

LinuxQMISDK

SLQS04.00.03

Generated by Doxygen 1.8.11

Contents

1	Welcome to the Sierra Wireless Linux QMI SDK API Reference Guide	1
1.1	Important Notice	1
1.2	Limitation of Liability	1
1.3	Patents	2
1.4	Copyright	2
1.5	Trademarks	2
1.6	Contact Information	2
2	Module Index	3
2.1	Modules	3
3	Namespace Index	5
3.1	Namespace List	5
4	Data Structure Index	7
4.1	Data Structures	7
5	File Index	29
5.1	File List	29

6	Module Documentation	33
6.1	Device Connectivity Service (DCS)	33
6.1.1	Detailed Description	33
6.2	Wireless Data Service (WDS)	34
6.2.1	Detailed Description	34
6.3	Device Management Service (DMS)	35
6.3.1	Detailed Description	35
6.4	Network Access Service (NAS)	36
6.4.1	Detailed Description	36
6.5	CallBack registration (CBK)	37
6.5.1	Detailed Description	37
6.6	Short Message Service (SMS)	38
6.6.1	Detailed Description	38
6.7	Position Determination Service (PDS)	39
6.7.1	Detailed Description	39
6.8	Card Application Toolkit (CAT)	40
6.8.1	Detailed Description	40
6.9	Remote Management Service (RMS)	41
6.9.1	Detailed Description	41
6.10	Firmware Management Service (FMS)	42
6.10.1	Detailed Description	42
6.11	Open Mobile Alliance Service (OMA)	43
6.11.1	Detailed Description	43
6.12	Specific Absorption Rate (SAR)	44
6.12.1	Detailed Description	44
6.13	SWI Open Mobile Alliance Service (SWIOMA)	45
6.13.1	Detailed Description	45
6.14	Voice Service (VOICE)	46
6.14.1	Detailed Description	46
6.15	Non-service specific APIs (SWI)	47

6.15.1 Detailed Description	47
6.16 User Identity Module Service (UIM)	48
6.16.1 Detailed Description	48
6.17 Audio Service (AUDIO)	49
6.17.1 Detailed Description	49
6.18 Quality of Service (QOS)	50
6.18.1 Detailed Description	50
6.19 IMS Service (IMS)	51
6.19.1 Detailed Description	51
6.20 SWI Audio Service(SWIAUDIO)	52
6.20.1 Detailed Description	52
6.21 Location Service(LOC)	53
6.21.1 Detailed Description	53
6.22 Thermal Mitigation Device(TMD)	54
6.22.1 Detailed Description	54
7 Namespace Documentation	55
7.1 Tables Namespace Reference	55
7.1.1 Detailed Description	55
8 Data Structure Documentation	57
8.1 _getIndicationRegResp Struct Reference	57
8.1.1 Detailed Description	57
8.1.2 Field Documentation	58
8.1.2.1 pRegCallStatInfoEvt	58
8.1.2.2 pRegTransLayerInfoEvt	58
8.1.2.3 pRegTransNWRegInfoEvt	58
8.2 _GetProfileSettingIn Struct Reference	58
8.2.1 Detailed Description	58
8.2.2 Field Documentation	58
8.2.2.1 ProfileID	58

8.2.2.2	ProfileType	58
8.3	_GetProfileSettingOut Struct Reference	58
8.3.1	Detailed Description	59
8.3.2	Field Documentation	59
8.3.2.1	curProfile	59
8.3.2.2	pExtErrCode	59
8.4	_getResetInfoNotification Struct Reference	59
8.4.1	Detailed Description	59
8.4.2	Field Documentation	60
8.4.2.1	source	60
8.4.2.2	type	60
8.5	_getTransLayerInfoResp Struct Reference	60
8.5.1	Detailed Description	60
8.5.2	Field Documentation	61
8.5.2.1	pRegInd	61
8.5.2.2	pTransLayerInfo	61
8.6	_getTransNWRRegInfoResp Struct Reference	61
8.6.1	Detailed Description	61
8.6.2	Field Documentation	61
8.6.2.1	pRegStatus	61
8.7	_MitigationDevInfo Struct Reference	61
8.7.1	Detailed Description	61
8.7.2	Field Documentation	62
8.7.2.1	deviceId	62
8.7.2.2	deviceIdLen	62
8.8	_modemTempNotification Struct Reference	62
8.8.1	Detailed Description	62
8.8.2	Field Documentation	62
8.8.2.1	ModemTemperature	63
8.8.2.2	ModemTempState	63

8.9	_packetSrvStatus Struct Reference	63
8.9.1	Detailed Description	63
8.9.2	Field Documentation	64
8.9.2.1	bearerID	64
8.9.2.2	connStatus	64
8.9.2.3	ipFamily	64
8.9.2.4	pQmiInterfaceInfo	64
8.9.2.5	reconfigReqd	64
8.9.2.6	sessionEndReason	64
8.9.2.7	techName	64
8.9.2.8	verboseSessnEndReason	64
8.9.2.9	verboseSessnEndReasonType	64
8.10	_qaQmi3GPP2BroadcastCfgInfo Struct Reference	64
8.10.1	Detailed Description	65
8.10.2	Field Documentation	65
8.10.2.1	activated_ind	65
8.10.2.2	CDMABroadcastConfig	65
8.10.2.3	num_instances	65
8.11	_qaQmi3GPPBroadcastCfgInfo Struct Reference	65
8.11.1	Detailed Description	65
8.11.2	Field Documentation	66
8.11.2.1	activated_ind	66
8.11.2.2	broadcastConfig	66
8.11.2.3	num_instances	66
8.12	_setIndicationRegReq Struct Reference	66
8.12.1	Detailed Description	66
8.12.2	Field Documentation	67
8.12.2.1	pRegCallStatInfoEvt	67
8.12.2.2	pRegTransLayerInfoEvt	67
8.12.2.3	pRegTransNWRegInfoEvt	67

8.13	_slqs3GPPConfigItem Struct Reference	67
8.13.1	Detailed Description	67
8.13.2	Field Documentation	69
8.13.2.1	LTEAttachProfileListLen	69
8.13.2.2	p3gppRelease	69
8.13.2.3	pDefaultPDNEnabled	69
8.13.2.4	pLTEAttachProfile	69
8.13.2.5	pLTEAttachProfileList	69
8.13.2.6	pProfileList	69
8.14	_SlqsNas3GppNetworkRAT_ Struct Reference	69
8.14.1	Detailed Description	69
8.14.2	Field Documentation	70
8.14.2.1	MCC	70
8.14.2.2	MNC	70
8.14.2.3	RAT	70
8.15	_slqsNetworkScanInfo Struct Reference	70
8.15.1	Detailed Description	70
8.15.2	Field Documentation	71
8.15.2.1	pNetworkInfo	71
8.15.2.2	pNetworkInfoInstances	71
8.15.2.3	pPCSDigitInfo	71
8.15.2.4	pPCSDigitInstances	71
8.15.2.5	pRATInfo	71
8.15.2.6	pRATInstances	71
8.15.2.7	pScanResult	71
8.16	_SLQSOMADMSessionInfo Struct Reference	71
8.16.1	Detailed Description	71
8.16.2	Field Documentation	73
8.16.2.1	pDate	73
8.16.2.2	pDateLength	73

8.16.2.3	pPkgDescLength	73
8.16.2.4	pPkgDescription	73
8.16.2.5	pPkgName	73
8.16.2.6	pPkgNameLength	74
8.16.2.7	pRetryCount	74
8.16.2.8	pSessionState	74
8.16.2.9	pSessionType	74
8.16.2.10	pSeverity	74
8.16.2.11	pSource	74
8.16.2.12	pSourceLength	74
8.16.2.13	pStatus	74
8.16.2.14	pTime	74
8.16.2.15	pTimeLength	74
8.16.2.16	pUpdateCompleteStatus	74
8.17	_SLQSOMADMSettings Struct Reference	74
8.17.1	Detailed Description	74
8.17.2	Field Documentation	76
8.17.2.1	pAutosdm	76
8.17.2.2	pFOTAdownload	76
8.17.2.3	pFOTAUpdate	76
8.17.2.4	pFwAutoCheck	76
8.17.2.5	pOMADMEEnabled	76
8.18	_SLQSOMADMSettingsReqParams Struct Reference	76
8.18.1	Detailed Description	76
8.18.2	Field Documentation	77
8.18.2.1	FOTAdownload	77
8.18.2.2	FOTAUpdate	77
8.18.2.3	pAutosdm	77
8.19	_SLQSOMADMSettingsReqParams3 Struct Reference	77
8.19.1	Detailed Description	77

8.19.2	Field Documentation	78
8.19.2.1	FOTAdownload	78
8.19.2.2	FOTAUpdate	78
8.19.2.3	pAutosdm	78
8.19.2.4	pFwAutoCheck	78
8.20	_SLQSSwiGetHostDevInfoParams Struct Reference	78
8.20.1	Detailed Description	78
8.20.2	Field Documentation	79
8.20.2.1	bManSize	79
8.20.2.2	bModelSize	79
8.20.2.3	bPlasmaIDSize	79
8.20.2.4	bSWVerSize	79
8.20.2.5	pManString	79
8.20.2.6	pModelString	79
8.20.2.7	pPlasmaIDString	79
8.20.2.8	pSWVerString	79
8.21	_SLQSSwiGetOSInfoParams Struct Reference	79
8.21.1	Detailed Description	79
8.21.2	Field Documentation	80
8.21.2.1	bNameSize	80
8.21.2.2	bVersionSize	80
8.21.2.3	pNameString	80
8.21.2.4	pVersionString	80
8.22	_SLQSSwiGetSerialNoExtParams Struct Reference	80
8.22.1	Detailed Description	80
8.22.2	Field Documentation	81
8.22.2.1	meidLength	81
8.22.2.2	pMeidString	81
8.23	_SLQSSwiSetHostDevInfoParams Struct Reference	81
8.23.1	Detailed Description	81

8.23.2	Field Documentation	82
8.23.2.1	bManSize	82
8.23.2.2	bModelSize	82
8.23.2.3	bPlasmaIDSize	82
8.23.2.4	bSWVerSize	82
8.23.2.5	pManString	82
8.23.2.6	pModelString	82
8.23.2.7	pPlasmaIDString	82
8.23.2.8	pSWVerString	82
8.24	_SLQSSwiSetOSInfoParams Struct Reference	82
8.24.1	Detailed Description	82
8.24.2	Field Documentation	83
8.24.2.1	bNameSize	83
8.24.2.2	bVersionSize	83
8.24.2.3	pNameString	83
8.24.2.4	pVersionString	83
8.25	_sysSelectPrefInfo Struct Reference	83
8.25.1	Detailed Description	83
8.25.2	Field Documentation	87
8.25.2.1	pBandPref	87
8.25.2.2	pEmerMode	87
8.25.2.3	pGWAcqOrderPref	87
8.25.2.4	pLTEBandPref	87
8.25.2.5	pModePref	87
8.25.2.6	pNetSelPref	87
8.25.2.7	pPRLPref	87
8.25.2.8	pRoamPref	88
8.25.2.9	pSrvDomainPref	88
8.26	_sysSelectPrefParams Struct Reference	88
8.26.1	Detailed Description	88

8.26.2	Field Documentation	93
8.26.2.1	pAcqOrderPref	93
8.26.2.2	pBandPref	93
8.26.2.3	pChgDuration	93
8.26.2.4	pCSGID	93
8.26.2.5	pEmerMode	93
8.26.2.6	pGWAcqOrderPref	93
8.26.2.7	pLTEBandPref	93
8.26.2.8	pMNCIncPCSDigStat	93
8.26.2.9	pModePref	93
8.26.2.10	pNetSelPref	93
8.26.2.11	pPRLPref	93
8.26.2.12	pRAT	94
8.26.2.13	pRoamPref	94
8.26.2.14	pSrvDomainPref	94
8.26.2.15	pSrvRegRestriction	94
8.26.2.16	pTdsdmaBandPref	94
8.27	_transLayerinfo Struct Reference	94
8.27.1	Detailed Description	94
8.27.2	Field Documentation	94
8.27.2.1	TransCap	94
8.27.2.2	TransType	94
8.28	_transLayerInfoNotification Struct Reference	94
8.28.1	Detailed Description	95
8.28.2	Field Documentation	95
8.28.2.1	pTransLayerInfo	95
8.28.2.2	regInd	95
8.29	_transNWRegInfoNotification Struct Reference	95
8.29.1	Detailed Description	95
8.29.2	Field Documentation	96

8.29.2.1	NWRegStat	96
8.30	accelAcceptReady_s Struct Reference	96
8.30.1	Detailed Description	96
8.30.2	Field Documentation	96
8.30.2.1	batchPerSec	96
8.30.2.2	injectEnable	96
8.30.2.3	samplesPerBatch	96
8.31	accelTempAcceptReady_s Struct Reference	97
8.31.1	Detailed Description	97
8.31.2	Field Documentation	97
8.31.2.1	batchPerSec	97
8.31.2.2	injectEnable	97
8.31.2.3	samplesPerBatch	97
8.32	acqOrderPref Struct Reference	97
8.32.1	Detailed Description	98
8.32.2	Field Documentation	98
8.32.2.1	acqOrdeLen	98
8.32.2.2	pAcqOrder	98
8.33	ActPilotPNElement Struct Reference	98
8.33.1	Detailed Description	98
8.33.2	Field Documentation	98
8.33.2.1	ActSetPilotPN	99
8.33.2.2	ActSetPilotPNStrength	99
8.34	AddCDMASysInfo Struct Reference	99
8.34.1	Detailed Description	99
8.34.2	Field Documentation	99
8.34.2.1	geoSysIdx	99
8.34.2.2	regPrd	99
8.35	AddSysInfo Struct Reference	99
8.35.1	Detailed Description	99

8.35.2	Field Documentation	100
8.35.2.1	cellBroadcastCap	100
8.35.2.2	geoSysIdx	100
8.36	airTimer Struct Reference	100
8.36.1	Detailed Description	100
8.36.2	Field Documentation	100
8.36.2.1	airTimerValue	101
8.36.2.2	namID	101
8.37	allCallsAlphaIDInfo Struct Reference	101
8.37.1	Detailed Description	101
8.37.2	Field Documentation	101
8.37.2.1	AlphaIDInfo	101
8.37.2.2	callID	101
8.38	allCallsDiagInfo Struct Reference	101
8.38.1	Detailed Description	101
8.38.2	Field Documentation	102
8.38.2.1	callID	102
8.38.2.2	DiagInfo	102
8.39	allCallsUUSInfo Struct Reference	102
8.39.1	Detailed Description	102
8.39.2	Field Documentation	102
8.39.2.1	callID	102
8.39.2.2	uusInfo	102
8.40	alphaIDInfo Struct Reference	102
8.40.1	Detailed Description	102
8.40.2	Field Documentation	103
8.40.2.1	alphaDcs	103
8.40.2.2	alphaLen	103
8.40.2.3	alphaText	103
8.41	altitudeSrcInfo Struct Reference	103

8.41.1 Detailed Description	103
8.41.2 Field Documentation	104
8.41.2.1 coverage	104
8.41.2.2 linkage	104
8.41.2.3 source	104
8.42 altSrcInfo_t Struct Reference	104
8.42.1 Detailed Description	104
8.42.2 Field Documentation	105
8.42.2.1 coverage	105
8.42.2.2 linkage	105
8.42.2.3 source	105
8.43 appStats Struct Reference	105
8.43.1 Detailed Description	105
8.43.2 Field Documentation	107
8.43.2.1 aidLength	107
8.43.2.2 aidVal	107
8.43.2.3 appState	108
8.43.2.4 appType	108
8.43.2.5 persoFeature	108
8.43.2.6 persoRetries	108
8.43.2.7 persoState	108
8.43.2.8 persoUnblockRetries	108
8.43.2.9 pin1Retries	108
8.43.2.10 pin1State	108
8.43.2.11 pin2Retries	108
8.43.2.12 pin2State	108
8.43.2.13 puk1Retries	108
8.43.2.14 puk2Retries	108
8.43.2.15 univPin	108
8.44 appStatus Struct Reference	108

8.44.1 Detailed Description	108
8.44.2 Field Documentation	110
8.44.2.1 aidLength	110
8.44.2.2 aidVal	110
8.44.2.3 appState	111
8.44.2.4 appType	111
8.44.2.5 persoFeature	111
8.44.2.6 persoRetries	111
8.44.2.7 persoState	111
8.44.2.8 persoUnblockRetries	111
8.44.2.9 pin1Retries	111
8.44.2.10 pin1State	111
8.44.2.11 pin2Retries	111
8.44.2.12 pin2State	111
8.44.2.13 puk1Retries	111
8.44.2.14 puk2Retries	111
8.44.2.15 univPin	111
8.45 arrAlertingPattern Struct Reference	111
8.45.1 Detailed Description	112
8.45.2 Field Documentation	112
8.45.2.1 alertingPattern	112
8.45.2.2 callID	112
8.45.2.3 numInstances	112
8.46 arrAlertingType Struct Reference	112
8.46.1 Detailed Description	112
8.46.2 Field Documentation	113
8.46.2.1 AlertingType	113
8.46.2.2 callID	113
8.46.2.3 numInstances	113
8.47 arrAlphaID Struct Reference	113

8.47.1 Detailed Description	113
8.47.2 Field Documentation	113
8.47.2.1 allCallsAlphaIDInfoArr	113
8.47.2.2 numInstances	114
8.48 arrCalledPartyNum Struct Reference	114
8.48.1 Detailed Description	114
8.48.2 Field Documentation	114
8.48.2.1 CalledPartyNum	114
8.48.2.2 numInstances	114
8.49 arrCallEndReason Struct Reference	114
8.49.1 Detailed Description	114
8.49.2 Field Documentation	115
8.49.2.1 callEndReason	115
8.49.2.2 callID	115
8.49.2.3 numInstances	115
8.50 arrCallInfo Struct Reference	115
8.50.1 Detailed Description	115
8.50.2 Field Documentation	115
8.50.2.1 getAllCallInfo	115
8.50.2.2 numInstances	116
8.51 arrConnectPartyNum Struct Reference	116
8.51.1 Detailed Description	116
8.51.2 Field Documentation	116
8.51.2.1 ConnectedPartyNum	116
8.51.2.2 numInstances	116
8.52 arrDiagInfo Struct Reference	116
8.52.1 Detailed Description	116
8.52.2 Field Documentation	117
8.52.2.1 DiagInfo	117
8.52.2.2 numInstances	117

8.53	arrRedirPartyNum Struct Reference	117
8.53.1	Detailed Description	117
8.53.2	Field Documentation	117
8.53.2.1	numInstances	117
8.53.2.2	RedirPartyNum	117
8.54	arrRemotePartyName Struct Reference	117
8.54.1	Detailed Description	117
8.54.2	Field Documentation	118
8.54.2.1	GetAllCallRmtPtyName	118
8.54.2.2	numInstances	118
8.55	arrRemotePartyNum Struct Reference	118
8.55.1	Detailed Description	118
8.55.2	Field Documentation	118
8.55.2.1	numInstances	118
8.55.2.2	RmtPtyNum	118
8.56	arrSvcOption Struct Reference	118
8.56.1	Detailed Description	119
8.56.2	Field Documentation	119
8.56.2.1	callID	119
8.56.2.2	numInstances	119
8.56.2.3	srvOption	119
8.57	arrUUSInfo Struct Reference	119
8.57.1	Detailed Description	119
8.57.2	Field Documentation	120
8.57.2.1	AllCallsUUSInfo	120
8.57.2.2	numInstances	120
8.58	authenticateResult Struct Reference	120
8.58.1	Detailed Description	120
8.58.2	Field Documentation	120
8.58.2.1	content	120

8.58.2.2	contentLen	120
8.59	authenticationData Struct Reference	120
8.59.1	Detailed Description	120
8.59.2	Field Documentation	121
8.59.2.1	context	121
8.59.2.2	data	121
8.59.2.3	dataLen	121
8.60	BandCapabilityResp Struct Reference	121
8.60.1	Detailed Description	121
8.60.2	Field Documentation	122
8.60.2.1	bandCapability	122
8.60.2.2	pLteBandCapability	123
8.60.2.3	pTdsBandCapability	123
8.61	BdsSV Struct Reference	123
8.61.1	Detailed Description	123
8.61.2	Field Documentation	123
8.61.2.1	id	123
8.61.2.2	mask	123
8.62	BdsSVInfo Struct Reference	123
8.62.1	Detailed Description	123
8.62.2	Field Documentation	124
8.62.2.1	len	124
8.62.2.2	pSV	124
8.63	BroadcastConfig Struct Reference	124
8.63.1	Detailed Description	124
8.63.2	Field Documentation	124
8.63.2.1	fromServiceId	124
8.63.2.2	selected	124
8.63.2.3	toServiceId	125
8.64	burstDTMFInfo Struct Reference	125

8.64.1 Detailed Description	125
8.64.2 Field Documentation	125
8.64.2.1 digitCnt	125
8.64.2.2 pCallID	125
8.64.2.3 pDigitBuff	125
8.65 CallBarringSysInfo Struct Reference	125
8.65.1 Detailed Description	126
8.65.2 Field Documentation	126
8.65.2.1 csBarStatus	126
8.65.2.2 psBarStatus	126
8.66 callBarStatus Struct Reference	126
8.66.1 Detailed Description	126
8.66.2 Field Documentation	127
8.66.2.1 csBarStatus	127
8.66.2.2 psBarStatus	127
8.67 calledPartyInfo Struct Reference	127
8.67.1 Detailed Description	127
8.67.2 Field Documentation	128
8.67.2.1 number	129
8.67.2.2 numLen	129
8.67.2.3 numPlan	129
8.67.2.4 numType	129
8.67.2.5 PI	129
8.67.2.6 SI	129
8.68 calledPartySubAdd Struct Reference	129
8.68.1 Detailed Description	129
8.68.2 Field Documentation	130
8.68.2.1 extBit	130
8.68.2.2 oddEvenInd	130
8.68.2.3 subAddr	130

8.68.2.4	subAddrLen	130
8.68.2.5	subAddrType	130
8.69	callerIDInfo Struct Reference	130
8.69.1	Detailed Description	130
8.69.2	Field Documentation	130
8.69.2.1	callerID	130
8.69.2.2	callerIDLen	130
8.69.2.3	PI	130
8.70	callFwdTypeAndPlan Struct Reference	130
8.70.1	Detailed Description	131
8.70.2	Field Documentation	131
8.70.2.1	numberPlan	131
8.70.2.2	numberType	131
8.71	callFWExtInfo Struct Reference	131
8.71.1	Detailed Description	132
8.71.2	Field Documentation	133
8.71.2.1	noReplyTimer	133
8.71.2.2	number	133
8.71.2.3	numLen	133
8.71.2.4	numPlan	133
8.71.2.5	numType	133
8.71.2.6	PI	133
8.71.2.7	SI	133
8.71.2.8	SvcClass	133
8.71.2.9	SvcStatus	133
8.72	callFWInfo Struct Reference	133
8.72.1	Detailed Description	134
8.72.2	Field Documentation	134
8.72.2.1	noReplyTimer	134
8.72.2.2	number	134

8.72.2.3 numLen	134
8.72.2.4 SvcClass	134
8.72.2.5 SvcStatus	134
8.73 callInfo Struct Reference	134
8.73.1 Detailed Description	135
8.73.2 Field Documentation	136
8.73.2.1 callID	136
8.73.2.2 callState	136
8.73.2.3 callType	136
8.73.2.4 direction	136
8.73.2.5 mode	136
8.74 callingPartyInfo Struct Reference	136
8.74.1 Detailed Description	136
8.74.2 Field Documentation	137
8.74.2.1 number	138
8.74.2.2 numLen	138
8.74.2.3 numPlan	138
8.74.2.4 numType	138
8.74.2.5 PI	138
8.74.2.6 SI	138
8.75 cardResult Struct Reference	138
8.75.1 Detailed Description	138
8.75.2 Field Documentation	138
8.75.2.1 sw1	138
8.75.2.2 sw2	138
8.76 cardStatus Struct Reference	138
8.76.1 Detailed Description	139
8.76.2 Field Documentation	139
8.76.2.1 index1xPri	139
8.76.2.2 index1xSec	139

8.76.2.3	indexGwPri	139
8.76.2.4	indexGwSec	139
8.76.2.5	numSlot	139
8.76.2.6	SlotInfo	140
8.77	CarrierImage_t Struct Reference	140
8.77.1	Detailed Description	140
8.77.2	Field Documentation	140
8.77.2.1	m_FwBuildId	140
8.77.2.2	m_FwImageId	141
8.77.2.3	m_nCarrierId	141
8.77.2.4	m_nFolderId	141
8.77.2.5	m_nStorage	141
8.77.2.6	m_PriBuildId	141
8.77.2.7	m_PriImageId	141
8.78	CatAlphaIdentifierTlv Struct Reference	141
8.78.1	Detailed Description	141
8.78.2	Field Documentation	141
8.78.2.1	AlphaID	141
8.78.2.2	AlphaIDLength	141
8.78.2.3	ReferenceID	141
8.79	CatCommonEventTlv Struct Reference	141
8.79.1	Field Documentation	142
8.79.1.1	CatEvent	142
8.79.1.2	EventID	142
8.79.1.3	EventLength	142
8.79.1.4	TlvPresent	142
8.80	CatEndProactiveSessionTlv Struct Reference	142
8.80.1	Detailed Description	142
8.80.2	Field Documentation	142
8.80.2.1	EndProactiveSession	142

8.81 CATEventDataType Struct Reference	142
8.81.1 Field Documentation	142
8.81.1.1 eventMask	142
8.81.1.2 pErrorMask	142
8.82 CatEventIDDataTlv Struct Reference	143
8.82.1 Detailed Description	143
8.82.2 Field Documentation	143
8.82.2.1 Data	143
8.82.2.2 DataLength	143
8.82.2.3 ReferenceID	143
8.83 CatEventListTlv Struct Reference	143
8.83.1 Detailed Description	143
8.83.2 Field Documentation	144
8.83.2.1 SetupEventList	144
8.84 CatRefreshTlv Struct Reference	144
8.84.1 Detailed Description	144
8.84.2 Field Documentation	144
8.84.2.1 RefreshMode	144
8.84.2.2 RefreshStage	144
8.85 ccSUPSType Struct Reference	144
8.85.1 Detailed Description	144
8.85.2 Field Documentation	145
8.85.2.1 reason	145
8.85.2.2 svcType	145
8.86 CDMABroadcastConfig Struct Reference	145
8.86.1 Detailed Description	145
8.86.2 Field Documentation	146
8.86.2.1 language	146
8.86.2.2 selected	146
8.86.2.3 serviceCategory	146

8.87	CDMAChannel Struct Reference	146
8.87.1	Detailed Description	146
8.87.2	Field Documentation	146
8.87.2.1	priChA	146
8.87.2.2	priChB	146
8.87.2.3	secChA	146
8.87.2.4	secChB	147
8.88	CDMAECIOThresh Struct Reference	147
8.88.1	Detailed Description	147
8.88.2	Field Documentation	147
8.88.2.1	CDMAECIOThreshListLen	147
8.88.2.2	pCDMAECIOThreshList	147
8.89	CDMAInfo Struct Reference	147
8.89.1	Detailed Description	147
8.89.2	Field Documentation	148
8.89.2.1	baseId	148
8.89.2.2	baseLat	148
8.89.2.3	baseLong	148
8.89.2.4	nid	148
8.89.2.5	refpn	148
8.89.2.6	sid	148
8.90	cdmaMsgDecodingParams Struct Reference	148
8.90.1	Detailed Description	149
8.90.2	Field Documentation	150
8.90.2.1	absoluteValidity	150
8.90.2.2	mcTimeStamp	151
8.90.2.3	messageLength	151
8.90.2.4	pAlertPriority	151
8.90.2.5	pCallbkAddr	151
8.90.2.6	pCallbkAddrLength	151

8.90.2.7	pDisplayMode	151
8.90.2.8	pLanguage	151
8.90.2.9	pMessage	151
8.90.2.10	pMessageID	151
8.90.2.11	pPriority	151
8.90.2.12	pPrivacy	151
8.90.2.13	pReadAcknowledgementReq	151
8.90.2.14	pRelativeValidity	151
8.90.2.15	pSenderAddr	151
8.90.2.16	pSenderAddrLength	151
8.90.2.17	pTextMsg	151
8.90.2.18	pTextMsgLength	151
8.90.2.19	pUserAcknowledgementReq	151
8.91	cdmaMsgEncodingParams Struct Reference	151
8.91.1	Detailed Description	151
8.91.2	Field Documentation	152
8.91.2.1	messageld	152
8.91.2.2	pCallbackAddr	153
8.91.2.3	pDestAddr	153
8.91.2.4	pEncodingAlphabet	153
8.91.2.5	pMessage	153
8.91.2.6	pMessageSize	153
8.91.2.7	pPriority	153
8.91.2.8	pRelValidity	153
8.91.2.9	pTextMsg	153
8.91.2.10	textMsgLength	153
8.92	CDMARSSIThresh Struct Reference	153
8.92.1	Detailed Description	153
8.92.2	Field Documentation	153
8.92.2.1	CDMARSSIThreshListLen	153

8.92.2.2	pCDMARSSIThreshList	153
8.93	CDMASSInfo Struct Reference	154
8.93.1	Detailed Description	154
8.93.2	Field Documentation	154
8.93.2.1	ecio	154
8.93.2.2	rsi	154
8.94	cdmaSSInfo Struct Reference	154
8.94.1	Detailed Description	154
8.94.2	Field Documentation	154
8.94.2.1	ecio	154
8.94.2.2	rsi	155
8.95	CDMASysInfo Struct Reference	155
8.95.1	Detailed Description	155
8.95.2	Field Documentation	157
8.95.2.1	baseId	157
8.95.2.2	baseLat	158
8.95.2.3	baseLong	158
8.95.2.4	bsInfoValid	158
8.95.2.5	bsPRev	158
8.95.2.6	bsPRevValid	158
8.95.2.7	ccsSupported	158
8.95.2.8	ccsSupportedValid	158
8.95.2.9	cdmaSysIdValid	158
8.95.2.10	isSysPrIMatch	158
8.95.2.11	isSysPrIMatchValid	158
8.95.2.12	MCC	158
8.95.2.13	MNC	158
8.95.2.14	networkID	158
8.95.2.15	networkIdValid	158
8.95.2.16	packetZone	158

8.95.2.17 packetZoneValid	158
8.95.2.18 pRevInUse	158
8.95.2.19 pRevInUseValid	158
8.95.2.20 sysInfoCDMA	158
8.95.2.21 systemID	158
8.96 CDMA SysInfoExt Struct Reference	158
8.96.1 Detailed Description	158
8.96.2 Field Documentation	159
8.96.2.1 imsi_11_12	159
8.96.2.2 MCC	159
8.97 CellIDb Struct Reference	159
8.97.1 Detailed Description	159
8.97.2 Field Documentation	159
8.97.2.1 mask	159
8.98 cellParams Struct Reference	159
8.98.1 Detailed Description	160
8.98.2 Field Documentation	160
8.98.2.1 pci	160
8.98.2.2 rsrp	160
8.98.2.3 rsrq	160
8.98.2.4 rssi	160
8.98.2.5 srxlev	160
8.99 changeUIMPIN Struct Reference	161
8.99.1 Detailed Description	161
8.99.2 Field Documentation	161
8.99.2.1 oldPINLen	161
8.99.2.2 oldPINVal	161
8.99.2.3 pinID	161
8.99.2.4 pinLen	161
8.99.2.5 pinVal	162

8.100ChannelRate Struct Reference	162
8.100.1 Detailed Description	162
8.100.2 Field Documentation	162
8.100.2.1 CurrChanRxRate	162
8.100.2.2 CurrChanTxRate	162
8.100.2.3 MaxChanRxRate	162
8.100.2.4 MaxChanTxRate	162
8.101channelRate Struct Reference	162
8.101.1 Detailed Description	163
8.101.2 Field Documentation	163
8.101.2.1 CurrChanRxRate	163
8.101.2.2 CurrChanTxRate	163
8.102CLIPResp Struct Reference	163
8.102.1 Detailed Description	163
8.102.2 Field Documentation	164
8.102.2.1 ActiveStatus	164
8.102.2.2 ProvisionStatus	164
8.103CLIRResp Struct Reference	164
8.103.1 Detailed Description	164
8.103.2 Field Documentation	164
8.103.2.1 ActiveStatus	164
8.103.2.2 ProvisionStatus	164
8.104ClkInfo Struct Reference	164
8.104.1 Detailed Description	165
8.104.2 Field Documentation	165
8.104.2.1 mask	165
8.105CNAPResp Struct Reference	165
8.105.1 Detailed Description	166
8.105.2 Field Documentation	166
8.105.2.1 ActiveStatus	166

8.105.2.2 ProvisionStatus	166
8.106COLPResp Struct Reference	166
8.106.1 Detailed Description	166
8.106.2 Field Documentation	167
8.106.2.1 ActiveStatus	167
8.106.2.2 ProvisionStatus	167
8.107COLRResp Struct Reference	167
8.107.1 Detailed Description	167
8.107.2 Field Documentation	168
8.107.2.1 ActiveStatus	168
8.107.2.2 ProvisionStatus	168
8.108CommInfo Struct Reference	168
8.108.1 Detailed Description	168
8.108.2 Field Documentation	169
8.108.2.1 imsRegState	169
8.108.2.2 modemMode	169
8.108.2.3 psState	169
8.108.2.4 systemMode	169
8.108.2.5 temperature	169
8.109ConnectionStatus Struct Reference	169
8.109.1 Detailed Description	170
8.109.2 Field Documentation	170
8.109.2.1 MDMCallDuration	170
8.109.2.2 MDMConnStatus	170
8.110connectionStatus Struct Reference	170
8.110.1 Detailed Description	170
8.110.2 Field Documentation	170
8.110.2.1 MDMCallDuration	170
8.110.2.2 MDMConnStatus	170
8.111connectNumInfo Struct Reference	171

8.111.1 Detailed Description	171
8.111.2 Field Documentation	172
8.111.2.1 callerID	172
8.111.2.2 callerIDLen	172
8.111.2.3 numPlan	172
8.111.2.4 numPresInd	172
8.111.2.5 numType	172
8.111.2.6 screeningInd	172
8.112CrashInfo Struct Reference	172
8.112.1 Detailed Description	172
8.112.2 Field Documentation	173
8.112.2.1 crashData	173
8.112.2.2 crashId	173
8.112.2.3 crashStrLen	173
8.112.2.4 gcDumpStrLen	173
8.112.2.5 numCrashes	173
8.112.2.6 pCrashString	173
8.112.2.7 pGCDumpString	173
8.113crashInfoParams Struct Reference	173
8.113.1 Detailed Description	173
8.113.2 Field Documentation	174
8.113.2.1 crashInfo	174
8.113.2.2 crashStatus	174
8.114CrashInfoParams Struct Reference	174
8.114.1 Detailed Description	174
8.114.2 Field Documentation	174
8.114.2.1 pCrashInfo	174
8.114.2.2 pDevCrashStatus	174
8.115crashInformation Struct Reference	174
8.115.1 Detailed Description	175

8.115.2 Field Documentation	175
8.115.2.1 crashData	175
8.115.2.2 crashId	175
8.115.2.3 crashString	175
8.115.2.4 crashStrlen	175
8.115.2.5 gcdumpString	175
8.115.2.6 gcdumpStrlen	176
8.115.2.7 numCrashes	176
8.116 CreateProfileIn Struct Reference	176
8.116.1 Detailed Description	176
8.116.2 Field Documentation	176
8.116.2.1 curProfile	176
8.116.2.2 pProfileID	176
8.116.2.3 pProfileType	176
8.117 CreateProfileOut Struct Reference	176
8.117.1 Detailed Description	177
8.117.2 Field Documentation	177
8.117.2.1 pExtErrorCode	177
8.117.2.2 pProfileIndex	177
8.117.2.3 pProfileType	177
8.118 CSGID Struct Reference	177
8.118.1 Detailed Description	177
8.118.2 Field Documentation	178
8.118.2.1 id	178
8.118.2.2 mcc	178
8.118.2.3 mnc	178
8.118.2.4 mncPcsDigits	178
8.118.2.5 rat	178
8.119 CUGInfo Struct Reference	178
8.119.1 Detailed Description	178

8.119.2 Field Documentation	179
8.119.2.1 CUGIndex	179
8.119.2.2 SuppOA	179
8.119.2.3 SuppPrefCUG	179
8.120 curAMRConfig Struct Reference	179
8.120.1 Detailed Description	179
8.120.2 Field Documentation	180
8.120.2.1 gsmAmrStat	180
8.120.2.2 wcdmaAmrStat	180
8.121 CurrDataSysStat Struct Reference	180
8.121.1 Detailed Description	180
8.121.2 Field Documentation	180
8.121.2.1 pCurrNetworkInfo	180
8.121.2.2 pNetworkInfoLen	180
8.121.2.3 pPrefNetwork	180
8.122 currentCatEvent Union Reference	180
8.122.1 Detailed Description	181
8.122.2 Field Documentation	181
8.122.2.1 CatAlphaldtfr	181
8.122.2.2 CatEndPS	181
8.122.2.3 CatEventLst	181
8.122.2.4 CatEvIDData	181
8.122.2.5 CatRefresh	181
8.123 CurrentImgLst Struct Reference	181
8.123.1 Detailed Description	182
8.123.2 Field Documentation	182
8.123.2.1 carrier	182
8.123.2.2 fwvers	182
8.123.2.3 numEntries	182
8.123.2.4 pCurrImglInfo	182

8.123.2.5 pkgver	182
8.123.2.6 priver	182
8.124currentPLMN Struct Reference	182
8.124.1 Detailed Description	183
8.124.2 Field Documentation	183
8.124.2.1 MCC	183
8.124.2.2 MNC	183
8.124.2.3 netDescr	183
8.124.2.4 netDescrLength	183
8.125CurrImageInfo Struct Reference	183
8.125.1 Detailed Description	183
8.125.2 Field Documentation	184
8.125.2.1 buildID	184
8.125.2.2 buildIDLen	184
8.125.2.3 imageType	184
8.125.2.4 uniqueID	184
8.126CurrNetworkInfo Struct Reference	184
8.126.1 Detailed Description	184
8.126.2 Field Documentation	186
8.126.2.1 NetworkType	186
8.126.2.2 RATMask	186
8.126.2.3 SOMask	186
8.127currNetworkInfo Struct Reference	186
8.127.1 Detailed Description	186
8.127.2 Field Documentation	187
8.127.2.1 NetworkType	187
8.127.2.2 RATMask	187
8.127.2.3 SOMask	187
8.128custFeaturesInfo Struct Reference	187
8.128.1 Detailed Description	187

8.128.2 Field Documentation	189
8.128.2.1 GpsEnable	189
8.128.2.2 pDHCPRelayEnabled	189
8.128.2.3 pDisableIMSI	189
8.128.2.4 pGPSP_LPM	189
8.128.2.5 pGPSSel	189
8.128.2.6 pIPFamSupport	189
8.128.2.7 plsVoiceEnabled	189
8.128.2.8 pRMAutoConnect	189
8.128.2.9 pSMSSupport	189
8.129custFeaturesSetting Struct Reference	189
8.129.1 Detailed Description	189
8.129.2 Field Documentation	191
8.129.2.1 pDHCPRelayEnabled	191
8.129.2.2 pGPSEnable	191
8.129.2.3 pGPSP_LPM	191
8.129.2.4 pGPSSel	191
8.129.2.5 plsVoiceEnabled	191
8.130custSettingInfo Struct Reference	191
8.130.1 Detailed Description	191
8.130.2 Field Documentation	192
8.130.2.1 cust_attr	192
8.130.2.2 cust_id	192
8.130.2.3 cust_value	192
8.130.2.4 id_length	192
8.130.2.5 value_length	192
8.131custSettingList Struct Reference	192
8.131.1 Detailed Description	192
8.131.2 Field Documentation	192
8.131.2.1 custSetting	192

8.131.2.2 list_type	192
8.131.2.3 num_instances	192
8.132dataBearers Struct Reference	192
8.132.1 Detailed Description	193
8.132.2 Field Documentation	193
8.132.2.1 dataBearerMask	193
8.132.2.2 pCurDataBearerTechnology	193
8.132.2.3 pLastCallDataBearerTechnology	193
8.133DataBearerTech Struct Reference	193
8.133.1 Detailed Description	193
8.133.2 Field Documentation	195
8.133.2.1 ratValue	195
8.133.2.2 soMask	195
8.133.2.3 techType	195
8.134DataBearerTechExt Struct Reference	195
8.134.1 Detailed Description	196
8.134.2 Field Documentation	196
8.134.2.1 pBearerTech	196
8.134.2.2 pLastBearerTech	196
8.135dataBearerTechnology Struct Reference	196
8.135.1 Detailed Description	196
8.135.2 Field Documentation	197
8.135.2.1 currentNetwork	197
8.135.2.2 ratMask	197
8.135.2.3 soMask	197
8.136dataRate Struct Reference	197
8.136.1 Detailed Description	197
8.136.2 Field Documentation	198
8.136.2.1 dataRateMax	198
8.136.2.2 guaranteedRate	198

8.137dataSrvCapabilities Struct Reference	198
8.137.1 Detailed Description	198
8.137.2 Field Documentation	199
8.137.2.1 dataCapabilities	199
8.137.2.2 dataCapabilitiesLen	199
8.138DataStatusDetail Struct Reference	199
8.138.1 Detailed Description	199
8.138.2 Field Documentation	200
8.138.2.1 IPAddress	200
8.138.2.2 LastErrCode	200
8.139DataULongLongTlv Struct Reference	201
8.139.1 Field Documentation	201
8.139.1.1 TlvPresent	201
8.139.1.2 ullData	201
8.140DataULongTlv Struct Reference	201
8.140.1 Field Documentation	201
8.140.1.1 TlvPresent	201
8.140.1.2 ulData	201
8.141DcsUsbPortNames Struct Reference	201
8.141.1 Field Documentation	201
8.141.1.1 AtCmdPort	201
8.141.1.2 DmPort	201
8.141.1.3 NmeaPort	201
8.142delAssistDataStatus Struct Reference	201
8.142.1 Detailed Description	201
8.142.2 Field Documentation	202
8.142.2.1 status	202
8.143depersonalizationInformation Struct Reference	202
8.143.1 Detailed Description	202
8.143.2 Field Documentation	203

8.143.2.1 ckLen	203
8.143.2.2 ckVal	203
8.143.2.3 feature	203
8.143.2.4 operation	203
8.144detailSvcInfo Struct Reference	203
8.144.1 Detailed Description	204
8.144.2 Field Documentation	204
8.144.2.1 hdrHybrid	205
8.144.2.2 hdrSrvStatus	205
8.144.2.3 isSysForbidden	205
8.144.2.4 srvCapability	205
8.144.2.5 srvStatus	205
8.145DeviceConfigDetail Struct Reference	205
8.145.1 Detailed Description	205
8.145.2 Field Documentation	206
8.145.2.1 Chipset	206
8.145.2.2 HWVersion	206
8.145.2.3 QLIC	206
8.145.2.4 Technology	206
8.146DHCPOption Struct Reference	206
8.146.1 Detailed Description	206
8.146.2 Field Documentation	206
8.146.2.1 optCode	206
8.146.2.2 optValLen	206
8.146.2.3 pOptVal	206
8.147DHCPOptionList Struct Reference	206
8.147.1 Detailed Description	207
8.147.2 Field Documentation	207
8.147.2.1 numOpt	207
8.147.2.2 pOptions	207

8.148diagInfo Struct Reference	207
8.148.1 Detailed Description	207
8.148.2 Field Documentation	208
8.148.2.1 diagInfoLen	208
8.148.2.2 diagnosticInfo	208
8.149dirNum Struct Reference	208
8.149.1 Detailed Description	208
8.149.2 Field Documentation	208
8.149.2.1 dirNum	208
8.149.2.2 dirNumLen	208
8.150dms_ActivationStatusTlv Struct Reference	208
8.150.1 Detailed Description	208
8.150.2 Field Documentation	209
8.150.2.1 activationStatus	209
8.150.2.2 TlvPresent	209
8.151dms_OperatingModeTlv Struct Reference	209
8.151.1 Detailed Description	209
8.151.2 Field Documentation	210
8.151.2.1 operatingMode	210
8.151.2.2 TlvPresent	210
8.152dmsCurrentPRLInfo Struct Reference	210
8.152.1 Detailed Description	210
8.152.2 Field Documentation	210
8.152.2.1 pPRLPreference	210
8.152.2.2 pPRLVersion	210
8.153DMScustSettingInfo Struct Reference	210
8.153.1 Detailed Description	211
8.153.2 Field Documentation	211
8.153.2.1 cust_attr	211
8.153.2.2 cust_id	211

8.153.2.3 cust_value	211
8.153.2.4 id_length	211
8.153.2.5 value_length	211
8.154DMScustSettingList Struct Reference	211
8.154.1 Detailed Description	211
8.154.2 Field Documentation	212
8.154.2.1 custSetting	212
8.154.2.2 list_type	212
8.154.2.3 num_instances	212
8.155DMSgetCustomFeatureV2 Struct Reference	212
8.155.1 Detailed Description	212
8.155.2 Field Documentation	212
8.155.2.1 pCustSettingInfo	212
8.155.2.2 pCustSettingList	212
8.155.2.3 pGetCustomInput	212
8.156DMSgetCustomInput Struct Reference	212
8.156.1 Detailed Description	213
8.156.2 Field Documentation	213
8.156.2.1 cust_id	213
8.156.2.2 list_type	213
8.157dmsIndicationRegisterReq Struct Reference	213
8.157.1 Detailed Description	213
8.157.2 Field Documentation	213
8.157.2.1 pSwiGetResetInd	213
8.158dmsSwiGetResetInfo Struct Reference	213
8.158.1 Detailed Description	214
8.158.2 Field Documentation	214
8.158.2.1 source	214
8.158.2.2 type	214
8.159Domain Struct Reference	214

8.159.1 Detailed Description	214
8.159.2 Field Documentation	214
8.159.2.1 domainLen	214
8.159.2.2 domainName	214
8.160 DomainNameList Struct Reference	215
8.160.1 Detailed Description	215
8.160.2 Field Documentation	215
8.160.2.1 domain	215
8.160.2.2 numInstances	215
8.161 DRCPParams Struct Reference	215
8.161.1 Detailed Description	215
8.161.2 Field Documentation	216
8.161.2.1 DRCCover	216
8.161.2.2 DRCValue	216
8.162 DTMFInfo Struct Reference	216
8.162.1 Detailed Description	216
8.162.2 Field Documentation	216
8.162.2.1 callID	216
8.162.2.2 digitBuff	216
8.162.2.3 digitCnt	217
8.162.2.4 DTMFEvent	217
8.163 DTMFLengths Struct Reference	217
8.163.1 Detailed Description	217
8.163.2 Field Documentation	217
8.163.2.1 DTMFInterdigitInterval	217
8.163.2.2 DTMFPulseWidth	217
8.164 DUNCallInfoInd Struct Reference	217
8.164.1 Field Documentation	218
8.164.1.1 CallEndReason	218
8.164.1.2 ChannelRate	218

8.164.1.3 DataBearerTech	218
8.164.1.4 DormancyStatus	218
8.164.1.5 MdmConnStatus	218
8.164.1.6 RXOKBytesCount	218
8.164.1.7 TXOKBytesCount	218
8.165dunchannelRate Struct Reference	218
8.165.1 Detailed Description	218
8.165.2 Field Documentation	218
8.165.2.1 CurrChanRxRate	218
8.165.2.2 CurrChanTxRate	219
8.165.2.3 MaxChanRxRate	219
8.165.2.4 MaxChanTxRate	219
8.166ecioListElement Struct Reference	219
8.166.1 Detailed Description	219
8.166.2 Field Documentation	219
8.166.2.1 ecio	219
8.166.2.2 radiolf	219
8.167ECIOThresh Struct Reference	219
8.167.1 Detailed Description	220
8.167.2 Field Documentation	220
8.167.2.1 ECIOThresListLen	220
8.167.2.2 pECIOThresList	220
8.168ECTNum Struct Reference	220
8.168.1 Detailed Description	220
8.168.2 Field Documentation	221
8.168.2.1 ECTCallState	221
8.168.2.2 number	221
8.168.2.3 presentationInd	221
8.169encryptedPIN1 Struct Reference	221
8.169.1 Detailed Description	221

8.169.2 Field Documentation	221
8.169.2.1 pin1Len	222
8.169.2.2 pin1Val	222
8.170eriDataparams Struct Reference	222
8.170.1 Field Documentation	222
8.170.1.1 eriData	222
8.170.1.2 eriDataLen	222
8.171ERIFileparams Struct Reference	222
8.171.1 Detailed Description	222
8.171.2 Field Documentation	222
8.171.2.1 pFile	222
8.171.2.2 pFileSize	222
8.172errorRateListElement Struct Reference	223
8.172.1 Detailed Description	223
8.172.2 Field Documentation	223
8.172.2.1 errorRate	223
8.172.2.2 radiolf	223
8.173eTWSPLMNInfoTlv Struct Reference	224
8.173.1 Detailed Description	224
8.173.2 Field Documentation	224
8.173.2.1 ETWSPLMNInfo	224
8.173.2.2 TlvPresent	224
8.174extDispRecInfo Struct Reference	224
8.174.1 Detailed Description	224
8.174.2 Field Documentation	225
8.174.2.1 dispType	225
8.174.2.2 extDispInfo	225
8.174.2.3 extDispInfoLen	225
8.175FactorySequenceNumber Struct Reference	225
8.175.1 Detailed Description	225

8.175.2 Field Documentation	225
8.175.2.1 FSNumber	225
8.176fileAttributes Struct Reference	225
8.176.1 Detailed Description	226
8.176.2 Field Documentation	228
8.176.2.1 fileID	228
8.176.2.2 fileSize	228
8.176.2.3 fileType	228
8.176.2.4 rawLen	228
8.176.2.5 rawValue	228
8.176.2.6 recordCount	228
8.176.2.7 recordSize	228
8.176.2.8 secActivate	228
8.176.2.9 secActivateMask	228
8.176.2.10secDeactivate	228
8.176.2.11secDeactivateMask	228
8.176.2.12secIncrease	228
8.176.2.13secIncreaseMask	228
8.176.2.14secRead	228
8.176.2.15secReadMask	228
8.176.2.16secWrite	229
8.176.2.17secWriteMask	229
8.177fileInfo Struct Reference	229
8.177.1 Detailed Description	229
8.177.2 Field Documentation	229
8.177.2.1 fileID	229
8.177.2.2 path	229
8.177.2.3 pathLen	229
8.178FirmwareUpdatStat Struct Reference	229
8.178.1 Detailed Description	230

8.178.2 Field Documentation	231
8.178.2.1 plmgType	231
8.178.2.2 pLogString	231
8.178.2.3 pLogStringLen	231
8.178.2.4 pRefData	231
8.178.2.5 pRefString	231
8.178.2.6 pRefStringLen	231
8.178.2.7 ResCode	231
8.179FMSImageElement Struct Reference	231
8.179.1 Detailed Description	231
8.179.2 Field Documentation	231
8.179.2.1 buildId	232
8.179.2.2 buildIdLength	232
8.179.2.3 imageId	232
8.179.2.4 imageType	232
8.180FMSImageIdElement Struct Reference	232
8.180.1 Detailed Description	232
8.180.2 Field Documentation	232
8.180.2.1 buildID	232
8.180.2.2 buildIDLength	232
8.180.2.3 failureCount	233
8.180.2.4 imageID	233
8.180.2.5 storageIndex	233
8.181FMSImageIDEntries Struct Reference	233
8.181.1 Detailed Description	233
8.181.2 Field Documentation	233
8.181.2.1 executingImage	233
8.181.2.2 imageIDElement	233
8.181.2.3 imageIDSize	234
8.181.2.4 imageType	234

8.181.2.5 maxImages	234
8.182FMSImageList Struct Reference	234
8.182.1 Detailed Description	234
8.182.2 Field Documentation	234
8.182.2.1 imageIDEntries	234
8.182.2.2 listSize	234
8.183FMSPrefImageList Struct Reference	234
8.183.1 Detailed Description	234
8.183.2 Field Documentation	235
8.183.2.1 listEntries	235
8.183.2.2 listSize	235
8.184fwinfo_s Struct Reference	235
8.184.1 Detailed Description	235
8.184.2 Field Documentation	236
8.184.2.1 Carrier	236
8.184.2.2 FirmwareID	236
8.184.2.3 GPSCapability	236
8.184.2.4 Region	236
8.184.2.5 Technology	236
8.185GERANInfo Struct Reference	236
8.185.1 Detailed Description	236
8.185.2 Field Documentation	237
8.185.2.1 arfcn	237
8.185.2.2 bsic	237
8.185.2.3 cellID	237
8.185.2.4 insNmrCellInfo	237
8.185.2.5 lac	237
8.185.2.6 nmrlnst	237
8.185.2.7 plmn	237
8.185.2.8 rxLev	237

8.185.2.9 timingAdvance	238
8.186geranInstInfo Struct Reference	238
8.186.1 Detailed Description	238
8.186.2 Field Documentation	238
8.186.2.1 geranArfcn	238
8.186.2.2 geranBsicBcc	238
8.186.2.3 geranBsicNcc	238
8.186.2.4 geranRssi	238
8.187getAllCallInformation Struct Reference	238
8.187.1 Detailed Description	239
8.187.2 Field Documentation	239
8.187.2.1 ALS	239
8.187.2.2 Callinfo	239
8.187.2.3 isEmpty	239
8.188getAllCallRmtPtyName Struct Reference	239
8.188.1 Detailed Description	239
8.188.2 Field Documentation	240
8.188.2.1 callID	240
8.188.2.2 RemotePartyName	240
8.189getAllCallRmtPtyNum Struct Reference	240
8.189.1 Detailed Description	240
8.189.2 Field Documentation	240
8.189.2.1 callID	240
8.189.2.2 RemotePartyNum	240
8.190GetAudioPathConfigReq Struct Reference	240
8.190.1 Detailed Description	240
8.190.2 Field Documentation	241
8.190.2.1 Item	241
8.190.2.2 Profile	241
8.191GetAudioPathConfigResp Struct Reference	241

8.191.1 Detailed Description	241
8.191.2 Field Documentation	242
8.191.2.1 pCodecSTGain	243
8.191.2.2 pDTMFTXGain	243
8.191.2.3 pECMode	243
8.191.2.4 pMICGainSelect	243
8.191.2.5 pNSEnable	243
8.191.2.6 pRXAGCList	243
8.191.2.7 pRXAVCAGCSwitch	243
8.191.2.8 pRXAVCList	243
8.191.2.9 pRXPCMIIRFiltr	243
8.191.2.10pTXAGCList	243
8.191.2.11pTXAVCSwitch	243
8.191.2.12pTXGain	243
8.191.2.13pTXPCMIIRFiltr	243
8.192GetAudioProfileReq Struct Reference	243
8.192.1 Detailed Description	243
8.192.2 Field Documentation	244
8.192.2.1 Generator	244
8.193GetAudioProfileResp Struct Reference	244
8.193.1 Detailed Description	244
8.193.2 Field Documentation	244
8.193.2.1 EarMute	245
8.193.2.2 MicMute	245
8.193.2.3 Profile	245
8.193.2.4 Volume	245
8.194GetAudioVoITLBConfigReq Struct Reference	245
8.194.1 Detailed Description	245
8.194.2 Field Documentation	245
8.194.2.1 Generator	245

8.194.2.2 Item	245
8.194.2.3 Profile	246
8.194.2.4 Volume	246
8.195GetAudioVoTLBConfigResp Struct Reference	246
8.195.1 Detailed Description	246
8.195.2 Field Documentation	246
8.195.2.1 ResCode	246
8.196getCallFWExtInfo Struct Reference	246
8.196.1 Detailed Description	246
8.196.2 Field Documentation	247
8.196.2.1 CallFWExtInfo	247
8.196.2.2 numInstances	247
8.197getCallFWInfo Struct Reference	247
8.197.1 Detailed Description	247
8.197.2 Field Documentation	247
8.197.2.1 CallFWInfo	247
8.197.2.2 numInstances	247
8.198getCustomFeatureV2 Struct Reference	247
8.198.1 Detailed Description	247
8.198.2 Field Documentation	248
8.198.2.1 pCustSettingInfo	248
8.198.2.2 pCustSettingList	248
8.198.2.3 pGetCustomInput	248
8.199getCustomInput Struct Reference	248
8.199.1 Detailed Description	248
8.199.2 Field Documentation	248
8.199.2.1 cust_id	248
8.199.2.2 list_type	248
8.200getDUNCallInfoReq Struct Reference	248
8.200.1 Detailed Description	249

8.200.2 Field Documentation	249
8.200.2.1 Mask	249
8.200.2.2 pReportChannelRate	249
8.200.2.3 pReportConnStatus	249
8.200.2.4 pReportDataBearerTech	250
8.200.2.5 pReportDormStatus	250
8.200.2.6 pTransferStatInd	250
8.201 getDUNCallInfoResp Struct Reference	250
8.201.1 Detailed Description	250
8.201.2 Field Documentation	252
8.201.2.1 pCallEndReason	252
8.201.2.2 pChannelRate	252
8.201.2.3 pConnectionStatus	252
8.201.2.4 pDataBearerTech	252
8.201.2.5 pDormancyStatus	252
8.201.2.6 pLastCallDataBearerTech	252
8.201.2.7 pLastCallRXOKBytesCnt	252
8.201.2.8 pLastCallTXOKBytesCnt	252
8.201.2.9 pMdmCallDurationActive	252
8.201.2.10 pRXOKBytesCount	252
8.201.2.11 pTXOKBytesCount	253
8.202 getDyingGaspCfg Struct Reference	253
8.202.1 Detailed Description	253
8.202.2 Field Documentation	253
8.202.2.1 pDestSMSContent	253
8.202.2.2 pDestSMSNum	253
8.203 getDyingGaspStatistics Struct Reference	253
8.203.1 Detailed Description	253
8.203.2 Field Documentation	253
8.203.2.1 pSMSAttemptedFlag	253

8.203.2.2 pTimeStamp	253
8.204GetErrRateResp Struct Reference	254
8.204.1 Detailed Description	254
8.204.2 Field Documentation	254
8.204.2.1 pCDMAFrameErrRate	254
8.204.2.2 pGSMBER	254
8.204.2.3 pHDRPackErrRate	254
8.204.2.4 pWCDMABER	254
8.205GetHRPDStatsResp Struct Reference	255
8.205.1 Detailed Description	255
8.205.2 Field Documentation	255
8.205.2.1 pDRCPParams	255
8.205.2.2 pPilotSetData	255
8.205.2.3 pUATI	255
8.206GetIMSSMSConfigParams Struct Reference	255
8.206.1 Detailed Description	255
8.206.2 Field Documentation	256
8.206.2.1 pPhoneCtxtURI	256
8.206.2.2 pPhoneCtxtURILen	256
8.206.2.3 pSettingResp	256
8.206.2.4 pSMSFormat	256
8.206.2.5 pSMSOverIPNwInd	256
8.207GetIMSUserConfigParams Struct Reference	256
8.207.1 Detailed Description	256
8.207.2 Field Documentation	257
8.207.2.1 pIMSDomain	257
8.207.2.2 pIMSDomainLen	257
8.207.2.3 pSettingResp	257
8.208GetIMSVoIPConfigResp Struct Reference	257
8.208.1 Detailed Description	257

8.208.2 Field Documentation	259
8.208.2.1 pAmrMode	259
8.208.2.2 pAmrOctetAligned	259
8.208.2.3 pAmrWbEnable	259
8.208.2.4 pAmrWBMode	259
8.208.2.5 pAmrWBOctetAligned	259
8.208.2.6 pMinSessionExpiryTimer	260
8.208.2.7 pRingBackTimer	260
8.208.2.8 pRingingTimer	260
8.208.2.9 pRTPRTCPInactTimer	260
8.208.2.10pScrAmrEnable	260
8.208.2.11pScrAmrWbEnable	260
8.208.2.12pSessionExpiryTimer	260
8.208.2.13pSettingResp	260
8.209GetInstIDResp Struct Reference	260
8.209.1 Field Documentation	260
8.209.1.1 pInstanceId	260
8.209.1.2 pIPFamily	260
8.210GetM2MAudioProfileReq Struct Reference	260
8.210.1 Detailed Description	260
8.210.2 Field Documentation	261
8.210.2.1 pGenerator	261
8.211GetM2MAudioProfileResp Struct Reference	261
8.211.1 Detailed Description	261
8.211.2 Field Documentation	262
8.211.2.1 CwtMute	262
8.211.2.2 EarMute	262
8.211.2.3 Generator	262
8.211.2.4 MicMute	262
8.211.2.5 Profile	262

8.211.2.6 Volume	262
8.212GetM2MAudioVolumeReq Struct Reference	262
8.212.1 Detailed Description	262
8.212.2 Field Documentation	262
8.212.2.1 Generator	262
8.212.2.2 Profile	262
8.213GetM2MAudioVolumeResp Struct Reference	262
8.213.1 Detailed Description	263
8.213.2 Field Documentation	263
8.213.2.1 Level	263
8.214GetM2MAVMuteReq Struct Reference	263
8.214.1 Detailed Description	263
8.214.2 Field Documentation	263
8.214.2.1 Profile	263
8.215GetM2MAVMuteResp Struct Reference	263
8.215.1 Detailed Description	264
8.215.2 Field Documentation	264
8.215.2.1 CwtMute	264
8.215.2.2 EarMute	264
8.215.2.3 MicMute	264
8.216GetM2MSpkrGainReq Struct Reference	264
8.216.1 Detailed Description	264
8.216.2 Field Documentation	265
8.216.2.1 Profile	265
8.217GetM2MSpkrGainResp Struct Reference	265
8.217.1 Detailed Description	265
8.217.2 Field Documentation	265
8.217.2.1 Value	265
8.218getMsgWaitingInfo Struct Reference	265
8.218.1 Detailed Description	265

8.218.2 Field Documentation	266
8.218.2.1 msgWaitInfo	266
8.218.2.2 numInstances	266
8.219GetNetworkTimeResp Struct Reference	266
8.219.1 Detailed Description	266
8.219.2 Field Documentation	266
8.219.2.1 p3GPP2TimeInfo	266
8.219.2.2 p3GPPTimeInfo	266
8.220GetRegMgrConfigParams Struct Reference	266
8.220.1 Detailed Description	266
8.220.2 Field Documentation	267
8.220.2.1 pIMSTestMode	267
8.220.2.2 pPCSCFPort	267
8.220.2.3 pPriCSCFPortName	267
8.220.2.4 pPriCSCFPortNameLen	267
8.220.2.5 pSettingResp	267
8.221GetSessionIDResp Struct Reference	267
8.221.1 Field Documentation	268
8.221.1.1 pSessionIDv4	268
8.221.1.2 pSessionIDv6	268
8.222GetSIPConfigResp Struct Reference	268
8.222.1 Detailed Description	268
8.222.2 Field Documentation	269
8.222.2.1 pSettingResp	269
8.222.2.2 pSigCompEnabled	269
8.222.2.3 pSIPLocalPort	269
8.222.2.4 pSubscribeTimer	269
8.222.2.5 pTimerSIPReg	269
8.222.2.6 pTimerT1	269
8.222.2.7 pTimerT2	269

8.222.2.8 pTimerTf	269
8.223GnssData Struct Reference	269
8.223.1 Detailed Description	269
8.223.2 Field Documentation	270
8.223.2.1 mask	270
8.224gnssSvInfoNotification Struct Reference	270
8.224.1 Detailed Description	271
8.224.2 Field Documentation	271
8.224.2.1 bAltitudeAssumed	271
8.224.2.2 pSatelliteInfo	271
8.225GPRSQoS Struct Reference	271
8.225.1 Detailed Description	271
8.225.2 Field Documentation	272
8.225.2.1 delayClass	272
8.225.2.2 meanThroughputClass	272
8.225.2.3 peakThroughputClass	272
8.225.2.4 precedenceClass	272
8.225.2.5 reliabilityClass	272
8.226GPRSRequestedQoS Struct Reference	272
8.226.1 Detailed Description	272
8.226.2 Field Documentation	273
8.226.2.1 delayClass	273
8.226.2.2 meanThroughputClass	273
8.226.2.3 peakThroughputClass	273
8.226.2.4 precedenceClass	273
8.226.2.5 reliabilityClass	273
8.227GPSSStateInfo Struct Reference	273
8.227.1 Detailed Description	274
8.227.2 Field Documentation	276
8.227.2.1 Altitude	276

8.227.2.2 EngineState	276
8.227.2.3 glo_almanac_sv_msk	276
8.227.2.4 glo_ephemeris_sv_msk	276
8.227.2.5 glo_health_sv_msk	276
8.227.2.6 glo_visible_sv_msk	276
8.227.2.7 gps_almanac_sv_msk	276
8.227.2.8 gps_ephemeris_sv_msk	276
8.227.2.9 gps_health_sv_msk	276
8.227.2.10gps_visible_sv_msk	277
8.227.2.11HorizontalUncertainty	277
8.227.2.12ono_valid	277
8.227.2.13Latitude	277
8.227.2.14Longitude	277
8.227.2.15sbas_almanac_sv_msk	277
8.227.2.16sbas_ephemeris_sv_msk	277
8.227.2.17sbas_health_sv_msk	277
8.227.2.18sbas_visible_sv_msk	277
8.227.2.19Time_uncert_ms	277
8.227.2.20TimeStmp_gps_week	277
8.227.2.21TimeStmp_tow_ms	277
8.227.2.22ValidMask	277
8.227.2.23VerticalUncertainty	277
8.227.2.24xtra_start_gps_minutes	277
8.227.2.25xtra_start_gps_week	277
8.227.2.26xtra_valid_duration_hours	277
8.228gpsTime_s Struct Reference	277
8.228.1 Detailed Description	277
8.228.2 Field Documentation	278
8.228.2.1 gpsTimeOfWeekMs	278
8.228.2.2 gpsWeek	278

8.229gsmCellInfo Struct Reference	278
8.229.1 Detailed Description	278
8.229.2 Field Documentation	279
8.229.2.1 arfcn	279
8.229.2.2 band1900	279
8.229.2.3 bsicld	279
8.229.2.4 cellldValid	279
8.229.2.5 rssi	279
8.229.2.6 srxlev	279
8.230GSMRSSIThresh Struct Reference	279
8.230.1 Detailed Description	279
8.230.2 Field Documentation	280
8.230.2.1 GSMRSSIThreshListLen	280
8.230.2.2 pGSMRSSIThreshList	280
8.231GSMSrvStatusInfo Struct Reference	280
8.231.1 Detailed Description	280
8.231.2 Field Documentation	280
8.231.2.1 isPrefDataPath	281
8.231.2.2 srvStatus	281
8.231.2.3 trueSrvStatus	281
8.232GSMSysInfo Struct Reference	281
8.232.1 Detailed Description	281
8.232.2 Field Documentation	283
8.232.2.1 cellld	283
8.232.2.2 cellldValid	283
8.232.2.3 dtmSupp	283
8.232.2.4 dtmSuppValid	283
8.232.2.5 egprsSupp	283
8.232.2.6 egprsSuppValid	283
8.232.2.7 lac	283

8.232.2.8 lacValid	283
8.232.2.9 MCC	283
8.232.2.10MNC	283
8.232.2.11networkIdValid	283
8.232.2.12regRejectInfoValid	283
8.232.2.13rejCause	284
8.232.2.14rejectSrvDomain	284
8.232.2.15sysInfoGSM	284
8.233gyroAcceptReady_s Struct Reference	284
8.233.1 Detailed Description	284
8.233.2 Field Documentation	284
8.233.2.1 batchPerSec	284
8.233.2.2 injectEnable	284
8.233.2.3 samplesPerBatch	284
8.234gyroTempAcceptReady_s Struct Reference	284
8.234.1 Detailed Description	285
8.234.2 Field Documentation	285
8.234.2.1 batchPerSec	285
8.234.2.2 injectEnable	285
8.234.2.3 samplesPerBatch	285
8.235HDRECIOThresh Struct Reference	285
8.235.1 Detailed Description	285
8.235.2 Field Documentation	286
8.235.2.1 HDRECIOThreshListLen	286
8.235.2.2 pHDRECIOThreshList	286
8.236HDRIOTThresh Struct Reference	286
8.236.1 Detailed Description	286
8.236.2 Field Documentation	286
8.236.2.1 HDRIOTThreshListLen	286
8.236.2.2 pHDRIOTThreshList	286

8.237HDRPersonalityInd Struct Reference	286
8.237.1 Field Documentation	287
8.237.1.1 pCurrentPersonality	287
8.237.1.2 pPersonalityListLength	287
8.237.1.3 pProtocolSubtypeElement	287
8.238HDRPersonalityResp Struct Reference	287
8.238.1 Detailed Description	287
8.238.2 Field Documentation	287
8.238.2.1 pCurrentPersonality	287
8.238.2.2 pPersonalityListLength	287
8.238.2.3 pProtocolSubtypeElement	287
8.239HDRProtSubtypResp Struct Reference	288
8.239.1 Detailed Description	288
8.239.2 Field Documentation	288
8.239.2.1 pAppSubType	288
8.239.2.2 pCurrentPrsnlty	288
8.239.2.3 pPersonalityListLength	288
8.239.2.4 pProtoSubTypElmnt	288
8.240HDRRSSIThresh Struct Reference	288
8.240.1 Detailed Description	289
8.240.2 Field Documentation	289
8.240.2.1 HDRRSSIThreshListLen	289
8.240.2.2 pHDRRSSIThreshList	289
8.241HDRSINRThresh Struct Reference	289
8.241.1 Detailed Description	289
8.241.2 Field Documentation	290
8.241.2.1 HDRSINRThresListLen	290
8.241.2.2 pHDRSINRThresList	290
8.242HDRSINRThreshold Struct Reference	290
8.242.1 Detailed Description	290

8.242.2 Field Documentation	290
8.242.2.1 HDRSINRThreshListLen	290
8.242.2.2 pHDRSINRThreshList	290
8.243HDRSSInfo Struct Reference	290
8.243.1 Detailed Description	291
8.243.2 Field Documentation	291
8.243.2.1 ecio	291
8.243.2.2 io	291
8.243.2.3 rssi	291
8.243.2.4 sinr	291
8.244hdrSSInfo Struct Reference	292
8.244.1 Detailed Description	292
8.244.2 Field Documentation	292
8.244.2.1 ecio	292
8.244.2.2 io	292
8.244.2.3 rssi	292
8.244.2.4 sinr	292
8.245HDRSysInfo Struct Reference	292
8.245.1 Detailed Description	292
8.245.2 Field Documentation	294
8.245.2.1 hdrActiveProt	294
8.245.2.2 hdrActiveProtValid	294
8.245.2.3 hdrPersonality	294
8.245.2.4 hdrPersonalityValid	294
8.245.2.5 is856SysId	294
8.245.2.6 is856SysIdValid	294
8.245.2.7 isSysPrIMatch	294
8.245.2.8 isSysPrIMatchValid	294
8.245.2.9 sysInfoHDR	294
8.246homeSIDNID Struct Reference	294

8.246.1 Detailed Description	294
8.246.2 Field Documentation	295
8.246.2.1 numInstances	295
8.246.2.2 SidNid	295
8.247hotSwapStatus Struct Reference	295
8.247.1 Detailed Description	295
8.247.2 Field Documentation	295
8.247.2.1 hotSwap	295
8.247.2.2 hotSwapLength	295
8.248image_info_t Struct Reference	295
8.248.1 Field Documentation	296
8.248.1.1 buildID	296
8.248.1.2 buildIDLen	296
8.248.1.3 imageType	296
8.248.1.4 uniqueID	296
8.249ImageElement Struct Reference	296
8.249.1 Detailed Description	296
8.249.2 Field Documentation	296
8.249.2.1 buildId	296
8.249.2.2 buildIdLength	296
8.249.2.3 imageId	296
8.249.2.4 imageType	297
8.250ImageIdElement Struct Reference	297
8.250.1 Detailed Description	297
8.250.2 Field Documentation	297
8.250.2.1 buildID	297
8.250.2.2 buildIDLength	297
8.250.2.3 failureCount	297
8.250.2.4 imageID	297
8.250.2.5 storageIndex	297

8.251 ImageIDEntries Struct Reference	298
8.251.1 Detailed Description	298
8.251.2 Field Documentation	298
8.251.2.1 executingImage	298
8.251.2.2 imageIDElement	298
8.251.2.3 imageIDSize	298
8.251.2.4 imageType	298
8.251.2.5 maxImages	298
8.252 ImageList Struct Reference	298
8.252.1 Detailed Description	299
8.252.2 Field Documentation	299
8.252.2.1 imageIDEntries	299
8.252.2.2 listSize	299
8.253 IMSAIndRegisterInfo Struct Reference	299
8.253.1 Detailed Description	299
8.253.2 Field Documentation	300
8.253.2.1 pPdpStatusConfig	300
8.253.2.2 pRatHandoverStatusConfig	300
8.253.2.3 pRegStatusConfig	300
8.253.2.4 pServiceStatusConfig	300
8.254 imsaPdpStatusInfo Struct Reference	300
8.254.1 Detailed Description	300
8.254.2 Field Documentation	301
8.254.2.1 connetionState	301
8.254.2.2 pFailErrorCode	301
8.255 imsaRatStatusInfo Struct Reference	301
8.255.1 Detailed Description	301
8.255.2 Field Documentation	301
8.255.2.1 pErrorCodeStr	301
8.255.2.2 pRATStatus	301

8.255.2.3 pSrcRAT	301
8.255.2.4 pTgtRAT	301
8.256IMSARegistrationStatus Struct Reference	301
8.256.1 Detailed Description	302
8.256.2 Field Documentation	302
8.256.2.1 plmsRegErrCode	302
8.256.2.2 plmsRegStatus	302
8.256.2.3 pNewlmsRegStatus	302
8.257imsaRegStatusInfo Struct Reference	302
8.257.1 Detailed Description	303
8.257.2 Field Documentation	303
8.257.2.1 pbIMSRegistered	303
8.257.2.2 plmsRegStatus	303
8.257.2.3 pRegStatusErrorCode	303
8.258IMSAServiceStatus Struct Reference	303
8.258.1 Detailed Description	303
8.258.2 Field Documentation	305
8.258.2.1 pSmsServiceRat	305
8.258.2.2 pSmsServiceStatus	305
8.258.2.3 pUtServiceRat	305
8.258.2.4 pUtServiceStatus	305
8.258.2.5 pVoipServiceRat	305
8.258.2.6 pVoipServiceStatus	305
8.258.2.7 pVsServiceRat	305
8.258.2.8 pVsServiceStatus	305
8.258.2.9 pVtServiceRat	305
8.258.2.10 pVtServiceStatus	305
8.259IMSASupportedFieldsResp Struct Reference	305
8.259.1 Detailed Description	305
8.259.2 Field Documentation	306

8.259.2.1 pIndFieldsList	306
8.259.2.2 pReqFieldsList	306
8.259.2.3 pRespFieldsList	306
8.260IMSASupportedMsgInfo Struct Reference	306
8.260.1 Detailed Description	306
8.260.2 Field Documentation	306
8.260.2.1 pSupportedMsgList	306
8.261imsaSvcStatusInfo Struct Reference	306
8.261.1 Detailed Description	307
8.261.2 Field Documentation	307
8.261.2.1 pSMSSvcRAT	307
8.261.2.2 pSMSSvcStatus	307
8.261.2.3 pUTSvcRAT	307
8.261.2.4 pUTSvcStatus	307
8.261.2.5 pVOIPSvcRAT	307
8.261.2.6 pVOIPSvcStatus	307
8.261.2.7 pVTSvcRAT	307
8.261.2.8 pVTSvcStatus	307
8.262imsCfgIndRegisterInfo Struct Reference	307
8.262.1 Detailed Description	308
8.262.2 Field Documentation	309
8.262.2.1 pRegMgrConfigEvents	309
8.262.2.2 pSIPConfigEvents	309
8.262.2.3 pSMSCConfigEvents	309
8.262.2.4 pUserConfigEvents	309
8.262.2.5 pVoIPConfigEvents	309
8.263imsRegMgrConfigInfo Struct Reference	309
8.263.1 Detailed Description	309
8.263.2 Field Documentation	309
8.263.2.1 pCSCFPortName	309

8.263.2.2 pIMSTestMode	309
8.263.2.3 pPriCSCFPort	309
8.264imsSIPConfigInfo Struct Reference	310
8.264.1 Detailed Description	310
8.264.2 Field Documentation	310
8.264.2.1 pSigCompEnabled	310
8.264.2.2 pSIPLocalPort	311
8.264.2.3 pSubscribeTimer	311
8.264.2.4 pTimerSIPReg	311
8.264.2.5 pTimerT1	311
8.264.2.6 pTimerT2	311
8.264.2.7 pTimerTf	311
8.265imsSMSConfigInfo Struct Reference	311
8.265.1 Detailed Description	311
8.265.2 Field Documentation	311
8.265.2.1 pPhoneCtxURI	311
8.265.2.2 pSMSFormat	311
8.265.2.3 pSMSOverIPNwInd	312
8.266imsUserConfigInfo Struct Reference	312
8.266.1 Detailed Description	312
8.266.2 Field Documentation	312
8.266.2.1 pIMSDomain	312
8.267imsVoIPConfigInfo Struct Reference	312
8.267.1 Detailed Description	312
8.267.2 Field Documentation	314
8.267.2.1 pAmrMode	314
8.267.2.2 pAmrOctetAligned	314
8.267.2.3 pAmrWbEnable	314
8.267.2.4 pAmrWBMode	314
8.267.2.5 pAmrWBOctetAligned	314

8.267.2.6 pMinSessionExpiryTimer	314
8.267.2.7 pRingBackTimer	314
8.267.2.8 pRingingTimer	314
8.267.2.9 pRTPRTCPInactTimer	314
8.267.2.10pScrAmrEnable	314
8.267.2.11pScrAmrWbEnable	315
8.267.2.12pSessionExpiryTimer	315
8.268IndFieldsList Struct Reference	315
8.268.1 Detailed Description	315
8.268.2 Field Documentation	315
8.268.2.1 indicationFields	315
8.268.2.2 indicationFieldsLen	315
8.269infoInterFreq Struct Reference	315
8.269.1 Detailed Description	315
8.269.2 Field Documentation	316
8.269.2.1 cell_resel_priority	316
8.269.2.2 cellInterFreqParams	316
8.269.2.3 cells_len	316
8.269.2.4 earfcn	316
8.269.2.5 threshXHigh	316
8.269.2.6 threshXLow	316
8.270IOThresh Struct Reference	316
8.270.1 Detailed Description	317
8.270.2 Field Documentation	317
8.270.2.1 IOThresListLen	317
8.270.2.2 pIOThresList	317
8.271IPv4Addr Struct Reference	317
8.271.1 Detailed Description	317
8.271.2 Field Documentation	317
8.271.2.1 addr	318

8.271.2.2 subnetMask	318
8.272IPv6Addr Struct Reference	318
8.272.1 Detailed Description	318
8.272.2 Field Documentation	318
8.272.2.1 addr	318
8.272.2.2 prefixLen	318
8.273IPv6AddressInfo Struct Reference	318
8.273.1 Detailed Description	318
8.273.2 Field Documentation	319
8.273.2.1 IPAddressV6	319
8.273.2.2 IPV6PrefixLen	319
8.274ipv6AddressInfo Struct Reference	319
8.274.1 Detailed Description	319
8.274.2 Field Documentation	319
8.274.2.1 IPAddressV6	319
8.274.2.2 IPV6PrefixLen	319
8.275IPv6GWAddressInfo Struct Reference	319
8.275.1 Detailed Description	319
8.275.2 Field Documentation	320
8.275.2.1 gwAddressV6	320
8.275.2.2 gwV6PrefixLen	320
8.276IPv6TrafCls Struct Reference	320
8.276.1 Detailed Description	320
8.276.2 Field Documentation	320
8.276.2.1 mask	320
8.276.2.2 val	320
8.277LibPackGPRSRequestedQoS Struct Reference	320
8.277.1 Detailed Description	321
8.277.2 Field Documentation	321
8.277.2.1 delayClass	321

8.277.2.2 meanThroughputClass	321
8.277.2.3 peakThroughputClass	321
8.277.2.4 precedenceClass	321
8.277.2.5 reliabilityClass	321
8.278LibpackProfile3GPP Struct Reference	321
8.278.1 Detailed Description	322
8.278.2 Field Documentation	326
8.278.2.1 pAddrAllocPref	326
8.278.2.2 pAPNClass	326
8.278.2.3 pAPNDisabledFlag	326
8.278.2.4 pAPNName	326
8.278.2.5 pAPNnameSize	326
8.278.2.6 pAuthenticationPref	326
8.278.2.7 pGPRSMinimumQoS	326
8.278.2.8 pGPRSRequestedQos	326
8.278.2.9 plmCnFlag	326
8.278.2.10pIPv4AddrPref	326
8.278.2.11pIPv6AddPref	326
8.278.2.12pPassword	326
8.278.2.13pPasswordSize	326
8.278.2.14pPcscfAddrUsingDhcp	326
8.278.2.15pPcscfAddrUsingPCO	326
8.278.2.16pPDNInactivTimeout	326
8.278.2.17pPdpAccessConFlag	326
8.278.2.18pPdpContext	326
8.278.2.19pPdpDataCompType	327
8.278.2.20pPdpHdrCompType	327
8.278.2.21pPDPtype	327
8.278.2.22pPriDNSIPv4AddPref	327
8.278.2.23pPriDNSIPv6addpref	327

8.278.2.24	pPrimaryID	327
8.278.2.25	pProfilename	327
8.278.2.26	pProfilenameSize	327
8.278.2.27	pQosClassID	327
8.278.2.28	pSecDNSIPv4AddPref	327
8.278.2.29	pSecDNSIPv6addpref	327
8.278.2.30	pSecondaryFlag	327
8.278.2.31	pTFTID1Params	327
8.278.2.32	pTFTID2Params	327
8.278.2.33	pUMTSMinQoS	327
8.278.2.34	pUMTSMinQoSSigInd	327
8.278.2.35	pUMTSReqQoS	327
8.278.2.36	pUMTSReqQoSSigInd	327
8.278.2.37	pUsername	327
8.278.2.38	pUsernameSize	327
8.279	LibpackProfile3GPP2 Struct Reference	327
8.279.1	Detailed Description	328
8.279.2	Field Documentation	331
8.279.2.1	pAllowLinger	331
8.279.2.2	pAPNClass3GPP2	332
8.279.2.3	pAPNEnabled3GPP2	332
8.279.2.4	pApnString	332
8.279.2.5	pApnStringSize	332
8.279.2.6	pAppPriority	332
8.279.2.7	pAppType	332
8.279.2.8	pAuthPassword	332
8.279.2.9	pAuthPasswordSize	332
8.279.2.10	pAuthProtocol	332
8.279.2.11	pAuthRetryCount	332
8.279.2.12	pAuthTimeout	332

8.279.2.13pDataMode	332
8.279.2.14pDataRate	332
8.279.2.15pIpcpAckTimeout	332
8.279.2.16pIpcpCreqRetryCount	332
8.279.2.17pIsPcscfAddressNedded	332
8.279.2.18pLcpAckTimeout	332
8.279.2.19pLcpCreqRetryCount	332
8.279.2.20pNegoDnsSrvrPref	332
8.279.2.21pPDNInactivTimeout3GPP2	332
8.279.2.22pPdnType	332
8.279.2.23pPppSessCloseTimer1x	332
8.279.2.24pPppSessCloseTimerDO	332
8.279.2.25pPrimaryV4DnsAddress	333
8.279.2.26pPriV6DnsAddress	333
8.279.2.27pRATType	333
8.279.2.28pSecondaryV4DnsAddress	333
8.279.2.29pSecV6DnsAddress	333
8.279.2.30pUserId	333
8.279.2.31pUserIdSize	333
8.280LibPackprofile_3GPP Struct Reference	333
8.280.1 Detailed Description	334
8.280.2 Field Documentation	337
8.280.2.1 pAddrAllocPref	337
8.280.2.2 pAPNClass	337
8.280.2.3 pAPNDisabledFlag	337
8.280.2.4 pAPNName	337
8.280.2.5 pAPNnameSize	337
8.280.2.6 pAuthenticationPref	337
8.280.2.7 pGPRSMinimumQoS	338
8.280.2.8 pGPRSRequestedQos	338

8.280.2.9	plmCnFlag	338
8.280.2.10	pIPv4AddrPref	338
8.280.2.11	pIPv6AddPref	338
8.280.2.12	pPassword	338
8.280.2.13	pPasswordSize	338
8.280.2.14	pPcscfAddrUsingDhcp	338
8.280.2.15	pPcscfAddrUsingPCO	338
8.280.2.16	pPDNInactivTimeout	338
8.280.2.17	pPdpAccessConFlag	338
8.280.2.18	pPdpContext	338
8.280.2.19	pPdpDataCompType	338
8.280.2.20	pPdpHdrCompType	338
8.280.2.21	pPDType	338
8.280.2.22	pPriDNSIPv4AddPref	338
8.280.2.23	pPriDNSIPv6addpref	338
8.280.2.24	pPrimaryID	338
8.280.2.25	pProfilename	338
8.280.2.26	pProfilenameSize	338
8.280.2.27	pQosClassID	338
8.280.2.28	pSecDNSIPv4AddPref	338
8.280.2.29	pSecDNSIPv6addpref	338
8.280.2.30	pSecondaryFlag	339
8.280.2.31	pTFTID1Params	339
8.280.2.32	pTFTID2Params	339
8.280.2.33	pUMTSMInQoS	339
8.280.2.34	pUMTSMInQoSSigInd	339
8.280.2.35	pUMTSReqQoS	339
8.280.2.36	pUMTSReqQoSSigInd	339
8.280.2.37	pUsername	339
8.280.2.38	pUsernameSize	339

8.281 LibPackprofile_3GPP2 Struct Reference	339
8.281.1 Detailed Description	340
8.281.2 Field Documentation	343
8.281.2.1 pAllowLinger	343
8.281.2.2 pAPNClass3GPP2	343
8.281.2.3 pAPNEnabled3GPP2	343
8.281.2.4 pApnString	343
8.281.2.5 pApnStringSize	343
8.281.2.6 pAppPriority	343
8.281.2.7 pAppType	343
8.281.2.8 pAuthPassword	343
8.281.2.9 pAuthPassword_tSize	343
8.281.2.10 pAuthProtocol	343
8.281.2.11 pAuthRetryCount	343
8.281.2.12 pAuthTimeout	343
8.281.2.13 pDataMode	343
8.281.2.14 pDataRate	343
8.281.2.15 pIpcpAckTimeout	343
8.281.2.16 pIpcpCreqRetryCount	344
8.281.2.17 pIsPcscfAddressNedded	344
8.281.2.18 pLcpAckTimeout	344
8.281.2.19 pLcpCreqRetryCount	344
8.281.2.20 pNegoDnsSrvrPref	344
8.281.2.21 pPDNInactivTimeout3GPP2	344
8.281.2.22 pPdnType	344
8.281.2.23 pPppSessCloseTimer1x	344
8.281.2.24 pPppSessCloseTimerDO	344
8.281.2.25 pPrimaryV4DnsAddress	344
8.281.2.26 pPriV6DnsAddress	344
8.281.2.27 pRATType	344

8.281.2.28pSecondaryV4DnsAddress	344
8.281.2.29pSecV6DnsAddress	344
8.281.2.30pUserId	344
8.281.2.31pUserIdSize	344
8.282LibPackQosClassID Struct Reference	344
8.282.1 Detailed Description	344
8.282.2 Field Documentation	345
8.282.2.1 gDIBitRate	345
8.282.2.2 gUIBitRate	345
8.282.2.3 maxDIBitRate	345
8.282.2.4 maxUIBitRate	345
8.282.2.5 QCI	345
8.283LibPackTFTIDParams Struct Reference	345
8.283.1 Detailed Description	346
8.283.2 Field Documentation	347
8.283.2.1 destPortRangeEnd	347
8.283.2.2 destPortRangeStart	347
8.283.2.3 eValid	347
8.283.2.4 filterId	347
8.283.2.5 flowLabel	347
8.283.2.6 IPSECSPi	347
8.283.2.7 ipVersion	347
8.283.2.8 nextHeader	347
8.283.2.9 pSourceIP	347
8.283.2.10sourceIPMask	347
8.283.2.11srcPortRangeEnd	347
8.283.2.12srcPortRangeStart	347
8.283.2.13osMask	347
8.284LibPackUMTSQoS Struct Reference	347
8.284.1 Detailed Description	347

8.284.2 Field Documentation	349
8.284.2.1 deliveryErrSDU	349
8.284.2.2 grntDownlinkBitrate	349
8.284.2.3 grntUplinkBitrate	349
8.284.2.4 maxDownlinkBitrate	349
8.284.2.5 maxSDUSize	349
8.284.2.6 maxUplinkBitrate	349
8.284.2.7 qosDeliveryOrder	349
8.284.2.8 resBerRatio	349
8.284.2.9 sduErrorRatio	349
8.284.2.10 trafficClass	349
8.284.2.11 trafficPriority	349
8.284.2.12 transferDelay	349
8.285 LibPackUMTSReqQoS SigInd Struct Reference	349
8.285.1 Detailed Description	350
8.285.2 Field Documentation	350
8.285.2.1 SigInd	350
8.285.2.2 UMTSReqQoS	350
8.286 lineCtrlInfo Struct Reference	350
8.286.1 Detailed Description	350
8.286.2 Field Documentation	351
8.286.2.1 polarityIncluded	351
8.286.2.2 pwrDenialTime	351
8.286.2.3 revPolarity	351
8.286.2.4 toggleMode	351
8.287 loc_BdsSV Struct Reference	351
8.287.1 Detailed Description	351
8.287.2 Field Documentation	351
8.287.2.1 id	351
8.287.2.2 mask	351

8.288loc_BdsSVInfo Struct Reference	351
8.288.1 Detailed Description	352
8.288.2 Field Documentation	352
8.288.2.1 len	352
8.288.2.2 pSV	352
8.289loc_CellDb Struct Reference	352
8.289.1 Detailed Description	352
8.289.2 Field Documentation	353
8.289.2.1 mask	353
8.290loc_ClkInfo Struct Reference	353
8.290.1 Detailed Description	353
8.290.2 Field Documentation	354
8.290.2.1 mask	354
8.291loc_GnssData Struct Reference	354
8.291.1 Detailed Description	354
8.291.2 Field Documentation	355
8.291.2.1 mask	355
8.292loc_gpsTime Struct Reference	355
8.292.1 Detailed Description	356
8.292.2 Field Documentation	356
8.292.2.1 gpsTimeOfWeekMs	356
8.292.2.2 gpsWeek	356
8.293loc_LocApplicationInfo Struct Reference	356
8.293.1 Detailed Description	356
8.293.2 Field Documentation	357
8.293.2.1 appNameLength	357
8.293.2.2 appProviderLength	357
8.293.2.3 appVersionLength	357
8.293.2.4 appVersionValid	357
8.293.2.5 pAppName	357

8.293.2.6 pAppProvider	357
8.293.2.7 pAppVersion	357
8.294loc_precisionDilution Struct Reference	357
8.294.1 Detailed Description	358
8.294.2 Field Documentation	358
8.294.2.1 HDOP	358
8.294.2.2 PDOP	358
8.294.2.3 VDOP	358
8.295loc_satelliteInfo Struct Reference	358
8.295.1 Detailed Description	358
8.295.2 Field Documentation	360
8.295.2.1 azimuth	360
8.295.2.2 elevation	360
8.295.2.3 gnssSvId	360
8.295.2.4 healthStatus	360
8.295.2.5 snr	360
8.295.2.6 svInfoMask	360
8.295.2.7 svListLen	360
8.295.2.8 svStatus	360
8.295.2.9 system	360
8.295.2.10validMask	360
8.296loc_sensorDataUsage Struct Reference	360
8.296.1 Detailed Description	361
8.296.2 Field Documentation	361
8.296.2.1 aidingIndicatorMask	361
8.296.2.2 usageMask	361
8.297loc_SV Struct Reference	361
8.297.1 Detailed Description	361
8.297.2 Field Documentation	362
8.297.2.1 id	362

8.297.2.2 mask	362
8.297.2.3 system	362
8.298loc_SVInfo Struct Reference	362
8.298.1 Detailed Description	362
8.298.2 Field Documentation	363
8.298.2.1 len	363
8.298.2.2 pSV	363
8.299loc_svUsedforFix Struct Reference	363
8.299.1 Detailed Description	363
8.299.2 Field Documentation	363
8.299.2.1 gnssSvUsedList	363
8.299.2.2 gnssSvUsedList_len	363
8.300LocApplicationInfo Struct Reference	363
8.300.1 Detailed Description	364
8.300.2 Field Documentation	364
8.300.2.1 appNameLength	364
8.300.2.2 appProviderLength	364
8.300.2.3 appVersionLength	364
8.300.2.4 appVersionValid	364
8.300.2.5 pAppName	364
8.300.2.6 pAppProvider	364
8.300.2.7 pAppVersion	364
8.301LocDelAssDataReq Struct Reference	365
8.301.1 Detailed Description	365
8.301.2 Field Documentation	365
8.301.2.1 pBdsSVInfo	365
8.301.2.2 pCellDb	365
8.301.2.3 pCikInfo	365
8.301.2.4 pGnssData	365
8.301.2.5 pSVInfo	365

8.302LOCEventRegisterReqResp Struct Reference	366
8.302.1 Detailed Description	366
8.302.2 Field Documentation	368
8.302.2.1 eventRegister	368
8.303LOCExtPowerStateReqResp Struct Reference	368
8.303.1 Detailed Description	368
8.303.2 Field Documentation	368
8.303.2.1 extPowerState	368
8.304LocInjectPositionReq Struct Reference	368
8.304.1 Detailed Description	369
8.304.2 Field Documentation	372
8.304.2.1 pAltitudeSrcInfo	372
8.304.2.2 pAltitudeWrtEllipsoid	372
8.304.2.3 pAltitudeWrtMeanSeaLevel	372
8.304.2.4 pHorConfidence	372
8.304.2.5 pHorReliability	372
8.304.2.6 pHorUncCircular	372
8.304.2.7 pLatitude	372
8.304.2.8 pLongitude	372
8.304.2.9 pPositionSrc	372
8.304.2.10pRawHorConfidence	372
8.304.2.11pRawHorUncCircular	372
8.304.2.12pTimestampAge	372
8.304.2.13pTimestampUtc	372
8.304.2.14pVertConfidence	372
8.304.2.15pVertReliability	372
8.304.2.16pVertUnc	372
8.305LocInjectSensorDataReq Struct Reference	372
8.305.1 Detailed Description	372
8.305.2 Field Documentation	373

8.305.2.1 pAcceleroData	373
8.305.2.2 pAcceleroTempData	373
8.305.2.3 pAcceleroTimeSrc	373
8.305.2.4 pGyroData	374
8.305.2.5 pGyroTempData	374
8.305.2.6 pGyroTimeSrc	374
8.305.2.7 pOpaqueIdentifier	374
8.306LocSetCradleMountReq Struct Reference	374
8.306.1 Detailed Description	374
8.306.2 Field Documentation	374
8.306.2.1 pConfidence	374
8.306.2.2 state	374
8.307LOCStartReq Struct Reference	374
8.307.1 Detailed Description	375
8.307.2 Field Documentation	376
8.307.2.1 pApplicationInfo	376
8.307.2.2 pConfigAltitudeAssumed	376
8.307.2.3 pHorizontalAccuracyLvl	376
8.307.2.4 pIntermediateReportState	376
8.307.2.5 pMinIntervalTime	376
8.307.2.6 pRecurrenceType	376
8.307.2.7 SessionId	376
8.308LOCStopReq Struct Reference	376
8.308.1 Detailed Description	376
8.308.2 Field Documentation	376
8.308.2.1 sessionId	377
8.309LteCQIParm Struct Reference	377
8.309.1 Detailed Description	377
8.309.2 Field Documentation	377
8.309.2.1 CQIValueCW0	377

8.309.2.2 CQIValueCW1	377
8.309.2.3 ValidityCW0	377
8.309.2.4 ValidityCW1	377
8.310lteEARFCN Struct Reference	377
8.310.1 Detailed Description	378
8.310.2 Field Documentation	378
8.310.2.1 earfcn0	378
8.310.2.2 earfcn1	378
8.310.2.3 status	378
8.311lteGsmCellInfo Struct Reference	378
8.311.1 Detailed Description	378
8.311.2 Field Documentation	379
8.311.2.1 cellReselPriority	379
8.311.2.2 cells_len	379
8.311.2.3 GsmCellInfo	379
8.311.2.4 nccPermitted	379
8.311.2.5 threshGsmHigh	379
8.311.2.6 threshGsmLow	379
8.312LTEInfo Struct Reference	379
8.312.1 Detailed Description	380
8.312.2 Field Documentation	381
8.312.2.1 band	381
8.312.2.2 bandwidth	381
8.312.2.3 emmConnState	381
8.312.2.4 emmState	381
8.312.2.5 emmSubState	381
8.312.2.6 RXChan	381
8.312.2.7 TXChan	382
8.313LTEInfoInterfreq Struct Reference	382
8.313.1 Detailed Description	382

8.313.2 Field Documentation	382
8.313.2.1 freqsLen	382
8.313.2.2 InfoInterfreq	382
8.313.2.3 ueInIdle	382
8.314LTEInfoIntrafreq Struct Reference	382
8.314.1 Detailed Description	383
8.314.2 Field Documentation	384
8.314.2.1 CellParams	384
8.314.2.2 cellReselPriority	384
8.314.2.3 cellsLen	384
8.314.2.4 earfcn	384
8.314.2.5 globalCellId	384
8.314.2.6 plmn	384
8.314.2.7 servingCellId	384
8.314.2.8 sIntraSearch	384
8.314.2.9 sNonIntraSearch	384
8.314.2.10tac	384
8.314.2.11threshServingLow	384
8.314.2.12ueInIdle	384
8.315LTEInfoNeighboringGSM Struct Reference	384
8.315.1 Detailed Description	385
8.315.2 Field Documentation	385
8.315.2.1 freqsLen	385
8.315.2.2 LteGsmCellInfo	385
8.315.2.3 ueInIdle	385
8.316LTEInfoNeighboringWCDMA Struct Reference	385
8.316.1 Detailed Description	385
8.316.2 Field Documentation	386
8.316.2.1 freqsLen	386
8.316.2.2 LTEWCDMACellInfo	386

8.316.2.3 ueInIdle	386
8.317LteNasReleaseInfo_s Struct Reference	386
8.317.1 Detailed Description	386
8.317.2 Field Documentation	386
8.317.2.1 nas_major	386
8.317.2.2 nas_minor	386
8.317.2.3 nas_release	386
8.318ltePCI Struct Reference	386
8.318.1 Detailed Description	387
8.318.2 Field Documentation	387
8.318.2.1 earfcn	387
8.318.2.2 pci	387
8.318.2.3 status	387
8.319lteRsrpInformation Struct Reference	387
8.319.1 Detailed Description	387
8.319.2 Field Documentation	387
8.319.2.1 rsrplevel	387
8.320LTERS RPThresh Struct Reference	388
8.320.1 Detailed Description	388
8.320.2 Field Documentation	388
8.320.2.1 LTERS RPThreshListLen	388
8.320.2.2 pLTERS RPThreshList	388
8.321LTERS RQThresh Struct Reference	388
8.321.1 Detailed Description	388
8.321.2 Field Documentation	389
8.321.2.1 LTERS RQThreshListLen	389
8.321.2.2 pLTERS RQThreshList	389
8.322LTERS SIThresh Struct Reference	389
8.322.1 Detailed Description	389
8.322.2 Field Documentation	389

8.322.2.1 LTERSSIThreshListLen	389
8.322.2.2 pLTERSSIThreshList	389
8.323LteSccRxInfoResp Struct Reference	389
8.323.1 Detailed Description	389
8.323.2 Field Documentation	390
8.323.2.1 pSccRxInfo	390
8.324LTESigRptCfg Struct Reference	390
8.324.1 Detailed Description	390
8.324.2 Field Documentation	391
8.324.2.1 avgPeriod	391
8.324.2.2 rptRate	391
8.325LTESigRptConfig Struct Reference	391
8.325.1 Detailed Description	391
8.325.2 Field Documentation	391
8.325.2.1 avgPeriod	391
8.325.2.2 rptRate	392
8.326LteSnrinformation Struct Reference	392
8.326.1 Detailed Description	392
8.326.2 Field Documentation	392
8.326.2.1 snrlevel	392
8.327LTESNRThresh Struct Reference	392
8.327.1 Detailed Description	392
8.327.2 Field Documentation	393
8.327.2.1 LTESNRThresListLen	393
8.327.2.2 pLTESNRThresList	393
8.328LTESNRThreshold Struct Reference	393
8.328.1 Detailed Description	393
8.328.2 Field Documentation	393
8.328.2.1 LTESNRThreshListLen	393
8.328.2.2 pLTESNRThreshList	393

8.329LTESSInfo Struct Reference	393
8.329.1 Detailed Description	393
8.329.2 Field Documentation	394
8.329.2.1 rsrp	394
8.329.2.2 rsrq	394
8.329.2.3 rssi	394
8.329.2.4 snr	394
8.330lteSSInfo Struct Reference	394
8.330.1 Detailed Description	394
8.330.2 Field Documentation	395
8.330.2.1 rsrp	395
8.330.2.2 rsrq	395
8.330.2.3 rssi	395
8.330.2.4 snr	395
8.331LTESysInfo Struct Reference	395
8.331.1 Detailed Description	395
8.331.2 Field Documentation	397
8.331.2.1 cellId	397
8.331.2.2 cellIdValid	397
8.331.2.3 lac	397
8.331.2.4 lacValid	397
8.331.2.5 MCC	397
8.331.2.6 MNC	397
8.331.2.7 networkIdValid	397
8.331.2.8 regRejectInfoValid	397
8.331.2.9 rejCause	397
8.331.2.10rejectSrvDomain	397
8.331.2.11sysInfoLTE	397
8.331.2.12lac	397
8.331.2.13lacValid	397

8.332IteWcdmaCellInfo Struct Reference	397
8.332.1 Detailed Description	397
8.332.2 Field Documentation	398
8.332.2.1 cellReselPriority	398
8.332.2.2 cellsLen	398
8.332.2.3 threshXhigh	398
8.332.2.4 threshXlow	398
8.332.2.5 uarfcn	398
8.332.2.6 WCDMACellInfo	398
8.333messageModeTlv Struct Reference	398
8.333.1 Detailed Description	398
8.333.2 Field Documentation	399
8.333.2.1 MessageModelInfo	399
8.333.2.2 TlvPresent	399
8.334messageWaitingInfoContent Struct Reference	399
8.334.1 Detailed Description	399
8.334.2 Field Documentation	399
8.334.2.1 activeInd	400
8.334.2.2 msgCount	400
8.334.2.3 msgType	400
8.335minBasedIMSI Struct Reference	400
8.335.1 Detailed Description	400
8.335.2 Field Documentation	400
8.335.2.1 imsiM1112	400
8.335.2.2 imsiMS1	400
8.335.2.3 imsiMS2	400
8.335.2.4 mccM	400
8.336mitigationDevList Struct Reference	400
8.336.1 Detailed Description	401
8.336.2 Field Documentation	401

8.336.2.1 maxMitigationLevel	401
8.336.2.2 mitigationDevId	401
8.336.2.3 mitigationDevIdLen	401
8.337MNRInfo Struct Reference	401
8.337.1 Detailed Description	401
8.337.2 Field Documentation	402
8.337.2.1 mcc	402
8.337.2.2 mnc	402
8.337.2.3 rat	402
8.338ModifyProfileIn Struct Reference	402
8.338.1 Detailed Description	402
8.338.2 Field Documentation	403
8.338.2.1 curProfile	403
8.338.2.2 pProfileID	403
8.338.2.3 pProfileType	403
8.339ModifyProfileOut Struct Reference	403
8.339.1 Detailed Description	403
8.339.2 Field Documentation	403
8.339.2.1 pExtErrorCode	403
8.340msgWaitingInfo Struct Reference	403
8.340.1 Detailed Description	403
8.340.2 Field Documentation	404
8.340.2.1 msgWaitInfo	404
8.340.2.2 numInstances	404
8.341namName Struct Reference	404
8.341.1 Detailed Description	404
8.341.2 Field Documentation	404
8.341.2.1 namName	404
8.341.2.2 namNameLen	404
8.342nas_acqOrderPref Struct Reference	404

8.342.1 Detailed Description	405
8.342.2 Field Documentation	405
8.342.2.1 acqOrdeLen	405
8.342.2.2 pAcqOrder	405
8.343nas_AddCDMASysInfo Struct Reference	405
8.343.1 Detailed Description	405
8.343.2 Field Documentation	406
8.343.2.1 geoSysIdx	406
8.343.2.2 regPrd	406
8.344nas_AddSysInfo Struct Reference	406
8.344.1 Detailed Description	406
8.344.2 Field Documentation	406
8.344.2.1 cellBroadcastCap	406
8.344.2.2 geoSysIdx	406
8.345nas_CallBarringSysInfo Struct Reference	406
8.345.1 Detailed Description	407
8.345.2 Field Documentation	407
8.345.2.1 csBarStatus	407
8.345.2.2 psBarStatus	407
8.346nas_callBarStatus Struct Reference	407
8.346.1 Detailed Description	407
8.346.2 Field Documentation	408
8.346.2.1 csBarStatus	408
8.346.2.2 psBarStatus	408
8.347nas_CDMAECIOThresh Struct Reference	408
8.347.1 Detailed Description	408
8.347.2 Field Documentation	409
8.347.2.1 CDMAECIOThreshListLen	409
8.347.2.2 pCDMAECIOThreshList	409
8.348nas_CDMAInfo Struct Reference	409

8.348.1 Detailed Description	409
8.348.2 Field Documentation	410
8.348.2.1 baseId	410
8.348.2.2 baseLat	410
8.348.2.3 baseLong	410
8.348.2.4 nid	410
8.348.2.5 refpn	410
8.348.2.6 sid	410
8.349nas_CDMARSSIThresh Struct Reference	410
8.349.1 Detailed Description	410
8.349.2 Field Documentation	410
8.349.2.1 CDMARSSIThreshListLen	410
8.349.2.2 pCDMARSSIThreshList	410
8.350nas_CDMA SysInfo Struct Reference	410
8.350.1 Detailed Description	411
8.350.2 Field Documentation	413
8.350.2.1 baseId	413
8.350.2.2 baseLat	413
8.350.2.3 baseLong	413
8.350.2.4 bsInfoValid	413
8.350.2.5 bsPRev	414
8.350.2.6 bsPRevValid	414
8.350.2.7 ccsSupported	414
8.350.2.8 ccsSupportedValid	414
8.350.2.9 cdmaSysIdValid	414
8.350.2.10sSysPriMatch	414
8.350.2.11isSysPriMatchValid	414
8.350.2.12MCC	414
8.350.2.13MNC	414
8.350.2.14networkID	414

8.350.2.15networkIdValid	414
8.350.2.16packetZone	414
8.350.2.17packetZoneValid	414
8.350.2.18pRevInUse	414
8.350.2.19pRevInUseValid	414
8.350.2.20sysInfoCDMA	414
8.350.2.21systemID	414
8.351nas_CDMASysInfoExt Struct Reference	414
8.351.1 Detailed Description	414
8.351.2 Field Documentation	415
8.351.2.1 imsi_11_12	415
8.351.2.2 MCC	415
8.352nas_cellParams Struct Reference	415
8.352.1 Detailed Description	415
8.352.2 Field Documentation	416
8.352.2.1 pci	416
8.352.2.2 rsrp	416
8.352.2.3 rsrq	416
8.352.2.4 rssi	416
8.352.2.5 srxlev	416
8.353nas_CommInfo Struct Reference	416
8.353.1 Detailed Description	416
8.353.2 Field Documentation	417
8.353.2.1 imsRegState	417
8.353.2.2 modemMode	417
8.353.2.3 psState	417
8.353.2.4 systemMode	417
8.353.2.5 temperature	417
8.354nas_CSGID Struct Reference	417
8.354.1 Detailed Description	418

8.354.2 Field Documentation	418
8.354.2.1 id	418
8.354.2.2 mcc	418
8.354.2.3 mnc	418
8.354.2.4 mncPcsDigits	418
8.354.2.5 rat	418
8.355nas_currentPLMN Struct Reference	418
8.355.1 Detailed Description	419
8.355.2 Field Documentation	419
8.355.2.1 MCC	419
8.355.2.2 MNC	419
8.355.2.3 netDescr	419
8.355.2.4 netDescrLength	419
8.356nas_dataSrvCapabilities Struct Reference	419
8.356.1 Detailed Description	419
8.356.2 Field Documentation	420
8.356.2.1 dataCapabilities	420
8.356.2.2 dataCapabilitiesLen	420
8.357nas_detailSvcInfo Struct Reference	420
8.357.1 Detailed Description	420
8.357.2 Field Documentation	421
8.357.2.1 hdrHybrid	421
8.357.2.2 hdrSrvStatus	421
8.357.2.3 isSysForbidden	421
8.357.2.4 srvCapability	421
8.357.2.5 srvStatus	422
8.358nas_ecioListElement Struct Reference	422
8.358.1 Detailed Description	422
8.358.2 Field Documentation	422
8.358.2.1 ecio	422

8.358.2.2 radiolf	422
8.359nas_errorRateListElement Struct Reference	422
8.359.1 Detailed Description	422
8.359.2 Field Documentation	423
8.359.2.1 errorRate	423
8.359.2.2 radiolf	423
8.360nas_GERANInfo Struct Reference	423
8.360.1 Detailed Description	424
8.360.2 Field Documentation	425
8.360.2.1 arfcn	425
8.360.2.2 bsic	425
8.360.2.3 cellID	425
8.360.2.4 insNmrCellInfo	425
8.360.2.5 lac	425
8.360.2.6 nmrlnst	425
8.360.2.7 plmn	425
8.360.2.8 rxLev	425
8.360.2.9 timingAdvance	425
8.361nas_geranInstInfo Struct Reference	425
8.361.1 Detailed Description	425
8.361.2 Field Documentation	426
8.361.2.1 geranArfcn	426
8.361.2.2 geranBsicBcc	426
8.361.2.3 geranBsicNcc	426
8.361.2.4 geranRssi	426
8.362nas_gsmCellInfo Struct Reference	426
8.362.1 Detailed Description	426
8.362.2 Field Documentation	427
8.362.2.1 arfcn	427
8.362.2.2 band1900	427

8.362.2.3 bsicld	427
8.362.2.4 cellIdValid	427
8.362.2.5 rssi	427
8.362.2.6 srxlev	427
8.363nas_GSMRSSIThresh Struct Reference	427
8.363.1 Detailed Description	427
8.363.2 Field Documentation	427
8.363.2.1 GSMRSSIThreshListLen	427
8.363.2.2 pGSMRSSIThreshList	427
8.364nas_GSMSrvStatusInfo Struct Reference	427
8.364.1 Detailed Description	428
8.364.2 Field Documentation	428
8.364.2.1 isPrefDataPath	428
8.364.2.2 srvStatus	428
8.364.2.3 trueSrvStatus	428
8.365nas_GSMSysInfo Struct Reference	428
8.365.1 Detailed Description	429
8.365.2 Field Documentation	431
8.365.2.1 cellId	431
8.365.2.2 cellIdValid	431
8.365.2.3 dtmSupp	431
8.365.2.4 dtmSuppValid	431
8.365.2.5 egprsSupp	431
8.365.2.6 egprsSuppValid	431
8.365.2.7 lac	431
8.365.2.8 lacValid	431
8.365.2.9 MCC	431
8.365.2.10MNC	431
8.365.2.11networkIdValid	431
8.365.2.12regRejectInfoValid	431

8.365.2.13rejCause	431
8.365.2.14rejectSrvDomain	431
8.365.2.15sysInfoGSM	431
8.366nas_HDRECIOThresh Struct Reference	431
8.366.1 Detailed Description	431
8.366.2 Field Documentation	432
8.366.2.1 HDRECIOThreshListLen	432
8.366.2.2 pHRECIOThreshList	432
8.367nas_HDRIOThresh Struct Reference	432
8.367.1 Detailed Description	432
8.367.2 Field Documentation	432
8.367.2.1 HDRIOThreshListLen	432
8.367.2.2 pHDRIOThreshList	432
8.368nas_HDRRSSIThresh Struct Reference	432
8.368.1 Detailed Description	433
8.368.2 Field Documentation	433
8.368.2.1 HDRRSSIThreshListLen	433
8.368.2.2 pHDRRSSIThreshList	433
8.369nas_HDRSINRThreshold Struct Reference	433
8.369.1 Detailed Description	433
8.369.2 Field Documentation	434
8.369.2.1 HDRSINRThreshListLen	434
8.369.2.2 pHDRSINRThreshList	434
8.370nas_HDRSysInfo Struct Reference	434
8.370.1 Detailed Description	434
8.370.2 Field Documentation	436
8.370.2.1 hdrActiveProt	436
8.370.2.2 hdrActiveProtValid	436
8.370.2.3 hdrPersonality	436
8.370.2.4 hdrPersonalityValid	436

8.370.2.5 is856SysId	436
8.370.2.6 is856SysIdValid	436
8.370.2.7 isSysPrIMatch	436
8.370.2.8 isSysPrIMatchValid	436
8.370.2.9 sysInfoHDR	436
8.371nas_infoInterFreq Struct Reference	436
8.371.1 Detailed Description	436
8.371.2 Field Documentation	437
8.371.2.1 cell_resel_priority	437
8.371.2.2 cellInterFreqParams	437
8.371.2.3 cells_len	437
8.371.2.4 earfcn	437
8.371.2.5 threshXHigh	437
8.371.2.6 threshXLow	437
8.372nas_lteGsmCellInfo Struct Reference	437
8.372.1 Detailed Description	438
8.372.2 Field Documentation	438
8.372.2.1 cellReselPriority	438
8.372.2.2 cells_len	438
8.372.2.3 GsmCellInfo	438
8.372.2.4 nccPermitted	438
8.372.2.5 threshGsmHigh	438
8.372.2.6 threshGsmLow	438
8.373nas_LTEInfo Struct Reference	439
8.373.1 Detailed Description	439
8.373.2 Field Documentation	440
8.373.2.1 band	440
8.373.2.2 bandwidth	440
8.373.2.3 emmConnState	440
8.373.2.4 emmState	440

8.373.2.5 emmSubState	440
8.373.2.6 RXChan	440
8.373.2.7 TXChan	441
8.374nas_LTEInfoInterfreq Struct Reference	441
8.374.1 Detailed Description	441
8.374.2 Field Documentation	441
8.374.2.1 freqsLen	441
8.374.2.2 InfoInterfreq	441
8.374.2.3 ueInIdle	441
8.375nas_LTEInfoIntrafreq Struct Reference	441
8.375.1 Detailed Description	442
8.375.2 Field Documentation	443
8.375.2.1 CellParams	443
8.375.2.2 cellReselPriority	443
8.375.2.3 cellsLen	443
8.375.2.4 earfcn	443
8.375.2.5 globalCellId	443
8.375.2.6 plmn	443
8.375.2.7 servingCellId	443
8.375.2.8 sIntraSearch	443
8.375.2.9 sNonIntraSearch	443
8.375.2.10tac	443
8.375.2.11threshServingLow	443
8.375.2.12ueInIdle	444
8.376nas_LTEInfoNeighboringGSM Struct Reference	444
8.376.1 Detailed Description	444
8.376.2 Field Documentation	444
8.376.2.1 freqsLen	444
8.376.2.2 LteGsmCellInfo	444
8.376.2.3 ueInIdle	444

8.377nas_LTEInfoNeighboringWCDMA Struct Reference	444
8.377.1 Detailed Description	445
8.377.2 Field Documentation	445
8.377.2.1 freqsLen	445
8.377.2.2 LTEWCDMACellInfo	445
8.377.2.3 ueInIdle	445
8.378nas_lteRsrpInformation Struct Reference	445
8.378.1 Detailed Description	445
8.378.2 Field Documentation	445
8.378.2.1 rsrplevel	446
8.379nas_LTERSRPThresh Struct Reference	446
8.379.1 Detailed Description	446
8.379.2 Field Documentation	446
8.379.2.1 LTERSRPThreshListLen	446
8.379.2.2 pLTERSRPThreshList	446
8.380nas_LTERSRQThresh Struct Reference	446
8.380.1 Detailed Description	446
8.380.2 Field Documentation	447
8.380.2.1 LTERSRQThreshListLen	447
8.380.2.2 pLTERSRQThreshList	447
8.381nas_LTERSSIThresh Struct Reference	447
8.381.1 Detailed Description	447
8.381.2 Field Documentation	447
8.381.2.1 LTERSSIThreshListLen	447
8.381.2.2 pLTERSSIThreshList	447
8.382nas_LTESigRptConfig Struct Reference	447
8.382.1 Detailed Description	447
8.382.2 Field Documentation	448
8.382.2.1 avgPeriod	448
8.382.2.2 rptRate	448

8.383nas_ItSnrinformation Struct Reference	448
8.383.1 Detailed Description	448
8.383.2 Field Documentation	449
8.383.2.1 snrlevel	449
8.384nas_LTESNRThreshold Struct Reference	449
8.384.1 Detailed Description	449
8.384.2 Field Documentation	449
8.384.2.1 LTESNRThreshListLen	449
8.384.2.2 pLTESNRThreshList	449
8.385nas_LTESysInfo Struct Reference	449
8.385.1 Detailed Description	450
8.385.2 Field Documentation	451
8.385.2.1 cellId	451
8.385.2.2 cellIdValid	451
8.385.2.3 lac	451
8.385.2.4 lacValid	451
8.385.2.5 MCC	451
8.385.2.6 MNC	451
8.385.2.7 networkIdValid	451
8.385.2.8 regRejectInfoValid	451
8.385.2.9 rejCause	452
8.385.2.10rejectSrvDomain	452
8.385.2.11sysInfoLTE	452
8.385.2.12tac	452
8.385.2.13tacValid	452
8.386nas_ItWcdmaCellInfo Struct Reference	452
8.386.1 Detailed Description	452
8.386.2 Field Documentation	453
8.386.2.1 cellReselPriority	453
8.386.2.2 cellsLen	453

8.386.2.3 threshXhigh	453
8.386.2.4 threshXlow	453
8.386.2.5 uarfcn	453
8.386.2.6 WCDMACellInfo	453
8.387nas_MNRInfo Struct Reference	453
8.387.1 Detailed Description	453
8.387.2 Field Documentation	453
8.387.2.1 mcc	453
8.387.2.2 mnc	453
8.387.2.3 rat	454
8.388nas_netSelectionPref Struct Reference	454
8.388.1 Detailed Description	454
8.388.2 Field Documentation	454
8.388.2.1 mcc	454
8.388.2.2 mnc	454
8.388.2.3 netReg	454
8.389nas_nmrCellInfo Struct Reference	454
8.389.1 Detailed Description	455
8.389.2 Field Documentation	455
8.389.2.1 nmrArfcn	455
8.389.2.2 nmrBsic	455
8.389.2.3 nmrCellID	455
8.389.2.4 nmrLac	455
8.389.2.5 nmrPlmn	456
8.389.2.6 nmrRxLev	456
8.390nas_PhyCaAggPcellInfo Struct Reference	456
8.390.1 Detailed Description	456
8.390.2 Field Documentation	456
8.390.2.1 dl_bw_value	456
8.390.2.2 freq	456

8.390.2.3 iLTEbandValue	456
8.390.2.4 pci	457
8.390.2.5 TlvPresent	457
8.391nas_PhyCaAggScellDIBw Struct Reference	457
8.391.1 Detailed Description	457
8.391.2 Field Documentation	457
8.391.2.1 dl_bw_value	457
8.391.2.2 TlvPresent	457
8.392nas_PhyCaAggScellIndex Struct Reference	457
8.392.1 Detailed Description	457
8.392.2 Field Documentation	458
8.392.2.1 scell_idx	458
8.392.2.2 TlvPresent	458
8.393nas_PhyCaAggScellIndType Struct Reference	458
8.393.1 Detailed Description	458
8.393.2 Field Documentation	458
8.393.2.1 freq	458
8.393.2.2 pci	458
8.393.2.3 scell_state	458
8.393.2.4 TlvPresent	458
8.394nas_PhyCaAggScellInfo Struct Reference	459
8.394.1 Detailed Description	459
8.394.2 Field Documentation	461
8.394.2.1 dl_bw_value	461
8.394.2.2 freq	461
8.394.2.3 iLTEbandValue	461
8.394.2.4 pci	461
8.394.2.5 scell_state	461
8.394.2.6 TlvPresent	461
8.395nas_qaQmi3Gpp2TimeZone Struct Reference	461

8.395.1 Detailed Description	461
8.395.2 Field Documentation	462
8.395.2.1 daylightSavings	462
8.395.2.2 leapSeconds	462
8.395.2.3 localTimeOffset	462
8.396nas_QmiNas3GppNetworkInfo Struct Reference	462
8.396.1 Detailed Description	462
8.396.2 Field Documentation	462
8.396.2.1 Description	462
8.396.2.2 Forbidden	462
8.396.2.3 InUse	462
8.396.2.4 MCC	462
8.396.2.5 MNC	462
8.396.2.6 Preferred	462
8.396.2.7 Roaming	463
8.397nas_QmiNas3GppNetworkRAT Struct Reference	463
8.397.1 Detailed Description	463
8.397.2 Field Documentation	463
8.397.2.1 MCC	463
8.397.2.2 MNC	463
8.397.2.3 RAT	463
8.398nas_QmisNasPcsDigit Struct Reference	463
8.398.1 Detailed Description	464
8.398.2 Field Documentation	464
8.398.2.1 includes_pcs_digit	464
8.398.2.2 MCC	464
8.398.2.3 MNC	464
8.399nas_RejectReasonTlv Struct Reference	464
8.399.1 Detailed Description	464
8.399.2 Field Documentation	464

8.399.2.1 rejectCause	464
8.399.2.2 serviceDomain	464
8.399.2.3 TlvPresent	465
8.400nas_RFInfoTlv Struct Reference	465
8.400.1 Detailed Description	465
8.400.2 Field Documentation	465
8.400.2.1 activeBandClass	465
8.400.2.2 activeChannel	465
8.400.2.3 radiolInterface	465
8.400.2.4 radiolInterfaceSize	465
8.400.2.5 TlvPresent	465
8.401nas_roamIndList Struct Reference	465
8.401.1 Detailed Description	465
8.401.2 Field Documentation	466
8.401.2.1 numInstances	466
8.401.2.2 radiolInterface	466
8.401.2.3 roamIndicator	466
8.402nas_rsrqInformation Struct Reference	466
8.402.1 Detailed Description	466
8.402.2 Field Documentation	467
8.402.2.1 radiolf	467
8.402.2.2 rsrq	467
8.403nas_RxSigInfo Struct Reference	467
8.403.1 Detailed Description	467
8.403.2 Field Documentation	468
8.403.2.1 isRadioTuned	468
8.403.2.2 rsrp	468
8.403.2.3 rxChainIndex	468
8.403.2.4 rxPower	468
8.404nas_rxSignalStrengthListElement Struct Reference	468

8.404.1 Detailed Description	468
8.404.2 Field Documentation	468
8.404.2.1 radiolf	468
8.404.2.2 rxSignalStrength	469
8.405nas_SccRxInfo Struct Reference	469
8.405.1 Detailed Description	469
8.405.2 Field Documentation	469
8.405.2.1 numInstances	469
8.405.2.2 rsrq	469
8.405.2.3 sigInfo	469
8.405.2.4 snr	470
8.405.2.5 TlvPresent	470
8.406nas_servSystem Struct Reference	470
8.406.1 Detailed Description	470
8.406.2 Field Documentation	471
8.406.2.1 csAttachState	471
8.406.2.2 numRadioInterfaces	471
8.406.2.3 psAttachState	471
8.406.2.4 radioInterface	471
8.406.2.5 regState	471
8.406.2.6 selNetwork	471
8.407nas_SignalStrengthTlv Struct Reference	471
8.407.1 Detailed Description	471
8.407.2 Field Documentation	472
8.407.2.1 radioInterface	472
8.407.2.2 signalStrength	472
8.407.2.3 TlvPresent	472
8.408nas_SLQSSignalStrengthsIndReq Struct Reference	472
8.408.1 Detailed Description	472
8.408.2 Field Documentation	473

8.408.2.1 ecioDelta	473
8.408.2.2 ecioThresholdList	473
8.408.2.3 ecioThresholdListLen	473
8.408.2.4 ioDelta	473
8.408.2.5 lteRsrpDelta	473
8.408.2.6 lteSnrDelta	473
8.408.2.7 rsrqDelta	473
8.408.2.8 rxSignalStrengthDelta	473
8.408.2.9 sinrDelta	473
8.408.2.10sinrThresholdList	473
8.408.2.11sinrThresholdListLen	473
8.409nas_SLQSSignalStrengthsInformation Struct Reference	473
8.409.1 Detailed Description	473
8.409.2 Field Documentation	474
8.409.2.1 ecioInfo	474
8.409.2.2 errorRateInfo	474
8.409.2.3 io	474
8.409.2.4 lteRsrpinfo	474
8.409.2.5 lteSnrinfo	474
8.409.2.6 rsrqInfo	474
8.409.2.7 rxSignalStrengthInfo	474
8.409.2.8 sinr	474
8.410nas_SLQSSignalStrengthsTlv Struct Reference	474
8.410.1 Detailed Description	474
8.410.2 Field Documentation	474
8.410.2.1 sSLQSSignalStrengthsInfo	474
8.410.2.2 TlvPresent	474
8.411nas_SrvStatusInfo Struct Reference	474
8.411.1 Detailed Description	475
8.411.2 Field Documentation	475

8.411.2.1 isPrefDataPath	475
8.411.2.2 srvStatus	475
8.412nas_sysInfoCommon Struct Reference	475
8.412.1 Detailed Description	475
8.412.2 Field Documentation	477
8.412.2.1 isSysForbidden	477
8.412.2.2 isSysForbiddenValid	477
8.412.2.3 roamStatus	477
8.412.2.4 roamStatusValid	477
8.412.2.5 srvCapability	477
8.412.2.6 srvCapabilityValid	477
8.412.2.7 srvDomain	478
8.412.2.8 srvDomainValid	478
8.413nas_TDSCDMAECIOThresh Struct Reference	478
8.413.1 Detailed Description	478
8.413.2 Field Documentation	478
8.413.2.1 pTDSCDMAECIOThreshList	478
8.413.2.2 TDSCDMAECIOThreshListLen	478
8.414nas_TDSCDMARSCPThresh Struct Reference	478
8.414.1 Detailed Description	478
8.414.2 Field Documentation	479
8.414.2.1 pTDSCDMARSCPThreshList	479
8.414.2.2 TDSCDMARSCPThreshListLen	479
8.415nas_TDSCDMARSSIThresh Struct Reference	479
8.415.1 Detailed Description	479
8.415.2 Field Documentation	479
8.415.2.1 pTDSCDMARSSIThreshList	479
8.415.2.2 TDSCDMARSSIThreshListLen	479
8.416nas_TDSCDMASINRThresh Struct Reference	479
8.416.1 Detailed Description	480

8.416.2 Field Documentation	480
8.416.2.1 pTDSCDMASINRThreshList	480
8.416.2.2 TDSCDMASINRThreshListLen	480
8.417nas_timeInfo Struct Reference	480
8.417.1 Detailed Description	480
8.417.2 Field Documentation	482
8.417.2.1 day	482
8.417.2.2 dayLtSavingAdj	482
8.417.2.3 dayOfWeek	482
8.417.2.4 hour	482
8.417.2.5 minute	482
8.417.2.6 month	482
8.417.2.7 radiolInterface	482
8.417.2.8 second	482
8.417.2.9 timeZone	482
8.417.2.10TivPresent	482
8.417.2.11year	482
8.418nas_UMTSInfo Struct Reference	482
8.418.1 Detailed Description	483
8.418.2 Field Documentation	483
8.418.2.1 cellID	484
8.418.2.2 ecio	484
8.418.2.3 geranInst	484
8.418.2.4 GeranInstInfo	484
8.418.2.5 lac	484
8.418.2.6 plmn	484
8.418.2.7 psc	484
8.418.2.8 rscp	484
8.418.2.9 uarfcn	484
8.418.2.10umtsInst	484

8.418.2.11UMTSInstInfo	484
8.419nas_UMTSInstInfo Struct Reference	484
8.419.1 Detailed Description	484
8.419.2 Field Documentation	485
8.419.2.1 umtsEcio	485
8.419.2.2 umtsPsc	485
8.419.2.3 umtsRscp	485
8.419.2.4 umtsUarfcn	485
8.420nas_umtsLTENbrCell Struct Reference	485
8.420.1 Detailed Description	485
8.420.2 Field Documentation	486
8.420.2.1 cellsTDD	486
8.420.2.2 earfcn	486
8.420.2.3 pci	486
8.420.2.4 rsrp	486
8.420.2.5 rsrq	486
8.420.2.6 srxlev	486
8.421nas_UniversalTime Struct Reference	486
8.421.1 Detailed Description	486
8.421.2 Field Documentation	487
8.421.2.1 day	487
8.421.2.2 dayOfWeek	487
8.421.2.3 hour	487
8.421.2.4 minute	487
8.421.2.5 month	487
8.421.2.6 second	487
8.421.2.7 year	487
8.422nas_wcdmaCellInfo Struct Reference	487
8.422.1 Detailed Description	488
8.422.2 Field Documentation	488

8.422.2.1 cpich_ecno	488
8.422.2.2 cpich_rscp	488
8.422.2.3 psc	488
8.422.2.4 srxlev	488
8.423nas_WCDMAECIOThresh Struct Reference	488
8.423.1 Detailed Description	488
8.423.2 Field Documentation	489
8.423.2.1 pWCDMAECIOThreshList	489
8.423.2.2 WCDMAECIOThreshListLen	489
8.424nas_WCDMAInfoLTENeighborCell Struct Reference	489
8.424.1 Detailed Description	489
8.424.2 Field Documentation	490
8.424.2.1 UMTSLTENbrCell	490
8.424.2.2 umtsLTENbrCellLen	490
8.424.2.3 wcdmaRRCCState	490
8.425nas_WCDMARSSIThresh Struct Reference	490
8.425.1 Detailed Description	490
8.425.2 Field Documentation	490
8.425.2.1 pWCDMARSSIThreshList	490
8.425.2.2 WCDMARSSIThreshListLen	490
8.426nas_WCDMASysInfo Struct Reference	490
8.426.1 Detailed Description	491
8.426.2 Field Documentation	493
8.426.2.1 cellId	493
8.426.2.2 cellIdValid	493
8.426.2.3 hsCallStatus	493
8.426.2.4 hsCallStatusValid	493
8.426.2.5 hsInd	493
8.426.2.6 hsIndValid	493
8.426.2.7 lac	493

8.426.2.8 lacValid	493
8.426.2.9 MCC	494
8.426.2.10MNC	494
8.426.2.11networkIdValid	494
8.426.2.12psc	494
8.426.2.13pscValid	494
8.426.2.14regRejectInfoValid	494
8.426.2.15rejCause	494
8.426.2.16rejectSrvDomain	494
8.426.2.17sysInfoWCDMA	494
8.427NASBandPreferenceTlv Struct Reference	494
8.427.1 Field Documentation	494
8.427.1.1 band_pref	494
8.427.1.2 TlvPresent	494
8.428nasCellLocationInfoResp Struct Reference	494
8.428.1 Detailed Description	494
8.428.2 Field Documentation	495
8.428.2.1 pCDMAInfo	495
8.428.2.2 pGERANInfo	495
8.428.2.3 pLTEInfoInterfreq	495
8.428.2.4 pLTEInfoIntrafreq	495
8.428.2.5 pLTEInfoNeighboringGSM	495
8.428.2.6 pLTEInfoNeighboringWCDMA	495
8.428.2.7 pUMTSCellID	495
8.428.2.8 pUMTSInfo	495
8.428.2.9 pWCDMAInfoLTENeighborCell	495
8.429NASEmergencyModeTlv Struct Reference	495
8.429.1 Field Documentation	496
8.429.1.1 EmerMode	496
8.429.1.2 TlvPresent	496

8.430nasGet3GPP2SubscriptionInfoReq Struct Reference	496
8.430.1 Detailed Description	496
8.430.2 Field Documentation	496
8.430.2.1 namID	496
8.431nasGet3GPP2SubscriptionInfoResp Struct Reference	496
8.431.1 Detailed Description	496
8.431.2 Field Documentation	497
8.431.2.1 pCDMAChannel	497
8.431.2.2 pDirNum	497
8.431.2.3 pHomeSIDNID	497
8.431.2.4 pMinBasedIMSI	497
8.431.2.5 pNAMNameInfo	497
8.431.2.6 pTrueIMSI	497
8.432nasGetHDRColorCodeResp Struct Reference	497
8.432.1 Detailed Description	497
8.432.2 Field Documentation	498
8.432.2.1 pColorCode	498
8.433nasGetLTECphyCa Struct Reference	498
8.433.1 Field Documentation	498
8.433.1.1 sPhyCaAggPcellInfo	498
8.433.1.2 sPhyCaAggScellIDBw	498
8.433.1.3 sPhyCaAggScellIndex	498
8.433.1.4 sPhyCaAggScellIndType	498
8.433.1.5 sPhyCaAggScellInfo	498
8.434NasGetLTECphyCaInfo Struct Reference	498
8.434.1 Field Documentation	498
8.434.1.1 PhyCaAggPcellInfo	498
8.434.1.2 PhyCaAggScellIDBw	498
8.434.1.3 PhyCaAggScellIndex	498
8.434.1.4 PhyCaAggScellIndType	498

8.434.1.5 PhyCaAggScellInfo	498
8.435nasGetLTECphyCaResp Struct Reference	498
8.435.1 Field Documentation	499
8.435.1.1 pPhyCaAggPcellInfo	499
8.435.1.2 pPhyCaAggScellIDIBw	499
8.435.1.3 pPhyCaAggScellIndex	499
8.435.1.4 pPhyCaAggScellIndType	499
8.435.1.5 pPhyCaAggScellInfo	499
8.436nasGetSigInfoResp Struct Reference	499
8.436.1 Detailed Description	499
8.436.2 Field Documentation	500
8.436.2.1 pCDMASSInfo	500
8.436.2.2 pGSMSSInfo	500
8.436.2.3 pHDRSSInfo	500
8.436.2.4 pLTESSInfo	500
8.436.2.5 pTDSCDMASigInfoExt	500
8.436.2.6 pTDSCDMASigInfoRscp	500
8.436.2.7 pWCDMASSInfo	500
8.437nasGetSysInfoResp Struct Reference	500
8.437.1 Detailed Description	501
8.437.2 Field Documentation	502
8.437.2.1 pAddCDMASysInfo	502
8.437.2.2 pAddGSMSysInfo	502
8.437.2.3 pAddHDRSysInfo	502
8.437.2.4 pAddLTESysInfo	502
8.437.2.5 pAddWCDMASysInfo	502
8.437.2.6 pCDMASrvStatusInfo	502
8.437.2.7 pCDMASysInfo	502
8.437.2.8 pGSMCallBarringSysInfo	502
8.437.2.9 pGSMCipherDomainSysInfo	502

8.437.2.10pGSMSTxStatusInfo	502
8.437.2.11pGSMSTxInfo	503
8.437.2.12pHDRSTxStatusInfo	503
8.437.2.13pHDRSTxInfo	503
8.437.2.14pLTERSTxStatusInfo	503
8.437.2.15pLTERSTxInfo	503
8.437.2.16pLTEVoiceSupportTxInfo	503
8.437.2.17pWCDMACTxBarringTxInfo	503
8.437.2.18pWCDMACTxCipherDomainTxInfo	503
8.437.2.19pWCDMASTxStatusInfo	503
8.437.2.20pWCDMASTxInfo	503
8.438nasGetTxRxInfoReq Struct Reference	503
8.438.1 Detailed Description	503
8.438.2 Field Documentation	503
8.438.2.1 radio_if	503
8.439nasGetTxRxInfoResp Struct Reference	504
8.439.1 Detailed Description	504
8.439.2 Field Documentation	504
8.439.2.1 pRXChain0Info	504
8.439.2.2 pRXChain1Info	504
8.439.2.3 pTXInfo	504
8.440NASGWAqOrderPrefTlv Struct Reference	504
8.440.1 Field Documentation	504
8.440.1.1 GWAqOrderPref	504
8.440.1.2 TlvPresent	504
8.441nasIndicationRegisterReq Struct Reference	504
8.441.1 Detailed Description	505
8.441.2 Field Documentation	507
8.441.2.1 pDDTMInd	507
8.441.2.2 pDualStandByPrefInd	507

8.441.2.3 pErrorRateInd	507
8.441.2.4 pHDRNewUATIAssInd	507
8.441.2.5 pHDRSessionCloseInd	507
8.441.2.6 pLTECphyCa	507
8.441.2.7 pManagedRoamingInd	507
8.441.2.8 pNetworkTimeInd	507
8.441.2.9 pServingSystemInd	507
8.441.2.10pSignalStrengthInd	507
8.441.2.11pSubscriptionInfoInd	507
8.441.2.12pSysInfoInd	507
8.441.2.13pSystemSelectionInd	507
8.442nasInitNetworkReg Struct Reference	507
8.442.1 Detailed Description	508
8.442.2 Field Documentation	508
8.442.2.1 pChangeDuration	508
8.442.2.2 pMncPcsDigitStatus	508
8.442.2.3 pMNRInfo	508
8.442.2.4 regAction	508
8.443NASLTEBandPreferenceTlv Struct Reference	508
8.443.1 Field Documentation	509
8.443.1.1 LTEBandPref	509
8.443.1.2 TlvPresent	509
8.444NASLteNasReleaseInfoTlv Struct Reference	509
8.444.1 Field Documentation	509
8.444.1.1 nas_major	509
8.444.1.2 nas_minor	509
8.444.1.3 nas_release	509
8.444.1.4 TlvPresent	509
8.445NASModePreferenceTlv Struct Reference	509
8.445.1 Field Documentation	509

8.445.1.1 ModePref	509
8.445.1.2 TlvPresent	509
8.446NASNetSelPreferenceTlv Struct Reference	509
8.446.1 Field Documentation	510
8.446.1.1 NetSelPref	510
8.446.1.2 TlvPresent	510
8.447nasNetworkTime Struct Reference	510
8.447.1 Detailed Description	510
8.447.2 Field Documentation	510
8.447.2.1 pDayltSavAdj	510
8.447.2.2 pRadioInterface	511
8.447.2.3 pTimeZone	511
8.447.2.4 universalTime	511
8.448nasOperatorNameResp Struct Reference	511
8.448.1 Detailed Description	511
8.448.2 Field Documentation	511
8.448.2.1 pNITZInformation	511
8.448.2.2 pOperatorNameString	511
8.448.2.3 pOperatorPLMNList	512
8.448.2.4 pPLMNNetworkName	512
8.448.2.5 pSvcProviderName	512
8.449NASOTAMessageTlv Struct Reference	512
8.449.1 Field Documentation	512
8.449.1.1 data_buf	512
8.449.1.2 data_len	512
8.449.1.3 message_type	512
8.449.1.4 TlvPresent	512
8.450NASPhyCaAggPcellInfo Struct Reference	512
8.450.1 Detailed Description	513
8.450.2 Field Documentation	513

8.450.2.1 dl_bw_value	513
8.450.2.2 freq	513
8.450.2.3 iLTEbandValue	513
8.450.2.4 pci	513
8.450.2.5 TlvPresent	513
8.451NASPhyCaAggScellDIBw Struct Reference	513
8.451.1 Detailed Description	513
8.451.2 Field Documentation	514
8.451.2.1 dl_bw_value	514
8.451.2.2 TlvPresent	514
8.452NASPhyCaAggScellIndex Struct Reference	514
8.452.1 Detailed Description	514
8.452.2 Field Documentation	514
8.452.2.1 scell_idx	514
8.452.2.2 TlvPresent	514
8.453NASPhyCaAggScellIndType Struct Reference	514
8.453.1 Detailed Description	515
8.453.2 Field Documentation	515
8.453.2.1 freq	515
8.453.2.2 pci	515
8.453.2.3 scell_state	515
8.453.2.4 TlvPresent	515
8.454NASPhyCaAggScellInfo Struct Reference	515
8.454.1 Detailed Description	515
8.454.2 Field Documentation	516
8.454.2.1 dl_bw_value	516
8.454.2.2 freq	516
8.454.2.3 iLTEbandValue	516
8.454.2.4 pci	516
8.454.2.5 scell_state	516

8.454.2.6 TlvPresent	516
8.455nasPLMNNameReq Struct Reference	516
8.455.1 Detailed Description	516
8.455.2 Field Documentation	517
8.455.2.1 mcc	517
8.455.2.2 mnc	517
8.455.2.3 pMncPcsStatus	517
8.456nasPLMNNameResp Struct Reference	517
8.456.1 Detailed Description	517
8.456.2 Field Documentation	519
8.456.2.1 longName	519
8.456.2.2 longNameCI	519
8.456.2.3 longNameEn	519
8.456.2.4 longNameLen	519
8.456.2.5 longNameSB	519
8.456.2.6 shortName	519
8.456.2.7 shortNameCI	519
8.456.2.8 shortNameEn	519
8.456.2.9 shortNameLen	520
8.456.2.10shortNameSB	520
8.456.2.11spn	520
8.456.2.12spnEncoding	520
8.456.2.13spnLength	520
8.457NASPRLPreferenceTlv Struct Reference	520
8.457.1 Field Documentation	520
8.457.1.1 PRLPref	520
8.457.1.2 TlvPresent	520
8.458NASQmiCbkNasSwiOTAMessageInd Struct Reference	520
8.458.1 Field Documentation	520
8.458.1.1 nasRelInfoTlv	520

8.458.1.2 otaMsgTlv	520
8.458.1.3 timeTlv	520
8.459NASQmiCbkNasSystemSelPrefInd Struct Reference	520
8.459.1 Field Documentation	521
8.459.1.1 BPTlv	521
8.459.1.2 EMTlv	521
8.459.1.3 GWAOPTlv	521
8.459.1.4 LBPTlv	521
8.459.1.5 MPTlv	521
8.459.1.6 NSPTlv	521
8.459.1.7 PRLPTlv	521
8.459.1.8 RPTlv	521
8.459.1.9 SDPTlv	521
8.460NASRoamPreferenceTlv Struct Reference	521
8.460.1 Field Documentation	521
8.460.1.1 RoamPref	521
8.460.1.2 TlvPresent	521
8.461NASServDomainPrefTlv Struct Reference	521
8.461.1 Field Documentation	522
8.461.1.1 SrvDomainPref	522
8.461.1.2 TlvPresent	522
8.462NASServingSystemInfo Struct Reference	522
8.462.1 Detailed Description	522
8.462.2 Field Documentation	523
8.462.2.1 csAttachState	523
8.462.2.2 hdrPersonality	523
8.462.2.3 psAttachState	523
8.462.2.4 radiolInterfaceList	523
8.462.2.5 radiolInterfaceNo	523
8.462.2.6 registrationState	523

8.462.2.7 selectedNetwork	523
8.463nasSigInfo Struct Reference	523
8.463.1 Detailed Description	523
8.463.2 Field Documentation	524
8.463.2.1 pCDMASigInfo	524
8.463.2.2 pGSMSigInfo	524
8.463.2.3 pHDRSigInfo	524
8.463.2.4 pLTESigInfo	524
8.463.2.5 pRscp	524
8.463.2.6 pTDSCDMASigInfoExt	524
8.463.2.7 pWCDMASigInfo	524
8.464nasSwiGetChannelLockResp Struct Reference	524
8.464.1 Detailed Description	525
8.464.2 Field Documentation	525
8.464.2.1 pLteEARFCN	525
8.464.2.2 pLtePCI	525
8.464.2.3 pWcdmaUARFCN	525
8.465NasSwiIndReg Struct Reference	525
8.465.1 Detailed Description	525
8.465.2 Field Documentation	526
8.465.2.1 gsmUmtsDI	526
8.465.2.2 gsmUmtsUI	526
8.465.2.3 lteEmmDI	526
8.465.2.4 lteEmmUI	526
8.465.2.5 lteEsmDI	526
8.465.2.6 lteEsmUI	526
8.465.2.7 pRankIndicatorInd	526
8.465.2.8 pTimer	527
8.466nasSwiSetChannelLockReq Struct Reference	527
8.466.1 Detailed Description	527

8.466.2 Field Documentation	527
8.466.2.1 pLteEARFCN	527
8.466.2.2 pLtePCI	527
8.466.2.3 pWcdmaUARFCN	527
8.467nasSysInfo Struct Reference	527
8.467.1 Detailed Description	528
8.467.2 Field Documentation	529
8.467.2.1 pAddCDMASysInfo	529
8.467.2.2 pAddGSMSysInfo	530
8.467.2.3 pAddHDRSysInfo	530
8.467.2.4 pAddLTESysInfo	530
8.467.2.5 pAddWCDMASysInfo	530
8.467.2.6 pCDMASrvStatusInfo	530
8.467.2.7 pCDMASysInfo	530
8.467.2.8 pGSMCallBarringSysInfo	530
8.467.2.9 pGSMCipherDomainSysInfo	530
8.467.2.10pGSMSrvStatusInfo	530
8.467.2.11pGSMSysInfo	530
8.467.2.12pHDRSrvStatusInfo	530
8.467.2.13pHDRSysInfo	530
8.467.2.14pLTESrvStatusInfo	530
8.467.2.15pLTESysInfo	530
8.467.2.16pLTEVoiceSupportSysInfo	530
8.467.2.17pSysInfoNoChange	530
8.467.2.18pWCDMACallBarringSysInfo	530
8.467.2.19pWCDMACipherDomainSysInfo	530
8.467.2.20pWCDMASrvStatusInfo	530
8.467.2.21pWCDMASysInfo	530
8.468NASTimeInfoTlv Struct Reference	530
8.468.1 Field Documentation	531

8.468.1.1 time	531
8.468.1.2 TlvPresent	531
8.469nasTimers Struct Reference	531
8.469.1 Detailed Description	531
8.469.2 Field Documentation	531
8.469.2.1 t3396_apn	531
8.469.2.2 t3396_plmn_id	531
8.469.2.3 t3396_val	531
8.470netSelectionPref Struct Reference	531
8.470.1 Detailed Description	532
8.470.2 Field Documentation	532
8.470.2.1 mcc	532
8.470.2.2 mnc	532
8.470.2.3 netReg	532
8.471NetStats Struct Reference	532
8.471.1 Detailed Description	532
8.471.2 Field Documentation	533
8.471.2.1 rx_bytes	533
8.471.2.2 rx_errors	533
8.471.2.3 rx_overflows	533
8.471.2.4 rx_packets	533
8.471.2.5 tx_bytes	533
8.471.2.6 tx_errors	533
8.471.2.7 tx_overflows	533
8.471.2.8 tx_packets	533
8.472NetworkDebugResp Struct Reference	533
8.472.1 Detailed Description	534
8.472.2 Field Documentation	534
8.472.2.1 pDataStatusDetail	534
8.472.2.2 pDeviceConfigDetail	534

8.472.2.3 pNetworkStat1x	534
8.472.2.4 pNetworkStatEVDO	534
8.472.2.5 pObjectVer	534
8.473NetworkStat1x Struct Reference	534
8.473.1 Detailed Description	535
8.473.2 Field Documentation	536
8.473.2.1 ActSetCnt	536
8.473.2.2 NeighborSetCnt	536
8.473.2.3 pActPilotPNElements	536
8.473.2.4 pNeighborSetPilotPN	536
8.473.2.5 RX_EC_IO	536
8.473.2.6 RX_PWR	536
8.473.2.7 SO	536
8.473.2.8 State	536
8.473.2.9 TX_PWR	536
8.474NetworkStatEVDO Struct Reference	536
8.474.1 Detailed Description	537
8.474.2 Field Documentation	538
8.474.2.1 MACIndex	538
8.474.2.2 PER	538
8.474.2.3 PilotEnergy	538
8.474.2.4 pSectorID	538
8.474.2.5 RX_PWR	538
8.474.2.6 SectorIDLen	538
8.474.2.7 SNR	538
8.474.2.8 State	538
8.475newMTMessageTlv Struct Reference	538
8.475.1 Detailed Description	538
8.475.2 Field Documentation	538
8.475.2.1 MTMessageInfo	538

8.475.2.2 TlvPresent	538
8.476newPwdData Struct Reference	538
8.476.1 Detailed Description	539
8.476.2 Field Documentation	539
8.476.2.1 newPwd	539
8.476.2.2 newPwdAgain	539
8.477nmrCellInfo Struct Reference	539
8.477.1 Detailed Description	539
8.477.2 Field Documentation	540
8.477.2.1 nmrArfcn	540
8.477.2.2 nmrBsic	540
8.477.2.3 nmrCellID	540
8.477.2.4 nmrLac	540
8.477.2.5 nmrPlmn	540
8.477.2.6 nmrRxLev	540
8.478NSSAudioCtrl Struct Reference	541
8.478.1 Detailed Description	541
8.478.2 Field Documentation	541
8.478.2.1 downLink	541
8.478.2.2 upLink	541
8.479NWProfile Struct Reference	541
8.479.1 Detailed Description	541
8.479.2 Field Documentation	542
8.479.2.1 pProfSz	542
8.479.2.2 pProfValues	542
8.479.2.3 tech	542
8.480omaDmConfigTlv Struct Reference	542
8.480.1 Detailed Description	542
8.480.2 Field Documentation	542
8.480.2.1 alertmsg	542

8.480.2.2 alertmsglength	543
8.480.2.3 state	543
8.480.2.4 userInputReq	543
8.480.2.5 userInputTimeout	543
8.481omaDmConfigTlvExt Struct Reference	543
8.481.1 Detailed Description	543
8.481.2 Field Documentation	545
8.481.2.1 alertmsg	545
8.481.2.2 alertmsglength	545
8.481.2.3 state	545
8.481.2.4 userInputReq	545
8.481.2.5 userInputTimeout	545
8.482omaDmFotaTlv Struct Reference	545
8.482.1 Detailed Description	545
8.482.2 Field Documentation	547
8.482.2.1 description	547
8.482.2.2 descriptionlength	547
8.482.2.3 fwdloadsize	547
8.482.2.4 fwloadComplete	547
8.482.2.5 namelength	547
8.482.2.6 package_name	547
8.482.2.7 sessionType	547
8.482.2.8 severity	547
8.482.2.9 state	547
8.482.2.10updateCompleteStatus	547
8.482.2.11userInputReq	547
8.482.2.12userInputTimeout	547
8.482.2.13version	547
8.482.2.14versionlength	547
8.483omaDmFotaTlvExt Struct Reference	547

8.483.1 Detailed Description	547
8.483.2 Field Documentation	549
8.483.2.1 description	549
8.483.2.2 descriptionlength	549
8.483.2.3 fumoResultCode	549
8.483.2.4 namelength	549
8.483.2.5 package_name	549
8.483.2.6 packageSize	549
8.483.2.7 receivedBytes	549
8.483.2.8 reserved	549
8.483.2.9 state	549
8.483.2.10userInputTimeout	549
8.483.2.11version	549
8.483.2.12versionlength	549
8.484omaDmNotificationsTlv Struct Reference	549
8.484.1 Field Documentation	549
8.484.1.1 notification	549
8.484.1.2 sessionStatus	549
8.485operatorNameString Struct Reference	549
8.485.1 Detailed Description	550
8.485.2 Field Documentation	550
8.485.2.1 PLMNName	550
8.486OperatorPLMNData Struct Reference	550
8.486.1 Detailed Description	550
8.486.2 Field Documentation	551
8.486.2.1 lac1	551
8.486.2.2 lac2	551
8.486.2.3 mcc	551
8.486.2.4 mnc	551
8.486.2.5 PLMNRecID	551

8.487operatorPLMNList Struct Reference	551
8.487.1 Detailed Description	551
8.487.2 Field Documentation	551
8.487.2.1 numInstance	551
8.487.2.2 PLMNData	551
8.488pack_dms_ActivateAutomatic_t Struct Reference	551
8.488.1 Detailed Description	552
8.488.2 Field Documentation	552
8.488.2.1 actCode	552
8.489pack_dms_GetCustFeaturesV2_t Struct Reference	552
8.489.1 Detailed Description	552
8.489.2 Field Documentation	552
8.489.2.1 cust_id	552
8.489.2.2 list_type	552
8.489.2.3 Tlvresult	552
8.490pack_dms_ResetToFactoryDefaults_t Struct Reference	553
8.490.1 Detailed Description	553
8.490.2 Field Documentation	553
8.490.2.1 spc	553
8.491pack_dms_SetActivationStatusCallback_t Struct Reference	553
8.491.1 Detailed Description	553
8.491.2 Field Documentation	553
8.491.2.1 activationState	553
8.492pack_dms_SetCrashAction_t Struct Reference	553
8.492.1 Detailed Description	554
8.492.2 Field Documentation	554
8.492.2.1 crashAction	554
8.493pack_dms_SetCustFeature_t Struct Reference	554
8.493.1 Field Documentation	554
8.493.1.1 DHCPRelayEnabled	554

8.493.1.2 DisableIMSI	554
8.493.1.3 GpsEnable	554
8.493.1.4 GPSPMP	554
8.493.1.5 GPSSel	555
8.493.1.6 IPFamSupport	555
8.493.1.7 IsVoiceEnabled	555
8.493.1.8 RMAutoConnect	555
8.493.1.9 SMSSupport	555
8.494pack_dms_SetCustFeaturesV2_t Struct Reference	555
8.494.1 Detailed Description	555
8.494.2 Field Documentation	555
8.494.2.1 cust_id	555
8.494.2.2 cust_value	555
8.494.2.3 Tlvresult	555
8.494.2.4 value_length	556
8.495pack_dms_SetEventReport_t Struct Reference	556
8.495.1 Field Documentation	556
8.495.1.1 mode	556
8.496pack_dms_SetPower_t Struct Reference	556
8.496.1 Field Documentation	556
8.496.1.1 mode	556
8.496.1.2 Tlvresult	556
8.497pack_dms_SetUSBComp_t Struct Reference	556
8.497.1 Field Documentation	556
8.497.1.1 Tlvresult	556
8.497.1.2 USBComp	556
8.498pack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference	556
8.498.1 Detailed Description	556
8.498.2 Field Documentation	557
8.498.2.1 resetInfoInd	557

8.499	pack_dms_SLQSSwiGetCrashInfo_t Struct Reference	557
8.499.1	Detailed Description	557
8.499.2	Field Documentation	557
8.499.2.1	clear	557
8.500	pack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference	557
8.500.1	Detailed Description	557
8.500.2	Field Documentation	558
8.500.2.1	pDestSMSContent	558
8.500.2.2	pDestSMSNum	558
8.501	pack_dms_SLQSSwiSetHostDevInfo_t Struct Reference	558
8.501.1	Detailed Description	558
8.501.2	Field Documentation	558
8.501.2.1	manString	558
8.501.2.2	modelString	559
8.501.2.3	plasmaIDString	559
8.501.2.4	swVerString	559
8.502	pack_dms_SLQSSwiSetOSInfo_t Struct Reference	559
8.502.1	Detailed Description	559
8.502.2	Field Documentation	559
8.502.2.1	nameString	559
8.502.2.2	versionString	559
8.503	pack_dms_UIMChangePIN_t Struct Reference	559
8.503.1	Detailed Description	559
8.503.2	Field Documentation	560
8.503.2.1	id	560
8.503.2.2	newValue	560
8.503.2.3	oldValue	560
8.504	pack_dms_UIMGetControlKeyStatus_t Struct Reference	560
8.504.1	Detailed Description	560
8.504.2	Field Documentation	560

8.504.2.1 facility	560
8.505pack_dms_UIMGetICCID_t Struct Reference	561
8.505.1 Detailed Description	561
8.505.2 Field Documentation	561
8.505.2.1 Tlvresult	561
8.506pack_dms_UIMSetControlKeyProtection_t Struct Reference	561
8.506.1 Detailed Description	561
8.506.2 Field Documentation	562
8.506.2.1 facility	562
8.506.2.2 facilityCk	562
8.506.2.3 facilityState	562
8.507pack_dms_UIMSetPINProtection_t Struct Reference	562
8.507.1 Detailed Description	562
8.507.2 Field Documentation	562
8.507.2.1 bEnable	562
8.507.2.2 id	562
8.507.2.3 value	562
8.508pack_dms_UIMUnblockControlKey_t Struct Reference	563
8.508.1 Detailed Description	563
8.508.2 Field Documentation	563
8.508.2.1 facility	563
8.508.2.2 facilityCk	563
8.509pack_dms_UIMUnblockPIN_t Struct Reference	563
8.509.1 Detailed Description	563
8.509.2 Field Documentation	564
8.509.2.1 id	564
8.509.2.2 newPin	564
8.509.2.3 pukValue	564
8.510pack_dms_UIMVerifyPIN_t Struct Reference	564
8.510.1 Detailed Description	564

8.510.2 Field Documentation	564
8.510.2.1 id	564
8.510.2.2 value	565
8.511 pack_fms_GetImagesPreference_t Struct Reference	565
8.511.1 Detailed Description	565
8.511.2 Field Documentation	565
8.511.2.1 Tlvresult	565
8.512 pack_fms_GetStoredImages_t Struct Reference	565
8.512.1 Detailed Description	565
8.512.2 Field Documentation	565
8.512.2.1 Tlvresult	565
8.513 pack_fms_SetImagesPreference_t Struct Reference	565
8.513.1 Detailed Description	566
8.513.2 Field Documentation	566
8.513.2.1 bForceDownload	566
8.513.2.2 imageListSize	566
8.513.2.3 modemindex	566
8.513.2.4 pImageList	566
8.513.2.5 Tlvresult	566
8.514 pack_loc_Delete_Assist_Data_t Struct Reference	566
8.514.1 Detailed Description	567
8.514.2 Field Documentation	567
8.514.2.1 pBdsSVInfo	567
8.514.2.2 pCellDb	567
8.514.2.3 pCikInfo	567
8.514.2.4 pGnssData	567
8.514.2.5 pSVInfo	567
8.514.2.6 Tlvresult	567
8.515 pack_loc_EventRegister_t Struct Reference	567
8.515.1 Detailed Description	568

8.515.2 Field Documentation	570
8.515.2.1 eventRegister	570
8.515.2.2 Tlvresult	570
8.516pack_loc_SetExtPowerState_t Struct Reference	570
8.516.1 Detailed Description	570
8.516.2 Field Documentation	570
8.516.2.1 extPowerState	570
8.516.2.2 Tlvresult	570
8.517pack_loc_SetOperationMode_t Struct Reference	571
8.517.1 Detailed Description	571
8.517.2 Field Documentation	571
8.517.2.1 mode	571
8.517.2.2 Tlvresult	571
8.518pack_loc_SLQSLOCGetBestAvailPos_t Struct Reference	571
8.518.1 Detailed Description	571
8.518.2 Field Documentation	572
8.518.2.1 Tlvresult	572
8.518.2.2 xid	572
8.519pack_loc_SLQSLOCInjectPosition_t Struct Reference	572
8.519.1 Detailed Description	573
8.519.2 Field Documentation	575
8.519.2.1 altitudeSrcInfo	575
8.519.2.2 altitudeWrtEllipsoid	576
8.519.2.3 altitudeWrtMeanSeaLevel	576
8.519.2.4 has_altitudeSrcInfo	576
8.519.2.5 has_altitudeWrtEllipsoid	576
8.519.2.6 has_altitudeWrtMeanSeaLevel	576
8.519.2.7 has_horConfidence	576
8.519.2.8 has_horReliability	576
8.519.2.9 has_horUncCircular	576

8.519.2.10has_latitude	576
8.519.2.11has_longitude	576
8.519.2.12has_positionSrc	576
8.519.2.13has_rawHorConfidence	576
8.519.2.14has_rawHorUncCircular	576
8.519.2.15has_timestampAge	576
8.519.2.16has_timestampUtc	576
8.519.2.17has_vertConfidence	576
8.519.2.18has_vertRelicability	576
8.519.2.19has_vertUnc	576
8.519.2.20horConfidence	576
8.519.2.21horReliability	576
8.519.2.22horUncCircular	576
8.519.2.23atitude	576
8.519.2.24ongitude	576
8.519.2.25positionSrc	577
8.519.2.26rawHorConfidence	577
8.519.2.27rawHorUncCircular	577
8.519.2.28imestampAge	577
8.519.2.29imestampUtc	577
8.519.2.30vertConfidence	577
8.519.2.31vertReliability	577
8.519.2.32vertUnc	577
8.520pack_loc_SLQSLOCInjectSensorData_t Struct Reference	577
8.520.1 Detailed Description	577
8.520.2 Field Documentation	578
8.520.2.1 acceleroData	578
8.520.2.2 acceleroTimeSrc	578
8.520.2.3 accelTemp	578
8.520.2.4 gyroData	579

8.520.2.5 gyroTemp	579
8.520.2.6 gyroTimeSrc	579
8.520.2.7 has_acceleroTimeSrc	579
8.520.2.8 has_accelTemp	579
8.520.2.9 has_acceleroData	579
8.520.2.10has_gyroData	579
8.520.2.11has_gyroTemp	579
8.520.2.12has_gyroTimeSrc	579
8.520.2.13has_opaqueId	579
8.520.2.14opaqueId	579
8.521pack_loc_SLQSLOCInjectUTCTime_t Struct Reference	579
8.521.1 Detailed Description	579
8.521.2 Field Documentation	579
8.521.2.1 timeMsec	579
8.521.2.2 timeUncMsec	579
8.522pack_loc_SLQSLOCSetCradleMountConfig_t Struct Reference	580
8.522.1 Detailed Description	580
8.522.2 Field Documentation	580
8.522.2.1 confidence	580
8.522.2.2 has_confidence	580
8.522.2.3 state	580
8.523pack_loc_Start_t Struct Reference	580
8.523.1 Detailed Description	581
8.523.2 Field Documentation	581
8.523.2.1 pApplicationInfo	581
8.523.2.2 pConfigAltitudeAssumed	582
8.523.2.3 pHorizontalAccuracyLvl	582
8.523.2.4 pIntermediateReportState	582
8.523.2.5 pMinIntervalTime	582
8.523.2.6 pRecurrenceType	582

8.523.2.7 SessionId	582
8.523.2.8 Tlvresult	582
8.524pack_loc_Stop_t Struct Reference	582
8.524.1 Detailed Description	582
8.524.2 Field Documentation	582
8.524.2.1 SessionId	582
8.524.2.2 Tlvresult	582
8.525pack_nas_SetACCOLC_t Struct Reference	582
8.525.1 Detailed Description	583
8.525.2 Field Documentation	583
8.525.2.1 accolc	583
8.525.2.2 spc	583
8.526pack_nas_SetNetworkPreference_t Struct Reference	583
8.526.1 Detailed Description	583
8.526.2 Field Documentation	584
8.526.2.1 Duration	584
8.526.2.2 TechnologyPref	584
8.526.2.3 Tlvresult	584
8.527pack_nas_SLQSGetPLMNName_t Struct Reference	584
8.527.1 Detailed Description	584
8.527.2 Field Documentation	584
8.527.2.1 mcc	584
8.527.2.2 mnc	584
8.527.2.3 pMncPcsStatus	585
8.528pack_nas_SLQSIInitiateNetworkRegistration_t Struct Reference	585
8.528.1 Detailed Description	585
8.528.2 Field Documentation	585
8.528.2.1 pChangeDuration	585
8.528.2.2 pMncPcsDigitStatus	585
8.528.2.3 pMNRInfo	585

8.528.2.4 regAction	586
8.529pack_nas_SLQSNasConfigSigInfo2_t Struct Reference	586
8.529.1 Detailed Description	586
8.529.2 Field Documentation	589
8.529.2.1 pCDMAECIODelta	589
8.529.2.2 pCDMAECIOThresh	589
8.529.2.3 pCDMARSSIDelta	589
8.529.2.4 pCDMARSSIThresh	589
8.529.2.5 pGSMRSSIDelta	589
8.529.2.6 pGSMRSSIThresh	589
8.529.2.7 pHDRCIDelta	589
8.529.2.8 pHDRCIOThresh	589
8.529.2.9 pHDRIDelta	589
8.529.2.10pHDRIOThresh	589
8.529.2.11pHDRRSSIDelta	589
8.529.2.12pHDRRSSIThresh	589
8.529.2.13pHDRSINRDelta	589
8.529.2.14pHDRSINRThresh	590
8.529.2.15pLTERSRPDelta	590
8.529.2.16pLTERSRPThresh	590
8.529.2.17pLTERSRQDelta	590
8.529.2.18pLTERSRQThresh	590
8.529.2.19pLTERSSIDelta	590
8.529.2.20pLTERSSIThresh	590
8.529.2.21pLTESigRptConfig	590
8.529.2.22pLTESNRDelta	590
8.529.2.23pLTESNRThresh	590
8.529.2.24pTDSCDMAECIODelta	590
8.529.2.25pTDSCDMAECIOThresh	590
8.529.2.26pTDSCDMARSCPDelta	590

8.529.2.27	pTDSCDMARSCPTresh	590
8.529.2.28	pTDSCDMARSSIDelta	590
8.529.2.29	pTDSCDMARSSITresh	590
8.529.2.30	pTDSCDMASINRDelta	590
8.529.2.31	pTDSCDMASINRTresh	590
8.529.2.32	pWCDMAECIODelta	590
8.529.2.33	pWCDMAECIOTresh	590
8.529.2.34	pWCDMARSSIDelta	590
8.529.2.35	pWCDMARSSITresh	590
8.530	pack_nas_SLQSNasIndicationRegisterExt_t Struct Reference	590
8.530.1	Detailed Description	591
8.530.2	Field Documentation	593
8.530.2.1	pDDTMInd	593
8.530.2.2	pDualStandByPrefInd	593
8.530.2.3	pErrorRateInd	593
8.530.2.4	pHDRNewUATIAssInd	593
8.530.2.5	pHDRSessionCloseInd	593
8.530.2.6	pLTECphyCa	593
8.530.2.7	pManagedRoamingInd	593
8.530.2.8	pNetworkTimeInd	593
8.530.2.9	pServingSystemInd	593
8.530.2.10	pSignalStrengthInd	593
8.530.2.11	pSubscriptionInfoInd	593
8.530.2.12	pSysInfoInd	593
8.530.2.13	pSystemSelectionInd	593
8.531	pack_nas_SLQSNasSwiIndicationRegister_t Struct Reference	593
8.531.1	Detailed Description	593
8.531.2	Field Documentation	594
8.531.2.1	gsmUmtsDI	594
8.531.2.2	gsmUmtsUI	594

8.531.2.3 lteEmmDI	594
8.531.2.4 lteEmmUI	594
8.531.2.5 lteEsmDI	594
8.531.2.6 lteEsmUI	594
8.531.2.7 pRankIndicatorInd	594
8.531.2.8 pTimer	595
8.532pack_nas_SLQSSetSignalStrengthsCallback_t Struct Reference	595
8.532.1 Detailed Description	595
8.532.2 Field Documentation	595
8.532.2.1 bEnable	595
8.532.2.2 pSigIndReq	595
8.533pack_nas_SLQSSetSysSelectionPref_t Struct Reference	595
8.533.1 Detailed Description	595
8.533.2 Field Documentation	599
8.533.2.1 pAcqOrderPref	599
8.533.2.2 pBandPref	599
8.533.2.3 pChgDuration	599
8.533.2.4 pCSGID	599
8.533.2.5 pEmerMode	599
8.533.2.6 pGWAcqOrderPref	600
8.533.2.7 pLTEBandPref	600
8.533.2.8 pMNCIncPCSDigStat	600
8.533.2.9 pModePref	600
8.533.2.10pNetSelPref	600
8.533.2.11pPRLPref	600
8.533.2.12pRAT	600
8.533.2.13pRoamPref	600
8.533.2.14pSrvDomainPref	600
8.533.2.15pSrvRegRestriction	600
8.533.2.16pTdsdmaBandPref	600

8.534	pack_qmi_t Struct Reference	600
8.534.1	Detailed Description	600
8.534.2	Field Documentation	600
8.534.2.1	msgid	600
8.534.2.2	svc	601
8.534.2.3	timeout	601
8.534.2.4	xid	601
8.535	pack_qos_SLQSQoSWiReadApnExtraParams_t Struct Reference	601
8.535.1	Detailed Description	601
8.535.2	Field Documentation	601
8.535.2.1	apnId	601
8.536	pack_qos_SLQSQoSWiReadDataStats_t Struct Reference	601
8.536.1	Detailed Description	601
8.536.2	Field Documentation	601
8.536.2.1	apnId	602
8.537	pack_qos_SLQSSetQoSEventCallback_t Struct Reference	602
8.537.1	Detailed Description	602
8.537.2	Field Documentation	602
8.537.2.1	enable	602
8.538	pack_sms_SendSMS_t Struct Reference	602
8.538.1	Detailed Description	602
8.538.2	Field Documentation	603
8.538.2.1	messageFormat	603
8.538.2.2	messageSize	603
8.538.2.3	pLinktimer	603
8.538.2.4	pMessage	603
8.539	pack_sms_SetNewSMSCallback_t Struct Reference	603
8.539.1	Detailed Description	603
8.539.2	Field Documentation	603
8.539.2.1	status	603

8.540	pack_sms_SLQSDDeleteSMS_t Struct Reference	603
8.540.1	Detailed Description	603
8.540.2	Field Documentation	604
8.540.2.1	pMessageIndex	604
8.540.2.2	pMessageMode	604
8.540.2.3	pMessageTag	604
8.540.2.4	storageType	604
8.541	pack_sms_SLQSGetSMS_t Struct Reference	604
8.541.1	Detailed Description	604
8.541.2	Field Documentation	605
8.541.2.1	messageIndex	605
8.541.2.2	pMessageMode	605
8.541.2.3	storageType	605
8.542	pack_sms_SLQSGetSMSList_t Struct Reference	605
8.542.1	Detailed Description	605
8.542.2	Field Documentation	605
8.542.2.1	pMessageMode	605
8.542.2.2	pRequestedTag	605
8.542.2.3	storageType	605
8.543	pack_sms_SLQSModifySMSStatus_t Struct Reference	605
8.543.1	Detailed Description	606
8.543.2	Field Documentation	606
8.543.2.1	messageIndex	606
8.543.2.2	messageTag	606
8.543.2.3	pMessageMode	606
8.543.2.4	storageType	606
8.544	pack_swiloc_SwiLocSetAutoStart_t Struct Reference	606
8.544.1	Detailed Description	607
8.544.2	Field Documentation	608
8.544.2.1	fix_rate	608

8.544.2.2 fix_type	608
8.544.2.3 function	608
8.544.2.4 max_dist	608
8.544.2.5 max_time	608
8.544.2.6 set_fix_rate	608
8.544.2.7 set_fix_type	608
8.544.2.8 set_function	608
8.544.2.9 set_max_dist	608
8.544.2.10set_max_time	608
8.545pack_swioama_SLQSOMADMCancelSession_t Struct Reference	608
8.545.1 Detailed Description	608
8.545.2 Field Documentation	608
8.545.2.1 sessionType	608
8.546pack_swioama_SLQSOMADMGetSessionInfo_t Struct Reference	609
8.546.1 Detailed Description	609
8.546.2 Field Documentation	609
8.546.2.1 SessionType	609
8.547pack_swioama_SLQSOMADMSendSelection_t Struct Reference	609
8.547.1 Detailed Description	609
8.547.2 Field Documentation	610
8.547.2.1 pDeferTime	610
8.547.2.2 pRejectReason	610
8.547.2.3 selection	610
8.548pack_swioama_SLQSOMADMSetSettings_t Struct Reference	610
8.548.1 Detailed Description	610
8.548.2 Field Documentation	610
8.548.2.1 FOTAdownload	611
8.548.2.2 FOTAUpdate	611
8.548.2.3 pAutosdm	611
8.548.2.4 pFwAutoCheck	611

8.549	pack_swima_SLQSOMADMStartSession_t Struct Reference	611
8.549.1	Detailed Description	611
8.549.2	Field Documentation	611
8.549.2.1	sessionType	611
8.550	pack_uim_ChangePin_t Struct Reference	611
8.550.1	Detailed Description	611
8.550.2	Field Documentation	612
8.550.2.1	changePIN	612
8.550.2.2	EncryptedPIN1	612
8.550.2.3	pIndicationToken	612
8.550.2.4	pKeyReferenceID	612
8.550.2.5	sessionInfo	612
8.550.2.6	Tlvresult	612
8.551	pack_uim_ReadTransparent_t Struct Reference	612
8.551.1	Detailed Description	613
8.551.2	Field Documentation	613
8.551.2.1	fileIndex	613
8.551.2.2	pEncryptData	613
8.551.2.3	pIndicationToken	613
8.551.2.4	readTransparent	613
8.551.2.5	sessionInfo	613
8.551.2.6	Tlvresult	613
8.552	pack_uim_SetPinProtection_t Struct Reference	613
8.552.1	Detailed Description	614
8.552.2	Field Documentation	614
8.552.2.1	EncryptedPIN1	614
8.552.2.2	pIndicationToken	614
8.552.2.3	pinProtection	614
8.552.2.4	pKeyReferenceID	614
8.552.2.5	sessionInfo	614

8.552.2.6 Tlvresult	614
8.553pack_uim_SLQSUIEventRegister_t Struct Reference	614
8.553.1 Detailed Description	615
8.553.2 Field Documentation	615
8.553.2.1 eventMask	615
8.554pack_uim_SLQSUIPowerDown_t Struct Reference	615
8.554.1 Detailed Description	615
8.554.2 Field Documentation	615
8.554.2.1 slot	615
8.555pack_uim_SLQSUIPowerUp_t Struct Reference	615
8.555.1 Detailed Description	615
8.555.2 Field Documentation	616
8.555.2.1 plgnoreHotSwapSwitch	616
8.555.2.2 slot	616
8.556pack_uim_SLQSUISwitchSlot_t Struct Reference	616
8.556.1 Detailed Description	616
8.556.2 Field Documentation	617
8.556.2.1 bLogicalSlot	617
8.556.2.2 ulPhysicalSlot	617
8.557pack_uim_UnblockPin_t Struct Reference	617
8.557.1 Detailed Description	617
8.557.2 Field Documentation	617
8.557.2.1 EncryptedPIN1	618
8.557.2.2 pIndicationToken	618
8.557.2.3 pinProtection	618
8.557.2.4 pKeyReferenceID	618
8.557.2.5 sessionInfo	618
8.557.2.6 Tlvresult	618
8.558pack_uim_VerifyPin_t Struct Reference	618
8.558.1 Detailed Description	618

8.558.2 Field Documentation	619
8.558.2.1 pEncryptedPIN1	619
8.558.2.2 pIndicationToken	619
8.558.2.3 pKeyReferenceID	619
8.558.2.4 sessionInfo	619
8.558.2.5 Tlvresult	619
8.558.2.6 verifyPIN	619
8.559pack_wds_DHCPv4ClientLeaseChange_t Struct Reference	619
8.559.1 Detailed Description	619
8.559.2 Field Documentation	619
8.559.2.1 pEnableNotification	619
8.560pack_wds_GetDefaultProfile_t Struct Reference	619
8.560.1 Detailed Description	620
8.560.2 Field Documentation	620
8.560.2.1 profiletype	620
8.561pack_wds_GetDefaultProfileNum_t Struct Reference	620
8.561.1 Detailed Description	620
8.561.2 Field Documentation	620
8.561.2.1 family	620
8.561.2.2 type	620
8.562pack_wds_GetDormancyState_t Struct Reference	620
8.563pack_wds_GetLastMobileIPError_t Struct Reference	620
8.564pack_wds_GetMobileIP_t Struct Reference	620
8.565pack_wds_GetMobileIPProfile_t Struct Reference	620
8.565.1 Detailed Description	621
8.565.2 Field Documentation	621
8.565.2.1 index	621
8.566pack_wds_GetPacketStatistics_t Struct Reference	621
8.566.1 Detailed Description	621
8.566.2 Field Documentation	621

8.566.2.1 pStatMask	621
8.567pack_wds_GetPacketStatus_t Struct Reference	621
8.567.1 Detailed Description	621
8.567.2 Field Documentation	621
8.567.2.1 statmask	621
8.568pack_wds_GetSessionDuration_t Struct Reference	622
8.569pack_wds_RMSetTransferStatistics_t Struct Reference	622
8.569.1 Detailed Description	622
8.569.2 Field Documentation	622
8.569.2.1 RmTrasnferStaticsReq	622
8.570pack_wds_SetAutoconnect_t Struct Reference	622
8.570.1 Detailed Description	622
8.570.2 Field Documentation	622
8.570.2.1 acroamsetting	622
8.570.2.2 acsetting	622
8.571pack_wds_SetDefaultProfile_t Struct Reference	623
8.571.1 Detailed Description	623
8.571.2 Field Documentation	623
8.571.2.1 authentication	623
8.571.2.2 ipAddress	623
8.571.2.3 pApnname	623
8.571.2.4 pdpType	623
8.571.2.5 pName	623
8.571.2.6 pPassword	623
8.571.2.7 primaryDNS	623
8.571.2.8 profileType	623
8.571.2.9 pUsername	624
8.571.2.10secondaryDNS	624
8.572pack_wds_SetDefaultProfileNum_t Struct Reference	624
8.572.1 Field Documentation	624

8.572.1.1 family	624
8.572.1.2 index	624
8.572.1.3 type	624
8.573pack_wds_SetMobileIP_t Struct Reference	624
8.573.1 Detailed Description	624
8.573.2 Field Documentation	624
8.573.2.1 mode	624
8.574pack_wds_SetMobileIPParameters_t Struct Reference	624
8.574.1 Detailed Description	625
8.574.2 Field Documentation	625
8.574.2.1 pHA2002bis	626
8.574.2.2 pHAAuthenticator	626
8.574.2.3 pMode	626
8.574.2.4 pReRegPeriod	626
8.574.2.5 pReRegTraffic	626
8.574.2.6 pRetryInterval	626
8.574.2.7 pRetryLimit	626
8.574.2.8 pSPC	626
8.575pack_wds_SetMobileIPProfile_t Struct Reference	626
8.575.1 Detailed Description	626
8.575.2 Field Documentation	627
8.575.2.1 index	627
8.575.2.2 pAAASPI	627
8.575.2.3 pAddress	627
8.575.2.4 pEnabled	627
8.575.2.5 pHASPI	627
8.575.2.6 pMNAAA	627
8.575.2.7 pMNHA	627
8.575.2.8 pNAI	627
8.575.2.9 pPrimaryHA	627

8.575.2.10pRevTunneling	627
8.575.2.11pSecondaryHA	627
8.575.2.12spc	627
8.576pack_wds_SLQSCreateProfile_t Struct Reference	627
8.576.1 Detailed Description	627
8.576.2 Field Documentation	628
8.576.2.1 pCurProfile	628
8.576.2.2 pProfileId	628
8.576.2.3 pProfileType	628
8.577pack_wds_SLQSDeleteProfile_t Struct Reference	628
8.577.1 Detailed Description	628
8.577.2 Field Documentation	628
8.577.2.1 profileIndex	628
8.577.2.2 profileType	628
8.578pack_wds_SLQSGetCurrDataSystemStat_t Struct Reference	628
8.579pack_wds_SLQSGetDataBearerTechnology_t Struct Reference	628
8.580pack_wds_SLQSGetDUNCallInfo_t Struct Reference	628
8.580.1 Detailed Description	629
8.580.2 Field Documentation	629
8.580.2.1 Mask	629
8.580.2.2 pReportChannelRate	629
8.580.2.3 pReportConnStatus	629
8.580.2.4 pReportDataBearerTech	629
8.580.2.5 pReportDormStatus	629
8.580.2.6 pTransferStatInd	629
8.581pack_wds_SLQSGetProfileSettings_t Struct Reference	629
8.581.1 Detailed Description	629
8.581.2 Field Documentation	630
8.581.2.1 ProfileId	630
8.581.2.2 ProfileType	630

8.582	pack_wds_SLQSGetRuntimeSettings_t Struct Reference	630
8.582.1	Detailed Description	630
8.582.2	Field Documentation	631
8.582.2.1	pReqSettings	631
8.583	pack_wds_SLQSModifyProfile_t Struct Reference	631
8.583.1	Detailed Description	631
8.583.2	Field Documentation	632
8.583.2.1	curProfile	632
8.583.2.2	pProfileId	632
8.583.2.3	pProfileType	632
8.584	pack_wds_SLQSSet3GPPConfigItem_t Struct Reference	632
8.584.1	Detailed Description	632
8.584.2	Field Documentation	633
8.584.2.1	LTEAttachProfileListLen	633
8.584.2.2	p3gppRelease	634
8.584.2.3	pDefaultPDNEnabled	634
8.584.2.4	pLTEAttachProfile	634
8.584.2.5	pLTEAttachProfileList	634
8.584.2.6	pProfileList	634
8.585	pack_wds_SLQSSetIPFamilyPreference_t Struct Reference	634
8.585.1	Detailed Description	634
8.585.2	Field Documentation	634
8.585.2.1	IPFamilyPreference	634
8.586	pack_wds_SLQSSetWdsEventCallback_t Struct Reference	634
8.586.1	Detailed Description	634
8.586.2	Field Documentation	635
8.586.2.1	currentDataBearer	635
8.586.2.2	dataBearer	635
8.586.2.3	dataSystemStatus	635
8.586.2.4	dormancyStatus	635

8.586.2.5 interval	635
8.586.2.6 mobileIP	635
8.586.2.7 transferStats	635
8.587pack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference	635
8.587.1 Detailed Description	635
8.587.2 Field Documentation	635
8.587.2.1 pProfileId	635
8.588pack_wds_SLQSSSetDHCPv4ClientConfig_t Struct Reference	635
8.588.1 Detailed Description	636
8.588.2 Field Documentation	636
8.588.2.1 pHwConfig	636
8.588.2.2 pProfileId	636
8.588.2.3 pRequestOptionList	636
8.589pack_wds_SLQSSSetLoopback_t Struct Reference	636
8.589.1 Detailed Description	636
8.589.2 Field Documentation	637
8.589.2.1 loopbackMode	637
8.589.2.2 loopbackMultiplier	637
8.590pack_wds_SLQSSStartDataSession_t Struct Reference	637
8.590.1 Detailed Description	637
8.590.2 Field Documentation	638
8.590.2.1 pAuth	638
8.590.2.2 pPass	638
8.590.2.3 pprofileid3gpp	638
8.590.2.4 pprofileid3gpp2	638
8.590.2.5 pTech	638
8.590.2.6 pUser	638
8.591pack_wds_SLQSSStopDataSession_t Struct Reference	638
8.591.1 Detailed Description	638
8.591.2 Field Documentation	638

8.591.2.1 psid	638
8.592pack_wds_SLQSWdsSetEventReport_t Struct Reference	638
8.592.1 Detailed Description	639
8.592.2 Field Documentation	640
8.592.2.1 pCurrChannelRateInd	640
8.592.2.2 pCurrDataBearerTechInd	640
8.592.2.3 pCurrPrefDataSysInd	640
8.592.2.4 pDataBearerTechInd	640
8.592.2.5 pDataCallStatusChangeInd	640
8.592.2.6 pDataSystemStatusChangeInd	640
8.592.2.7 pDormancyStatusInd	640
8.592.2.8 pEVDOPageMonPerChangeInd	640
8.592.2.9 pMIPStatusInd	640
8.592.2.10pTransferStatInd	640
8.593pack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference	640
8.593.1 Detailed Description	640
8.593.2 Field Documentation	640
8.593.2.1 contextId	640
8.593.2.2 contextType	640
8.594PackCreateProfileOut Struct Reference	641
8.594.1 Field Documentation	641
8.594.1.1 ExtErrorCode	641
8.594.1.2 ProfileIndex	641
8.594.1.3 ProfileType	641
8.595packgetDyingGaspCfg Struct Reference	641
8.595.1 Detailed Description	641
8.595.2 Field Documentation	641
8.595.2.1 pDestSMSContent	641
8.595.2.2 pDestSMSNum	641
8.596packgetDyingGaspStatistics Struct Reference	641

8.596.1 Detailed Description	642
8.596.2 Field Documentation	642
8.596.2.1 pSMSAttemptedFlag	642
8.596.2.2 pTimeStamp	642
8.597PCMparams Struct Reference	642
8.597.1 Detailed Description	642
8.597.2 Field Documentation	642
8.597.2.1 iFaceTab	642
8.597.2.2 iFaceTabLen	642
8.598PCSCFFQDNAddress Struct Reference	642
8.598.1 Detailed Description	643
8.598.2 Field Documentation	643
8.598.2.1 fqdnAddr	643
8.598.2.2 fqdnLen	643
8.599PCSCFFQDNAddressList Struct Reference	643
8.599.1 Detailed Description	643
8.599.2 Field Documentation	643
8.599.2.1 numInstances	644
8.599.2.2 pcsfQDNAddress	644
8.600PCSCFIPv4ServerAddressList Struct Reference	644
8.600.1 Detailed Description	644
8.600.2 Field Documentation	644
8.600.2.1 numInstances	644
8.600.2.2 pscsfIPv4Addr	644
8.601PDSPositionData Struct Reference	644
8.601.1 Detailed Description	645
8.601.2 Field Documentation	646
8.601.2.1 pAltitudeWrtEllipsoid	646
8.601.2.2 pAltitudeWrtSealevel	646
8.601.2.3 pHorizontalConfidence	646

8.601.2.4 pHorizontalUncCircular	646
8.601.2.5 pLatitude	646
8.601.2.6 pLongitude	646
8.601.2.7 pPositionSource	646
8.601.2.8 pTimeStamp	646
8.601.2.9 pTimeType	646
8.601.2.10pVerticalConfidence	646
8.601.2.11pVerticalUnc	646
8.602PDSPosMethodStateReq Struct Reference	646
8.602.1 Detailed Description	647
8.602.2 Field Documentation	647
8.602.2.1 pWifiState	647
8.602.2.2 pXtraDataState	647
8.602.2.3 pXtraTimeState	647
8.603peerNumberInfo Struct Reference	647
8.603.1 Detailed Description	647
8.603.2 Field Documentation	649
8.603.2.1 callID	649
8.603.2.2 number	649
8.603.2.3 numLen	649
8.603.2.4 numPI	649
8.603.2.5 numPlan	649
8.603.2.6 numSI	649
8.603.2.7 numType	649
8.604personalizationStatus Struct Reference	649
8.604.1 Detailed Description	649
8.604.2 Field Documentation	650
8.604.2.1 feature	650
8.604.2.2 numFeatures	650
8.604.2.3 unblockLeft	650

8.604.2.4 verifyLeft	650
8.605PhyCaAggPcellInfo Struct Reference	650
8.605.1 Detailed Description	650
8.605.2 Field Documentation	650
8.605.2.1 dl_bw_value	651
8.605.2.2 freq	651
8.605.2.3 iLTEbandValue	651
8.605.2.4 pci	651
8.605.2.5 TlvPresent	651
8.606PhyCaAggScellIDBw Struct Reference	651
8.606.1 Detailed Description	651
8.606.2 Field Documentation	651
8.606.2.1 dl_bw_value	651
8.606.2.2 TlvPresent	651
8.607PhyCaAggScellIndex Struct Reference	651
8.607.1 Detailed Description	651
8.607.2 Field Documentation	652
8.607.2.1 scell_idx	652
8.607.2.2 TlvPresent	652
8.608PhyCaAggScellIndType Struct Reference	652
8.608.1 Detailed Description	652
8.608.2 Field Documentation	652
8.608.2.1 freq	653
8.608.2.2 pci	653
8.608.2.3 scell_state	653
8.608.2.4 TlvPresent	653
8.609PhyCaAggScellInfo Struct Reference	653
8.609.1 Detailed Description	653
8.609.2 Field Documentation	655
8.609.2.1 dl_bw_value	655

8.609.2.2 freq	655
8.609.2.3 iLTEbandValue	655
8.609.2.4 pci	655
8.609.2.5 scell_state	655
8.609.2.6 TlvPresent	655
8.610PilotSetData Struct Reference	655
8.610.1 Detailed Description	655
8.610.2 Field Documentation	656
8.610.2.1 NumPilots	656
8.610.2.2 pPilotSetInfo	656
8.611PilotSetParams Struct Reference	656
8.611.1 Detailed Description	656
8.611.2 Field Documentation	656
8.611.2.1 PilotPN	656
8.611.2.2 PilotStrength	656
8.611.2.3 PilotType	656
8.612pktErrRate Struct Reference	656
8.612.1 Detailed Description	657
8.612.2 Field Documentation	657
8.612.2.1 exponent	657
8.612.2.2 multiplier	657
8.613PLMNNetworkName Struct Reference	657
8.613.1 Detailed Description	657
8.613.2 Field Documentation	657
8.613.2.1 numInstance	657
8.613.2.2 PLMNNetName	657
8.614PLMNNetworkNameData Struct Reference	657
8.614.1 Detailed Description	658
8.614.2 Field Documentation	659
8.614.2.1 codingScheme	659

8.614.2.2 countryInitials	659
8.614.2.3 longName	659
8.614.2.4 longNameLen	659
8.614.2.5 longNameSpareBits	659
8.614.2.6 shortName	659
8.614.2.7 shortNameLen	659
8.614.2.8 shortNameSpareBits	659
8.615Port Struct Reference	659
8.615.1 Detailed Description	659
8.615.2 Field Documentation	660
8.615.2.1 port	660
8.615.2.2 range	660
8.616precisionDilution_s Struct Reference	660
8.616.1 Detailed Description	660
8.616.2 Field Documentation	660
8.616.2.1 HDOP	660
8.616.2.2 PDOP	660
8.616.2.3 VDOP	660
8.617PrefImageList Struct Reference	660
8.617.1 Detailed Description	661
8.617.2 Field Documentation	661
8.617.2.1 listEntries	661
8.617.2.2 listSize	661
8.618prefVoiceSO Struct Reference	661
8.618.1 Detailed Description	661
8.618.2 Field Documentation	663
8.618.2.1 evrcCapability	663
8.618.2.2 homeOrigVoiceSO	663
8.618.2.3 homePageVoiceSO	663
8.618.2.4 namID	663

8.618.2.5 roamOrigVoiceSO	663
8.619Profile3GPP Struct Reference	663
8.619.1 Detailed Description	664
8.619.2 Field Documentation	667
8.619.2.1 pAddrAllocPref	667
8.619.2.2 pAPNClass	667
8.619.2.3 pAPNDisabledFlag	667
8.619.2.4 pAPNName	667
8.619.2.5 pAPNnameSize	667
8.619.2.6 pAuthenticationPref	667
8.619.2.7 pGPRSMinimumQoS	668
8.619.2.8 pGPRSRequestedQos	668
8.619.2.9 plmCnFlag	668
8.619.2.10pIPv4AddrPref	668
8.619.2.11pIPv6AddPref	668
8.619.2.12pPassword	668
8.619.2.13pPasswordSize	668
8.619.2.14pPcscfAddrUsingDhcp	668
8.619.2.15pPcscfAddrUsingPCO	668
8.619.2.16pPDNInactivTimeout	668
8.619.2.17pPdpAccessConFlag	668
8.619.2.18pPdpContext	668
8.619.2.19pPdpDataCompType	668
8.619.2.20pPdpHdrCompType	668
8.619.2.21pPDType	668
8.619.2.22pPriDNSIPv4AddPref	668
8.619.2.23pPriDNSIPv6addpref	668
8.619.2.24pPrimaryID	668
8.619.2.25pProfilename	668
8.619.2.26pProfilenameSize	668

8.619.2.27	pQosClassID	668
8.619.2.28	pSecDNSIPv4AddPref	668
8.619.2.29	pSecDNSIPv6addpref	668
8.619.2.30	pSecondaryFlag	669
8.619.2.31	pTFTID1Params	669
8.619.2.32	pTFTID2Params	669
8.619.2.33	pUMTSMInQoS	669
8.619.2.34	pUMTSMInQoSSigInd	669
8.619.2.35	pUMTSReqQoS	669
8.619.2.36	pUMTSReqQoSSigInd	669
8.619.2.37	pUsername	669
8.619.2.38	pUsernameSize	669
8.620	Profile3GPP2 Struct Reference	669
8.620.1	Detailed Description	670
8.620.2	Field Documentation	673
8.620.2.1	pAllowLinger	673
8.620.2.2	pAPNClass3GPP2	673
8.620.2.3	pAPNEnabled3GPP2	673
8.620.2.4	pApnString	673
8.620.2.5	pApnStringSize	673
8.620.2.6	pAppPriority	673
8.620.2.7	pAppType	673
8.620.2.8	pAuthPassword	673
8.620.2.9	pAuthPasswordSize	673
8.620.2.10	pAuthProtocol	673
8.620.2.11	pAuthRetryCount	673
8.620.2.12	pAuthTimeout	673
8.620.2.13	pDataMode	673
8.620.2.14	pDataRate	673
8.620.2.15	pIpcpAckTimeout	673

8.620.2.16pIpcpCreqRetryCount	674
8.620.2.17pIsPcscfAddressNedded	674
8.620.2.18pLcpAckTimeout	674
8.620.2.19pLcpCreqRetryCount	674
8.620.2.20pNegoDnsSrvrPref	674
8.620.2.21pPDNInactivTimeout3GPP2	674
8.620.2.22pPdnType	674
8.620.2.23pPppSessCloseTimer1x	674
8.620.2.24pPppSessCloseTimerDO	674
8.620.2.25pPrimaryV4DnsAddress	674
8.620.2.26pPriV6DnsAddress	674
8.620.2.27pRATType	674
8.620.2.28pSecondaryV4DnsAddress	674
8.620.2.29pSecV6DnsAddress	674
8.620.2.30pUserId	674
8.620.2.31pUserIdSize	674
8.621 ProfileIdentifier Struct Reference	674
8.621.1 Detailed Description	674
8.621.2 Field Documentation	675
8.621.2.1 profileIndex	675
8.621.2.2 profileType	675
8.622 protocolSubtypeElement Struct Reference	675
8.622.1 Detailed Description	675
8.622.2 Field Documentation	676
8.622.2.1 AccessMac	676
8.622.2.2 AuthProt	676
8.622.2.3 ControlMac	676
8.622.2.4 EncryptProt	676
8.622.2.5 ForwardMac	676
8.622.2.6 IdleState	676

8.622.2.7 KeyExchange	676
8.622.2.8 MultDisc	676
8.622.2.9 PhysicalLayer	676
8.622.2.10ReverseMac	677
8.622.2.11SecProt	677
8.622.2.12VirtStream	677
8.623PSDetachReq Struct Reference	677
8.623.1 Detailed Description	677
8.623.2 Field Documentation	677
8.623.2.1 pDetachAction	677
8.624qaQmi3Gpp2TimeZone Struct Reference	677
8.624.1 Detailed Description	677
8.624.2 Field Documentation	678
8.624.2.1 daylightSavings	678
8.624.2.2 leapSeconds	678
8.624.2.3 localTimeOffset	678
8.625qaQmiInterfaceInfo Struct Reference	678
8.625.1 Detailed Description	678
8.625.2 Field Documentation	679
8.625.2.1 qaQmiinstanceid	679
8.625.2.2 qaQmisvctype	679
8.625.2.3 v4sessionId	679
8.625.2.4 v6sessionId	679
8.626qaQmiServingSystemParam Struct Reference	679
8.626.1 Detailed Description	679
8.626.2 Field Documentation	682
8.626.2.1 BasestationID	682
8.626.2.2 BasestationLatitude	682
8.626.2.3 BasestationLongitude	682
8.626.2.4 CallBarStatus	682

8.626.2.5 CDMA_P_Rev	682
8.626.2.6 CDMASystemInfoExt	682
8.626.2.7 CellID	682
8.626.2.8 concSvcInfo	682
8.626.2.9 CurrentPLMN	682
8.626.2.10DataSrvCapabilities	682
8.626.2.11defaultRoamInd	682
8.626.2.12DetailedSvcInfo	682
8.626.2.13DTMInd	682
8.626.2.14Gpp2TimeZone	682
8.626.2.15GppNetworkDSTAdjustment	682
8.626.2.16GppTimeZone	683
8.626.2.17hdrPersonality	683
8.626.2.18Lac	683
8.626.2.19NetworkID	683
8.626.2.20PRLInd	683
8.626.2.21roamIndicatorVal	683
8.626.2.22RoamingIndicatorList	683
8.626.2.23ServingSystem	683
8.626.2.24SystemID	683
8.626.2.25trackAreaCode	683
8.627QmiCbkCatEventStatusReportInd Struct Reference	683
8.627.1 Field Documentation	683
8.627.1.1 CCETlv	683
8.627.1.2 event_Index	683
8.628QmiCbkLocBestAvailPosInd Struct Reference	683
8.628.1 Detailed Description	684
8.628.2 Field Documentation	688
8.628.2.1 pAltitudeWrtEllipsoid	688
8.628.2.2 pAltitudeWrtMeanSeaLevel	688

8.628.2.3 pGpsTime	688
8.628.2.4 pHeading	688
8.628.2.5 pHeadingUnc	688
8.628.2.6 pHorCirConf	688
8.628.2.7 pHorEllpConf	688
8.628.2.8 pHorReliability	688
8.628.2.9 pHorUncCircular	689
8.628.2.10pHorUncEllipseOrientAzimuth	689
8.628.2.11pHorUncEllipseSemiMajor	689
8.628.2.12pHorUncEllipseSemiMinor	689
8.628.2.13pLatitude	689
8.628.2.14pLongitude	689
8.628.2.15pMagneticDeviation	689
8.628.2.16pPrecisionDilution	689
8.628.2.17pSensorDataUsage	689
8.628.2.18pSpeedHorizontal	689
8.628.2.19pSpeedUnc	689
8.628.2.20pSpeedVertical	689
8.628.2.21pSpeedVerticalUnc	689
8.628.2.22pSvUsedforFix	689
8.628.2.23pTechnologyMask	689
8.628.2.24pTimeSrc	689
8.628.2.25pTimestampUtc	689
8.628.2.26pTimeUnc	689
8.628.2.27pVertConfidence	689
8.628.2.28pVertReliability	689
8.628.2.29pVertUnc	689
8.628.2.30pXid	689
8.628.2.31status	689
8.629QmiCbkLocCradleMountInd Struct Reference	689

8.629.1 Detailed Description	690
8.629.2 Field Documentation	690
8.629.2.1 cradleMountConfigStatus	690
8.630QmiCbkLocEngineStateInd Struct Reference	690
8.630.1 Detailed Description	690
8.630.2 Field Documentation	690
8.630.2.1 engineState	690
8.631QmiCbkLocEventTimeSyncInd Struct Reference	691
8.631.1 Detailed Description	691
8.631.2 Field Documentation	691
8.631.2.1 timeSyncRefCounter	691
8.632QmiCbkLocInjectPositionInd Struct Reference	691
8.632.1 Detailed Description	691
8.632.2 Field Documentation	692
8.632.2.1 status	692
8.633QmiCbkLocInjectSensorDataInd Struct Reference	692
8.633.1 Detailed Description	692
8.633.2 Field Documentation	693
8.633.2.1 injectSensorDataStatus	693
8.633.2.2 pAccelSamplesAccepted	693
8.633.2.3 pAccelTempSamplesAccepted	693
8.633.2.4 pGyroSamplesAccepted	693
8.633.2.5 pGyroTempSamplesAccepted	693
8.633.2.6 pOpaqueIdentifier	693
8.634QmiCbkLocInjectTimeInd Struct Reference	693
8.634.1 Detailed Description	694
8.634.2 Field Documentation	694
8.634.2.1 injectTimeSyncStatus	694
8.635QmiCbkLocInjectUTCTimeInd Struct Reference	694
8.635.1 Detailed Description	694

8.635.2 Field Documentation	695
8.635.2.1 status	695
8.636QmiCbkLocPositionReportInd Struct Reference	695
8.636.1 Detailed Description	695
8.636.2 Field Documentation	699
8.636.2.1 pAltitudeAssumed	699
8.636.2.2 pAltitudeWrtEllipsoid	699
8.636.2.3 pAltitudeWrtMeanSeaLevel	699
8.636.2.4 pFixId	700
8.636.2.5 pGpsTime	700
8.636.2.6 pHeading	700
8.636.2.7 pHeadingUnc	700
8.636.2.8 pHorConfidence	700
8.636.2.9 pHorReliability	700
8.636.2.10pHorUncCircular	700
8.636.2.11pHorUncEllipseOrientAzimuth	700
8.636.2.12pHorUncEllipseSemiMajor	700
8.636.2.13pHorUncEllipseSemiMinor	700
8.636.2.14pLatitude	700
8.636.2.15pLeapSeconds	700
8.636.2.16pLongitude	700
8.636.2.17pMagneticDeviation	700
8.636.2.18pPrecisionDilution	700
8.636.2.19pSensorDataUsage	700
8.636.2.20pSpeedHorizontal	700
8.636.2.21pSpeedUnc	700
8.636.2.22pSpeedVertical	700
8.636.2.23pSvUsedforFix	700
8.636.2.24pTechnologyMask	700
8.636.2.25pTimeSrc	700

8.636.2.26	pTimestampUtc	700
8.636.2.27	pTimeUnc	701
8.636.2.28	pVertConfidence	701
8.636.2.29	pVertReliability	701
8.636.2.30	pVertUnc	701
8.636.2.31	sessionId	701
8.636.2.32	sessionStatus	701
8.637	QmiCbkLocSensorStreamingInd Struct Reference	701
8.637.1	Detailed Description	701
8.637.2	Field Documentation	701
8.637.2.1	pAccelAcceptReady	701
8.637.2.2	pAccelTempAcceptReady	701
8.637.2.3	pGyroAcceptReady	702
8.637.2.4	pGyroTempAcceptReady	702
8.638	QmiCbkLocSetExtPowerConfigInd Struct Reference	702
8.638.1	Detailed Description	702
8.638.2	Field Documentation	702
8.638.2.1	status	702
8.639	QmiCbkNasLTECphyCaInfo Struct Reference	702
8.639.1	Detailed Description	703
8.639.2	Field Documentation	703
8.639.2.1	sPhyCaAggPcellInfo	703
8.639.2.2	sPhyCaAggScellIDBw	703
8.639.2.3	sPhyCaAggScellIndex	703
8.639.2.4	sPhyCaAggScellIndType	703
8.639.2.5	sPhyCaAggScellInfo	703
8.640	QmiCbkSwiOmaDmEventStatusReportInd Struct Reference	703
8.640.1	Field Documentation	703
8.640.1.1	SITlv	703
8.641	QmiCbkSwiOmaDmEventStatusReportIndExt Struct Reference	703

8.641.1 Field Documentation	704
8.641.1.1 SITlv	704
8.642QmiCbkTmdMitiLvlRptInd Struct Reference	704
8.642.1 Detailed Description	704
8.642.2 Field Documentation	704
8.642.2.1 currentMitigationLvl	704
8.642.2.2 MitigationDevInfo	704
8.643QmiCbkWdsStatisticsIndState Struct Reference	704
8.643.1 Detailed Description	704
8.643.2 Field Documentation	705
8.643.2.1 RxDropConutTlv	705
8.643.2.2 RxOkByteCountTlv	705
8.643.2.3 RxOkConutTlv	705
8.643.2.4 TxDropConutTlv	705
8.643.2.5 TxOkByteCountTlv	705
8.643.2.6 TxOkConutTlv	705
8.644qmifwinfo_s Struct Reference	705
8.644.1 Detailed Description	705
8.644.2 Field Documentation	706
8.644.2.1 dev	706
8.644.2.2 g	706
8.644.2.3 s	706
8.645QmiNas3GppNetworkInfo Struct Reference	706
8.645.1 Detailed Description	706
8.645.2 Field Documentation	707
8.645.2.1 pDescription	707
8.645.2.2 pForbidden	707
8.645.2.3 pInUse	707
8.645.2.4 pMCC	707
8.645.2.5 pMNC	707

8.645.2.6 pPreferred	707
8.645.2.7 pRoaming	707
8.646QmiNasGetRFBandInfoResp Struct Reference	708
8.646.1 Field Documentation	708
8.646.1.1 pInstancesSize	708
8.646.1.2 pRFBandInfoElements	708
8.646.1.3 results	708
8.647QmiNasPerformNetworkScanResp Struct Reference	708
8.647.1 Field Documentation	708
8.647.1.1 pInstances	708
8.647.1.2 pInstanceSize	708
8.647.1.3 results	708
8.648qmiSmsMessageList Struct Reference	708
8.648.1 Detailed Description	708
8.648.2 Field Documentation	709
8.648.2.1 messageIndex	709
8.648.2.2 messageTag	709
8.649qmiWSDDataBearerTechnology Struct Reference	709
8.649.1 Detailed Description	709
8.649.2 Field Documentation	709
8.649.2.1 currentNetwork	709
8.649.2.2 ratMask	709
8.649.2.3 soMask	709
8.650QmiWdsIpAddressInfo Struct Reference	709
8.650.1 Detailed Description	709
8.650.2 Field Documentation	710
8.650.2.1 pIPAddressV4	710
8.650.2.2 pIPAddressV6	710
8.650.2.3 pIPv6prefixlen	710
8.651qmiWdsRunTimeSettings Struct Reference	710

8.651.1 Detailed Description	711
8.651.2 Field Documentation	713
8.651.2.1 pAPNName	713
8.651.2.2 pAuthentication	713
8.651.2.3 pDomainList	713
8.651.2.4 pGPRSGrantedQoS	713
8.651.2.5 pGWAddressV4	713
8.651.2.6 pIMCNflag	713
8.651.2.7 pIPAddressV4	713
8.651.2.8 pIPFamilyPreference	713
8.651.2.9 pIPv6AddrInfo	713
8.651.2.10 pIPv6GWAddrInfo	713
8.651.2.11 pMtu	713
8.651.2.12 pPCSCFAddrPCO	713
8.651.2.13 pPCSCFFQDNAddrList	713
8.651.2.14 pPDPTType	713
8.651.2.15 pPrimaryDNSV4	714
8.651.2.16 pPrimaryDNSV6	714
8.651.2.17 pProfileID	714
8.651.2.18 pProfileName	714
8.651.2.19 pSecondaryDNSV4	714
8.651.2.20 pSecondaryDNSV6	714
8.651.2.21 pServerAddrList	714
8.651.2.22 pSubnetMaskV4	714
8.651.2.23 pTechnology	714
8.651.2.24 pUMTSGrantedQoS	714
8.651.2.25 pUsername	714
8.652 QoSClassID Struct Reference	714
8.652.1 Detailed Description	714
8.652.2 Field Documentation	715

8.652.2.1 gDIBitRate	715
8.652.2.2 gUIBitRate	715
8.652.2.3 maxDIBitRate	715
8.652.2.4 maxUIBitRate	715
8.652.2.5 QCI	715
8.653QosEventInfo Struct Reference	715
8.653.1 Detailed Description	715
8.653.2 Field Documentation	716
8.653.2.1 pDataBearer	716
8.653.2.2 pPacketsCountRX	716
8.653.2.3 pPacketsCountTX	716
8.653.2.4 pTotalBytesRX	716
8.653.2.5 pTotalBytesTX	716
8.654QosFlowInfo Struct Reference	716
8.654.1 Detailed Description	716
8.654.2 Field Documentation	717
8.654.2.1 pBearerID	717
8.654.2.2 pQFlowState	717
8.654.2.3 pRxQFilter	717
8.654.2.4 pRxQFlowGranted	717
8.654.2.5 pTxQFilter	717
8.654.2.6 pTxQFlowGranted	717
8.655QosFlowInfoState Struct Reference	717
8.655.1 Detailed Description	717
8.655.2 Field Documentation	718
8.655.2.1 id	718
8.655.2.2 isNewFlow	718
8.655.2.3 state	718
8.656QosMap Struct Reference	718
8.656.1 Detailed Description	718

8.656.2 Field Documentation	719
8.656.2.1 dscp	719
8.656.2.2 qos_id	719
8.656.2.3 state	719
8.657 RankIndicatorInd Struct Reference	719
8.657.1 Field Documentation	719
8.657.1.1 Count1	719
8.657.1.2 Count2	719
8.658 readResult Struct Reference	719
8.658.1 Detailed Description	719
8.658.2 Field Documentation	719
8.658.2.1 content	720
8.658.2.2 contentLen	720
8.659 readTransparentInfo Struct Reference	720
8.659.1 Detailed Description	720
8.659.2 Field Documentation	720
8.659.2.1 length	720
8.659.2.2 offset	720
8.660 redirNumInfo Struct Reference	720
8.660.1 Detailed Description	720
8.660.2 Field Documentation	722
8.660.2.1 number	722
8.660.2.2 numLen	722
8.660.2.3 numPlan	722
8.660.2.4 numType	722
8.660.2.5 PI	722
8.660.2.6 reason	722
8.660.2.7 SI	722
8.661 registerRefresh Struct Reference	722
8.661.1 Detailed Description	722

8.661.2 Field Documentation	723
8.661.2.1 arrfileInfo	723
8.661.2.2 numFiles	723
8.661.2.3 registerFlag	723
8.661.2.4 voteForInit	723
8.662remainingRetries Struct Reference	723
8.662.1 Detailed Description	723
8.662.2 Field Documentation	723
8.662.2.1 unblockLeft	723
8.662.2.2 verifyLeft	723
8.663remotePartyName Struct Reference	723
8.663.1 Detailed Description	724
8.663.2 Field Documentation	724
8.663.2.1 callerName	724
8.663.2.2 codingScheme	724
8.663.2.3 nameLen	724
8.663.2.4 namePI	724
8.664remotePartyNum Struct Reference	724
8.664.1 Detailed Description	725
8.664.2 Field Documentation	725
8.664.2.1 numLen	725
8.664.2.2 presentationInd	725
8.664.2.3 remPartyNumber	725
8.665ReqFieldsList Struct Reference	725
8.665.1 Detailed Description	725
8.665.2 Field Documentation	726
8.665.2.1 requestFields	726
8.665.2.2 requestFieldsLen	726
8.666RespFieldsList Struct Reference	726
8.666.1 Detailed Description	726

8.666.2 Field Documentation	726
8.666.2.1 responseFields	726
8.666.2.2 responseFieldsLen	726
8.667RFBandInfoElements Struct Reference	726
8.667.1 Detailed Description	727
8.667.2 Field Documentation	727
8.667.2.1 activeBandClass	727
8.667.2.2 activeBandClass	727
8.667.2.3 activeChannel	727
8.667.2.4 activeChannel	727
8.667.2.5 radiolInterface	727
8.667.2.6 radiolInterface	727
8.668rmTrasferStaticsReq Struct Reference	727
8.668.1 Detailed Description	727
8.668.2 Field Documentation	728
8.668.2.1 bResetStatistics	728
8.668.2.2 ulMask	728
8.669roamIndList Struct Reference	728
8.669.1 Detailed Description	728
8.669.2 Field Documentation	728
8.669.2.1 numInstances	728
8.669.2.2 radiolInterface	729
8.669.2.3 roamIndicator	729
8.670RoamingInfo Struct Reference	729
8.670.1 Field Documentation	729
8.670.1.1 roaming_ind	729
8.670.1.2 TlvPresent	729
8.671roamTimer Struct Reference	729
8.671.1 Detailed Description	729
8.671.2 Field Documentation	729

8.671.2.1 namID	729
8.671.2.2 roamTimerValue	730
8.672RSRPThresh Struct Reference	730
8.672.1 Detailed Description	730
8.672.2 Field Documentation	730
8.672.2.1 pRSRPThresList	730
8.672.2.2 RSRPThresListLen	730
8.673rsrqInformation Struct Reference	730
8.673.1 Detailed Description	730
8.673.2 Field Documentation	731
8.673.2.1 radiolf	731
8.673.2.2 rsrq	731
8.674RSRQThresh Struct Reference	731
8.674.1 Detailed Description	731
8.674.2 Field Documentation	731
8.674.2.1 pRSRQThresList	731
8.674.2.2 RSRQThresListLen	732
8.675RSSIThresh Struct Reference	732
8.675.1 Detailed Description	732
8.675.2 Field Documentation	732
8.675.2.1 pRSSIThresList	732
8.675.2.2 RSSIThresListLen	732
8.676RXAGCList Struct Reference	732
8.676.1 Detailed Description	733
8.676.2 Field Documentation	733
8.676.2.1 pRXAIG	733
8.676.2.2 pRXComprSlope	733
8.676.2.3 pRXComprThres	733
8.676.2.4 pRXExpSlope	733
8.676.2.5 pRXExpThres	733

8.676.2.6 pRXStaticGain	733
8.677RXAVCList Struct Reference	733
8.677.1 Detailed Description	733
8.677.2 Field Documentation	734
8.677.2.1 pAVRXAVCHheadroom	734
8.677.2.2 pAVRXAVCSens	734
8.678rxInfo Struct Reference	734
8.678.1 Detailed Description	734
8.678.2 Field Documentation	735
8.678.2.1 ecio	735
8.678.2.2 isRadioTuned	735
8.678.2.3 phase	735
8.678.2.4 rscp	735
8.678.2.5 rsrp	735
8.678.2.6 rxPower	735
8.679RXPCMIIRFiltr Struct Reference	735
8.679.1 Detailed Description	735
8.679.2 Field Documentation	737
8.679.2.1 pFlag	737
8.679.2.2 pStage0Val	737
8.679.2.3 pStage1Val	737
8.679.2.4 pStage2Val	737
8.679.2.5 pStage3Val	737
8.679.2.6 pStage4Val	737
8.679.2.7 pStageCnt	737
8.680RxSigInfo Struct Reference	737
8.680.1 Detailed Description	737
8.680.2 Field Documentation	738
8.680.2.1 isRadioTuned	738
8.680.2.2 rsrp	738

8.680.2.3 rxChainIndex	738
8.680.2.4 rxPower	738
8.681 rxSignalStrengthListElement Struct Reference	738
8.681.1 Detailed Description	738
8.681.2 Field Documentation	739
8.681.2.1 radiolf	739
8.681.2.2 rxSignalStrength	739
8.682 sApnExtraParams Struct Reference	739
8.682.1 Detailed Description	739
8.682.2 Field Documentation	740
8.682.2.1 ambr_dl	740
8.682.2.2 ambr_dl_ext	740
8.682.2.3 ambr_dl_ext2	740
8.682.2.4 ambr_ul	740
8.682.2.5 ambr_ul_ext	740
8.682.2.6 ambr_ul_ext2	740
8.682.2.7 apnId	740
8.683 satelliteInfo Struct Reference	740
8.683.1 Detailed Description	740
8.683.2 Field Documentation	742
8.683.2.1 azimuth	742
8.683.2.2 elevation	742
8.683.2.3 gnssSvId	742
8.683.2.4 healthStatus	742
8.683.2.5 snr	742
8.683.2.6 svInfoMask	742
8.683.2.7 svListLen	742
8.683.2.8 svStatus	742
8.683.2.9 system	742
8.683.2.10 validMask	742

8.684SccRxInfo Struct Reference	742
8.684.1 Detailed Description	742
8.684.2 Field Documentation	743
8.684.2.1 numInstances	743
8.684.2.2 rsrq	743
8.684.2.3 sigInfo	743
8.684.2.4 snr	743
8.684.2.5 TlvPresent	743
8.685sensorData Struct Reference	743
8.685.1 Detailed Description	743
8.685.2 Field Documentation	744
8.685.2.1 flags	744
8.685.2.2 sensorDataLen	744
8.685.2.3 timeOfFirstSample	744
8.685.2.4 timeOffset	744
8.685.2.5 xAxis	745
8.685.2.6 yAxis	745
8.685.2.7 zAxis	745
8.686sensorData_t Struct Reference	745
8.686.1 Detailed Description	745
8.686.2 Field Documentation	746
8.686.2.1 flags	746
8.686.2.2 sensorDataLen	746
8.686.2.3 timeOfFirstSample	746
8.686.2.4 timeOffset	746
8.686.2.5 xAxis	746
8.686.2.6 yAxis	746
8.686.2.7 zAxis	746
8.687sensorDataUsage_s Struct Reference	746
8.687.1 Detailed Description	746

8.687.2 Field Documentation	747
8.687.2.1 aidingIndicatorMask	747
8.687.2.2 usageMask	747
8.688serialNumbersInfo Struct Reference	747
8.688.1 Detailed Description	747
8.688.2 Field Documentation	748
8.688.2.1 esnSize	748
8.688.2.2 imeiSize	748
8.688.2.3 imeiSvnSize	748
8.688.2.4 meidSize	748
8.688.2.5 pESNString	748
8.688.2.6 pIMEIString	748
8.688.2.7 plmeiSvnString	748
8.688.2.8 pMEIDString	748
8.689serviceProviderName Struct Reference	749
8.689.1 Detailed Description	749
8.689.2 Field Documentation	749
8.689.2.1 displayCondition	749
8.689.2.2 spn	749
8.689.2.3 spnLength	749
8.690ServingSystemInfo Struct Reference	749
8.690.1 Detailed Description	750
8.690.2 Field Documentation	750
8.690.2.1 csAttachState	751
8.690.2.2 hdrPersonality	751
8.690.2.3 psAttachState	751
8.690.2.4 radiolInterfaceList	751
8.690.2.5 radiolInterfaceNo	751
8.690.2.6 registrationState	751
8.690.2.7 selectedNetwork	751

8.691servSystem Struct Reference	751
8.691.1 Detailed Description	751
8.691.2 Field Documentation	752
8.691.2.1 csAttachState	752
8.691.2.2 numRadioInterfaces	752
8.691.2.3 psAttachState	752
8.691.2.4 radiolInterface	752
8.691.2.5 regState	752
8.691.2.6 selNetwork	752
8.692sessionInfo Union Reference	752
8.692.1 Detailed Description	753
8.692.2 Field Documentation	753
8.692.2.1 omaDmConfig	753
8.692.2.2 omaDmFota	753
8.692.2.3 omaDmNotifications	753
8.693sessionInfoExt Union Reference	753
8.693.1 Detailed Description	753
8.693.2 Field Documentation	753
8.693.2.1 omaDmConfig	753
8.693.2.2 omaDmFota	753
8.694sessionInfoTlv Struct Reference	753
8.694.1 Detailed Description	753
8.694.2 Field Documentation	754
8.694.2.1 sessionInfo	754
8.694.2.2 sessionType	754
8.694.2.3 TlvPresent	754
8.695sessionInfoTlvExt Struct Reference	754
8.695.1 Detailed Description	754
8.695.2 Field Documentation	754
8.695.2.1 sessionInfo	754

8.695.2.2 sessionType	754
8.695.2.3 TlvPresent	754
8.696SetAudioPathConfigReq Struct Reference	754
8.696.1 Detailed Description	755
8.696.2 Field Documentation	756
8.696.2.1 pCodecSTGain	756
8.696.2.2 pDTMFTXGain	756
8.696.2.3 pECMode	756
8.696.2.4 pNSEnable	756
8.696.2.5 Profile	756
8.696.2.6 pRXAGCList	756
8.696.2.7 pRXAVCAGCSwitch	756
8.696.2.8 pRXAVCList	756
8.696.2.9 pRXPCMIIRFiltr	756
8.696.2.10pTXAGCList	756
8.696.2.11pTXAVCSwitch	756
8.696.2.12pTXGain	756
8.696.2.13pTXPCMIIRFiltr	756
8.697SetAudioProfileReq Struct Reference	756
8.697.1 Detailed Description	757
8.697.2 Field Documentation	757
8.697.2.1 EarMute	757
8.697.2.2 Generator	757
8.697.2.3 MicMute	757
8.697.2.4 Profile	757
8.697.2.5 Volume	758
8.698SetAudioVolTLBConfigReq Struct Reference	758
8.698.1 Detailed Description	758
8.698.2 Field Documentation	758
8.698.2.1 Generator	758

8.698.2.2 Item	758
8.698.2.3 Profile	758
8.698.2.4 Volume	759
8.698.2.5 VolValue	759
8.699SetAudioVolTLBConfigResp Struct Reference	759
8.699.1 Detailed Description	759
8.699.2 Field Documentation	759
8.699.2.1 ResCode	759
8.700setCustomSettingV2 Struct Reference	759
8.700.1 Detailed Description	759
8.700.2 Field Documentation	760
8.700.2.1 cust_id	760
8.700.2.2 cust_value	760
8.700.2.3 value_length	760
8.701setDyingGaspCfg Struct Reference	760
8.701.1 Detailed Description	760
8.701.2 Field Documentation	760
8.701.2.1 pDestSMSContent	760
8.701.2.2 pDestSMSNum	760
8.702SetIMSSMSConfigReq Struct Reference	760
8.702.1 Detailed Description	760
8.702.2 Field Documentation	761
8.702.2.1 pPhoneCtxtURI	761
8.702.2.2 pPhoneCtxtURILen	761
8.702.2.3 pSMSFormat	761
8.702.2.4 pSMSOverIPNwInd	761
8.703SetIMSSMSConfigResp Struct Reference	761
8.703.1 Detailed Description	761
8.703.2 Field Documentation	762
8.703.2.1 pSettingResp	762

8.704SetIMSUserConfigReq Struct Reference	762
8.704.1 Detailed Description	762
8.704.2 Field Documentation	762
8.704.2.1 pIMSDomain	762
8.704.2.2 pIMSDomainLen	762
8.705SetIMSUserConfigResp Struct Reference	762
8.705.1 Detailed Description	762
8.705.2 Field Documentation	763
8.705.2.1 pSettingResp	763
8.706SetIMSVoIPConfigReq Struct Reference	763
8.706.1 Detailed Description	763
8.706.2 Field Documentation	764
8.706.2.1 pAmrMode	765
8.706.2.2 pAmrOctetAligned	765
8.706.2.3 pAmrWbEnable	765
8.706.2.4 pAmrWBMode	765
8.706.2.5 pAmrWBOctetAligned	765
8.706.2.6 pMinSessionExpiryTimer	765
8.706.2.7 pRingBackTimer	765
8.706.2.8 pRingingTimer	765
8.706.2.9 pRTPRTCPInactTimer	765
8.706.2.10pScrAmrEnable	765
8.706.2.11pScrAmrWbEnable	765
8.706.2.12pSessionExpiryTimer	765
8.707SetIMSVoIPConfigResp Struct Reference	765
8.707.1 Detailed Description	765
8.707.2 Field Documentation	765
8.707.2.1 pSettingResp	765
8.708SetM2MAudioAVCFGRReq Struct Reference	766
8.708.1 Detailed Description	766

8.708.2 Field Documentation	766
8.708.2.1 Device	766
8.708.2.2 PIFACEId	766
8.708.2.3 pPCMPParams	766
8.708.2.4 Profile	766
8.709SetM2MAudioLPBKReq Struct Reference	766
8.709.1 Detailed Description	767
8.709.2 Field Documentation	767
8.709.2.1 Enable	767
8.710SetM2MAudioProfileReq Struct Reference	767
8.710.1 Detailed Description	767
8.710.2 Field Documentation	768
8.710.2.1 pCwtMute	768
8.710.2.2 pEarMute	768
8.710.2.3 pGenerator	768
8.710.2.4 pMicMute	768
8.710.2.5 Profile	768
8.710.2.6 pVolume	768
8.711SetM2MAudioVolumeReq Struct Reference	768
8.711.1 Detailed Description	768
8.711.2 Field Documentation	769
8.711.2.1 Generator	769
8.711.2.2 Level	769
8.711.2.3 Profile	769
8.712SetM2MAVMuteReq Struct Reference	769
8.712.1 Detailed Description	769
8.712.2 Field Documentation	769
8.712.2.1 EarMute	769
8.712.2.2 MicMute	769
8.712.2.3 pCwtMute	769

8.712.2.4 Profile	770
8.713SetM2MSpkrGainReq Struct Reference	770
8.713.1 Detailed Description	770
8.713.2 Field Documentation	770
8.713.2.1 Profile	770
8.713.2.2 Value	770
8.714setPINProtection Struct Reference	770
8.714.1 Detailed Description	770
8.714.2 Field Documentation	771
8.714.2.1 pinID	771
8.714.2.2 pinLength	771
8.714.2.3 pinOperation	771
8.714.2.4 pinValue	771
8.715SetRegMgrConfigReq Struct Reference	771
8.715.1 Detailed Description	771
8.715.2 Field Documentation	772
8.715.2.1 pCSCFPortName	772
8.715.2.2 pCSCFPortNameLen	772
8.715.2.3 pIMSTestMode	772
8.715.2.4 pPriCSCFPort	772
8.716SetRegMgrConfigResp Struct Reference	772
8.716.1 Detailed Description	772
8.716.2 Field Documentation	772
8.716.2.1 pSettingResp	773
8.717setSignalStrengthInfo Struct Reference	773
8.717.1 Detailed Description	773
8.717.2 Field Documentation	776
8.717.2.1 pCDMAECIODelta	776
8.717.2.2 pCDMAECIOThresh	776
8.717.2.3 pCDMARSSIDelta	776

8.717.2.4 pCDMARSSIThresh	776
8.717.2.5 pGSMRSSIDelta	776
8.717.2.6 pGSMRSSIThresh	776
8.717.2.7 pHDRICIODelta	776
8.717.2.8 pHDRICIOThresh	777
8.717.2.9 pHDRIODelta	777
8.717.2.10pHDRIOThresh	777
8.717.2.11pHDRRSSIDelta	777
8.717.2.12pHDRRSSIThresh	777
8.717.2.13pHDRSINRDelta	777
8.717.2.14pHDRSINRThresh	777
8.717.2.15pLTERSRPDelta	777
8.717.2.16pLTERSRPThresh	777
8.717.2.17pLTERSRQDelta	777
8.717.2.18pLTERSRQThresh	777
8.717.2.19pLTERSSIDelta	777
8.717.2.20pLTERSSIThresh	777
8.717.2.21pLTESigRptConfig	777
8.717.2.22pLTESNRDelta	777
8.717.2.23pLTESNRThresh	777
8.717.2.24pTDSCDMAECIODelta	777
8.717.2.25pTDSCDMAECIOThresh	777
8.717.2.26pTDSCDMARSCPDelta	777
8.717.2.27pTDSCDMARSCPThresh	777
8.717.2.28pTDSCDMARSSIDelta	777
8.717.2.29pTDSCDMARSSIThresh	777
8.717.2.30pTDSCDMASINRDelta	777
8.717.2.31pTDSCDMASINRThresh	778
8.717.2.32pWCDMAECIODelta	778
8.717.2.33pWCDMAECIOThresh	778

8.717.2.34 pWCDMARSSIDelta	778
8.717.2.35 pWCDMARSSIThresh	778
8.718SetSIPConfigReq Struct Reference	778
8.718.1 Detailed Description	778
8.718.2 Field Documentation	779
8.718.2.1 pSigCompEnabled	779
8.718.2.2 pSIPLocalPort	779
8.718.2.3 pSubscribeTimer	779
8.718.2.4 pTimerSIPReg	779
8.718.2.5 pTimerT1	779
8.718.2.6 pTimerT2	779
8.718.2.7 pTimerTf	779
8.719SetSIPConfigResp Struct Reference	779
8.719.1 Detailed Description	779
8.719.2 Field Documentation	779
8.719.2.1 pSettingResp	779
8.720sGetDeviceSeriesResult Struct Reference	779
8.720.1 Detailed Description	780
8.720.2 Field Documentation	780
8.720.2.1 eDevice	780
8.720.2.2 uResult	780
8.721sidNid Struct Reference	780
8.721.1 Detailed Description	780
8.721.2 Field Documentation	780
8.721.2.1 nid	780
8.721.2.2 sid	780
8.722sigInfo Struct Reference	780
8.722.1 Detailed Description	781
8.722.2 Field Documentation	782
8.722.2.1 pECIOThresh	782

8.722.2.2 pHDRSINRThresh	782
8.722.2.3 pIOTresh	782
8.722.2.4 pLTESigRptCfg	782
8.722.2.5 pLTESNRThresh	782
8.722.2.6 pRSRPThresh	782
8.722.2.7 pRSRQThresh	782
8.722.2.8 pRSSIThresh	782
8.722.2.9 pTDSCDMASINRCONFThresh	782
8.723signalInfo Struct Reference	782
8.723.1 Detailed Description	782
8.723.2 Field Documentation	782
8.723.2.1 alertPitch	783
8.723.2.2 signal	783
8.723.2.3 signalType	783
8.724SignalStrengthDataType Struct Reference	783
8.724.1 Field Documentation	783
8.724.1.1 thresholds	783
8.724.1.2 thresholdsSize	783
8.725slot_t Struct Reference	783
8.725.1 Detailed Description	783
8.725.2 Field Documentation	784
8.725.2.1 bICCID	784
8.725.2.2 bICCIDLength	784
8.725.2.3 bLogicalSlot	784
8.725.2.4 uPhyCardStatus	784
8.725.2.5 uPhySlotStatus	784
8.726slotInf Struct Reference	784
8.726.1 Detailed Description	785
8.726.2 Field Documentation	786
8.726.2.1 AppStatus	786

8.726.2.2 cardState	786
8.726.2.3 errorState	786
8.726.2.4 numApp	786
8.726.2.5 upinRetries	786
8.726.2.6 upinState	786
8.726.2.7 upukRetries	786
8.727slotInfo Struct Reference	786
8.727.1 Detailed Description	786
8.727.2 Field Documentation	787
8.727.2.1 AppStatus	787
8.727.2.2 cardState	787
8.727.2.3 errorState	787
8.727.2.4 numApp	787
8.727.2.5 upinRetries	787
8.727.2.6 upinState	787
8.727.2.7 upukRetries	787
8.728slots_t Struct Reference	787
8.728.1 Field Documentation	788
8.728.1.1 uimSlotStatus	788
8.729slqsautoconnect Struct Reference	788
8.729.1 Detailed Description	788
8.729.2 Field Documentation	788
8.729.2.1 acroamsetting	788
8.729.2.2 acsetting	788
8.729.2.3 action	788
8.730SLQSDelateProfileParams Struct Reference	788
8.730.1 Detailed Description	789
8.730.2 Field Documentation	789
8.730.2.1 profileIndex	789
8.730.2.2 profileType	789

8.731slqsfwinfo_s Struct Reference	789
8.731.1 Detailed Description	789
8.731.2 Field Documentation	790
8.731.2.1 appversion_str	790
8.731.2.2 bootversion_str	790
8.731.2.3 carrier_str	790
8.731.2.4 cur_carr_name	790
8.731.2.5 cur_carr_rev	790
8.731.2.6 modelid_str	790
8.731.2.7 packageid_str	790
8.731.2.8 priversion_str	790
8.731.2.9 sku_str	791
8.732SlqsNas3GppNetworkInfo Struct Reference	791
8.732.1 Detailed Description	791
8.732.2 Field Documentation	792
8.732.2.1 Description	792
8.732.2.2 Forbidden	792
8.732.2.3 InUse	792
8.732.2.4 MCC	792
8.732.2.5 MNC	792
8.732.2.6 Preferred	792
8.732.2.7 Roaming	792
8.733SlqsNasPcsDigit Struct Reference	792
8.733.1 Detailed Description	792
8.733.2 Field Documentation	792
8.733.2.1 includes_pcs_digit	792
8.733.2.2 MCC	793
8.733.2.3 MNC	793
8.734slqssendasyncsmsparams_s Struct Reference	793
8.734.1 Detailed Description	793

8.734.2 Field Documentation	794
8.734.2.1 messageFormat	794
8.734.2.2 messageSize	794
8.734.2.3 pFollowOnDC	794
8.734.2.4 pForceOnDC	794
8.734.2.5 pLinktimer	794
8.734.2.6 pMessage	794
8.734.2.7 pRetryMessage	795
8.734.2.8 pRetryMessageId	795
8.734.2.9 pServiceOption	795
8.734.2.10 pSmsOnIms	795
8.734.2.11 pUserData	795
8.735slqssendsmsparams_s Struct Reference	795
8.735.1 Detailed Description	795
8.735.2 Field Documentation	796
8.735.2.1 messageFailureCode	796
8.735.2.2 messageFormat	796
8.735.2.3 messageId	796
8.735.2.4 messageSize	796
8.735.2.5 pLinktimer	796
8.735.2.6 pMessage	796
8.735.2.7 pSmsOnIms	796
8.736slqsSessionStateInfo Struct Reference	796
8.736.1 Detailed Description	797
8.736.2 Field Documentation	797
8.736.2.1 pQmiInterfaceInfo	797
8.736.2.2 reconfiguration_required	797
8.736.2.3 sessionEndReason	797
8.736.2.4 state	797
8.737slqsSignalStrengthInfo Struct Reference	797

8.737.1 Detailed Description	798
8.737.2 Field Documentation	799
8.737.2.1 ecioList	799
8.737.2.2 ecioListLen	799
8.737.2.3 errorRateList	799
8.737.2.4 errorRateListLen	799
8.737.2.5 lo	799
8.737.2.6 ltersrp	799
8.737.2.7 ltesnr	799
8.737.2.8 rsrqInfo	799
8.737.2.9 rxSignalStrengthList	799
8.737.2.10rxSignalStrengthListLen	799
8.737.2.11signalStrengthReqMask	799
8.737.2.12sinr	800
8.738SLQSSignalStrengthsIndReq Struct Reference	800
8.738.1 Detailed Description	800
8.738.2 Field Documentation	801
8.738.2.1 ecioDelta	801
8.738.2.2 ecioThresholdList	801
8.738.2.3 ecioThresholdListLen	801
8.738.2.4 ioDelta	801
8.738.2.5 lteRsrpDelta	801
8.738.2.6 lteSnrDelta	801
8.738.2.7 rsrqDelta	801
8.738.2.8 rxSignalStrengthDelta	801
8.738.2.9 sinrDelta	802
8.738.2.10sinrThresholdList	802
8.738.2.11sinrThresholdListLen	802
8.739SLQSSignalStrengthsInformation Struct Reference	802
8.739.1 Detailed Description	802

8.739.2 Field Documentation	803
8.739.2.1 eciInfo	803
8.739.2.2 errorRateInfo	803
8.739.2.3 io	803
8.739.2.4 lteRsrpinfo	803
8.739.2.5 lteSnrinfo	803
8.739.2.6 rsrqInfo	803
8.739.2.7 rxSignalStrengthInfo	803
8.739.2.8 sinr	803
8.740slqsWdsEventInfo Struct Reference	803
8.740.1 Detailed Description	803
8.740.2 Field Documentation	804
8.740.2.1 pDataBearer	804
8.740.2.2 pDormancyStatus	804
8.740.2.3 pPacketsCountRX	804
8.740.2.4 pPacketsCountTX	805
8.740.2.5 pQmiInterfaceInfo	805
8.740.2.6 pTotalBytesRX	805
8.740.2.7 pTotalBytesTX	805
8.741SMSAsyncRawSend_s Struct Reference	805
8.741.1 Detailed Description	805
8.741.2 Field Documentation	806
8.741.2.1 alphaIDLen	806
8.741.2.2 causeCode	806
8.741.2.3 errorClass	806
8.741.2.4 messageID	806
8.741.2.5 msgDelFailureCause	806
8.741.2.6 msgDelFailureType	806
8.741.2.7 pAlphaID	806
8.741.2.8 RPCause	806

8.741.2.9 sendStatus	806
8.741.2.10TPCause	806
8.741.2.11userData	807
8.742sMSCAddress Struct Reference	807
8.742.1 Detailed Description	807
8.742.2 Field Documentation	807
8.742.2.1 data	807
8.742.2.2 length	807
8.743SMSCAddress Struct Reference	807
8.743.1 Detailed Description	807
8.743.2 Field Documentation	808
8.743.2.1 data	808
8.743.2.2 length	808
8.744sMSCAddressTlv Struct Reference	808
8.744.1 Detailed Description	808
8.744.2 Field Documentation	808
8.744.2.1 SMSCInfo	808
8.744.2.2 TlvPresent	808
8.745sMSEtwsMessage Struct Reference	808
8.745.1 Detailed Description	808
8.745.2 Field Documentation	809
8.745.2.1 data	809
8.745.2.2 length	809
8.745.2.3 notificationType	809
8.746SMSEtwsMessage Struct Reference	809
8.746.1 Detailed Description	809
8.746.2 Field Documentation	809
8.746.2.1 data	809
8.746.2.2 length	810
8.746.2.3 notificationType	810

8.747sMSEtwsMessageTlv Struct Reference	810
8.747.1 Detailed Description	810
8.747.2 Field Documentation	810
8.747.2.1 EtwsMessageInfo	810
8.747.2.2 TlvPresent	810
8.748sMSEtwsPlmn Struct Reference	810
8.748.1 Detailed Description	810
8.748.2 Field Documentation	811
8.748.2.1 mobileCountryCode	811
8.748.2.2 mobileNetworkCode	811
8.749SMSEtwsPlmn Struct Reference	811
8.749.1 Detailed Description	811
8.749.2 Field Documentation	811
8.749.2.1 mobileCountryCode	811
8.749.2.2 mobileNetworkCode	811
8.750SMSEventInfo_s Struct Reference	811
8.750.1 Detailed Description	812
8.750.2 Field Documentation	812
8.750.2.1 pEtwsMessageInfo	812
8.750.2.2 pEtwsPlmnInfo	812
8.750.2.3 pMessageModelInfo	812
8.750.2.4 pMTMessageInfo	812
8.750.2.5 pSMSCAddressInfo	812
8.750.2.6 pSMSOnIMSInfo	813
8.750.2.7 pTransferRouteMTMessageInfo	813
8.750.2.8 smsEventType	813
8.751smsMaxStorageSizeReq Struct Reference	813
8.751.1 Detailed Description	813
8.751.2 Field Documentation	813
8.751.2.1 pMessageMode	813

8.751.2.2 storageType	813
8.752smsMaxStorageSizeResp Struct Reference	813
8.752.1 Detailed Description	814
8.752.2 Field Documentation	814
8.752.2.1 freeSlots	814
8.752.2.2 maxStorageSize	814
8.753SMSMemoryInfo Struct Reference	814
8.753.1 Detailed Description	814
8.753.2 Field Documentation	814
8.753.2.1 messageMode	814
8.753.2.2 storageType	815
8.754sMSMessageMode Struct Reference	815
8.754.1 Detailed Description	815
8.754.2 Field Documentation	815
8.754.2.1 messageMode	815
8.755SMSMessageMode Struct Reference	815
8.755.1 Detailed Description	815
8.755.2 Field Documentation	815
8.755.2.1 messageMode	815
8.756smsMsgprotocolResp Struct Reference	815
8.756.1 Detailed Description	816
8.756.2 Field Documentation	816
8.756.2.1 msgProtocol	816
8.757sMSMTMessage Struct Reference	816
8.757.1 Detailed Description	816
8.757.2 Field Documentation	816
8.757.2.1 messageIndex	816
8.757.2.2 storageType	816
8.758SMSMTMessage Struct Reference	816
8.758.1 Detailed Description	817

8.758.2 Field Documentation	817
8.758.2.1 messageIndex	817
8.758.2.2 storageType	817
8.759SMSOnIMS Struct Reference	817
8.759.1 Detailed Description	817
8.759.2 Field Documentation	817
8.759.2.1 smsOnIMS	817
8.760sMSOnIMS Struct Reference	817
8.760.1 Detailed Description	818
8.760.2 Field Documentation	818
8.760.2.1 smsOnIMS	818
8.761sMSOnIMSTlv Struct Reference	818
8.761.1 Detailed Description	818
8.761.2 Field Documentation	818
8.761.2.1 IMSInfo	818
8.761.2.2 TlvPresent	818
8.762smsRouteEntry Struct Reference	818
8.762.1 Detailed Description	819
8.762.2 Field Documentation	819
8.762.2.1 messageClass	819
8.762.2.2 messageType	819
8.762.2.3 receiptAction	819
8.762.2.4 routeStorage	819
8.763smsSetRoutesReq Struct Reference	819
8.763.1 Detailed Description	820
8.763.2 Field Documentation	820
8.763.2.1 numOfRoutes	820
8.763.2.2 pTransferStatusReport	820
8.763.2.3 routeList	820
8.764sMSTransferRouteMTMessage Struct Reference	820

8.764.1 Detailed Description	820
8.764.2 Field Documentation	821
8.764.2.1 ackIndicator	821
8.764.2.2 data	821
8.764.2.3 format	821
8.764.2.4 length	821
8.764.2.5 transactionID	821
8.765SMSTransferRouteMTMessage Struct Reference	821
8.765.1 Detailed Description	821
8.765.2 Field Documentation	822
8.765.2.1 ackIndicator	822
8.765.2.2 data	822
8.765.2.3 format	822
8.765.2.4 length	822
8.765.2.5 transactionID	822
8.766sQosFlowStat Struct Reference	822
8.766.1 Detailed Description	822
8.766.2 Field Documentation	822
8.766.2.1 bearerId	822
8.766.2.2 tx_bytes	823
8.766.2.3 tx_bytes_drp	823
8.766.2.4 tx_pkt	823
8.766.2.5 tx_pkt_drp	823
8.767sQosStat Struct Reference	823
8.767.1 Detailed Description	823
8.767.2 Field Documentation	824
8.767.2.1 apnId	824
8.767.2.2 numQosFlow	824
8.767.2.3 qosFlow	824
8.767.2.4 total_rx_bytes	824

8.767.2.5 total_rx_pkt	824
8.767.2.6 total_tx_bytes	824
8.767.2.7 total_tx_bytes_drp	824
8.767.2.8 total_tx_pkt	824
8.767.2.9 total_tx_pkt_drp	824
8.768SrvStatusInfo Struct Reference	824
8.768.1 Detailed Description	825
8.768.2 Field Documentation	825
8.768.2.1 isPrefDataPath	825
8.768.2.2 srvStatus	825
8.769ssdatasession_params Struct Reference	825
8.769.1 Detailed Description	826
8.769.2 Field Documentation	826
8.769.2.1 action	826
8.769.2.2 failureReason	826
8.769.2.3 failureReasonv4	826
8.769.2.4 failureReasonv6	826
8.769.2.5 instanceId	826
8.769.2.6 ipfamily	826
8.769.2.7 pAuthentication	826
8.769.2.8 pPassword	826
8.769.2.9 pProfileId3GPP	826
8.769.2.10pProfileId3GPP2	827
8.769.2.11pTechnology	827
8.769.2.12pUsername	827
8.769.2.13cv4	827
8.769.2.14cv6	827
8.769.2.15sessionId	827
8.769.2.16v4sessionId	827
8.769.2.17v6sessionId	827

8.769.2.18verbFailReason	827
8.769.2.19verbFailReasonType	827
8.770SupportedMsgList Struct Reference	827
8.770.1 Detailed Description	827
8.770.2 Field Documentation	827
8.770.2.1 supportedMsgLen	827
8.770.2.2 supportedMsgs	828
8.771SUPSInfo Struct Reference	828
8.771.1 Detailed Description	828
8.771.2 Field Documentation	828
8.771.2.1 isModByCC	828
8.771.2.2 svcType	828
8.772SV Struct Reference	828
8.772.1 Detailed Description	829
8.772.2 Field Documentation	829
8.772.2.1 id	829
8.772.2.2 mask	829
8.772.2.3 system	829
8.773SVInfo Struct Reference	829
8.773.1 Detailed Description	829
8.773.2 Field Documentation	830
8.773.2.1 len	830
8.773.2.2 pSV	830
8.774svUsedforFix_s Struct Reference	830
8.774.1 Detailed Description	830
8.774.2 Field Documentation	830
8.774.2.1 gnssSvUsedList	830
8.774.2.2 gnssSvUsedList_len	831
8.775SWI_STRUCT_CarrierImage Struct Reference	831
8.775.1 Detailed Description	831

8.775.2 Field Documentation	831
8.775.2.1 m_FwBuildId	831
8.775.2.2 m_FwImagId	832
8.775.2.3 m_nCarrierId	832
8.775.2.4 m_nFolderId	832
8.775.2.5 m_nStorage	832
8.775.2.6 m_PriBuildId	832
8.775.2.7 m_PrImagId	832
8.776SwiLocGetAutoStartResp Struct Reference	832
8.776.1 Detailed Description	832
8.776.2 Field Documentation	833
8.776.2.1 fix_rate	833
8.776.2.2 fix_rate_reported	833
8.776.2.3 fix_type	833
8.776.2.4 fix_type_reported	833
8.776.2.5 function	833
8.776.2.6 function_reported	833
8.776.2.7 max_dist	833
8.776.2.8 max_dist_reported	833
8.776.2.9 max_time	833
8.776.2.10max_time_reported	834
8.777SwiLocSetAutoStartReq Struct Reference	834
8.777.1 Detailed Description	834
8.777.2 Field Documentation	835
8.777.2.1 fix_rate	835
8.777.2.2 fix_type	835
8.777.2.3 function	835
8.777.2.4 max_dist	835
8.777.2.5 max_time	835
8.777.2.6 set_fix_rate	835

8.777.2.7 set_fix_type	835
8.777.2.8 set_function	835
8.777.2.9 set_max_dist	835
8.777.2.10set_max_time	835
8.778swiModemStatusResp Struct Reference	835
8.778.1 Detailed Description	835
8.778.2 Field Documentation	836
8.778.2.1 commonInfo	836
8.778.2.2 pLTEInfo	836
8.779SwiOTAMsg_s Struct Reference	836
8.779.1 Detailed Description	836
8.779.2 Field Documentation	837
8.779.2.1 data	837
8.779.2.2 data_len	837
8.779.2.3 pLteNasRelInfo	837
8.779.2.4 pTime	837
8.779.2.5 type	837
8.780swiPDPRuntimeSettingsReq Struct Reference	837
8.780.1 Detailed Description	837
8.780.2 Field Documentation	837
8.780.2.1 contextId	837
8.780.2.2 contextType	837
8.781swiPDPRuntimeSettingsResp Struct Reference	837
8.781.1 Detailed Description	838
8.781.2 Field Documentation	839
8.781.2.1 pAPNName	839
8.781.2.2 pBearerId	839
8.781.2.3 pContextId	839
8.781.2.4 pIPv4Address	839
8.781.2.5 pIPv4GWAddress	839

8.781.2.6 pIPv6Address	839
8.781.2.7 pIPv6GWAddress	840
8.781.2.8 pPrDNSIPv4Address	840
8.781.2.9 pPrDNSIPv6Address	840
8.781.2.10pPrPCSCFIPv4Address	840
8.781.2.11pPrPCSCFIPv6Address	840
8.781.2.12pSeDNSIPv4Address	840
8.781.2.13pSeDNSIPv6Address	840
8.781.2.14pSePCSCFIPv4Address	840
8.781.2.15pSePCSCFIPv6Address	840
8.782swiQosFilter Struct Reference	840
8.782.1 Detailed Description	840
8.782.2 Field Documentation	842
8.782.2.1 index	842
8.782.2.2 pEspSpi	842
8.782.2.3 pld	842
8.782.2.4 pIPv4DstAddr	842
8.782.2.5 pIPv4SrcAddr	842
8.782.2.6 pIPv6DstAddr	842
8.782.2.7 pIPv6Label	842
8.782.2.8 pIPv6SrcAddr	842
8.782.2.9 pIPv6TrafCls	842
8.782.2.10pNxtHdrProto	842
8.782.2.11pPrecedence	842
8.782.2.12pTCPPDstPort	842
8.782.2.13pTCPSrcPort	842
8.782.2.14pTos	842
8.782.2.15pTranDstPort	842
8.782.2.16pTranSrcPort	842
8.782.2.17pUDPDstPort	842

8.782.2.18pUDPSrcPort	842
8.782.2.19version	842
8.783swiQosFlow Struct Reference	842
8.783.1 Detailed Description	843
8.783.2 Field Documentation	845
8.783.2.1 index	845
8.783.2.2 p3GPP2Pri	845
8.783.2.3 p3GPPImCn	845
8.783.2.4 p3GPPResResidualBER	845
8.783.2.5 p3GPPSigInd	845
8.783.2.6 p3GPPTraHdlPri	845
8.783.2.7 pDataRate	845
8.783.2.8 pJitter	845
8.783.2.9 pLatency	845
8.783.2.10pLteQci	845
8.783.2.11pMaxAllowedPktSz	845
8.783.2.12pMinPolicedPktSz	846
8.783.2.13pPktErrRate	846
8.783.2.14pProfileId3GPP2	846
8.783.2.15pTokenBucket	846
8.783.2.16pTrafficClass	846
8.784swiQosGranted Struct Reference	846
8.784.1 Detailed Description	846
8.784.2 Field Documentation	846
8.784.2.1 pRxFlow	846
8.784.2.2 pTxFlow	846
8.785swiQosIds Struct Reference	846
8.785.1 Detailed Description	846
8.785.2 Field Documentation	847
8.785.2.1 plds	847

8.785.2.2 sz	847
8.786swiQosModifyReq Struct Reference	847
8.786.1 Detailed Description	847
8.786.2 Field Documentation	847
8.786.2.1 id	847
8.786.2.2 pRxFilter	847
8.786.2.3 pRxFlow	847
8.786.2.4 pTxFilter	847
8.786.2.5 pTxFlow	847
8.787swiQosReq Struct Reference	848
8.787.1 Detailed Description	848
8.787.2 Field Documentation	848
8.787.2.1 index	848
8.787.2.2 pRxFilter	848
8.787.2.3 pRxFlow	848
8.787.2.4 pTxFilter	848
8.787.2.5 pTxFlow	848
8.788swiRMTrasnferStaticsReq Struct Reference	848
8.788.1 Detailed Description	849
8.788.2 Field Documentation	849
8.788.2.1 bResetStatistics	849
8.788.2.2 ulMask	849
8.789sysInfoCommon Struct Reference	849
8.789.1 Detailed Description	849
8.789.2 Field Documentation	851
8.789.2.1 isSysForbidden	851
8.789.2.2 isSysForbiddenValid	851
8.789.2.3 roamStatus	851
8.789.2.4 roamStatusValid	851
8.789.2.5 srvCapability	851

8.789.2.6 srvCapabilityValid	851
8.789.2.7 srvDomain	852
8.789.2.8 srvDomainValid	852
8.790t_gpsTime Struct Reference	852
8.790.1 Field Documentation	852
8.790.1.1 gpsTimeOfWeekMs	852
8.790.1.2 gpsWeek	852
8.791t_sensor Struct Reference	852
8.791.1 Field Documentation	852
8.791.1.1 aidingIndicatorMask	852
8.791.1.2 usageMask	852
8.792t_Sv Struct Reference	852
8.792.1 Field Documentation	852
8.792.1.1 entries	852
8.792.1.2 len	852
8.793TDSCDMAECIOThresh Struct Reference	852
8.793.1 Detailed Description	853
8.793.2 Field Documentation	853
8.793.2.1 pTDSCDMAECIOThreshList	853
8.793.2.2 TDSCDMAECIOThreshListLen	853
8.794TDSCDMARSCPThresh Struct Reference	853
8.794.1 Detailed Description	853
8.794.2 Field Documentation	853
8.794.2.1 pTDSCDMARSCPThreshList	853
8.794.2.2 TDSCDMARSCPThreshListLen	854
8.795TDSCDMARSSIThresh Struct Reference	854
8.795.1 Detailed Description	854
8.795.2 Field Documentation	854
8.795.2.1 pTDSCDMARSSIThreshList	854
8.795.2.2 TDSCDMARSSIThreshListLen	854

8.796TDSCDMASigInfoExt Struct Reference	854
8.796.1 Detailed Description	854
8.796.2 Field Documentation	855
8.796.2.1 ecio	855
8.796.2.2 rscp	855
8.796.2.3 rssi	855
8.796.2.4 sinr	855
8.797tdscdmaSigInfoExt Struct Reference	855
8.797.1 Detailed Description	855
8.797.2 Field Documentation	855
8.797.2.1 ecio	855
8.797.2.2 rscp	855
8.797.2.3 rssi	856
8.797.2.4 sinr	856
8.798TDSCDMASINRCONFThresh Struct Reference	856
8.798.1 Detailed Description	856
8.798.2 Field Documentation	856
8.798.2.1 pTDSCDMASINRCONFThreshList	856
8.798.2.2 TDSCDMASINRCONFThreshListLen	856
8.799TDSCDMASINRThresh Struct Reference	856
8.799.1 Detailed Description	856
8.799.2 Field Documentation	857
8.799.2.1 pTDSCDMASINRThreshList	857
8.799.2.2 TDSCDMASINRThreshListLen	857
8.800tempData_t Struct Reference	857
8.800.1 Detailed Description	857
8.800.2 Field Documentation	858
8.800.2.1 temperature	858
8.800.2.2 temperatureDataLen	858
8.800.2.3 timeOfFirstSample	858

8.800.2.4 timeOffset	858
8.800.2.5 timeSource	858
8.801 temperatureData Struct Reference	858
8.801.1 Detailed Description	858
8.801.2 Field Documentation	859
8.801.2.1 temperature	859
8.801.2.2 temperatureDataLen	859
8.801.2.3 timeOfFirstSample	859
8.801.2.4 timeOffset	859
8.801.2.5 timeSource	859
8.802 TFTIDParams Struct Reference	859
8.802.1 Detailed Description	859
8.802.2 Field Documentation	860
8.802.2.1 destPortRangeEnd	860
8.802.2.2 destPortRangeStart	860
8.802.2.3 eValid	860
8.802.2.4 filterId	860
8.802.2.5 flowLabel	860
8.802.2.6 IPSEC SPI	860
8.802.2.7 ipVersion	860
8.802.2.8 nextHeader	861
8.802.2.9 pSourceIP	861
8.802.2.10 sourceIPMask	861
8.802.2.11 srcPortRangeEnd	861
8.802.2.12 srcPortRangeStart	861
8.802.2.13 tosMask	861
8.803 timeInfo Struct Reference	861
8.803.1 Detailed Description	861
8.803.2 Field Documentation	862
8.803.2.1 day	862

8.803.2.2 dayLtSavingAdj	862
8.803.2.3 dayOfWeek	862
8.803.2.4 hour	862
8.803.2.5 minute	862
8.803.2.6 month	862
8.803.2.7 radiolInterface	863
8.803.2.8 second	863
8.803.2.9 timeZone	863
8.803.2.10TlvPresent	863
8.803.2.11year	863
8.804TmdDeRegNotMitigationLvlReq Struct Reference	863
8.804.1 Detailed Description	863
8.804.2 Field Documentation	863
8.804.2.1 mitigationDevID	863
8.804.2.2 mitigationDevIDLen	863
8.805TmdGetMitigationDevListResp Struct Reference	863
8.805.1 Detailed Description	864
8.805.2 Field Documentation	864
8.805.2.1 pMitigationDevList	864
8.805.2.2 pMitigationDevListLen	864
8.806TmdGetMitigationLvlReq Struct Reference	864
8.806.1 Detailed Description	864
8.806.2 Field Documentation	864
8.806.2.1 mitigationDevID	864
8.806.2.2 mitigationDevIDLen	865
8.807TmdGetMitigationLvlResp Struct Reference	865
8.807.1 Detailed Description	865
8.807.2 Field Documentation	865
8.807.2.1 pCurrentmitigationLvl	865
8.807.2.2 pReqMitigationLvl	865

8.808TmdMitigationLvIndReq Struct Reference	865
8.808.1 Detailed Description	865
8.808.2 Field Documentation	866
8.808.2.1 mitigationDevID	866
8.808.2.2 mitigationDevIDLen	866
8.809TmdRegNotMitigationLvReq Struct Reference	866
8.809.1 Detailed Description	866
8.809.2 Field Documentation	866
8.809.2.1 mitigationDevID	866
8.809.2.2 mitigationDevIDLen	866
8.810tokenBucket Struct Reference	866
8.810.1 Detailed Description	866
8.810.2 Field Documentation	867
8.810.2.1 bucketSz	867
8.810.2.2 peakRate	867
8.810.2.3 tokenRate	867
8.811Tos Struct Reference	867
8.811.1 Detailed Description	867
8.811.2 Field Documentation	867
8.811.2.1 mask	867
8.811.2.2 val	867
8.812transferRouteMessageTlv Struct Reference	867
8.812.1 Detailed Description	868
8.812.2 Field Documentation	868
8.812.2.1 TlvPresent	868
8.812.2.2 TransferRouteMTMessageInfo	868
8.813TransferStatInd Struct Reference	868
8.813.1 Detailed Description	868
8.813.2 Field Documentation	869
8.813.2.1 StatsMask	869

8.813.2.2 StatsPeriod	869
8.814transferStatInd Struct Reference	869
8.814.1 Detailed Description	869
8.814.2 Field Documentation	869
8.814.2.1 StatsMask	869
8.814.2.2 StatsPeriod	869
8.815TransferStatsDataType Struct Reference	869
8.815.1 Field Documentation	869
8.815.1.1 interval	869
8.816TrStatInd Struct Reference	869
8.816.1 Detailed Description	869
8.816.2 Field Documentation	870
8.816.2.1 statsMask	870
8.816.2.2 statsPeriod	870
8.817trueIMSI Struct Reference	870
8.817.1 Detailed Description	870
8.817.2 Field Documentation	871
8.817.2.1 imsiT1112	871
8.817.2.2 imsiTaddrNum	871
8.817.2.3 imsiTS1	871
8.817.2.4 imsiTS2	871
8.817.2.5 mccT	871
8.818TXAGCList Struct Reference	871
8.818.1 Detailed Description	871
8.818.2 Field Documentation	872
8.818.2.1 pTXAIG	872
8.818.2.2 pTXComprSlope	872
8.818.2.3 pTXComprThres	872
8.818.2.4 pTXExpSlope	872
8.818.2.5 pTXExpThres	872

8.818.2.6 pTXStaticGain	872
8.819txInfo Struct Reference	872
8.819.1 Detailed Description	872
8.819.2 Field Documentation	873
8.819.2.1 isInTraffic	873
8.819.2.2 txPower	873
8.820TXPCMIIRFtr Struct Reference	873
8.820.1 Detailed Description	873
8.820.2 Field Documentation	874
8.820.2.1 pFlag	874
8.820.2.2 pStage0Val	874
8.820.2.3 pStage1Val	874
8.820.2.4 pStage2Val	874
8.820.2.5 pStage3Val	874
8.820.2.6 pStage4Val	874
8.820.2.7 pStageCnt	874
8.821uim_appStatus Struct Reference	874
8.821.1 Detailed Description	875
8.821.2 Field Documentation	877
8.821.2.1 aidLength	877
8.821.2.2 aidVal	877
8.821.2.3 appState	877
8.821.2.4 appType	877
8.821.2.5 persoFeature	877
8.821.2.6 persoRetries	877
8.821.2.7 persoState	877
8.821.2.8 persoUnblockRetries	877
8.821.2.9 pin1Retries	877
8.821.2.10pin1State	877
8.821.2.11pin2Retries	877

8.821.2.12 pin2State	877
8.821.2.13 puk1Retries	877
8.821.2.14 puk2Retries	877
8.821.2.15 univPin	877
8.822uim_cardResult Struct Reference	877
8.822.1 Detailed Description	878
8.822.2 Field Documentation	878
8.822.2.1 sw1	878
8.822.2.2 sw2	878
8.823uim_cardStatus Struct Reference	878
8.823.1 Detailed Description	878
8.823.2 Field Documentation	879
8.823.2.1 index1xPri	879
8.823.2.2 index1xSec	879
8.823.2.3 indexGwPri	879
8.823.2.4 indexGwSec	879
8.823.2.5 numSlot	879
8.823.2.6 SlotInfo	879
8.824uim_changeUIMPIN Struct Reference	879
8.824.1 Detailed Description	879
8.824.2 Field Documentation	880
8.824.2.1 oldPINLen	880
8.824.2.2 oldPINVal	880
8.824.2.3 pinID	880
8.824.2.4 pinLen	880
8.824.2.5 pinVal	880
8.825uim_encryptedPIN1 Struct Reference	880
8.825.1 Detailed Description	880
8.825.2 Field Documentation	881
8.825.2.1 pin1Len	881

8.825.2.2 pin1Val	881
8.826uim_fileInfo Struct Reference	881
8.826.1 Detailed Description	881
8.826.2 Field Documentation	881
8.826.2.1 fileID	881
8.826.2.2 path	882
8.826.2.3 pathLen	882
8.827uim_hotSwapStatus Struct Reference	882
8.827.1 Detailed Description	882
8.827.2 Field Documentation	882
8.827.2.1 hotSwap	882
8.827.2.2 hotSwapLength	882
8.828uim_readResult Struct Reference	882
8.828.1 Detailed Description	882
8.828.2 Field Documentation	883
8.828.2.1 content	883
8.828.2.2 contentLen	883
8.829uim_readTransparentInfo Struct Reference	883
8.829.1 Detailed Description	883
8.829.2 Field Documentation	883
8.829.2.1 length	883
8.829.2.2 offset	883
8.830uim_remainingRetries Struct Reference	883
8.830.1 Detailed Description	884
8.830.2 Field Documentation	884
8.830.2.1 unblockLeft	884
8.830.2.2 verifyLeft	884
8.831uim_sessionInformation Struct Reference	884
8.831.1 Detailed Description	884
8.831.2 Field Documentation	885

8.831.2.1 aid	885
8.831.2.2 aidLength	885
8.831.2.3 sessionType	885
8.832uim_setPINProtection Struct Reference	885
8.832.1 Detailed Description	885
8.832.2 Field Documentation	886
8.832.2.1 pinID	886
8.832.2.2 pinLength	886
8.832.2.3 pinOperation	886
8.832.2.4 pinValue	886
8.833uim_slotInfo Struct Reference	886
8.833.1 Detailed Description	886
8.833.2 Field Documentation	887
8.833.2.1 AppStatus	887
8.833.2.2 cardState	887
8.833.2.3 errorState	887
8.833.2.4 numApp	887
8.833.2.5 upinRetries	887
8.833.2.6 upinState	887
8.833.2.7 upukRetries	887
8.834uim_UIMSessionInformation Struct Reference	887
8.834.1 Detailed Description	888
8.834.2 Field Documentation	888
8.834.2.1 aid	888
8.834.2.2 aidLength	888
8.834.2.3 sessionType	888
8.835uim_unblockUIMPIN Struct Reference	888
8.835.1 Detailed Description	888
8.835.2 Field Documentation	889
8.835.2.1 newPINLen	889

8.835.2.2 newPINVal	889
8.835.2.3 pinID	889
8.835.2.4 pukLen	889
8.835.2.5 pukVal	889
8.836uim_verifyUIMPIN Struct Reference	889
8.836.1 Detailed Description	889
8.836.2 Field Documentation	890
8.836.2.1 pinID	890
8.836.2.2 pinLen	890
8.836.2.3 pinVal	890
8.837UIMAuthenticateReq Struct Reference	890
8.837.1 Detailed Description	890
8.837.2 Field Documentation	891
8.837.2.1 authData	891
8.837.2.2 pIndicationToken	891
8.837.2.3 sessionInfo	891
8.838UIMAuthenticateResp Struct Reference	891
8.838.1 Detailed Description	891
8.838.2 Field Documentation	891
8.838.2.1 pAuthenticateResult	891
8.838.2.2 pCardResult	891
8.838.2.3 pIndicationToken	892
8.839UIMChangePinReq Struct Reference	892
8.839.1 Detailed Description	892
8.839.2 Field Documentation	892
8.839.2.1 changePIN	892
8.839.2.2 pIndicationToken	892
8.839.2.3 pKeyReferenceID	892
8.839.2.4 sessionInfo	892
8.840UIMDepersonalizationReq Struct Reference	893

8.840.1 Detailed Description	893
8.840.2 Field Documentation	893
8.840.2.1 depersonalisationInfo	893
8.841 UIMDepersonalizationResp Struct Reference	893
8.841.1 Detailed Description	893
8.841.2 Field Documentation	893
8.841.2.1 pRemainingRetries	893
8.842 UIMEventRegisterReqResp Struct Reference	894
8.842.1 Detailed Description	894
8.842.2 Field Documentation	894
8.842.2.1 eventMask	894
8.843 UIMGetCardStatusResp Struct Reference	894
8.843.1 Detailed Description	894
8.843.2 Field Documentation	895
8.843.2.1 pCardStatus	895
8.843.2.2 pHotSwapStatus	895
8.844 UIMGetConfigurationReq Struct Reference	895
8.844.1 Detailed Description	895
8.844.2 Field Documentation	895
8.844.2.1 pConfigurationMask	895
8.845 UIMGetConfigurationResp Struct Reference	895
8.845.1 Detailed Description	896
8.845.2 Field Documentation	896
8.845.2.1 pAutoSelection	896
8.845.2.2 pHaltSubscription	896
8.845.2.3 pPersonalizationStatus	896
8.846 UIMGetFileAttributesReq Struct Reference	896
8.846.1 Detailed Description	896
8.846.2 Field Documentation	897
8.846.2.1 fileIndex	897

8.846.2.2 pIndicationToken	897
8.846.2.3 sessionInfo	897
8.847UIMGetFileAttributesResp Struct Reference	897
8.847.1 Detailed Description	897
8.847.2 Field Documentation	897
8.847.2.1 pCardResult	897
8.847.2.2 pFileAttributes	897
8.847.2.3 pIndicationToken	898
8.848UIMGetSlotsStatusResp Struct Reference	898
8.848.1 Detailed Description	898
8.848.2 Field Documentation	898
8.848.2.1 pNumberOfPhySlot	898
8.848.2.2 pUimSlotsStatus	898
8.849UIMPinResp Struct Reference	898
8.849.1 Detailed Description	898
8.849.2 Field Documentation	899
8.849.2.1 pEncryptedPIN1	899
8.849.2.2 pIndicationToken	899
8.849.2.3 pRemainingRetries	899
8.850UIMPowerDownReq Struct Reference	899
8.850.1 Detailed Description	899
8.850.2 Field Documentation	899
8.850.2.1 slot	899
8.851UIMPowerUpReq Struct Reference	900
8.851.1 Detailed Description	900
8.851.2 Field Documentation	900
8.851.2.1 pIgnoreHotSwapSwitch	900
8.851.2.2 slot	900
8.852UIMReadTransparentReq Struct Reference	900
8.852.1 Detailed Description	900

8.852.2 Field Documentation	901
8.852.2.1 fileIndex	901
8.852.2.2 pEncryptData	901
8.852.2.3 pIndicationToken	901
8.852.2.4 readTransparent	901
8.852.2.5 sessionInfo	901
8.853UIMReadTransparentResp Struct Reference	901
8.853.1 Detailed Description	901
8.853.2 Field Documentation	902
8.853.2.1 pCardResult	902
8.853.2.2 pEncryptedData	902
8.853.2.3 pIndicationToken	902
8.853.2.4 pReadResult	902
8.854UIMRefreshCompleteReq Struct Reference	902
8.854.1 Detailed Description	902
8.854.2 Field Documentation	903
8.854.2.1 refreshComplete	903
8.854.2.2 sessionInfo	903
8.855UIMRefreshEvent Struct Reference	903
8.855.1 Detailed Description	903
8.855.2 Field Documentation	904
8.855.2.1 aid	904
8.855.2.2 aidLength	904
8.855.2.3 arrfileInfo	904
8.855.2.4 mode	904
8.855.2.5 numOfFiles	904
8.855.2.6 sessionType	904
8.855.2.7 stage	904
8.856UIMRefreshGetLastEventReq Struct Reference	904
8.856.1 Detailed Description	904

8.856.2 Field Documentation	905
8.856.2.1 sessionInfo	905
8.857UIMRefreshGetLastEventResp Struct Reference	905
8.857.1 Detailed Description	905
8.857.2 Field Documentation	905
8.857.2.1 pRefreshEvent	905
8.858UIMRefreshOKReq Struct Reference	905
8.858.1 Detailed Description	905
8.858.2 Field Documentation	906
8.858.2.1 OKtoRefresh	906
8.858.2.2 sessionInfo	906
8.859UIMRefreshRegisterReq Struct Reference	906
8.859.1 Detailed Description	906
8.859.2 Field Documentation	906
8.859.2.1 regRefresh	906
8.859.2.2 sessionInfo	906
8.860UIMSessionInformation Struct Reference	906
8.860.1 Detailed Description	907
8.860.2 Field Documentation	907
8.860.2.1 aid	907
8.860.2.2 aidLength	907
8.860.2.3 sessionType	907
8.861UIMSetPinProtectionReq Struct Reference	907
8.861.1 Detailed Description	908
8.861.2 Field Documentation	908
8.861.2.1 pIndicationToken	908
8.861.2.2 pinProtection	908
8.861.2.3 pKeyReferenceID	908
8.861.2.4 sessionInfo	908
8.862UIMSlotsStatus Struct Reference	908

8.862.1 Detailed Description	908
8.862.2 Field Documentation	909
8.862.2.1 uimSlotStatus	909
8.863UIMSlotStatus Struct Reference	909
8.863.1 Detailed Description	909
8.863.2 Field Documentation	910
8.863.2.1 bICCID	910
8.863.2.2 bICCIDLength	910
8.863.2.3 bLogicalSlot	910
8.863.2.4 uPhyCardStatus	910
8.863.2.5 uPhySlotStatus	910
8.864UIMSlotStatusChangeInfo Struct Reference	910
8.864.1 Detailed Description	910
8.864.2 Field Documentation	910
8.864.2.1 bNumberOfPhySlots	910
8.864.2.2 slotsstatusChange	910
8.865UIMStatusChangeInfo Struct Reference	910
8.865.1 Detailed Description	911
8.865.2 Field Documentation	911
8.865.2.1 statusChange	911
8.866UIMSwitchSlotReq Struct Reference	911
8.866.1 Detailed Description	911
8.866.2 Field Documentation	911
8.866.2.1 bLogicalSlot	911
8.866.2.2 ulPhysicalSlot	912
8.867UIMUnblockPinReq Struct Reference	912
8.867.1 Detailed Description	912
8.867.2 Field Documentation	912
8.867.2.1 pIndicationToken	912
8.867.2.2 pKeyReferenceID	912

8.867.2.3 sessionInfo	912
8.867.2.4 unblockPIN	912
8.868UIMVerifyPinReq Struct Reference	913
8.868.1 Detailed Description	913
8.868.2 Field Documentation	913
8.868.2.1 pEncryptedPIN1	913
8.868.2.2 pIndicationToken	913
8.868.2.3 pKeyReferenceID	913
8.868.2.4 sessionInfo	913
8.868.2.5 verifyPIN	914
8.869UMTSInfo Struct Reference	914
8.869.1 Detailed Description	914
8.869.2 Field Documentation	915
8.869.2.1 cellID	915
8.869.2.2 ecio	915
8.869.2.3 geranInst	915
8.869.2.4 GeranInstInfo	915
8.869.2.5 lac	915
8.869.2.6 plmn	915
8.869.2.7 psc	915
8.869.2.8 rscp	915
8.869.2.9 uarfcn	915
8.869.2.10umtsInst	915
8.869.2.11UMTSInstInfo	915
8.870UMTSinstInfo Struct Reference	915
8.870.1 Detailed Description	916
8.870.2 Field Documentation	916
8.870.2.1 umtsEcio	916
8.870.2.2 umtsPsc	916
8.870.2.3 umtsRscp	916

8.870.2.4 umtsUarfcn	916
8.871 umtsLTENbrCell Struct Reference	916
8.871.1 Detailed Description	916
8.871.2 Field Documentation	917
8.871.2.1 cellIsTDD	917
8.871.2.2 earfcn	917
8.871.2.3 pci	917
8.871.2.4 rsrp	917
8.871.2.5 rsrq	917
8.871.2.6 srxlev	917
8.872UMTSMinQoS Struct Reference	917
8.872.1 Detailed Description	918
8.872.2 Field Documentation	919
8.872.2.1 deliveryErrSDU	919
8.872.2.2 grntDownlinkBitrate	919
8.872.2.3 grntUplinkBitrate	919
8.872.2.4 maxDownlinkBitrate	919
8.872.2.5 maxSDUSize	919
8.872.2.6 maxUplinkBitrate	919
8.872.2.7 qosDeliveryOrder	920
8.872.2.8 resBerRatio	920
8.872.2.9 sduErrorRatio	920
8.872.2.10trafficClass	920
8.872.2.11trafficPriority	920
8.872.2.12transferDelay	920
8.873UMTSQoS Struct Reference	920
8.873.1 Detailed Description	920
8.873.2 Field Documentation	922
8.873.2.1 deliveryErrSDU	922
8.873.2.2 grntDownlinkBitrate	922

8.873.2.3 grntUplinkBitrate	922
8.873.2.4 maxDownlinkBitrate	922
8.873.2.5 maxSDUSize	922
8.873.2.6 maxUplinkBitrate	922
8.873.2.7 qosDeliveryOrder	922
8.873.2.8 resBerRatio	922
8.873.2.9 sduErrorRatio	922
8.873.2.10 trafficClass	922
8.873.2.11 trafficPriority	922
8.873.2.12 transferDelay	922
8.874UMTSReqQoS SigInd Struct Reference	922
8.874.1 Detailed Description	923
8.874.2 Field Documentation	923
8.874.2.1 SigInd	923
8.874.2.2 UMTSReqQoS	923
8.875unlockUIMPIN Struct Reference	923
8.875.1 Detailed Description	923
8.875.2 Field Documentation	924
8.875.2.1 newPINLen	924
8.875.2.2 newPINVal	924
8.875.2.3 pinID	924
8.875.2.4 pukLen	924
8.875.2.5 pukVal	924
8.876UniversalTime Struct Reference	924
8.876.1 Detailed Description	924
8.876.2 Field Documentation	925
8.876.2.1 day	925
8.876.2.2 dayOfWeek	925
8.876.2.3 hour	925
8.876.2.4 minute	925

8.876.2.5 month	925
8.876.2.6 second	925
8.876.2.7 year	925
8.877unpack_dms_GetActivationState_t Struct Reference	925
8.877.1 Detailed Description	926
8.877.2 Field Documentation	926
8.877.2.1 state	926
8.878unpack_dms_GetBandCapability_t Struct Reference	926
8.878.1 Field Documentation	926
8.878.1.1 BandCapability	926
8.878.1.2 Tlvresult	926
8.879unpack_dms_GetCrashAction_t Struct Reference	926
8.879.1 Field Documentation	927
8.879.1.1 DevCrashState	927
8.879.1.2 Tlvresult	927
8.880unpack_dms_GetCustFeature_t Struct Reference	927
8.880.1 Field Documentation	927
8.880.1.1 DHCPRelayEnabled	927
8.880.1.2 DisableIMSI	927
8.880.1.3 GpsEnable	927
8.880.1.4 GPSLPM	927
8.880.1.5 GPSSel	927
8.880.1.6 IPFamSupport	927
8.880.1.7 IsVoiceEnabled	927
8.880.1.8 RMAutoConnect	927
8.880.1.9 SMSSupport	927
8.880.1.10Tlvresult	927
8.881unpack_dms_GetCustFeaturesV2_t Struct Reference	927
8.881.1 Detailed Description	927
8.881.2 Field Documentation	928

8.881.2.1 GetCustomFeatureV2	928
8.881.2.2 Tlvresult	928
8.882unpack_dms_GetDeviceCap_t Struct Reference	928
8.882.1 Field Documentation	928
8.882.1.1 DataServiceCapability	928
8.882.1.2 MaxRXChannelRate	928
8.882.1.3 MaxTXChannelRate	928
8.882.1.4 Radiolfaces	928
8.882.1.5 RadiolfacesSize	928
8.882.1.6 SimCapability	928
8.882.1.7 Tlvresult	928
8.883unpack_dms_GetDeviceCapabilities_t Struct Reference	928
8.883.1 Detailed Description	928
8.883.2 Field Documentation	929
8.883.2.1 dataServiceCaCapability	929
8.883.2.2 maxRxChannelRate	929
8.883.2.3 maxTxChannelRate	929
8.883.2.4 Radiolfaces	929
8.883.2.5 radiolfacesSize	929
8.883.2.6 simCapability	929
8.884unpack_dms_GetDeviceHardwareRev_t Struct Reference	929
8.884.1 Field Documentation	929
8.884.1.1 String	929
8.884.1.2 stringSize	929
8.884.1.3 Tlvresult	929
8.885unpack_dms_GetDeviceMfr_t Struct Reference	929
8.885.1 Field Documentation	930
8.885.1.1 String	930
8.885.1.2 stringSize	930
8.885.1.3 Tlvresult	930

8.886unpack_dms_GetDeviceSerialNumbers_t Struct Reference	930
8.886.1 Field Documentation	930
8.886.1.1 esnSize	930
8.886.1.2 ESNString	930
8.886.1.3 imeiSize	930
8.886.1.4 IMEIString	930
8.886.1.5 imeiSvnSize	930
8.886.1.6 ImeiSvnString	930
8.886.1.7 meidSize	930
8.886.1.8 MEIDString	930
8.886.1.9 Tlvresult	930
8.887unpack_dms_GetFirmwareInfo_t Struct Reference	930
8.887.1 Detailed Description	931
8.887.2 Field Documentation	931
8.887.2.1 appversion_str	931
8.887.2.2 bootversion_str	931
8.887.2.3 carrier_str	931
8.887.2.4 cur_carr_name	931
8.887.2.5 cur_carr_rev	931
8.887.2.6 modelid_str	931
8.887.2.7 packageid_str	931
8.887.2.8 priversion_str	931
8.887.2.9 sku_str	931
8.887.2.10Tlvresult	931
8.888unpack_dms_GetFirmwareRevision_t Struct Reference	931
8.888.1 Field Documentation	932
8.888.1.1 amssSize	932
8.888.1.2 AMSSString	932
8.888.1.3 PRIString	932
8.888.1.4 Tlvresult	932

8.889unpack_dms_GetFirmwareRevisions_t Struct Reference	932
8.889.1 Detailed Description	932
8.889.2 Field Documentation	932
8.889.2.1 amssSize	932
8.889.2.2 AMSSString	932
8.889.2.3 bootSize	932
8.889.2.4 BootString	932
8.889.2.5 priSize	932
8.889.2.6 PRIString	932
8.889.2.7 Tlvresult	932
8.890unpack_dms_GetFSN_t Struct Reference	933
8.890.1 Field Documentation	933
8.890.1.1 String	933
8.890.1.2 Tlvresult	933
8.891unpack_dms_GetHardwareRevision_t Struct Reference	933
8.891.1 Detailed Description	933
8.891.2 Field Documentation	933
8.891.2.1 hwVer	933
8.892unpack_dms_GetIMSI_t Struct Reference	933
8.892.1 Field Documentation	933
8.892.1.1 imsi	933
8.892.1.2 Tlvresult	933
8.893unpack_dms_GetManufacturer_t Struct Reference	933
8.893.1 Detailed Description	934
8.893.2 Field Documentation	934
8.893.2.1 manufacturer	934
8.893.2.2 Tlvresult	934
8.894unpack_dms_GetModelID_t Struct Reference	934
8.894.1 Detailed Description	934
8.894.2 Field Documentation	934

8.894.2.1 modelid	934
8.894.2.2 Tlvresult	934
8.895unpack_dms_GetNetworkTime_t Struct Reference	934
8.895.1 Detailed Description	935
8.895.2 Field Documentation	935
8.895.2.1 source	935
8.895.2.2 timestamp	935
8.895.2.3 Tlvresult	935
8.896unpack_dms_GetOfflineReason_t Struct Reference	935
8.896.1 Detailed Description	935
8.896.2 Field Documentation	936
8.896.2.1 pbPlatform	936
8.896.2.2 pReasonMask	936
8.896.2.3 Tlvresult	936
8.897unpack_dms_GetPower_t Struct Reference	936
8.897.1 Detailed Description	936
8.897.2 Field Documentation	936
8.897.2.1 HardwareControlledMode	937
8.897.2.2 OfflineReason	937
8.897.2.3 OperationMode	937
8.897.2.4 Tlvresult	937
8.898unpack_dms_GetPRLVersion_t Struct Reference	937
8.898.1 Field Documentation	937
8.898.1.1 Tlvresult	937
8.898.1.2 u16PRLVersion	937
8.898.1.3 u8PRLPreference	937
8.899unpack_dms_GetSerialNumbers_t Struct Reference	937
8.899.1 Detailed Description	937
8.899.2 Field Documentation	937
8.899.2.1 esn	937

8.899.2.2 imei_no	938
8.899.2.3 imeisv_svn	938
8.899.2.4 meid	938
8.900unpack_dms_GetUSBComp_t Struct Reference	938
8.900.1 Field Documentation	938
8.900.1.1 NumSupUSBComps	938
8.900.1.2 SupUSBComps	938
8.900.1.3 Tlvresult	938
8.900.1.4 USBComp	938
8.901unpack_dms_GetVoiceNumber_t Struct Reference	938
8.901.1 Field Documentation	938
8.901.1.1 MIN	938
8.901.1.2 minSize	938
8.901.1.3 Tlvresult	938
8.901.1.4 VoiceNumber	938
8.901.1.5 voiceNumberSize	938
8.902unpack_dms_ResetToFactoryDefaults_t Struct Reference	938
8.902.1 Detailed Description	939
8.902.2 Field Documentation	939
8.902.2.1 Tlvresult	939
8.903unpack_dms_SetActivationStatusCallback_t Struct Reference	939
8.903.1 Detailed Description	939
8.903.2 Field Documentation	939
8.903.2.1 Tlvresult	939
8.904unpack_dms_SetCrashAction_t Struct Reference	939
8.904.1 Detailed Description	939
8.904.2 Field Documentation	940
8.904.2.1 notused	940
8.905unpack_dms_SetCustFeature_t Struct Reference	940
8.905.1 Field Documentation	940

8.905.1.1 Tlvresult	940
8.906unpack_dms_SetCustFeaturesV2_t Struct Reference	940
8.906.1 Detailed Description	940
8.906.2 Field Documentation	940
8.906.2.1 Tlvresult	940
8.907unpack_dms_SetEventReport_ind_t Struct Reference	940
8.907.1 Detailed Description	941
8.907.2 Field Documentation	941
8.907.2.1 ActivationStatusTlv	941
8.907.2.2 OperatingModeTlv	941
8.907.2.3 Tlvresult	941
8.908unpack_dms_SetEventReport_t Struct Reference	941
8.908.1 Field Documentation	941
8.908.1.1 Tlvresult	941
8.909unpack_dms_SetFirmwarePreference_t Struct Reference	941
8.909.1 Field Documentation	941
8.909.1.1 Tlvresult	941
8.910unpack_dms_SetPower_t Struct Reference	941
8.910.1 Field Documentation	942
8.910.1.1 Tlvresult	942
8.911unpack_dms_SetUSBComp_t Struct Reference	942
8.911.1 Field Documentation	942
8.911.1.1 Tlvresult	942
8.912unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t Struct Reference	942
8.912.1 Detailed Description	942
8.912.2 Field Documentation	942
8.912.2.1 source	942
8.912.2.2 Tlvresult	942
8.912.2.3 type	942
8.913unpack_dms_SLQSDmsSwiGetResetInfo_t Struct Reference	942

8.913.1 Detailed Description	943
8.913.2 Field Documentation	943
8.913.2.1 source	943
8.913.2.2 Tlvresult	943
8.913.2.3 type	943
8.914unpack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference	943
8.914.1 Detailed Description	943
8.914.2 Field Documentation	943
8.914.2.1 Tlvresult	943
8.915unpack_dms_SLQSGetBandCapability_t Struct Reference	943
8.915.1 Detailed Description	944
8.915.2 Field Documentation	946
8.915.2.1 bandCapability	946
8.915.2.2 is_LteBandCapability_Available	946
8.915.2.3 is_TdsBandCapability_Available	946
8.915.2.4 LteBandCapability	946
8.915.2.5 TdsBandCapability	946
8.916unpack_dms_SLQSGetERIFile_t Struct Reference	946
8.916.1 Detailed Description	946
8.916.2 Field Documentation	946
8.916.2.1 eriFile	946
8.916.2.2 Tlvresult	946
8.917unpack_dms_SLQSSwiClearDyingGaspStatistics_t Struct Reference	947
8.917.1 Detailed Description	947
8.917.2 Field Documentation	947
8.917.2.1 Tlvresult	947
8.918unpack_dms_SLQSSwiGetCrashInfo_t Struct Reference	947
8.918.1 Detailed Description	947
8.918.2 Field Documentation	947
8.918.2.1 crashInfoParam	947

8.918.2.2 Tlvresult	947
8.919unpack_dms_SLQSSwiGetDyingGaspCfg_t Struct Reference	948
8.919.1 Detailed Description	948
8.919.2 Field Documentation	948
8.919.2.1 pGetDyingGaspCfg	948
8.919.2.2 Tlvresult	948
8.920unpack_dms_SLQSSwiGetDyingGaspStatistics_t Struct Reference	948
8.920.1 Detailed Description	948
8.920.2 Field Documentation	948
8.920.2.1 pGetDyingGaspStatistics	948
8.920.2.2 Tlvresult	949
8.921unpack_dms_SLQSSwiGetFirmwareCurr_t Struct Reference	949
8.921.1 Detailed Description	949
8.921.2 Field Documentation	949
8.921.2.1 carrier	949
8.921.2.2 fwvers	949
8.921.2.3 numEntries	949
8.921.2.4 pCurrImgInfo	949
8.921.2.5 pkgver	949
8.921.2.6 priver	950
8.922unpack_dms_SLQSSwiGetFwUpdateStatus_t Struct Reference	950
8.922.1 Detailed Description	950
8.922.2 Field Documentation	951
8.922.2.1 imgType	951
8.922.2.2 logString	951
8.922.2.3 refData	951
8.922.2.4 refString	951
8.922.2.5 ResCode	951
8.922.2.6 Tlvresult	951
8.923unpack_dms_SLQSSwiGetHostDevInfo_t Struct Reference	951

8.923.1 Detailed Description	951
8.923.2 Field Documentation	952
8.923.2.1 manString	952
8.923.2.2 modelString	952
8.923.2.3 plasmaIDString	952
8.923.2.4 swVerString	952
8.923.2.5 Tlvresult	952
8.924unpack_dms_SLQSSwiGetOSInfo_t Struct Reference	952
8.924.1 Detailed Description	952
8.924.2 Field Documentation	952
8.924.2.1 nameString	952
8.924.2.2 Tlvresult	952
8.924.2.3 versionString	952
8.925unpack_dms_SLQSSwiGetSerialNoExt_t Struct Reference	952
8.925.1 Detailed Description	953
8.925.2 Field Documentation	953
8.925.2.1 meidString	953
8.925.2.2 Tlvresult	953
8.926unpack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference	953
8.926.1 Detailed Description	953
8.926.2 Field Documentation	953
8.926.2.1 Tlvresult	953
8.927unpack_dms_SLQSSwiSetHostDevInfo_t Struct Reference	953
8.927.1 Detailed Description	954
8.927.2 Field Documentation	954
8.927.2.1 Tlvresult	954
8.928unpack_dms_SLQSSwiSetOSInfo_t Struct Reference	954
8.928.1 Detailed Description	954
8.928.2 Field Documentation	954
8.928.2.1 Tlvresult	954

8.929unpack_dms_SLQSUIMGetState_t Struct Reference	954
8.929.1 Detailed Description	955
8.929.2 Field Documentation	955
8.929.2.1 state	955
8.929.2.2 Tlvresult	955
8.930unpack_dms_UIMGetControlKeyStatus_t Struct Reference	955
8.930.1 Detailed Description	955
8.930.2 Field Documentation	956
8.930.2.1 facilityState	956
8.930.2.2 Tlvresult	956
8.930.2.3 unblockRetriesLeft	956
8.930.2.4 verifyRetriesLeft	956
8.931unpack_dms_UIMGetICCID_t Struct Reference	956
8.931.1 Detailed Description	956
8.931.2 Field Documentation	956
8.931.2.1 String	956
8.931.2.2 stringSize	956
8.931.2.3 Tlvresult	956
8.932unpack_dms_UIMGetPINStatus_t Struct Reference	956
8.932.1 Detailed Description	957
8.932.2 Field Documentation	958
8.932.2.1 p1Status	958
8.932.2.2 p1UnblockRetriesLeft	958
8.932.2.3 p1VerifyRetriesLeft	958
8.932.2.4 p2Status	958
8.932.2.5 p2UnblockRetriesLeft	958
8.932.2.6 p2VerifyRetriesLeft	958
8.932.2.7 Tlvresult	958
8.933unpack_dms_UIMSetControlKeyProtection_t Struct Reference	958
8.933.1 Detailed Description	958

8.933.2 Field Documentation	959
8.933.2.1 Tlvresult	959
8.933.2.2 verifyRetriesLeft	959
8.934unpack_dms_UIMSetPINProtection_t Struct Reference	959
8.934.1 Detailed Description	959
8.934.2 Field Documentation	959
8.934.2.1 Tlvresult	959
8.934.2.2 unblockRetriesLeft	959
8.934.2.3 verifyRetriesLeft	959
8.935unpack_dms_UIMUnblockControlKey_t Struct Reference	959
8.935.1 Detailed Description	960
8.935.2 Field Documentation	960
8.935.2.1 Tlvresult	960
8.935.2.2 unblockRetriesLeft	960
8.936unpack_fms_GetImagesPreference_t Struct Reference	960
8.936.1 Detailed Description	960
8.936.2 Field Documentation	960
8.936.2.1 ImageListSize	961
8.936.2.2 plmageList	961
8.936.2.3 Tlvresult	961
8.937unpack_fms_GetStoredImages_t Struct Reference	961
8.937.1 Detailed Description	961
8.937.2 Field Documentation	961
8.937.2.1 imageList	961
8.937.2.2 imagelistSize	961
8.937.2.3 Tlvresult	961
8.938unpack_fms_SetImagesPreference_t Struct Reference	961
8.938.1 Detailed Description	962
8.938.2 Field Documentation	962
8.938.2.1 ImageTypes	962

8.938.2.2 ImageTypesSize	962
8.938.2.3 Tlvresult	962
8.939unpack_loc_BestAvailPos_Ind_t Struct Reference	962
8.939.1 Detailed Description	963
8.939.2 Field Documentation	966
8.939.2.1 pAltitudeWrtEllipsoid	966
8.939.2.2 pAltitudeWrtMeanSeaLevel	967
8.939.2.3 pGpsTime	967
8.939.2.4 pHeading	967
8.939.2.5 pHeadingUnc	967
8.939.2.6 pHorCirConf	967
8.939.2.7 pHorEllpConf	967
8.939.2.8 pHorReliability	967
8.939.2.9 pHorUncCircular	967
8.939.2.10pHorUncEllipseOrientAzimuth	967
8.939.2.11pHorUncEllipseSemiMajor	967
8.939.2.12pHorUncEllipseSemiMinor	967
8.939.2.13pLatitude	967
8.939.2.14pLongitude	967
8.939.2.15pMagneticDeviation	967
8.939.2.16pPrecisionDilution	967
8.939.2.17pSensorDataUsage	967
8.939.2.18pSpeedHorizontal	967
8.939.2.19pSpeedUnc	967
8.939.2.20pSpeedVertical	967
8.939.2.21pSpeedVerticalUnc	967
8.939.2.22pSvUsedforFix	967
8.939.2.23pTechnologyMask	967
8.939.2.24pTimeSrc	967
8.939.2.25pTimestampUtc	968

8.939.2.26	pTimeUnc	968
8.939.2.27	pVertConfidence	968
8.939.2.28	pVertReliability	968
8.939.2.29	pVertUnc	968
8.939.2.30	pXid	968
8.939.2.31	status	968
8.939.2.32	Tlvresult	968
8.940	unpack_loc_Delete_Assist_Data_t Struct Reference	968
8.940.1	Detailed Description	968
8.940.2	Field Documentation	968
8.940.2.1	Tlvresult	968
8.941	unpack_loc_DeleteAssistData_Ind_t Struct Reference	968
8.941.1	Detailed Description	969
8.941.2	Field Documentation	969
8.941.2.1	status	969
8.941.2.2	Tlvresult	969
8.942	unpack_loc_EngineState_Ind_t Struct Reference	969
8.942.1	Detailed Description	969
8.942.2	Field Documentation	970
8.942.2.1	engineState	970
8.942.2.2	Tlvresult	970
8.943	unpack_loc_EventRegister_t Struct Reference	970
8.943.1	Detailed Description	970
8.943.2	Field Documentation	970
8.943.2.1	Tlvresult	970
8.944	unpack_loc_GnssSvInfo_Ind_t Struct Reference	970
8.944.1	Detailed Description	971
8.944.2	Field Documentation	971
8.944.2.1	altitudeAssumed	971
8.944.2.2	pSatelliteInfo	971

8.944.2.3 Tlvresult	971
8.945unpack_loc_PositionRpt_Ind_t Struct Reference	971
8.945.1 Detailed Description	972
8.945.2 Field Documentation	976
8.945.2.1 pAltitudeAssumed	976
8.945.2.2 pAltitudeWrtEllipsoid	976
8.945.2.3 pAltitudeWrtMeanSeaLevel	976
8.945.2.4 pFixId	976
8.945.2.5 pGpsTime	976
8.945.2.6 pHeading	976
8.945.2.7 pHeadingUnc	976
8.945.2.8 pHorConfidence	976
8.945.2.9 pHorReliability	976
8.945.2.10pHorUncCircular	976
8.945.2.11pHorUncEllipseOrientAzimuth	976
8.945.2.12pHorUncEllipseSemiMajor	976
8.945.2.13pHorUncEllipseSemiMinor	976
8.945.2.14pLatitude	976
8.945.2.15pLeapSeconds	976
8.945.2.16pLongitude	976
8.945.2.17pMagneticDeviation	976
8.945.2.18pPrecisionDilution	976
8.945.2.19pSensorDataUsage	976
8.945.2.20pSpeedHorizontal	977
8.945.2.21pSpeedUnc	977
8.945.2.22pSpeedVertical	977
8.945.2.23pSvUsedforFix	977
8.945.2.24pTechnologyMask	977
8.945.2.25pTimeSrc	977
8.945.2.26pTimestampUtc	977

8.945.2.27pTimeUnc	977
8.945.2.28pVertConfidence	977
8.945.2.29pVertReliability	977
8.945.2.30pVertUnc	977
8.945.2.31sessionId	977
8.945.2.32sessionStatus	977
8.945.2.33Tlvresult	977
8.946unpack_loc_SetExtPowerConfig_Ind_t Struct Reference	977
8.946.1 Detailed Description	978
8.946.2 Field Documentation	978
8.946.2.1 status	978
8.946.2.2 Tlvresult	978
8.947unpack_loc_SetExtPowerState_t Struct Reference	978
8.947.1 Detailed Description	978
8.947.2 Field Documentation	979
8.947.2.1 Tlvresult	979
8.948unpack_loc_SetOperationMode_Ind_t Struct Reference	979
8.948.1 Detailed Description	979
8.948.2 Field Documentation	979
8.948.2.1 status	979
8.948.2.2 Tlvresult	979
8.949unpack_loc_SetOperationMode_t Struct Reference	979
8.949.1 Detailed Description	980
8.949.2 Field Documentation	980
8.949.2.1 Tlvresult	980
8.950unpack_loc_SLQSLOCGetBestAvailPos_t Struct Reference	980
8.950.1 Detailed Description	980
8.950.2 Field Documentation	980
8.950.2.1 Tlvresult	980
8.951unpack_loc_Start_t Struct Reference	980

8.951.1 Detailed Description	980
8.951.2 Field Documentation	981
8.951.2.1 Tlvresult	981
8.952unpack_loc_Stop_t Struct Reference	981
8.952.1 Detailed Description	981
8.952.2 Field Documentation	981
8.952.2.1 Tlvresult	981
8.953unpack_nas_GetCDMANetworkParameters_t Struct Reference	981
8.953.1 Detailed Description	982
8.953.2 Field Documentation	982
8.953.2.1 Application	982
8.953.2.2 Broadcast	982
8.953.2.3 CustomSCP	982
8.953.2.4 ForceRev0	982
8.953.2.5 Protocol	982
8.953.2.6 RegForeignNID	982
8.953.2.7 RegForeignSID	982
8.953.2.8 RegHomeSID	982
8.953.2.9 Roaming	982
8.953.2.10SCI	982
8.953.2.11SCM	982
8.954unpack_nas_GetHomeNetwork_t Struct Reference	982
8.954.1 Detailed Description	982
8.954.2 Field Documentation	983
8.954.2.1 mcc	983
8.954.2.2 mnc	983
8.954.2.3 name	983
8.954.2.4 nid	983
8.954.2.5 sid	983
8.955unpack_nas_GetNetworkPreference_t Struct Reference	983

8.955.1 Detailed Description	983
8.955.2 Field Documentation	984
8.955.2.1 ActiveTechPref	984
8.955.2.2 Duration	984
8.955.2.3 PersistentTechPref	984
8.955.2.4 Tlvresult	984
8.956unpack_nas_GetRFInfo_t Struct Reference	984
8.956.1 Detailed Description	984
8.956.2 Field Documentation	984
8.956.2.1 instancesSize	984
8.956.2.2 RFBandInfoElements	985
8.957unpack_nas_GetServingNetwork_t Struct Reference	985
8.957.1 Detailed Description	985
8.957.2 Field Documentation	985
8.957.2.1 CSDomain	985
8.957.2.2 DataCaps	985
8.957.2.3 DataCapsLen	985
8.957.2.4 MCC	985
8.957.2.5 MNC	986
8.957.2.6 Name	986
8.957.2.7 nameSize	986
8.957.2.8 PSDomain	986
8.957.2.9 Radiolfaces	986
8.957.2.10RadiolfacesSize	986
8.957.2.11RAN	986
8.957.2.12RegistrationState	986
8.957.2.13Roaming	986
8.958unpack_nas_GetServingNetworkCapabilities_t Struct Reference	986
8.958.1 Detailed Description	986
8.958.2 Field Documentation	986

8.958.2.1 DataCaps	986
8.958.2.2 DataCapsLen	986
8.959unpack_nas_GetSignalStrengths_t Struct Reference	986
8.959.1 Detailed Description	987
8.959.2 Field Documentation	987
8.959.2.1 len	987
8.959.2.2 radio	987
8.959.2.3 rssi	987
8.960unpack_nas_PerformNetworkScan_t Struct Reference	987
8.960.1 Detailed Description	987
8.960.2 Field Documentation	987
8.960.2.1 p3GppNetworkInfoInstances	987
8.960.2.2 p3GppNetworkInstanceSize	987
8.960.2.3 pPCSIInstance	987
8.960.2.4 pPCSIInstanceSize	987
8.960.2.5 pRATINInstance	987
8.960.2.6 pRATInstanceSize	988
8.960.2.7 pScanResult	988
8.961unpack_nas_SetDataCapabilitiesCallback_ind_t Struct Reference	988
8.961.1 Detailed Description	988
8.961.2 Field Documentation	988
8.961.2.1 dataCaps	988
8.961.2.2 dataCapsSize	988
8.962unpack_nas_SetEventReportInd_t Struct Reference	988
8.962.1 Detailed Description	988
8.962.2 Field Documentation	988
8.962.2.1 RFTIv	988
8.962.2.2 RRTIv	989
8.962.2.3 SLQSSSTIv	989
8.962.2.4 SSTIv	989

8.963unpack_nas_SetNasLTECphyCaIndCallback_ind_t Struct Reference	989
8.963.1 Detailed Description	989
8.963.2 Field Documentation	989
8.963.2.1 sPhyCaAggPcellInfo	989
8.963.2.2 sPhyCaAggScellIDBw	989
8.963.2.3 sPhyCaAggScellIndex	989
8.963.2.4 sPhyCaAggScellIndType	989
8.963.2.5 sPhyCaAggScellInfo	990
8.964unpack_nas_SetNetworkPreference_t Struct Reference	990
8.964.1 Detailed Description	990
8.964.2 Field Documentation	990
8.964.2.1 Tlvresult	990
8.965unpack_nas_SetRoamingIndicatorCallback_ind_t Struct Reference	990
8.965.1 Detailed Description	991
8.965.2 Field Documentation	991
8.965.2.1 roaming	991
8.966unpack_nas_SetServingSystemCallback_ind_t Struct Reference	991
8.966.1 Detailed Description	991
8.966.2 Field Documentation	991
8.966.2.1 SSInfo	991
8.966.2.2 Tlvresult	991
8.967unpack_nas_SlqsGetLTECphyCAInfo_t Struct Reference	991
8.967.1 Detailed Description	992
8.967.2 Field Documentation	992
8.967.2.1 LTECphyCAInfo	992
8.967.2.2 Tlvresult	992
8.968unpack_nas_SLQSGetNetworkTime_t Struct Reference	992
8.968.1 Detailed Description	992
8.968.2 Field Documentation	992
8.968.2.1 p3GPP2TimeInfo	992

8.968.2.2 p3GPPTIMEInfo	993
8.969unpack_nas_SLQSGetPLMNName_t Struct Reference	993
8.969.1 Field Documentation	993
8.969.1.1 longName	993
8.969.1.2 longNameCI	993
8.969.1.3 longNameEn	993
8.969.1.4 longNameLen	993
8.969.1.5 longNameSB	993
8.969.1.6 shortName	993
8.969.1.7 shortNameCI	993
8.969.1.8 shortNameEn	993
8.969.1.9 shortNameLen	993
8.969.1.10shortNameSB	993
8.969.1.11spn	993
8.969.1.12spnEncoding	993
8.969.1.13spnLength	993
8.970unpack_nas_SLQSGetServingSystem_t Struct Reference	993
8.970.1 Detailed Description	994
8.970.2 Field Documentation	995
8.970.2.1 BasestationID	995
8.970.2.2 BasestationLatitude	995
8.970.2.3 BasestationLongitude	995
8.970.2.4 CallBarStatus	995
8.970.2.5 CDMA_P_Rev	995
8.970.2.6 CDMASystemInfoExt	995
8.970.2.7 CellID	995
8.970.2.8 ConcSvcInfo	995
8.970.2.9 CurrentPLMN	995
8.970.2.10DataSrvCapabilities	995
8.970.2.11DefaultRoamInd	995

8.970.2.12DetailedSvcInfo	995
8.970.2.13DTMInd	995
8.970.2.14Gpp2TimeZone	995
8.970.2.15GppNetworkDSTAdjustment	995
8.970.2.16GppTimeZone	995
8.970.2.17HdrPersonality	995
8.970.2.18Lac	995
8.970.2.19NetworkID	996
8.970.2.20PRLInd	996
8.970.2.21RoamIndicatorVal	996
8.970.2.22RoamingIndicatorList	996
8.970.2.23ServingSystem	996
8.970.2.24SystemID	996
8.970.2.25TrackAreaCode	996
8.971unpack_nas_SLQSGetSignalStrength_t Struct Reference	996
8.971.1 Detailed Description	996
8.971.2 Field Documentation	997
8.971.2.1 ecioList	997
8.971.2.2 ecioListLen	997
8.971.2.3 errorRateList	997
8.971.2.4 errorRateListLen	997
8.971.2.5 lo	997
8.971.2.6 ltersrp	997
8.971.2.7 ltesnr	997
8.971.2.8 rsrqInfo	997
8.971.2.9 rxSignalStrengthList	997
8.971.2.10rxSignalStrengthListLen	997
8.971.2.11signalStrengthReqMask	997
8.971.2.12sinr	997
8.972unpack_nas_SLQSGetSysInfo_t Struct Reference	997

8.972.1 Detailed Description	997
8.972.2 Field Documentation	999
8.972.2.1 pAddCDMASysInfo	999
8.972.2.2 pAddGSMSysInfo	999
8.972.2.3 pAddHDRSysInfo	999
8.972.2.4 pAddLTESysInfo	999
8.972.2.5 pAddWCDMASysInfo	999
8.972.2.6 pCDMASrvStatusInfo	999
8.972.2.7 pCDMASysInfo	999
8.972.2.8 pGSMCallBarringSysInfo	999
8.972.2.9 pGSMCipherDomainSysInfo	999
8.972.2.10pGSMSrvStatusInfo	999
8.972.2.11pGSMSysInfo	999
8.972.2.12pHDRSrvStatusInfo	1000
8.972.2.13pHDRSysInfo	1000
8.972.2.14pLTESrvStatusInfo	1000
8.972.2.15pLTESysInfo	1000
8.972.2.16pLTEVoiceSupportSysInfo	1000
8.972.2.17pWCDMACallBarringSysInfo	1000
8.972.2.18pWCDMACipherDomainSysInfo	1000
8.972.2.19pWCDMASrvStatusInfo	1000
8.972.2.20pWCDMASysInfo	1000
8.973unpack_nas_SLQSGetSysSelectionPref_t Struct Reference	1000
8.973.1 Detailed Description	1000
8.973.2 Field Documentation	1003
8.973.2.1 pBandPref	1003
8.973.2.2 pEmerMode	1003
8.973.2.3 pGWAcqOrderPref	1003
8.973.2.4 pLTEBandPref	1003
8.973.2.5 pModePref	1003

8.973.2.6 pNetSelPref	1003
8.973.2.7 pPRLPref	1003
8.973.2.8 pRoamPref	1003
8.973.2.9 pSrvDomainPref	1003
8.974unpack_nas_SLQSNasGetCellLocationInfo_t Struct Reference	1003
8.974.1 Detailed Description	1004
8.974.2 Field Documentation	1004
8.974.2.1 pCDMAInfo	1004
8.974.2.2 pGERANInfo	1004
8.974.2.3 pLTEInfoInterfreq	1005
8.974.2.4 pLTEInfoIntrafreq	1005
8.974.2.5 pLTEInfoNeighboringGSM	1005
8.974.2.6 pLTEInfoNeighboringWCDMA	1005
8.974.2.7 pUMTSCellID	1005
8.974.2.8 pUMTSInfo	1005
8.974.2.9 pWCDMAInfoLTENeighborCell	1005
8.975unpack_nas_SLQSNasGetSigInfo_t Struct Reference	1005
8.975.1 Detailed Description	1005
8.975.2 Field Documentation	1005
8.975.2.1 CDMASSInfo	1005
8.975.2.2 GSMSSInfo	1005
8.975.2.3 HDRSSInfo	1005
8.975.2.4 LTESInfo	1005
8.975.2.5 WCDMASSInfo	1005
8.976unpack_nas_SLQSNasNetworkTimeCallBack_ind_t Struct Reference	1006
8.976.1 Detailed Description	1006
8.976.2 Field Documentation	1006
8.976.2.1 pDayltSavAdj	1006
8.976.2.2 pRadioInterface	1006
8.976.2.3 pTimeZone	1006

8.976.2.4 universalTime	1006
8.977unpack_nas_SLQSNasSigInfoCallback_ind_t Struct Reference	1007
8.977.1 Detailed Description	1007
8.977.2 Field Documentation	1007
8.977.2.1 pCDMASigInfo	1007
8.977.2.2 pGSMASigInfo	1007
8.977.2.3 pHDRSigInfo	1007
8.977.2.4 pLTESigInfo	1007
8.977.2.5 pRscp	1007
8.977.2.6 pTDSCDMASigInfoExt	1007
8.977.2.7 pWCDMASigInfo	1007
8.978unpack_nas_SLQSNasSwiModemStatus_t Struct Reference	1007
8.978.1 Detailed Description	1008
8.978.2 Field Documentation	1008
8.978.2.1 commonInfo	1008
8.978.2.2 pLTEInfo	1008
8.979unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t Struct Reference	1008
8.979.1 Detailed Description	1008
8.979.2 Field Documentation	1008
8.979.2.1 Info	1008
8.979.2.2 Tlvresult	1008
8.980unpack_nas_SLQSNasTimerCallback_ind_t Struct Reference	1008
8.980.1 Detailed Description	1009
8.980.2 Field Documentation	1009
8.980.2.1 t3396_apn	1009
8.980.2.2 t3396_plmn_id	1009
8.980.2.3 t3396_val	1009
8.981unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t Struct Reference	1009
8.981.1 Detailed Description	1009
8.981.2 Field Documentation	1010

8.981.2.1 Info	1010
8.981.2.2 Tlvresult	1010
8.982unpack_nas_SLQSSwiGetLteCQI_t Struct Reference	1010
8.982.1 Detailed Description	1010
8.982.2 Field Documentation	1010
8.982.2.1 CQIValueCW0	1010
8.982.2.2 CQIValueCW1	1010
8.982.2.3 ValidityCW0	1010
8.982.2.4 ValidityCW1	1010
8.983unpack_nas_SLQSSwiGetLteSccRxInfo_t Struct Reference	1011
8.983.1 Detailed Description	1011
8.983.2 Field Documentation	1011
8.983.2.1 pSccRxInfo	1011
8.984unpack_nas_SLQSSysInfoCallback_ind_t Struct Reference	1011
8.984.1 Detailed Description	1012
8.984.2 Field Documentation	1013
8.984.2.1 pAddCDMASysInfo	1013
8.984.2.2 pAddGSMSysInfo	1013
8.984.2.3 pAddHDRSysInfo	1013
8.984.2.4 pAddLTESysInfo	1013
8.984.2.5 pAddWCDMASysInfo	1013
8.984.2.6 pCDMASrvStatusInfo	1013
8.984.2.7 pCDMASysInfo	1013
8.984.2.8 pGSMCallBarringSysInfo	1013
8.984.2.9 pGSMCipherDomainSysInfo	1013
8.984.2.10pGSMSrvStatusInfo	1014
8.984.2.11pGSMSysInfo	1014
8.984.2.12pHDRSrvStatusInfo	1014
8.984.2.13pHDRSysInfo	1014
8.984.2.14pLTESrvStatusInfo	1014

8.984.2.15pLTESysInfo	1014
8.984.2.16pLTEVoiceSupportSysInfo	1014
8.984.2.17pSysInfoNoChange	1014
8.984.2.18pWCDMACallBarringSysInfo	1014
8.984.2.19pWCDMACipherDomainSysInfo	1014
8.984.2.20pWCDMASrvStatusInfo	1014
8.984.2.21pWCDMASysInfo	1014
8.985unpack_omaDmConfigTlv_t Struct Reference	1014
8.985.1 Detailed Description	1014
8.985.2 Field Documentation	1015
8.985.2.1 alertmsg	1015
8.985.2.2 alertmsglength	1015
8.985.2.3 state	1015
8.985.2.4 userInputReq	1015
8.985.2.5 userInputTimeout	1015
8.986unpack_omaDmFotaTlv_t Struct Reference	1015
8.986.1 Detailed Description	1016
8.986.2 Field Documentation	1017
8.986.2.1 description	1017
8.986.2.2 descriptionlength	1017
8.986.2.3 fwdloadsize	1017
8.986.2.4 fwloadComplete	1017
8.986.2.5 namelength	1017
8.986.2.6 package_name	1017
8.986.2.7 sessionType	1017
8.986.2.8 severity	1017
8.986.2.9 state	1017
8.986.2.10updateCompleteStatus	1017
8.986.2.11userInputReq	1017
8.986.2.12userInputTimeout	1017

8.986.2.13	version	1017
8.986.2.14	versionlength	1017
8.987	unpack_omaDmNotificationsTlv_t Struct Reference	1017
8.987.1	Field Documentation	1018
8.987.1.1	notification	1018
8.987.1.2	sessionStatus	1018
8.988	unpack_qmi_t Struct Reference	1018
8.988.1	Detailed Description	1018
8.988.2	Field Documentation	1018
8.988.2.1	msgid	1018
8.988.2.2	type	1018
8.988.2.3	xid	1018
8.989	unpack_qos_dataRate_t Struct Reference	1018
8.989.1	Detailed Description	1018
8.989.2	Field Documentation	1019
8.989.2.1	dataRateMax	1019
8.989.2.2	guaranteedRate	1019
8.990	unpack_qos_IPv4Addr_t Struct Reference	1019
8.990.1	Detailed Description	1019
8.990.2	Field Documentation	1019
8.990.2.1	addr	1019
8.990.2.2	subnetMask	1019
8.991	unpack_qos_IPv6Addr_t Struct Reference	1019
8.991.1	Detailed Description	1020
8.991.2	Field Documentation	1020
8.991.2.1	addr	1020
8.991.2.2	prefixLen	1020
8.992	unpack_qos_IPv6TrafCls_t Struct Reference	1020
8.992.1	Detailed Description	1020
8.992.2	Field Documentation	1020

8.992.2.1 mask	1020
8.992.2.2 val	1020
8.993unpack_qos_pktErrRate_t Struct Reference	1021
8.993.1 Detailed Description	1021
8.993.2 Field Documentation	1021
8.993.2.1 exponent	1021
8.993.2.2 multiplier	1021
8.994unpack_qos_Port_t Struct Reference	1021
8.994.1 Detailed Description	1021
8.994.2 Field Documentation	1021
8.994.2.1 port	1021
8.994.2.2 range	1021
8.995unpack_qos_QosFlowInfo_t Struct Reference	1022
8.995.1 Detailed Description	1022
8.995.2 Field Documentation	1023
8.995.2.1 BearerID	1023
8.995.2.2 is_RxQFlowGranted_Available	1023
8.995.2.3 is_TxQFlowGranted_Available	1023
8.995.2.4 NumRxFilters	1023
8.995.2.5 NumTxFilters	1023
8.995.2.6 QFlowState	1023
8.995.2.7 RxQFilter	1023
8.995.2.8 RxQFlowGranted	1023
8.995.2.9 TxQFilter	1023
8.995.2.10TxQFlowGranted	1023
8.996unpack_qos_QosFlowInfoState_t Struct Reference	1023
8.996.1 Detailed Description	1024
8.996.2 Field Documentation	1024
8.996.2.1 id	1024
8.996.2.2 isNewFlow	1024

8.996.2.3 state	1024
8.997unpack_qos_SLQSQosGetNetworkStatus_t Struct Reference	1024
8.997.1 Detailed Description	1024
8.997.2 Field Documentation	1024
8.997.2.1 NWQoSStatus	1025
8.998unpack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference	1025
8.998.1 Detailed Description	1025
8.998.2 Field Documentation	1026
8.998.2.1 ambr_dl	1026
8.998.2.2 ambr_dl_ext	1026
8.998.2.3 ambr_dl_ext2	1026
8.998.2.4 ambr_ul	1026
8.998.2.5 ambr_ul_ext	1026
8.998.2.6 ambr_ul_ext2	1026
8.998.2.7 apnId	1026
8.999unpack_qos_SLQSQosSwiReadDataStats_t Struct Reference	1026
8.999.1 Detailed Description	1026
8.999.2 Field Documentation	1027
8.999.2.1 apnId	1027
8.999.2.2 numQosFlow	1027
8.999.2.3 qosFlow	1027
8.999.2.4 total_rx_bytes	1027
8.999.2.5 total_rx_pkt	1027
8.999.2.6 total_tx_bytes	1027
8.999.2.7 total_tx_bytes_drp	1027
8.999.2.8 total_tx_pkt	1027
8.999.2.9 total_tx_pkt_drp	1027
8.1000unpack_qos_SLQSSetQosEventCallback_ind_t Struct Reference	1027
8.1000.1 Detailed Description	1027
8.1000.2 Field Documentation	1028

8.1000.2.1NumFlows	1028
8.1000.2.2QosFlowInfo	1028
8.1001unpack_qos_SLQSSetQosNWStatusCallback_ind_t Struct Reference	1028
8.1001.1Detailed Description	1028
8.1001.2Field Documentation	1028
8.1001.2.1status	1028
8.1002unpack_qos_SLQSSetQosPriEventCallback_ind_t Struct Reference	1028
8.1002.1Detailed Description	1029
8.1002.2Field Documentation	1029
8.1002.2.1event	1029
8.1003unpack_qos_SLQSSetQosStatusCallback_ind_t Struct Reference	1029
8.1003.1Detailed Description	1029
8.1003.2Field Documentation	1030
8.1003.2.1event	1030
8.1003.2.2d	1030
8.1003.2.3reason	1030
8.1003.2.4status	1030
8.1004unpack_qos_swiQosFilter_t Struct Reference	1030
8.1004.1Detailed Description	1031
8.1004.2Field Documentation	1032
8.1004.2.1EspSpi	1032
8.1004.2.2d	1033
8.1004.2.3index	1033
8.1004.2.4Pv4DstAddr	1033
8.1004.2.5Pv4SrcAddr	1033
8.1004.2.6Pv4Tos	1033
8.1004.2.7Pv6DstAddr	1033
8.1004.2.8Pv6Label	1033
8.1004.2.9Pv6SrcAddr	1033
8.1004.2.10Pv6TrafCls	1033

8.1004.2.1i4_EspSpi_Available	1033
8.1004.2.1i8_Id_Available	1033
8.1004.2.1i9_IPv4DstAddr_Available	1033
8.1004.2.1i4_IPv4SrcAddr_Available	1033
8.1004.2.1i5_IPv4Tos_Available	1033
8.1004.2.1i6_IPv6DstAddr_Available	1033
8.1004.2.1i7_IPv6Label_Available	1033
8.1004.2.1i8_IPv6SrcAddr_Available	1033
8.1004.2.1i9_IPv6TrafCls_Available	1033
8.1004.2.20_NxtHdrProto_Available	1033
8.1004.2.21_Precedence_Available	1033
8.1004.2.28_TCPDstPort_Available	1033
8.1004.2.29_TCPSrcPort_Available	1033
8.1004.2.24_TranDstPort_Available	1033
8.1004.2.25_TranSrcPort_Available	1034
8.1004.2.26_UDPDstPort_Available	1034
8.1004.2.27_UDPSrcPort_Available	1034
8.1004.2.28_NxtHdrProto	1034
8.1004.2.29_Precedence	1034
8.1004.2.30_TCPDstPort	1034
8.1004.2.31_TCPSrcPort	1034
8.1004.2.32_TranDstPort	1034
8.1004.2.33_TranSrcPort	1034
8.1004.2.34_UDPDstPort	1034
8.1004.2.35_UDPSrcPort	1034
8.1004.2.36_Version	1034
8.1005.1npack_qos_swiQosFlow_t Struct Reference	1034
8.1005.1Detailed Description	1035
8.1005.2Field Documentation	1037
8.1005.2.1DataRate	1037

8.1005.2.2index	1037
8.1005.2.3s_DataRate_Available	1037
8.1005.2.4s_Jitter_Available	1037
8.1005.2.5s_Latency_Available	1037
8.1005.2.6s_LteQci_Available	1037
8.1005.2.7s_MaxAllowedPktSz_Available	1037
8.1005.2.8s_MinPolicedPktSz_Available	1037
8.1005.2.9s_PktErrRate_Available	1037
8.1005.2.10_ProfileId3GPP2_Available	1037
8.1005.2.11s_TokenBucket_Available	1037
8.1005.2.118_TrafficClass_Available	1037
8.1005.2.113_val_3GPP2Pri_Available	1037
8.1005.2.114_val_3GPPImCn_Available	1037
8.1005.2.115_val_3GPPResResidualBER_Available	1037
8.1005.2.116_val_3GPPSigInd_Available	1037
8.1005.2.117_val_3GPPTraHdlPri_Available	1037
8.1005.2.118_Jitter	1038
8.1005.2.119_Latency	1038
8.1005.2.120_LteQci	1038
8.1005.2.121_MaxAllowedPktSz	1038
8.1005.2.122_MinPolicedPktSz	1038
8.1005.2.123_PktErrRate	1038
8.1005.2.124_ProfileId3GPP2	1038
8.1005.2.125_TokenBucket	1038
8.1005.2.126_TrafficClass	1038
8.1005.2.127_val_3GPP2Pri	1038
8.1005.2.128_val_3GPPImCn	1038
8.1005.2.129_val_3GPPResResidualBER	1038
8.1005.2.130_val_3GPPSigInd	1038
8.1005.2.131_val_3GPPTraHdlPri	1038

8.1006	npack_qos_tokenBucket_t Struct Reference	1038
8.1006.1	Detailed Description	1038
8.1006.2	Field Documentation	1039
8.1006.2.1	bucketSz	1039
8.1006.2.2	peakRate	1039
8.1006.2.3	tokenRate	1039
8.1007	npack_qos_Tos_t Struct Reference	1039
8.1007.1	Detailed Description	1039
8.1007.2	Field Documentation	1039
8.1007.2.1	mask	1039
8.1007.2.2	val	1039
8.1008	npack_QosFlowStat_t Struct Reference	1039
8.1008.1	Detailed Description	1040
8.1008.2	Field Documentation	1040
8.1008.2.1	bearerId	1040
8.1008.2.2	tx_bytes	1040
8.1008.2.3	tx_bytes_drp	1040
8.1008.2.4	tx_pkt	1040
8.1008.2.5	tx_pkt_drp	1040
8.1009	npack_RMTransferStatistics_ind_t Struct Reference	1040
8.1009.1	Detailed Description	1040
8.1009.2	Field Documentation	1041
8.1009.2.1	RxDropConutTlv	1041
8.1009.2.2	RxOkByteCountTlv	1041
8.1009.2.3	RxOkConutTlv	1041
8.1009.2.4	TxDropConutTlv	1041
8.1009.2.5	TxOkByteCountTlv	1041
8.1009.2.6	TxOkConutTlv	1041
8.1010	npack_sms_SendSMS_t Struct Reference	1041
8.1010.1	Detailed Description	1041

8.1010.2Field Documentation	1042
8.1010.2.1messageFailureCode	1042
8.1010.2.2messageID	1042
8.1011unpack_sms_SetNewSMSCallback_ind_t Struct Reference	1042
8.1011.1Detailed Description	1042
8.1011.2Field Documentation	1043
8.1011.2.1ETWSPLMNTlv	1043
8.1011.2.2ETWSTlv	1043
8.1011.2.3MSTlv	1043
8.1011.2.4MMTlv	1043
8.1011.2.5NewMMTlv	1043
8.1011.2.6SMSCtlv	1043
8.1011.2.7TRMessageTlv	1043
8.1012unpack_sms_SetNewSMSCallback_t Struct Reference	1043
8.1013unpack_sms_SLQSDelateSMS_t Struct Reference	1043
8.1014unpack_sms_SLQSGetSMS_t Struct Reference	1043
8.1014.1Detailed Description	1043
8.1014.2Field Documentation	1044
8.1014.2.1message	1044
8.1014.2.2messageFormat	1044
8.1014.2.3messageSize	1044
8.1014.2.4messageTag	1044
8.1015unpack_sms_SLQSGetSMSList_t Struct Reference	1044
8.1015.1Detailed Description	1044
8.1015.2Field Documentation	1044
8.1015.2.1messageList	1044
8.1015.2.2messageListSize	1044
8.1016unpack_sms_SLQSModifySMSStatus_t Struct Reference	1044
8.1017unpack_sms_SLQSWmsMemoryFullCallBack_ind_t Struct Reference	1045
8.1017.1Detailed Description	1045

8.1017.2Field Documentation	1045
8.1017.2.1messageMode	1045
8.1017.2.2storageType	1045
8.1018unpack_swiloc_SwiLocGetAutoStart_t Struct Reference	1045
8.1018.1Detailed Description	1045
8.1018.2Field Documentation	1046
8.1018.2.1fix_rate	1046
8.1018.2.2fix_rate_reported	1046
8.1018.2.3fix_type	1047
8.1018.2.4fix_type_reported	1047
8.1018.2.5function	1047
8.1018.2.6function_reported	1047
8.1018.2.7max_dist	1047
8.1018.2.8max_dist_reported	1047
8.1018.2.9max_time	1047
8.1018.2.10max_time_reported	1047
8.1019unpack_swioama_SLQSOMADMAAlertCallback_ind_t Struct Reference	1047
8.1019.1Detailed Description	1047
8.1019.2Field Documentation	1048
8.1019.2.1eventType	1048
8.1019.2.2SessionInfoConfig	1048
8.1019.2.3SessionInfoFota	1048
8.1019.2.4SessionInfoNotification	1048
8.1020unpack_swioama_SLQSOMADMGetSessionInfo_t Struct Reference	1048
8.1020.1Detailed Description	1048
8.1020.2Field Documentation	1050
8.1020.2.1Date	1050
8.1020.2.2DateLength	1050
8.1020.2.3PkgDescLength	1050
8.1020.2.4PkgDescription	1050

8.1020.2.5PkgName	1050
8.1020.2.6PkgNameLength	1050
8.1020.2.7RetryCount	1050
8.1020.2.8SessionState	1050
8.1020.2.9SessionType	1050
8.1020.2.10Severity	1050
8.1020.2.11Source	1050
8.1020.2.12SourceLength	1051
8.1020.2.13Status	1051
8.1020.2.14Time	1051
8.1020.2.15TimeLength	1051
8.1020.2.16UpdateCompleteStatus	1051
8.1021unpack_swima_SLQSOMADMGetSettings_t Struct Reference	1051
8.1021.1Detailed Description	1051
8.1021.2Field Documentation	1052
8.1021.2.1Autosdm	1052
8.1021.2.2FOTAdownload	1052
8.1021.2.3FOTAUpdate	1052
8.1021.2.4FwAutoCheck	1052
8.1021.2.5OMADMEEnabled	1052
8.1022unpack_swima_SLQSOMADMStartSession_t Struct Reference	1052
8.1022.1Detailed Description	1052
8.1022.2Field Documentation	1053
8.1022.2.1FwAvailability	1053
8.1023unpack_uim_ChangePin_t Struct Reference	1053
8.1023.1Detailed Description	1053
8.1023.2Field Documentation	1053
8.1023.2.1pEncryptedPIN1	1053
8.1023.2.2pIndicationToken	1054
8.1023.2.3pRemainingRetries	1054

8.1023.2.4Tlvresult	1054
8.1024unpack_uim_GetCardStatus_t Struct Reference	1054
8.1024.1Detailed Description	1054
8.1024.2Field Documentation	1054
8.1024.2.1pCardStatus	1054
8.1024.2.2pHotSwapStatus	1054
8.1024.2.3Tlvresult	1054
8.1025unpack_uim_ReadTransparent_t Struct Reference	1054
8.1025.1Detailed Description	1055
8.1025.2Field Documentation	1055
8.1025.2.1pCardResult	1055
8.1025.2.2pEncryptedData	1055
8.1025.2.3pIndicationToken	1055
8.1025.2.4pReadResult	1055
8.1025.2.5Tlvresult	1055
8.1026unpack_uim_SetPinProtection_t Struct Reference	1055
8.1026.1Detailed Description	1055
8.1026.2Field Documentation	1056
8.1026.2.1pEncryptedPIN1	1056
8.1026.2.2pIndicationToken	1056
8.1026.2.3pRemainingRetries	1056
8.1026.2.4Tlvresult	1056
8.1027unpack_uim_SetUimSlotStatusChangeCallback_ind_t Struct Reference	1056
8.1027.1Detailed Description	1056
8.1027.2Field Documentation	1057
8.1027.2.1bNumberOfPhySlots	1057
8.1027.2.2slotsstatusChange	1057
8.1028unpack_uim_SLQSUIEventRegister_t Struct Reference	1057
8.1028.1Detailed Description	1057
8.1028.2Field Documentation	1057

8.1028.2.1eventMask	1057
8.1029unpack_uim_SLQSUIGetSlotsStatus_t Struct Reference	1057
8.1029.1Detailed Description	1057
8.1029.2Field Documentation	1058
8.1029.2.1pNumberOfPhySlot	1058
8.1029.2.2pUimSlotsStatus	1058
8.1030unpack_uim_SLQSUISetStatusChangeCallBack_ind_t Struct Reference	1058
8.1030.1Detailed Description	1058
8.1030.2Field Documentation	1058
8.1030.2.1pCardStatus	1058
8.1031unpack_uim_UnblockPin_t Struct Reference	1058
8.1031.1Detailed Description	1058
8.1031.2Field Documentation	1059
8.1031.2.1pEncryptedPIN1	1059
8.1031.2.2pIndicationToken	1059
8.1031.2.3pRemainingRetries	1059
8.1031.2.4Tlvresult	1059
8.1032unpack_uim_VerifyPin_t Struct Reference	1059
8.1032.1Detailed Description	1059
8.1032.2Field Documentation	1060
8.1032.2.1pEncryptedPIN1	1060
8.1032.2.2pIndicationToken	1060
8.1032.2.3pRemainingRetries	1060
8.1032.2.4Tlvresult	1060
8.1033unpack_wds_DHCPv4ClientLease_ind_t Struct Reference	1060
8.1033.1Field Documentation	1060
8.1033.1.1DHCPv4LeaseOptTlv	1060
8.1033.1.2DHCPv4LeaseStateTlv	1060
8.1033.1.3Pv4AddrTlv	1060
8.1033.1.4ProfileIdTlv	1060

8.1034	npack_wds_GetAutoconnect_t Struct Reference	1060
8.1034.1	Detailed Description	1060
8.1034.2	Field Documentation	1061
8.1034.2.1	psetting	1061
8.1035	npack_wds_GetByteTotals_t Struct Reference	1061
8.1035.1	Detailed Description	1061
8.1035.2	Field Documentation	1061
8.1035.2.1	pRXTotalBytes	1061
8.1035.2.2	pTXTotalBytes	1061
8.1036	npack_wds_GetConnectionRate_t Struct Reference	1061
8.1036.1	Detailed Description	1061
8.1036.2	Field Documentation	1062
8.1036.2.1	currentChannelRXRate	1062
8.1036.2.2	currentChannelTXRate	1062
8.1036.2.3	maxChannelRXRate	1062
8.1036.2.4	maxChannelTXRate	1062
8.1037	npack_wds_GetDataBearerTechnology_t Struct Reference	1062
8.1037.1	Detailed Description	1062
8.1037.2	Field Documentation	1063
8.1037.2.1	pDataBearer	1063
8.1038	npack_wds_GetDefaultProfile_t Struct Reference	1063
8.1038.1	Detailed Description	1063
8.1038.2	Field Documentation	1063
8.1038.2.1	pnnname	1063
8.1038.2.2	pnnsize	1063
8.1038.2.3	auth	1063
8.1038.2.4	paddr	1064
8.1038.2.5	paddrv6	1064
8.1038.2.6	name	1064
8.1038.2.7	namesize	1064

8.1038.2.8	pdptype	1064
8.1038.2.9	pridns	1064
8.1038.2.10	pridnsv6	1064
8.1038.2.11	secdns	1064
8.1038.2.12	secdnsv6	1064
8.1038.2.13	username	1064
8.1038.2.14	user size	1064
8.1039	unpack_wds_GetDefaultProfileNum_t Struct Reference	1064
8.1039.1	Detailed Description	1064
8.1039.2	Field Documentation	1064
8.1039.2.1	index	1064
8.1040	unpack_wds_GetDormancyState_t Struct Reference	1064
8.1040.1	Detailed Description	1065
8.1040.2	Field Documentation	1065
8.1040.2.1	dormancyState	1065
8.1041	unpack_wds_GetLastMobileIPError_t Struct Reference	1065
8.1041.1	Detailed Description	1065
8.1041.2	Field Documentation	1065
8.1041.2.1	error	1065
8.1042	unpack_wds_GetMobileIP_t Struct Reference	1065
8.1042.1	Detailed Description	1065
8.1042.2	Field Documentation	1065
8.1042.2.1	mipMode	1065
8.1043	unpack_wds_GetMobileIPProfile_t Struct Reference	1065
8.1043.1	Detailed Description	1066
8.1043.2	Field Documentation	1066
8.1043.2.1	AAASPI	1066
8.1043.2.2	AAASState	1066
8.1043.2.3	address	1066
8.1043.2.4	enabled	1066

8.1043.2.5	HASPI	1066
8.1043.2.6	HASState	1066
8.1043.2.7	NAI	1066
8.1043.2.8	naiSize	1066
8.1043.2.9	primaryHA	1066
8.1043.2.10	evTunneling	1067
8.1043.2.11	secondaryHA	1067
8.1044	inpack_wds_GetPacketStatistics_t Struct Reference	1067
8.1044.1	Detailed Description	1067
8.1044.2	Field Documentation	1068
8.1044.2.1	pRXDroppedCount	1068
8.1044.2.2	pRXOkBytesCount	1068
8.1044.2.3	pRXOKBytesLastCall	1068
8.1044.2.4	pRXPacketErrors	1068
8.1044.2.5	pRXPacketOverflows	1068
8.1044.2.6	pRXPacketSuccesses	1068
8.1044.2.7	pTXDroppedCount	1068
8.1044.2.8	pTXOkBytesCount	1068
8.1044.2.9	pTXOKBytesLastCall	1068
8.1044.2.10	pTXPacketErrors	1068
8.1044.2.11	pTXPacketOverflows	1068
8.1044.2.12	pTXPacketSuccesses	1068
8.1045	inpack_wds_GetPacketStatus_t Struct Reference	1068
8.1045.1	Detailed Description	1069
8.1045.2	Field Documentation	1069
8.1045.2.1	rXDroppedCount	1069
8.1045.2.2	rXOkBytesCount	1069
8.1045.2.3	rXOKBytesLastCall	1069
8.1045.2.4	rXPacketErrors	1069
8.1045.2.5	rXPacketOverflows	1069

8.1045.2.6XPaketSuccesses	1069
8.1045.2.7XDroppedCount	1069
8.1045.2.8XOkBytesCount	1070
8.1045.2.9XOkBytesLastCall	1070
8.1045.2.10XPaketErrors	1070
8.1045.2.11XPaketOverflows	1070
8.1045.2.12XPaketSuccesses	1070
8.1046npack_wds_GetSessionDuration_t Struct Reference	1070
8.1046.1Detailed Description	1070
8.1046.2Field Documentation	1070
8.1046.2.1callDuration	1070
8.1047npack_wds_GetSessionState_t Struct Reference	1070
8.1047.1Detailed Description	1070
8.1047.2Field Documentation	1070
8.1047.2.1connectionStatus	1070
8.1048npack_wds_RMSetTransferStatistics_t Struct Reference	1071
8.1049npack_wds_SetMobileIPProfile_t Struct Reference	1071
8.1050npack_wds_SLQSCreateProfile_t Struct Reference	1071
8.1050.1Detailed Description	1071
8.1050.2Field Documentation	1071
8.1050.2.1pCreateProfileOut	1071
8.1050.2.2pProfileID	1071
8.1050.2.3Tlvresult	1071
8.1051npack_wds_SLQSDeleteProfile_t Struct Reference	1071
8.1051.1Detailed Description	1071
8.1051.2Field Documentation	1071
8.1051.2.1extendedErrorCode	1071
8.1052npack_wds_SLQSGet3GPPConfigItem_t Struct Reference	1072
8.1052.1Detailed Description	1072
8.1052.2Field Documentation	1073

8.1052.2.1_3gppRelease	1073
8.1052.2.2defaultPDNEnabled	1073
8.1052.2.3LTEAttachProfile	1073
8.1052.2.4LTEAttachProfileList	1073
8.1052.2.5LTEAttachProfileListLen	1073
8.1052.2.6profileList	1073
8.1053inpack_wds_SLQSGetCurrDataSystemStat_t Struct Reference	1073
8.1053.1Detailed Description	1073
8.1053.2Field Documentation	1073
8.1053.2.1currNetworkInfo	1073
8.1053.2.2networkInfoLen	1073
8.1053.2.3prefNetwork	1073
8.1054inpack_wds_SLQSGetCurrentChannelRate_t Struct Reference	1074
8.1054.1Detailed Description	1074
8.1054.2Field Documentation	1074
8.1054.2.1current_channel_rx_rate	1074
8.1054.2.2current_channel_tx_rate	1074
8.1054.2.3max_channel_rx_rate	1074
8.1054.2.4max_channel_tx_rate	1074
8.1055inpack_wds_SLQSGetDataBearerTechnology_t Struct Reference	1075
8.1055.1Detailed Description	1075
8.1055.2Field Documentation	1075
8.1055.2.1curDataBearerTechnology	1075
8.1055.2.2dataBearerMask	1075
8.1055.2.3astCallDataBearerTechnology	1075
8.1056inpack_wds_SLQSGetDUNCallInfo_t Struct Reference	1075
8.1056.1Detailed Description	1075
8.1056.2Field Documentation	1076
8.1056.2.1callEndReason	1076
8.1056.2.2channelRate	1076

8.1056.2.3connectionStatus	1076
8.1056.2.4dataBearerTech	1076
8.1056.2.5dormancyStatus	1076
8.1056.2.6astCallDataBearerTech	1076
8.1056.2.7astCallRXOKBytesCnt	1076
8.1056.2.8astCallTXOKBytesCnt	1076
8.1056.2.9mdmCallDurationActive	1076
8.1056.2.10OKBytesCount	1076
8.1056.2.11OKBytesCount	1076
8.1057unpack_wds_SLQSGetProfileSettings_t Struct Reference	1076
8.1057. Field Documentation	1077
8.1057.1.1pProfileSettings	1077
8.1057.1.2ProfileType	1077
8.1057.1.3Tlvresult	1077
8.1058unpack_wds_SLQSGetRuntimeSettings_t Struct Reference	1077
8.1058.1Detailed Description	1077
8.1058.2Field Documentation	1078
8.1058.2.1APNName	1078
8.1058.2.2Authentication	1078
8.1058.2.3DomainList	1078
8.1058.2.4GPRSGrantedQoS	1078
8.1058.2.5GWAddressV4	1078
8.1058.2.6MCNflag	1078
8.1058.2.7PFamilyPreference	1078
8.1058.2.8Pv4	1078
8.1058.2.9PV6AddrInfo	1078
8.1058.2.10PV6GWAddrInfo	1078
8.1058.2.11Mtu	1078
8.1058.2.12CSCFAddrPCO	1078
8.1058.2.13CSCFFQDNAddrList	1078

8.1058.2.14	DPTType	1079
8.1058.2.15	PrimaryDNSV4	1079
8.1058.2.16	PrimaryDNSV6	1079
8.1058.2.17	ProfileID	1079
8.1058.2.18	ProfileName	1079
8.1058.2.19	SecondaryDNSV4	1079
8.1058.2.20	SecondaryDNSV6	1079
8.1058.2.23	ServerAddrList	1079
8.1058.2.24	SubnetMaskV4	1079
8.1058.2.25	Technology	1079
8.1058.2.24	MTSGrantedQoS	1079
8.1058.2.25	Username	1079
8.1059	npack_wds_SLQSMModifyProfile_t Struct Reference	1079
8.1059.1	Detailed Description	1079
8.1059.2	Field Documentation	1079
8.1059.2.1	pExtErrorCode	1079
8.1060	npack_wds_SLQSSetIPFamilyPreference_t Struct Reference	1079
8.1060.1	Detailed Description	1080
8.1060.2	Field Documentation	1080
8.1060.2.1	Tlvresult	1080
8.1061	npack_wds_SLQSSetPacketSrvStatusCallback_t Struct Reference	1080
8.1061.1	Detailed Description	1080
8.1061.2	Field Documentation	1080
8.1061.2.1	bearerID	1080
8.1061.2.2	conn_status	1080
8.1061.2.3	pFamily	1081
8.1061.2.4	reconfigReqd	1081
8.1061.2.5	sessionEndReason	1081
8.1061.2.6	techName	1081
8.1061.2.7	verboseSessnEndReason	1081

8.1061.2.8verboseSessnEndReasonType	1081
8.1062npack_wds_SLQSSetWdsEventCallback_ind_t Struct Reference	1081
8.1062.1Detailed Description	1081
8.1062.2Field Documentation	1082
8.1062.2.1currDBTechAvail	1082
8.1062.2.2currNWInfo	1082
8.1062.2.3dataSysStatAvail	1082
8.1062.2.4dBTechAvail	1082
8.1062.2.5dBTechnology	1082
8.1062.2.6dormancyStatAvail	1082
8.1062.2.7dormancyStatus	1082
8.1062.2.8nipstatAvail	1082
8.1062.2.9nipStatus	1082
8.1062.2.10netInfoLen	1082
8.1062.2.11prefNetwork	1082
8.1062.2.12atMask	1082
8.1062.2.13_bytes	1082
8.1062.2.14_pkts	1082
8.1062.2.15Mask	1082
8.1062.2.16_bytes	1082
8.1062.2.17_pkts	1082
8.1062.2.18erStatAvail	1082
8.1063npack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference	1082
8.1063.1Detailed Description	1082
8.1063.2Field Documentation	1083
8.1063.2.1pHwConfig	1083
8.1063.2.2pRequestOptionList	1083
8.1064npack_wds_SLQSSGetLoopback_t Struct Reference	1083
8.1064.1Detailed Description	1083
8.1064.2Field Documentation	1083

8.1064.2.1ByteLoopbackMode	1083
8.1064.2.2ByteLoopbackMultiplier	1083
8.1065Inpack_wds_SLQSStartDataSession_t Struct Reference	1083
8.1065.1Detailed Description	1084
8.1065.2Field Documentation	1084
8.1065.2.1pFailureReason	1084
8.1065.2.2psid	1084
8.1065.2.3pVerboseFailReasonType	1084
8.1065.2.4pVerboseFailureReason	1084
8.1066Inpack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference	1084
8.1066.1Detailed Description	1084
8.1066.2Field Documentation	1085
8.1066.2.1apnName	1085
8.1066.2.2bearerId	1085
8.1066.2.3contextId	1085
8.1066.2.4pv4Address	1085
8.1066.2.5pv4GWAddress	1085
8.1066.2.6pv6Address	1085
8.1066.2.7pv6GWAddress	1085
8.1066.2.8pDNSIPv4Address	1085
8.1066.2.9pDNSIPv6Address	1085
8.1066.2.10pPCSCFIPv4Address	1085
8.1066.2.11pPCSCFIPv6Address	1085
8.1066.2.12pDNSIPv4Address	1086
8.1066.2.13pDNSIPv6Address	1086
8.1066.2.14pPCSCFIPv4Address	1086
8.1066.2.15pPCSCFIPv6Address	1086
8.1067UnPackGetProfileSettingOut Struct Reference	1086
8.1067.1Field Documentation	1086
8.1067.1.1curProfile	1086

8.1067.1.2pExtErrCode	1086
8.1068npackWdsProfileParam Union Reference	1086
8.1068.1Field Documentation	1086
8.1068.1.1SlqsProfile3GPP	1086
8.1068.1.2SlqsProfile3GPP2	1086
8.1069USBCompConfig Struct Reference	1086
8.1069.1Detailed Description	1086
8.1069.2Field Documentation	1087
8.1069.2.1pUSBComp	1087
8.1070USBCompParams Struct Reference	1087
8.1070.1Detailed Description	1087
8.1070.2Field Documentation	1089
8.1070.2.1pNumSupUSBComps	1089
8.1070.2.2pSupUSBComps	1089
8.1070.2.3pUSBComp	1089
8.1071USSDNoWaitIndicationInfo Struct Reference	1089
8.1071.1Detailed Description	1089
8.1071.2Field Documentation	1090
8.1071.2.1pAlphaIdentifier	1090
8.1071.2.2pError	1090
8.1071.2.3pFailureCause	1090
8.1071.2.4pUSSDData	1090
8.1072USSDRespFNetwork Struct Reference	1090
8.1072.1Detailed Description	1090
8.1072.2Field Documentation	1090
8.1072.2.1pRespData	1090
8.1072.2.2pTypeCode	1091
8.1073USSInfo Struct Reference	1091
8.1073.1Detailed Description	1091
8.1073.2Field Documentation	1091

8.1073.2.1ussData	1091
8.1073.2.2ussDCS	1091
8.1073.2.3ussLen	1091
8.1074. USSResp Struct Reference	1091
8.1074.1. Field Documentation	1092
8.1074.1.1pAlphaIDInfo	1092
8.1074.1.2pCallId	1092
8.1074.1.3pCcResultType	1092
8.1074.1.4pCCSuppsType	1092
8.1074.1.5pfailureCause	1092
8.1074.1.6pUSSDInfo	1092
8.1075. UUSInfo Struct Reference	1092
8.1075.1. Detailed Description	1092
8.1075.2. Field Documentation	1093
8.1075.2.1UUSData	1093
8.1075.2.2JUUSDatalen	1093
8.1075.2.3JUUSDcs	1093
8.1075.2.4JUUSType	1093
8.1076. VerifyUIMPIN Struct Reference	1093
8.1076.1. Detailed Description	1093
8.1076.2. Field Documentation	1094
8.1076.2.1pinID	1094
8.1076.2.2pinLen	1094
8.1076.2.3pinVal	1094
8.1077. VoiceALSSelectLineInfo Struct Reference	1094
8.1077.1. Detailed Description	1094
8.1077.2. Field Documentation	1094
8.1077.2.1lineValue	1094
8.1078. VoiceALSSetLineSwitchInfo Struct Reference	1094
8.1078.1. Detailed Description	1094

8.1078.2Field Documentation	1095
8.1078.2.1switchOption	1095
8.1079VoiceAnswerCall Struct Reference	1095
8.1079.1Detailed Description	1095
8.1079.2Field Documentation	1095
8.1079.2.1pCallId	1095
8.1080VoiceBindSubscriptionInfo Struct Reference	1095
8.1080.1Detailed Description	1095
8.1080.2Field Documentation	1095
8.1080.2.1subsType	1096
8.1081VoiceBurstDTMFInfo Struct Reference	1096
8.1081.1Detailed Description	1096
8.1081.2Field Documentation	1096
8.1081.2.1BurstDTMFInfo	1096
8.1081.2.2BurstDTMFLengths	1096
8.1082VoiceCallInfoReq Struct Reference	1096
8.1082.1Detailed Description	1096
8.1082.2Field Documentation	1097
8.1082.2.1callID	1097
8.1083VoiceCallInfoResp Struct Reference	1097
8.1083.1Detailed Description	1097
8.1083.2Field Documentation	1099
8.1083.2.1pAlertingPattern	1099
8.1083.2.2pAlertType	1099
8.1083.2.3pAlphaIDInfo	1099
8.1083.2.4pCallInfo	1099
8.1083.2.5pConnectNumInfo	1099
8.1083.2.6pDiagInfo	1099
8.1083.2.7pOTASPStatus	1099
8.1083.2.8pRemotePartyName	1099

8.1083.2.9pRemotePartyNum	1099
8.1083.2.10pSrvOpt	1099
8.1083.2.11pUUSInfo	1099
8.1083.2.12pVoicePrivacy	1099
8.1084VoiceCallRequestParams Struct Reference	1099
8.1084.1Detailed Description	1100
8.1084.2Field Documentation	1101
8.1084.2.1callNumber	1101
8.1084.2.2pCallPartySubAdd	1101
8.1084.2.3pCallType	1101
8.1084.2.4pCLIRType	1101
8.1084.2.5pCUGInfo	1101
8.1084.2.6pEmergencyCategory	1101
8.1084.2.7pSvcType	1101
8.1084.2.8pUUSInfo	1101
8.1085VoiceCallResponseParams Struct Reference	1101
8.1085.1Detailed Description	1102
8.1085.2Field Documentation	1102
8.1085.2.1pAlphaIDInfo	1102
8.1085.2.2pCallID	1102
8.1085.2.3pCCResultType	1102
8.1085.2.4pCCSUPSType	1102
8.1086VoiceContDTMFInfo Struct Reference	1102
8.1086.1Detailed Description	1102
8.1086.2Field Documentation	1103
8.1086.2.1DTMFdigit	1103
8.1086.2.2pCallID	1103
8.1087VoiceDTMFEventInfo Struct Reference	1103
8.1087.1Detailed Description	1103
8.1087.2Field Documentation	1104

8.1087.2.1DTMFInformation	1104
8.1087.2.2pOffLength	1104
8.1087.2.3pOnLength	1104
8.1088VoiceFlashInfo Struct Reference	1104
8.1088.1Detailed Description	1104
8.1088.2Field Documentation	1104
8.1088.2.1pCallID	1104
8.1088.2.2pFlashPayLd	1104
8.1088.2.3pFlashType	1104
8.1089VoiceGetAllCallInfo Struct Reference	1105
8.1089.1Detailed Description	1105
8.1089.2Field Documentation	1107
8.1089.2.1pArrAlertingPattern	1107
8.1089.2.2pArrAlertingType	1107
8.1089.2.3pArrAlphaID	1107
8.1089.2.4pArrCalledPartyNum	1107
8.1089.2.5pArrCallEndReason	1107
8.1089.2.6pArrCallInfo	1107
8.1089.2.7pArrConnectPartyNum	1107
8.1089.2.8pArrDiagInfo	1107
8.1089.2.9pArrRedirPartyNum	1107
8.1089.2.10pArrRemotePartyName	1107
8.1089.2.11pArrRemotePartyNum	1107
8.1089.2.12pArrSvcOption	1107
8.1089.2.13pArrUUSInfo	1107
8.1089.2.14pOTASPSStatus	1107
8.1089.2.15pVoicePrivacy	1107
8.1090VoiceGetCallBarringReq Struct Reference	1107
8.1090.1Detailed Description	1107
8.1090.2Field Documentation	1108

8.1090.2.1pSvcClass	1108
8.1090.2.2Reason	1108
8.1091VoiceGetCallBarringResp Struct Reference	1108
8.1091.1Detailed Description	1108
8.1091.2Field Documentation	1109
8.1091.2.1pAlphaIDInfo	1109
8.1091.2.2pCallID	1109
8.1091.2.3pCCResType	1109
8.1091.2.4pCCSUPSType	1109
8.1091.2.5pFailCause	1109
8.1091.2.6pSvcClass	1110
8.1092VoiceGetCallFWReq Struct Reference	1110
8.1092.1Detailed Description	1110
8.1092.2Field Documentation	1110
8.1092.2.1pSvcClass	1110
8.1092.2.2Reason	1110
8.1093VoiceGetCallFWResp Struct Reference	1110
8.1093.1Detailed Description	1111
8.1093.2Field Documentation	1111
8.1093.2.1pAlphaIDInfo	1111
8.1093.2.2pCallID	1111
8.1093.2.3pCCResType	1112
8.1093.2.4pCCSUPSType	1112
8.1093.2.5pFailCause	1112
8.1093.2.6pGetCallFWExtInfo	1112
8.1093.2.7pGetCallFWInfo	1112
8.1094VoiceGetCallWaitInfo Struct Reference	1112
8.1094.1Detailed Description	1112
8.1094.2Field Documentation	1113
8.1094.2.1pAlphaIDInfo	1113

8.1094.2.2pCallID	1113
8.1094.2.3pCCResType	1113
8.1094.2.4pCCSUPSType	1113
8.1094.2.5pFailCause	1113
8.1094.2.6pSvcClass	1113
8.1095VoiceGetCLIPResp Struct Reference	1113
8.1095.1Detailed Description	1113
8.1095.2Field Documentation	1114
8.1095.2.1pAlphaIDInfo	1114
8.1095.2.2pCallID	1114
8.1095.2.3pCCResType	1114
8.1095.2.4pCCSUPSType	1114
8.1095.2.5pCLIPResp	1114
8.1095.2.6pFailCause	1114
8.1096VoiceGetCLIRResp Struct Reference	1115
8.1096.1Detailed Description	1115
8.1096.2Field Documentation	1116
8.1096.2.1pAlphaIDInfo	1116
8.1096.2.2pCallID	1116
8.1096.2.3pCCResType	1116
8.1096.2.4pCCSUPSType	1116
8.1096.2.5pCLIRResp	1116
8.1096.2.6pFailCause	1116
8.1097VoiceGetCNAPResp Struct Reference	1116
8.1097.1Detailed Description	1116
8.1097.2Field Documentation	1117
8.1097.2.1pAlphaIDInfo	1117
8.1097.2.2pCallID	1117
8.1097.2.3pCCResType	1117
8.1097.2.4pCCSUPSType	1117

8.1097.2.5pCNAPResp	1117
8.1097.2.6pFailCause	1117
8.1098oiceGetCOLPResp Struct Reference	1117
8.1098.1Detailed Description	1117
8.1098.2Field Documentation	1118
8.1098.2.1pAlphaIDInfo	1118
8.1098.2.2pCallID	1118
8.1098.2.3pCCResType	1118
8.1098.2.4pCCSUPSType	1118
8.1098.2.5pCOLPResp	1118
8.1098.2.6pFailCause	1118
8.1099oiceGetCOLRResp Struct Reference	1119
8.1099.1Detailed Description	1119
8.1099.2Field Documentation	1120
8.1099.2.1pAlphaIDInfo	1120
8.1099.2.2pCallID	1120
8.1099.2.3pCCResType	1120
8.1099.2.4pCCSUPSType	1120
8.1099.2.5pCOLRResp	1120
8.1099.2.6pFailCause	1120
8.1100oiceGetConfigReq Struct Reference	1120
8.1100.1Detailed Description	1120
8.1100.2Field Documentation	1121
8.1100.2.1pAirTimer	1121
8.1100.2.2pAMRStatus	1121
8.1100.2.3pAutoAnswer	1121
8.1100.2.4pNamID	1121
8.1100.2.5pPrefVoicePrivacy	1121
8.1100.2.6pPrefVoiceSO	1121
8.1100.2.7pRoamTimer	1121

8.1100.2.8pTTYMode	1122
8.1100.2.9pVoiceDomainPref	1122
8.1101.1VoiceGetConfigResp Struct Reference	1122
8.1101.1.1Detailed Description	1122
8.1101.1.2Field Documentation	1123
8.1101.2.1pAirTimerCnt	1123
8.1101.2.2pAutoAnswerStat	1123
8.1101.2.3pCurAMRConfig	1123
8.1101.2.4pCurPrefVoiceSO	1123
8.1101.2.5pCurrTTYMode	1123
8.1101.2.6pCurVoiceDomainPref	1123
8.1101.2.7pCurVoicePrivacyPref	1123
8.1101.2.8pRoamTimerCnt	1124
8.1102.1VoiceIndicationRegisterInfo Struct Reference	1124
8.1102.1.1Detailed Description	1124
8.1102.1.2Field Documentation	1124
8.1102.2.1pRegDTMFEvents	1124
8.1102.2.2pRegVoicePrivacyEvents	1124
8.1102.2.3pSuppsNotifEvents	1125
8.1103.1VoiceInfoRec Struct Reference	1125
8.1103.1.1Detailed Description	1125
8.1103.1.2Field Documentation	1126
8.1103.2.1callID	1126
8.1103.2.2pCalledPartyInfo	1126
8.1103.2.3pCallerIDInfo	1126
8.1103.2.4pCallerNameInfo	1126
8.1103.2.5pCallingPartyInfo	1126
8.1103.2.6pCallWaitInd	1126
8.1103.2.7pCLIRCause	1127
8.1103.2.8pConnectNumInfo	1127

8.1103.2.9	DispInfo	1127
8.1103.2.10	ExtDispInfo	1127
8.1103.2.11	ExtDispRecInfo	1127
8.1103.2.12	LineCtrlInfo	1127
8.1103.2.13	NSSAudioCtrl	1127
8.1103.2.14	NSSRelease	1127
8.1103.2.15	RedirNumInfo	1127
8.1103.2.16	SignalInfo	1127
8.1104	VoiceManageCallsReq Struct Reference	1127
8.1104.1	Detailed Description	1127
8.1104.2	Field Documentation	1128
8.1104.2.1	CallID	1128
8.1104.2.2	SUPSType	1128
8.1105	VoiceManageCallsResp Struct Reference	1128
8.1105.1	Detailed Description	1128
8.1105.2	Field Documentation	1128
8.1105.2.1	FailCause	1128
8.1106	VoiceOrigUSSDNoWaitInfo Struct Reference	1128
8.1106.1	Detailed Description	1128
8.1106.2	Field Documentation	1129
8.1106.2.1	USSInformation	1129
8.1107	VoiceOTASPStatusInfo Struct Reference	1129
8.1107.1	Detailed Description	1129
8.1107.2	Field Documentation	1130
8.1107.2.1	CallID	1130
8.1107.2.2	OTASPStatus	1130
8.1108	VoicePrivacyInfo Struct Reference	1130
8.1108.1	Detailed Description	1130
8.1108.2	Field Documentation	1130
8.1108.2.1	CallID	1130

8.1108.2.2	voicePrivacy	1130
8.1109	voiceSetAllCallStatusCbkInfo Struct Reference	1130
8.1109.1	Detailed Description	1131
8.1109.2	Field Documentation	1132
8.1109.2.1	arrCallInfomation	1132
8.1109.2.2	arrAlertingPattern	1132
8.1109.2.3	arrAlertingType	1132
8.1109.2.4	arrAlphaID	1132
8.1109.2.5	arrCalledPartyNum	1132
8.1109.2.6	arrCallEndReason	1132
8.1109.2.7	arrConnectPartyNum	1132
8.1109.2.8	arrDiagInfo	1132
8.1109.2.9	arrRedirPartyNum	1132
8.1109.2.10	arrRemotePartyName	1133
8.1109.2.11	arrRemotePartyNum	1133
8.1109.2.12	arrSvcOption	1133
8.1110	voiceSetCallBarringPwdInfo Struct Reference	1133
8.1110.1	Detailed Description	1133
8.1110.2	Field Documentation	1134
8.1110.2.1	newPasswd	1134
8.1110.2.2	newPasswdAgain	1134
8.1110.2.3	oldPasswd	1134
8.1110.2.4	Reason	1134
8.1111	voiceSetCallBarringPwdResp Struct Reference	1134
8.1111.1	Detailed Description	1134
8.1111.2	Field Documentation	1135
8.1111.2.1	alphaIDInfo	1135
8.1111.2.2	callID	1135
8.1111.2.3	ccResType	1135
8.1111.2.4	ccSUPSType	1135

8.1111.2.5pFailCause	1135
8.1112oiceSetConfigReq Struct Reference	1135
8.1112.1Detailed Description	1135
8.1112.2Field Documentation	1136
8.1112.2.1pAirTimerConfig	1136
8.1112.2.2pAutoAnswer	1136
8.1112.2.3pPrefVoiceDomain	1136
8.1112.2.4pPrefVoiceSO	1136
8.1112.2.5pRoamTimerConfig	1136
8.1112.2.6pTTYMode	1136
8.1113oiceSetConfigResp Struct Reference	1136
8.1113.1Detailed Description	1137
8.1113.2Field Documentation	1138
8.1113.2.1pAirTimerStatus	1138
8.1113.2.2pAutoAnsStatus	1138
8.1113.2.3pPrefVoiceSOStatus	1138
8.1113.2.4pRoamTimerStatus	1138
8.1113.2.5pTTYConfigStatus	1138
8.1113.2.6pVoiceDomainPrefStatus	1138
8.1114oiceSetPrefPrivacy Struct Reference	1138
8.1114.1Detailed Description	1138
8.1114.2Field Documentation	1138
8.1114.2.1privacyPref	1138
8.1115oiceSetSUPSServiceReq Struct Reference	1138
8.1115.1Detailed Description	1139
8.1115.2Field Documentation	1140
8.1115.2.1pCallBarringPasswd	1140
8.1115.2.2pCallForwardingNumber	1140
8.1115.2.3pCallFwdTypeAndPlan	1140
8.1115.2.4pServiceClass	1140

8.1115.2.5pTimerVal	1140
8.1115.2.6reason	1140
8.1115.2.7voiceSvc	1140
8.1116voiceSetSUPSServiceResp Struct Reference	1140
8.1116.1Detailed Description	1140
8.1116.2Field Documentation	1141
8.1116.2.1pAlphaIDInfo	1141
8.1116.2.2pCallID	1141
8.1116.2.3pCCResultType	1141
8.1116.2.4pCCSUPSType	1141
8.1116.2.5pFailCause	1141
8.1117voiceStopContDTMFInfo Struct Reference	1141
8.1117.1Detailed Description	1141
8.1117.2Field Documentation	1142
8.1117.2.1callID	1142
8.1118voiceSUPSInfo Struct Reference	1142
8.1118.1Detailed Description	1142
8.1118.2Field Documentation	1144
8.1118.2.1pAlphaIDInfo	1144
8.1118.2.2pCallBarPasswd	1144
8.1118.2.3pCallFwdInfo	1144
8.1118.2.4pCallFWNum	1144
8.1118.2.5pCallFWTimerVal	1144
8.1118.2.6pCallID	1144
8.1118.2.7pCLIPstatus	1144
8.1118.2.8pCLIRstatus	1144
8.1118.2.9pCNAPstatus	1144
8.1118.2.10pCOLPstatus	1144
8.1118.2.11pCOLRstatus	1144
8.1118.2.12pDataSrc	1144

8.1118.2.10	FailCause	1145
8.1118.2.11	NewPwdData	1145
8.1118.2.12	Reason	1145
8.1118.2.13	SvcClass	1145
8.1118.2.14	USSInfo	1145
8.1118.2.15	SUPSInformation	1145
8.1119	VoiceSUPSNotification Struct Reference	1145
8.1119.1	Detailed Description	1145
8.1119.2	Field Documentation	1146
8.1119.2.1	callID	1147
8.1119.2.2	notifType	1147
8.1119.2.3	CUGIndex	1147
8.1119.2.4	ECTNum	1147
8.1120	WcdmaCellInfo Struct Reference	1147
8.1120.1	Detailed Description	1147
8.1120.2	Field Documentation	1147
8.1120.2.1	cpich_ecno	1147
8.1120.2.2	cpich_rscp	1147
8.1120.2.3	psc	1148
8.1120.2.4	rxlev	1148
8.1121	WCDMAECIOTresh Struct Reference	1148
8.1121.1	Detailed Description	1148
8.1121.2	Field Documentation	1148
8.1121.2.1	WCDMAECIOTreshList	1148
8.1121.2.2	WCDMAECIOTreshListLen	1148
8.1122	WCDMAInfoLTENNeighborCell Struct Reference	1148
8.1122.1	Detailed Description	1148
8.1122.2	Field Documentation	1149
8.1122.2.1	UMTSLTENbrCell	1149
8.1122.2.2	umtsLTENbrCellLen	1149

8.1122.2.3wcdmaRRCState	1149
8.1123wcdmaLongMsgDecodingParams Struct Reference	1149
8.1123.1Detailed Description	1150
8.1123.2Field Documentation	1151
8.1123.2.1Date	1151
8.1123.2.2pIsUDHPresent	1151
8.1123.2.3pMessage	1151
8.1123.2.4pPartNum	1151
8.1123.2.5pReferenceNum	1151
8.1123.2.6pScAddr	1151
8.1123.2.7pScAddrLength	1151
8.1123.2.8pSenderAddr	1151
8.1123.2.9pSenderAddrLength	1151
8.1123.2.10pTextMsg	1151
8.1123.2.11pTextMsgLength	1151
8.1123.2.12pTotalNum	1151
8.1123.2.13pTime	1151
8.1124wcdmaMsgDecodingParams Struct Reference	1151
8.1124.1Detailed Description	1151
8.1124.2Field Documentation	1152
8.1124.2.1Date	1152
8.1124.2.2pMessage	1152
8.1124.2.3pScAddr	1152
8.1124.2.4pScAddrLength	1152
8.1124.2.5pSenderAddr	1152
8.1124.2.6pSenderAddrLength	1152
8.1124.2.7pTextMsg	1153
8.1124.2.8pTextMsgLength	1153
8.1124.2.9pTime	1153
8.1125wcdmaMsgEncodingParams Struct Reference	1153

8.1125.1Detailed Description	1153
8.1125.2Field Documentation	1153
8.1125.2.1alphabet	1153
8.1125.2.2messageSize	1153
8.1125.2.3pDestAddr	1154
8.1125.2.4pPDUMessage	1154
8.1125.2.5pTextMsg	1154
8.1126.WCDMARSSIThresh Struct Reference	1154
8.1126.1Detailed Description	1154
8.1126.2Field Documentation	1154
8.1126.2.1pWCDMARSSIThreshList	1154
8.1126.2.2WCDMARSSIThreshListLen	1154
8.1127.WCDMASysInfo Struct Reference	1154
8.1127.1Detailed Description	1155
8.1127.2Field Documentation	1157
8.1127.2.1cellId	1157
8.1127.2.2cellIdValid	1157
8.1127.2.3hsCallStatus	1157
8.1127.2.4hsCallStatusValid	1157
8.1127.2.5hsInd	1157
8.1127.2.6hsIndValid	1157
8.1127.2.7lac	1157
8.1127.2.8lacValid	1157
8.1127.2.9MCC	1157
8.1127.2.10MNC	1157
8.1127.2.11networkIdValid	1158
8.1127.2.12sc	1158
8.1127.2.13scValid	1158
8.1127.2.14regRejectInfoValid	1158
8.1127.2.15jCause	1158

8.1127.2.1ObjectSrvDomain	1158
8.1127.2.1SysInfoWCDMA	1158
8.1128.1UcdmaUARFCN Struct Reference	1158
8.1128.1Detailed Description	1158
8.1128.2Field Documentation	1158
8.1128.2.1status	1158
8.1128.2.2uarfcn	1158
8.1129.1Uds_currNetworkInfo Struct Reference	1158
8.1129.1Detailed Description	1159
8.1129.2Field Documentation	1160
8.1129.2.1NetworkType	1160
8.1129.2.2RATMask	1160
8.1129.2.3SOMask	1160
8.1130.1Uds_DataULongLongTlv Struct Reference	1160
8.1130.1Field Documentation	1160
8.1130.1.1TlvPresent	1160
8.1130.1.2UlData	1160
8.1131.1Uds_DataULongTlv Struct Reference	1160
8.1131.1Field Documentation	1161
8.1131.1.1TlvPresent	1161
8.1131.1.2UlData	1161
8.1132.1Uds_DHCPLeaseOptTlv Struct Reference	1161
8.1132.1Field Documentation	1161
8.1132.1.1numOpt	1161
8.1132.1.2optList	1161
8.1132.1.3optListData	1161
8.1132.1.4TlvPresent	1161
8.1133.1Uds_DHCPLeaseStateTlv Struct Reference	1161
8.1133.1Field Documentation	1161
8.1133.1.1leaseState	1161

8.1133.1.2TlvPresent	1161
8.1134.1.1wds_DHCOpt Struct Reference	1161
8.1134.1.1Field Documentation	1162
8.1134.1.1.1optCode	1162
8.1134.1.1.2optValLen	1162
8.1134.1.1.3optVal	1162
8.1135.1.1wds_DHCPProfileIdTlv Struct Reference	1162
8.1135.1.1Field Documentation	1162
8.1135.1.1.1profileId	1162
8.1135.1.1.2profileType	1162
8.1135.1.1.3TlvPresent	1162
8.1136.1.1wds_DHCPv4HWConfig Struct Reference	1162
8.1136.1.1Detailed Description	1162
8.1136.1.2Field Documentation	1163
8.1136.1.2.1chaddr	1163
8.1136.1.2.2chaddrLen	1163
8.1136.1.2.3hwType	1163
8.1137.1.1wds_DHCPv4Option Struct Reference	1163
8.1137.1.1Detailed Description	1163
8.1137.1.2Field Documentation	1163
8.1137.1.2.1optCode	1163
8.1137.1.2.2optVal	1163
8.1137.1.2.3optValLen	1163
8.1138.1.1wds_DHCPv4OptionList Struct Reference	1163
8.1138.1.1Detailed Description	1164
8.1138.1.2Field Documentation	1164
8.1138.1.2.1numOpt	1164
8.1138.1.2.2optList	1164
8.1139.1.1wds_DHCPv4ProfileId Struct Reference	1164
8.1139.1.1Detailed Description	1164

8.1139.2Field Documentation	1164
8.1139.2.1profileId	1164
8.1139.2.2profileType	1164
8.1140wds_Domain Struct Reference	1165
8.1140.1Detailed Description	1165
8.1140.2Field Documentation	1165
8.1140.2.1domainLen	1165
8.1140.2.2domainName	1165
8.1141wds_DomainNameList Struct Reference	1165
8.1141.1Detailed Description	1165
8.1141.2Field Documentation	1166
8.1141.2.1domain	1166
8.1141.2.2numInstances	1166
8.1142wds_GPRSQoS Struct Reference	1166
8.1142.1Detailed Description	1166
8.1142.2Field Documentation	1166
8.1142.2.1delayClass	1166
8.1142.2.2meanThroughputClass	1166
8.1142.2.3peakThroughputClass	1166
8.1142.2.4precedenceClass	1166
8.1142.2.5reliabilityClass	1167
8.1143wds_IPv4AdTlv Struct Reference	1167
8.1143.1Field Documentation	1167
8.1143.1.1IPv4Addr	1167
8.1143.1.2TlvPresent	1167
8.1144wds_IPV6AddressInfo Struct Reference	1167
8.1144.1Detailed Description	1167
8.1144.2Field Documentation	1167
8.1144.2.1IPAddressV6	1167
8.1144.2.2IPV6PrefixLen	1167

8.1145. nds_IPV6GWAddressInfo Struct Reference	1167
8.1145.1 Detailed Description	1168
8.1145.2 Field Documentation	1168
8.1145.2.1 gwAddressV6	1168
8.1145.2.2 gwV6PrefixLen	1168
8.1146. nds_PCSCFFQDNAddress Struct Reference	1168
8.1146.1 Detailed Description	1168
8.1146.2 Field Documentation	1168
8.1146.2.1 fqdnAddr	1169
8.1146.2.2 fqdnLen	1169
8.1147. nds_PCSCFFQDNAddressList Struct Reference	1169
8.1147.1 Detailed Description	1169
8.1147.2 Field Documentation	1169
8.1147.2.1 numInstances	1169
8.1147.2.2 pcsfQDNAddress	1169
8.1148. nds_PCSCFIPv4ServerAddressList Struct Reference	1169
8.1148.1 Detailed Description	1169
8.1148.2 Field Documentation	1170
8.1148.2.1 numInstances	1170
8.1148.2.2 pcsfIPv4Addr	1170
8.1149. nds_ProfileIdentifier Struct Reference	1170
8.1149.1 Detailed Description	1170
8.1149.2 Field Documentation	1170
8.1149.2.1 profileIndex	1170
8.1149.2.2 profileType	1170
8.1150. nds_profileInfo Union Reference	1170
8.1150.1 Detailed Description	1170
8.1150.2 Field Documentation	1171
8.1150.2.1 SlqsProfile3GPP	1171
8.1150.2.2 SlqsProfile3GPP2	1171

8.1151. Wds_TrStatInd Struct Reference	1171
8.1151.1. Detailed Description	1171
8.1151.2. Field Documentation	1171
8.1151.2.1. statsMask	1171
8.1151.2.2. statsPeriod	1171
8.1152. Wds_UMTSMInQoS Struct Reference	1171
8.1152.1. Detailed Description	1172
8.1152.2. Field Documentation	1173
8.1152.2.1. deliveryErrSDU	1173
8.1152.2.2. grntDownlinkBitrate	1174
8.1152.2.3. grntUplinkBitrate	1174
8.1152.2.4. maxDownlinkBitrate	1174
8.1152.2.5. maxSDUSize	1174
8.1152.2.6. maxUplinkBitrate	1174
8.1152.2.7. qosDeliveryOrder	1174
8.1152.2.8. resBerRatio	1174
8.1152.2.9. sduErrorRatio	1174
8.1152.2.10. trafficClass	1174
8.1152.2.11. trafficPriority	1174
8.1152.2.12. transferDelay	1174
8.1153. WdsByteTotals Struct Reference	1174
8.1153.1. Detailed Description	1174
8.1153.2. Field Documentation	1175
8.1153.2.1. ByteTotalsElmntsV4	1175
8.1153.2.2. ByteTotalsElmntsV6	1175
8.1153.2.3. v4sessionId	1175
8.1153.2.4. v6sessionId	1175
8.1154. WdsByteTotalsElmnts Struct Reference	1175
8.1154.1. Detailed Description	1175
8.1154.2. Field Documentation	1175

8.1154.2.1pRXTotalBytes	1175
8.1154.2.2pTXTotalBytes	1175
8.1155WdsClientLeaseChange Struct Reference	1175
8.1155.1Detailed Description	1176
8.1155.2Field Documentation	1176
8.1155.2.1pEnableNotification	1176
8.1156WdsConnectionRate Struct Reference	1176
8.1156.1Detailed Description	1176
8.1156.2Field Documentation	1177
8.1156.2.1ConnRateElmntsV4	1177
8.1156.2.2ConnRateElmntsV6	1177
8.1156.2.3pV4sessionId	1177
8.1156.2.4pV6sessionId	1177
8.1157WdsConnectionRateElmnts Struct Reference	1177
8.1157.1Detailed Description	1177
8.1157.2Field Documentation	1177
8.1157.2.1pCurrentChannelRXRate	1177
8.1157.2.2pCurrentChannelTXRate	1177
8.1157.2.3pMaxChannelRXRate	1177
8.1157.2.4pMaxChannelTXRate	1178
8.1158WdsDHCPv4ClientLeaseInd Struct Reference	1178
8.1158.1Detailed Description	1178
8.1158.2Field Documentation	1178
8.1158.2.1pIPv4Addr	1178
8.1158.2.2pLeaseState	1178
8.1158.2.3pOptList	1178
8.1158.2.4pProfileId	1178
8.1159WdsDHCPv4Config Struct Reference	1179
8.1159.1Detailed Description	1179
8.1159.2Field Documentation	1179

8.1159.2.1pHwConfig	1179
8.1159.2.2pProfileId	1179
8.1159.2.3pRequestOptionList	1179
8.1160WdsDHCPv4HwConfig Struct Reference	1179
8.1160.1Detailed Description	1179
8.1160.2Field Documentation	1180
8.1160.2.1chaddr	1180
8.1160.2.2chaddrLen	1180
8.1160.2.3hwType	1180
8.1161WdsDhcpv4HwConfig Struct Reference	1180
8.1161.1Detailed Description	1180
8.1161.2Field Documentation	1180
8.1161.2.1chaddr	1180
8.1161.2.2chaddrLen	1180
8.1161.2.3hwType	1181
8.1162WdsDhcpv4Option Struct Reference	1181
8.1162.1Detailed Description	1181
8.1162.2Field Documentation	1181
8.1162.2.1optCode	1181
8.1162.2.2optVal	1181
8.1162.2.3optValLen	1181
8.1163WdsDHCPv4Option Struct Reference	1181
8.1163.1Detailed Description	1181
8.1163.2Field Documentation	1182
8.1163.2.1optCode	1182
8.1163.2.2optVal	1182
8.1163.2.3optValLen	1182
8.1164WdsDHCPv4OptionList Struct Reference	1182
8.1164.1Detailed Description	1182
8.1164.2Field Documentation	1182

8.1164.2.1numOpt	1182
8.1164.2.2pOptList	1183
8.1165WdsDhcpv4OptionList Struct Reference	1183
8.1165.1Detailed Description	1183
8.1165.2Field Documentation	1183
8.1165.2.1numOpt	1183
8.1165.2.2pOptList	1183
8.1166WdsDHCPv4ProfileId Struct Reference	1183
8.1166.1Detailed Description	1183
8.1166.2Field Documentation	1183
8.1166.2.1profileId	1184
8.1166.2.2profileType	1184
8.1167WdsDhcpv4ProfileId Struct Reference	1184
8.1167.1Detailed Description	1184
8.1167.2Field Documentation	1184
8.1167.2.1profileId	1184
8.1167.2.2profileType	1184
8.1168WdsGetLoopbackData Struct Reference	1184
8.1168.1Detailed Description	1184
8.1168.2Field Documentation	1185
8.1168.2.1ByteLoopbackMode	1185
8.1168.2.2ByteLoopbackMultiplier	1185
8.1169WdsIpAddressInfoReq Struct Reference	1185
8.1169.1Field Documentation	1185
8.1169.1.1ip	1185
8.1169.1.2pv4sessionId	1185
8.1169.1.3pv6sessionId	1185
8.1170WdsPktStatisticsElmnts Struct Reference	1185
8.1170.1Detailed Description	1186
8.1170.2Field Documentation	1187

8.1170.2.1pRXDroppedCount	1187
8.1170.2.2pRXOkBytesCount	1187
8.1170.2.3pRXOKBytesLastCall	1187
8.1170.2.4pRXPacketErrors	1187
8.1170.2.5pRXPacketOverflows	1187
8.1170.2.6pRXPacketSuccesses	1187
8.1170.2.7pTXDroppedCount	1187
8.1170.2.8pTXOkBytesCount	1187
8.1170.2.9pTXOKBytesLastCall	1187
8.1170.2.10pTXPacketErrors	1187
8.1170.2.11pTXPacketOverflows	1187
8.1170.2.12pTXPacketSuccesses	1187
8.1171WdsPktStatisticsReq Struct Reference	1187
8.1171.1Detailed Description	1187
8.1171.2Field Documentation	1188
8.1171.2.1pStatMask	1188
8.1172WdsPktStatisticsResp Struct Reference	1188
8.1172.1Detailed Description	1188
8.1172.2Field Documentation	1188
8.1172.2.1PktStatElmntsV4	1188
8.1172.2.2PktStatElmntsV6	1189
8.1172.2.3pV4sessionId	1189
8.1172.2.4pV6sessionId	1189
8.1173WdsProfileParam Union Reference	1189
8.1173.1Detailed Description	1189
8.1173.2Field Documentation	1189
8.1173.2.1SlqsProfile3GPP	1189
8.1173.2.2SlqsProfile3GPP2	1189
8.1174WdsRunTimeSettings Struct Reference	1189
8.1174.1Detailed Description	1189

8.1174.2Field Documentation	1190
8.1174.2.1rts	1190
8.1174.2.24sessionId	1190
8.1174.2.36sessionId	1190
8.1175WdsSetEventReportReq Struct Reference	1190
8.1175.1Detailed Description	1190
8.1175.2Field Documentation	1191
8.1175.2.1pCurrChannelRateInd	1191
8.1175.2.2pCurrDataBearerTechInd	1191
8.1175.2.3pCurrPrefDataSysInd	1191
8.1175.2.4pDataBearerTechInd	1191
8.1175.2.5pDataCallStatusChangeInd	1191
8.1175.2.6pDataSystemStatusChangeInd	1191
8.1175.2.7pDormancyStatusInd	1192
8.1175.2.8pEVDOPageMonPerChangeInd	1192
8.1175.2.9pMIPStatusInd	1192
8.1175.2.10pTransferStatInd	1192
8.1176WDSSetLoopbackData Struct Reference	1192
8.1176.1Detailed Description	1192
8.1176.2Field Documentation	1192
8.1176.2.1pLoopbackMode	1192
8.1176.2.2pLoopbackMultiplier	1192
8.1177WDSSWICurrentChannelRates Struct Reference	1192
8.1177.1Detailed Description	1193
8.1177.2Field Documentation	1193
8.1177.2.1current_channel_rx_rate	1193
8.1177.2.2current_channel_tx_rate	1193
8.1177.2.3max_channel_rx_rate	1193
8.1177.2.4max_channel_tx_rate	1193

9	File Documentation	1195
9.1	apdoxypages.c File Reference	1195
9.1.1	Detailed Description	1195
9.2	common.h File Reference	1195
9.2.1	Macro Definition Documentation	1197
9.2.1.1	DEAULT_LOC_TIMEOUT_IN_SEC	1197
9.2.1.2	MINREQBKLEN	1197
9.2.1.3	MSGID_AND_LEN	1197
9.2.1.4	MSGID_DONT_CARE	1197
9.2.1.5	SDK_VALIDATE_INPUT_PACK_PARAM	1197
9.2.1.6	SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL_XID	1197
9.2.1.7	SDU_HDR_LEN	1197
9.2.1.8	UNUSEDPARAM	1197
9.2.2	Typedef Documentation	1197
9.2.2.1	logger	1197
9.2.3	Enumeration Type Documentation	1197
9.2.3.1	eLOG_LEVEL	1197
9.2.3.2	eQMI_SVC	1198
9.2.3.3	eTimeout	1198
9.2.3.4	msgtype	1198
9.2.4	Function Documentation	1199
9.2.4.1	fill_pack_ctx(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t svc, int timeout)	1199
9.2.4.2	fill_sdu_hdr(pack_qmi_t *pCtx, uint8_t *pReqBuf)	1199
9.2.4.3	get_version()	1199
9.2.4.4	helper_get_resp_ctx(uint8_t svc, uint8_t *pbuf, uint32_t len, unpack_qmi_t *pCtx)	1199
9.2.4.5	helper_get_xid(uint8_t *qmi_resp)	1199
9.2.4.6	helper_set_log_func(logger func)	1199
9.2.4.7	helper_set_log_lvl(uint8_t lvl)	1199
9.2.4.8	libpack_GetVersion()	1199
9.2.4.9	libpack_log(uint8_t lvl, const char *fmt,...)	1200

9.2.4.10	unpack_result_code_only(uint8_t *pMdmResp)	1200
9.2.5	Variable Documentation	1200
9.2.5.1	glog	1200
9.2.5.2	gloglvl	1200
9.3	dms.h File Reference	1200
9.3.1	Macro Definition Documentation	1206
9.3.1.1	ACT_CODE_MAX_SIZE	1206
9.3.1.2	CK_MAX_SIZE	1206
9.3.1.3	DMS_IMGDETAILS_LEN	1206
9.3.1.4	DMS_MAX_CUST_ID_LEN	1206
9.3.1.5	DMS_MAX_CUST_VALUE_LEN	1206
9.3.1.6	DMS_MAX_FWUPDATE_LOG_STR_SZ	1207
9.3.1.7	DMS_MAX_FWUPDATE_REF_STR_SZ	1207
9.3.1.8	DMS_PM_FACTORY	1207
9.3.1.9	DMS_PM_LOW	1207
9.3.1.10	DMS_PM_OFFLINE	1207
9.3.1.11	DMS_PM_ONLINE	1207
9.3.1.12	DMS_PM_PERSISTENT_LOW	1207
9.3.1.13	DMS_PM_RESET	1207
9.3.1.14	DMS_PM_SHUT_DOWN	1207
9.3.1.15	DMS_SET_REPORT_DISABLE	1207
9.3.1.16	DMS_SET_REPORT_ENABLE	1207
9.3.1.17	DMS_SLQSFWINFO_APPVERSION_SZ	1207
9.3.1.18	DMS_SLQSFWINFO_BOOTVERSION_SZ	1207
9.3.1.19	DMS_SLQSFWINFO_CARRIER_SZ	1207
9.3.1.20	DMS_SLQSFWINFO_CUR_CARR_NAME	1207
9.3.1.21	DMS_SLQSFWINFO_CUR_CARR_REV	1207
9.3.1.22	DMS_SLQSFWINFO_MODELID_SZ	1207
9.3.1.23	DMS_SLQSFWINFO_PACKAGEID_SZ	1207
9.3.1.24	DMS_SLQSFWINFO_PRIVERSION_SZ	1207

9.3.1.25	DMS_SLQSFWINFO_SKU_SZ	1207
9.3.1.26	DMS_SWI_SET_IND_DISABLE	1207
9.3.1.27	DMS_SWI_SET_IND_ENABLE	1207
9.3.1.28	DMS_UINT8_MAX_STRING_SZ	1207
9.3.1.29	ERI_DATA_MAX_SIZE	1208
9.3.1.30	MAX_BUILD_ID_LEN	1208
9.3.1.31	MEID_MAX_SIZE	1208
9.3.1.32	SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH	1208
9.3.1.33	SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH	1208
9.3.1.34	SPC_SIZE	1208
9.3.1.35	UNIQUE_ID_LEN	1208
9.3.2	Function Documentation	1208
9.3.2.1	pack_dms_ActivateAutomatic(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_ActivateAutomatic_t *pReq)	1208
9.3.2.2	pack_dms_GetActivationState(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1208
9.3.2.3	pack_dms_GetBandCapability(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)	1209
9.3.2.4	pack_dms_GetCrashAction(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)	1209
9.3.2.5	pack_dms_GetCustFeature(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)	1210
9.3.2.6	pack_dms_GetCustFeaturesV2(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_GetCustFeaturesV2_t *reqArg)	1210
9.3.2.7	pack_dms_GetDeviceCap(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)	1210
9.3.2.8	pack_dms_GetDeviceCapabilities(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)	1210
9.3.2.9	pack_dms_GetDeviceHardwareRev(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)	1211
9.3.2.10	pack_dms_GetDeviceMfr(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)	1211
9.3.2.11	pack_dms_GetDeviceSerialNumbers(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)	1212
9.3.2.12	pack_dms_GetFirmwareInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)	1212

9.3.2.13	<code>pack_dms_GetFirmwareRevision(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1212
9.3.2.14	<code>pack_dms_GetFirmwareRevisions(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1213
9.3.2.15	<code>pack_dms_GetFSN(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1213
9.3.2.16	<code>pack_dms_GetHardwareRevision(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1214
9.3.2.17	<code>pack_dms_GetIMSI(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1214
9.3.2.18	<code>pack_dms_GetManufacturer(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1214
9.3.2.19	<code>pack_dms_GetModelID(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1215
9.3.2.20	<code>pack_dms_GetNetworkTime(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1215
9.3.2.21	<code>pack_dms_GetOfflineReason(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1216
9.3.2.22	<code>pack_dms_GetPower(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1216
9.3.2.23	<code>pack_dms_GetPRLVersion(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1216
9.3.2.24	<code>pack_dms_GetSerialNumbers(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1217
9.3.2.25	<code>pack_dms_GetUSBComp(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1217
9.3.2.26	<code>pack_dms_GetVoiceNumber(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1218
9.3.2.27	<code>pack_dms_ResetToFactoryDefaults(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_ResetToFactoryDefaults_t *pReq)</code>	1218
9.3.2.28	<code>pack_dms_SetActivationStatusCallback(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetActivationStatusCallback_t *reqArg)</code>	1218
9.3.2.29	<code>pack_dms_SetCrashAction(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetCrashAction_t reqArg)</code>	1219
9.3.2.30	<code>pack_dms_SetCustFeature(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetCustFeature_t *reqArg)</code>	1220
9.3.2.31	<code>pack_dms_SetCustFeaturesV2(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetCustFeaturesV2_t *reqArg)</code>	1220
9.3.2.32	<code>pack_dms_SetEventReport(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetEventReport_t *reqArg)</code>	1220

9.3.2.33	<code>pack_dms_SetFirmwarePreference(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1221
9.3.2.34	<code>pack_dms_SetPower(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetPower_t *reqArg)</code>	1221
9.3.2.35	<code>pack_dms_SetUSBComp(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetUSBComp_t *reqArg)</code>	1222
9.3.2.36	<code>pack_dms_SLQSDmsSwiGetResetInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1222
9.3.2.37	<code>pack_dms_SLQSDmsSwiIndicationRegister(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSDmsSwiIndicationRegister_t *reqArg)</code>	1222
9.3.2.38	<code>pack_dms_SLQSGetBandCapability(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)</code>	1223
9.3.2.39	<code>pack_dms_SLQSGetERIFFile(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1223
9.3.2.40	<code>pack_dms_SLQSSwiClearDyingGaspStatistics(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1224
9.3.2.41	<code>pack_dms_SLQSSwiGetCrashInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiGetCrashInfo_t *pReq)</code>	1224
9.3.2.42	<code>pack_dms_SLQSSwiGetDyingGaspCfg(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1225
9.3.2.43	<code>pack_dms_SLQSSwiGetDyingGaspStatistics(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1225
9.3.2.44	<code>pack_dms_SLQSSwiGetFirmwareCurr(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1225
9.3.2.45	<code>pack_dms_SLQSSwiGetFwUpdateStatus(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1226
9.3.2.46	<code>pack_dms_SLQSSwiGetHostDevInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1226
9.3.2.47	<code>pack_dms_SLQSSwiGetOSInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1226
9.3.2.48	<code>pack_dms_SLQSSwiGetSerialNoExt(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1227
9.3.2.49	<code>pack_dms_SLQSSwiSetDyingGaspCfg(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiSetDyingGaspCfg_t *reqArg)</code>	1227
9.3.2.50	<code>pack_dms_SLQSSwiSetHostDevInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiSetHostDevInfo_t *pReq)</code>	1228
9.3.2.51	<code>pack_dms_SLQSSwiSetOSInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiSetOSInfo_t *pReq)</code>	1228
9.3.2.52	<code>pack_dms_SLQSUIGetState(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1228

9.3.2.53	<code>pack_dms_UIMChangePIN(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMChangePIN_t *pReq)</code>	1229
9.3.2.54	<code>pack_dms_UIMGetControlKeyStatus(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMGetControlKeyStatus_t *pReq)</code>	1229
9.3.2.55	<code>pack_dms_UIMGetICCID(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMGetICCID_t *reqArg)</code>	1230
9.3.2.56	<code>pack_dms_UIMGetPINStatus(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1230
9.3.2.57	<code>pack_dms_UIMSetControlKeyProtection(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMSetControlKeyProtection_t *pReq)</code>	1231
9.3.2.58	<code>pack_dms_UIMSetPINProtection(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMSetPINProtection_t *pReq)</code>	1231
9.3.2.59	<code>pack_dms_UIMUnblockControlKey(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMUnblockControlKey_t *pReq)</code>	1231
9.3.2.60	<code>pack_dms_UIMUnblockPIN(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMUnblockPIN_t *pReq)</code>	1232
9.3.2.61	<code>pack_dms_UIMVerifyPIN(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMVerifyPIN_t *pReq)</code>	1232
9.3.2.62	<code>pack_dms_ValidateSPC(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_ResetToFactoryDefaults_t *pReq)</code>	1233
9.3.2.63	<code>unpack_dms_ActivateAutomatic(uint8_t *pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t *pOutput)</code>	1233
9.3.2.64	<code>unpack_dms_GetActivationState(uint8_t *pResp, uint16_t respLen, unpack_dms_GetActivationState_t *pOutput)</code>	1233
9.3.2.65	<code>unpack_dms_GetBandCapability(uint8_t *pResp, uint16_t respLen, unpack_dms_GetBandCapability_t *pOutput)</code>	1234
9.3.2.66	<code>unpack_dms_GetCrashAction(uint8_t *pResp, uint16_t respLen, unpack_dms_GetCrashAction_t *pOutput)</code>	1234
9.3.2.67	<code>unpack_dms_GetCustFeature(uint8_t *pResp, uint16_t respLen, unpack_dms_GetCustFeature_t *pOutput)</code>	1235
9.3.2.68	<code>unpack_dms_GetCustFeaturesV2(uint8_t *pResp, uint16_t respLen, unpack_dms_GetCustFeaturesV2_t *pOutput)</code>	1235
9.3.2.69	<code>unpack_dms_GetDeviceCap(uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceCap_t *pOutput)</code>	1235
9.3.2.70	<code>unpack_dms_GetDeviceCapabilities(uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceCapabilities_t *pOutput)</code>	1235
9.3.2.71	<code>unpack_dms_GetDeviceHardwareRev(uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceHardwareRev_t *pOutput)</code>	1236
9.3.2.72	<code>unpack_dms_GetDeviceMfr(uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceMfr_t *pOutput)</code>	1236

9.3.2.73	<code>unpack_dms_GetDeviceSerialNumbers(uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceSerialNumbers_t *pOutput)</code>	1237
9.3.2.74	<code>unpack_dms_GetFirmwareInfo(uint8_t *pResp, uint16_t respLen, unpack_dms_GetFirmwareInfo_t *pOutput)</code>	1237
9.3.2.75	<code>unpack_dms_GetFirmwareRevision(uint8_t *pResp, uint16_t respLen, unpack_dms_GetFirmwareRevision_t *pOutput)</code>	1237
9.3.2.76	<code>unpack_dms_GetFirmwareRevisions(uint8_t *pResp, uint16_t respLen, unpack_dms_GetFirmwareRevisions_t *pOutput)</code>	1238
9.3.2.77	<code>unpack_dms_GetFSN(uint8_t *pResp, uint16_t respLen, unpack_dms_GetFSN_t *pOutput)</code>	1238
9.3.2.78	<code>unpack_dms_GetHardwareRevision(uint8_t *pResp, uint16_t respLen, unpack_dms_GetHardwareRevision_t *pOutput)</code>	1239
9.3.2.79	<code>unpack_dms_GetIMSI(uint8_t *pResp, uint16_t respLen, unpack_dms_GetIMSI_t *pOutput)</code>	1239
9.3.2.80	<code>unpack_dms_GetManufacturer(uint8_t *pResp, uint16_t respLen, unpack_dms_GetManufacturer_t *pOutput)</code>	1239
9.3.2.81	<code>unpack_dms_GetModelID(uint8_t *pResp, uint16_t respLen, unpack_dms_GetModelID_t *pOutput)</code>	1240
9.3.2.82	<code>unpack_dms_GetNetworkTime(uint8_t *pResp, uint16_t respLen, unpack_dms_GetNetworkTime_t *pOutput)</code>	1240
9.3.2.83	<code>unpack_dms_GetOfflineReason(uint8_t *pResp, uint16_t respLen, unpack_dms_GetOfflineReason_t *pOutput)</code>	1241
9.3.2.84	<code>unpack_dms_GetPower(uint8_t *pResp, uint16_t respLen, unpack_dms_GetPower_t *pOutput)</code>	1241
9.3.2.85	<code>unpack_dms_GetPRLVersion(uint8_t *pResp, uint16_t respLen, unpack_dms_GetPRLVersion_t *pOutput)</code>	1241
9.3.2.86	<code>unpack_dms_GetSerialNumbers(uint8_t *pResp, uint16_t respLen, unpack_dms_GetSerialNumbers_t *pOutput)</code>	1242
9.3.2.87	<code>unpack_dms_GetUSBComp(uint8_t *pResp, uint16_t respLen, unpack_dms_GetUSBComp_t *pOutput)</code>	1242
9.3.2.88	<code>unpack_dms_GetVoiceNumber(uint8_t *pResp, uint16_t respLen, unpack_dms_GetVoiceNumber_t *pOutput)</code>	1243
9.3.2.89	<code>unpack_dms_ResetToFactoryDefaults(uint8_t *pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t *pOutput)</code>	1243
9.3.2.90	<code>unpack_dms_SetActivationStatusCallback(uint8_t *pResp, uint16_t respLen, unpack_dms_SetActivationStatusCallback_t *pOutput)</code>	1243
9.3.2.91	<code>unpack_dms_SetCrashAction(uint8_t *pResp, uint16_t respLen, unpack_dms_SetCrashAction_t *pOutput)</code>	1244
9.3.2.92	<code>unpack_dms_SetCustFeature(uint8_t *pResp, uint16_t respLen, unpack_dms_SetCustFeature_t *pOutput)</code>	1244

9.3.2.93	unpack_dms_SetCustFeaturesV2(uint8_t *pResp, uint16_t respLen, unpack_dms_SetCustFeaturesV2_t *pOutput)	1245
9.3.2.94	unpack_dms_SetEventReport(uint8_t *pResp, uint16_t respLen, unpack_dms_SetEventReport_t *pOutput)	1245
9.3.2.95	unpack_dms_SetEventReport_ind(uint8_t *pResp, uint16_t respLen, unpack_dms_SetEventReport_ind_t *pOutput)	1245
9.3.2.96	unpack_dms_SetFirmwarePreference(uint8_t *pResp, uint16_t respLen, unpack_dms_SetFirmwarePreference_t *pOutput)	1246
9.3.2.97	unpack_dms_SetPower(uint8_t *pResp, uint16_t respLen, unpack_dms_SetPower_t *pOutput)	1246
9.3.2.98	unpack_dms_SetUSBComp(uint8_t *pResp, uint16_t respLen, unpack_dms_SetUSBComp_t *pOutput)	1247
9.3.2.99	unpack_dms_SLQSDmsSwiGetResetInfo(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiGetResetInfo_t *pOutput)	1247
9.3.2.100	unpack_dms_SLQSDmsSwiGetResetInfo_Ind(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t *pOutput)	1247
9.3.2.101	unpack_dms_SLQSDmsSwiIndicationRegister(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiIndicationRegister_t *pOutput)	1248
9.3.2.102	unpack_dms_SLQSGetBandCapability(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSGetBandCapability_t *pOutput)	1248
9.3.2.103	unpack_dms_SLQSGetERIFile(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSGetERIFile_t *pOutput)	1249
9.3.2.104	unpack_dms_SLQSSwiClearDyingGaspStatistics(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiClearDyingGaspStatistics_t *pOutput)	1249
9.3.2.105	unpack_dms_SLQSSwiGetCrashInfo(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetCrashInfo_t *pOutput)	1250
9.3.2.106	unpack_dms_SLQSSwiGetDyingGaspCfg(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetDyingGaspCfg_t *pOutput)	1250
9.3.2.107	unpack_dms_SLQSSwiGetDyingGaspStatistics(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetDyingGaspStatistics_t *pOutput)	1250
9.3.2.108	unpack_dms_SLQSSwiGetFirmwareCurr(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetFirmwareCurr_t *pOutput)	1251
9.3.2.109	unpack_dms_SLQSSwiGetFwUpdateStatus(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetFwUpdateStatus_t *pOutput)	1251
9.3.2.110	unpack_dms_SLQSSwiGetHostDevInfo(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetHostDevInfo_t *pOutput)	1252
9.3.2.111	unpack_dms_SLQSSwiGetOSInfo(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetOSInfo_t *pOutput)	1252
9.3.2.112	unpack_dms_SLQSSwiGetSerialNoExt(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetSerialNoExt_t *pOutput)	1252

9.3.2.113	unpack_dms_SLQSSwiSetDyingGaspCfg(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiSetDyingGaspCfg_t *pOutput)	1253
9.3.2.114	unpack_dms_SLQSSwiSetHostDevInfo(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiSetHostDevInfo_t *pOutput)	1253
9.3.2.115	unpack_dms_SLQSSwiSetOSInfo(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiSetOSInfo_t *pOutput)	1254
9.3.2.116	unpack_dms_SLQSUIMGetState(uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSUIMGetState_t *pOutput)	1254
9.3.2.117	unpack_dms_UIMChangePIN(uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)	1254
9.3.2.118	unpack_dms_UIMGetControlKeyStatus(uint8_t *pResp, uint16_t respLen, unpack_dms_UIMGetControlKeyStatus_t *pOutput)	1255
9.3.2.119	unpack_dms_UIMGetICCID(uint8_t *pResp, uint16_t respLen, unpack_dms_UIMGetICCID_t *pOutput)	1255
9.3.2.120	unpack_dms_UIMGetPINStatus(uint8_t *pResp, uint16_t respLen, unpack_dms_UIMGetPINStatus_t *pOutput)	1256
9.3.2.121	unpack_dms_UIMSetControlKeyProtection(uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetControlKeyProtection_t *pOutput)	1256
9.3.2.122	unpack_dms_UIMSetPINProtection(uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)	1256
9.3.2.123	unpack_dms_UIMUnblockControlKey(uint8_t *pResp, uint16_t respLen, unpack_dms_UIMUnblockControlKey_t *pOutput)	1257
9.3.2.124	unpack_dms_UIMUnblockPIN(uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)	1257
9.3.2.125	unpack_dms_UIMVerifyPIN(uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)	1258
9.3.2.126	unpack_dms_ValidateSPC(uint8_t *pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t *pOutput)	1258
9.4	fms.h File Reference	1258
9.4.1	Macro Definition Documentation	1259
9.4.1.1	FMS_FW_PRI_BUILD_MATCH_LEN	1259
9.4.1.2	FMS_GOBI_LISTENTRIES_MAX	1259
9.4.1.3	FMS_GOBI_MBN_BUILD_ID_STR_LEN	1259
9.4.1.4	FMS_GOBI_MBN_IMG_ID_STR_LEN	1260
9.4.1.5	FMS_IMAGE_ID_BUILD_ID_LEN	1260
9.4.1.6	FMS_IMAGE_ID_IMG_ID_LEN	1260
9.4.1.7	FMS_IMAGE_ID_MAX_ENTRIES	1260

9.4.1.8	FMS_IMAGE_ID_PRI_IMGTYPE	1260
9.4.1.9	FMS_MAX_IMAGE_ID_ELEMENT	1260
9.4.1.10	FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE	1260
9.4.2	Function Documentation	1260
9.4.2.1	GetValidFwPriCombinations(FMSImageList *pStoredImageList, uint32_t *pValidCombinationSize, CarrierImage_t *pValidCombinations)	1260
9.4.2.2	pack_fms_GetImagesPreference(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_GetImagesPreference_t *reqArg)	1260
9.4.2.3	pack_fms_GetStoredImages(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_GetStoredImages_t *reqArg)	1261
9.4.2.4	pack_fms_SetImagesPreference(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_SetImagesPreference_t *reqArg)	1261
9.4.2.5	unpack_fms_GetImagesPreference(uint8_t *pResp, uint16_t respLen, unpack_fms_GetImagesPreference_t *pOutput)	1261
9.4.2.6	unpack_fms_GetStoredImages(uint8_t *pResp, uint16_t respLen, unpack_fms_GetStoredImages_t *pOutput)	1261
9.4.2.7	unpack_fms_SetImagesPreference(uint8_t *pResp, uint16_t respLen, unpack_fms_SetImagesPreference_t *pOutput)	1262
9.5	loc.h File Reference	1262
9.5.1	Macro Definition Documentation	1264
9.5.1.1	LOC_UINT8_MAX_STRING_SZ	1264
9.5.1.2	LOCEVENTMASKBATCHFULLNOTIFICATION	1264
9.5.1.3	LOCEVENTMASKENGINESTATE	1265
9.5.1.4	LOCEVENTMASKFIXSESSIONSTATE	1265
9.5.1.5	LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION	1265
9.5.1.6	LOCEVENTMASKGEOFENCEBREACHNOTIFICATION	1265
9.5.1.7	LOCEVENTMASKGEOFENCEGENALERT	1265
9.5.1.8	LOCEVENTMASKGNSSMEASUREMENTREPORT	1265
9.5.1.9	LOCEVENTMASKGNSSSVINFO	1265
9.5.1.10	LOCEVENTMASKINJECTPOSITIONREQ	1265
9.5.1.11	LOCEVENTMASKINJECTPREDICTEDORBITSREQ	1265
9.5.1.12	LOCEVENTMASKINJECTTIMERREQ	1266
9.5.1.13	LOCEVENTMASKINJECTWIFIAPDATAREQ	1266

9.5.1.14	LOCEVENTMASKINVALIDVALUE	1266
9.5.1.15	LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT	1266
9.5.1.16	LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ	1266
9.5.1.17	LOCEVENTMASKMOTIONDATACONTROL	1266
9.5.1.18	LOCEVENTMASKNIGEOFENCENOTIFICATION	1266
9.5.1.19	LOCEVENTMASKNINOTIFYVERIFYREQ	1266
9.5.1.20	LOCEVENTMASKNMEA	1266
9.5.1.21	LOCEVENTMASKPEDOMETERCONTROL	1267
9.5.1.22	LOCEVENTMASKPOSITIONREPORT	1267
9.5.1.23	LOCEVENTMASKSENSORSTREAMINGREADYSTATUS	1267
9.5.1.24	LOCEVENTMASKSETSPISTREAMINGREPORT	1267
9.5.1.25	LOCEVENTMASKTIMESYNCREQ	1267
9.5.1.26	LOCEVENTMASKVEHICLEDATAREADYSTATUS	1267
9.5.1.27	LOCEVENTMASKWIFIREQ	1267
9.5.1.28	MAX_SENSOR_DATA_LEN	1267
9.5.1.29	MAX_TEMP_DATA_LEN	1267
9.5.2	Enumeration Type Documentation	1267
9.5.2.1	anonymous enum	1267
9.5.3	Function Documentation	1268
9.5.3.1	pack_loc_DeleteAssistData(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_Delete_Assist_Data_t *reqArg)	1268
9.5.3.2	pack_loc_EventRegister(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_EventRegister_t *reqArg)	1268
9.5.3.3	pack_loc_SetExtPowerState(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SetExtPowerState_t *reqArg)	1268
9.5.3.4	pack_loc_SetOperationMode(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SetOperationMode_t *reqArg)	1269
9.5.3.5	pack_loc_SLQSLOCGetBestAvailPos(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCGetBestAvailPos_t *reqArg)	1269
9.5.3.6	pack_loc_SLQSLOCInjectPosition(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCInjectPosition_t *reqArg)	1270
9.5.3.7	pack_loc_SLQSLOCInjectSensorData(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCInjectSensorData_t *reqArg)	1270

9.5.3.8	pack_loc_SLQSLOCInjectUTCTime(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCInjectUTCTime_t *reqArg)	1270
9.5.3.9	pack_loc_SLQSLOCSetCradleMountConfig(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCSetCradleMountConfig_t *reqArg) . . .	1271
9.5.3.10	pack_loc_Start(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_Start_t *reqArg)	1271
9.5.3.11	pack_loc_Stop(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_Stop_t *reqArg)	1271
9.5.3.12	unpack_loc_BestAvailPos_Ind(uint8_t *pResp, uint16_t respLen, unpack_loc_BestAvailPos_Ind_t *pOutput)	1272
9.5.3.13	unpack_loc_DeleteAssistData(uint8_t *pResp, uint16_t respLen, unpack_loc_Delete_Assist_Data_t *pOutput)	1272
9.5.3.14	unpack_loc_DeleteAssistData_Ind(uint8_t *pResp, uint16_t respLen, unpack_loc_DeleteAssistData_Ind_t *pOutput)	1273
9.5.3.15	unpack_loc_EngineState_Ind(uint8_t *pResp, uint16_t respLen, unpack_loc_EngineState_Ind_t *pOutput)	1273
9.5.3.16	unpack_loc_EventRegister(uint8_t *pResp, uint16_t respLen, unpack_loc_EventRegister_t *pOutput)	1273
9.5.3.17	unpack_loc_GnssSvInfo_Ind(uint8_t *pResp, uint16_t respLen, unpack_loc_GnssSvInfo_Ind_t *pOutput)	1274
9.5.3.18	unpack_loc_PositionRpt_Ind(uint8_t *pResp, uint16_t respLen, unpack_loc_PositionRpt_Ind_t *pOutput)	1274
9.5.3.19	unpack_loc_SetExtPowerConfig_Ind(uint8_t *pResp, uint16_t respLen, unpack_loc_SetExtPowerConfig_Ind_t *pOutput)	1275
9.5.3.20	unpack_loc_SetExtPowerState(uint8_t *pResp, uint16_t respLen, unpack_loc_SetExtPowerState_t *pOutput)	1275
9.5.3.21	unpack_loc_SetOperationMode(uint8_t *pResp, uint16_t respLen, unpack_loc_SetOperationMode_t *pOutput)	1275
9.5.3.22	unpack_loc_SetOperationMode_Ind(uint8_t *pResp, uint16_t respLen, unpack_loc_SetOperationMode_Ind_t *pOutput)	1276
9.5.3.23	unpack_loc_SLQSLOCGetBestAvailPos(uint8_t *pResp, uint16_t respLen, unpack_loc_SLQSLOCGetBestAvailPos_t *pOutput)	1276
9.5.3.24	unpack_loc_SLQSLOCInjectPosition(uint8_t *pResp, uint16_t respLen)	1277
9.5.3.25	unpack_loc_SLQSLOCInjectSensorData(uint8_t *pResp, uint16_t respLen)	1277
9.5.3.26	unpack_loc_SLQSLOCInjectUTCTime(uint8_t *pResp, uint16_t respLen)	1277
9.5.3.27	unpack_loc_SLQSLOCSetCradleMountConfig(uint8_t *pResp, uint16_t respLen)	1277
9.5.3.28	unpack_loc_Start(uint8_t *pResp, uint16_t respLen, unpack_loc_Start_t *pOutput)	1278
9.5.3.29	unpack_loc_Stop(uint8_t *pResp, uint16_t respLen, unpack_loc_Stop_t *pOutput)	1278

9.6	nas.h File Reference	1279
9.6.1	Macro Definition Documentation	1284
9.6.1.1	NAS_MAX_DESCRIPTION_LENGTH	1284
9.6.1.2	NAS_MAX_NUM_NETWORKS	1284
9.6.1.3	NAS_MAX_SCC_RX_INFO_INSTANCES	1284
9.6.1.4	NAS_OTA_MESSAGE_MAX_BUF_SIZE	1284
9.6.1.5	NAS_PLMN_LENGTH	1284
9.6.1.6	NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST	1284
9.6.2	Enumeration Type Documentation	1285
9.6.2.1	LIBPACK_NAS_LTE_CPHY_CA_BW_NRB	1285
9.6.2.2	LIBPACK_NAS_LTE_CPHY_SCELL_STATE	1285
9.6.2.3	NAS_LTE_CPHY_CA_BW_NRB_LITE	1285
9.6.2.4	NAS_LTE_CPHY_SCELL_STATE_LITE	1285
9.6.3	Function Documentation	1285
9.6.3.1	pack_nas_GetACCOLC(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1285
9.6.3.2	pack_nas_GetANAAAuthenticationStatus(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1286
9.6.3.3	pack_nas_GetCDMANetworkParameters(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1286
9.6.3.4	pack_nas_GetHomeNetwork(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1286
9.6.3.5	pack_nas_GetNetworkPreference(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1287
9.6.3.6	pack_nas_GetRFInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1287
9.6.3.7	pack_nas_GetServingNetwork(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1287
9.6.3.8	pack_nas_GetServingNetworkCapabilities(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1288
9.6.3.9	pack_nas_GetSignalStrengths(pack_qmi_t *pCtx, uint8_t *pReq, uint16_t *pLen)	1288
9.6.3.10	pack_nas_PerformNetworkScan(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1288
9.6.3.11	pack_nas_SetACCOLC(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SetACCOLC_t reqParam)	1289
9.6.3.12	pack_nas_SetLURejectCallback(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t *pBenable)	1289

9.6.3.13	<code>pack_nas_SetNetworkPreference(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SetNetworkPreference_t *reqArg)</code>	1289
9.6.3.14	<code>pack_nas_SetRFInfoCallback(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t *pBenable)</code>	1290
9.6.3.15	<code>pack_nas_SlqsGetLTECphyCAInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1290
9.6.3.16	<code>pack_nas_SLQSGetNetworkTime(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1290
9.6.3.17	<code>pack_nas_SLQSGetPLMNName(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSGetPLMNName_t *reqArg)</code>	1290
9.6.3.18	<code>pack_nas_SLQSGetServingSystem(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1291
9.6.3.19	<code>pack_nas_SLQSGetSignalStrength(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint16_t reqMask)</code>	1291
9.6.3.20	<code>pack_nas_SLQSGetSysInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1291
9.6.3.21	<code>pack_nas_SLQSGetSysSelectionPref(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1292
9.6.3.22	<code>pack_nas_SLQSIInitiateNetworkRegistration(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSIInitiateNetworkRegistration_t *pReqParam)</code>	1292
9.6.3.23	<code>pack_nas_SLQSNasConfigSigInfo2(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasConfigSigInfo2_t *pReqParam)</code>	1292
9.6.3.24	<code>pack_nas_SLQSNasGetCellLocationInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1293
9.6.3.25	<code>pack_nas_SLQSNasGetSigInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1293
9.6.3.26	<code>pack_nas_SLQSNasIndicationRegisterExt(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasIndicationRegisterExt_t *pReqParam)</code>	1293
9.6.3.27	<code>pack_nas_SLQSNasSwiIndicationRegister(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasSwiIndicationRegister_t *pReqParam)</code>	1294
9.6.3.28	<code>pack_nas_SLQSNasSwiModemStatus(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1294
9.6.3.29	<code>pack_nas_SLQSSetBandPreference(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint64_t bandPref)</code>	1294
9.6.3.30	<code>pack_nas_SLQSSetSignalStrengthsCallback(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSetSignalStrengthsCallback_t *pReqParam)</code>	1295
9.6.3.31	<code>pack_nas_SLQSSetSysSelectionPref(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSetSysSelectionPref_t *pReqParam)</code>	1295
9.6.3.32	<code>pack_nas_SLQSSwiGetLteCQI(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)</code>	1295

9.6.3.33	pack_nas_SLQSSwiGetLteSccRxInfo(pack_qmi_t *pCtx, uint8_t *pReq, uint16_t *pLen)	1296
9.6.3.34	unpack_nas_GetACCOLC(uint8_t *pResp, uint16_t respLen, uint8_t *pAccolc) .	1296
9.6.3.35	unpack_nas_GetANAAAAuthenticationStatus(uint8_t *pResp, uint16_t respLen, uint32_t *pAuthStatus)	1296
9.6.3.36	unpack_nas_GetCDMANetworkParameters(uint8_t *pResp, uint16_t respLen, unpack_nas_GetCDMANetworkParameters_t *pOutput)	1297
9.6.3.37	unpack_nas_GetHomeNetwork(uint8_t *pResp, uint16_t respLen, unpack_nas_GetHomeNetwork_t *pOutput)	1297
9.6.3.38	unpack_nas_GetNetworkPreference(uint8_t *pResp, uint16_t respLen, unpack_nas_GetNetworkPreference_t *pOutput)	1297
9.6.3.39	unpack_nas_GetRFInfo(uint8_t *pResp, uint16_t respLen, unpack_nas_GetRFInfo_t *pOutput)	1297
9.6.3.40	unpack_nas_GetServingNetwork(uint8_t *pResp, uint16_t respLen, unpack_nas_GetServingNetwork_t *pOutput)	1298
9.6.3.41	unpack_nas_GetServingNetworkCapabilities(uint8_t *pResp, uint16_t respLen, unpack_nas_GetServingNetworkCapabilities_t *pOutput)	1298
9.6.3.42	unpack_nas_GetSignalStrengths(uint8_t *pResp, uint16_t respLen, unpack_nas_GetSignalStrengths_t *pOutput)	1299
9.6.3.43	unpack_nas_PerformNetworkScan(uint8_t *pResp, uint16_t respLen, unpack_nas_PerformNetworkScan_t *pOutput)	1299
9.6.3.44	unpack_nas_SetACCOLC(uint8_t *pResp, uint16_t respLen)	1299
9.6.3.45	unpack_nas_SetDataCapabilitiesCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_nas_SetDataCapabilitiesCallback_ind_t *pOutput)	1300
9.6.3.46	unpack_nas_SetEventReportInd(uint8_t *pResp, uint16_t respLen, unpack_nas_SetEventReportInd_t *pOutput)	1300
9.6.3.47	unpack_nas_SetLURRejectCallback(uint8_t *pResp, uint16_t respLen)	1300
9.6.3.48	unpack_nas_SetNasLTECphyCalndCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_nas_SetNasLTECphyCalndCallback_ind_t *pOutput)	1300
9.6.3.49	unpack_nas_SetNetworkPreference(uint8_t *pResp, uint16_t respLen, unpack_nas_SetNetworkPreference_t *pOutput)	1301
9.6.3.50	unpack_nas_SetRFInfoCallback(uint8_t *pResp, uint16_t respLen)	1301
9.6.3.51	unpack_nas_SetRoamingIndicatorCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_nas_SetRoamingIndicatorCallback_ind_t *pOutput)	1301
9.6.3.52	unpack_nas_SetServingSystemCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_nas_SetServingSystemCallback_ind_t *pOutput)	1301
9.6.3.53	unpack_nas_SlqsGetLTECphyCAInfo(uint8_t *pResp, uint16_t respLen, unpack_nas_SlqsGetLTECphyCAInfo_t *pOutput)	1302

9.6.3.54	unpack_nas_SLQSGetNetworkTime(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetNetworkTime_t *pOutput)	1302
9.6.3.55	unpack_nas_SLQSGetPLMNName(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetPLMNName_t *pOutput)	1302
9.6.3.56	unpack_nas_SLQSGetServingSystem(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetServingSystem_t *pOutput)	1302
9.6.3.57	unpack_nas_SLQSGetSignalStrength(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSignalStrength_t *pOutput)	1303
9.6.3.58	unpack_nas_SLQSGetSysInfo(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSysInfo_t *pOutput)	1303
9.6.3.59	unpack_nas_SLQSGetSysSelectionPref(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSysSelectionPref_t *pOutput)	1303
9.6.3.60	unpack_nas_SLQSInitiateNetworkRegistration(uint8_t *pResp, uint16_t respLen)	1304
9.6.3.61	unpack_nas_SLQSNasConfigSigInfo2(uint8_t *pResp, uint16_t respLen)	1304
9.6.3.62	unpack_nas_SLQSNasGetCellLocationInfo(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetCellLocationInfo_t *pOutput)	1304
9.6.3.63	unpack_nas_SLQSNasGetSigInfo(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetSigInfo_t *pOutput)	1305
9.6.3.64	unpack_nas_SLQSNasIndicationRegisterExt(uint8_t *pResp, uint16_t respLen)	1305
9.6.3.65	unpack_nas_SLQSNasNetworkTimeCallBack_ind(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasNetworkTimeCallBack_ind_t *pOutput)	1305
9.6.3.66	unpack_nas_SLQSNasSigInfoCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasSigInfoCallback_ind_t *pOutput)	1306
9.6.3.67	unpack_nas_SLQSNasSwiIndicationRegister(uint8_t *pResp, uint16_t respLen)	1306
9.6.3.68	unpack_nas_SLQSNasSwiModemStatus(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasSwiModemStatus_t *pOutput)	1306
9.6.3.69	unpack_nas_SLQSNasSwiOTAMessageCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t *pOutput)	1307
9.6.3.70	unpack_nas_SLQSNasSysInfoCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSysInfoCallback_ind_t *pOutput)	1307
9.6.3.71	unpack_nas_SLQSNasTimerCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasTimerCallback_ind_t *pOutput)	1307
9.6.3.72	unpack_nas_SLQSSetBandPreference(uint8_t *pResp, uint16_t respLen)	1308
9.6.3.73	unpack_nas_SLQSSetSignalStrengthsCallback(uint8_t *pResp, uint16_t respLen)	1308
9.6.3.74	unpack_nas_SLQSSetSysSelectionPref(uint8_t *pResp, uint16_t respLen)	1308
9.6.3.75	unpack_nas_SLQSSetSysSelectionPrefCallBack_ind(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t *pOutput)	1309

9.6.3.76	unpack_nas_SLQSSwiGetLteCQI(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetLteCQI_t *pOutput)	1309
9.6.3.77	unpack_nas_SLQSSwiGetLteSccRxInfo(uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetLteSccRxInfo_t *pOutput)	1309
9.7	qaCbkCatEventReportInd.h File Reference	1310
9.7.1	Macro Definition Documentation	1311
9.7.1.1	QMI_CAN_COMMON_EVENT_TLV_NUMBER	1311
9.7.1.2	QMI_MAX_CAT_EVENT_DATA_LENGTH	1311
9.7.2	Enumeration Type Documentation	1311
9.7.2.1	eQMI_CAT_EVENT_REPORT_IND_TLV	1311
9.7.2.2	eQMI_CAT_EVENT_REPORT_IND_TLV_LENGTH	1311
9.7.3	Function Documentation	1311
9.7.3.1	UpkQmiCbkCatEventReportInd(BYTE *pMdmResp, struct QmiCbkCatEventStatusReportInd *pAipResp)	1311
9.8	qaCbkSwiOmaDmEventReportInd.h File Reference	1311
9.8.1	Macro Definition Documentation	1312
9.8.1.1	QMI_SWIOMA_DM_CONFIG	1312
9.8.1.2	QMI_SWIOMA_DM_FOTA	1312
9.8.1.3	QMI_SWIOMA_DM_NOT	1312
9.8.2	Enumeration Type Documentation	1312
9.8.2.1	eQMI_SWIOMA_DM_EVENT_REPORT_IND	1312
9.8.3	Function Documentation	1312
9.8.3.1	UpkQmiCbkSwiOmaDmEventReportInd(BYTE *pMdmResp, struct QmiCbkSwiOmaDmEventStatusReportInd *pApiResp)	1312
9.8.3.2	UpkQmiCbkSwiOmaDmEventReportIndExt(BYTE *pMdmResp, struct QmiCbkSwiOmaDmEventStatusReportInd *pApiResp)	1312
9.9	qaGobiApiAudio.h File Reference	1312
9.9.1	Detailed Description	1313
9.9.2	Function Documentation	1313
9.9.2.1	SLQSGetAudioPathConfig(GetAudioPathConfigReq *pGetAudioPathConfigReq, GetAudioPathConfigResp *pGetAudioPathConfigResp)	1313
9.9.2.2	SLQSGetAudioProfile(GetAudioProfileReq *pGetAudioProfileReq, GetAudioProfileResp *pGetAudioProfileResp)	1314

9.9.2.3	SLQSGetAudioVoTLBConfig(GetAudioVoTLBConfigReq *pGetAudioVoTLB↔ CfgReq, GetAudioVoTLBConfigResp *pGetAudioVoTLBCfgResp)	1314
9.9.2.4	SLQSSetAudioPathConfig(SetAudioPathConfigReq *pSetAudioPathConfigReq)	1315
9.9.2.5	SLQSSetAudioProfile(SetAudioProfileReq *pSetAudioProfileReq)	1316
9.9.2.6	SLQSSetAudioVoTLBConfig(SetAudioVoTLBConfigReq *pSetAudioVoTLB↔ CfgReq, SetAudioVoTLBConfigResp *pSetAudioVoTLBCfgResp)	1316
9.10	qaGobiApiCat.h File Reference	1317
9.10.1	Detailed Description	1317
9.10.2	Function Documentation	1317
9.10.2.1	CATSendEnvelopeCommand(ULONG cmdID, ULONG dataLen, BYTE *pData)	1317
9.10.2.2	CATSendTerminalResponse(ULONG refID, ULONG dataLen, BYTE *pData)	1318
9.11	qaGobiApiCbK.h File Reference	1318
9.11.1	Detailed Description	1326
9.11.2	Macro Definition Documentation	1326
9.11.2.1	CBK_DISABLE_EVENT	1326
9.11.2.2	CBK_ENABLE_EVENT	1326
9.11.2.3	CBK_NOCHANGE	1326
9.11.2.4	DEREGISTER_EVENT	1326
9.11.2.5	DEREGISTER_SRV	1326
9.11.2.6	DHCP_MAX_NUM_OPTIONS	1326
9.11.2.7	DHCP_OPTION_DATA_BUF_SIZE	1326
9.11.2.8	EVENT_MASK_CARD	1326
9.11.2.9	EVENT_MASK_DEREGISTER_ALL	1326
9.11.2.10	EVENT_MASK_PHY_SLOT_STATUS	1326
9.11.2.11	FIRST_INSTANCE	1326
9.11.2.12	INVALID_INSTACNE	1326
9.11.2.13	IPV4	1327
9.11.2.14	IPV4V6	1327
9.11.2.15	IPV6	1327
9.11.2.16	LOC_EVENT_MASK_ENG_STATE	1327
9.11.2.17	LOC_EVENT_MASK_GNSS_SV_INFO	1327

9.11.2.18 LOC_EVENT_MASK_INJECT_TIME	1327
9.11.2.19 LOC_EVENT_MASK_SENSOR_STREAM	1327
9.11.2.20 LOC_EVENT_MASK_TIME_SYNC	1327
9.11.2.21 LOC_EVENT_POSITION_REPORT	1327
9.11.2.22 MAX_MITIGATION_DEV_ID_LEN	1327
9.11.2.23 MAX_NO_OF_APPLICATIONS	1327
9.11.2.24 MAX_NO_OF_CALLS	1327
9.11.2.25 MAX_NO_OF_FILES	1327
9.11.2.26 MAX_NO_OF_SLOTS	1327
9.11.2.27 MAX_NO_OF_UUSINFO	1327
9.11.2.28 MAX_PATH_LENGTH	1327
9.11.2.29 MAX_RADIO_INTERFACE_LIST	1327
9.11.2.30 MAXUSSDLENGTH	1327
9.11.2.31 NAS_SRV	1327
9.11.2.32 NUM_OF_SET	1327
9.11.2.33 PDS_SRV	1327
9.11.2.34 QMI_ETWS_MAX_PAYLOAD_LENGTH	1327
9.11.2.35 QMI_MAX_VOICE_NUMBER_LENGTH	1327
9.11.2.36 QMI_WMS_MAX_PAYLOAD_LENGTH	1328
9.11.2.37 REGISTER_EVENT	1328
9.11.2.38 REGISTER_SRV	1328
9.11.2.39 SECOND_INSTANCE	1328
9.11.2.40 SIGSTRENGTH_THRESHOLD_ARR_SZ	1328
9.11.2.41 THIRD_INSTANCE	1328
9.11.2.42 USSD_DCS_8BIT	1328
9.11.2.43 USSD_DCS_ASCII	1328
9.11.2.44 USSD_DCS_UCS2	1328
9.11.2.45 VOICE_SRV	1328
9.11.2.46 WDS_SRV	1328
9.11.3 Typedef Documentation	1328

9.11.3.1	accelAcceptReady	1328
9.11.3.2	accelTempAcceptReady	1329
9.11.3.3	eDevState	1329
9.11.3.4	eSMSEventType	1329
9.11.3.5	gpsTime	1329
9.11.3.6	gyroAcceptReady	1330
9.11.3.7	gyroTempAcceptReady	1330
9.11.3.8	LteNasReleaseInfo	1330
9.11.3.9	modemTempNotification	1331
9.11.3.10	packetSrvStatus	1331
9.11.3.11	precisionDilution	1332
9.11.3.12	ResetInfoNotification	1333
9.11.3.13	sensorDataUsage	1333
9.11.3.14	sessionInformation	1334
9.11.3.15	sessionInformationExt	1334
9.11.3.16	SMSAsyncRawSend	1334
9.11.3.17	SMSCAddressInfo	1335
9.11.3.18	SMSEtwsMessageInfo	1336
9.11.3.19	SMSEtwsPlmnInfo	1336
9.11.3.20	SMSEventInfo	1336
9.11.3.21	SMSMessageModelInfo	1337
9.11.3.22	SMSMTMessageInfo	1337
9.11.3.23	SMSONIMSInfo	1337
9.11.3.24	SMSTransferRouteMTMessageInfo	1338
9.11.3.25	svUsedforFix	1338
9.11.3.26	SwiOTAMsg	1339
9.11.3.27	tFNActivationStatus	1339
9.11.3.28	tFNAllCallStatus	1340
9.11.3.29	tFNASwiLTECphyCallInfo	1340
9.11.3.30	tFNASwiOTAMsg	1340

9.11.3.31 tFNAsyncRawSend	1340
9.11.3.32 tFNBandPreference	1340
9.11.3.33 tFNBstAvailPos	1342
9.11.3.34 tFNCATEvent	1342
9.11.3.35 tFNCbkUimSlotStatusChangeInd	1342
9.11.3.36 tFNDDataCapabilities	1342
9.11.3.37 tFNDDataSysStatus	1343
9.11.3.38 tFNDelAssistData	1343
9.11.3.39 tFNDeviceStateChange	1343
9.11.3.40 tFNDHCPv4ClientLeaseStatus	1344
9.11.3.41 tFNDTMFEvent	1344
9.11.3.42 tFNDUNCallInfo	1344
9.11.3.43 tFNEventPosition	1344
9.11.3.44 tFNFwDidCompletion	1344
9.11.3.45 tFNGnssSvInfo	1345
9.11.3.46 tFNHDRPersonaity	1345
9.11.3.47 tFNImsaPdpStatus	1345
9.11.3.48 tFNImsaRatStatus	1346
9.11.3.49 tFNImsaRegStatus	1346
9.11.3.50 tFNImsaSvcStatus	1346
9.11.3.51 tFNImRegMgrConfig	1346
9.11.3.52 tFNImSIPConfig	1347
9.11.3.53 tFNImSMSConfig	1347
9.11.3.54 tFNImUserConfig	1347
9.11.3.55 tFNImVoIPConfig	1347
9.11.3.56 tFNInfoRec	1347
9.11.3.57 tFNInjectPosition	1348
9.11.3.58 tFNInjectSensorData	1348
9.11.3.59 tFNInjectTimeStatus	1348
9.11.3.60 tFNInjectUTCTime	1348

9.11.3.61 tFNLURreject	1348
9.11.3.62 tFNMemoryFull	1349
9.11.3.63 tFNMessageWaiting	1350
9.11.3.64 tFNMiniLvlRpt	1350
9.11.3.65 tFNMobileIPStatus	1350
9.11.3.66 tFNModemTempInfo	1350
9.11.3.67 tFNNasTimer	1350
9.11.3.68 tFNNet	1351
9.11.3.69 tFNNetworkTime	1351
9.11.3.70 tFNNewGPS	1351
9.11.3.71 tFNNewNMEA	1352
9.11.3.72 tFNNewRMTransferStatistics	1352
9.11.3.73 tFNNewSMS	1353
9.11.3.74 tFNOMADMState	1353
9.11.3.75 tFNOpMode	1354
9.11.3.76 tFNOTASPStatus	1354
9.11.3.77 tFNPacketSrvState	1354
9.11.3.78 tFNPDSState	1354
9.11.3.79 tFNPower	1355
9.11.3.80 tFNPrivacyChange	1355
9.11.3.81 tFNQosNWStatus	1355
9.11.3.82 tFNQosPriEvent	1355
9.11.3.83 tFNQosStatus	1356
9.11.3.84 tFNRankIndicator	1357
9.11.3.85 tFNResetInfo	1357
9.11.3.86 tFNRFInfo	1357
9.11.3.87 tFNRoamingIndicator	1358
9.11.3.88 tFNSDKTerminated	1358
9.11.3.89 tFNSensorStreaming	1358
9.11.3.90 tFNServingSystem	1358

9.11.3.91 tFNSetCradleMount	1359
9.11.3.92 tFNSetEngineState	1359
9.11.3.93 tFNSetEventTimeSync	1359
9.11.3.94 tFNSetExtPowerConfig	1359
9.11.3.95 tFNSigInfo	1359
9.11.3.96 tFNSignalStrength	1359
9.11.3.97 tFNSLQSOMADMAAlert	1359
9.11.3.98 tFNSLQSQOSEvent	1360
9.11.3.99 tFNSLQSSessionState	1360
9.11.3.100 tFNSLQSSignalStrengths	1360
9.11.3.101 tFNSLQSWDSEvent	1360
9.11.3.102 FNSMSEvents	1361
9.11.3.103 FNSUPInfo	1361
9.11.3.104 FNSUPSNotification	1361
9.11.3.105 FNSysInfo	1361
9.11.3.106 FNSysSelectionPref	1362
9.11.3.107 FNtransLayerInfo	1362
9.11.3.108 FNtransNWRegInfo	1362
9.11.3.109 FNUIMRefresh	1362
9.11.3.110 FNUIMStatusChangeInfo	1362
9.11.3.111 tFNUSSDNotification	1363
9.11.3.112 FNUSSDNoWaitIndication	1363
9.11.3.113 FNUSSDRelease	1363
9.11.3.114 transLayerNotification	1363
9.11.3.115 transNWRegInfoNotification	1364
9.11.4 Enumeration Type Documentation	1364
9.11.4.1 device_state_enum	1364
9.11.4.2 eQaQMIService	1364
9.11.4.3 SMSEventType	1365
9.11.5 Function Documentation	1365

9.11.5.1	iSetCATEventCallback(tFNCATEvent pCallback)	1365
9.11.5.2	iSetSignalStrengthCallback(tFNSignalStrength pCallback)	1365
9.11.5.3	iSLQSSetDUNCallInfoCallback(tFNDUNCallInfo pCallback)	1365
9.11.5.4	iSLQSSetSignalStrengthsCallback(tFNSLQSSignalStrengths pCallback)	1365
9.11.5.5	iSLQSSetWdsFirstInstEventCallback(tFNSLQSWDSEvent pCallback)	1365
9.11.5.6	iSLQSSetWdsSecondInstEventCallback(tFNSLQSWDSEvent pCallback)	1365
9.11.5.7	iSLQSSetWdsThirdInstEventCallback(tFNSLQSWDSEvent pCallback)	1365
9.11.5.8	iSLQSSetWdsXferStatsFirstInstCallback(tFNSLQSWDSEvent pCallback)	1365
9.11.5.9	iSLQSSetWdsXferStatsSecondInstCallback(tFNSLQSWDSEvent pCallback)	1365
9.11.5.10	SetActivationStatusCallback(tFNActivationStatus pCallback)	1365
9.11.5.11	SetCATEventCallback(tFNCATEvent pCallback, ULONG eventMask, ULONG *pErrorMask)	1366
9.11.5.12	SetDataCapabilitiesCallback(tFNDataCapabilities pCallback)	1367
9.11.5.13	SetDeviceStateChangeCbK(tFNDeviceStateChange pCallback)	1367
9.11.5.14	SetFwDldCompletionCbK(tFNFwDldCompletion pCallback)	1368
9.11.5.15	SetGPSCallback(tFNNewGPS pCallback)	1368
9.11.5.16	SetLocBestAvailPosCallback(tFNBestAvailPos pCallback)	1368
9.11.5.17	SetLocCradleMountCallback(tFNSetCradleMount pCallback)	1369
9.11.5.18	SetLocDeleteAssistDataCallback(tFNDeAssistData pCallback)	1369
9.11.5.19	SetLocEngineStateCallback(tFNSetEngineState pCallback)	1369
9.11.5.20	SetLocEventPositionCallback(tFNEventPosition pCallback)	1370
9.11.5.21	SetLocEventTimeSyncCallback(tFNSetEventTimeSync pCallback)	1370
9.11.5.22	SetLocGnssSvInfoCallback(tFNGnssSvInfo pCallback)	1370
9.11.5.23	SetLocInjectSensorDataCallback(tFNInjectSensorData pCallback)	1371
9.11.5.24	SetLocInjectTimeCallback(tFNInjectTimeStatus pCallback)	1371
9.11.5.25	SetLocOpModeCallback(tFNOpMode pCallback)	1371
9.11.5.26	SetLocSensorStreamingCallback(tFNSensorStreaming pCallback)	1372
9.11.5.27	SetLocSetExtPowerConfigCallback(tFNSetExtPowerConfig pCallback)	1372
9.11.5.28	SetLURejectCallback(tFNLUReject pCallback)	1372
9.11.5.29	SetMobileIPStatusCallback(tFNMobileIPStatus pCallback)	1373
9.11.5.30	SetNasLTECphyCalIndCallback(tFNASwiLTECphyCalInfo pCallback)	1373

9.11.5.31 SetNetChangeCbk(BYTE instance, tFNNet pCallback, ULONG loMark, ULONG hiMark, ULONG period)	1374
9.11.5.32 SetNewSMSCallback(tFNNewSMS pCallback)	1374
9.11.5.33 SetNMEACallback(tFNNewNMEA pCallback)	1375
9.11.5.34 SetOMADMStateCallback(tFNOMADMState pCallback)	1375
9.11.5.35 SetPDSSStateCallback(tFNPDSState pCallback)	1376
9.11.5.36 SetPowerCallback(tFNPower pCallback)	1376
9.11.5.37 SetRankIndicatorCallback(tFNRankIndicator pCallback)	1376
9.11.5.38 SetRFInfoCallback(tFNRFInfo pCallback)	1376
9.11.5.39 SetRMTransferStatisticsCallback(tFNNewRMTransferStatistics pCallback)	1377
9.11.5.40 SetRoamingIndicatorCallback(tFNRoamingIndicator pCallback)	1377
9.11.5.41 SetSignalStrengthCallback(tFNSignalStrength pCallback, BYTE thresholdsSize, INT8 *pThresholds)	1378
9.11.5.42 SetSLQSOMADMAAlertCallback(tFNSLQSOMADMAAlert pCallback)	1379
9.11.5.43 SetSLQSOMADMAAlertCallbackExt(tFNSLQSOMADMAAlert pCallback)	1379
9.11.5.44 SetUimSlotStatusChangeCallback(tFNCbkUimSlotStatusChangeInd pCallback) .	1379
9.11.5.45 SetUSSDNotificationCallback(tFNUSSDNotification pCallback)	1380
9.11.5.46 SetUSSDNoWaitIndicationCallback(tFNUSSDNoWaitIndication pCallback) . . .	1380
9.11.5.47 SetUSSDReleaseCallback(tFNUSSDRelease pCallback)	1381
9.11.5.48 SLQSNasNetworkTimeCallBack(tFNNetworkTime pCallback)	1381
9.11.5.49 SLQSNasSigInfo2CallBack(tFNSigInfo pCallback, setSignalStrengthInfo *pSigInfo2)	1382
9.11.5.50 SLQSNasSigInfoCallBack(tFNSigInfo pCallback, sigInfo *pSigInfo)	1382
9.11.5.51 SLQSNasSwiOTAMessageCallback(NasSwiIndReg *req, tFNASwiOTAMsg pCallback)	1383
9.11.5.52 SLQSNasSysInfoCallBack(tFNSysInfo pCallback)	1383
9.11.5.53 SLQSNasTimerCallback(tFNNasTimer pCallback)	1384
9.11.5.54 SLQSSetBandPreferenceCbk(tFNBandPreference pCallback)	1384
9.11.5.55 SLQSSetDataSystemStatusCallback(tFNDataSysStatus pCallback)	1385
9.11.5.56 SLQSSetDHCPv4ClientLeaseStatusCallback(BYTE instance, tFNDHCPv4ClientLeaseStatus pCallback)	1385
9.11.5.57 SLQSSetDUNCallInfoCallback(BYTE StatsPeriod, tFNDUNCallInfo pCallback) .	1385
9.11.5.58 SLQSSetIMSAPdpStatusCallback(tFNImsaPdpStatus pCallback)	1386

9.11.5.59 SLQSSetIMSAStatusCallback(tFNImsaRatStatus pCallback)	1386
9.11.5.60 SLQSSetIMSARegStatusCallback(tFNImsaRegStatus pCallback)	1387
9.11.5.61 SLQSSetIMSASvcStatusCallback(tFNImsaSvcStatus pCallback)	1387
9.11.5.62 SLQSSetIMSSMSConfigCallback(tFNImSMSConfig pCallback)	1388
9.11.5.63 SLQSSetIMSUserConfigCallback(tFNImUserConfig pCallback)	1388
9.11.5.64 SLQSSetIMSVoIPConfigCallback(tFNImVoIPConfig pCallback)	1389
9.11.5.65 SLQSSetLocInjectPositionCallback(tFNInjectPosition pCallback)	1389
9.11.5.66 SLQSSetLocInjectUTCTimeCallback(tFNInjectUTCTime pCallback)	1390
9.11.5.67 SLQSSetModemTempCallback(tFNModemTempInfo pCallback)	1390
9.11.5.68 SLQSSetPacketSrvStatusCallback(tFNPacketSrvState pCallback)	1390
9.11.5.69 SLQSSetQosEventCallback(BYTE instance, tFNSLQSQOSEvent pCallback) . .	1391
9.11.5.70 SLQSSetQosNWStatusCallback(tFNQosNWStatus pCallback)	1391
9.11.5.71 SLQSSetQosPriEventCallback(tFNQosPriEvent pCallback)	1392
9.11.5.72 SLQSSetQosStatusCallback(BYTE instance, tFNQosStatus pCallback)	1392
9.11.5.73 SLQSSetRegMgrConfigCallback(tFNImRegMgrConfig pCallback)	1393
9.11.5.74 SLQSSetSDKTerminatedCallback(tFNSDKTerminated pCallback)	1393
9.11.5.75 SLQSSetServingSystemCallback(tFNServingSystem pCallback)	1394
9.11.5.76 SLQSSetSessionStateCallback(tFNSLQSSessionState pCallback)	1394
9.11.5.77 SLQSSetSignalStrengthsCallback(tFNSLQSSignalStrengths pCallback, struct SLQSSignalStrengthsIndReq *pSLQSSignalStrengthsIndReq)	1395
9.11.5.78 SLQSSetSIPConfigCallback(tFNImSIPConfig pCallback)	1395
9.11.5.79 SLQSSetSMSEventCallback(tFNSMSEvents pCallback)	1396
9.11.5.80 SLQSSetSwiGetResetInfoCallback(tFNResetInfo pCallback)	1396
9.11.5.81 SLQSSetSwiHDRPersCallback(tFNHDRPersonality pCallback)	1396
9.11.5.82 SLQSSetSysSelectionPrefCallBack(tFNSysSelectionPref pCallback)	1397
9.11.5.83 SLQSSetTransLayerInfoCallback(tFNtransLayerInfo pCallback)	1397
9.11.5.84 SLQSSetTransNWRegInfoCallback(tFNtransNWRegInfo pCallback)	1398
9.11.5.85 SLQSSetWdsEventCallback(tFNSLQSWDSEvent pCallback, BYTE interval, B↔ YTE instanceid, BYTE ipfamily)	1398
9.11.5.86 SLQSSetWdsTransferStatisticCallback(tFNSLQSWDSEvent pXferStatsCb, BY↔ TE interval, BYTE instanceid, BYTE ipfamily)	1399

9.11.5.87 SLQSTmdMitigationLvIRptCallback(TmdMitigationLvIRpt *req, tFNMitLvIRpt pCallback)	1400
9.11.5.88 SLQSUIMSetRefreshCallBack(tFNUIMRefresh pCallback)	1400
9.11.5.89 SLQSUIMSetStatusChangeCallBack(tFNUIMStatusChangeInfo pCallback)	1401
9.11.5.90 SLQSVoiceInfoRecCallback(tFNInfoRec pCallback)	1401
9.11.5.91 SLQSVoiceSetAllCallStatusCallBack(tFNAllCallStatus pCallback)	1402
9.11.5.92 SLQSVoiceSetDTMFEventCallBack(tFNDTMFEvent pCallback)	1402
9.11.5.93 SLQSVoiceSetOTASPStatusCallBack(tFNOTASPStatus pCallback)	1402
9.11.5.94 SLQSVoiceSetPrivacyChangeCallBack(tFNPrivacyChange pCallback)	1403
9.11.5.95 SLQSVoiceSetSUPSCallBack(tFNSUPSInfo pCallback)	1403
9.11.5.96 SLQSVoiceSetSUPSNotificationCallback(tFNSUPSNotification pCallback)	1404
9.11.5.97 SLQSWmsAsyncRawSendCallBack(tFNAsyncRawSend pCallback)	1404
9.11.5.98 SLQSWmsMemoryFullCallBack(tFNMemoryFull pCallback)	1405
9.11.5.99 SLQSWmsMessageWaitingCallBack(tFNMessageWaiting pCallback)	1405
9.12 qaGobiApiDcs.h File Reference	1406
9.12.1 Detailed Description	1406
9.12.2 Macro Definition Documentation	1406
9.12.2.1 LEN	1406
9.12.2.2 PORTNAM_LEN	1406
9.12.3 Function Documentation	1407
9.12.3.1 QCWWAN2kConnect(CHAR *pDeviceID, CHAR *pDeviceKey)	1407
9.12.3.2 QCWWAN2kEnumerateDevices(BYTE *pDevicesSize, BYTE *pDevices)	1407
9.12.3.3 QCWWAN2kGetConnectedDeviceID(ULONG deviceIDSize, CHAR *pDeviceID, ULONG deviceKeySize, CHAR *pDeviceKey)	1408
9.12.3.4 QCWWANConnect(CHAR *pDeviceID, CHAR *pDeviceKey)	1409
9.12.3.5 QCWWANDisconnect()	1409
9.12.3.6 QCWWANEnumerateDevices(BYTE *pDevicesSize, BYTE *pDevices)	1409
9.12.3.7 SetSDKImagePath(LPCSTR pPath)	1410
9.12.3.8 SLQSGetDeviceMode(BYTE *pDeviceMode)	1410
9.12.3.9 SLQSGetNetStatistic(struct NetStats *pNetStatistic, BYTE instance)	1411
9.12.3.10 SLQSGetUsbPortNames(struct DcsUsbPortNames *pUsbPortNames)	1411

9.12.3.11 SLQSKillSDKProcess()	1412
9.12.3.12 SLQSSetLoggingMask(BYTE mask)	1412
9.12.3.13 SLQSStart(BYTE modem_index, CHAR *sn)	1413
9.12.3.14 SLQSStart_AVAgent(BYTE modem_index)	1413
9.12.3.15 SLQSStartSrv(BYTE action, BYTE mask)	1414
9.13 qaGobiApiDms.h File Reference	1414
9.13.1 Detailed Description	1417
9.13.2 Macro Definition Documentation	1417
9.13.2.1 MAX_CUST_ID_LEN	1417
9.13.2.2 MAX_CUST_VALUE_LEN	1417
9.13.2.3 MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH	1417
9.13.2.4 MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH	1417
9.13.2.5 MAX_FSN_LENGTH	1417
9.13.3 Typedef Documentation	1417
9.13.3.1 custFeaturesInfo	1417
9.13.3.2 custFeaturesSetting	1419
9.13.3.3 dmsCurrentPRLInfo	1420
9.13.3.4 ERIFileparams	1420
9.13.3.5 serialNumbersInfo	1421
9.13.3.6 SLQSSwiGetHostDevInfoParams	1421
9.13.3.7 SLQSSwiGetOSInfoParams	1422
9.13.3.8 SLQSSwiGetSerialNoExtParams	1423
9.13.3.9 SLQSSwiSetHostDevInfoParams	1423
9.13.3.10 SLQSSwiSetOSInfoParams	1424
9.13.4 Function Documentation	1424
9.13.4.1 ActivateAutomatic(CHAR *pActivationCode)	1424
9.13.4.2 GetActivationState(ULONG *pActivationState)	1424
9.13.4.3 GetDeviceCapabilities(ULONG *pMaxTXChannelRate, ULONG *pMaxRX↵ ChannelRate, ULONG *pDataServiceCapability, ULONG *pSimCapability, UL↵ ONG *pRadiolfacesSize, BYTE *pRadiolfaces)	1425
9.13.4.4 GetFirmwareRevision(BYTE stringSize, CHAR *pString)	1426

9.13.4.5	GetFirmwareRevisions(BYTE amssSize, CHAR *pAMSSString, BYTE bootSize, CHAR *pBootString, BYTE priSize, CHAR *pPRIString)	1427
9.13.4.6	GetHardwareRevision(BYTE stringSize, CHAR *pString)	1427
9.13.4.7	GetIMSI(BYTE stringSize, CHAR *pString)	1428
9.13.4.8	GetManufacturer(BYTE stringSize, CHAR *pString)	1429
9.13.4.9	GetModelID(BYTE stringSize, CHAR *pString)	1429
9.13.4.10	GetNetworkTime(ULONGLONG *pTimeStamp, ULONG *pTimeSource)	1430
9.13.4.11	GetOfflineReason(ULONG *pReasonMask, ULONG *pbPlatform)	1430
9.13.4.12	GetPower(ULONG *pPowerMode)	1431
9.13.4.13	GetPRLVersion(WORD *pPRLVersion)	1431
9.13.4.14	GetSerialNumbers(BYTE esnSize, CHAR *pESNString, BYTE imeiSize, CHAR *pIMEIString, BYTE meidSize, CHAR *pMEIDString)	1432
9.13.4.15	GetVoiceNumber(BYTE voiceNumberSize, CHAR *pVoiceNumber, BYTE minSize, CHAR *pMIN)	1433
9.13.4.16	ResetToFactoryDefaults(CHAR *pSPC)	1433
9.13.4.17	SetPower(ULONG powerMode)	1434
9.13.4.18	SLQSDmsSwiGetResetInfo(dmsSwiGetResetInfo *pGetResetInfoResp)	1434
9.13.4.19	SLQSDmsSwiIndicationRegister(dmsIndicationRegisterReq *pIndicationRegisterReq)	1435
9.13.4.20	SLQSGetBandCapabilities(BandCapabilityResp *pBandCapability)	1435
9.13.4.21	SLQSGetBandCapability(ULONGLONG *pBandCapability)	1435
9.13.4.22	SLQSGetCurrentPRLInfo(dmsCurrentPRLInfo *pCurrentPRLInfo)	1437
9.13.4.23	SLQSGetCustFeatures(custFeaturesInfo *pCustFeaturesInfo)	1437
9.13.4.24	SLQSGetCustFeaturesV2(getCustomFeatureV2 *pGetCustomFeatureV2)	1438
9.13.4.25	SLQSGetERIFile(ERIFileparams *pERIFileparams)	1438
9.13.4.26	SLQSGetSerialNumbers(serialNumbersInfo *pSerialNumbersInfo)	1439
9.13.4.27	SLQSSetCustFeatures(custFeaturesSetting *pCustFeaturesSetting)	1439
9.13.4.28	SLQSSetCustFeaturesV2(setCustomSettingV2 *pSetCustSetting)	1439
9.13.4.29	SLQSSwiClearDyingGaspStatistics()	1440
9.13.4.30	SLQSSwiGetCrashAction(BYTE *pDevCrashState)	1440
9.13.4.31	SLQSSwiGetCrashInfo(BYTE *pClear, CrashInfoParams *pCrashInfoParams)	1441
9.13.4.32	SLQSSwiGetDyingGaspCfg(getDyingGaspCfg *pConfig)	1441

9.13.4.33 SLQSSwiGetDyingGaspStatistics(getDyingGaspStatistics *pStatistics)	1441
9.13.4.34 SLQSSwiGetFSN(FactorySequenceNumber *pFSNumber)	1442
9.13.4.35 SLQSSwiGetFwUpdateStatus(FirmwareUpdatStat *pFirmwareUpdatStat)	1442
9.13.4.36 SLQSSwiGetHostDevInfo(SLQSSwiGetHostDevInfoParams *pGetHostDevInfoParams)	1443
9.13.4.37 SLQSSwiGetOSInfo(SLQSSwiGetOSInfoParams *pParams)	1443
9.13.4.38 SLQSSwiGetSerialNoExt(SLQSSwiGetSerialNoExtParams *pParams)	1444
9.13.4.39 SLQSSwiGetUSBComp(USBCompParams *pUSBCompParams)	1444
9.13.4.40 SLQSSwiSetCrashAction(BYTE crashActionParams)	1445
9.13.4.41 SLQSSwiSetDyingGaspCfg(setDyingGaspCfg *pConfig)	1445
9.13.4.42 SLQSSwiSetHostDevInfo(SLQSSwiSetHostDevInfoParams *pSetHostDevInfoParams)	1445
9.13.4.43 SLQSSwiSetOSInfo(SLQSSwiSetOSInfoParams *pParams)	1446
9.13.4.44 SLQSSwiSetUSBComp(USBCompConfig *pUSBCompConfig)	1446
9.13.4.45 SLQSUIMGetState(ULONG *pUIMState)	1447
9.13.4.46 UIMChangePIN(ULONG id, CHAR *pOldValue, CHAR *pNewValue, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)	1448
9.13.4.47 UIMGetControlKeyStatus(ULONG id, ULONG *pStatus, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)	1448
9.13.4.48 UIMGetICCID(BYTE stringSize, CHAR *pString)	1449
9.13.4.49 UIMGetPINStatus(ULONG id, ULONG *pStatus, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)	1450
9.13.4.50 UIMSetControlKeyProtection(ULONG id, ULONG status, CHAR *pValue, ULONG *pVerifyRetriesLeft)	1451
9.13.4.51 UIMSetPINProtection(ULONG id, ULONG bEnable, CHAR *pValue, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)	1452
9.13.4.52 UIMUnblockControlKey(ULONG id, CHAR *pValue, ULONG *pUnblockRetriesLeft)	1452
9.13.4.53 UIMUnblockPIN(ULONG id, CHAR *pPUKValue, CHAR *pNewValue, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)	1453
9.13.4.54 UIMVerifyPIN(ULONG id, CHAR *pValue, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)	1454
9.13.4.55 ValidateSPC(CHAR *pSPC)	1455
9.14 qagobiApiFms.h File Reference	1455
9.14.1 Detailed Description	1458

9.14.2 Macro Definition Documentation	1459
9.14.2.1 BUILD_ID_LEN	1459
9.14.2.2 BUILD_ID_MAX_LEN	1459
9.14.2.3 DEVICE_OFFLINE	1459
9.14.2.4 DEVICE_RESET	1459
9.14.2.5 DEVICE_SHUTDOWN	1459
9.14.2.6 FIRMWARE_UPDATE_FAIL	1459
9.14.2.7 FIRMWARE_UPDATE_SUCCESS	1459
9.14.2.8 FIRMWARE_UPGRADE_SUCCESS	1459
9.14.2.9 G3K_FIRMWARE_DOWNLOAD	1459
9.14.2.10 GOBI_LISTENTRIES_MAX	1459
9.14.2.11 GOBI_MBN_BUILD_ID_STR_LEN	1459
9.14.2.12 GOBI_MBN_IMG_ID_STR_LEN	1459
9.14.2.13 GOBI_SET_IMG_PREF_RSPLN	1459
9.14.2.14 IMG_ID_LEN	1459
9.14.2.15 IMGDETAILS_LEN	1459
9.14.2.16 MAX_IMAGE_IDE_ELEMENTS	1459
9.14.2.17 PRI_UPDATE_FAIL	1459
9.14.2.18 SLQSFWINFO_APPVERSION_SZ	1459
9.14.2.19 SLQSFWINFO_BOOTVERSION_SZ	1459
9.14.2.20 SLQSFWINFO_CARRIER_SZ	1459
9.14.2.21 SLQSFWINFO_CUR_CARR_NAME	1459
9.14.2.22 SLQSFWINFO_CUR_CARR_REV	1459
9.14.2.23 SLQSFWINFO_MODELID_SZ	1460
9.14.2.24 SLQSFWINFO_PACKAGEID_SZ	1460
9.14.2.25 SLQSFWINFO_PRIVERSION_SZ	1460
9.14.2.26 SLQSFWINFO_SKU_SZ	1460
9.14.2.27 SPKG_FIRMWARE_DOWNLOAD	1460
9.14.2.28 UNIQUE_ID_LEN	1460
9.14.3 Enumeration Type Documentation	1460

9.14.3.1	eGobiDeviceSeries	1460
9.14.3.2	eGobiImageCarrier	1460
9.14.3.3	eGobiImageGPS	1461
9.14.3.4	eGobiImageRegion	1462
9.14.3.5	eGobiImageTech	1462
9.14.4	Function Documentation	1462
9.14.4.1	DeleteStoredImage(ULONG imageInfoSize, BYTE *pImageInfo)	1462
9.14.4.2	DownloadToSlot(CHAR *path, struct slqsfwinfo_s fwImgInfo, struct slqsfwinfo_s priImgInfo, BYTE slot, BYTE forceDownload)	1463
9.14.4.3	eGetDeviceSeries(struct sGetDeviceSeriesResult *)	1463
9.14.4.4	GetImagesPreference(ULONG *pImageListSize, struct PrefImageList *pImageList)	1463
9.14.4.5	GetImageStore(WORD imageStorePathSize, CHAR *pImageStorePath)	1464
9.14.4.6	GetStoredImages(ULONG *pImageListSize, BYTE *pImageList)	1464
9.14.4.7	SetImagesPreference(ULONG imageListSize, BYTE *pImageList, ULONG bForceDownload, BYTE modemIndex, ULONG *pImageTypesSize, BYTE *pImageTypes)	1465
9.14.4.8	SLQSDownloadFirmwareToSlot(CHAR *pPath, BYTE slot_index, BYTE force_download)	1466
9.14.4.9	SLQSGetBootVersionNumber(ULONG *bootversion)	1467
9.14.4.10	SLQSGetFirmwareInfo(struct qmifwinfo_s *pinfo)	1467
9.14.4.11	SLQSGetImageInfo(LPCSTR path, struct qmifwinfo_s *pinfo)	1468
9.14.4.12	SLQSGetImageInfo_9x15(LPCSTR path, BYTE imgType, struct slqsfwinfo_s *pinfo)	1468
9.14.4.13	SLQSGetImageInfoMC77xx(LPCSTR path, struct qmifwinfo_s *pinfo)	1469
9.14.4.14	SLQSGetImageInfoMC83xx(LPCSTR path, struct qmifwinfo_s *pinfo)	1470
9.14.4.15	SLQSGetValidFwPriCombinations(struct ImageList *pStoredImageList, ULONG *pValidCombinationSize, struct SWI_STRUCT_CarrierImage *pValidCombinations)	1470
9.14.4.16	SLQSIspkgFormatRequired(void)	1471
9.14.4.17	SLQSSetCrashStateCheckIgnore(BOOL ignore)	1471
9.14.4.18	SLQSSetSIMBasedImageSwitching(void)	1472
9.14.4.19	SLQSSetSpkgFormatRequired(BYTE isneeded)	1472
9.14.4.20	SLQSSwiGetAllCarrierImages(ULONG *pNumOfItems, struct SWI_STRUCT_CarrierImage *pCarrierImages, char *pFolderPath)	1472

9.14.4.21	SLQSSwiGetFirmwareCurr(CurrentImgList *pCurrentImgList)	1473
9.14.4.22	SLQSUgradeFirmware9x15(CHAR *pDestinationPath)	1474
9.14.4.23	upgrade_mc77xx_fw(LPCSTR path)	1475
9.14.4.24	UpgradeFirmware2k(CHAR *pDestinationPath)	1475
9.15	qaGobiApilms.h File Reference	1476
9.15.1	Detailed Description	1477
9.15.2	Function Documentation	1477
9.15.2.1	SLQSGetIMSSMSConfig(GetIMSSMSConfigParams *pGetIMSSMSConfigParams)	1477
9.15.2.2	SLQSGetIMSUserConfig(GetIMSUserConfigParams *pGetIMSUserConfigParams)	1477
9.15.2.3	SLQSGetIMSVoIPConfig(GetIMSVoIPConfigResp *pGetIMSVoIPConfigResp)	1478
9.15.2.4	SLQSGetRegMgrConfig(GetRegMgrConfigParams *pGetRegMgrConfigParams)	1478
9.15.2.5	SLQSGetSIPConfig(GetSIPConfigResp *pGetSIPConfigResp)	1479
9.15.2.6	SLQSImsConfigIndicationRegister(imsCfgIndRegisterInfo *pImsCfgIndRegisterInfo)	1479
9.15.2.7	SLQSSetIMSSMSConfig(SetIMSSMSConfigReq *pSetIMSSMSConfigReq, SetIMSSMSConfigResp *pSetIMSSMSConfigResp)	1480
9.15.2.8	SLQSSetIMSUserConfig(SetIMSUserConfigReq *pSetIMSUserConfigReq, SetIMSUserConfigResp *pSetIMSUserConfigResp)	1481
9.15.2.9	SLQSSetIMSVoIPConfig(SetIMSVoIPConfigReq *pSetIMSVoIPConfigReq, SetIMSVoIPConfigResp *pSetIMSVoIPConfigResp)	1481
9.15.2.10	SLQSSetRegMgrConfig(SetRegMgrConfigReq *pSetRegMgrConfigReq, SetRegMgrConfigResp *pSetRegMgrConfigResp)	1482
9.15.2.11	SLQSSetSIPConfig(SetSIPConfigReq *pSetSIPConfigReq, SetSIPConfigResp *pSetSIPConfigResp)	1482
9.16	qaGobiApilmsa.h File Reference	1483
9.16.1	Detailed Description	1483
9.16.2	Function Documentation	1483
9.16.2.1	SLQSGetIMSARegStatus(IMSARegistrationStatus *pIMSARegistrationStatus)	1483
9.16.2.2	SLQSGetIMSAServiceStatus(IMSAServiceStatus *pIMSAServiceStatus)	1484
9.16.2.3	SLQSGetIMSASupportedFields(WORD messageID, IMSASupportedFieldsResp *pIMSASupportedFieldsResp)	1485
9.16.2.4	SLQSGetIMSASupportedMsg(IMSASupportedMsgInfo *pIMSASupportedMsgInfo)	1485
9.16.2.5	SLQSRegisterIMSIndication(IMSIndRegisterInfo *pImsIndRegisterInfo)	1486

9.17	qaGobiApiLoc.h File Reference	1486
9.17.1	Detailed Description	1487
9.17.2	Macro Definition Documentation	1487
9.17.2.1	MAX_SENSOR_DATA_LEN	1487
9.17.2.2	MAX_TEMP_DATA_LEN	1487
9.17.3	Function Documentation	1487
9.17.3.1	SLQSLOCDeIAssData(LocDeIAssDataReq request)	1487
9.17.3.2	SLQSLOCEventRegister(LOCEventRegisterReqResp *pLOCEventRegisterReqResp)	1488
9.17.3.3	SLQSLOCGetBestAvailPos(ULONG xid)	1488
9.17.3.4	SLQSLOCInjectPosition(LocInjectPositionReq *pLocInjectPositionReq)	1489
9.17.3.5	SLQSLOCInjectSensorData(LocInjectSensorDataReq *pLocInjectSensorDataReq)	1489
9.17.3.6	SLQSLOCInjectUTCTime(ULONGLONG timeMsec, ULONG timeUncMsec)	1490
9.17.3.7	SLQSLOCSetCradleMountConfig(LocSetCradleMountReq *pLocSetCradleMountReq)	1490
9.17.3.8	SLQSLOCSetExtPowerState(LOCExtPowerStateReqResp *pLOCExtPowerStateReqResp)	1491
9.17.3.9	SLQSLOCSetOpMode(ULONG mode)	1491
9.17.3.10	SLQSLOCStart(LOCStartReq *pLOCStartReq)	1492
9.17.3.11	SLQSLOCStop(LOCStopReq *pLOCStopReq)	1492
9.17.3.12	SwiLocGetAutoStart(SwiLocGetAutoStartResp *resp)	1493
9.17.3.13	SwiLocSetAutoStart(SwiLocSetAutoStartReq *req)	1493
9.18	qaGobiApiNas.h File Reference	1493
9.18.1	Detailed Description	1499
9.18.2	Macro Definition Documentation	1499
9.18.2.1	IMSI_M_S1_LENGTH	1499
9.18.2.2	IMSI_M_S2_LENGTH	1499
9.18.2.3	MAX_DATA_SRV_CAPABILITIES	1499
9.18.2.4	MAX_DESCRIPTION_LENGTH	1499
9.18.2.5	MAX_PILOT_SETS	1499
9.18.2.6	MAX_SERV_SYSTEM_RADIO_INTERFACES	1499
9.18.2.7	NAM_NAME_LENGTH	1499

9.18.2.8	NAS_MAX_SCC_RX_INFO_INSTANCES	1499
9.18.2.9	NAS_SIG_INFO_MAX_TDSCDMA_THRESHOLDS_LIST_SIZE	1499
9.18.2.10	NAS_SIG_INFO_MIN_dB_FLOAT_VALUE	1499
9.18.2.11	NAS_SIG_INFO_MIN_dBm_FLOAT_VALUE	1499
9.18.2.12	PLMN_LENGTH	1499
9.18.2.13	SLQS_SS_INFO_LIST_MAX_ELEMENTS	1499
9.18.2.14	SLQS_SYSTEM_ID_SIZE	1499
9.18.2.15	UATISIZE	1499
9.18.3	Typedef Documentation	1499
9.18.3.1	SlqsNas3GppNetworkRAT	1499
9.18.3.2	slqsNetworkScanInfo	1500
9.18.3.3	sysSelectPrefInfo	1501
9.18.3.4	sysSelectPrefParams	1505
9.18.4	Enumeration Type Documentation	1511
9.18.4.1	_NAMS_RADIO_IF_TECHNOLOGY_	1511
9.18.4.2	eSYS_SRV_DOMAIN	1511
9.18.4.3	NAS_LTE_CPHY_CA_BW_NRB	1512
9.18.4.4	NAS_LTE_CPHY_SCELL_STATE	1512
9.18.5	Function Documentation	1512
9.18.5.1	GetACCOLC(BYTE *pACCOLC)	1512
9.18.5.2	GetANAAAAAuthenticationStatus(ULONG *pStatus)	1512
9.18.5.3	GetCDMANetworkParameters(BYTE *pSCI, BYTE *pSCM, BYTE *pReg← HomeSID, BYTE *pRegForeignSID, BYTE *pRegForeignNID, BYTE *pForce← Rev0, BYTE *pCustomSCP, ULONG *pProtocol, ULONG *pBroadcast, ULONG *pApplication, ULONG *pRoaming)	1513
9.18.5.4	GetHomeNetwork(WORD *pMCC, WORD *pMNC, BYTE nameSize, CHAR *p← Name, WORD *pSID, WORD *pNID)	1515
9.18.5.5	GetHomeNetwork3GPP2(WORD *pMCC, WORD *pMNC, BYTE nameSize, C← HAR *pName, WORD *pSID, WORD *pNID, WORD *pNw2MCC, WORD *p← Nw2MNC, BYTE *pNw2DescDisp, BYTE *pNw2DescEnc, BYTE *pNw2Desc← Len, BYTE *pNw2Name)	1516
9.18.5.6	GetNetworkPreference(ULONG *pTechnologyPref, ULONG *pDuration, ULONG *pPersistentTechnologyPref)	1518
9.18.5.7	GetRFInfo(BYTE *pInstanceSize, struct RFBandInfoElements *pRFBandInfo)	1518

9.18.5.8	GetServingNetwork(ULONG *pRegistrationState, ULONG *pCSDomain, ULONG *pPSDomain, ULONG *pRAN, BYTE *pRadiolfacesSize, BYTE *pRadiolfaces, ULONG *pRoaming, WORD *pMCC, WORD *pMNC, BYTE nameSize, CHAR *pName)	1519
9.18.5.9	GetServingNetworkCapabilities(BYTE *pDataCapsSize, BYTE *pDataCaps)	1520
9.18.5.10	GetSignalStrengths(ULONG *pArraySizes, INT8 *pSignalStrength, ULONG *pRadiolInterface)	1521
9.18.5.11	InitiateDomainAttach(ULONG action)	1522
9.18.5.12	InitiateNetworkRegistration(ULONG regType, WORD mcc, WORD mnc, ULONG rat)	1523
9.18.5.13	PerformNetworkScan(BYTE *pInstanceSize, BYTE *pInstances)	1523
9.18.5.14	SetACCOLC(CHAR *spc, BYTE accolc)	1524
9.18.5.15	SetCDMANetworkParameters(CHAR *pSPC, BYTE *pForceRev0, BYTE *pCustomSCP, ULONG *pProtocol, ULONG *pBroadcast, ULONG *pApplication, ULONG *pRoaming)	1524
9.18.5.16	SetNetworkPreference(ULONG technologyPref, ULONG duration)	1526
9.18.5.17	SLQSConfigSigInfo(sigInfo *pSigInfo)	1527
9.18.5.18	SLQSGetErrorRate(GetErrRateResp *pGetErrRateResp)	1527
9.18.5.19	SLQSGetNetworkTime(GetNetworkTimeResp *pGetNetworkTimeResp)	1528
9.18.5.20	SLQSGetOperatorNameData(nasOperatorNameResp *pOperatorNameData)	1528
9.18.5.21	SLQSGetPLMNName(nasPLMNNameReq *pPLMNNameReq, nasPLMNNameResp *pPLMNNameResp)	1529
9.18.5.22	SLQSGetServingSystem(qaQmiServingSystemParam *pServingSystem)	1529
9.18.5.23	SLQSGetSignalStrength(struct slqsSignalStrengthInfo *pSignalInfo)	1530
9.18.5.24	SLQSGetSysSelectionPref(sysSelectPrefInfo *pSysSelectPrefInfo)	1530
9.18.5.25	SLQSInitiateNetworkRegistration(nasInitNetworkReg *pNasInitNetRegistrationReg)	1531
9.18.5.26	SLQSNasConfigSigInfo2(setSignalStrengthInfo *pSetSignalStrengthInfo)	1531
9.18.5.27	SLQSNasGet3GPP2Subscription(nasGet3GPP2SubscriptionInfoReq *pGet3GPP2SubsInfoReq, nasGet3GPP2SubscriptionInfoResp *pGet3GPP2SubsInfoResp)	1532
9.18.5.28	SLQSNasGetCellLocationInfo(nasCellLocationInfoResp *pNasCellLocationInfoResp)	1532
9.18.5.29	SLQSNasGetHDRColorCode(nasGetHDRColorCodeResp *pGetHDRColorCodeResp)	1533
9.18.5.30	SLQSNASGetLTEPHYCaInfo(nasGetLTEPHYCa *pLTEPHYCa)	1533

9.18.5.31 SLQSNasGetSigInfo(nasGetSigInfoResp *pGetSigInfoResp)	1534
9.18.5.32 SLQSNasGetSysInfo(nasGetSysInfoResp *pGetSysInfoResp)	1534
9.18.5.33 SLQSNasGetTxRxInfo(nasGetTxRxInfoReq *pGetTxRxInfoReq, nasGetTxRxInfoResp *pGetTxRxInfoResp)	1535
9.18.5.34 SLQSNasIndicationRegister(BYTE systemSelectionInd, BYTE DDTMInd, BYTE servingSystemInd)	1535
9.18.5.35 SLQSNasIndicationRegisterExt(nasIndicationRegisterReq *pIndicationRegisterReq)	1536
9.18.5.36 SLQSNasIndicationRegisterLTECphyCa(BYTE *bStatus)	1537
9.18.5.37 SLQSNASSwiGetChannelLock(nasSwiGetChannelLockResp *pNasSwiGetChannelLockResp)	1537
9.18.5.38 SLQSNasSwiIndicationRegister(NasSwiIndReg *pIndRegReq)	1537
9.18.5.39 SLQSNasSwiModemStatus(swiModemStatusResp *pModemStatusResp)	1538
9.18.5.40 SLQSNASSwiSetChannelLock(nasSwiSetChannelLockReq *pNasSwiSetChannelLockReq)	1538
9.18.5.41 SLQSPerformNetworkScan(slqsNetworkScanInfo *pNetworkInfo)	1539
9.18.5.42 SLQSSetBandPreference(ULONGLONG bandpreference)	1539
9.18.5.43 SLQSSetSysSelectionPref(sysSelectPrefParams *pSysSelectPrefParams)	1541
9.18.5.44 SLQSSwiGetHDRPersonality(HDRPersonalityResp *pHDRPersonalityResp)	1541
9.18.5.45 SLQSSwiGetHDRProtSubtype(HDRProtSubtypResp *pHDRProtSubtypResp)	1542
9.18.5.46 SLQSSwiGetHRPDStats(GetHRPDStatsResp *pGetHRPDStatsResp)	1542
9.18.5.47 SLQSSwiGetLteCQI(LteCQIParm *pLteCQIResp)	1543
9.18.5.48 SLQSSwiGetLteSccRxInfo(LteSccRxInfoResp *pLteSccRxInfoResp)	1543
9.18.5.49 SLQSSwiNetworkDebug(NetworkDebugResp *pNetworkDebugResp)	1544
9.18.5.50 SLQSSwiPSDetach(PSDetachReq *pPSDetachReq)	1544
9.19 qaGobiApiOadm.h File Reference	1545
9.19.1 Detailed Description	1545
9.19.2 Function Documentation	1545
9.19.2.1 OMADMCancelSession()	1545
9.19.2.2 OMADMGetPendingNIA(ULONG *pSessionType, USHORT *pSessionID)	1546
9.19.2.3 OMADMGetSessionInfo(ULONG *pSessionState, ULONG *pSessionType, ULONG *pFailureReason, BYTE *pRetryCount, WORD *pSessionPause, WORD *pTimeRemaining)	1546

9.19.2.4	OMADMStartSession(ULONG sessionType)	1548
9.20	qaGobiApiPds.h File Reference	1548
9.20.1	Detailed Description	1549
9.20.2	Macro Definition Documentation	1549
9.20.2.1	DEFAULTBYTEVALUE	1549
9.20.2.2	DEFAULTLONGVALUE	1549
9.20.2.3	DEFAULTWORDVALUE	1549
9.20.3	Enumeration Type Documentation	1549
9.20.3.1	anonymous enum	1549
9.20.4	Function Documentation	1550
9.20.4.1	ForceXTRADownload()	1550
9.20.4.2	GetPDSDetails(ULONG *pOperation, BYTE *pTimeout, ULONG *pInterval, ULONG *pAccuracy)	1550
9.20.4.3	GetPDSSState(ULONG *pEnabledStatus, ULONG *pTrackingStatus)	1551
9.20.4.4	GetPortAutomaticTracking(ULONG *pbAuto)	1551
9.20.4.5	GetServiceAutomaticTracking(ULONG *pbAuto)	1552
9.20.4.6	GetXTRAAutomaticDownload(ULONG *pbEnabled, USHORT *pInterval)	1552
9.20.4.7	GetXTRANetwork(ULONG *pPreference)	1553
9.20.4.8	GetXTRAValidity(USHORT *pGPSWeek, USHORT *pGPSWeekOffset, USHORT *pDuration)	1554
9.20.4.9	PDSInjectTimeReference(ULONGLONG systemTime, USHORT systemDiscontinuities)	1554
9.20.4.10	ResetPDSDData(ULONG *pGPSDataMask, ULONG *pCellDataMask)	1555
9.20.4.11	SetPDSDetails(ULONG operation, BYTE timeout, ULONG interval, ULONG accuracy)	1556
9.20.4.12	SetPDSSState(ULONG enable)	1556
9.20.4.13	SetPortAutomaticTracking(ULONG bAuto)	1557
9.20.4.14	SetServiceAutomaticTracking(ULONG bAuto)	1557
9.20.4.15	SetXTRAAutomaticDownload(ULONG bEnabled, USHORT interval)	1558
9.20.4.16	SetXTRANetwork(ULONG preference)	1558
9.20.4.17	SLQSGetAGPSConfig(ULONG *pServerAddress, ULONG *pServerPort, BYTE *pServerURL, BYTE *pServerURLLength, BYTE *pNetworkMode)	1559
9.20.4.18	SLQSGetGPSSStateInfo(GPSSStateInfo *pGPSSStateInfo)	1560

9.20.4.19	SLQSPDSDeterminePosition()	1560
9.20.4.20	SLQSPDSInjectAbsoluteTimeReference(ULONGLONG timeMsec, ULONG↵ G timeUncMsec, BYTE timeBase, BYTE forceFlag)	1561
9.20.4.21	SLQSPDSInjectPositionData(struct PDSPositionData *pPositionData)	1561
9.20.4.22	SLQSSetAGPSConfig(ULONG *pServerAddress, ULONG *pServerPort, BYTE *pServerURL, BYTE *pServerURLLength, BYTE *pNetworkMode)	1562
9.20.4.23	SLQSSetPositionMethodState(PDSPosMethodStateReq *pDSPosMethod↵ StateReq)	1562
9.20.4.24	StartPDSTrackingSessionExt(BYTE sessionControl, BYTE sessionType, BYT↵ E sessionOperation, BYTE sessionServerOption, BYTE fixTimeout, ULONG fix↵ Interval, ULONG fixCount, ULONG fixAccuracy)	1563
9.20.4.25	StopPDSTrackingSession()	1564
9.21	qaGobiApiQos.h File Reference	1564
9.21.1	Detailed Description	1565
9.21.2	Macro Definition Documentation	1565
9.21.2.1	MAX_QOS_FILTER_TLV	1565
9.21.2.2	MAX_QOS_SPEC_PER_APN	1565
9.21.3	Function Documentation	1565
9.21.3.1	SLQSQosGetFlowStatus(BYTE instance, ULONG id, BYTE *pStatus)	1565
9.21.3.2	SLQSQosGetGranted(BYTE instance, ULONG id, swiQosGranted *pGranted)	1566
9.21.3.3	SLQSQosGetNetworkStatus(BYTE instance, BYTE *pStatus)	1566
9.21.3.4	SLQSQosGetNWProf(BYTE instance, BYTE *pSz, NWProfile *pProfile)	1567
9.21.3.5	SLQSQosModify(BYTE instance, swiQosModifyReq *pReq)	1567
9.21.3.6	SLQSQosRel(BYTE instance, swiQosIds *pQosIds)	1568
9.21.3.7	SLQSQosReq(BYTE instance, swiQosReq *pQosReq, swiQosIds *pQosResp)	1568
9.21.3.8	SLQSQosReset(BYTE instance)	1569
9.21.3.9	SLQSQosResume(BYTE instance, swiQosIds *pQosIds)	1569
9.21.3.10	SLQSQosSuspend(BYTE instance, swiQosIds *pQosIds)	1570
9.21.3.11	SLQSQosSwiReadApnExtraParams(BYTE instance, ULONG apnId, sApn↵ ExtraParams *pApnExtraParams)	1570
9.21.3.12	SLQSQosSwiReadDataStats(BYTE instance, ULONG apnId, sQosStat *pQosStat)	1571
9.22	qaGobiApiRms.h File Reference	1571
9.22.1	Detailed Description	1571

9.22.2	Function Documentation	1571
9.22.2.1	GetSMSWake(ULONG *pEnabled, ULONG *pWakeMask)	1571
9.22.2.2	SetSMSWake(ULONG bEnable, ULONG wakeMask)	1572
9.23	qaGobiApiSar.h File Reference	1573
9.23.1	Detailed Description	1573
9.23.2	Enumeration Type Documentation	1573
9.23.2.1	eQMISARRFState	1573
9.23.3	Function Documentation	1574
9.23.3.1	SLQSGetRfSarState(ULONG *pSarRFState)	1574
9.23.3.2	SLQSSetRfSarState(ULONG RfSarState)	1574
9.24	qaGobiApiSms.h File Reference	1575
9.24.1	Detailed Description	1577
9.24.2	Macro Definition Documentation	1577
9.24.2.1	ABSOLUTE_VALIDITY	1577
9.24.2.2	CONFIG_LEN	1577
9.24.2.3	MAX_SMS_ROUTES	1577
9.24.2.4	NUM_OF_SET	1577
9.24.2.5	TIME_DATE_BUF	1577
9.24.2.6	TIME_STAMP_BUF	1577
9.24.3	Typedef Documentation	1577
9.24.3.1	getIndicationRegResp	1577
9.24.3.2	getTransLayerInfoResp	1578
9.24.3.3	getTransNWRegInfoResp	1578
9.24.3.4	qaQmi3GPP2BroadcastCfgInfo	1579
9.24.3.5	qaQmi3GPPBroadcastCfgInfo	1579
9.24.3.6	setIndicationRegReq	1580
9.24.3.7	transLayerInfo	1580
9.24.4	Function Documentation	1581
9.24.4.1	GetSMSCAddress(BYTE addressSize, CHAR *pSMSCAddress, BYTE typeSize, CHAR *pSMSCType)	1581

9.24.4.2	SaveSMS(ULONG storageType, ULONG messageFormat, ULONG messageSize, BYTE *pMessage, ULONG *pMessageIndex)	1582
9.24.4.3	SendSMS(ULONG messageFormat, ULONG messageSize, BYTE *pMessage, ULONG *pMessageFailureCode, BYTE *pSmsOnIms)	1582
9.24.4.4	SetSMSCAddress(CHAR *pSMSCAddress, CHAR *pSMSCType)	1583
9.24.4.5	SLQSCDMADecodeMTTextMsg(struct cdmaMsgDecodingParams *pCdmaMsgDecodingParams)	1584
9.24.4.6	SLQSCDMAEncodeMOTextMsg(struct cdmaMsgEncodingParams *pCdmaMsgEncodingParams)	1584
9.24.4.7	SLQSDeleteSMS(ULONG storageType, ULONG *pMessageIndex, ULONG *pMessageTag, BYTE *pMessageMode)	1585
9.24.4.8	SLQSGetIndicationRegister(getIndicationRegResp *pGetIndicationRegInfo)	1586
9.24.4.9	SLQSGetMessageWaiting(getMsgWaitingInfo *pGetMsgWaitingInfoResp)	1586
9.24.4.10	SLQSGetSMS(ULONG storageType, ULONG messageIndex, ULONG *pMessageTag, ULONG *pMessageFormat, ULONG *pMessageSize, BYTE *pMessage, BYTE *pMessageMode)	1587
9.24.4.11	SLQSGetSmsBroadcastConfig(BYTE mode, qaQmi3GPPBroadcastCfgInfo *pBroadcastConfig, qaQmi3GPP2BroadcastCfgInfo *pCDMABroadcastConfig)	1588
9.24.4.12	SLQSGetSMSList(ULONG storageType, ULONG *pRequestedTag, ULONG *pMessageListSize, BYTE *pMessageList, BYTE *pMessageMode)	1588
9.24.4.13	SLQSGetTransLayerInfo(getTransLayerInfoResp *pGetTransLayerInfoResp)	1589
9.24.4.14	SLQSGetTransNWRegInfo(getTransNWRegInfoResp *pGetTransNWRegInfoResp)	1590
9.24.4.15	SLQSModifySMSStatus(ULONG storageType, ULONG messageIndex, ULONG messageTag, BYTE *pMessageMode)	1590
9.24.4.16	SLQSSendAsyncSMS(slqssendasyncsmsparams_s *pSendSmsParams)	1591
9.24.4.17	SLQSSendLongSMS(ULONG messageFormat, ULONG messageSize, CHAR *pMessage, BYTE encodingScheme, ULONG *pMessageFailureCode, CHAR *pMobileNum, BYTE *pSmsOnIMS)	1592
9.24.4.18	SLQSSendSMS(slqssendsmsparams_s *pSendSmsParams)	1593
9.24.4.19	SLQSSetIndicationRegister(setIndicationRegReq *pSetIndicationRegReq)	1593
9.24.4.20	SLQSSetSmsBroadcastActivation(BYTE mode, BYTE broadcastActivate)	1594
9.24.4.21	SLQSSetSmsBroadcastConfig(BYTE mode, qaQmi3GPPBroadcastCfgInfo *pBroadcastConfig, qaQmi3GPP2BroadcastCfgInfo *pCDMABroadcastConfig)	1594
9.24.4.22	SLQSSetSmsStorage(BYTE smsStorage)	1595
9.24.4.23	SLQSSmsGetMaxStorageSize(smsMaxStorageSizeReq *pMaxStorageSizeReq, smsMaxStorageSizeResp *pMaxStorageSizeResp)	1595
9.24.4.24	SLQSSmsGetMessageProtocol(smsMsgprotocolResp *pMessageProtocol)	1596

9.24.4.25	SLQSSmsSetRoutes(smsSetRoutesReq *pSetRoutesReq)	1596
9.24.4.26	SLQSSwiGetSMSStorage(ULONG *pSmsStorage)	1597
9.24.4.27	SLQSWCDMADecodeLongTextMsg(struct wcdmaLongMsgDecodingParams *pWcdmaLongMsgDecodingParams)	1597
9.24.4.28	SLQSWCDMADecodeMTTextMsg(struct wcdmaMsgDecodingParams *pWcdmaMsgDecodingParams)	1598
9.24.4.29	SLQSWCDMAEncodeMOTextMsg(struct wcdmaMsgEncodingParams *pWcdmaMsgEncodingParams)	1598
9.25	qaGobiApiSwi.h File Reference	1599
9.25.1	Detailed Description	1599
9.25.2	Function Documentation	1599
9.25.2.1	SLQSGetPidof(CHAR *pProcName)	1599
9.25.2.2	SLQSGetSdkVersion(CHAR **sdkversionpp)	1600
9.25.2.3	SLQSSendRawQMI(BYTE *pReqBuf, USHORT service, USHORT length, ULONG timeout, BYTE **ppInParm, USHORT *pParamLength)	1600
9.26	qaGobiApiSwiAudio.h File Reference	1600
9.26.1	Detailed Description	1601
9.26.2	Macro Definition Documentation	1601
9.26.2.1	MAX_LEN_IFACE_TABLE	1601
9.26.3	Function Documentation	1601
9.26.3.1	SLQSGetM2MAudioProfile(GetM2MAudioProfileReq *pGetM2MAudioProfileReq, GetM2MAudioProfileResp *pGetM2MAudioProfileResp)	1601
9.26.3.2	SLQSGetM2MAudioVolume(GetM2MAudioVolumeReq *pGetM2MAudioVolumeReq, GetM2MAudioVolumeResp *pGetM2MAudioVolumeResp)	1602
9.26.3.3	SLQSGetM2MAVMute(GetM2MAVMuteReq *pGetM2MAVMuteReq, GetM2MAVMuteResp *pGetM2MAVMuteResp)	1602
9.26.3.4	SLQSGetM2MSpkrGain(GetM2MSpkrGainReq *pSpkrGainReq, GetM2MSpkrGainResp *pSpkrGainResp)	1603
9.26.3.5	SLQSSetM2MAudioAVCFG(SetM2MAudioAVCFGReq *pSetM2MAudioAVCFGReq)	1603
9.26.3.6	SLQSSetM2MAudioLPBK(SetM2MAudioLPBKReq *pSetM2MAudioLPBKReq)	1604
9.26.3.7	SLQSSetM2MAudioNVDef()	1604
9.26.3.8	SLQSSetM2MAudioProfile(SetM2MAudioProfileReq *pSetM2MAudioProfileReq)	1605
9.26.3.9	SLQSSetM2MAudioVolume(SetM2MAudioVolumeReq *pSetM2MAudioVolumeReq)	1605

9.26.3.10 SLQSSetM2MAVMute(SetM2MAVMuteReq *pSetM2MAVMuteReq)	1606
9.26.3.11 SLQSSetM2MSpkrGain(SetM2MSpkrGainReq *pSpkrGainReq)	1606
9.27 qaGobiApiSwiOmadms.h File Reference	1606
9.27.1 Detailed Description	1607
9.27.2 Typedef Documentation	1607
9.27.2.1 SLQSOMADMSessionInfo	1607
9.27.2.2 SLQSOMADMSettings	1609
9.27.2.3 SLQSOMADMSettingsReqParams	1610
9.27.2.4 SLQSOMADMSettingsReqParams3	1611
9.27.3 Function Documentation	1612
9.27.3.1 SLQSOMADMCancelSession(ULONG session)	1612
9.27.3.2 SLQSOMADMGetSessionInfo(ULONG *pSessionType, SLQSOMADM↔ SessionInfo *pResp)	1612
9.27.3.3 SLQSOMADMGetSettings(ULONG *pbOMADMEEnabled, ULONG *pbFOT↔ Adownload, ULONG *pbFOTAUpdate)	1613
9.27.3.4 SLQSOMADMGetSettings2(SLQSOMADMSettings *pSLQSOMADMSettings) .	1614
9.27.3.5 SLQSOMADMSendSelection(ULONG selection)	1614
9.27.3.6 SLQSOMADMSendSelection2(ULONG selection, ULONG *pDeferTime, ULO↔ NG *pRejectReason)	1615
9.27.3.7 SLQSOMADMSetSettings(ULONG bFOTAdownload, ULONG bFOTAUpdate) .	1615
9.27.3.8 SLQSOMADMSetSettings2(SLQSOMADMSettingsReqParams *pSLQSOMA↔ DMSettingsReqParams)	1616
9.27.3.9 SLQSOMADMSetSettings3(SLQSOMADMSettingsReqParams3 *pSLQSOMA↔ DMSettingsReqParams3)	1616
9.27.3.10 SLQSOMADMStartSession(ULONG sessionType)	1616
9.27.3.11 SLQSOMADMStartSession2(ULONG sessionType, ULONG *pFwAvailability) .	1617
9.28 qaGobiApiTableBandClasses.h File Reference	1618
9.28.1 Detailed Description	1618
9.28.2 Band Classes (Value - Description)	1618
9.28.2.1 LTE Bands	1619
9.29 qaGobiApiTableCallControlReturnReasons.h File Reference	1621
9.29.1 Detailed Description	1621
9.29.2 Coding Group Bits 7..4(0000)	1621

9.30	qaGobiApiTableCallEndReasons.h File Reference	1621
9.30.1	Detailed Description	1622
9.30.2	Call end reason codes (Code - Reason)	1622
9.30.2.1	Technology-agnostic call end reasons	1622
9.30.2.2	CDMA	1622
9.30.2.3	WCDMA/GSM call end reasons	1623
9.30.2.4	CDMA	1625
9.30.2.5	call end reason type	1625
9.30.2.6	Mobile IP call end reasons (Type=1)	1626
9.30.2.7	Internal call end reasons (Type=2)	1627
9.30.2.8	Call Manager defined call end reasons (Type=3)	1629
9.30.2.9	3GPP specification defined call end reasons (Type=6)	1634
9.30.2.10	PPP call end reasons (Type=7)	1636
9.30.2.11	EHRPD call end reasons (Type=8)	1636
9.30.2.12	IPV6 call end reasons (Type=9)	1637
9.31	qaGobiApiTableCarrierCodes.h File Reference	1637
9.31.1	Detailed Description	1637
9.31.2	Carrier Codes (Number - Carrier)	1638
9.32	qaGobiApiTableCodingScheme.h File Reference	1639
9.32.1	Detailed Description	1639
9.32.2	Coding Group Bits 7..4(0000)	1640
9.32.2.1	Use of bits 3..0\n\n	1640
9.32.3	Coding Group Bits 7..4(0001)	1640
9.32.3.1	use of bits 3..0	1640
9.32.4	Coding Group Bits 7..4(0010)	1640
9.32.4.1	use of bits 3..0	1640
9.32.5	Coding Group Bits 7..4(0011)	1641
9.32.5.1	use of bits 3..0	1641
9.32.6	Coding Group Bits 7..4(01xx)	1641
9.32.6.1	use of bits 3..0	1641

9.32.7 Coding Group Bits 7..4(1001)	1641
9.32.7.1 Reserved coding groups	1641
9.32.8 Coding Group Bits 7..4(1010..1101)	1642
9.32.8.1 Reserved coding groups	1642
9.32.9 Coding Group Bits 7..4(1110)	1642
9.32.9.1 Defined by the WAP Forum	1642
9.32.10 Coding Group Bits 7..4 (1111)	1642
9.32.10.1 Data coding / message handling	1642
9.32.11 Macro Definition Documentation	1642
9.32.11.1 __GOBI_API_CODING_SCHEME_H__	1642
9.33 qaGobiApiTableGpsCapabilityCodes.h File Reference	1642
9.33.1 Detailed Description	1642
9.33.2 GPS capability (Value - Capability)	1642
9.34 qaGobiApiTablePowerModes.h File Reference	1643
9.34.1 Detailed Description	1643
9.34.2 Power Modes (Value - Description)	1643
9.35 qaGobiApiTableRadioInterfaces.h File Reference	1643
9.35.1 Detailed Description	1643
9.35.2 Radio interface	1644
9.35.2.1 Technology (Value - Radio Interface Technology)	1644
9.36 qaGobiApiTableRegionCodes.h File Reference	1644
9.36.1 Detailed Description	1644
9.36.2 Region Codes (Code - Region)	1644
9.37 qaGobiApiTableServiceOptions.h File Reference	1644
9.37.1 Detailed Description	1645
9.37.2 Service Option codes (Code - Reason)	1645
9.37.2.1 Description	1645
9.38 qaGobiApiTableSupServiceInfoClasses.h File Reference	1647
9.38.1 Detailed Description	1647
9.38.2 Supplementary Service Information Classes (Value - Service Class)	1647

9.39	qaGobiApiTableSwiAudio.h File Reference	1647
9.39.1	Detailed Description	1647
9.39.2	ACDB Device (Device ID - description)	1648
9.39.3	Physical Interface (Device ID - description - Interface parameters)	1648
9.40	qaGobiApiTableSwiOMADMUpdateCompleteStatus.h File Reference	1648
9.40.1	Detailed Description	1648
9.40.2	OMA DM Update Complete Status (Update Complete Status - Meaning - Usage)	1648
9.41	qaGobiApiTableVoiceCallEndReasons.h File Reference	1649
9.41.1	Detailed Description	1650
9.41.2	Voice Call and supplementary services end reason codes (Code - Reason)	1650
9.41.2.1	General	1650
9.41.2.2	service Errors	1651
9.41.2.3	control cause values	1652
9.41.2.4	reject causes	1654
9.41.2.5	reject causes	1655
9.41.2.6	reject causes	1655
9.41.2.7	stratum reject causes	1655
9.41.2.8	reject causes	1655
9.41.2.9	IP end reasons	1655
9.42	qaGobiApiTmd.h File Reference	1656
9.42.1	Detailed Description	1656
9.42.2	Macro Definition Documentation	1657
9.42.2.1	MAX_MITIGATION_DEV_ID_LEN	1657
9.42.2.2	MAX_MITIGATION_DEV_LIST_LEN	1657
9.42.3	Function Documentation	1657
9.42.3.1	SLQSTmdDeRegNotMitigationLvl(TmdDeRegNotMitigationLvlReq *pTmdDeRegNotMitigationLvlReq)	1657
9.42.3.2	SLQSTmdGetMitigationDevList(TmdGetMitigationDevListResp *pTmdGetMitigationDevListResp)	1657
9.42.3.3	SLQSTmdGetMitigationLvl(TmdGetMitigationLvlReq *pTmdGetMitigationLvlReq, TmdGetMitigationLvlResp *pTmdGetMitigationLvlResp)	1658

9.42.3.4	SLQSTmdRegNotMitigationLvl(TmdRegNotMitigationLvlReq *pTmdRegNotMitigationLvlReq)	1658
9.43	qaGobiApiUim.h File Reference	1658
9.43.1	Detailed Description	1660
9.43.2	Macro Definition Documentation	1661
9.43.2.1	MAX_ACTIVE_PERS_FEATURES	1661
9.43.2.2	MAX_CONTENT_LENGTH	1661
9.43.2.3	MAX_DESCRIPTION_LENGTH	1661
9.43.2.4	MAX_ICCID_LENGTH	1661
9.43.2.5	MAX_NO_OF_APPLICATIONS	1661
9.43.2.6	MAX_NO_OF_SLOTS	1661
9.43.2.7	MAX_PATH_LENGTH	1661
9.43.2.8	MAX_PUK_LENGTH	1661
9.43.2.9	MAX_SLOTS_STATUS	1661
9.43.3	Function Documentation	1661
9.43.3.1	SLQSUIMAuthenticate(UIMAuthenticateReq *pUIMAuthenticateReq, UIMAuthenticateResp *pUIMAuthenticateResp)	1661
9.43.3.2	SLQSUIMChangePin(UIMChangePinReq *pUIMChangePinReq, UIMPinResp *pUIMChangePinResp)	1662
9.43.3.3	SLQSUIMDepersonalization(UIMDepersonalizationReq *pUIMDepersonalizationReq, UIMDepersonalizationResp *pUIMDepersonalizationResp)	1662
9.43.3.4	SLQSUIEventRegister(UIMEventRegisterReqResp *pUIMEventRegisterReqResp)	1663
9.43.3.5	SLQSUIMGetCardStatus(UIMGetCardStatusResp *pUIMGetCardStatusResp)	1663
9.43.3.6	SLQSUIMGetConfiguration(UIMGetConfigurationReq *pUIMGetConfigurationReq, UIMGetConfigurationResp *pUIMGetConfigurationResp)	1664
9.43.3.7	SLQSUIMGetFileAttributes(UIMGetFileAttributesReq *pUIMGetFileAttributesReq, UIMGetFileAttributesResp *pUIMGetFileAttributesResp)	1664
9.43.3.8	SLQSUIMGetSlotsStatus(UIMGetSlotsStatusResp *pResp)	1665
9.43.3.9	SLQSUIPowerDown(UIMPowerDownReq *pUIMPowerDownReq)	1666
9.43.3.10	SLQSUIPowerUp(UIMPowerUpReq *pUIMPowerUpReq)	1666
9.43.3.11	SLQSUIReadTransparent(UIMReadTransparentReq *pUIMReadTransparentReq, UIMReadTransparentResp *pUIMReadTransparentResp)	1667

9.43.3.12 SLQSUIMRefreshComplete(UIMRefreshCompleteReq *pUIMRefreshCompleteReq)	1667
9.43.3.13 SLQSUIMRefreshGetLastEvent(UIMRefreshGetLastEventReq *pUIMRefreshGetLastEventReq, UIMRefreshGetLastEventResp *pUIMRefreshGetLastEventResp)	1668
9.43.3.14 SLQSUIMRefreshOK(UIMRefreshOKReq *pUIMRefreshOKReq)	1668
9.43.3.15 SLQSUIMRefreshRegister(UIMRefreshRegisterReq *pUIMRefreshRegisterReq)	1669
9.43.3.16 SLQSUIMReset()	1669
9.43.3.17 SLQSUIMSetPinProtection(UIMSetPinProtectionReq *pUIMSetPinProtectionReq, UIMPinResp *pUIMSetPinProtectionResp)	1670
9.43.3.18 SLQSUIMSwitchSlot(UIMSwitchSlotReq *pReq)	1671
9.43.3.19 SLQSUIMUnblockPin(UIMUnblockPinReq *pUIMUnblockPinReq, UIMPinResp *pUIMUnblockPinResp)	1671
9.43.3.20 SLQSUIMVerifyPin(UIMVerifyPinReq *pUIMVerifyPinReq, UIMPinResp *pUIMVerifyPinResp)	1672
9.44 qaGobiApiVoice.h File Reference	1672
9.44.1 Detailed Description	1675
9.44.2 Macro Definition Documentation	1676
9.44.2.1 MAX_CALL_NO_LEN	1676
9.44.2.2 MAX_DESCRIPTION_LENGTH	1676
9.44.2.3 MAX_NO_OF_CALLS	1676
9.44.2.4 MAXUSSDLENGTH	1676
9.44.2.5 PASSWORD_LENGTH	1676
9.44.3 Enumeration Type Documentation	1676
9.44.3.1 serviceClassInformation	1676
9.44.4 Function Documentation	1676
9.44.4.1 AnswerUSSD(BYTE *pInfo)	1676
9.44.4.2 CancelUSSD()	1677
9.44.4.3 OriginateUSSD(BYTE *pInfo)	1677
9.44.4.4 SLQSOriinateUSSD(struct USSInfo *pReq, struct USSResp *pResp)	1678
9.44.4.5 SLQSVoiceALSSelectLine(voiceALSSelectLineInfo *pVoiceALSSelectLineInfo)	1678
9.44.4.6 SLQSVoiceALSSetLineSwitching(voiceALSSetLineSwitchInfo *pVoiceALSSetLineSwitchInfo)	1679

9.44.4.7	SLQSVoiceAnswerCall(voiceAnswerCall *pVoiceAnswerCall)	1679
9.44.4.8	SLQSVoiceBindSubscription(voiceBindSubscriptionInfo *pVoiceBindSubscriptionInfo)	1680
9.44.4.9	SLQSVoiceBurstDTMF(voiceBurstDTMFInfo *pBurstDTMFInfo)	1680
9.44.4.10	SLQSVoiceDialCall(voiceCallRequestParams *pCallRequestParams, voiceCallResponseParams *pCallResponseParams)	1681
9.44.4.11	SLQSVoiceEndCall(BYTE *pCallId)	1682
9.44.4.12	SLQSVoiceGetAllCallInfo(voiceGetAllCallInfo *pGetAllCallInfo)	1682
9.44.4.13	SLQSVoiceGetCallBarring(voiceGetCallBarringReq *pVoiceGetCallBarringReq, voiceGetCallBarringResp *pVoiceGetCallBarringResp)	1683
9.44.4.14	SLQSVoiceGetCallForwardingStatus(voiceGetCallFWReq *pVoiceGetCallFWReq, voiceGetCallFWResp *pVoiceGetCallFWResp)	1683
9.44.4.15	SLQSVoiceGetCallInfo(voiceCallInfoReq *pGetCallInfoReq, voiceCallInfoResp *pGetCallInfoResp)	1684
9.44.4.16	SLQSVoiceGetCallWaiting(voiceGetCallWaitInfo *pVoiceGetCallWaitInfo)	1684
9.44.4.17	SLQSVoiceGetCLIP(voiceGetCLIPResp *pVoiceGetCLIPResp)	1685
9.44.4.18	SLQSVoiceGetCLIR(voiceGetCLIRResp *pVoiceGetCLIRResp)	1686
9.44.4.19	SLQSVoiceGetCNAP(voiceGetCNAPResp *pVoiceGetCNAPResp)	1686
9.44.4.20	SLQSVoiceGetCOLP(voiceGetCOLPResp *pVoiceGetCOLPResp)	1687
9.44.4.21	SLQSVoiceGetCOLR(voiceGetCOLRResp *pVoiceGetCOLRResp)	1687
9.44.4.22	SLQSVoiceGetConfig(voiceGetConfigReq *pVoiceGetConfigReq, voiceGetConfigResp *pVoiceGetConfigResp)	1688
9.44.4.23	SLQSVoiceIndicationRegister(voiceIndicationRegisterInfo *pVoiceIndicationRegisterInfo)	1689
9.44.4.24	SLQSVoiceManageCalls(voiceManageCallsReq *pVoiceManageCallsReq, voiceManageCallsResp *pVoiceManageCallsResp)	1689
9.44.4.25	SLQSVoiceOrigUSSDNoWait(voiceOrigUSSDNoWaitInfo *pVoiceOrigUSSDNoWaitInfo)	1690
9.44.4.26	SLQSVoiceSendFlash(voiceFlashInfo *pFlashInfo)	1690
9.44.4.27	SLQSVoiceSetCallBarringPassword(voiceSetCallBarringPwdInfo *pVoiceSetCallBarringPwdInfo, voiceSetCallBarringPwdResp *pSetCallBarringPwdResp)	1691
9.44.4.28	SLQSVoiceSetConfig(voiceSetConfigReq *pVoiceSetConfigReq, voiceSetConfigResp *pVoiceSetConfigResp)	1692
9.44.4.29	SLQSVoiceSetPreferredPrivacy(voiceSetPrefPrivacy *pSetPrefPrivacy)	1692
9.44.4.30	SLQSVoiceSetSUPSService(voiceSetSUPSServiceReq *pVoiceSetSUPSServiceReq, voiceSetSUPSServiceResp *pVoiceSetSUPSServiceResp)	1693

9.44.4.31	SLQSVoiceStartContDTMF(voiceContDTMFinfo *pContDTMFinfo)	1693
9.44.4.32	SLQSVoiceStopContDTMF(voiceStopContDTMFinfo *pVoiceStopContDTMFinfo)	1694
9.45	qaGobiApiWds.h File Reference	1694
9.45.1	Detailed Description	1698
9.45.2	Macro Definition Documentation	1698
9.45.2.1	IPV6_ADDRESS_ARRAY_SIZE	1698
9.45.3	Typedef Documentation	1698
9.45.3.1	GetProfileSettingIn	1698
9.45.3.2	GetProfileSettingOut	1699
9.45.3.3	QmiProfileInfo	1699
9.45.3.4	QmiWDSDataBearers	1699
9.45.3.5	QmiWDSDataBearerTechnology	1699
9.45.3.6	slqs3GPPConfigItem	1700
9.45.4	Enumeration Type Documentation	1701
9.45.4.1	qmiDataBearerMasks	1702
9.45.5	Function Documentation	1702
9.45.5.1	GetAutoconnect(ULONG *pSetting)	1702
9.45.5.2	GetByteTotals(ULONGLONG *pTXTotalBytes, ULONGLONG *pRXTotalBytes, BYTE instance)	1702
9.45.5.3	GetConnectionRate(ULONG *pCurrentChannelTXRate, ULONG *pCurrentChannelRXRate, ULONG *pMaxChannelTXRate, ULONG *pMaxChannelRXRate, BYTE instance)	1703
9.45.5.4	GetDataBearerTechnology(ULONG *pDataBearer, BYTE instance)	1704
9.45.5.5	GetDefaultProfile(ULONG profileType, ULONG *pPDPTType, ULONG *pIPAddress, ULONG *pPrimaryDNS, ULONG *pSecondaryDNS, ULONG *pAuthentication, BYTE nameSize, CHAR *pName, BYTE apnSize, CHAR *pAPNName, BYTE userSize, CHAR *pUsername)	1705
9.45.5.6	GetDefaultProfileLTE(ULONG profileType, ULONG *pPDPTType, ULONG *pIPAddressv4, ULONG *pPrimaryDNSv4, ULONG *pSecondaryDNSv4, USHORT *pIPAddressv6, USHORT *pPrimaryDNSv6, USHORT *pSecondaryDNSv6, ULONG *pAuthentication, BYTE nameSize, CHAR *pName, BYTE apnSize, CHAR *pAPNName, BYTE userSize, CHAR *pUsername)	1706
9.45.5.7	GetDefaultProfileNum(BYTE profile_type, BYTE profile_family, BYTE *pProfile_no)	1708
9.45.5.8	GetDormancyState(ULONG *pDormancyState, BYTE instance)	1708

9.45.5.9	GetIPAddressLTE(WdsIpAddressInfoReq *)	1709
9.45.5.10	GetLastMobileIPError(ULONG *pError)	1709
9.45.5.11	GetMobileIP(ULONG *pMode)	1710
9.45.5.12	GetMobileIPProfile(BYTE index, BYTE *pEnabled, ULONG *pAddress, ULONG *pPrimaryHA, ULONG *pSecondaryHA, BYTE *pRevTunneling, BYTE naiSize, CHAR *pNAI, ULONG *pHASPI, ULONG *pAAASPI, ULONG *pHASState, ULONG *pAAASState)	1711
9.45.5.13	GetPacketStatistics(struct WdsPktStatisticsReq *pStatMask, struct WdsPktStatisticsElmnts *pPktStatisticsElmnt, BYTE instance)	1712
9.45.5.14	GetPacketStatus(ULONG *pTXPacketSuccesses, ULONG *pRXPacketSuccesses, ULONG *pTXPacketErrors, ULONG *pRXPacketErrors, ULONG *pTXPacketOverflows, ULONG *pRXPacketOverflows, BYTE instance)	1713
9.45.5.15	GetSessionDuration(ULONGLONG *pDuration, BYTE instance)	1713
9.45.5.16	GetSessionState(ULONG *pState, BYTE instance)	1714
9.45.5.17	iGetByteTotals(ULONG *pv4sessionId, ULONG *pv6sessionId, struct WdsByteTotalsElmnts *pByteTotalsElmnt)	1714
9.45.5.18	iGetConnectionRate(ULONG *pv4sessionId, ULONG *pv6sessionId, struct WdsConnectionRateElmnts *pConnectionRateElmnt)	1714
9.45.5.19	iGetPacketStatistics(ULONG *pV4sessionId, ULONG *pV6sessionId, struct WdsPktStatisticsReq *pStatMask, struct WdsPktStatisticsElmnts *pPktStatisticsElmnt)	1715
9.45.5.20	iLQSMISetIPFamilyPreference(BYTE IPFamilyPreference, BYTE instance)	1715
9.45.5.21	RMSetTransferStatistics(swiRMTrasferStaticsReq *pSwiRMTrasferStaticsReq)	1715
9.45.5.22	SetActiveMobileIPProfile(CHAR *pSPC, BYTE index)	1715
9.45.5.23	SetAutoconnect(ULONG setting)	1716
9.45.5.24	SetDefaultProfile(ULONG profileType, ULONG *pPDPTType, ULONG *pIPAddress, ULONG *pPrimaryDNS, ULONG *pSecondaryDNS, ULONG *pAuthentication, CHAR *pName, CHAR *pAPNName, CHAR *pUsername, CHAR *pPassword)	1716
9.45.5.25	SetDefaultProfileLTE(ULONG profileType, ULONG *pPDPTType, ULONG *pIPAddressv4, ULONG *pPrimaryDNSv4, ULONG *pSecondaryDNSv4, USHORT *pIPAddressv6, USHORT *PrimaryDNSv6, USHORT *pSecondaryDNSv6, ULONG *pAuthentication, CHAR *pName, CHAR *pAPNName, CHAR *pUsername, CHAR *pPassword)	1717
9.45.5.26	SetDefaultProfileLTEV2(ULONG profileType, ULONG *pPDPTType, ULONG *pIPAddressv4, ULONG *pPrimaryDNSv4, ULONG *pSecondaryDNSv4, USHORT *pIPAddressv6, USHORT *PrimaryDNSv6, USHORT *pSecondaryDNSv6, ULONG *pAuthentication, CHAR *pName, CHAR *pAPNName, CHAR *pUsername, CHAR *pPassword)	1719
9.45.5.27	SetDefaultProfileNum(BYTE profile_type, BYTE profile_family, BYTE profile_index)	1720

9.45.5.28 SetMobileIP(ULONG mode)	1721
9.45.5.29 SetMobileIPParameters(CHAR *pSPC, ULONG *pMode, BYTE *pRetryLimit, BYTE *pRetryInterval, BYTE *pReRegPeriod, BYTE *pReRegTraffic, BYTE *p← HAAAuthenticator, BYTE *pHA2002bis)	1722
9.45.5.30 SetMobileIPProfile(CHAR *pSPC, BYTE index, BYTE *pEnabled, ULON← G *pAddress, ULONG *pPrimaryHA, ULONG *pSecondaryHA, BYTE *p← RevTunneling, CHAR *pNAI, ULONG *pHASPI, ULONG *pAAASPI, CHAR *pMNHA, CHAR *pMNAAB)	1723
9.45.5.31 SLQSAutoConnect(struct slqsaautoconnect *pacreq)	1724
9.45.5.32 SLQSCreateProfile(struct CreateProfileIn *pReq, struct CreateProfileOut *pResp)1724	
9.45.5.33 SLQSDeleteProfile(struct SLQSDeleteProfileParams *pProfileToDelete, WOR← D *pExtendedErrorCode)	1725
9.45.5.34 SLQSGet3GPPConfigItem(slqs3GPPConfigItem *pSLQS3GPPConfigItem) . . .	1725
9.45.5.35 SLQSGetByteTotals(struct WdsByteTotals *pByteTotals)	1726
9.45.5.36 SLQSGetConnectionRate(struct WdsConnectionRate *pConnectionRate) . . .	1726
9.45.5.37 SLQSGetCurrDataSystemStat(CurrDataSysStat *pCurrDataSysStat)	1727
9.45.5.38 SLQSGetCurrentChannelRate(WDSSWICurrentChannelRates *pRates, BYT← E instance)	1727
9.45.5.39 SLQSGetDataBearerTechnology(QmiWSDDataBearers *pDataBearers, BYT← E instance)	1728
9.45.5.40 SLQSGetDataBearerTechnologyExt(DataBearerTechExt *pDataBearerTech, B← YTE instance)	1728
9.45.5.41 SLQSGetDUNCallInfo(getDUNCallInfoReq *pGetDUNCallInfoReq, getDUN← CallInfoResp *pGetDUNCallInfoResp)	1729
9.45.5.42 SLQSGetPacketStatistics(struct WdsPktStatisticsReq *pStatMask, struct Wds← PktStatisticsResp *pPktStatistics)	1729
9.45.5.43 SLQSGetProfile(ULONG profileType, BYTE profileId, ULONG *pPDPTType, UL← ONG *pIPAddress, ULONG *pPrimaryDNS, ULONG *pSecondaryDNS, ULONG *pAuthentication, BYTE nameSize, CHAR *pName, BYTE apnSize, CHAR *p← APNName, BYTE userSize, CHAR *pUsername, WORD *pExtendedErrorCode) 1730	
9.45.5.44 SLQSGetProfileSettings(GetProfileSettingIn *pReq, GetProfileSettingOut *pResp)1731	
9.45.5.45 SLQSGetRuntimeSettings(struct WdsRunTimeSettings *pRunTimeSettings) . .	1732
9.45.5.46 SLQSGetSessionState(ULONG *pStateV4, ULONG *pStateV6, BYTE instance) 1732	
9.45.5.47 SLQSModifyProfile(struct ModifyProfileIn *pReq, struct ModifyProfileOut *pResp)1733	
9.45.5.48 SLQSResetPacketStatics()	1734
9.45.5.49 SLQSSet3GPPConfigItem(slqs3GPPConfigItem *pSLQS3GPPConfigItem) . . .	1734

9.45.5.50	SLQSSetProfile(ULONG profileType, BYTE profileId, ULONG *pPDPTType, UL↵ ONG *pIPAddress, ULONG *pPrimaryDNS, ULONG *pSecondaryDNS, ULONG *pAuthentication, CHAR *pName, CHAR *pAPNName, CHAR *pUsername, C↵ HAR *pPassword)	1734
9.45.5.51	SLQSSetDHCPv4ClientConfig(WdsDHCPv4Config *pReqResp)	1736
9.45.5.52	SLQSSetLoopback(WDSGetLoopbackData *data)	1736
9.45.5.53	SLQSSetDHCPv4ClientConfig(WdsDHCPv4Config *pReq)	1737
9.45.5.54	SLQSSetLoopback(WDSSetLoopbackData *pReq)	1737
9.45.5.55	SLQSSetDataSession(struct ssdatasession_params *pin)	1737
9.45.5.56	SLQSWdsGoActive(void)	1738
9.45.5.57	SLQSWdsGoDormant(void)	1738
9.45.5.58	SLQSWdsSetEventReport(wdsSetEventReportReq *pSetEventReportReq) . . .	1739
9.45.5.59	SLQSWdsSwiPDPRuntimeSettings(swiPDPRuntimeSettingsReq *pPDP↵ RuntimeSettingsReq, swiPDPRuntimeSettingsResp *pPDPRuntimeSettingsResp) .	1739
9.45.5.60	WDS_IsGobiDevice()	1740
9.46	qaNasGetRFBandInfo.h File Reference	1740
9.46.1	Enumeration Type Documentation	1740
9.46.1.1	eQMI_NAS_GET_RF_INFO_RESP	1740
9.46.2	Function Documentation	1741
9.46.2.1	PkQmiNasGetRFBandInfo(WORD *pMlength, BYTE *pBuffer)	1741
9.46.2.2	UpkQmiNasGetRFBandInfo(BYTE *pMdmResp, struct QmiNasGetRFBand↵ InfoResp *pApiResp)	1741
9.47	qaNasPerformNetworkScan.h File Reference	1741
9.47.1	Macro Definition Documentation	1741
9.47.1.1	FORBIDDEN_INDEX	1741
9.47.1.2	INDEX_ZERO	1741
9.47.1.3	MAX_DESCRIPTION_LENGTH	1741
9.47.1.4	PREFERRED_INDEX	1741
9.47.1.5	QMI_NAS_MAX_INSTANCES	1741
9.47.1.6	QMI_NAS_NETSTATUS_MASK	1741
9.47.1.7	ROAMING_INDEX	1741
9.47.2	Enumeration Type Documentation	1741
9.47.2.1	eQMI_NAS_PERFORM_NETWORK_SCAN_RESP	1741

9.47.3	Function Documentation	1742
9.47.3.1	PkQmiNasPerformNetworkScan(WORD *pMlength, BYTE *pParamField)	1742
9.47.3.2	UpkQmiNasPerformNetworkScan(BYTE *pMdmResp, struct QmiNasPerformNetworkScanResp *pAipResp)	1742
9.48	qmerrno.h File Reference	1742
9.48.1	Enumeration Type Documentation	1744
9.48.1.1	eQCWWANError	1744
9.48.1.2	qm_wds_ds_profile_extended_err_codes	1749
9.49	qos.h File Reference	1749
9.49.1	Macro Definition Documentation	1751
9.49.1.1	LIBPACK_MAX_QOS_FILTERS	1751
9.49.1.2	LIBPACK_MAX_QOS_FLOW_PER_APN_STATS	1751
9.49.1.3	LIBPACK_MAX_QOS_FLOWS	1751
9.49.2	Function Documentation	1751
9.49.2.1	pack_qos_SLQSQosGetNetworkStatus(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1751
9.49.2.2	pack_qos_SLQSQosSwiReadApnExtraParams(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_qos_SLQSQosSwiReadApnExtraParams_t reqParam)	1751
9.49.2.3	pack_qos_SLQSQosSwiReadDataStats(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_qos_SLQSQosSwiReadDataStats_t reqParam)	1752
9.49.2.4	pack_qos_SLQSSetQosEventCallback(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_qos_SLQSSetQosEventCallback_t reqParam)	1753
9.49.2.5	unpack_qos_SLQSQosGetNetworkStatus(uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSQosGetNetworkStatus_t *pOutput)	1754
9.49.2.6	unpack_qos_SLQSQosSwiReadApnExtraParams(uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSQosSwiReadApnExtraParams_t *pOutput)	1754
9.49.2.7	unpack_qos_SLQSQosSwiReadDataStats(uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSQosSwiReadDataStats_t *pOutput)	1755
9.49.2.8	unpack_qos_SLQSSetQosEventCallback(uint8_t *pResp, uint16_t respLen)	1755
9.49.2.9	unpack_qos_SLQSSetQosEventCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosEventCallback_ind_t *pOutput)	1755
9.49.2.10	unpack_qos_SLQSSetQosNWStatusCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosNWStatusCallback_ind_t *pOutput)	1756
9.49.2.11	unpack_qos_SLQSSetQosPriEventCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosPriEventCallback_ind_t *pOutput)	1757

9.49.2.12 unpack_qos_SLQSSetQosStatusCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosStatusCallback_ind_t *pOutput)	1757
9.50 sms.h File Reference	1758
9.50.1 Macro Definition Documentation	1760
9.50.1.1 MAX_CDMA_ENC_MO_TXT_MSG_SIZE	1760
9.50.1.2 MAX_MS_TRANSFER_ROUTE_MSG	1760
9.50.1.3 MAX_MSC_ADDRESS_SIZE	1760
9.50.1.4 MAX_MSE_TWS_MSG	1760
9.50.1.5 MAX_SMS_LIST_SIZE	1760
9.50.1.6 MAX_SMS_MESSAGE_SIZE	1760
9.50.2 Typedef Documentation	1760
9.50.2.1 sMSCAddressInfo	1760
9.50.2.2 sMSEtwsMessageInfo	1760
9.50.2.3 sMSEtwsPlmnInfo	1761
9.50.2.4 sSMSMessageModelInfo	1761
9.50.2.5 sSMSMTMessageInfo	1761
9.50.2.6 sMSOnIMSInfo	1761
9.50.2.7 sMSTransferRouteMTMessageInfo	1761
9.50.3 Enumeration Type Documentation	1762
9.50.3.1 eqmiCbkSetStatus	1762
9.50.4 Function Documentation	1762
9.50.4.1 pack_sms_SendSMS(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SendSMS_t *reqParam)	1762
9.50.4.2 pack_sms_SetNewSMSCallback(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SetNewSMSCallback_t reqParam)	1763
9.50.4.3 pack_sms_SLQSDeleteSMS(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSDeleteSMS_t *reqParam)	1763
9.50.4.4 pack_sms_SLQSGetSMS(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSGetSMS_t *reqParam)	1763
9.50.4.5 pack_sms_SLQSGetSMSList(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSGetSMSList_t *reqParam)	1764
9.50.4.6 pack_sms_SLQSModifySMSStatus(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSModifySMSStatus_t *reqParam)	1764

9.50.4.7	unpack_sms_SendSMS(uint8_t *pResp, uint16_t respLen, unpack_sms_SendSMS_t *pOutput)	1765
9.50.4.8	unpack_sms_SetNewSMSCallback(uint8_t *pResp, uint16_t respLen, unpack_sms_SetNewSMSCallback_t *Output)	1765
9.50.4.9	unpack_sms_SetNewSMSCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_sms_SetNewSMSCallback_ind_t *pOutput)	1765
9.50.4.10	unpack_sms_SLQSDeleteSMS(uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSDeleteSMS_t *pOutput)	1766
9.50.4.11	unpack_sms_SLQSGetSMS(uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetSMS_t *pOutput)	1766
9.50.4.12	unpack_sms_SLQSGetSMSList(uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetSMSList_t *pOutput)	1766
9.50.4.13	unpack_sms_SLQSModifySMSSStatus(uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSModifySMSSStatus_t *pOutput)	1767
9.50.4.14	unpack_sms_SLQSWmsMemoryFullCallBack_ind(uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSWmsMemoryFullCallBack_ind_t *pOutput)	1767
9.51	SwiDataTypes.h File Reference	1768
9.51.1	Detailed Description	1768
9.51.2	Macro Definition Documentation	1768
9.51.2.1	QMI_NO_LTE_FW_SUPPORT	1768
9.51.2.2	QMI_TLV_PLACEHOLDER	1768
9.51.2.3	SWI_API	1768
9.51.2.4	UNUSEDPARAM	1768
9.51.3	Typedef Documentation	1769
9.51.3.1	BOOL	1769
9.51.3.2	BYTE	1769
9.51.3.3	CHAR	1769
9.51.3.4	FLOAT	1769
9.51.3.5	INT32	1769
9.51.3.6	INT8	1769
9.51.3.7	LPCSTR	1769
9.51.3.8	SHORT	1769
9.51.3.9	ULONG	1769
9.51.3.10	ULONGLONG	1769

9.51.3.11 USHORT	1769
9.51.3.12 WORD	1769
9.52 swiloc.h File Reference	1769
9.52.1 Function Documentation	1769
9.52.1.1 pack_swiloc_SwiLocGetAutoStart(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1769
9.52.1.2 pack_swiloc_SwiLocSetAutoStart(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiloc_SwiLocSetAutoStart_t *reqArg)	1770
9.52.1.3 unpack_swiloc_SwiLocGetAutoStart(uint8_t *pResp, uint16_t respLen, unpack_swiloc_SwiLocGetAutoStart_t *pOutput)	1770
9.52.1.4 unpack_swiloc_SwiLocSetAutoStart(uint8_t *pResp, uint16_t respLen)	1771
9.53 swioma.h File Reference	1771
9.53.1 Macro Definition Documentation	1772
9.53.1.1 LIBPACK_MAX_SWIOMA_STR_LEN	1772
9.53.2 Function Documentation	1772
9.53.2.1 pack_swioma_SLQSOMADMAAlertCallback(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1772
9.53.2.2 pack_swioma_SLQSOMADMCancelSession(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMCancelSession_t reqParam)	1773
9.53.2.3 pack_swioma_SLQSOMADMGetSessionInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMGetSessionInfo_t reqParam)	1774
9.53.2.4 pack_swioma_SLQSOMADMGetSettings(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1774
9.53.2.5 pack_swioma_SLQSOMADMSendSelection(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMSendSelection_t reqParam)	1775
9.53.2.6 pack_swioma_SLQSOMADMSetSettings(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMSetSettings_t reqParam)	1775
9.53.2.7 pack_swioma_SLQSOMADMStartSession(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMStartSession_t reqParam)	1776
9.53.2.8 unpack_swioma_SLQSOMADMAAlertCallback(uint8_t *pResp, uint16_t respLen)	1777
9.53.2.9 unpack_swioma_SLQSOMADMAAlertCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMAAlertCallback_ind_t *pOutput)	1777
9.53.2.10 unpack_swioma_SLQSOMADMCancelSession(uint8_t *pResp, uint16_t respLen)	1778
9.53.2.11 unpack_swioma_SLQSOMADMGetSessionInfo(uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMGetSessionInfo_t *pOutput)	1778

9.53.2.12	unpack_swioma_SLQSOMADMGetSettings(uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMGetSettings_t *pOutput)	1779
9.53.2.13	unpack_swioma_SLQSOMADMSendSelection(uint8_t *pResp, uint16_t respLen)	1779
9.53.2.14	unpack_swioma_SLQSOMADMSetSettings(uint8_t *pResp, uint16_t respLen) .	1780
9.53.2.15	unpack_swioma_SLQSOMADMStartSession(uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMStartSession_t *pOutput)	1780
9.54	SWIWWANCMAPI.h File Reference	1781
9.55	uim.h File Reference	1781
9.55.1	Macro Definition Documentation	1782
9.55.1.1	MAX_DESCRIPTION_LENGTH	1782
9.55.1.2	MAX_ICCID_LENGTH	1782
9.55.1.3	MAX_NO_OF_APPLICATIONS	1782
9.55.1.4	MAX_NO_OF_SLOTS	1782
9.55.1.5	MAX_SLOTS_STATUS	1783
9.55.1.6	UIM_MAX_DESCRIPTION_LENGTH	1783
9.55.1.7	UIM_MAX_NO_OF_APPLICATIONS	1783
9.55.1.8	UIM_MAX_NO_OF_SLOTS	1783
9.55.1.9	UIM_UINT8_MAX_STRING_SZ	1783
9.55.2	Function Documentation	1783
9.55.2.1	pack_uim_ChangePin(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_ChangePin_t *reqArg)	1783
9.55.2.2	pack_uim_GetCardStatus(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)	1783
9.55.2.3	pack_uim_ReadTransparent(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_ReadTransparent_t *reqArg)	1784
9.55.2.4	pack_uim_SetPinProtection(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SetPinProtection_t *reqArg)	1784
9.55.2.5	pack_uim_SLQSUIEventRegister(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIEventRegister_t *reqArg)	1784
9.55.2.6	pack_uim_SLQSUIGetSlotsStatus(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1785
9.55.2.7	pack_uim_SLQSUIPowerDown(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIPowerDown_t *reqArg)	1785
9.55.2.8	pack_uim_SLQSUIPowerUp(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIPowerUp_t *reqArg)	1786

9.55.2.9	pack_uim_SLQSUIMSwitchSlot(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMSwitchSlot_t *reqArg)	1786
9.55.2.10	pack_uim_UnblockPin(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_UnblockPin_t *reqArg)	1786
9.55.2.11	pack_uim_VerifyPin(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_VerifyPin_t *reqArg)	1787
9.55.2.12	unpack_uim_ChangePin(uint8_t *pResp, uint16_t respLen, unpack_uim_ChangePin_t *pOutput)	1787
9.55.2.13	unpack_uim_GetCardStatus(uint8_t *pResp, uint16_t respLen, unpack_uim_GetCardStatus_t *pOutput)	1788
9.55.2.14	unpack_uim_ReadTransparent(uint8_t *pResp, uint16_t respLen, unpack_uim_ReadTransparent_t *pOutput)	1788
9.55.2.15	unpack_uim_SetPinProtection(uint8_t *pResp, uint16_t respLen, unpack_uim_SetPinProtection_t *pOutput)	1788
9.55.2.16	unpack_uim_SetUimSlotStatusChangeCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_uim_SetUimSlotStatusChangeCallback_ind_t *pOutput)	1789
9.55.2.17	unpack_uim_SLQSUIEventRegister(uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIEventRegister_t *pOutput)	1789
9.55.2.18	unpack_uim_SLQSUIGetSlotsStatus(uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIGetSlotsStatus_t *pOutput)	1790
9.55.2.19	unpack_uim_SLQSUIPowerDown(uint8_t *pResp, uint16_t respLen)	1790
9.55.2.20	unpack_uim_SLQSUIPowerUp(uint8_t *pResp, uint16_t respLen)	1790
9.55.2.21	unpack_uim_SLQSUISetStatusChangeCallBack_ind(uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUISetStatusChangeCallBack_ind_t *pOutput)	1791
9.55.2.22	unpack_uim_SLQSUIMSwitchSlot(uint8_t *pResp, uint16_t respLen)	1791
9.55.2.23	unpack_uim_UnblockPin(uint8_t *pResp, uint16_t respLen, unpack_uim_UnblockPin_t *pOutput)	1792
9.55.2.24	unpack_uim_VerifyPin(uint8_t *pResp, uint16_t respLen, unpack_uim_VerifyPin_t *pOutput)	1792
9.56	wds.h File Reference	1792
9.56.1	Macro Definition Documentation	1798
9.56.1.1	BYT_STAT_STAT_MASK	1798
9.56.1.2	IPV6_ADDRESS_ARRAY_SIZE	1798
9.56.1.3	MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE	1798
9.56.1.4	PACK_WDS_IPV4	1798
9.56.1.5	PACK_WDS_IPV6	1798

9.56.1.6	WDS_DHCP_MAX_NUM_OPTIONS	1798
9.56.1.7	WDS_DHCP_OPTION_DATA_BUF_SIZE	1798
9.56.2	Typedef Documentation	1798
9.56.2.1	UnpackQmiProfileInfo	1798
9.56.3	Function Documentation	1798
9.56.3.1	pack_wds_DHCPv4ClientLeaseChange(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_DHCPv4ClientLeaseChange_t *reqArg)	1798
9.56.3.2	pack_wds_GetAutoconnect(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1799
9.56.3.3	pack_wds_GetByteTotals(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1799
9.56.3.4	pack_wds_GetConnectionRate(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1800
9.56.3.5	pack_wds_GetDataBearerTechnology(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1800
9.56.3.6	pack_wds_GetDefaultProfile(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetDefaultProfile_t *reqParam)	1801
9.56.3.7	pack_wds_GetDefaultProfileNum(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetDefaultProfileNum_t *reqParam)	1801
9.56.3.8	pack_wds_GetDormancyState(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetDormancyState_t *reqParam)	1801
9.56.3.9	pack_wds_GetLastMobileIPError(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetLastMobileIPError_t *reqParam)	1802
9.56.3.10	pack_wds_GetMobileIP(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetMobileIP_t *reqParam)	1802
9.56.3.11	pack_wds_GetMobileIPProfile(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetMobileIPProfile_t *reqParam)	1803
9.56.3.12	pack_wds_GetPacketStatistics(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetPacketStatistics_t *pReq)	1803
9.56.3.13	pack_wds_GetPacketStatus(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetPacketStatus_t *reqParam)	1804
9.56.3.14	pack_wds_GetSessionDuration(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetSessionDuration_t *reqParam)	1804
9.56.3.15	pack_wds_GetSessionState(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1805
9.56.3.16	pack_wds_RMSetTransferStatistics(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_RMSetTransferStatistics_t *reqParam)	1805
9.56.3.17	pack_wds_SetAutoconnect(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetAutoconnect_t *reqArg)	1806

9.56.3.18	pack_wds_SetDefaultProfile(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetDefaultProfile_t *reqParam)	1806
9.56.3.19	pack_wds_SetDefaultProfileNum(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetDefaultProfileNum_t *pReqParam)	1807
9.56.3.20	pack_wds_SetMobileIP(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetMobileIP_t *reqArg)	1807
9.56.3.21	pack_wds_SetMobileIPParameters(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetMobileIPParameters_t *reqArg)	1807
9.56.3.22	pack_wds_SetMobileIPProfile(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetMobileIPProfile_t *reqParam)	1808
9.56.3.23	pack_wds_SLQSCreateProfile(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSCreateProfile_t *reqArg)	1808
9.56.3.24	pack_wds_SLQSDeleteProfile(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSDeleteProfile_t *reqParam)	1809
9.56.3.25	pack_wds_SLQSGet3GPPConfigItem(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1809
9.56.3.26	pack_wds_SLQSGetCurrDataSystemStat(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetCurrDataSystemStat_t *pReqParam) . . .	1810
9.56.3.27	pack_wds_SLQSGetCurrentChannelRate(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1810
9.56.3.28	pack_wds_SLQSGetDataBearerTechnology(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetDataBearerTechnology_t *pReqParam)	1811
9.56.3.29	pack_wds_SLQSGetDUNCallInfo(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetDUNCallInfo_t *reqParam)	1811
9.56.3.30	pack_wds_SLQSGetProfileSettings(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetProfileSettings_t *reqArg)	1812
9.56.3.31	pack_wds_SLQSGetRuntimeSettings(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetRuntimeSettings_t *reqArg)	1812
9.56.3.32	pack_wds_SLQSModifyProfile(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSModifyProfile_t *reqArg)	1812
9.56.3.33	pack_wds_SLQSResetPacketStatics(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1813
9.56.3.34	pack_wds_SLQSSet3GPPConfigItem(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSet3GPPConfigItem_t *reqParam)	1813
9.56.3.35	pack_wds_SLQSSetIPFamilyPreference(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSetIPFamilyPreference_t *pReqParam)	1814
9.56.3.36	pack_wds_SLQSSetWdsEventCallback(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSetWdsEventCallback_t *reqArg)	1814
9.56.3.37	pack_wds_SLQSSetDHCPv4ClientConfig(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSetDHCPv4ClientConfig_t *pReq) . . .	1815

9.56.3.38	pack_wds_SLQSSGetLoopback(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1815
9.56.3.39	pack_wds_SLQSSSetDHCPv4ClientConfig(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSSetDHCPv4ClientConfig_t *reqArg) . . .	1816
9.56.3.40	pack_wds_SLQSSSetLoopback(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSSetLoopback_t *reqArg)	1816
9.56.3.41	pack_wds_SLQSStartDataSession(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSStartDataSession_t *reqArg)	1817
9.56.3.42	pack_wds_SLQSStopDataSession(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSStopDataSession_t *reqArg)	1817
9.56.3.43	pack_wds_SLQSWdsGoActive(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1818
9.56.3.44	pack_wds_SLQSWdsGoDormant(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)	1818
9.56.3.45	pack_wds_SLQSWdsSetEventReport(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSWdsSetEventReport_t *reqArg)	1818
9.56.3.46	pack_wds_SLQSWdsSwiPDPRuntimeSettings(pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSWdsSwiPDPRuntimeSettings_t *reqParam)	1819
9.56.3.47	unpack_wds_DHCPv4ClientLease_ind(uint8_t *pResp, uint16_t respLen, unpack_wds_DHCPv4ClientLease_ind_t *pOutput)	1819
9.56.3.48	unpack_wds_DHCPv4ClientLeaseChange(uint8_t *pResp, uint16_t respLen) . .	1820
9.56.3.49	unpack_wds_GetAutoconnect(uint8_t *pResp, uint16_t respLen, unpack_wds_GetAutoconnect_t *pOutput)	1820
9.56.3.50	unpack_wds_GetByteTotals(uint8_t *pResp, uint16_t respLen, unpack_wds_GetByteTotals_t *pOutput)	1821
9.56.3.51	unpack_wds_GetConnectionRate(uint8_t *pResp, uint16_t respLen, unpack_wds_GetConnectionRate_t *pOutput)	1821
9.56.3.52	unpack_wds_GetDataBearerTechnology(uint8_t *pResp, uint16_t respLen, unpack_wds_GetDataBearerTechnology_t *pOutput)	1821
9.56.3.53	unpack_wds_GetDefaultProfile(uint8_t *pResp, uint16_t respLen, unpack_wds_GetDefaultProfile_t *pOutput)	1822
9.56.3.54	unpack_wds_GetDefaultProfileNum(uint8_t *pResp, uint16_t respLen, unpack_wds_GetDefaultProfileNum_t *pOutput)	1822
9.56.3.55	unpack_wds_GetDormancyState(uint8_t *pResp, uint16_t respLen, unpack_wds_GetDormancyState_t *pOutput)	1822
9.56.3.56	unpack_wds_GetLastMobileIPError(uint8_t *pResp, uint16_t respLen, unpack_wds_GetLastMobileIPError_t *pOutput)	1823
9.56.3.57	unpack_wds_GetMobileIP(uint8_t *pResp, uint16_t respLen, unpack_wds_GetMobileIP_t *pOutput)	1823

9.56.3.58	unpack_wds_GetMobileIPProfile(uint8_t *pResp, uint16_t respLen, unpack_wds_GetMobileIPProfile_t *pOutput)	1824
9.56.3.59	unpack_wds_GetPacketStatistics(uint8_t *pResp, uint16_t respLen, unpack_wds_GetPacketStatistics_t *pOutput)	1824
9.56.3.60	unpack_wds_GetPacketStatus(uint8_t *pResp, uint16_t respLen, unpack_wds_GetPacketStatus_t *pOutput)	1824
9.56.3.61	unpack_wds_GetSessionDuration(uint8_t *pResp, uint16_t respLen, unpack_wds_GetSessionDuration_t *pOutput)	1825
9.56.3.62	unpack_wds_GetSessionState(uint8_t *pResp, uint16_t respLen, unpack_wds_GetSessionState_t *pOutput)	1825
9.56.3.63	unpack_wds_RMSetTransferStatistics(uint8_t *pResp, uint16_t respLen, unpack_wds_RMSetTransferStatistics_t *pOutput)	1826
9.56.3.64	unpack_wds_RMTransferStatistics_ind(uint8_t *pResp, uint16_t respLen, unpack_RMTransferStatistics_ind_t *pOutput)	1826
9.56.3.65	unpack_wds_SetAutoconnect(uint8_t *pResp, uint16_t respLen)	1826
9.56.3.66	unpack_wds_SetDefaultProfile(uint8_t *pResp, uint16_t respLen)	1827
9.56.3.67	unpack_wds_SetDefaultProfileNum(uint8_t *pResp, uint16_t respLen)	1827
9.56.3.68	unpack_wds_SetMobileIP(uint8_t *pResp, uint16_t respLen)	1828
9.56.3.69	unpack_wds_SetMobileIPParameters(uint8_t *pResp, uint16_t respLen)	1828
9.56.3.70	unpack_wds_SetMobileIPProfile(uint8_t *pResp, uint16_t respLen, unpack_wds_SetMobileIPProfile_t *pOutput)	1828
9.56.3.71	unpack_wds_SLQSCreateProfile(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSCreateProfile_t *pOutput)	1829
9.56.3.72	unpack_wds_SLQSDeleteProfile(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSDeleteProfile_t *pOutput)	1829
9.56.3.73	unpack_wds_SLQSGet3GPPConfigItem(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGet3GPPConfigItem_t *pOutput)	1829
9.56.3.74	unpack_wds_SLQSGetCurrDataSystemStat(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetCurrDataSystemStat_t *pOutput)	1830
9.56.3.75	unpack_wds_SLQSGetCurrentChannelRate(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetCurrentChannelRate_t *pOutput)	1830
9.56.3.76	unpack_wds_SLQSGetDataBearerTechnology(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetDataBearerTechnology_t *pOutput)	1831
9.56.3.77	unpack_wds_SLQSGetDUNCallInfo(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetDUNCallInfo_t *pOutput)	1831
9.56.3.78	unpack_wds_SLQSGetProfileSettings(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetProfileSettings_t *pOutput)	1831

9.56.3.79	unpack_wds_SLQSGetRuntimeSettings(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetRuntimeSettings_t *pOutput)	1832
9.56.3.80	unpack_wds_SLQSModifyProfile(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSModifyProfile_t *pOutput)	1832
9.56.3.81	unpack_wds_SLQSResetPacketStatics(uint8_t *pResp, uint16_t respLen) . . .	1832
9.56.3.82	unpack_wds_SLQSSet3GPPConfigItem(uint8_t *pResp, uint16_t respLen) . . .	1833
9.56.3.83	unpack_wds_SLQSSetIPFamilyPreference(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetIPFamilyPreference_t *pOutput)	1833
9.56.3.84	unpack_wds_SLQSSetPacketSrvStatusCallback(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetPacketSrvStatusCallback_t *pOutput)	1833
9.56.3.85	unpack_wds_SLQSSetWdsEventCallback(uint8_t *pResp, uint16_t respLen) . .	1834
9.56.3.86	unpack_wds_SLQSSetWdsEventCallback_ind(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetWdsEventCallback_ind_t *pOutput)	1834
9.56.3.87	unpack_wds_SLQSSetDHCPv4ClientConfig(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetDHCPv4ClientConfig_t *pOutput)	1835
9.56.3.88	unpack_wds_SLQSSetLoopback(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetLoopback_t *pOutput)	1835
9.56.3.89	unpack_wds_SLQSSetDHCPv4ClientConfig(uint8_t *pResp, uint16_t respLen)	1835
9.56.3.90	unpack_wds_SLQSSetLoopback(uint8_t *pResp, uint16_t respLen)	1836
9.56.3.91	unpack_wds_SLQSSetDataSession(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetDataSession_t *pOutput)	1836
9.56.3.92	unpack_wds_SLQSSetStopDataSession(uint8_t *pResp, uint16_t respLen)	1836
9.56.3.93	unpack_wds_SLQSSetWdsGoActive(uint8_t *pResp, uint16_t respLen)	1837
9.56.3.94	unpack_wds_SLQSSetWdsGoDormant(uint8_t *pResp, uint16_t respLen)	1837
9.56.3.95	unpack_wds_SLQSSetWdsSetEventReport(uint8_t *pResp, uint16_t respLen) . . .	1838
9.56.3.96	unpack_wds_SLQSSetWdsSwiPDPRuntimeSettings(uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetWdsSwiPDPRuntimeSettings_t *pOutput)	1838

Chapter 1

Welcome to the Sierra Wireless Linux QMI SDK API Reference Guide

This API reference guide contains information about all the modules, in the Sierra Wireless Linux QMI SDK (SLQS). Use the tabs at the top of the page to navigate the reference guide.

- Modules tab – lists all the service modules and provides a link to the API header file in each module.
- References tab – links to reference material.

1.1 Important Notice

Due to the nature of wireless communications, transmission and reception of data can never be guaranteed. Data may be delayed, corrupted (i.e., have errors) or be totally lost. Although significant delays or losses of data are rare when wireless devices such as the Sierra Wireless modem are used in a normal manner with a well-constructed network, the Sierra Wireless modem should not be used in situations where failure to transmit or receive data could result in damage of any kind to the user or any other party, including but not limited to personal injury, death, or loss of property. Sierra Wireless accepts no responsibility for damages of any kind resulting from delays or errors in data transmitted or received using the Sierra Wireless modem, or for failure of the Sierra Wireless modem to transmit or receive such data.

1.2 Limitation of Liability

The information in this manual is subject to change without notice and does not represent a commitment on the part of Sierra Wireless. SIERRA WIRELESS AND ITS AFFILIATES SPECIFICALLY DISCLAIM LIABILITY FOR ANY AND ALL DIRECT, INDIRECT, SPECIAL, GENERAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES INCLUDING, BUT NOT LIMITED TO, LOSS OF PROFITS OR REVENUE OR ANTICIPATED PROFITS OR REVENUE ARISING OUT OF THE USE OR INABILITY TO USE ANY SIERRA WIRELESS PRODUCT, EVEN IF SIERRA WIRELESS AND/OR ITS AFFILIATES HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES OR THEY ARE FORESEEABLE OR FOR CLAIMS BY ANY THIRD PARTY.

Notwithstanding the foregoing, in no event shall Sierra Wireless and/or its affiliates aggregate liability arising under or in connection with the Sierra Wireless product, regardless of the number of events, occurrences, or claims giving rise to liability, be in excess of the price paid by the purchaser for the Sierra Wireless product.

1.3 Patents

Patents This product may contain technology developed by or for Sierra Wireless Inc. This product includes technology licensed from QUALCOMM® 3G. This product is manufactured or sold by Sierra Wireless Inc. or its affiliates under one or more patents licensed from InterDigital Group.

1.4 Copyright

© 2011-2015 Sierra Wireless. All rights reserved.

1.5 Trademarks

AirCard® and Heart of the Wireless Machine® are registered trademarks of Sierra Wireless. Watcher® is a trademark of Sierra Wireless, registered in the European Community. Sierra Wireless, the Sierra Wireless logo, the red wave design, and the red-tipped antenna are trademarks of Sierra Wireless. Windows® is a registered trademark of Microsoft Corporation. QUALCOMM® is a registered trademark of QUALCOMM Incorporated. Used under license. Linux is a registered trademark of Linus Torvalds. Other trademarks are the property of the respective owners.

1.6 Contact Information

If you have any questions about the Sierra Wireless Linux SDK, contact your Sierra Wireless account manager.

Consult our website for up-to-date product descriptions, documentation, application notes, firmware upgrades, troubleshooting tips, and press releases: <http://www.sierrawireless.com>.

Chapter 2

Module Index

2.1 Modules

Here is a list of all modules:

Device Connectivity Service (DCS)	33
Wireless Data Service (WDS)	34
Device Management Service (DMS)	35
Network Access Service (NAS)	36
CallBack registration (CBK)	37
Short Message Service (SMS)	38
Position Determination Service (PDS)	39
Card Application Toolkit (CAT)	40
Remote Management Service (RMS)	41
Firmware Management Service (FMS)	42
Open Mobile Alliance Service (OMA)	43
Specific Absorption Rate (SAR)	44
SWI Open Mobile Alliance Service (SWIOMA)	45
Voice Service (VOICE)	46
Non-service specific APIs (SWI)	47
User Identity Module Service (UIM)	48
Audio Service (AUDIO)	49
Quality of Service (QOS)	50
IMS Service (IMS)	51
SWI Audio Service(SWIAUDIO)	52
Location Service(LOC)	53
Thermal Mitigation Device(TMD)	54

Chapter 3

Namespace Index

3.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Tables	55
------------------------	----

Chapter 4

Data Structure Index

4.1 Data Structures

Here are the data structures with brief descriptions:

_getIndicationRegResp	57
_GetProfileSettingIn	58
_GetProfileSettingOut	58
_getResetInfoNotification	59
_getTransLayerInfoResp	60
_getTransNWRegInfoResp	61
_MitigationDevInfo	61
_modemTempNotification	62
_packetSrvStatus	63
_qaQmi3GPP2BroadcastCfgInfo	64
_qaQmi3GPPBroadcastCfgInfo	65
_setIndicationRegReq	66
_slqs3GPPConfigItem	67
SlqsNas3GppNetworkRAT	69
_slqsNetworkScanInfo	70
_SLQSOMADMSessionInfo	71
_SLQSOMADMSettings	74
_SLQSOMADMSettingsReqParams	76
_SLQSOMADMSettingsReqParams3	77
_SLQSSwiGetHostDevInfoParams	78
_SLQSSwiGetOSInfoParams	79
_SLQSSwiGetSerialNoExtParams	80
_SLQSSwiSetHostDevInfoParams	81
_SLQSSwiSetOSInfoParams	82
_sysSelectPrefInfo	83
_sysSelectPrefParams	88
_transLayerinfo	94
_transLayerInfoNotification	94
_transNWRegInfoNotification	95
accelAcceptReady_s	96
accelTempAcceptReady_s	97
acqOrderPref	97
ActPilotPNElement	98
AddCDMASysInfo	99
AddSysInfo	99

airTimer	100
allCallsAlphaIDInfo	101
allCallsDiagInfo	101
allCallsUUSInfo	102
alphaIDInfo	102
altitudeSrcInfo	103
altSrcInfo_t	104
appStats	105
appStatus	108
arrAlertingPattern	111
arrAlertingType	112
arrAlphaID	113
arrCalledPartyNum	114
arrCallEndReason	114
arrCallInfo	115
arrConnectPartyNum	116
arrDiagInfo	116
arrRedirPartyNum	117
arrRemotePartyName	117
arrRemotePartyNum	118
arrSvcOption	118
arrUUSInfo	119
authenticateResult	120
authenticationData	120
BandCapabilityResp	121
BdsSV	123
BdsSVInfo	123
BroadcastConfig	124
burstDTMFInfo	125
CallBarringSysInfo	125
callBarStatus	126
calledPartyInfo	127
calledPartySubAdd	129
callerIDInfo	130
callFwdTypeAndPlan	130
callFWExtInfo	131
callFWInfo	133
callInfo	134
callingPartyInfo	136
cardResult	138
cardStatus	138
CarrierImage_t	140
CatAlphaIdentifierTlv	141
CatCommonEventTlv	141
CatEndProactiveSessionTlv	142
CATEventDataType	142
CatEventIDDDataTlv	143
CatEventListTlv	143
CatRefreshTlv	144
ccSUPSType	144
CDMABroadcastConfig	145
CDMAChannel	146
CDMAECIOThresh	147
CDMAInfo	147
cdmaMsgDecodingParams	148
cdmaMsgEncodingParams	151
CDMARSSIThresh	153
CDMASSInfo	154

cdmaSSInfo	154
CDMASysInfo	155
CDMASysInfoExt	158
CellDb	159
cellParams	159
changeUIMPIN	161
ChannelRate	162
channelRate	162
CLIPResp	163
CLIRResp	164
CikInfo	164
CNAPResp	165
COLPResp	166
COLRResp	167
CommInfo	168
ConnectionStatus	169
connectionStatus	170
connectNumInfo	171
CrashInfo	172
crashInfoParams	173
CrashInfoParams	174
crashInformation	174
CreateProfileIn	176
CreateProfileOut	176
CSGID	177
CUGInfo	178
curAMRConfig	179
CurrDataSysStat	180
currentCatEvent	180
CurrentImgList	181
currentPLMN	182
CurrImageInfo	183
CurrNetworkInfo	184
currNetworkInfo	186
custFeaturesInfo	187
custFeaturesSetting	189
custSettingInfo	191
custSettingList	192
dataBearers	192
DataBearerTech	193
DataBearerTechExt	195
dataBearerTechnology	196
dataRate	197
dataSrvCapabilities	198
DataStatusDetail	199
DataULongLongTlv	201
DataULongTlv	201
DcsUsbPortNames	201
delAssistDataStatus	201
depersonalizationInformation	202
detailSvcInfo	203
DeviceConfigDetail	205
DHCPOption	206
DHCPOptionList	206
diagInfo	207
dirNum	208
dms_ActivationStatusTlv	208
dms_OperatingModeTlv	209

dmsCurrentPRLInfo	210
DMScustSettingInfo	210
DMScustSettingList	211
DMSgetCustomFeatureV2	212
DMSgetCustomInput	212
dmsIndicationRegisterReq	213
dmsSwiGetResetInfo	213
Domain	214
DomainNameList	215
DRCPParams	215
DTMFInfo	216
DTMFLengths	217
DUNCallInfoInd	217
dunchannelRate	218
ecioListElement	219
ECIOThresh	219
ECTNum	220
encryptedPIN1	221
eriDataparams	222
ERIFileparams	222
errorRateListElement	223
eTWSPLMNInfoTlv	224
extDispRecInfo	224
FactorySequenceNumber	225
fileAttributes	225
fileInfo	229
FirmwareUpdatStat	229
FMSImageElement	231
FMSImageIdElement	232
FMSImageIDEntries	233
FMSImageList	234
FMSPrefImageList	234
fwinfo_s	235
GERANInfo	236
geranInstInfo	238
getAllCallInformation	238
getAllCallRmtPtyName	239
getAllCallRmtPtyNum	240
GetAudioPathConfigReq	240
GetAudioPathConfigResp	241
GetAudioProfileReq	243
GetAudioProfileResp	244
GetAudioVoTLBConfigReq	245
GetAudioVoTLBConfigResp	246
getCallFWExtInfo	246
getCallFWInfo	247
getCustomFeatureV2	247
getCustomInput	248
getDUNCallInfoReq	248
getDUNCallInfoResp	250
getDyingGaspCfg	253
getDyingGaspStatistics	253
GetErrRateResp	254
GetHRPDStatsResp	255
GetIMSSMSConfigParams	255
GetIMSUserConfigParams	256
GetIMSVoIPConfigResp	257
GetInstIDResp	260

GetM2MAudioProfileReq	260
GetM2MAudioProfileResp	261
GetM2MAudioVolumeReq	262
GetM2MAudioVolumeResp	262
GetM2MAVMuteReq	263
GetM2MAVMuteResp	263
GetM2MSpkrGainReq	264
GetM2MSpkrGainResp	265
getMsgWaitingInfo	265
GetNetworkTimeResp	266
GetRegMgrConfigParams	266
GetSessionIDResp	267
GetSIPConfigResp	268
GnssData	269
gnssSvInfoNotification	270
GPRSQoS	271
GPRSRequestedQoS	272
GPSSStateInfo	273
gpsTime_s	277
gsmCellInfo	278
GSMRSSIThresh	279
GSMSrvStatusInfo	280
GSMSysInfo	281
gyroAcceptReady_s	284
gyroTempAcceptReady_s	284
HDRECIOThresh	285
HDRIOThresh	286
HDRPersonalityInd	286
HDRPersonalityResp	287
HDRProtSubtypResp	288
HDRRSSIThresh	288
HDRSINRThresh	289
HDRSINRThreshold	290
HDRSSInfo	290
hdrSSInfo	292
HDRSysInfo	292
homeSIDNID	294
hotSwapStatus	295
image_info_t	295
ImageElement	296
ImageIdElement	297
ImageIDentries	298
ImageList	298
IMSAIndRegisterInfo	299
imsaPdpStatusInfo	300
imsaRatStatusInfo	301
IMSARegistrationStatus	301
imsaRegStatusInfo	302
IMSAServiceStatus	303
IMSASupportedFieldsResp	305
IMSASupportedMsgInfo	306
imsaSvcStatusInfo	306
imsCfgIndRegisterInfo	307
imsRegMgrConfigInfo	309
imsSIPConfigInfo	310
imsSMSConfigInfo	311
imsUserConfigInfo	312
imsVoIPConfigInfo	312

IndFieldsList	315
infoInterFreq	315
IOThresh	316
IPv4Addr	317
IPv6Addr	318
IPV6AddressInfo	318
ipv6AddressInfo	319
IPV6GWAddressInfo	319
IPv6TrafCls	320
LibPackGPRSRequestedQoS	320
LibpackProfile3GPP	321
LibpackProfile3GPP2	327
LibPackprofile_3GPP	333
LibPackprofile_3GPP2	339
LibPackQoSClassID	344
LibPackTFTIDParams	345
LibPackUMTSQoS	347
LibPackUMTSReqQoSSigInd	349
lineCtrlInfo	350
loc_BdsSV	351
loc_BdsSVInfo	351
loc_CellDb	352
loc_ClkInfo	353
loc_GnssData	354
loc_gpsTime	355
loc_LocApplicationInfo	356
loc_precisionDilution	357
loc_satelliteInfo	358
loc_sensorDataUsage	360
loc_SV	361
loc_SVInfo	362
loc_svUsedforFix	363
LocApplicationInfo	363
LocDelAssDataReq	365
LOCEventRegisterReqResp	366
LOCExtPowerStateReqResp	368
LocInjectPositionReq	368
LocInjectSensorDataReq	372
LocSetCradleMountReq	374
LOCStartReq	374
LOCStopReq	376
LteCQIParm	377
lteEARFCN	377
lteGsmCellInfo	378
LTEInfo	379
LTEInfoInterfreq	382
LTEInfoIntrafreq	382
LTEInfoNeighboringGSM	384
LTEInfoNeighboringWCDMA	385
LteNasReleaseInfo_s	386
ltePCI	386
lteRsrpinformation	387
LTERSRPThresh	388
LTERSRQThresh	388
LTERSSIThresh	389
LteSccRxInfoResp	389
LTESigRptCfg	390
LTESigRptConfig	391

IteSnrinformation	392
LTESNRThresh	392
LTESNRThreshold	393
LTESSInfo	393
IteSSInfo	394
LTESysInfo	395
IteWcdmaCellInfo	397
messageModeTlv	398
messageWaitingInfoContent	399
minBasedIMSI	400
mitigationDevList	400
MNRInfo	401
ModifyProfileIn	402
ModifyProfileOut	403
msgWaitingInfo	403
namName	404
nas_acqOrderPref	404
nas_AddCDMASysInfo	405
nas_AddSysInfo	406
nas_CallBarringSysInfo	406
nas_callBarStatus	407
nas_CDMAECIOThresh	408
nas_CDMAInfo	409
nas_CDMARSSIThresh	410
nas_CDMASysInfo	410
nas_CDMASysInfoExt	414
nas_cellParams	415
nas_CommInfo	416
nas_CSGID	417
nas_currentPLMN	418
nas_dataSrvCapabilities	419
nas_detailSvcInfo	420
nas_ecioListElement	422
nas_errorRateListElement	422
nas_GERANInfo	423
nas_geranInstInfo	425
nas_gsmCellInfo	426
nas_GSMRSSIThresh	427
nas_GSMSrvStatusInfo	427
nas_GSMSysInfo	428
nas_HDRECIOThresh	431
nas_HDRIOThresh	432
nas_HDRRSSIThresh	432
nas_HDRSINRThreshold	433
nas_HDRSysInfo	434
nas_infoInterFreq	436
nas_IteGsmCellInfo	437
nas_LTEInfo	439
nas_LTEInfoInterfreq	441
nas_LTEInfoIntrafreq	441
nas_LTEInfoNeighboringGSM	444
nas_LTEInfoNeighboringWCDMA	444
nas_IteRsrpinformation	445
nas_LTERSRRPThresh	446
nas_LTERSRRQThresh	446
nas_LTERSSIThresh	447
nas_LTESigRptConfig	447
nas_IteSnrinformation	448

nas_LTESNRThreshold	449
nas_LTESysInfo	449
nas_lteWcdmaCellInfo	452
nas_MNRInfo	453
nas_netSelectionPref	454
nas_nmrCellInfo	454
nas_PhyCaAggPcellInfo	456
nas_PhyCaAggScellDIBw	457
nas_PhyCaAggScellIndex	457
nas_PhyCaAggScellIndType	458
nas_PhyCaAggScellInfo	459
nas_qaQmi3Gpp2TimeZone	461
nas_QmiNas3GppNetworkInfo	462
nas_QmiNas3GppNetworkRAT	463
nas_QmisNasPcsDigit	463
nas_RejectReasonTlv	464
nas_RFInfoTlv	465
nas_roamIndList	465
nas_rsrqInformation	466
nas_RxSigInfo	467
nas_rxSignalStrengthListElement	468
nas_SccRxInfo	469
nas_servSystem	470
nas_SignalStrengthTlv	471
nas_SLQSSignalStrengthsIndReq	472
nas_SLQSSignalStrengthsInformation	473
nas_SLQSSignalStrengthsTlv	474
nas_SrvStatusInfo	474
nas_sysInfoCommon	475
nas_TDSCDMAECIOThresh	478
nas_TDSCDMARSCPTthresh	478
nas_TDSCDMARSSIThresh	479
nas_TDSCDMASINRThresh	479
nas_timeInfo	480
nas_UMTSInfo	482
nas_UMTSinstInfo	484
nas_umtsLTENbrCell	485
nas_UniversalTime	486
nas_wcdmaCellInfo	487
nas_WCDMAECIOThresh	488
nas_WCDMAInfoLTENeighborCell	489
nas_WCDMARSSIThresh	490
nas_WCDMASysInfo	490
NASBandPreferenceTlv	494
nasCellLocationInfoResp	494
NASEmergencyModeTlv	495
nasGet3GPP2SubscriptionInfoReq	496
nasGet3GPP2SubscriptionInfoResp	496
nasGetHDRColorCodeResp	497
nasGetLTECphyCa	498
NasGetLTECphyCaInfo	498
nasGetLTECphyCaResp	498
nasGetSigInfoResp	499
nasGetSysInfoResp	500
nasGetTxRxInfoReq	503
nasGetTxRxInfoResp	504
NASGWAcqOrderPrefTlv	504
nasIndicationRegisterReq	504

nasInitNetworkReg	507
NASLTEBandPreferenceTlv	508
NASLteNasReleaseInfoTlv	509
NASModePreferenceTlv	509
NASNetSelPreferenceTlv	509
nasNetworkTime	510
nasOperatorNameResp	511
NASOTAMessageTlv	512
NASPhyCaAggPcellInfo	512
NASPhyCaAggScellIDIBw	513
NASPhyCaAggScellIndex	514
NASPhyCaAggScellIndType	514
NASPhyCaAggScellInfo	515
nasPLMNNameReq	516
nasPLMNNameResp	517
NASPRLPreferenceTlv	520
NASQmiCbkNasSwiOTAMessageInd	520
NASQmiCbkNasSystemSelPrefInd	520
NASRoamPreferenceTlv	521
NASServDomainPrefTlv	521
NASServingSystemInfo	522
nasSigInfo	523
nasSwiGetChannelLockResp	524
NasSwiIndReg	525
nasSwiSetChannelLockReq	527
nasSysInfo	527
NASTimeInfoTlv	530
nasTimers	531
netSelectionPref	531
NetStats	532
NetworkDebugResp	533
NetworkStat1x	534
NetworkStatEVDO	536
newMTMessageTlv	538
newPwdData	538
nmrCellInfo	539
NSSAudioCtrl	541
NWProfile	541
omaDmConfigTlv	542
omaDmConfigTlvExt	543
omaDmFotaTlv	545
omaDmFotaTlvExt	547
omaDmNotificationsTlv	549
operatorNameString	549
OperatorPLMNData	550
operatorPLMNList	551
pack_dms_ActivateAutomatic_t	551
pack_dms_GetCustFeaturesV2_t	552
pack_dms_ResetToFactoryDefaults_t	553
pack_dms_SetActivationStatusCallback_t	553
pack_dms_SetCrashAction_t	553
pack_dms_SetCustFeature_t	554
pack_dms_SetCustFeaturesV2_t	555
pack_dms_SetEventReport_t	556
pack_dms_SetPower_t	556
pack_dms_SetUSBComp_t	556
pack_dms_SLQSDmsSwiIndicationRegister_t	556
pack_dms_SLQSSwiGetCrashInfo_t	557

pack_dms_SLQSSwiSetDyingGaspCfg_t	557
pack_dms_SLQSSwiSetHostDevInfo_t	558
pack_dms_SLQSSwiSetOSInfo_t	559
pack_dms_UIMChangePIN_t	559
pack_dms_UIMGetControlKeyStatus_t	560
pack_dms_UIMGetICCID_t	561
pack_dms_UIMSetControlKeyProtection_t	561
pack_dms_UIMSetPINProtection_t	562
pack_dms_UIMUnblockControlKey_t	563
pack_dms_UIMUnblockPIN_t	563
pack_dms_UIMVerifyPIN_t	564
pack_fms_GetImagesPreference_t	565
pack_fms_GetStoredImages_t	565
pack_fms_SetImagesPreference_t	565
pack_loc_Delete_Assist_Data_t	566
pack_loc_EventRegister_t	567
pack_loc_SetExtPowerState_t	570
pack_loc_SetOperationMode_t	571
pack_loc_SLQSLOCGetBestAvailPos_t	571
pack_loc_SLQSLOCInjectPosition_t	572
pack_loc_SLQSLOCInjectSensorData_t	577
pack_loc_SLQSLOCInjectUTCTime_t	579
pack_loc_SLQSLOCSetCradleMountConfig_t	580
pack_loc_Start_t	580
pack_loc_Stop_t	582
pack_nas_SetACCOLC_t	582
pack_nas_SetNetworkPreference_t	583
pack_nas_SLQSGetPLMNName_t	584
pack_nas_SLQSInitiateNetworkRegistration_t	585
pack_nas_SLQSNasConfigSigInfo2_t	586
pack_nas_SLQSNasIndicationRegisterExt_t	590
pack_nas_SLQSNasSwiIndicationRegister_t	593
pack_nas_SLQSSetSignalStrengthsCallback_t	595
pack_nas_SLQSSetSysSelectionPref_t	595
pack_qmi_t	600
pack_qos_SLQSQosSwiReadApnExtraParams_t	601
pack_qos_SLQSQosSwiReadDataStats_t	601
pack_qos_SLQSSetQosEventCallback_t	602
pack_sms_SendSMS_t	602
pack_sms_SetNewSMSCallback_t	603
pack_sms_SLQSDeleteSMS_t	603
pack_sms_SLQSGetSMS_t	604
pack_sms_SLQSGetSMSList_t	605
pack_sms_SLQSModifySMSStatus_t	605
pack_swiloc_SwiLocSetAutoStart_t	606
pack_swioama_SLQSOMADMCancelSession_t	608
pack_swioama_SLQSOMADMGetSessionInfo_t	609
pack_swioama_SLQSOMADMSelectSession_t	609
pack_swioama_SLQSOMADMSetSettings_t	610
pack_swioama_SLQSOMADMStartSession_t	611
pack_uim_ChangePin_t	611
pack_uim_ReadTransparent_t	612
pack_uim_SetPinProtection_t	613
pack_uim_SLQSUIEventRegister_t	614
pack_uim_SLQSUIPowerDown_t	615
pack_uim_SLQSUIPowerUp_t	615
pack_uim_SLQSUIPowerSwitchSlot_t	616
pack_uim_UnblockPin_t	617

pack_uim_VerifyPin_t	618
pack_wds_DHCPv4ClientLeaseChange_t	619
pack_wds_GetDefaultProfile_t	619
pack_wds_GetDefaultProfileNum_t	620
pack_wds_GetDormancyState_t	620
pack_wds_GetLastMobileIPError_t	620
pack_wds_GetMobileIP_t	620
pack_wds_GetMobileIPProfile_t	620
pack_wds_GetPacketStatistics_t	621
pack_wds_GetPacketStatus_t	621
pack_wds_GetSessionDuration_t	622
pack_wds_RMSetTransferStatistics_t	622
pack_wds_SetAutoconnect_t	622
pack_wds_SetDefaultProfile_t	623
pack_wds_SetDefaultProfileNum_t	624
pack_wds_SetMobileIP_t	624
pack_wds_SetMobileIPParameters_t	624
pack_wds_SetMobileIPProfile_t	626
pack_wds_SLQSCreateProfile_t	627
pack_wds_SLQSDeleteProfile_t	628
pack_wds_SLQSGetCurrDataSystemStat_t	628
pack_wds_SLQSGetDataBearerTechnology_t	628
pack_wds_SLQSGetDUNCallInfo_t	628
pack_wds_SLQSGetProfileSettings_t	629
pack_wds_SLQSGetRuntimeSettings_t	630
pack_wds_SLQSModifyProfile_t	631
pack_wds_SLQSSet3GPPConfigItem_t	632
pack_wds_SLQSSetIPFamilyPreference_t	634
pack_wds_SLQSSetWdsEventCallback_t	634
pack_wds_SLQSSetDHCPv4ClientConfig_t	635
pack_wds_SLQSSetDHCPv4ClientConfig_t	635
pack_wds_SLQSSetLoopback_t	636
pack_wds_SLQSStartDataSession_t	637
pack_wds_SLQSStopDataSession_t	638
pack_wds_SLQSWdsSetEventReport_t	638
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t	640
PackCreateProfileOut	641
packgetDyingGaspCfg	641
packgetDyingGaspStatistics	641
PCMparams	642
PCSCFFQDNAddress	642
PCSCFFQDNAddressList	643
PCSCFIPv4ServerAddressList	644
PDSPositionData	644
PDSPosMethodStateReq	646
peerNumberInfo	647
personalizationStatus	649
PhyCaAggPcellInfo	650
PhyCaAggScellIDBw	651
PhyCaAggScellIndex	651
PhyCaAggScellIndType	652
PhyCaAggScellInfo	653
PilotSetData	655
PilotSetParams	656
pktErrRate	656
PLMNNetworkName	657
PLMNNetworkNameData	657
Port	659

precisionDilution_s	660
PrefImageList	660
prefVoiceSO	661
Profile3GPP	663
Profile3GPP2	669
ProfileIdentifier	674
protocolSubtypeElement	675
PSDetachReq	677
qaQmi3Gpp2TimeZone	677
qaQmiInterfaceInfo	678
qaQmiServingSystemParam	679
QmiCbkCatEventStatusReportInd	683
QmiCbkLocBestAvailPosInd	683
QmiCbkLocCradleMountInd	689
QmiCbkLocEngineStateInd	690
QmiCbkLocEventTimeSyncInd	691
QmiCbkLocInjectPositionInd	691
QmiCbkLocInjectSensorDataInd	692
QmiCbkLocInjectTimeInd	693
QmiCbkLocInjectUTCTimeInd	694
QmiCbkLocPositionReportInd	695
QmiCbkLocSensorStreamingInd	701
QmiCbkLocSetExtPowerConfigInd	702
QmiCbkNasLTECphyCalInfo	702
QmiCbkSwiOmaDmEventStatusReportInd	703
QmiCbkSwiOmaDmEventStatusReportIndExt	703
QmiCbkTmdMitiLvIRptInd	704
QmiCbkWdsStatisticsIndState	704
qmifwinfo_s	705
QmiNas3GppNetworkInfo	706
QmiNasGetRFBandInfoResp	708
QmiNasPerformNetworkScanResp	708
qmiSmsMessageList	708
qmiWDSDataBearerTechnology	709
QmiWdsIpAddressInfo	709
qmiWdsRunTimeSettings	710
QosClassID	714
QosEventInfo	715
QosFlowInfo	716
QosFlowInfoState	717
QosMap	718
RankIndicatorInd	719
readResult	719
readTransparentInfo	720
redirNumInfo	720
registerRefresh	722
remainingRetries	723
remotePartyName	723
remotePartyNum	724
ReqFieldsList	725
RespFieldsList	726
RFBandInfoElements	726
rmTrasferStaticsReq	727
roamIndList	728
RoamingInfo	729
roamTimer	729
RSRPThresh	730
rsrqInformation	730

RSRQThresh	731
RSSIThresh	732
RXAGCList	732
RXAVCList	733
rxInfo	734
RXPCMIIRFitr	735
RxSigInfo	737
rxSignalStrengthListElement	738
sApnExtraParams	739
satelliteInfo	740
SccRxInfo	742
sensorData	743
sensorData_t	745
sensorDataUsage_s	746
serialNumbersInfo	747
serviceProviderName	749
ServingSystemInfo	749
servSystem	751
sessionInfo	752
sessionInfoExt	753
sessionInfoTlv	753
sessionInfoTlvExt	754
SetAudioPathConfigReq	754
SetAudioProfileReq	756
SetAudioVolTLBConfigReq	758
SetAudioVolTLBConfigResp	759
setCustomSettingV2	759
setDyingGaspCfg	760
SetIMSSMSCConfigReq	760
SetIMSSMSCConfigResp	761
SetIMSUserConfigReq	762
SetIMSUserConfigResp	762
SetIMSVoIPConfigReq	763
SetIMSVoIPConfigResp	765
SetM2MAudioAVCFGRReq	766
SetM2MAudioLPBKReq	766
SetM2MAudioProfileReq	767
SetM2MAudioVolumeReq	768
SetM2MAVMuteReq	769
SetM2MSpkrGainReq	770
setPINProtection	770
SetRegMgrConfigReq	771
SetRegMgrConfigResp	772
setSignalStrengthInfo	773
SetSIPConfigReq	778
SetSIPConfigResp	779
sGetDeviceSeriesResult	779
sidNid	780
sigInfo	780
signalInfo	782
SignalStrengthDataType	783
slot_t	783
slotInf	784
slotInfo	786
slots_t	787
slqsautoconnect	788
SLQSDdeleteProfileParams	788
slqsfwinfo_s	789

SlqsNas3GppNetworkInfo	791
SlqsNasPcsDigit	792
slqssendasyncsmsparams_s	793
slqssendsmsparams_s	795
slqsSessionStateInfo	796
slqsSignalStrengthInfo	797
SLQSSignalStrengthsIndReq	800
SLQSSignalStrengthsInformation	802
slqsWdsEventInfo	803
SMSAsyncRawSend_s	805
sMSCAddress	807
SMSCAddress	807
sMSCAddressTlv	808
sMSEtwsMessage	808
SMSEtwsMessage	809
sMSEtwsMessageTlv	810
sMSEtwsPlmn	810
SMSEtwsPlmn	811
SMSEventInfo_s	811
smsMaxStorageSizeReq	813
smsMaxStorageSizeResp	813
SMSMemoryInfo	814
sMSMessageMode	815
SMSMessageMode	815
smsMsgprotocolResp	815
sMSMTMessage	816
SMSMTMessage	816
SMSOnIMS	817
sMSOnIMS	817
sMSOnIMSTlv	818
smsRouteEntry	818
smsSetRoutesReq	819
sMSTransferRouteMTMessage	820
SMSTransferRouteMTMessage	821
sQosFlowStat	822
sQosStat	823
SrvStatusInfo	824
ssdatasession_params	825
SupportedMsgList	827
SUPSInfo	828
SV	828
SVInfo	829
svUsedforFix_s	830
SWI_STRUCT_CarrierImage	831
SwiLocGetAutoStartResp	832
SwiLocSetAutoStartReq	834
swiModemStatusResp	835
SwiOTAMsg_s	836
swiPDPRuntimeSettingsReq	837
swiPDPRuntimeSettingsResp	837
swiQosFilter	840
swiQosFlow	842
swiQosGranted	846
swiQosIds	846
swiQosModifyReq	847
swiQosReq	848
swiRMTrasnferStaticsReq	848
sysInfoCommon	849

t_gpsTime	852
t_sensor	852
t_Sv	852
TDSCDMAECIOThresh	852
TDSCDMARSCPThresh	853
TDSCDMARSSIThresh	854
TDSCDMASigInfoExt	854
tdscdmaSigInfoExt	855
TDSCDMASINRCONFTresh	856
TDSCDMASINRThresh	856
tempData_t	857
tempratureData	858
TFTIDParams	859
timeInfo	861
TmdDeRegNotMitigationLvlReq	863
TmdGetMitigationDevListResp	863
TmdGetMitigationLvlReq	864
TmdGetMitigationLvlResp	865
TmdMitigationLvlIndReq	865
TmdRegNotMitigationLvlReq	866
tokenBucket	866
Tos	867
transferRouteMessageTlv	867
TransferStatInd	868
transferStatInd	869
TransferStatsDataType	869
TrStatInd	869
trueIMSI	870
TXAGCList	871
txInfo	872
TXPCMIIRFtr	873
uim_appStatus	874
uim_cardResult	877
uim_cardStatus	878
uim_changeUIMPIN	879
uim_encryptedPIN1	880
uim_fileInfo	881
uim_hotSwapStatus	882
uim_readResult	882
uim_readTransparentInfo	883
uim_remainingRetries	883
uim_sessionInformation	884
uim_setPINProtection	885
uim_slotInfo	886
uim_UIMSessionInformation	887
uim_unblockUIMPIN	888
uim_verifyUIMPIN	889
UIMAuthenticateReq	890
UIMAuthenticateResp	891
UIMChangePinReq	892
UIMDepersonalizationReq	893
UIMDepersonalizationResp	893
UIMEventRegisterReqResp	894
UIMGetCardStatusResp	894
UIMGetConfigurationReq	895
UIMGetConfigurationResp	895
UIMGetFileAttributesReq	896
UIMGetFileAttributesResp	897

UIMGetSlotsStatusResp	898
UIMPinResp	898
UIMPowerDownReq	899
UIMPowerUpReq	900
UIMReadTransparentReq	900
UIMReadTransparentResp	901
UIMRefreshCompleteReq	902
UIMRefreshEvent	903
UIMRefreshGetLastEventReq	904
UIMRefreshGetLastEventResp	905
UIMRefreshOKReq	905
UIMRefreshRegisterReq	906
UIMSessionInformation	906
UIMSetPinProtectionReq	907
UIMSlotsStatus	908
UIMSlotStatus	909
UIMSlotStatusChangeInfo	910
UIMStatusChangeInfo	910
UIMSwitchSlotReq	911
UIMUnblockPinReq	912
UIMVerifyPinReq	913
UMTSInfo	914
UMTSInstInfo	915
umtsLTENbrCell	916
UMTSMinQoS	917
UMTSQoS	920
UMTSReqQoSsigInd	922
unblockUIMPIN	923
UniversalTime	924
unpack_dms_GetActivationState_t	925
unpack_dms_GetBandCapability_t	926
unpack_dms_GetCrashAction_t	926
unpack_dms_GetCustFeature_t	927
unpack_dms_GetCustFeaturesV2_t	927
unpack_dms_GetDeviceCap_t	928
unpack_dms_GetDeviceCapabilities_t	928
unpack_dms_GetDeviceHardwareRev_t	929
unpack_dms_GetDeviceMfr_t	929
unpack_dms_GetDeviceSerialNumbers_t	930
unpack_dms_GetFirmwareInfo_t	930
unpack_dms_GetFirmwareRevision_t	931
unpack_dms_GetFirmwareRevisions_t	932
unpack_dms_GetFSN_t	933
unpack_dms_GetHardwareRevision_t	933
unpack_dms_GetIMSI_t	933
unpack_dms_GetManufacturer_t	933
unpack_dms_GetModelID_t	934
unpack_dms_GetNetworkTime_t	934
unpack_dms_GetOfflineReason_t	935
unpack_dms_GetPower_t	936
unpack_dms_GetPRLVersion_t	937
unpack_dms_GetSerialNumbers_t	937
unpack_dms_GetUSBComp_t	938
unpack_dms_GetVoiceNumber_t	938
unpack_dms_ResetToFactoryDefaults_t	938
unpack_dms_SetActivationStatusCallback_t	939
unpack_dms_SetCrashAction_t	939
unpack_dms_SetCustFeature_t	940

unpack_dms_SetCustFeaturesV2_t	940
unpack_dms_SetEventReport_ind_t	940
unpack_dms_SetEventReport_t	941
unpack_dms_SetFirmwarePreference_t	941
unpack_dms_SetPower_t	941
unpack_dms_SetUSBComp_t	942
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t	942
unpack_dms_SLQSDmsSwiGetResetInfo_t	942
unpack_dms_SLQSDmsSwiIndicationRegister_t	943
unpack_dms_SLQSGetBandCapability_t	943
unpack_dms_SLQSGetERIFile_t	946
unpack_dms_SLQSSwiClearDyingGaspStatistics_t	947
unpack_dms_SLQSSwiGetCrashInfo_t	947
unpack_dms_SLQSSwiGetDyingGaspCfg_t	948
unpack_dms_SLQSSwiGetDyingGaspStatistics_t	948
unpack_dms_SLQSSwiGetFirmwareCurr_t	949
unpack_dms_SLQSSwiGetFwUpdateStatus_t	950
unpack_dms_SLQSSwiGetHostDevInfo_t	951
unpack_dms_SLQSSwiGetOSInfo_t	952
unpack_dms_SLQSSwiGetSerialNoExt_t	952
unpack_dms_SLQSSwiSetDyingGaspCfg_t	953
unpack_dms_SLQSSwiSetHostDevInfo_t	953
unpack_dms_SLQSSwiSetOSInfo_t	954
unpack_dms_SLQSUIMGetState_t	954
unpack_dms_UIMGetControlKeyStatus_t	955
unpack_dms_UIMGetICCID_t	956
unpack_dms_UIMGetPINStatus_t	956
unpack_dms_UIMSetControlKeyProtection_t	958
unpack_dms_UIMSetPINProtection_t	959
unpack_dms_UIMUnblockControlKey_t	959
unpack_fms_GetImagesPreference_t	960
unpack_fms_GetStoredImages_t	961
unpack_fms_SetImagesPreference_t	961
unpack_loc_BestAvailPos_Ind_t	962
unpack_loc_Delete_Assist_Data_t	968
unpack_loc_DeleteAssistData_Ind_t	968
unpack_loc_EngineState_Ind_t	969
unpack_loc_EventRegister_t	970
unpack_loc_GnssSvInfo_Ind_t	970
unpack_loc_PositionRpt_Ind_t	971
unpack_loc_SetExtPowerConfig_Ind_t	977
unpack_loc_SetExtPowerState_t	978
unpack_loc_SetOperationMode_Ind_t	979
unpack_loc_SetOperationMode_t	979
unpack_loc_SLQSLOCGetBestAvailPos_t	980
unpack_loc_Start_t	980
unpack_loc_Stop_t	981
unpack_nas_GetCDMANetworkParameters_t	981
unpack_nas_GetHomeNetwork_t	982
unpack_nas_GetNetworkPreference_t	983
unpack_nas_GetRFInfo_t	984
unpack_nas_GetServingNetwork_t	985
unpack_nas_GetServingNetworkCapabilities_t	986
unpack_nas_GetSignalStrengths_t	986
unpack_nas_PerformNetworkScan_t	987
unpack_nas_SetDataCapabilitiesCallback_ind_t	988
unpack_nas_SetEventReportInd_t	988
unpack_nas_SetNasLTECphyCalIndCallback_ind_t	989

unpack_nas_SetNetworkPreference_t	990
unpack_nas_SetRoamingIndicatorCallback_ind_t	990
unpack_nas_SetServingSystemCallback_ind_t	991
unpack_nas_SLqsGetLTECphyCAInfo_t	991
unpack_nas_SLQSGetNetworkTime_t	992
unpack_nas_SLQSGetPLMNName_t	993
unpack_nas_SLQSGetServingSystem_t	993
unpack_nas_SLQSGetSignalStrength_t	996
unpack_nas_SLQSGetSysInfo_t	997
unpack_nas_SLQSGetSysSelectionPref_t	1000
unpack_nas_SLQSNasGetCellLocationInfo_t	1003
unpack_nas_SLQSNasGetSigInfo_t	1005
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t	1006
unpack_nas_SLQSNasSigInfoCallback_ind_t	1007
unpack_nas_SLQSNasSwiModemStatus_t	1007
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t	1008
unpack_nas_SLQSNasTimerCallback_ind_t	1008
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t	1009
unpack_nas_SLQSSwiGetLteCQI_t	1010
unpack_nas_SLQSSwiGetLteSccRxInfo_t	1011
unpack_nas_SLQSSysInfoCallback_ind_t	1011
unpack_omaDmConfigTlv_t	1014
unpack_omaDmFotaTlv_t	1015
unpack_omaDmNotificationsTlv_t	1017
unpack_qmi_t	1018
unpack_qos_dataRate_t	1018
unpack_qos_IPv4Addr_t	1019
unpack_qos_IPv6Addr_t	1019
unpack_qos_IPv6TrafCls_t	1020
unpack_qos_pktErrRate_t	1021
unpack_qos_Port_t	1021
unpack_qos_QosFlowInfo_t	1022
unpack_qos_QosFlowInfoState_t	1023
unpack_qos_SLQSQosGetNetworkStatus_t	1024
unpack_qos_SLQSQosSwiReadApnExtraParams_t	1025
unpack_qos_SLQSQosSwiReadDataStats_t	1026
unpack_qos_SLQSSetQosEventCallback_ind_t	1027
unpack_qos_SLQSSetQosNWStatusCallback_ind_t	1028
unpack_qos_SLQSSetQosPriEventCallback_ind_t	1028
unpack_qos_SLQSSetQosStatusCallback_ind_t	1029
unpack_qos_swiQosFilter_t	1030
unpack_qos_swiQosFlow_t	1034
unpack_qos_tokenBucket_t	1038
unpack_qos_Tos_t	1039
unpack_QosFlowStat_t	1039
unpack_RMTransferStatistics_ind_t	1040
unpack_sms_SendSMS_t	1041
unpack_sms_SetNewSMSCallback_ind_t	1042
unpack_sms_SetNewSMSCallback_t	1043
unpack_sms_SLQSDeleteSMS_t	1043
unpack_sms_SLQSGetSMS_t	1043
unpack_sms_SLQSGetSMSList_t	1044
unpack_sms_SLQSModifySMSStatus_t	1044
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t	1045
unpack_swiloc_SwiLocGetAutoStart_t	1045
unpack_swioama_SLQSOMADMAAlertCallback_ind_t	1047
unpack_swioama_SLQSOMADMGetSessionInfo_t	1048
unpack_swioama_SLQSOMADMGetSettings_t	1051

unpack_swioma_SLQSOMADMStartSession_t	1052
unpack_uim_ChangePin_t	1053
unpack_uim_GetCardStatus_t	1054
unpack_uim_ReadTransparent_t	1054
unpack_uim_SetPinProtection_t	1055
unpack_uim_SetUimSlotStatusChangeCallback_ind_t	1056
unpack_uim_SLQSUIEventRegister_t	1057
unpack_uim_SLQSUIGetSlotsStatus_t	1057
unpack_uim_SLQSUISetStatusChangeCallBack_ind_t	1058
unpack_uim_UnblockPin_t	1058
unpack_uim_VerifyPin_t	1059
unpack_wds_DHCPv4ClientLease_ind_t	1060
unpack_wds_GetAutoconnect_t	1060
unpack_wds_GetByteTotals_t	1061
unpack_wds_GetConnectionRate_t	1061
unpack_wds_GetDataBearerTechnology_t	1062
unpack_wds_GetDefaultProfile_t	1063
unpack_wds_GetDefaultProfileNum_t	1064
unpack_wds_GetDormancyState_t	1064
unpack_wds_GetLastMobileIPError_t	1065
unpack_wds_GetMobileIP_t	1065
unpack_wds_GetMobileIPProfile_t	1065
unpack_wds_GetPacketStatistics_t	1067
unpack_wds_GetPacketStatus_t	1068
unpack_wds_GetSessionDuration_t	1070
unpack_wds_GetSessionState_t	1070
unpack_wds_RMSetTransferStatistics_t	1071
unpack_wds_SetMobileIPProfile_t	1071
unpack_wds_SLQSCreateProfile_t	1071
unpack_wds_SLQSDeleteProfile_t	1071
unpack_wds_SLQSGet3GPPConfigItem_t	1072
unpack_wds_SLQSGetCurrDataSystemStat_t	1073
unpack_wds_SLQSGetCurrentChannelRate_t	1074
unpack_wds_SLQSGetDataBearerTechnology_t	1075
unpack_wds_SLQSGetDUNCallInfo_t	1075
unpack_wds_SLQSGetProfileSettings_t	1076
unpack_wds_SLQSGetRuntimeSettings_t	1077
unpack_wds_SLQSModifyProfile_t	1079
unpack_wds_SLQSSetIPFamilyPreference_t	1079
unpack_wds_SLQSSetPacketSrvStatusCallback_t	1080
unpack_wds_SLQSSetWdsEventCallback_ind_t	1081
unpack_wds_SLQSSetDHCPv4ClientConfig_t	1082
unpack_wds_SLQSSetLoopback_t	1083
unpack_wds_SLQSStartDataSession_t	1083
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t	1084
UnPackGetProfileSettingOut	1086
unpackWdsProfileParam	1086
USBCompConfig	1086
USBCompParams	1087
USSDNoWaitIndicationInfo	1089
USSDRespFNetwork	1090
USSInfo	1091
USSResp	1091
UUSInfo	1092
verifyUIMPIN	1093
voiceALSSelectLineInfo	1094
voiceALSSetLineSwitchInfo	1094
voiceAnswerCall	1095

voiceBindSubscriptionInfo	1095
voiceBurstDTMFInfo	1096
voiceCallInfoReq	1096
voiceCallInfoResp	1097
voiceCallRequestParams	1099
voiceCallResponseParams	1101
voiceContDTMFInfo	1102
voiceDTMFEventInfo	1103
voiceFlashInfo	1104
voiceGetAllCallInfo	1105
voiceGetCallBarringReq	1107
voiceGetCallBarringResp	1108
voiceGetCallFWReq	1110
voiceGetCallFWResp	1110
voiceGetCallWaitInfo	1112
voiceGetCLIPResp	1113
voiceGetCLIRResp	1115
voiceGetCNAPResp	1116
voiceGetCOLPResp	1117
voiceGetCOLRResp	1119
voiceGetConfigReq	1120
voiceGetConfigResp	1122
voiceIndicationRegisterInfo	1124
voiceInfoRec	1125
voiceManageCallsReq	1127
voiceManageCallsResp	1128
voiceOrigUSSDNoWaitInfo	1128
voiceOTASPStatusInfo	1129
voicePrivacyInfo	1130
voiceSetAllCallStatusCbkiInfo	1130
voiceSetCallBarringPwdInfo	1133
voiceSetCallBarringPwdResp	1134
voiceSetConfigReq	1135
voiceSetConfigResp	1136
voiceSetPrefPrivacy	1138
voiceSetSUPSServiceReq	1138
voiceSetSUPSServiceResp	1140
voiceStopContDTMFInfo	1141
voiceSUPSInfo	1142
voiceSUPSNotification	1145
wcdmaCellInfo	1147
WCDMAECIOTresh	1148
WCDMAInfoLTENeighborCell	1148
wcdmaLongMsgDecodingParams	1149
wcdmaMsgDecodingParams	1151
wcdmaMsgEncodingParams	1153
WCDMARSSITresh	1154
WCDMASysInfo	1154
wcdmaUARFCN	1158
wds_currNetworkInfo	1158
wds_DataULongLongTlv	1160
wds_DataULongTlv	1160
wds_DHCPLeaseOptTlv	1161
wds_DHCPLeaseStateTlv	1161
wds_DHCPOpt	1161
wds_DHCPProfileIdTlv	1162
wds_DHCPv4HWConfig	1162
wds_DHCPv4Option	1163

wds_DHCPv4OptionList	1163
wds_DHCPv4ProfileId	1164
wds_Domain	1165
wds_DomainNameList	1165
wds_GPRSQoS	1166
wds_IPv4AdTlv	1167
wds_IPV6AddressInfo	1167
wds_IPV6GWAddressInfo	1167
wds_PCSCFFQDNAddress	1168
wds_PCSCFFQDNAddressList	1169
wds_PCSCFIPv4ServerAddressList	1169
wds_ProfileIdentifier	1170
wds_profileInfo	1170
wds_TrStatInd	1171
wds_UMTSMInQoS	1171
WdsByteTotals	1174
WdsByteTotalsElmnts	1175
WdsClientLeaseChange	1175
WdsConnectionRate	1176
WdsConnectionRateElmnts	1177
WdsDHCPv4ClientLeaseInd	1178
WdsDHCPv4Config	1179
WdsDHCPv4HWConfig	1179
wdsDhcpv4HwConfig	1180
wdsDhcpv4Option	1181
WdsDHCPv4Option	1181
WdsDHCPv4OptionList	1182
wdsDhcpv4OptionList	1183
WdsDHCPv4ProfileId	1183
wdsDhcpv4ProfileId	1184
WDSGetLoopbackData	1184
WdsIpAddressInfoReq	1185
WdsPktStatisticsElmnts	1185
WdsPktStatisticsReq	1187
WdsPktStatisticsResp	1188
WdsProfileParam	1189
WdsRunTimeSettings	1189
wdsSetEventReportReq	1190
WDSSetLoopbackData	1192
WDSWICurrentChannelRates	1192

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

apdoxypages.c	Contains the module declaration for the Doxygen output. Also contains the content of the main page and related pages	1195
common.h		1195
dms.h		1200
fms.h		1258
loc.h		1262
nas.h		1279
qaCbkCatEventReportInd.h		1310
qaCbkSwiOmaDmEventReportInd.h		1311
qaGobiApiAudio.h	Audio Service API function prototypes	1312
qaGobiApiCat.h	Card Application Toolkit API function headers	1317
qaGobiApiCbk.h	Callback Service API function prototypes	1318
qaGobiApiDcs.h	Device Connectivity Service API function prototypes	1406
qaGobiApiDms.h	Device Management Service API function prototypes	1414
qaGobiApiFms.h	Firmware Management Service API function prototypes	1455
qaGobiApiIms.h	IMS Service API function prototypes	1476
qaGobiApiImsa.h	IMSA Service API function prototypes	1483
qaGobiApiLoc.h	Location API function prototypes	1486
qaGobiApiNas.h	Network Access Service API function prototypes	1493
qaGobiApiOmadm.h	Open Mobile Alliance Device Management Service API function prototypes	1545
qaGobiApiPds.h	Position Determination Service API function prototypes	1548
qaGobiApiQos.h	Quality of Service API function prototypes	1564

qaGobiApiRms.h	
Remote Management Service API function prototypes	1571
qaGobiApiSar.h	
Specific Absorption Rate API function prototypes	1573
qaGobiApiSms.h	
Short Message Service API function prototypes	1575
qaGobiApiSwi.h	
SWI API function prototypes	1599
qaGobiApiSwiAudio.h	
M2M Audio Service API function prototypes	1600
qaGobiApiSwiOmadms.h	
SWI Open Mobile Alliance Device Management Service API function prototypes SWI OMA-DM	
QMI Service revision 1.6	1606
qaGobiApiTableBandClasses.h	
Network Access Service API Band Classes table	1618
qaGobiApiTableCallControlReturnReasons.h	
Call Control Return Reasons table	1621
qaGobiApiTableCallEndReasons.h	
Wireless Data Service Call End Reasons	1621
qaGobiApiTableCarrierCodes.h	
Carrier Codes table	1637
qaGobiApiTableCodingScheme.h	
Data Coding Scheme	1639
qaGobiApiTableGpsCapabilityCodes.h	
Position Determination Service API GPS Capability Codes	1642
qaGobiApiTablePowerModes.h	
Device Management Service API Power Modes table	1643
qaGobiApiTableRadioInterfaces.h	
Network Access Service API Radio Interfaces table	1643
qaGobiApiTableRegionCodes.h	
Region Codes table	1644
qaGobiApiTableServiceOptions.h	
Voice Service Options	1644
qaGobiApiTableSupServiceInfoClasses.h	
Voice Supplementary Service Information Classes	1647
qaGobiApiTableSwiAudio.h	
Swi Audio related tables	1647
qaGobiApiTableSwiOMADMUpdateCompleteStatus.h	
Update Complete Status table	1648
qaGobiApiTableVoiceCallEndReasons.h	
Voice Service Call and supplementary services end reasons	1649
qaGobiApiTmd.h	
Thermal Mitigation Device API function prototypes	1656
qaGobiApiUim.h	
Uim Service API function prototypes	1658
qaGobiApiVoice.h	
Voice Service API function prototypes	1672
qaGobiApiWds.h	
Wireless Data Service API function prototypes	1694
qaNasGetRFBandInfo.h	1740
qaNasPerformNetworkScan.h	1741
qmerrno.h	1742
qos.h	1749
sms.h	1758
SwiDataTypes.h	
SWI data types	1768
swiloc.h	1769
swioma.h	1771

SWIWWANCMAPI.h	1781
uim.h	1781
wds.h	1792

Chapter 6

Module Documentation

6.1 Device Connectivity Service (DCS)

Files

- file [qaGobiApiDcs.h](#)
Device Connectivity Service API function prototypes.

6.1.1 Detailed Description

6.2 Wireless Data Service (WDS)

Files

- file [qaGobiApiTableCallEndReasons.h](#)
Wireless Data Service Call End Reasons.
- file [qaGobiApiTableCarrierCodes.h](#)
Carrier Codes table.
- file [qaGobiApiTableRegionCodes.h](#)
Region Codes table.
- file [qaGobiApiWds.h](#)
Wireless Data Service API function prototypes.

6.2.1 Detailed Description

6.3 Device Management Service (DMS)

Files

- file [qaGobiApiTablePowerModes.h](#)
Device Management Service API Power Modes table.
- file [qaGobiApiDms.h](#)
Device Management Service API function prototypes.

6.3.1 Detailed Description

6.4 Network Access Service (NAS)

Files

- file [qaGobiApiTableRadioInterfaces.h](#)
Network Access Service API Radio Interfaces table.
- file [qaGobiApiTableBandClasses.h](#)
Network Access Service API Band Classes table.
- file [qaGobiApiNas.h](#)
Network Access Service API function prototypes.

6.4.1 Detailed Description

6.5 CallBack registration (CBK)

Files

- file [qaGobiApiCbK.h](#)

Callback Service API function prototypes.

6.5.1 Detailed Description

6.6 Short Message Service (SMS)

Files

- file [qaGobiApiSms.h](#)
Short Message Service API function prototypes.

6.6.1 Detailed Description

6.7 Position Determination Service (PDS)

Files

- file [qaGobiApiTableGpsCapabilityCodes.h](#)
Position Determination Service API GPS Capability Codes.
- file [qaGobiApiPds.h](#)
Position Determination Service API function prototypes.

6.7.1 Detailed Description

6.8 Card Application Toolkit (CAT)

Files

- file [qaGobiApiCat.h](#)

Card Application Toolkit API function headers.

6.8.1 Detailed Description

6.9 Remote Management Service (RMS)

Files

- file [qaGobiApiRms.h](#)

Remote Management Service API function prototypes.

6.9.1 Detailed Description

6.10 Firmware Management Service (FMS)

Files

- file [qaGobiApiFms.h](#)
Firmware Management Service API function prototypes.

6.10.1 Detailed Description

6.11 Open Mobile Alliance Service (OMA)

Files

- file [qaGobiApiOmadm.h](#)

Open Mobile Alliance Device Management Service API function prototypes.

6.11.1 Detailed Description

6.12 Specific Absorption Rate (SAR)

Files

- file [qaGobiApiSar.h](#)

Specific Absorption Rate API function prototypes.

6.12.1 Detailed Description

6.13 SWI Open Mobile Alliance Service (SWIOMA)

Files

- file [qaGobiApiTableSwiOMADMUpdateCompleteStatus.h](#)
Update Complete Status table.
- file [qaGobiApiSwiOmadms.h](#)
SWI Open Mobile Alliance Device Management Service API function prototypes SWI OMA-DM QMI Service revision 1.6.

6.13.1 Detailed Description

6.14 Voice Service (VOICE)

Files

- file [qaGobiApiTableServiceOptions.h](#)
Voice Service Options.
- file [qaGobiApiTableVoiceCallEndReasons.h](#)
Voice Service Call and supplementary services end reasons.
- file [qaGobiApiTableCodingScheme.h](#)
Data Coding Scheme.
- file [qaGobiApiTableCallControlReturnReasons.h](#)
Call Control Return Reasons table.
- file [qaGobiApiTableSupServiceInfoClasses.h](#)
Voice Supplementary Service Information Classes.
- file [qaGobiApiVoice.h](#)
Voice Service API function prototypes.

6.14.1 Detailed Description

6.15 Non-service specific APIs (SWI)

Files

- file [qaGobiApiSwi.h](#)
SWI API function prototypes.

6.15.1 Detailed Description

6.16 User Identity Module Service (UIM)

Files

- file [qaGobiApiUim.h](#)
Uim Service API function prototypes.

6.16.1 Detailed Description

6.17 Audio Service (AUDIO)

Files

- file [qaGobiApiAudio.h](#)
Audio Service API function prototypes.

6.17.1 Detailed Description

6.18 Quality of Service (QOS)

Files

- file [qaGobiApiQos.h](#)

Quality of Service API function prototypes.

6.18.1 Detailed Description

6.19 IMS Service (IMS)

Files

- file [qaGobiApilms.h](#)
IMS Service API function prototypes.
- file [qaGobiApilmsa.h](#)
IMSA Service API function prototypes.

6.19.1 Detailed Description

6.20 SWI Audio Service(SWIAUDIO)

Files

- file [qaGobiApiTableSwiAudio.h](#)
Swi Audio related tables.
- file [qaGobiApiSwiAudio.h](#)
M2M Audio Service API function prototypes.

6.20.1 Detailed Description

6.21 Location Service(LOC)

Files

- file [qaGobiApiLoc.h](#)

Location API function prototypes.

6.21.1 Detailed Description

6.22 Thermal Mitigation Device(TMD)

Files

- file [qaGobiApiTmd.h](#)
Thermal Mitigation Device API function prototypes.

6.22.1 Detailed Description

Chapter 7

Namespace Documentation

7.1 Tables Namespace Reference

7.1.1 Detailed Description

[Tables](#) referenced in the API function headers:

- Table 1 - Call End Reason Codes The reason a call (either in process or connected) was ended. [qaGobiApiTableCallEndReasons.h](#)
- Table 2 - Carrier codes List of carrier identification codes. [qaGobiApiTableCarrierCodes.h](#)
- Table 3 - Region codes List of region identification codes. [qaGobiApiTableRegionCodes.h](#)
- Table 4 - GPS capability codes List of GPS capability codes. [qaGobiApiTableGpsCapabilityCodes.h](#)
- Table 5 - Radio Interfaces List of radio interface technologies. [qaGobiApiTableRadioInterfaces.h](#)
- Table 6 - Band classes List of band classes. [qaGobiApiTableBandClasses.h](#)
- Table 7 - Power modes List of operating modes. [qaGobiApiTablePowerModes.h](#)
- Table 8 - Service Options List of Service Options. [qaGobiApiTableServiceOptions.h](#)
- Table 9 - Voice Call End Reason List of Voice Call End Reason. [qaGobiApiTableVoiceCallEndReasons.h](#)
- Table 10 - Data Coding Scheme List of Data Coding Scheme. [qaGobiApiTableCodingScheme.h](#)
- Table 11 - Call Control Return Reasons List of Voice Call Control Return Reasons. [qaGobiApiTableCallControlReturnReasons.h](#)
- Table 12 - Supplementary Service Information Classes List of Voice Supplementary Service Information Classes. [qaGobiApiTableSupServiceInfoClasses.h](#)
- Table 13 - Audio Calibration Data Base(ACDB) Device List of ACDB devices. [qaGobiApiTableSwiAudio.h](#)
- Table 14 - PIFACE List of physical interfaces. [qaGobiApiTableSwiAudio.h](#)

Chapter 8

Data Structure Documentation

8.1 `_getIndicationRegResp` Struct Reference

Data Fields

- [BYTE](#) * `pRegTransLayerInfoEvt`
- [BYTE](#) * `pRegTransNWRegInfoEvt`
- [BYTE](#) * `pRegCallStatInfoEvt`

8.1.1 Detailed Description

This structure contains Get Indication Register Response parameters

Parameters

<code>pRegTrans↔ LayerInfoEvt</code>	- <ul style="list-style-type: none">• Optional 1 BYTE parameter indicating registration status of transport layer information events• Values:<ul style="list-style-type: none">– 0x00 - Disabled– 0x01 - Enabled• function SLQSGetIndicationRegister() returns a default value 0xFF if this parameter is allocated memory in the structure and no response is received from the device.
<code>pRegTransNW↔ RegInfoEvt</code>	- <ul style="list-style-type: none">• Optional 1 BYTEparameter indicating registration status of transport network registration information events• Values:<ul style="list-style-type: none">– 0x00 - Disabled– 0x01 - Enabled• function SLQSGetIndicationRegister() returns a default value 0xFF if this parameter is allocated memory in the structure and no response is received from the device.
<code>pRegCallStat↔ InfoEvt</code>	- <ul style="list-style-type: none">• Optional 1 BYTE parameter indicating registration status of call status information events• Values:<ul style="list-style-type: none">– 0x00 - Disabled– 0x01 - Enabled• function SLQSGetIndicationRegister() returns a default value 0xFF if this parameter is allocated memory in the structure and no response is received from the device.

8.1.2 Field Documentation

8.1.2.1 **BYTE*** `_getIndicationRegResp::pRegCallStatInfoEvt`

8.1.2.2 **BYTE*** `_getIndicationRegResp::pRegTransLayerInfoEvt`

8.1.2.3 **BYTE*** `_getIndicationRegResp::pRegTransNWRegInfoEvt`

8.2 `_GetProfileSettingIn` Struct Reference

Data Fields

- [BYTE ProfileType](#)
- [BYTE ProfileID](#)

8.2.1 Detailed Description

This structure contains the input parameters for `SLQSGetProfileSettings`

Parameters

<i>ProfileType</i>	<ul style="list-style-type: none"> Identifies the technology type of the profile <ul style="list-style-type: none"> 0x00 - 3GPP 0x01 - 3GPP2
<i>ProfileID</i>	<ul style="list-style-type: none"> index identifying the profile

8.2.2 Field Documentation

8.2.2.1 **BYTE** `_GetProfileSettingIn::ProfileID`

8.2.2.2 **BYTE** `_GetProfileSettingIn::ProfileType`

8.3 `_GetProfileSettingOut` Struct Reference

Data Fields

- [QmiProfileInfo curProfile](#)
- **WORD *** `pExtErrCode`

8.3.1 Detailed Description

This structure contains the profile settings retrieved by the API SLQSGetProfileSettings

Parameters

<i>curProfile</i>	<ul style="list-style-type: none"> • Structure containing details of the profile • See QmiProfileInfo for more details
<i>pExtErrCode</i>	<ul style="list-style-type: none"> • pointer to a 2 byte extended error code • Error code will only be present if error code eQCWWAN_ERR_QMI_EXTENDED_INTER↵NAL is returned by device. • See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description.

8.3.2 Field Documentation

8.3.2.1 QmiProfileInfo _GetProfileSettingOut::curProfile

8.3.2.2 WORD* _GetProfileSettingOut::pExtErrCode

8.4 _getResetInfoNotification Struct Reference

Data Fields

- [BYTE type](#)
- [BYTE source](#)

8.4.1 Detailed Description

Contains the parameters passed for SLQSSetSwiGetResetInfoCallback by the device.

Parameters

<i>type</i>	<ul style="list-style-type: none"> • type of reset or power down, possible values listed below: <ul style="list-style-type: none"> – 0 - unknown – 1 - warm – 2 - hard – 3 - crash – 4 - power down
-------------	--

<i>source</i>	<ul style="list-style-type: none"> entity which initiated the reset or power down, possible values listed below: <ul style="list-style-type: none"> 0 - unknown 1 - user requested 2 - hardware switch 3 - temperature critical 4 - voltage critical 5 - configuration update 6 - LWM2M 7 - OMA-DM 8 - FOTA
---------------	--

Note

None

8.4.2 Field Documentation

8.4.2.1 BYTE _getResetInfoNotification::source

8.4.2.2 BYTE _getResetInfoNotification::type

8.5 _getTransLayerInfoResp Struct Reference

Data Fields

- BYTE * [pRegInd](#)
- [transLayerInfo](#) * [pTransLayerInfo](#)

8.5.1 Detailed Description

This structure contains Get Transport Layer Info Response parameters

Parameters

<i>pRegInd</i>	<ul style="list-style-type: none"> Optional parameter indicating if transport layer is registered Values: <ul style="list-style-type: none"> 0x00 - Transport layer is not registered 0x01 - Transport layer is registered function SLQSGetTransLayerInfo() returns a default value 0xFF if no response is received from the device.
<i>pTransLayerInfo</i>	<ul style="list-style-type: none"> Pointer to structure of transLayerInfo. <ul style="list-style-type: none"> Optional parameter See transLayerInfo for more information
	<ul style="list-style-type: none"> function SLQSGetTransLayerInfo() returns a default value 0xFF for parameter values if no response is received from the device.

8.5.2 Field Documentation

8.5.2.1 `BYTE* _getTransLayerInfoResp::pRegInd`

8.5.2.2 `transLayerInfo* _getTransLayerInfoResp::pTransLayerInfo`

8.6 _getTransNWRegInfoResp Struct Reference

Data Fields

- `BYTE * pRegStatus`

8.6.1 Detailed Description

This structure contains transport network registration info parameter

Parameters

<i>pRegStatus</i>	-
	<ul style="list-style-type: none"> • Optional 1 BYTE parameter indicating transport layer network registration status • Values: <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - In progress – 0x02 - Failed – 0x03 - Limited Service – 0x04 - Full Service • function SLQSGetTransNWRegInfo() returns a default value 0xFF if no response is received from the device.

8.6.2 Field Documentation

8.6.2.1 `BYTE* _getTransNWRegInfoResp::pRegStatus`

8.7 _MitigationDevInfo Struct Reference

Data Fields

- `BYTE deviceIdLen`
- `CHAR deviceId [255]`

8.7.1 Detailed Description

This structure contains mitigation Level Indication request parameters

Parameters

<i>deviceIDLen</i>	<ul style="list-style-type: none"> • Number of sets of the following elements <ul style="list-style-type: none"> – deviceID
<i>deviceID</i>	<ul style="list-style-type: none"> • Mitigation device ID

8.7.2 Field Documentation

8.7.2.1 CHAR _MitigationDevInfo::deviceID[255]

8.7.2.2 BYTE _MitigationDevInfo::deviceIDLen

8.8 _modemTempNotification Struct Reference

Data Fields

- [BYTE ModemTempState](#)
- [WORD ModemTemperature](#)

8.8.1 Detailed Description

Contains the parameters passed for SLQSSetModemTempCallback by the device.

Parameters

<i>ModemTemp↔ State</i>	<ul style="list-style-type: none"> • provides the temperature state of the modem • Values: <ul style="list-style-type: none"> – 0 - unknown – 1 - normal – 2 - high(warning) – 3 - high(critical) – 4 - low(critical)
<i>Modem↔ Temperature</i>	<ul style="list-style-type: none"> • provides the temperature of the modem

Note

None

8.8.2 Field Documentation

8.8.2.1 WORD _modemTempNotification::ModemTemperature

8.8.2.2 BYTE _modemTempNotification::ModemTempState

8.9 _packetSrvStatus Struct Reference

Data Fields

- [qaQmiInterfaceInfo](#) * [pQmiInterfaceInfo](#)
- [BYTE](#) connStatus
- [BYTE](#) reconfigReqd
- [WORD](#) sessionEndReason
- [WORD](#) verboseSessnEndReasonType
- [WORD](#) verboseSessnEndReason
- [BYTE](#) ipFamily
- [WORD](#) techName
- [BYTE](#) bearerID

8.9.1 Detailed Description

Contains the parameters passed for SLQSSetPacketSrvStatusCallback by the device.

Parameters

<i>pQmiInterfaceInfo</i>	<ul style="list-style-type: none"> • See qaQmiInterfaceInfo for more information
<i>connStatus</i>	<ul style="list-style-type: none"> • Current Link Status <ul style="list-style-type: none"> – 1 - Disconnected – 2 - Connected – 3 - Suspended – 4 - Authenticating
<i>reconfigReqd</i>	<ul style="list-style-type: none"> • Indicates if the network interface on the host needs to be reconfigured <ul style="list-style-type: none"> – 0 - No need to reconfigure – 1 - Reconfiguration required
<i>sessionEndReason</i>	<ul style="list-style-type: none"> • See qaGobiApiTableCallEndReasons.h for Call End Reason, 0xFFFF means invalid value
<i>verboseSessnEndReasonType</i>	<ul style="list-style-type: none"> • Call End Reason Type <ul style="list-style-type: none"> – 0 - Unspecified – 1 - Mobile IP – 2 - Internal – 3 - Call Manager defined – 6 - 3GPP Specification defined – 7 - PPP
Generated by Doxygen	<ul style="list-style-type: none"> – 8 - EHRPD – 9 - IPv6 – 0xFFFF - invalid value

<i>verboseSessn↔ EndReason</i>	<ul style="list-style-type: none"> See qaGobiApiTableCallEndReasons.h for verbose Call End Reason. The values depend on verboseSessnEndReasonType parameter 0xFFFF means invalid value
<i>ipFamily</i>	<ul style="list-style-type: none"> IP Family of the packet data connection <ul style="list-style-type: none"> 4 - IPv4 6 - IPv6 0xFF - invalid value
<i>techName</i>	<ul style="list-style-type: none"> Technology name of the packet data connection. <ul style="list-style-type: none"> 32767 - CDMA 32764 - UMTS 30592 - EPC 30590 - EMBMS 30584 - Modem Link Local 0xFFFF - invalid value EPC is a logical interface to support LTE/eHRPD handoff. Modem Link is an interface for transferring data between entities on the AP and modem.
<i>bearerID</i>	<ul style="list-style-type: none"> Bearer ID (3GPP) or RLP ID (3GPP2) of the packet data connection 0xFF means invalid value

Note

Any parameter not returned by the device is returned as its maximum unsigned value by the callback.

8.9.2 Field Documentation

8.9.2.1 **BYTE** _packetSrvStatus::bearerID

8.9.2.2 **BYTE** _packetSrvStatus::connStatus

8.9.2.3 **BYTE** _packetSrvStatus::ipFamily

8.9.2.4 **qaQmiInterfaceInfo*** _packetSrvStatus::pQmiInterfaceInfo

8.9.2.5 **BYTE** _packetSrvStatus::reconfigReqd

8.9.2.6 **WORD** _packetSrvStatus::sessionEndReason

8.9.2.7 **WORD** _packetSrvStatus::techName

8.9.2.8 **WORD** _packetSrvStatus::verboseSessnEndReason

8.9.2.9 **WORD** _packetSrvStatus::verboseSessnEndReasonType

8.10 _qaQmi3GPP2BroadcastCfgInfo Struct Reference**Data Fields**

- BYTE** activated_ind
- WORD** num_instances
- struct [CDMABroadcastConfig](#) CDMABroadcastConfig [0x05]

8.10.1 Detailed Description

This structure contains the 3GPP2 Broadcast Configuration Information parameters

Parameters

<i>activated_ind</i>	<ul style="list-style-type: none"> • Broadcast SMS <ul style="list-style-type: none"> – 0x00 - Deactivated – 0x01 - Activated
<i>num_instances</i>	<ul style="list-style-type: none"> • Number of sets (N) of parameters Following each set describes one entry in the broadcast configuration table. <ul style="list-style-type: none"> – serviceCategory – language – selected
<i>broadcastConfig</i>	<ul style="list-style-type: none"> • A CDMABroadcastConfig structure array. • Further defined by the structure CDMABroadcastConfig

8.10.2 Field Documentation

8.10.2.1 **BYTE** _qaQmi3GPP2BroadcastCfgInfo::activated_ind

8.10.2.2 **struct CDMABroadcastConfig** _qaQmi3GPP2BroadcastCfgInfo::CDMABroadcastConfig[0x05]

8.10.2.3 **WORD** _qaQmi3GPP2BroadcastCfgInfo::num_instances

8.11 _qaQmi3GPPBroadcastCfgInfo Struct Reference

Data Fields

- [BYTE](#) activated_ind
- [WORD](#) num_instances
- struct [BroadcastConfig](#) broadcastConfig [0x05]

8.11.1 Detailed Description

This structure contains the 3GPP Broadcast Configuration Information parameters

Parameters

<i>activated_ind</i>	<ul style="list-style-type: none"> Broadcast SMS <ul style="list-style-type: none"> 0x00 - Deactivated 0x01 - Activated
<i>num_instances</i>	<ul style="list-style-type: none"> Number of sets (N) of parameters Following each set describes one entry in the broadcast configuration table. <ul style="list-style-type: none"> fromServiceId toServiceId selected
<i>broadcastConfig</i>	<ul style="list-style-type: none"> A BroadcastConfig structure array. Further defined by the structure BroadcastConfig

8.11.2 Field Documentation

8.11.2.1 BYTE _qaQmi3GPPBroadcastCfgInfo::activated_ind

8.11.2.2 struct BroadcastConfig _qaQmi3GPPBroadcastCfgInfo::broadcastConfig[0x05]

8.11.2.3 WORD _qaQmi3GPPBroadcastCfgInfo::num_instances

8.12 _setIndicationRegReq Struct Reference

Data Fields

- BYTE * [pRegTransLayerInfoEvt](#)
- BYTE * [pRegTransNWRegInfoEvt](#)
- BYTE * [pRegCallStatInfoEvt](#)

8.12.1 Detailed Description

This structure contains Indication Register request parameters

Parameters

<i>pRegTrans↔ LayerInfoEvt</i>	<ul style="list-style-type: none"> Optional 1 BYTE parameter indicating registration status of transport layer information events Values: <ul style="list-style-type: none"> 0x00 - Disabled 0x01 - Enabled NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device
------------------------------------	--

<i>pRegTransNW↔ RegInfoEvt</i>	- <ul style="list-style-type: none"> Optional 1 BYTE parameter indicating registration status of transport network registration information events Values: <ul style="list-style-type: none"> 0x00 - Disabled 0x01 - Enabled NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device
<i>pRegCallStat↔ InfoEvt</i>	- <ul style="list-style-type: none"> Optional 1 BYTE parameter indicating registration status of call status information events Values: <ul style="list-style-type: none"> 0x00 - Disabled 0x01 - Enabled NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device

8.12.2 Field Documentation

8.12.2.1 **BYTE*** `_setIndicationRegReq::pRegCallStatInfoEvt`

8.12.2.2 **BYTE*** `_setIndicationRegReq::pRegTransLayerInfoEvt`

8.12.2.3 **BYTE*** `_setIndicationRegReq::pRegTransNWRegInfoEvt`

8.13 _slqs3GPPConfigItem Struct Reference

Data Fields

- [WORD *](#) `pLTEAttachProfile`
- [WORD *](#) `pProfileList`
- [BYTE *](#) `pDefaultPDNEnabled`
- [BYTE *](#) `p3gppRelease`
- [WORD](#) `LTEAttachProfileListLen`
- [WORD *](#) `pLTEAttachProfileList`

8.13.1 Detailed Description

This structure contains the 3gpp Configuration Item information.

Parameters

<i>pLTEAttachProfile</i>	<ul style="list-style-type: none"> • LTE Attach Profile <ul style="list-style-type: none"> – points to a single WORD Value indicating the attached LTE Profile – Optional parameter with possible values 1-16 (EM/MC73xx or earlier) – function SLQSGet3GPPConfigItem() returns a default value 255 if no LTE Attach Profile is configured • This setting is deprecated on MC/EM74xx
<i>pProfileList</i>	<ul style="list-style-type: none"> • Profile List <ul style="list-style-type: none"> – an array of 4 profile configurations – Each element points to a single WORD value indicating profile – Optional parameter with possible values <ul style="list-style-type: none"> * 1 - 16 (MC/EM73xx and before) * 1 - 24 (MC/EM74xx and onwards) – function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present
<i>pDefaultPDNEnabled</i>	<ul style="list-style-type: none"> • Always Connect Default PDN <ul style="list-style-type: none"> – A single BYTE value indicating the status of Always connect default PDN <ul style="list-style-type: none"> * 0 - disabled * 1 - enabled – Optional parameter – function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present
<i>p3gppRelease</i>	<ul style="list-style-type: none"> • 3gpp release <ul style="list-style-type: none"> – A single BYTE value indicating the 3gpp release <ul style="list-style-type: none"> * 0 - Release 99 * 1 - Release 5 * 2 - Release 6 * 3 - Release 7 * 4 - Release 8 – In 9x30 and onwards <ul style="list-style-type: none"> * 5 - Release 9 * 6 - Release 10 * 7 - Release 11 – Optional parameter – function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present
<i>pLTEAttachProfileList</i>	<ul style="list-style-type: none"> • pointer to WORD array indicating LTE Attach Profile List <ul style="list-style-type: none"> – Optional parameter – possible values: 1-24 – This setting is only supported for MC/EM74xx onwards – The new equivalent option for "pLTEAttachProfile" on 74xx modems is "pLTEAttachProfileList". Please provide attach profiles in order of decreasing priority in this list.

<i>LTEAttachProfileListLen</i>	<ul style="list-style-type: none"> Number of element in pLTEAttachProfileList <ul style="list-style-type: none"> valid range: 1-24 * This setting is only supported for MC/EM74xx onwards
--------------------------------	---

8.13.2 Field Documentation

8.13.2.1 **WORD** _slqs3GPPConfigItem::LTEAttachProfileListLen

8.13.2.2 **BYTE*** _slqs3GPPConfigItem::p3gppRelease

8.13.2.3 **BYTE*** _slqs3GPPConfigItem::pDefaultPDNEnabled

8.13.2.4 **WORD*** _slqs3GPPConfigItem::pLTEAttachProfile

8.13.2.5 **WORD*** _slqs3GPPConfigItem::pLTEAttachProfileList

8.13.2.6 **WORD*** _slqs3GPPConfigItem::pProfileList

8.14 _SlqsNas3GppNetworkRAT_ Struct Reference

Data Fields

- [WORD MCC](#)
- [WORD MNC](#)
- [BYTE RAT](#)

8.14.1 Detailed Description

Contain the 3GPP radio access technology information.

Parameters

<i>MCC</i>	<ul style="list-style-type: none"> Mobile Country Code
<i>MNC</i>	<ul style="list-style-type: none"> Mobile Network Code
<i>RAT</i>	<ul style="list-style-type: none"> Radio Access Technology <ul style="list-style-type: none"> 0x04 - GERAN 0x05 - UMTS 0x08 - LTE 0x09 - TD-SCDMA

8.14.2 Field Documentation

8.14.2.1 **WORD** _SlqsNas3GppNetworkRAT_::MCC

8.14.2.2 **WORD** _SlqsNas3GppNetworkRAT_::MNC

8.14.2.3 **BYTE** _SlqsNas3GppNetworkRAT_::RAT

8.15 _slqsNetworkScanInfo Struct Reference

Data Fields

- **BYTE** * pNetworkInfoInstances
- struct SlqsNas3GppNetworkInfo * pNetworkInfo
- **BYTE** * pRATInstances
- SlqsNas3GppNetworkRAT * pRATInfo
- **BYTE** * pPCSDigitInstances
- struct SlqsNasPcsDigit * pPCSDigitInfo
- **ULONG** * pScanResult

8.15.1 Detailed Description

Contain the network scan information.

Parameters

<i>pNetworkInfo</i> ↔ <i>Instances</i> [IN↔ OUT]	<ul style="list-style-type: none"> • Upon input, maximum number of elements that the network info instance array can contain. • Upon successful output, the actual number of elements in the network info instance array.
<i>pNetworkInfo</i> ↔ OUT]	<ul style="list-style-type: none"> • Network info instance array <ul style="list-style-type: none"> – See SlqsNas3GppNetworkInfo for more information
<i>pRAT</i> ↔ <i>Instances</i> [IN↔ OUT]	<ul style="list-style-type: none"> • Upon input, maximum number of elements that the RAT info instance array can contain. • Upon successful output, the actual number of elements in the RAT info instance array.
<i>pRATInfo</i> [OUT]	<ul style="list-style-type: none"> • RAT info instance array <ul style="list-style-type: none"> – See SlqsNas3GppNetworkRAT for more information
<i>pPCSDigit</i> ↔ <i>Instances</i> [IN↔ OUT]	<ul style="list-style-type: none"> • Upon input, maximum number of elements that the PCS Digit info instance array can contain. • Upon successful output, the actual number of elements in the PCS Digit info instance array.
<i>pPCSDigitInfo</i> ↔ OUT]	<ul style="list-style-type: none"> • PCS Digit info instance array <ul style="list-style-type: none"> – See SlqsNasPcsDigit for more information
<i>pScanResult</i> ↔ OUT]	<ul style="list-style-type: none"> • status of network scan • 0x00 - scan successful • 0x01 - scan was aborted • 0x02 - scan did not complete due to a radio link failure recovery in progress

8.15.2 Field Documentation

8.15.2.1 struct SlqsNas3GppNetworkInfo* _slqsNetworkScanInfo::pNetworkInfo

8.15.2.2 BYTE* _slqsNetworkScanInfo::pNetworkInfoInstances

8.15.2.3 struct SlqsNasPcsDigit* _slqsNetworkScanInfo::pPCSDigitInfo

8.15.2.4 BYTE* _slqsNetworkScanInfo::pPCSDigitInstances

8.15.2.5 SlqsNas3GppNetworkRAT* _slqsNetworkScanInfo::pRATInfo

8.15.2.6 BYTE* _slqsNetworkScanInfo::pRATInstances

8.15.2.7 ULONG* _slqsNetworkScanInfo::pScanResult

8.16 _SLQSOMADMSessionInfo Struct Reference

Data Fields

- [BYTE * pStatus](#)
- [WORD * pUpdateCompleteStatus](#)
- [BYTE * pSeverity](#)
- [WORD * pSourceLength](#)
- [BYTE * pSource](#)
- [WORD * pPkgNameLength](#)
- [BYTE * pPkgName](#)
- [WORD * pPkgDescLength](#)
- [BYTE * pPkgDescription](#)
- [WORD * pDateLength](#)
- [BYTE * pDate](#)
- [WORD * pTimeLength](#)
- [BYTE * pTime](#)
- [BYTE * pSessionType](#)
- [BYTE * pSessionState](#)
- [WORD * pRetryCount](#)

8.16.1 Detailed Description

Structure containing the OMA DM Session Info returned by the device. Also used as input parameter to specify the size of variable parameters. (ref. notes)

Parameters

<i>pStatus</i>	<ul style="list-style-type: none"> • 1 Byte parameter indicating status(optional) <ul style="list-style-type: none"> – 0x01 - No Firmware available – 0x02 - Query Firmware Download – 0x03 - Firmware Downloading – 0x04 - Firmware Downloaded – 0x05 - Query Firmware Update – 0x06 - Firmware Updating – 0x07 - Firmware Updated
<i>pUpdate↔</i> <i>CompleteStatus</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Update Complete Status(optional) <ul style="list-style-type: none"> – See qaGobiApiTableSwiOMADMUpdateCompleteStatus.h Update Complete Status
<i>pSeverity</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating severity(optional) <ul style="list-style-type: none"> – 0x01 - Mandatory – 0x02 - Optional
<i>pSourceLength</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Vendor Name String in Bytes.(optional)
<i>pSource</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Vendor Name in ASCII(optional)
<i>pPkgName↔</i> <i>Length</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Package Name String in Bytes.(optional)
<i>pPkgName</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Package Name in ASCII(optional)
<i>pPkgDesc↔</i> <i>Length</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Package Description String in Bytes.(optional)
<i>pPkgDescription</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Package Description in ASCII(optional)
<i>pDateLength</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Package Description String in Bytes.(optional)
<i>pDate</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Package Description in ASCII
<i>pTimeLength</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Time String in Bytes.(optional)
<i>pTime</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Time String in ASCII(optional)

<i>pSessionType</i>	<ul style="list-style-type: none"> • 1 byte parameter reflects the last session started for Sprint(optional) <ul style="list-style-type: none"> – 0x00 - No session since boot – 0x01 - Sprint CI-DC Session – 0x02 - Sprint CI-PRL Session – 0x03 - Sprint CI-FUMO Session – 0x04 - Sprint HFA-DC Session – 0x05 - Sprint HFA-PRL Session – 0x06 - Sprint HFA-FUMO Session – 0x07 - Sprint NI Session
<i>pSessionState</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating session state(optional) <ul style="list-style-type: none"> – 0x01 - idle – 0x02 - active – 0x03 - pending
<i>pRetryCount</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating retries left count(optional) <ul style="list-style-type: none"> – valid values 0 to 6

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

As input parameter the members pSourceLength, pPkgNameLength pPkgDescLength, pDateLength, pTimeLength have to be specified. These should contain the initialized size of pSource, pPkgName, pPkgDescription, pDate, pTime respectively.

8.16.2 Field Documentation**8.16.2.1 BYTE* _SLQSOMADMSessionInfo::pDate****8.16.2.2 WORD* _SLQSOMADMSessionInfo::pDateLength****8.16.2.3 WORD* _SLQSOMADMSessionInfo::pPkgDescLength****8.16.2.4 BYTE* _SLQSOMADMSessionInfo::pPkgDescription****8.16.2.5 BYTE* _SLQSOMADMSessionInfo::pPkgName**

8.16.2.6 WORD* _SLQSOMADMSessionInfo::pPkgNameLength

8.16.2.7 WORD* _SLQSOMADMSessionInfo::pRetryCount

8.16.2.8 BYTE* _SLQSOMADMSessionInfo::pSessionState

8.16.2.9 BYTE* _SLQSOMADMSessionInfo::pSessionType

8.16.2.10 BYTE* _SLQSOMADMSessionInfo::pSeverity

8.16.2.11 BYTE* _SLQSOMADMSessionInfo::pSource

8.16.2.12 WORD* _SLQSOMADMSessionInfo::pSourceLength

8.16.2.13 BYTE* _SLQSOMADMSessionInfo::pStatus

8.16.2.14 BYTE* _SLQSOMADMSessionInfo::pTime

8.16.2.15 WORD* _SLQSOMADMSessionInfo::pTimeLength

8.16.2.16 WORD* _SLQSOMADMSessionInfo::pUpdateCompleteStatus

8.17 _SLQSOMADMSettings Struct Reference

Data Fields

- ULONG * pOMADMEEnabled
- BYTE * pFOTAdownload
- BYTE * pFOTAUpdate
- BYTE * pAutosdm
- BYTE * pFwAutoCheck

8.17.1 Detailed Description

Structure containing the OMA DM settings retrieved from the device

Parameters

<i>pOMADM↔ Enabled[OUT]</i>	<ul style="list-style-type: none"> 4 byte parameter indicating OMADM service enabled <ul style="list-style-type: none"> 0x00000001 - Client-initiated device configuration 0x00000002 - Network-initiated device configuration 0x00000010 - Client-initiated FUMO 0x00000020 - Network-initiated FUMO function SLQSOMADMGetSettings2() returns a default value 0xFFFFFFFF in case this parameter is not returned by the modem.
<i>pFOTA↔ Adownload[OUT]</i>	<ul style="list-style-type: none"> 1 Byte parameter indicating support for FOTA Automatic download <ul style="list-style-type: none"> 0x00 - Host permission required before downloading 0x01 - Automatically start downloading, no host permission required 0x02 - Automatically start downloading, while not roaming 0x03 - Automatically reject download 0x04 - Automatically reject download with “Enterprise Reject Policy” function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.
<i>pFOTA↔ Update[OUT]</i>	<ul style="list-style-type: none"> 1 byte parameter indicating FOTA Automatic update <ul style="list-style-type: none"> 0x00 - User permission required before updating firmware 0x01 - No user permission required before updating firmware 0x02 - User permission required, auto update on power up function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.
<i>pAutosdm[OUT]</i>	<ul style="list-style-type: none"> 1 byte parameter indicating OMA Automatic UI Alert Response <ul style="list-style-type: none"> 0x00 - Disabled 0x01 - Enabled Accept 0x02 - Enabled Reject function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.
<i>pFwAuto↔ Check[OUT]</i>	<ul style="list-style-type: none"> Optional 1 byte parameter indicating OMA Automatic Check for Firmware Update on Power-Up Response <ul style="list-style-type: none"> 0x00 - Disabled 0x01 - Enabled function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

8.17.2 Field Documentation

8.17.2.1 **BYTE*** _SLQSOMADMSettings::pAutosdm

8.17.2.2 **BYTE*** _SLQSOMADMSettings::pFOTAdownload

8.17.2.3 **BYTE*** _SLQSOMADMSettings::pFOTAUpdate

8.17.2.4 **BYTE*** _SLQSOMADMSettings::pFwAutoCheck

8.17.2.5 **ULONG*** _SLQSOMADMSettings::pOMADMEabled

8.18 _SLQSOMADMSettingsReqParams Struct Reference

Data Fields

- [BYTE FOTAdownload](#)
- [BYTE FOTAUpdate](#)
- [BYTE *](#) [pAutosdm](#)

8.18.1 Detailed Description

Structure containing the OMA DM settings to be set on the device

Parameters

<i>FOTAdownload</i>	<ul style="list-style-type: none">• 1 Byte parameter indicating support for FOTA Automatic download<ul style="list-style-type: none">– 0x00 - Firmware auto download FALSE– 0x01 - Firmware auto download TRUE
<i>FOTAUpdate</i>	<ul style="list-style-type: none">• 1 byte parameter indicating FOTA Automatic update<ul style="list-style-type: none">– 0x00 - Firmware auto update FALSE– 0x01 - Firmware auto update TRUE
<i>pAutosdm[IN]</i>	<ul style="list-style-type: none">• Optional 1 byte parameter indicating OMA Automatic UI Alert Response<ul style="list-style-type: none">– 0x00 - Disabled– 0x01 - Enabled Accept– 0x02 - Enabled Reject

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

8.18.2 Field Documentation

8.18.2.1 BYTE _SLQSOMADMSettingsReqParams::FOTAdownload

8.18.2.2 BYTE _SLQSOMADMSettingsReqParams::FOTAUpdate

8.18.2.3 BYTE* _SLQSOMADMSettingsReqParams::pAutosdm

8.19 _SLQSOMADMSettingsReqParams3 Struct Reference

Data Fields

- [BYTE FOTAdownload](#)
- [BYTE FOTAUpdate](#)
- [BYTE * pAutosdm](#)
- [BYTE * pFwAutoCheck](#)

8.19.1 Detailed Description

Structure containing the OMA DM settings to be set on the device

Parameters

<i>FOTAdownload</i>	<ul style="list-style-type: none"> • 1 Byte parameter indicating support for FOTA Automatic download <ul style="list-style-type: none"> – 0x00 - Firmware auto download FALSE – 0x01 - Firmware auto download TRUE
<i>FOTAUpdate</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating FOTA Automatic update <ul style="list-style-type: none"> – 0x00 - Firmware auto update FALSE – 0x01 - Firmware auto update TRUE
<i>pAutosdm[IN]</i>	<ul style="list-style-type: none"> • Optional 1 byte parameter indicating OMA Automatic UI Alert Response <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled Accept – 0x02 - Enabled Reject
<i>pFwAutoCheck[IN]</i>	<ul style="list-style-type: none"> • Optional 1 byte parameter indicating OMA Automatic Check for Firmware Update on Power-Up Response <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

8.19.2 Field Documentation

8.19.2.1 **BYTE** _SLQSOMADMSettingsReqParams3::FOTAdownload

8.19.2.2 **BYTE** _SLQSOMADMSettingsReqParams3::FOTAUpdate

8.19.2.3 **BYTE*** _SLQSOMADMSettingsReqParams3::pAutosdm

8.19.2.4 **BYTE*** _SLQSOMADMSettingsReqParams3::pFwAutoCheck

8.20 _SLQSSwiGetHostDevInfoParams Struct Reference**Data Fields**

- [BYTE](#) bManSize
- [CHAR](#) * pManString
- [BYTE](#) bModelSize
- [CHAR](#) * pModelString
- [BYTE](#) bSWVerSize
- [CHAR](#) * pSWVerString
- [BYTE](#) bPlasmaIDSize
- [CHAR](#) * pPlasmaIDString

8.20.1 Detailed Description

This structure is used to Get Host Device Information

Parameters

<i>bManSize</i> [IN↔ OUT]	<ul style="list-style-type: none"> • Host Device Manufacturer String Size
<i>pManString</i> [O↔ UT]	<ul style="list-style-type: none"> • Host Device Manufacturer Name(Optional parameter) • Null terminated ASCII String
<i>bModelSize</i> [I↔ N/OUT]	<ul style="list-style-type: none"> • Host Device Model String Size

<i>pModelString</i> [↔ OUT]	<ul style="list-style-type: none"> • Host Device Model String(Optional parameter) • Null terminated ASCII string.
<i>bSWVerSize</i> [↔ N/OUT]	<ul style="list-style-type: none"> • Host Device Software Version String Size
<i>pSWVerString</i> [↔ OUT]	<ul style="list-style-type: none"> • Host Device Software Version String(Optional parameter) • Null terminated ASCII string
<i>bPlasmaID</i> ↔ Size[IN/OUT]	<ul style="list-style-type: none"> • Host Device Plasma ID String Size
<i>pPlasmaID</i> ↔ String[OUT]	<ul style="list-style-type: none"> • Host Device Plasma ID String(Optional parameter) • Null terminated alphanumeric ASCII String.

8.20.2 Field Documentation

8.20.2.1 **BYTE** _SLQSSwiGetHostDevInfoParams::bManSize

8.20.2.2 **BYTE** _SLQSSwiGetHostDevInfoParams::bModelSize

8.20.2.3 **BYTE** _SLQSSwiGetHostDevInfoParams::bPlasmaIDSize

8.20.2.4 **BYTE** _SLQSSwiGetHostDevInfoParams::bSWVerSize

8.20.2.5 **CHAR*** _SLQSSwiGetHostDevInfoParams::pManString

8.20.2.6 **CHAR*** _SLQSSwiGetHostDevInfoParams::pModelString

8.20.2.7 **CHAR*** _SLQSSwiGetHostDevInfoParams::pPlasmaIDString

8.20.2.8 **CHAR*** _SLQSSwiGetHostDevInfoParams::pSWVerString

8.21 _SLQSSwiGetOSInfoParams Struct Reference

Data Fields

- [BYTE bNameSize](#)
- [CHAR * pNameString](#)
- [BYTE bVersionSize](#)
- [CHAR * pVersionString](#)

8.21.1 Detailed Description

This structure is used to Get OS Information

Parameters

<i>bNameSize</i> [I↔ N/OUT]	<ul style="list-style-type: none"> Size of Operating System Name
<i>pNameString</i> [↔ OUT]	<ul style="list-style-type: none"> Operating System Name(Optional parameter) Null terminated ASCII string
<i>bVersionSize</i> [I↔ N/OUT]	<ul style="list-style-type: none"> Operating System Version Size
<i>pVersionString</i> [↔ OUT]	<ul style="list-style-type: none"> Operating System Version String(Optional parameter) Null terminated ASCII string.

8.21.2 Field Documentation

8.21.2.1 **BYTE** _SLQSSwiGetOSInfoParams::bNameSize

8.21.2.2 **BYTE** _SLQSSwiGetOSInfoParams::bVersionSize

8.21.2.3 **CHAR*** _SLQSSwiGetOSInfoParams::pNameString

8.21.2.4 **CHAR*** _SLQSSwiGetOSInfoParams::pVersionString

8.22 _SLQSSwiGetSerialNoExtParams Struct Reference

Data Fields

- [BYTE](#) meidLength
- [CHAR *](#) pMeidString

8.22.1 Detailed Description

This structure is used to store MEID Information

Parameters

<i>meidLength</i> [O↔ UT]	<ul style="list-style-type: none"> String length of the of MEID received
<i>pMeidString</i> [O↔ UT]	<ul style="list-style-type: none"> Optional parameter Pointer to receive String containing the Mobile Equipment Identifier(MEID) of the device.

8.22.2 Field Documentation

8.22.2.1 `BYTE _SLQSSwiGetSerialNoExtParams::meidLength`

8.22.2.2 `CHAR* _SLQSSwiGetSerialNoExtParams::pMeidString`

8.23 _SLQSSwiSetHostDevInfoParams Struct Reference

Data Fields

- `BYTE bManSize`
- `CHAR * pManString`
- `BYTE bModelSize`
- `CHAR * pModelString`
- `BYTE bSWVerSize`
- `CHAR * pSWVerString`
- `BYTE bPlasmaIDSize`
- `CHAR * pPlasmaIDString`

8.23.1 Detailed Description

This structure is used to Set Host Device Information

Parameters

<i>bManSize</i> [IN]	<ul style="list-style-type: none"> • Host Device Manufacturer String Size
<i>pManString</i> [IN]	<ul style="list-style-type: none"> • Host Device Manufacturer Name(Optional parameter) • Null terminated ASCII String
<i>bModelSize</i> [IN]	<ul style="list-style-type: none"> • Host Device Model String Size
<i>pModelString</i> [IN]	<ul style="list-style-type: none"> • Host Device Model String(Optional parameter) • Null terminated ASCII string.
<i>bSWVerSize</i> [IN]	<ul style="list-style-type: none"> • Host Device Software Version String Size
<i>pSWVerString</i> [↔ IN]	<ul style="list-style-type: none"> • Host Device Software Version String(Optional parameter) • Null terminated ASCII string
<i>bPlasmaID</i> ↔ Size[IN]	<ul style="list-style-type: none"> • Host Device Plasma ID String Size
<i>pPlasmaID</i> ↔ String[IN]	<ul style="list-style-type: none"> • Host Device Plasma ID String(Optional parameter)
Generated by Doxygen	<ul style="list-style-type: none"> • Null terminated alphanumeric ASCII String.

8.23.2 Field Documentation

8.23.2.1 **BYTE** _SLQSSwiSetHostDevInfoParams::bManSize

8.23.2.2 **BYTE** _SLQSSwiSetHostDevInfoParams::bModelSize

8.23.2.3 **BYTE** _SLQSSwiSetHostDevInfoParams::bPlasmaIDSize

8.23.2.4 **BYTE** _SLQSSwiSetHostDevInfoParams::bSWVerSize

8.23.2.5 **CHAR*** _SLQSSwiSetHostDevInfoParams::pManString

8.23.2.6 **CHAR*** _SLQSSwiSetHostDevInfoParams::pModelString

8.23.2.7 **CHAR*** _SLQSSwiSetHostDevInfoParams::pPlasmaIDString

8.23.2.8 **CHAR*** _SLQSSwiSetHostDevInfoParams::pSWVerString

8.24 _SLQSSwiSetOSInfoParams Struct Reference

Data Fields

- [BYTE](#) bNameSize
- [CHAR *](#) pNameString
- [BYTE](#) bVersionSize
- [CHAR *](#) pVersionString

8.24.1 Detailed Description

This structure is used to Set OS Information

Parameters

<i>bNameSize</i> [IN]	<ul style="list-style-type: none"> • Size of Operating System Name
<i>pNameString</i> [IN]	<ul style="list-style-type: none"> • Operating System Name(Optional parameter) • Null terminated ASCII string
<i>bVersionSize</i> [IN]	<ul style="list-style-type: none"> • Operating System Version Size
<i>pVersion↔String</i> [IN]	<ul style="list-style-type: none"> • Operating System Version String(Optional parameter) • Null terminated ASCII string.

8.24.2 Field Documentation

8.24.2.1 **BYTE** _SLQSSwiSetOSInfoParams::bNameSize

8.24.2.2 **BYTE** _SLQSSwiSetOSInfoParams::bVersionSize

8.24.2.3 **CHAR*** _SLQSSwiSetOSInfoParams::pNameString

8.24.2.4 **CHAR*** _SLQSSwiSetOSInfoParams::pVersionString

8.25 _sysSelectPrefInfo Struct Reference

Data Fields

- **BYTE** * pEmerMode
- **WORD** * pModePref
- **ULONGLONG** * pBandPref
- **WORD** * pPRLPref
- **WORD** * pRoamPref
- **ULONGLONG** * pLTEBandPref
- **BYTE** * pNetSelPref
- **ULONG** * pSrvDomainPref
- **ULONG** * pGWAacOrderPref

8.25.1 Detailed Description

Structure for storing the current preferred system selection settings for the device.

Parameters

<i>pEmerMode</i>	<ul style="list-style-type: none"> • Optional parameter specifying the emergency Mode • Values: <ul style="list-style-type: none"> – 0 - OFF (normal) – 1 - ON (Emergency) • function SLQSGetSysSelectionPref() returns a default value FF if no value is returned by the device.
<i>pModePref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit Mask indicating the radio technology mode preference • Bit values: <ul style="list-style-type: none"> – Bit 0 - cdma2000 1x – Bit 1 - cdma2000 HRPD(1xEV-DO) – Bit 2 - GSM – Bit 3 - UMTS – Bit 4 - LTE • function SLQSGetSysSelectionPref() returns a default value FF if no value is returned by the device.

<i>pBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the band preference • Bit values: <ul style="list-style-type: none"> – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved • function SLQSGetSysSelectionPref() returns a default value FFFFFFFFFFFFFFFF if no value is returned by the device.
------------------	--

<i>pPRLPref</i>	<ul style="list-style-type: none">• Optional parameter indicating the CDMA PRL Preference• Values:<ul style="list-style-type: none">– 0x0001 - Acquire available system only on the A side– 0x0002 - Acquire available system only on the B side– 0x3FFF - Acquire any available systems• function SLQSGetSysSelectionPref() returns a default value FFFF if no value is returned by the device.
<i>pRoamPref</i>	<ul style="list-style-type: none">• Optional parameter indicating the roaming Preference• Values:<ul style="list-style-type: none">– 0x01 - Acquire only systems for which the roaming indicator is off– 0x02 - Acquire a system as long as its roaming indicator is not off– 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only– 0xFF - Acquire systems, regardless of their roaming indicator• function SLQSGetSysSelectionPref() returns a default value FFFF if no value is returned by the device.

<i>pLTEBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the LTE band preference • Bit Values <ul style="list-style-type: none"> – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – All other bits are reserved • function SLQSGetSysSelectionPref() returns a default value FFFFFFFFFFFFFFFF if no value is returned by the device.
---------------------	--

<i>pNetSelPref</i>	<ul style="list-style-type: none"> Optional parameter indicating network selection preference Values: <ul style="list-style-type: none"> 0x00 - Automatic network selection 0x01 - Manual network selection function SLQSGetSysSelectionPref() returns a default value FF if no value is returned by the device.
<i>pSrvDomainPref</i>	<ul style="list-style-type: none"> Optional parameter indicating Service domain preference Values: <ul style="list-style-type: none"> 0x00 - Circuit switched only 0x01 - Packet switched only 0x02 - Circuit switched and packet switched 0x03 - Packet switched attach 0x04 - Packet switched detach function SLQSGetSysSelectionPref() returns a default value FFFFFFFF if no value is returned by the device.
<i>pGWAcqOrder</i> ↔ <i>Pref</i>	<ul style="list-style-type: none"> Optional parameter indicating GSM/WCDMA Acquisition order Preference Values: <ul style="list-style-type: none"> 0x00 - Automatic 0x01 - GSM then WCDMA 0x02 - WCDMA then GSM function SLQSGetSysSelectionPref() returns a default value FFFFFFFF if no value is returned by the device.

Note

None

8.25.2 Field Documentation**8.25.2.1** `ULONGLONG* _sysSelectPrefInfo::pBandPref`**8.25.2.2** `BYTE* _sysSelectPrefInfo::pEmerMode`**8.25.2.3** `ULONG* _sysSelectPrefInfo::pGWAcqOrderPref`**8.25.2.4** `ULONGLONG* _sysSelectPrefInfo::pLTEBandPref`**8.25.2.5** `WORD* _sysSelectPrefInfo::pModePref`**8.25.2.6** `BYTE* _sysSelectPrefInfo::pNetSelPref`**8.25.2.7** `WORD* _sysSelectPrefInfo::pPRLPref`

8.25.2.8 **WORD*** `_sysSelectPrefInfo::pRoamPref`

8.25.2.9 **ULONG*** `_sysSelectPrefInfo::pSrvDomainPref`

8.26 `_sysSelectPrefParams` Struct Reference

Data Fields

- **BYTE *** `pEmerMode`
- **WORD *** `pModePref`
- **ULONGLONG *** `pBandPref`
- **WORD *** `pPRLPref`
- **WORD *** `pRoamPref`
- **ULONGLONG *** `pLTEBandPref`
- **struct** `netSelectionPref` * `pNetSelPref`
- **BYTE *** `pChgDuration`
- **BYTE *** `pMNCIncPCSDigStat`
- **ULONG *** `pSrvDomainPref`
- **ULONG *** `pGWAcqOrderPref`
- **ULONGLONG *** `pTdsdmaBandPref`
- **struct** `acqOrderPref` * `pAcqOrderPref`
- **ULONG *** `pSrvRegRestriction`
- **struct** `CSGID` * `pCSGID`
- **BYTE *** `pRAT`

8.26.1 Detailed Description

Contain the system selection preferences.

Parameters

<i>pEmerMode</i>	<ul style="list-style-type: none"> • Optional parameter specifying the emergency Mode • Values: <ul style="list-style-type: none"> – 0 - OFF (normal) – 1 - ON (Emergency)
<i>pModePref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit Mask indicating the radio technology mode preference • Bit values: <ul style="list-style-type: none"> – Bit 0 - cdma2000 1x – Bit 1 - cdma2000 HRPD(1xEV-DO) – Bit 2 - GSM – Bit 3 - UMTS – Bit 4 - LTE

<i>pBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the band preference • Bit values: <ul style="list-style-type: none"> – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved
------------------	--

<i>pPRLPref</i>	<ul style="list-style-type: none">• Optional parameter indicating the CDMA PRL Preference• Values:<ul style="list-style-type: none">– 0x0001 - Acquire available system only on the A side– 0x0002 - Acquire available system only on the B side– 0x3FFF - Acquire any available systems
<i>pRoamPref</i>	<ul style="list-style-type: none">• Optional parameter indicating the roaming Preference• Values:<ul style="list-style-type: none">– 0x01 - Acquire only systems for which the roaming indicator is off– 0x02 - Acquire a system as long as its roaming indicator is not off– 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only– 0xFF - Acquire systems, regardless of their roaming indicator

<i>pLTETestBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the LTE band preference • Bit Values <ul style="list-style-type: none"> – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – All other bits are reserved
<i>pNetSelPref</i>	<ul style="list-style-type: none"> - netSelectionPref <ul style="list-style-type: none"> • Optional parameter for specifying Network Selection Preference • Modem selects networks based on this parameter(if present). • Either of pNetSelPref or pCSGID can be set. • see netSelectionPref for more information

<i>pChgDuration</i>	<ul style="list-style-type: none"> Optional parameter specifying the duration of the change At least one system selection setting to be set if pChgDuration is populated. Values: <ul style="list-style-type: none"> 0x00 - Power cycle - Remains active until the next device power cycle 0x01 - Permanent - Remains active through power cycles until changed by client Device will use "0x01 - permanent" as default if this parameter is omitted
<i>pMNCIncPCS↔ DigStat</i>	<ul style="list-style-type: none"> Optional parameter indicating if MNC includes PCS digit pNetSelPref is expected if MNC includes PCS digit is set to 1. Values: <ul style="list-style-type: none"> TRUE - MNC is a 3 digit value; e.g., a reported value of 90 corresponds to an MNC value of 090 FALSE - MNC is a 2-digit value; e.g., a reported value of 90 corresponds to an MNC value of 90
<i>pSrvDomainPref</i>	<ul style="list-style-type: none"> Optional parameter indicating Service domain preference Values: <ul style="list-style-type: none"> 0x00 - Circuit switched only 0x01 - Packet switched only 0x02 - Circuit switched and packet switched 0x03 - Packet switched attach 0x04 - Packet switched detach
<i>pGWAcqOrder↔ Pref</i>	<ul style="list-style-type: none"> Optional parameter indicating GSM/WCDMA Acquisition order Preference Values: <ul style="list-style-type: none"> 0x00 - Automatic 0x01 - GSM then WCDMA 0x02 - WCDMA then GSM
<i>pTdscdma↔ BandPref</i>	<ul style="list-style-type: none"> Optional parameter indicating bitmask representing the TD-SCDMA band preference to be set. Values: <ul style="list-style-type: none"> 0x01 - TD-SCDMA Band A 0x02 - TD-SCDMA Band B 0x04 - TD-SCDMA Band C 0x08 - TD-SCDMA Band D 0x10 - TD-SCDMA Band E 0x20 - TD-SCDMA Band F All other bits are reserved
<i>pAcqOrderPref</i>	<ul style="list-style-type: none"> - acqOrderPref Optional parameter for specifying Acquisition Order Preference see acqOrderPref for more information

<i>pSrvReg</i> <i>Restriction</i>	<ul style="list-style-type: none"> Optional parameter indicating Network Selection Registration Restriction Preference Values: <ul style="list-style-type: none"> 0x00 - Device follows the normal registration process 0x01 - Device camps on the network according to its provisioning, but does not register 0x02 - Device selects the network for limited service All other values are reserved.
<i>pCSGID</i>	<ul style="list-style-type: none"> - CSGID Optional parameter for specifying CSG ID Either of pNetSelPref or pCSGID can be set. see CSGID for more information
<i>pRAT</i>	<ul style="list-style-type: none"> Optional parameter Radio Access Technology order Preference Values: <ul style="list-style-type: none"> 0x04 - GSM 0x05 - UMTS 0x08 - LTE 0x09 - TDSCDMA

8.26.2 Field Documentation

8.26.2.1 `struct acqOrderPref* _sysSelectPrefParams::pAcqOrderPref`

8.26.2.2 `ULONGLONG* _sysSelectPrefParams::pBandPref`

8.26.2.3 `BYTE* _sysSelectPrefParams::pChgDuration`

8.26.2.4 `struct CSGID* _sysSelectPrefParams::pCSGID`

8.26.2.5 `BYTE* _sysSelectPrefParams::pEmerMode`

8.26.2.6 `ULONG* _sysSelectPrefParams::pGWAcqOrderPref`

8.26.2.7 `ULONGLONG* _sysSelectPrefParams::pLTEBandPref`

8.26.2.8 `BYTE* _sysSelectPrefParams::pMNCIncPCSDigStat`

8.26.2.9 `WORD* _sysSelectPrefParams::pModePref`

8.26.2.10 `struct netSelectionPref* _sysSelectPrefParams::pNetSelPref`

8.26.2.11 `WORD* _sysSelectPrefParams::pPRLPref`

8.26.2.12 **BYTE*** _sysSelectPrefParams::pRAT

8.26.2.13 **WORD*** _sysSelectPrefParams::pRoamPref

8.26.2.14 **ULONG*** _sysSelectPrefParams::pSrvDomainPref

8.26.2.15 **ULONG*** _sysSelectPrefParams::pSrvRegRestriction

8.26.2.16 **ULONGLONG*** _sysSelectPrefParams::pTdsdmaBandPref

8.27 _transLayerinfo Struct Reference

Data Fields

- [BYTE TransType](#)
- [BYTE TransCap](#)

8.27.1 Detailed Description

This structure contains Transport Layer Information

Parameters

<i>TransType</i>	<ul style="list-style-type: none"> • Transport Type <ul style="list-style-type: none"> – 0x00 - IMS
<i>TransCap</i>	<ul style="list-style-type: none"> • Transport Capability • Values: <ul style="list-style-type: none"> – 0x00 - CDMA – 0x01 - GW

8.27.2 Field Documentation

8.27.2.1 **BYTE** _transLayerinfo::TransCap

8.27.2.2 **BYTE** _transLayerinfo::TransType

8.28 _transLayerInfoNotification Struct Reference

Data Fields

- [BYTE regInd](#)
- [transLayerInfo *](#) pTransLayerInfo

8.28.1 Detailed Description

Contains the parameters passed for SLQSSetTransLayerInfoCallback by the device.

Parameters

<i>regInd</i>	<ul style="list-style-type: none">Indicates whether the transport layer is registered or notValues:<ul style="list-style-type: none">0x00 - Transport layer is not registered0x01 - Transport layer is registered
<i>pTransLayerInfo</i>	<ul style="list-style-type: none">Optional parameterSee transLayerInfo for more information

Note

None

8.28.2 Field Documentation

8.28.2.1 [transLayerInfo*](#) [_transLayerInfoNotification::pTransLayerInfo](#)

8.28.2.2 [BYTE](#) [_transLayerInfoNotification::regInd](#)

8.29 _transNWRegInfoNotification Struct Reference

Data Fields

- [BYTE](#) [NWRegStat](#)

8.29.1 Detailed Description

Contains the parameters passed for SLQSSetTransNWRegInfoCallback by the device.

Parameters

<i>NWRegStat</i>	<ul style="list-style-type: none">provides the transport network registration informationValues:<ul style="list-style-type: none">0x00 - No Service0x01 - In Progress0x02 - Failed0x03 - Limited Service0x04 - Full Service
------------------	--

Note

None

8.29.2 Field Documentation

8.29.2.1 BYTE _transNWRegInfoNotification::NWRegStat

8.30 accelAcceptReady_s Struct Reference

Data Fields

- [BYTE injectEnable](#)
- [WORD samplesPerBatch](#)
- [WORD batchPerSec](#)

8.30.1 Detailed Description

This structure contains Accelerometer Accept Ready Info

Parameters

<i>injectEnable</i>	<ul style="list-style-type: none"> • GNSS location engine is ready to accept data from sensor. • Values <ul style="list-style-type: none"> • 0x01 - Ready to accept sensor data • 0x00 - Not ready to accept sensor data
<i>samplesPerBatch</i>	<ul style="list-style-type: none"> • number of samples per batch the GNSS location engine is to receive. • $\text{samplingFrequency} = \text{samplesPerBatch} * \text{batchesPerSecond}$ • samplesPerBatch must be a nonzero positive value.
<i>batchPerSec</i>	<ul style="list-style-type: none"> • LTE NAS version minor • Number of sensor-data batches the GNSS location engine is to receive per second. • BatchesPerSecond must be a nonzero positive value.

8.30.2 Field Documentation

8.30.2.1 WORD accelAcceptReady_s::batchPerSec

8.30.2.2 BYTE accelAcceptReady_s::injectEnable

8.30.2.3 WORD accelAcceptReady_s::samplesPerBatch

8.31 accelTempAcceptReady_s Struct Reference

Data Fields

- [BYTE injectEnable](#)
- [WORD samplesPerBatch](#)
- [WORD batchPerSec](#)

8.31.1 Detailed Description

This structure contains Accelerometer Temperature Accept Ready Info

Parameters

<i>injectEnable</i>	<ul style="list-style-type: none"> • GNSS location engine is ready to accept data from sensor. • Values • 0x01 - Ready to accept sensor data • 0x00 - Not ready to accept sensor data
<i>samplesPerBatch</i>	<ul style="list-style-type: none"> • number of samples per batch the GNSS location engine is to receive. • $\text{samplingFrequency} = \text{samplesPerBatch} * \text{batchesPerSecond}$ • samplesPerBatch must be a nonzero positive value.
<i>batchPerSec</i>	<ul style="list-style-type: none"> • LTE NAS version minor • Number of sensor-data batches the GNSS location engine is to receive per second. • BatchesPerSecond must be a nonzero positive value.

8.31.2 Field Documentation

8.31.2.1 **WORD** accelTempAcceptReady_s::batchPerSec

8.31.2.2 **BYTE** accelTempAcceptReady_s::injectEnable

8.31.2.3 **WORD** accelTempAcceptReady_s::samplesPerBatch

8.32 acqOrderPref Struct Reference

Data Fields

- [BYTE acqOrdeLen](#)
- [BYTE * pAcqOrder](#)

8.32.1 Detailed Description

Contain the Acquisition Order Preference.

Parameters

<i>acqOrdeLen</i>	<ul style="list-style-type: none"> Number of sets of the following elements.
<i>pAcqOrder</i>	<ul style="list-style-type: none"> Acquisition order preference to be set. Values: <ul style="list-style-type: none"> 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X 0x02 - NAS_RADIO_IF_CDMA_1xEVDO - cdma2000 HRPD (1xEV-DO) 0x04 - NAS_RADIO_IF_GSM - GSM 0x05 - NAS_RADIO_IF_UMTS - UMTS 0x08 - NAS_RADIO_IF_LTE - LTE 0x09 - NAS_RADIO_IF_TDSCDMA - TD-SCDMA

8.32.2 Field Documentation

8.32.2.1 **BYTE** acqOrderPref::acqOrdeLen

8.32.2.2 **BYTE*** acqOrderPref::pAcqOrder

8.33 ActPilotPNElement Struct Reference

Data Fields

- [WORD ActSetPilotPN](#)
- [BYTE ActSetPilotPNStrength](#)

8.33.1 Detailed Description

This structure describes Active Pilot PN elements

Parameters

<i>ActSetPilotPN</i>	<ul style="list-style-type: none"> Active Pilot PN
<i>ActSetPilotPN↔ Strength</i>	<ul style="list-style-type: none"> Active Pilot PN strength

8.33.2 Field Documentation

8.33.2.1 **WORD** ActPilotPNElement::ActSetPilotPN

8.33.2.2 **BYTE** ActPilotPNElement::ActSetPilotPNStrength

8.34 AddCDMASysInfo Struct Reference

Data Fields

- [WORD geoSysIdx](#)
- [WORD regPrd](#)

8.34.1 Detailed Description

Structure for storing the Additional CDMA System Information.

Parameters

<i>geoSysIdx</i>	<ul style="list-style-type: none">• System table index referencing the beginning of the geo in which the current serving system is present.• When the system index is not known, 0xFFFF is used.
<i>regPrd</i>	<ul style="list-style-type: none">• Registration period after the CDMA system is acquired.• When the CDMA registration period is not valid, 0xFFFF is used.

8.34.2 Field Documentation

8.34.2.1 **WORD** AddCDMASysInfo::geoSysIdx

8.34.2.2 **WORD** AddCDMASysInfo::regPrd

8.35 AddSysInfo Struct Reference

Data Fields

- [WORD geoSysIdx](#)
- [ULONG cellBroadcastCap](#)

8.35.1 Detailed Description

Structure for storing the Additional GSM and WCDMA System Information.

Parameters

<i>geoSysIdx</i>	<ul style="list-style-type: none"> • System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
<i>cellBroadcastCap</i>	<ul style="list-style-type: none"> • Cell broadcast capability of the serving system. • When the CDMA registration period is not valid, 0xFFFF is used. <ul style="list-style-type: none"> – NAS_CELL_BROADCAST_CAP_UNKNOWN - Cell broadcast support is unknown – NAS_CELL_BROADCAST_CAP_OFF - Cell broadcast is not supported – NAS_CELL_BROADCAST_CAP_ON - Cell broadcast is supported

8.35.2 Field Documentation

8.35.2.1 ULONG AddSysInfo::cellBroadcastCap

8.35.2.2 WORD AddSysInfo::geoSysIdx

8.36 airTimer Struct Reference

Data Fields

- [BYTE namID](#)
- [ULONG airTimerValue](#)

8.36.1 Detailed Description

This structure contains information about the Air Timer.

Parameters

<i>namID</i>	<ul style="list-style-type: none"> • Index of the NAM(Number Assignment Module) to be configured. • Range 0 to 3. • Some modems support only 1 or 2 NAMs. • 0xFF,if not available.
<i>airTimerValue</i>	<ul style="list-style-type: none"> • Time in minutes. • Cumulative air time is slammed. • 0xFFFFFFFF,if not available.

8.36.2 Field Documentation

8.36.2.1 **ULONG** airTimer::airTimerValue

8.36.2.2 **BYTE** airTimer::namID

8.37 allCallsAlphaIDInfo Struct Reference

Data Fields

- [BYTE](#) callID
- [alphaIDInfo](#) AlphaIDInfo

8.37.1 Detailed Description

This structure contains information for Alpha Identifier for All Calls

Parameters

<i>callID</i>	<ul style="list-style-type: none">• Unique call identifier for the call.
<i>AlphaIDInfo</i>	<ul style="list-style-type: none">• See alphaIDInfo for more information.

8.37.2 Field Documentation

8.37.2.1 [alphaIDInfo](#) allCallsAlphaIDInfo::AlphaIDInfo

8.37.2.2 **BYTE** allCallsAlphaIDInfo::callID

8.38 allCallsDiagInfo Struct Reference

Data Fields

- [BYTE](#) callID
- [diagInfo](#) DiagInfo

8.38.1 Detailed Description

This structure contains Diagnostic Information for All Calls

Parameters

<i>callID</i>	<ul style="list-style-type: none">• Unique call identifier for the call.
<i>DiagInfo</i> Generated by Doxygen	<ul style="list-style-type: none">• See diagInfo for more information.

8.38.2 Field Documentation

8.38.2.1 **BYTE** `allCallsDiagInfo::callID`

8.38.2.2 **diagInfo** `allCallsDiagInfo::DiagInfo`

8.39 allCallsUUSInfo Struct Reference

Data Fields

- [BYTE](#) `callID`
- [UUSInfo](#) `uusInfo`

8.39.1 Detailed Description

This structure contains information for User to User Signaling Service for All Calls.

Parameters

<i>callID</i>	<ul style="list-style-type: none">• Unique call identifier for the call.
<i>uusInfo</i>	<ul style="list-style-type: none">• See UUSInfo for more information.

8.39.2 Field Documentation

8.39.2.1 **BYTE** `allCallsUUSInfo::callID`

8.39.2.2 **UUSInfo** `allCallsUUSInfo::uusInfo`

8.40 alphaIDInfo Struct Reference

Data Fields

- [BYTE](#) `alphaDcs`
- [BYTE](#) `alphaLen`
- [BYTE](#) `alphaText` [255]

8.40.1 Detailed Description

This structure contains information about the Alpha Identifier.

Parameters

<i>alphaDcs</i>	<ul style="list-style-type: none"> Alpha coding scheme <ul style="list-style-type: none"> 0x01 - GSM Default_Char 0x02 - UCS2 0xFF - Not Available
<i>alphaLen</i>	<ul style="list-style-type: none"> Number of sets of the following elements: <ul style="list-style-type: none"> pAlpha_text If zero(0) then no further information exists.
<i>alphaText</i> [MAX_DESCRIPTOR_LENGTH]	<ul style="list-style-type: none"> Data encoded as per the alpha_dcs

8.40.2 Field Documentation

8.40.2.1 **BYTE** alphaInfo::alphaDcs

8.40.2.2 **BYTE** alphaInfo::alphaLen

8.40.2.3 **BYTE** alphaInfo::alphaText[255]

8.41 altitudeSrcInfo Struct Reference

Data Fields

- [ULONG source](#)
- [ULONG linkage](#)
- [ULONG coverage](#)

8.41.1 Detailed Description

This structure specifies information regarding the altitude source

Parameters

<i>source</i>	<ul style="list-style-type: none"> Specifies the source of the altitude Valid values <ul style="list-style-type: none"> 0 - Source is unknown 1 - GPS is the source 2 - Cell ID provided the source 3 - Source is enhanced cell ID 4 - Wi-Fi is the source 5 - Terrestrial source 6 - Hybrid terrestrial source 7 - Altitude database is the source
Generated by Doxygen	<ul style="list-style-type: none"> 8 - Barometric altimeter is the source 9 - Other sources

<i>linkage</i>	<ul style="list-style-type: none"> • Specifies the dependency between the horizontal and altitude position components • Valid values <ul style="list-style-type: none"> – 0 - Not specified – 1 - Fully interdependent – 2 - Depends on latitude and longitude – 3 - Fully independent
<i>coverage</i>	<ul style="list-style-type: none"> • Specifies the region of uncertainty. • Valid values <ul style="list-style-type: none"> – 0 - Not specified – 1 - Altitude uncertainty is valid at the injected horizontal position coordinates only – 2 - Altitude uncertainty applies to the position of the device regardless of horizontal position

8.41.2 Field Documentation

8.41.2.1 ULONG altitudeSrcInfo::coverage

8.41.2.2 ULONG altitudeSrcInfo::linkage

8.41.2.3 ULONG altitudeSrcInfo::source

8.42 altSrcInfo_t Struct Reference

Data Fields

- uint32_t [source](#)
- uint32_t [linkage](#)
- uint32_t [coverage](#)

8.42.1 Detailed Description

This structure specifies information regarding the altitude source

Parameters

<i>source</i>	<ul style="list-style-type: none"> • Specifies the source of the altitude • Valid values <ul style="list-style-type: none"> – 0 - Source is unknown – 1 - GPS is the source – 2 - Cell ID provided the source – 3 - Source is enhanced cell ID – 4 - Wi-Fi is the source – 5 - Terrestrial source – 6 - Hybrid terrestrial source – 7 - Altitude database is the source – 8 - Barometric altimeter is the source – 9 - Other sources
	Generated by Doxygen

<i>linkage</i>	<ul style="list-style-type: none"> • Specifies the dependency between the horizontal and altitude position components • Valid values <ul style="list-style-type: none"> – 0 - Not specified – 1 - Fully interdependent – 2 - Depends on latitude and longitude – 3 - Fully independent
<i>coverage</i>	<ul style="list-style-type: none"> • Specifies the region of uncertainty. • Valid values <ul style="list-style-type: none"> – 0 - Not specified – 1 - Altitude uncertainty is valid at the injected horizontal position coordinates only – 2 - Altitude uncertainty applies to the position of the device regardless of horizontal position

8.42.2 Field Documentation

8.42.2.1 `uint32_t altSrcInfo_t::coverage`

8.42.2.2 `uint32_t altSrcInfo_t::linkage`

8.42.2.3 `uint32_t altSrcInfo_t::source`

8.43 appStats Struct Reference

Data Fields

- `uint8_t appType`
- `uint8_t appState`
- `uint8_t persoState`
- `uint8_t persoFeature`
- `uint8_t persoRetries`
- `uint8_t persoUnblockRetries`
- `uint8_t aidLength`
- `uint8_t aidVal [255]`
- `uint8_t univPin`
- `uint8_t pin1State`
- `uint8_t pin1Retries`
- `uint8_t puk1Retries`
- `uint8_t pin2State`
- `uint8_t pin2Retries`
- `uint8_t puk2Retries`

8.43.1 Detailed Description

This structure contains Application Status Information loaded on the card.

Parameters

<i>appType</i>	<ul style="list-style-type: none"> Indicates the type of the application. <ul style="list-style-type: none"> 0 - Unknown 1 - SIM card 2 - USIM application 3 - RUIM card 4 - CSIM application 5 - ISIM application Other values are reserved for the future and are to be handled as "Unknown".
<i>appState</i>	<ul style="list-style-type: none"> Indicates the state of the application. <ul style="list-style-type: none"> 0 - Unknown 1 - Detected 2 - PIN1 or UPIN is required 3 - PUK1 or PUK for UPIN is required 4 - Personalization state must be checked 5 - PIN1 is blocked 6 - Illegal 7 - Ready
<i>persoState</i>	<ul style="list-style-type: none"> Indicates the state of the personalization for the application. <ul style="list-style-type: none"> 0 - Unknown 1 - Personalization operation is in progress 2 - Ready 3 - Personalization code is required 4 - PUK for personalization code is required 5 - Permanently blocked
<i>persoFeature</i>	<ul style="list-style-type: none"> Indicates the personalization feature. This applies only when a personalization code is required to deactivate or unblock personalization. <ul style="list-style-type: none"> 0 - GW network personalization 1 - GW network subset personalization 2 - GW service provider personalization 3 - GW corporate personalization 4 - GW UIM personalization 5 - 1X network type 1 personalization 6 - 1X network type 2 personalization 7 - 1X HRPD personalization 8 - 1X service provider personalization 9 - 1X corporate personalization 10 - 1X RUIM personalization 11 - Unknown
<i>persoRetries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to disable the personalization.

<i>persoUnblockRetries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to unblock the personalization.
<i>aidLength</i>	<ul style="list-style-type: none"> Number of sets of the following elements. i.e. aidVal If zero(0) then no aidVal information exists.
<i>aidVal[MAX_DESCRIPTION_LENGTH]</i>	<ul style="list-style-type: none"> Application identifier value.
<i>univPin</i>	<ul style="list-style-type: none"> Indicates whether UPIN replaces PIN1. <ul style="list-style-type: none"> 0 - PIN1 is used 1 - UPIN replaces PIN1
<i>pin1State</i>	<ul style="list-style-type: none"> Indicates the state of PIN1. <ul style="list-style-type: none"> 0 - Unknown 1 - Enabled and not verified 2 - Enabled and verified 3 - Disabled 4 - Blocked 5 - Permanently blocked
<i>pin1Retries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to verify PIN1.
<i>puk1Retries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to unblock PIN1.
<i>pin2State</i>	<ul style="list-style-type: none"> Indicates the state of PIN2. <ul style="list-style-type: none"> 0 - Unknown 1 - Enabled and not verified 2 - Enabled and verified 3 - Disabled 4 - Blocked 5 - Permanently blocked
<i>pin2Retries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to verify PIN2.
<i>puk2Retries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to unblock PIN2.

8.43.2 Field Documentation

8.43.2.1 uint8_t appStats::aidLength

8.43.2.2 uint8_t appStats::aidVal[255]

8.43.2.3 uint8_t appStats::appState

8.43.2.4 uint8_t appStats::appType

8.43.2.5 uint8_t appStats::persoFeature

8.43.2.6 uint8_t appStats::persoRetries

8.43.2.7 uint8_t appStats::persoState

8.43.2.8 uint8_t appStats::persoUnblockRetries

8.43.2.9 uint8_t appStats::pin1Retries

8.43.2.10 uint8_t appStats::pin1State

8.43.2.11 uint8_t appStats::pin2Retries

8.43.2.12 uint8_t appStats::pin2State

8.43.2.13 uint8_t appStats::puk1Retries

8.43.2.14 uint8_t appStats::puk2Retries

8.43.2.15 uint8_t appStats::univPin

8.44 appStatus Struct Reference

Data Fields

- [BYTE appType](#)
- [BYTE appState](#)
- [BYTE persoState](#)
- [BYTE persoFeature](#)
- [BYTE persoRetries](#)
- [BYTE persoUnblockRetries](#)
- [BYTE aidLength](#)
- [BYTE aidVal \[255\]](#)
- [BYTE univPin](#)
- [BYTE pin1State](#)
- [BYTE pin1Retries](#)
- [BYTE puk1Retries](#)
- [BYTE pin2State](#)
- [BYTE pin2Retries](#)
- [BYTE puk2Retries](#)

8.44.1 Detailed Description

This structure contains Application Status Information loaded on the card.

Parameters

<i>appType</i>	<ul style="list-style-type: none"> Indicates the type of the application. <ul style="list-style-type: none"> 0 - Unknown 1 - SIM card 2 - USIM application 3 - RUIM card 4 - CSIM application 5 - ISIM application Other values are reserved for the future and are to be handled as "Unknown".
<i>appState</i>	<ul style="list-style-type: none"> Indicates the state of the application. <ul style="list-style-type: none"> 0 - Unknown 1 - Detected 2 - PIN1 or UPIN is required 3 - PUK1 or PUK for UPIN is required 4 - Personalization state must be checked 5 - PIN1 is blocked 6 - Illegal 7 - Ready
<i>persoState</i>	<ul style="list-style-type: none"> Indicates the state of the personalization for the application. <ul style="list-style-type: none"> 0 - Unknown 1 - Personalization operation is in progress 2 - Ready 3 - Personalization code is required 4 - PUK for personalization code is required 5 - Permanently blocked
<i>persoFeature</i>	<ul style="list-style-type: none"> Indicates the personalization feature. This applies only when a personalization code is required to deactivate or unblock personalization. <ul style="list-style-type: none"> 0 - GW network personalization 1 - GW network subset personalization 2 - GW service provider personalization 3 - GW corporate personalization 4 - GW UIM personalization 5 - 1X network type 1 personalization 6 - 1X network type 2 personalization 7 - 1X HRPD personalization 8 - 1X service provider personalization 9 - 1X corporate personalization 10 - 1X RUIM personalization 11 - Unknown
<i>persoRetries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to disable the personalization.

<i>persoUnblock</i> <i>Retries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to unblock the personalization.
<i>aidLength</i>	<ul style="list-style-type: none"> Number of sets of the following elements. i.e. aidVal If zero(0) then no aidVal information exists.
<i>aidVal</i> [MAX_DESCRIPTION_LENGTH]	<ul style="list-style-type: none"> Application identifier value.
<i>univPin</i>	<ul style="list-style-type: none"> Indicates whether UPIN replaces PIN1. <ul style="list-style-type: none"> 0 - PIN1 is used 1 - UPIN replaces PIN1
<i>pin1State</i>	<ul style="list-style-type: none"> Indicates the state of PIN1. <ul style="list-style-type: none"> 0 - Unknown 1 - Enabled and not verified 2 - Enabled and verified 3 - Disabled 4 - Blocked 5 - Permanently blocked
<i>pin1Retries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to verify PIN1.
<i>puk1Retries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to unblock PIN1.
<i>pin2State</i>	<ul style="list-style-type: none"> Indicates the state of PIN2. <ul style="list-style-type: none"> 0 - Unknown 1 - Enabled and not verified 2 - Enabled and verified 3 - Disabled 4 - Blocked 5 - Permanently blocked
<i>pin2Retries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to verify PIN2.
<i>puk2Retries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to unblock PIN2.

8.44.2 Field Documentation

8.44.2.1 BYTE appStatus::aidLength

8.44.2.2 BYTE appStatus::aidVal[255]

8.44.2.3 BYTE appStatus::appState

8.44.2.4 BYTE appStatus::appType

8.44.2.5 BYTE appStatus::persoFeature

8.44.2.6 BYTE appStatus::persoRetries

8.44.2.7 BYTE appStatus::persoState

8.44.2.8 BYTE appStatus::persoUnblockRetries

8.44.2.9 BYTE appStatus::pin1Retries

8.44.2.10 BYTE appStatus::pin1State

8.44.2.11 BYTE appStatus::pin2Retries

8.44.2.12 BYTE appStatus::pin2State

8.44.2.13 BYTE appStatus::puk1Retries

8.44.2.14 BYTE appStatus::puk2Retries

8.44.2.15 BYTE appStatus::univPin

8.45 arrAlertingPattern Struct Reference

Data Fields

- [BYTE numInstances](#)
- [BYTE callID](#) [20]
- [ULONG alertingPattern](#) [20]

8.45.1 Detailed Description

This structure contains an array of Alerting Pattern.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of callID, alertingPattern that follow. • If zero(0) then no further information exists.
<i>callID</i> [<i>MAX_NO_OF_CALLS</i>]	<ul style="list-style-type: none"> • Array of Unique call identifier for the call.
<i>alertingPattern</i> [<i>MAX_NO_OF_CALLS</i>]	<ul style="list-style-type: none"> • Array of Alerting pattern. <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_ALERTING_PATTERN_1 - Pattern 1 – 0x01 - QMI_VOICE_ALERTING_PATTERN_2 - Pattern 2 – 0x02 - QMI_VOICE_ALERTING_PATTERN_3 - Pattern 3 – 0x04 - QMI_VOICE_ALERTING_PATTERN_5 - Pattern 5 – 0x05 - QMI_VOICE_ALERTING_PATTERN_6 - Pattern 6 – 0x06 - QMI_VOICE_ALERTING_PATTERN_7 - Pattern 7 – 0x07 - QMI_VOICE_ALERTING_PATTERN_8 - Pattern 8 – 0x08 - QMI_VOICE_ALERTING_PATTERN_9 - Pattern 9

8.45.2 Field Documentation

8.45.2.1 **ULONG** arrAlertingPattern::alertingPattern[20]

8.45.2.2 **BYTE** arrAlertingPattern::callID[20]

8.45.2.3 **BYTE** arrAlertingPattern::numInstances

8.46 arrAlertingType Struct Reference

Data Fields

- [BYTE](#) numInstances
- [BYTE](#) callID [20]
- [BYTE](#) AlertingType [20]

8.46.1 Detailed Description

This structure contains an array of Alerting Type.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of callID, AlertingType that follow. • If zero(0) then no further information exists.
<i>callID</i> [<i>MAX_NO_OF_CALLS</i>]	<ul style="list-style-type: none"> • Array of Unique call identifier for the call.
<i>AlertingType</i> [<i>MAX_NO_OF_CALLS</i>]	<ul style="list-style-type: none"> • Array of Alerting type. <ul style="list-style-type: none"> – 0x00 - ALERTING_LOCAL - Local – 0x01 - ALERTING_REMOTE - Remote

8.46.2 Field Documentation

8.46.2.1 **BYTE** arrAlertingType::AlertingType[20]8.46.2.2 **BYTE** arrAlertingType::callID[20]8.46.2.3 **BYTE** arrAlertingType::numInstances

8.47 arrAlphaID Struct Reference

Data Fields

- [BYTE](#) numInstances
- [allCallsAlphaIDInfo](#) allCallsAlphaIDInfoArr [20]

8.47.1 Detailed Description

This structure contains an array of Alpha ID Info

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of allCallsAlphaIDInfo that follow. • If zero(0) then no further information exists.
<i>allCallsAlphaIDInfo</i> [<i>MAX_NO_OF_CALLS</i>]	<ul style="list-style-type: none"> • Array of allCallsAlphaIDInfo. • See allCallsAlphaIDInfo for more information.

8.47.2 Field Documentation

8.47.2.1 **allCallsAlphaIDInfo** arrAlphaID::allCallsAlphaIDInfoArr[20]

8.47.2.2 **BYTE** arrAlphaID::numInstances

8.48 arrCalledPartyNum Struct Reference

Data Fields

- [BYTE](#) numInstances
- [peerNumberInfo](#) [CalledPartyNum](#) [20]

8.48.1 Detailed Description

This structure contains an array of Called Party Numbers consisting of information of all the numbers which have been called from the device.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of calledPartyNum that follow. • If zero(0) then no further information exists.
<i>CalledPartyNum</i> ↔ <i>Num</i> [<i>MAX_NUM_OF_CALLS</i>]	<ul style="list-style-type: none"> • Array of CalledPartyNum. • See peerNumberInfo for more information.

8.48.2 Field Documentation

8.48.2.1 **peerNumberInfo** arrCalledPartyNum::CalledPartyNum[20]8.48.2.2 **BYTE** arrCalledPartyNum::numInstances

8.49 arrCallEndReason Struct Reference

Data Fields

- [BYTE](#) numInstances
- [BYTE](#) callID [20]
- [WORD](#) callEndReason [20]

8.49.1 Detailed Description

This structure contains an array of Call End Reasons.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of callID, callEndReason that follow. • If zero(0) then no further information exists.
<i>callID</i> [MAX_NO_OF_CALLS]	<ul style="list-style-type: none"> • Array of Unique call identifier for the call.
<i>callEndReason</i> [MAX_NO_OF_CALLS]	<ul style="list-style-type: none"> • Array of Call End Reason . • See Table9 qaGobiApiTableVoiceCallEndReasons.h for a list of valid voice-related call end reasons

8.49.2 Field Documentation

8.49.2.1 WORD arrCallEndReason::callEndReason[20]

8.49.2.2 BYTE arrCallEndReason::callID[20]

8.49.2.3 BYTE arrCallEndReason::numInstances

8.50 arrCallInfo Struct Reference

Data Fields

- [BYTE numInstances](#)
- [getAllCallInformation getAllCallInfo](#) [20]

8.50.1 Detailed Description

This structure contains an array of Call Info

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of getAllCallInfo that follow. • If zero(0) then no further information exists.
<i>getAllCallInfo</i> [MAX_NO_OF_CALLS]	<ul style="list-style-type: none"> • Array of CallInfo. • See getAllCallInfo for more information.

8.50.2 Field Documentation

8.50.2.1 getAllCallInformation arrCallInfo::getAllCallInfo[20]

8.50.2.2 BYTE arrCallInfo::numInstances

8.51 arrConnectPartyNum Struct Reference

Data Fields

- [BYTE numInstances](#)
- [peerNumberInfo ConnectedPartyNum](#) [20]

8.51.1 Detailed Description

This structure contains an array of Connected Party Numbers consisting of information regarding all the devices connected.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of ConnectedPartyNum that follow. • If zero(0) then no further information exists.
<i>ConnectedPartyNum</i> [MAX_NO_OF_CALLS]	<ul style="list-style-type: none"> • Array of ConnectedPartyNum. • See peerNumberInfo for more information.

8.51.2 Field Documentation

8.51.2.1 peerNumberInfo arrConnectPartyNum::ConnectedPartyNum[20]

8.51.2.2 BYTE arrConnectPartyNum::numInstances

8.52 arrDiagInfo Struct Reference

Data Fields

- [BYTE numInstances](#)
- [allCallsDiagInfo DiagInfo](#) [20]

8.52.1 Detailed Description

This structure contains an array of Diagnostic Information.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of DiagInfo that follow. • If zero(0) then no further information exists.
<i>DiagInfo</i> [MAX_NO_OF_CALLS]	<ul style="list-style-type: none"> • Array of DiagInfo. • See allCallsDiagInfo for more information.

8.52.2 Field Documentation

8.52.2.1 `allCallsDiagInfo arrDiagInfo::DiagInfo[20]`

8.52.2.2 `BYTE arrDiagInfo::numInstances`

8.53 arrRedirPartyNum Struct Reference

Data Fields

- [BYTE numInstances](#)
- [peerNumberInfo RedirPartyNum](#) [20]

8.53.1 Detailed Description

This structure contains an array of Redirecting Party Numbers consisting of information of all the numbers which have been redirected from the device.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of redirPartyNum that follow. • If zero(0) then no further information exists.
<i>RedirPartyNum</i> <i>Num</i> [MAX_NUM_OF_CALLS]	<ul style="list-style-type: none"> • Array of RedirPartyNum. • See peerNumberInfo for more information.

8.53.2 Field Documentation

8.53.2.1 `BYTE arrRedirPartyNum::numInstances`

8.53.2.2 `peerNumberInfo arrRedirPartyNum::RedirPartyNum[20]`

8.54 arrRemotePartyName Struct Reference

Data Fields

- [BYTE numInstances](#)
- [getAllCallRmtPtyName GetAllCallRmtPtyName](#) [20]

8.54.1 Detailed Description

This structure contains an array of Remote Party Names

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of remotePartyName that follow. • If zero(0) then no further information exists.
<i>GetAllCallRmtPtyName[MAX_NO_OF_CALLS]</i>	<ul style="list-style-type: none"> • Array of remotePartyName. • See getAllCallRmtPtyName for more information.

8.54.2 Field Documentation

8.54.2.1 `getAllCallRmtPtyName arrRemotePartyName::GetAllCallRmtPtyName[20]`

8.54.2.2 `BYTE arrRemotePartyName::numInstances`

8.55 arrRemotePartyNum Struct Reference

Data Fields

- [BYTE numInstances](#)
- [getAllCallRmtPtyNum RmtPtyNum \[20\]](#)

8.55.1 Detailed Description

This structure contains an array of Remote Party Numbers

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of remotePartyNum that follow. • If zero(0) then no further information exists.
<i>RmtPtyNum[MAX_NO_OF_CALLS]</i>	<ul style="list-style-type: none"> • Array of remotePartyNum. • See getAllCallRmtPtyNum for more information.

8.55.2 Field Documentation

8.55.2.1 `BYTE arrRemotePartyNum::numInstances`

8.55.2.2 `getAllCallRmtPtyNum arrRemotePartyNum::RmtPtyNum[20]`

8.56 arrSvcOption Struct Reference

Data Fields

- [BYTE numInstances](#)
- [BYTE callID](#) [20]
- [WORD srvOption](#) [20]

8.56.1 Detailed Description

This structure contains array an of Servicing option.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of callID, srvOption that follow. • If zero(0) then no further information exists.
<i>callID</i> [<i>MAX_NUM_OF_CALLS</i>]	<ul style="list-style-type: none"> • Array of Unique call identifier for the call.
<i>srvOption</i> [<i>MAX_NUM_OF_CALLS</i>]	<ul style="list-style-type: none"> • Array of Service option. • See Table9 qaGobiApiTableServiceOptions.h for standard service option number assignments.

8.56.2 Field Documentation

8.56.2.1 **BYTE** arrSvcOption::callID[20]

8.56.2.2 **BYTE** arrSvcOption::numInstances

8.56.2.3 **WORD** arrSvcOption::srvOption[20]

8.57 arrUUSInfo Struct Reference

Data Fields

- [BYTE numInstances](#)
- [allCallsUUSInfo AllCallsUUSInfo](#) [20]

8.57.1 Detailed Description

This structure contains an array of User to User Signaling Service Information

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of allCallsUUSInfo that follow. • If zero(0) then no further information exists.
Generated by Doxygen	
<i>AllCallsUUSInfo</i> [<i>MAX_NUM_OF_CALLS</i>]	<ul style="list-style-type: none"> • Array of allCallsUUSInfo. • See allCallsUUSInfo for more information.

8.57.2 Field Documentation

8.57.2.1 `allCallsUUSInfo arrUUSInfo::AllCallsUUSInfo[20]`

8.57.2.2 `BYTE arrUUSInfo::numInstances`

8.58 authenticateResult Struct Reference

Data Fields

- [WORD contentLen](#)
- [BYTE content](#) [1024]

8.58.1 Detailed Description

This structure contains the information about the authenticate result.

Parameters

<i>contentLen</i>	<ul style="list-style-type: none"> • Length of the following elements i.e. content.
<i>content</i> [<i>MAX_</i> ↔ <i>CONTENT_L</i> ↔ <i>ENGTH</i>]	<ul style="list-style-type: none"> • Authenticate data. • This value is a sequence of bytes returned from the card.

8.58.2 Field Documentation

8.58.2.1 `BYTE authenticateResult::content[1024]`

8.58.2.2 `WORD authenticateResult::contentLen`

8.59 authenticationData Struct Reference

Data Fields

- [BYTE context](#)
- [WORD dataLen](#)
- [BYTE data](#) [1024]

8.59.1 Detailed Description

This structure contains the Session Information.

Parameters

<i>context</i>	<ul style="list-style-type: none"> Authenticate context. <ul style="list-style-type: none"> 0 - Runs the GSM algorithm (valid only on a 2GSIM card) 1 - Runs the CAVE algorithm (valid only on a RUIM card) 2 - GSM security context (valid only on a USIM application) 3 - 3G security context (valid only on a USIM application) 4 - VGCS/VBS security context (valid only on aUSIM application) 5 - GBA security context, Bootstrapping mode (valid only on a USIM or ISIM application) 6 - GBA security context, NAF Derivation mode (valid only on a USIM or ISIM application) 7 - MBMS security context, MSK Update mode (valid only on a USIM application) 8 - MBMS security context, MTK Generation mode (valid only on a USIM application) 9 - MBMS security context, MSK Deletion mode (valid only on a USIM application) 10 - MBMS security context, MUK Deletion mode (valid only on a USIM application) 11 - IMS AKA security context (valid only on aISIM application) 12 - HTTP-digest security context (valid only onan ISIM application) 13 - Compute IP authentication, CHAP (valid onlyon RUIM or CSIM) 14 - Compute IP authentication, MN-HA authenticator (valid only on RUIM or CSIM) 15 - Compute IP authentication, MIP-RRQ hash (valid only on RUIM or CSIM) 16 - Compute IP authentication, MN-AAA authenticator (valid only on RUIM or CSIM) 17 - Compute IP authentication, HRPD access authenticator (valid only on RUIM or CSIM) Other values are possible and reserved for future use.
<i>dataLen</i>	<ul style="list-style-type: none"> Length of the following elements i.e. data.
<i>data</i> [MAX_DE↔ SCRIPTION_L↔ ENGTH]	<ul style="list-style-type: none"> Authenticate Data.

8.59.2 Field Documentation

8.59.2.1 **BYTE** authenticationData::context

8.59.2.2 **BYTE** authenticationData::data[1024]

8.59.2.3 **WORD** authenticationData::dataLen

8.60 BandCapabilityResp Struct Reference

Data Fields

- [ULONGLONG bandCapability](#)
- [ULONGLONG * pLteBandCapability](#)
- [ULONGLONG * pTdsBandCapability](#)

8.60.1 Detailed Description

This structure contains the TLV required to Get Band Capability.

Parameters

<i>band</i> ↔ <i>Capability[OUT]</i>	Bitmask of bands supported by the device <ul style="list-style-type: none"> • Bit 0 - Band class 0, A-system • Bit 1 - Band class 0, B-system • Bit 2 - Band class 1, all blocks • Bit 3 - Band class 2 • Bit 4 - Band class 3, A-system • Bit 5 - Band class 4, all blocks • Bit 6 - Band class 5, all blocks • Bit 7 - GSM DCS band (1800) • Bit 8 - GSM Extended GSM (E-GSM) band (900) • Bit 9 - GSM Primary GSM (P-GSM) band (900) • Bit 10 - Band class 6 • Bit 11 - Band class 7 • Bit 12 - Band class 8 • Bit 13 - Band class 9 • Bit 14 - Band class 10 • Bit 15 - Band class 11 • Bit 16 - GSM 450 band • Bit 17 - GSM 480 band • Bit 18 - GSM 750 band • Bit 19 - GSM 850 band • Bit 20 - GSM railways GSM band (900) • Bit 21 - GSM PCS band (1900) • Bit 22 - WCDMA (Europe, Japan, and China) 2100 band • Bit 23 - WCDMA US PCS 1900 band • Bit 24 - WCDMA (Europe and China) DCS 1800 band • Bit 25 - WCDMA US 1700 band • Bit 26 - WCDMA US 850 band • Bit 27 - WCDMA Japan 800 band • Bit 28 - Band class 12 • Bit 29 - Band class 14 • Bit 30 - Reserved • Bit 31 - Band class 15 • Bits 32 through 47 - Reserved • Bit 48 - WCDMA Europe 2600 band • Bit 49 - WCDMA Europe and Japan 900 band • Bit 50 - WCDMA Japan 1700 band • Bits 51 through 55 - Reserved • Bit 56 - Band class 16 • Bit 57 - Band class 17 • Bit 58 - Band class 18 • Bit 59 - Band class 19

8.60.2 Field Documentation

8.60.2.1 ULONGLONG BandCapabilityResp::bandCapability

8.60.2.2 **ULONGLONG*** BandCapabilityResp::pLteBandCapability

8.60.2.3 **ULONGLONG*** BandCapabilityResp::pTdsBandCapability

8.61 BdsSV Struct Reference

Data Fields

- [WORD](#) *id*
- [BYTE](#) *mask*

8.61.1 Detailed Description

This structure contains the BDS [SV](#) Info

Parameters

<i>id</i>	<ul style="list-style-type: none">• SV ID of the satellite whose data is to be deleted.<ul style="list-style-type: none">– Range for BDS: 201 to 237
<i>mask</i>	<ul style="list-style-type: none">• Indicates if the ephemeris or almanac for a satellite is to be deleted• Valid values:<ul style="list-style-type: none">– QMI_LOC_MASK_DELETE_EPHEMERIS (0x01) - Delete ephemeris for the satellite– QMI_LOC_MASK_DELETE_ALMANAC (0x02) - Delete almanac for the satellite

8.61.2 Field Documentation

8.61.2.1 **WORD** BdsSV::id

8.61.2.2 **BYTE** BdsSV::mask

8.62 BdsSVInfo Struct Reference

Data Fields

- [BYTE](#) *len*
- [BdsSV](#) * *pSV*

8.62.1 Detailed Description

This structure contains the number of sets of the BDS SVN Info

Parameters

<i>len</i>	<ul style="list-style-type: none"> Number of sets of the following elements: <ul style="list-style-type: none"> gnssSvId deleteSvInfoMask
<i>pSV</i>	<ul style="list-style-type: none"> Pointer to struct BdsSV. See BdsSV for more information

8.62.2 Field Documentation

8.62.2.1 BYTE BdsSVInfo::len

8.62.2.2 BdsSV* BdsSVInfo::pSV

8.63 BroadcastConfig Struct Reference

Data Fields

- [WORD fromServiceId](#)
- [WORD toServiceId](#)
- [BYTE selected](#)

8.63.1 Detailed Description

This structure contains [BroadcastConfig](#) parameters

Parameters

<i>fromServiceId</i>	<ul style="list-style-type: none"> Starting point of range of CBM message identifiers
<i>toServiceId</i>	<ul style="list-style-type: none"> Ending point of range of CBM message identifiers
<i>selected</i>	<ul style="list-style-type: none"> Range of CBM message identifiers indicated by from_service_id and to_service_id <ul style="list-style-type: none"> 0x00 – Not selected 0x01 – Selected

8.63.2 Field Documentation

8.63.2.1 WORD BroadcastConfig::fromServiceId

8.63.2.2 BYTE BroadcastConfig::selected

8.63.2.3 WORD BroadcastConfig::toServiceId

8.64 burstDTMFInfo Struct Reference

Data Fields

- [BYTE](#) * pCallID
- [BYTE](#) digitCnt
- [BYTE](#) pDigitBuff [255]

8.64.1 Detailed Description

This structure contains Voice Burst DTMF Information

Parameters

<i>pCallID</i> [IN/OUT]	<ul style="list-style-type: none"> • Call ID associated with call on which the DTMF information has to be sent. A burst DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device
<i>digitCnt</i>	<ul style="list-style-type: none"> • Length of DTMF digit buffer which follows
<i>pDigitBuff</i> [MAX_DESCRIPTOR_LENGTH]	<ul style="list-style-type: none"> • DTMF digit buffer in ASCII, NULL terminated

8.64.2 Field Documentation

8.64.2.1 BYTE burstDTMFInfo::digitCnt

8.64.2.2 BYTE* burstDTMFInfo::pCallID

8.64.2.3 BYTE burstDTMFInfo::pDigitBuff[255]

8.65 CallBarringSysInfo Struct Reference

Data Fields

- [ULONG](#) csBarStatus
- [ULONG](#) psBarStatus

8.65.1 Detailed Description

Structure for storing the GSM and WCDMA Call Barring System Information.

Parameters

<i>csBarStatus</i>	<ul style="list-style-type: none"> • Call barring status for circuit-switched calls. <ul style="list-style-type: none"> – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown
<i>psBarStatus</i>	<ul style="list-style-type: none"> • Call barring status for packet-switched calls. <ul style="list-style-type: none"> – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown

8.65.2 Field Documentation

8.65.2.1 **ULONG** CallBarringSysInfo::csBarStatus

8.65.2.2 **ULONG** CallBarringSysInfo::psBarStatus

8.66 callBarStatus Struct Reference

Data Fields

- [ULONG csBarStatus](#)
- [ULONG psBarStatus](#)

8.66.1 Detailed Description

This structure contains Call Barring Status.

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>csBarStatus</i>	<ul style="list-style-type: none"> • Call Barring Status for circuit-switched calls. • Values: <ul style="list-style-type: none"> • NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only • NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only • NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type • NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types • NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown
<i>psBarStatus</i>	<ul style="list-style-type: none"> • Call Barring Status for packet-switched calls. • Values: <ul style="list-style-type: none"> – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown

8.66.2 Field Documentation

8.66.2.1 ULONG callBarStatus::csBarStatus

8.66.2.2 ULONG callBarStatus::psBarStatus

8.67 calledPartyInfo Struct Reference

Data Fields

- [BYTE PI](#)
- [BYTE SI](#)
- [BYTE numType](#)
- [BYTE numPlan](#)
- [BYTE numLen](#)
- [BYTE number](#) [255]

8.67.1 Detailed Description

This structure contains Called party Number Information

Parameters

<i>PI</i>	<ul style="list-style-type: none"> • Presentation indicator; refer to [S1, Table 2.7.4.4-1] for valid values.
<i>SI</i>	<ul style="list-style-type: none"> • Number of sets of following elements <ul style="list-style-type: none"> – Caller Id
<i>SI</i>	<ul style="list-style-type: none"> • Number screening indicator. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network
<i>numType</i>	<ul style="list-style-type: none"> • Number type. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension
<i>numPlan</i>	<ul style="list-style-type: none"> • Number plan. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension
<i>numLen</i>	<ul style="list-style-type: none"> • Provides the length of number which follow.
<i>number[255]</i>	<ul style="list-style-type: none"> • number of numLen length, NULL terminated.

8.67.2 Field Documentation

8.67.2.1 BYTE calledPartyInfo::number[255]

8.67.2.2 BYTE calledPartyInfo::numLen

8.67.2.3 BYTE calledPartyInfo::numPlan

8.67.2.4 BYTE calledPartyInfo::numType

8.67.2.5 BYTE calledPartyInfo::PI

8.67.2.6 BYTE calledPartyInfo::SI

8.68 calledPartySubAdd Struct Reference

Data Fields

- [BYTE extBit](#)
- [BYTE subAddrType](#)
- [BYTE oddEvenInd](#)
- [BYTE subAddrLen](#)
- [BYTE subAddr](#) [255]

8.68.1 Detailed Description

This structure contains information about the Called Sub Party Addresses.

Parameters

<i>extBit</i>	<ul style="list-style-type: none">• Extension bit.
<i>subAddrType</i>	<ul style="list-style-type: none">• Subaddress type.<ul style="list-style-type: none">– 0x00 - NSAP– 0x01 - USER
<i>oddEvenInd</i>	<ul style="list-style-type: none">• Even/odd indicator.<ul style="list-style-type: none">– 0x00 - Even number of address signals– 0x01 - Odd number of address signals
<i>subAddrLen</i>	<ul style="list-style-type: none">• Number of sets of the following elements:<ul style="list-style-type: none">– SubAddress
<i>subAddr</i> [MAX← _DESCRIPTI← ON_LENGTH]	<ul style="list-style-type: none">• Array of the SubAddress in BCD number format.

8.68.2 Field Documentation

8.68.2.1 **BYTE** calledPartySubAdd::extBit

8.68.2.2 **BYTE** calledPartySubAdd::oddEvenInd

8.68.2.3 **BYTE** calledPartySubAdd::subAddr[255]

8.68.2.4 **BYTE** calledPartySubAdd::subAddrLen

8.68.2.5 **BYTE** calledPartySubAdd::subAddrType

8.69 callerIDInfo Struct Reference

Data Fields

- [BYTE PI](#)
- [BYTE callerIDLen](#)
- [BYTE callerID \[255\]](#)

8.69.1 Detailed Description

This structure contains Caller ID Information

Parameters

<i>PI</i>	<ul style="list-style-type: none"> • Presentation indicator; refer to [S1, Table 2.7.4.4-1] for valid values.
<i>callerIDLen</i>	<ul style="list-style-type: none"> • Number of sets of following elements <ul style="list-style-type: none"> – Caller Id
<i>pCallerID</i>	<ul style="list-style-type: none"> • Caller ID in ASCII string.

8.69.2 Field Documentation

8.69.2.1 **BYTE** callerIDInfo::callerID[255]

8.69.2.2 **BYTE** callerIDInfo::callerIDLen

8.69.2.3 **BYTE** callerIDInfo::PI

8.70 callFwdTypeAndPlan Struct Reference

Data Fields

- [BYTE numberType](#)
- [BYTE numberPlan](#)

8.70.1 Detailed Description

This structure contains Supplementary Service request parameters related to different features and their activation, deactivation, registration and erasure (applicable only for 3GPP)

Parameters

<i>numberType</i>	<ul style="list-style-type: none"> • Call forwarding number type <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION Reserved extension
<i>numberPlan</i>	<ul style="list-style-type: none"> • Call forwarding number plan <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION Reserved extension

8.70.2 Field Documentation

8.70.2.1 BYTE callFwdTypeAndPlan::numberPlan

8.70.2.2 BYTE callFwdTypeAndPlan::numberType

8.71 callFWExtInfo Struct Reference

Data Fields

- [BYTE SvcStatus](#)
- [BYTE SvcClass](#)
- [BYTE noReplyTimer](#)

- [BYTE PI](#)
- [BYTE SI](#)
- [BYTE numType](#)
- [BYTE numPlan](#)
- [BYTE numLen](#)
- [BYTE number](#) [255]

8.71.1 Detailed Description

This structure contains information for Get Call Forwarding Extended Information.

Parameters

<i>SvcStatus</i>	<ul style="list-style-type: none"> • Service status. Values: <ul style="list-style-type: none"> – 0x00 - SERVICE_STATUS_INACTIVE - Inactive – 0x01 - SERVICE_STATUS_ACTIVE - Active
<i>SvcClass</i>	<ul style="list-style-type: none"> • Service Class is a combination (sum) of information class constants • See qaGobiApiTableSupServiceInfoClasses.h for service classes.
<i>noReplyTimer</i>	<ul style="list-style-type: none"> • No reply timer value in seconds • A value of 0 indicates that no_reply_timer is ignored.
<i>PI</i>	<ul style="list-style-type: none"> • Presentation indicator; refer to [S1, Table 2.7.4.4-1] for valid values.
<i>SI</i>	<ul style="list-style-type: none"> • Number screening indicator. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network
<i>numType</i>	<ul style="list-style-type: none"> • Number type. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension

<i>numPlan</i>	<ul style="list-style-type: none"> • Number plan. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension
<i>numLen</i>	<ul style="list-style-type: none"> • Provides the length of number which follow.
<i>number[255]</i>	<ul style="list-style-type: none"> • number of numLen length, NULL terminated.

8.71.2 Field Documentation

8.71.2.1 **BYTE** callFWExtInfo::noReplyTimer

8.71.2.2 **BYTE** callFWExtInfo::number[255]

8.71.2.3 **BYTE** callFWExtInfo::numLen

8.71.2.4 **BYTE** callFWExtInfo::numPlan

8.71.2.5 **BYTE** callFWExtInfo::numType

8.71.2.6 **BYTE** callFWExtInfo::PI

8.71.2.7 **BYTE** callFWExtInfo::SI

8.71.2.8 **BYTE** callFWExtInfo::SvcClass

8.71.2.9 **BYTE** callFWExtInfo::SvcStatus

8.72 callFWInfo Struct Reference

Data Fields

- [BYTE SvcStatus](#)
- [BYTE SvcClass](#)
- [BYTE numLen](#)
- [BYTE number \[255\]](#)
- [BYTE noReplyTimer](#)

8.72.1 Detailed Description

This structure contains information for Get Call Forwarding Information.

Parameters

<i>SvcStatus</i>	<ul style="list-style-type: none"> Service status. Values: <ul style="list-style-type: none"> 0x00 - SERVICE_STATUS_INACTIVE - Inactive 0x01 - SERVICE_STATUS_ACTIVE - Active
<i>SvcClass</i>	<ul style="list-style-type: none"> Service Class is a combination (sum) of information class constants See qaGobiApiTableSupServiceInfoClasses.h for service classes.
<i>numLen</i>	<ul style="list-style-type: none"> Provides the length of number which follow.
<i>number[255]</i>	<ul style="list-style-type: none"> number of numLen length, NULL terminated.
<i>noReplyTimer</i>	<ul style="list-style-type: none"> No reply timer value in seconds A value of 0 indicates that noReplyTimer is ignored.

8.72.2 Field Documentation

8.72.2.1 **BYTE** callFWInfo::noReplyTimer

8.72.2.2 **BYTE** callFWInfo::number[255]

8.72.2.3 **BYTE** callFWInfo::numLen

8.72.2.4 **BYTE** callFWInfo::SvcClass

8.72.2.5 **BYTE** callFWInfo::SvcStatus

8.73 callInfo Struct Reference

Data Fields

- [BYTE](#) callIID
- [BYTE](#) callState
- [BYTE](#) callType
- [BYTE](#) direction
- [BYTE](#) mode

8.73.1 Detailed Description

This structure contains Information about call state changes. For example, when an incoming call is received, this structure is populated and indicate the incoming call information. When this incoming call is answered, the call status changes from INCOMING to CONVERSATION, which means a change in the call information and this structure is populated again with the changes and notified to/retrived by the user.

Parameters

<i>callID</i>	<ul style="list-style-type: none"> • Call identifier for the call queried for information. • If zero(0) then invalid.
<i>callState</i>	<ul style="list-style-type: none"> • Call state. <ul style="list-style-type: none"> – 0x01 - CALL_STATE_ORIGINATION - Origination – 0x02 - CALL_STATE_INCOMING - Incoming – 0x03 - CALL_STATE_CONVERSATION - Conversation – 0x04 - CALL_STATE_CC_IN_PROGRESS - Call is originating but waiting for call control to complete – 0x05 - CALL_STATE_ALERTING - Alerting – 0x06 - CALL_STATE_HOLD - Hold – 0x07 - CALL_STATE_WAITING - Waiting – 0x08 - CALL_STATE_DISCONNECTING - Disconnecting – 0x09 - CALL_STATE_END - End – 0x0A - CALL_STATE_SETUP - MT call is in Setup state in 3GPP – 0xFF - Not Available
<i>callType</i>	<ul style="list-style-type: none"> • Call type. <ul style="list-style-type: none"> – 0x00 - CALL_TYPE_VOICE - Voice – 0x02 - CALL_TYPE_VOICE_IP - Voice over IP – 0x06 - CALL_TYPE_OTAPA - OTAPA – 0x07 - CALL_TYPE_STD_OTASP - Standard OTASP – 0x08 - CALL_TYPE_NON_STD_OTASP - Nonstandard OTASP – 0x09 - CALL_TYPE_EMERGENCY - Emergency – 0xFF - Not Available
<i>direction</i>	<ul style="list-style-type: none"> • Direction. <ul style="list-style-type: none"> – 0x01 - CALL_DIRECTION_MO - MO call – 0x02 - CALL_DIRECTION_MT - MT call – 0xFF - Not Available

<i>mode</i>	<ul style="list-style-type: none"> • Mode. • If the mode field is "0x01 - CDMA", the optional Service Option, Voice Privacy, and OTASP Status (only for OTASP calls) TLVs are included in the response. <ul style="list-style-type: none"> – 0x01 - CALL_MODE_CDMA - CDMA – 0x02 - CALL_MODE_GSM - GSM – 0x03 - CALL_MODE_UMTS - UMTS – 0x04 - CALL_MODE_LTE - LTE – 0x05 - CALL_MODE_TDS - TD-SCDMA – 0xFF - Not Available
-------------	--

8.73.2 Field Documentation

8.73.2.1 BYTE callInfo::callID

8.73.2.2 BYTE callInfo::callState

8.73.2.3 BYTE callInfo::callType

8.73.2.4 BYTE callInfo::direction

8.73.2.5 BYTE callInfo::mode

8.74 callingPartyInfo Struct Reference

Data Fields

- [BYTE PI](#)
- [BYTE SI](#)
- [BYTE numType](#)
- [BYTE numPlan](#)
- [BYTE numLen](#)
- [BYTE number \[255\]](#)

8.74.1 Detailed Description

This structure contains Calling party Number Information

Parameters

<i>PI</i>	<ul style="list-style-type: none"> • Presentation indicator; refer to [S1, Table 2.7.4.4-1] for valid values.
<i>SI</i>	<ul style="list-style-type: none"> • Number of sets of following elements <ul style="list-style-type: none"> – Caller Id
<i>SI</i>	<ul style="list-style-type: none"> • Number screening indicator. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network
<i>numType</i>	<ul style="list-style-type: none"> • Number type. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension
<i>numPlan</i>	<ul style="list-style-type: none"> • Number plan. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension
<i>numLen</i>	<ul style="list-style-type: none"> • Provides the length of number which follow.
<i>number[255]</i>	<ul style="list-style-type: none"> • number of numLen length, NULL terminated.

8.74.2 Field Documentation

8.74.2.1 **BYTE** callingPartyInfo::number[255]

8.74.2.2 **BYTE** callingPartyInfo::numLen

8.74.2.3 **BYTE** callingPartyInfo::numPlan

8.74.2.4 **BYTE** callingPartyInfo::numType

8.74.2.5 **BYTE** callingPartyInfo::PI

8.74.2.6 **BYTE** callingPartyInfo::SI

8.75 cardResult Struct Reference

Data Fields

- [BYTE sw1](#)
- [BYTE sw2](#)

8.75.1 Detailed Description

This structure contains the information about the card result.

Parameters

<i>sw1</i>	<ul style="list-style-type: none">• SW1 received from the card.
<i>sw2</i>	<ul style="list-style-type: none">• SW2 received from the card.

8.75.2 Field Documentation

8.75.2.1 **BYTE** cardResult::sw1

8.75.2.2 **BYTE** cardResult::sw2

8.76 cardStatus Struct Reference

Data Fields

- [WORD indexGwPri](#)
- [WORD index1xPri](#)
- [WORD indexGwSec](#)
- [WORD index1xSec](#)
- [BYTE numSlot](#)
- [slotInfo SlotInfo](#) [5]

8.76.1 Detailed Description

This structure contains Card Status Information.

Parameters

<i>indexGwPri</i>	<ul style="list-style-type: none"> • Index of the primary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
<i>index1xPri</i>	<ul style="list-style-type: none"> • Index of the primary 1X provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
<i>indexGwSec</i>	<ul style="list-style-type: none"> • Index of the secondary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
<i>index1xSec</i>	<ul style="list-style-type: none"> • Index of the secondary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
<i>numSlot</i>	<ul style="list-style-type: none"> • Indicates the number of slots available on the device. • The following block is repeated for each slot. i.e. cardState • If zero(0) then no cardState information exists.
<i>SlotInfo</i> [MAX_↔ NO_OF_SLO↔ TS]	<ul style="list-style-type: none"> • See slotInfo for more information.

8.76.2 Field Documentation

8.76.2.1 WORD cardStatus::index1xPri

8.76.2.2 WORD cardStatus::index1xSec

8.76.2.3 WORD cardStatus::indexGwPri

8.76.2.4 WORD cardStatus::indexGwSec

8.76.2.5 BYTE cardStatus::numSlot

8.76.2.6 slotInfo cardStatus::SlotInfo[5]

8.77 CarrierImage_t Struct Reference

Data Fields

- uint32_t [m_nCarrierId](#)
- uint32_t [m_nFolderId](#)
- uint32_t [m_nStorage](#)
- uint8_t [m_FwImageId](#) [16]
- uint8_t [m_FwBuildId](#) [32]
- uint8_t [m_PriImageId](#) [16]
- uint8_t [m_PriBuildId](#) [32]

8.77.1 Detailed Description

This structure contains the Carrier Image parameters.

Parameters

<i>m_nCarrierId</i>	<ul style="list-style-type: none"> • Unique numeric carrier ID indicating the carrier that the following images belong to
<i>m_nFolderId</i>	<ul style="list-style-type: none"> • Unique numeric folder ID indicating the folder where the images should reside on the host storage.
<i>m_nStorage</i>	<ul style="list-style-type: none"> • Information of storage type • Values <ul style="list-style-type: none"> – 0 - Device – 1 - Host
<i>m_FwImageId</i>	<ul style="list-style-type: none"> • Firmware image ID
<i>m_FwBuildId</i>	<ul style="list-style-type: none"> • Firmware build ID
<i>m_PriImageId</i>	<ul style="list-style-type: none"> • PRI image ID
<i>m_PriBuildId</i>	<ul style="list-style-type: none"> • PRI build ID

8.77.2 Field Documentation

8.77.2.1 uint8_t CarrierImage_t::m_FwBuildId[32]

8.77.2.2 `uint8_t CarrierImage_t::m_FwImageld[16]`

8.77.2.3 `uint32_t CarrierImage_t::m_nCarrierId`

8.77.2.4 `uint32_t CarrierImage_t::m_nFolderId`

8.77.2.5 `uint32_t CarrierImage_t::m_nStorage`

8.77.2.6 `uint8_t CarrierImage_t::m_PriBuildId[32]`

8.77.2.7 `uint8_t CarrierImage_t::m_PrImageld[16]`

8.78 CatAlPhalIdentifierTlv Struct Reference

Data Fields

- [BYTE ReferenceID](#)
- [USHORT AlphaDLength](#)
- [BYTE AlphaID \[255\]](#)

8.78.1 Detailed Description

structure used to store all Alpha Identifier parameters.

Parameters

<i>ReferenceID</i>	- proactive command type that included the alpha identifier – 0x01; sends SMS proactive command
<i>AlphaDLength</i>	- length of AlphaID (in bytes)
<i>AlphaID</i>	- alpha identifier, encoded as in ETSI TS 102 223 [Section 8.2]

8.78.2 Field Documentation

8.78.2.1 `BYTE CatAlPhalIdentifierTlv::AlphaID[255]`

8.78.2.2 `USHORT CatAlPhalIdentifierTlv::AlphaDLength`

8.78.2.3 `BYTE CatAlPhalIdentifierTlv::ReferenceID`

8.79 CatCommonEventTlv Struct Reference

Data Fields

- [BYTE TlvPresent](#)
- [BYTE EventID](#)
- [WORD EventLength](#)
- union [currentCatEvent](#) [CatEvent](#)

8.79.1 Field Documentation

8.79.1.1 union `currentCatEvent` `CatCommonEventTlv::CatEvent`

8.79.1.2 **BYTE** `CatCommonEventTlv::EventID`

8.79.1.3 **WORD** `CatCommonEventTlv::EventLength`

8.79.1.4 **BYTE** `CatCommonEventTlv::TlvPresent`

8.80 CatEndProactiveSessionTlv Struct Reference

Data Fields

- [BYTE](#) `EndProactiveSession`

8.80.1 Detailed Description

structure used to store End Proactive Session event parameters.

Parameters

<i>EndProactiveSession</i>	- The proactive session end type values are: <ul style="list-style-type: none"> • 0x01 – End proactive session command type received from the card • 0x02 – End proactive session internal to ME
----------------------------	--

8.80.2 Field Documentation

8.80.2.1 **BYTE** `CatEndProactiveSessionTlv::EndProactiveSession`

8.81 CATEventDataType Struct Reference

Data Fields

- [ULONG](#) `eventMask`
- [ULONG *](#) `pErrorMask`

8.81.1 Field Documentation

8.81.1.1 **ULONG** `CATEventDataType::eventMask`

8.81.1.2 **ULONG*** `CATEventDataType::pErrorMask`

8.82 CatEventIDDataTlv Struct Reference

Data Fields

- [ULONG ReferenceID](#)
- [USHORT DataLength](#)
- [BYTE Data](#) [255]

8.82.1 Detailed Description

structure used to store all Common CAT Event parameters.

Parameters

<i>ReferenceID</i>	- proactive command reference ID.
<i>DataLength</i>	- length of pData (in Bytes)
<i>Data</i>	- command specific to the CAT event ID, encoded as in ETSI TS 102 223 [Section 6.6.X]

8.82.2 Field Documentation

8.82.2.1 **BYTE** CatEventIDDataTlv::Data[255]

8.82.2.2 **USHORT** CatEventIDDataTlv::DataLength

8.82.2.3 **ULONG** CatEventIDDataTlv::ReferenceID

8.83 CatEventListTlv Struct Reference

Data Fields

- [ULONG SetupEventList](#)

8.83.1 Detailed Description

structure used to store all Event List parameters.

Parameters

<i>SetupEventList</i>	- Setup event list bit mask <ul style="list-style-type: none">• 0x00000001 – User Activity Notify• 0x00000002 – Idle Screen Available• 0x00000004 – Lang Selection Notify Each set bit indicates the availability of the corresponding event in Setup Event list proactive command; all unlisted bits are reserved for future use and will be ignored
-----------------------	---

8.83.2 Field Documentation

8.83.2.1 ULONG CatEventListTlv::SetupEventList

8.84 CatRefreshTlv Struct Reference

Data Fields

- [USHORT RefreshMode](#)
- [BYTE RefreshStage](#)

8.84.1 Detailed Description

structure used to store all Refresh Event parameters.

Parameters

<i>RefreshMode</i>	- The Refresh Event as in ETSI TS 102 223 [Section 8.6]
<i>RefreshStage</i>	- Stage of a refresh procedure <ul style="list-style-type: none">• 0x01 – Refresh start• 0x02 – Refresh success• 0x03 – Refresh failed

8.84.2 Field Documentation

8.84.2.1 USHORT CatRefreshTlv::RefreshMode

8.84.2.2 BYTE CatRefreshTlv::RefreshStage

8.85 ccSUPSType Struct Reference

Data Fields

- [BYTE svcType](#)
- [BYTE reason](#)

8.85.1 Detailed Description

This structure contains information about the Call Control Supplementary Service Types

Parameters

<i>svcType</i>	<ul style="list-style-type: none"> Service type. <ul style="list-style-type: none"> 0x01 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_ACTIVATE - Activate 0x02 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_DEACTIVATE - Deactivate 0x03 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_REGISTER - Register 0x04 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_ERASE - Erase 0x05 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_INTERROGATE - Interrogate 0x06 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_REGISTER_PASSWORD - Register password 0x07 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_USSD - USSD 0xFF - Not Available
<i>reason</i>	<ul style="list-style-type: none"> Call control supplementary service result reason Values: <ul style="list-style-type: none"> See qaGobiApiTableCallControlReturnReasons.h for return reasons.

8.85.2 Field Documentation

8.85.2.1 BYTE ccSUPSType::reason

8.85.2.2 BYTE ccSUPSType::svcType

8.86 CDMABroadcastConfig Struct Reference

Data Fields

- [WORD serviceCategory](#)
- [WORD language](#)
- [BYTE selected](#)

8.86.1 Detailed Description

This structure contains [CDMABroadcastConfig](#) parameters

Parameters

<i>serviceCategory</i>	<ul style="list-style-type: none"> Service category
<i>language</i>	<ul style="list-style-type: none"> Language
<i>selected</i>	<ul style="list-style-type: none"> Specified service_category and language <ul style="list-style-type: none"> 0x00 – Not selected 0x01 – Selected

8.86.2 Field Documentation

8.86.2.1 WORD CDMABroadcastConfig::language

8.86.2.2 BYTE CDMABroadcastConfig::selected

8.86.2.3 WORD CDMABroadcastConfig::serviceCategory

8.87 CDMAChannel Struct Reference

Data Fields

- [WORD priChA](#)
- [WORD priChB](#)
- [WORD secChA](#)
- [WORD secChB](#)

8.87.1 Detailed Description

This structure contains the parameters for CDMA Channel Information

Parameters

<i>priChA</i>	<ul style="list-style-type: none"> • A Channel number for the primary carrier. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>priChB</i>	<ul style="list-style-type: none"> • B Channel number for the primary carrier. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>secChA</i>	<ul style="list-style-type: none"> • A Channel number for the secondary carrier. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>secChB</i>	<ul style="list-style-type: none"> • B Channel number for the secondary carrier. <ul style="list-style-type: none"> – 0xFFFF - Not Available

8.87.2 Field Documentation

8.87.2.1 WORD CDMAChannel::priChA

8.87.2.2 WORD CDMAChannel::priChB

8.87.2.3 WORD CDMAChannel::secChA

8.87.2.4 WORD CDMAChannel::secChB

8.88 CDMAECIOThresh Struct Reference

Data Fields

- [BYTE CDMAECIOThreshListLen](#)
- [WORD * pCDMAECIOThreshList](#)

8.88.1 Detailed Description

This structure contains CDMA ECIO threshold related parameters.

Parameters

<i>CDMAECIO</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none">• Length of the CDMA ECIO threshold list parameter to follow
<i>pCDMAECIO</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none">• Array of ECIO thresholds (in units of 0.1 dB)• Maximum of 32 values. Range for ECIO values: -31.5 to 0 (in dB)

8.88.2 Field Documentation

8.88.2.1 BYTE CDMAECIOThresh::CDMAECIOThreshListLen

8.88.2.2 WORD* CDMAECIOThresh::pCDMAECIOThreshList

8.89 CDMAInfo Struct Reference

Data Fields

- [WORD sid](#)
- [WORD nid](#)
- [WORD baselId](#)
- [WORD refpn](#)
- [ULONG baseLat](#)
- [ULONG baseLong](#)

8.89.1 Detailed Description

This structure contains information about the CDMA Network.

Parameters

<i>sid</i>	<ul style="list-style-type: none"> • System ID. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>nid</i>	<ul style="list-style-type: none"> • Network ID. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>baseId</i>	<ul style="list-style-type: none"> • Base station ID. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>refpn</i>	<ul style="list-style-type: none"> • Reference PN. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>baseLat</i>	<ul style="list-style-type: none"> • Latitude of the current base station in units of 0.25 sec. <ul style="list-style-type: none"> – 0xFFFFFFFF - Not Available
<i>baseLong</i>	<ul style="list-style-type: none"> • Longitude of the current base station in units of 0.25 sec. <ul style="list-style-type: none"> – 0xFFFFFFFF - Not Available

8.89.2 Field Documentation

8.89.2.1 WORD CDMAInfo::baseId

8.89.2.2 ULONG CDMAInfo::baseLat

8.89.2.3 ULONG CDMAInfo::baseLong

8.89.2.4 WORD CDMAInfo::nid

8.89.2.5 WORD CDMAInfo::refpn

8.89.2.6 WORD CDMAInfo::sid

8.90 cdmaMsgDecodingParams Struct Reference

Data Fields

- ULONG [messageLength](#)
- BYTE * [pMessage](#)
- ULONG * [pMessageID](#)
- BYTE * [pSenderAddrLength](#)
- CHAR * [pSenderAddr](#)

- BYTE * pTextMsgLength
- WORD * pTextMsg
- BYTE * pPriority
- BYTE * pPrivacy
- BYTE * pLanguage
- BYTE mcTimeStamp [0x08]
- BYTE absoluteValidity [0x08]
- BYTE * pRelativeValidity
- BYTE * pDisplayMode
- BOOL * pUserAcknowledgementReq
- BOOL * pReadAcknowledgementReq
- BYTE * pAlertPriority
- BYTE * pCallbkAddrLength
- CHAR * pCallbkAddr

8.90.1 Detailed Description

Structure contains parameters which need to be decoded from message

Parameters

<i>message</i> ↔ <i>Length</i> [IN]	<ul style="list-style-type: none"> Length of the message to be decoded in bytes
<i>pMessage</i> [IN]	<ul style="list-style-type: none"> Message read off the device via GetSMS
<i>pSenderAddr</i> ↔ <i>Length</i> [IN/OUT]	<ul style="list-style-type: none"> Upon input, indicates the maximum number of ASCII characters (including NULL termination) that the pSenderAddr buffer can accommodate. Note that a length of 14 is reasonable. Upon successful output, returns the length of originating address string (including the NULL termination)
<i>pSenderAddr</i> ↔ <i>OUT</i>	<ul style="list-style-type: none"> Returns NULL-terminated ASCII String containing the originating address. International number will be prepended with a '+' character
<i>pTextMsg</i> ↔ <i>Length</i> [IN/OUT]	<ul style="list-style-type: none"> Upon input, specifies the number of UCS2 characters the given text message buffer can accommodate. Upon successful output, returns the number of UCS2 characters returns in the given text messagebuffer(including NULL-terminator)
<i>pTextMsg</i> [OUT]	<ul style="list-style-type: none"> Returns the text message as NULL-terminated UCS2 string
<i>pPriority</i> [OUT]	(optional parameter) <ul style="list-style-type: none"> Returns the priority setting of the message 0x00 - normal 0x01 - interactive 0x02 - urgent 0x03 - emergency 0xFF - unavailable setting
<i>pPrivacy</i> [OUT↔ <i>T</i>](optional	parameter) <ul style="list-style-type: none"> Returns the privacy setting of the message 0x00 - not restricted 0x01 - restricted 0x02 - confidential 0x03 - secret 0xFF - unavailable setting

<i>pLanguage</i> [O↔ UT]	(optional parameter) <ul style="list-style-type: none"> Returns the language setting of the message 0x00 - unspecified 0x01 - english 0x02 - french 0x03 - spanish 0x04 - japanese 0x05 - korean 0x06 - chinese 0x07 - hebrew 0xFF - unavailable setting
<i>mcTime</i> ↔ <i>Stamp</i> [8][OUT]	(optional parameter) <ul style="list-style-type: none"> Returns the message center timestamp which takes the form: YYMMDDHHMMSSTZ where YY - year MM - month DD - day HH - hour MM - minute SS - second TZ - timezone All values are in decimal. Timezone is in relation to GMT, one unit is equal to 15 minutes and MSB indicates a negative value.If this information is unavailable for message then this field will be filled with 0xFF
<i>absolute</i> ↔ <i>Validity</i> [8][OUT]	(optional parameter) <ul style="list-style-type: none"> Returns the absolute validity period setting for this message.This field takes the same form as mcTimeStamp
<i>pRelative</i> ↔ <i>Validity</i> [OUT]	(optional parameter) <ul style="list-style-type: none"> Returns the relative validity period.Values have the following meanings: 0 to 143: validity period =(value + 1)* 5 minutes 144 to 167: validity period =12 hours+(value - 143)*30 minutes 168 to 196: validity period = (value - 166) * 1 day 197 to 244: validity period = (value - 192) * 1 week 245: validity period = indefinite 246: validity period = immediate 247: validity period = valid until mobile becomes inactive 248: validity period = valid until registration area changes 249 to 254: reserved 255: unavailable information
<i>pDisplayMode</i> ↔ OUT]	(optional parameter) <ul style="list-style-type: none"> Returns the display mode parameter 0x00 - immediate display 0x01 - mobile default setting 0x02 - user invoked 0x03 - reserved 0xFF - unavailable parameter
<i>pUser</i> ↔ <i>Acknowledgement</i> <i>Req</i> [OUT]	(optional parameter) <ul style="list-style-type: none"> Returns the user (manual) acknowledgment request parameter TRUE - means the user is requested to manually acknowledge the delivery of the message. FALSE - means no such user acknowledgement is requested
<i>pRead</i> ↔ <i>Acknowledgement</i> <i>Req</i> [OUT]	(optional parameter) <ul style="list-style-type: none"> Returns the read acknowledgement request parameter TRUE - means acknowledgment of the message being viewed is requested. FALSE - means no such read acknowledgement is requested
<i>pAlertPriority</i> ↔ OUT]	(optional parameter) <ul style="list-style-type: none"> Returns the alerting parameter setting 0x00 - use default alert 0x01 - use low priority alert 0x02 - use medium priority alert 0x03 - use high priority alert 0xFF - unavailable parameter
<i>pCallbkAddr</i> ↔ <i>Length</i> [OUT]	(optional parameter) <ul style="list-style-type: none"> returns the length of Callback address string (including the NULL termination)
<i>pCallbkAddr</i> [O↔ UT]	(optional parameter) <ul style="list-style-type: none"> returns NULL-terminated ASCII String containing callback address String containing the Call Back number with a 32 maximum characters.

8.90.2 Field Documentation

8.90.2.1 BYTE cdmaMsgDecodingParams::absoluteValidity[0x08]

- 8.90.2.2 **BYTE** cdmaMsgDecodingParams::mcTimeStamp[0x08]
- 8.90.2.3 **ULONG** cdmaMsgDecodingParams::messageLength
- 8.90.2.4 **BYTE*** cdmaMsgDecodingParams::pAlertPriority
- 8.90.2.5 **CHAR*** cdmaMsgDecodingParams::pCallbkAddr
- 8.90.2.6 **BYTE*** cdmaMsgDecodingParams::pCallbkAddrLength
- 8.90.2.7 **BYTE*** cdmaMsgDecodingParams::pDisplayMode
- 8.90.2.8 **BYTE*** cdmaMsgDecodingParams::pLanguage
- 8.90.2.9 **BYTE*** cdmaMsgDecodingParams::pMessage
- 8.90.2.10 **ULONG*** cdmaMsgDecodingParams::pMessageld
- 8.90.2.11 **BYTE*** cdmaMsgDecodingParams::pPriority
- 8.90.2.12 **BYTE*** cdmaMsgDecodingParams::pPrivacy
- 8.90.2.13 **BOOL*** cdmaMsgDecodingParams::pReadAcknowledgementReq
- 8.90.2.14 **BYTE*** cdmaMsgDecodingParams::pRelativeValidity
- 8.90.2.15 **CHAR*** cdmaMsgDecodingParams::pSenderAddr
- 8.90.2.16 **BYTE*** cdmaMsgDecodingParams::pSenderAddrLength
- 8.90.2.17 **WORD*** cdmaMsgDecodingParams::pTextMsg
- 8.90.2.18 **BYTE*** cdmaMsgDecodingParams::pTextMsgLength
- 8.90.2.19 **BOOL*** cdmaMsgDecodingParams::pUserAcknowledgementReq

8.91 cdmaMsgEncodingParams Struct Reference

Data Fields

- **BYTE *** pMessageSize
- **BYTE *** pMessage
- **BYTE** messageld
- **CHAR *** pDestAddr
- **CHAR *** pCallbackAddr
- **ULONG** textMsgLength
- **WORD *** pTextMsg
- **BYTE *** pPriority
- **BYTE *** pEncodingAlphabet
- **BYTE *** pRelValidity

8.91.1 Detailed Description

Structure contains parameters for message to be encoded

Parameters

<i>pMessageSize</i> [IN/OUT]	<ul style="list-style-type: none"> Upon input, specifies the total number of bytes that the given pMessage buffer can hold (a buffer of length 240 is recommended). Upon successful output, specifies the length of the constructed message placed in the pMessage buffer (in bytes)
<i>pMessage</i> [OUT]	- The constructed raw message
<i>messageId</i> [IN]	<ul style="list-style-type: none"> The message reference number for this message. This value should be incremented for every message the host application sends
<i>pDestAddr</i> [IN]	<ul style="list-style-type: none"> Gives NULL-terminated ASCII String containing a destination address. International number will be prepended with a '+' character
<i>pCallbackAddr</i> [IN]	<ul style="list-style-type: none"> Gives NULL-terminated ASCII String containing a callback address. International number will be prepended with a '+' character
<i>textMsgLength</i> [IN]	<ul style="list-style-type: none"> Number of UCS2 characters in the text message(excluding NULL)
<i>pTextMsg</i> [IN]	<ul style="list-style-type: none"> Text message to be encoded
<i>pPriority</i> [IN/OUT](optional)	parameter) <ul style="list-style-type: none"> Gives the priority of the outgoing message: 0 - normal (default if NULL pointer is given) 1 - interactive 2 - urgent 3 - emergency 64 - 64 is decoded value for URGENT VZAM Support interactive. 128 - 128 is decoded value for URGENT VZAM Support urgent. 192 - 128 is decoded value for URGENT VZAM Support emergency.
<i>pEncodingAlphabet</i> [IN/OUT/UT](optional)	parameter) <ul style="list-style-type: none"> Upon input, specifies the alphabet the text message should be encoded in 0 - 8bit ASCII (not supported at this time) 1 - IS91EP (not supported at this time) 2 - 7bit ASCII (default if NULL pointer is given) 3 - IA5 (not supported at this time) 4 - unicode (not supported at this time) 5 - shift JIS (not supported at this time) 6 - korean (not supported at this time) 7 - latin hebrew (not supported at this time) 8 - latin (not supported at this time) 9 - GSM 7 bit default Upon successful output, specifies the alphabet used to encode the message.
<i>pRelValidity</i> [IN/OUT/UT](optional)	parameter) <ul style="list-style-type: none"> Gives the relative validity period of the outgoing message 0 - Set Relative validity to 11 1 - Set Relative validity to 71 2 - Set Relative validity to 167 3 - Set Relative validity to 169 4 - Set Relative validity to 171 Values have the following meanings: 0 to 143: validity period = (value + 1)* 5 minutes 144 to 167: validity period = 12 hours + (value - 143)*30 minutes 168 to 196: validity period = (value - 166) * 1 day 197 to 244: validity period = (value - 192) * 1 week 245: validity period = indefinite

Currently only encoding of 7bit ASCII messages is supported.

8.91.2 Field Documentation

8.91.2.1 BYTE cdmaMsgEncodingParams::messageId

8.91.2.2 **CHAR*** cdmaMsgEncodingParams::pCallbackAddr

8.91.2.3 **CHAR*** cdmaMsgEncodingParams::pDestAddr

8.91.2.4 **BYTE*** cdmaMsgEncodingParams::pEncodingAlphabet

8.91.2.5 **BYTE*** cdmaMsgEncodingParams::pMessage

8.91.2.6 **BYTE*** cdmaMsgEncodingParams::pMessageSize

8.91.2.7 **BYTE*** cdmaMsgEncodingParams::pPriority

8.91.2.8 **BYTE*** cdmaMsgEncodingParams::pRelValidity

8.91.2.9 **WORD*** cdmaMsgEncodingParams::pTextMsg

8.91.2.10 **ULONG** cdmaMsgEncodingParams::textMsgLength

8.92 CDMARSSIThresh Struct Reference

Data Fields

- [BYTE](#) CDMARSSIThreshListLen
- [WORD](#) * pCDMARSSIThreshList

8.92.1 Detailed Description

This structure contains CDMA RSSI threshold related parameters.

Parameters

<i>CDMARSSI</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> • Length of the CDMARSSI threshold list parameter to follow
<i>pCDMARSSI</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> • Array of RSSI thresholds (in units of 0.1 dBm) • maximum of 32 values. • Range for RSSI values:-105 to -21 (in dBm).

8.92.2 Field Documentation

8.92.2.1 **BYTE** CDMARSSIThresh::CDMARSSIThreshListLen

8.92.2.2 **WORD*** CDMARSSIThresh::pCDMARSSIThreshList

8.93 CDMASSInfo Struct Reference

Data Fields

- [INT8](#) *rss*
- [SHORT](#) *ecio*

8.93.1 Detailed Description

This structure contains the parameters for CDMA/WCDMA Signal Strength Information

Parameters

<i>rss</i>	<ul style="list-style-type: none">• RSSI in dBm (signed value).• A value of -125 dBm or lower is used to indicate No Signal.
<i>ecio</i>	<ul style="list-style-type: none">• ECIO value representing negative 0.5 dBm increments, i.e., 2 means -1 dBm (14 means -7 dBm, 63 means -31.5 dBm).

8.93.2 Field Documentation

8.93.2.1 SHORT CDMASSInfo::ecio

8.93.2.2 INT8 CDMASSInfo::rss

8.94 cdmaSSInfo Struct Reference

Data Fields

- [int8_t](#) *rss*
- [int16_t](#) *ecio*

8.94.1 Detailed Description

Parameters

<i>rss</i>	RSSI in dBm.
<i>ecio</i>	ECIO value representing negative 0.5 dBm increment

8.94.2 Field Documentation

8.94.2.1 int16_t cdmaSSInfo::ecio

8.94.2.2 int8_t cdmaSSInfo::rssi

8.95 CDMA SysInfo Struct Reference

Data Fields

- [sysInfoCommon](#) [sysInfoCDMA](#)
- [BYTE](#) [isSysPrIMatchValid](#)
- [BYTE](#) [isSysPrIMatch](#)
- [BYTE](#) [pRevInUseValid](#)
- [BYTE](#) [pRevInUse](#)
- [BYTE](#) [bsPRevValid](#)
- [BYTE](#) [bsPRev](#)
- [BYTE](#) [ccsSupportedValid](#)
- [BYTE](#) [ccsSupported](#)
- [BYTE](#) [cdmaSysIdValid](#)
- [WORD](#) [systemID](#)
- [WORD](#) [networkID](#)
- [BYTE](#) [bsInfoValid](#)
- [WORD](#) [baseId](#)
- [ULONG](#) [baseLat](#)
- [ULONG](#) [baseLong](#)
- [BYTE](#) [packetZoneValid](#)
- [WORD](#) [packetZone](#)
- [BYTE](#) [networkIdValid](#)
- [BYTE](#) [MCC](#) [3]
- [BYTE](#) [MNC](#) [3]

8.95.1 Detailed Description

Structure for storing the CDMA System Information.

Parameters

<i>sysInfoCDMA</i>	<ul style="list-style-type: none"> • See sysInfoCommon for more information.
<i>isSysPrIMatch</i> ↔ <i>Valid</i>	<ul style="list-style-type: none"> • Indicates whether the system PRL match is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>isSysPrIMatch</i>	<ul style="list-style-type: none"> • Indicates whether the system is in a PRL. • Only applies to CDMA/HDR. <ul style="list-style-type: none"> – 0x00 - System is not in a PRL – 0x01 - System is in a PRL – 0xFF - Not Available • If the system is not in a PRL, <code>roam_status</code> carries the value from the default roaming indicator in the PRL. • If the system is in a PRL, <code>roam_status</code> is set to the value based on the standard specification.
Generated by Doxygen	

<i>pRevInUseValid</i>	<ul style="list-style-type: none"> Indicates whether the P_Rev in use is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>pRevInUse</i>	<ul style="list-style-type: none"> The lesser of the base station P_Rev and mobile P_Rev Only applicable for CDMA. <ul style="list-style-type: none"> 0xFF - Not Available
<i>bsPRevValid</i>	<ul style="list-style-type: none"> Indicates whether the base station P_Rev is valid <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>bsPRev</i>	<ul style="list-style-type: none"> Base station P_Rev. Only applicable for CDMA. <ul style="list-style-type: none"> 0xFF - Not Available
<i>ccsSupportedValid</i>	<ul style="list-style-type: none"> Indicates whether the supported concurrent service is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>ccsSupported</i>	<ul style="list-style-type: none"> Whether concurrent service is supported. Only applicable for CDMA. <ul style="list-style-type: none"> 0x00 - Not supported 0x01 - Supported 0xFF - Not Available
<i>cdmaSysIdValid</i>	<ul style="list-style-type: none"> Indicates whether the CDMA system ID is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>systemID</i>	<ul style="list-style-type: none"> System ID. <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>networkID</i>	<ul style="list-style-type: none"> Network ID. <ul style="list-style-type: none"> 0xFFFF - Not Available

<i>bsInfoValid</i>	<ul style="list-style-type: none"> Indicates whether the base station information is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>baseLat</i>	<ul style="list-style-type: none"> Base station latitude in units of 0.25 sec. Expressed as a two's complement signed number with positive numbers signifying North latitudes. <ul style="list-style-type: none"> 0xFFFFFFFF - Not Available
<i>baseLong</i>	<ul style="list-style-type: none"> Base station longitude in units of 0.25 sec. Expressed as a two's complement signed number with positive numbers signifying East latitudes. <ul style="list-style-type: none"> 0xFFFFFFFF - Not Available
<i>packetZoneValid</i>	<ul style="list-style-type: none"> Indicates whether the packet zone is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>packetZone</i>	<ul style="list-style-type: none"> Packet zone (8-bit). <ul style="list-style-type: none"> 0xFFFF indicates no packet zone. Only applicable for CDMA.
<i>networkIdValid</i>	<ul style="list-style-type: none"> Indicates whether the network ID is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>MCC[PLMN_LEN-1:0]</i>	<ul style="list-style-type: none"> Mobile Country Code. MCC digits in ASCII characters MCC wildcard value is returned as {'3', 0xFF, 0xFF}.
<i>MNC[PLMN_LEN-1:0]</i>	<ul style="list-style-type: none"> Mobile Network Code. MNC digits in ASCII characters An unused byte is set to 0xFF. MNC wildcard value is returned as {'7', 0xFF, 0xFF}.

8.95.2 Field Documentation

8.95.2.1 WORD CDMA SysInfo::baseId

- 8.95.2.2 **ULONG** CDMA SysInfo::baseLat
- 8.95.2.3 **ULONG** CDMA SysInfo::baseLong
- 8.95.2.4 **BYTE** CDMA SysInfo::bsInfoValid
- 8.95.2.5 **BYTE** CDMA SysInfo::bsPRev
- 8.95.2.6 **BYTE** CDMA SysInfo::bsPRevValid
- 8.95.2.7 **BYTE** CDMA SysInfo::ccsSupported
- 8.95.2.8 **BYTE** CDMA SysInfo::ccsSupportedValid
- 8.95.2.9 **BYTE** CDMA SysInfo::cdmaSysIdValid
- 8.95.2.10 **BYTE** CDMA SysInfo::isSysPrIMatch
- 8.95.2.11 **BYTE** CDMA SysInfo::isSysPrIMatchValid
- 8.95.2.12 **BYTE** CDMA SysInfo::MCC[3]
- 8.95.2.13 **BYTE** CDMA SysInfo::MNC[3]
- 8.95.2.14 **WORD** CDMA SysInfo::networkID
- 8.95.2.15 **BYTE** CDMA SysInfo::networkIdValid
- 8.95.2.16 **WORD** CDMA SysInfo::packetZone
- 8.95.2.17 **BYTE** CDMA SysInfo::packetZoneValid
- 8.95.2.18 **BYTE** CDMA SysInfo::pRevInUse
- 8.95.2.19 **BYTE** CDMA SysInfo::pRevInUseValid
- 8.95.2.20 **sysInfoCommon** CDMA SysInfo::sysInfoCDMA
- 8.95.2.21 **WORD** CDMA SysInfo::systemID

8.96 CDMA SysInfoExt Struct Reference

Data Fields

- [WORD MCC](#)
- [BYTE imsi_11_12](#)

8.96.1 Detailed Description

This structure contains CDMA system information extension

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>MCC</i>	<ul style="list-style-type: none">• Mobile Country Code
<i>imsi_11_12</i>	<ul style="list-style-type: none">• IMSI_11_12

8.96.2 Field Documentation

8.96.2.1 BYTE CDMA SysInfoExt::imsi_11_12

8.96.2.2 WORD CDMA SysInfoExt::MCC

8.97 CellDb Struct Reference

Data Fields

- [ULONG mask](#)

8.97.1 Detailed Description

This structure contains the cell database

Parameters

<i>mask</i>	<ul style="list-style-type: none">• Mask for the cell database assistance data that is to be deleted• Valid values:<ul style="list-style-type: none">– 0x00000001 - DELETE_CELLDB_POS– 0x00000002 - DELETE_CELLDB_LATEST_GPS_POS– 0x00000004 - DELETE_CELLDB_OTA_POS– 0x00000008 - DELETE_CELLDB_EXT_REF_POS– 0x00000010 - DELETE_CELLDB_TIMETAG– 0x00000020 - DELETE_CELLDB_CELLID– 0x00000040 - DELETE_CELLDB_CACHED_CELLID– 0x00000080 - DELETE_CELLDB_LAST_SRV_CELL– 0x00000100 - DELETE_CELLDB_CUR_SRV_CELL– 0x00000200 - DELETE_CELLDB_NEIGHBOR_INFO
-------------	--

8.97.2 Field Documentation

8.97.2.1 ULONG CellDb::mask

8.98 cellParams Struct Reference

Data Fields

- [WORD pci](#)
- [SHORT rsrq](#)
- [SHORT rsrp](#)
- [SHORT rssi](#)
- [SHORT srxlev](#)

8.98.1 Detailed Description

This structure contains information about the Cell parameters.

Parameters

<i>pci</i>	<ul style="list-style-type: none"> • Physical cell ID. • Range: 0 to 503.
<i>rsrq</i>	<ul style="list-style-type: none"> • Current RSRQ in 1/10 dB as measured by L1. • Range: -20.0 dB to -3.0 dB.
<i>rsrp</i>	<ul style="list-style-type: none"> • Current RSRP in 1/10 dBm as measured by L1. • Range: -140.0 dBm to -44.0 dBm.
<i>rssi</i>	<ul style="list-style-type: none"> • Current RSSI in 1/10 dBm as measured by L1. • Range: -120.0 dBm to 0.
<i>srxlev</i>	<ul style="list-style-type: none"> • Cell selection Rx level (Srxlev) value. • Range: -128 to 128. • This field is only valid when ue_in_idle is TRUE.

8.98.2 Field Documentation

8.98.2.1 WORD cellParams::pci

8.98.2.2 SHORT cellParams::rsrp

8.98.2.3 SHORT cellParams::rsrq

8.98.2.4 SHORT cellParams::rssi

8.98.2.5 SHORT cellParams::srxlev

8.99 changeUIMPIN Struct Reference

Data Fields

- [BYTE pinID](#)
- [BYTE oldPINLen](#)
- [BYTE oldPINVal](#) [255]
- [BYTE pinLen](#)
- [BYTE pinVal](#) [255]

8.99.1 Detailed Description

This structure contains the information about the pin parameters that need to be verified.

Parameters

<i>pinID</i>	<ul style="list-style-type: none"> Indicates the PIN ID to be changed. <ul style="list-style-type: none"> 1 - PIN1 (also called PIN) 2 - PIN2 3 - Universal PIN 4 - Hidden key
<i>oldPINLen</i>	<ul style="list-style-type: none"> Length of the following elements i.e. old pin value.
<i>oldPINVal</i> [<i>MAX_DESCRIPTOR_LENGTH</i>]	<ul style="list-style-type: none"> Old PIN value. This value is a sequence of ASCII characters.
<i>pinLen</i>	<ul style="list-style-type: none"> Length of the following elements i.e. new pin value.
<i>pinVal</i> [<i>MAX_DESCRIPTOR_LENGTH</i>]	<ul style="list-style-type: none"> New PIN value. This value is a sequence of ASCII characters.

8.99.2 Field Documentation

8.99.2.1 BYTE changeUIMPIN::oldPINLen

8.99.2.2 BYTE changeUIMPIN::oldPINVal[255]

8.99.2.3 BYTE changeUIMPIN::pinID

8.99.2.4 BYTE changeUIMPIN::pinLen

8.99.2.5 **BYTE** changeUIMPIN::pinVal[255]

8.100 ChannelRate Struct Reference

Data Fields

- [ULONG CurrChanTxRate](#)
- [ULONG CurrChanRxRate](#)
- [ULONG MaxChanTxRate](#)
- [ULONG MaxChanRxRate](#)

8.100.1 Detailed Description

This structure contains Channel Rate

Parameters

<i>CurrChanTxRate</i>	<ul style="list-style-type: none"> • Instantaneous channel Tx rate in bits per second
<i>CurrChanRx↔ Rate</i>	<ul style="list-style-type: none"> • Instantaneous channel Rx rate in bits per second
<i>MaxChanTxRate</i>	<ul style="list-style-type: none"> • maximum Tx rate that can be assigned to the device by the serving system in bits per second
<i>MaxChanRx↔ Rate</i>	<ul style="list-style-type: none"> • maximum Rx rate that can be assigned to the device by the serving system in bits per second

8.100.2 Field Documentation

8.100.2.1 **ULONG** ChannelRate::CurrChanRxRate

8.100.2.2 **ULONG** ChannelRate::CurrChanTxRate

8.100.2.3 **ULONG** ChannelRate::MaxChanRxRate

8.100.2.4 **ULONG** ChannelRate::MaxChanTxRate

8.101 channelRate Struct Reference

Data Fields

- [ULONG CurrChanTxRate](#)
- [ULONG CurrChanRxRate](#)

8.101.1 Detailed Description

This structure contains Channel Rate

Parameters

<i>CurrChanTxRate</i>	<ul style="list-style-type: none">• Max channel Tx rate in bits per second
<i>CurrChanRx↔ Rate</i>	<ul style="list-style-type: none">• Max channel Rx rate in bits per second

8.101.2 Field Documentation

8.101.2.1 **ULONG** channelRate::CurrChanRxRate

8.101.2.2 **ULONG** channelRate::CurrChanTxRate

8.102 CLIPResp Struct Reference

Data Fields

- [BYTE](#) *ActiveStatus*
- [BYTE](#) *ProvisionStatus*

8.102.1 Detailed Description

This structure contains information about the Calling Line Identification Presentation (CLIP) supplementary service responses.

Parameters

<i>ActiveStatus</i>	<ul style="list-style-type: none">• Active status.• Values:<ul style="list-style-type: none">– 0x00 - ACTIVE_STATUS_INACTIVE - Inactive– 0x01 - ACTIVE_STATUS_ACTIVE - Active– 0xFF - Not Available
<i>ProvisionStatus</i>	<ul style="list-style-type: none">• Provisioned status.• Values:<ul style="list-style-type: none">– 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned– 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned– 0xFF - Not Available

8.102.2 Field Documentation

8.102.2.1 BYTE CLIPResp::ActiveStatus

8.102.2.2 BYTE CLIPResp::ProvisionStatus

8.103 CLIRResp Struct Reference

Data Fields

- [BYTE ActiveStatus](#)
- [BYTE ProvisionStatus](#)

8.103.1 Detailed Description

This structure contains information about the Calling Line Identification Restriction (CLIR) supplementary service responses.

Parameters

<i>ActiveStatus</i>	<ul style="list-style-type: none"> • Active status. • Values: <ul style="list-style-type: none"> – 0x00 - ACTIVE_STATUS_INACTIVE - Inactive – 0x01 - ACTIVE_STATUS_ACTIVE - Active – 0xFF - Not Available
<i>ProvisionStatus</i>	<ul style="list-style-type: none"> • Provisioned status. • Values: <ul style="list-style-type: none"> – 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned – 0x01 - PROVISION_STATUS_PROVISIONED_PERMANENT - Permanently provisioned – 0x02 - PROVISION_STATUS_PRESENTATION_RESTRICTED - Restricted presentation – 0x03 - PROVISION_STATUS_PRESENTATION_ALLOWED - Allowed presentation – 0xFF - Not Available

8.103.2 Field Documentation

8.103.2.1 BYTE CLIRResp::ActiveStatus

8.103.2.2 BYTE CLIRResp::ProvisionStatus

8.104 ClkInfo Struct Reference

Data Fields

- [ULONG mask](#)

8.104.1 Detailed Description

This structure contains the clock Info

Parameters

<i>mask</i>	<ul style="list-style-type: none"> • Mask for the clock information assistance data that is to be deleted • Valid bitmasks: <ul style="list-style-type: none"> – QMI_LOC_MASK_DELETE_CLOCK_INFO_TIME_EST (0x00000001) - Mask to delete time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_FREQ_EST (0x00000002) - Mask to delete frequency estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_WEEK_NUMBER (0x00000004) - Mask to delete week number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_RTC_TIME (0x00000008) - Mask to delete RTC time from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_TIME_TRANSFER (0x00000010) - Mask to delete time transfer from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GPSTIME_EST (0x00000020) - Mask to delete GPS time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLOTIME_EST (0x00000040) - Mask to delete GLONASS time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLODAY_NUMBER (0x00000080) - Mask to delete GLONASS day number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLO4YEAR_NUMBER (0x00000100) - Mask to delete GLONASS four year number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLO_RF_GRP_DELAY (0x00000200) - Mask to delete GLONASS RF GRP delay from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_DISABLE_TT (0x00000400) - Mask to delete disable TT from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GG_LEAPSEC (0x00000800) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GG_GGTB (0x00001000) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDSTIME_EST (0x00002000) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GB_GBTB (0x00004000) - Mask to delete Glonass-to-BDS time bias-related information from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BG_BGTB (0x00008000) - Mask to delete BDS-to-GLONASS time bias-related information from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDSWEEK_NUMBER (0x00010000) - Mask to delete the BDS week number from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDS_RF_GRP_DELAY (0x00020000) - Mask to delete the BDS RF GRP delay from the clock information
-------------	--

8.104.2 Field Documentation

8.104.2.1 ULONG CkInfo::mask

8.105 CNAPResp Struct Reference

Data Fields

- [BYTE ActiveStatus](#)
- [BYTE ProvisionStatus](#)

8.105.1 Detailed Description

This structure contains information about the Calling Name Presentation (CNAP) supplementary service responses.

Parameters

<i>ActiveStatus</i>	<ul style="list-style-type: none">• Active status.• Values:<ul style="list-style-type: none">– 0x00 - ACTIVE_STATUS_INACTIVE - Inactive– 0x01 - ACTIVE_STATUS_ACTIVE - Active– 0xFF - Not Available
<i>ProvisionStatus</i>	<ul style="list-style-type: none">• Provisioned status.• Values:<ul style="list-style-type: none">– 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned– 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned– 0xFF - Not Available

8.105.2 Field Documentation

8.105.2.1 BYTE CNAPResp::ActiveStatus

8.105.2.2 BYTE CNAPResp::ProvisionStatus

8.106 COLPResp Struct Reference

Data Fields

- [BYTE ActiveStatus](#)
- [BYTE ProvisionStatus](#)

8.106.1 Detailed Description

This structure contains information about the Connected Line Identification Presentation (COLP) supplementary service responses.

Parameters

<i>ActiveStatus</i>	<ul style="list-style-type: none">• Active status.• Values:<ul style="list-style-type: none">– 0x00 - ACTIVE_STATUS_INACTIVE - Inactive– 0x01 - ACTIVE_STATUS_ACTIVE - Active– 0xFF - Not Available
<i>ProvisionStatus</i>	<ul style="list-style-type: none">• Provisioned status.• Values:<ul style="list-style-type: none">– 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned– 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned– 0xFF - Not Available

8.106.2 Field Documentation

8.106.2.1 BYTE COLPResp::ActiveStatus

8.106.2.2 BYTE COLPResp::ProvisionStatus

8.107 COLRResp Struct Reference

Data Fields

- [BYTE ActiveStatus](#)
- [BYTE ProvisionStatus](#)

8.107.1 Detailed Description

This structure contains information about the Connected Line Identification Restriction (COLR) supplementary service responses.

Parameters

<i>ActiveStatus</i>	<ul style="list-style-type: none">• Active status.• Values:<ul style="list-style-type: none">– 0x00 - ACTIVE_STATUS_INACTIVE - Inactive– 0x01 - ACTIVE_STATUS_ACTIVE - Active– 0xFF - Not Available
<i>ProvisionStatus</i>	<ul style="list-style-type: none">• Provisioned status.• Values:<ul style="list-style-type: none">– 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned– 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned– 0xFF - Not Available
Generated by Doxygen	

8.107.2 Field Documentation

8.107.2.1 BYTE COLRResp::ActiveStatus

8.107.2.2 BYTE COLRResp::ProvisionStatus

8.108 CommInfo Struct Reference

Data Fields

- [BYTE temperature](#)
- [BYTE modemMode](#)
- [BYTE systemMode](#)
- [BYTE imsRegState](#)
- [BYTE psState](#)

8.108.1 Detailed Description

Structure for storing the common information for the device.

Parameters

<i>temperature</i>	<ul style="list-style-type: none">• Temperature.<ul style="list-style-type: none">– 8-bit signed integer– 0xFF - Not Available.
<i>modemMode</i>	<ul style="list-style-type: none">• Modem Operating Mode.<ul style="list-style-type: none">– 0x00 - POWERING OFF– 0x01 - FACTORY TEST– 0x02 - OFFLINE– 0x03 - OFFLINE_AMPS– 0x04 - OFFLINE_CDMA– 0x05 - ONLINE– 0x06 - LOW POWER MODE– 0x07 - RESETTING– 0x08 - NETWORK TEST– 0x09 - OFFLINE REQUEST– 0x0A - PSEUDO ONLINE– 0x0B - RESETTING MODEM– 0xFF - Unknown

<i>systemMode</i>	<ul style="list-style-type: none">• System Acquisition Mode.<ul style="list-style-type: none">– 0x00 - No service– 0x01 - AMPS– 0x02 - CDMA– 0x03 - GSM– 0x04 - HDR– 0x05 - WCDMA– 0x06 - GPS– 0x08 - WLAN– 0x09 - LTE– 0xFF - Unknown
<i>imsRegState</i>	<ul style="list-style-type: none">• IMS Registration State.<ul style="list-style-type: none">– 0x00 - NO SRV– 0x01 - IN PROG– 0x02 - FAILED– 0x03 - LIMITED– 0x04 - FULL SRV– 0xFF - Unknown
<i>psState</i>	<ul style="list-style-type: none">• PS Attach State.<ul style="list-style-type: none">– 0x00 - Attached– 0x01 - Detached– 0xFF - Unknown

8.108.2 Field Documentation

8.108.2.1 **BYTE** CommInfo::imsRegState

8.108.2.2 **BYTE** CommInfo::modemMode

8.108.2.3 **BYTE** CommInfo::psState

8.108.2.4 **BYTE** CommInfo::systemMode

8.108.2.5 **BYTE** CommInfo::temperature

8.109 ConnectionStatus Struct Reference

Data Fields

- [BYTE](#) MDMConnStatus
- [ULONGLONG](#) MDMCallDuration

8.109.1 Detailed Description

This structure contains modem connection status

Parameters

<i>MDMConn↔ Status</i>	<ul style="list-style-type: none"> Current link status <ul style="list-style-type: none"> 0x01 - DISCONNECTED 0x02 - CONNECTED
<i>MDMCall↔ Duration</i>	<ul style="list-style-type: none"> Call duration in milliseconds. If the modem connection status is connected, this represent the duration of the current DUN call. If the modem connection status is disconnected, this represents the duration of the last DUN call since the device was powered up (zero, if no call has been made or if the last call was not DUN).

8.109.2 Field Documentation

8.109.2.1 ULONGLONG ConnectionStatus::MDMCallDuration

8.109.2.2 BYTE ConnectionStatus::MDMConnStatus

8.110 connectionStatus Struct Reference

Data Fields

- uint8_t [MDMConnStatus](#)
- uint64_t [MDMCallDuration](#)

8.110.1 Detailed Description

Parameters

<i>MDMConn↔ Status</i>	Current link status 1-Disconnected 2-Connected
<i>MDMCall↔ Duration</i>	Call duration in milliseconds

8.110.2 Field Documentation

8.110.2.1 uint64_t connectionStatus::MDMCallDuration

8.110.2.2 uint8_t connectionStatus::MDMConnStatus

8.111 connectNumInfo Struct Reference

Data Fields

- [BYTE numPresInd](#)
- [BYTE screeningInd](#)
- [BYTE numType](#)
- [BYTE numPlan](#)
- [BYTE callerIDLen](#)
- [BYTE callerID \[81\]](#)

8.111.1 Detailed Description

This structure contains information about the numbers connected to a device. It contains information such as number type, eg International or Local.

Parameters

<i>numPresInd</i>	<ul style="list-style-type: none"> • Presentation indicator <ul style="list-style-type: none"> – 0x00 - PRESENTATION_ALLOWED - Allowed presentation – 0x01 - PRESENTATION_RESTRICTED - Restricted presentation – 0x02 - PRESENTATION_NUM_UNAVAILABLE - Unavailable presentation – 0x04 - PRESENTATION_PAYPHONE - Payphone presentation (GSM/UMTS specific) – 0xFF - Not Available
<i>screeningInd</i>	<ul style="list-style-type: none"> • Screening indicator. <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network – 0xFF - Not Available
<i>numType</i>	<ul style="list-style-type: none"> • Number type. <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension – 0xFF - Not Available

<i>numPlan</i>	<ul style="list-style-type: none"> • Number plan. <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension – 0xFF - Not Available
<i>callerIDLen</i>	<ul style="list-style-type: none"> • Provides the length of caller ID which follow. • If zero(0) then no further information exists.
<i>callerID</i> [MAX_ CALL_NO_LEN]	<ul style="list-style-type: none"> • callerID of numLen length, NULL terminated.

8.111.2 Field Documentation

8.111.2.1 **BYTE** connectNumInfo::callerID[81]

8.111.2.2 **BYTE** connectNumInfo::callerIDLen

8.111.2.3 **BYTE** connectNumInfo::numPlan

8.111.2.4 **BYTE** connectNumInfo::numPresInd

8.111.2.5 **BYTE** connectNumInfo::numType

8.111.2.6 **BYTE** connectNumInfo::screeningInd

8.112 CrashInfo Struct Reference

Data Fields

- [WORD](#) numCrashes
- [ULONG](#) crashId
- [ULONG](#) crashData
- [WORD](#) crashStrLen
- [CHAR](#) * pCrashString
- [WORD](#) gcDumpStrLen
- [CHAR](#) * pGCDumpString

8.112.1 Detailed Description

This structure is used to store Crash Information

Parameters

<i>numCrashes</i> [↔ OUT]	<ul style="list-style-type: none"> Number of instances of the remaining fields
<i>crashId</i> [OUT]	<ul style="list-style-type: none"> Random crash id assigned at crash
<i>crashData</i> [OUT]	<ul style="list-style-type: none"> Crash Data
<i>crashStrLen</i> [↔ N/OUT]	<ul style="list-style-type: none"> Length of pCrashString as an INPUT, length of the pCrashString field returned by API as an OUTPUT
<i>pCrashString</i> [↔ OUT]	<ul style="list-style-type: none"> Pointer to store crash string
<i>gcDumpStrLen</i> [↔ IN/OUT]	<ul style="list-style-type: none"> Length of pGCDumpString as an INPUT, Length of the pGCDumpString field returned by API as an OUTPUT
<i>pGCDumpString</i> [↔ OUT]	<ul style="list-style-type: none"> gcdump string for the crash

8.112.2 Field Documentation

8.112.2.1 **ULONG** CrashInfo::crashData

8.112.2.2 **ULONG** CrashInfo::crashId

8.112.2.3 **WORD** CrashInfo::crashStrLen

8.112.2.4 **WORD** CrashInfo::gcDumpStrLen

8.112.2.5 **WORD** CrashInfo::numCrashes

8.112.2.6 **CHAR*** CrashInfo::pCrashString

8.112.2.7 **CHAR*** CrashInfo::pGCDumpString

8.113 crashInfoParams Struct Reference

Data Fields

- [uint8_t](#) [crashStatus](#)
- [crashInformation](#) [crashInfo](#)

8.113.1 Detailed Description

This structure contains [crashInfoParams](#)

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none"> • Unpack Result
<i>crashInfo</i> ↔ <i>Param[OUT]</i>	<ul style="list-style-type: none"> • See crashInfoParams

8.113.2 Field Documentation

8.113.2.1 crashInformation crashInfoParams::crashInfo

8.113.2.2 uint8_t crashInfoParams::crashStatus

8.114 CrashInfoParams Struct Reference

Data Fields

- [BYTE](#) * [pDevCrashStatus](#)
- [CrashInfo](#) * [pCrashInfo](#)

8.114.1 Detailed Description

This structure is used to store Crash Information

Parameters

<i>pDevCrash</i> ↔ <i>Status[OUT]</i>	<ul style="list-style-type: none"> • Device Crash Status • 0 - no crash • 1 - crash has occurred
<i>pCrashInfo</i> [O↔ <i>UT]</i>	<ul style="list-style-type: none"> • Pointer to structure CrashInfo (Optional parameter) • See CrashInfo for more information

8.114.2 Field Documentation

8.114.2.1 CrashInfo* CrashInfoParams::pCrashInfo

8.114.2.2 BYTE* CrashInfoParams::pDevCrashStatus

8.115 crashInformation Struct Reference

Data Fields

- uint16_t [numCrashes](#)

- uint32_t [crashId](#)
- uint32_t [crashData](#)
- uint16_t [crashStrlen](#)
- char [crashString](#) [255]
- uint16_t [gcdumpStrlen](#)
- char [gcdumpString](#) [1024]

8.115.1 Detailed Description

This structure contains [crashInformation](#)

Parameters

<i>numCrashes</i> [↔ OUT]	<ul style="list-style-type: none"> • number of instances of the remaining fields
<i>crashId</i> [OUT]	<ul style="list-style-type: none"> • random crash id assigned at crash
<i>crashData</i>	<ul style="list-style-type: none"> • crash data[OUT]
<i>crashStrlen</i> [↔ N/OUT]	<ul style="list-style-type: none"> • length of the <i>crashString</i> field as an input, length of the <i>crashString</i> field returned by API as an OUTPUT
<i>crashString</i>	<ul style="list-style-type: none"> • crash string
<i>gcdumpStrlen</i> [↔ OUT]	<ul style="list-style-type: none"> • length of the <i>gcdumpString</i> field as an input, length of the <i>gcdumpString</i> field returned by API as an OUTPUT
<i>gcdumpString</i> [↔ OUT]	<ul style="list-style-type: none"> • gcdump string for the crash

8.115.2 Field Documentation

8.115.2.1 uint32_t [crashInformation::crashData](#)

8.115.2.2 uint32_t [crashInformation::crashId](#)

8.115.2.3 char [crashInformation::crashString](#)[255]

8.115.2.4 uint16_t [crashInformation::crashStrlen](#)

8.115.2.5 char [crashInformation::gcdumpString](#)[1024]

8.115.2.6 `uint16_t crashInformation::gcdumpStrlen`

8.115.2.7 `uint16_t crashInformation::numCrashes`

8.116 CreateProfileIn Struct Reference

Data Fields

- [BYTE * pProfileID](#)
- [BYTE * pProfileType](#)
- [QmiProfileInfo curProfile](#)

8.116.1 Detailed Description

This structure contains the input parameters for `SLQSCreateProfile`

Parameters

<i>ProfileID</i>	<ul style="list-style-type: none"> • 1 to 16 for 3GPP profile (EM/MC73xx or earlier) • 1 to 24 for 3GPP profile (EM/MC74xx onwards) • 101 to 106 for 3GPP2 profile
<i>ProfileType</i>	<ul style="list-style-type: none"> • Identifies the technology type of the profile <ul style="list-style-type: none"> – 0x00 - 3GPP – 0x01 - 3GPP2 – NULL is not allowed
<i>curProfile</i>	<ul style="list-style-type: none"> • union of Profile3GPP and Profile3GPP2

Note

- If profileID is NULL, 3GPP profile will be created and index will be assigned based on availability in device.
- If profileID is not NULL depending on pProfileType 3GPP/3GPP2 relevant profile will be created

8.116.2 Field Documentation

8.116.2.1 `QmiProfileInfo CreateProfileIn::curProfile`

8.116.2.2 `BYTE* CreateProfileIn::pProfileID`

8.116.2.3 `BYTE* CreateProfileIn::pProfileType`

8.117 CreateProfileOut Struct Reference

Data Fields

- [BYTE](#) * [pProfileType](#)
- [BYTE](#) * [pProfileIndex](#)
- [USHORT](#) * [pExtErrorCode](#)

8.117.1 Detailed Description

structure contains out parameter Information

Parameters

<i>profileType</i>	<ul style="list-style-type: none">• Identifies the type of profile 0x00 = 3GPP 0x01 = 3GPP2
<i>profileIndex</i>	<ul style="list-style-type: none">• Index identifying the profile that was created
<i>pExtErrorCode</i>	<ul style="list-style-type: none">• The extended error code received from DS Profile subsystem

8.117.2 Field Documentation

8.117.2.1 [USHORT](#)* [CreateProfileOut::pExtErrorCode](#)

8.117.2.2 [BYTE](#)* [CreateProfileOut::pProfileIndex](#)

8.117.2.3 [BYTE](#)* [CreateProfileOut::pProfileType](#)

8.118 CSGID Struct Reference

Data Fields

- [WORD](#) [mcc](#)
- [WORD](#) [mnc](#)
- [BYTE](#) [mncPcsDigits](#)
- [ULONG](#) [id](#)
- [BYTE](#) [rat](#)

8.118.1 Detailed Description

Contain the [CSGID](#).

Parameters

<i>mcc</i>	<ul style="list-style-type: none"> • MCC value. Range 0 to 999
<i>mnc</i>	<ul style="list-style-type: none"> • MNC value. Range 0 to 999
<i>mncPcsDigits</i>	<ul style="list-style-type: none"> • TRUE - MNC is a three-digit value; e.g., a reported value of 90 corresponds to an MNC value of 090 • FALSE - MNC is a two-digit value; e.g., a reported value of 90 corresponds to an MNC value of 90
<i>id</i>	<ul style="list-style-type: none"> • Closed subscriber group identifier.
<i>rat</i>	<ul style="list-style-type: none"> • Radio interface technology of the CSG network. Values: <ul style="list-style-type: none"> – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS – 0x08 - RADIO_IF_LTE - LTE – 0x09 - RADIO_IF_TDSCDMA - TDS

8.118.2 Field Documentation

8.118.2.1 ULONG CSGID::id

8.118.2.2 WORD CSGID::mcc

8.118.2.3 WORD CSGID::mnc

8.118.2.4 BYTE CSGID::mncPcsDigits

8.118.2.5 BYTE CSGID::rat

8.119 CUGInfo Struct Reference

Data Fields

- [WORD CUGIndex](#)
- [BYTE SuppPrefCUG](#)
- [BYTE SuppOA](#)

8.119.1 Detailed Description

This structure contains Closed User Group Information

Parameters

<i>CUGIndex</i>	<ul style="list-style-type: none"> • Range 0x00... 0x7FFF
<i>SuppPrefCUG</i>	<ul style="list-style-type: none"> • Suppress preferential CUG <ul style="list-style-type: none"> – 0x00 - FALSE – 0x01 - TRUE
<i>SuppOA</i>	<ul style="list-style-type: none"> • Suppress OA subscription option <ul style="list-style-type: none"> – 0x00 - FALSE – 0x01 - TRUE

8.119.2 Field Documentation

8.119.2.1 WORD CUGInfo::CUGIndex

8.119.2.2 BYTE CUGInfo::SuppOA

8.119.2.3 BYTE CUGInfo::SuppPrefCUG

8.120 curAMRConfig Struct Reference

Data Fields

- [BYTE gsmAmrStat](#)
- [BYTE wcdmaAmrStat](#)

8.120.1 Detailed Description

This structure contains the Current Adaptive Multi Rate Configuration Information.

Parameters

<i>gsmAmrStat</i>	<ul style="list-style-type: none"> • GSM AMR Status <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable – 0xFF - Not Available
<i>wcdmaAmrStat</i>	<ul style="list-style-type: none"> • WCDMA AMR Status • One or a combination of the following bitmask values: <ul style="list-style-type: none"> – Bit 0 - AMR codec advertised is not supported – Bit 1 - Controls WCDMA AMR wideband – Bit 2 - Controls GSM half rate AMR – Bit 3 - Controls GSM AMR wideband – Bit 4 - Controls GSM AMR narrowband
Generated by Doxygen	<ul style="list-style-type: none"> • 0xFF, if not available

8.120.2 Field Documentation

8.120.2.1 **BYTE** curAMRConfig::gsmAmrStat

8.120.2.2 **BYTE** curAMRConfig::wcdmaAmrStat

8.121 CurrDataSysStat Struct Reference

Data Fields

- [BYTE](#) * pPrefNetwork
- [BYTE](#) * pNetworkInfoLen
- [CurrNetworkInfo](#) * pCurrNetworkInfo

8.121.1 Detailed Description

Data System Status

Parameters

<i>pPrefNetwork</i>	[OUT] <ul style="list-style-type: none"> • Preferred Network • Values: <ul style="list-style-type: none"> – 0 - 3GPP – 1 - 3GPP2
<i>pNetworkInfoLen</i>	[IN/OUT] <ul style="list-style-type: none"> • As input parameter size assigned to next parameter i.e. network information • As output the actual number of network information elements returned by the device
<i>pCurrNetworkInfo</i>	[OUT] <ul style="list-style-type: none"> • Network information • See CurrNetworkInfo for more details

8.121.2 Field Documentation

8.121.2.1 **CurrNetworkInfo*** CurrDataSysStat::pCurrNetworkInfo

8.121.2.2 **BYTE*** CurrDataSysStat::pNetworkInfoLen

8.121.2.3 **BYTE*** CurrDataSysStat::pPrefNetwork

8.122 currentCatEvent Union Reference

Data Fields

- struct [CatEventIDDDataTlv](#) CatEvIDDData

- struct [CatAlPhalIdentifierTlv](#) [CatAlphaldtfr](#)
- struct [CatEventListTlv](#) [CatEventLst](#)
- struct [CatRefreshTlv](#) [CatRefresh](#)
- struct [CatEndProactiveSessionTlv](#) [CatEndPS](#)

8.122.1 Detailed Description

Union used to represent the current CAT Event Data. Choose the structure based on the EventID received.

- Use [CatEventIDDataTlv](#) if the Event ID is any of the below.
 - 16
 - 17
 - 18
 - 19
 - 20
 - 23
 - 24
- Use [CatAlPhalIdentifierTlv](#) if the Event ID is 21
- Use [CatEventListTlv](#) if the Event ID is 22
- Use [CatRefreshTlv](#) if the Event ID is 25
- Use [CatEndProactiveSessionTlv](#) if the Event ID is 26

8.122.2 Field Documentation

8.122.2.1 struct [CatAlPhalIdentifierTlv](#) [currentCatEvent::CatAlphaldtfr](#)

8.122.2.2 struct [CatEndProactiveSessionTlv](#) [currentCatEvent::CatEndPS](#)

8.122.2.3 struct [CatEventListTlv](#) [currentCatEvent::CatEventLst](#)

8.122.2.4 struct [CatEventIDDataTlv](#) [currentCatEvent::CatEvIDData](#)

8.122.2.5 struct [CatRefreshTlv](#) [currentCatEvent::CatRefresh](#)

8.123 CurrentImgList Struct Reference

Data Fields

- [BYTE](#) [numEntries](#)
- [CurrImageInfo](#) * [pCurrImgInfo](#)
- [CHAR](#) [priver](#) [16]
- [CHAR](#) [pkgver](#) [16]
- [CHAR](#) [fwvers](#) [16]
- [CHAR](#) [carrier](#) [16]

8.123.1 Detailed Description

This structure is used to store image list

Parameters

<i>numEntries</i> [\leftrightarrow N/OUT]	<ul style="list-style-type: none"> Number of entries in the image list to follow The size of the list pCurrImgInfo must be specified when calling the API
<i>pCurrImgInfo</i> [\leftrightarrow OUT]	<ul style="list-style-type: none"> Currently Active Image List
<i>priver</i> [OUT]	<ul style="list-style-type: none"> PR1 version of the currently running firmware
<i>pkgver</i> [OUT]	<ul style="list-style-type: none"> Package version of the currently running firmware
<i>fwvers</i> [OUT]	<ul style="list-style-type: none"> firmware version of the currently running firmware
<i>carrier</i> [OUT]	<ul style="list-style-type: none"> Carrier string of the currently running firmware

8.123.2 Field Documentation

8.123.2.1 **CHAR** CurrentImgList::carrier[16]

8.123.2.2 **CHAR** CurrentImgList::fwvers[16]

8.123.2.3 **BYTE** CurrentImgList::numEntries

8.123.2.4 **CurrImageInfo*** CurrentImgList::pCurrImgInfo

8.123.2.5 **CHAR** CurrentImgList::pkgver[16]

8.123.2.6 **CHAR** CurrentImgList::priver[16]

8.124 currentPLMN Struct Reference

Data Fields

- [WORD](#) MCC
- [WORD](#) MNC
- [BYTE](#) netDescrLength
- [BYTE](#) netDescr [255]

8.124.1 Detailed Description

This structure contains the current PLMN parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>MCC</i>	<ul style="list-style-type: none"> • mobile country code <ul style="list-style-type: none"> – A 16 bit representation of MCC – Range 0 to 999
<i>MNC</i>	<ul style="list-style-type: none"> • mobile network code <ul style="list-style-type: none"> – A 16 bit representation of MNC – Range 0 to 999
<i>netDescrLength</i>	<ul style="list-style-type: none"> • Length of Network description field • Defaults to zero
<i>netDescr</i>	<ul style="list-style-type: none"> • Network Description <ul style="list-style-type: none"> – optional string containing network name or description

8.124.2 Field Documentation

8.124.2.1 WORD currentPLMN::MCC

8.124.2.2 WORD currentPLMN::MNC

8.124.2.3 BYTE currentPLMN::netDescr[255]

8.124.2.4 BYTE currentPLMN::netDescrLength

8.125 CurrlmageInfo Struct Reference

Data Fields

- [BYTE imageType](#)
- [BYTE uniqueID \[16\]](#)
- [BYTE buildIDLen](#)
- [BYTE buildID \[255\]](#)

8.125.1 Detailed Description

This structure is used to store image information

Parameters

<i>imageType</i> [O↔ UT]	<ul style="list-style-type: none"> • Image Type • Values: <ul style="list-style-type: none"> – 0 - FW – 1 - configuration
<i>uniqueID</i> [OUT]	<ul style="list-style-type: none"> • Image Unique Identifier (ASCII characters)
<i>buildIDLen</i> [OUT]	<ul style="list-style-type: none"> • Length of build ID string to follow • If set to zero, build ID string will be blank
<i>buildID</i> [OUT]	<ul style="list-style-type: none"> • String containing image information(ASCII characters) • Maximum length of this string is 255 chars

8.125.2 Field Documentation

8.125.2.1 **BYTE** CurrImageInfo::buildID[255]

8.125.2.2 **BYTE** CurrImageInfo::buildIDLen

8.125.2.3 **BYTE** CurrImageInfo::imageType

8.125.2.4 **BYTE** CurrImageInfo::uniqueID[16]

8.126 CurrNetworkInfo Struct Reference

Data Fields

- [BYTE NetworkType](#)
- [ULONG RATMask](#)
- [ULONG SOMask](#)

8.126.1 Detailed Description

Network information structure

Parameters

<i>NetworkType</i>	<ul style="list-style-type: none">• Values:<ul style="list-style-type: none">– 0 - 3GPP– 1 - 3GPP2
<i>RATMask</i>	<ul style="list-style-type: none">• Radio Access Technology (RAT) mask to indicate the type of technology.• Values:<ul style="list-style-type: none">– 0 - Don't Care– 0x8000 - NULL Bearer• CDMA RAT mask values:<ul style="list-style-type: none">– 0x01 - CDMA_1x– 0x02 - EVDO_REV0– 0x04 - EVDO_REVA– 0x08 - EVDO_REVB– 0x10 - EHRPD– 0x20 - FMC• UMTS RAT mask values:<ul style="list-style-type: none">– 0x01 - WCDMA– 0x02 - GPRS– 0x04 - HSDPA– 0x08 - HSUPA– 0x10 - EDGE– 0x20 - LTE– 0x40 - HSDPA+– 0x80 - DC_HSDPA+– 0x100 - 64_QAM– 0x200 - TDSCDMA

<i>SOMask</i>	<ul style="list-style-type: none"> • Service Option (SO) mask to indicate the service option or type of application. • Values: <ul style="list-style-type: none"> – 0 - Don't Care • CDMA 1x SO mask values: <ul style="list-style-type: none"> – 0x01 - CDMA_1X_IS95 – 0x02 - CDMA_1X_IS2000 – 0x04 - CDMA_1X_IS2000_REL_A • CDMA EV-DO Rev 0 SO mask values: <ul style="list-style-type: none"> – 0x01 - DPA • CDMA EV-DO Rev A SO mask values: <ul style="list-style-type: none"> – 0x01 - DPA – 0x02 - MFPA – 0x04 - EMPA – 0x08 - EMPA_EHRPD • CDMA EV-DO Rev B SO mask values: <ul style="list-style-type: none"> – 0x01 - DPA – 0x02 - MFPA – 0x04 - EMPA – 0x08 - EMPA_EHRPD – 0x10 - MMPA – 0x20 - MMPA_EHRPD
---------------	--

8.126.2 Field Documentation

8.126.2.1 BYTE CurrNetworkInfo::NetworkType

8.126.2.2 ULONG CurrNetworkInfo::RATMask

8.126.2.3 ULONG CurrNetworkInfo::SOMask

8.127 currNetworkInfo Struct Reference

Data Fields

- uint8_t [NetworkType](#)
- uint32_t [RATMask](#)
- uint32_t [SOMask](#)

8.127.1 Detailed Description

Parameters

<i>NetworkType</i>	network type 0-3GPP 1-3GPP2
<i>RATMask</i>	Radio access technology mask to indicate technology.
<i>SOMask</i>	Service Option mask to indicate the service option.

8.127.2 Field Documentation

8.127.2.1 `uint8_t currNetworkInfo::NetworkType`

8.127.2.2 `uint32_t currNetworkInfo::RATMask`

8.127.2.3 `uint32_t currNetworkInfo::SOMask`

8.128 custFeaturesInfo Struct Reference

Data Fields

- [ULONG GpsEnable](#)
- [BYTE * pDisableIMSI](#)
- [WORD * pIPFamSupport](#)
- [BYTE * pRMAutoConnect](#)
- [BYTE * pGPSSel](#)
- [BYTE * pSMSSupport](#)
- [BYTE * plsVoiceEnabled](#)
- [BYTE * pDHCPRelayEnabled](#)
- [BYTE * pGPSLPM](#)

8.128.1 Detailed Description

This structure contains current settings of custom features

Parameters

<i>GpsEnable</i> [\leftrightarrow <i>UT</i>]	<ul style="list-style-type: none"> • describes if GPS is enabled or disabled • values: <ul style="list-style-type: none"> – 0x00 - GPS is disabled – 0x01 - GPS is enabled • function SLQSGetCustFeatures() returns a default value FFFFFFFF if no value is returned by the modem
<i>pDisableIMSI</i> [\leftrightarrow <i>OUT</i>]	<ul style="list-style-type: none"> • optional 1 byte parameter • describes if IMSI display is enabled or disabled • values: <ul style="list-style-type: none"> – 0x00 - Allow display of IMSI – 0x01 - Do not display IMSI • function SLQSGetCustFeatures() returns a default value FF if no value is returned by the modem

<p><i>pIPFam</i>↔ <i>Support[OUT]</i></p>	<ul style="list-style-type: none"> • optional 2 byte BitMask • bitmask representing the IP families supported • values: <ul style="list-style-type: none"> – 0x01 - IPv4 – 0x02 - IPv6 – 0x04 - IPv4v6 • function SLQSGetCustFeatures() returns a default value FFFF if no value is returned by the modem
<p><i>pRMAuto</i>↔ <i>Connect[OUT]</i></p>	<ul style="list-style-type: none"> • optional 1 byte parameter • QMI Mode RM Net Auto Connect Support • values: <ul style="list-style-type: none"> – 0x00 - Not Supported – 0x01 - Supported • function SLQSGetCustFeatures() returns a default value FF if no value is returned by the modem
<p><i>pGPSSel[OUT]</i></p>	<ul style="list-style-type: none"> • optional 1 byte parameter • GPS Antenna Select • values: <ul style="list-style-type: none"> – 0x00 - Dedicated GPS Port – 0x01 - GPS Rx over AUX Port – 0x02 - GPS Rx over dedicated GPS port with no bias voltage applied • function SLQSGetCustFeatures() returns a default value FF if no value is returned by the modem
<p><i>pSMSSupport</i>↔ <i>OUT]</i></p>	<ul style="list-style-type: none"> • optional 1 byte parameter • SMS support • values: <ul style="list-style-type: none"> – 0x00 - Not supported – 0x01 - supported • Used to determine whether or not to hide SMS from user • function SLQSGetCustFeatures() returns a default value FF if no value is returned by the modem. In this case assume, SMS is supported.
<p><i>pIsVoice</i>↔ <i>Enabled[OUT]</i></p>	<ul style="list-style-type: none"> • optional 1 byte parameter • Voice support • values: <ul style="list-style-type: none"> – 0x00 - Enable voice on both AT and QMI interface (default) – 0x01 - Reserved – 0x02 - Disable voice on both AT and QMI interface

<i>pDHCPRelayEnabled</i> [OUT]	<ul style="list-style-type: none"> • optional 1 byte parameter • DHCP Relay support • values: <ul style="list-style-type: none"> – 0x00 - Disable DHCP relay – 0x01 - Enable DHCP relay
<i>pGPSLPM</i> [OUT]	<ul style="list-style-type: none"> • optional 1 byte parameter • GPSLPM support • values: <ul style="list-style-type: none"> – 0x00 - Enable GPS in Low Power Mode – 0x01 - Disable GPS in Low Power Mode

8.128.2 Field Documentation

8.128.2.1 **ULONG** custFeaturesInfo::GpsEnable

8.128.2.2 **BYTE*** custFeaturesInfo::pDHCPRelayEnabled

8.128.2.3 **BYTE*** custFeaturesInfo::pDisableIMSI

8.128.2.4 **BYTE*** custFeaturesInfo::pGPSLPM

8.128.2.5 **BYTE*** custFeaturesInfo::pGPSSel

8.128.2.6 **WORD*** custFeaturesInfo::pIPFamSupport

8.128.2.7 **BYTE*** custFeaturesInfo::plsVoiceEnabled

8.128.2.8 **BYTE*** custFeaturesInfo::pRMAutoConnect

8.128.2.9 **BYTE*** custFeaturesInfo::pSMSSupport

8.129 custFeaturesSetting Struct Reference

Data Fields

- **BYTE *** pGPSSel
- **ULONG *** pGPSEnable
- **BYTE *** plsVoiceEnabled
- **BYTE *** pDHCPRelayEnabled
- **BYTE *** pGPSLPM

8.129.1 Detailed Description

This structure contains settings to be used for custom features

Parameters

<i>pGPSSel</i>	<ul style="list-style-type: none"> • optional 1 byte parameter • GPS Antenna Select • values: <ul style="list-style-type: none"> – 0x00 - Dedicated GPS Port – 0x01 - GPS Rx over AUX Port – 0x02 - GPS Rx over dedicated GPS port with no bias voltage applied
<i>pGPSEnable</i>	<ul style="list-style-type: none"> • optional 4 byte parameter • GPS Enable/Disable • values: The value of 7 least significant bits: <ul style="list-style-type: none"> – 0 - Disabled – 1 - MT & MO enabled – 2 - MO enabled – 3 - MT enabled – 4 - MT & MO enabled if GPS_DISABLE pin is not asserted – 5 - MO GPS enabled if GPS_DISABLE pin is not asserted – 6 - MT GPS enabled if GPS_DISABLE pin is not asserted

Note

Only MC7750 3.5.x firmware supports above 0x04, 0x05 and 0x06 settings. To disable GLONASS, set the most significant bit - 0x80. This setting is only applicable if GPS is not Disabled.

Parameters

<i>plsVoiceEnabled</i>	<ul style="list-style-type: none"> • optional 1 byte parameter • voice enabled/disabled • values: <ul style="list-style-type: none"> – 0 - Enable voice on both AT and QMI interface (default) – 1 - Reserved – 2 - Disable voice on both AT and QMI interface
<i>pDHCPRelay↔ Enabled</i>	<ul style="list-style-type: none"> • optional 1 byte parameter • DHCPRELAYENABLE support • values: <ul style="list-style-type: none"> – 0 - Disable DHCP relay – 1 - Enable DHCP relay
<i>pGPSLPM</i>	<ul style="list-style-type: none"> • optional 1 byte parameter • GPSLPM support • values: <ul style="list-style-type: none"> – 0 - Enable GPS in Low Power Mode – 1 - Disable GPS in Low Power Mode

8.129.2 Field Documentation

8.129.2.1 **BYTE*** custFeaturesSetting::pDHCPRelayEnabled

8.129.2.2 **ULONG*** custFeaturesSetting::pGPSEnable

8.129.2.3 **BYTE*** custFeaturesSetting::pGPSLPM

8.129.2.4 **BYTE*** custFeaturesSetting::pGPSSel

8.129.2.5 **BYTE*** custFeaturesSetting::plsVoiceEnabled

8.130 custSettingInfo Struct Reference

Data Fields

- [WORD id_length](#)
- [CHAR cust_id \[64+1\]](#)
- [WORD value_length](#)
- [BYTE cust_value \[8+1\]](#)
- [WORD cust_attr](#)

8.130.1 Detailed Description

This structure contains information about Customization Setting. This TLV is only applicable for 9x30 modules so far

Parameters

<i>id_length</i>	<ul style="list-style-type: none">• length of cust_id field
<i>cust_id</i>	<ul style="list-style-type: none">• Customization ID (Maximum 64 bytes)
<i>value_length</i>	<ul style="list-style-type: none">• length of cust_value field
<i>cust_value</i>	<ul style="list-style-type: none">• Customization Setting Value (Maximum 8 bytes)
<i>cust_attr</i>	<ul style="list-style-type: none">• Customization Setting attribute through QMI<ul style="list-style-type: none">– bit 0: Values:<ul style="list-style-type: none">* 0 - read only* 1 - read/write

8.130.2 Field Documentation

8.130.2.1 WORD custSettingInfo::cust_attr

8.130.2.2 CHAR custSettingInfo::cust_id[64+1]

8.130.2.3 BYTE custSettingInfo::cust_value[8+1]

8.130.2.4 WORD custSettingInfo::id_length

8.130.2.5 WORD custSettingInfo::value_length

8.131 custSettingList Struct Reference

Data Fields

- [BYTE list_type](#)
- [WORD num_instances](#)
- [custSettingInfo custSetting](#) [256]

8.131.1 Detailed Description

This structure contains the fields of TLV Customization Setting List. This TLV is only applicable for 9x30 modules so far

Parameters

<i>list_type</i>	<ul style="list-style-type: none"> • list type requested
<i>num_instances</i>	<ul style="list-style-type: none"> • number of instances of customization setting
<i>custSetting</i>	<ul style="list-style-type: none"> • See custSettingInfo for more information

8.131.2 Field Documentation

8.131.2.1 [custSettingInfo custSettingList::custSetting](#)[256]

8.131.2.2 BYTE custSettingList::list_type

8.131.2.3 WORD custSettingList::num_instances

8.132 dataBearers Struct Reference

Data Fields

- [BYTE dataBearerMask](#)
- [QmiWDSDataBearerTechnology](#) * [pCurDataBearerTechnology](#)
- [QmiWDSDataBearerTechnology](#) * [pLastCallDataBearerTechnology](#)

8.132.1 Detailed Description

Structure to hold the data bearer technology values

Parameters

<i>dataBearerMask</i> [OUT]	<ul style="list-style-type: none"> • This bit mask indicates if data bearer information for the current and/or last call has been received from the device. If a bit is set, then the information is available in the corresponding structure i.e. the one provided by the caller. Refer to qmiDataBearerMasks for bit-mask positions.
<i>pCurDataBearerTechnology</i> [OUT]	<ul style="list-style-type: none"> • current data bearer technology value. <ul style="list-style-type: none"> – NULL if the parameter is not required
<i>pLastCallDataBearerTechnology</i> [OUT]	<ul style="list-style-type: none"> • last call data bearer technology value. <ul style="list-style-type: none"> – NULL if the parameter is not required

8.132.2 Field Documentation

8.132.2.1 **BYTE** `dataBearers::dataBearerMask`

8.132.2.2 **QmiWDSDataBearerTechnology*** `dataBearers::pCurDataBearerTechnology`

8.132.2.3 **QmiWDSDataBearerTechnology*** `dataBearers::pLastCallDataBearerTechnology`

8.133 DataBearerTech Struct Reference

Data Fields

- [ULONG techType](#)
- [ULONG ratValue](#)
- [ULONGLONG soMask](#)

8.133.1 Detailed Description

Network information structure

Parameters

<i>TechType</i>	<ul style="list-style-type: none">• Technology type• Values:<ul style="list-style-type: none">– 0 - WDS_BEARER_TECH_NETWORK_3GPP - 3GPP– 1 - WDS_BEARER_TECH_NETWORK_3GPP2 - 3GPP2
<i>ratValue</i>	<ul style="list-style-type: none">• Radio Access Technology (RAT) value• Values:<ul style="list-style-type: none">– 0x00 - WDS_BEARER_TECH_RAT_EX_NULL_BEARER - NULL bearer– 0x01 - WDS_BEARER_TECH_RAT_EX_3GPP_WCDMA - 3GPP WCDMA– 0x02 - WDS_BEARER_TECH_RAT_EX_3GPP_GERAN - 3GPP GERAN– 0x03 - WDS_BEARER_TECH_RAT_EX_3GPP_LTE - 3GPP LTE– 0x04 - WDS_BEARER_TECH_RAT_EX_3GPP_TDSCDMA - 3GPP TDSCDMA– 0x05 - WDS_BEARER_TECH_RAT_EX_3GPP_WLAN - 3GPP WLAN– 0x64 - WDS_BEARER_TECH_RAT_EX_3GPP_MAX - 3GPP maximum– 0x65 - WDS_BEARER_TECH_RAT_EX_3GPP2_1X - 3GPP2 1X– 0x66 - WDS_BEARER_TECH_RAT_EX_3GPP2_HRPD - 3GPP2 HRPD– 0x67 - WDS_BEARER_TECH_RAT_EX_3GPP2_EHRPD - 3GPP2 EHRPD– 0x68 - WDS_BEARER_TECH_RAT_EX_3GPP2_WLAN - 3GPP2 WLAN– 0xC8 - WDS_BEARER_TECH_RAT_EX_3GPP2_MAX - 3GPP2 maximum

<i>SOMask</i>	<ul style="list-style-type: none"> • Service Option (SO) mask to indicate the service option or type of application. An SO mask value of zero indicates that this field is ignored. • Values: <ul style="list-style-type: none"> – 0x00 - SO mask unspecified • 3GPP SO mask: <ul style="list-style-type: none"> – 0x01 - WCDMA – 0x02 - HSDPA – 0x04 - HSUPA – 0x08 - HSDPAPLUS – 0x10 - DC HSDPAPLUS – 0x20 - 64 QAM – 0x40 - HSPA – 0x80 - GPRS – 0x100 - EDGE – 0x200 - GSM – 0x400 - S2B – 0x800 - LTE limited service – 0x1000 - LTE FDD – 0x2000 - LTE TDD • 3GPP2 SO mask: <ul style="list-style-type: none"> – 0x01000000 - 1X IS95 – 0x02000000 - 1X IS2000 – 0x04000000 - 1X IS2000 REL A – 0x08000000 - HDR REV0 DPA – 0x10000000 - HDR REVA DPA – 0x20000000 - HDR REVB DPA – 0x40000000 - HDR REVA MPA – 0x80000000 - HDR REVB MPA – 0x100000000 - HDR REVA EMPA – 0x200000000 - HDR REVB EMPA – 0x400000000 - HDR REVB MMPA – 0x800000000 - HDR EVDO FMC
---------------	---

8.133.2 Field Documentation

8.133.2.1 **ULONG** DataBearerTech::ratValue

8.133.2.2 **ULONGLONG** DataBearerTech::soMask

8.133.2.3 **ULONG** DataBearerTech::techType

8.134 DataBearerTechExt Struct Reference

Data Fields

- [DataBearerTech](#) * [pBearerTech](#)
- [DataBearerTech](#) * [pLastBearerTech](#)

8.134.1 Detailed Description

Data Bearer Technology Ext

Parameters

<i>pBearerTech</i>	[OUT] <ul style="list-style-type: none">• See DataBearerTech for more information
<i>pLastBearerTech</i>	[OUT] <ul style="list-style-type: none">• See DataBearerTech for more information

8.134.2 Field Documentation

8.134.2.1 [DataBearerTech](#)* [DataBearerTechExt](#)::pBearerTech

8.134.2.2 [DataBearerTech](#)* [DataBearerTechExt](#)::pLastBearerTech

8.135 dataBearerTechnology Struct Reference

Data Fields

- [BYTE](#) currentNetwork
- [ULONG](#) ratMask
- [ULONG](#) soMask

8.135.1 Detailed Description

Structure to hold the current data bearer technology values

Parameters

<i>pCurrentNetwork</i> <i>Network[OUT]</i>	<ul style="list-style-type: none">• current selected network<ul style="list-style-type: none">– 0 - UNKNOWN– 1 - 3GPP2– 2 - 3GPP
---	--

<i>pRatMask[OUT]</i>	<ul style="list-style-type: none"> • Radio Access Technology (RAT) mask to indicate the type of technology (RAT mask value of zero indicates that this field is ignored) <ul style="list-style-type: none"> – 0x8000 - NULL Bearer – 0x0000 - DO_NOT_CARE CDMA RAT mask – 0x01 - CDMA_1X – 0x02 - EVDO_REV0 – 0x04 - EVDO_REVA UMTS RAT mask – 0x01 - WCDMA – 0x02 - GPRS – 0x04 - HSDPA – 0x08 - HSUPA – 0x10 - EDGE – 0x20 - LTE – 0x40 - HSDPA+ – 0x80 - DC_HSDPA+
<i>pSoMask[OUT]</i>	<ul style="list-style-type: none"> • Service Option (SO) mask to indicate the SO or type of application (SO mask value of zero indicates that this field is ignored) <ul style="list-style-type: none"> – 0x00 - DO_NOT_CARE CDMA 1X SO mask – 0x01 - CDMA_1X_IS95 – 0x02 - CDMA_1X_IS2000 – 0x04 - CDMA_1X_IS2000_REL_A CDMA EV-DO Rev A SO mask – 0x01 - EVDO_REVA_DPA – 0x02 - EVDO_REVA_MFPA – 0x04 - EVDO_REVA_EMPA – 0x08 - EVDO_REVA_EMPA_EHRPD

8.135.2 Field Documentation

8.135.2.1 **BYTE** dataBearerTechnology::currentNetwork

8.135.2.2 **ULONG** dataBearerTechnology::ratMask

8.135.2.3 **ULONG** dataBearerTechnology::soMask

8.136 dataRate Struct Reference

Data Fields

- [ULONG dataRateMax](#)
- [ULONG guaranteedRate](#)

8.136.1 Detailed Description

This structure contains the IP flow data rate min max

Parameters

<i>dataRateMax</i>	Maximum required data rate (bits per second)
<i>guaranteedRate</i>	Minimum guaranteed data rate (bits per second)

8.136.2 Field Documentation

8.136.2.1 **ULONG** *dataRate::dataRateMax*

8.136.2.2 **ULONG** *dataRate::guaranteedRate*

8.137 dataSrvCapabilities Struct Reference

Data Fields

- **BYTE** *dataCapabilitiesLen*
- **BYTE** *dataCapabilities* [0x20]

8.137.1 Detailed Description

This structure contains the data services capability

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>dataCapabilitiesLen</i>	<ul style="list-style-type: none"> • Length of data capabilities list • Defaults to zero
<i>dataCapabilities</i>	<ul style="list-style-type: none"> • List of data capabilities • Values: <ul style="list-style-type: none"> – 0x01 - GPRS – 0x02 - EDGE – 0x03 - HSDPA – 0x04 - HSUPA – 0x05 - WCDMA – 0x06 - CDMA – 0x07 - EV-DO Rev0 – 0x08 - EV-DO RevA – 0x09 - GSM – 0x0A - EV-DO Rev B – 0x0B - LTE – 0x0C - HSDPA+ – 0x0D - DC-HSDPA+

8.137.2 Field Documentation

8.137.2.1 **BYTE** dataSrvCapabilities::dataCapabilities[0x20]

8.137.2.2 **BYTE** dataSrvCapabilities::dataCapabilitiesLen

8.138 DataStatusDetail Struct Reference

Data Fields

- [ULONG](#) IPAddress
- [BYTE](#) LastErrCode

8.138.1 Detailed Description

This structure contains Data Status Details

Parameters

<i>IPAddress</i>	<ul style="list-style-type: none">• IP Address• 0xABCDEFGH - AB.CD.EF.GH• Example:<ul style="list-style-type: none">– 0x12345678 - 18.52.86.120 0x12=18 0x34=52 0x56=86 0x78=120• 0xFFFFFFFF - NA
------------------	--

<i>LastErrCode</i>	<ul style="list-style-type: none"> • MIP Error code <ul style="list-style-type: none"> – 0x00 - MIP_RRP_CODE_SUCCESS – 0x01 - MIP_RRP_CODE_SUCCESS_NO_SIM_BINDINGS – 0x40 - MIP_RRP_CODE_FAILURE_FA_REASON_UNSPECIFIED – 0x41 - MIP_RRP_CODE_FAILURE_FA_ADMIN_PROHIBITED – 0x42 - MIP_RRP_CODE_FAILURE_FA_INSUFFICIENT_RESOURCES – 0x43 - MIP_RRP_CODE_FAILURE_FA_MOBILE_NODE_FAILED_AUTH – 0x44 - MIP_RRP_CODE_FAILURE_FA_HA_FAILED_AUTH – 0x45 - MIP_RRP_CODE_FAILURE_FA_REQUESTED_LIFETIME_TOO_LONG – 0x46 - MIP_RRP_CODE_FAILURE_FA_MALFORMED_REQUEST – 0x47 - MIP_RRP_CODE_FAILURE_FA_MALFORMED_REPLY – 0x48 - MIP_RRP_CODE_FAILURE_FA_ENCAPSULATION_UNAVAILABLE – 0x49 - MIP_RRP_CODE_FAILURE_FA_VJHC_UNAVAILABLE – 0x4A - MIP_RRP_CODE_FAILURE_FA_CANT_REV_TUN – 0x4B - MIP_RRP_CODE_FAILURE_FA_MUST_REV_TUN – 0x4C - MIP_RRP_CODE_FAILURE_FA_BAD_TTL – 0x4D - MIP_RRP_CODE_FAILURE_INVALID_COA – 0x4F - MIP_RRP_CODE_FAILURE_FA_DELIVERY_STYLE_NOT_SUPPORTED – 0x59 - MIP_RRP_CODE_FAILURE_FA_VS_REASON – 0x61 - MIP_RRP_CODE_FAILURE_MISSING_NAI – 0x62 - MIP_RRP_CODE_FAILURE_MISSING_HA_ADDR – 0x63 - MIP_RRP_CODE_FAILURE_MISSING_HOMEADDR – 0x68 - MIP_RRP_CODE_FAILURE_UNKNOWN_CHALLENGE – 0x69 - MIP_RRP_CODE_FAILURE_MISSING_CHALLENGE – 0x6A - MIP_RRP_CODE_FAILURE_STALE_CHALLENGE – 0x6B - MIP_RRP_CODE_FAILURE_MISSING_MN_FA – 0x80 - MIP_RRP_CODE_FAILURE_HA_REASON_UNSPECIFIED – 0x81 - MIP_RRP_CODE_FAILURE_HA_ADMIN_PROHIBITED – 0x82 - MIP_RRP_CODE_FAILURE_HA_INSUFFICIENT_RESOURCES – 0x83 - MIP_RRP_CODE_FAILURE_HA_MOBILE_NODE_FAILED_AUTH – 0x84 - MIP_RRP_CODE_FAILURE_HA_FA_FAILED_AUTH – 0x85 - MIP_RRP_CODE_FAILURE_HA_REG_ID_MISMATCH – 0x86 - MIP_RRP_CODE_FAILURE_HA_MALFORMED_REQUEST – 0x88 - MIP_RRP_CODE_FAILURE_UNKNOWN_HA – 0x89 - MIP_RRP_CODE_FAILURE_HA_CANT_REV_TUN – 0x8A - MIP_RRP_CODE_FAILURE_HA_MUST_REV_TUN – 0x8B - MIP_RRP_CODE_FAILURE_HA_ENCAPSULATION_UNAVAILABLE – 0x8F - MIP_RRP_CODE_FAILURE_REDIRECTED_HA – 0x90 - MIP_RRP_CODE_FAILURE_HA_BAD_AAA_AUTH – 0xFF - NA
--------------------	--

8.138.2 Field Documentation

8.138.2.1 ULONG DataStatusDetail::IPAddress

8.138.2.2 BYTE DataStatusDetail::LastErrCode

8.139 DataULongLongTlv Struct Reference

Data Fields

- [BYTE TlvPresent](#)
- [ULONGLONG ullData](#)

8.139.1 Field Documentation

8.139.1.1 **BYTE** DataULongLongTlv::TlvPresent

8.139.1.2 **ULONGLONG** DataULongLongTlv::ullData

8.140 DataULongTlv Struct Reference

Data Fields

- [BYTE TlvPresent](#)
- [ULONG ulData](#)

8.140.1 Field Documentation

8.140.1.1 **BYTE** DataULongTlv::TlvPresent

8.140.1.2 **ULONG** DataULongTlv::ulData

8.141 DcsUsbPortNames Struct Reference

Data Fields

- [CHAR AtCmdPort](#) [32]
- [CHAR NmeaPort](#) [32]
- [CHAR DmPort](#) [32]

8.141.1 Field Documentation

8.141.1.1 **CHAR** DcsUsbPortNames::AtCmdPort[32]

8.141.1.2 **CHAR** DcsUsbPortNames::DmPort[32]

8.141.1.3 **CHAR** DcsUsbPortNames::NmeaPort[32]

8.142 delAssistDataStatus Struct Reference

Data Fields

- [ULONG status](#)

8.142.1 Detailed Description

Contain the parameters passed for SetLocDeleteAssistDataCallback by the device.

Parameters

<i>status</i>	<ul style="list-style-type: none"> • Status of the Delete Assist Data request • Valid values: <ul style="list-style-type: none"> – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure
---------------	---

Note

None

8.142.2 Field Documentation**8.142.2.1 ULONG delAssistDataStatus::status****8.143 depersonalizationInformation Struct Reference****Data Fields**

- [BYTE feature](#)
- [BYTE operation](#)
- [BYTE ckLen](#)
- [BYTE ckVal \[255\]](#)

8.143.1 Detailed Description

This structure contains the Depersonalization Information.

Parameters

<i>feature</i>	<ul style="list-style-type: none"> Indicates the personalization feature to de-activate or unblock. <ul style="list-style-type: none"> 0 - GW network personalization 1 - GW network subset personalization 2 - GW service provider personalization 3 - GW corporate personalization 4 - GW UIM personalization 5 - 1X network type 1 personalization 6 - 1X network type 2 personalization 7 - 1X HRPD personalization 8 - 1X service provider personalization 9 - 1X corporate personalization 10 - 1X RUIM personalization
<i>operation</i>	<ul style="list-style-type: none"> Indicates the operation to perform. <ul style="list-style-type: none"> 0 - Deactivate personalization. 1 - Unblock personalization.
<i>ckLen</i>	<ul style="list-style-type: none"> Length of the following elements i.e. control key value.
<i>ckVal</i> [<i>MAX_DESCRIPTOR_LENGTH</i>]	<ul style="list-style-type: none"> Control key value. This value is a sequence of ASCII characters.

8.143.2 Field Documentation

8.143.2.1 BYTE depersonalizationInformation::ckLen

8.143.2.2 BYTE depersonalizationInformation::ckVal[255]

8.143.2.3 BYTE depersonalizationInformation::feature

8.143.2.4 BYTE depersonalizationInformation::operation

8.144 detailSvcInfo Struct Reference

Data Fields

- [BYTE srvStatus](#)
- [BYTE srvCapability](#)
- [BYTE hdrSrvStatus](#)
- [BYTE hdrHybrid](#)
- [BYTE isSysForbidden](#)

8.144.1 Detailed Description

This structure contains Detailed Service information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>srvStatus</i>	<ul style="list-style-type: none"> • Service status • Values: <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Limited service – 0x02 - Service available – 0x03 - Limited regional service – 0x04 - MS in power save or deep sleep
<i>srvCapability</i>	<ul style="list-style-type: none"> • System's service capability • Values: <ul style="list-style-type: none"> – 0x00 - No Service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - MS found the right system but not yet registered/attached
<i>hdrSrvStatus</i>	<ul style="list-style-type: none"> • HDR service status • Values: <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Limited service – 0x02 - Service available – 0x03 - Limited regional service – 0x04 - MS in power save or deep sleep
<i>hdrHybrid</i>	<ul style="list-style-type: none"> • HDR hybrid information • Values: <ul style="list-style-type: none"> – 0x00 - System is not hybrid – 0x01 - System is hybrid
<i>isSysForbidden</i>	<ul style="list-style-type: none"> • Forbidden system information • Values: <ul style="list-style-type: none"> – 0x00 - System is not a forbidden system – 0x01 - System is a forbidden system

8.144.2 Field Documentation

8.144.2.1 **BYTE** detailSvcInfo::hdrHybrid8.144.2.2 **BYTE** detailSvcInfo::hdrSrvStatus8.144.2.3 **BYTE** detailSvcInfo::isSysForbidden8.144.2.4 **BYTE** detailSvcInfo::srvCapability8.144.2.5 **BYTE** detailSvcInfo::srvStatus

8.145 DeviceConfigDetail Struct Reference

Data Fields

- [BYTE Technology](#)
- [BYTE QLIC](#)
- [BYTE Chipset](#)
- [BYTE HWVersion](#)

8.145.1 Detailed Description

This structure contains Device Configuration Details

Parameters

<i>Technology</i>	<ul style="list-style-type: none"> Current technology in use <ul style="list-style-type: none"> 0x00 - 1x RTT 0x01 - EVDO Rev 0 0x02 - EVDO Rev A 0xFF - NA
<i>QLIC</i>	<ul style="list-style-type: none"> Quasi Linear Interference Cancellation <ul style="list-style-type: none"> 0x00 - Not supported 0x01 - Supported
<i>Chipset</i>	<ul style="list-style-type: none"> Qualcomm platform <ul style="list-style-type: none"> 0x4E - MDM6200 0x4F - MDM6600 0xFF - NA
<i>HWVersion</i>	<ul style="list-style-type: none"> Hardware version <ul style="list-style-type: none"> 0x00 - BSHWREV0 0x01 - BSHWREV1 0x02 - BSHWREV2 0x03 - BSHWREV3 0x04 - BSHWREVMAX 0xFF - BSHWREVUNKNOWN
Generated by Doxygen	

8.145.2 Field Documentation

8.145.2.1 **BYTE** DeviceConfigDetail::Chipset

8.145.2.2 **BYTE** DeviceConfigDetail::HWVersion

8.145.2.3 **BYTE** DeviceConfigDetail::QLIC

8.145.2.4 **BYTE** DeviceConfigDetail::Technology

8.146 DHCPOption Struct Reference

Data Fields

- [BYTE optCode](#)
- [BYTE optValLen](#)
- [BYTE * pOptVal](#)

8.146.1 Detailed Description

This structure contains DHCPv4 lease option values

Parameters

<i>optCode</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – Option code
<i>optValLen</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – Option value length
<i>pOptValue</i>	<ul style="list-style-type: none"> • Val <ul style="list-style-type: none"> – Option value

8.146.2 Field Documentation

8.146.2.1 **BYTE** DHCPOption::optCode

8.146.2.2 **BYTE** DHCPOption::optValLen

8.146.2.3 **BYTE*** DHCPOption::pOptVal

8.147 DHCPOptionList Struct Reference

Data Fields

- [BYTE numOpt](#)
- [DHCPOption * pOptions](#)

8.147.1 Detailed Description

This structure contains DHCPv4 lease option list

Parameters

<i>optListSize</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – Size of Option List
<i>pOptions</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – Options

8.147.2 Field Documentation

8.147.2.1 BYTE DHCPOptionList::numOpt

8.147.2.2 DHCPOption* DHCPOptionList::pOptions

8.148 diagInfo Struct Reference

Data Fields

- [BYTE diagInfoLen](#)
- [BYTE diagnosticInfo \[255\]](#)

8.148.1 Detailed Description

This structure contains Diagnostic Information

Parameters

<i>diagInfoLen</i>	<ul style="list-style-type: none"> • Provides the length of information which follow. • If zero(0) then no further information exists.
<i>diagnosticInfo</i> <i>MAX_DESCRIP</i> <i>TION_LENGTH</i> <i>TH]</i>	<ul style="list-style-type: none"> • Diagnostic information.

8.148.2 Field Documentation

8.148.2.1 **BYTE** diagInfo::diagInfoLen

8.148.2.2 **BYTE** diagInfo::diagnosticInfo[255]

8.149 dirNum Struct Reference

Data Fields

- [BYTE](#) dirNumLen
- [BYTE](#) dirNum [255]

8.149.1 Detailed Description

This structure contains the parameters for Directory Number Information

Parameters

<i>dirNumLen</i>	<ul style="list-style-type: none"> • Number of sets of the following elements: <ul style="list-style-type: none"> – dir_num • If zero(0), then no information follows.
<i>dirNum</i>	<ul style="list-style-type: none"> • Directory number in ASCII characters.

8.149.2 Field Documentation

8.149.2.1 **BYTE** dirNum::dirNum[255]

8.149.2.2 **BYTE** dirNum::dirNumLen

8.150 dms_ActivationStatusTlv Struct Reference

Data Fields

- uint16_t [TlvPresent](#)
- uint32_t [activationStatus](#)

8.150.1 Detailed Description

Activation Status Tlv

Parameters

<i>activationStatus</i>	<ul style="list-style-type: none">• Service Activation Code<ul style="list-style-type: none">– 0 - Service not activated– 1 - Service activated– 2 - Activation connecting– 3 - Activation connected– 4 - OTASP security authenticated– 5 - OTASP NAM downloaded– 6 - OTASP MDN downloaded– 7 - OTASP IMSI downloaded– 8 - OTASP PRL downloaded– 9 - OTASP SPC downloaded– 10 - OTASP settings committed
-------------------------	--

8.150.2 Field Documentation

8.150.2.1 uint32_t dms_ActivationStatusTlv::activationStatus

8.150.2.2 uint16_t dms_ActivationStatusTlv::TlvPresent

8.151 dms_OperatingModeTlv Struct Reference

Data Fields

- uint16_t [TlvPresent](#)
- uint32_t [operatingMode](#)

8.151.1 Detailed Description

Operating Mode Tlv

Parameters

<i>operatingMode</i>	<ul style="list-style-type: none">• 0 - Online• 1 - Low power• 2 - Factory test mode• 3 - Offline• 4 - Resetting• 5 - Shutting down• 6 - Persistent low power• 7 - Mode-only low power
----------------------	---

8.151.2 Field Documentation

8.151.2.1 uint32_t dms_OperatingModeTlv::operatingMode

8.151.2.2 uint16_t dms_OperatingModeTlv::TlvPresent

8.152 dmsCurrentPRLInfo Struct Reference

Data Fields

- WORD * pPRLVersion
- BYTE * pPRLPreference

8.152.1 Detailed Description

This structure contains GetCurrentPRLInfo response parameter

Parameters

<i>pPRLVersion</i> [↔ OUT]	- Optional <ul style="list-style-type: none"> • PRL version of device.
<i>pPRLPreference</i>	[OUT]- Optional <ul style="list-style-type: none"> • PRL Preference <ul style="list-style-type: none"> – 0 - Unset – 1 - Set

8.152.2 Field Documentation

8.152.2.1 BYTE* dmsCurrentPRLInfo::pPRLPreference

8.152.2.2 WORD* dmsCurrentPRLInfo::pPRLVersion

8.153 DMScustSettingInfo Struct Reference

Data Fields

- uint16_t id_length
- uint8_t cust_id [64+1]
- uint16_t value_length
- uint8_t cust_value [8+1]
- uint16_t cust_attr

8.153.1 Detailed Description

This structure contains information about Customization Setting. This TLV is only applicable for 9x30 modules so far

Parameters

<i>id_length</i>	<ul style="list-style-type: none"> length of cust_id field
<i>cust_id</i>	<ul style="list-style-type: none"> Customization ID (Maximum 64 bytes)
<i>value_length</i>	<ul style="list-style-type: none"> length of cust_value field
<i>cust_value</i>	<ul style="list-style-type: none"> Customization Setting Value (Maximum 8 bytes)
<i>cust_attr</i>	<ul style="list-style-type: none"> Customization Setting attribute through QMI <ul style="list-style-type: none"> bit 0: Values: <ul style="list-style-type: none"> * 0 - read only * 1 - read/write

8.153.2 Field Documentation

8.153.2.1 `uint16_t DMScustSettingInfo::cust_attr`

8.153.2.2 `uint8_t DMScustSettingInfo::cust_id[64+1]`

8.153.2.3 `uint8_t DMScustSettingInfo::cust_value[8+1]`

8.153.2.4 `uint16_t DMScustSettingInfo::id_length`

8.153.2.5 `uint16_t DMScustSettingInfo::value_length`

8.154 DMScustSettingList Struct Reference

Data Fields

- `uint8_t list_type`
- `uint16_t num_instances`
- `DMScustSettingInfo custSetting [255+1]`

8.154.1 Detailed Description

This structure contains the fields of TLV Customization Setting List. This TLV is only applicable for 9x30 modules so far

Parameters

<i>list_type</i>	<ul style="list-style-type: none"> list type requested
<i>num_instances</i>	<ul style="list-style-type: none"> number of instances of customization setting
<i>custSetting</i>	<ul style="list-style-type: none"> See custSettingInfo for more information

8.154.2 Field Documentation

8.154.2.1 **DMScustSettingInfo** DMScustSettingList::custSetting[255+1]

8.154.2.2 **uint8_t** DMScustSettingList::list_type

8.154.2.3 **uint16_t** DMScustSettingList::num_instances

8.155 DMSgetCustomFeatureV2 Struct Reference

Data Fields

- [DMSgetCustomInput](#) * [pGetCustomInput](#)
- [DMScustSettingInfo](#) * [pCustSettingInfo](#)
- [DMScustSettingList](#) * [pCustSettingList](#)

8.155.1 Detailed Description

This struture contains the TLV required to get the Customization Info and customization list.

Parameters

--	--

8.155.2 Field Documentation

8.155.2.1 **DMScustSettingInfo*** DMSgetCustomFeatureV2::pCustSettingInfo

8.155.2.2 **DMScustSettingList*** DMSgetCustomFeatureV2::pCustSettingList

8.155.2.3 **DMSgetCustomInput*** DMSgetCustomFeatureV2::pGetCustomInput

8.156 DMSgetCustomInput Struct Reference

Data Fields

- uint8_t [cust_id](#) [64+1]
- uint8_t [list_type](#)

8.156.1 Detailed Description

This structure contains which customization id or the list type want to retrieve from modem. This TLV is only applicable for 9x30 modules so far

Parameters

<i>cust_id</i>	<ul style="list-style-type: none"> • Customization ID (Maximum 64 bytes)
<i>list_type</i>	<ul style="list-style-type: none"> • list type requested

8.156.2 Field Documentation

8.156.2.1 uint8_t DMSgetCustomInput::cust_id[64+1]

8.156.2.2 uint8_t DMSgetCustomInput::list_type

8.157 dmsIndicationRegisterReq Struct Reference

Data Fields

- [BYTE](#) * [pSwiGetResetInd](#)

8.157.1 Detailed Description

This structure contains the SLQSDmsSwiIndicationRegister request parameters.

Parameters

--	--

8.157.2 Field Documentation

8.157.2.1 [BYTE](#)* dmsIndicationRegisterReq::pSwiGetResetInd

8.158 dmsSwiGetResetInfo Struct Reference

Data Fields

- [BYTE type](#)
- [BYTE source](#)

8.158.1 Detailed Description

This structure contains the TLV required to Get Reset Info.

Parameters

--	--

8.158.2 Field Documentation

8.158.2.1 **BYTE** dmsSwiGetResetInfo::source

8.158.2.2 **BYTE** dmsSwiGetResetInfo::type

8.159 Domain Struct Reference

Data Fields

- [WORD domainLen](#)
- [CHAR domainName](#) [256]

8.159.1 Detailed Description

This structure contains the DomainName Information

Parameters

<i>domainLen</i>	<ul style="list-style-type: none">• length of the received Domain name
<i>domainName</i>	<ul style="list-style-type: none">• Domain name(Max 256 characters)

8.159.2 Field Documentation

8.159.2.1 **WORD** Domain::domainLen

8.159.2.2 **CHAR** Domain::domainName[256]

8.160 DomainNameList Struct Reference

Data Fields

- [BYTE numInstances](#)
- struct [Domain domain](#) [10]

8.160.1 Detailed Description

This structure contains the [DomainNameList](#) Information

Parameters

<i>numInstances</i>	<ul style="list-style-type: none">• Number of Domain name received
<i>domain</i>	<ul style="list-style-type: none">• Domain name information(Max 10 Domain names)

8.160.2 Field Documentation

8.160.2.1 struct [Domain](#) [DomainNameList::domain](#)[10]

8.160.2.2 [BYTE](#) [DomainNameList::numInstances](#)

8.161 DRCPParams Struct Reference

Data Fields

- [BYTE DRCValue](#)
- [BYTE DRCCover](#)

8.161.1 Detailed Description

This structure contains Data Rate Channel parameters

Parameters

<i>DRCValue</i>	<ul style="list-style-type: none">• Current Data Rate Channel
<i>DRCCover</i>	<ul style="list-style-type: none">• Current Data Rate Channel cover

8.161.2 Field Documentation

8.161.2.1 BYTE DRCPParams::DRCCover

8.161.2.2 BYTE DRCPParams::DRCValue

8.162 DTMFInfo Struct Reference

Data Fields

- [BYTE callID](#)
- [BYTE DTMFEvent](#)
- [BYTE digitCnt](#)
- [BYTE digitBuff](#) [255]

8.162.1 Detailed Description

This structure contains information about the DTMF (Dual Tone Multi-Frequency).

Parameters

<i>callID</i>	<ul style="list-style-type: none"> • Call identifier for the current call.
<i>DTMFEvent</i>	<ul style="list-style-type: none"> • DTMF event <ul style="list-style-type: none"> – 0x00 - DTMF_EVENT_REV_BURST - Sends a CDMA-burst DTMF – 0x01 - DTMF_EVENT_REV_START_CONT - Starts a continuous DTMF tone – 0x03 - DTMF_EVENT_REV_STOP_CONT - Stops a continuous DTMF tone – 0x05 - DTMF_EVENT_FWD_BURST - Received a CDMA-burst DTMF message – 0x06 - DTMF_EVENT_FWD_START_CONT - Received a start-continuous DTMF tone order – 0x07 - DTMF_EVENT_FWD_STOP_CONT - Received a stop-continuous DTMF tone order
<i>digitCnt</i>	<ul style="list-style-type: none"> • Number of set of following element i.e. digitBuff.
<i>digitBuff</i> [MAX_ _DESCRIPTI ON_LENGTH]	<ul style="list-style-type: none"> • DTMF digit buffer in ASCII string which is NULL terminated

8.162.2 Field Documentation

8.162.2.1 BYTE DTMFInfo::callID

8.162.2.2 BYTE DTMFInfo::digitBuff[255]

8.162.2.3 BYTE DTMFInfo::digitCnt

8.162.2.4 BYTE DTMFInfo::DTMFEvent

8.163 DTMFLengths Struct Reference

Data Fields

- [BYTE DTMFPulseWidth](#)
- [BYTE DTMFInterdigitInterval](#)

8.163.1 Detailed Description

This structure contains Voice Burst DTMF pulse length information

Parameters

<i>DTMFPulse↔ Width</i>	<ul style="list-style-type: none"> DTMF pulse width. Values: <ul style="list-style-type: none"> 0x00 - DTMF_ONLENGTH_95MS - 95 ms 0x01 - DTMF_ONLENGTH_150MS - 150 ms 0x02 - DTMF_ONLENGTH_200MS - 200 ms 0x03 - DTMF_ONLENGTH_250MS - 250 ms 0x04 - DTMF_ONLENGTH_300MS - 300 ms 0x05 - DTMF_ONLENGTH_350MS - 350 ms 0x06 - DTMF_ONLENGTH_SMS SMS Tx special pulse width
<i>DTMF↔ InterdigitInterval</i>	<ul style="list-style-type: none"> DTMF interdigit interval Values: <ul style="list-style-type: none"> 0x00 - DTMF_OFFLENGTH_60MS - 60 ms 0x01 - DTMF_OFFLENGTH_100MS - 100 ms 0x02 - DTMF_OFFLENGTH_150MS - 150 ms 0x03 - DTMF_OFFLENGTH_200MS - 200 ms

8.163.2 Field Documentation

8.163.2.1 BYTE DTMFLengths::DTMFInterdigitInterval

8.163.2.2 BYTE DTMFLengths::DTMFPulseWidth

8.164 DUNCallInfoInd Struct Reference

Data Fields

- [BYTE MdmConnStatus](#)

- [WORD CallEndReason](#)
- [ULONGLONG TXOKBytesCount](#)
- [ULONGLONG RXOKBytesCount](#)
- [BYTE DormancyStatus](#)
- [BYTE DataBearerTech](#)
- [channelRate ChannelRate](#)

8.164.1 Field Documentation

8.164.1.1 **WORD** DUNCallInfoInd::CallEndReason

8.164.1.2 **channelRate** DUNCallInfoInd::ChannelRate

8.164.1.3 **BYTE** DUNCallInfoInd::DataBearerTech

8.164.1.4 **BYTE** DUNCallInfoInd::DormancyStatus

8.164.1.5 **BYTE** DUNCallInfoInd::MdmConnStatus

8.164.1.6 **ULONGLONG** DUNCallInfoInd::RXOKBytesCount

8.164.1.7 **ULONGLONG** DUNCallInfoInd::TXOKBytesCount

8.165 dunchannelRate Struct Reference

Data Fields

- [uint32_t CurrChanTxRate](#)
- [uint32_t CurrChanRxRate](#)
- [uint32_t MaxChanTxRate](#)
- [uint32_t MaxChanRxRate](#)

8.165.1 Detailed Description

Parameters

<i>CurrChanTxRate</i>	instantaneous channel Tx rate in bits per second
<i>CurrChanRx↔ Rate</i>	instantaneous channel Rx rate in bits per second
<i>MaxChanTxRate</i>	maximum Tx rate that can be assigned to the device
<i>MaxChanRx↔ Rate</i>	maximum Rx rate that can be assigned to the device

8.165.2 Field Documentation

8.165.2.1 **uint32_t** dunchannelRate::CurrChanRxRate

8.165.2.2 `uint32_t dunchannelRate::CurrChanTxRate`

8.165.2.3 `uint32_t dunchannelRate::MaxChanRxRate`

8.165.2.4 `uint32_t dunchannelRate::MaxChanTxRate`

8.166 `ecioListElement` Struct Reference

Data Fields

- [SHORT `ecio`](#)
- [BYTE `radiolf`](#)

8.166.1 Detailed Description

This structure contains the ECIO Information

Parameters

<i>ecio</i>	<ul style="list-style-type: none">• ECIO value in dBm
<i>radiolf</i>	<ul style="list-style-type: none">• Radio interface technology of the signal being measured<ul style="list-style-type: none">– 0x00 - RADIO_IF_NO_SVC - None (no service)– 0x01 - RADIO_IF_CDMA_1X - cdma2000 1X– 0x02 - RADIO_IF_CDMA_1XEVD0 - cdma2000 HRPD (1xEV-DO)– 0x03 - RADIO_IF_AMPS - AMPS– 0x04 - RADIO_IF_GSM - GSM– 0x05 - RADIO_IF_UMTS - UMTS

8.166.2 Field Documentation

8.166.2.1 **SHORT** `ecioListElement::ecio`

8.166.2.2 **BYTE** `ecioListElement::radiolf`

8.167 `ECIOThresh` Struct Reference

Data Fields

- [BYTE `ECIOThresListLen`](#)
- [SHORT * `pECIOThresList`](#)

8.167.1 Detailed Description

This structure contains ECIO threshold related parameters.

Parameters

<i>ECIOThresListLen</i>	<ul style="list-style-type: none"> Length of the ECIO threshold list parameter to follow
<i>pECIOThresList</i>	<ul style="list-style-type: none"> Sequence of thresholds delimiting ECIO event reporting bands Every time a new ECIO value crosses a threshold value, an event report indication message with the new ECIO value is sent to the requesting control point. For this field <ul style="list-style-type: none"> Each ECIO threshold value is a signed 2 byte value Each ECIO threshold value increments in negative 0.5 dB, e.g., an ECIO threshold value of 2 means -1dB Maximum number of threshold values is 16 At least one value must be specified Threshold values specified above are used for all RATs

8.167.2 Field Documentation

8.167.2.1 **BYTE** ECIOThresh::ECIOThresListLen

8.167.2.2 **SHORT*** ECIOThresh::pECIOThresList

8.168 ECTNum Struct Reference

Data Fields

- [BYTE](#) ECTCallState
- [BYTE](#) presentationInd
- [BYTE](#) number [81]

8.168.1 Detailed Description

Contains the parameters passed for Explicit Communication Transfer by the device.

Parameters

<i>ECTCallState</i>	<ul style="list-style-type: none"> ECT call state: <ul style="list-style-type: none"> 0x00 - ECT_CALL_STATE_NONE - None 0x01 - ECT_CALL_STATE_ALERTING - Alerting 0x02 - ECT_CALL_STATE_ACTIVE - Active
---------------------	--

<i>presentationInd</i>	<ul style="list-style-type: none"> • Presentation indicator <ul style="list-style-type: none"> – 0x00 - presentationAllowedAddress – 0x01 - presentationRestricted – 0x02 - numberNotAvailable – 0x04 - presentationRestrictedAddress
<i>number</i>	<ul style="list-style-type: none"> • Number in ASCII characters terminated by NULL

8.168.2 Field Documentation

8.168.2.1 **BYTE** ECTNum::ECTCallState

8.168.2.2 **BYTE** ECTNum::number[81]

8.168.2.3 **BYTE** ECTNum::presentationInd

8.169 encryptedPIN1 Struct Reference

Data Fields

- [BYTE pin1Len](#)
- [BYTE pin1Val](#) [255]

8.169.1 Detailed Description

This structure contains the encrypted PIN1 Information.

Parameters

<i>pin1Len</i>	<ul style="list-style-type: none"> • Number of sets of the following elements ie encrypted PIN1 value. • If zero(0), no information follows.
<i>pin1Val</i>	<ul style="list-style-type: none"> • Encrypted PIN1 value.

Note

This value is returned only when PIN1 is enabled successfully and the feature is supported.

8.169.2 Field Documentation

8.169.2.1 BYTE encryptedPIN1::pin1Len

8.169.2.2 BYTE encryptedPIN1::pin1Val[255]

8.170 eriDataparams Struct Reference

Data Fields

- uint16_t eriDataLen
- uint8_t eriData [1024]

8.170.1 Field Documentation

8.170.1.1 uint8_t eriDataparams::eriData[1024]

8.170.1.2 uint16_t eriDataparams::eriDataLen

8.171 ERIFileparams Struct Reference

Data Fields

- WORD * pFileSize
- BYTE * pFile

8.171.1 Detailed Description

This structure contains Extended Roaming Indicator(ERI) file parameters

Parameters

<i>pFileSize</i> [IN/OUT] <i>UT</i>	<ul style="list-style-type: none"> • Upon input, the maximum number of bytes that file contents array can contain. • Upon successful output, actual number of bytes written to file contents array
<i>pFile</i> [OUT]	<ul style="list-style-type: none"> • ERI data read from persistent storage(Max size is 1024)

8.171.2 Field Documentation

8.171.2.1 BYTE* ERIFileparams::pFile

8.171.2.2 WORD* ERIFileparams::pFileSize

8.172 errorRateListElement Struct Reference

Data Fields

- [USHORT errorRate](#)
- [BYTE radiolf](#)

8.172.1 Detailed Description

This structure contains the Error Rate Information

Parameters

<i>errorRate</i>	<ul style="list-style-type: none"> • Error rate value corresponds to the RAT that is currently registered. <ul style="list-style-type: none"> – For CDMA, the error rate reported is Frame Error Rate: <ul style="list-style-type: none"> * Valid error rate values between 1 and 10000 are returned to indicate percentage, e.g., a value of 300 means the error rate is 3% * A value of 0xFFFF indicates that the error rate is unknown or unavailable – For HDR, the error rate reported is Packet Error Rate: <ul style="list-style-type: none"> * Valid error rate values between 1 and 10000 are returned to indicate percentage, e.g., a value of 300 means the error rate is 3% * A value of 0xFFFF indicates that the error rate is unknown or unavailable – For GSM, the error rate reported is Bit Error Rate: <ul style="list-style-type: none"> * Valid values are 0, 100, 200, 300, 400, 500, 600, and 700 The reported value divided by 100 gives the error rate as an RxQual value, e.g., a value of 300 represents an RxQual value of 3. * A value of 25500 indicates No Data – For WCDMA, the error rate reported is Block Error Rate (BLER): <ul style="list-style-type: none"> * Valid values are 1 to 10000 * The reported value divided by 100 provides the error rate in percentages, e.g., a value of 300 represents a BLER of 3%. * A value of 0 indicates No Data
<i>radiolf</i>	<ul style="list-style-type: none"> • Radio interface technology of the signal being measured <ul style="list-style-type: none"> – 0x00 - RADIO_IF_NO_SVC - None (no service) – 0x01 - RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - RADIO_IF_CDMA_1xEVDO - cdma2000 HRPD (1xEV-DO) – 0x03 - RADIO_IF_AMPS - AMPS – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS

8.172.2 Field Documentation

8.172.2.1 USHORT errorRateListElement::errorRate

8.172.2.2 BYTE errorRateListElement::radiolf

8.173 eTWSPLMNInfoTlv Struct Reference

Data Fields

- [uint8_t TlvPresent](#)
- [sMSEtwsPlmnInfo](#) [ETWSPLMNInfo](#)

8.173.1 Detailed Description

Parameters

<i>TlvPresent</i>	<ul style="list-style-type: none"> • Boolean indicating the presence of the TLV in the QMI response
<i>ETWSPLMNInfo</i>	<ul style="list-style-type: none"> • ETWS PLMN Information • See sMSEtwsPlmnInfo for more information

8.173.2 Field Documentation

8.173.2.1 [sMSEtwsPlmnInfo](#) [eTWSPLMNInfoTlv::ETWSPLMNInfo](#)

8.173.2.2 [uint8_t](#) [eTWSPLMNInfoTlv::TlvPresent](#)

8.174 extDispRecInfo Struct Reference

Data Fields

- [BYTE](#) [dispType](#)
- [BYTE](#) [extDispInfoLen](#)
- [BYTE](#) [extDispInfo](#) [255]

8.174.1 Detailed Description

This structure contains Line Control Information

Parameters

<i>dispType</i>	<ul style="list-style-type: none"> • Values are per [S1, Table 3.7.5.16-1].
<i>extDispInfoLen</i>	<ul style="list-style-type: none"> • Number of sets of the following elements: <ul style="list-style-type: none"> – <code>ext_display_info</code>
<i>extDispInfo</i>	<ul style="list-style-type: none"> • Extended display information buffer containing the display record; refer to [S1, Section 3.7.5.16] for the format information of the buffer contents.
	Generated by Doxygen

8.174.2 Field Documentation

8.174.2.1 **BYTE** extDispRecInfo::dispType

8.174.2.2 **BYTE** extDispRecInfo::extDispInfo[255]

8.174.2.3 **BYTE** extDispRecInfo::extDispInfoLen

8.175 FactorySequenceNumber Struct Reference

Data Fields

- **BYTE** FSNumber [255]

8.175.1 Detailed Description

This structure used to store Factory Sequence Number parameter

Parameters

<i>FSNumber</i> [$O \leftrightarrow UT$]	<ul style="list-style-type: none">• Facorty Sequence Number• Maximum Length is 255 Bytes
--	---

8.175.2 Field Documentation

8.175.2.1 **BYTE** FactorySequenceNumber::FSNumber[255]

8.176 fileAttributes Struct Reference

Data Fields

- **WORD** fileSize
- **WORD** fileID
- **BYTE** fileType
- **WORD** recordSize
- **WORD** recordCount
- **BYTE** secRead
- **WORD** secReadMask
- **BYTE** secWrite
- **WORD** secWriteMask
- **BYTE** secIncrease
- **WORD** secIncreaseMask
- **BYTE** secDeactivate
- **WORD** secDeactivateMask
- **BYTE** secActivate
- **WORD** secActivateMask
- **WORD** rawLen
- **BYTE** rawValue [255]

8.176.1 Detailed Description

This structure contains the information about the File Attributes.

Parameters

<i>fileSize</i>	<ul style="list-style-type: none"> Indicates the size of the file.
<i>fileID</i>	<ul style="list-style-type: none"> Indicates the ID of the file.
<i>fileType</i>	<ul style="list-style-type: none"> Indicates the type of the file. <ul style="list-style-type: none"> 0 - Transparent 1 - Cyclic 2 - Linear fixed 3 - Dedicated file 4 - Master file
<i>recordSize</i>	<ul style="list-style-type: none"> Indicates the size of the records. Only for cyclic and linear fixed files
<i>recordCount</i>	<ul style="list-style-type: none"> Indicates the total no. of the records. Only for linear fixed files
<i>secRead</i>	<ul style="list-style-type: none"> Read security attributes. <ul style="list-style-type: none"> 0 - Always 1 - Never 2 - AND condition 3 - OR condition 4 - Single condition
<i>secReadMask</i>	<ul style="list-style-type: none"> Mask with read security attributes. This field is valid only when required by security attributes. <ul style="list-style-type: none"> Bit 0 - PIN1 Bit 1 - PIN2 Bit 2 - UPIN Bit 3 - ADM
<i>secWrite</i>	<ul style="list-style-type: none"> Write security attributes. <ul style="list-style-type: none"> 0 - Always 1 - Never 2 - AND condition 3 - OR condition 4 - Single condition

<i>secWriteMask</i>	<ul style="list-style-type: none"> • Mask with write security attributes. • This field is valid only when required by security attributes. <ul style="list-style-type: none"> – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
<i>secIncrease</i>	<ul style="list-style-type: none"> • Increase security attributes. <ul style="list-style-type: none"> – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
<i>secIncrease↔ Mask</i>	<ul style="list-style-type: none"> • Mask with increase security attributes. • This field is valid only when required by security attributes. <ul style="list-style-type: none"> – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
<i>secDeactivate</i>	<ul style="list-style-type: none"> • Deactivate security attributes. <ul style="list-style-type: none"> – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
<i>secDeactivate↔ Mask</i>	<ul style="list-style-type: none"> • Mask with deactivate security attributes. • This field is valid only when required by security attributes. <ul style="list-style-type: none"> – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
<i>secActivate</i>	<ul style="list-style-type: none"> • Activate security attributes. <ul style="list-style-type: none"> – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition

<i>secActivateMask</i>	<ul style="list-style-type: none"> • Mask with activate security attributes. • This field is valid only when required by security attributes. <ul style="list-style-type: none"> – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
<i>rawLen</i>	<ul style="list-style-type: none"> • Length of the following elements i.e. raw value.
<i>rawValue</i> [MAX↔ _DESCRIPTI↔ ON_LENGTH]	<ul style="list-style-type: none"> • Raw value of file attributes.

8.176.2 Field Documentation

8.176.2.1 WORD fileAttributes::fileID

8.176.2.2 WORD fileAttributes::fileSize

8.176.2.3 BYTE fileAttributes::fileType

8.176.2.4 WORD fileAttributes::rawLen

8.176.2.5 BYTE fileAttributes::rawValue[255]

8.176.2.6 WORD fileAttributes::recordCount

8.176.2.7 WORD fileAttributes::recordSize

8.176.2.8 BYTE fileAttributes::secActivate

8.176.2.9 WORD fileAttributes::secActivateMask

8.176.2.10 BYTE fileAttributes::secDeactivate

8.176.2.11 WORD fileAttributes::secDeactivateMask

8.176.2.12 BYTE fileAttributes::secIncrease

8.176.2.13 WORD fileAttributes::secIncreaseMask

8.176.2.14 BYTE fileAttributes::secRead

8.176.2.15 WORD fileAttributes::secReadMask

8.176.2.16 **BYTE** fileAttributes::secWrite

8.176.2.17 **WORD** fileAttributes::secWriteMask

8.177 fileInfo Struct Reference

Data Fields

- [WORD](#) fileID
- [BYTE](#) pathLen
- [WORD](#) path [255]

8.177.1 Detailed Description

This structure contains paramaters for file Information

Parameters

<i>fileID</i>	<ul style="list-style-type: none">• This is Identifier to SIM files; e.g. in UIM "6F07" is Identifier of IMSI File
<i>pathLen</i>	<ul style="list-style-type: none">• Length of file Path
<i>path</i>	<ul style="list-style-type: none">• Path value. This value must be the complete path of the file, which is a sequence block of 2 bytes (e.g., 0x3F00 0x7FFF).

8.177.2 Field Documentation

8.177.2.1 **WORD** fileInfo::fileID

8.177.2.2 **WORD** fileInfo::path[255]

8.177.2.3 **BYTE** fileInfo::pathLen

8.178 FirmwareUpdatStat Struct Reference

Data Fields

- [ULONG](#) ResCode
- [BYTE](#) * pImgType
- [ULONG](#) * pRefData
- [BYTE](#) * pRefStringLen
- [BYTE](#) * pRefString
- [BYTE](#) * pLogStringLen
- [BYTE](#) * pLogString

8.178.1 Detailed Description

This structure is used to store Firmware Update Status

Parameters

<i>ResCode[OUT]</i>	<ul style="list-style-type: none"> FW Update Result Code Values: <ul style="list-style-type: none"> 0x00000001 - Successful 0xFFFFFFFF - Unknown (due to power off reset after firmware update) 0x100000nn - File update errors while nn will be the exact error number: <ul style="list-style-type: none"> * 00 - General error 0x200000nn - NVUP update errors while nn will be the exact error number: <ul style="list-style-type: none"> * 00 - General error 0x40000nnn - FOTA update agent errors while nnn will be the exact error number: <ul style="list-style-type: none"> * 000 ~ 0FF - Insignia defined error code * 100 ~ 1FF - Sierra defined error code * See qaGobiApiTableFwDldErrorCodes.h for more detailed information 0x800000nn - FDT/SSDP reported errors while nn will be the exact error number <ul style="list-style-type: none"> * See qaGobiApiTableFwDldErrorCodes.h for more detailed information
<i>pImgType[OUT]</i>	<ul style="list-style-type: none"> Optional parameter Firmware image type that failed the update
<i>pRefData[OUT]</i>	<ul style="list-style-type: none"> Optional parameter Failed image reference data This is normally the offset of the image that caused the failure
<i>pRefString</i> <i>Len[IN/OUT]</i>	<ul style="list-style-type: none"> Length of Reference String parameter to follow As input parameter specifies length assigned to pRefString parameter As output parameter specifies length of actual value retrieved from the device
<i>pRefString[OUT]</i>	<ul style="list-style-type: none"> Optional parameter Failed image reference string. This is normally the partition name of the image that caused the failure if applicable.
<i>pLogString</i> <i>Len[IN/OUT]</i>	<ul style="list-style-type: none"> Length of Reference String parameter to follow As input parameter specifies length assigned to pRefString parameter As output parameter specifies length of actual value retrieved from the device
<i>pLogString</i> <i>[OUT]</i>	<ul style="list-style-type: none"> Optional parameter Failed image reference string. This is normally the partition name of the image that caused the failure if applicable.

8.178.2 Field Documentation

8.178.2.1 **BYTE*** FirmwareUpdatStat::pImgType

8.178.2.2 **BYTE*** FirmwareUpdatStat::pLogString

8.178.2.3 **BYTE*** FirmwareUpdatStat::pLogStringLen

8.178.2.4 **ULONG*** FirmwareUpdatStat::pRefData

8.178.2.5 **BYTE*** FirmwareUpdatStat::pRefString

8.178.2.6 **BYTE*** FirmwareUpdatStat::pRefStringLen

8.178.2.7 **ULONG** FirmwareUpdatStat::ResCode

8.179 FMSImageElement Struct Reference

Data Fields

- uint8_t [imageType](#)
- uint8_t [imageId](#) [16]
- uint8_t [buildIdLength](#)
- uint8_t [buildId](#) [100]

8.179.1 Detailed Description

This structure contains the Image Element information

Parameters

<i>imageType</i>	<ul style="list-style-type: none">• Type of image 0 - Modem 1 - PRI
<i>imageId</i>	<ul style="list-style-type: none">• Unique image identifier
<i>buildIdLength</i>	<ul style="list-style-type: none">• Length of the build ID string (may be zero)
<i>pBuildId</i>	<ul style="list-style-type: none">• Build ID ANSI string with length provided by the previous field

8.179.2 Field Documentation

8.179.2.1 `uint8_t FMSImageElement::buildId[100]`

8.179.2.2 `uint8_t FMSImageElement::buildIdLength`

8.179.2.3 `uint8_t FMSImageElement::imageId[16]`

8.179.2.4 `uint8_t FMSImageElement::imageType`

8.180 FMSImageIdElement Struct Reference

Data Fields

- `uint8_t storageIndex`
- `uint8_t failureCount`
- `uint8_t imageID [16]`
- `uint8_t buildIDLength`
- `uint8_t buildID [100]`

8.180.1 Detailed Description

This structure contains the Image ID list element Information

Parameters

<i>storageIndex</i>	<ul style="list-style-type: none"> • Index in storage where the image is located(a value of 0xFF indicates that the storage for this type of image is not relevant)
<i>failureCount</i>	<ul style="list-style-type: none"> • Number of consecutive write attempts to this storage index that have failed(a value of 0xFF indicates unspecified)
<i>imageID</i>	<ul style="list-style-type: none"> • Image unique identifier(max 16 chars.)
<i>buildIDLength</i>	<ul style="list-style-type: none"> • Length of the build ID string. If there is no build ID, this field will be 0 and no data will follow.
<i>buildID</i>	<ul style="list-style-type: none"> • String containing image build information(Max 100 characters)

8.180.2 Field Documentation

8.180.2.1 `uint8_t FMSImageIdElement::buildID[100]`

8.180.2.2 `uint8_t FMSImageIdElement::buildIDLength`

8.180.2.3 `uint8_t` FMSImageIDElement::failureCount

8.180.2.4 `uint8_t` FMSImageIDElement::imageID[16]

8.180.2.5 `uint8_t` FMSImageIDElement::storageIndex

8.181 FMSImageIDEntries Struct Reference

Data Fields

- `uint8_t` [imageType](#)
- `uint8_t` [maxImages](#)
- `uint8_t` [executingImage](#)
- `uint8_t` [imageIDSize](#)
- [FMSImageIDElement](#) [imageIDElement](#) [50]

8.181.1 Detailed Description

This structure contains the list entry Information

Parameters

<i>imageType</i>	<ul style="list-style-type: none"> • Type of image <ul style="list-style-type: none"> – 0 - Modem – 1 - PRI
<i>maxImages</i>	<ul style="list-style-type: none"> • Maximum number of images of this type that may be stored concurrently on the device
<i>executingImage</i>	<ul style="list-style-type: none"> • Index (into the next array) of image that is currently executing
<i>imageIDSize</i>	<ul style="list-style-type: none"> • The number of elements in the image ID list
<i>imageIDElement</i>	<ul style="list-style-type: none"> • Array of ImageIDElement Structure (Max 50 elements) • See FMSImageIDElement

8.181.2 Field Documentation

8.181.2.1 `uint8_t` FMSImageIDEntries::executingImage

8.181.2.2 [FMSImageIDElement](#) FMSImageIDEntries::imageIDElement[50]

8.181.2.3 `uint8_t FMSImageIDEntries::imageIDSize`

8.181.2.4 `uint8_t FMSImageIDEntries::imageType`

8.181.2.5 `uint8_t FMSImageIDEntries::maxImages`

8.182 FMSImageList Struct Reference

Data Fields

- `uint8_t listSize`
- `FMSImageIDEntries imageIDEntries [2]`

8.182.1 Detailed Description

This structure contains the Get Stored Images List

Parameters

<i>listSize</i>	<ul style="list-style-type: none"> • The number of elements in the image list
<i>imageIDEntries</i>	<ul style="list-style-type: none"> • Array of <code>ImageIDEntries</code> Structure (Max 2 entries)

8.182.2 Field Documentation

8.182.2.1 `FMSImageIDEntries FMSImageList::imageIDEntries[2]`

8.182.2.2 `uint8_t FMSImageList::listSize`

8.183 FMSPrefImageList Struct Reference

Data Fields

- `uint8_t listSize`
- `FMSImageElement listEntries [2]`

8.183.1 Detailed Description

This structure contains the Preference Image List information

Parameters

<i>listSize</i>	<ul style="list-style-type: none">• The number of elements in the image list
<i>pListEntries</i>	<ul style="list-style-type: none">• Array of Image entries with size provided by previous field• See FMSImageElement

8.183.2 Field Documentation

8.183.2.1 [FMSImageElement](#) FMSPrefImageList::listEntries[2]

8.183.2.2 `uint8_t` FMSPrefImageList::listSize

8.184 fwinfo_s Struct Reference

Data Fields

- [ULONG](#) FirmwareID
- [ULONG](#) Technology
- [ULONG](#) Carrier
- [ULONG](#) Region
- [ULONG](#) GPSCapability

8.184.1 Detailed Description

Gobi firmware image info structure

Parameters

<i>FirmwareID</i>	<ul style="list-style-type: none">• Firmware ID obtained from the firmware image
<i>Technology</i>	<ul style="list-style-type: none">• Technology (0xFFFFFFFF if unknown)
<i>Carrier</i>	<ul style="list-style-type: none">• Carrier (0xFFFFFFFF if unknown)
<i>Region</i>	<ul style="list-style-type: none">• Region (0xFFFFFFFF if unknown)
<i>GPSCapability</i>	<ul style="list-style-type: none">• GPS capability (0xFFFFFFFF if unknown)

8.184.2 Field Documentation

8.184.2.1 **ULONG** `fwinfo_s::Carrier`

8.184.2.2 **ULONG** `fwinfo_s::FirmwareID`

8.184.2.3 **ULONG** `fwinfo_s::GPSCapability`

8.184.2.4 **ULONG** `fwinfo_s::Region`

8.184.2.5 **ULONG** `fwinfo_s::Technology`

8.185 GERANInfo Struct Reference

Data Fields

- [ULONG](#) `cellID`
- [BYTE](#) `plmn` [3]
- [WORD](#) `lac`
- [WORD](#) `arfcn`
- [BYTE](#) `bsic`
- [ULONG](#) `timingAdvance`
- [WORD](#) `rxLev`
- [BYTE](#) `nmrInst`
- [nmrCellInfo](#) `insNmrCellInfo` [255]

8.185.1 Detailed Description

This structure contains information about the GERAN Network.

Parameters

<i>cellID</i>	<ul style="list-style-type: none"> • Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.
<i>plmn</i> [<i>PLMN_LEN</i> ↔ <i>ENGTH</i>]	<ul style="list-style-type: none"> • MCC/MNC information coded as octet 3, 4, and 5. • This field is ignored when <code>nmrCellID</code> is not present.
<i>lac</i>	<ul style="list-style-type: none"> • Location area code. • This field is ignored when <code>nmrCellID</code> is not present. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>arfcn</i>	<ul style="list-style-type: none"> • Absolute RF channel number. <ul style="list-style-type: none"> – 0xFFFF - Not Available

<i>bsic</i>	<ul style="list-style-type: none"> Base station identity code. <ul style="list-style-type: none"> 0xFF - Not Available
<i>timingAdvance</i>	<ul style="list-style-type: none"> Measured delay (in bit periods; 1 bit period = 48/13 microsecond) of access burst transmission on RACH or PRACH to the expected signal from an MS at zero distance under static channel conditions. <ul style="list-style-type: none"> 0xFFFFFFFF - Not Available
<i>rxLev</i>	<ul style="list-style-type: none"> Serving Cell Rx measurement. Values range between 0 and 63. Mapped to a measured signal level: <ul style="list-style-type: none"> Rxlev 0 is a signal strength less than -110 dBm Rxlev 1 is -110 dBm to -109 dBm Rxlev 2 is -109 dBm to -108 dBm ... Rxlev 62 is -49 dBm to -48 dBm Rxlev 63 is greater than -48 dBm 0xFFFF - Not Available
<i>nmlInst</i>	<ul style="list-style-type: none"> Provides the number of set of instances which follow. If 0(zero), then no information follows it.
<i>insNmrCellInfo</i> [MAX_DESCRIPTION_LENGTH]	<ul style="list-style-type: none"> See nmrCellInfo for more information.

8.185.2 Field Documentation

8.185.2.1 WORD GERANInfo::arfcn

8.185.2.2 BYTE GERANInfo::bsic

8.185.2.3 ULONG GERANInfo::cellID

8.185.2.4 nmrCellInfo GERANInfo::insNmrCellInfo[255]

8.185.2.5 WORD GERANInfo::lac

8.185.2.6 BYTE GERANInfo::nmlInst

8.185.2.7 BYTE GERANInfo::plmn[3]

8.185.2.8 WORD GERANInfo::rxLev

8.185.2.9 **ULONG** GERANInfo::timingAdvance

8.186 geranInstInfo Struct Reference

Data Fields

- [WORD](#) geranArfcn
- [BYTE](#) geranBsicNcc
- [BYTE](#) geranBsicBcc
- [SHORT](#) geranRssi

8.186.1 Detailed Description

This structure contains information about the GERAN Instances in UMTS Network.

Parameters

<i>geranArfcn</i>	<ul style="list-style-type: none"> • Absolute RF channel number.
<i>geranBsicNcc</i>	<ul style="list-style-type: none"> • Base station identity code network color code. • 0xFF indicates information is not present.
<i>geranBsicBcc</i>	<ul style="list-style-type: none"> • Base station identity code base station color code. • 0xFF indicates information is not present.
<i>geranRssi</i>	<ul style="list-style-type: none"> • Received signal strength indicator.

8.186.2 Field Documentation

8.186.2.1 **WORD** geranInstInfo::geranArfcn

8.186.2.2 **BYTE** geranInstInfo::geranBsicBcc

8.186.2.3 **BYTE** geranInstInfo::geranBsicNcc

8.186.2.4 **SHORT** geranInstInfo::geranRssi

8.187 getAllCallInformation Struct Reference

Data Fields

- [callInfo](#) Callinfo
- [BYTE](#) isEmpty
- [BYTE](#) ALS

8.187.1 Detailed Description

This structure contains information related to call state change.

Parameters

<i>Callinfo</i>	<ul style="list-style-type: none">• See callInfo for more information.
<i>isEmpty</i>	<ul style="list-style-type: none">• Multiparty indicator.<ul style="list-style-type: none">– 0x00 - False– 0x01 - True
<i>ALS</i>	<ul style="list-style-type: none">• Alternate Line Service line indicator.• Feature for supporting two different phone numbers on the same mobile device.<ul style="list-style-type: none">– 0x00 - ALS_LINE1 - Line 1 (default)– 0x01 - ALS_LINE2 - Line 2

8.187.2 Field Documentation

8.187.2.1 **BYTE** getAllCallInformation::ALS

8.187.2.2 **callInfo** getAllCallInformation::Callinfo

8.187.2.3 **BYTE** getAllCallInformation::isEmpty

8.188 getAllCallRmtPtyName Struct Reference

Data Fields

- [BYTE](#) callID
- [remotePartyName](#) RemotePartyName

8.188.1 Detailed Description

This structure contains information for All Call Remote Party Names

Parameters

<i>callID</i>	<ul style="list-style-type: none">• Unique call identifier for the call.
<i>RemoteParty↔ Name</i>	<ul style="list-style-type: none">• See remotePartyName for more information.

8.188.2 Field Documentation

8.188.2.1 **BYTE** `getAllCallRmtPtyName::callID`

8.188.2.2 **remotePartyName** `getAllCallRmtPtyName::RemotePartyName`

8.189 `getAllCallRmtPtyNum` Struct Reference

Data Fields

- [BYTE](#) `callID`
- [remotePartyNum](#) `RemotePartyNum`

8.189.1 Detailed Description

This structure contains information for All Call Remote Party Numbers

Parameters

<i>callID</i>	<ul style="list-style-type: none">• Unique call identifier for the call.
<i>RemotePartyNum</i>	<ul style="list-style-type: none">• See remotePartyNum for more information.

8.189.2 Field Documentation

8.189.2.1 **BYTE** `getAllCallRmtPtyNum::callID`

8.189.2.2 **remotePartyNum** `getAllCallRmtPtyNum::RemotePartyNum`

8.190 `GetAudioPathConfigReq` Struct Reference

Data Fields

- [BYTE](#) `Profile`
- [BYTE](#) `Item`

8.190.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig request parameters

Parameters

<i>Profile</i>	<ul style="list-style-type: none"> • Audio Profile <ul style="list-style-type: none"> – 0-9
<i>Item</i>	<ul style="list-style-type: none"> • Item <ul style="list-style-type: none"> – 0 - AV_EC – 1 - AV_NS – 2 - AV_TXVOL – 3 - AV_DTMFTXG – 4 - AV_CODEECSTG – 5 - AV_TXPCMIIRFLTR – 6 - AV_RXPCMIIRFLTR – 7 - AV_MICGAIN – 8 - AV_RXAGC – 9 - AV_TXAGC – 10 - AV_RXAGCLIST – 11 - AV_RXAVCLIST – 12 - AV_TXAGCLIST

8.190.2 Field Documentation

8.190.2.1 **BYTE** GetAudioPathConfigReq::Item

8.190.2.2 **BYTE** GetAudioPathConfigReq::Profile

8.191 GetAudioPathConfigResp Struct Reference

Data Fields

- **BYTE** * pECMode
- **BYTE** * pNSEnable
- **WORD** * pTXGain
- **WORD** * pDTMFTXGain
- **WORD** * pCodecSTGain
- **TXPCMIIRFiltr** * pTXPCMIIRFiltr
- **RXPCMIIRFiltr** * pRXPCMIIRFiltr
- **BYTE** * pMICGainSelect
- **BYTE** * pRXAVCAGCSwitch
- **BYTE** * pTXAVCSwitch
- **RXAGCList** * pRXAGCList
- **RXAVCList** * pRXAVCList
- **TXAGCList** * pTXAGCList

8.191.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig response parameters.

Parameters

<i>pECMode</i>	[Optional] <ul style="list-style-type: none"> • AV_EC <ul style="list-style-type: none"> – 0 - Echo cancellation off – 1 - Handset mode – 2 - Headset mode – 3 - Car kit mode – 4 - Speaker Mode
<i>pNSEnable</i>	[Optional] <ul style="list-style-type: none"> • AV_NS <ul style="list-style-type: none"> – 0 - Noise suppression off – 1 - Noise suppression on
<i>pTXGain</i>	[Optional] <ul style="list-style-type: none"> • AV_TXVOL <ul style="list-style-type: none"> – 0x0000 - 0xffff
<i>pDTMFTXGain</i>	[Optional] <ul style="list-style-type: none"> • AV_DTMFTXG <ul style="list-style-type: none"> – 0x0000 - 0xffff
<i>pCodecSTGain</i>	[Optional] <ul style="list-style-type: none"> • AV_CODECSTG <ul style="list-style-type: none"> – 0x0000 - 0xffff
<i>pTXPCMIIRFiltr</i>	[Optional] <ul style="list-style-type: none"> • See TXPCMIIRFiltr for more information
<i>pRXPCMIIRFiltr</i>	[Optional] <ul style="list-style-type: none"> • See RXPCMIIRFiltr for more information
<i>pMICGainSelect</i>	[Optional] <ul style="list-style-type: none"> • AV_MICGAIN
<i>pRXAVCAGC↔ Switch</i>	[Optional] <ul style="list-style-type: none"> • RX AVC/AGC Switch
<i>pTXAVCSwitch</i>	[Optional] <ul style="list-style-type: none"> • TX AVC Switch
<i>pRXAGCList</i>	[Optional] <ul style="list-style-type: none"> • See RXAGCList for more information
<i>pRXAVCList</i>	[Optional] <ul style="list-style-type: none"> • See RXAVCList for more information
<i>pTXAGCList</i>	[Optional] <ul style="list-style-type: none"> • See TXAGCList for more information

8.191.2 Field Documentation

- 8.191.2.1 **WORD*** GetAudioPathConfigResp::pCodecSTGain
- 8.191.2.2 **WORD*** GetAudioPathConfigResp::pDTMFTXGain
- 8.191.2.3 **BYTE*** GetAudioPathConfigResp::pECMode
- 8.191.2.4 **BYTE*** GetAudioPathConfigResp::pMICGainSelect
- 8.191.2.5 **BYTE*** GetAudioPathConfigResp::pNSEnable
- 8.191.2.6 **RXAGCList*** GetAudioPathConfigResp::pRXAGCList
- 8.191.2.7 **BYTE*** GetAudioPathConfigResp::pRXAVCAGCSwitch
- 8.191.2.8 **RXAVCList*** GetAudioPathConfigResp::pRXAVCList
- 8.191.2.9 **RXPCMIIRFiltr*** GetAudioPathConfigResp::pRXPCMIIRFiltr
- 8.191.2.10 **TXAGCList*** GetAudioPathConfigResp::pTXAGCList
- 8.191.2.11 **BYTE*** GetAudioPathConfigResp::pTXAVCSwitch
- 8.191.2.12 **WORD*** GetAudioPathConfigResp::pTXGain
- 8.191.2.13 **TXPCMIIRFiltr*** GetAudioPathConfigResp::pTXPCMIIRFiltr

8.192 GetAudioProfileReq Struct Reference

Data Fields

- [BYTE Generator](#)

8.192.1 Detailed Description

This structure contains the SLQSGetAudioProfile request parameters

Parameters

<i>Generator</i>	<ul style="list-style-type: none">• Audio Generator<ul style="list-style-type: none">– 0 - Voice– 1 - Key Beep– 2 - MIDI
------------------	--

8.192.2 Field Documentation

8.192.2.1 BYTE GetAudioProfileReq::Generator

8.193 GetAudioProfileResp Struct Reference

Data Fields

- [BYTE Profile](#)
- [BYTE EarMute](#)
- [BYTE MicMute](#)
- [BYTE Volume](#)

8.193.1 Detailed Description

This structure contains the SLQSGetAudioProfile response parameters.

Parameters

<i>Profile</i>	<ul style="list-style-type: none">• Audio Profile<ul style="list-style-type: none">– 0 - Handset– 1 - Headset– 2 - Car Kit– 3 - Speaker phone– 4 - Auxiliary– 5 - TTY– 6 - Auxiliary external PCM– 7 - Primary external PCM– 8 - External slave PCM– 9 - I2S
<i>EarMute</i>	<ul style="list-style-type: none">• Ear Mute Setting<ul style="list-style-type: none">– 0 - unmuted– 1 - muted
<i>MicMute</i>	<ul style="list-style-type: none">• MIC Mute Setting<ul style="list-style-type: none">– 0 - unmuted– 1 - muted
<i>Volume</i>	<ul style="list-style-type: none">• Audio Volume Level<ul style="list-style-type: none">– 0 to 7

8.193.2 Field Documentation

8.193.2.1 **BYTE** GetAudioProfileResp::EarMute

8.193.2.2 **BYTE** GetAudioProfileResp::MicMute

8.193.2.3 **BYTE** GetAudioProfileResp::Profile

8.193.2.4 **BYTE** GetAudioProfileResp::Volume

8.194 GetAudioVoITLBConfigReq Struct Reference

Data Fields

- [BYTE Profile](#)
- [BYTE Generator](#)
- [BYTE Volume](#)
- [BYTE Item](#)

8.194.1 Detailed Description

This structure contains the SLQSGetAudioVoITLBConfig request parameters

Parameters

<i>Profile</i>	<ul style="list-style-type: none">• Audio Profile<ul style="list-style-type: none">– 0-9
<i>Generator</i>	<ul style="list-style-type: none">• Audio Generator<ul style="list-style-type: none">– 0-2
<i>Volume</i>	<ul style="list-style-type: none">• Audio Volume Level<ul style="list-style-type: none">– 0-7
<i>Item</i>	<ul style="list-style-type: none">• Item<ul style="list-style-type: none">– 13 - AV_RXVOLDB– 14 - AV_DTMFVOLDB– 15 - AV_PAD

8.194.2 Field Documentation

8.194.2.1 **BYTE** GetAudioVoITLBConfigReq::Generator

8.194.2.2 **BYTE** GetAudioVoITLBConfigReq::Item

8.194.2.3 **BYTE** GetAudioVolTLBConfigReq::Profile

8.194.2.4 **BYTE** GetAudioVolTLBConfigReq::Volume

8.195 GetAudioVolTLBConfigResp Struct Reference

Data Fields

- [WORD](#) ResCode

8.195.1 Detailed Description

This structure contains the SLQSGetAudioVolTLBConfig response parameters.

Parameters

<i>ResCode</i>	<ul style="list-style-type: none"> • Result of requested item
----------------	--

8.195.2 Field Documentation

8.195.2.1 **WORD** GetAudioVolTLBConfigResp::ResCode

8.196 getCallFWExtInfo Struct Reference

Data Fields

- [BYTE](#) numInstances
- [callFWExtInfo](#) CallFWExtInfo [20]

8.196.1 Detailed Description

This structure contains an array of Call Forwarded Extended Information.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of callFWExtInfo that follow. • If zero(0) then no further information exists.
<i>CallFWExtInfo</i> [MAX_NO_OF_CALLS]	<ul style="list-style-type: none"> • Array of CallFWExtInfo. <ul style="list-style-type: none"> – See CallFWExtInfo for more information.

8.196.2 Field Documentation

8.196.2.1 `callFWExtInfo` `getCallFWExtInfo::CallFWExtInfo[20]`

8.196.2.2 `BYTE` `getCallFWExtInfo::numInstances`

8.197 getCallFWInfo Struct Reference

Data Fields

- `BYTE` `numInstances`
- `callFWInfo` `CallFWInfo` [20]

8.197.1 Detailed Description

This structure contains an array of Call Forwarded Information.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of <code>callFWInfo</code> that follow. • If zero(0) then no further information exists.
<i>CallFWInfo[M↔ AX_NO_OF_↔ CALLS]</i>	<ul style="list-style-type: none"> • Array of <code>callFWInfo</code>. <ul style="list-style-type: none"> – See <code>callFWInfo</code> for more information.

8.197.2 Field Documentation

8.197.2.1 `callFWInfo` `getCallFWInfo::CallFWInfo[20]`

8.197.2.2 `BYTE` `getCallFWInfo::numInstances`

8.198 getCustomFeatureV2 Struct Reference

Data Fields

- `getCustomInput` * `pGetCustomInput`
- `custSettingInfo` * `pCustSettingInfo`
- `custSettingList` * `pCustSettingList`

8.198.1 Detailed Description

This struture contains the TLV required to get the Customization Info and customization list.

Parameters

--	--

8.198.2 Field Documentation

8.198.2.1 `custSettingInfo*` `getCustomFeatureV2::pCustSettingInfo`

8.198.2.2 `custSettingList*` `getCustomFeatureV2::pCustSettingList`

8.198.2.3 `getCustomInput*` `getCustomFeatureV2::pGetCustomInput`

8.199 getCustomInput Struct Reference

Data Fields

- [CHAR](#) `cust_id` [64+1]
- [BYTE](#) `list_type`

8.199.1 Detailed Description

This structure contains which customization id or the list type want to retrieve from modem. This TLV is only applicable for 9x30 modules so far

Parameters

<i>cust_id</i>	<ul style="list-style-type: none"> • Customization ID (Maximum 64 bytes)
<i>list_type</i>	<ul style="list-style-type: none"> • list type requested

8.199.2 Field Documentation

8.199.2.1 `CHAR` `getCustomInput::cust_id`[64+1]

8.199.2.2 `BYTE` `getCustomInput::list_type`

8.200 getDUNCallInfoReq Struct Reference

Data Fields

- [ULONG](#) `Mask`
- [BYTE](#) * `pReportConnStatus`
- [TransferStatInd](#) * `pTransferStatInd`
- [BYTE](#) * `pReportDormStatus`
- [BYTE](#) * `pReportDataBearerTech`
- [BYTE](#) * `pReportChannelRate`

8.200.1 Detailed Description

This structure contains the DUN Call Info Request parameters.

Parameters

<i>Mask</i>	<ul style="list-style-type: none"> • Mandatory parameter • Set the bits corresponding to the information requested to 1 • All other bits must be set to 0. • If any values are not available or applicable, the corresponding TLVs are not returned in the response. <ul style="list-style-type: none"> – Bit 0 - Connection Status – Bit 1 - Last call end reason – Bit 2 - Tx/Rx bytes OK – Bit 3 - Dormancy status – Bit 4 - Data bearer – Bit 5 - Channel rate – Bit 6 - Call active duration
<i>pReportConn↔ Status</i>	<ul style="list-style-type: none"> • Connect Status Indicator <ul style="list-style-type: none"> – 0 - Do not report – 1 - Report connection status and call end reason
<i>pTransferStatInd</i>	<ul style="list-style-type: none"> • See TransferStatInd for more information
<i>pReportDorm↔ Status</i>	<ul style="list-style-type: none"> • Dormancy Status Indicator <ul style="list-style-type: none"> – 0 - Do not report – 1 - Report traffic channel state of interface used for data connection
<i>pReportData↔ BearerTech</i>	<ul style="list-style-type: none"> • Current Data Bearer Technology Indicator <ul style="list-style-type: none"> – 0 - Do not report – 1 - Report radio interface used for data transfer when it changes
<i>pReport↔ ChannelRate</i>	<ul style="list-style-type: none"> • Channel Rate Indicator <ul style="list-style-type: none"> – 0 - Do not report – 1 - Report channel rate

8.200.2 Field Documentation

8.200.2.1 **ULONG** getDUNCallInfoReq::Mask

8.200.2.2 **BYTE*** getDUNCallInfoReq::pReportChannelRate

8.200.2.3 **BYTE*** getDUNCallInfoReq::pReportConnStatus

8.200.2.4 **BYTE*** `getDUNCallInfoReq::pReportDataBearerTech`

8.200.2.5 **BYTE*** `getDUNCallInfoReq::pReportDormStatus`

8.200.2.6 **TransferStatInd*** `getDUNCallInfoReq::pTransferStatInd`

8.201 getDUNCallInfoResp Struct Reference

Data Fields

- [ConnectionStatus](#) * `pConnectionStatus`
- [WORD](#) * `pCallEndReason`
- [ULONGLONG](#) * `pTXOKBytesCount`
- [ULONGLONG](#) * `pRXOKBytesCount`
- [BYTE](#) * `pDormancyStatus`
- [BYTE](#) * `pDataBearerTech`
- [ChannelRate](#) * `pChannelRate`
- [ULONGLONG](#) * `pLastCallTXOKBytesCnt`
- [ULONGLONG](#) * `pLastCallRXOKBytesCnt`
- [ULONGLONG](#) * `pMdmCallDurationActive`
- [BYTE](#) * `pLastCallDataBearerTech`

8.201.1 Detailed Description

This structure contains the DUN Call Info response parameters

Parameters

<i>pConnection↔ Status</i>	<ul style="list-style-type: none"> • See ConnectionStatus for more information
<i>pCallEndReason</i>	<ul style="list-style-type: none"> • Last modem call end reason • See qaGobiApiTableCallEndReasons.h for Call End Reason • Only valid if the last call made was DUN, else zero is returned
<i>pTXOKBytes↔ Count</i>	<ul style="list-style-type: none"> • Number of bytes transmitted without error • Returned only if a data call is up
<i>pRXOKBytes↔ Count</i>	<ul style="list-style-type: none"> • Number of bytes received without error • Returned only if a data call is up
<i>pDormancy↔ Status</i>	<ul style="list-style-type: none"> • Current traffic channel status • Returned if a data call is up <ul style="list-style-type: none"> – 0x01 - Traffic channel dormant – 0x02 - Traffic channel active

<p><i>pDataBearer</i>↔ <i>Tech</i></p>	<ul style="list-style-type: none"> • Current data bearer technology • Returned only if a data call is up <ul style="list-style-type: none"> – 0x01 - cdma2000 1X – 0x02 - cdma2000 HRPD (1xEV-DO) – 0x03 - GSM – 0x04 - UMTS – 0x05 - cdma200 HRPD (1xEV-DO RevA) – 0x06 - EDGE – 0x07 - HSDPA and WCDMA – 0x08 - WCDMA and HSUPA – 0x09 - HSDPA and HSUPA – 0x0A - LTE – 0x0B - cdma2000 EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA – 0x10 - HSDPA+ and 64QAM – 0x11 - HSDPA+, 64QAM and HSUPA – 0x12 - TDSCDMA – 0x13 - TDSCDMA and HSDPA – 0xFF - Unknown
<p><i>pChannelRate</i></p>	<ul style="list-style-type: none"> • See ChannelRate for more information
<p><i>pLastCallTXO</i>↔ <i>KBytesCnt</i></p>	<ul style="list-style-type: none"> • Number of bytes transmitted without error during the last data call (0 if no call was made). • Return only if not in a call and the previous call was made using DUN.
<p><i>pLastCallRXO</i>↔ <i>KBytesCnt</i></p>	<ul style="list-style-type: none"> • Number of bytes transmitted without error during the last data call (0 if no call was made). • Return only if not in a call and the previous call was made using DUN.
<p><i>pMdmCall</i>↔ <i>DurationActive</i></p>	<ul style="list-style-type: none"> • Duration that the call is active in milliseconds • If the modem connection status is connected, this represents the active duration of the current DUN call • If the modem connection status is disconnected, this represents the duration of the last DUN call since the device was powered up (0 if no call has been made or if the last call was not DUN)

<p><i>pLastCallData</i>↔ <i>BearerTech</i></p>	<ul style="list-style-type: none"> • Last Call Data Bearer Technology • Returned only if not in a call and when the previous call was made using DUN <ul style="list-style-type: none"> – 0x01 - cdma2000 1X – 0x02 - cdma2000 HRPD (1xEV-DO) – 0x03 - GSM – 0x04 - UMTS – 0x05 - cdma200 HRPD (1xEV-DO Rev A) – 0x06 - EDGE – 0x07 - HSDPA and WCDMA – 0x08 - WCDMA and HSUPA – 0x09 - HSDPA and HSUPA – 0x0A - LTE – 0x0B - cdma2000 EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA – 0x10 - HSDPA+ and 64QAM – 0x11 - HSDPA+, 64QAM and HSUPA – 0x12 - TDSCDMA – 0x13 - TDSCDMA and HSDPA – 0xFF - Unknown
--	---

8.201.2 Field Documentation

8.201.2.1 **WORD*** `getDUNCallInfoResp::pCallEndReason`

8.201.2.2 **ChannelRate*** `getDUNCallInfoResp::pChannelRate`

8.201.2.3 **ConnectionStatus*** `getDUNCallInfoResp::pConnectionStatus`

8.201.2.4 **BYTE*** `getDUNCallInfoResp::pDataBearerTech`

8.201.2.5 **BYTE*** `getDUNCallInfoResp::pDormancyStatus`

8.201.2.6 **BYTE*** `getDUNCallInfoResp::pLastCallDataBearerTech`

8.201.2.7 **ULONGLONG*** `getDUNCallInfoResp::pLastCallRXOKBytesCnt`

8.201.2.8 **ULONGLONG*** `getDUNCallInfoResp::pLastCallTXOKBytesCnt`

8.201.2.9 **ULONGLONG*** `getDUNCallInfoResp::pMdmCallDurationActive`

8.201.2.10 **ULONGLONG*** `getDUNCallInfoResp::pRXOKBytesCount`

8.201.2.11 **ULONGLONG*** getDUNCallInfoResp::pTXOKBytesCount

8.202 getDyingGaspCfg Struct Reference

Data Fields

- **BYTE** * pDestSMSNum
- **BYTE** * pDestSMSContent

8.202.1 Detailed Description

This struture contains the TLV required to get the Dying GASP Config.

Parameters

--	--

8.202.2 Field Documentation

8.202.2.1 **BYTE*** getDyingGaspCfg::pDestSMSContent

8.202.2.2 **BYTE*** getDyingGaspCfg::pDestSMSNum

8.203 getDyingGaspStatistics Struct Reference

Data Fields

- **ULONG** * pTimeStamp
- **BYTE** * pSMSAttemptedFlag

8.203.1 Detailed Description

This struture contains the TLV required to get the Dying GASP Statistics.

Parameters

--	--

8.203.2 Field Documentation

8.203.2.1 **BYTE*** getDyingGaspStatistics::pSMSAttemptedFlag

8.203.2.2 **ULONG*** getDyingGaspStatistics::pTimeStamp

8.204 GetErrRateResp Struct Reference

Data Fields

- WORD * pCDMAFrameErrRate
- WORD * pHDRPackErrRate
- BYTE * pGSMBER
- BYTE * pWCDMABER

8.204.1 Detailed Description

This structure contains information about the SLQSGetErrorRate response parameters.

Parameters

<i>pCDMAFrameErrRate[Out]</i>	<ul style="list-style-type: none"> • CDMA Frame Error Rate • Valid error rate values between 1 and 10000 are returned to indicate the percentage, e.g., a value of 300 means the error rate is 3%. • A value of 0xFFFF indicates that the error rate is unknown/unavailable.
<i>pHDRPackErrRate[Out]</i>	<ul style="list-style-type: none"> • HDR Packet Error Rate • Valid error rate values between 1 and 10000 are returned to indicate the percentage, e.g., a value of 300 means the error rate is 3%. • A value of 0xFFFF indicates that the error rate is unknown/unavailable.
<i>pGSMBER[Out]</i>	<ul style="list-style-type: none"> • GSM Bit Error Rate • Valid error rate values between 1 and 100 are returned to indicate the percentage value. • A 0% block error rate (BLER) indicates No Data.
<i>pWCDMABER[Out]</i>	<ul style="list-style-type: none"> • WCDMA Block Error Rate • Valid error rate values between 1 and 100 are returned to indicate the percentage value. • A value of 0xFF indicates that the error rate is unknown/unavailable.

8.204.2 Field Documentation

8.204.2.1 WORD* GetErrRateResp::pCDMAFrameErrRate

8.204.2.2 BYTE* GetErrRateResp::pGSMBER

8.204.2.3 WORD* GetErrRateResp::pHDRPackErrRate

8.204.2.4 BYTE* GetErrRateResp::pWCDMABER

8.205 GetHRPDStatsResp Struct Reference

Data Fields

- [DRCParams](#) * [pDRCParams](#)
- [BYTE](#) * [pUATI](#)
- [PilotSetData](#) * [pPilotSetData](#)

8.205.1 Detailed Description

This structure contains information about the SLQSSwiGetHRPDStats response parameters.

Parameters

<i>pDRCParams</i> [↔ Out]	<ul style="list-style-type: none"> • See DRCParams for more information.
<i>pUATI</i> [Out]	<ul style="list-style-type: none"> • A 128-bit address that includes the access terminal identifier and subnet ID • Size must be 16 bytes
<i>pPilotSetData</i> [↔ Out]	<ul style="list-style-type: none"> • See PilotSetData for more information.

8.205.2 Field Documentation

8.205.2.1 **DRCParams*** GetHRPDStatsResp::pDRCParams

8.205.2.2 **PilotSetData*** GetHRPDStatsResp::pPilotSetData

8.205.2.3 **BYTE*** GetHRPDStatsResp::pUATI

8.206 GetIMSSMSConfigParams Struct Reference

Data Fields

- [BYTE](#) * [pSettingResp](#)
- [BYTE](#) * [pSMSFormat](#)
- [BYTE](#) * [pSMSOverIPNwInd](#)
- [BYTE](#) * [pPhoneCtxtURLen](#)
- [BYTE](#) * [pPhoneCtxtURI](#)

8.206.1 Detailed Description

This structure contains the SLQSGetIMSSMSConfig response parameters.

Parameters

<i>pSettingResp</i>	<ul style="list-style-type: none"> • Settings Response
<i>pSMSFormat</i>	<ul style="list-style-type: none"> • SMS format <ul style="list-style-type: none"> – 0 - 3GPP – 1 - 3GPP2
<i>pSMSOverIP↔ NwInd</i>	<ul style="list-style-type: none"> • SMS over IP Network Indication Flag <ul style="list-style-type: none"> – TRUE - Turn on mobile-originated SMS – FALSE - Turn off mobile-originated SMS
<i>pPhoneCtxtU↔ RILen[IN/OUT]</i>	<ul style="list-style-type: none"> • Size in bytes assigned to the Phone context Universal Resource Identifier to follow
<i>pPhoneCtxtURI</i>	<ul style="list-style-type: none"> • Phone context universal resource identifier • Length of this string must be specified in pPhoneCtxtURILen parameter

8.206.2 Field Documentation

8.206.2.1 **BYTE*** GetIMSSMSConfigParams::pPhoneCtxtURI

8.206.2.2 **BYTE*** GetIMSSMSConfigParams::pPhoneCtxtURILen

8.206.2.3 **BYTE*** GetIMSSMSConfigParams::pSettingResp

8.206.2.4 **BYTE*** GetIMSSMSConfigParams::pSMSFormat

8.206.2.5 **BYTE*** GetIMSSMSConfigParams::pSMSOverIPNwInd

8.207 GetIMSUserConfigParams Struct Reference

Data Fields

- **BYTE *** [pSettingResp](#)
- **BYTE *** [pIMSDomainLen](#)
- **BYTE *** [pIMSDomain](#)

8.207.1 Detailed Description

This structure contains the SLQSGetIMSUserConfig response parameters.

Parameters

<i>pSettingResp</i>	<ul style="list-style-type: none"> Settings Response
<i>pIMSDomainLen</i> [IN/OUT]	<ul style="list-style-type: none"> Length of IMS Domain Name to follow
<i>pIMSDomain</i>	<ul style="list-style-type: none"> IMS domain name Length of this string must be specified in pIMSDomainLen parameter

8.207.2 Field Documentation

8.207.2.1 **BYTE*** GetIMSUserConfigParams::pIMSDomain

8.207.2.2 **BYTE*** GetIMSUserConfigParams::pIMSDomainLen

8.207.2.3 **BYTE*** GetIMSUserConfigParams::pSettingResp

8.208 GetIMSVoIPConfigResp Struct Reference

Data Fields

- BYTE *** [pSettingResp](#)
- WORD *** [pSessionExpiryTimer](#)
- WORD *** [pMinSessionExpiryTimer](#)
- BYTE *** [pAmrWbEnable](#)
- BYTE *** [pScrAmrEnable](#)
- BYTE *** [pScrAmrWbEnable](#)
- BYTE *** [pAmrMode](#)
- WORD *** [pAmrWBMode](#)
- BYTE *** [pAmrOctetAligned](#)
- BYTE *** [pAmrWBOctetAligned](#)
- WORD *** [pRingingTimer](#)
- WORD *** [pRingBackTimer](#)
- WORD *** [pRTPRTCPInactTimer](#)

8.208.1 Detailed Description

This structure contains the SLQSGetIMSVoIPConfig request parameters.

Parameters

<i>pSettingResp</i>	<ul style="list-style-type: none"> Settings Response. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE
<i>pSession↔</i> <i>ExpiryTimer</i>	<ul style="list-style-type: none"> Session duration, in seconds
<i>pMinSession↔</i> <i>ExpiryTimer</i>	<ul style="list-style-type: none"> Minimum allowed value for session expiry timer, in seconds
<i>pAmrWbEnable</i>	<ul style="list-style-type: none"> Flag to enable/disable Adaptive Multirate Codec(AMR) WideBand(WB) audio Values: <ul style="list-style-type: none"> – True - Enable – False - Disable
<i>pScrAmrEnable</i>	<ul style="list-style-type: none"> Flag to enable/disable Source Control Rate(SCR) for AMR NarrowBand (NB) Values: <ul style="list-style-type: none"> – True - Enable – False - Disable
<i>pScrAmrWb↔</i> <i>Enable</i>	<ul style="list-style-type: none"> Flag to enable/disable SCR for AMR WB Audio Values: <ul style="list-style-type: none"> – True - Enable – False - Disable
<i>pAmrMode</i>	<ul style="list-style-type: none"> BitMask for AMR NB modes allowed Values: <ul style="list-style-type: none"> – 0x1 - 4.75 kbps – 0x2 - 5.15 kbps – 0x4 - 5.9 kbps – 0x8 - 6.17 kbps – 0x10 - 7.4 kbps – 0x20 - 7.95 kbps – 0x40 - 10.2 kbps – 0x80 - 12.2 kbps

<i>pAmrWBMode</i>	<ul style="list-style-type: none"> • BitMask for AMR WB modes allowed • Values: <ul style="list-style-type: none"> – 0x1 - 6.60 kbps – 0x2 - 8.85 kbps – 0x4 - 12.65 kbps – 0x8 - 14.25 kbps – 0x10 - 15.85 kbps – 0x20 - 18.25 kbps – 0x40 - 19.85 kbps – 0x80 - 23.05 kbps – 0x100 - 23.85 kbps
<i>pAmrOctet↔ Aligned</i>	<ul style="list-style-type: none"> • Flag to indicate if the octet is aligned for AMR NB Audio • Values: <ul style="list-style-type: none"> – True - Aligned – False - Not aligned, Bandwidth Efficient mode
<i>pAmrWBOctet↔ Aligned</i>	<ul style="list-style-type: none"> • Flag to indicate if the octet is aligned for AMR WB Audio • Values: <ul style="list-style-type: none"> – True - Aligned – False - Not aligned, Bandwidth Efficient mode
<i>pRingingTimer</i>	<ul style="list-style-type: none"> • Duration of ringing timer, in seconds. The ringing timer starts on the ringing event. If the call is not answered within the duration of this timer, the call is disconnected.
<i>pRingBackTimer</i>	<ul style="list-style-type: none"> • Duration of ringback timer, in seconds. The ringback timer starts on the ringback event. If the call is not answered within the duration of this timer, the call is disconnected.
<i>pRTPRTCP↔ InactTimer</i>	<ul style="list-style-type: none"> • Duration of RTP/RTCP inactivity timer, in seconds. If no RTP/RTCP packet is received prior to the expiry of this timer, the call is disconnected.

8.208.2 Field Documentation

8.208.2.1 **BYTE*** GetIMSVolIPConfigResp::pAmrMode

8.208.2.2 **BYTE*** GetIMSVolIPConfigResp::pAmrOctetAligned

8.208.2.3 **BYTE*** GetIMSVolIPConfigResp::pAmrWbEnable

8.208.2.4 **WORD*** GetIMSVolIPConfigResp::pAmrWBMode

8.208.2.5 **BYTE*** GetIMSVolIPConfigResp::pAmrWBOctetAligned

8.208.2.6 WORD* GetIMSVoIPConfigResp::pMinSessionExpiryTimer

8.208.2.7 WORD* GetIMSVoIPConfigResp::pRingBackTimer

8.208.2.8 WORD* GetIMSVoIPConfigResp::pRingingTimer

8.208.2.9 WORD* GetIMSVoIPConfigResp::pRTPRTCPlnactTimer

8.208.2.10 BYTE* GetIMSVoIPConfigResp::pScrAmrEnable

8.208.2.11 BYTE* GetIMSVoIPConfigResp::pScrAmrWbEnable

8.208.2.12 WORD* GetIMSVoIPConfigResp::pSessionExpiryTimer

8.208.2.13 BYTE* GetIMSVoIPConfigResp::pSettingResp

8.209 GetInstIDResp Struct Reference

Data Fields

- [BYTE * pInstanceID](#)
- [BYTE * pIPFamily](#)

8.209.1 Field Documentation

8.209.1.1 BYTE* GetInstIDResp::pInstanceID

8.209.1.2 BYTE* GetInstIDResp::pIPFamily

8.210 GetM2MAudioProfileReq Struct Reference

Data Fields

- [BYTE * pGenerator](#)

8.210.1 Detailed Description

This structure contains the SLQSGetM2MAudioProfile request parameters.

Parameters

$p \leftrightarrow$ <i>Generator[optional]</i>	<ul style="list-style-type: none"> • Generator <ul style="list-style-type: none"> – 0 - Voice
---	--

8.210.2 Field Documentation

8.210.2.1 BYTE* GetM2MAudioProfileReq::pGenerator

8.211 GetM2MAudioProfileResp Struct Reference

Data Fields

- [BYTE Profile](#)
- [BYTE EarMute](#)
- [BYTE MicMute](#)
- [BYTE Generator](#)
- [BYTE Volume](#)
- [BYTE CwtMute](#)

8.211.1 Detailed Description

This structure contains the SLQSGetM2MAudioProfile response parameters.

Parameters

<i>Profile</i>	<ul style="list-style-type: none">• Audio Profile<ul style="list-style-type: none">– 0-5
<i>EarMute</i>	<ul style="list-style-type: none">• Ear Mute<ul style="list-style-type: none">– 0 - Mute– 1 - Unmute
<i>MicMute</i>	<ul style="list-style-type: none">• MIC Mute<ul style="list-style-type: none">– 0 - Mute– 1 - Unmute
<i>Generator</i>	<ul style="list-style-type: none">• Generator<ul style="list-style-type: none">– 0 - Voice
<i>Volume</i>	<ul style="list-style-type: none">• RX volume level<ul style="list-style-type: none">– 0-5
<i>CwtMute</i>	<ul style="list-style-type: none">• Call waiting tone Mute<ul style="list-style-type: none">– 0 - Mute– 1 - Unmute

8.211.2 Field Documentation

8.211.2.1 **BYTE** GetM2MAudioProfileResp::CwtMute

8.211.2.2 **BYTE** GetM2MAudioProfileResp::EarMute

8.211.2.3 **BYTE** GetM2MAudioProfileResp::Generator

8.211.2.4 **BYTE** GetM2MAudioProfileResp::MicMute

8.211.2.5 **BYTE** GetM2MAudioProfileResp::Profile

8.211.2.6 **BYTE** GetM2MAudioProfileResp::Volume

8.212 GetM2MAudioVolumeReq Struct Reference

Data Fields

- [BYTE Profile](#)
- [BYTE Generator](#)

8.212.1 Detailed Description

This structure contains the SLQSGetM2MAudioVolume request parameters.

Parameters

<i>Profile</i>	<ul style="list-style-type: none">• Audio Profile<ul style="list-style-type: none">– 0-5
<i>Generator</i>	<ul style="list-style-type: none">• Generator<ul style="list-style-type: none">– 0 - Voice

8.212.2 Field Documentation

8.212.2.1 **BYTE** GetM2MAudioVolumeReq::Generator

8.212.2.2 **BYTE** GetM2MAudioVolumeReq::Profile

8.213 GetM2MAudioVolumeResp Struct Reference

Data Fields

- [BYTE Level](#)

8.213.1 Detailed Description

This structure contains the SLQSGetM2MAudioVolume response parameters.

Parameters

<i>Level</i>	<ul style="list-style-type: none">• The RX Volume Level<ul style="list-style-type: none">– 0-5
--------------	--

8.213.2 Field Documentation

8.213.2.1 **BYTE** GetM2MAudioVolumeResp::Level

8.214 GetM2MAVMuteReq Struct Reference

Data Fields

- [BYTE Profile](#)

8.214.1 Detailed Description

This structure contains the SLQSGetM2MAVMute request parameters.

Parameters

<i>Profile</i>	<ul style="list-style-type: none">• Audio Profile Number<ul style="list-style-type: none">– 0-5
----------------	---

8.214.2 Field Documentation

8.214.2.1 **BYTE** GetM2MAVMuteReq::Profile

8.215 GetM2MAVMuteResp Struct Reference

Data Fields

- [BYTE EarMute](#)
- [BYTE MicMute](#)
- [BYTE CwtMute](#)

8.215.1 Detailed Description

This structure contains the SLQSGetM2MAVMute response parameters.

Parameters

<i>pEarMute</i>	<ul style="list-style-type: none"> • Ear Mute <ul style="list-style-type: none"> – 0-Mute – 1-UnMute
<i>pMicMute</i>	<ul style="list-style-type: none"> • Mic Mute <ul style="list-style-type: none"> – 0-Mute – 1-unmute
<i>CwtMute</i>	<ul style="list-style-type: none"> • Waiting tone Mute <ul style="list-style-type: none"> – 0-5

8.215.2 Field Documentation

8.215.2.1 **BYTE** GetM2MAVMuteResp::CwtMute

8.215.2.2 **BYTE** GetM2MAVMuteResp::EarMute

8.215.2.3 **BYTE** GetM2MAVMuteResp::MicMute

8.216 GetM2MSpkrGainReq Struct Reference

Data Fields

- [BYTE Profile](#)

8.216.1 Detailed Description

This structure contains the SLQSGetM2MSpkrGain request parameters.

Parameters

<i>pProfile</i>	<ul style="list-style-type: none"> • Audio Profile Number <ul style="list-style-type: none"> – 0-5
-----------------	---

8.216.2 Field Documentation

8.216.2.1 BYTE GetM2MSpkrGainReq::Profile

8.217 GetM2MSpkrGainResp Struct Reference

Data Fields

- [WORD Value](#)

8.217.1 Detailed Description

This structure contains the SLQSGetM2MSpkrGain response parameters.

Parameters

<i>Value</i>	<ul style="list-style-type: none">• RX speakerphone gain<ul style="list-style-type: none">– 0x0 - 0x7fff
--------------	--

8.217.2 Field Documentation

8.217.2.1 WORD GetM2MSpkrGainResp::Value

8.218 getMsgWaitingInfo Struct Reference

Data Fields

- [BYTE numInstances](#)
- [messageWaitingInfoContent msgWaitInfo](#) [0xFF]

8.218.1 Detailed Description

This structure contains Get Message Waiting Info Response parameters

Parameters

<i>numInstances</i>	<ul style="list-style-type: none">• Number of sets of the elements in structure messageWaitingInfoContent
<i>pMsgWaitInfo</i>	<ul style="list-style-type: none">• Pointer to structure of messageWaitingInfoContent.<ul style="list-style-type: none">– See messageWaitingInfoContent for more information.

8.218.2 Field Documentation

8.218.2.1 `messageWaitingInfoContent getMsgWaitingInfo::msgWaitInfo[0xFF]`

8.218.2.2 `BYTE getMsgWaitingInfo::numInstances`

8.219 GetNetworkTimeResp Struct Reference

Data Fields

- [timeInfo](#) * [p3GPP2TimeInfo](#)
- [timeInfo](#) * [p3GPPTimeInfo](#)

8.219.1 Detailed Description

This structure contains information about the GetNetworkTime response parameters.

Parameters

<i>p3GPP2TimeInfo</i>	[Optional] • See timeInfo for more information
<i>p3GPPTimeInfo</i>	[Optional] • See timeInfo for more information

8.219.2 Field Documentation

8.219.2.1 `timeInfo* GetNetworkTimeResp::p3GPP2TimeInfo`

8.219.2.2 `timeInfo* GetNetworkTimeResp::p3GPPTimeInfo`

8.220 GetRegMgrConfigParams Struct Reference

Data Fields

- `BYTE` * [pSettingResp](#)
- `WORD` * [pPCSCFPort](#)
- `BYTE` * [pPriCSCFPortNameLen](#)
- `BYTE` * [pPriCSCFPortName](#)
- `BYTE` * [pIMSTestMode](#)

8.220.1 Detailed Description

This structure contains the SLQSGetRegMgrConfig response parameters.

Parameters

<i>pSettingResp</i>	<ul style="list-style-type: none"> • Settings Response
<i>pPCSCFPort</i>	<ul style="list-style-type: none"> • Proxy call session control function port
<i>pPriCSCFPortNameLen</i> (IN/OUT)	<ul style="list-style-type: none"> • Size in bytes assigned to the primary CSCF Port name parameter to follow
<i>pPriCSCFPortName</i>	<ul style="list-style-type: none"> • Call Session control port, fully qualified domain name • Length of this string must be specified in pPriCSCFPortNameLen parameter
<i>pIMSTestMode</i>	<ul style="list-style-type: none"> • IMS Test mode Enabled. <ul style="list-style-type: none"> – TRUE - Enabled – FALSE - Disabled

Note

pPriCSCFPortNameLen must be set to a valid value during API call to retrieve pPriCSCFPortName.

8.220.2 Field Documentation

8.220.2.1 **BYTE*** GetRegMgrConfigParams::pIMSTestMode

8.220.2.2 **WORD*** GetRegMgrConfigParams::pPCSCFPort

8.220.2.3 **BYTE*** GetRegMgrConfigParams::pPriCSCFPortName

8.220.2.4 **BYTE*** GetRegMgrConfigParams::pPriCSCFPortNameLen

8.220.2.5 **BYTE*** GetRegMgrConfigParams::pSettingResp

8.221 GetSessionIDResp Struct Reference**Data Fields**

- **ULONG *** pSessionIDv4
- **ULONG *** pSessionIDv6

8.221.1 Field Documentation

8.221.1.1 **ULONG*** *GetSessionIDResp::pSessionIDv4*

8.221.1.2 **ULONG*** *GetSessionIDResp::pSessionIDv6*

8.222 GetSIPConfigResp Struct Reference

Data Fields

- **BYTE *** *pSettingResp*
- **WORD *** *pSIPLocalPort*
- **ULONG *** *pTimerSIPReg*
- **ULONG *** *pSubscribeTimer*
- **ULONG *** *pTimerT1*
- **ULONG *** *pTimerT2*
- **ULONG *** *pTimerTf*
- **BYTE *** *pSigCompEnabled*

8.222.1 Detailed Description

This structure contains the SLQSGetSIPConfig response parameters.

Parameters

<i>pSettingResp</i>	<ul style="list-style-type: none"> • Settings Response
<i>pSIPLocalPort</i>	<ul style="list-style-type: none"> • Primary call session control function SIP port number
<i>pTimerSIPReg</i>	<ul style="list-style-type: none"> • Initial SIP registration duration from the User equipment, in seconds
<i>pSubscribeTimer</i>	<ul style="list-style-type: none"> • Duration of the subscription by the UE for IMS registration notifications, in seconds
<i>pTimerT1</i>	<ul style="list-style-type: none"> • RTT estimate, in milliseconds
<i>pTimerT2</i>	<ul style="list-style-type: none"> • The maximum retransmit interval for non-invite requests and invite responses, in milliseconds
<i>pTimerTf</i>	<ul style="list-style-type: none"> • Non-invite transaction timeout timer, in milliseconds
<i>pSigCompEnabled</i>	<ul style="list-style-type: none"> • Sig Comp Status <ul style="list-style-type: none"> – TRUE - Sig Comp Enabled – FALSE - Sig Comp Disabled

8.222.2 Field Documentation

8.222.2.1 **BYTE*** GetSIPConfigResp::pSettingResp

8.222.2.2 **BYTE*** GetSIPConfigResp::pSigCompEnabled

8.222.2.3 **WORD*** GetSIPConfigResp::pSIPLocalPort

8.222.2.4 **ULONG*** GetSIPConfigResp::pSubscribeTimer

8.222.2.5 **ULONG*** GetSIPConfigResp::pTimerSIPReg

8.222.2.6 **ULONG*** GetSIPConfigResp::pTimerT1

8.222.2.7 **ULONG*** GetSIPConfigResp::pTimerT2

8.222.2.8 **ULONG*** GetSIPConfigResp::pTimerTf

8.223 GnssData Struct Reference

Data Fields

- [ULONGLONG mask](#)

8.223.1 Detailed Description

This structure contains the GNSS data

Parameters

<i>mask</i>	<ul style="list-style-type: none"> • Mask for the GNSS data that is to be deleted • Valid values: <ul style="list-style-type: none"> – QMI_LOC_MASK_DELETE_GPS_SVDIR (0x00000001) - Mask to delete GPS SVDIR – QMI_LOC_MASK_DELETE_GPS_SVSTEER (0x00000002) - Mask to delete GPS SVS↔TEER – QMI_LOC_MASK_DELETE_GPS_TIME (0x00000004) - Mask to delete GPS time – QMI_LOC_MASK_DELETE_GPS_ALM_CORR (0x00000008) - Mask to delete almanac correlation – QMI_LOC_MASK_DELETE_GLO_SVDIR (0x00000010) - Mask to delete GLONASS SV↔DIR – QMI_LOC_MASK_DELETE_GLO_SVSTEER (0x00000020) - Mask to delete GLONASS SVSTEER – QMI_LOC_MASK_DELETE_GLO_TIME (0x00000040) - Mask to delete GLONASS time – QMI_LOC_MASK_DELETE_GLO_ALM_CORR (0x00000080) - Mask to delete GLONASS almanac correlation – QMI_LOC_MASK_DELETE_SBAS_SVDIR (0x00000100) - Mask to delete SBAS SVDIR – QMI_LOC_MASK_DELETE_SBAS_SVSTEER (0x00000200) - Mask to delete SBAS SV↔STEER – QMI_LOC_MASK_DELETE_POSITION (0x00000400) - Mask to delete position estimate – QMI_LOC_MASK_DELETE_TIME (0x00000800) - Mask to delete time estimate – QMI_LOC_MASK_DELETE_IONO (0x00001000) - Mask to delete IONO – QMI_LOC_MASK_DELETE_UTC (0x00002000) - Mask to delete UTC estimate – QMI_LOC_MASK_DELETE_HEALTH (0x00004000) - Mask to delete SV health record – QMI_LOC_MASK_DELETE_SADATA (0x00008000) - Mask to delete SADATA – QMI_LOC_MASK_DELETE_RTI (0x00010000) - Mask to delete RTI – QMI_LOC_MASK_DELETE_SV_NO_EXIST (0x00020000) - Mask to delete SV_NO_EX↔IST – QMI_LOC_MASK_DELETE_FREQ_BIAS_EST (0x00040000) - Mask to delete frequency bias estimate – QMI_LOC_MASK_DELETE_BDS_SVDIR (0x00080000) - Mask to delete BDS SVDIR – QMI_LOC_MASK_DELETE_BDS_SVSTEER (0x00100000) - Mask to delete BDS SVS↔TEER – QMI_LOC_MASK_DELETE_BDS_TIME (0x00200000) - Mask to delete BDS time – QMI_LOC_MASK_DELETE_BDS_ALM_CORR (0x00400000) - Mask to delete BDS almanac correlation – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_GPS (0x00800000) - Mask to delete GNSS SV blacklist GPS – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_GLO (0x01000000) - Mask to delete GNSS SV blacklist GLO – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_BDS (0x02000000) - Mask to delete GNSS SV blacklist BDS
-------------	--

8.223.2 Field Documentation

8.223.2.1 ULONGLONG GnssData::mask

8.224 gnssSvInfoNotification Struct Reference

Data Fields

- [BYTE bAltitudeAssumed](#)
- [satelliteInfo](#) * [pSatelliteInfo](#)

8.224.1 Detailed Description

Contain the parameters passed for SetLocGnssSvInfoCallback by the device.

Parameters

<i>bAltitudeAssumed</i>	<ul style="list-style-type: none"> • Indicates whether altitude is assumed or calculated <ul style="list-style-type: none"> – 0x00 (FALSE) - Valid altitude is calculated – 0x01 (TRUE) - Valid altitude is assumed; there may not be enough satellites to determine precise altitude
<i>pSatelliteInfo</i>	<ul style="list-style-type: none"> • See satelliteInfo for more information.

Note

None

8.224.2 Field Documentation

8.224.2.1 **BYTE gnssSvInfoNotification::bAltitudeAssumed**

8.224.2.2 **satelliteInfo* gnssSvInfoNotification::pSatelliteInfo**

8.225 GPRSQoS Struct Reference

Data Fields

- [ULONG precedenceClass](#)
- [ULONG delayClass](#)
- [ULONG reliabilityClass](#)
- [ULONG peakThroughputClass](#)
- [ULONG meanThroughputClass](#)

8.225.1 Detailed Description

This structure contains the GPRS Quality Of Service Information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>precedence</i> ↔ Class	<ul style="list-style-type: none"> • Precedence class
<i>delayClass</i>	<ul style="list-style-type: none"> • Delay class
<i>reliabilityClass</i>	<ul style="list-style-type: none"> • Reliability class
<i>peak</i> ↔ Throughput↔ Class	<ul style="list-style-type: none"> • Peak throughput class
<i>mean</i> ↔ Throughput↔ Class	<ul style="list-style-type: none"> • Mean throughput class

8.225.2 Field Documentation

8.225.2.1 **ULONG** GPRSQoS::delayClass

8.225.2.2 **ULONG** GPRSQoS::meanThroughputClass

8.225.2.3 **ULONG** GPRSQoS::peakThroughputClass

8.225.2.4 **ULONG** GPRSQoS::precedenceClass

8.225.2.5 **ULONG** GPRSQoS::reliabilityClass

8.226 GPRSRequestedQoS Struct Reference

Data Fields

- [ULONG precedenceClass](#)
- [ULONG delayClass](#)
- [ULONG reliabilityClass](#)
- [ULONG peakThroughputClass](#)
- [ULONG meanThroughputClass](#)

8.226.1 Detailed Description

This structure contains the GPRS Quality Of Service Information

Parameters

<i>precedence</i> ↔ <i>Class</i>	• Precedence class
<i>delayClass</i>	• Delay class
<i>reliabilityClass</i>	• Reliability class
<i>peak</i> ↔ <i>Throughput</i> ↔ <i>Class</i>	• Peak throughput class
<i>mean</i> ↔ <i>Throughput</i> ↔ <i>Class</i>	• Mean throughput class

8.226.2 Field Documentation

8.226.2.1 **ULONG** GPRSRequestedQoS::delayClass8.226.2.2 **ULONG** GPRSRequestedQoS::meanThroughputClass8.226.2.3 **ULONG** GPRSRequestedQoS::peakThroughputClass8.226.2.4 **ULONG** GPRSRequestedQoS::precedenceClass8.226.2.5 **ULONG** GPRSRequestedQoS::reliabilityClass

8.227 GPSSStateInfo Struct Reference

Data Fields

- [BYTE](#) EngineState
- [ULONG](#) ValidMask
- [ULONGLONG](#) Latitude
- [ULONGLONG](#) Longitude
- [ULONG](#) HorizontalUncertainty
- [ULONG](#) Altitude
- [ULONG](#) VerticalUncertainty
- [ULONG](#) TimeStmp_tow_ms
- [WORD](#) TimeStmp_gps_week
- [ULONG](#) Time_uncert_ms
- [BYTE](#) Iono_valid
- [ULONG](#) gps_ephemeris_sv_msk
- [ULONG](#) gps_almanac_sv_msk
- [ULONG](#) gps_health_sv_msk
- [ULONG](#) gps_visible_sv_msk
- [ULONG](#) glo_ephemeris_sv_msk
- [ULONG](#) glo_almanac_sv_msk

- [ULONG glo_health_sv_msk](#)
- [ULONG glo_visible_sv_msk](#)
- [ULONG sbas_ephemeris_sv_msk](#)
- [ULONG sbas_almanac_sv_msk](#)
- [ULONG sbas_health_sv_msk](#)
- [ULONG sbas_visible_sv_msk](#)
- [WORD xtra_start_gps_week](#)
- [WORD xtra_start_gps_minutes](#)
- [WORD xtra_valid_duration_hours](#)

8.227.1 Detailed Description

GPS state Info.

Parameters

<i>EngineState</i>	<ul style="list-style-type: none"> • Values: <ul style="list-style-type: none"> – 0 - OFF – 1 - ON • This field is always valid
<i>ValidMask</i>	<ul style="list-style-type: none"> • Mask of valid state information data. • Values: <ul style="list-style-type: none"> – 0x00000001 - Position(latitude/longitude/horizontal uncertainty) – 0x00000002 - Altitude and vertical uncertainty – 0x00000004 - Time ms – 0x00000008 - Time week number – 0x00000010 - Time uncertainty – 0x00000020 - Iono validity – 0x00000040 - GPS ephemeris – 0x00000080 - GPS almanac – 0x00000100 - GPS health – 0x00000200 - GPS visible SVs – 0x00000400 - GLONASS ephemeris – 0x00000800 - GLONASS almanac – 0x00001000 - GLONASS health – 0x00002000 - GLONASS visible SVs – 0x00004000 - SBAS ephemeris – 0x00008000 - SBAS almanac – 0x00010000 - SBAS health – 0x00020000 - SBAS visible SVs – 0x00040000 - XTRA information
<i>Latitude</i>	<ul style="list-style-type: none"> • Latitude position referenced to the WGS-84 reference ellipsoid, counting positive angles north of the equator and negative angles south of the equator. • Units: Decimal degrees • Range: -90 to +90 degrees. • Value is in double float format (refer to IEEE Std 754-1985)

<i>Longitude</i>	<ul style="list-style-type: none"> Longitude position referenced to the WGS-84 reference ellipsoid, counting positive angles east of the Greenwich Meridian and negative angles west of Greenwich meridian. Units: Decimal degrees Range: -180 to +180 degrees Value is in double float format (refer to IEEE Std 754-1985)
<i>Horizontal↔ Uncertainty</i>	<ul style="list-style-type: none"> Circular horizontal uncertainty (in meters). The uncertainty is provided at 63 percent confidence. Value is in single float format (refer to IEEE Std 754-1985)
<i>Altitude</i>	<ul style="list-style-type: none"> Height above the WGS-84 reference ellipsoid. Value conveys height (in meters) plus 500 m Range -500 to 15883 Value in single float format (refer to IEEE Std 754-1985)
<i>Vertical↔ Uncertainty</i>	<ul style="list-style-type: none"> Vertical uncertainty (in meters). The uncertainty is provided at 68 percent confidence. Value in single float format (refer to IEEE Std 754-1985)
<i>TimeStmp_↔ tow_ms</i>	<ul style="list-style-type: none"> Time stamp in GPS time of week(in milliseconds)
<i>TimeStmp_↔ gps_week</i>	<ul style="list-style-type: none"> GPS week number
<i>Time_uncert_ms</i>	<ul style="list-style-type: none"> Time uncertainty (in milliseconds). The uncertainty is provided at 99 percent confidence.
<i>lono_valid</i>	<ul style="list-style-type: none"> lono validity. Values: <ul style="list-style-type: none"> 0 - Invalid 1 - Valid
<i>gps_↔ ephemeris_sv↔ _msk</i>	<ul style="list-style-type: none"> GPS SV mask for ephemeris; if the bit is set, ephemeris for that SV is available.
<i>gps_almanac_↔ sv_msk</i>	<ul style="list-style-type: none"> GPS SV mask for almanac; if the bit is set, almanac for that SV is available.
<i>gps_health_sv↔ _msk</i>	<ul style="list-style-type: none"> GPS SV mask for health; if the bit is set, health for that SV is available.
<i>gps_visible_sv↔ _msk</i>	<ul style="list-style-type: none"> GPS SV mask for visible Svs; if the bit is set, the SV is available.
<i>glo_ephemeris↔ _sv_msk</i>	<ul style="list-style-type: none"> GLONASS SV mask for ephemeris; if the bit is set, ephemeris for that SV is available.
<i>glo_almanac_↔ sv_msk</i>	<ul style="list-style-type: none"> GLONASS SV mask for almanac; if the bit is set, almanac for that SV is available.

<i>glo_health_sv↔ _msk</i>	<ul style="list-style-type: none"> GLONASS SV mask for health; if the bit is set, health for that SV is available.
<i>glo_visible_sv↔ _msk</i>	<ul style="list-style-type: none"> GLONASS SV mask for visible SVs; if the bit is set, the SV is available.
<i>sbas↔ ephemeris_sv↔ _msk</i>	<ul style="list-style-type: none"> SBAS SV mask for ephemeris; if the bit is set, ephemeris for that SV is available.
<i>sbas_almanac↔ _sv_msk</i>	<ul style="list-style-type: none"> SBAS SV mask for almanac; if the bit is set, almanac for that SV is available.
<i>sbas_health↔ sv_msk</i>	<ul style="list-style-type: none"> SBAS SV mask for health; if the bit is set, health for that SV is available.
<i>sbas_visible↔ sv_msk</i>	<ul style="list-style-type: none"> SBAS SV mask for visible SVs; if the bit is set, the SV is available.
<i>xtra_start_gps↔ _week</i>	<ul style="list-style-type: none"> Current XTRA information is valid starting from this GPS week number
<i>xtra_start_gps↔ _minutes</i>	<ul style="list-style-type: none"> Current XTRA information is valid starting from the GPS minutes with the GPS week
<i>xtra_valid↔ duration_hours</i>	<ul style="list-style-type: none"> XTRA information is valid for this many hours starting from the specified GPS week/minutes

8.227.2 Field Documentation

8.227.2.1 ULONG GPSSStateInfo::Altitude

8.227.2.2 BYTE GPSSStateInfo::EngineState

8.227.2.3 ULONG GPSSStateInfo::glo_almanac_sv_msk

8.227.2.4 ULONG GPSSStateInfo::glo_ephemeris_sv_msk

8.227.2.5 ULONG GPSSStateInfo::glo_health_sv_msk

8.227.2.6 ULONG GPSSStateInfo::glo_visible_sv_msk

8.227.2.7 ULONG GPSSStateInfo::gps_almanac_sv_msk

8.227.2.8 ULONG GPSSStateInfo::gps_ephemeris_sv_msk

8.227.2.9 ULONG GPSSStateInfo::gps_health_sv_msk

- 8.227.2.10 **ULONG** GPSSStateInfo::gps_visible_sv_msk
- 8.227.2.11 **ULONG** GPSSStateInfo::HorizontalUncertainty
- 8.227.2.12 **BYTE** GPSSStateInfo::lono_valid
- 8.227.2.13 **ULONGLONG** GPSSStateInfo::Latitude
- 8.227.2.14 **ULONGLONG** GPSSStateInfo::Longitude
- 8.227.2.15 **ULONG** GPSSStateInfo::sbas_almanac_sv_msk
- 8.227.2.16 **ULONG** GPSSStateInfo::sbas_ephemeris_sv_msk
- 8.227.2.17 **ULONG** GPSSStateInfo::sbas_health_sv_msk
- 8.227.2.18 **ULONG** GPSSStateInfo::sbas_visible_sv_msk
- 8.227.2.19 **ULONG** GPSSStateInfo::Time_uncert_ms
- 8.227.2.20 **WORD** GPSSStateInfo::TimeStmp_gps_week
- 8.227.2.21 **ULONG** GPSSStateInfo::TimeStmp_tow_ms
- 8.227.2.22 **ULONG** GPSSStateInfo::ValidMask
- 8.227.2.23 **ULONG** GPSSStateInfo::VerticalUncertainty
- 8.227.2.24 **WORD** GPSSStateInfo::xtra_start_gps_minutes
- 8.227.2.25 **WORD** GPSSStateInfo::xtra_start_gps_week
- 8.227.2.26 **WORD** GPSSStateInfo::xtra_valid_duration_hours

8.228 gpsTime_s Struct Reference

Data Fields

- [WORD gpsWeek](#)
- [ULONG gpsTimeOfWeekMs](#)

8.228.1 Detailed Description

This structure contains GPS Time info.

Parameters

<i>gpsWeek</i>	<ul style="list-style-type: none"> • Current GPS week as calculated from midnight, Jan. 6, 1980. • Units - Weeks
<i>gpsTimeOf↵ WeekMs</i>	<ul style="list-style-type: none"> • Amount of time into the current GPS week. • Units - Milliseconds

8.228.2 Field Documentation

8.228.2.1 **ULONG** *gpsTime_s::gpsTimeOfWeekMs*

8.228.2.2 **WORD** *gpsTime_s::gpsWeek*

8.229 gsmCellInfo Struct Reference

Data Fields

- [WORD](#) *arfcn*
- [BYTE](#) *band1900*
- [BYTE](#) *cellIdValid*
- [BYTE](#) *bsicId*
- [SHORT](#) *rsi*
- [SHORT](#) *srxlev*

8.229.1 Detailed Description

This structure contains information about the GSM Cell.

Parameters

<i>arfcn</i>	<ul style="list-style-type: none"> • GSM frequency being reported. • Range: 0 to 1023.
<i>band1900</i>	<ul style="list-style-type: none"> • Band indicator for the GSM ARFCN • This field is only valid if <i>arfcn</i> is in the overlapping region. • If TRUE and the cell is in the overlapping region, the ARFCN is on the 1900 band. • If FALSE, it is on the 1800 band.
<i>cellIdValid</i>	<ul style="list-style-type: none"> • Flag indicating whether the base station identity code ID is valid.
<i>bsicId</i>	<ul style="list-style-type: none"> • Base station identity code ID, including base station color code and network color code. • The lower 6 bits can be set to any value.

<i>rssI</i>	<ul style="list-style-type: none">• Measured RSSI value in 1/10 dB.• Range: -200.0 dB to 0
<i>srxlev</i>	<ul style="list-style-type: none">• Cell selection Rx level (Srxlev) value.• Range: -128 to 128.• This field is only valid when ue_in_idle is TRUE.

8.229.2 Field Documentation

8.229.2.1 WORD gsmCellInfo::arfcn

8.229.2.2 BYTE gsmCellInfo::band1900

8.229.2.3 BYTE gsmCellInfo::bsicld

8.229.2.4 BYTE gsmCellInfo::cellIdValid

8.229.2.5 SHORT gsmCellInfo::rssI

8.229.2.6 SHORT gsmCellInfo::srxlev

8.230 GSMRSSIThresh Struct Reference

Data Fields

- [BYTE GSMRSSIThreshListLen](#)
- [WORD * pGSMRSSIThreshList](#)

8.230.1 Detailed Description

This structure contains GSM RSSI threshold related parameters.

Parameters

<i>GSMRSSI</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none">• Length of the GSM RSSI threshold list parameter to follow
<i>pGSMRSSI</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none">• Array of RSSI thresholds (in units of 0.1 dBm)• Maximum of 32 values• Range for RSSI values: -111 to -48 (in dBm)

8.230.2 Field Documentation

8.230.2.1 **BYTE** GSMRSSIthresh::GSMRSSIthreshListLen

8.230.2.2 **WORD*** GSMRSSIthresh::pGSMRSSIthreshList

8.231 GSMSrvStatusInfo Struct Reference

Data Fields

- [BYTE](#) *srvStatus*
- [BYTE](#) *trueSrvStatus*
- [BYTE](#) *isPrefDataPath*

8.231.1 Detailed Description

Structure for storing the service status information for GSM, WCDMA and LTE networks.

Parameters

<i>srvStatus</i>	<ul style="list-style-type: none"> • Service status of the system. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save – 0xFF - Not Available
<i>trueSrvStatus</i>	<ul style="list-style-type: none"> • True service status of the system. • Not applicable to CDMA/HDR. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save – 0xFF - Not Available
<i>isPrefDataPath</i>	<ul style="list-style-type: none"> • Whether the RAT is the preferred data path. <ul style="list-style-type: none"> – 0x00 - Not preferred – 0x01 - Preferred – 0xFF - Not Available

8.231.2 Field Documentation

8.231.2.1 **BYTE** GSMSrvStatusInfo::isPrefDataPath

8.231.2.2 **BYTE** GSMSrvStatusInfo::srvStatus

8.231.2.3 **BYTE** GSMSrvStatusInfo::trueSrvStatus

8.232 GSMSysInfo Struct Reference

Data Fields

- [sysInfoCommon](#) [sysInfoGSM](#)
- **BYTE** lacValid
- **WORD** lac
- **BYTE** cellIdValid
- **ULONG** cellId
- **BYTE** regRejectInfoValid
- **BYTE** rejectSrvDomain
- **BYTE** rejCause
- **BYTE** networkIdValid
- **BYTE** MCC [3]
- **BYTE** MNC [3]
- **BYTE** egprsSuppValid
- **BYTE** egprsSupp
- **BYTE** dtmSuppValid
- **BYTE** dtmSupp

8.232.1 Detailed Description

Structure for storing the GSM System Information.

Parameters

<i>sysInfoGSM</i>	<ul style="list-style-type: none">• See sysInfoCommon for more information.
<i>lacValid</i>	<ul style="list-style-type: none">• Indicates whether the location area code is valid..<ul style="list-style-type: none">– 0x00 - Invalid– 0x01 - Valid– 0xFF - Not Available
<i>lac</i>	<ul style="list-style-type: none">• Location area code.• Only applies to 3GPP.<ul style="list-style-type: none">– 0xFFFF - Not Available
<i>cellIdValid</i>	<ul style="list-style-type: none">• Indicates whether the cell ID is valid..<ul style="list-style-type: none">– 0x00 - Invalid– 0x01 - Valid– 0xFF - Not Available
Generated by Doxygen	

<i>cellId</i>	<ul style="list-style-type: none"> Cell ID. <ul style="list-style-type: none"> 0xFFFFFFFF - Not Available
<i>regRejectInfo</i> <i>Valid</i>	<ul style="list-style-type: none"> Indicates whether the registration reject information is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>rejectSrvDomain</i>	<ul style="list-style-type: none"> Type of service domain in which the registration is rejected. <ul style="list-style-type: none"> 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service 0x01 - Circuit-switched only 0x02 - Packet-switched only 0x03 - Circuit-switched and packet-switched 0x04 - Camped 0xFF - Not Available
<i>rejCause</i>	<ul style="list-style-type: none"> Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6]. <ul style="list-style-type: none"> 0xFF - Not Available
<i>networkIdValid</i>	<ul style="list-style-type: none"> Indicates whether the network ID is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>MCC</i> [<i>PLMN_L</i> <i>ENGTH</i>]	<ul style="list-style-type: none"> Mobile Country Code. MCC digits in ASCII characters
<i>MNC</i> [<i>PLMN_L</i> <i>ENGTH</i>]	<ul style="list-style-type: none"> Mobile Network Code. MNC digits in ASCII characters An unused byte is set to 0xFF. In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.
<i>egprsSuppValid</i>	<ul style="list-style-type: none"> Indicates whether the EGPRS support is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available

<i>egprsSupp</i>	<ul style="list-style-type: none"> • EGPRS support indication. • Only applicable for GSM. <ul style="list-style-type: none"> – 0x00 - Not available – 0x01 - Available – 0xFF - Not Available
<i>dtmSuppValid</i>	<ul style="list-style-type: none"> • Indicates whether Dual Transfer mode support is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>dtmSupp</i>	<ul style="list-style-type: none"> • Dual Transfer mode support indication. • Only applicable for GSM. <ul style="list-style-type: none"> – 0x00 - Not available – 0x01 - Available – 0xFF - Not Available

8.232.2 Field Documentation

8.232.2.1 **ULONG** GSMSysInfo::cellId

8.232.2.2 **BYTE** GSMSysInfo::cellIdValid

8.232.2.3 **BYTE** GSMSysInfo::dtmSupp

8.232.2.4 **BYTE** GSMSysInfo::dtmSuppValid

8.232.2.5 **BYTE** GSMSysInfo::egprsSupp

8.232.2.6 **BYTE** GSMSysInfo::egprsSuppValid

8.232.2.7 **WORD** GSMSysInfo::lac

8.232.2.8 **BYTE** GSMSysInfo::lacValid

8.232.2.9 **BYTE** GSMSysInfo::MCC[3]

8.232.2.10 **BYTE** GSMSysInfo::MNC[3]

8.232.2.11 **BYTE** GSMSysInfo::networkIdValid

8.232.2.12 **BYTE** GSMSysInfo::regRejectInfoValid

8.232.2.13 **BYTE** GSMSysInfo::rejCause

8.232.2.14 **BYTE** GSMSysInfo::rejectSrvDomain

8.232.2.15 **sysInfoCommon** GSMSysInfo::sysInfoGSM

8.233 gyroAcceptReady_s Struct Reference

Data Fields

- [BYTE injectEnable](#)
- [WORD samplesPerBatch](#)
- [WORD batchPerSec](#)

8.233.1 Detailed Description

This structure contains Gyroscope Accept Ready Info

Parameters

<i>injectEnable</i>	<ul style="list-style-type: none"> • GNSS location engine is ready to accept data from sensor. • Values <ul style="list-style-type: none"> • 0x01 - Ready to accept sensor data • 0x00 - Not ready to accept sensor data
<i>samplesPerBatch</i>	<ul style="list-style-type: none"> • number of samples per batch the GNSS location engine is to receive. • $\text{samplingFrequency} = \text{samplesPerBatch} * \text{batchesPerSecond}$ • samplesPerBatch must be a nonzero positive value.
<i>batchPerSec</i>	<ul style="list-style-type: none"> • LTE NAS version minor • Number of sensor-data batches the GNSS location engine is to receive per second. • BatchesPerSecond must be a nonzero positive value.

8.233.2 Field Documentation

8.233.2.1 **WORD** gyroAcceptReady_s::batchPerSec

8.233.2.2 **BYTE** gyroAcceptReady_s::injectEnable

8.233.2.3 **WORD** gyroAcceptReady_s::samplesPerBatch

8.234 gyroTempAcceptReady_s Struct Reference

Data Fields

- [BYTE injectEnable](#)
- [WORD samplesPerBatch](#)
- [WORD batchPerSec](#)

8.234.1 Detailed Description

This structure contains Gyroscope Temperature Accept Ready Info

Parameters

<i>injectEnable</i>	<ul style="list-style-type: none"> • GNSS location engine is ready to accept data from sensor. • Values <ul style="list-style-type: none"> • 0x01 - Ready to accept sensor data • 0x00 - Not ready to accept sensor data
<i>samplesPerBatch</i>	<ul style="list-style-type: none"> • number of samples per batch the GNSS location engine is to receive. • $\text{samplingFrequency} = \text{samplesPerBatch} * \text{batchesPerSecond}$ • samplesPerBatch must be a nonzero positive value.
<i>batchPerSec</i>	<ul style="list-style-type: none"> • LTE NAS version minor • Number of sensor-data batches the GNSS location engine is to receive per second. • BatchesPerSecond must be a nonzero positive value.

8.234.2 Field Documentation

8.234.2.1 **WORD** gyroTempAcceptReady_s::batchPerSec

8.234.2.2 **BYTE** gyroTempAcceptReady_s::injectEnable

8.234.2.3 **WORD** gyroTempAcceptReady_s::samplesPerBatch

8.235 HDRECIOTresh Struct Reference

Data Fields

- [BYTE HDRECIOTreshListLen](#)
- [WORD * pHDR](#)ECIOTreshList

8.235.1 Detailed Description

This structure contains HDR ECIO threshold related parameters.

Parameters

<i>HDRECIO</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> Length of the HDR ECIO threshold list parameter to follow
<i>pHDRECIO</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> Array of ECIO thresholds (in units of 0.1 dB) Maximum of 32 values Range for ECIO values: -31.5 to 0 (in dB).

8.235.2 Field Documentation

8.235.2.1 **BYTE** HDRECIOThresh::HDRECIOThreshListLen

8.235.2.2 **WORD*** HDRECIOThresh::pHDRECIOThreshList

8.236 HDRIOThresh Struct Reference

Data Fields

- [BYTE HDRIOThreshListLen](#)
- [WORD * pHDRIOThreshList](#)

8.236.1 Detailed Description

This structure contains HDR IO threshold related parameters.

Parameters

<i>HDRIOThresh</i> ↔ <i>ListLen</i>	<ul style="list-style-type: none"> Length of the HDR IO threshold list parameter to follow
<i>pHDRIO</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> Array of IO thresholds (in units of 0.1 dBm) Maximum of 32 values Range for IO values: -128 to -13 (in dBm).

8.236.2 Field Documentation

8.236.2.1 **BYTE** HDRIOThresh::HDRIOThreshListLen

8.236.2.2 **WORD*** HDRIOThresh::pHDRIOThreshList

8.237 HDRPersonalityInd Struct Reference

Data Fields

- [WORD](#) * [pCurrentPersonality](#)
- [BYTE](#) * [pPersonalityListLength](#)
- [protocolSubtypeElement](#) * [pProtocolSubtypeElement](#)

8.237.1 Field Documentation

8.237.1.1 [WORD](#)* HDRPersonalityInd::pCurrentPersonality

8.237.1.2 [BYTE](#)* HDRPersonalityInd::pPersonalityListLength

8.237.1.3 [protocolSubtypeElement](#)* HDRPersonalityInd::pProtocolSubtypeElement

8.238 HDRPersonalityResp Struct Reference

Data Fields

- [WORD](#) * [pCurrentPersonality](#)
- [BYTE](#) * [pPersonalityListLength](#)
- [protocolSubtypeElement](#) * [pProtocolSubtypeElement](#)

8.238.1 Detailed Description

This structure contains information about the SLQSSwiGetHDRPersonality response parameters.

Parameters

<i>pCurrentPersonality</i> [Out]	<ul style="list-style-type: none">• Current active personality index.
<i>pPersonalityListLength</i> [In/Out]	<ul style="list-style-type: none">• Number of Personality Protocol Subtype contains in this response.• maximum input value is 3
<i>pProtocolSubtypeElement</i> [Out]	<ul style="list-style-type: none">• See protocolSubtypeElement for more information.

8.238.2 Field Documentation

8.238.2.1 [WORD](#)* HDRPersonalityResp::pCurrentPersonality

8.238.2.2 [BYTE](#)* HDRPersonalityResp::pPersonalityListLength

8.238.2.3 [protocolSubtypeElement](#)* HDRPersonalityResp::pProtocolSubtypeElement

8.239 HDRProtSubtypResp Struct Reference

Data Fields

- [WORD](#) * [pCurrentPrsnlty](#)
- [BYTE](#) * [pPersonalityListLength](#)
- [protocolSubtypeElement](#) * [pProtoSubTypElmnt](#)
- [ULONGLONG](#) * [pAppSubType](#)

8.239.1 Detailed Description

This structure contains information about the SLQSSwiGetHDRProtSubtype response parameters.

Parameters

<i>pCurrent↔ Personality[Out]</i>	<ul style="list-style-type: none"> • Current active personality index.
<i>pPersonality↔ ListLength[In/↔ Out]</i>	<ul style="list-style-type: none"> • Number of Personality Protocol Subtype contains in this response. • maximum input value is 4
<i>pProtocol↔ Subtype↔ Element[Out]</i>	<ul style="list-style-type: none"> • See protocolSubtypeElement for more information.
<i>pAppSubType[↔ Out]</i>	<ul style="list-style-type: none"> • Stream application subtype • Application subtype for each stream,

8.239.2 Field Documentation

8.239.2.1 [ULONGLONG](#)* HDRProtSubtypResp::pAppSubType

8.239.2.2 [WORD](#)* HDRProtSubtypResp::pCurrentPrsnlty

8.239.2.3 [BYTE](#)* HDRProtSubtypResp::pPersonalityListLength

8.239.2.4 [protocolSubtypeElement](#)* HDRProtSubtypResp::pProtoSubTypElmnt

8.240 HDRRSSIThresh Struct Reference

Data Fields

- [BYTE](#) HDRRSSIThreshListLen
- [WORD](#) * [pHDRRSSIThreshList](#)

8.240.1 Detailed Description

This structure contains HDR RSSI threshold related parameters.

Parameters

<i>HDRRSSI</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> Length of the HDR RSSI threshold list parameter to follow
<i>pHDRRSSI</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> Array of RSSI thresholds (in units of 0.1 dBm) Maximum of 32 values. Range for RSSI values: -118 to -13 (in dBm).

8.240.2 Field Documentation

8.240.2.1 **BYTE** HDRRSSIThresh::HDRRSSIThreshListLen

8.240.2.2 **WORD*** HDRRSSIThresh::pHDRRSSIThreshList

8.241 HDRSINRThresh Struct Reference

Data Fields

- [BYTE HDRSINRThresListLen](#)
- [BYTE * pHDRSINRThresList](#)

8.241.1 Detailed Description

This structure contains HDR SINR threshold related parameters.

Parameters

<i>HDRSINR</i> ↔ <i>ThresListLen</i>	<ul style="list-style-type: none"> Length of the HDR SINR threshold list parameter to follow
<i>pHDRSINR</i> ↔ <i>ThresList</i>	<ul style="list-style-type: none"> Sequence of thresholds delimiting SINR event reporting bands Every time a new SINR value crosses a threshold value, an event report indication message with the new SINR value is sent to the requesting control point. For this field <ul style="list-style-type: none"> SINR is reported only for HDR Each SINR threshold value is an unsigned 1 byte value Maximum number of threshold values is 16 At least one value must be specified

8.241.2 Field Documentation

8.241.2.1 **BYTE** HDRSINRThresh::HDRSINRThresListLen

8.241.2.2 **BYTE*** HDRSINRThresh::pHDRSINRThresList

8.242 HDRSINRThreshold Struct Reference

Data Fields

- [BYTE HDRSINRThreshListLen](#)
- [WORD * pHDRSINRThreshList](#)

8.242.1 Detailed Description

This structure contains HDR SINR threshold related parameters.

Parameters

<i>HDRSINR</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> • Length of the HDR ECIO threshold list parameter to follow
<i>pHDRSINR</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> • Array of SINR level thresholds (in units of 1) • maximum of 32 values. • Valid levels are 0 to 8 <ul style="list-style-type: none"> – 0x00 - SINR_LEVEL_0 is -9 dB – 0x01 - SINR_LEVEL_1 is -6 dB – 0x02 - SINR_LEVEL_2 is -4.5 dB – 0x03 - SINR_LEVEL_3 is -3 dB – 0x04 - SINR_LEVEL_4 is -2 dB – 0x05 - SINR_LEVEL_5 is +1 dB – 0x06 - SINR_LEVEL_6 is +3 dB – 0x07 - SINR_LEVEL_7 is +6 dB – 0x08 - SINR_LEVEL_8 is +9 dB

8.242.2 Field Documentation

8.242.2.1 **BYTE** HDRSINRThreshold::HDRSINRThreshListLen

8.242.2.2 **WORD*** HDRSINRThreshold::pHDRSINRThreshList

8.243 HDRSSInfo Struct Reference

Data Fields

- [INT8 rssi](#)

- [SHORT ecio](#)
- [BYTE sinr](#)
- [INT32 io](#)

8.243.1 Detailed Description

This structure contains the parameters for HDR Signal Strength Information

Parameters

<i>rssI</i>	<ul style="list-style-type: none"> • RSSI in dBm (signed value). • A value of -125 dBm or lower is used to indicate No Signal.
<i>ecio</i>	<ul style="list-style-type: none"> • ECIO value representing negative 0.5 dBm increments, i.e., 2 means -1 dBm (14 means -7 dBm, 63 means -31.5 dBm).
<i>sinr</i>	<ul style="list-style-type: none"> • SINR level. • SINR is only applicable for 1xEV-DO. • Valid levels are 0 to 8, where the maximum value for: <ul style="list-style-type: none"> – 0 - SINR_LEVEL_0 is -9 dB – 1 - SINR_LEVEL_1 is -6 dB – 2 - SINR_LEVEL_2 is -4.5 dB – 3 - SINR_LEVEL_3 is -3 dB – 4 - SINR_LEVEL_4 is -2 dB – 5 - SINR_LEVEL_5 is +1 dB – 6 - SINR_LEVEL_6 is +3 dB – 7 - SINR_LEVEL_7 is +6 dB – 8 - SINR_LEVEL_8 is +9 dB – 0xFF - Not Available
<i>io</i>	<ul style="list-style-type: none"> • Received IO in dBm. • IO is only applicable for 1xEV-DO.

8.243.2 Field Documentation

8.243.2.1 SHORT HDRSSInfo::ecio

8.243.2.2 INT32 HDRSSInfo::io

8.243.2.3 INT8 HDRSSInfo::rssI

8.243.2.4 BYTE HDRSSInfo::sinr

8.244 hdrSSInfo Struct Reference

Data Fields

- [int8_t rssi](#)
- [int16_t ecio](#)
- [uint8_t sinr](#)
- [int32_t io](#)

8.244.1 Detailed Description

Parameters

<i>rssi</i>	RSSI in dBm.
<i>ecio</i>	ECIO value representing negative 0.5 dBm increment
<i>sinr</i>	SINR level.
<i>io</i>	Received IO in dBm.

8.244.2 Field Documentation

8.244.2.1 [int16_t hdrSSInfo::ecio](#)

8.244.2.2 [int32_t hdrSSInfo::io](#)

8.244.2.3 [int8_t hdrSSInfo::rssi](#)

8.244.2.4 [uint8_t hdrSSInfo::sinr](#)

8.245 HDRSysInfo Struct Reference

Data Fields

- [sysInfoCommon sysInfoHDR](#)
- [BYTE isSysPrIMatchValid](#)
- [BYTE isSysPrIMatch](#)
- [BYTE hdrPersonalityValid](#)
- [BYTE hdrPersonality](#)
- [BYTE hdrActiveProtValid](#)
- [BYTE hdrActiveProt](#)
- [BYTE is856SysIdValid](#)
- [BYTE is856SysId \[16\]](#)

8.245.1 Detailed Description

Structure for storing the HDR System Information.

Parameters

<i>sysInfoHDR</i>	<ul style="list-style-type: none"> • See sysInfoCommon for more information.
<i>isSysPrIMatch</i> ↔ <i>Valid</i>	<ul style="list-style-type: none"> • Indicates whether the system PRL match is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>isSysPrIMatch</i>	<ul style="list-style-type: none"> • Indicates whether the system is in a PRL. • Only applies to CDMA/HDR. <ul style="list-style-type: none"> – 0x00 - System is not in a PRL – 0x01 - System is in a PRL – 0xFF - Not Available • If the system is not in a PRL, roam_status carries the value from the default roaming indicator in the PRL. • If the system is in a PRL, roam_status is set to the value based on the standard specification.
<i>hdrPersonality</i> ↔ <i>Valid</i>	<ul style="list-style-type: none"> • Indicates whether the HDR personality is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>hdrPersonality</i>	<ul style="list-style-type: none"> • HDR personality information. • Only applicable for HDR. <ul style="list-style-type: none"> – 0x00 - None – 0x02 - HRPD – 0x03 - eHRPD – 0xFF - Not Available
<i>hdrActiveProt</i> ↔ <i>Valid</i>	<ul style="list-style-type: none"> • Indicates whether the HDR active protocol revision information is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>hdrActiveProt</i>	<ul style="list-style-type: none"> • HDR active protocol revision information . • Only applicable for HDR. <ul style="list-style-type: none"> – 0x00 - None – 0x02 - HDR Rel 0 – 0x03 - HDR Rel A – 0x04 - HDR Rel B – 0xFF - Not Available

<i>is856SysIdValid</i>	<ul style="list-style-type: none"> Indicates whether the IS-856 system ID is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>is856SysId[SL↔ QS_SYSTEM↔ _ID_SIZE]</i>	<ul style="list-style-type: none"> IS-856 system ID. Only applicable for HDR.

8.245.2 Field Documentation

8.245.2.1 **BYTE** HDRSysInfo::hdrActiveProt

8.245.2.2 **BYTE** HDRSysInfo::hdrActiveProtValid

8.245.2.3 **BYTE** HDRSysInfo::hdrPersonality

8.245.2.4 **BYTE** HDRSysInfo::hdrPersonalityValid

8.245.2.5 **BYTE** HDRSysInfo::is856SysId[16]

8.245.2.6 **BYTE** HDRSysInfo::is856SysIdValid

8.245.2.7 **BYTE** HDRSysInfo::isSysPrIMatch

8.245.2.8 **BYTE** HDRSysInfo::isSysPrIMatchValid

8.245.2.9 **sysInfoCommon** HDRSysInfo::sysInfoHDR

8.246 homeSIDNID Struct Reference

Data Fields

- [BYTE](#) numInstances
- [sidNid](#) SidNid [255]

8.246.1 Detailed Description

This structure contains the parameters for Home SID/NID Information

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> Number of sets of the following elements: <ul style="list-style-type: none"> sid nid If zero(0), then no information follows.
---------------------	---

8.246.2 Field Documentation

8.246.2.1 `BYTE homeSIDNID::numInstances`

8.246.2.2 `sidNid homeSIDNID::SidNid[255]`

8.247 hotSwapStatus Struct Reference

Data Fields

- `BYTE hotSwapLength`
- `BYTE hotSwap [255]`

8.247.1 Detailed Description

This structure contains Hot Swap Status Information.

Parameters

<i>hotSwapLength</i>	<ul style="list-style-type: none"> • Number of sets of the following elements. i.e. hot_swap
<i>hotSwap[$\text{MAX}_{\leftrightarrow}$ _DESCRIPTI\leftrightarrow ON_LENGTH]</i>	<ul style="list-style-type: none"> • Indicates the status of the hot-swap switch. <ul style="list-style-type: none"> – 0 - Hot-swap is not supported – 1 - Hot-swap is supported, but the status of the switch is not supported – 2 - Switch indicates that the card is present – 3 - Switch indicates that the card is not present

8.247.2 Field Documentation

8.247.2.1 `BYTE hotSwapStatus::hotSwap[255]`

8.247.2.2 `BYTE hotSwapStatus::hotSwapLength`

8.248 image_info_t Struct Reference

Data Fields

- `uint8_t imageType`
- `uint8_t uniqueID [16]`
- `uint8_t buildIDLen`
- `uint8_t buildID [255]`

8.248.1 Field Documentation

8.248.1.1 `uint8_t image_info_t::buildID[255]`

8.248.1.2 `uint8_t image_info_t::buildIDLen`

8.248.1.3 `uint8_t image_info_t::imageType`

8.248.1.4 `uint8_t image_info_t::uniqueID[16]`

8.249 ImageElement Struct Reference

Data Fields

- [BYTE imageType](#)
- [BYTE imageId \[16\]](#)
- [BYTE buildIdLength](#)
- [CHAR buildId \[100\]](#)

8.249.1 Detailed Description

push current alignment to stack set alignment to 1 byte boundary This structure contains the Image Element information

Parameters

<i>imageType</i>	<ul style="list-style-type: none"> • Type of image 0 - Modem 1 - PRI
<i>imageId</i>	<ul style="list-style-type: none"> • Unique image identifier
<i>buildIdLength</i>	<ul style="list-style-type: none"> • Length of the build ID string (may be zero)
<i>buildId</i>	<ul style="list-style-type: none"> • Build ID ANSI string(Max 100 characters)

8.249.2 Field Documentation

8.249.2.1 `CHAR ImageElement::buildId[100]`

8.249.2.2 `BYTE ImageElement::buildIdLength`

8.249.2.3 `BYTE ImageElement::imageId[16]`

8.249.2.4 BYTE ImageElement::imageType

8.250 ImageIdElement Struct Reference

Data Fields

- [BYTE storageIndex](#)
- [BYTE failureCount](#)
- [BYTE imageID \[16\]](#)
- [BYTE buildIDLength](#)
- [CHAR buildID \[100\]](#)

8.250.1 Detailed Description

push current alignment to stack set alignment to 1 byte boundary This structure contains the Image ID list element Information

Parameters

<i>storageIndex</i>	<ul style="list-style-type: none"> • Index in storage where the image is located(a value of 0xFF indicates that the storage for this type of image is not relevant)
<i>failureCount</i>	<ul style="list-style-type: none"> • Number of consecutive write attempts to this storage index that have failed(a value of 0xFF indicates unspecified)
<i>imageID</i>	<ul style="list-style-type: none"> • Image unique identifier(max 16 chars.)
<i>buildIDLength</i>	<ul style="list-style-type: none"> • Length of the build ID string. If there is no build ID, this field will be 0 and no data will follow.
<i>buildID</i>	<ul style="list-style-type: none"> • String containing image build information(Max 100 characters)

8.250.2 Field Documentation

8.250.2.1 CHAR ImageIdElement::buildID[100]

8.250.2.2 BYTE ImageIdElement::buildIDLength

8.250.2.3 BYTE ImageIdElement::failureCount

8.250.2.4 BYTE ImageIdElement::imageID[16]

8.250.2.5 BYTE ImageIdElement::storageIndex

8.251 ImageIDEntries Struct Reference

Data Fields

- [BYTE imageType](#)
- [BYTE maxImages](#)
- [BYTE executingImage](#)
- [BYTE imageIDSize](#)
- struct [ImageIDElement imageIDElement](#) [50]

8.251.1 Detailed Description

This structure contains the list entry Information

Parameters

<i>imageType</i>	<ul style="list-style-type: none"> • Type of image <ul style="list-style-type: none"> – 0 - Modem – 1 - PRI
<i>maxImages</i>	<ul style="list-style-type: none"> • Maximum number of images of this type that may be stored concurrently on the device
<i>executingImage</i>	<ul style="list-style-type: none"> • Index (into the next array) of image that is currently executing
<i>imageIDSize</i>	<ul style="list-style-type: none"> • The number of elements in the image ID list
<i>imageIDElement</i>	<ul style="list-style-type: none"> • Array of ImageIDElement Structure (Max 50 elements)

8.251.2 Field Documentation

8.251.2.1 **BYTE** ImageIDEntries::executingImage

8.251.2.2 **struct ImageIDElement** ImageIDEntries::imageIDElement[50]

8.251.2.3 **BYTE** ImageIDEntries::imageIDSize

8.251.2.4 **BYTE** ImageIDEntries::imageType

8.251.2.5 **BYTE** ImageIDEntries::maxImages

8.252 ImageList Struct Reference

Data Fields

- [BYTE listSize](#)
- struct [ImageIDEntries imageIDEntries](#) [2]

8.252.1 Detailed Description

This structure contains the Get Stored Images List

Parameters

<i>listSize</i>	<ul style="list-style-type: none"> The number of elements in the image list
<i>imageIDEntries</i>	<ul style="list-style-type: none"> Array of ImageIDEntries Structure (Max 2 entries)

8.252.2 Field Documentation

8.252.2.1 struct ImageIDEntries ImageList::imageIDEntries[2]

8.252.2.2 BYTE ImageList::listSize

8.253 IMSAIndRegisterInfo Struct Reference

Data Fields

- [BYTE](#) * [pRegStatusConfig](#)
- [BYTE](#) * [pServiceStatusConfig](#)
- [BYTE](#) * [pRatHandoverStatusConfig](#)
- [BYTE](#) * [pPdpStatusConfig](#)

8.253.1 Detailed Description

This structure contains parameters of IMSA Config Indication Register

Parameters

<i>pRegStatus↔ Config(optional)</i>	<ul style="list-style-type: none"> Register Indication For Registration status. When this registration is enabled, the device learns of Registration status via the QMI_IMSA↔_REGISTRATION_STATUS_IND indication. <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pService↔ Status↔ Config(optional)</i>	<ul style="list-style-type: none"> Register Indication For Service status Events. When this registration is enabled, the device learns of Service status via the QMI_IMSA_SE↔RVICE_STATUS_IND indication. <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable

<i>pRatHandover↔ Status↔ Config(optional)</i>	<ul style="list-style-type: none"> Registration Indication For RAT handover status. When this registration is enabled, the device learns of RAT handover status via the QMI_IMS↔ A_RAT_HANOVER_STATUS_IND indication. <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pPdpStatus↔ Config(optional)</i>	<ul style="list-style-type: none"> PDP Status Configuration. <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable

Note

One of the optional parameter is mandatory to be present in the request.

8.253.2 Field Documentation

8.253.2.1 **BYTE*** IMSAIndRegisterInfo::pPdpStatusConfig

8.253.2.2 **BYTE*** IMSAIndRegisterInfo::pRatHandoverStatusConfig

8.253.2.3 **BYTE*** IMSAIndRegisterInfo::pRegStatusConfig

8.253.2.4 **BYTE*** IMSAIndRegisterInfo::pServiceStatusConfig

8.254 imsaPdpStatusInfo Struct Reference**Data Fields**

- [BYTE](#) connetionState
- [ULONG *](#) pFailErrorCode

8.254.1 Detailed Description

Contains the parameters passed for SLQSSetIMSAPdpStatusCallback by the device.

Parameters

<i>connetionState</i>	<ul style="list-style-type: none"> IMS PDP connection state information. Values <ul style="list-style-type: none"> TRUE – IMS PDP is connected FALSE – IMS PDP is not connected
<i>pFailErrorCode</i>	<ul style="list-style-type: none"> IMS PDP connection failure error reason code when the IMS PDP Connection State TLV is FALSE. Values <ul style="list-style-type: none"> 0 - Generic failure reason for other than specified

8.254.2 Field Documentation

8.254.2.1 **BYTE** `imsaPdpStatusInfo::connetionState`

8.254.2.2 **ULONG*** `imsaPdpStatusInfo::pFailErrorCode`

8.255 imsaRatStatusInfo Struct Reference

Data Fields

- **ULONG *** `pRATStatus`
- **ULONG *** `pSrcRAT`
- **ULONG *** `pTgtRAT`
- **BYTE *** `pErrorCodeStr`

8.255.1 Detailed Description

Contains the parameters passed for `SLQSSetIMSAStatusCallback` by the device.

Parameters

<i>pRATStatus</i>	<ul style="list-style-type: none">• RAT handover Status
<i>pSrcRAT</i>	<ul style="list-style-type: none">• Source RAT
<i>pTgtRAT</i>	<ul style="list-style-type: none">• Target RAT
<i>pErrorCodeStr</i>	<ul style="list-style-type: none">• Error Code String

8.255.2 Field Documentation

8.255.2.1 **BYTE*** `imsaRatStatusInfo::pErrorCodeStr`

8.255.2.2 **ULONG*** `imsaRatStatusInfo::pRATStatus`

8.255.2.3 **ULONG*** `imsaRatStatusInfo::pSrcRAT`

8.255.2.4 **ULONG*** `imsaRatStatusInfo::pTgtRAT`

8.256 IMSARegistrationStatus Struct Reference

Data Fields

- **BYTE *** `plmsRegStatus`

- [WORD](#) * [plmsRegErrCode](#)
- [ULONG](#) * [pNewImsRegStatus](#)

8.256.1 Detailed Description

This structure contains response parameters of registration status.

Parameters

<i>plmsRegStatus</i>	<ul style="list-style-type: none"> • IMS Registration Status (Deprecated). • Values <ul style="list-style-type: none"> – TRUE - UE is registered on the IMS network – FALSE - UE is not registered on the IMS network
<i>plmsRegErr↔ Code</i>	<ul style="list-style-type: none"> • IMS Registration Error Code. • An error code is returned when the IMS registration status is IMSA_STATUS_NOT_REGIST↔ERED. -Values <ul style="list-style-type: none"> – 3xx – Redirection responses – 4xx – Client failure responses – 5xx – Server failure responses – 6xx – Global failure responses
<i>pNewImsReg↔ Status</i>	<ul style="list-style-type: none"> • New IMS Registration Status • Values <ul style="list-style-type: none"> – 0 - Not registered for IMS – 1 - Registering for IMS – 2 - Registered for IMS

8.256.2 Field Documentation

8.256.2.1 **WORD*** IMSARegistrationStatus::plmsRegErrCode

8.256.2.2 **BYTE*** IMSARegistrationStatus::plmsRegStatus

8.256.2.3 **ULONG*** IMSARegistrationStatus::pNewImsRegStatus

8.257 imsaRegStatusInfo Struct Reference

Data Fields

- [BYTE](#) * [pbIMSRegistered](#)
- [WORD](#) * [pRegStatusErrorCode](#)
- [ULONG](#) * [plmsRegStatus](#)

8.257.1 Detailed Description

Contains the parameters passed for SLQSSetIMSARegStatusCallback by the device.

Parameters

<i>pbIMSRegistered</i>	<ul style="list-style-type: none"> • TRUE/FALSE
<i>pRegStatusErrorCode</i>	<ul style="list-style-type: none"> • if IMSA_STATUS_NOT_REGISTERED. Values: 3xx – Redirection responses 4xx – Client failure responses 5xx – Server failure responses 6xx – Global failure responses
<i>plmsRegStatus</i>	IMS registration status. Values: IMSA_STATUS_NOT_REGISTERED - 0 IMSA_STATUS_REGISTERING - 1 IMSA_STATUS_REGISTERED - 2

8.257.2 Field Documentation

8.257.2.1 **BYTE*** `imsaRegStatusInfo::pbIMSRegistered`

8.257.2.2 **ULONG*** `imsaRegStatusInfo::plmsRegStatus`

8.257.2.3 **WORD*** `imsaRegStatusInfo::pRegStatusErrorCode`

8.258 IMSAServiceStatus Struct Reference

Data Fields

- [ULONG *](#) `pSmsServiceStatus`
- [ULONG *](#) `pVoipServiceStatus`
- [ULONG *](#) `pVtServiceStatus`
- [ULONG *](#) `pSmsServiceRat`
- [ULONG *](#) `pVoipServiceRat`
- [ULONG *](#) `pVtServiceRat`
- [ULONG *](#) `pUtServiceStatus`
- [ULONG *](#) `pUtServiceRat`
- [ULONG *](#) `pVsServiceStatus`
- [ULONG *](#) `pVsServiceRat`

8.258.1 Detailed Description

This structure contains response parameters of service status for various IMS services.

Parameters

<i>pSmsService</i> ↔ <i>Status</i>	<ul style="list-style-type: none"> • SMS Service Status. • Values <ul style="list-style-type: none"> – 0 - IMS SMS service is not available – 1 - IMS SMS is in limited service – 2 - IMS SMS is in full service
<i>pVoipService</i> ↔ <i>Status</i>	<ul style="list-style-type: none"> • VoIP Service Status. -Values <ul style="list-style-type: none"> – 0 - IMS VoIP service is not available – 2 - IMS VoIP is in full service
<i>pVtService</i> ↔ <i>Status</i>	<ul style="list-style-type: none"> • VT Service Status • Values <ul style="list-style-type: none"> – 0 - IMS VT service is not available – 2 - IMS VT is in full service
<i>pSmsServiceRat</i>	<ul style="list-style-type: none"> • SMS service RAT • Values <ul style="list-style-type: none"> – 0 - IMS service is registered on WLAN – 1 - IMS service is registered on WWAN – 2 - IMS service is registered on interworking WLAN
<i>pVoipServiceRat</i>	<ul style="list-style-type: none"> • VoIP service RAT. • Values <ul style="list-style-type: none"> – 0 - IMS service is registered on WLAN – 1 - IMS service is registered on WWAN – 2 - IMS service is registered on interworking WLAN
<i>pVtServiceRat</i>	<ul style="list-style-type: none"> • VT service RAT. • Values <ul style="list-style-type: none"> – 0 - IMS service is registered on WLAN – 1 - IMS service is registered on WWAN – 2 - IMS service is registered on interworking WLAN
<i>pUtService</i> ↔ <i>Status</i>	<ul style="list-style-type: none"> • UT service Status. • Values <ul style="list-style-type: none"> – 0 - IMS UT service is not available – 2 - IMS UT is in full service
<i>pUtServiceRat</i>	<ul style="list-style-type: none"> • UT service RAT. • Values <ul style="list-style-type: none"> – 0 - IMS service is registered on WLAN – 1 - IMS service is registered on WWAN – 2 - IMS service is registered on interworking WLAN

<i>pVsServiceStatus</i>	<ul style="list-style-type: none"> VS service Status. Values <ul style="list-style-type: none"> 0 - IMS UT service is not available 2 - IMS UT is in full service
<i>pVsServiceRat</i>	<ul style="list-style-type: none"> VS service RAT. Values <ul style="list-style-type: none"> 0 - IMS service is registered on WLAN 1 - IMS service is registered on WWAN 2 - IMS service is registered on interworking WLAN

8.258.2 Field Documentation

8.258.2.1 **ULONG*** IMSAServiceStatus::pSmsServiceRat

8.258.2.2 **ULONG*** IMSAServiceStatus::pSmsServiceStatus

8.258.2.3 **ULONG*** IMSAServiceStatus::pUtServiceRat

8.258.2.4 **ULONG*** IMSAServiceStatus::pUtServiceStatus

8.258.2.5 **ULONG*** IMSAServiceStatus::pVoipServiceRat

8.258.2.6 **ULONG*** IMSAServiceStatus::pVoipServiceStatus

8.258.2.7 **ULONG*** IMSAServiceStatus::pVsServiceRat

8.258.2.8 **ULONG*** IMSAServiceStatus::pVsServiceStatus

8.258.2.9 **ULONG*** IMSAServiceStatus::pVtServiceRat

8.258.2.10 **ULONG*** IMSAServiceStatus::pVtServiceStatus

8.259 IMSASupportedFieldsResp Struct Reference

Data Fields

- struct [ReqFieldsList](#) * pReqFieldsList
- struct [RespFieldsList](#) * pRespFieldsList
- struct [IndFieldsList](#) * pIndFieldsList

8.259.1 Detailed Description

This structure contains response of supported fields by the currently running software.

Parameters

<i>pReqFieldsList</i>	<ul style="list-style-type: none"> List of Supported Request Fields. See ReqFieldsList for more information
<i>pRespFieldsList</i>	<ul style="list-style-type: none"> List of Supported Request Fields. See RespFieldsList for more information
<i>pIndFieldsList</i>	<ul style="list-style-type: none"> List of Supported Request Fields. See IndFieldsList for more information

8.259.2 Field Documentation

8.259.2.1 struct **IndFieldsList*** IMSASupportedFieldsResp::pIndFieldsList

8.259.2.2 struct **ReqFieldsList*** IMSASupportedFieldsResp::pReqFieldsList

8.259.2.3 struct **RespFieldsList*** IMSASupportedFieldsResp::pRespFieldsList

8.260 IMSASupportedMsgInfo Struct Reference

Data Fields

- struct [SupportedMsgList](#) * pSupportedMsgList

8.260.1 Detailed Description

This structure contains Queries the set of messages implemented by the currently running software.

Parameters

<i>pSupported↔ MsgList</i>	<ul style="list-style-type: none"> List of Supported Messages. See SupportedMsgList for more information
--------------------------------	--

8.260.2 Field Documentation

8.260.2.1 struct **SupportedMsgList*** IMSASupportedMsgInfo::pSupportedMsgList

8.261 imsaSvcStatusInfo Struct Reference

Data Fields

- [ULONG](#) * [pSMSSvcStatus](#)
- [ULONG](#) * [pVOIPSvcStatus](#)
- [ULONG](#) * [pVTSvcStatus](#)
- [ULONG](#) * [pSMSSvcRAT](#)
- [ULONG](#) * [pVOIPSvcRAT](#)
- [ULONG](#) * [pVTSvcRAT](#)
- [ULONG](#) * [pUTSvcStatus](#)
- [ULONG](#) * [pUTSvcRAT](#)

8.261.1 Detailed Description

Contains the parameters passed for SLQSSetIMSASvcStatusCallback by the device.

Parameters

<i>pSMSSvcStatus</i>	IMS SMS NOT AVAILABLE - 0 IMS SMS LIMITED SERVICE - 1 IMS SMS FULL SERVICE - 2
<i>pVOIPSvcStatus</i>	IMS VOIP NOT AVAILABLE - 0 IMS SMS FULL SERVICE -2
<i>pVTSvcStatus</i>	IMS VT NOT AVAILABLE - 0 IMS VT FULL SERVICE - 2
<i>pSMSSvcRAT</i>	IMS service is registered on WLAN - 0 IMS service is registered on WWAN - 1 IMS service is registered on interworking WLAN -2
<i>pVOIPSvcRAT</i>	IMS service is registered on WLAN - 0 IMS service is registered on WWAN - 1 IMS service is registered on interworking WLAN -2 *
<i>pVTSvcRAT</i>	IMS service is registered on WLAN - 0 IMS service is registered on WWAN - 1 IMS service is registered on interworking WLAN -2
<i>pUTSvcStatus</i>	IMS VOIP NOT AVAILABLE - 0 IMS SMS FULL SERVICE -2
<i>pUTSvcRAT</i>	IMS service is registered on WLAN - 0 IMS service is registered on WWAN - 1 IMS service is registered on interworking WLAN -2

8.261.2 Field Documentation

8.261.2.1 **ULONG*** [imsaSvcStatusInfo::pSMSSvcRAT](#)

8.261.2.2 **ULONG*** [imsaSvcStatusInfo::pSMSSvcStatus](#)

8.261.2.3 **ULONG*** [imsaSvcStatusInfo::pUTSvcRAT](#)

8.261.2.4 **ULONG*** [imsaSvcStatusInfo::pUTSvcStatus](#)

8.261.2.5 **ULONG*** [imsaSvcStatusInfo::pVOIPSvcRAT](#)

8.261.2.6 **ULONG*** [imsaSvcStatusInfo::pVOIPSvcStatus](#)

8.261.2.7 **ULONG*** [imsaSvcStatusInfo::pVTSvcRAT](#)

8.261.2.8 **ULONG*** [imsaSvcStatusInfo::pVTSvcStatus](#)

8.262 imsCfgIndRegisterInfo Struct Reference

Data Fields

- [BYTE](#) * [pSIPConfigEvents](#)

- BYTE * pRegMgrConfigEvents
- BYTE * pSMSConfigEvents
- BYTE * pUserConfigEvents
- BYTE * pVoIPConfigEvents

8.262.1 Detailed Description

This structure contains parameters of IMS Config Indication Register

Parameters

<i>pSIPConfigEvents(optional)</i>	<ul style="list-style-type: none"> • Registration Indication For SIP Configuration Events. • When this registration is enabled, the device learns of SIP config events via the QMI_IMS_SIP_CONFIG_IND indication. <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pRegMgrConfigEvents(optional)</i>	<ul style="list-style-type: none"> • Registration Indication For Registration Manager Configuration Events. • When this registration is enabled, the device learns of Reg Mgr config events via the QMI_IMS_REG_MGR_CONFIG_IND indication. <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pSMSConfigEvents(optional)</i>	<ul style="list-style-type: none"> • Registration Indication For SMS Configuration Events. • When this registration is enabled, the device learns of SMS config events via the QMI_IMS_SMS_CONFIG_IND indication. <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pUserConfigEvents(optional)</i>	<ul style="list-style-type: none"> • Registration Indication For User Configuration Events. • When this registration is enabled, the device learns of user config events via the QMI_IMS_USER_CONFIG_IND indication. <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pVoIPConfigEvents(optional)</i>	<ul style="list-style-type: none"> • Registration Indication For VoIP Configuration Events. • When this registration is enabled, the device learns of VOIP config events via the QMI_IMS_VOIP_CONFIG_IND indication. <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable

Note

One of the optional parameter is mandatory to be present in the request.

8.262.2 Field Documentation

8.262.2.1 **BYTE*** imsCfgIndRegisterInfo::pRegMgrConfigEvents

8.262.2.2 **BYTE*** imsCfgIndRegisterInfo::pSIPConfigEvents

8.262.2.3 **BYTE*** imsCfgIndRegisterInfo::pSMSConfigEvents

8.262.2.4 **BYTE*** imsCfgIndRegisterInfo::pUserConfigEvents

8.262.2.5 **BYTE*** imsCfgIndRegisterInfo::pVoIPConfigEvents

8.263 imsRegMgrConfigInfo Struct Reference

Data Fields

- **WORD*** pPriCSCFPort
- **BYTE*** pCSCFPortName
- **BYTE*** pIMSTestMode

8.263.1 Detailed Description

Contains the parameters passed for SLQSSetRegMgrConfigCallback by the device.

Parameters

<i>pPriCSCFPort</i>	<ul style="list-style-type: none"> • Primary call session control function port
<i>pCSCFPortName</i>	<ul style="list-style-type: none"> • Call Session control port, fully qualified domain name • Length of this string can be of maximum 255 bytes
<i>pIMSTestMode</i>	<ul style="list-style-type: none"> • IMS Test mode Enabled. <ul style="list-style-type: none"> – TRUE - Enable, no IMS registration – FALSE - Disable, IMS registration is initiated

8.263.2 Field Documentation

8.263.2.1 **BYTE*** imsRegMgrConfigInfo::pCSCFPortName

8.263.2.2 **BYTE*** imsRegMgrConfigInfo::pIMSTestMode

8.263.2.3 **WORD*** imsRegMgrConfigInfo::pPriCSCFPort

8.264 imsSIPConfigInfo Struct Reference

Data Fields

- WORD * pSIPLocalPort
- ULONG * pTimerSIPReg
- ULONG * pSubscribeTimer
- ULONG * pTimerT1
- ULONG * pTimerT2
- ULONG * pTimerTf
- BYTE * pSigCompEnabled

8.264.1 Detailed Description

Contains the parameters passed for SLQSSetSIPConfigCallback by the device.

Parameters

<i>pSIPLocalPort</i>	<ul style="list-style-type: none"> Primary call session control function SIP port number
<i>pTimerSIPReg</i>	<ul style="list-style-type: none"> Initial SIP registration duration from the User equipment, in seconds
<i>pSubscribeTimer</i>	<ul style="list-style-type: none"> Duration of the subscription by the UE for IMS registration notifications, in seconds
<i>pTimerT1</i>	<ul style="list-style-type: none"> RTT estimate, in milliseconds
<i>pTimerT2</i>	<ul style="list-style-type: none"> The maximum retransmit interval for non-invite requests and invite responses, in milliseconds
<i>pTimerTf</i>	<ul style="list-style-type: none"> Non-invite transaction timeout timer, in milliseconds
<i>pSigCompEnabled</i>	<ul style="list-style-type: none"> Sig Comp Status <ul style="list-style-type: none"> TRUE - Enable FALSE - Disable

Note

None

8.264.2 Field Documentation

8.264.2.1 BYTE* imsSIPConfigInfo::pSigCompEnabled

8.264.2.2 **WORD*** `imsSIPConfigInfo::pSIPLocalPort`

8.264.2.3 **ULONG*** `imsSIPConfigInfo::pSubscribeTimer`

8.264.2.4 **ULONG*** `imsSIPConfigInfo::pTimerSIPReg`

8.264.2.5 **ULONG*** `imsSIPConfigInfo::pTimerT1`

8.264.2.6 **ULONG*** `imsSIPConfigInfo::pTimerT2`

8.264.2.7 **ULONG*** `imsSIPConfigInfo::pTimerTf`

8.265 imsSMSConfigInfo Struct Reference

Data Fields

- **BYTE *** `pSMSFormat`
- **BYTE *** `pSMSOverIPNwInd`
- **BYTE *** `pPhoneCtxtURI`

8.265.1 Detailed Description

Contains the parameters passed for `SLQSSetIMSSMSConfigCallback` by the device.

Parameters

<i>pSMSFormat</i>	<ul style="list-style-type: none"> • SMS format <ul style="list-style-type: none"> – 0 - 3GPP – 1 - 3GPP2
<i>pSMSOverIP↔ NwInd</i>	<ul style="list-style-type: none"> • SMS over IP Network Indication Flag <ul style="list-style-type: none"> – TRUE - Mobile-Originated(MO) SMS turned on – FALSE - MO SMS turned off
<i>pPhoneCtxtURI</i>	<ul style="list-style-type: none"> • Phone context universal resource identifier • Length of this string can be of maximum 255 bytes

8.265.2 Field Documentation

8.265.2.1 **BYTE*** `imsSMSConfigInfo::pPhoneCtxtURI`

8.265.2.2 **BYTE*** `imsSMSConfigInfo::pSMSFormat`

8.265.2.3 **BYTE*** `imsSMSConfigInfo::pSMSOverIPNwnd`

8.266 `imsUserConfigInfo` Struct Reference

Data Fields

- **BYTE *** `pIMSDomain`

8.266.1 Detailed Description

Contains the parameters passed for `SLQSSetIMSUserConfigCallback` by the device.

Parameters

<i>pIMSDomain</i>	<ul style="list-style-type: none">• IMS domain name• Length of this string can be of maximum 255 bytes
-------------------	---

8.266.2 Field Documentation

8.266.2.1 **BYTE*** `imsUserConfigInfo::pIMSDomain`

8.267 `imsVoIPConfigInfo` Struct Reference

Data Fields

- **WORD *** `pSessionExpiryTimer`
- **WORD *** `pMinSessionExpiryTimer`
- **BYTE *** `pAmrWbEnable`
- **BYTE *** `pScrAmrEnable`
- **BYTE *** `pScrAmrWbEnable`
- **BYTE *** `pAmrMode`
- **WORD *** `pAmrWBMode`
- **BYTE *** `pAmrOctetAligned`
- **BYTE *** `pAmrWBOctetAligned`
- **WORD *** `pRingingTimer`
- **WORD *** `pRingBackTimer`
- **WORD *** `pRTPRTCPInactTimer`

8.267.1 Detailed Description

Contains the parameters passed for `SLQSSetIMSVoIPConfigCallback` by the device.

Parameters

<i>pSessionExpiryTimer</i>	<ul style="list-style-type: none"> • Session duration, in seconds
<i>pMinSessionExpiryTimer</i>	<ul style="list-style-type: none"> • Minimum allowed value for session expiry timer, in seconds
<i>pAmrWbEnable</i>	<ul style="list-style-type: none"> • Flag to enable/disable Adaptive Multirate Codec(AMR) WideBand(WB) audio • Values: <ul style="list-style-type: none"> – True - Enabled – False - Disabled
<i>pScrAmrEnable</i>	<ul style="list-style-type: none"> • Flag to enable/disable Source Control Rate(SCR) for AMR NarrowBand (NB) • Values: <ul style="list-style-type: none"> – True - Enabled – False - Disabled
<i>pScrAmrWbEnable</i>	<ul style="list-style-type: none"> • Flag to enable/disable SCR for AMR WB Audio • Values: <ul style="list-style-type: none"> – True - Enabled – False - Disabled
<i>pAmrMode</i>	<ul style="list-style-type: none"> • BitMask for AMR NB modes allowed • Values: <ul style="list-style-type: none"> – 0x1 - 4.75 kbps – 0x2 - 5.15 kbps – 0x4 - 5.9 kbps – 0x8 - 6.17 kbps – 0x10 - 7.4 kbps – 0x20 - 7.95 kbps – 0x40 - 10.2 kbps – 0x80 - 12.2 kbps
<i>pAmrWBMode</i>	<ul style="list-style-type: none"> • BitMask for AMR WB modes allowed • Values: <ul style="list-style-type: none"> – 0x1 - 6.60 kbps – 0x2 - 8.85 kbps – 0x4 - 12.65 kbps – 0x8 - 14.25 kbps – 0x10 - 15.85 kbps – 0x20 - 18.25 kbps – 0x40 - 19.85 kbps – 0x80 - 23.05 kbps – 0x100 - 23.85 kbps

<i>pAmrOctet</i> ↔ <i>Aligned</i>	<ul style="list-style-type: none"> • Flag to indicate if the octet is aligned for AMR NB Audio • Values: <ul style="list-style-type: none"> – True - Aligned – False - Not aligned, Bandwidth Efficient mode
<i>pAmrWBOctet</i> ↔ <i>Aligned</i>	<ul style="list-style-type: none"> • Flag to indicate if the octet is aligned for AMR WB Audio • Values: <ul style="list-style-type: none"> – True - Aligned – False - Not aligned, Bandwidth Efficient mode
<i>pRingingTimer</i>	<ul style="list-style-type: none"> • Duration of ringing timer, in seconds. The ringing timer starts on the ringing event. If the call is not answered within the duration of this timer, the call is disconnected.
<i>pRingBackTimer</i>	<ul style="list-style-type: none"> • Duration of ringback timer, in seconds. The ringback timer starts on the ringback event. If the call is not answered within the duration of this timer, the call is disconnected.
<i>pRTPRTCP</i> ↔ <i>InactTimer</i>	<ul style="list-style-type: none"> • Duration of RTP/RTCP inactivity timer, in seconds. If no RTP/RTCP packet is received prior to the expiry of this timer, the call is disconnected.

8.267.2 Field Documentation

8.267.2.1 **BYTE*** `imsVoIPConfigInfo::pAmrMode`

8.267.2.2 **BYTE*** `imsVoIPConfigInfo::pAmrOctetAligned`

8.267.2.3 **BYTE*** `imsVoIPConfigInfo::pAmrWbEnable`

8.267.2.4 **WORD*** `imsVoIPConfigInfo::pAmrWBMode`

8.267.2.5 **BYTE*** `imsVoIPConfigInfo::pAmrWBOctetAligned`

8.267.2.6 **WORD*** `imsVoIPConfigInfo::pMinSessionExpiryTimer`

8.267.2.7 **WORD*** `imsVoIPConfigInfo::pRingBackTimer`

8.267.2.8 **WORD*** `imsVoIPConfigInfo::pRingingTimer`

8.267.2.9 **WORD*** `imsVoIPConfigInfo::pRTPRTCPInactTimer`

8.267.2.10 **BYTE*** `imsVoIPConfigInfo::pScrAmrEnable`

8.267.2.11 **BYTE*** `imsVoIPConfigInfo::pScrAmrWbEnable`

8.267.2.12 **WORD*** `imsVoIPConfigInfo::pSessionExpiryTimer`

8.268 IndFieldsList Struct Reference

Data Fields

- [BYTE indicationFieldsLen](#)
- [BYTE indicationFields](#) [256]

8.268.1 Detailed Description

This structure contains the Supported Indication Fields List Information

Parameters

<i>indication↔ FieldsLen</i>	<ul style="list-style-type: none">• Number of sets of the indication fields.
<i>indicationFields</i>	<ul style="list-style-type: none">• Describes which optional field IDs are supported in QMI indication.• Format is same as request field.

8.268.2 Field Documentation

8.268.2.1 **BYTE** `IndFieldsList::indicationFields`[256]

8.268.2.2 **BYTE** `IndFieldsList::indicationFieldsLen`

8.269 infoInterFreq Struct Reference

Data Fields

- [WORD earfcn](#)
- [BYTE threshXLow](#)
- [BYTE threshXHigh](#)
- [BYTE cell_resel_priority](#)
- [BYTE cells_len](#)
- [cellParams cellInterFreqParams](#) [255]

8.269.1 Detailed Description

This structure contains information about the inter-frequency.

Parameters

<i>earfcn</i>	<ul style="list-style-type: none"> E-UTRA absolute radio frequency channel number of the serving cell. Range: 0 to 65535.
<i>threshXLow</i>	<ul style="list-style-type: none"> Cell Srxlev low threshold. Range: 0 to 31. When the serving cell does not exceed thresh_serving_low, the value of an evaluated cell must be smaller than this value to be considered for re-selection.
<i>threshXHigh</i>	<ul style="list-style-type: none"> Cell Srxlev high threshold. Range: 0 to 31. When the serving cell exceeds thresh_serving_low, the value of an evaluated cell must be greater than this value to be considered for re-selection.
<i>cell_resel_↔ priority</i>	<ul style="list-style-type: none"> Cell re-selection priority Range: 0 to 7. This field is only valid when ue_in_idle is TRUE.
<i>cells_len</i>	<ul style="list-style-type: none"> Provides the number of set of cell params.
<i>cellInterFreq↔ Params[MAX↔ DESCRIPTIO↔ N_LENGTH]</i>	<ul style="list-style-type: none"> See cellParams for more information.

8.269.2 Field Documentation

8.269.2.1 **BYTE** infoInterFreq::cell_resel_priority

8.269.2.2 **cellParams** infoInterFreq::cellInterFreqParams[255]

8.269.2.3 **BYTE** infoInterFreq::cells_len

8.269.2.4 **WORD** infoInterFreq::earfcn

8.269.2.5 **BYTE** infoInterFreq::threshXHigh

8.269.2.6 **BYTE** infoInterFreq::threshXLow

8.270 IOTresh Struct Reference

Data Fields

- [BYTE](#) IOTreshListLen
- [INT32](#) * pIOTreshList

8.270.1 Detailed Description

This structure contains IO threshold related parameters.

Parameters

<i>IOThresListLen</i>	<ul style="list-style-type: none"> Length of the LTE SNR threshold list parameter to follow
<i>plIOThresList</i>	<ul style="list-style-type: none"> Sequence of thresholds delimiting IO event reporting bands Every time a new IO value crosses a threshold value, an event report indication message with the new IO value is sent to the requesting control point. For this field <ul style="list-style-type: none"> IO is applicable only for HDR Each IO threshold value is a signed 4 byte value Maximum number of threshold values is 16 At least one value must be specified

8.270.2 Field Documentation

8.270.2.1 BYTE IOThresh::IOThresListLen

8.270.2.2 INT32* IOThresh::plIOThresList

8.271 IPv4Addr Struct Reference

Data Fields

- [ULONG addr](#)
- [ULONG subnetMask](#)

8.271.1 Detailed Description

This structure contains the IPv4 filter address

Parameters

<i>addr</i>	IPv4 address
<i>subnetMask</i>	A packet matches if: <ul style="list-style-type: none"> (addr and subnetMask) == (IP pkt addr & subnetMask) Callers to set up a filter with a range of source addresses, if needed; subnet mask of all 1s (255.255.255.255) specifies a single address value

8.271.2 Field Documentation

8.271.2.1 **ULONG** IPv4Addr::addr

8.271.2.2 **ULONG** IPv4Addr::subnetMask

8.272 IPv6Addr Struct Reference

Data Fields

- [BYTE](#) addr [16]
- [BYTE](#) prefixLen

8.272.1 Detailed Description

This structure contains the IPv6 filter address

Parameters

<i>addr</i>	IPv6 address (in network byte order); this is a 16-byte byte array (in Big-endian format)
<i>prefixLen</i>	IPv6 filter prefix length; can take a value between 0 and 128 Note: A packet matches if the IPv6 source address bytes until the prefix lengths are equal. Therefore prefix length can be used to set a filter with a range of source addresses. A prefix length of 128 specifies a single address value.

8.272.2 Field Documentation

8.272.2.1 **BYTE** IPv6Addr::addr[16]

8.272.2.2 **BYTE** IPv6Addr::prefixLen

8.273 IPV6AddressInfo Struct Reference

Data Fields

- [BYTE](#) IPV6PrefixLen
- [USHORT](#) IPAddressV6 [8]

8.273.1 Detailed Description

This structure contains the IPV6 Address Information

Parameters

<i>IPV6PrefixLen</i>	<ul style="list-style-type: none"> Length of the received IPv6 address in no. of bits; can take value between 0 and 128 <ul style="list-style-type: none"> 0xFF - Not Available
<i>IPAddressV6</i>	<ul style="list-style-type: none"> IPv6 address(in network byte order); This is an 8-element array of 16 bit numbers, each of which is in big endian format.

8.273.2 Field Documentation

8.273.2.1 USHORT IPV6AddressInfo::IPAddressV6[8]

8.273.2.2 BYTE IPV6AddressInfo::IPV6PrefixLen

8.274 ipv6AddressInfo Struct Reference

Data Fields

- uint8_t [IPV6PrefixLen](#)
- uint16_t [IPAddressV6](#) [8]

8.274.1 Detailed Description

Parameters

<i>IPV6PrefixLen</i>	Length of the received IPv6 address
<i>IPAddressV6</i>	IPv6 address(in network byte order)

8.274.2 Field Documentation

8.274.2.1 uint16_t ipv6AddressInfo::IPAddressV6[8]

8.274.2.2 uint8_t ipv6AddressInfo::IPV6PrefixLen

8.275 IPV6GWAddressInfo Struct Reference

Data Fields

- [BYTE](#) [gwV6PrefixLen](#)
- [USHORT](#) [gwAddressV6](#) [8]

8.275.1 Detailed Description

This structure contains the IPV6 Gateway Address Information

Parameters

<i>gwV6PrefixLen</i>	<ul style="list-style-type: none">• Length of the received IPv6 Gateway address in no. of bits; can take value between 0 and 128
<i>IPAddressV6</i>	<ul style="list-style-type: none">• IPv6 Gateway address(in network byte order); This is an 8-element array of 16 bit numbers, each of which is in big endian format.

8.275.2 Field Documentation

8.275.2.1 USHORT IPV6GWAddressInfo::gwAddressV6[8]

8.275.2.2 BYTE IPV6GWAddressInfo::gwV6PrefixLen

8.276 IPv6TrafCls Struct Reference

Data Fields

- [BYTE val](#)
- [BYTE mask](#)

8.276.1 Detailed Description

This structure contains the IPv6 filter traffic class

Parameters

<i>val</i>	The traffic class value
<i>mask</i>	<p>The packet matches the traffic class filter if: (IPv6_filter_traffic_class_val and IPv6_filter_traffic_class_mask) == (Traffic class value in the IP packet & IPv6_filter_traffic_class_mask) Example:</p> <ul style="list-style-type: none"> • IPv6_filter_tc_val = 00101000 • IPv6_filter_tc_mask = 11111100 Filter will compare only the first 6 bits in IPv6_filter_traffic_class with the first 6 bits in the traffic class field of the IP packet; first 6 bits in the traffic class field of the IP packet must be 001010 to match filter; last 2 bits can be anything, since they are ignored by filtering

8.276.2 Field Documentation

8.276.2.1 BYTE IPv6TrafCls::mask

8.276.2.2 BYTE IPv6TrafCls::val

8.277 LibPackGPRSRequestedQoS Struct Reference

Data Fields

- uint32_t [precedenceClass](#)
- uint32_t [delayClass](#)
- uint32_t [reliabilityClass](#)
- uint32_t [peakThroughputClass](#)
- uint32_t [meanThroughputClass](#)

8.277.1 Detailed Description

This structure contains the GPRS Quality Of Service Information

Parameters

<i>precedence</i> ↔ Class	<ul style="list-style-type: none"> • Precedence class
<i>delayClass</i>	<ul style="list-style-type: none"> • Delay class
<i>reliabilityClass</i>	<ul style="list-style-type: none"> • Reliability class
<i>peak</i> ↔ Throughput↔ Class	<ul style="list-style-type: none"> • Peak throughput class
<i>mean</i> ↔ Throughput↔ Class	<ul style="list-style-type: none"> • Mean throughput class

8.277.2 Field Documentation

8.277.2.1 `uint32_t LibPackGPRSRequestedQoS::delayClass`

8.277.2.2 `uint32_t LibPackGPRSRequestedQoS::meanThroughputClass`

8.277.2.3 `uint32_t LibPackGPRSRequestedQoS::peakThroughputClass`

8.277.2.4 `uint32_t LibPackGPRSRequestedQoS::precedenceClass`

8.277.2.5 `uint32_t LibPackGPRSRequestedQoS::reliabilityClass`

8.278 LibpackProfile3GPP Struct Reference

Data Fields

- `uint8_t * pProfileName`
- `uint16_t * pProfileNameSize`
- `uint8_t * pDPType`
- `uint8_t * pPdpHdrCompType`
- `uint8_t * pPdpDataCompType`
- `uint8_t * pAPNName`
- `uint16_t * pAPNNameSize`
- `uint32_t * pPriDNSIPv4AddPref`
- `uint32_t * pSecDNSIPv4AddPref`
- `LibPackUMTSQoS * pUMTSReqQoS`
- `LibPackUMTSQoS * pUMTSMinQoS`
- `LibPackGPRSRequestedQoS * pGPRSRequestedQoS`

- [LibPackGPRSRequestedQoS](#) * [pGPRSMinimumQoS](#)
- [uint8_t](#) * [pUsername](#)
- [uint16_t](#) * [pUsernameSize](#)
- [uint8_t](#) * [pPassword](#)
- [uint16_t](#) * [pPasswordSize](#)
- [uint8_t](#) * [pAuthenticationPref](#)
- [uint32_t](#) * [pIPv4AddrPref](#)
- [uint8_t](#) * [pPscfAddrUsingPCO](#)
- [uint8_t](#) * [pPdpAccessConFlag](#)
- [uint8_t](#) * [pPscfAddrUsingDhcp](#)
- [uint8_t](#) * [pImCnFlag](#)
- [LibPackTFTIDParams](#) * [pTFTID1Params](#)
- [LibPackTFTIDParams](#) * [pTFTID2Params](#)
- [uint8_t](#) * [pPdpContext](#)
- [uint8_t](#) * [pSecondaryFlag](#)
- [uint8_t](#) * [pPrimaryID](#)
- [uint16_t](#) * [pIPv6AddPref](#)
- [LibPackUMTSReqQoSsigInd](#) * [pUMTSReqQoSsigInd](#)
- [LibPackUMTSReqQoSsigInd](#) * [pUMTSMinQoSsigInd](#)
- [uint16_t](#) * [pPriDNSIPv6addpref](#)
- [uint16_t](#) * [pSecDNSIPv6addpref](#)
- [uint8_t](#) * [pAddrAllocPref](#)
- [LibPackQoSClassID](#) * [pQoSClassID](#)
- [uint8_t](#) * [pAPNDisabledFlag](#)
- [uint32_t](#) * [pPDNInactivTimeout](#)
- [uint8_t](#) * [pAPNClass](#)

8.278.1 Detailed Description

Parameters

<i>extended</i>	error
<i>profile</i>	<p>info This structure contains Input parameters of SLQSCreateProfile and SLQSModifyProfile and output parameters of SLQSGetProfileSettings</p> <ul style="list-style-type: none"> Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.
<i>pProfileName</i>	<ul style="list-style-type: none"> One or more uint8_ts describing the profile
<i>pProfileNameSize;</i>	<ul style="list-style-type: none"> This parameter is an input parameter and should be initialised to the size of pProfileName field. Size of this parameter is 2 uint8_ts.
<i>pPDPTYPE</i>	<ul style="list-style-type: none"> Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile <ul style="list-style-type: none"> 0x00 - PDP-IP (IPv4) 0x01 - PDP-PPP 0x02 - PDP-IPV6 0x03 - PDP-IPV4V6

<i>pPdpHdr</i> ↔ <i>CompType</i>	<ul style="list-style-type: none"> • PDP header compression type <ul style="list-style-type: none"> – 0 - PDP header compression is OFF – 1 - Manufacturer preferred compression – 2 - PDP header compression based on RFC 1144 – 3 - PDP header compression based on RFC 25074 PDP header compression based on RFC 3095
<i>pPdpData</i> ↔ <i>CompType</i>	<ul style="list-style-type: none"> • PDP data compression type <ul style="list-style-type: none"> – 0 - PDP data compression is OFF – 1 - Manufacturer preferred compression – 2 - V.42BIS data compression – 3 - V.44 data compression
<i>pAPNName</i>	<ul style="list-style-type: none"> • Access point name
<i>pAPNnameSize;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pAPNName field. Size of this parameter is 2 uint8_ts.
<i>pPriDNSIPv4</i> ↔ <i>AddPref</i>	<ul style="list-style-type: none"> • Primary DNS IPv4 Address Preference
<i>pSecDNSIPv4</i> ↔ <i>AddPref</i>	<ul style="list-style-type: none"> • Secondary DNS IPv4 Address Preference
<i>pUMTSReqQoS</i>	<ul style="list-style-type: none"> • UMTS Requested QoS
<i>pUMTSMinQoS</i>	<ul style="list-style-type: none"> • UMTS Minimum QoS
<i>pGPRS</i> ↔ <i>RequestedQoS</i>	<ul style="list-style-type: none"> • GPRS Minimum QoS
<i>pUsername</i>	<ul style="list-style-type: none"> • User name
<i>pUsernameSize;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pUsername field. Size of this parameter is 2 uint8_ts.
<i>pPassword</i>	<ul style="list-style-type: none"> • Password
<i>pPasswordSize;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pPassword field. Size of this parameter is 2 uint8_ts.

<i>pAuthPref</i>	<ul style="list-style-type: none"> • Authentication Preference <ul style="list-style-type: none"> – Bit map that indicates the authentication algorithm preference <ul style="list-style-type: none"> * Bit 0 - PAP preference <ul style="list-style-type: none"> · 0 - PAP is never performed · 1 - PAP may be performed * Bit 1 - CHAP preference <ul style="list-style-type: none"> · 0 - CHAP is never performed · 1 - CHAP may be performed * If more than one bit is set, then the device decides which authentication procedure is performed while setting up the data session. For example, the device may have a policy to select the most secure authentication mechanism.
<i>pIPv4AddrPref</i>	<ul style="list-style-type: none"> • IPv4 Address Preference
<i>pPcscfAddrUsingPCO</i>	<ul style="list-style-type: none"> • P-CSCF Address using PCO Flag <ul style="list-style-type: none"> – 1 - (TRUE) implies request PCSCF address using PCO – 0 - (FALSE) implies do not request By default, this value is 0
<i>pPdpAccessConFlag</i>	<ul style="list-style-type: none"> • PDP access control flag <ul style="list-style-type: none"> – 0 - PDP access control none – 1 - PDP access control reject – 2 - PDP access control permission
<i>pPcscfAddrUsingDhcp</i>	<ul style="list-style-type: none"> • P-CSCF address using DHCP <ul style="list-style-type: none"> – 1 - (TRUE) implies Request PCSCF address using DHCP – 0 - (FALSE) implies do not request By default, value is 0
<i>pImCnFlag</i>	<ul style="list-style-type: none"> • IM CN flag <ul style="list-style-type: none"> – 1 - (TRUE) implies request IM CN flag for this profile – 0 - (FALSE) implies do not request IM CN flag for this profile
<i>pTFTID1Params</i>	<ul style="list-style-type: none"> • Traffic Flow Template
<i>pTFTID2Params</i>	<ul style="list-style-type: none"> • Traffic Flow Template
<i>pPdpContext</i>	<ul style="list-style-type: none"> • PDP context number
<i>pSecondaryFlag</i>	<ul style="list-style-type: none"> • PDP context secondary flag <ul style="list-style-type: none"> – 1 - (TRUE) implies this is secondary profile – 0 - (FALSE) implies this is not secondary profile

<i>pPrimaryID</i>	<ul style="list-style-type: none"> PDP context primary ID function SLQSGetProfileSettings() returns a default value 0xFF if this parameter is not returned by the device
<i>pIPv6AddPref</i>	<ul style="list-style-type: none"> IPv6 address preference Preferred IPv6 address to be assigned to the TE; actual assigned address is negotiated with the network and may differ from this value; if not specified, the IPv6 address is obtained automatically from the network
<i>pUMTSReqQoS</i> ↔ <i>SSigInd</i>	<ul style="list-style-type: none"> UMTS requested QoS with Signalling Indication flag
<i>pUMTSMinQoS</i> ↔ <i>SSigInd</i>	<ul style="list-style-type: none"> UMTS minimum QoS with Signalling Indication flag
<i>pPrimaryDNS</i> ↔ <i>Pv6addpref</i>	<ul style="list-style-type: none"> Primary DNS IPv6 address preference <ul style="list-style-type: none"> The value may be used as a preference during negotiation with the network; if not specified, the wireless device will attempt to obtain the DNS address automatically from the network; the negotiated value is provided to the host via DHCP
<i>pSecondaryDNS</i> ↔ <i>IPv6addpref</i>	<ul style="list-style-type: none"> Secondary DNS IPv6 address preference
<i>paddr</i> ↔ <i>AllocationPref</i>	<ul style="list-style-type: none"> DHCP/NAS preference <ul style="list-style-type: none"> This enumerated value may be used to indicate the address allocation preference <ul style="list-style-type: none"> * 0 - NAS signaling is used for address allocation * 1 - DHCP is used for address allocation
<i>pQoSClassID</i>	<ul style="list-style-type: none"> 3GPP LTE QoS parameters
<i>pAPNDisabled</i> ↔ <i>Flag</i>	<ul style="list-style-type: none"> Optional 1 uint8_t Flag indicating if the APN is disabled/enabled If set, the profile can not be used for making data calls Any data call is failed locally Values: <ul style="list-style-type: none"> 0 - FALSE(default) 1 - True This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().
<i>pPDNInactiv</i> ↔ <i>Timeout</i>	<ul style="list-style-type: none"> Optional 4 uint8_ts indicating the duration of inactivity timer in seconds If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected Default value of zero indicates infinite value This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().

<i>pAPNClass</i>	<ul style="list-style-type: none"> • Optional 1 uint8_t numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().
------------------	---

8.278.2 Field Documentation

8.278.2.1 uint8_t* LibpackProfile3GPP::pAddrAllocPref

8.278.2.2 uint8_t* LibpackProfile3GPP::pAPNClass

8.278.2.3 uint8_t* LibpackProfile3GPP::pAPNDisabledFlag

8.278.2.4 uint8_t* LibpackProfile3GPP::pAPNName

8.278.2.5 uint16_t* LibpackProfile3GPP::pAPNnameSize

8.278.2.6 uint8_t* LibpackProfile3GPP::pAuthenticationPref

8.278.2.7 LibPackGPRSRequestedQoS* LibpackProfile3GPP::pGPRSMinimumQoS

8.278.2.8 LibPackGPRSRequestedQoS* LibpackProfile3GPP::pGPRSRequestedQoS

8.278.2.9 uint8_t* LibpackProfile3GPP::pImCnFlag

8.278.2.10 uint32_t* LibpackProfile3GPP::pIPv4AddrPref

8.278.2.11 uint16_t* LibpackProfile3GPP::pIPv6AddPref

8.278.2.12 uint8_t* LibpackProfile3GPP::pPassword

8.278.2.13 uint16_t* LibpackProfile3GPP::pPasswordSize

8.278.2.14 uint8_t* LibpackProfile3GPP::pPcsfAddrUsingDhcp

8.278.2.15 uint8_t* LibpackProfile3GPP::pPcsfAddrUsingPCO

8.278.2.16 uint32_t* LibpackProfile3GPP::pPDNInactivTimeout

8.278.2.17 uint8_t* LibpackProfile3GPP::pPdpAccessConFlag

8.278.2.18 uint8_t* LibpackProfile3GPP::pPdpContext

- 8.278.2.19 `uint8_t*` LibpackProfile3GPP::pPdpDataCompType
- 8.278.2.20 `uint8_t*` LibpackProfile3GPP::pPdpHdrCompType
- 8.278.2.21 `uint8_t*` LibpackProfile3GPP::pPDPTtype
- 8.278.2.22 `uint32_t*` LibpackProfile3GPP::pPriDNSIPv4AddPref
- 8.278.2.23 `uint16_t*` LibpackProfile3GPP::pPriDNSIPv6addpref
- 8.278.2.24 `uint8_t*` LibpackProfile3GPP::pPrimaryID
- 8.278.2.25 `uint8_t*` LibpackProfile3GPP::pProfilename
- 8.278.2.26 `uint16_t*` LibpackProfile3GPP::pProfilenameSize
- 8.278.2.27 `LibPackQosClassID*` LibpackProfile3GPP::pQosClassID
- 8.278.2.28 `uint32_t*` LibpackProfile3GPP::pSecDNSIPv4AddPref
- 8.278.2.29 `uint16_t*` LibpackProfile3GPP::pSecDNSIPv6addpref
- 8.278.2.30 `uint8_t*` LibpackProfile3GPP::pSecondaryFlag
- 8.278.2.31 `LibPackTFTIDParams*` LibpackProfile3GPP::pTFTID1Params
- 8.278.2.32 `LibPackTFTIDParams*` LibpackProfile3GPP::pTFTID2Params
- 8.278.2.33 `LibPackUMTSQoS*` LibpackProfile3GPP::pUMTSMinQoS
- 8.278.2.34 `LibPackUMTSReqQoSSigInd*` LibpackProfile3GPP::pUMTSMinQoSsigInd
- 8.278.2.35 `LibPackUMTSQoS*` LibpackProfile3GPP::pUMTSReqQoS
- 8.278.2.36 `LibPackUMTSReqQoSSigInd*` LibpackProfile3GPP::pUMTSReqQoSSigInd
- 8.278.2.37 `uint8_t*` LibpackProfile3GPP::pUsername
- 8.278.2.38 `uint16_t*` LibpackProfile3GPP::pUsernameSize

8.279 LibpackProfile3GPP2 Struct Reference

Data Fields

- `uint8_t *` [pNegoDnsSvrPref](#)
- `uint32_t *` [pPppSessCloseTimerDO](#)

- uint32_t * pPppSessCloseTimer1x
- uint8_t * pAllowLinger
- uint16_t * pLcpAckTimeout
- uint16_t * plpcpAckTimeout
- uint16_t * pAuthTimeout
- uint8_t * pLcpCreqRetryCount
- uint8_t * plpcpCreqRetryCount
- uint8_t * pAuthRetryCount
- uint8_t * pAuthProtocol
- uint8_t * pUserId
- uint16_t * pUserIdSize
- uint8_t * pAuthPassword
- uint16_t * pAuthPasswordSize
- uint8_t * pDataRate
- uint32_t * pAppType
- uint8_t * pDataMode
- uint8_t * pAppPriority
- uint8_t * pApnString
- uint16_t * pApnStringSize
- uint8_t * pPdnType
- uint8_t * plsPcscfAddressNedded
- uint32_t * pPrimaryV4DnsAddress
- uint32_t * pSecondaryV4DnsAddress
- uint16_t * pPriV6DnsAddress
- uint16_t * pSecV6DnsAddress
- uint8_t * pRATType
- uint8_t * pAPNEnabled3GPP2
- uint32_t * pPDNInactivTimeout3GPP2
- uint8_t * pAPNClass3GPP2

8.279.1 Detailed Description

This structure contains the 3GPP2 profile parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>pNegoDnsSrvr</i> ↔ <i>Pref</i>	<ul style="list-style-type: none"> • Negotiate DNS Server Preference <ul style="list-style-type: none"> – 1 - (TRUE)implies request DNS addresses from the PDSN – 0 - (FALSE)implies do not request DNS addresses from the PDSN – Default value is 1 (TRUE)
<i>pPppSess</i> ↔ <i>CloseTimerDO</i>	<ul style="list-style-type: none"> • PPP Session Close Timer for DO <ul style="list-style-type: none"> – Timer value (in seconds) on DO indicating how long the PPP Session should linger before closing down

<i>pPppSess↔ CloseTimer1x</i>	<ul style="list-style-type: none"> • PPP Session Close Timer for 1X <ul style="list-style-type: none"> – Timer value (in seconds) on 1X indicating how long the PPP session should linger before closing down
<i>pAllowLinger</i>	<ul style="list-style-type: none"> • Allow/disallow lingering of interface <ul style="list-style-type: none"> – 1 -(TRUE) implies allow lingering – 0 -(FALSE) implies do not allow lingering
<i>pLcpAckTimeout</i>	<ul style="list-style-type: none"> • LCP ACK Timeout <ul style="list-style-type: none"> – Value of LCP ACK Timeout in milliseconds
<i>plpcpAck↔ Timeout</i>	<ul style="list-style-type: none"> • IPCP ACK Timeout <ul style="list-style-type: none"> – Value of IPCP ACK Timeout in milliseconds
<i>pAuthTimeout</i>	<ul style="list-style-type: none"> • AUTH Timeout <ul style="list-style-type: none"> – Value of Authentication Timeout in milliseconds
<i>pLcpCreq↔ RetryCount</i>	<ul style="list-style-type: none"> • LCP Configuration Request Retry Count
<i>plpcpCreq↔ RetryCount</i>	<ul style="list-style-type: none"> • IPCP Configuration Request Retry Count
<i>pAuthRetry↔ Count</i>	<ul style="list-style-type: none"> • Authentication Retry Count value
<i>pAuthProtocol</i>	<ul style="list-style-type: none"> • Authentication Protocol <ul style="list-style-type: none"> – 1 - PAP – 2 - CHAP – 3 - PAP or CHAP
<i>pUserId</i>	<ul style="list-style-type: none"> • User ID to be used during data network authentication • maximum length allowed is 127 uint8_ts; • QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.
<i>pUserIdSize;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pUserId field. Size of this parameter is 2 uint8_ts.
<i>pAuthPassword</i>	<ul style="list-style-type: none"> • Password to be used during data network authentication; • maximum length allowed is 127 uint8_ts • QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.

<i>pAuth</i> ↔ <i>PasswordSize;</i>	<ul style="list-style-type: none"> This parameter is an input parameter and should be initialised to the size of <i>pAuthPassword</i> field. Size of this parameter is 2 <i>uint8_ts</i>.
<i>pDataRate</i>	<ul style="list-style-type: none"> Data Rate Requested <ul style="list-style-type: none"> 0 - Low (Low speed Service Options (SO15) only) 1 - Medium (SO33 + low R-SCH) 2 - High (SO33 + high R-SCH) Default is 2
<i>pAppType</i>	<ul style="list-style-type: none"> Application Type: <ul style="list-style-type: none"> 0x00000001 - Default Application Type 0x00000020 - LBS Application Type 0x00000040 - Tethered Application Type This parameter is not used while creating/modifying a profile
<i>pDataMode</i>	<ul style="list-style-type: none"> Data Mode to use: <ul style="list-style-type: none"> 0 - CDMA or HDR (Hybrid 1X/1xEV-DO) 1 - CDMA Only (1X only) 2 - HDR Only (1xEV-DO only) Default is 0
<i>pAppPriority</i>	<ul style="list-style-type: none"> Application Priority <ul style="list-style-type: none"> Numerical 1 <i>uint8_t</i> value defining the application priority; higher value implies higher priority This parameter is not used while creating/modifying a profile
<i>pApnString</i>	<ul style="list-style-type: none"> String representing the Access Point Name maximum length allowed is 100 <i>uint8_ts</i> QMI_ERR_ARG_TOO_LONG will be returned if the APN name is too long.
<i>pApnStringSize;</i>	<ul style="list-style-type: none"> This parameter is an input parameter and should be initialised to the size of <i>pApnString</i> field. Size of this parameter is 2 <i>uint8_ts</i>.
<i>pPdnType</i>	<ul style="list-style-type: none"> Packed Data Network Type Requested: <ul style="list-style-type: none"> 0 - IPv4 PDN Type 1 - IPv6 PDN Type 2 - IPv4 or IPv6 PDN Type 3 - Unspecified PDN Type (implying no preference)
<i>plsPcsct</i> ↔ <i>AddressNedded</i>	<ul style="list-style-type: none"> This boolean value is used to control if PCSCF address is requested from PDSN <ul style="list-style-type: none"> 1 -(TRUE) implies request for PCSCF value from the PDSN 0 -(FALSE) implies do not request for PCSCF value from the PDSN

<i>pPrimaryV4↔ DnsAddress</i>	<ul style="list-style-type: none"> IPv4 Primary DNS address <ul style="list-style-type: none"> The Primary IPv4 DNS address that can be statically assigned to the UE
<i>pSecondaryV4↔ DnsAddress</i>	<ul style="list-style-type: none"> IPv4 Secondary DNS address <ul style="list-style-type: none"> The Secondary IPv4 DNS address that can be statically assigned to the UE
<i>pPriV6Dns↔ Address</i>	<ul style="list-style-type: none"> Primary IPv6 DNS address <ul style="list-style-type: none"> The Primary IPv6 DNS address that can be statically assigned to the UE
<i>pSecV6Dns↔ Address</i>	<ul style="list-style-type: none"> Secondary IPv6 DNS address <ul style="list-style-type: none"> The Secondary IPv6 DNS address that can be statically assigned to the UE
<i>pRATType</i>	<ul style="list-style-type: none"> Optional 1 uint8_t Flag indicating RAT Type Values: <ul style="list-style-type: none"> 1 - HRPD 2 - EHRPD 3 - HRPD_EHRPD This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().
<i>pAPN↔ Enabled3GPP2</i>	<ul style="list-style-type: none"> Optional 1 uint8_t Flag indicating if the APN is disabled/enabled If disabled, the profile can not be used for making data calls Values: <ul style="list-style-type: none"> 0 - Disabled 1 - Enabled(default value) This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().
<i>pPDNInactiv↔ Timeout3GPP2</i>	<ul style="list-style-type: none"> Optional 4 uint8_ts indicating the duration of inactivity timer in seconds If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected Default value of zero indicates infinite value This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().
<i>pAPNClass3G↔ PP2</i>	<ul style="list-style-type: none"> Optional 1 uint8_t numeric identifier representing the APN in profile Can be set and queried but is not used by the modem This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().

8.279.2 Field Documentation

8.279.2.1 uint8_t* LibpackProfile3GPP2::pAllowLinger

- 8.279.2.2 `uint8_t*` `LibpackProfile3GPP2::pAPNClass3GPP2`
- 8.279.2.3 `uint8_t*` `LibpackProfile3GPP2::pAPNEnabled3GPP2`
- 8.279.2.4 `uint8_t*` `LibpackProfile3GPP2::pApnString`
- 8.279.2.5 `uint16_t*` `LibpackProfile3GPP2::pApnStringSize`
- 8.279.2.6 `uint8_t*` `LibpackProfile3GPP2::pAppPriority`
- 8.279.2.7 `uint32_t*` `LibpackProfile3GPP2::pAppType`
- 8.279.2.8 `uint8_t*` `LibpackProfile3GPP2::pAuthPassword`
- 8.279.2.9 `uint16_t*` `LibpackProfile3GPP2::pAuthPasswordSize`
- 8.279.2.10 `uint8_t*` `LibpackProfile3GPP2::pAuthProtocol`
- 8.279.2.11 `uint8_t*` `LibpackProfile3GPP2::pAuthRetryCount`
- 8.279.2.12 `uint16_t*` `LibpackProfile3GPP2::pAuthTimeout`
- 8.279.2.13 `uint8_t*` `LibpackProfile3GPP2::pDataMode`
- 8.279.2.14 `uint8_t*` `LibpackProfile3GPP2::pDataRate`
- 8.279.2.15 `uint16_t*` `LibpackProfile3GPP2::plpcpAckTimeout`
- 8.279.2.16 `uint8_t*` `LibpackProfile3GPP2::plpcpCreqRetryCount`
- 8.279.2.17 `uint8_t*` `LibpackProfile3GPP2::plsPscsfAddressNedded`
- 8.279.2.18 `uint16_t*` `LibpackProfile3GPP2::pLcpAckTimeout`
- 8.279.2.19 `uint8_t*` `LibpackProfile3GPP2::pLcpCreqRetryCount`
- 8.279.2.20 `uint8_t*` `LibpackProfile3GPP2::pNegoDnsSrvrPref`
- 8.279.2.21 `uint32_t*` `LibpackProfile3GPP2::pPDNInactivTimeout3GPP2`
- 8.279.2.22 `uint8_t*` `LibpackProfile3GPP2::pPdnType`
- 8.279.2.23 `uint32_t*` `LibpackProfile3GPP2::pPppSessCloseTimer1x`
- 8.279.2.24 `uint32_t*` `LibpackProfile3GPP2::pPppSessCloseTimerDO`

- 8.279.2.25 `uint32_t*` LibpackProfile3GPP2::pPrimaryV4DnsAddress
- 8.279.2.26 `uint16_t*` LibpackProfile3GPP2::pPriV6DnsAddress
- 8.279.2.27 `uint8_t*` LibpackProfile3GPP2::pRATType
- 8.279.2.28 `uint32_t*` LibpackProfile3GPP2::pSecondaryV4DnsAddress
- 8.279.2.29 `uint16_t*` LibpackProfile3GPP2::pSecV6DnsAddress
- 8.279.2.30 `uint8_t*` LibpackProfile3GPP2::pUserId
- 8.279.2.31 `uint16_t*` LibpackProfile3GPP2::pUserIdSize

8.280 LibPackprofile_3GPP Struct Reference

Data Fields

- `uint8_t*` [pProfilename](#)
- `uint16_t*` [pProfilenameSize](#)
- `uint8_t*` [pPDPTtype](#)
- `uint8_t*` [pPdpHdrCompType](#)
- `uint8_t*` [pPdpDataCompType](#)
- `uint8_t*` [pAPNName](#)
- `uint16_t*` [pAPNnameSize](#)
- `uint32_t*` [pPriDNSIPv4AddPref](#)
- `uint32_t*` [pSecDNSIPv4AddPref](#)
- [LibPackUMTSQoS](#) * [pUMTSReqQoS](#)
- [LibPackUMTSQoS](#) * [pUMTSMinQoS](#)
- [LibPackGPRSRequestedQoS](#) * [pGPRSRequestedQoS](#)
- [LibPackGPRSRequestedQoS](#) * [pGPRSMinimumQoS](#)
- `uint8_t*` [pUsername](#)
- `uint16_t*` [pUsernameSize](#)
- `uint8_t*` [pPassword](#)
- `uint16_t*` [pPasswordSize](#)
- `uint8_t*` [pAuthenticationPref](#)
- `uint32_t*` [pIPv4AddrPref](#)
- `uint8_t*` [pPcsfAddrUsingPCO](#)
- `uint8_t*` [pPdpAccessConFlag](#)
- `uint8_t*` [pPcsfAddrUsingDhcp](#)
- `uint8_t*` [pImCnFlag](#)
- [LibPackTFTIDParams](#) * [pTFTID1Params](#)
- [LibPackTFTIDParams](#) * [pTFTID2Params](#)
- `uint8_t*` [pPdpContext](#)
- `uint8_t*` [pSecondaryFlag](#)
- `uint8_t*` [pPrimaryID](#)
- `uint16_t*` [pIPv6AddPref](#)
- [LibPackUMTSReqQoSSigInd](#) * [pUMTSReqQoSSigInd](#)
- [LibPackUMTSReqQoSSigInd](#) * [pUMTSMinQoSsigInd](#)
- `uint16_t*` [pPriDNSIPv6addpref](#)
- `uint16_t*` [pSecDNSIPv6addpref](#)
- `uint8_t*` [pAddrAllocPref](#)
- [LibPackQoSClassID](#) * [pQoSClassID](#)
- `uint8_t*` [pAPNDisabledFlag](#)
- `uint32_t*` [pPDNInactivTimeout](#)
- `uint8_t*` [pAPNClass](#)

8.280.1 Detailed Description

This structure contains Input parameters of SLQSCreateProfile and SLQSModifyProfile and output parameters of SLQSGetProfileSettings

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>pProfileName</i>	<ul style="list-style-type: none"> • One or more bytes describing the profile
<i>pProfilename</i> ↔ <i>Size;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pProfileName field. Size of this parameter is 2 bytes.
<i>pPDPTType</i>	<ul style="list-style-type: none"> • Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile <ul style="list-style-type: none"> – 0x00 - PDP-IP (IPv4) – 0x01 - PDP-PPP – 0x02 - PDP-IPV6 – 0x03 - PDP-IPV4V6
<i>pPdpHdr</i> ↔ <i>CompType</i>	<ul style="list-style-type: none"> • PDP header compression type <ul style="list-style-type: none"> – 0 - PDP header compression is OFF – 1 - Manufacturer preferred compression – 2 - PDP header compression based on RFC 1144 – 3 - PDP header compression based on RFC 25074 PDP header compression based on RFC 3095
<i>pPdpData</i> ↔ <i>CompType</i>	<ul style="list-style-type: none"> • PDP data compression type <ul style="list-style-type: none"> – 0 - PDP data compression is OFF – 1 - Manufacturer preferred compression – 2 - V.42BIS data compression – 3 - V.44 data compression
<i>pAPNName</i>	<ul style="list-style-type: none"> • Access point name
<i>pAPNnameSize;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pAPNName field. Size of this parameter is 2 bytes.
<i>pPriDNSIPv4</i> ↔ <i>AddPref</i>	<ul style="list-style-type: none"> • Primary DNS IPv4 Address Preference
<i>pSecDNSIPv4</i> ↔ <i>AddPref</i>	<ul style="list-style-type: none"> • Secondary DNS IPv4 Address Preference

<i>pUMTSReqQoS</i>	<ul style="list-style-type: none"> • UMTS Requested QoS
<i>pUMTSMinQoS</i>	<ul style="list-style-type: none"> • UMTS Minimum QoS
<i>pGPRS↔ RequestedQoS</i>	<ul style="list-style-type: none"> • GPRS Minimum QoS
<i>pUsername</i>	<ul style="list-style-type: none"> • User name
<i>pUsernameSize;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pUsername field. Size of this parameter is 2 bytes.
<i>pPassword</i>	<ul style="list-style-type: none"> • Password
<i>pPasswordSize;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pPassword field. Size of this parameter is 2 bytes.
<i>p↔ Authentication↔ Pref</i>	<ul style="list-style-type: none"> • Authentication Preference <ul style="list-style-type: none"> – Bit map that indicates the authentication algorithm preference <ul style="list-style-type: none"> * Bit 0 - PAP preference <ul style="list-style-type: none"> • 0 - PAP is never performed • 1 - PAP may be performed * Bit 1 - CHAP preference <ul style="list-style-type: none"> • 0 - CHAP is never performed • 1 - CHAP may be performed * If more than one bit is set, then the device decides which authentication procedure is performed while setting up the data session. For example, the device may have a policy to select the most secure authentication mechanism.
<i>pIPv4AddrPref</i>	<ul style="list-style-type: none"> • IPv4 Address Preference
<i>pPcscfAddr↔ UsingPCO</i>	<ul style="list-style-type: none"> • P-CSCF Address using PCO Flag <ul style="list-style-type: none"> – 1 - (TRUE) implies request PCSCF address using PCO – 0 - (FALSE) implies do not request By default, this value is 0
<i>pPdpAccess↔ ConFlag</i>	<ul style="list-style-type: none"> • PDP access control flag <ul style="list-style-type: none"> – 0 - PDP access control none – 1 - PDP access control reject – 2 - PDP access control permission

<i>pPcscfAddr</i> ↔ <i>UsingDhcp</i>	<ul style="list-style-type: none"> • P-CSCF address using DHCP <ul style="list-style-type: none"> – 1 - (TRUE) implies Request PCSCF address using DHCP – 0 - (FALSE) implies do not request By default, value is 0
<i>pImCnFlag</i>	<ul style="list-style-type: none"> • IM CN flag <ul style="list-style-type: none"> – 1 - (TRUE) implies request IM CN flag for this profile – 0 - (FALSE) implies do not request IM CN flag for this profile
<i>pTFTID1Params</i>	<ul style="list-style-type: none"> • Traffic Flow Template
<i>pTFTID2Params</i>	<ul style="list-style-type: none"> • Traffic Flow Template
<i>pPdpContext</i>	<ul style="list-style-type: none"> • PDP context number
<i>pSecondaryFlag</i>	<ul style="list-style-type: none"> • PDP context secondary flag <ul style="list-style-type: none"> – 1 - (TRUE) implies this is secondary profile – 0 - (FALSE) implies this is not secondary profile
<i>pPrimaryID</i>	<ul style="list-style-type: none"> • PDP context primary ID • function SLQSGetProfileSettings() returns a default value 0xFF if this parameter is not returned by the device
<i>pIPv6AddPref</i>	<ul style="list-style-type: none"> • IPv6 address preference Preferred IPv6 address to be assigned to the TE; actual assigned address is negotiated with the network and may differ from this value; if not specified, the IPv6 address is obtained automatically from the network
<i>pUMTSReqQo</i> ↔ <i>SSigInd</i>	<ul style="list-style-type: none"> • UMTS requested QoS with Signalling Indication flag
<i>pUMTSMinQo</i> ↔ <i>SSigInd</i>	<ul style="list-style-type: none"> • UMTS minimum QoS with Signalling Indication flag
<i>pPrimaryDNSI</i> ↔ <i>Pv6addpref</i>	<ul style="list-style-type: none"> • Primary DNS IPv6 address preference <ul style="list-style-type: none"> – The value may be used as a preference during negotiation with the network; if not specified, the wireless device will attempt to obtain the DNS address automatically from the network; the negotiated value is provided to the host via DHCP
<i>pSecondaryD</i> ↔ <i>NSIPv6addpref</i>	<ul style="list-style-type: none"> • Secondary DNS IPv6 address preference

<i>pAddr</i> ↔ <i>AllocationPref</i>	<ul style="list-style-type: none"> DHCP/NAS preference <ul style="list-style-type: none"> This enumerated value may be used to indicate the address allocation preference <ul style="list-style-type: none"> * 0 - NAS signaling is used for address allocation * 1 - DHCP is used for address allocation
<i>pQosClassID</i>	<ul style="list-style-type: none"> 3GPP LTE QoS parameters
<i>pAPNDisabled</i> ↔ <i>Flag</i>	<ul style="list-style-type: none"> Optional 1 uint8_t Flag indicating if the APN is disabled/enabled If set, the profile can not be used for making data calls Any data call is failed locally Values: <ul style="list-style-type: none"> 0 - FALSE(default) 1 - True This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().
<i>pPDNInactiv</i> ↔ <i>Timeout</i>	<ul style="list-style-type: none"> Optional 4 Bytes indicating the duration of inactivity timer in seconds If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected Default value of zero indicates infinite value This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().
<i>pAPNClass</i>	<ul style="list-style-type: none"> Optional 1 uint8_t numeric identifier representing the APN in profile Can be set and queried but is not used by the modem This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().

8.280.2 Field Documentation

8.280.2.1 uint8_t* LibPackprofile_3GPP::pAddrAllocPref

8.280.2.2 uint8_t* LibPackprofile_3GPP::pAPNClass

8.280.2.3 uint8_t* LibPackprofile_3GPP::pAPNDisabledFlag

8.280.2.4 uint8_t* LibPackprofile_3GPP::pAPNName

8.280.2.5 uint16_t* LibPackprofile_3GPP::pAPNNameSize

8.280.2.6 uint8_t* LibPackprofile_3GPP::pAuthenticationPref

8.280.2.7 **LibPackGPRSRequestedQoS*** LibPackprofile_3GPP::pGPRSMinimumQoS

8.280.2.8 **LibPackGPRSRequestedQoS*** LibPackprofile_3GPP::pGPRSRequestedQos

8.280.2.9 **uint8_t*** LibPackprofile_3GPP::plmCnFlag

8.280.2.10 **uint32_t*** LibPackprofile_3GPP::pIPv4AddrPref

8.280.2.11 **uint16_t*** LibPackprofile_3GPP::pIPv6AddPref

8.280.2.12 **uint8_t*** LibPackprofile_3GPP::pPassword

8.280.2.13 **uint16_t*** LibPackprofile_3GPP::pPasswordSize

8.280.2.14 **uint8_t*** LibPackprofile_3GPP::pPcscfAddrUsingDhcp

8.280.2.15 **uint8_t*** LibPackprofile_3GPP::pPcscfAddrUsingPCO

8.280.2.16 **uint32_t*** LibPackprofile_3GPP::pPDNInactivTimeout

8.280.2.17 **uint8_t*** LibPackprofile_3GPP::pPdpAccessConFlag

8.280.2.18 **uint8_t*** LibPackprofile_3GPP::pPdpContext

8.280.2.19 **uint8_t*** LibPackprofile_3GPP::pPdpDataCompType

8.280.2.20 **uint8_t*** LibPackprofile_3GPP::pPdpHdrCompType

8.280.2.21 **uint8_t*** LibPackprofile_3GPP::pPDPTtype

8.280.2.22 **uint32_t*** LibPackprofile_3GPP::pPriDNSIPv4AddPref

8.280.2.23 **uint16_t*** LibPackprofile_3GPP::pPriDNSIPv6addpref

8.280.2.24 **uint8_t*** LibPackprofile_3GPP::pPrimaryID

8.280.2.25 **uint8_t*** LibPackprofile_3GPP::pProfilename

8.280.2.26 **uint16_t*** LibPackprofile_3GPP::pProfilenameSize

8.280.2.27 **LibPackQosClassID*** LibPackprofile_3GPP::pQosClassID

8.280.2.28 **uint32_t*** LibPackprofile_3GPP::pSecDNSIPv4AddPref

8.280.2.29 **uint16_t*** LibPackprofile_3GPP::pSecDNSIPv6addpref

- 8.280.2.30 `uint8_t*` `LibPackprofile_3GPP::pSecondaryFlag`
- 8.280.2.31 `LibPackTFTIDParams*` `LibPackprofile_3GPP::pTFTID1Params`
- 8.280.2.32 `LibPackTFTIDParams*` `LibPackprofile_3GPP::pTFTID2Params`
- 8.280.2.33 `LibPackUMTSQoS*` `LibPackprofile_3GPP::pUMTSMinQoS`
- 8.280.2.34 `LibPackUMTSReqQoSSigInd*` `LibPackprofile_3GPP::pUMTSMinQoSsigInd`
- 8.280.2.35 `LibPackUMTSQoS*` `LibPackprofile_3GPP::pUMTSReqQoS`
- 8.280.2.36 `LibPackUMTSReqQoSSigInd*` `LibPackprofile_3GPP::pUMTSReqQoSSigInd`
- 8.280.2.37 `uint8_t*` `LibPackprofile_3GPP::pUsername`
- 8.280.2.38 `uint16_t*` `LibPackprofile_3GPP::pUsernameSize`

8.281 LibPackprofile_3GPP2 Struct Reference

Data Fields

- `uint8_t*` `pNegoDnsSrvrPref`
- `uint32_t*` `pPppSessCloseTimerDO`
- `uint32_t*` `pPppSessCloseTimer1x`
- `uint8_t*` `pAllowLinger`
- `uint16_t*` `pLcpAckTimeout`
- `uint16_t*` `pIpccpAckTimeout`
- `uint16_t*` `pAuthTimeout`
- `uint8_t*` `pLcpCreqRetryCount`
- `uint8_t*` `pIpccpCreqRetryCount`
- `uint8_t*` `pAuthRetryCount`
- `uint8_t*` `pAuthProtocol`
- `uint8_t*` `pUserId`
- `uint16_t*` `pUserIdSize`
- `uint8_t*` `pAuthPassword`
- `uint16_t*` `pAuthPassword_tSize`
- `uint8_t*` `pDataRate`
- `uint32_t*` `pAppType`
- `uint8_t*` `pDataMode`
- `uint8_t*` `pAppPriority`
- `uint8_t*` `pApnString`
- `uint16_t*` `pApnStringSize`
- `uint8_t*` `pPdnType`
- `uint8_t*` `pIsPcscfAddressNedded`
- `uint32_t*` `pPrimaryV4DnsAddress`
- `uint32_t*` `pSecondaryV4DnsAddress`
- `uint16_t*` `pPriV6DnsAddress`
- `uint16_t*` `pSecV6DnsAddress`
- `uint8_t*` `pRATType`
- `uint8_t*` `pAPNEnabled3GPP2`
- `uint32_t*` `pPDNInactivTimeout3GPP2`
- `uint8_t*` `pAPNClass3GPP2`

8.281.1 Detailed Description

This structure contains the 3GPP2 profile parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>pNegoDnsSrvr</i> ↔ <i>Pref</i>	<ul style="list-style-type: none"> • Negotiate DNS Server Preference <ul style="list-style-type: none"> – 1 - (TRUE) implies request DNS addresses from the PDSN – 0 - (FALSE) implies do not request DNS addresses from the PDSN – Default value is 1 (TRUE)
<i>pPppSess</i> ↔ <i>CloseTimerDO</i>	<ul style="list-style-type: none"> • PPP Session Close Timer for DO <ul style="list-style-type: none"> – Timer value (in seconds) on DO indicating how long the PPP Session should linger before closing down
<i>pPppSess</i> ↔ <i>CloseTimer1x</i>	<ul style="list-style-type: none"> • PPP Session Close Timer for 1X <ul style="list-style-type: none"> – Timer value (in seconds) on 1X indicating how long the PPP session should linger before closing down
<i>pAllowLinger</i>	<ul style="list-style-type: none"> • Allow/disallow lingering of interface <ul style="list-style-type: none"> – 1 -(TRUE) implies allow lingering – 0 -(FALSE) implies do not allow lingering
<i>pLcpAckTimeout</i>	<ul style="list-style-type: none"> • LCP ACK Timeout <ul style="list-style-type: none"> – Value of LCP ACK Timeout in milliseconds
<i>pIpccpAck</i> ↔ <i>Timeout</i>	<ul style="list-style-type: none"> • IPCP ACK Timeout <ul style="list-style-type: none"> – Value of IPCP ACK Timeout in milliseconds
<i>pAuthTimeout</i>	<ul style="list-style-type: none"> • AUTH Timeout <ul style="list-style-type: none"> – Value of Authentication Timeout in milliseconds
<i>pLcpCreq</i> ↔ <i>RetryCount</i>	<ul style="list-style-type: none"> • LCP Configuration Request Retry Count
<i>pIpccpCreq</i> ↔ <i>RetryCount</i>	<ul style="list-style-type: none"> • IPCP Configuration Request Retry Count
<i>pAuthRetry</i> ↔ <i>Count</i>	<ul style="list-style-type: none"> • Authentication Retry Count value
<i>pAuthProtocol</i>	<ul style="list-style-type: none"> • Authentication Protocol <ul style="list-style-type: none"> – 1 - PAP – 2 - CHAP – 3 - PAP or CHAP
	Generated by Doxygen

<i>pUserId</i>	<ul style="list-style-type: none"> User ID to be used during data network authentication maximum length allowed is 127 bytes; QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.
<i>pUserIdSize;</i>	<ul style="list-style-type: none"> This parameter is an input parameter and should be initialised to the size of pUserId field. Size of this parameter is 2 bytes.
<i>pAuthPassword</i>	<ul style="list-style-type: none"> Password to be used during data network authentication; maximum length allowed is 127 bytes QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.
<i>pAuthPasswordSize;</i>	<ul style="list-style-type: none"> This parameter is an input parameter and should be initialised to the size of pAuthPassword field. Size of this parameter is 2 bytes.
<i>pDataRate</i>	<ul style="list-style-type: none"> Data Rate Requested <ul style="list-style-type: none"> 0 - Low (Low speed Service Options (SO15) only) 1 - Medium (SO33 + low R-SCH) 2 - High (SO33 + high R-SCH) Default is 2
<i>pAppType</i>	<ul style="list-style-type: none"> Application Type: <ul style="list-style-type: none"> 0x00000001 - Default Application Type 0x00000020 - LBS Application Type 0x00000040 - Tethered Application Type This parameter is not used while creating/modifying a profile
<i>pDataMode</i>	<ul style="list-style-type: none"> Data Mode to use: <ul style="list-style-type: none"> 0 - CDMA or HDR (Hybrid 1X/1xEV-DO) 1 - CDMA Only (1X only) 2 - HDR Only (1xEV-DO only) Default is 0
<i>pAppPriority</i>	<ul style="list-style-type: none"> Application Priority <ul style="list-style-type: none"> Numerical 1 uint8_t value defining the application priority; higher value implies higher priority This parameter is not used while creating/modifying a profile
<i>pApnString</i>	<ul style="list-style-type: none"> String representing the Access Point Name maximum length allowed is 100 bytes QMI_ERR_ARG_TOO_LONG will be returned if the APN name is too long.

<i>pApnStringSize;</i>	<ul style="list-style-type: none"> This parameter is an input parameter and should be initialised to the size of pApnString field. Size of this parameter is 2 bytes.
<i>pPdnType</i>	<ul style="list-style-type: none"> Packed Data Network Type Requested: <ul style="list-style-type: none"> 0 - IPv4 PDN Type 1 - IPv6 PDN Type 2 - IPv4 or IPv6 PDN Type 3 - Unspecified PDN Type (implying no preference)
<i>plsPcscf↔ AddressNedded</i>	<ul style="list-style-type: none"> This boolean value is used to control if PCSCF address is requested from PDSN <ul style="list-style-type: none"> 1 -(TRUE) implies request for PCSCF value from the PDSN 0 -(FALSE) implies do not request for PCSCF value from the PDSN
<i>pPrimaryV4↔ DnsAddress</i>	<ul style="list-style-type: none"> IPv4 Primary DNS address <ul style="list-style-type: none"> The Primary IPv4 DNS address that can be statically assigned to the UE
<i>pSecondaryV4↔ DnsAddress</i>	<ul style="list-style-type: none"> IPv4 Secondary DNS address <ul style="list-style-type: none"> The Secondary IPv4 DNS address that can be statically assigned to the UE
<i>pPriV6Dns↔ Address</i>	<ul style="list-style-type: none"> Primary IPv6 DNS address <ul style="list-style-type: none"> The Primary IPv6 DNS address that can be statically assigned to the UE
<i>pSecV6Dns↔ Address</i>	<ul style="list-style-type: none"> Secondary IPv6 DNS address <ul style="list-style-type: none"> The Secondary IPv6 DNS address that can be statically assigned to the UE
<i>pRATType</i>	<ul style="list-style-type: none"> Optional 1 uint8_t Flag indicating RAT Type Values: <ul style="list-style-type: none"> 1 - HRPD 2 - EHRPD 3 - HRPD_EHRPD This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().
<i>pAPN↔ Enabled3GPP2</i>	<ul style="list-style-type: none"> Optional 1 uint8_t Flag indicating if the APN is disabled/enabled If disabled, the profile can not be used for making data calls Values: <ul style="list-style-type: none"> 0 - Disabled 1 - Enabled(default value) This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().

<i>pPDNInactiv↔ Timeout3GPP2</i>	<ul style="list-style-type: none"> • Optional 4 Bytes indicating the duration of inactivity timer in seconds • If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected • Default value of zero indicates infinite value • This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().
<i>pAPNClass3G↔ PP2</i>	<ul style="list-style-type: none"> • Optional 1 uint8_t numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().

8.281.2 Field Documentation

8.281.2.1 uint8_t* LibPackprofile_3GPP2::pAllowLinger

8.281.2.2 uint8_t* LibPackprofile_3GPP2::pAPNClass3GPP2

8.281.2.3 uint8_t* LibPackprofile_3GPP2::pAPNEnabled3GPP2

8.281.2.4 uint8_t* LibPackprofile_3GPP2::pApnString

8.281.2.5 uint16_t* LibPackprofile_3GPP2::pApnStringSize

8.281.2.6 uint8_t* LibPackprofile_3GPP2::pAppPriority

8.281.2.7 uint32_t* LibPackprofile_3GPP2::pAppType

8.281.2.8 uint8_t* LibPackprofile_3GPP2::pAuthPassword

8.281.2.9 uint16_t* LibPackprofile_3GPP2::pAuthPassword_tSize

8.281.2.10 uint8_t* LibPackprofile_3GPP2::pAuthProtocol

8.281.2.11 uint8_t* LibPackprofile_3GPP2::pAuthRetryCount

8.281.2.12 uint16_t* LibPackprofile_3GPP2::pAuthTimeout

8.281.2.13 uint8_t* LibPackprofile_3GPP2::pDataMode

8.281.2.14 uint8_t* LibPackprofile_3GPP2::pDataRate

8.281.2.15 uint16_t* LibPackprofile_3GPP2::plpcpAckTimeout

- 8.281.2.16 `uint8_t*` `LibPackprofile_3GPP2::plpcpCreqRetryCount`
- 8.281.2.17 `uint8_t*` `LibPackprofile_3GPP2::plsPscsfAddressNedded`
- 8.281.2.18 `uint16_t*` `LibPackprofile_3GPP2::pLcpAckTimeout`
- 8.281.2.19 `uint8_t*` `LibPackprofile_3GPP2::pLcpCreqRetryCount`
- 8.281.2.20 `uint8_t*` `LibPackprofile_3GPP2::pNegoDnsSrvrPref`
- 8.281.2.21 `uint32_t*` `LibPackprofile_3GPP2::pPDNInactivTimeout3GPP2`
- 8.281.2.22 `uint8_t*` `LibPackprofile_3GPP2::pPdnType`
- 8.281.2.23 `uint32_t*` `LibPackprofile_3GPP2::pPppSessCloseTimer1x`
- 8.281.2.24 `uint32_t*` `LibPackprofile_3GPP2::pPppSessCloseTimerDO`
- 8.281.2.25 `uint32_t*` `LibPackprofile_3GPP2::pPrimaryV4DnsAddress`
- 8.281.2.26 `uint16_t*` `LibPackprofile_3GPP2::pPriV6DnsAddress`
- 8.281.2.27 `uint8_t*` `LibPackprofile_3GPP2::pRATType`
- 8.281.2.28 `uint32_t*` `LibPackprofile_3GPP2::pSecondaryV4DnsAddress`
- 8.281.2.29 `uint16_t*` `LibPackprofile_3GPP2::pSecV6DnsAddress`
- 8.281.2.30 `uint8_t*` `LibPackprofile_3GPP2::pUserId`
- 8.281.2.31 `uint16_t*` `LibPackprofile_3GPP2::pUserIdSize`

8.282 `LibPackQosClassID` Struct Reference

Data Fields

- `uint8_t` [QCI](#)
- `uint8_t` [gDIBitRate](#)
- `uint32_t` [maxDIBitRate](#)
- `uint32_t` [gUIBitRate](#)
- `uint32_t` [maxUIBitRate](#)

8.282.1 Detailed Description

structure contains 3GPP LTE QoS parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>QCI</i>	<ul style="list-style-type: none"> QOS specified using the QOS Class Identifier (QOS) values QCI value 0 - Requests the network to assign the appropriate QCI value QCI values 1-4 - Associated with guaranteed bit rates QCI values 5-9 - Associated with non-guaranteed bit rates
<i>gDIBitRate</i>	<ul style="list-style-type: none"> Guaranteed DL bit rate
<i>maxDIBitRate</i>	<ul style="list-style-type: none"> maxDIBitRate
<i>gUIBitRate</i>	<ul style="list-style-type: none"> Guaranteed UL bit rate
<i>maxUIBitRate</i>	<ul style="list-style-type: none"> Maximum UL bit rate

8.282.2 Field Documentation

8.282.2.1 `uint8_t LibPackQosClassID::gDIBitRate`

8.282.2.2 `uint32_t LibPackQosClassID::gUIBitRate`

8.282.2.3 `uint32_t LibPackQosClassID::maxDIBitRate`

8.282.2.4 `uint32_t LibPackQosClassID::maxUIBitRate`

8.282.2.5 `uint8_t LibPackQosClassID::QCI`

8.283 LibPackTFTIDParams Struct Reference

Data Fields

- `uint8_t filterId`
- `uint8_t eValid`
- `uint8_t ipVersion`
- `uint16_t * pSourceIP`
- `uint8_t sourceIPMask`
- `uint8_t nextHeader`
- `uint32_t destPortRangeStart`
- `uint16_t destPortRangeEnd`
- `uint16_t srcPortRangeStart`
- `uint16_t srcPortRangeEnd`
- `uint32_t IPSECSPi`
- `uint16_t tosMask`
- `uint32_t flowLabel`

8.283.1 Detailed Description

structure contains traffic flow template parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>filterId</i>	<ul style="list-style-type: none"> • Filter identifier
<i>eValid</i>	<ul style="list-style-type: none"> • Evaluation precedence index
<i>pVersion</i>	<ul style="list-style-type: none"> • IP version number <ul style="list-style-type: none"> – 4 - IPv4 – 6 - IPv6
<i>sourceIP</i>	<ul style="list-style-type: none"> • Source IP address <ul style="list-style-type: none"> – IPv4 - Fill the first 4 uint8_ts – IPv6 - Fill all the 16 uint8_ts
<i>sourceIPMask</i>	<ul style="list-style-type: none"> • Mask value for the source address
<i>nextHeader</i>	<ul style="list-style-type: none"> • Next header/protocol value
<i>destPort</i> ↔ <i>RangeStart</i>	<ul style="list-style-type: none"> • Start value of the destination port range
<i>destPort</i> ↔ <i>RangeEnd</i>	<ul style="list-style-type: none"> • End value of the destination port range
<i>srcPortRange</i> ↔ <i>Start</i>	<ul style="list-style-type: none"> • Start value of the source port range
<i>srcPortRange</i> ↔ <i>End</i>	<ul style="list-style-type: none"> • End value of the source port range
<i>IPSECSPi</i>	<ul style="list-style-type: none"> • IPSEC security parameter index
<i>tosMask</i>	<ul style="list-style-type: none"> • TOS mask (Traffic class for IPv6)
<i>flowLabel</i>	<ul style="list-style-type: none"> • Flow label

8.283.2 Field Documentation

8.283.2.1 `uint16_t` LibPackTFTIDParams::destPortRangeEnd

8.283.2.2 `uint32_t` LibPackTFTIDParams::destPortRangeStart

8.283.2.3 `uint8_t` LibPackTFTIDParams::eValid

8.283.2.4 `uint8_t` LibPackTFTIDParams::filterId

8.283.2.5 `uint32_t` LibPackTFTIDParams::flowLabel

8.283.2.6 `uint32_t` LibPackTFTIDParams::IPSECSPi

8.283.2.7 `uint8_t` LibPackTFTIDParams::ipVersion

8.283.2.8 `uint8_t` LibPackTFTIDParams::nextHeader

8.283.2.9 `uint16_t*` LibPackTFTIDParams::pSourceIP

8.283.2.10 `uint8_t` LibPackTFTIDParams::sourceIPMask

8.283.2.11 `uint16_t` LibPackTFTIDParams::srcPortRangeEnd

8.283.2.12 `uint16_t` LibPackTFTIDParams::srcPortRangeStart

8.283.2.13 `uint16_t` LibPackTFTIDParams::tosMask

8.284 LibPackUMTSQoS Struct Reference

Data Fields

- `uint8_t` [trafficClass](#)
- `uint32_t` [maxUplinkBitrate](#)
- `uint32_t` [maxDownlinkBitrate](#)
- `uint32_t` [grntUplinkBitrate](#)
- `uint32_t` [grntDownlinkBitrate](#)
- `uint8_t` [qosDeliveryOrder](#)
- `uint32_t` [maxSDUSize](#)
- `uint8_t` [sduErrorRatio](#)
- `uint8_t` [resBerRatio](#)
- `uint8_t` [deliveryErrSDU](#)
- `uint32_t` [transferDelay](#)
- `uint32_t` [trafficPriority](#)

8.284.1 Detailed Description

This structure contains the UMTS Quality Of Service Information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>trafficClass</i>	<ul style="list-style-type: none"> • 0x00 - Subscribed • 0x01 - Conversational • 0x02 - Streaming • 0x03 - Interactive • 0x04 - Background
<i>maxUplinkBitrate</i>	<ul style="list-style-type: none"> • Maximum uplink bit rate in bits/sec
<i>maxDownlink↔ Bitrate</i>	<ul style="list-style-type: none"> • Maximum downlink bit rate in bits/sec
<i>grntUplinkBitrate</i>	<ul style="list-style-type: none"> • Guaranteed uplink bit rate in bits/sec
<i>grntDownlink↔ Bitrate</i>	<ul style="list-style-type: none"> • Guranteed downlink bit rate in bits/sec
<i>qosDelivery↔ Order</i>	<ul style="list-style-type: none"> - Qos delivery order • 0x00 - Subscribe • 0x01 - delivery order on • 0x02 - delivery order off
<i>maxSDUSize</i>	<ul style="list-style-type: none"> • Maximum SDU size
<i>sduErrorRatio</i>	<ul style="list-style-type: none"> - SDU error ratio • Target value for fraction of SDUs lost or detected as erroneous. • 0x00 - Subscribe • 0x01 - $1 \cdot 10^{(-2)}$ • 0x02 - $7 \cdot 10^{(-3)}$ • 0x03 - $1 \cdot 10^{(-3)}$ • 0x04 - $1 \cdot 10^{(-4)}$ • 0x05 - $1 \cdot 10^{(-5)}$ • 0x06 - $1 \cdot 10^{(-6)}$ • 0x07 - $1 \cdot 10^{(-1)}$
<i>resBerRatio</i>	<ul style="list-style-type: none"> - Residual bit error ratio • Target value for undetected bit error ratio in in the delivered SDUs. • 0x00 - Subscribe • 0x01 - $5 \cdot 10^{(-2)}$ • 0x02 - $1 \cdot 10^{(-2)}$ • 0x03 - $5 \cdot 10^{(-3)}$ • 0x04 - $4 \cdot 10^{(-3)}$ • 0x05 - $1 \cdot 10^{(-3)}$ • 0x06 - $1 \cdot 10^{(-4)}$ • 0x07 - $1 \cdot 10^{(-5)}$ • 0x08 - $1 \cdot 10^{(-6)}$ • 0x09 - $1 \cdot 10^{(-8)}$

<i>deliveryErrSDU</i>	- Delivery of erroneous SDUs <ul style="list-style-type: none"> • Indicates whether SDUs detected as erroneous shall be delivered or not. • 0x00 - Subscribe • 0x01 - $5 \cdot 10^{-2}$ • 0x02 - $1 \cdot 10^{-2}$ • 0x03 - $5 \cdot 10^{-3}$ • 0x04 - $4 \cdot 10^{-3}$ • 0x05 - $1 \cdot 10^{-3}$ • 0x06 - $1 \cdot 10^{-4}$ • 0x07 - $1 \cdot 10^{-5}$ • 0x08 - $1 \cdot 10^{-6}$ • 0x09 - $1 \cdot 10^{-8}$
<i>transferDelay</i>	- Transfer delay (ms) <ul style="list-style-type: none"> • Indicates the targeted time between a request to transfer an SDU at one SAP to its delivery at the other SAP in milliseconds.
<i>trafficPriority</i>	- Transfer handling priority <ul style="list-style-type: none"> • Specifies the relative importance for handling of SDUs that belong to the UMTS bearer, compared to the SDUs of other bearers.

8.284.2 Field Documentation

- 8.284.2.1 `uint8_t LibPackUMTSQoS::deliveryErrSDU`
- 8.284.2.2 `uint32_t LibPackUMTSQoS::grntDownlinkBitrate`
- 8.284.2.3 `uint32_t LibPackUMTSQoS::grntUplinkBitrate`
- 8.284.2.4 `uint32_t LibPackUMTSQoS::maxDownlinkBitrate`
- 8.284.2.5 `uint32_t LibPackUMTSQoS::maxSDUSize`
- 8.284.2.6 `uint32_t LibPackUMTSQoS::maxUplinkBitrate`
- 8.284.2.7 `uint8_t LibPackUMTSQoS::qosDeliveryOrder`
- 8.284.2.8 `uint8_t LibPackUMTSQoS::resBerRatio`
- 8.284.2.9 `uint8_t LibPackUMTSQoS::sduErrorRatio`
- 8.284.2.10 `uint8_t LibPackUMTSQoS::trafficClass`
- 8.284.2.11 `uint32_t LibPackUMTSQoS::trafficPriority`
- 8.284.2.12 `uint32_t LibPackUMTSQoS::transferDelay`

8.285 LibPackUMTSReqQoSsigInd Struct Reference

Data Fields

- [LibPackUMTSQoS UMTSReqQoS](#)
- `uint8_t` [SigInd](#)

8.285.1 Detailed Description

structure contains UMTS requested QoS with Signaling Indication flag

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>UMTSReqQoS</i>	<ul style="list-style-type: none"> • Contains the UMTS Quality Of Service Information
<i>SigInd</i>	<ul style="list-style-type: none"> - Signaling Indication flag • TRUE - Signaling indication ON • FALSE - Signaling indication OFF

8.285.2 Field Documentation

8.285.2.1 `uint8_t LibPackUMTSReqQoSSigInd::SigInd`

8.285.2.2 `LibPackUMTSQoS LibPackUMTSReqQoSSigInd::UMTSReqQoS`

8.286 lineCtrlInfo Struct Reference

Data Fields

- [BYTE polarityIncluded](#)
- [BYTE toggleMode](#)
- [BYTE revPolarity](#)
- [BYTE pwrDenialTime](#)

8.286.1 Detailed Description

This structure contains Line Control Information

Parameters

<i>polarityIncluded</i>	<ul style="list-style-type: none"> • Included Polarity; Boolean Value
<i>toggleMode</i>	<ul style="list-style-type: none"> • Toggle mode; Boolean Value
<i>revPolarity</i>	<ul style="list-style-type: none"> • Reverse Polarity; Boolean Value
<i>pwrDenialTime</i>	<ul style="list-style-type: none"> • Power denial time; refer to [S1, Section 3.7.5.15 Line Control] for valid values

8.286.2 Field Documentation

8.286.2.1 BYTE lineCtrlInfo::polarityIncluded

8.286.2.2 BYTE lineCtrlInfo::pwrDenialTime

8.286.2.3 BYTE lineCtrlInfo::revPolarity

8.286.2.4 BYTE lineCtrlInfo::toggleMode

8.287 loc_BdsSV Struct Reference

Data Fields

- uint16_t [id](#)
- uint8_t [mask](#)

8.287.1 Detailed Description

This structure contains the BDS [SV](#) Info

Parameters

<i>id</i>	<ul style="list-style-type: none">• SV ID of the satellite whose data is to be deleted.<ul style="list-style-type: none">– Range for BDS: 201 to 237
<i>mask</i>	<ul style="list-style-type: none">• Indicates if the ephemeris or almanac for a satellite is to be deleted• Valid values:<ul style="list-style-type: none">– QMI_LOC_MASK_DELETE_EPHEMERIS (0x01) - Delete ephemeris for the satellite– QMI_LOC_MASK_DELETE_ALMANAC (0x02) - Delete almanac for the satellite

8.287.2 Field Documentation

8.287.2.1 uint16_t loc_BdsSV::id

8.287.2.2 uint8_t loc_BdsSV::mask

8.288 loc_BdsSVInfo Struct Reference

Data Fields

- uint8_t [len](#)
- [loc_BdsSV](#) * [pSV](#)

8.288.1 Detailed Description

This structure contains the number of sets of the BDS SVN Info

Parameters

<i>len</i>	<ul style="list-style-type: none"> Number of sets of the following elements: <ul style="list-style-type: none"> gnssSvId deleteSvInfoMask
<i>pSV</i>	<ul style="list-style-type: none"> Pointer to struct loc_BdsSV. See loc_BdsSV for more information

8.288.2 Field Documentation

8.288.2.1 `uint8_t loc_BdsSVInfo::len`

8.288.2.2 `loc_BdsSV* loc_BdsSVInfo::pSV`

8.289 loc_CellDb Struct Reference

Data Fields

- `uint32_t` [mask](#)

8.289.1 Detailed Description

This structure contains the cell database

Parameters

<i>mask</i>	<ul style="list-style-type: none"> Mask for the cell database assistance data that is to be deleted Valid values: <ul style="list-style-type: none"> 0x00000001 - DELETE_CELLDB_POS 0x00000002 - DELETE_CELLDB_LATEST_GPS_POS 0x00000004 - DELETE_CELLDB_OTA_POS 0x00000008 - DELETE_CELLDB_EXT_REF_POS 0x00000010 - DELETE_CELLDB_TIMETAG 0x00000020 - DELETE_CELLDB_CELLID 0x00000040 - DELETE_CELLDB_CACHED_CELLID 0x00000080 - DELETE_CELLDB_LAST_SRV_CELL 0x00000100 - DELETE_CELLDB_CUR_SRV_CELL 0x00000200 - DELETE_CELLDB_NEIGHBOR_INFO
-------------	--

8.289.2 Field Documentation

8.289.2.1 uint32_t loc_CellDb::mask

8.290 loc_ClkInfo Struct Reference

Data Fields

- uint32_t [mask](#)

8.290.1 Detailed Description

This structure contains the clock Info

Parameters

<i>mask</i>	<ul style="list-style-type: none"> • Mask for the clock information assistance data that is to be deleted • Valid bitmasks: <ul style="list-style-type: none"> – QMI_LOC_MASK_DELETE_CLOCK_INFO_TIME_EST (0x00000001) - Mask to delete time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_FREQ_EST (0x00000002) - Mask to delete frequency estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_WEEK_NUMBER (0x00000004) - Mask to delete week number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_RTC_TIME (0x00000008) - Mask to delete RTC time from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_TIME_TRANSFER (0x00000010) - Mask to delete time transfer from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GPSTIME_EST (0x00000020) - Mask to delete GPS time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLOTIME_EST (0x00000040) - Mask to delete GLONASS time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLODAY_NUMBER (0x00000080) - Mask to delete GLONASS day number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLO4YEAR_NUMBER (0x00000100) - Mask to delete GLONASS four year number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLO_RF_GRP_DELAY (0x00000200) - Mask to delete GLONASS RF GRP delay from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_DISABLE_TT (0x00000400) - Mask to delete disable TT from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GG_LEAPSEC (0x00000800) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GG_GGTB (0x00001000) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDSTIME_EST (0x00002000) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GB_GBTB (0x00004000) - Mask to delete Glonass-to-BDS time bias-related information from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BG_BGTB (0x00008000) - Mask to delete BDS-to-GLONASS time bias-related information from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDSWEEK_NUMBER (0x00010000) - Mask to delete the BDS week number from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDS_RF_GRP_DELAY (0x00020000) - Mask to delete the BDS RF GRP delay from the clock information
Generated by Doxygen	

8.290.2 Field Documentation

8.290.2.1 `uint32_t loc_ClkInfo::mask`

8.291 `loc_GnssData` Struct Reference

Data Fields

- `uint64_t` [mask](#)

8.291.1 Detailed Description

This structure contains the GNSS data

Parameters

<i>mask</i>	<ul style="list-style-type: none"> • Mask for the GNSS data that is to be deleted • Valid values: <ul style="list-style-type: none"> – QMI_LOC_MASK_DELETE_GPS_SVDIR (0x00000001) - Mask to delete GPS SVDIR – QMI_LOC_MASK_DELETE_GPS_SVSTEER (0x00000002) - Mask to delete GPS SVSTEER – QMI_LOC_MASK_DELETE_GPS_TIME (0x00000004) - Mask to delete GPS time – QMI_LOC_MASK_DELETE_GPS_ALM_CORR (0x00000008) - Mask to delete almanac correlation – QMI_LOC_MASK_DELETE_GLO_SVDIR (0x00000010) - Mask to delete GLONASS SVDIR – QMI_LOC_MASK_DELETE_GLO_SVSTEER (0x00000020) - Mask to delete GLONASS SVSTEER – QMI_LOC_MASK_DELETE_GLO_TIME (0x00000040) - Mask to delete GLONASS time – QMI_LOC_MASK_DELETE_GLO_ALM_CORR (0x00000080) - Mask to delete GLONASS almanac correlation – QMI_LOC_MASK_DELETE_SBAS_SVDIR (0x00000100) - Mask to delete SBAS SVDIR – QMI_LOC_MASK_DELETE_SBAS_SVSTEER (0x00000200) - Mask to delete SBAS SVSTEER – QMI_LOC_MASK_DELETE_POSITION (0x00000400) - Mask to delete position estimate – QMI_LOC_MASK_DELETE_TIME (0x00000800) - Mask to delete time estimate – QMI_LOC_MASK_DELETE_IONO (0x00001000) - Mask to delete IONO – QMI_LOC_MASK_DELETE_UTC (0x00002000) - Mask to delete UTC estimate – QMI_LOC_MASK_DELETE_HEALTH (0x00004000) - Mask to delete SV health record – QMI_LOC_MASK_DELETE_SADATA (0x00008000) - Mask to delete SADATA – QMI_LOC_MASK_DELETE_RTI (0x00010000) - Mask to delete RTI – QMI_LOC_MASK_DELETE_SV_NO_EXIST (0x00020000) - Mask to delete SV_NO_EXIST – QMI_LOC_MASK_DELETE_FREQ_BIAS_EST (0x00040000) - Mask to delete frequency bias estimate – QMI_LOC_MASK_DELETE_BDS_SVDIR (0x00080000) - Mask to delete BDS SVDIR – QMI_LOC_MASK_DELETE_BDS_SVSTEER (0x00100000) - Mask to delete BDS SVSTEER – QMI_LOC_MASK_DELETE_BDS_TIME (0x00200000) - Mask to delete BDS time – QMI_LOC_MASK_DELETE_BDS_ALM_CORR (0x00400000) - Mask to delete BDS almanac correlation – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_GPS (0x00800000) - Mask to delete GNSS SV blacklist GPS – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_GLO (0x01000000) - Mask to delete GNSS SV blacklist GLO – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_BDS (0x02000000) - Mask to delete GNSS SV blacklist BDS
-------------	---

8.291.2 Field Documentation

8.291.2.1 `uint64_t loc_GnssData::mask`

8.292 loc_gpsTime Struct Reference

Data Fields

- uint16_t [gpsWeek](#)
- uint32_t [gpsTimeOfWeekMs](#)

8.292.1 Detailed Description

This structure contains GPS Time info.

Parameters

<i>gpsWeek</i>	<ul style="list-style-type: none"> • Current GPS week as calculated from midnight, Jan. 6, 1980. • Units - Weeks
<i>gpsTimeOfWeekMs</i>	<ul style="list-style-type: none"> • Amount of time into the current GPS week. • Units - Milliseconds

8.292.2 Field Documentation

8.292.2.1 uint32_t loc_gpsTime::gpsTimeOfWeekMs

8.292.2.2 uint16_t loc_gpsTime::gpsWeek

8.293 loc_LocApplicationInfo Struct Reference

Data Fields

- uint8_t [appProviderLength](#)
- uint8_t * [pAppProvider](#)
- uint8_t [appNameLength](#)
- uint8_t * [pAppName](#)
- uint8_t [appVersionValid](#)
- uint8_t [appVersionLength](#)
- uint8_t * [pAppVersion](#)

8.293.1 Detailed Description

This structure contains the Application Information

Parameters

<i>appProvider</i> ↔ <i>Length</i>	<ul style="list-style-type: none"> Length of the Application Provider
<i>pAppProvider</i>	<ul style="list-style-type: none"> Application Provider Depends upon the Length of application Provider
<i>appNameLength</i>	<ul style="list-style-type: none"> Length of Application Name
<i>pAppName</i>	<ul style="list-style-type: none"> Application Name Depends upon the Length of application Name
<i>appVersionValid</i>	<ul style="list-style-type: none"> Specifies whether the application version string contains a valid value 0x00 (FALSE) Application version string is invalid 0x01 (TRUE) Application version string is valid
<i>appVersion</i> ↔ <i>Length</i>	<ul style="list-style-type: none"> Length of Application Version
<i>pAppVersion</i>	<ul style="list-style-type: none"> Application Version Depends upon the Length of application Version

8.293.2 Field Documentation

8.293.2.1 uint8_t loc_LocApplicationInfo::appNameLength

8.293.2.2 uint8_t loc_LocApplicationInfo::appProviderLength

8.293.2.3 uint8_t loc_LocApplicationInfo::appVersionLength

8.293.2.4 uint8_t loc_LocApplicationInfo::appVersionValid

8.293.2.5 uint8_t* loc_LocApplicationInfo::pAppName

8.293.2.6 uint8_t* loc_LocApplicationInfo::pAppProvider

8.293.2.7 uint8_t* loc_LocApplicationInfo::pAppVersion

8.294 loc_precisionDilution Struct Reference

Data Fields

- uint32_t [PDOP](#)
- uint32_t [HDOP](#)
- uint32_t [VDOP](#)

8.294.1 Detailed Description

This structure contains Dilution of precision associated with this position.

Parameters

<i>PDOP</i>	<ul style="list-style-type: none"> Position dilution of precision. Range - 1 (highest accuracy) to 50 (lowest accuracy) PDOP = square root of (Square of HDOP + Square of VDOP2)
<i>HDOP</i>	<ul style="list-style-type: none"> Horizontal dilution of precision. Range - 1 (highest accuracy) to 50 (lowest accuracy)
<i>VDOP</i>	<ul style="list-style-type: none"> Vertical dilution of precision. Range- 1 (highest accuracy) to 50 (lowest accuracy)

8.294.2 Field Documentation

8.294.2.1 `uint32_t loc_precisionDilution::HDOP`

8.294.2.2 `uint32_t loc_precisionDilution::PDOP`

8.294.2.3 `uint32_t loc_precisionDilution::VDOP`

8.295 `loc_satelliteInfo` Struct Reference

Data Fields

- `uint8_t svListLen`
- `uint32_t validMask`
- `uint32_t system`
- `uint16_t gnssSvId`
- `uint8_t healthStatus`
- `uint32_t svStatus`
- `uint8_t svInfoMask`
- `float elevation`
- `float azimuth`
- `float snr`

8.295.1 Detailed Description

Contain fields in struct `loc_satelliteInfo`

Parameters

<i>svListLen</i>	<ul style="list-style-type: none"> number of sets of the following elements: <ul style="list-style-type: none"> validMask system gnssSvid healthStatus svStatus svInfoMask elevation azimuth snr
<i>validMask</i>	<ul style="list-style-type: none"> Bitmask indicating which of the fields in this TLV are valid. Valid bitmasks: <ul style="list-style-type: none"> 0x00000001 - VALID_SYSTEM 0x00000002 - VALID_GNSS_SVID 0x00000004 - VALID_HEALTH_STATUS 0x00000008 - VALID_PROCESS_STATUS 0x00000010 - VALID_SVINFO_MASK 0x00000020 - VALID_ELEVATION 0x00000040 - VALID_AZIMUTH 0x00000080 - VALID_SNR
<i>system</i>	<ul style="list-style-type: none"> Indicates to which constellation this SV belongs. Valid values: <ul style="list-style-type: none"> eQMI_LOC_SV_SYSTEM_GPS (1) - GPS satellite eQMI_LOC_SV_SYSTEM_GALILEO (2) - GALILEO satellite eQMI_LOC_SV_SYSTEM_SBAS (3) - SBAS satellite eQMI_LOC_SV_SYSTEM_COMPASS (4) - COMPASS satellite eQMI_LOC_SV_SYSTEM_GLONASS (5) - GLONASS satellite eQMI_LOC_SV_SYSTEM_BDS (6) - BDS satellite
<i>gnssSvid</i>	<ul style="list-style-type: none"> GNSS SV ID. The GPS and GLONASS SVs can be disambiguated using the system field. Range: <ul style="list-style-type: none"> FOR GPS: 1 to 32 FOR GLONASS: 1 to 32 FOR SBAS: 120 to 151 for BDS: 201 to 237
<i>healthStatus</i>	<ul style="list-style-type: none"> health status. Range: 0 - 1 <ul style="list-style-type: none"> 0 - unhealthy 1 - healthy
<i>svStatus</i>	<ul style="list-style-type: none"> SV process status. Valid values: <ul style="list-style-type: none"> eQMI_LOC_SV_STATUS_IDLE (1) - SV is not being actively processed eQMI_LOC_SV_STATUS_SEARCH (2) - The system is searching for this SV eQMI_LOC_SV_STATUS_TRACK (3) - SV is being tracked

<i>svInfoMask</i>	<ul style="list-style-type: none"> Indicates whether almanac and ephemeris information is available. Valid bitmasks: <ul style="list-style-type: none"> 0x01 - SVINFO_HAS_EPHEMERIS 0x02 - SVINFO_HAS_ALMANAC
<i>elevation</i>	<ul style="list-style-type: none"> SV elevation angle. <ul style="list-style-type: none"> Units: Degrees Range: 0 to 90
<i>azimuth</i>	<ul style="list-style-type: none"> SV azimuth angle. <ul style="list-style-type: none"> Units: Degrees Range: 0 to 360
<i>snr</i>	<ul style="list-style-type: none"> SV signal-to-noise ratio <ul style="list-style-type: none"> Units: dB-Hz

8.295.2 Field Documentation

8.295.2.1 float loc_satelliteInfo::azimuth

8.295.2.2 float loc_satelliteInfo::elevation

8.295.2.3 uint16_t loc_satelliteInfo::gnssSvId

8.295.2.4 uint8_t loc_satelliteInfo::healthStatus

8.295.2.5 float loc_satelliteInfo::snr

8.295.2.6 uint8_t loc_satelliteInfo::svInfoMask

8.295.2.7 uint8_t loc_satelliteInfo::svListLen

8.295.2.8 uint32_t loc_satelliteInfo::svStatus

8.295.2.9 uint32_t loc_satelliteInfo::system

8.295.2.10 uint32_t loc_satelliteInfo::validMask

8.296 loc_sensorDataUsage Struct Reference

Data Fields

- uint32_t [usageMask](#)
- uint32_t [aidingIndicatorMask](#)

8.296.1 Detailed Description

This structure contains Sensor Data Usage info.

Parameters

<i>usageMask</i>	<ul style="list-style-type: none">• Specifies which sensors were used in calculating the position in the position report.
------------------	---

- Value
 - 0x00000001 - Accelerometer used
 - 0x00000002 - Gyroscope used

Parameters

<i>aidingIndicator↔ Mask</i>	
----------------------------------	--

- Specifies which results were aided by sensors.
- Value
 - 0x00000001 - AIDED_HEADING
 - 0x00000002 - AIDED_SPEED
 - 0x00000004 - AIDED_POSITION
 - 0x00000008 - AIDED_VELOCITY

8.296.2 Field Documentation

8.296.2.1 `uint32_t loc_sensorDataUsage::aidingIndicatorMask`

8.296.2.2 `uint32_t loc_sensorDataUsage::usageMask`

8.297 loc_SV Struct Reference

Data Fields

- `uint16_t id`
- `uint32_t system`
- `uint8_t mask`

8.297.1 Detailed Description

This structure contains the Delete LOC [SV](#) Info

Parameters

<i>id</i>	<ul style="list-style-type: none"> • LOC SV ID of the satellite whose data is to be deleted • Range: <ul style="list-style-type: none"> – For GPS: 1 to 32 – For SBAS: 33 to 64 – For GLONASS: 65 to 96
<i>system</i>	<ul style="list-style-type: none"> • Indicates to which constellation this loc_SV belongs • Valid values: <ul style="list-style-type: none"> – eQMI_LOC_SV_SYSTEM_GPS (1) - GPS satellite – eQMI_LOC_SV_SYSTEM_GALILEO (2) - GALILEO satellite – eQMI_LOC_SV_SYSTEM_SBAS (3) - SBAS satellite – eQMI_LOC_SV_SYSTEM_COMPASS (4) - COMPASS satellite – eQMI_LOC_SV_SYSTEM_GLONASS (5) - GLONASS satellite – eQMI_LOC_SV_SYSTEM_BDS (6) - BDS satellite
<i>mask</i>	<ul style="list-style-type: none"> • Indicates if the ephemeris or almanac for a satellite is to be deleted • Valid values: <ul style="list-style-type: none"> – 0x01 - DELETE_EPHEMERIS – 0x02 - DELETE_ALMANAC

8.297.2 Field Documentation

8.297.2.1 `uint16_t loc_SV::id`8.297.2.2 `uint8_t loc_SV::mask`8.297.2.3 `uint32_t loc_SV::system`8.298 `loc_SVInfo` Struct Reference

Data Fields

- `uint8_t len`
- `loc_SV * pSV`

8.298.1 Detailed Description

This structure contains the elements of Delete LOC [SV](#) Info

Parameters

<i>len</i>	<ul style="list-style-type: none"> • Number of sets of the following elements in struct loc_SV: <ul style="list-style-type: none"> – <code>gnssSvId</code> – <code>system</code> – <code>deleteSvInfoMask</code>
<i>pSV</i>	

8.298.2 Field Documentation

8.298.2.1 `uint8_t loc_SVInfo::len`

8.298.2.2 `loc_SV* loc_SVInfo::pSV`

8.299 loc_svUsedforFix Struct Reference

Data Fields

- `uint8_t gnssSvUsedList_len`
- `uint16_t gnssSvUsedList [255]`

8.299.1 Detailed Description

This structure contains SVs Used to Calculate the Fix.

Parameters

<i>gnssSvUsedList_len</i>	<ul style="list-style-type: none"> • Number of sets of gnssSvUsedList
<i>pGnssSvUsedList</i>	<ul style="list-style-type: none"> • Entry in the list contains the SV ID of a satellite used for calculating this position report. • Following information is associated with each SV ID: <ul style="list-style-type: none"> – GPS - 1 to 32 – SBAS - 33 to 64 – GLONASS - 65 to 96 – QZSS - 193 to 197 – BDS - 201 to 237

8.299.2 Field Documentation

8.299.2.1 `uint16_t loc_svUsedforFix::gnssSvUsedList[255]`

8.299.2.2 `uint8_t loc_svUsedforFix::gnssSvUsedList_len`

8.300 LocApplicationInfo Struct Reference

Data Fields

- `BYTE appProviderLength`
- `CHAR * pAppProvider`
- `BYTE appNameLength`
- `CHAR * pAppName`
- `BYTE appVersionValid`
- `CHAR appVersionLength`
- `CHAR * pAppVersion`

8.300.1 Detailed Description

This structure contains the Application Information

Parameters

<i>appProvider</i> ↔ <i>Length</i>	<ul style="list-style-type: none"> Length of the Application Provider
<i>pAppProvider</i>	<ul style="list-style-type: none"> Application Provider Depends upon the Length of application Provider
<i>appNameLength</i>	<ul style="list-style-type: none"> Length of Application Name
<i>pAppName</i>	<ul style="list-style-type: none"> Application Name Depends upon the Length of application Name
<i>appVersionValid</i>	<ul style="list-style-type: none"> Specifies whether the application version string contains a valid value 0x00 (FALSE) – Application version string is invalid 0x01 (TRUE) – Application version string is valid
<i>appVersion</i> ↔ <i>Length</i>	<ul style="list-style-type: none"> Length of Application Version
<i>pAppVersion</i>	<ul style="list-style-type: none"> Application Version Depends upon the Length of application Version

8.300.2 Field Documentation

8.300.2.1 **BYTE** LocApplicationInfo::appNameLength

8.300.2.2 **BYTE** LocApplicationInfo::appProviderLength

8.300.2.3 **CHAR** LocApplicationInfo::appVersionLength

8.300.2.4 **BYTE** LocApplicationInfo::appVersionValid

8.300.2.5 **CHAR*** LocApplicationInfo::pAppName

8.300.2.6 **CHAR*** LocApplicationInfo::pAppProvider

8.300.2.7 **CHAR*** LocApplicationInfo::pAppVersion

8.301 LocDelAssDataReq Struct Reference

Data Fields

- [SVInfo](#) * [pSVInfo](#)
- [GnssData](#) * [pGnssData](#)
- [CellDb](#) * [pCellDb](#)
- [ClkInfo](#) * [pClkInfo](#)
- [BdsSVInfo](#) * [pBdsSVInfo](#)

8.301.1 Detailed Description

This structure contains LOC delete assist data request

Parameters

<i>pSVInfo</i> [IN]	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct SVInfo. See SVInfo for more information
<i>pGnssData</i> [IN]	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct GnssData. See GnssData for more information
<i>pCellDb</i> [IN]	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct CellDb. See CellDb for more information
<i>pClkInfo</i> [IN]	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct ClkInfo. See ClkInfo for more information
<i>pBdsSVInfo</i> [IN]	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct BdsSVInfo. See BdsSVInfo for more information

8.301.2 Field Documentation

8.301.2.1 **BdsSVInfo*** LocDelAssDataReq::pBdsSVInfo

8.301.2.2 **CellDb*** LocDelAssDataReq::pCellDb

8.301.2.3 **ClkInfo*** LocDelAssDataReq::pClkInfo

8.301.2.4 **GnssData*** LocDelAssDataReq::pGnssData

8.301.2.5 **SVInfo*** LocDelAssDataReq::pSVInfo

8.302 LOEventRegisterReqResp Struct Reference

Data Fields

- [ULONGLONG eventRegister](#)

8.302.1 Detailed Description

This structure contains the Parameter for RegisterEvents

Parameters

<i>pEventRegMask</i>	<ul style="list-style-type: none"> • Specifies the events that the control point is interested in receiving. -Values <ul style="list-style-type: none"> – 0x00000001 - to receive position report event indications – 0x00000002 - to receive satellite report event indications. These reports are sent at a 1 Hz rate. – 0x00000004 - to receive NMEA reports for position and satellites in view. The report is at a 1 Hz rate. – 0x00000008 - to receive NI Notify/Verify request event indications – 0x00000010 - to receive time injection request event indications. – 0x00000020 - to receive predicted orbits request event indications. – 0x00000040 - to receive position injection request event indications. – 0x00000080 - to receive engine state report event indications. – 0x00000100 - to receive fix session status report event indications. – 0x00000200 - to receive Wi-Fi position request event indications. – 0x00000400 - to receive notifications from the location engine indicating its readiness to accept data from the sensors (accelerometer, gyroscope, etc.). – 0x00000800 - to receive time sync requests from the GPS engine. Time sync enables the GPS engine to synchronize its clock with the sensor processor's clock. – 0x00001000 - to receive Stationary Position Indicator (SPI) streaming report indications. – 0x00002000 - to receive location server requests. These requests are generated when the service wishes to establish a connection with a location server. – 0x00004000 - to receive notifications related to network-initiated Geofences. These events notify the client when a network-initiated Geofence is added, deleted, or edited. – 0x00008000 - to receive Geofence alerts. These alerts are generated to inform the client of the changes that may affect a Geofence, e.g., if GPS is turned off or if the network is unavailable. – 0x00010000 - to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach report is for a single Geofence. – 0x00020000 - to register for pedometer control requests from the location engine. The location engine sends this event to control the injection of pedometer reports. – 0x00040000 - to register for motion data control requests from the location engine. The location engine sends this event to control the injection of motion data. – 0x00080000 - to receive notification when a batch is full. The location engine sends this event to notify of Batch Full for ongoing batching session. – 0x00100000 - to receive position report indications along with an ongoing batching session. The location engine sends this event to notify the batched position report while a batching session is ongoing. – 0x00200000 - to receive Wi-Fi Access Point (AP) data inject request event indications. – 0x00400000 - to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach notification is for multiple Geofences. Breaches from multiple Geofences are all batched and sent in the same notification. – 0x00800000 - to receive notifications from the location engine indicating its readiness to accept vehicle data (vehicle accelerometer, vehicle angular rate, vehicle odometry, etc.). – 0x01000000 - to receive system clock and satellite measurement report events (system clock, SV time, Doppler, etc.). – 0x02000000 - to receive satellite position reports as polynomials. Reports are generated only for the GNSS satellite constellations that are enabled using QMI_LOC_SET_GNSS←_CONSTELL_REPORT_CONFIG.
----------------------	---

Note

Multiple events can be registered by OR the individual masks and sending them in this TLV. All unused bits in this mask must be set to 0.

8.302.2 Field Documentation**8.302.2.1 ULONGLONG LOCEventRegisterReqResp::eventRegister****8.303 LOCExtPowerStateReqResp Struct Reference****Data Fields**

- [ULONG extPowerState](#)

8.303.1 Detailed Description

This structure contains the Parameter External Power Source State.

Parameters

<p><i>pLOCEventRegisterReqResp</i></p>	<ul style="list-style-type: none"> • Specifies the Power state; injected by the control point. • Values <ul style="list-style-type: none"> – 0 - Device is not connected to an external power source – 1 - Device is connected to an external power source – 2 - Unknown external power state
--	---

8.303.2 Field Documentation**8.303.2.1 ULONG LOCExtPowerStateReqResp::extPowerState****8.304 LocInjectPositionReq Struct Reference****Data Fields**

- [ULONGLONG * pLatitude](#)
- [ULONGLONG * pLongitude](#)
- [ULONG * pHorUncCircular](#)
- [BYTE * pHorConfidence](#)
- [ULONG * pHorReliability](#)
- [ULONG * pAltitudeWrtEllipsoid](#)
- [ULONG * pAltitudeWrtMeanSeaLevel](#)
- [ULONG * pVertUnc](#)
- [BYTE * pVertConfidence](#)
- [ULONG * pVertReliability](#)

- altitudeSrcInfo * pAltitudeSrcInfo
- ULONGLONG * pTimestampUtc
- ULONG * pTimestampAge
- ULONG * pPositionSrc
- ULONG * pRawHorUncCircular
- BYTE * pRawHorConfidence

8.304.1 Detailed Description

This structure contains LOC Inject Position parameters

Parameters

<i>pLatitude</i>	<ul style="list-style-type: none"> • Optional parameter • Type - Floating point • Units - Degrees • Range - -90.0 to 90.0 • Positive values indicate northern latitude • Negative values indicate southern latitude • Note - This field must be specified together with pLongitude and pHorUncCircular.
<i>pLongitude</i>	<ul style="list-style-type: none"> • Optional parameter • Type - Floating point • Units - Degrees • Range - -180.0 to 180.0 • Positive values indicate eastern latitude • Negative values indicate western latitude • Note - This field must be specified together with pLatitude and pHorUncCircular.
<i>pHorUncCircular</i>	<ul style="list-style-type: none"> • Optional parameter • Horizontal position uncertainty. • Units - Meters • Note - This field must be specified together with pLatitude and pLongitude.
<i>pHorConfidence</i>	<ul style="list-style-type: none"> • Optional parameter • Horizontal confidence. • Units - Percent • Values <ul style="list-style-type: none"> – Valid Values - 1 to 99 – Invalid Values - 0, 101 to 255 – If 100 is received, reinterpret to 99 • Note - This field must be specified together with horizontal uncertainty. If not specified when pHorUncCircular is set, the default value is 50.
<i>pHorReliability</i>	<ul style="list-style-type: none"> • Optional parameter

- Values
 - 0 - Location reliability is not set.
 - 1 - Location reliability is very low; use it at your own risk
 - 2 - Location reliability is low; little or no cross-checking is possible.
 - 3 - Location reliability is medium; limited cross-check passed
 - 4 - Location reliability is high; strong cross-check passed

Parameters

<i>pAltitudeWrtEllipsoid</i>	<ul style="list-style-type: none"> • Optional parameter • Altitude With Respect to Ellipsoid. • Units - Meters • Values <ul style="list-style-type: none"> – Positive - height – Negative = depth
<i>pAltitudeWrtMeanSeaLevel</i>	<ul style="list-style-type: none"> • Optional parameter • Altitude With Respect to Sea Level. • Units - Meters
<i>pVertUnc</i>	<ul style="list-style-type: none"> • Optional parameter • Vertical uncertainty. • Units - Meters • Note - This is mandatory if either <i>pAltitudeWrtEllipsoid</i> or <i>pAltitudeWrtMeanSeaLevel</i> is specified.
<i>pVertConfidence</i>	<ul style="list-style-type: none"> • Optional parameter • Vertical confidence. • Units - Percentage • Values <ul style="list-style-type: none"> – Valid Values - 0 to 99 – Invalid Values - 0, 100-256 – If 100 is received, reinterpret to 99 • Note - This field must be specified together with the vertical uncertainty. If not specified, the default value will be 50.
<i>pVertReliability</i>	<ul style="list-style-type: none"> • Optional parameter

- Values
 - 0 - Location reliability is not set.
 - 1 - Location reliability is very low; use it at your own risk.
 - 2 - Location reliability is low; little or no cross-checking is possible
 - 3 - Location reliability is medium; limited cross-check passed
 - 4 - Location reliability is high; strong cross-check passed

Parameters

<i>pAltitudeSrcInfo</i>	<ul style="list-style-type: none"> Optional parameter
-------------------------	--

- Pointer to struct [altitudeSrcInfo](#). See [altitudeSrcInfo](#) for more information

Parameters

<i>pTimestampUtc</i>	<ul style="list-style-type: none"> Optional parameter UTC timestamp Units - Milliseconds since Jan. 1, 1970
<i>pTimestampAge</i>	<ul style="list-style-type: none"> Optional parameter Position age, which is an estimate of how long ago this fix was made. Units - Milliseconds
<i>pPositionSrc</i>	<ul style="list-style-type: none"> Optional parameter Source from which this position was obtained Valid values <ul style="list-style-type: none"> 0 - Position source is GNSS 1 - Position source is Cell ID 2 - Position source is Enhanced Cell ID 3 - Position source is Wi-Fi 4 - Position source is Terrestrial 5 - Position source is GNSS Terrestrial Hybrid 6 - Other sources Note - If altitude is specified and the altitude source is not specified, the engine assumes that the altitude was obtained using the specified position source. <ul style="list-style-type: none"> If both altitude and altitude source are specified, the engine assumes that only latitude and longitude were obtained using the specified position source.
<i>pRawHorUnc</i> ↔ <i>Circular</i>	<ul style="list-style-type: none"> Optional parameter Horizontal position uncertainty (circular) without any optimization. Units - Meters
<i>pRawHor</i> ↔ <i>Confidence</i>	<ul style="list-style-type: none"> Optional parameter Horizontal confidence associated with raw horizontal uncertainty Units: Percent Values <ul style="list-style-type: none"> Valid values - 1 to 99 Invalid values - 0, 101 to 255 If 100 is received, reinterpret to 99 Note - This field must be specified together with raw horizontal uncertainty. If not specified when <i>rawHorUncCircular</i> is set, the default value is 50.

8.304.2 Field Documentation

- 8.304.2.1 **altitudeSrcInfo*** **LocInjectPositionReq::pAltitudeSrcInfo**
- 8.304.2.2 **ULONG*** **LocInjectPositionReq::pAltitudeWrtEllipsoid**
- 8.304.2.3 **ULONG*** **LocInjectPositionReq::pAltitudeWrtMeanSeaLevel**
- 8.304.2.4 **BYTE*** **LocInjectPositionReq::pHorConfidence**
- 8.304.2.5 **ULONG*** **LocInjectPositionReq::pHorReliability**
- 8.304.2.6 **ULONG*** **LocInjectPositionReq::pHorUncCircular**
- 8.304.2.7 **ULONGLONG*** **LocInjectPositionReq::pLatitude**
- 8.304.2.8 **ULONGLONG*** **LocInjectPositionReq::pLongitude**
- 8.304.2.9 **ULONG*** **LocInjectPositionReq::pPositionSrc**
- 8.304.2.10 **BYTE*** **LocInjectPositionReq::pRawHorConfidence**
- 8.304.2.11 **ULONG*** **LocInjectPositionReq::pRawHorUncCircular**
- 8.304.2.12 **ULONG*** **LocInjectPositionReq::pTimestampAge**
- 8.304.2.13 **ULONGLONG*** **LocInjectPositionReq::pTimestampUtc**
- 8.304.2.14 **BYTE*** **LocInjectPositionReq::pVertConfidence**
- 8.304.2.15 **ULONG*** **LocInjectPositionReq::pVertReliability**
- 8.304.2.16 **ULONG*** **LocInjectPositionReq::pVertUnc**

8.305 LocInjectSensorDataReq Struct Reference

Data Fields

- [ULONG](#) * [pOpaqueIdentifier](#)
- [sensorData](#) * [pAcceleroData](#)
- [sensorData](#) * [pGyroData](#)
- [ULONG](#) * [pAcceleroTimeSrc](#)
- [ULONG](#) * [pGyroTimeSrc](#)
- [temperatureData](#) * [pAcceleroTempData](#)
- [temperatureData](#) * [pGyroTempData](#)

8.305.1 Detailed Description

This structure contains parameters to inject sensor data into the GNSS location engine

Parameters

<i>pOpaque</i> ↔ <i>Identifier</i>	<ul style="list-style-type: none"> • Opaque Identifier (Optional parameter) • An opaque identifier that is sent in by the client that will be echoed in the indication so the client can relate the indication to the request.
<i>pAcceleroData</i>	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct sensorData. See sensorData for more information
<i>pGyroData</i>	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct sensorData. See sensorData for more information
<i>pAccelero</i> ↔ <i>TimeSrc</i>	<ul style="list-style-type: none"> • 3-Axis Accelerometer Data Time Source (Optional parameter) • The location service uses this field to identify the time reference used in the accelerometer data time stamps. • If not specified, the location service assumes that the time source for the accelerometer data is unknown. • Valid values <ul style="list-style-type: none"> – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine
<i>pGyroTimeSrc</i>	<ul style="list-style-type: none"> • 3-Axis Gyroscope Data Time Source (Optional) • The location service uses this field to identify the time reference used in the gyroscope data time stamps. • If not specified, the location service assumes that the time source for the gyroscope data is unknown. • Valid values <ul style="list-style-type: none"> – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine
<i>pAccelero</i> ↔ <i>TempData</i>	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct tempratureData. See tempratureData for more information
<i>pGyroTempData</i>	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct tempratureData. See tempratureData for more information

8.305.2 Field Documentation

8.305.2.1 **sensorData*** LocInjectSensorDataReq::pAcceleroData

8.305.2.2 **tempratureData*** LocInjectSensorDataReq::pAcceleroTempData

8.305.2.3 **ULONG*** LocInjectSensorDataReq::pAcceleroTimeSrc

8.305.2.4 **sensorData*** LocInjectSensorDataReq::pGyroData

8.305.2.5 **tempratureData*** LocInjectSensorDataReq::pGyroTempData

8.305.2.6 **ULONG*** LocInjectSensorDataReq::pGyroTimeSrc

8.305.2.7 **ULONG*** LocInjectSensorDataReq::pOpaqueldentifier

8.306 LocSetCradleMountReq Struct Reference

Data Fields

- [ULONG](#) state
- [BYTE](#) * pConfidence

8.306.1 Detailed Description

This structure contains parameters to set current cradle mount configuration

Parameters

<i>state</i>	<ul style="list-style-type: none"> • Cradle Mount State • Valid values: <ul style="list-style-type: none"> – 0 - Device is mounted on the cradle – 1 - Device is not mounted on the cradle – 2 - Unknown cradle mount state
<i>pConfidence</i>	<ul style="list-style-type: none"> • Cradle Mount Confidence (Optional) • Confidence in the Cradle Mount state expressed as a percentage. • Range - 0 to 100

8.306.2 Field Documentation

8.306.2.1 **BYTE*** LocSetCradleMountReq::pConfidence

8.306.2.2 **ULONG** LocSetCradleMountReq::state

8.307 LOCStartReq Struct Reference

Data Fields

- [BYTE](#) SessionId
- [ULONG](#) * pRecurrenceType

- [ULONG](#) * [pHorizontalAccuracyLvl](#)
- [ULONG](#) * [pIntermediateReportState](#)
- [ULONG](#) * [pMinIntervalTime](#)
- [struct LocApplicationInfo](#) * [pApplicationInfo](#)
- [ULONG](#) * [pConfigAltitudeAssumed](#)

8.307.1 Detailed Description

This structure contains the LOC Start Request

Parameters

<i>SessionId</i> [IN]	<ul style="list-style-type: none"> • ID of the session as identified by the control point. • Range: 0 to 255
<i>pRecurrence</i> <i>Type</i> [IN]	<ul style="list-style-type: none"> • Optional Parameter • Specifies the type of session in which the control point is interested. • Defaults to SINGLE. -Values <ul style="list-style-type: none"> – 1 - Request periodic position fixes – 2 - Request a single position fix
<i>pHorizontal</i> <i>AccuracyLvl</i> [IN]	<ul style="list-style-type: none"> • Optional Parameter • Specifies the horizontal accuracy level required by the control point. • Defaults to LOW • Values <ul style="list-style-type: none"> – 1 - Low accuracy – 2 - Medium accuracy – 3 - High accuracy
<i>pIntermediate</i> <i>ReportState</i> [IN]	<ul style="list-style-type: none"> • Optional Parameter • Specifies if the control point is interested in receiving intermediate reports. • ON by default. • Values <ul style="list-style-type: none"> – 1 - Intermediate reports are turned on – 2 - Intermediate reports are turned off
<i>pMinInterval</i> <i>Time</i> [IN]	<ul style="list-style-type: none"> • Optional Parameter • Minimum time interval, specified by the control point, that must elapse between position reports. • Units - Milliseconds • Default - 1000 ms
<i>LocApplication</i> <i>Info</i> [IN]	<ul style="list-style-type: none"> • Optional Parameter • LOC Application Parameters • See LocApplicationInfo for more information

<p><i>pConfig</i>↔</p> <p><i>Altitude</i>↔</p> <p><i>Assumed</i>[IN]</p>	<ul style="list-style-type: none"> • Optional Parameter • Configuration for Altitude Assumed Info in GNSS SV Info Event • Defaults to ENABLED. • Values <ul style="list-style-type: none"> – 1 - Enable Altitude Assumed information in GNSS SV Info Event – 2 - Disable Altitude Assumed information in GNSS SV Info Event
--	--

8.307.2 Field Documentation

8.307.2.1 **struct** `LocApplicationInfo*` `LOCStartReq::pApplicationInfo`

8.307.2.2 **ULONG*** `LOCStartReq::pConfigAltitudeAssumed`

8.307.2.3 **ULONG*** `LOCStartReq::pHorizontalAccuracyLvl`

8.307.2.4 **ULONG*** `LOCStartReq::pIntermediateReportState`

8.307.2.5 **ULONG*** `LOCStartReq::pMinIntervalTime`

8.307.2.6 **ULONG*** `LOCStartReq::pRecurrenceType`

8.307.2.7 **BYTE** `LOCStartReq::SessionId`

8.308 LOCStopReq Struct Reference

Data Fields

- [BYTE](#) `sessionId`

8.308.1 Detailed Description

This structure contains the LOC Stop Request

Parameters

<i>sessionId</i>	<ul style="list-style-type: none"> • ID of the session as identified by the control point. • Range: 0 to 255
------------------	--

8.308.2 Field Documentation

8.308.2.1 BYTE LOCStopReq::sessionId

8.309 LteCQIParm Struct Reference

Data Fields

- BYTE ValidityCW0
- BYTE CQIValueCW0
- BYTE ValidityCW1
- BYTE CQIValueCW1

8.309.1 Detailed Description

This structure contains information about the SLQSSwiGetLteCQI response parameters.

Parameters

<i>pValidityCW0</i> [↔ OUT]	<ul style="list-style-type: none">• Values<ul style="list-style-type: none">– 0- Invalid.– 1- Valid.
<i>pCQIValueC</i> ↔ <i>W0</i> [OUT]	<ul style="list-style-type: none">• Values<ul style="list-style-type: none">– Range 0~15
<i>pValidityCW1</i> [↔ OUT]	<ul style="list-style-type: none">• Values<ul style="list-style-type: none">– 0- Invalid.– 1- Valid.
<i>pCQIValueC</i> ↔ <i>W1</i> [OUT]	<ul style="list-style-type: none">• Values<ul style="list-style-type: none">– Range 0~15

8.309.2 Field Documentation

8.309.2.1 BYTE LteCQIParm::CQIValueCW0

8.309.2.2 BYTE LteCQIParm::CQIValueCW1

8.309.2.3 BYTE LteCQIParm::ValidityCW0

8.309.2.4 BYTE LteCQIParm::ValidityCW1

8.310 lteEARFCN Struct Reference

Data Fields

- [BYTE status](#)
- [ULONG earfcn0](#)
- [ULONG earfcn1](#)

8.310.1 Detailed Description

This structure contains the parameters for WCDMA UARCFN.

Parameters

<i>status</i>	<ul style="list-style-type: none">• 0 - Disable• 1 - Enable
<i>earfcn0</i>	<ul style="list-style-type: none">• Primary DL EARFCN to which the UE is locked
<i>earfcn1</i>	<ul style="list-style-type: none">• Secondary DL EARFCN to which the UE is locked• Note : Make earfcn1 value equal to earfcn0 if only one EARFCN is desired.

8.310.2 Field Documentation

8.310.2.1 **ULONG** lteEARFCN::earfcn0

8.310.2.2 **ULONG** lteEARFCN::earfcn1

8.310.2.3 **BYTE** lteEARFCN::status

8.311 lteGsmCellInfo Struct Reference

Data Fields

- [BYTE cellReselPriority](#)
- [BYTE threshGsmHigh](#)
- [BYTE threshGsmLow](#)
- [BYTE nccPermitted](#)
- [BYTE cells_len](#)
- [gsmCellInfo](#) [GsmCellInfo](#) [255]

8.311.1 Detailed Description

This structure contains information about the LTE GSM Cell.

Parameters

<i>cellReselPriority</i>	<ul style="list-style-type: none"> • Priority of this frequency group. • Range: 0 to 7. • This field is only valid when ue_in_idle is TRUE.
<i>threshGsmHigh</i>	<ul style="list-style-type: none"> • Reselection threshold for high priority layers. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE.
<i>threshGsmLow</i>	<ul style="list-style-type: none"> • Reselection threshold for low priority layers. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE.
<i>nccPermitted</i>	<ul style="list-style-type: none"> • Bitmask specifying whether a neighbor with a specific network color code is to be reported. • Range: 0 to 255. • Bit n set to 1 means a neighbor with NCC n must be included in the report. This flag is synonymous with a blacklist in other RATs. • This field is only valid when ue_in_idle is TRUE.
<i>cells_len</i>	<ul style="list-style-type: none"> • Provides the number of set of gsm cells.
<i>GsmCellInfo[M↵ AX_DESCRIP↵ TION_LENGTH]</i>	<ul style="list-style-type: none"> • See gsmCellInfo for more information.

8.311.2 Field Documentation

8.311.2.1 **BYTE** `IteGsmCellInfo::cellReselPriority`

8.311.2.2 **BYTE** `IteGsmCellInfo::cells_len`

8.311.2.3 **gsmCellInfo** `IteGsmCellInfo::GsmCellInfo[255]`

8.311.2.4 **BYTE** `IteGsmCellInfo::nccPermitted`

8.311.2.5 **BYTE** `IteGsmCellInfo::threshGsmHigh`

8.311.2.6 **BYTE** `IteGsmCellInfo::threshGsmLow`

8.312 LTEInfo Struct Reference

Data Fields

- [BYTE](#) `band`
- [BYTE](#) `bandwidth`
- [WORD](#) `RXChan`
- [WORD](#) `TXChan`
- [BYTE](#) `emmState`
- [BYTE](#) `emmSubState`
- [BYTE](#) `emmConnState`

8.312.1 Detailed Description

Structure for storing the LTE information for the device.

Parameters

<i>band</i>	<ul style="list-style-type: none"> • LTE Band <ul style="list-style-type: none"> – 1 ~ 41 (Band in decimal) – 0xFF - Invalid
<i>bandwidth</i>	<ul style="list-style-type: none"> • BandWidth. <ul style="list-style-type: none"> – 0x00 - 1.4 MHz – 0x01 - 3 MHz – 0x02 - 5 MHz – 0x03 - 10 MHz – 0x04 - 15 MHz – 0x05 - 20 MHz – 0x06 - Invalid – 0xFF - Unknown
<i>RXChan</i>	<ul style="list-style-type: none"> • RX channel number in decimal <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>TXChan</i>	<ul style="list-style-type: none"> • TX channel number in decimal <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>emmState</i>	<ul style="list-style-type: none"> • EMM State. <ul style="list-style-type: none"> – 0x00 - Deregistered – 0x01 - Reg Initiated – 0x02 - Registered – 0x03 - TAU Initiated – 0x04 - SR Initiated – 0x05 - Dereg Initiated – 0x06 - Invalid – 0xFF - Unknown

<i>emmSubState</i>	<ul style="list-style-type: none"> • EMM Sub State. <ul style="list-style-type: none"> – 0xFF - NOT Applicable • When EMM_state is 0x00: <ul style="list-style-type: none"> – 0x00 - No IMSI – 0x01 - PLMN Search – 0x02 - Attach Needed – 0x03 - No Cell – 0x04 - Attaching – 0x05 - Normal Service – 0x06 - Limited Service – 0x07 - Waiting for PDN • When EMM_state is 0x01: <ul style="list-style-type: none"> – 0x00 - Waiting for NW – 0x01 - Waiting for ESM • When EMM_state is 0x02: <ul style="list-style-type: none"> – 0x00 - Normal Service – 0x01 - Update Needed – 0x02 - Attempt Update – 0x03 - No Cell – 0x04 - PLMN Search – 0x05 - Limited Service – 0x06 - MM Update – 0x07 - IMSI Detach – 0x08 - Waiting for ESM
<i>emmConnState</i>	<ul style="list-style-type: none"> • EMM Connected Mode State. <ul style="list-style-type: none"> – 0x00 - RRC Idle – 0x01 - Waiting RRC Cfm – 0x02 - RRC Connected – 0x03 - RRC Releasing – 0xFF - Unknown

8.312.2 Field Documentation

8.312.2.1 **BYTE** LTEInfo::band

8.312.2.2 **BYTE** LTEInfo::bandwidth

8.312.2.3 **BYTE** LTEInfo::emmConnState

8.312.2.4 **BYTE** LTEInfo::emmState

8.312.2.5 **BYTE** LTEInfo::emmSubState

8.312.2.6 **WORD** LTEInfo::RXChan

8.312.2.7 WORD LTEInfo::TXChan

8.313 LTEInfoInterfreq Struct Reference

Data Fields

- [BYTE ueInIdle](#)
- [BYTE freqsLen](#)
- [infoInterFreq InfoInterfreq](#) [255]

8.313.1 Detailed Description

This structure contains information about the LTE Inter-Frequency Network.

Parameters

<i>ueInIdle</i>	<ul style="list-style-type: none"> • TRUE if the UE is in Idle mode, otherwise FALSE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>freqsLen</i>	<ul style="list-style-type: none"> • Provides the number of set of inter frequency information. • If 0(zero), then no information follows it.
<i>InfoInterfreq</i> [<i>M</i> ↔ <i>AX_DESCRIP</i> ↔ <i>TION_LENGTH</i>]	<ul style="list-style-type: none"> • See infoInterFreq for more information.

8.313.2 Field Documentation

8.313.2.1 BYTE LTEInfoInterfreq::freqsLen

8.313.2.2 infoInterFreq LTEInfoInterfreq::InfoInterfreq[255]

8.313.2.3 BYTE LTEInfoInterfreq::ueInIdle

8.314 LTEInfoIntrafreq Struct Reference

Data Fields

- [BYTE ueInIdle](#)
- [BYTE plmn](#) [3]
- [WORD tac](#)
- [ULONG globalCellId](#)
- [WORD earfcn](#)
- [WORD servingCellId](#)
- [BYTE cellReselPriority](#)
- [BYTE sNonIntraSearch](#)
- [BYTE threshServingLow](#)
- [BYTE sIntraSearch](#)
- [BYTE cellsLen](#)
- [cellParams CellParams](#) [255]

8.314.1 Detailed Description

This structure contains information about the LTE Intra-Frequency Network.

Parameters

<i>ueInIdle</i>	<ul style="list-style-type: none"> • TRUE if the UE is in Idle mode, otherwise FALSE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>plmn</i> [<i>PLMN_L</i> ↔ <i>ENGTH</i>]	<ul style="list-style-type: none"> • PLMN ID coded as octet 3, 4, and 5.
<i>tac</i>	<ul style="list-style-type: none"> • Tracking area code. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>globalCellId</i>	<ul style="list-style-type: none"> • Global cell ID in the system information block. <ul style="list-style-type: none"> – 0xFFFFFFFF - Not Available
<i>earfcn</i>	<ul style="list-style-type: none"> • E-UTRA absolute radio frequency channel number of the serving cell. • Range: 0 to 65535. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>servingCellId</i>	<ul style="list-style-type: none"> • LTE serving cell ID. • Range: 0 to 503. • This is the cell ID of the serving cell and can be found in the cell list. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>cellReselPriority</i>	<ul style="list-style-type: none"> • Priority for serving frequency. • Range: 0 to 7. • This field is only valid when <i>ue_in_idle</i> is TRUE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>sNonIntraSearch</i>	<ul style="list-style-type: none"> • S non-intra search threshold to control non-intrafrequency searches. • Range: 0 to 31. • This field is only valid when <i>ue_in_idle</i> is TRUE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>threshServing</i> ↔ <i>Low</i>	<ul style="list-style-type: none"> • Serving cell low threshold. • Range: 0 to 31. • This field is only valid when <i>ue_in_idle</i> is TRUE. <ul style="list-style-type: none"> – 0xFF - Not Available

<i>sIntraSearch</i>	<ul style="list-style-type: none"> • S Intra search threshold. • Range: 0 to 31. • The current cell measurement must fall below this threshold to consider intrafrequency for reselection. • This field is only valid when ue_in_idle is TRUE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>cellsLen</i>	<ul style="list-style-type: none"> • Provides the number of set of cell params. • If 0(zero), then no information follows it.
<i>CellParams</i> [<i>M</i> ↔ <i>AX_DESCRIP</i> ↔ <i>TION_LENGTH</i>]	<ul style="list-style-type: none"> • See cellParams for more information.

8.314.2 Field Documentation

8.314.2.1 **cellParams** LTEInfoIntrafreq::CellParams[255]

8.314.2.2 **BYTE** LTEInfoIntrafreq::cellReselPriority

8.314.2.3 **BYTE** LTEInfoIntrafreq::cellsLen

8.314.2.4 **WORD** LTEInfoIntrafreq::earfcn

8.314.2.5 **ULONG** LTEInfoIntrafreq::globalCellId

8.314.2.6 **BYTE** LTEInfoIntrafreq::plmn[3]

8.314.2.7 **WORD** LTEInfoIntrafreq::servingCellId

8.314.2.8 **BYTE** LTEInfoIntrafreq::sIntraSearch

8.314.2.9 **BYTE** LTEInfoIntrafreq::sNonIntraSearch

8.314.2.10 **WORD** LTEInfoIntrafreq::tac

8.314.2.11 **BYTE** LTEInfoIntrafreq::threshServingLow

8.314.2.12 **BYTE** LTEInfoIntrafreq::ueInIdle

8.315 LTEInfoNeighboringGSM Struct Reference

Data Fields

- [BYTE](#) ueInIdle
- [BYTE](#) freqsLen
- [lteGsmCellInfo](#) [LteGsmCellInfo](#) [255]

8.315.1 Detailed Description

This structure contains information about the LTE Neighboring GSM Network.

Parameters

<i>ueIdle</i>	<ul style="list-style-type: none"> • TRUE if the UE is in Idle mode, otherwise FALSE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>freqsLen</i>	<ul style="list-style-type: none"> • Provides the number of set of LTE GSM cell information. • If 0(zero), then no information follows it.
<i>LteGsmCellInfo</i> [MAX_DESCRIPTION_LENGTH]	<ul style="list-style-type: none"> • See LteGsmCellInfo for more information.

8.315.2 Field Documentation

8.315.2.1 BYTE LTEInfoNeighboringGSM::freqsLen

8.315.2.2 LteGsmCellInfo LTEInfoNeighboringGSM::LteGsmCellInfo[255]

8.315.2.3 BYTE LTEInfoNeighboringGSM::ueIdle

8.316 LTEInfoNeighboringWCDMA Struct Reference

Data Fields

- [BYTE ueIdle](#)
- [BYTE freqsLen](#)
- [LteWcdmaCellInfo](#) [LTEWCDMACellInfo](#) [255]

8.316.1 Detailed Description

This structure contains information about the LTE Neighboring WCDMA Network.

Parameters

<i>ueIdle</i>	<ul style="list-style-type: none"> • TRUE if the UE is in Idle mode, otherwise FALSE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>freqsLen</i>	<ul style="list-style-type: none"> • Provides the number of set of LTE WCDMA cell information. • If 0(zero), then no information follows it.
<i>LTEWCDMACellInfo</i> [MAX_DESCRIPTION_LENGTH]	<ul style="list-style-type: none"> • See LteWcdmaCellInfo for more information.

8.316.2 Field Documentation

8.316.2.1 **BYTE** LTEInfoNeighboringWCDMA::freqsLen

8.316.2.2 **lteWcdmaCellInfo** LTEInfoNeighboringWCDMA::LTEWCDMACellInfo[255]

8.316.2.3 **BYTE** LTEInfoNeighboringWCDMA::ueInIdle

8.317 LteNasReleaseInfo_s Struct Reference

Data Fields

- [BYTE nas_release](#)
- [BYTE nas_major](#)
- [BYTE nas_minor](#)

8.317.1 Detailed Description

This structure contains LTE Nas Release Information

Parameters

<i>nas_release</i>	<ul style="list-style-type: none">• LTE NAS release
<i>nas_major</i>	<ul style="list-style-type: none">• LTE NAS version major
<i>nas_minor</i>	<ul style="list-style-type: none">• LTE NAS version minor

8.317.2 Field Documentation

8.317.2.1 **BYTE** LteNasReleaseInfo_s::nas_major

8.317.2.2 **BYTE** LteNasReleaseInfo_s::nas_minor

8.317.2.3 **BYTE** LteNasReleaseInfo_s::nas_release

8.318 ltePCI Struct Reference

Data Fields

- [BYTE status](#)
- [ULONG earfcn](#)
- [ULONG pci](#)

8.318.1 Detailed Description

This structure contains the parameters for WCDMA UARFCN.

Parameters

<i>status</i>	<ul style="list-style-type: none">• 0 - Disable• 1 - Enable
<i>earfcn</i>	<ul style="list-style-type: none">• UARFCN to which UE is locked
<i>pci</i>	<ul style="list-style-type: none">• PCI to which the UE is locked

8.318.2 Field Documentation

8.318.2.1 **ULONG** ItePCI::earfcn

8.318.2.2 **ULONG** ItePCI::pci

8.318.2.3 **BYTE** ItePCI::status

8.319 IteRsrpinformation Struct Reference

Data Fields

- [SHORT](#) rsrplevel

8.319.1 Detailed Description

This structure contains the LTE RSRP Information

Parameters

<i>rsrplevel</i>	<ul style="list-style-type: none">• LTE RSRP in dBm as a mesaured by L1. Range: -44 to -140(-44 means -44dBm, -140 means -140dBm).
------------------	--

8.319.2 Field Documentation

8.319.2.1 **SHORT** IteRsrpinformation::rsrplevel

8.320 LTERSRPThresh Struct Reference

Data Fields

- [BYTE LTERSRPThreshListLen](#)
- [WORD * pLTERSRPThreshList](#)

8.320.1 Detailed Description

This structure contains LTE RSRP threshold related parameters.

Parameters

<i>LTERSRP_↔</i> <i>ThreshListLen</i>	<ul style="list-style-type: none"> Length of the LTE RSRP threshold list parameter to follow
<i>pLTERSRP_↔</i> <i>ThreshList</i>	<ul style="list-style-type: none"> Array of RSRP thresholds (in units of 0.1 dBm) Maximum of 32 values Range for RSRP values: -140 to -44 (in dBm).

8.320.2 Field Documentation

8.320.2.1 **BYTE** LTERSRPThresh::LTERSRPThreshListLen

8.320.2.2 **WORD*** LTERSRPThresh::pLTERSRPThreshList

8.321 LTERSRQThresh Struct Reference

Data Fields

- [BYTE LTERSRQThreshListLen](#)
- [WORD * pLTERSRQThreshList](#)

8.321.1 Detailed Description

This structure contains LTE RSRQ threshold related parameters.

Parameters

<i>LTERSRQ_↔</i> <i>ThreshListLen</i>	<ul style="list-style-type: none"> Length of the LTE RSRQ threshold list parameter to follow
<i>pLTERSRQ_↔</i> <i>ThreshList</i>	<ul style="list-style-type: none"> Array of RSRQ thresholds (in units of 0.1 dBm) Maximum of 32 values. Range for RSRQ values: -20 to -3 (in dBm)
	Generated by Doxygen

8.321.2 Field Documentation

8.321.2.1 **BYTE** LTERSQRThresh::LTERSQRThreshListLen

8.321.2.2 **WORD*** LTERSQRThresh::pLTERSQRThreshList

8.322 LTERSSIThresh Struct Reference

Data Fields

- [BYTE](#) LTERSSIThreshListLen
- [WORD *](#) [pLTERSSIThreshList](#)

8.322.1 Detailed Description

This structure contains LTE RSSI threshold related parameters.

Parameters

<i>LTERSSI</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none">• Length of the LTE RSSI threshold list parameter to follow
<i>pLTERSSI</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none">• Array of RSSI thresholds (in units of 0.1 dBm)• Maximum of 32 values.• Range for RSSI values: -120 to 0 (in dBm)

8.322.2 Field Documentation

8.322.2.1 **BYTE** LTERSSIThresh::LTERSSIThreshListLen

8.322.2.2 **WORD*** LTERSSIThresh::pLTERSSIThreshList

8.323 LteSccRxInfoResp Struct Reference

Data Fields

- [SccRxInfo](#) * [pSccRxInfo](#)

8.323.1 Detailed Description

This structure contains information about the SLQSSwiGetLteSccRxInfo response parameters.

Parameters

<i>pSccRxInfo</i>	[Optional] <ul style="list-style-type: none"> • See SccRxInfo for more information
-------------------	---

8.323.2 Field Documentation

8.323.2.1 SccRxInfo* LteSccRxInfoResp::pSccRxInfo

8.324 LTESigRptCfg Struct Reference

Data Fields

- [BYTE rptRate](#)
- [BYTE avgPeriod](#)

8.324.1 Detailed Description

This structure contains LTE Signal Report Config parameters.

Parameters

<i>rptRate</i>	<ul style="list-style-type: none"> • Rate on how often the LTE signal must be checked for reporting • Values: <ul style="list-style-type: none"> – 0 - Report using the default configuration – 1 - Report every 1 sec – 2 - Report every 2 sec – 3 - Report every 3 sec – 4 - Report every 4 sec – 5 - Report every 5 sec
<i>avgPeriod</i>	<ul style="list-style-type: none"> • Averaging period to be used for the LTE signal • Values: <ul style="list-style-type: none"> – 0 - Average using the default configuration – 1 - Average over 1 sec – 2 - Average over 2 sec – 3 - Average over 3 sec – 4 - Average over 4 sec – 5 - Average over 5 sec – 6 - Average over 6 sec – 7 - Average over 7 sec – 8 - Average over 8 sec – 9 - Average over 9 sec – 10 - Average over 10 sec

8.324.2 Field Documentation

8.324.2.1 BYTE LTESigRptCfg::avgPeriod

8.324.2.2 BYTE LTESigRptCfg::rptRate

8.325 LTESigRptConfig Struct Reference

Data Fields

- [BYTE rptRate](#)
- [BYTE avgPeriod](#)

8.325.1 Detailed Description

This structure contains LTE RSRP threshold related parameters.

Parameters

<i>rptRate</i>	<ul style="list-style-type: none">• Rate on how often the LTE signal must be checked for reporting Values• 0 - Report using the default configuration• 1 - Report every 1 sec• 2 - Report every 2 sec• 3 - Report every 3 sec• 4 - Report every 4 sec• 5 - Report every 5 sec
<i>avgPeriod</i>	<ul style="list-style-type: none">• Averaging period to be used for the LTE signal.• Values<ul style="list-style-type: none">– 0 - Average using the default configuration– 1 - Average over 1 sec– 2 - Average over 2 sec– 3 - Average over 3 sec– 4 - Average over 4 sec– 5 - Average over 5 sec– 6 - Average over 6 sec– 7 - Average over 7 sec– 8 - Average over 8 sec– 9 - Average over 9 sec– 10 - Average over 10 sec

8.325.2 Field Documentation

8.325.2.1 BYTE LTESigRptConfig::avgPeriod

8.325.2.2 BYTE LTESigRptConfig::rptRate

8.326 lteSnrinformation Struct Reference

Data Fields

- [SHORT snrlevel](#)

8.326.1 Detailed Description

This structure contains the LTE SNR Information

Parameters

<i>snrlevel</i>	<ul style="list-style-type: none"> • LTE SNR level as a scaled integer in units of 0.1dB e.g. -16dB has a value of -160 and 24.6dB has value of 246.
-----------------	---

8.326.2 Field Documentation

8.326.2.1 SHORT lteSnrinformation::snrlevel

8.327 LTESNRThresh Struct Reference

Data Fields

- [BYTE LTESNRThresListLen](#)
- [SHORT * pLTESNRThresList](#)

8.327.1 Detailed Description

This structure contains LTE SNR threshold related parameters.

Parameters

<i>LTESNRThresListLen</i>	<ul style="list-style-type: none"> • Length of the LTE SNR threshold list parameter to follow
<i>pLTESNRThresList</i>	<ul style="list-style-type: none"> • Sequence of thresholds delimiting SNR event reporting bands • Every time a SNR value crosses a threshold value, an event report indication message with the new SNR value is sent to the requesting control point. For this field <ul style="list-style-type: none"> – For LTE, each SNR threshold value is a signed 2 Byte value – Maximum number of threshold values is 16 – At least one value must be specified – SNR level as a scaled integer in units of 0.1 dB; e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246
	Generated by Doxygen

8.327.2 Field Documentation

8.327.2.1 BYTE LTESNRThresh::LTESNRThresListLen

8.327.2.2 SHORT* LTESNRThresh::pLTESNRThresList

8.328 LTESNRThreshold Struct Reference

Data Fields

- [BYTE LTESNRThreshListLen](#)
- [WORD * pLTESNRThreshList](#)

8.328.1 Detailed Description

This structure contains LTE SNR threshold related parameters.

Parameters

<i>LTESNR↔ ThreshListLen</i>	<ul style="list-style-type: none">• Length of the LTE SNR threshold list parameter to follow
<i>pLTESNR↔ ThreshList</i>	<ul style="list-style-type: none">• Array of SNR thresholds (in units of 0.1 dB)• Maximum of 32 values• Range for SNR values: -20 to 30 (in dB).

8.328.2 Field Documentation

8.328.2.1 BYTE LTESNRThreshold::LTESNRThreshListLen

8.328.2.2 WORD* LTESNRThreshold::pLTESNRThreshList

8.329 LTESSInfo Struct Reference

Data Fields

- [INT8 rssi](#)
- [INT8 rsrq](#)
- [SHORT rsrp](#)
- [SHORT snr](#)

8.329.1 Detailed Description

This structure contains the parameters for LTE Signal Strength Information

Parameters

<i>rsqi</i>	<ul style="list-style-type: none"> • RSSI in dBm (signed value). • A value of -125 dBm or lower is used to indicate No Signal. <ul style="list-style-type: none"> – For CDMA and UMTS, this indicates forward link pilot Ec – For GSM, this indicates received signal strength
<i>rsrq</i>	<ul style="list-style-type: none"> • RSRQ value in dB (signed integer value) as measured by L1. • Range: -3 to -20 (-3 means -3 dB, -20 means -20 dB).
<i>rsrp</i>	<ul style="list-style-type: none"> • Current RSRP in dBm as measured by L1. • Range: -44 to -140 (-44 means -44 dBm, -140 means -140 dBm).
<i>snr</i>	<ul style="list-style-type: none"> • SNR level as a scaled integer in units of 0.1 dB. e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246,

8.329.2 Field Documentation

8.329.2.1 SHORT LTESSInfo::rsrp

8.329.2.2 INT8 LTESSInfo::rsrq

8.329.2.3 INT8 LTESSInfo::rsqi

8.329.2.4 SHORT LTESSInfo::snr

8.330 lteSSInfo Struct Reference

Data Fields

- int8_t [rsqi](#)
- int8_t [rsrq](#)
- int16_t [rsrp](#)
- int16_t [snr](#)

8.330.1 Detailed Description

Parameters

<i>rsqi</i>	RSSI in dBm.
<i>rsrq</i>	RSRQ value in dB
<i>rsrp</i>	Current RSRP in dBm as measured by L1.
<i>snr</i>	SNR level as a scaled integer in units of 0.1 dB.

8.330.2 Field Documentation

8.330.2.1 `int16_t lteSSInfo::rsrp`

8.330.2.2 `int8_t lteSSInfo::rsrq`

8.330.2.3 `int8_t lteSSInfo::rssi`

8.330.2.4 `int16_t lteSSInfo::snr`

8.331 LTESysInfo Struct Reference

Data Fields

- [sysInfoCommon](#) [sysInfoLTE](#)
- [BYTE](#) `lacValid`
- [WORD](#) `lac`
- [BYTE](#) `cellIdValid`
- [ULONG](#) `cellId`
- [BYTE](#) `regRejectInfoValid`
- [BYTE](#) `rejectSrvDomain`
- [BYTE](#) `rejCause`
- [BYTE](#) `networkIdValid`
- [BYTE](#) `MCC` [3]
- [BYTE](#) `MNC` [3]
- [BYTE](#) `tacValid`
- [WORD](#) `tac`

8.331.1 Detailed Description

Structure for storing the LTE System Information.

Parameters

<i>sysInfoLTE</i>	<ul style="list-style-type: none">• See sysInfoCommon for more information.
<i>lacValid</i>	<ul style="list-style-type: none">• Indicates whether the location area code is valid..<ul style="list-style-type: none">– 0x00 - Invalid– 0x01 - Valid– 0xFF - Not Available
<i>lac</i>	<ul style="list-style-type: none">• Location area code.• Only applies to 3GPP.<ul style="list-style-type: none">– 0xFFFF - Not Available

<i>cellIdValid</i>	<ul style="list-style-type: none"> Indicates whether the cell ID is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>cellId</i>	<ul style="list-style-type: none"> Cell ID. <ul style="list-style-type: none"> 0xFFFFFFFF - Not Available
<i>regRejectInfo</i> <i>Valid</i>	<ul style="list-style-type: none"> Indicates whether the registration reject information is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>rejectSrvDomain</i>	<ul style="list-style-type: none"> Type of service domain in which the registration is rejected. <ul style="list-style-type: none"> 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service 0x01 - Circuit-switched only 0x02 - Packet-switched only 0x03 - Circuit-switched and packet-switched 0x04 - Camped 0xFF - Not Available
<i>rejCause</i>	<ul style="list-style-type: none"> Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6]. <ul style="list-style-type: none"> 0xFF - Not Available
<i>networkIdValid</i>	<ul style="list-style-type: none"> Indicates whether the network ID is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>MCC</i> [<i>PLMN_L</i> <i>ENGTH</i>]	<ul style="list-style-type: none"> Mobile Country Code. MCC digits in ASCII characters
<i>MNC</i> [<i>PLMN_L</i> <i>ENGTH</i>]	<ul style="list-style-type: none"> Mobile Network Code. MNC digits in ASCII characters An unused byte is set to 0xFF. In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.
<i>tacValid</i>	<ul style="list-style-type: none"> Indicates whether tracking area code is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available

<i>tac</i>	<ul style="list-style-type: none">• Tracking area code.• Only applicable for LTE.<ul style="list-style-type: none">– 0xFFFF - Not Available
------------	--

8.331.2 Field Documentation

8.331.2.1 **ULONG** LTESysInfo::cellId

8.331.2.2 **BYTE** LTESysInfo::cellIdValid

8.331.2.3 **WORD** LTESysInfo::lac

8.331.2.4 **BYTE** LTESysInfo::lacValid

8.331.2.5 **BYTE** LTESysInfo::MCC[3]

8.331.2.6 **BYTE** LTESysInfo::MNC[3]

8.331.2.7 **BYTE** LTESysInfo::networkIdValid

8.331.2.8 **BYTE** LTESysInfo::regRejectInfoValid

8.331.2.9 **BYTE** LTESysInfo::rejCause

8.331.2.10 **BYTE** LTESysInfo::rejectSrvDomain

8.331.2.11 **sysInfoCommon** LTESysInfo::sysInfoLTE

8.331.2.12 **WORD** LTESysInfo::tac

8.331.2.13 **BYTE** LTESysInfo::tacValid

8.332 lteWcdmaCellInfo Struct Reference

Data Fields

- [WORD](#) uarfcn
- [BYTE](#) cellReselPriority
- [WORD](#) threshXhigh
- [WORD](#) threshXlow
- [BYTE](#) cellsLen
- [wcdmaCellInfo](#) [WCDMACellInfo](#) [255]

8.332.1 Detailed Description

This structure contains information about the LTE WCDMA Cell.

Parameters

<i>uarfcn</i>	<ul style="list-style-type: none"> • WCDMA layer frequency. • Range: 0 to 16383.
<i>cellReselPriority</i>	<ul style="list-style-type: none"> • Cell re-selection priority. • Range: 0 to 7. • This field is only valid when <i>ue_in_idle</i> is TRUE.
<i>threshXhigh</i>	<ul style="list-style-type: none"> • Re-selection low threshold. • Range: 0 to 31. • This field is only valid when <i>ue_in_idle</i> is TRUE.
<i>threshXlow</i>	<ul style="list-style-type: none"> • Re-selection high threshold. • Range: 0 to 31. • This field is only valid when <i>ue_in_idle</i> is TRUE.
<i>cellsLen</i>	<ul style="list-style-type: none"> • Provides the number of set of WCDMA cells.
<i>WCDMACellInfo[MAX_DESCRIPTION_LENGTH]</i>	<ul style="list-style-type: none"> • See wcdmaCellInfo for more information.

8.332.2 Field Documentation

8.332.2.1 **BYTE** *IteWcdmaCellInfo::cellReselPriority*

8.332.2.2 **BYTE** *IteWcdmaCellInfo::cellsLen*

8.332.2.3 **WORD** *IteWcdmaCellInfo::threshXhigh*

8.332.2.4 **WORD** *IteWcdmaCellInfo::threshXlow*

8.332.2.5 **WORD** *IteWcdmaCellInfo::uarfcn*

8.332.2.6 **wcdmaCellInfo** *IteWcdmaCellInfo::WCDMACellInfo[255]*

8.333 messageModeTlv Struct Reference

Data Fields

- [uint8_t TlvPresent](#)
- [sMSMessageModelInfo MessageModelInfo](#)

8.333.1 Detailed Description

Parameters

<i>TlvPresent</i>	<ul style="list-style-type: none"> • Boolean indicating the presence of the TLV in the QMI response
<i>MessageModeInfo</i>	<ul style="list-style-type: none"> • Message Mode • See sMSMessageModelInfo for more information

8.333.2 Field Documentation

8.333.2.1 `sMSMessageModelInfo messageModeTlv::MessageModelInfo`

8.333.2.2 `uint8_t messageModeTlv::TlvPresent`

8.334 messageWaitingInfoContent Struct Reference

Data Fields

- [BYTE msgType](#)
- [BYTE activeInd](#)
- [BYTE msgCount](#)

8.334.1 Detailed Description

This structure contains message waiting information per instance

Parameters

<i>msgType</i>	<ul style="list-style-type: none"> • Message type <ul style="list-style-type: none"> – 0x00 - MWI_MESSAGE_TYPE_VOICMAIL - Voicemail – 0x01 - MWI_MESSAGE_TYPE_FAX - Fax – 0x02 - MWI_MESSAGE_TYPE_EMAIL - Email – 0x03 - MWI_MESSAGE_TYPE_OTHER - Other – 0x04 - MWI_MESSAGE_TYPE_VIDEOMAIL - Videomail
<i>activeInd</i>	<ul style="list-style-type: none"> • Indicates whether the indication is active <ul style="list-style-type: none"> – 0x00 - Inactive – 0x01 - Active
<i>msgCount</i>	<ul style="list-style-type: none"> • Number of messages

8.334.2 Field Documentation

8.334.2.1 **BYTE** `messageWaitingInfoContent::activeInd`

8.334.2.2 **BYTE** `messageWaitingInfoContent::msgCount`

8.334.2.3 **BYTE** `messageWaitingInfoContent::msgType`

8.335 minBasedIMSI Struct Reference

Data Fields

- [BYTE](#) `mccM` [3]
- [WORD](#) `imsiM1112`
- [BYTE](#) `imsiMS1` [7]
- [BYTE](#) `imsiMS2` [3]

8.335.1 Detailed Description

This structure contains the parameters for Min based IMSI Information

Parameters

<i>mccM</i>	<ul style="list-style-type: none"> • ASCII character representation of MCC_M
<i>imsiM1112</i>	<ul style="list-style-type: none"> • ASCII character representation of IMSI_M_11_12 value <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>imsiMS1</i>	<ul style="list-style-type: none"> • ASCII character representation of IMSI_M_S1 value
<i>imsiMS2</i>	<ul style="list-style-type: none"> • ASCII character representation of IMSI_M_S2 value

8.335.2 Field Documentation

8.335.2.1 **WORD** `minBasedIMSI::imsiM1112`

8.335.2.2 **BYTE** `minBasedIMSI::imsiMS1`[7]

8.335.2.3 **BYTE** `minBasedIMSI::imsiMS2`[3]

8.335.2.4 **BYTE** `minBasedIMSI::mccM`[3]

8.336 mitigationDevList Struct Reference

Data Fields

- [BYTE mitigationDevIdLen](#)
- [CHAR mitigationDevId](#) [255]
- [BYTE maxMitigationLevel](#)

8.336.1 Detailed Description

This structure contains mitigation devices list

Parameters

<i>mitigationDevIdLen</i>	<ul style="list-style-type: none"> • Number of sets of the following elements <ul style="list-style-type: none"> – mitigation_dev_id
<i>mitigationDevId</i>	<ul style="list-style-type: none"> • Mitigation device ID.
<i>maxMitigationLevel</i>	<ul style="list-style-type: none"> • Maximum valid mitigation level. • Valid range - 0 to max_mitigation_level.

8.336.2 Field Documentation

8.336.2.1 **BYTE** mitigationDevList::maxMitigationLevel

8.336.2.2 **CHAR** mitigationDevList::mitigationDevId[255]

8.336.2.3 **BYTE** mitigationDevList::mitigationDevIdLen

8.337 MNRInfo Struct Reference

Data Fields

- [WORD mcc](#)
- [WORD mnc](#)
- [ULONG rat](#)

8.337.1 Detailed Description

Structure contains Manual Network Register Information parameters

Parameters

<i>mcc</i>	<ul style="list-style-type: none"> • A 16-bit integer representation of Mobile Country Code. Range - 0 to 999.
<i>mnc</i>	<ul style="list-style-type: none"> • A 16-bit integer representation of Mobile Network Code. Range - 0 to 999.
<i>rat</i>	<ul style="list-style-type: none"> • Radio access technology for which to register. <ul style="list-style-type: none"> – 0x04 - RADIO_IF_GSM – 0x05 - RADIO_IF_UMTS – 0x08 - RADIO_IF_LTE

8.337.2 Field Documentation

8.337.2.1 WORD MNRInfo::mcc

8.337.2.2 WORD MNRInfo::mnc

8.337.2.3 ULONG MNRInfo::rat

8.338 ModifyProfileIn Struct Reference

Data Fields

- [BYTE * pProfileID](#)
- [BYTE * pProfileType](#)
- [QmiProfileInfo curProfile](#)

8.338.1 Detailed Description

This structure contains input parameters for SLQSMModifyProfile

Parameters

<i>ProfileID</i>	<ul style="list-style-type: none"> • 1 to 16 for 3GPP profile (EM/MC73xx or earlier) • 1 to 24 for 3GPP profile (EM/MC74xx onwards) • 101 to 106 for 3GPP2 profile
<i>ProfileType</i>	<ul style="list-style-type: none"> • Identifies the technology type of the profile <ul style="list-style-type: none"> – 0x00 - 3GPP – 0x01 - 3GPP2 – NULL is not allowed
<i>curProfile</i>	<ul style="list-style-type: none"> • Contains Union of profile(3GPP/3GPP2) structures
	Generated by Doxygen

8.338.2 Field Documentation

8.338.2.1 QmiProfileInfo ModifyProfileIn::curProfile

8.338.2.2 BYTE* ModifyProfileIn::pProfileID

8.338.2.3 BYTE* ModifyProfileIn::pProfileType

8.339 ModifyProfileOut Struct Reference

Data Fields

- USHORT * pExtErrorCode

8.339.1 Detailed Description

This structure contains out parameters for SLQSMModifyProfile

Parameters

<i>pExtErrorCode</i>	<ul style="list-style-type: none">• The extended error code received from DS Profile subsystem of type eWDS_ERR_PROFILE↔_REG_XXX.• Error code will only be present if error code eQCWWAN_ERR_QMI_EXTENDED_INTER↔NAL is returned by device.• See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description.
----------------------	--

8.339.2 Field Documentation

8.339.2.1 USHORT* ModifyProfileOut::pExtErrorCode

8.340 msgWaitingInfo Struct Reference

Data Fields

- BYTE numInstances
- messageWaitingInfoContent msgWaitInfo [0xFF]

8.340.1 Detailed Description

This structure holds information related to message waiting information

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • Number of sets of the elements in structure messageWaitingInfoContent
<i>msgWaitInfo</i>	<ul style="list-style-type: none"> • Pointer to structure of messageWaitingInfoContent. <ul style="list-style-type: none"> – See messageWaitingInfoContent for more information.

8.340.2 Field Documentation

8.340.2.1 [messageWaitingInfoContent](#) `msgWaitingInfo::msgWaitInfo[0xFF]`

8.340.2.2 **BYTE** `msgWaitingInfo::numInstances`

8.341 [namName](#) Struct Reference

Data Fields

- [BYTE](#) `namNameLen`
- [BYTE](#) `namName` [12]

8.341.1 Detailed Description

This structure contains the parameters for NAM Name Information

Parameters

<i>namNameLen</i>	<ul style="list-style-type: none"> • Number of sets of the following elements: <ul style="list-style-type: none"> – <code>nam_name</code> • If zero(0), then no information follows.
<i>namName</i>	<ul style="list-style-type: none"> • Name information in ASCII. The maximum length of <code>nam_name</code> is 12.

8.341.2 Field Documentation

8.341.2.1 **BYTE** `namName::namName[12]`

8.341.2.2 **BYTE** `namName::namNameLen`

8.342 [nas_acqOrderPref](#) Struct Reference

Data Fields

- `uint8_t` [acqOrdeLen](#)
- `uint8_t *` [pAcqOrder](#)

8.342.1 Detailed Description

Contain the Acquisition Order Preference.

Parameters

<i>acqOrdeLen</i>	<ul style="list-style-type: none"> Number of sets of the following elements.
<i>pAcqOrder</i>	<ul style="list-style-type: none"> Acquisition order preference to be set. Values: <ul style="list-style-type: none"> 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X 0x02 - NAS_RADIO_IF_CDMA_1XEVD0 - cdma2000 HRPD (1xEV-DO) 0x04 - NAS_RADIO_IF_GSM - GSM 0x05 - NAS_RADIO_IF_UMTS - UMTS 0x08 - NAS_RADIO_IF_LTE - LTE 0x09 - NAS_RADIO_IF_TDSCDMA - TD-SCDMA

8.342.2 Field Documentation

8.342.2.1 `uint8_t nas_acqOrderPref::acqOrdeLen`

8.342.2.2 `uint8_t* nas_acqOrderPref::pAcqOrder`

8.343 nas_AddCDMASysInfo Struct Reference

Data Fields

- `uint16_t` [geoSysIdx](#)
- `uint16_t` [regPrd](#)

8.343.1 Detailed Description

Structure for storing the Additional CDMA System Information.

Parameters

<i>geoSysIdx</i>	<ul style="list-style-type: none"> System table index referencing the beginning of the geo in which the current serving system is present. When the system index is not known, 0xFFFF is used.
<i>regPrd</i>	<ul style="list-style-type: none"> Registration period after the CDMA system is acquired. When the CDMA registration period is not valid, 0xFFFF is used.

8.343.2 Field Documentation

8.343.2.1 uint16_t nas_AddCDMASysInfo::geoSysIdx

8.343.2.2 uint16_t nas_AddCDMASysInfo::regPrd

8.344 nas_AddSysInfo Struct Reference

Data Fields

- uint16_t [geoSysIdx](#)
- uint32_t [cellBroadcastCap](#)

8.344.1 Detailed Description

Structure for storing the Additional GSM and WCDMA System Information.

Parameters

<i>geoSysIdx</i>	<ul style="list-style-type: none"> • System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
<i>cellBroadcastCap</i>	<ul style="list-style-type: none"> • Cell broadcast capability of the serving system. • When the CDMA registration period is not valid, 0xFFFF is used. <ul style="list-style-type: none"> – NAS_CELL_BROADCAST_CAP_UNKNOWN - Cell broadcast support is unknown – NAS_CELL_BROADCAST_CAP_OFF - Cell broadcast is not supported – NAS_CELL_BROADCAST_CAP_ON - Cell broadcast is supported

8.344.2 Field Documentation

8.344.2.1 uint32_t nas_AddSysInfo::cellBroadcastCap

8.344.2.2 uint16_t nas_AddSysInfo::geoSysIdx

8.345 nas_CallBarringSysInfo Struct Reference

Data Fields

- uint32_t [csBarStatus](#)
- uint32_t [psBarStatus](#)

8.345.1 Detailed Description

Structure for storing the GSM and WCDMA Call Barring System Information.

Parameters

<i>csBarStatus</i>	<ul style="list-style-type: none"> • Call barring status for circuit-switched calls. <ul style="list-style-type: none"> – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown
<i>psBarStatus</i>	<ul style="list-style-type: none"> • Call barring status for packet-switched calls. <ul style="list-style-type: none"> – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown

8.345.2 Field Documentation

8.345.2.1 uint32_t nas_CallBarringSysInfo::csBarStatus

8.345.2.2 uint32_t nas_CallBarringSysInfo::psBarStatus

8.346 nas_callBarStatus Struct Reference

Data Fields

- uint32_t [csBarStatus](#)
- uint32_t [psBarStatus](#)

8.346.1 Detailed Description

This structure contains Call Barring Status.

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>csBarStatus</i>	<ul style="list-style-type: none"> • Call Barring Status for circuit-switched calls. • Values: <ul style="list-style-type: none"> • NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only • NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only • NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type • NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types • NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown
<i>psBarStatus</i>	<ul style="list-style-type: none"> • Call Barring Status for packet-switched calls. • Values: <ul style="list-style-type: none"> – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown

8.346.2 Field Documentation

8.346.2.1 uint32_t nas_callBarStatus::csBarStatus

8.346.2.2 uint32_t nas_callBarStatus::psBarStatus

8.347 nas_CDMAECIOThresh Struct Reference

Data Fields

- uint8_t [CDMAECIOThreshListLen](#)
- int16_t * [pCDMAECIOThreshList](#)

8.347.1 Detailed Description

This structure contains CDMA ECIO threshold related parameters.

Parameters

<i>CDMAECIO↔ ThreshListLen</i>	<ul style="list-style-type: none"> • Length of the CDMA ECIO threshold list parameter to follow
<i>pCDMAECIO↔ ThreshList</i>	<ul style="list-style-type: none"> • Array of ECIO thresholds (in units of 0.1 dB) • Maximum of 32 values. Range for ECIO values: -31.5 to 0 (in dB)

8.347.2 Field Documentation

8.347.2.1 `uint8_t nas_CDMAECIOThresh::CDMAECIOThreshListLen`

8.347.2.2 `int16_t* nas_CDMAECIOThresh::pCDMAECIOThreshList`

8.348 nas_CDMAInfo Struct Reference

Data Fields

- `uint16_t sid`
- `uint16_t nid`
- `uint16_t baselId`
- `uint16_t refpn`
- `uint32_t baseLat`
- `uint32_t baseLong`

8.348.1 Detailed Description

This structure contains information about the CDMA Network.

Parameters

<i>sid</i>	<ul style="list-style-type: none">• System ID.<ul style="list-style-type: none">– 0xFFFF - Not Available
<i>nid</i>	<ul style="list-style-type: none">• Network ID.<ul style="list-style-type: none">– 0xFFFF - Not Available
<i>baselId</i>	<ul style="list-style-type: none">• Base station ID.<ul style="list-style-type: none">– 0xFFFF - Not Available
<i>refpn</i>	<ul style="list-style-type: none">• Reference PN.<ul style="list-style-type: none">– 0xFFFF - Not Available
<i>baseLat</i>	<ul style="list-style-type: none">• Latitude of the current base station in units of 0.25 sec.<ul style="list-style-type: none">– 0xFFFFFFFF - Not Available
<i>baseLong</i>	<ul style="list-style-type: none">• Longitude of the current base station in units of 0.25 sec.<ul style="list-style-type: none">– 0xFFFFFFFF - Not Available

8.348.2 Field Documentation

8.348.2.1 `uint16_t nas_CDMAInfo::baseId`

8.348.2.2 `uint32_t nas_CDMAInfo::baseLat`

8.348.2.3 `uint32_t nas_CDMAInfo::baseLong`

8.348.2.4 `uint16_t nas_CDMAInfo::nid`

8.348.2.5 `uint16_t nas_CDMAInfo::refpn`

8.348.2.6 `uint16_t nas_CDMAInfo::sid`

8.349 nas_CDMARSSIThresh Struct Reference

Data Fields

- `uint8_t` [CDMARSSIThreshListLen](#)
- `int16_t *` [pCDMARSSIThreshList](#)

8.349.1 Detailed Description

This structure contains CDMA RSSI threshold related parameters.

Parameters

<i>CDMARSSI</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> • Length of the CDMARSSI threshold list parameter to follow
<i>pCDMARSSI</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> • Array of RSSI thresholds (in units of 0.1 dBm) • maximum of 32 values. • Range for RSSI values:-105 to -21 (in dBm).

8.349.2 Field Documentation

8.349.2.1 `uint8_t nas_CDMARSSIThresh::CDMARSSIThreshListLen`

8.349.2.2 `int16_t*` `nas_CDMARSSIThresh::pCDMARSSIThreshList`

8.350 nas_CDMA SysInfo Struct Reference

Data Fields

- [nas_sysInfoCommon sysInfoCDMA](#)

- uint8_t [isSysPrIMatchValid](#)
- uint8_t [isSysPrIMatch](#)
- uint8_t [pRevInUseValid](#)
- uint8_t [pRevInUse](#)
- uint8_t [bsPRevValid](#)
- uint8_t [bsPRev](#)
- uint8_t [ccsSupportedValid](#)
- uint8_t [ccsSupported](#)
- uint8_t [cdmaSysIdValid](#)
- uint16_t [systemID](#)
- uint16_t [networkID](#)
- uint8_t [bsInfoValid](#)
- uint16_t [baseId](#)
- uint32_t [baseLat](#)
- uint32_t [baseLong](#)
- uint8_t [packetZoneValid](#)
- uint16_t [packetZone](#)
- uint8_t [networkIdValid](#)
- uint8_t [MCC](#) [3]
- uint8_t [MNC](#) [3]

8.350.1 Detailed Description

Structure for storing the CDMA System Information.

Parameters

<i>sysInfoCDMA</i>	<ul style="list-style-type: none"> • See sysInfoCommon for more information.
<i>isSysPrIMatchValid</i>	<ul style="list-style-type: none"> • Indicates whether the system PRL match is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>isSysPrIMatch</i>	<ul style="list-style-type: none"> • Indicates whether the system is in a PRL. • Only applies to CDMA/HDR. <ul style="list-style-type: none"> – 0x00 - System is not in a PRL – 0x01 - System is in a PRL – 0xFF - Not Available • If the system is not in a PRL, roam_status carries the value from the default roaming indicator in the PRL. • If the system is in a PRL, roam_status is set to the value based on the standard specification.
<i>pRevInUseValid</i>	<ul style="list-style-type: none"> • Indicates whether the P_Rev in use is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

<i>pRevInUse</i>	<ul style="list-style-type: none"> • The lesser of the base station P_Rev and mobile P_Rev • Only applicable for CDMA. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>bsPRevValid</i>	<ul style="list-style-type: none"> • Indicates whether the base station P_Rev is valid <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>bsPRev</i>	<ul style="list-style-type: none"> • Base station P_Rev. • Only applicable for CDMA. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>ccsSupported</i> ↔ <i>Valid</i>	<ul style="list-style-type: none"> • Indicates whether the supported concurrent service is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>ccsSupported</i>	<ul style="list-style-type: none"> • Whether concurrent service is supported. • Only applicable for CDMA. <ul style="list-style-type: none"> – 0x00 - Not supported – 0x01 - Supported – 0xFF - Not Available
<i>cdmaSysIdValid</i>	<ul style="list-style-type: none"> • Indicates whether the CDMA system ID is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>systemID</i>	<ul style="list-style-type: none"> • System ID. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>networkID</i>	<ul style="list-style-type: none"> • Network ID. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>bsInfoValid</i>	<ul style="list-style-type: none"> • Indicates whether the base station information is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

<i>baseLat</i>	<ul style="list-style-type: none"> • Base station latitude in units of 0.25 sec. • Expressed as a two's complement signed number with positive numbers signifying North latitudes. <ul style="list-style-type: none"> – 0xFFFFFFFF - Not Available
<i>baseLong</i>	<ul style="list-style-type: none"> • Base station longitude in units of 0.25 sec. • Expressed as a two's complement signed number with positive numbers signifying East latitudes. <ul style="list-style-type: none"> – 0xFFFFFFFF - Not Available
<i>packetZoneValid</i>	<ul style="list-style-type: none"> • Indicates whether the packet zone is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>packetZone</i>	<ul style="list-style-type: none"> • Packet zone (8-bit). <ul style="list-style-type: none"> – 0xFFFF indicates no packet zone. • Only applicable for CDMA.
<i>networkIdValid</i>	<ul style="list-style-type: none"> • Indicates whether the network ID is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>MCC[PLMN_LEN-1:0]</i>	<ul style="list-style-type: none"> • Mobile Country Code. • MCC digits in ASCII characters • MCC wildcard value is returned as {'3', 0xFF, 0xFF}.
<i>MNC[PLMN_LEN-1:0]</i>	<ul style="list-style-type: none"> • Mobile Network Code. • MNC digits in ASCII characters • An unused byte is set to 0xFF. • MNC wildcard value is returned as {'7', 0xFF, 0xFF}.

8.350.2 Field Documentation

8.350.2.1 uint16_t nas_CDMA SysInfo::baseId

8.350.2.2 uint32_t nas_CDMA SysInfo::baseLat

8.350.2.3 uint32_t nas_CDMA SysInfo::baseLong

8.350.2.4 uint8_t nas_CDMA SysInfo::bsInfoValid

- 8.350.2.5 `uint8_t nas_CDMASysInfo::bsPRev`
- 8.350.2.6 `uint8_t nas_CDMASysInfo::bsPRevValid`
- 8.350.2.7 `uint8_t nas_CDMASysInfo::ccsSupported`
- 8.350.2.8 `uint8_t nas_CDMASysInfo::ccsSupportedValid`
- 8.350.2.9 `uint8_t nas_CDMASysInfo::cdmaSysIdValid`
- 8.350.2.10 `uint8_t nas_CDMASysInfo::isSysPriMatch`
- 8.350.2.11 `uint8_t nas_CDMASysInfo::isSysPriMatchValid`
- 8.350.2.12 `uint8_t nas_CDMASysInfo::MCC[3]`
- 8.350.2.13 `uint8_t nas_CDMASysInfo::MNC[3]`
- 8.350.2.14 `uint16_t nas_CDMASysInfo::networkID`
- 8.350.2.15 `uint8_t nas_CDMASysInfo::networkIdValid`
- 8.350.2.16 `uint16_t nas_CDMASysInfo::packetZone`
- 8.350.2.17 `uint8_t nas_CDMASysInfo::packetZoneValid`
- 8.350.2.18 `uint8_t nas_CDMASysInfo::pRevInUse`
- 8.350.2.19 `uint8_t nas_CDMASysInfo::pRevInUseValid`
- 8.350.2.20 `nas_sysInfoCommon nas_CDMASysInfo::sysInfoCDMA`
- 8.350.2.21 `uint16_t nas_CDMASysInfo::systemID`

8.351 `nas_CDMASysInfoExt` Struct Reference

Data Fields

- `uint16_t` [MCC](#)
- `uint8_t` [imsi_11_12](#)

8.351.1 Detailed Description

This structure contains CDMA system information extension

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>MCC</i>	<ul style="list-style-type: none">• Mobile Country Code
<i>imsi_11_12</i>	<ul style="list-style-type: none">• IMSI_11_12

8.351.2 Field Documentation

8.351.2.1 uint8_t nas_CDMA SysInfoExt::imsi_11_12

8.351.2.2 uint16_t nas_CDMA SysInfoExt::MCC

8.352 nas_cellParams Struct Reference

Data Fields

- uint16_t [pci](#)
- int16_t [rsrq](#)
- int16_t [rsrp](#)
- int16_t [rssi](#)
- int16_t [srxlev](#)

8.352.1 Detailed Description

This structure contains information about the Cell parameters.

Parameters

<i>pci</i>	<ul style="list-style-type: none">• Physical cell ID.• Range: 0 to 503.
<i>rsrq</i>	<ul style="list-style-type: none">• Current RSRQ in 1/10 dB as measured by L1.• Range: -20.0 dB to -3.0 dB.
<i>rsrp</i>	<ul style="list-style-type: none">• Current RSRP in 1/10 dBm as measured by L1.• Range: -140.0 dBm to -44.0 dBm.
<i>rssi</i>	<ul style="list-style-type: none">• Current RSSI in 1/10 dBm as measured by L1.• Range: -120.0 dBm to 0.
<i>srxlev</i>	<ul style="list-style-type: none">• Cell selection Rx level (Srxlev) value.• Range: -128 to 128.• This field is only valid when ue_in_idle is TRUE.
Generated by Doxygen	

8.352.2 Field Documentation

8.352.2.1 uint16_t nas_cellParams::pci

8.352.2.2 int16_t nas_cellParams::rsrp

8.352.2.3 int16_t nas_cellParams::rsrq

8.352.2.4 int16_t nas_cellParams::rssi

8.352.2.5 int16_t nas_cellParams::srxlev

8.353 nas_CommInfo Struct Reference

Data Fields

- int8_t [temperature](#)
- uint8_t [modemMode](#)
- uint8_t [systemMode](#)
- uint8_t [imsRegState](#)
- uint8_t [psState](#)

8.353.1 Detailed Description

Structure for storing the common information for the device.

Parameters

<i>temperature</i>	<ul style="list-style-type: none"> • Temperature. <ul style="list-style-type: none"> – 8-bit signed integer – 0xFF - Not Available.
<i>modemMode</i>	<ul style="list-style-type: none"> • Modem Operating Mode. <ul style="list-style-type: none"> – 0x00 - POWERING OFF – 0x01 - FACTORY TEST – 0x02 - OFFLINE – 0x03 - OFFLINE_AMPS – 0x04 - OFFLINE_CDMA – 0x05 - ONLINE – 0x06 - LOW POWER MODE – 0x07 - RESETTING – 0x08 - NETWORK TEST – 0x09 - OFFLINE REQUEST – 0x0A - PSEUDO ONLINE – 0x0B - RESETTING MODEM – 0xFF - Unknown

<i>systemMode</i>	<ul style="list-style-type: none"> • System Acquisition Mode. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - AMPS – 0x02 - CDMA – 0x03 - GSM – 0x04 - HDR – 0x05 - WCDMA – 0x06 - GPS – 0x08 - WLAN – 0x09 - LTE – 0xFF - Unknown
<i>imsRegState</i>	<ul style="list-style-type: none"> • IMS Registration State. <ul style="list-style-type: none"> – 0x00 - NO SRV – 0x01 - IN PROG – 0x02 - FAILED – 0x03 - LIMITED – 0x04 - FULL SRV – 0xFF - Unknown
<i>psState</i>	<ul style="list-style-type: none"> • PS Attach State. <ul style="list-style-type: none"> – 0x00 - Attached – 0x01 - Detached – 0xFF - Unknown

8.353.2 Field Documentation

8.353.2.1 `uint8_t nas_CommInfo::imsRegState`

8.353.2.2 `uint8_t nas_CommInfo::modemMode`

8.353.2.3 `uint8_t nas_CommInfo::psState`

8.353.2.4 `uint8_t nas_CommInfo::systemMode`

8.353.2.5 `int8_t nas_CommInfo::temperature`

8.354 nas_CSGID Struct Reference

Data Fields

- `uint16_t mcc`
- `uint16_t mnc`
- `uint8_t mncPcsDigits`
- `uint32_t id`
- `uint8_t rat`

8.354.1 Detailed Description

Contain the [CSGID](#).

Parameters

<i>mcc</i>	<ul style="list-style-type: none"> MCC value. Range 0 to 999
<i>mnc</i>	<ul style="list-style-type: none"> MNC value. Range 0 to 999
<i>mncPcsDigits</i>	<ul style="list-style-type: none"> TRUE - MNC is a three-digit value; e.g., a reported value of 90 corresponds to an MNC value of 090 FALSE - MNC is a two-digit value; e.g., a reported value of 90 corresponds to an MNC value of 90
<i>id</i>	<ul style="list-style-type: none"> Closed subscriber group identifier.
<i>rat</i>	<ul style="list-style-type: none"> Radio interface technology of the CSG network. Values: <ul style="list-style-type: none"> 0x04 - RADIO_IF_GSM - GSM 0x05 - RADIO_IF_UMTS - UMTS 0x08 - RADIO_IF_LTE - LTE 0x09 - RADIO_IF_TDSCDMA - TDS

8.354.2 Field Documentation

8.354.2.1 `uint32_t nas_CSGID::id`

8.354.2.2 `uint16_t nas_CSGID::mcc`

8.354.2.3 `uint16_t nas_CSGID::mnc`

8.354.2.4 `uint8_t nas_CSGID::mncPcsDigits`

8.354.2.5 `uint8_t nas_CSGID::rat`

8.355 nas_currentPLMN Struct Reference

Data Fields

- `uint16_t` [MCC](#)
- `uint16_t` [MNC](#)
- `uint8_t` [netDescrLength](#)
- `uint8_t` [netDescr](#) [255]

8.355.1 Detailed Description

This structure contains the current PLMN parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>MCC</i>	<ul style="list-style-type: none"> • mobile country code <ul style="list-style-type: none"> – A 16 bit representation of MCC – Range 0 to 999
<i>MNC</i>	<ul style="list-style-type: none"> • mobile network code <ul style="list-style-type: none"> – A 16 bit representation of MNC – Range 0 to 999
<i>netDescrLength</i>	<ul style="list-style-type: none"> • Length of Network description field • Defaults to zero
<i>netDescr</i>	<ul style="list-style-type: none"> • Network Description <ul style="list-style-type: none"> – optional string containing network name or description

8.355.2 Field Documentation

8.355.2.1 uint16_t nas_currentPLMN::MCC

8.355.2.2 uint16_t nas_currentPLMN::MNC

8.355.2.3 uint8_t nas_currentPLMN::netDescr[255]

8.355.2.4 uint8_t nas_currentPLMN::netDescrLength

8.356 nas_dataSrvCapabilities Struct Reference

Data Fields

- uint8_t [dataCapabilitiesLen](#)
- uint8_t [dataCapabilities](#) [32]

8.356.1 Detailed Description

This structure contains the data services capability

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>data↔</i> <i>CapabilitiesLen</i>	<ul style="list-style-type: none"> • Length of data capabilities list • Defaults to zero
<i>dataCapabilities</i>	<ul style="list-style-type: none"> • List of data capabilities • Values: <ul style="list-style-type: none"> – 0x01 - GPRS – 0x02 - EDGE – 0x03 - HSDPA – 0x04 - HSUPA – 0x05 - WCDMA – 0x06 - CDMA – 0x07 - EV-DO Rev0 – 0x08 - EV-DO RevA – 0x09 - GSM – 0x0A - EV-DO Rev B – 0x0B - LTE – 0x0C - HSDPA+ – 0x0D - DC-HSDPA+

8.356.2 Field Documentation

8.356.2.1 `uint8_t nas_dataSrvCapabilities::dataCapabilities[32]`

8.356.2.2 `uint8_t nas_dataSrvCapabilities::dataCapabilitiesLen`

8.357 nas_detailSvcInfo Struct Reference

Data Fields

- `uint8_t srvStatus`
- `uint8_t srvCapability`
- `uint8_t hdrSrvStatus`
- `uint8_t hdrHybrid`
- `uint8_t isSysForbidden`

8.357.1 Detailed Description

This structure contains Detailed Service information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>srvStatus</i>	<ul style="list-style-type: none"> • Service status • Values: <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Limited service – 0x02 - Service available – 0x03 - Limited regional service – 0x04 - MS in power save or deep sleep
<i>srvCapability</i>	<ul style="list-style-type: none"> • System's service capability • Values: <ul style="list-style-type: none"> – 0x00 - No Service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - MS found the right system but not yet registered/attached
<i>hdrSrvStatus</i>	<ul style="list-style-type: none"> • HDR service status • Values: <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Limited service – 0x02 - Service available – 0x03 - Limited regional service – 0x04 - MS in power save or deep sleep
<i>hdrHybrid</i>	<ul style="list-style-type: none"> • HDR hybrid information • Values: <ul style="list-style-type: none"> – 0x00 - System is not hybrid – 0x01 - System is hybrid
<i>isSysForbidden</i>	<ul style="list-style-type: none"> • Forbidden system information • Values: <ul style="list-style-type: none"> – 0x00 - System is not a forbidden system – 0x01 - System is a forbidden system

8.357.2 Field Documentation

8.357.2.1 `uint8_t nas_detailSvcInfo::hdrHybrid`

8.357.2.2 `uint8_t nas_detailSvcInfo::hdrSrvStatus`

8.357.2.3 `uint8_t nas_detailSvcInfo::isSysForbidden`

8.357.2.4 `uint8_t nas_detailSvcInfo::srvCapability`

8.357.2.5 `uint8_t nas_detailSvcInfo::srvStatus`

8.358 `nas_ecioListElement` Struct Reference

Data Fields

- `int16_t` [ecio](#)
- `uint8_t` [radioIf](#)

8.358.1 Detailed Description

This structure contains the ECIO Information

Parameters

<i>ecio</i>	<ul style="list-style-type: none"> • ECIO value in dBm
<i>radioIf</i>	<ul style="list-style-type: none"> • Radio interface technology of the signal being measured <ul style="list-style-type: none"> – 0x00 – RADIO_IF_NO_SVC – None (no service) – 0x01 – RADIO_IF_CDMA_1X – cdma2000® 1X – 0x02 – RADIO_IF_CDMA_1XEVD0 – cdma2000 HRPD (1xEV-DO) – 0x03 – RADIO_IF_AMPS – AMPS – 0x04 – RADIO_IF_GSM – GSM – 0x05 – RADIO_IF_UMTS – UMTS

8.358.2 Field Documentation

8.358.2.1 `int16_t nas_ecioListElement::ecio`

8.358.2.2 `uint8_t nas_ecioListElement::radioIf`

8.359 `nas_errorRateListElement` Struct Reference

Data Fields

- `uint16_t` [errorRate](#)
- `uint8_t` [radioIf](#)

8.359.1 Detailed Description

This structure contains the Error Rate Information

Parameters

<i>errorRate</i>	<ul style="list-style-type: none"> • Error rate value corresponds to the RAT that is currently registered. <ul style="list-style-type: none"> – For CDMA, the error rate reported is Frame Error Rate: <ul style="list-style-type: none"> * Valid error rate values between 1 and 10000 are returned to indicate percentage, e.g., a value of 300 means the error rate is 3% * A value of 0xFFFF indicates that the error rate is unknown or unavailable – For HDR, the error rate reported is Packet Error Rate: <ul style="list-style-type: none"> * Valid error rate values between 1 and 10000 are returned to indicate percentage, e.g., a value of 300 means the error rate is 3% * A value of 0xFFFF indicates that the error rate is unknown or unavailable – For GSM, the error rate reported is Bit Error Rate: <ul style="list-style-type: none"> * Valid values are 0, 100, 200, 300, 400, 500, 600, and 700 The reported value divided by 100 gives the error rate as an RxQual value, e.g., a value of 300 represents an RxQual value of 3. * A value of 25500 indicates No Data – For WCDMA, the error rate reported is Block Error Rate (BLER): <ul style="list-style-type: none"> * Valid values are 1 to 10000 * The reported value divided by 100 provides the error rate in percentages, e.g., a value of 300 represents a BLER of 3%. * A value of 0 indicates No Data
<i>radioIf</i>	<ul style="list-style-type: none"> • Radio interface technology of the signal being measured <ul style="list-style-type: none"> – 0x00 – RADIO_IF_NO_SVC – None (no service) – 0x01 – RADIO_IF_CDMA_1X – cdma2000@ 1X – 0x02 – RADIO_IF_CDMA_1xEVDO – cdma2000 HRPD (1xEV-DO) – 0x03 – RADIO_IF_AMPS – AMPS – 0x04 – RADIO_IF_GSM – GSM – 0x05 – RADIO_IF_UMTS – UMTS

8.359.2 Field Documentation

8.359.2.1 `uint16_t nas_errorRateListElement::errorRate`

8.359.2.2 `uint8_t nas_errorRateListElement::radioIf`

8.360 nas_GERANInfo Struct Reference

Data Fields

- `uint32_t cellID`
- `uint8_t plmn` [3]
- `uint16_t lac`
- `uint16_t arfcn`
- `uint8_t bsic`
- `uint32_t timingAdvance`
- `uint16_t rxLev`
- `uint8_t nmrInst`
- `nas_nmrCellInfo insNmrCellInfo` [255]

8.360.1 Detailed Description

This structure contains information about the GERAN Network.

Parameters

<i>cellID</i>	<ul style="list-style-type: none"> Cell ID. 0xFFFFFFFF indicates cell ID information is not present.
<i>plmn</i> [<i>NAS_PL</i> ↔ <i>MN_LENGTH</i>]	<ul style="list-style-type: none"> MCC/MNC information coded as octet 3, 4, and 5. This field is ignored when <i>nmrCellID</i> is not present.
<i>lac</i>	<ul style="list-style-type: none"> Location area code. This field is ignored when <i>nmrCellID</i> is not present. <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>arfcn</i>	<ul style="list-style-type: none"> Absolute RF channel number. <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>bsic</i>	<ul style="list-style-type: none"> Base station identity code. <ul style="list-style-type: none"> 0xFF - Not Available
<i>timingAdvance</i>	<ul style="list-style-type: none"> Measured delay (in bit periods; 1 bit period = 48/13 microsecond) of access burst transmission on RACH or PRACH to the expected signal from an MS at zero distance under static channel conditions. <ul style="list-style-type: none"> 0xFFFFFFFF - Not Available
<i>rxLev</i>	<ul style="list-style-type: none"> Serving Cell Rx measurement. Values range between 0 and 63. Mapped to a measured signal level: <ul style="list-style-type: none"> Rxlev 0 is a signal strength less than -110 dBm Rxlev 1 is -110 dBm to -109 dBm Rxlev 2 is -109 dBm to -108 dBm ... Rxlev 62 is -49 dBm to -48 dBm Rxlev 63 is greater than -48 dBm 0xFFFF - Not Available
<i>nmrInst</i>	<ul style="list-style-type: none"> Provides the number of set of instances which follow. If 0(zero), then no information follows it.
<i>insNmrCell</i> ↔ <i>Info</i> [<i>NAS_MA</i> ↔ <i>X_DESCRIPTOR</i> ↔ <i>ON_LENGTH</i>]	<ul style="list-style-type: none"> See nas_nmrCellInfo for more information.

8.360.2 Field Documentation

8.360.2.1 uint16_t nas_GERANInfo::arfcn

8.360.2.2 uint8_t nas_GERANInfo::bsic

8.360.2.3 uint32_t nas_GERANInfo::cellID

8.360.2.4 nas_nmrCellInfo nas_GERANInfo::insNmrCellInfo[255]

8.360.2.5 uint16_t nas_GERANInfo::lac

8.360.2.6 uint8_t nas_GERANInfo::nmrInst

8.360.2.7 uint8_t nas_GERANInfo::plmn[3]

8.360.2.8 uint16_t nas_GERANInfo::rxLev

8.360.2.9 uint32_t nas_GERANInfo::timingAdvance

8.361 nas_geranInstInfo Struct Reference

Data Fields

- uint16_t [geranArfcn](#)
- uint8_t [geranBsicNcc](#)
- uint8_t [geranBsicBcc](#)
- int16_t [geranRssi](#)

8.361.1 Detailed Description

This structure contains information about the GERAN Instances in UMTS Network.

Parameters

<i>geranArfcn</i>	<ul style="list-style-type: none"> • Absolute RF channel number.
<i>geranBsicNcc</i>	<ul style="list-style-type: none"> • Base station identity code network color code. • 0xFF indicates information is not present.
<i>geranBsicBcc</i>	<ul style="list-style-type: none"> • Base station identity code base station color code. • 0xFF indicates information is not present.
<i>geranRssi</i>	<ul style="list-style-type: none"> • Received signal strength indicator.
Generated by Doxygen	

8.361.2 Field Documentation

8.361.2.1 uint16_t nas_geranInstInfo::geranArfcn

8.361.2.2 uint8_t nas_geranInstInfo::geranBsicBcc

8.361.2.3 uint8_t nas_geranInstInfo::geranBsicNcc

8.361.2.4 int16_t nas_geranInstInfo::geranRssi

8.362 nas_gsmCellInfo Struct Reference

Data Fields

- uint16_t [arfcn](#)
- uint8_t [band1900](#)
- uint8_t [cellIdValid](#)
- uint8_t [bsicId](#)
- int16_t [rssi](#)
- int16_t [srxlev](#)

8.362.1 Detailed Description

This structure contains information about the GSM Cell.

Parameters

<i>arfcn</i>	<ul style="list-style-type: none"> • GSM frequency being reported. • Range: 0 to 1023.
<i>band1900</i>	<ul style="list-style-type: none"> • Band indicator for the GSM ARFCN • This field is only valid if arfcn is in the overlapping region. • If TRUE and the cell is in the overlapping region, the ARFCN is on the 1900 band. • If FALSE, it is on the 1800 band.
<i>cellIdValid</i>	<ul style="list-style-type: none"> • Flag indicating whether the base station identity code ID is valid.
<i>bsicId</i>	<ul style="list-style-type: none"> • Base station identity code ID, including base station color code and network color code. • The lower 6 bits can be set to any value.
<i>rssi</i>	<ul style="list-style-type: none"> • Measured RSSI value in 1/10 dB. • Range: -200.0 dB to 0
<i>srxlev</i>	<ul style="list-style-type: none"> • Cell selection Rx level (Srxlev) value.
	<ul style="list-style-type: none"> • Range: -128 to 128. • This field is only valid when ue_in_idle is TRUE.

8.362.2 Field Documentation

8.362.2.1 uint16_t nas_gsmCellInfo::arfcn

8.362.2.2 uint8_t nas_gsmCellInfo::band1900

8.362.2.3 uint8_t nas_gsmCellInfo::bsicld

8.362.2.4 uint8_t nas_gsmCellInfo::cellIdValid

8.362.2.5 int16_t nas_gsmCellInfo::rssi

8.362.2.6 int16_t nas_gsmCellInfo::srxlev

8.363 nas_GSMRSSIThresh Struct Reference

Data Fields

- uint8_t [GSMRSSIThreshListLen](#)
- int16_t * [pGSMRSSIThreshList](#)

8.363.1 Detailed Description

This structure contains GSM RSSI threshold related parameters.

Parameters

<i>GSMRSSI</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> • Length of the GSM RSSI threshold list parameter to follow
<i>pGSMRSSI</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> • Array of RSSI thresholds (in units of 0.1 dBm) • Maximum of 32 values • Range for RSSI values: -111 to -48 (in dBm)

8.363.2 Field Documentation

8.363.2.1 uint8_t nas_GSMRSSIThresh::GSMRSSIThreshListLen

8.363.2.2 int16_t* nas_GSMRSSIThresh::pGSMRSSIThreshList

8.364 nas_GSMSrvStatusInfo Struct Reference

Data Fields

- uint8_t [srvStatus](#)

- [uint8_t trueSrvStatus](#)
- [uint8_t isPrefDataPath](#)

8.364.1 Detailed Description

Structure for storing the service status information for GSM, WCDMA and LTE networks.

Parameters

<i>srvStatus</i>	<ul style="list-style-type: none"> • Service status of the system. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save – 0xFF - Not Available
<i>trueSrvStatus</i>	<ul style="list-style-type: none"> • True service status of the system. • Not applicable to CDMA/HDR. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save – 0xFF - Not Available
<i>isPrefDataPath</i>	<ul style="list-style-type: none"> • Whether the RAT is the preferred data path. <ul style="list-style-type: none"> – 0x00 - Not preferred – 0x01 - Preferred – 0xFF - Not Available

8.364.2 Field Documentation

8.364.2.1 [uint8_t nas_GSMsSrvStatusInfo::isPrefDataPath](#)

8.364.2.2 [uint8_t nas_GSMsSrvStatusInfo::srvStatus](#)

8.364.2.3 [uint8_t nas_GSMsSrvStatusInfo::trueSrvStatus](#)

8.365 nas_GSMsSysInfo Struct Reference

Data Fields

- [nas_sysInfoCommon sysInfoGSM](#)

- uint8_t [lacValid](#)
- uint16_t [lac](#)
- uint8_t [cellIdValid](#)
- uint32_t [cellId](#)
- uint8_t [regRejectInfoValid](#)
- uint8_t [rejectSrvDomain](#)
- uint8_t [rejCause](#)
- uint8_t [networkIdValid](#)
- uint8_t [MCC](#) [3]
- uint8_t [MNC](#) [3]
- uint8_t [egprsSuppValid](#)
- uint8_t [egprsSupp](#)
- uint8_t [dtmSuppValid](#)
- uint8_t [dtmSupp](#)

8.365.1 Detailed Description

Structure for storing the GSM System Information.

Parameters

<i>sysInfoGSM</i>	<ul style="list-style-type: none"> • See sysInfoCommon for more information.
<i>lacValid</i>	<ul style="list-style-type: none"> • Indicates whether the location area code is valid.. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>lac</i>	<ul style="list-style-type: none"> • Location area code. • Only applies to 3GPP. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>cellIdValid</i>	<ul style="list-style-type: none"> • Indicates whether the cell ID is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>cellId</i>	<ul style="list-style-type: none"> • Cell ID. <ul style="list-style-type: none"> – 0xFFFFFFFF - Not Available
<i>regRejectInfoValid</i>	<ul style="list-style-type: none"> • Indicates whether the registration reject information is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

<i>rejectSrvDomain</i>	<ul style="list-style-type: none"> • Type of service domain in which the registration is rejected. <ul style="list-style-type: none"> – 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped – 0xFF - Not Available
<i>rejCause</i>	<ul style="list-style-type: none"> • Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6]. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>networkIdValid</i>	<ul style="list-style-type: none"> • Indicates whether the network ID is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>MCC[PLMN_L↔ENGTH]</i>	<ul style="list-style-type: none"> • Mobile Country Code. • MCC digits in ASCII characters
<i>MNC[PLMN_L↔ENGTH]</i>	<ul style="list-style-type: none"> • Mobile Network Code. • MNC digits in ASCII characters • An unused byte is set to 0xFF. • In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.
<i>egprsSuppValid</i>	<ul style="list-style-type: none"> • Indicates whether the EGPRS support is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>egprsSupp</i>	<ul style="list-style-type: none"> • EGPRS support indication. • Only applicable for GSM. <ul style="list-style-type: none"> – 0x00 - Not available – 0x01 - Available – 0xFF - Not Available
<i>dtmSuppValid</i>	<ul style="list-style-type: none"> • Indicates whether Dual Transfer mode support is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

<i>dtmSupp</i>	<ul style="list-style-type: none"> • Dual Transfer mode support indication. • Only applicable for GSM. <ul style="list-style-type: none"> – 0x00 - Not available – 0x01 - Available – 0xFF - Not Available
----------------	--

8.365.2 Field Documentation

8.365.2.1 `uint32_t nas_GSM.sysInfo::cellId`

8.365.2.2 `uint8_t nas_GSM.sysInfo::cellIdValid`

8.365.2.3 `uint8_t nas_GSM.sysInfo::dtmSupp`

8.365.2.4 `uint8_t nas_GSM.sysInfo::dtmSuppValid`

8.365.2.5 `uint8_t nas_GSM.sysInfo::egprsSupp`

8.365.2.6 `uint8_t nas_GSM.sysInfo::egprsSuppValid`

8.365.2.7 `uint16_t nas_GSM.sysInfo::lac`

8.365.2.8 `uint8_t nas_GSM.sysInfo::lacValid`

8.365.2.9 `uint8_t nas_GSM.sysInfo::MCC[3]`

8.365.2.10 `uint8_t nas_GSM.sysInfo::MNC[3]`

8.365.2.11 `uint8_t nas_GSM.sysInfo::networkIdValid`

8.365.2.12 `uint8_t nas_GSM.sysInfo::regRejectInfoValid`

8.365.2.13 `uint8_t nas_GSM.sysInfo::rejCause`

8.365.2.14 `uint8_t nas_GSM.sysInfo::rejectSrvDomain`

8.365.2.15 `nas_sysInfoCommon nas_GSM.sysInfo::sysInfoGSM`

8.366 nas_HDRECIOThresh Struct Reference

Data Fields

- `uint8_t HDRECIOThreshListLen`
- `int16_t * pHRECIOThreshList`

8.366.1 Detailed Description

This structure contains HDR ECIO threshold related parameters.

Parameters

<i>HDRECIO</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> Length of the HDR ECIO threshold list parameter to follow
<i>pHDRECIO</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> Array of ECIO thresholds (in units of 0.1 dB) Maximum of 32 values Range for ECIO values: -31.5 to 0 (in dB).

8.366.2 Field Documentation

8.366.2.1 `uint8_t nas_HDRECIOThresh::HDRECIOThreshListLen`

8.366.2.2 `int16_t* nas_HDRECIOThresh::pHDRECIOThreshList`

8.367 nas_HDRIOThresh Struct Reference

Data Fields

- `uint8_t` [HDRIOThreshListLen](#)
- `int16_t *` [pHDRIOThreshList](#)

8.367.1 Detailed Description

This structure contains HDR IO threshold related parameters.

Parameters

<i>HDRIOThresh</i> ↔ <i>ListLen</i>	<ul style="list-style-type: none"> Length of the HDR IO threshold list parameter to follow
<i>pHDRIOThresh</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> Array of IO thresholds (in units of 0.1 dBm) Maximum of 32 values Range for IO values: -128 to -13 (in dBm).

8.367.2 Field Documentation

8.367.2.1 `uint8_t nas_HDRIOThresh::HDRIOThreshListLen`

8.367.2.2 `int16_t* nas_HDRIOThresh::pHDRIOThreshList`

8.368 nas_HDRRSSIThresh Struct Reference

Data Fields

- uint8_t [HDRRSSIThreshListLen](#)
- int16_t * [pHDRRSSIThreshList](#)

8.368.1 Detailed Description

This structure contains HDR RSSI threshold related parameters.

Parameters

<i>HDRRSSI↔ ThreshListLen</i>	<ul style="list-style-type: none">• Length of the HDR RSSI threshold list parameter to follow
<i>pHDRRSSI↔ ThreshList</i>	<ul style="list-style-type: none">• Array of RSSI thresholds (in units of 0.1 dBm)• Maximum of 32 values.• Range for RSSI values: -118 to -13 (in dBm).

8.368.2 Field Documentation

8.368.2.1 uint8_t nas_HDRRSSIThresh::HDRRSSIThreshListLen

8.368.2.2 int16_t* nas_HDRRSSIThresh::pHDRRSSIThreshList

8.369 nas_HDRSINRThreshold Struct Reference

Data Fields

- uint8_t [HDRSINRThreshListLen](#)
- uint16_t * [pHDRSINRThreshList](#)

8.369.1 Detailed Description

This structure contains HDR SINR threshold related parameters.

Parameters

<i>HDRSINR↔ ThreshListLen</i>	<ul style="list-style-type: none">• Length of the HDR ECIO threshold list parameter to follow
-----------------------------------	---

<p><i>pHDRSINR</i>↔ <i>ThreshList</i></p>	<ul style="list-style-type: none"> • Array of SINR level thresholds (in units of 1) • maximum of 32 values. • Valid levels are 0 to 8 <ul style="list-style-type: none"> – 0x00 - SINR_LEVEL_0 is -9 dB – 0x01 - SINR_LEVEL_1 is -6 dB – 0x02 - SINR_LEVEL_2 is -4.5 dB – 0x03 - SINR_LEVEL_3 is -3 dB – 0x04 - SINR_LEVEL_4 is -2 dB – 0x05 - SINR_LEVEL_5 is +1 dB – 0x06 - SINR_LEVEL_6 is +3 dB – 0x07 - SINR_LEVEL_7 is +6 dB – 0x08 - SINR_LEVEL_8 is +9 dB
---	--

8.369.2 Field Documentation

8.369.2.1 `uint8_t nas_HDRSINRThreshold::HDRSINRThreshListLen`

8.369.2.2 `uint16_t* nas_HDRSINRThreshold::pHDRSINRThreshList`

8.370 nas_HDRSysInfo Struct Reference

Data Fields

- [nas_sysInfoCommon sysInfoHDR](#)
- `uint8_t isSysPrIMatchValid`
- `uint8_t isSysPrIMatch`
- `uint8_t hdrPersonalityValid`
- `uint8_t hdrPersonality`
- `uint8_t hdrActiveProtValid`
- `uint8_t hdrActiveProt`
- `uint8_t is856SysIdValid`
- `uint8_t is856SysId [16]`

8.370.1 Detailed Description

Structure for storing the HDR System Information.

Parameters

<i>sysInfoHDR</i>	<ul style="list-style-type: none"> • See sysInfoCommon for more information.
<i>isSysPrIMatch</i> ↔ <i>Valid</i>	<ul style="list-style-type: none"> • Indicates whether the system PRL match is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>isSysPrIMatch</i>	<ul style="list-style-type: none"> • Indicates whether the system is in a PRL. • Only applies to CDMA/HDR. <ul style="list-style-type: none"> – 0x00 - System is not in a PRL – 0x01 - System is in a PRL – 0xFF - Not Available • If the system is not in a PRL, roam_status carries the value from the default roaming indicator in the PRL. • If the system is in a PRL, roam_status is set to the value based on the standard specification.
<i>hdrPersonality</i> ↔ <i>Valid</i>	<ul style="list-style-type: none"> • Indicates whether the HDR personality is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>hdrPersonality</i>	<ul style="list-style-type: none"> • HDR personality information. • Only applicable for HDR. <ul style="list-style-type: none"> – 0x00 - None – 0x02 - HRPD – 0x03 - eHRPD – 0xFF - Not Available
<i>hdrActiveProt</i> ↔ <i>Valid</i>	<ul style="list-style-type: none"> • Indicates whether the HDR active protocol revision information is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>hdrActiveProt</i>	<ul style="list-style-type: none"> • HDR active protocol revision information . • Only applicable for HDR. <ul style="list-style-type: none"> – 0x00 - None – 0x02 - HDR Rel 0 – 0x03 - HDR Rel A – 0x04 - HDR Rel B – 0xFF - Not Available

<i>is856SysIdValid</i>	<ul style="list-style-type: none"> Indicates whether the IS-856 system ID is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>is856SysId[SL↔ QS_SYSTEM↔ _ID_SIZE]</i>	<ul style="list-style-type: none"> IS-856 system ID. Only applicable for HDR.

8.370.2 Field Documentation

8.370.2.1 `uint8_t nas_HDRSysInfo::hdrActiveProt`

8.370.2.2 `uint8_t nas_HDRSysInfo::hdrActiveProtValid`

8.370.2.3 `uint8_t nas_HDRSysInfo::hdrPersonality`

8.370.2.4 `uint8_t nas_HDRSysInfo::hdrPersonalityValid`

8.370.2.5 `uint8_t nas_HDRSysInfo::is856SysId[16]`

8.370.2.6 `uint8_t nas_HDRSysInfo::is856SysIdValid`

8.370.2.7 `uint8_t nas_HDRSysInfo::isSysPrMatch`

8.370.2.8 `uint8_t nas_HDRSysInfo::isSysPrMatchValid`

8.370.2.9 `nas_sysInfoCommon nas_HDRSysInfo::sysInfoHDR`

8.371 nas_infoInterFreq Struct Reference

Data Fields

- `uint16_t earfcn`
- `uint8_t threshXLow`
- `uint8_t threshXHigh`
- `uint8_t cell_resel_priority`
- `uint8_t cells_len`
- `nas_cellParams cellInterFreqParams [255]`

8.371.1 Detailed Description

This structure contains information about the inter-frequency.

Parameters

<i>earfcn</i>	<ul style="list-style-type: none"> E-UTRA absolute radio frequency channel number of the serving cell. Range: 0 to 65535.
<i>threshXLow</i>	<ul style="list-style-type: none"> Cell Srxlev low threshold. Range: 0 to 31. When the serving cell does not exceed thresh_serving_low, the value of an evaluated cell must be smaller than this value to be considered for re-selection.
<i>threshXHigh</i>	<ul style="list-style-type: none"> Cell Srxlev high threshold. Range: 0 to 31. When the serving cell exceeds thresh_serving_low, the value of an evaluated cell must be greater than this value to be considered for re-selection.
<i>cell_resel_↔ priority</i>	<ul style="list-style-type: none"> Cell re-selection priority Range: 0 to 7. This field is only valid when ue_in_idle is TRUE.
<i>cells_len</i>	<ul style="list-style-type: none"> Provides the number of set of cell params.
<i>cellInterFreq↔ Params[NAS ↔ MAX_DESCR↔ PTION LENG↔ TH]</i>	<ul style="list-style-type: none"> See nas_cellParams for more information.

8.371.2 Field Documentation

8.371.2.1 uint8_t nas_infoInterFreq::cell_resel_priority

8.371.2.2 nas_cellParams nas_infoInterFreq::cellInterFreqParams[255]

8.371.2.3 uint8_t nas_infoInterFreq::cells_len

8.371.2.4 uint16_t nas_infoInterFreq::earfcn

8.371.2.5 uint8_t nas_infoInterFreq::threshXHigh

8.371.2.6 uint8_t nas_infoInterFreq::threshXLow

8.372 nas_lteGsmCellInfo Struct Reference

Data Fields

- uint8_t [cellReselPriority](#)
- uint8_t [threshGsmHigh](#)
- uint8_t [threshGsmLow](#)
- uint8_t [nccPermitted](#)
- uint8_t [cells_len](#)
- [nas_gsmCellInfo](#) [GsmCellInfo](#) [255]

8.372.1 Detailed Description

This structure contains information about the LTE GSM Cell.

Parameters

<i>cellReselPriority</i>	<ul style="list-style-type: none"> • Priority of this frequency group. • Range: 0 to 7. • This field is only valid when <code>ue_in_idle</code> is TRUE.
<i>threshGsmHigh</i>	<ul style="list-style-type: none"> • Reselection threshold for high priority layers. • Range: 0 to 31. • This field is only valid when <code>ue_in_idle</code> is TRUE.
<i>threshGsmLow</i>	<ul style="list-style-type: none"> • Reselection threshold for low priority layers. • Range: 0 to 31. • This field is only valid when <code>ue_in_idle</code> is TRUE.
<i>nccPermitted</i>	<ul style="list-style-type: none"> • Bitmask specifying whether a neighbor with a specific network color code is to be reported. • Range: 0 to 255. • Bit <code>n</code> set to 1 means a neighbor with NCC <code>n</code> must be included in the report. This flag is synonymous with a blacklist in other RATs. • This field is only valid when <code>ue_in_idle</code> is TRUE.
<i>cells_len</i>	<ul style="list-style-type: none"> • Provides the number of set of gsm cells.
<i>GsmCellInfo</i> [<i>N</i> ↔ <i>AS_MAX_DE</i> ↔ <i>SCRIPTION_L</i> ↔ <i>ENGTH</i>]	<ul style="list-style-type: none"> • See nas_gsmCellInfo for more information.

8.372.2 Field Documentation

8.372.2.1 `uint8_t nas_lteGsmCellInfo::cellReselPriority`

8.372.2.2 `uint8_t nas_lteGsmCellInfo::cells_len`

8.372.2.3 `nas_gsmCellInfo nas_lteGsmCellInfo::GsmCellInfo[255]`

8.372.2.4 `uint8_t nas_lteGsmCellInfo::nccPermitted`

8.372.2.5 `uint8_t nas_lteGsmCellInfo::threshGsmHigh`

8.372.2.6 `uint8_t nas_lteGsmCellInfo::threshGsmLow`

8.373 nas_LTEInfo Struct Reference

Data Fields

- uint8_t [band](#)
- uint8_t [bandwidth](#)
- uint16_t [RXChan](#)
- uint16_t [TXChan](#)
- uint8_t [emmState](#)
- uint8_t [emmSubState](#)
- uint8_t [emmConnState](#)

8.373.1 Detailed Description

Structure for storing the LTE information for the device.

Parameters

<i>band</i>	<ul style="list-style-type: none"> • LTE Band <ul style="list-style-type: none"> – 1 ~ 40 (Band in decimal) – 0xFF - Invalid
<i>bandwidth</i>	<ul style="list-style-type: none"> • BandWidth. <ul style="list-style-type: none"> – 0x00 - 1.4 MHz – 0x01 - 3 MHz – 0x02 - 5 MHz – 0x03 - 10 MHz – 0x04 - 15 MHz – 0x05 - 20 MHz – 0x06 - Invalid – 0xFF - Unknown
<i>RXChan</i>	<ul style="list-style-type: none"> • RX channel number in decimal <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>TXChan</i>	<ul style="list-style-type: none"> • TX channel number in decimal <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>emmState</i>	<ul style="list-style-type: none"> • EMM State. <ul style="list-style-type: none"> – 0x00 - Deregistered – 0x01 - Reg Initiated – 0x02 - Registered – 0x03 - TAU Initiated – 0x04 - SR Initiated – 0x05 - Dereg Initiated – 0x06 - Invalid – 0xFF - Unknown
Generated by Doxygen	

<i>emmSubState</i>	<ul style="list-style-type: none"> • EMM Sub State. <ul style="list-style-type: none"> – 0xFF - NOT Applicable • When EMM_state is 0x00: <ul style="list-style-type: none"> – 0x00 - No IMSI – 0x01 - PLMN Search – 0x02 - Attach Needed – 0x03 - No Cell – 0x04 - Attaching – 0x05 - Normal Service – 0x06 - Limited Service – 0x07 - Waiting for PDN • When EMM_state is 0x01: <ul style="list-style-type: none"> – 0x00 - Waiting for NW – 0x01 - Waiting for ESM • When EMM_state is 0x02: <ul style="list-style-type: none"> – 0x00 - Normal Service – 0x01 - Update Needed – 0x02 - Attempt Update – 0x03 - No Cell – 0x04 - PLMN Search – 0x05 - Limited Service – 0x06 - MM Update – 0x07 - IMSI Detach – 0x08 - Waiting for ESM
<i>emmConnState</i>	<ul style="list-style-type: none"> • EMM Connected Mode State. <ul style="list-style-type: none"> – 0x00 - RRC Idle – 0x01 - Waiting RRC Cfm – 0x02 - RRC Connected – 0x03 - RRC Releasing – 0xFF - Unknown

8.373.2 Field Documentation

8.373.2.1 uint8_t nas_LTEInfo::band

8.373.2.2 uint8_t nas_LTEInfo::bandwidth

8.373.2.3 uint8_t nas_LTEInfo::emmConnState

8.373.2.4 uint8_t nas_LTEInfo::emmState

8.373.2.5 uint8_t nas_LTEInfo::emmSubState

8.373.2.6 uint16_t nas_LTEInfo::RXChan

8.373.2.7 uint16_t nas_LTEInfo::TXChan

8.374 nas_LTEInfoInterfreq Struct Reference

Data Fields

- uint8_t [ueIdle](#)
- uint8_t [freqsLen](#)
- [nas_infoInterFreq](#) [InfoInterfreq](#) [255]

8.374.1 Detailed Description

This structure contains information about the LTE Inter-Frequency Network.

Parameters

<i>ueIdle</i>	<ul style="list-style-type: none"> • TRUE if the UE is in Idle mode, otherwise FALSE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>freqsLen</i>	<ul style="list-style-type: none"> • Provides the number of set of inter frequency information. • If 0(zero), then no information follows it.
<i>InfoInterfreq[N↔ AS_MAX_DE↔ SCRIPTION_L↔ ENGTH]</i>	<ul style="list-style-type: none"> • See nas_infoInterFreq for more information.

8.374.2 Field Documentation

8.374.2.1 uint8_t nas_LTEInfoInterfreq::freqsLen

8.374.2.2 nas_infoInterFreq nas_LTEInfoInterfreq::InfoInterfreq[255]

8.374.2.3 uint8_t nas_LTEInfoInterfreq::ueIdle

8.375 nas_LTEInfoIntrafreq Struct Reference

Data Fields

- uint8_t [ueIdle](#)
- uint8_t [plmn](#) [3]
- uint16_t [tac](#)
- uint32_t [globalCellId](#)
- uint16_t [earfcn](#)
- uint16_t [servingCellId](#)

- uint8_t [cellReselPriority](#)
- uint8_t [sNonIntraSearch](#)
- uint8_t [threshServingLow](#)
- uint8_t [sIntraSearch](#)
- uint8_t [cellsLen](#)
- [nas_cellParams](#) [CellParams](#) [255]

8.375.1 Detailed Description

This structure contains information about the LTE Intra-Frequency Network.

Parameters

<i>ueInIdle</i>	<ul style="list-style-type: none"> • TRUE if the UE is in Idle mode, otherwise FALSE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>plmn[NAS_PLMN_LENGTH]</i>	<ul style="list-style-type: none"> • PLMN ID coded as octet 3, 4, and 5.
<i>tac</i>	<ul style="list-style-type: none"> • Tracking area code. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>globalCellId</i>	<ul style="list-style-type: none"> • Global cell ID in the system information block. <ul style="list-style-type: none"> – 0xFFFFFFFF - Not Available
<i>earfcn</i>	<ul style="list-style-type: none"> • E-UTRA absolute radio frequency channel number of the serving cell. • Range: 0 to 65535. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>servingCellId</i>	<ul style="list-style-type: none"> • LTE serving cell ID. • Range: 0 to 503. • This is the cell ID of the serving cell and can be found in the cell list. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>cellReselPriority</i>	<ul style="list-style-type: none"> • Priority for serving frequency. • Range: 0 to 7. • This field is only valid when <i>ue_in_idle</i> is TRUE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>sNonIntraSearch</i>	<ul style="list-style-type: none"> • S non-intra search threshold to control non-intrafrequency searches. • Range: 0 to 31. • This field is only valid when <i>ue_in_idle</i> is TRUE. <ul style="list-style-type: none"> – 0xFF - Not Available

<i>threshServingLow</i>	<ul style="list-style-type: none"> Serving cell low threshold. Range: 0 to 31. This field is only valid when ue_in_idle is TRUE. <ul style="list-style-type: none"> 0xFF - Not Available
<i>sIntraSearch</i>	<ul style="list-style-type: none"> S Intra search threshold. Range: 0 to 31. The current cell measurement must fall below this threshold to consider intrafrequency for reselection. This field is only valid when ue_in_idle is TRUE. <ul style="list-style-type: none"> 0xFF - Not Available
<i>cellsLen</i>	<ul style="list-style-type: none"> Provides the number of set of cell params. If 0(zero), then no information follows it.
<i>CellParams[NAS_MAX_DESCRIPTION_LENGTH]</i>	<ul style="list-style-type: none"> See nas_cellParams for more information.

8.375.2 Field Documentation

8.375.2.1 `nas_cellParams nas_LTEInfoIntrafreq::CellParams[255]`

8.375.2.2 `uint8_t nas_LTEInfoIntrafreq::cellReselPriority`

8.375.2.3 `uint8_t nas_LTEInfoIntrafreq::cellsLen`

8.375.2.4 `uint16_t nas_LTEInfoIntrafreq::earfcn`

8.375.2.5 `uint32_t nas_LTEInfoIntrafreq::globalCellId`

8.375.2.6 `uint8_t nas_LTEInfoIntrafreq::plmn[3]`

8.375.2.7 `uint16_t nas_LTEInfoIntrafreq::servingCellId`

8.375.2.8 `uint8_t nas_LTEInfoIntrafreq::sIntraSearch`

8.375.2.9 `uint8_t nas_LTEInfoIntrafreq::sNonIntraSearch`

8.375.2.10 `uint16_t nas_LTEInfoIntrafreq::tac`

8.375.2.11 `uint8_t nas_LTEInfoIntrafreq::threshServingLow`

8.375.2.12 `uint8_t nas_LTEInfoIntrafreq::ueInIdle`

8.376 nas_LTEInfoNeighboringGSM Struct Reference

Data Fields

- `uint8_t ueInIdle`
- `uint8_t freqsLen`
- `nas_lteGsmCellInfo lteGsmCellInfo` [255]

8.376.1 Detailed Description

This structure contains information about the LTE Neighboring GSM Network.

Parameters

<i>ueInIdle</i>	<ul style="list-style-type: none"> • TRUE if the UE is in Idle mode, otherwise FALSE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>freqsLen</i>	<ul style="list-style-type: none"> • Provides the number of set of LTE GSM cell information. • If 0(zero), then no information follows it.
<i>LteGsmCellInfo[NAS_MAX_DESCRIPTOR_LENGTH]</i>	<ul style="list-style-type: none"> • See nas_lteGsmCellInfo for more information.

8.376.2 Field Documentation

8.376.2.1 `uint8_t nas_LTEInfoNeighboringGSM::freqsLen`

8.376.2.2 `nas_lteGsmCellInfo nas_LTEInfoNeighboringGSM::lteGsmCellInfo`[255]

8.376.2.3 `uint8_t nas_LTEInfoNeighboringGSM::ueInIdle`

8.377 nas_LTEInfoNeighboringWCDMA Struct Reference

Data Fields

- `uint8_t ueInIdle`
- `uint8_t freqsLen`
- `nas_lteWcdmaCellInfo lteWCDMACellInfo` [255]

8.377.1 Detailed Description

This structure contains information about the LTE Neighboring WCDMA Network.

Parameters

<i>ueIdle</i>	<ul style="list-style-type: none"> • TRUE if the UE is in Idle mode, otherwise FALSE. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>freqsLen</i>	<ul style="list-style-type: none"> • Provides the number of set of LTE WCDMA cell information. • If 0(zero), then no information follows it.
<i>LTEWCDMACellInfo[NAS_MAX_DESCRIPTOR_LENGTH]</i>	<ul style="list-style-type: none"> • See nas_IteWcdmaCellInfo for more information.

8.377.2 Field Documentation

8.377.2.1 `uint8_t nas_LTEInfoNeighboringWCDMA::freqsLen`

8.377.2.2 `nas_IteWcdmaCellInfo nas_LTEInfoNeighboringWCDMA::LTEWCDMACellInfo[255]`

8.377.2.3 `uint8_t nas_LTEInfoNeighboringWCDMA::ueIdle`

8.378 nas_IteRsrpinformation Struct Reference

Data Fields

- `int16_t rsrplevel`

8.378.1 Detailed Description

This structure contains the LTE RSRP Information

Parameters

<i>rsrplevel</i>	<ul style="list-style-type: none"> • LTE RSRP in dBm as a measured by L1. Range: -44 to -140(-44 means -44dBm, -140 means -140dBm).
------------------	--

8.378.2 Field Documentation

8.378.2.1 int16_t nas_lteRsrpInformation::rsrplevel

8.379 nas_LTERSRPThresh Struct Reference

Data Fields

- uint8_t [LTERSRPThreshListLen](#)
- int16_t * [pLTERSRPThreshList](#)

8.379.1 Detailed Description

This structure contains LTE RSRP threshold related parameters.

Parameters

<i>LTERSRP</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> • Length of the LTE RSRP threshold list parameter to follow
<i>pLTERSRP</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> • Array of RSRP thresholds (in units of 0.1 dBm) • Maximum of 32 values • Range for RSRP values: -140 to -44 (in dBm).

8.379.2 Field Documentation

8.379.2.1 uint8_t nas_LTERSRPThresh::LTERSRPThreshListLen

8.379.2.2 int16_t* nas_LTERSRPThresh::pLTERSRPThreshList

8.380 nas_LTERSRQThresh Struct Reference

Data Fields

- uint8_t [LTERSRQThreshListLen](#)
- int16_t * [pLTERSRQThreshList](#)

8.380.1 Detailed Description

This structure contains LTE RSRQ threshold related parameters.

Parameters

<i>LTERSRQ</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> • Length of the LTE RSRQ threshold list parameter to follow
<i>pLTERSRQ</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> • Array of RSRQ thresholds (in units of 0.1 dBm) • Maximum of 32 values. • Range for RSRQ values: -20 to -3 (in dBm)

8.380.2 Field Documentation

8.380.2.1 `uint8_t nas_LTERSRQThresh::LTERSRQThreshListLen`

8.380.2.2 `int16_t* nas_LTERSRQThresh::pLTERSRQThreshList`

8.381 nas_LTERSSIThresh Struct Reference

Data Fields

- `uint8_t LTERSSIThreshListLen`
- `int16_t * pLTERSSIThreshList`

8.381.1 Detailed Description

This structure contains LTE RSSI threshold related parameters.

Parameters

<i>LTERSSI↔ ThreshListLen</i>	<ul style="list-style-type: none">• Length of the LTE RSSI threshold list parameter to follow
<i>pLTERSSI↔ ThreshList</i>	<ul style="list-style-type: none">• Array of RSSI thresholds (in units of 0.1 dBm)• Maximum of 32 values.• Range for RSSI values: -120 to 0 (in dBm)

8.381.2 Field Documentation

8.381.2.1 `uint8_t nas_LTERSSIThresh::LTERSSIThreshListLen`

8.381.2.2 `int16_t* nas_LTERSSIThresh::pLTERSSIThreshList`

8.382 nas_LTESigRptConfig Struct Reference

Data Fields

- `uint8_t rptRate`
- `uint8_t avgPeriod`

8.382.1 Detailed Description

This structure contains LTE RSRP threshold related parameters.

Parameters

<i>rptRate</i>	<ul style="list-style-type: none"> • Rate on how often the LTE signal must be checked for reporting Values • 0 - Report using the default configuration • 1 - Report every 1 sec • 2 - Report every 2 sec • 3 - Report every 3 sec • 4 - Report every 4 sec • 5 - Report every 5 sec
<i>avgPeriod</i>	<ul style="list-style-type: none"> • Averaging period to be used for the LTE signal. • Values <ul style="list-style-type: none"> – 0 - Average using the default configuration – 1 - Average over 1 sec – 2 - Average over 2 sec – 3 - Average over 3 sec – 4 - Average over 4 sec – 5 - Average over 5 sec – 6 - Average over 6 sec – 7 - Average over 7 sec – 8 - Average over 8 sec – 9 - Average over 9 sec – 10 - Average over 10 sec

8.382.2 Field Documentation

8.382.2.1 uint8_t nas_LTESigRptConfig::avgPeriod

8.382.2.2 uint8_t nas_LTESigRptConfig::rptRate

8.383 nas_IteSnrinformation Struct Reference

Data Fields

- int16_t [snrlevel](#)

8.383.1 Detailed Description

This structure contains the LTE SNR Information

Parameters

<i>snrlevel</i>	<ul style="list-style-type: none"> • LTE SNR level as a scaled integer in units of 0.1dB e.g. -16dB has a value of -160 and 24.6dB has value of 246.
-----------------	---

8.383.2 Field Documentation

8.383.2.1 int16_t nas_lteSnrinformation::snrlevel

8.384 nas_LTESNRThreshold Struct Reference

Data Fields

- uint8_t [LTESNRThreshListLen](#)
- int16_t * [pLTESNRThreshList](#)

8.384.1 Detailed Description

This structure contains LTE SNR threshold related parameters.

Parameters

<i>LTESNR_↔</i> <i>ThreshListLen</i>	<ul style="list-style-type: none"> • Length of the LTE SNR threshold list parameter to follow
<i>pLTESNR_↔</i> <i>ThreshList</i>	<ul style="list-style-type: none"> • Array of SNR thresholds (in units of 0.1 dB) • Maximum of 32 values • Range for SNR values: -20 to 30 (in dB).

8.384.2 Field Documentation

8.384.2.1 uint8_t nas_LTESNRThreshold::LTESNRThreshListLen

8.384.2.2 int16_t* nas_LTESNRThreshold::pLTESNRThreshList

8.385 nas_LTESysInfo Struct Reference

Data Fields

- [nas_sysInfoCommon](#) sysInfoLTE
- uint8_t [lacValid](#)
- uint16_t [lac](#)
- uint8_t [cellIdValid](#)
- uint32_t [cellId](#)
- uint8_t [regRejectInfoValid](#)
- uint8_t [rejectSrvDomain](#)
- uint8_t [rejCause](#)
- uint8_t [networkIdValid](#)
- uint8_t [MCC](#) [3]
- uint8_t [MNC](#) [3]
- uint8_t [tacValid](#)
- uint16_t [tac](#)

8.385.1 Detailed Description

Structure for storing the LTE System Information.

Parameters

<i>sysInfoLTE</i>	<ul style="list-style-type: none"> See sysInfoCommon for more information.
<i>lacValid</i>	<ul style="list-style-type: none"> Indicates whether the location area code is valid.. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>lac</i>	<ul style="list-style-type: none"> Location area code. Only applies to 3GPP. <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>cellIdValid</i>	<ul style="list-style-type: none"> Indicates whether the cell ID is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>cellId</i>	<ul style="list-style-type: none"> Cell ID. <ul style="list-style-type: none"> 0xFFFFFFFF - Not Available
<i>regRejectInfoValid</i>	<ul style="list-style-type: none"> Indicates whether the registration reject information is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>rejectSrvDomain</i>	<ul style="list-style-type: none"> Type of service domain in which the registration is rejected. <ul style="list-style-type: none"> 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service 0x01 - Circuit-switched only 0x02 - Packet-switched only 0x03 - Circuit-switched and packet-switched 0x04 - Camped 0xFF - Not Available
<i>rejCause</i>	<ul style="list-style-type: none"> Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6]. <ul style="list-style-type: none"> 0xFF - Not Available

<i>networkIdValid</i>	<ul style="list-style-type: none"> Indicates whether the network ID is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>MCC[PLMN_L↔ENGTH]</i>	<ul style="list-style-type: none"> Mobile Country Code. MCC digits in ASCII characters
<i>MNC[PLMN_L↔ENGTH]</i>	<ul style="list-style-type: none"> Mobile Network Code. MNC digits in ASCII characters An unused byte is set to 0xFF. In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.
<i>tacValid</i>	<ul style="list-style-type: none"> Indicates whether tracking area code is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>tac</i>	<ul style="list-style-type: none"> Tracking area code. Only applicable for LTE. <ul style="list-style-type: none"> 0xFFFF - Not Available

8.385.2 Field Documentation

8.385.2.1 `uint32_t nas_LTESysInfo::cellId`

8.385.2.2 `uint8_t nas_LTESysInfo::cellIdValid`

8.385.2.3 `uint16_t nas_LTESysInfo::lac`

8.385.2.4 `uint8_t nas_LTESysInfo::lacValid`

8.385.2.5 `uint8_t nas_LTESysInfo::MCC[3]`

8.385.2.6 `uint8_t nas_LTESysInfo::MNC[3]`

8.385.2.7 `uint8_t nas_LTESysInfo::networkIdValid`

8.385.2.8 `uint8_t nas_LTESysInfo::regRejectInfoValid`

8.385.2.9 `uint8_t nas_LTESysInfo::rejCause`

8.385.2.10 `uint8_t nas_LTESysInfo::rejectSrvDomain`

8.385.2.11 `nas_sysInfoCommon nas_LTESysInfo::sysInfoLTE`

8.385.2.12 `uint16_t nas_LTESysInfo::tac`

8.385.2.13 `uint8_t nas_LTESysInfo::tacValid`

8.386 `nas_lteWcdmaCellInfo` Struct Reference

Data Fields

- `uint16_t uarfcn`
- `uint8_t cellReselPriority`
- `uint16_t threshXhigh`
- `uint16_t threshXlow`
- `uint8_t cellsLen`
- `nas_wcdmaCellInfo WCDMACellInfo` [255]

8.386.1 Detailed Description

This structure contains information about the LTE WCDMA Cell.

Parameters

<i>uarfcn</i>	<ul style="list-style-type: none"> • WCDMA layer frequency. • Range: 0 to 16383.
<i>cellReselPriority</i>	<ul style="list-style-type: none"> • Cell re-selection priority. • Range: 0 to 7. • This field is only valid when <code>ue_in_idle</code> is TRUE.
<i>threshXhigh</i>	<ul style="list-style-type: none"> • Re-selection low threshold. • Range: 0 to 31. • This field is only valid when <code>ue_in_idle</code> is TRUE.
<i>threshXlow</i>	<ul style="list-style-type: none"> • Re-selection high threshold. • Range: 0 to 31. • This field is only valid when <code>ue_in_idle</code> is TRUE.
<i>cellsLen</i>	<ul style="list-style-type: none"> • Provides the number of set of WCDMA cells.
<i>WCDMACellInfo[NAS_MAX_DESCRIPTOR_LENGTH]</i>	<ul style="list-style-type: none"> • See wcdmaCellInfo for more information.

8.386.2 Field Documentation

8.386.2.1 `uint8_t nas_lteWcdmaCellInfo::cellReselPriority`

8.386.2.2 `uint8_t nas_lteWcdmaCellInfo::cellsLen`

8.386.2.3 `uint16_t nas_lteWcdmaCellInfo::threshXhigh`

8.386.2.4 `uint16_t nas_lteWcdmaCellInfo::threshXlow`

8.386.2.5 `uint16_t nas_lteWcdmaCellInfo::uarfcn`

8.386.2.6 `nas_wcdmaCellInfo nas_lteWcdmaCellInfo::WCDMACellInfo[255]`

8.387 nas_MNRInfo Struct Reference

Data Fields

- `uint16_t mcc`
- `uint16_t mnc`
- `uint32_t rat`

8.387.1 Detailed Description

Structure contains Manual Network Register Information parameters

Parameters

<i>mcc</i>	<ul style="list-style-type: none">• A 16-bit integer representation of Mobile Country Code. Range - 0 to 999.
<i>mnc</i>	<ul style="list-style-type: none">• A 16-bit integer representation of Mobile Network Code. Range - 0 to 999.
<i>rat</i>	<ul style="list-style-type: none">• Radio access technology for which to register.<ul style="list-style-type: none">– 0x04 - RADIO_IF_GSM– 0x05 - RADIO_IF_UMTS– 0x08 - RADIO_IF_LTE

8.387.2 Field Documentation

8.387.2.1 `uint16_t nas_MNRInfo::mcc`

8.387.2.2 `uint16_t nas_MNRInfo::mnc`

8.387.2.3 uint32_t nas_MNRInfo::rat

8.388 nas_netSelectionPref Struct Reference

Data Fields

- uint8_t [netReg](#)
- uint16_t [mcc](#)
- uint16_t [mnc](#)

8.388.1 Detailed Description

Contain the network selection preference.

Parameters

<i>netReg</i>	<ul style="list-style-type: none">• specifies one of the following actions:<ul style="list-style-type: none">– 0x00 - Automatic registration<ul style="list-style-type: none">* Device registers according to its provisioning; mcc and mnc fields are ignored– 0x01 - Manual Registration<ul style="list-style-type: none">* Device registers to specified network; mcc and mnc must contain valid values
<i>mcc</i>	<ul style="list-style-type: none">• MCC value. Range 0 to 999
<i>mnc</i>	<ul style="list-style-type: none">• MNC value. Range 0 to 999

8.388.2 Field Documentation

8.388.2.1 uint16_t nas_netSelectionPref::mcc

8.388.2.2 uint16_t nas_netSelectionPref::mnc

8.388.2.3 uint8_t nas_netSelectionPref::netReg

8.389 nas_nmrCellInfo Struct Reference

Data Fields

- uint32_t [nmrCellID](#)
- uint8_t [nmrPlmn](#) [3]
- uint16_t [nmrLac](#)
- uint16_t [nmrArfcn](#)
- uint8_t [nmrBsic](#)
- uint16_t [nmrRxLev](#)

8.389.1 Detailed Description

This structure contains information about the Network Measurement Report (NMR) Cell Information.

Parameters

<i>nmrCellID</i>	<ul style="list-style-type: none"> Cell ID. 0xFFFFFFFF indicates cell ID information is not present.
<i>nmrPlmn[NAS↔ _PLMN_LEN↔ GTH]</i>	<ul style="list-style-type: none"> MCC/MNC information coded as octet 3, 4, and 5. This field is ignored when nmrCellID is not present.
<i>nmrLac</i>	<ul style="list-style-type: none"> Location area code. This field is ignored when nmrCellID is not present. <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>nmrArfcn</i>	<ul style="list-style-type: none"> Absolute RF channel number. <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>nmrBsic</i>	<ul style="list-style-type: none"> Base station identity code. <ul style="list-style-type: none"> 0xFF - Not Available
<i>nmrRxLev</i>	<ul style="list-style-type: none"> Cell Rx measurement. Values range between 0 and 63. Mapped to a measured signal level: <ul style="list-style-type: none"> Rxlev 0 is a signal strength less than -110 dBm Rxlev 1 is -110 dBm to -109 dBm Rxlev 2 is -109 dBm to -108 dBm ... Rxlev 62 is -49 dBm to -48 dBm Rxlev 63 is greater than -48 dBm 0xFFFF - Not Available

8.389.2 Field Documentation

8.389.2.1 uint16_t nas_nmrCellInfo::nmrArfcn

8.389.2.2 uint8_t nas_nmrCellInfo::nmrBsic

8.389.2.3 uint32_t nas_nmrCellInfo::nmrCellID

8.389.2.4 uint16_t nas_nmrCellInfo::nmrLac

8.389.2.5 `uint8_t nas_nmrCellInfo::nmrPlmn[3]`

8.389.2.6 `uint16_t nas_nmrCellInfo::nmrRxLev`

8.390 nas_PhyCaAggPcellInfo Struct Reference

Data Fields

- `uint16_t pci`
- `uint16_t freq`
- `NAS_LTE_CPHY_CA_BW_NRB_LITE dl_bw_value`
- `uint16_t iLTEbandValue`
- `uint8_t TlvPresent`

8.390.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Pcell Information.

Parameters

<i>pci</i>	<ul style="list-style-type: none"> • Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
<i>freq</i>	<ul style="list-style-type: none"> • Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
<i>dl_bw_value</i>	<ul style="list-style-type: none"> • Downlink Bandwidth Values. • See NAS_LTE_CPHY_CA_BW_NRB_LITE for more information.
<i>scell_state</i>	<ul style="list-style-type: none"> • SCell state Values. • See NAS_LTE_CPHY_SCELL_STATE_LITE for more information.
<i>TlvPresent</i>	<ul style="list-style-type: none"> • Tlv Present.

8.390.2 Field Documentation

8.390.2.1 `NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggPcellInfo::dl_bw_value`

8.390.2.2 `uint16_t nas_PhyCaAggPcellInfo::freq`

8.390.2.3 `uint16_t nas_PhyCaAggPcellInfo::iLTEbandValue`

8.390.2.4 uint16_t nas_PhyCaAggPcellInfo::pci

8.390.2.5 uint8_t nas_PhyCaAggPcellInfo::TlvPresent

8.391 nas_PhyCaAggScellIDIBw Struct Reference

Data Fields

- [NAS_LTE_CPHY_CA_BW_NRB_LITE dl_bw_value](#)
- uint8_t [TlvPresent](#)

8.391.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation Downlink Bandwidth of Scell.

Parameters

<i>dl_bw_value</i>	<ul style="list-style-type: none">• Downlink Bandwidth Values.• See NAS_LTE_CPHY_CA_BW_NRB_LITE for more information.
--------------------	--

8.391.2 Field Documentation

8.391.2.1 [NAS_LTE_CPHY_CA_BW_NRB_LITE](#) nas_PhyCaAggScellIDIBw::dl_bw_value

8.391.2.2 uint8_t nas_PhyCaAggScellIDIBw::TlvPresent

8.392 nas_PhyCaAggScellIndex Struct Reference

Data Fields

- uint8_t [scell_idx](#)
- uint8_t [TlvPresent](#)

8.392.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Index.

Parameters

<i>scell_idx</i>	<ul style="list-style-type: none">• Physical cell ID of the SCell Range.• Range for ID values: 0 to 503.
<i>TlvPresent</i>	
Generated by Doxygen	<ul style="list-style-type: none">• Tlv Present.

8.392.2 Field Documentation

8.392.2.1 `uint8_t nas_PhyCaAggScellIndex::scell_idx`

8.392.2.2 `uint8_t nas_PhyCaAggScellIndex::TlvPresent`

8.393 nas_PhyCaAggScellIndType Struct Reference

Data Fields

- `uint16_t pci`
- `uint16_t freq`
- `NAS_LTE_CPHY_SCELL_STATE_LITE scell_state`
- `uint8_t TlvPresent`

8.393.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Indicator Type.

Parameters

<i>pci</i>	<ul style="list-style-type: none"> • Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
<i>freq</i>	<ul style="list-style-type: none"> • Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
<i>scell_state</i>	<ul style="list-style-type: none"> • Scell state Values. • See NAS_LTE_CPHY_SCELL_STATE_LITE for more information.
<i>TlvPresent</i>	<ul style="list-style-type: none"> • Tlv Present.

8.393.2 Field Documentation

8.393.2.1 `uint16_t nas_PhyCaAggScellIndType::freq`

8.393.2.2 `uint16_t nas_PhyCaAggScellIndType::pci`

8.393.2.3 `NAS_LTE_CPHY_SCELL_STATE_LITE nas_PhyCaAggScellIndType::scell_state`

8.393.2.4 `uint8_t nas_PhyCaAggScellIndType::TlvPresent`

8.394 nas_PhyCaAggScellInfo Struct Reference

Data Fields

- uint16_t [pci](#)
- uint16_t [freq](#)
- [NAS_LTE_CPHY_CA_BW_NRB_LITE](#) [dl_bw_value](#)
- uint16_t [iLTEbandValue](#)
- [NAS_LTE_CPHY_SCELL_STATE_LITE](#) [scell_state](#)
- uint8_t [TlvPresent](#)

8.394.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Information.

Parameters

<i>pci</i>	<ul style="list-style-type: none">• Physical cell ID of the SCell Range.• Range for ID values: 0 to 503.
<i>freq</i>	<ul style="list-style-type: none">• Frequency of the absolute cell Range.• Range for ID values: 0 to 65535.
<i>dl_bw_value</i>	<ul style="list-style-type: none">• Downlink Bandwidth Values.• See NAS_LTE_CPHY_CA_BW_NRB_LITE for more information.

<i>ltebandValue</i>	<ul style="list-style-type: none"> • Band value. • Range for LTE Band class 120 to 160. <ul style="list-style-type: none"> – 120 - LTE E-UTRA Operating Band 1 – 121 - LTE E-UTRA Operating Band 2 – 122 - LTE E-UTRA Operating Band 3 – 123 - LTE E-UTRA Operating Band 4 – 124 - LTE E-UTRA Operating Band 5 – 125 - LTE E-UTRA Operating Band 6 – 126 - LTE E-UTRA Operating Band 7 – 127 - LTE E-UTRA Operating Band 8 – 128 - LTE E-UTRA Operating Band 9 – 129 - LTE E-UTRA Operating Band 10 – 130 - LTE E-UTRA Operating Band 11 – 131 - LTE E-UTRA Operating Band 12 – 132 - LTE E-UTRA Operating Band 13 – 133 - LTE E-UTRA Operating Band 14 – 134 - LTE E-UTRA Operating Band 17 – 135 - LTE E-UTRA Operating Band 33 – 136 - LTE E-UTRA Operating Band 34 – 137 - LTE E-UTRA Operating Band 35 – 138 - LTE E-UTRA Operating Band 36 – 139 - LTE E-UTRA Operating Band 37 – 140 - LTE E-UTRA Operating Band 38 – 141 - LTE E-UTRA Operating Band 39 – 142 - LTE E-UTRA Operating Band 40 – 143 - LTE E-UTRA Operating Band 18 – 144 - LTE E-UTRA Operating Band 19 – 145 - LTE E-UTRA Operating Band 20 – 146 - LTE E-UTRA Operating Band 21 – 147 - LTE E-UTRA Operating Band 24 – 148 - LTE E-UTRA Operating Band 25 – 149 - LTE E-UTRA Operating Band 41 – 150 - LTE E-UTRA Operating Band 42 – 151 - LTE E-UTRA Operating Band 43 – 152 - LTE E-UTRA Operating Band 23 – 153 - LTE E-UTRA Operating Band 26 – 154 - LTE E-UTRA Operating Band 32 – 155 - LTE E-UTRA Operating Band 125 – 156 - LTE E-UTRA Operating Band 126 – 157 - LTE E-UTRA Operating Band 127 – 158 - LTE E-UTRA Operating Band 28 – 159 - LTE E-UTRA Operating Band 29 – 160 - LTE E-UTRA Operating Band 30
<i>cell_state</i>	<ul style="list-style-type: none"> • Cell state Values. • See NAS_LTE_CPHY_CELL_STATE_LITE for more information.

<i>TlvPresent</i>	<ul style="list-style-type: none">• Tlv Present.
-------------------	--

8.394.2 Field Documentation

8.394.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggScellInfo::dl_bw_value

8.394.2.2 uint16_t nas_PhyCaAggScellInfo::freq

8.394.2.3 uint16_t nas_PhyCaAggScellInfo::ltebandValue

8.394.2.4 uint16_t nas_PhyCaAggScellInfo::pci

8.394.2.5 NAS_LTE_CPHY_SCELL_STATE_LITE nas_PhyCaAggScellInfo::scell_state

8.394.2.6 uint8_t nas_PhyCaAggScellInfo::TlvPresent

8.395 nas_qaQmi3Gpp2TimeZone Struct Reference

Data Fields

- uint8_t [leapSeconds](#)
- uint8_t [localTimeOffset](#)
- uint8_t [daylightSavings](#)

8.395.1 Detailed Description

This structure contains the 3GPP2TimeZone parameters

Parameters

<i>leapSeconds</i>	<ul style="list-style-type: none">• leap seconds - Number of leap seconds since the start of CDMA system time.
<i>localTimeOffset</i>	<ul style="list-style-type: none">• Local Time Offset - Offset of system time in units of 30 minutes; the value in this field conveys as 8 bit 2's compliment number.
<i>daylightSavings</i>	<ul style="list-style-type: none">• Day Light Savings Indicator<ul style="list-style-type: none">– 0x00 - OFF (daylight savings not in effect)– 0x01 - ON (daylight savings in effect)

8.395.2 Field Documentation

8.395.2.1 `uint8_t nas_qaQmi3Gpp2TimeZone::daylightSavings`

8.395.2.2 `uint8_t nas_qaQmi3Gpp2TimeZone::leapSeconds`

8.395.2.3 `uint8_t nas_qaQmi3Gpp2TimeZone::localTimeOffset`

8.396 nas_QmiNas3GppNetworkInfo Struct Reference

Data Fields

- `uint16_t` [MCC](#)
- `uint16_t` [MNC](#)
- `uint32_t` [InUse](#)
- `uint32_t` [Roaming](#)
- `uint32_t` [Forbidden](#)
- `uint32_t` [Preferred](#)
- `char` [Description](#) [255]

8.396.1 Detailed Description

Parameters

<i>MCC</i>	Mobile Country Code
<i>MNC</i>	Mobile Network Code
<i>InUse</i>	current network or not
<i>Roaming</i>	Home/Roam Status of the Network
<i>Forbidden</i>	Network in the forbidden network list or not
<i>Preferred</i>	Network in the Preferred network list or not
<i>Description</i>	Network Name/Description

8.396.2 Field Documentation

8.396.2.1 `char nas_QmiNas3GppNetworkInfo::Description[255]`

8.396.2.2 `uint32_t nas_QmiNas3GppNetworkInfo::Forbidden`

8.396.2.3 `uint32_t nas_QmiNas3GppNetworkInfo::InUse`

8.396.2.4 `uint16_t nas_QmiNas3GppNetworkInfo::MCC`

8.396.2.5 `uint16_t nas_QmiNas3GppNetworkInfo::MNC`

8.396.2.6 `uint32_t nas_QmiNas3GppNetworkInfo::Preferred`

8.396.2.7 uint32_t nas_QmiNas3GppNetworkInfo::Roaming

8.397 nas_QmiNas3GppNetworkRAT Struct Reference

Data Fields

- uint16_t [MCC](#)
- uint16_t [MNC](#)
- uint8_t [RAT](#)

8.397.1 Detailed Description

Contain the 3GPP radio access technology information.

Parameters

<i>MCC</i>	<ul style="list-style-type: none">• Mobile Country Code
<i>MNC</i>	<ul style="list-style-type: none">• Mobile Network Code
<i>RAT</i>	<ul style="list-style-type: none">• Radio Access Technology<ul style="list-style-type: none">– 0x04 - GERAN– 0x05 - UMTS– 0x08 - LTE– 0x09 - TD-SCDMA

8.397.2 Field Documentation

8.397.2.1 uint16_t nas_QmiNas3GppNetworkRAT::MCC

8.397.2.2 uint16_t nas_QmiNas3GppNetworkRAT::MNC

8.397.2.3 uint8_t nas_QmiNas3GppNetworkRAT::RAT

8.398 nas_QmisNasPcsDigit Struct Reference

Data Fields

- uint16_t [MCC](#)
- uint16_t [MNC](#)
- uint8_t [includes_pcs_digit](#)

8.398.1 Detailed Description

Contain the PCS Digit information

Parameters

<i>MCC</i>	<ul style="list-style-type: none"> • Mobile Country Code
<i>MNC</i>	<ul style="list-style-type: none"> • Mobile Network Code
<i>includes_pcs_↔ digit</i>	<ul style="list-style-type: none"> • this field is use to interpret the length of corresponding MNC reported • 0x01 - MNC is a three-digit value • 0x00 - MNC is a two-digit value

8.398.2 Field Documentation

8.398.2.1 `uint8_t nas_QmisNasPcsDigit::includes_pcs_digit`

8.398.2.2 `uint16_t nas_QmisNasPcsDigit::MCC`

8.398.2.3 `uint16_t nas_QmisNasPcsDigit::MNC`

8.399 nas_RejectReasonTlv Struct Reference

Data Fields

- `uint8_t TlvPresent`
- `uint32_t serviceDomain`
- `uint32_t rejectCause`

8.399.1 Detailed Description

Parameters

<i>TlvPresent</i>	indicating the presence of the TLV in the QMI ind
<i>serviceDomain</i>	service domain
<i>rejectCause</i>	cause of reject

8.399.2 Field Documentation

8.399.2.1 `uint32_t nas_RejectReasonTlv::rejectCause`

8.399.2.2 `uint32_t nas_RejectReasonTlv::serviceDomain`

8.399.2.3 uint8_t nas_RejectReasonTlv::TlvPresent

8.400 nas_RFInfoTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- uint8_t [radioInterfaceSize](#)
- uint32_t [radioInterface](#) [255]
- uint32_t [activeBandClass](#) [255]
- uint32_t [activeChannel](#) [255]

8.400.1 Detailed Description

Parameters

<i>TlvPresent</i>	indicating the presence of the TLV in the QMI ind
<i>radioInterface</i>	radio interface technology of the signal being measured
<i>activeBandClass</i>	active band class
<i>activeChannel</i>	active channel

8.400.2 Field Documentation

8.400.2.1 uint32_t nas_RFInfoTlv::activeBandClass[255]

8.400.2.2 uint32_t nas_RFInfoTlv::activeChannel[255]

8.400.2.3 uint32_t nas_RFInfoTlv::radioInterface[255]

8.400.2.4 uint8_t nas_RFInfoTlv::radioInterfaceSize

8.400.2.5 uint8_t nas_RFInfoTlv::TlvPresent

8.401 nas_roamIndList Struct Reference

Data Fields

- uint8_t [numInstances](#)
- uint8_t [radioInterface](#) [32]
- uint8_t [roamIndicator](#) [32]

8.401.1 Detailed Description

This structure contains the Roaming Indicator List

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • number of sets of radio interface currently in use and roaming indicator <ul style="list-style-type: none"> – defaults to zero
<i>radioInterface</i>	<ul style="list-style-type: none"> • Radio Interface currently in use • Values: <ul style="list-style-type: none"> – 0x01 - RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x03 - RADIO_IF_AMPS - AMPS – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS – 0x08 - RADIO_IF_LTE - LTE
<i>roamIndicator</i>	<ul style="list-style-type: none"> • Roaming Indicator • Values: <ul style="list-style-type: none"> – 0x00 - Roaming – 0x01 - Home

8.401.2 Field Documentation

8.401.2.1 `uint8_t nas_roamIndList::numInstances`

8.401.2.2 `uint8_t nas_roamIndList::radioInterface[32]`

8.401.2.3 `uint8_t nas_roamIndList::roamIndicator[32]`

8.402 nas_rsrqInformation Struct Reference

Data Fields

- `int8_t rsrq`
- `uint8_t radiolf`

8.402.1 Detailed Description

This structure contains the RSRQ Information

Parameters

<i>rsrq</i>	<ul style="list-style-type: none"> • RSRQ value in dB (signed integer value); valid range is -3 to -20 (-3 means -3 dB, -20 means -20 dB)
<i>radiolf</i>	<ul style="list-style-type: none"> • Radio interface technology of the signal being measured
	<ul style="list-style-type: none"> – 0x08 – LTE

8.402.2 Field Documentation

8.402.2.1 `uint8_t nas_rsrqInformation::radiolf`

8.402.2.2 `int8_t nas_rsrqInformation::rsrq`

8.403 nas_RxSigInfo Struct Reference

Data Fields

- `uint8_t rxChainIndex`
- `uint8_t isRadioTuned`
- `int32_t rxPower`
- `int32_t rsrp`

8.403.1 Detailed Description

This structure contains the parameters for Rx Signal Info.

Parameters

<i>rxChainIndex</i>	<ul style="list-style-type: none">• Rx antenna path• Valid Values<ul style="list-style-type: none">– 0 - Primary Rx– 1 - Diversity Rx
<i>isRadioTuned</i>	<ul style="list-style-type: none">• Rx path is tuned to a channel or Not• Values<ul style="list-style-type: none">– 0x00 - Not tuned– 0x01 - Tuned

Note

If the radio is tuned, the instantaneous values are set for the fields below. If the radio is not tuned, the values set below may be invalid.

Parameters

<i>rxPower</i>	<ul style="list-style-type: none">• Rx power value in 1/10 dBm resolution
<i>rsrp</i>	<ul style="list-style-type: none">• Current reference signal received power in 1/10 dBm resolution

8.403.2 Field Documentation

8.403.2.1 `uint8_t nas_RxSigInfo::isRadioTuned`

8.403.2.2 `int32_t nas_RxSigInfo::rsrp`

8.403.2.3 `uint8_t nas_RxSigInfo::rxChainIndex`

8.403.2.4 `int32_t nas_RxSigInfo::rxPower`

8.404 `nas_rxSignalStrengthListElement` Struct Reference

Data Fields

- `int16_t rxSignalStrength`
- `uint8_t radiolf`

8.404.1 Detailed Description

This structure contains the Received Signal Strength Information

Parameters

<i>rxSignalStrength</i>	<ul style="list-style-type: none"> • Received signal strength in dBm <ul style="list-style-type: none"> – For CDMA and UMTS, this indicates forward link pilotEc. – For GSM, the received signal strength. – For LTE, this indicates the total received wideband power observed by UE.
<i>radiolf</i>	<ul style="list-style-type: none"> • Radio interface technology of the signal being radio_if measured <ul style="list-style-type: none"> – 0x00 – RADIO_IF_NO_SVC – None (no service) – 0x01 – RADIO_IF_CDMA_1X – cdma2000@ 1X – 0x02 – RADIO_IF_CDMA_1XEVD0 – cdma2000 HRPD (1xEV-DO) – 0x03 – RADIO_IF_AMPS – AMPS – 0x04 – RADIO_IF_GSM – GSM – 0x05 – RADIO_IF_UMTS – UMTS – 0x08 – RADIO_IF_LTE – LTE

Note

First elemnet of the RSSI list always contains the current Signal strength and Radio Interface.

8.404.2 Field Documentation

8.404.2.1 `uint8_t nas_rxSignalStrengthListElement::radiolf`

8.404.2.2 int16_t nas_rxSignalStrengthListElement::rxSignalStrength

8.405 nas_SccRxInfo Struct Reference

Data Fields

- int32_t [rsrq](#)
- int16_t [snr](#)
- uint8_t [numInstances](#)
- [nas_RxSigInfo sigInfo](#) [255]
- uint8_t [TlvPresent](#)

8.405.1 Detailed Description

This structure contains information about the [SccRxInfo](#) parameters.

Parameters

<i>rsrq</i>	<ul style="list-style-type: none"> • Current reference signal • Receive quality in 1/10 dB resolution
<i>snr</i>	<ul style="list-style-type: none"> • Reference signal signal-to-noise ratio in dB. • Range -10 to 30
<i>numInstances</i>	<ul style="list-style-type: none"> • Number of sets of the following <ul style="list-style-type: none"> – rxChainIndex – isRadioTuned – rxPower – rsrp
<i>sigInfo</i>	<ul style="list-style-type: none"> • See nas_RxSigInfo for more information
<i>TlvPresent</i>	<ul style="list-style-type: none"> • Tlv Present.

8.405.2 Field Documentation

8.405.2.1 uint8_t nas_SccRxInfo::numInstances

8.405.2.2 int32_t nas_SccRxInfo::rsrq

8.405.2.3 nas_RxSigInfo nas_SccRxInfo::sigInfo[255]

8.405.2.4 int16_t nas_SccRxInfo::snr

8.405.2.5 uint8_t nas_SccRxInfo::TlvPresent

8.406 nas_servSystem Struct Reference

Data Fields

- uint8_t [regState](#)
- uint8_t [csAttachState](#)
- uint8_t [psAttachState](#)
- uint8_t [selNetwork](#)
- uint8_t [numRadioInterfaces](#)
- uint8_t [radioInterface](#) [32]

8.406.1 Detailed Description

This structure contains the Serving System parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>regState</i>	<ul style="list-style-type: none"> • Registration state - Registration state of the mobile • Values: <ul style="list-style-type: none"> – 0 - Not Registered; mobile is not currently searching for a new network to provide service – 1 - Registered with a network – 2 - Not registered, but mobile is currently searching for a new network to provide service – 3 - Registration denied by visible network – 4 - Registration state is unknown
<i>csAttachState</i>	<ul style="list-style-type: none"> • CS Attach State - Circuit-switched domain attach state of the mobile • Values: <ul style="list-style-type: none"> – 0 - Unknown or not applicable – 1 - Attached – 2 - Detached
<i>psAttachState</i>	<ul style="list-style-type: none"> • PS Attach State - Packet-switched domain attach state of the mobile • Values: <ul style="list-style-type: none"> – 0 - Unknown or not applicable – 1 - Attached – 2 - Detached

<i>selNetwork</i>	<ul style="list-style-type: none"> Selected Network - Type of selected radio access network Values: <ul style="list-style-type: none"> 0 - Unknown 1 - 3GPP2 network 2 - 3GPP network
<i>numRadio↔ Interfaces</i>	<ul style="list-style-type: none"> In Use Radio Interfaces Number <ul style="list-style-type: none"> Number of radio interfaces currently in use defaults to zero
<i>radioInterface</i>	<ul style="list-style-type: none"> Radio Interface currently in use Values: <ul style="list-style-type: none"> 0x00 - RADIO_IF_NO_SVC - None(no service) 0x01 - RADIO_IF_CDMA_1X - cdma2000 1X 0x02 - RADIO_IF_CDMA_1XEVD0 - cdma2000 HRPD (1xEV-DO) 0x03 - RADIO_IF_AMPS - AMPS 0x04 - RADIO_IF_GSM - GSM 0x05 - RADIO_IF_UMTS - UMTS 0x08 - RADIO_IF_LTE - LTE

8.406.2 Field Documentation

8.406.2.1 uint8_t nas_servSystem::csAttachState

8.406.2.2 uint8_t nas_servSystem::numRadioInterfaces

8.406.2.3 uint8_t nas_servSystem::psAttachState

8.406.2.4 uint8_t nas_servSystem::radioInterface[32]

8.406.2.5 uint8_t nas_servSystem::regState

8.406.2.6 uint8_t nas_servSystem::selNetwork

8.407 nas_SignalStrengthTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- int8_t [signalStrength](#)
- uint32_t [radioInterface](#)

8.407.1 Detailed Description

Parameters

<i>TlvPresent</i>	indicating the presence of the TLV in the QMI ind
<i>signalStrength</i>	signal strength
<i>radioInterface</i>	radio interface technology of the signal being measured

8.407.2 Field Documentation

8.407.2.1 uint32_t nas_SignalStrengthTlv::radioInterface

8.407.2.2 int8_t nas_SignalStrengthTlv::signalStrength

8.407.2.3 uint8_t nas_SignalStrengthTlv::TlvPresent

8.408 nas_SLQSSignalStrengthsIndReq Struct Reference

Data Fields

- uint8_t rxSignalStrengthDelta
- uint8_t ecioDelta
- uint8_t ioDelta
- uint8_t sinrDelta
- uint8_t rsrqDelta
- uint8_t ecioThresholdListLen
- int16_t ecioThresholdList [10]
- uint8_t sinrThresholdListLen
- uint8_t sinrThresholdList [5]
- uint16_t lteSnrDelta
- uint8_t lteRsrpDelta

8.408.1 Detailed Description

Parameters

<i>rxSignalStrengthDelta</i>	RSSI delta(in dBm) at which an event report indication
<i>ecioDelta</i>	ecio delta
<i>ioDelta</i>	io delta
<i>sinrDelta</i>	sinr delta
<i>rsrqDelta</i>	rsrq delta
<i>ecioThresholdListLen</i>	
<i>ecioThresholdList</i>	
<i>sinrThresholdListLen</i>	
<i>sinrThresholdList</i>	
<i>lteSnrDelta</i>	lte snr delta
<i>lteRsrpDelta</i>	lte rsrp delta

8.408.2 Field Documentation

- 8.408.2.1 `uint8_t` `nas_SLQSSignalStrengthsIndReq::ecioDelta`
- 8.408.2.2 `int16_t` `nas_SLQSSignalStrengthsIndReq::ecioThresholdList[10]`
- 8.408.2.3 `uint8_t` `nas_SLQSSignalStrengthsIndReq::ecioThresholdListLen`
- 8.408.2.4 `uint8_t` `nas_SLQSSignalStrengthsIndReq::ioDelta`
- 8.408.2.5 `uint8_t` `nas_SLQSSignalStrengthsIndReq::lteRsrpDelta`
- 8.408.2.6 `uint16_t` `nas_SLQSSignalStrengthsIndReq::lteSnrDelta`
- 8.408.2.7 `uint8_t` `nas_SLQSSignalStrengthsIndReq::rsrqDelta`
- 8.408.2.8 `uint8_t` `nas_SLQSSignalStrengthsIndReq::rxSignalStrengthDelta`
- 8.408.2.9 `uint8_t` `nas_SLQSSignalStrengthsIndReq::sinrDelta`
- 8.408.2.10 `uint8_t` `nas_SLQSSignalStrengthsIndReq::sinrThresholdList[5]`
- 8.408.2.11 `uint8_t` `nas_SLQSSignalStrengthsIndReq::sinrThresholdListLen`

8.409 nas_SLQSSignalStrengthsInformation Struct Reference

Data Fields

- [nas_rxSignalStrengthListElement](#) `rxSignalStrengthInfo`
- [nas_ecioListElement](#) `ecioInfo`
- `uint32_t` `io`
- `uint8_t` `sinr`
- [nas_errorRateListElement](#) `errorRateInfo`
- [nas_rsrqInformation](#) `rsrqInfo`
- [nas_lteSnrinformation](#) `lteSnrinfo`
- [nas_lteRsrpinformation](#) `lteRsrpinfo`

8.409.1 Detailed Description

Parameters

<i>rxSignalStrengthInfo</i>	signal strength info list
<i>ecioInfo</i>	ecio info list
<i>io</i>	received IO in dBm; IO is only applicable for 1xEV-DO
<i>sinr</i>	SINR level
<i>errorRateInfo</i>	error rate info
<i>rsrqInfo</i>	rsrq info
<i>lteSnrinfo</i>	lte Snr information
<i>lteRsrpinfo</i>	lte rsrp info

8.409.2 Field Documentation

8.409.2.1 `nas_ecioListElement` `nas_SLQSSignalStrengthsInformation::ecioInfo`

8.409.2.2 `nas_errorRateListElement` `nas_SLQSSignalStrengthsInformation::errorRateInfo`

8.409.2.3 `uint32_t` `nas_SLQSSignalStrengthsInformation::io`

8.409.2.4 `nas_lteRsrpInformation` `nas_SLQSSignalStrengthsInformation::lteRsrpInfo`

8.409.2.5 `nas_lteSnrInformation` `nas_SLQSSignalStrengthsInformation::lteSnrInfo`

8.409.2.6 `nas_rsrqInformation` `nas_SLQSSignalStrengthsInformation::rsrqInfo`

8.409.2.7 `nas_rxSignalStrengthListElement` `nas_SLQSSignalStrengthsInformation::rxSignalStrengthInfo`

8.409.2.8 `uint8_t` `nas_SLQSSignalStrengthsInformation::sinr`

8.410 `nas_SLQSSignalStrengthsTlv` Struct Reference

Data Fields

- `uint8_t` [TlvPresent](#)
- `nas_SLQSSignalStrengthsInformation` [sSLQSSignalStrengthsInfo](#)

8.410.1 Detailed Description

Parameters

<i>TlvPresent</i>	indicating the presence of the TLV in the QMI ind
<i>sSLQSSignalStrengthsInfo</i>	signal strength info

8.410.2 Field Documentation

8.410.2.1 `nas_SLQSSignalStrengthsInformation` `nas_SLQSSignalStrengthsTlv::sSLQSSignalStrengthsInfo`

8.410.2.2 `uint8_t` `nas_SLQSSignalStrengthsTlv::TlvPresent`

8.411 `nas_SrvStatusInfo` Struct Reference

Data Fields

- `uint8_t` [srvStatus](#)
- `uint8_t` [isPrefDataPath](#)

8.411.1 Detailed Description

Structure for storing the service status information for CDMA and HDR networks.

Parameters

<i>srvStatus</i>	<ul style="list-style-type: none">• Service status of the system.<ul style="list-style-type: none">– 0x00 - No service– 0x01 - Limited service– 0x02 - Service– 0x03 - Limited regional service– 0x04 - Power save– 0xFF - Not Available
<i>isPrefDataPath</i>	<ul style="list-style-type: none">• Whether the RAT is the preferred data path.<ul style="list-style-type: none">– 0x00 - Not preferred– 0x01 - Preferred– 0xFF - Not Available

8.411.2 Field Documentation

8.411.2.1 `uint8_t nas_SrvStatusInfo::isPrefDataPath`

8.411.2.2 `uint8_t nas_SrvStatusInfo::srvStatus`

8.412 nas_sysInfoCommon Struct Reference

Data Fields

- `uint8_t srvDomainValid`
- `uint8_t srvDomain`
- `uint8_t srvCapabilityValid`
- `uint8_t srvCapability`
- `uint8_t roamStatusValid`
- `uint8_t roamStatus`
- `uint8_t isSysForbiddenValid`
- `uint8_t isSysForbidden`

8.412.1 Detailed Description

Structure for storing the System Information common to CDMA, HDR, GSM, WCDMA and LTE networks.

Parameters

<i>srvDomainValid</i>	<ul style="list-style-type: none"> Indicates whether the service domain is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>srvDomain</i>	<ul style="list-style-type: none"> Service domain registered on the system. <ul style="list-style-type: none"> 0x00 - No service 0x01 - Circuit-switched only 0x02 - Packet-switched only 0x03 - Circuit-switched and packet-switched 0x04 - Camped 0xFF - Not Available
<i>srvCapability</i> ↔ <i>Valid</i>	<ul style="list-style-type: none"> Indicates whether the service capability is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>srvCapability</i>	<ul style="list-style-type: none"> Current system's service capability. <ul style="list-style-type: none"> 0x00 - No service 0x01 - Circuit-switched only 0x02 - Packet-switched only 0x03 - Circuit-switched and packet-switched 0x04 - Camped 0xFF - Not Available
<i>roamStatusValid</i>	<ul style="list-style-type: none"> Indicates whether the roaming status is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available

<i>roamStatus</i>	<ul style="list-style-type: none"> • Current roaming status. <ul style="list-style-type: none"> – 0x00 - Off – 0x01 - On – 0x02 - Blinking – 0x03 - Out of the neighborhood – 0x04 - Out of the building – 0x05 - Preferred system – 0x06 - Available system – 0x07 - Alliance partner – 0x08 - Premium partner – 0x09 - Full service – 0x0A - Partial service – 0x0B - Banner is on – 0x0C - Banner is off – 0x0D to 0x3F - Reserved for Standard Enhanced Roaming Indicator Numbers – 0x40 to 0x7F - Reserved for Non-Standard Enhanced Roaming Indicator Numbers – 0x40 to 0xFF - Reserved. – 0xFF - Not Available • Values from 0x02 onward are only applicable for 3GPP2
<i>isSys↔ ForbiddenValid</i>	<ul style="list-style-type: none"> • Indicates whether the forbidden system is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>isSysForbidden</i>	<ul style="list-style-type: none"> • Whether the system is forbidden. <ul style="list-style-type: none"> – 0x00 - Not forbidden – 0x01 - Forbidden – 0xFF - Not Available

8.412.2 Field Documentation

8.412.2.1 `uint8_t nas_sysInfoCommon::isSysForbidden`

8.412.2.2 `uint8_t nas_sysInfoCommon::isSysForbiddenValid`

8.412.2.3 `uint8_t nas_sysInfoCommon::roamStatus`

8.412.2.4 `uint8_t nas_sysInfoCommon::roamStatusValid`

8.412.2.5 `uint8_t nas_sysInfoCommon::srvCapability`

8.412.2.6 `uint8_t nas_sysInfoCommon::srvCapabilityValid`

8.412.2.7 `uint8_t nas_sysInfoCommon::srvDomain`

8.412.2.8 `uint8_t nas_sysInfoCommon::srvDomainValid`

8.413 `nas_TDSCDMAECIOThresh` Struct Reference

Data Fields

- `uint8_t` [TDSCDMAECIOThreshListLen](#)
- `float *` [pTDSCDMAECIOThreshList](#)

8.413.1 Detailed Description

This structure contains TDSCDMA ECIO threshold related parameters.

Parameters

<i>TDSCDMAECIOThreshListLen</i>	<ul style="list-style-type: none"> • Length of the TDSCDMA ECIO threshold list parameter to follow
<i>pTDSCDMAECIOThreshList</i>	<ul style="list-style-type: none"> • Array of ECIO thresholds (in dB) used by TD-SCDMA • Maximum of 32 values.

8.413.2 Field Documentation

8.413.2.1 `float* nas_TDSCDMAECIOThresh::pTDSCDMAECIOThreshList`

8.413.2.2 `uint8_t nas_TDSCDMAECIOThresh::TDSCDMAECIOThreshListLen`

8.414 `nas_TDSCDMARSCPThresh` Struct Reference

Data Fields

- `uint8_t` [TDSCDMARSCPThreshListLen](#)
- `int16_t *` [pTDSCDMARSCPThreshList](#)

8.414.1 Detailed Description

This structure contains TDSCDMA RSCP threshold related parameters.

Parameters

<i>TDSCDMARSCPThreshListLen</i>	<ul style="list-style-type: none"> Length of the TDSCDMA RSCP threshold list parameter to follow
<i>pTDSCDMARSCPThreshList</i>	<ul style="list-style-type: none"> Array of RSCP thresholds (in units of 0.1 dBm) Maximum of 32 values Range for RSCP values: -120 to -25 (in dBm).

8.414.2 Field Documentation

8.414.2.1 int16_t* nas_TDSCDMARSCPThresh::pTDSCDMARSCPThreshList

8.414.2.2 uint8_t nas_TDSCDMARSCPThresh::TDSCDMARSCPThreshListLen

8.415 nas_TDSCDMARSSIThresh Struct Reference

Data Fields

- uint8_t TDSCDMARSSIThreshListLen
- float * pTDSCDMARSSIThreshList

8.415.1 Detailed Description

This structure contains TDSCDMA RSSI threshold related parameters.

Parameters

<i>TDSCDMARSSIThreshListLen</i>	<ul style="list-style-type: none"> Length of the TDSCDMA RSSI threshold list parameter to follow
<i>pTDSCDMARSSIThreshList</i>	<ul style="list-style-type: none"> Array of RSSI thresholds (in dBm) used by TD-SCDMA Maximum of 32 values.

8.415.2 Field Documentation

8.415.2.1 float* nas_TDSCDMARSSIThresh::pTDSCDMARSSIThreshList

8.415.2.2 uint8_t nas_TDSCDMARSSIThresh::TDSCDMARSSIThreshListLen

8.416 nas_TDSCDMASINRThresh Struct Reference

Data Fields

- uint8_t [TDSCDMASINRThreshListLen](#)
- float * [pTDSCDMASINRThreshList](#)

8.416.1 Detailed Description

This structure contains TDSCDMA SINR threshold related parameters.

Parameters

<i>TDSCDMASINRThreshListLen</i>	<ul style="list-style-type: none"> • Length of the TDSCDMA SINR threshold list parameter to follow
<i>pTDSCDMASINRThreshList</i>	<ul style="list-style-type: none"> • Array of SINR thresholds (in dB) used by TD-SCDMA • Maximum of 32 values

8.416.2 Field Documentation

8.416.2.1 float* nas_TDSCDMASINRThresh::pTDSCDMASINRThreshList

8.416.2.2 uint8_t nas_TDSCDMASINRThresh::TDSCDMASINRThreshListLen

8.417 nas_timeInfo Struct Reference

Data Fields

- uint16_t [year](#)
- uint8_t [month](#)
- uint8_t [day](#)
- uint8_t [hour](#)
- uint8_t [minute](#)
- uint8_t [second](#)
- uint8_t [dayOfWeek](#)
- int8_t [timeZone](#)
- uint8_t [dayLtSavingAdj](#)
- uint8_t [radioInterface](#)
- uint8_t [TlvPresent](#)

8.417.1 Detailed Description

This structure contains the parameters for Network Time.

Parameters

<i>year</i>	<ul style="list-style-type: none"> • Year
<i>month</i>	<ul style="list-style-type: none"> • Month • 1 is January and 12 is December
<i>day</i>	<ul style="list-style-type: none"> • Day • Range - 1 to 31
<i>hour</i>	<ul style="list-style-type: none"> • Hour • Range - 0 to 59
<i>minute</i>	<ul style="list-style-type: none"> • Minute • Range - 0 to 59
<i>second</i>	<ul style="list-style-type: none"> • Second • Range - 0 to 59
<i>dayOfWeek</i>	<ul style="list-style-type: none"> • Day of the week • 0 is Monday and 6 is Sunday
<i>timeZone</i>	<ul style="list-style-type: none"> • Offset from Universal time • The difference between local time and Universal time, in increments of 15 min • Signed Value
<i>dayLtSavingAdj</i>	<ul style="list-style-type: none"> • Daylight saving adjustment in hours • Possible values - 0, 1, and 2. • This field is ignored if radio_if is NAS_RADIO_IF_CDMA_1XEVDO
<i>radioInterface</i>	<ul style="list-style-type: none"> • Radio interface from which the information comes • Values <ul style="list-style-type: none"> – 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - NAS_RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA - TD-SCDMA
<i>TlvPresent</i>	<ul style="list-style-type: none"> • Tlv Present.

8.417.2 Field Documentation

8.417.2.1 `uint8_t nas_timeInfo::day`

8.417.2.2 `uint8_t nas_timeInfo::dayLtSavingAdj`

8.417.2.3 `uint8_t nas_timeInfo::dayOfWeek`

8.417.2.4 `uint8_t nas_timeInfo::hour`

8.417.2.5 `uint8_t nas_timeInfo::minute`

8.417.2.6 `uint8_t nas_timeInfo::month`

8.417.2.7 `uint8_t nas_timeInfo::radioInterface`

8.417.2.8 `uint8_t nas_timeInfo::second`

8.417.2.9 `int8_t nas_timeInfo::timeZone`

8.417.2.10 `uint8_t nas_timeInfo::TlvPresent`

8.417.2.11 `uint16_t nas_timeInfo::year`

8.418 nas_UMTSInfo Struct Reference

Data Fields

- `uint16_t cellID`
- `uint8_t plmn` [3]
- `uint16_t lac`
- `uint16_t uarfcn`
- `uint16_t psc`
- `int16_t rscp`
- `int16_t ecio`
- `uint8_t umtsInst`
- `nas_UMTSInstInfo UMTSInstInfo` [255]
- `uint8_t geranInst`
- `nas_geranInstInfo GeranInstInfo` [255]

8.418.1 Detailed Description

This structure contains information about the UMTS Network.

Parameters

<i>cellID</i>	<ul style="list-style-type: none"> Cell ID. 0xFFFFFFFF indicates cell ID information is not present.
<i>plmn</i> [<i>NAS_PL</i> ↔ <i>MN_LENGTH</i>]	<ul style="list-style-type: none"> MCC/MNC information coded as octet 3, 4, and 5. This field is ignored when nmrCellID is not present.
<i>lac</i>	<ul style="list-style-type: none"> Location area code. This field is ignored when nmrCellID is not present. <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>uarfcn</i>	<ul style="list-style-type: none"> UTRA absolute RF channel number. <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>psc</i>	<ul style="list-style-type: none"> Primary scrambling code. <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>rscp</i>	<ul style="list-style-type: none"> Received signal code power. <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>ecio</i>	<ul style="list-style-type: none"> ECIO(Signal-to-Interference-ratio). <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>umtsInst</i>	<ul style="list-style-type: none"> Provides the number of set of UMTS info instances. If 0(zero), then no information follows it.
<i>UMTSInstInfo</i> [↔ <i>NAS_MAX_D</i> ↔ <i>ESCRPTION</i> ↔ <i>_LENGTH</i>]	<ul style="list-style-type: none"> See nas_UMTSInstInfo for more information.
<i>geranInst</i>	<ul style="list-style-type: none"> Provides the number of set of GERAN info instances. If 0(zero), then no information follows it.
<i>GeranInstInfo</i> [↔ <i>NAS_MAX_D</i> ↔ <i>ESCRPTION</i> ↔ <i>_LENGTH</i>]	<ul style="list-style-type: none"> See nas_geranInstInfo for more information.

8.418.2 Field Documentation

- 8.418.2.1 `uint16_t nas_UMTSInfo::cellID`
- 8.418.2.2 `int16_t nas_UMTSInfo::ecio`
- 8.418.2.3 `uint8_t nas_UMTSInfo::geranInst`
- 8.418.2.4 `nas_geranInstInfo nas_UMTSInfo::GeranInstInfo[255]`
- 8.418.2.5 `uint16_t nas_UMTSInfo::lac`
- 8.418.2.6 `uint8_t nas_UMTSInfo::plmn[3]`
- 8.418.2.7 `uint16_t nas_UMTSInfo::psc`
- 8.418.2.8 `int16_t nas_UMTSInfo::rscp`
- 8.418.2.9 `uint16_t nas_UMTSInfo::uarfcn`
- 8.418.2.10 `uint8_t nas_UMTSInfo::umtsInst`
- 8.418.2.11 `nas_UMTSInstInfo nas_UMTSInfo::UMTSInstInfo[255]`

8.419 `nas_UMTSInstInfo` Struct Reference

Data Fields

- `uint16_t` [umtsUarfcn](#)
- `uint16_t` [umtsPsc](#)
- `int16_t` [umtsRscp](#)
- `int16_t` [umtsEcio](#)

8.419.1 Detailed Description

This structure contains information about the UMTS Instances in UMTS Network.

Parameters

<i>umtsUarfcn</i>	<ul style="list-style-type: none"> • UTRA absolute RF channel number.
<i>umtsPsc</i>	<ul style="list-style-type: none"> • Primary scrambling code.
<i>umtsRscp</i>	<ul style="list-style-type: none"> • Received signal code power.
<i>umtsEcio</i>	<ul style="list-style-type: none"> • ECIO(Signal-to-Interference-ratio).
	Generated by Doxygen

8.419.2 Field Documentation

8.419.2.1 int16_t nas_UMTSinstInfo::umtsEcio

8.419.2.2 uint16_t nas_UMTSinstInfo::umtsPsc

8.419.2.3 int16_t nas_UMTSinstInfo::umtsRscp

8.419.2.4 uint16_t nas_UMTSinstInfo::umtsUarfcn

8.420 nas_umtsLTENbrCell Struct Reference

Data Fields

- uint16_t [earfcn](#)
- uint16_t [pci](#)
- uint32_t [rsrp](#)
- uint32_t [rsrq](#)
- int16_t [srxlev](#)
- uint8_t [cellsTDD](#)

8.420.1 Detailed Description

This structure contains information about the UMTS LTE neighbour Cell.

Parameters

<i>earfcn</i>	<ul style="list-style-type: none"> • E-UTRA absolute RF channel number of the detected cell.
<i>pci</i>	<ul style="list-style-type: none"> • Physical cell ID of the detected cell. • Range is defined in 3GPP TS 36.211
<i>rsrp</i>	<ul style="list-style-type: none"> • Current received signal strength indication (in dBm) of the detected cell.
<i>rsrq</i>	<ul style="list-style-type: none"> • Current reference signal received quality (in dB) of the detected cell.
<i>srxlev</i>	<ul style="list-style-type: none"> • Cell selection Rx level (Srxlev) value of the detected cell in linear scale. • This field is only valid when wcdma_rrc_state is not NAS_WCDMA_RRC_STATE_CEL_FACH or NAS_WCDMA_RRC_STATE_CELL_DCH.
<i>cellsTDD</i>	<ul style="list-style-type: none"> • TRUE if the cell is TDD; FALSE if the cell is FDD.

8.420.2 Field Documentation

8.420.2.1 `uint8_t nas_umtsLTENbrCell::cellsTDD`

8.420.2.2 `uint16_t nas_umtsLTENbrCell::earfcn`

8.420.2.3 `uint16_t nas_umtsLTENbrCell::pci`

8.420.2.4 `uint32_t nas_umtsLTENbrCell::rsrp`

8.420.2.5 `uint32_t nas_umtsLTENbrCell::rsrq`

8.420.2.6 `int16_t nas_umtsLTENbrCell::srxlev`

8.421 `nas_UniversalTime` Struct Reference

Data Fields

- `uint16_t year`
- `uint8_t month`
- `uint8_t day`
- `uint8_t hour`
- `uint8_t minute`
- `uint8_t second`
- `uint8_t dayOfWeek`

8.421.1 Detailed Description

This structure contains the parameters for Universal Time Information.

Parameters

<i>year</i>	<ul style="list-style-type: none"> • Year.
<i>month</i>	<ul style="list-style-type: none"> • Month. <ul style="list-style-type: none"> – 1 is January and 12 is December.
<i>day</i>	<ul style="list-style-type: none"> • Day. <ul style="list-style-type: none"> – Range 1 to 31.
<i>hour</i>	<ul style="list-style-type: none"> • Hour. <ul style="list-style-type: none"> – Range 0 to 59.

<i>minute</i>	<ul style="list-style-type: none">• Minute.<ul style="list-style-type: none">– Range 0 to 59.
<i>second</i>	<ul style="list-style-type: none">• Second.<ul style="list-style-type: none">– Range 0 to 59.
<i>dayOfWeek</i>	<ul style="list-style-type: none">• Day of the Week.<ul style="list-style-type: none">– 0 is Monday and 6 is Sunday.

8.421.2 Field Documentation

8.421.2.1 uint8_t nas_UniversalTime::day

8.421.2.2 uint8_t nas_UniversalTime::dayOfWeek

8.421.2.3 uint8_t nas_UniversalTime::hour

8.421.2.4 uint8_t nas_UniversalTime::minute

8.421.2.5 uint8_t nas_UniversalTime::month

8.421.2.6 uint8_t nas_UniversalTime::second

8.421.2.7 uint16_t nas_UniversalTime::year

8.422 nas_wcdmaCellInfo Struct Reference

Data Fields

- uint16_t [psc](#)
- int16_t [cpich_rscp](#)
- int16_t [cpich_ecno](#)
- int16_t [srxlev](#)

8.422.1 Detailed Description

This structure contains information about the WCDMA Cell.

Parameters

<i>psc</i>	<ul style="list-style-type: none"> Primary scrambling code. Range: 0 to 511.
<i>cpich_rscp</i>	<ul style="list-style-type: none"> Absolute power level (in 1/10 dBm) of the common pilot channel as received by the UE. Range: -120.0 dBm to -25.0 dBm
<i>cpich_ecno</i>	<ul style="list-style-type: none"> CPICH Ec/No; ratio (in 1/10 dB) of the received energy per PN chip for the CPICH to the total received power spectral density at the UE antenna connector. Range: -50.0 dB to 0.
<i>srxlev</i>	<ul style="list-style-type: none"> Cell selection Rx level (Srxlev) value. Range: -128 to 128. This field is only valid when ue_in_idle is TRUE.

8.422.2 Field Documentation

8.422.2.1 int16_t nas_wcdmaCellInfo::cpich_ecno

8.422.2.2 int16_t nas_wcdmaCellInfo::cpich_rscp

8.422.2.3 uint16_t nas_wcdmaCellInfo::psc

8.422.2.4 int16_t nas_wcdmaCellInfo::srxlev

8.423 nas_WCDMAECIOThresh Struct Reference

Data Fields

- uint8_t [WCDMAECIOThreshListLen](#)
- int16_t * [pWCDMAECIOThreshList](#)

8.423.1 Detailed Description

This structure contains WCDMA ECIO threshold related parameters.

Parameters

<i>WCDMAECIO</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> Length of the WCDMA ECIO threshold list parameter to follow
<i>pWCDMAECIO</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> Array of ECIO thresholds (in units of 0.1 dB) Maximum of 32 values Range for ECIO values: -31.5 to 0 (in dB)

8.423.2 Field Documentation

8.423.2.1 int16_t* nas_WCDMAECIOThresh::pWCDMAECIOThreshList

8.423.2.2 uint8_t nas_WCDMAECIOThresh::WCDMAECIOThreshListLen

8.424 nas_WCDMAInfoLTENeighborCell Struct Reference

Data Fields

- uint32_t [wcdmaRRState](#)
- uint8_t [umtsLTENbrCellLen](#)
- [nas_umtsLTENbrCell](#) [UMTSLTENbrCell](#) [255]

8.424.1 Detailed Description

This structure contains information about the WCDMA - LTE Neighboring Cell Info Set.

Parameters

<i>wcdmaRR</i> ↔ <i>State</i>	<ul style="list-style-type: none"> WCDMA RRC states. Defined in 3GPP TS 25.331 Values: <ul style="list-style-type: none"> 0x00 - NAS_WCDMA_RRC_STATE_DISCONNECTED * WCDMA RRC State is IDLE 0x01 - NAS_WCDMA_RRC_STATE_CELL_PCH * WCDMA RRC state is CELL_PCH 0x02 - NAS_WCDMA_RRC_STATE_URA_PCH * WCDMA RRC state is URA_PCH 0x03 - NAS_WCDMA_RRC_STATE_CELL_FACH * WCDMA RRC state is CELL_FACH 0x04 - NAS_WCDMA_RRC_STATE_CELL_DCH * WCDMA RRC state is CELL_DCH
<i>umtsLTENbr</i> ↔ <i>CellLen</i>	<ul style="list-style-type: none"> Number of sets of UMTS LTE Neighbors.
<i>UMTSLTENbr</i> ↔ <i>Cell</i>	
Generated by Doxygen	<ul style="list-style-type: none"> See nas_umtsLTENbrCell for more information.

8.424.2 Field Documentation

8.424.2.1 `nas_umtsLTENbrCell` `nas_WCDMAInfoLTENeighborCell::UMTSLTENbrCell[255]`

8.424.2.2 `uint8_t` `nas_WCDMAInfoLTENeighborCell::umtsLTENbrCellLen`

8.424.2.3 `uint32_t` `nas_WCDMAInfoLTENeighborCell::wcdmaRRState`

8.425 `nas_WCDMARSSIThresh` Struct Reference

Data Fields

- `uint8_t` [WCDMARSSIThreshListLen](#)
- `int16_t *` [pWCDMARSSIThreshList](#)

8.425.1 Detailed Description

This structure contains WCDMA RSSI threshold related parameters.

Parameters

<i>WCDMARSSIThreshListLen</i>	<ul style="list-style-type: none"> • Length of the WCDMA RSSI threshold list parameter to follow
<i>pWCDMARSSIThreshList</i>	<ul style="list-style-type: none"> • Array of RSSI thresholds (in units of 0.1 dBm) • Maximum of 32 values. • Range for RSSI values: -121 to 0 (in dBm)

8.425.2 Field Documentation

8.425.2.1 `int16_t *` `nas_WCDMARSSIThresh::pWCDMARSSIThreshList`

8.425.2.2 `uint8_t` `nas_WCDMARSSIThresh::WCDMARSSIThreshListLen`

8.426 `nas_WCDMASysInfo` Struct Reference

Data Fields

- `nas_sysInfoCommon` [sysInfoWCDMA](#)
- `uint8_t` [lacValid](#)
- `uint16_t` [lac](#)
- `uint8_t` [cellIdValid](#)
- `uint32_t` [cellId](#)
- `uint8_t` [regRejectInfoValid](#)

- uint8_t [rejectSrvDomain](#)
- uint8_t [rejCause](#)
- uint8_t [networkIdValid](#)
- uint8_t [MCC](#) [3]
- uint8_t [MNC](#) [3]
- uint8_t [hsCallStatusValid](#)
- uint8_t [hsCallStatus](#)
- uint8_t [hsIndValid](#)
- uint8_t [hsInd](#)
- uint8_t [pscValid](#)
- uint16_t [psc](#)

8.426.1 Detailed Description

Structure for storing the WCDMA System Information.

Parameters

<i>sysInfoWCDMA</i>	<ul style="list-style-type: none"> • See sysInfoCommon for more information.
<i>lacValid</i>	<ul style="list-style-type: none"> • Indicates whether the location area code is valid.. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>lac</i>	<ul style="list-style-type: none"> • Location area code. • Only applies to 3GPP. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>cellIdValid</i>	<ul style="list-style-type: none"> • Indicates whether the cell ID is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>cellId</i>	<ul style="list-style-type: none"> • Cell ID. <ul style="list-style-type: none"> – 0xFFFFFFFF - Not Available
<i>regRejectInfo</i> ↔ <i>Valid</i>	<ul style="list-style-type: none"> • Indicates whether the registration reject information is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

<i>rejectSrvDomain</i>	<ul style="list-style-type: none"> Type of service domain in which the registration is rejected. <ul style="list-style-type: none"> 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service 0x01 - Circuit-switched only 0x02 - Packet-switched only 0x03 - Circuit-switched and packet-switched 0x04 - Camped 0xFF - Not Available
<i>rejCause</i>	<ul style="list-style-type: none"> Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6]. <ul style="list-style-type: none"> 0xFF - Not Available
<i>networkIdValid</i>	<ul style="list-style-type: none"> Indicates whether the network ID is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>MCC[PLMN_L↔ENGTH]</i>	<ul style="list-style-type: none"> Mobile Country Code. MCC digits in ASCII characters
<i>MNC[PLMN_L↔ENGTH]</i>	<ul style="list-style-type: none"> Mobile Network Code. MNC digits in ASCII characters An unused byte is set to 0xFF. In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.
<i>hsCallStatus↔Valid</i>	<ul style="list-style-type: none"> Indicates whether the high-speed call status is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>hsCallStatus</i>	<ul style="list-style-type: none"> Call status on high speed. Only applicable for WCDMA. <ul style="list-style-type: none"> 0x00 - HSDPA and HSUPA are unsupported 0x01 - HSDPA is supported 0x02 - HSUPA is supported 0x03 - HSDPA and HSUPA are supported 0x04 - HSDPA+ is supported 0x05 - HSDPA+ and HSUPA are supported 0x06 - Dual-cell HSDPA+ is supported 0x07 - Dual-cell HSDPA+ and HSUPA are supported 0xFF - Not Available

<i>hsIndValid</i>	<ul style="list-style-type: none"> Indicates whether high-speed service indication is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>hsInd</i>	<ul style="list-style-type: none"> High-speed service indication Only applicable for WCDMA. <ul style="list-style-type: none"> 0x00 - HSDPA and HSUPA are unsupported 0x01 - HSDPA is supported 0x02 - HSUPA is supported 0x03 - HSDPA and HSUPA are supported 0x04 - HSDPA+ is supported 0x05 - HSDPA+ and HSUPA are supported 0x06 - Dual-cell HSDPA+ is supported 0x07 - Dual-cell HSDPA+ and HSUPA are supported 0xFF - Not Available
<i>pscValid</i>	<ul style="list-style-type: none"> Indicates whether primary scrambling code is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>psc</i>	<ul style="list-style-type: none"> Primary scrambling code. <ul style="list-style-type: none"> 0xFFFF - Not Available

8.426.2 Field Documentation

8.426.2.1 `uint32_t nas_WCDMASysInfo::cellId`

8.426.2.2 `uint8_t nas_WCDMASysInfo::cellIdValid`

8.426.2.3 `uint8_t nas_WCDMASysInfo::hsCallStatus`

8.426.2.4 `uint8_t nas_WCDMASysInfo::hsCallStatusValid`

8.426.2.5 `uint8_t nas_WCDMASysInfo::hsInd`

8.426.2.6 `uint8_t nas_WCDMASysInfo::hsIndValid`

8.426.2.7 `uint16_t nas_WCDMASysInfo::lac`

8.426.2.8 `uint8_t nas_WCDMASysInfo::lacValid`

- 8.426.2.9 `uint8_t nas_WCDMASysInfo::MCC[3]`
- 8.426.2.10 `uint8_t nas_WCDMASysInfo::MNC[3]`
- 8.426.2.11 `uint8_t nas_WCDMASysInfo::networkIdValid`
- 8.426.2.12 `uint16_t nas_WCDMASysInfo::psc`
- 8.426.2.13 `uint8_t nas_WCDMASysInfo::pscValid`
- 8.426.2.14 `uint8_t nas_WCDMASysInfo::regRejectInfoValid`
- 8.426.2.15 `uint8_t nas_WCDMASysInfo::rejCause`
- 8.426.2.16 `uint8_t nas_WCDMASysInfo::rejectSrvDomain`
- 8.426.2.17 `nas_sysInfoCommon nas_WCDMASysInfo::sysInfoWCDMA`

8.427 NASBandPreferenceTlv Struct Reference

Data Fields

- `uint8_t TlvPresent`
- `uint64_t band_pref`

8.427.1 Field Documentation

- 8.427.1.1 `uint64_t NASBandPreferenceTlv::band_pref`
- 8.427.1.2 `uint8_t NASBandPreferenceTlv::TlvPresent`

8.428 nasCellLocationInfoResp Struct Reference

Data Fields

- `GERANInfo * pGERANInfo`
- `UMTSInfo * pUMTSInfo`
- `CDMAInfo * pCDMAInfo`
- `LTEInfoIntrafreq * pLTEInfoIntrafreq`
- `LTEInfoInterfreq * pLTEInfoInterfreq`
- `LTEInfoNeighboringGSM * pLTEInfoNeighboringGSM`
- `LTEInfoNeighboringWCDMA * pLTEInfoNeighboringWCDMA`
- `ULONG * pUMTSCellID`
- `WCDMAInfoLTENeighborCell * pWCDMAInfoLTENeighborCell`

8.428.1 Detailed Description

This structure contains information about the Get Cell Location response parameters.

Parameters

<i>pGERANInfo</i>	<ul style="list-style-type: none"> See GERANInfo for more information.
<i>pUMTSInfo</i>	<ul style="list-style-type: none"> See UMTSInfo for more information.
<i>pCDMAInfo</i>	<ul style="list-style-type: none"> See CDMAInfo for more information.
<i>pLTEInfoIntrafreq</i>	<ul style="list-style-type: none"> See LTEInfoIntrafreq for more information.
<i>pLTEInfoInterfreq</i>	<ul style="list-style-type: none"> See LTEInfoInterfreq for more information.
<i>pLTEInfoNeighboringGSM</i>	<ul style="list-style-type: none"> See LTEInfoNeighboringGSM for more information.
<i>pLTEInfoNeighboringWCDMA</i>	<ul style="list-style-type: none"> See LTEInfoNeighboringWCDMA for more information.
<i>pUMTSCellID</i>	<ul style="list-style-type: none"> Cell ID. 0xFFFFFFFF indicates cell ID information is not present.
<i>pWCDMAInfoLTENeighborCell</i>	<ul style="list-style-type: none"> See WCDMAInfoLTENeighborCell for more information.

8.428.2 Field Documentation

8.428.2.1 **CDMAInfo*** nasCellLocationInfoResp::pCDMAInfo

8.428.2.2 **GERANInfo*** nasCellLocationInfoResp::pGERANInfo

8.428.2.3 **LTEInfoInterfreq*** nasCellLocationInfoResp::pLTEInfoInterfreq

8.428.2.4 **LTEInfoIntrafreq*** nasCellLocationInfoResp::pLTEInfoIntrafreq

8.428.2.5 **LTEInfoNeighboringGSM*** nasCellLocationInfoResp::pLTEInfoNeighboringGSM

8.428.2.6 **LTEInfoNeighboringWCDMA*** nasCellLocationInfoResp::pLTEInfoNeighboringWCDMA

8.428.2.7 **ULONG*** nasCellLocationInfoResp::pUMTSCellID

8.428.2.8 **UMTSInfo*** nasCellLocationInfoResp::pUMTSInfo

8.428.2.9 **WCDMAInfoLTENeighborCell*** nasCellLocationInfoResp::pWCDMAInfoLTENeighborCell

8.429 NASEmergencyModeTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- uint8_t [EmerMode](#)

8.429.1 Field Documentation

8.429.1.1 `uint8_t` `NASEmergencyModeTlv::EmerMode`

8.429.1.2 `uint8_t` `NASEmergencyModeTlv::TlvPresent`

8.430 nasGet3GPP2SubscriptionInfoReq Struct Reference

Data Fields

- [BYTE](#) `namID`

8.430.1 Detailed Description

This structure contains the `Get3GPP2SubscriptionInfo` request parameters

Parameters

<i>namID</i>	[Mandatory] <ul style="list-style-type: none"> • NAM ID of the information to be retrieved. The index starts from 0. A <code>nam_id</code> of 0xFF is used to retrieve information of current NAM.
--------------	---

8.430.2 Field Documentation

8.430.2.1 `BYTE` `nasGet3GPP2SubscriptionInfoReq::namID`

8.431 nasGet3GPP2SubscriptionInfoResp Struct Reference

Data Fields

- `namName` * [pNAMNameInfo](#)
- `dirNum` * [pDirNum](#)
- `homeSIDNID` * [pHomeSIDNID](#)
- `minBasedIMSI` * [pMinBasedIMSI](#)
- `trueIMSI` * [pTrueIMSI](#)
- `CDMAChannel` * [pCDMAChannel](#)

8.431.1 Detailed Description

This structure contains the `SLQSNasGet3GPP2Subscription` response parameters.

Parameters

<i>pNAMNameInfo</i>	[Optional] • See namName for more information
<i>pDirNum</i>	[Optional] • See dirNum for more information
<i>pHomeSIDNID</i>	[Optional] • See homeSIDNID for more information
<i>pMinBasedIMSI</i>	[Optional] • See minBasedIMSI for more information
<i>pTrueIMSI</i>	[Optional] • See trueIMSI for more information
<i>pCDMAChannel</i>	[Optional] • See CDMAChannel for more information

8.431.2 Field Documentation

8.431.2.1 **CDMAChannel*** nasGet3GPP2SubscriptionInfoResp::pCDMAChannel8.431.2.2 **dirNum*** nasGet3GPP2SubscriptionInfoResp::pDirNum8.431.2.3 **homeSIDNID*** nasGet3GPP2SubscriptionInfoResp::pHomeSIDNID8.431.2.4 **minBasedIMSI*** nasGet3GPP2SubscriptionInfoResp::pMinBasedIMSI8.431.2.5 **namName*** nasGet3GPP2SubscriptionInfoResp::pNAMNameInfo8.431.2.6 **trueIMSI*** nasGet3GPP2SubscriptionInfoResp::pTrueIMSI

8.432 nasGetHDRColorCodeResp Struct Reference

Data Fields

- **BYTE *** [pColorCode](#)

8.432.1 Detailed Description

Structure for storing the current preferred system selection settings for the device.

Parameters

<i>pColorCode</i>	[Optional] • Color code value
Generated by Doxygen	<ul style="list-style-type: none"> • Color code corresponding to the sector to which the AT is sending the access probe • See 3GPP2 C.S0024-B V3.0, Section 7.11.6.2.1 for more information. <ul style="list-style-type: none"> – 0xFF - Not Available

8.432.2 Field Documentation

8.432.2.1 **BYTE*** nasGetHDRColorCodeResp::pColorCode

8.433 nasGetLTECphyCa Struct Reference

Data Fields

- [PhyCaAggScellIndType](#) sPhyCaAggScellIndType
- [PhyCaAggScellDIBw](#) sPhyCaAggScellDIBw
- [PhyCaAggScellInfo](#) sPhyCaAggScellInfo
- [PhyCaAggPcellInfo](#) sPhyCaAggPcellInfo
- [PhyCaAggScellIndex](#) sPhyCaAggScellIndex

8.433.1 Field Documentation

8.433.1.1 **PhyCaAggPcellInfo** nasGetLTECphyCa::sPhyCaAggPcellInfo

8.433.1.2 **PhyCaAggScellDIBw** nasGetLTECphyCa::sPhyCaAggScellDIBw

8.433.1.3 **PhyCaAggScellIndex** nasGetLTECphyCa::sPhyCaAggScellIndex

8.433.1.4 **PhyCaAggScellIndType** nasGetLTECphyCa::sPhyCaAggScellIndType

8.433.1.5 **PhyCaAggScellInfo** nasGetLTECphyCa::sPhyCaAggScellInfo

8.434 NasGetLTECphyCaInfo Struct Reference

Data Fields

- [NASPhyCaAggScellIndType](#) PhyCaAggScellIndType
- [NASPhyCaAggScellDIBw](#) PhyCaAggScellDIBw
- [NASPhyCaAggScellInfo](#) PhyCaAggScellInfo
- [NASPhyCaAggPcellInfo](#) PhyCaAggPcellInfo
- [NASPhyCaAggScellIndex](#) PhyCaAggScellIndex

8.434.1 Field Documentation

8.434.1.1 **NASPhyCaAggPcellInfo** NasGetLTECphyCaInfo::PhyCaAggPcellInfo

8.434.1.2 **NASPhyCaAggScellDIBw** NasGetLTECphyCaInfo::PhyCaAggScellDIBw

8.434.1.3 **NASPhyCaAggScellIndex** NasGetLTECphyCaInfo::PhyCaAggScellIndex

8.434.1.4 **NASPhyCaAggScellIndType** NasGetLTECphyCaInfo::PhyCaAggScellIndType

8.434.1.5 **NASPhyCaAggScellInfo** NasGetLTECphyCaInfo::PhyCaAggScellInfo

8.435 nasGetLTECphyCaResp Struct Reference

Data Fields

- [PhyCaAggScellIndType](#) * pPhyCaAggScellIndType
- [PhyCaAggScellDIBw](#) * pPhyCaAggScellDIBw
- [PhyCaAggScellInfo](#) * pPhyCaAggScellInfo
- [PhyCaAggPcellInfo](#) * pPhyCaAggPcellInfo
- [PhyCaAggScellIndex](#) * pPhyCaAggScellIndex

8.435.1 Field Documentation

8.435.1.1 **PhyCaAggPcellInfo*** nasGetLTECphyCaResp::pPhyCaAggPcellInfo

8.435.1.2 **PhyCaAggScellIDIBw*** nasGetLTECphyCaResp::pPhyCaAggScellIDIBw

8.435.1.3 **PhyCaAggScellIndex*** nasGetLTECphyCaResp::pPhyCaAggScellIndex

8.435.1.4 **PhyCaAggScellIndType*** nasGetLTECphyCaResp::pPhyCaAggScellIndType

8.435.1.5 **PhyCaAggScellInfo*** nasGetLTECphyCaResp::pPhyCaAggScellInfo

8.436 nasGetSigInfoResp Struct Reference

Data Fields

- [CDMASSInfo](#) * [pCDMASSInfo](#)
- [HDRSSInfo](#) * [pHDRSSInfo](#)
- [INT8](#) * [pGSMSSInfo](#)
- [CDMASSInfo](#) * [pWCDMASSInfo](#)
- [LTESSInfo](#) * [pLTESSInfo](#)
- [INT8](#) * [pTDSCDMASigInfoRscp](#)
- [TDSCDMASigInfoExt](#) * [pTDSCDMASigInfoExt](#)

8.436.1 Detailed Description

This structure contains the SLQSNasGetSigInfo response parameters.

Parameters

<i>pCDMASSInfo</i>	[Optional] • See CDMASSInfo for more information
<i>pHDRSSInfo</i>	[Optional] • See HDRSSInfo for more information
<i>pGSMSSInfo</i>	[Optional] • GSM signal strength is the RSSI in dBm (signed value). • A value of -125 dBm or lower is used to indicate No Signal.
<i>pWCDMASSInfo</i>	[Optional] • See CDMASSInfo for more information
<i>pLTESSInfo</i>	[Optional] • See LTESSInfo for more information
<i>pTDSCDMA↔ SigInfoRscp</i>	[Optional] • RSCP of the Primary Common Control Physical Channel (PCCPCH) in dBm. • Measurement range: -120 dBm to -25 dBm.
<i>pTDSCDMA↔ SigInfoExt</i>	[Optional] • See TDSCDMASigInfoExt for more information.

8.436.2 Field Documentation

8.436.2.1 **CDMASSInfo*** nasGetSigInfoResp::pCDMASSInfo

8.436.2.2 **INT8*** nasGetSigInfoResp::pGSMSSInfo

8.436.2.3 **HDRSSInfo*** nasGetSigInfoResp::pHDRSSInfo

8.436.2.4 **LTESSInfo*** nasGetSigInfoResp::pLTESSInfo

8.436.2.5 **TDSCDMASigInfoExt*** nasGetSigInfoResp::pTDSCDMASigInfoExt

8.436.2.6 **INT8*** nasGetSigInfoResp::pTDSCDMASigInfoRscp

8.436.2.7 **CDMASSInfo*** nasGetSigInfoResp::pWCDMASSInfo

8.437 nasGetSysInfoResp Struct Reference

Data Fields

- [SrvStatusInfo](#) * [pCDMASrvStatusInfo](#)
- [SrvStatusInfo](#) * [pHDRSrvStatusInfo](#)
- [GSMSrvStatusInfo](#) * [pGSMSrvStatusInfo](#)
- [GSMSrvStatusInfo](#) * [pWCDMASrvStatusInfo](#)
- [GSMSrvStatusInfo](#) * [pLTESrvStatusInfo](#)
- [CDMASysInfo](#) * [pCDMASysInfo](#)
- [HDRSysInfo](#) * [pHDRSysInfo](#)
- [GSMSysInfo](#) * [pGSMSysInfo](#)
- [WCDMASysInfo](#) * [pWCDMASysInfo](#)
- [LTESysInfo](#) * [pLTESysInfo](#)
- [AddCDMASysInfo](#) * [pAddCDMASysInfo](#)
- [WORD](#) * [pAddHDRSysInfo](#)
- [AddSysInfo](#) * [pAddGSMSysInfo](#)
- [AddSysInfo](#) * [pAddWCDMASysInfo](#)
- [WORD](#) * [pAddLTESysInfo](#)
- [CallBarringSysInfo](#) * [pGSMCallBarringSysInfo](#)
- [CallBarringSysInfo](#) * [pWCDMACallBarringSysInfo](#)
- [BYTE](#) * [pLTEVoiceSupportSysInfo](#)
- [BYTE](#) * [pGSMCipherDomainSysInfo](#)
- [BYTE](#) * [pWCDMACipherDomainSysInfo](#)

8.437.1 Detailed Description

Structure for storing the SLQSNasGetSysInfo response parameters.

Parameters

<i>pCDMASrv↔ StatusInfo</i>	<ul style="list-style-type: none"> See SrvStatusInfo for more information.
<i>pHDRSrv↔ StatusInfo</i>	<ul style="list-style-type: none"> See SrvStatusInfo for more information.
<i>pGSMSrv↔ StatusInfo</i>	<ul style="list-style-type: none"> See GSMSrvStatusInfo for more information.
<i>pWCDMASrv↔ StatusInfo</i>	<ul style="list-style-type: none"> See GSMSrvStatusInfo for more information.
<i>pLTESrv↔ StatusInfo</i>	<ul style="list-style-type: none"> See GSMSrvStatusInfo for more information.
<i>pCDMASysInfo</i>	<ul style="list-style-type: none"> See CDMASysInfo for more information.
<i>pHDRSysInfo</i>	<ul style="list-style-type: none"> See HDRSysInfo for more information.
<i>pGSMSysInfo</i>	<ul style="list-style-type: none"> See GSMSysInfo for more information.
<i>pWCDMASys↔ Info</i>	<ul style="list-style-type: none"> See WCDMASysInfo for more information.
<i>pLTESysInfo</i>	<ul style="list-style-type: none"> See LTESysInfo for more information.
<i>pAddCDMA↔ SysInfo</i>	<ul style="list-style-type: none"> See AddCDMASysInfo for more information.
<i>pAddHDRSys↔ Info</i>	<ul style="list-style-type: none"> System table index referencing the beginning of the geo in which the current serving system is present. When the system index is not known, 0xFFFF is used.
<i>pAddGSMSys↔ Info</i>	<ul style="list-style-type: none"> See AddSysInfo for more information.
<i>pAddWCDMA↔ SysInfo</i>	<ul style="list-style-type: none"> See AddSysInfo for more information.
<i>pAddLTESysInfo</i>	<ul style="list-style-type: none"> System table index referencing the beginning of the geo in which the current serving system is present. When the system index is not known, 0xFFFF is used.

<i>pGSMCall</i> ↔ <i>BarringSysInfo</i>	<ul style="list-style-type: none"> • See CallBarringSysInfo for more information.
<i>pWCDMACall</i> ↔ <i>BarringSysInfo</i>	<ul style="list-style-type: none"> • See CallBarringSysInfo for more information.
<i>pLTEVoice</i> ↔ <i>SupportSysInfo</i>	<ul style="list-style-type: none"> • Indicates voice support status on LTE. <ul style="list-style-type: none"> – 0x00 - Voice is not supported – 0x01 - Voice is supported
<i>pGSMCipher</i> ↔ <i>DomainSysInfo</i>	<ul style="list-style-type: none"> • Ciphering on the service domain. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched
<i>pWCDMA</i> ↔ <i>CipherDomain</i> ↔ <i>SysInfo</i>	<ul style="list-style-type: none"> • Ciphering on the service domain. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched

8.437.2 Field Documentation

8.437.2.1 **AddCDMASysInfo*** nasGetSysInfoResp::pAddCDMASysInfo

8.437.2.2 **AddSysInfo*** nasGetSysInfoResp::pAddGSMSysInfo

8.437.2.3 **WORD*** nasGetSysInfoResp::pAddHDRSysInfo

8.437.2.4 **WORD*** nasGetSysInfoResp::pAddLTSysInfo

8.437.2.5 **AddSysInfo*** nasGetSysInfoResp::pAddWCDMASysInfo

8.437.2.6 **SrvStatusInfo*** nasGetSysInfoResp::pCDMASrvStatusInfo

8.437.2.7 **CDMASysInfo*** nasGetSysInfoResp::pCDMASysInfo

8.437.2.8 **CallBarringSysInfo*** nasGetSysInfoResp::pGSMCallBarringSysInfo

8.437.2.9 **BYTE*** nasGetSysInfoResp::pGSMCipherDomainSysInfo

8.437.2.10 **GSMSrvStatusInfo*** nasGetSysInfoResp::pGSMSrvStatusInfo

- 8.437.2.11 **GSMSysInfo*** nasGetSysInfoResp::pGSMSysInfo
- 8.437.2.12 **SrvStatusInfo*** nasGetSysInfoResp::pHDRSrvStatusInfo
- 8.437.2.13 **HDRSysInfo*** nasGetSysInfoResp::pHDRSysInfo
- 8.437.2.14 **GSMSrvStatusInfo*** nasGetSysInfoResp::pLTESrvStatusInfo
- 8.437.2.15 **LTESysInfo*** nasGetSysInfoResp::pLTESysInfo
- 8.437.2.16 **BYTE*** nasGetSysInfoResp::pLTEVoiceSupportSysInfo
- 8.437.2.17 **CallBarringSysInfo*** nasGetSysInfoResp::pWCDMACallBarringSysInfo
- 8.437.2.18 **BYTE*** nasGetSysInfoResp::pWCDMACipherDomainSysInfo
- 8.437.2.19 **GSMSrvStatusInfo*** nasGetSysInfoResp::pWCDMASrvStatusInfo
- 8.437.2.20 **WCDMASysInfo*** nasGetSysInfoResp::pWCDMASysInfo

8.438 nasGetTxRxInfoReq Struct Reference

Data Fields

- [BYTE](#) `radio_if`

8.438.1 Detailed Description

This structure contains the GetTxRxInfoReq request parameters

Parameters

<i>radio_if</i>	<div><div>[Mandatory]</div><ul style="list-style-type: none">• Radio interface technology of the signal being measured• Valid Values<ul style="list-style-type: none">– 0x01 - NAS_RADIO_IF_CDMA_1X - CDMA– 0x02 - NAS_RADIO_IF_CDMA_1XEVD0 - HDR– 0x04 - NAS_RADIO_IF_GSM - GSM– 0x05 - NAS_RADIO_IF_UMTS - UMTS– 0x08 - NAS_RADIO_IF_LTE - LTE</div>
-----------------	---

8.438.2 Field Documentation

- 8.438.2.1 **BYTE** nasGetTxRxInfoReq::radio_if

8.439 nasGetTxRxInfoResp Struct Reference

Data Fields

- [rxInfo](#) * [pRXChain0Info](#)
- [rxInfo](#) * [pRXChain1Info](#)
- [txInfo](#) * [pTXInfo](#)

8.439.1 Detailed Description

This structure contains the GetTxRxInfoResp response parameters.

Parameters

<i>pRXChain0Info</i>	[Optional] • See rxInfo for more information.
<i>pRXChain1Info</i>	[Optional] • See rxInfo for more information.
<i>pTXInfo</i>	[Optional] • See txInfo for more information.

8.439.2 Field Documentation

8.439.2.1 [rxInfo](#)* [nasGetTxRxInfoResp::pRXChain0Info](#)

8.439.2.2 [rxInfo](#)* [nasGetTxRxInfoResp::pRXChain1Info](#)

8.439.2.3 [txInfo](#)* [nasGetTxRxInfoResp::pTXInfo](#)

8.440 NASGWAcqOrderPrefTlv Struct Reference

Data Fields

- [uint8_t](#) [TlvPresent](#)
- [uint32_t](#) [GWAcqOrderPref](#)

8.440.1 Field Documentation

8.440.1.1 [uint32_t](#) [NASGWAcqOrderPrefTlv::GWAcqOrderPref](#)

8.440.1.2 [uint8_t](#) [NASGWAcqOrderPrefTlv::TlvPresent](#)

8.441 nasIndicationRegisterReq Struct Reference

Data Fields

- [BYTE](#) * [pSystemSelectionInd](#)

- BYTE * pDDTMInd
- BYTE * pServingSystemInd
- BYTE * pDualStandByPrefInd
- BYTE * pSubscriptionInfoInd
- BYTE * pNetworkTimeInd
- BYTE * pSysInfoInd
- BYTE * pSignalStrengthInd
- BYTE * pErrorRateInd
- BYTE * pHDRNewUATIAssInd
- BYTE * pHDRSessionCloseInd
- BYTE * pManagedRoamingInd
- BYTE * pLTECphyCa

8.441.1 Detailed Description

This structure contains the SLQSNasIndicationRegisterExt request parameters.

Parameters

<i>pSystem↔ SelectionInd</i>	[Optional] <ul style="list-style-type: none"> System Selection Preference indication registration. The following callbacks would not be invoked if the indication is disabled. tFNRoamingIndicator tFNDataCapabilities and tFNServingSystem <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pDDTMInd</i>	[Optional] <ul style="list-style-type: none"> DDTM (Data Dedicated Transmission Mode) indication registration. The following callbacks would not be invoked if the indication is disabled. tFNDDTM <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pServing↔ SystemInd</i>	[Optional] <ul style="list-style-type: none"> Serving System indication registration. The following callbacks would not be invoked if the indication is disabled. tFNBandPreference <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pDualStandBy↔ PrefInd</i>	[Optional] <ul style="list-style-type: none"> Dual Standby Preference indication registration. The following callbacks would not be invoked if the indication is disabled. tFNDualStandByPref <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pSubscription↔ InfoInd</i>	[Optional] <ul style="list-style-type: none"> Subscription Information indication registration. The following callbacks would not be invoked if the indication is disabled. tFNSubscriptionInfo <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable

<i>pNetworkTime</i> ↔ <i>Ind</i>	[Optional] <ul style="list-style-type: none"> Network Time indication registration. The following callbacks would not be invoked if the indication is disabled. tFNNetworkTime <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pSysInfoInd</i>	[Optional] <ul style="list-style-type: none"> System Information indication registration. The following callbacks would not be invoked if the indication is disabled. tFNSysInfo <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pSignal</i> ↔ <i>StrengthInd</i>	[Optional] <ul style="list-style-type: none"> Signal Strength indication registration. The following callbacks would not be invoked if the indication is disabled. tFNSigInfo <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pErrorRateInd</i>	[Optional] <ul style="list-style-type: none"> Error Rate indication registration. The following callbacks would not be invoked if the indication is disabled. tFNErrRate <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pHDRNewUA</i> ↔ <i>TIAssInd</i>	[Optional] <ul style="list-style-type: none"> HDR New UATI Assigned indication registration. The following callbacks would not be invoked if the indication is disabled. tFNHDRUATIUpdate <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pHDRSession</i> ↔ <i>CloseInd</i>	[Optional] <ul style="list-style-type: none"> HDR Session Closed indication registration. The following callbacks would not be invoked if the indication is disabled. tFNHDRSessionClose <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pManaged</i> ↔ <i>RoamingInd</i>	[Optional] <ul style="list-style-type: none"> Managed Roaming indication registration. The following callbacks would not be invoked if the indication is disabled. tFNManagedRoaming <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable

Note

Atleast one parameter must be provided as request. 'NULL' value confirms that the indication value is not sent.

8.441.2 Field Documentation

- 8.441.2.1 **BYTE*** nasIndicationRegisterReq::pDDTMInd
- 8.441.2.2 **BYTE*** nasIndicationRegisterReq::pDualStandByPrefInd
- 8.441.2.3 **BYTE*** nasIndicationRegisterReq::pErrorRateInd
- 8.441.2.4 **BYTE*** nasIndicationRegisterReq::pHDRNewUATIAssInd
- 8.441.2.5 **BYTE*** nasIndicationRegisterReq::pHDRSessionCloseInd
- 8.441.2.6 **BYTE*** nasIndicationRegisterReq::pLTECphyCa
- 8.441.2.7 **BYTE*** nasIndicationRegisterReq::pManagedRoamingInd
- 8.441.2.8 **BYTE*** nasIndicationRegisterReq::pNetworkTimeInd
- 8.441.2.9 **BYTE*** nasIndicationRegisterReq::pServingSystemInd
- 8.441.2.10 **BYTE*** nasIndicationRegisterReq::pSignalStrengthInd
- 8.441.2.11 **BYTE*** nasIndicationRegisterReq::pSubscriptionInfoInd
- 8.441.2.12 **BYTE*** nasIndicationRegisterReq::pSysInfoInd
- 8.441.2.13 **BYTE*** nasIndicationRegisterReq::pSystemSelectionInd

8.442 nasInitNetworkReg Struct Reference

Data Fields

- [ULONG](#) regAction
- [MNRInfo](#) * pMNRInfo
- [ULONG](#) * pChangeDuration
- [BOOL](#) * pMncPcsDigitStatus

8.442.1 Detailed Description

This structure contains Initiate Network Registration request parameters

Parameters

<i>regAction</i>	<ul style="list-style-type: none"> Specifies one of the following register actions : <ul style="list-style-type: none"> AUTO_REGISTER - Device registers according to its provisioning and optional parameters supplied with the command are ignored. MANUAL_REGISTER - Device registers to a specified network and the optional Manual Network Register Information parameter pMNRInfo must also be included for the command to process successfully and supported only for 3GPP.
<i>pMNRInfo</i>	[Optional] <ul style="list-style-type: none"> Pointer to structure MNRInfo <ul style="list-style-type: none"> See MNRInfo for more information
<i>pChange↔ Duration</i>	[Optional] <ul style="list-style-type: none"> Duration of the change. <ul style="list-style-type: none"> 0x00 - Power cycle - Remains active until the next device power cycle 0x01 - Permanent - Remains active through power cycles until changed by the client
<i>pMncPcsDigit↔ Status</i>	[Optional] <ul style="list-style-type: none"> MNC PCS Digit Include Status <ul style="list-style-type: none"> True - MNC is a 3-digit value. False - MNC is a 2-digit value.

8.442.2 Field Documentation

8.442.2.1 **ULONG*** nasInitNetworkReg::pChangeDuration

8.442.2.2 **BOOL*** nasInitNetworkReg::pMncPcsDigitStatus

8.442.2.3 **MNRInfo*** nasInitNetworkReg::pMNRInfo

8.442.2.4 **ULONG** nasInitNetworkReg::regAction

8.443 NASLTEBandPreferenceTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- uint64_t [LTEBandPref](#)

8.443.1 Field Documentation

8.443.1.1 uint64_t NASLTEBandPreferenceTlv::LTEBandPref

8.443.1.2 uint8_t NASLTEBandPreferenceTlv::TlvPresent

8.444 NASLteNasReleaseInfoTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- uint8_t [nas_release](#)
- uint8_t [nas_major](#)
- uint8_t [nas_minor](#)

8.444.1 Field Documentation

8.444.1.1 uint8_t NASLteNasReleaseInfoTlv::nas_major

8.444.1.2 uint8_t NASLteNasReleaseInfoTlv::nas_minor

8.444.1.3 uint8_t NASLteNasReleaseInfoTlv::nas_release

8.444.1.4 uint8_t NASLteNasReleaseInfoTlv::TlvPresent

8.445 NASModePreferenceTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- uint16_t [ModePref](#)

8.445.1 Field Documentation

8.445.1.1 uint16_t NASModePreferenceTlv::ModePref

8.445.1.2 uint8_t NASModePreferenceTlv::TlvPresent

8.446 NASNetSelPreferenceTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- uint8_t [NetSelPref](#)

8.446.1 Field Documentation

8.446.1.1 uint8_t NASNetSelPreferenceTlv::NetSelPref

8.446.1.2 uint8_t NASNetSelPreferenceTlv::TlvPresent

8.447 nasNetworkTime Struct Reference

Data Fields

- [UniversalTime](#) universalTime
- [BYTE](#) * pTimeZone
- [BYTE](#) * pDayltSavAdj
- [BYTE](#) * pRadioInterface

8.447.1 Detailed Description

Structure for storing the [nasSysInfo](#) indication parameters.

Parameters

<i>universalTime</i>	<ul style="list-style-type: none"> • See UniversalTime for more information.
<i>pTimeZone</i>	<ul style="list-style-type: none"> • Time Zone. • Offset from Universal time, i.e., the difference between local time and Universal time, in increments of 15 min (signed value).
<i>pDayltSavAdj</i>	<ul style="list-style-type: none"> • Daylight Saving Adjustment. • Daylight saving adjustment in hr. <ul style="list-style-type: none"> – Possible values: 0, 1, and 2.
<i>pRadioInterface</i>	<ul style="list-style-type: none"> • Radio interface from which the information comes • Values <ul style="list-style-type: none"> – 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - NAS_RADIO_IF_CDMA_1XEVD0 - cdma2000 HRPD (1xEV-DO) – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA - TD-SCDMA

8.447.2 Field Documentation

8.447.2.1 [BYTE](#)* nasNetworkTime::pDayltSavAdj

8.447.2.2 **BYTE*** nasNetworkTime::pRadioInterface8.447.2.3 **BYTE*** nasNetworkTime::pTimeZone8.447.2.4 **UniversalTime** nasNetworkTime::universalTime

8.448 nasOperatorNameResp Struct Reference

Data Fields

- [serviceProviderName](#) * [pSvcProviderName](#)
- [operatorPLMNList](#) * [pOperatorPLMNList](#)
- [PLMNNetworkName](#) * [pPLMNNetworkName](#)
- [operatorNameString](#) * [pOperatorNameString](#)
- [PLMNNetworkNameData](#) * [pNITZInformation](#)

8.448.1 Detailed Description

This structure contains Operator Name Data related from multiple sources.

Parameters

<i>pSvcProvider↔ Name</i>	<ul style="list-style-type: none"> • Refer serviceProviderName for details (Optional). • Can provide NULL if this parameter is not required.
<i>pOperatorPLM↔ NList</i>	<ul style="list-style-type: none"> • Refer operatorPLMNList for details (Optional). • Can provide NULL if this parameter is not required.
<i>pPLMN↔ NetworkName</i>	<ul style="list-style-type: none"> • Refer PLMNNetworkName for details (Optional). • Can provide NULL if this parameter is not required.
<i>pOperator↔ NameString</i>	<ul style="list-style-type: none"> • Refer operatorNameString for details (Optional). • Can provide NULL if this parameter is not required.
<i>pNITZ↔ Information</i>	<ul style="list-style-type: none"> • Refer PLMNNetworkNameData for details (Optional). • Can provide NULL if this parameter is not required.

8.448.2 Field Documentation

8.448.2.1 **PLMNNetworkNameData*** nasOperatorNameResp::pNITZInformation8.448.2.2 **operatorNameString*** nasOperatorNameResp::pOperatorNameString

8.448.2.3 **operatorPLMNList*** nasOperatorNameResp::pOperatorPLMNList

8.448.2.4 **PLMNNetworkName*** nasOperatorNameResp::pPLMNNetworkName

8.448.2.5 **serviceProviderName*** nasOperatorNameResp::pSvcProviderName

8.449 NASOTAMessageTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- uint32_t [message_type](#)
- uint16_t [data_len](#)
- uint8_t [data_buf](#) [2048]

8.449.1 Field Documentation

8.449.1.1 uint8_t NASOTAMessageTlv::data_buf[2048]

8.449.1.2 uint16_t NASOTAMessageTlv::data_len

8.449.1.3 uint32_t NASOTAMessageTlv::message_type

8.449.1.4 uint8_t NASOTAMessageTlv::TlvPresent

8.450 NASPhyCaAggPcellInfo Struct Reference

Data Fields

- uint32_t [pci](#)
- uint32_t [freq](#)
- [LIBPACK_NAS_LTE_CPHY_CA_BW_NRB](#) dl_bw_value
- uint32_t [iLTEbandValue](#)
- uint8_t [TlvPresent](#)

8.450.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Pcell Information.

Parameters

<i>pci</i>	<ul style="list-style-type: none"> Physical cell ID of the SCell Range. Range for ID values: 0 to 503.
<i>freq</i>	<ul style="list-style-type: none"> Frequency of the absolute cell Range. Range for ID values: 0 to 65535.
<i>dl_bw_value</i>	<ul style="list-style-type: none"> Downlink Bandwidth Values. See NAS_LTE_CPHY_CA_BW_NRB for more information.
<i>scell_state</i>	<ul style="list-style-type: none"> Scell state Values. See NAS_LTE_CPHY_SCELL_STATE for more information.
<i>TlvPresent</i>	<ul style="list-style-type: none"> Tlv Present.

8.450.2 Field Documentation

8.450.2.1 `LIBPACK_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggPcellInfo::dl_bw_value`

8.450.2.2 `uint32_t NASPhyCaAggPcellInfo::freq`

8.450.2.3 `uint32_t NASPhyCaAggPcellInfo::iLTEbandValue`

8.450.2.4 `uint32_t NASPhyCaAggPcellInfo::pci`

8.450.2.5 `uint8_t NASPhyCaAggPcellInfo::TlvPresent`

8.451 NASPhyCaAggScellIDIBw Struct Reference

Data Fields

- [LIBPACK_NAS_LTE_CPHY_CA_BW_NRB dl_bw_value](#)
- `uint8_t` [TlvPresent](#)

8.451.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation Downlink Bandwidth of Scell.

Parameters

<i>dl_bw_value</i>	<ul style="list-style-type: none"> • Downlink Bandwidth Values. • See NAS_LTE_CPHY_CA_BW_NRB for more information.
--------------------	--

8.451.2 Field Documentation

8.451.2.1 `LIBPACK_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggScellIDBw::dl_bw_value`

8.451.2.2 `uint8_t NASPhyCaAggScellIDBw::TlvPresent`

8.452 NASPhyCaAggScellIndex Struct Reference

Data Fields

- `uint8_t` [scell_idx](#)
- `uint8_t` [TlvPresent](#)

8.452.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Index.

Parameters

<i>scell_idx</i>	<ul style="list-style-type: none"> • Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
<i>TlvPresent</i>	<ul style="list-style-type: none"> • Tlv Present.

8.452.2 Field Documentation

8.452.2.1 `uint8_t NASPhyCaAggScellIndex::scell_idx`

8.452.2.2 `uint8_t NASPhyCaAggScellIndex::TlvPresent`

8.453 NASPhyCaAggScellIndType Struct Reference

Data Fields

- `uint32_t` [pci](#)
- `uint32_t` [freq](#)
- [LIBPACK_NAS_LTE_CPHY_SCELL_STATE](#) [scell_state](#)
- `uint8_t` [TlvPresent](#)

8.453.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Indicator Type.

Parameters

<i>pci</i>	<ul style="list-style-type: none"> Physical cell ID of the SCell Range. Range for ID values: 0 to 503.
<i>freq</i>	<ul style="list-style-type: none"> Frequency of the absolute cell Range. Range for ID values: 0 to 65535.
<i>scell_state</i>	<ul style="list-style-type: none"> Scell state Values. See NAS_LTE_CPHY_SCELL_STATE for more information.
<i>TlvPresent</i>	<ul style="list-style-type: none"> Tlv Present.

8.453.2 Field Documentation

8.453.2.1 `uint32_t NASPhyCaAggScellIndType::freq`

8.453.2.2 `uint32_t NASPhyCaAggScellIndType::pci`

8.453.2.3 `LIBPACK_NAS_LTE_CPHY_SCELL_STATE NASPhyCaAggScellIndType::scell_state`

8.453.2.4 `uint8_t NASPhyCaAggScellIndType::TlvPresent`

8.454 NASPhyCaAggScellInfo Struct Reference

Data Fields

- `uint32_t pci`
- `uint32_t freq`
- `LIBPACK_NAS_LTE_CPHY_CA_BW_NRB dl_bw_value`
- `uint32_t iLTEbandValue`
- `LIBPACK_NAS_LTE_CPHY_SCELL_STATE scell_state`
- `uint8_t TlvPresent`

8.454.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Information.

Parameters

<i>pci</i>	<ul style="list-style-type: none"> Physical cell ID of the SCell Range. Range for ID values: 0 to 503.
<i>freq</i>	<ul style="list-style-type: none"> Frequency of the absolute cell Range. Range for ID values: 0 to 65535.
<i>dl_bw_value</i>	<ul style="list-style-type: none"> Downlink Bandwidth Values. See NAS_LTE_CPHY_CA_BW_NRB for more information.
<i>iLTEbandValue</i>	<ul style="list-style-type: none"> Band value. Range for LTE Band class 120 to 160.
<i>scell_state</i>	<ul style="list-style-type: none"> Scell state Values. See NAS_LTE_CPHY_SCELL_STATE for more information.
<i>TlvPresent</i>	<ul style="list-style-type: none"> Tlv Present.

8.454.2 Field Documentation

8.454.2.1 **LIBPACK_NAS_LTE_CPHY_CA_BW_NRB** NASPhyCaAggScellInfo::dl_bw_value

8.454.2.2 **uint32_t** NASPhyCaAggScellInfo::freq

8.454.2.3 **uint32_t** NASPhyCaAggScellInfo::iLTEbandValue

8.454.2.4 **uint32_t** NASPhyCaAggScellInfo::pci

8.454.2.5 **LIBPACK_NAS_LTE_CPHY_SCELL_STATE** NASPhyCaAggScellInfo::scell_state

8.454.2.6 **uint8_t** NASPhyCaAggScellInfo::TlvPresent

8.455 nasPLMNNameReq Struct Reference

Data Fields

- [WORD](#) mcc
- [WORD](#) mnc
- [BYTE](#) * pMncPcsStatus

8.455.1 Detailed Description

Structure for storing the PLMN Name request parameters

Parameters

<i>mcc</i>	<ul style="list-style-type: none">• A 16-bit integer representation of MCC. Range: 0 to 999
<i>mnc</i>	<ul style="list-style-type: none">• A 16-bit integer representation of MNC. Range: 0 to 999
<i>pMncPcsStatus</i>	<ul style="list-style-type: none">• MNC PCS Digit Include Status• Used to interpret the length of the corresponding MNC reported in the PLMN TLV(0x01).• Values<ul style="list-style-type: none">– TRUE - MNC is a three-digit value. e.g. a reported value of 90 corresponds to an MNC value of 090– FALSE - MNC is a two-digit value. e.g. a reported value of 90 corresponds to an MNC value of 90

Note

If pMncPcsStatus is not present, an MNC smaller than 100 is assumed to be a two-digit value, and an MNC greater than or equal to 100 is assumed to be a three digit value.

8.455.2 Field Documentation

8.455.2.1 **WORD** nasPLMNNameReq::mcc

8.455.2.2 **WORD** nasPLMNNameReq::mnc

8.455.2.3 **BYTE*** nasPLMNNameReq::pMncPcsStatus

8.456 nasPLMNNameResp Struct Reference**Data Fields**

- [BYTE spnEncoding](#)
- [BYTE spnLength](#)
- [BYTE spn \[255\]](#)
- [BYTE shortNameEn](#)
- [BYTE shortNameCI](#)
- [BYTE shortNameSB](#)
- [BYTE shortNameLen](#)
- [BYTE shortName \[255\]](#)
- [BYTE longNameEn](#)
- [BYTE longNameCI](#)
- [BYTE longNameSB](#)
- [BYTE longNameLen](#)
- [BYTE longName \[255\]](#)

8.456.1 Detailed Description

Structure for storing the PLMN Name response parameters

Parameters

<i>spnEncoding</i>	<ul style="list-style-type: none"> Coding scheme used for service provider name. This value is ignored if spn_len is zero Values: <ul style="list-style-type: none"> 0x00 - SMS default 7-bit coded alphabet as defined in 3GPP TS 23.038 with bit 8 set to 0 0x01 - UCS2 (16 bit, little-endian) 3GPP TS 23.038
<i>spnLength</i>	<ul style="list-style-type: none"> Length of SPN which follows
<i>spn</i>	<ul style="list-style-type: none"> Service Provider name string
<i>shortNameEn</i>	<ul style="list-style-type: none"> Coding scheme used for PLMN short name. This value is ignored if PLMN short name length is zero Values: <ul style="list-style-type: none"> 0x00 - SMS default 7-bit coded alphabet as defined in 3GPP TS 23.038 with bit 8 set to 0 0x01 - UCS2 (16 bit, little-endian) 3GPP TS 23.038
<i>shortNameCI</i>	<ul style="list-style-type: none"> Indicates whether the country initials are to be added to the shortName. This value is ignored if shortNameLen is zero. Values: <ul style="list-style-type: none"> 0x00 - Do not add the letters for the countrys initials to the name 0x01 - Add the countrys initials and a text string to the name 0xFF - Not specified
<i>shortNameSB</i>	<ul style="list-style-type: none"> PLMN short name spare bits. This value is ignored if shortNameLen is zero. Values: <ul style="list-style-type: none"> 0x01 - Bit 8 is spare and set to 0 in octet 0x02 - Bits 7 and 8 are spare and set to 0 in octet n 0x03 - Bits 6 to 8 (inclusive) are spare and set to 0 in octet n 0x04 - Bits 5 to 8 (inclusive) are spare and set to 0 in octet n 0x05 - Bits 4 to 8 (inclusive) are spare and set to 0 in octet n 0x06 - Bits 3 to 8 (inclusive) are spare and set to 0 in octet n 0x07 - Bits 2 to 8 (inclusive) are spare and set to 0 in octet n 0x00 - Carries no information about the number of spare bits in octet n
<i>shortNameLen</i>	<ul style="list-style-type: none"> Length of shortName which follows
<i>shortName</i>	<ul style="list-style-type: none"> PLMN short name
<i>longNameEn</i>	<ul style="list-style-type: none"> Coding scheme used for PLMN long name. This value is ignored if PLMN long name length is zero Values: <ul style="list-style-type: none"> 0x00 - SMS default 7-bit coded alphabet as defined in 3GPP TS 23.038 with bit 8 set to 0 0x01 - UCS2 (16 bit, little-endian) 3GPP TS 23.038

<i>longNameCI</i>	<ul style="list-style-type: none"> Indicates whether the country initials are to be added to the longName. This value is ignored if longNameLen is zero. Values: <ul style="list-style-type: none"> 0x00 - Do not add the letters for the countrys initials to the name 0x01 - Add the countrys initials and a text string to the name 0xFF - Not specified
<i>longNameSB</i>	<ul style="list-style-type: none"> PLMN long name spare bits. This value is ignored if longNameLen is zero. Values: <ul style="list-style-type: none"> 0x01 - Bit 8 is spare and set to 0 in octet 0x02 - Bits 7 and 8 are spare and set to 0 in octet n 0x03 - Bits 6 to 8 (inclusive) are spare and set to 0 in octet n 0x04 - Bits 5 to 8 (inclusive) are spare and set to 0 in octet n 0x05 - Bits 4 to 8 (inclusive) are spare and set to 0 in octet n 0x06 - Bits 3 to 8 (inclusive) are spare and set to 0 in octet n 0x07 - Bits 2 to 8 (inclusive) are spare and set to 0 in octet n 0x00 - Carries no information about the number of spare bits in octet n
<i>longNameLen</i>	<ul style="list-style-type: none"> Length of longName which follows
<i>longName</i>	<ul style="list-style-type: none"> PLMN long name

Note

None

8.456.2 Field Documentation**8.456.2.1** BYTE nasPLMNNameResp::longName[255]**8.456.2.2** BYTE nasPLMNNameResp::longNameCI**8.456.2.3** BYTE nasPLMNNameResp::longNameEn**8.456.2.4** BYTE nasPLMNNameResp::longNameLen**8.456.2.5** BYTE nasPLMNNameResp::longNameSB**8.456.2.6** BYTE nasPLMNNameResp::shortName[255]**8.456.2.7** BYTE nasPLMNNameResp::shortNameCI**8.456.2.8** BYTE nasPLMNNameResp::shortNameEn

8.456.2.9 **BYTE** nasPLMNNameResp::shortNameLen

8.456.2.10 **BYTE** nasPLMNNameResp::shortNameSB

8.456.2.11 **BYTE** nasPLMNNameResp::spn[255]

8.456.2.12 **BYTE** nasPLMNNameResp::spnEncoding

8.456.2.13 **BYTE** nasPLMNNameResp::spnLength

8.457 NASPRLPreferenceTlv Struct Reference

Data Fields

- [uint8_t TlvPresent](#)
- [uint16_t PRLPref](#)

8.457.1 Field Documentation

8.457.1.1 [uint16_t NASPRLPreferenceTlv::PRLPref](#)

8.457.1.2 [uint8_t NASPRLPreferenceTlv::TlvPresent](#)

8.458 NASQmiCbkJasSwtOTAMessageInd Struct Reference

Data Fields

- [NASOTAMessageTlv otaMsgTlv](#)
- [NASLteJasReleaseInfoTlv jasRelInfoTlv](#)
- [NASTimeInfoTlv timeTlv](#)

8.458.1 Field Documentation

8.458.1.1 [NASLteJasReleaseInfoTlv NASQmiCbkJasSwtOTAMessageInd::jasRelInfoTlv](#)

8.458.1.2 [NASOTAMessageTlv NASQmiCbkJasSwtOTAMessageInd::otaMsgTlv](#)

8.458.1.3 [NASTimeInfoTlv NASQmiCbkJasSwtOTAMessageInd::timeTlv](#)

8.459 NASQmiCbkJasSystemSelPrefInd Struct Reference

Data Fields

- [NASEmergencyModeTlv EMTlv](#)
- [NASModePreferenceTlv MPTlv](#)
- [NASBandPreferenceTlv BPTlv](#)
- [NASPRLPreferenceTlv PRLPTlv](#)
- [NASRoamPreferenceTlv RPTlv](#)
- [NASLTEBandPreferenceTlv LBPTlv](#)
- [NASNetSelPreferenceTlv NSPTlv](#)
- [NASServDomainPrefTlv SDPTlv](#)
- [NASGWAqOrderPrefTlv GWAOPTlv](#)

8.459.1 Field Documentation

8.459.1.1 **NASBandPreferenceTlv** NASQmiCbkNasSystemSelPrefInd::BPTlv

8.459.1.2 **NASEmergencyModeTlv** NASQmiCbkNasSystemSelPrefInd::EMTlv

8.459.1.3 **NASGWAcqOrderPrefTlv** NASQmiCbkNasSystemSelPrefInd::GWAOPTlv

8.459.1.4 **NASLTEBandPreferenceTlv** NASQmiCbkNasSystemSelPrefInd::LBPTlv

8.459.1.5 **NASModePreferenceTlv** NASQmiCbkNasSystemSelPrefInd::MPTlv

8.459.1.6 **NASNetSelPreferenceTlv** NASQmiCbkNasSystemSelPrefInd::NSPTlv

8.459.1.7 **NASPRLPreferenceTlv** NASQmiCbkNasSystemSelPrefInd::PRLPTlv

8.459.1.8 **NASRoamPreferenceTlv** NASQmiCbkNasSystemSelPrefInd::RPTlv

8.459.1.9 **NASServDomainPrefTlv** NASQmiCbkNasSystemSelPrefInd::SDPTlv

8.460 NASRoamPreferenceTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- uint16_t [RoamPref](#)

8.460.1 Field Documentation

8.460.1.1 uint16_t NASRoamPreferenceTlv::RoamPref

8.460.1.2 uint8_t NASRoamPreferenceTlv::TlvPresent

8.461 NASServDomainPrefTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- uint32_t [SrvDomainPref](#)

8.461.1 Field Documentation

8.461.1.1 uint32_t NAServDomainPrefTlv::SrvDomainPref

8.461.1.2 uint8_t NAServDomainPrefTlv::TlvPresent

8.462 NAServingSystemInfo Struct Reference

Data Fields

- uint8_t [registrationState](#)
- uint8_t [csAttachState](#)
- uint8_t [psAttachState](#)
- uint8_t [selectedNetwork](#)
- uint8_t [radioInterfaceNo](#)
- uint8_t [radioInterfaceList](#) [255]
- uint8_t [hdrPersonality](#)

8.462.1 Detailed Description

This structure will hold the serving system parameters information

Parameters

<i>registrationState</i>	- Registration state of the mobile <ul style="list-style-type: none"> • 0 - QMI_NAS_NOT_REGISTERED Not registered;mobile is not currently searching for a new network to provide service • 1 - QMI_NAS_REGISTERED Registered with a network • 2 - QMI_NAS_NOT_REGISTERED_SEARCHING Not registered, but mobile is currently searching for a new network to provide service • 3 - QMI_NAS_REGISTRATION_DENIED Registration denied by the visible network • 4 - QMI_NAS_REGISTRATION_UNKNOWN Registration state is unknown
<i>csAttachState</i>	- Circuit Switch domain attach state of the mobile <ul style="list-style-type: none"> • 0 - Unknown or not applicable • 1 - Attached • 2 - Detached
<i>psAttachState</i>	- Packet domain attach state of the mobile <ul style="list-style-type: none"> • 0 - Unknown or not applicable • 1 - Attached • 2 - Detached
<i>selectedNetwork</i>	- Type of selected radio access network <ul style="list-style-type: none"> • 0x00 - Unknown • 0x01 - 3GPP2 network • 0x02 - 3GPP network
<i>radioInterfaceNo</i>	- Number of radio interfaces currently in use; this indicates how many radio_if identifiers follow this field

<i>radioInterfaceList</i>	- Radio interface currently in use (each is 1 byte) <ul style="list-style-type: none"> • 0x00 - None (no service) • 0x01 - cdma2000 1X • 0x02 - cdma2000 HRPD (1xEV-DO) • 0x03 - AMPS • 0x04 - GSM • 0x05 - UMTS • 0x08 - LTE
<i>hdrPersonality</i>	- HDR personality information (valid only for EVDO) <ul style="list-style-type: none"> • 0x00 - Unknown • 0x01 - HRPD • 0x02 - eHRPD

Note: None

8.462.2 Field Documentation

8.462.2.1 uint8_t NAServingSystemInfo::csAttachState

8.462.2.2 uint8_t NAServingSystemInfo::hdrPersonality

8.462.2.3 uint8_t NAServingSystemInfo::psAttachState

8.462.2.4 uint8_t NAServingSystemInfo::radioInterfaceList[255]

8.462.2.5 uint8_t NAServingSystemInfo::radioInterfaceNo

8.462.2.6 uint8_t NAServingSystemInfo::registrationState

8.462.2.7 uint8_t NAServingSystemInfo::selectedNetwork

8.463 nasSigInfo Struct Reference

Data Fields

- [CDMASSInfo](#) * [pCDMASigInfo](#)
- [HDRSSInfo](#) * [pHDRSigInfo](#)
- [INT8](#) * [pGSMSigInfo](#)
- [CDMASSInfo](#) * [pWCDMASigInfo](#)
- [LTESSInfo](#) * [pLTESigInfo](#)
- [INT8](#) * [pRscp](#)
- [TDSCDMASigInfoExt](#) * [pTDSCDMASigInfoExt](#)

8.463.1 Detailed Description

Structure for storing the [nasSigInfo](#) indication parameters.

Parameters

<i>pCDMASigInfo</i>	<ul style="list-style-type: none"> • See CDMASSInfo for more information.
<i>pHDRSigInfo</i>	<ul style="list-style-type: none"> • See HDRSSInfo for more information.
<i>pGSMSigInfo</i>	<ul style="list-style-type: none"> • one byte value, GSM signal strength is the RSSI in dBm (signed value). A value of -125 dBm or lower is used to indicate No Signal
<i>pWCDMASigInfo</i>	<ul style="list-style-type: none"> • See CDMASSInfo for more information.
<i>pLTESigInfo</i>	<ul style="list-style-type: none"> • See LTESSInfo for more information.
<i>pRscp</i>	<ul style="list-style-type: none"> • RSCP of the Primary Common Control Physical Channel (PCCPCH) in dBm. Measurement range: -120 dBm to -25 dBm.
<i>pTDSCDMA_↔ SigInfoExt</i>	<ul style="list-style-type: none"> • See TDSCDMASigInfoExt for more information.

8.463.2 Field Documentation

8.463.2.1 **CDMASSInfo*** nasSigInfo::pCDMASigInfo

8.463.2.2 **INT8*** nasSigInfo::pGSMSigInfo

8.463.2.3 **HDRSSInfo*** nasSigInfo::pHDRSigInfo

8.463.2.4 **LTESInfo*** nasSigInfo::pLTESigInfo

8.463.2.5 **INT8*** nasSigInfo::pRscp

8.463.2.6 **TDSCDMASigInfoExt*** nasSigInfo::pTDSCDMASigInfoExt

8.463.2.7 **CDMASSInfo*** nasSigInfo::pWCDMASigInfo

8.464 nasSwiGetChannelLockResp Struct Reference

Data Fields

- [wcdmaUARFCN](#) * [pWcdmaUARFCN](#)
- [lteEARFCN](#) * [pLteEARFCN](#)
- [ltePCI](#) * [pLtePCI](#)

8.464.1 Detailed Description

This structure contains the SLQSNASSwiGetChannelLock response parameters.

Parameters

<i>pWcdmaUARFCN</i>	[Optional] <ul style="list-style-type: none">See wcdmaUARFCN for more information
<i>pLteEARFCN</i>	[Optional] <ul style="list-style-type: none">See lteEARFCN for more information
<i>pLtePCI</i>	[Optional] <ul style="list-style-type: none">See ltePCI for more information

8.464.2 Field Documentation

8.464.2.1 **lteEARFCN*** nasSwiGetChannelLockResp::pLteEARFCN

8.464.2.2 **ltePCI*** nasSwiGetChannelLockResp::pLtePCI

8.464.2.3 **wcdmaUARFCN*** nasSwiGetChannelLockResp::pWcdmaUARFCN

8.465 NasSwiIndReg Struct Reference

Data Fields

- [BYTE lteEsmUI](#)
- [BYTE lteEsmDI](#)
- [BYTE lteEmmUI](#)
- [BYTE lteEmmDI](#)
- [BYTE gsmUmtsUI](#)
- [BYTE gsmUmtsDI](#)
- [BYTE * pRankIndicatorInd](#)
- [BYTE * pTimer](#)

8.465.1 Detailed Description

This structure contains the OTA message indication.

Parameters

<i>lteEsmUI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report LTE ESM uplink messages
<i>lteEsmDI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report LTE ESM downlink messages
<i>lteEmmUI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report LTE EMM uplink messages
<i>lteEmmDI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report GSM/UMTS uplink messages
<i>gsmUmtsUI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report GSM/UMTS uplink messages
<i>gsmUmtsDI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report GSM/UMTS downlink messages
<i>pRankIndicatorInd</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report Rank Indicator messages
<i>pTimer</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report Timer Indicator messages

8.465.2 Field Documentation

8.465.2.1 BYTE NasSwiIndReg::gsmUmtsDI

8.465.2.2 BYTE NasSwiIndReg::gsmUmtsUI

8.465.2.3 BYTE NasSwiIndReg::lteEmmDI

8.465.2.4 BYTE NasSwiIndReg::lteEmmUI

8.465.2.5 BYTE NasSwiIndReg::lteEsmDI

8.465.2.6 BYTE NasSwiIndReg::lteEsmUI

8.465.2.7 BYTE* NasSwiIndReg::pRankIndicatorInd

8.465.2.8 BYTE* NasSwiIndReg::pTimer

8.466 nasSwiSetChannelLockReq Struct Reference

Data Fields

- [wcdmaUARFCN](#) * [pWcdmaUARFCN](#)
- [lteEARFCN](#) * [pLteEARFCN](#)
- [ltePCI](#) * [pLtePCI](#)

8.466.1 Detailed Description

This structure contains the SLQSNASSwiSetChannelLock response parameters.

Parameters

<i>pWcdmaUARFCN</i>	[Optional] • See wcdmaUARFCN for more information
<i>pLteEARFCN</i>	[Optional] • See lteEARFCN for more information
<i>pLtePCI</i>	[Optional] • See ltePCI for more information

8.466.2 Field Documentation

8.466.2.1 [lteEARFCN](#)* nasSwiSetChannelLockReq::pLteEARFCN8.466.2.2 [ltePCI](#)* nasSwiSetChannelLockReq::pLtePCI8.466.2.3 [wcdmaUARFCN](#)* nasSwiSetChannelLockReq::pWcdmaUARFCN

8.467 nasSysInfo Struct Reference

Data Fields

- [SrvStatusInfo](#) * [pCDMASrvStatusInfo](#)
- [SrvStatusInfo](#) * [pHDRSrvStatusInfo](#)
- [GSMSrvStatusInfo](#) * [pGSMSrvStatusInfo](#)
- [GSMSrvStatusInfo](#) * [pWCDMASrvStatusInfo](#)
- [GSMSrvStatusInfo](#) * [pLTESrvStatusInfo](#)
- [CDMASysInfo](#) * [pCDMASysInfo](#)
- [HDRSysInfo](#) * [pHDRSysInfo](#)
- [GSMSysInfo](#) * [pGSMSysInfo](#)
- [WCDMASysInfo](#) * [pWCDMASysInfo](#)

- [LTESysInfo](#) * [pLTESysInfo](#)
- [AddCDMASysInfo](#) * [pAddCDMASysInfo](#)
- [WORD](#) * [pAddHDRSysInfo](#)
- [AddSysInfo](#) * [pAddGSMSysInfo](#)
- [AddSysInfo](#) * [pAddWCDMASysInfo](#)
- [WORD](#) * [pAddLTESysInfo](#)
- [CallBarringSysInfo](#) * [pGSMCallBarringSysInfo](#)
- [CallBarringSysInfo](#) * [pWCDMACallBarringSysInfo](#)
- [BYTE](#) * [pLTEVoiceSupportSysInfo](#)
- [BYTE](#) * [pGSMCipherDomainSysInfo](#)
- [BYTE](#) * [pWCDMACipherDomainSysInfo](#)
- [BYTE](#) * [pSysInfoNoChange](#)

8.467.1 Detailed Description

Structure for storing the [nasSysInfo](#) indication parameters.

Parameters

<i>pCDMASrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See SrvStatusInfo for more information.
<i>pHDRSrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See SrvStatusInfo for more information.
<i>pGSMSrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See GSMSrvStatusInfo for more information.
<i>pWCDMASrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See GSMSrvStatusInfo for more information.
<i>pLTESrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See GSMSrvStatusInfo for more information.
<i>pCDMASysInfo</i>	<ul style="list-style-type: none"> • See CDMASysInfo for more information.
<i>pHDRSysInfo</i>	<ul style="list-style-type: none"> • See HDRSysInfo for more information.
<i>pGSMSysInfo</i>	<ul style="list-style-type: none"> • See GSMSysInfo for more information.
<i>pWCDMASys↔ Info</i>	<ul style="list-style-type: none"> • See WCDMASysInfo for more information.
<i>pLTESysInfo</i>	<ul style="list-style-type: none"> • See LTESysInfo for more information.
<i>pAddCDMA↔ SysInfo</i>	<ul style="list-style-type: none"> • See AddCDMASysInfo for more information.

<i>pAddHDRSysInfo</i>	<ul style="list-style-type: none"> • System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
<i>pAddGSMSysInfo</i>	<ul style="list-style-type: none"> • See AddSysInfo for more information.
<i>pAddWCDMA SysInfo</i>	<ul style="list-style-type: none"> • See AddSysInfo for more information.
<i>pAddLTESysInfo</i>	<ul style="list-style-type: none"> • System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
<i>pGSMCallBarringSysInfo</i>	<ul style="list-style-type: none"> • See CallBarringSysInfo for more information.
<i>pWCDMACallBarringSysInfo</i>	<ul style="list-style-type: none"> • See CallBarringSysInfo for more information.
<i>pLTEVoiceSupportSysInfo</i>	<ul style="list-style-type: none"> • Indicates voice support status on LTE. <ul style="list-style-type: none"> – 0x00 - Voice is not supported – 0x01 - Voice is supported
<i>pGSMCipherDomainSysInfo</i>	<ul style="list-style-type: none"> • Ciphering on the service domain. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched
<i>pWCDMA CipherDomain SysInfo</i>	<ul style="list-style-type: none"> • Ciphering on the service domain. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched
<i>pSysInfoNoChange</i>	<ul style="list-style-type: none"> • System Info No Change. • Flag used to notify clients that a request to select a network ended with no change in the PLMN. <ul style="list-style-type: none"> – 0x01 - No change in system information

8.467.2 Field Documentation

8.467.2.1 AddCDMASysInfo* nasSysInfo::pAddCDMASysInfo

- 8.467.2.2 **AddSysInfo*** nasSysInfo::pAddGSMSysInfo
- 8.467.2.3 **WORD*** nasSysInfo::pAddHDRSysInfo
- 8.467.2.4 **WORD*** nasSysInfo::pAddLTESysInfo
- 8.467.2.5 **AddSysInfo*** nasSysInfo::pAddWCDMASysInfo
- 8.467.2.6 **SrvStatusInfo*** nasSysInfo::pCDMASrvStatusInfo
- 8.467.2.7 **CDMASysInfo*** nasSysInfo::pCDMASysInfo
- 8.467.2.8 **CallBarringSysInfo*** nasSysInfo::pGSMCallBarringSysInfo
- 8.467.2.9 **BYTE*** nasSysInfo::pGSMCipherDomainSysInfo
- 8.467.2.10 **GSMSrvStatusInfo*** nasSysInfo::pGSMSrvStatusInfo
- 8.467.2.11 **GSMSysInfo*** nasSysInfo::pGSMSysInfo
- 8.467.2.12 **SrvStatusInfo*** nasSysInfo::pHDRSrvStatusInfo
- 8.467.2.13 **HDRSysInfo*** nasSysInfo::pHDRSysInfo
- 8.467.2.14 **GSMSrvStatusInfo*** nasSysInfo::pLTESrvStatusInfo
- 8.467.2.15 **LTESysInfo*** nasSysInfo::pLTESysInfo
- 8.467.2.16 **BYTE*** nasSysInfo::pLTEVoiceSupportSysInfo
- 8.467.2.17 **BYTE*** nasSysInfo::pSysInfoNoChange
- 8.467.2.18 **CallBarringSysInfo*** nasSysInfo::pWCDMACallBarringSysInfo
- 8.467.2.19 **BYTE*** nasSysInfo::pWCDMACipherDomainSysInfo
- 8.467.2.20 **GSMSrvStatusInfo*** nasSysInfo::pWCDMASrvStatusInfo
- 8.467.2.21 **WCDMASysInfo*** nasSysInfo::pWCDMASysInfo

8.468 NASTimeInfoTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- uint64_t [time](#)

8.468.1 Field Documentation

8.468.1.1 `uint64_t` NASTimeInfoTlv::time

8.468.1.2 `uint8_t` NASTimeInfoTlv::TlvPresent

8.469 nasTimers Struct Reference

Data Fields

- `char` [t3396_apn](#) [101]
- `BYTE` [t3396_plmn_id](#) [3]
- `ULONG` [t3396_val](#)

8.469.1 Detailed Description

This structure contains Network Timer

Parameters

<i>t3396_apn</i>	<ul style="list-style-type: none">• APN name
<i>t3396_plmn_id</i>	<ul style="list-style-type: none">• PLMN ID
<i>t3396_val</i>	<ul style="list-style-type: none">• T3396 timer value

8.469.2 Field Documentation

8.469.2.1 `char` nasTimers::t3396_apn[101]

8.469.2.2 `BYTE` nasTimers::t3396_plmn_id[3]

8.469.2.3 `ULONG` nasTimers::t3396_val

8.470 netSelectionPref Struct Reference

Data Fields

- `BYTE` [netReg](#)
- `WORD` [mcc](#)
- `WORD` [mnc](#)

8.470.1 Detailed Description

Contain the network selection preference.

Parameters

<i>netReg</i>	<ul style="list-style-type: none"> specifies one of the following actions: <ul style="list-style-type: none"> 0x00 - Automatic registration <ul style="list-style-type: none"> Device registers according to its provisioning; mcc and mnc fields are ignored 0x01 - Manual Registration <ul style="list-style-type: none"> Device registers to specified network; mcc and mnc must contain valid values
<i>mcc</i>	<ul style="list-style-type: none"> MCC value. Range 0 to 999
<i>mnc</i>	<ul style="list-style-type: none"> MNC value. Range 0 to 999

8.470.2 Field Documentation

8.470.2.1 WORD netSelectionPref::mcc

8.470.2.2 WORD netSelectionPref::mnc

8.470.2.3 BYTE netSelectionPref::netReg

8.471 NetStats Struct Reference

Data Fields

- [ULONG rx_packets](#)
- [ULONG tx_packets](#)
- [ULONGLONG rx_bytes](#)
- [ULONGLONG tx_bytes](#)
- [ULONG rx_errors](#)
- [ULONG tx_errors](#)
- [ULONG rx_overflows](#)
- [ULONG tx_overflows](#)

8.471.1 Detailed Description

This structure contains the SLQSGetNetStatistic Information

Parameters

<i>rx_packets</i>	<ul style="list-style-type: none"> • Number of received Packets without error
<i>tx_packets</i>	<ul style="list-style-type: none"> • Number of transmitted Packets without error
<i>rx_bytes</i>	<ul style="list-style-type: none"> • Number of bytes recieved without error
<i>tx_bytes</i>	<ul style="list-style-type: none"> • NNumbero of bytes transmitted without error
<i>rx_error</i>	<ul style="list-style-type: none"> • Number of incoming packets with framing errors
<i>tx_error</i>	<ul style="list-style-type: none"> • Number of outgoing packets with framing errors
<i>rx_overflows</i>	<ul style="list-style-type: none"> • Number of packets dropped because Rx buffer overflowed
<i>tx_overflows</i>	<ul style="list-style-type: none"> • Number of packets dropped because Tx buffer overflowed

8.471.2 Field Documentation

8.471.2.1 **ULONGLONG** NetStats::rx_bytes8.471.2.2 **ULONG** NetStats::rx_errors8.471.2.3 **ULONG** NetStats::rx_overflows8.471.2.4 **ULONG** NetStats::rx_packets8.471.2.5 **ULONGLONG** NetStats::tx_bytes8.471.2.6 **ULONG** NetStats::tx_errors8.471.2.7 **ULONG** NetStats::tx_overflows8.471.2.8 **ULONG** NetStats::tx_packets

8.472 NetworkDebugResp Struct Reference

Data Fields

- [BYTE](#) * [pObjectVer](#)
- [NetworkStat1x](#) * [pNetworkStat1x](#)
- [NetworkStatEVDO](#) * [pNetworkStatEVDO](#)
- [DeviceConfigDetail](#) * [pDeviceConfigDetail](#)
- [DataStatusDetail](#) * [pDataStatusDetail](#)

8.472.1 Detailed Description

This structure contains information about the SLQSSwiNetworkDebug response parameters.

Parameters

<i>pObjectVer</i>	<ul style="list-style-type: none"> Object's version number for the host to handle <ul style="list-style-type: none"> 0xFF - NA Others - shows in decimal
<i>pNetworkStat1x</i>	<ul style="list-style-type: none"> See NetworkStat1x for more information
<i>pNetworkStat↔ EVDO</i>	<ul style="list-style-type: none"> See NetworkStatEVDO for more information.
<i>pDeviceConfig↔ Detail</i>	<ul style="list-style-type: none"> See DeviceConfigDetail for more information.
<i>pDataStatus↔ Detail</i>	<ul style="list-style-type: none"> See DataStatusDetail for more information.

8.472.2 Field Documentation

8.472.2.1 **DataStatusDetail*** NetworkDebugResp::pDataStatusDetail

8.472.2.2 **DeviceConfigDetail*** NetworkDebugResp::pDeviceConfigDetail

8.472.2.3 **NetworkStat1x*** NetworkDebugResp::pNetworkStat1x

8.472.2.4 **NetworkStatEVDO*** NetworkDebugResp::pNetworkStatEVDO

8.472.2.5 **BYTE*** NetworkDebugResp::pObjectVer

8.473 NetworkStat1x Struct Reference

Data Fields

- [BYTE](#) State
- [WORD](#) SO
- [ULONG](#) RX_PWR
- [WORD](#) RX_EC_IO
- [ULONG](#) TX_PWR
- [BYTE](#) ActSetCnt
- [ActPilotPNElement](#) * [pActPilotPNElements](#)
- [BYTE](#) NeighborSetCnt
- [WORD](#) * [pNeighborSetPilotPN](#)

8.473.1 Detailed Description

This structure contains 1x network status details

Parameters

<i>State</i>	<ul style="list-style-type: none"> • CDMA current access state <ul style="list-style-type: none"> – 0x01 - Searching – 0x02 - Idle – 0x03 - Traffic – Others - NA
<i>SO</i>	<ul style="list-style-type: none"> • CDMA service option <ul style="list-style-type: none"> – 0xFFFF - Not in a call – 0x0001 - Basic Variable Rate Voice Service(8kbps) – 0x0002 - Mobile Station Loopback(8kbps) – 0x0003 - Enhanced Variable Rate Codec(EVRC) Voice Service(8kbps) – 0x0006 - Short message Services(Rate Set 1) – 0x0009 - Mobile Station Loopback(13kbps) – 0x000E - Short Message Service (Rate Set 2) – 0x0011 - High Rate Voice Service(13kbps) – 0x0020 - Test Data Service Option(TDSO) – 0x0021 - cdma2000 High Speed Packet Data Service, Internet or ISO Protocol Stack – 0x0044 - EVRC-B Voice Service(8 kbps) – 0x0046 - EVRC-WB Voice Service(8 kbps) – 0x0049 - Voice Echo mode supports smart blanking(EVRC-NW) – 0x004B - Enhanced loopback – 0x8000 - Proprietary Service Option (Qualcomm Inc.)
<i>RX_PWR</i>	<ul style="list-style-type: none"> • RX Pwr(dBm) <ul style="list-style-type: none"> – 0xABCD00EF - -ABCD.EF dBm – ABCD00EF should be transferred to decimal while displaying – Example: 0x12340056 - -4660.86dBm 0x1234 = 4660, 0x0056 = 86 – 0xFFFFFFFF - NA
<i>RX_EC_IO</i>	<ul style="list-style-type: none"> • RX EC/IO(dB) <ul style="list-style-type: none"> – 0xABCD - -AB.CD dB – ABCD should be transferred to decimal while displaying – Example: 0x1234 - -18.52dB 0x12 = 18, 0x34 = 52 – 0xFFFF - NA
<i>TX_PWR</i>	<ul style="list-style-type: none"> • TX PWR(dBm) <ul style="list-style-type: none"> – 0xFFFFFFFF - NA – Others - display actual value in decimal – Example: 0x1234 - -4660dBm 0x1234 = 4660

<i>ActSetCnt</i> (IN/OUT) <ul style="list-style-type: none"> Count of active pilot PN elements As input specifies number of sets of parameter pActPilotElements for which memory has been assigned As output specifies the actual number of sets of parameter pActPilotElements returned by device
<i>pActPilotPN↔ Elements</i>	<ul style="list-style-type: none"> See ActPilotPNElement for more information
<i>NeighborSetCnt</i> (IN/OUT) <ul style="list-style-type: none"> Count of neighbor pilot PN elements As input specifies number of sets of parameter pNeighborSetPilotPN for which memory has been assigned As output specifies the actual number of sets of parameter pNeighborSetPilotPN returned by device
<i>pNeighborSet↔ PilotPN</i>	<ul style="list-style-type: none"> Neighbor pilot PN

8.473.2 Field Documentation

8.473.2.1 BYTE NetworkStat1x::ActSetCnt

8.473.2.2 BYTE NetworkStat1x::NeighborSetCnt

8.473.2.3 ActPilotPNElement* NetworkStat1x::pActPilotPNElements

8.473.2.4 WORD* NetworkStat1x::pNeighborSetPilotPN

8.473.2.5 WORD NetworkStat1x::RX_EC_IO

8.473.2.6 ULONG NetworkStat1x::RX_PWR

8.473.2.7 WORD NetworkStat1x::SO

8.473.2.8 BYTE NetworkStat1x::State

8.473.2.9 ULONG NetworkStat1x::TX_PWR

8.474 NetworkStatEVDO Struct Reference

Data Fields

- [BYTE State](#)
- [BYTE MACIndex](#)
- [BYTE SectorIDLen](#)
- [WORD * pSectorID](#)
- [WORD RX_PWR](#)
- [WORD PER](#)
- [WORD PilotEnergy](#)
- [BYTE SNR](#)

8.474.1 Detailed Description

This structure contains EVDO network status details

Parameters

<i>State</i>	<ul style="list-style-type: none"> • EVDO network access state <ul style="list-style-type: none"> – 0x00 - Sleep – 0x01 - Searching – 0x02 - Idle – 0x03 - Active – 0xFF - NA
<i>MACIndex</i>	<ul style="list-style-type: none"> • HDR Mac index <ul style="list-style-type: none"> – 0xFF - NA – Others - Display the actual value in decimal – Example: 0x12 - 18 0x12 = 18
<i>SectorIDLen</i>	(IN/OUT) <ul style="list-style-type: none"> • Sector ID length • As input specifies length of parameter pSectorID for which memory has been assigned • As output specifies the actual length of parameter pSectorID returned by device
<i>pSectorID</i>	<ul style="list-style-type: none"> • Sector ID
<i>RX_PWR</i>	<ul style="list-style-type: none"> • TX PWR(dBm) <ul style="list-style-type: none"> – 0xABCD - -ABCD dBm – ABCD should be transferred to decimal while displaying – Example: 0x1234 - -4660dBm 0x1234 = 4660 – 0xFFFF - NA
<i>PER</i>	<ul style="list-style-type: none"> • HDR Packet Error Rate <ul style="list-style-type: none"> – 0xFFFF - Unknown – Others - display the actual value in decimal – Example: 0x1234 - -4660dBm 0x1234 = 4660
<i>PilotEnergy</i>	<ul style="list-style-type: none"> • Pilt Energy (dB) <ul style="list-style-type: none"> – 0xFFFF - NA – 0xABCD should be transferred to decimal while displaying – Example: 0x1234 - -4660dBm 0x1234 = 4660
<i>SNR</i>	<ul style="list-style-type: none"> • Signal to Noise ratio (dB)

8.474.2 Field Documentation

8.474.2.1 **BYTE** NetworkStatEVDO::MACIndex

8.474.2.2 **WORD** NetworkStatEVDO::PER

8.474.2.3 **WORD** NetworkStatEVDO::PilotEnergy

8.474.2.4 **WORD*** NetworkStatEVDO::pSectorID

8.474.2.5 **WORD** NetworkStatEVDO::RX_PWR

8.474.2.6 **BYTE** NetworkStatEVDO::SectorIDLen

8.474.2.7 **BYTE** NetworkStatEVDO::SNR

8.474.2.8 **BYTE** NetworkStatEVDO::State

8.475 newMTMessageTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- [sMSMTMessageInfo](#) [MTMessageInfo](#)

8.475.1 Detailed Description

Parameters

<i>TlvPresent</i>	<ul style="list-style-type: none"> • Boolean indicating the presence of the TLV in the QMI response
<i>MTMessageInfo</i>	<ul style="list-style-type: none"> • MT Message • See sMSMTMessageInfo for more information

8.475.2 Field Documentation

8.475.2.1 **sMSMTMessageInfo** newMTMessageTlv::MTMessageInfo

8.475.2.2 **uint8_t** newMTMessageTlv::TlvPresent

8.476 newPwdData Struct Reference

Data Fields

- [BYTE newPwd](#) [4]
- [BYTE newPwdAgain](#) [4]

8.476.1 Detailed Description

This structure contains New Password Data.

Parameters

<i>newPwd</i> [<i>PASSWORD_LENGTH</i>]	<ul style="list-style-type: none">• New password.<ul style="list-style-type: none">– Password consists of 4 ASCII digits.– Range: 0000 to 9999.
<i>newPwdAgain</i> [<i>PASSWORD_LENGTH</i>]	<ul style="list-style-type: none">• New password again.<ul style="list-style-type: none">– Password consists of 4 ASCII digits.– Range: 0000 to 9999.

8.476.2 Field Documentation

8.476.2.1 [BYTE newPwdData::newPwd](#)[4]

8.476.2.2 [BYTE newPwdData::newPwdAgain](#)[4]

8.477 nmrCellInfo Struct Reference

Data Fields

- [ULONG nmrCellID](#)
- [BYTE nmrPlmn](#) [3]
- [WORD nmrLac](#)
- [WORD nmrArfcn](#)
- [BYTE nmrBsic](#)
- [WORD nmrRxLev](#)

8.477.1 Detailed Description

This structure contains information about the Network Measurement Report (NMR) Cell Information.

Parameters

<i>nmrCellID</i>	<ul style="list-style-type: none"> • Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.
<i>nmrPlmn</i> [<i>PLM</i> ↔ <i>N_LENGTH</i>]	<ul style="list-style-type: none"> • MCC/MNC information coded as octet 3, 4, and 5. • This field is ignored when <i>nmrCellID</i> is not present.
<i>nmrLac</i>	<ul style="list-style-type: none"> • Location area code. • This field is ignored when <i>nmrCellID</i> is not present. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>nmrArfcn</i>	<ul style="list-style-type: none"> • Absolute RF channel number. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>nmrBsic</i>	<ul style="list-style-type: none"> • Base station identity code. <ul style="list-style-type: none"> – 0xFF - Not Available
<i>nmrRxLev</i>	<ul style="list-style-type: none"> • Cell Rx measurement. • Values range between 0 and 63. • Mapped to a measured signal level: <ul style="list-style-type: none"> – Rxlev 0 is a signal strength less than -110 dBm – Rxlev 1 is -110 dBm to -109 dBm – Rxlev 2 is -109 dBm to -108 dBm – ... – Rxlev 62 is -49 dBm to -48 dBm – Rxlev 63 is greater than -48 dBm – 0xFFFF - Not Available

8.477.2 Field Documentation

8.477.2.1 WORD nmrCellInfo::nmrArfcn

8.477.2.2 BYTE nmrCellInfo::nmrBsic

8.477.2.3 ULONG nmrCellInfo::nmrCellID

8.477.2.4 WORD nmrCellInfo::nmrLac

8.477.2.5 BYTE nmrCellInfo::nmrPlmn[3]

8.477.2.6 WORD nmrCellInfo::nmrRxLev

8.478 NSSAudioCtrl Struct Reference

Data Fields

- [BYTE upLink](#)
- [BYTE downLink](#)

8.478.1 Detailed Description

This structure contains National Supplementary Services - Audio Control Information

Parameters

<i>upLink</i>	<ul style="list-style-type: none">• Values as per[S24, 4.10 Reservation response].
<i>downLink</i>	<ul style="list-style-type: none">• Values as per[S24, 4.10 Reservation response].

8.478.2 Field Documentation

8.478.2.1 **BYTE** NSSAudioCtrl::downLink

8.478.2.2 **BYTE** NSSAudioCtrl::upLink

8.479 NWProfile Struct Reference

Data Fields

- [WORD tech](#)
- [BYTE * pProfSz](#)
- [WORD * pProfValues](#)

8.479.1 Detailed Description

This structure contains Network supported QoS profile

Parameters

<i>tech</i>	Technology on which the network supported QoS profiles are being returned: <ul style="list-style-type: none">• CDMA – 0x8001
<i>exponent</i>	

8.479.2 Field Documentation

8.479.2.1 **BYTE*** NWProfile::pProfSz

8.479.2.2 **WORD*** NWProfile::pProfValues

8.479.2.3 **WORD** NWProfile::tech

8.480 omaDmConfigTlv Struct Reference

Data Fields

- [BYTE](#) state
- [BYTE](#) userInputReq
- [USHORT](#) userInputTimeout
- [USHORT](#) alertmsglength
- [BYTE](#) alertmsg [256]

8.480.1 Detailed Description

This structure will hold the SwiOmaDmConfig session parameters information.

Parameters

<i>state</i>	<ul style="list-style-type: none"> • 0x01 - OMA-DM Read Request • 0x02 - OMA-DM Change Request • 0x03 - OMA-DM Config Complete
<i>user_input_req</i>	- Bit mask of available user inputs <ul style="list-style-type: none"> • 0x00 - No user input required. Informational indication • 0x01 - Accept • 0x02 - Reject
<i>user_input↔ timeout</i>	<ul style="list-style-type: none"> • Timeout for user input in minutes. A value of 0 means no time-out
<i>alertmsglength</i>	<ul style="list-style-type: none"> • Length of Alert message string in bytes
<i>alertmsg</i>	<ul style="list-style-type: none"> • Alert message in UCS2 (Max 256 characters)

8.480.2 Field Documentation

8.480.2.1 **BYTE** omaDmConfigTlv::alertmsg[256]

8.480.2.2 **USHORT** omaDmConfigTlv::alertmsglength

8.480.2.3 **BYTE** omaDmConfigTlv::state

8.480.2.4 **BYTE** omaDmConfigTlv::userInputReq

8.480.2.5 **USHORT** omaDmConfigTlv::userInputTimeout

8.481 omaDmConfigTlvExt Struct Reference

Data Fields

- [BYTE](#) state
- [BYTE](#) userInputReq
- [USHORT](#) userInputTimeout
- [USHORT](#) alertmsglength
- [BYTE](#) alertmsg [256]

8.481.1 Detailed Description

This structure will hold the SwiOmaDmConfig session parameters information.

Parameters

<i>state</i>	<ul style="list-style-type: none"> • 1 - reserved • 2 - reserved • 3 - reserved • 4 - CI DC Success • 5 - CI DC Failure • 6 - User/device initiated PRL update success. • 7 - User/device initiated PRL update failure. • 8 - HFA DC session start • 9 - HFA DC sucess. • 10 - HFA is cancelled. • 11 - HFA retry. UI Screen 13[1] with 0 percent progress bar should be shown. • 12 - HFA fail after 5 retries. UI Screen 2[1] should be displayed. • 13 - HFA retry down counter. Used to update the process bar of UI Screen 13[1]. • 14 - HFA PRL session start, UI screen 4[1] should be displayed. • 15 - HFA PRL update success. • 16 - Device is launching a NI session. UI Screen 1[1] should be displayed. • 17 - An empty session. UI Screen 2[1] should be displayed. • 18 - No network coverage. • 19 - HFA is not enabled. • 20 - CI DC Start, UI Screen 1[1] should be displayed. • 21 - CI PRL start, UI screen 4[1] should be displayed. • 22 - HFA PRL updates fail. • 23 - Device reboot. • 24 - CI DC is cancelled. • 25 - User/device initiated PRL update is cancelled. • 26 - NI session is cancelled. • 27 - Current NI session is not enabled. • 28 - NI DC success. • 29 - NI DC Fail. • 30 - NI PRL success • 31 - NI PRL fail. • 32 - Reserved • 33 - NI fumo fail • 34 - NI session fail, unable to point out the session type.
<i>user_input_req</i>	- OMA task stop to wait user's input if this field is valid. until user input selection or after "UI Timer out (next field). In the case of timeout, a default selection of "YES/OK" is accepted. Note that this option is valid when DM state is 4/6/12/28/30. 0 - user/host doesn't need to input anything, and OMA task doesn't blocked by UI. 1 - user/host must input "YES/OK/CANCEL". 2 - User/host must input "NO/CANCEL". 3 - user/host must input "YES/OK/NO/CANCEL".
<i>user_input_↔ timeout</i>	<ul style="list-style-type: none"> • Timeout for user input in seconds. This indicates how many seconds OMA task stop to wait for host/user's response.
<i>alertmsglength</i>	<ul style="list-style-type: none"> • Length of Alert message string in word(16-bit)

<i>alertmsg</i>	<ul style="list-style-type: none">• Alert message in UCS2 (Max 256 characters)• This string is printed by host
-----------------	---

8.481.2 Field Documentation

8.481.2.1 **BYTE** omaDmConfigTlvExt::alertmsg[256]

8.481.2.2 **USHORT** omaDmConfigTlvExt::alertmsglength

8.481.2.3 **BYTE** omaDmConfigTlvExt::state

8.481.2.4 **BYTE** omaDmConfigTlvExt::userInputReq

8.481.2.5 **USHORT** omaDmConfigTlvExt::userInputTimeout

8.482 omaDmFotaTlv Struct Reference

Data Fields

- [BYTE](#) state
- [BYTE](#) userInputReq
- [USHORT](#) userInputTimeout
- [ULONG](#) fwdloadsize
- [ULONG](#) fwloadComplete
- [USHORT](#) updateCompleteStatus
- [BYTE](#) severity
- [USHORT](#) versionlength
- [BYTE](#) version [256]
- [USHORT](#) namelength
- [BYTE](#) package_name [256]
- [USHORT](#) descriptionlength
- [BYTE](#) description [256]
- [BYTE](#) sessionType

8.482.1 Detailed Description

This structure will hold the SwiOmaDmFota session parameters information.

Parameters

<i>state</i>	<ul style="list-style-type: none"> • 0x01 - No Firmware available • 0x02 - Query Firmware Download • 0x03 - Firmware Downloading • 0x04 - Firmware downloaded • 0x05 - Query Firmware Update • 0x06 - Firmware updating • 0x07 - Firmware updated
<i>user_input_req</i>	<ul style="list-style-type: none"> - Bit mask of available user inputs • 0x00 - No user input required. Informational indication • 0x01 - Accept • 0x02 - Reject
<i>user_input_↔ timeout</i>	<ul style="list-style-type: none"> • Timeout for user input in minutes. A value of 0 means no time-out
<i>fw_dload_size</i>	<ul style="list-style-type: none"> • The size (in bytes) of the firmware update package
<i>fw_dload_↔ complete</i>	<ul style="list-style-type: none"> • The number of bytes downloaded. Need to determine how often to send this message for progress bar notification. Every 500ms or 5% increment.
<i>update_↔ complete_status</i>	<ul style="list-style-type: none"> • See table below.
<i>severity</i>	<ul style="list-style-type: none"> • 0x01 - Mandatory • 0x02 - Optional
<i>versionlength</i>	<ul style="list-style-type: none"> • Length of FW Version string in bytes
<i>version</i>	<ul style="list-style-type: none"> • FW Version string in ASCII (Max 256 characters)
<i>namelength</i>	<ul style="list-style-type: none"> • Length Package Name string in bytes
<i>package_name</i>	<ul style="list-style-type: none"> • Package Name in UCS2 (Max 256 characters)
<i>descriptionlength</i>	<ul style="list-style-type: none"> • Length of description in bytes
<i>description</i>	<ul style="list-style-type: none"> • Description of Update Package in USC2 (Max 256 characters)
<i>sessionType</i>	<ul style="list-style-type: none"> • 0x00 - Client initiated • 0x01 - Network initiated

8.482.2 Field Documentation

- 8.482.2.1 **BYTE** omaDmFotaTlv::description[256]
- 8.482.2.2 **USHORT** omaDmFotaTlv::descriptionlength
- 8.482.2.3 **ULONG** omaDmFotaTlv::fwdloadsize
- 8.482.2.4 **ULONG** omaDmFotaTlv::fwloadComplete
- 8.482.2.5 **USHORT** omaDmFotaTlv::namelength
- 8.482.2.6 **BYTE** omaDmFotaTlv::package_name[256]
- 8.482.2.7 **BYTE** omaDmFotaTlv::sessionType
- 8.482.2.8 **BYTE** omaDmFotaTlv::severity
- 8.482.2.9 **BYTE** omaDmFotaTlv::state
- 8.482.2.10 **USHORT** omaDmFotaTlv::updateCompleteStatus
- 8.482.2.11 **BYTE** omaDmFotaTlv::userInputReq
- 8.482.2.12 **USHORT** omaDmFotaTlv::userInputTimeout
- 8.482.2.13 **BYTE** omaDmFotaTlv::version[256]
- 8.482.2.14 **USHORT** omaDmFotaTlv::versionlength

8.483 omaDmFotaTlvExt Struct Reference

Data Fields

- [BYTE state](#)
- [BYTE reserved](#)
- [USHORT userInputTimeout](#)
- [ULONG packageSize](#)
- [ULONG receivedBytes](#)
- [USHORT fumoResultCode](#)
- [USHORT versionlength](#)
- [BYTE version \[256\]](#)
- [USHORT namelength](#)
- [BYTE package_name \[256\]](#)
- [USHORT descriptionlength](#)
- [BYTE description \[256\]](#)

8.483.1 Detailed Description

This structure will hold the SwiOmaDmFota session parameters information.

Parameters

<i>state</i>	<ul style="list-style-type: none"> • 0x01 - No Firmware available • 0x02 - reserved • 0x03 - Update progress bar, UI screen 7[1] should be displayed • 0x04 - reserved • 0x05 - FUMO image download success, UI screen 8[1] should be displayed • 0x06 - reserved • 0x07 - FUMO image installation result, used to return error code. • 0x08 - FUMO session start • 0x09 - UI Screen 7[1] with 0 percent bar should be displayed • 0x0A - FUMO image installation is cancelled by user. • 0x0B - FUMO session fail • 0x0C - Device is sending a report to OMA Server. • 0x0D - Report to server success. • 0x0E - Report to server fails. • 0x0F - reserved • 0x10 - FUMO session is cancelled before image download success. • 0x11 - UI Screen 16[1] should be displayed, FUMO delay option, OMA task is blocked until a valid response is received.
<i>reserved</i>	- For sprint reserved
<i>user_input_↔ timeout</i>	<ul style="list-style-type: none"> • How many seconds OMA task stop to wait for user/host response.
<i>packageSize</i>	<ul style="list-style-type: none"> • The size (in bytes) of the firmware update package (only valid for states 3/5/7).
<i>receivedBytes</i>	<ul style="list-style-type: none"> • The number of bytes downloaded. Useful for FUMO state 3.
<i>fumoResultCode</i>	<ul style="list-style-type: none"> • Used when fumo state is 7/11. REsult code of FUMO image installation <ul style="list-style-type: none"> – 200 image install success • Others: image install fail
<i>versionlength</i>	<ul style="list-style-type: none"> • Length of FW Version string in bytes
<i>version</i>	<ul style="list-style-type: none"> • FW Version string in ASCII (Max 256 characters)
<i>namelength</i>	<ul style="list-style-type: none"> • Length Package Name string in bytes
<i>package_name</i>	<ul style="list-style-type: none"> • Package Name in UCS2 (Max 256 characters)
<i>descriptionlength</i>	<ul style="list-style-type: none"> • Length of description in bytes

<i>description</i>	<ul style="list-style-type: none">• Description of Update Package in USC2 (Max 256 characters)
--------------------	--

8.483.2 Field Documentation

8.483.2.1 **BYTE** omaDmFotaTlvExt::description[256]

8.483.2.2 **USHORT** omaDmFotaTlvExt::descriptionlength

8.483.2.3 **USHORT** omaDmFotaTlvExt::fumoResultCode

8.483.2.4 **USHORT** omaDmFotaTlvExt::namelength

8.483.2.5 **BYTE** omaDmFotaTlvExt::package_name[256]

8.483.2.6 **ULONG** omaDmFotaTlvExt::packageSize

8.483.2.7 **ULONG** omaDmFotaTlvExt::receivedBytes

8.483.2.8 **BYTE** omaDmFotaTlvExt::reserved

8.483.2.9 **BYTE** omaDmFotaTlvExt::state

8.483.2.10 **USHORT** omaDmFotaTlvExt::userInputTimeout

8.483.2.11 **BYTE** omaDmFotaTlvExt::version[256]

8.483.2.12 **USHORT** omaDmFotaTlvExt::versionlength

8.484 omaDmNotificationsTlv Struct Reference

Data Fields

- [BYTE](#) notification
- [USHORT](#) sessionStatus

8.484.1 Field Documentation

8.484.1.1 **BYTE** omaDmNotificationsTlv::notification

8.484.1.2 **USHORT** omaDmNotificationsTlv::sessionStatus

8.485 operatorNameString Struct Reference

Data Fields

- [BYTE](#) PLMNName [255]

8.485.1 Detailed Description

This structure contains Operator Name String as defined in CPHS4_2.WW6(Feb 27, 1997) (Section B.4.1.2) from multiple sources.

Parameters

<i>PLMNName</i>	<ul style="list-style-type: none">• PLMN name must be coded in a default 7-bit alphabet with b8 set to 0.
-----------------	---

8.485.2 Field Documentation

8.485.2.1 BYTE operatorNameString::PLMNName[255]

8.486 OperatorPLMNData Struct Reference

Data Fields

- [BYTE mcc](#) [3]
- [BYTE mnc](#) [3]
- [WORD lac1](#)
- [WORD lac2](#)
- [BYTE PLMNRecID](#)

8.486.1 Detailed Description

This structure contains Operator PLMN Data from multiple sources.

Parameters

<i>mcc</i>	<ul style="list-style-type: none">• MCC in ASCII string (a value of D in any of the digits is to be used to indicate a "wild" value for that corresponding digit).
<i>mnc</i>	<ul style="list-style-type: none">• MNC in ASCII string (a value of D in any of the digits is to be used to indicate a "wild" value for that corresponding digit; digit 3 in MNC is optional and when not present, will be set as ASCII F).
<i>lac1</i>	<ul style="list-style-type: none">• Location area code 1.
<i>lac2</i>	<ul style="list-style-type: none">• Location area code 1.
<i>PLMNRecID</i>	<ul style="list-style-type: none">• PLMN network name record identifier.

8.486.2 Field Documentation

8.486.2.1 WORD OperatorPLMNData::lac1

8.486.2.2 WORD OperatorPLMNData::lac2

8.486.2.3 BYTE OperatorPLMNData::mcc[3]

8.486.2.4 BYTE OperatorPLMNData::mnc[3]

8.486.2.5 BYTE OperatorPLMNData::PLMNRecID

8.487 operatorPLMNList Struct Reference

Data Fields

- [WORD numInstance](#)
- [OperatorPLMNData PLMNData](#) [255]

8.487.1 Detailed Description

This structure contains Operator PLMN List as defined in 3GPP TS 31.102 (Section 4.2.59) from multiple sources.

Parameters

<i>numInstance</i>	<ul style="list-style-type: none">• Number of sets of the elements.
<i>PLMNData</i>	<ul style="list-style-type: none">• Refer OperatorPLMNData for details (Optional).

8.487.2 Field Documentation

8.487.2.1 WORD operatorPLMNList::numInstance

8.487.2.2 OperatorPLMNData operatorPLMNList::PLMNData[255]

8.488 pack_dms_ActivateAutomatic_t Struct Reference

Data Fields

- uint8_t [actCode](#) [81]

8.488.1 Detailed Description

This structure contains UIM activate automation information

Parameters

<i>spc</i> [<i>IN</i>]	<ul style="list-style-type: none"> Service programming code in ASCII format (digits 0 to 9 only)
--------------------------	---

8.488.2 Field Documentation

8.488.2.1 `uint8_t pack_dms_ActivateAutomatic_t::actCode[81]`

8.489 `pack_dms_GetCustFeaturesV2_t` Struct Reference

Data Fields

- `uint8_t cust_id` [64+1]
- `uint8_t list_type`
- `uint16_t Tlvresult`

8.489.1 Detailed Description

This structure contains which customization id or the list type want to retrieve from modem. This TLV is only applicable for 9x30 modules so far

Parameters

<i>cust_id</i>	<ul style="list-style-type: none"> Customization ID (Maximum 64 bytes)
<i>list_type</i>	<ul style="list-style-type: none"> list type requested
<i>Tlvresult</i>	<ul style="list-style-type: none"> Pack Result

8.489.2 Field Documentation

8.489.2.1 `uint8_t pack_dms_GetCustFeaturesV2_t::cust_id[64+1]`

8.489.2.2 `uint8_t pack_dms_GetCustFeaturesV2_t::list_type`

8.489.2.3 `uint16_t pack_dms_GetCustFeaturesV2_t::Tlvresult`

8.490 pack_dms_ResetToFactoryDefaults_t Struct Reference

Data Fields

- uint8_t [spc](#) [6]

8.490.1 Detailed Description

This structure contains UIM reset to factory default information

Parameters

<i>spc</i> [IN]	<ul style="list-style-type: none">• Service programming code in ASCII format (digits 0 to 9 only)
-----------------	---

8.490.2 Field Documentation

8.490.2.1 uint8_t pack_dms_ResetToFactoryDefaults_t::spc[6]

8.491 pack_dms_SetActivationStatusCallback_t Struct Reference

Data Fields

- uint8_t [activationState](#)

8.491.1 Detailed Description

This structure is used to store Service Activation Status parameter .

Parameters

<i>activationState</i>	<ul style="list-style-type: none">• Service activation state.• Values<ul style="list-style-type: none">– 0 - Do not report– 1 - Report activation state changes
------------------------	---

8.491.2 Field Documentation

8.491.2.1 uint8_t pack_dms_SetActivationStatusCallback_t::activationState

8.492 pack_dms_SetCrashAction_t Struct Reference

Data Fields

- uint8_t [crashAction](#)

8.492.1 Detailed Description

Modem action in case of a crash

Parameters

<i>crashAction</i>	<ul style="list-style-type: none"> • 0 - USB Memory Download. Modem will reset after a crash and will stay in USB download mode with only DM port enumerated. • 1 - Reset. Modem will reset and come back in ONLINE mode. Minimal crash data will be available and can be extracted with at!gcdump? AT command • 2 - No Action
--------------------	---

8.492.2 Field Documentation

8.492.2.1 uint8_t pack_dms_SetCrashAction_t::crashAction

8.493 pack_dms_SetCustFeature_t Struct Reference

Data Fields

- uint32_t [GpsEnable](#)
- uint8_t [DisableIMSI](#)
- uint16_t [IPFamSupport](#)
- uint8_t [RMAutoConnect](#)
- uint8_t [GPSSel](#)
- uint8_t [SMSSupport](#)
- uint8_t [IsVoiceEnabled](#)
- uint8_t [DHCPRelayEnabled](#)
- uint8_t [GPSLPM](#)

8.493.1 Field Documentation

8.493.1.1 uint8_t pack_dms_SetCustFeature_t::DHCPRelayEnabled

8.493.1.2 uint8_t pack_dms_SetCustFeature_t::DisableIMSI

8.493.1.3 uint32_t pack_dms_SetCustFeature_t::GpsEnable

8.493.1.4 uint8_t pack_dms_SetCustFeature_t::GPSLPM

8.493.1.5 `uint8_t pack_dms_SetCustFeature_t::GPSSel`

8.493.1.6 `uint16_t pack_dms_SetCustFeature_t::IPFamSupport`

8.493.1.7 `uint8_t pack_dms_SetCustFeature_t::IsVoiceEnabled`

8.493.1.8 `uint8_t pack_dms_SetCustFeature_t::RMAutoConnect`

8.493.1.9 `uint8_t pack_dms_SetCustFeature_t::SMSSupport`

8.494 pack_dms_SetCustFeaturesV2_t Struct Reference

Data Fields

- `uint8_t cust_id` [64+1]
- `uint16_t value_length`
- `uint8_t cust_value` [8+1]
- `uint16_t Tlvresult`

8.494.1 Detailed Description

This structure contains customization settings set to modem pack

Parameters

<i>cust_id</i>	<ul style="list-style-type: none">• Customization ID (Maximum 64 bytes)
<i>value_length</i>	<ul style="list-style-type: none">• length of cust_value field
<i>cust_value</i>	<ul style="list-style-type: none">• Customization Setting Value (Maximum 8 bytes)
<i>Tlvresult</i>	<ul style="list-style-type: none">• Pack Result

8.494.2 Field Documentation

8.494.2.1 `uint8_t pack_dms_SetCustFeaturesV2_t::cust_id`[64+1]

8.494.2.2 `uint8_t pack_dms_SetCustFeaturesV2_t::cust_value`[8+1]

8.494.2.3 `uint16_t pack_dms_SetCustFeaturesV2_t::Tlvresult`

8.494.2.4 uint16_t pack_dms_SetCustFeaturesV2_t::value_length

8.495 pack_dms_SetEventReport_t Struct Reference

Data Fields

- uint8_t [mode](#)

8.495.1 Field Documentation

8.495.1.1 uint8_t pack_dms_SetEventReport_t::mode

8.496 pack_dms_SetPower_t Struct Reference

Data Fields

- uint32_t [mode](#)
- uint16_t [Tlvresult](#)

8.496.1 Field Documentation

8.496.1.1 uint32_t pack_dms_SetPower_t::mode

8.496.1.2 uint16_t pack_dms_SetPower_t::Tlvresult

8.497 pack_dms_SetUSBComp_t Struct Reference

Data Fields

- uint8_t [USBComp](#)
- uint16_t [Tlvresult](#)

8.497.1 Field Documentation

8.497.1.1 uint16_t pack_dms_SetUSBComp_t::Tlvresult

8.497.1.2 uint8_t pack_dms_SetUSBComp_t::USBComp

8.498 pack_dms_SLQSDmsSwilIndicationRegister_t Struct Reference

Data Fields

- uint8_t [resetInfoInd](#)

8.498.1 Detailed Description

Parameters

<i>resetInfoInd</i> [IN]	<ul style="list-style-type: none">• Values<ul style="list-style-type: none">– 0 - Disable– 1 - Enable
--------------------------	--

8.498.2 Field Documentation

8.498.2.1 uint8_t pack_dms_SLQSDmsSwiIndicationRegister_t::resetInfoInd

8.499 pack_dms_SLQSSwiGetCrashInfo_t Struct Reference

Data Fields

- uint8_t [clear](#)

8.499.1 Detailed Description

This structure contains [crashInfoParams](#)

Parameters

<i>clear</i>	<ul style="list-style-type: none">• 0 - do not clear crash data after response• 1 - clear crash data after response
--------------	--

8.499.2 Field Documentation

8.499.2.1 uint8_t pack_dms_SLQSSwiGetCrashInfo_t::clear

8.500 pack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference

Data Fields

- uint8_t * [pDestSMSNum](#)
- uint8_t * [pDestSMSContent](#)

8.500.1 Detailed Description

Parameters

<i>pDestSMS</i> ↔ <i>Num</i> [IN]	<ul style="list-style-type: none"> SMS Destination Number as string of 8 bit ASCII Characters Max 20 chars. Optional parameter.
<i>pDestSMS</i> ↔ <i>Content</i> [IN]	<ul style="list-style-type: none"> SMS Content as a string of 8 bit ASCII text characters Max 160 chars. Optional parameter.

8.500.2 Field Documentation

8.500.2.1 uint8_t* pack_dms_SLQSSwiSetDyingGaspCfg_t::pDestSMSContent

8.500.2.2 uint8_t* pack_dms_SLQSSwiSetDyingGaspCfg_t::pDestSMSNum

8.501 pack_dms_SLQSSwiSetHostDevInfo_t Struct Reference

Data Fields

- char [manString](#) [255]
- char [modelString](#) [255]
- char [swVerString](#) [255]
- char [plasmaIDString](#) [255]

8.501.1 Detailed Description

This structure contains SWI set host device info unpack information

Parameters

<i>manString</i> [IN]	<ul style="list-style-type: none"> optional parameter, host device manufacture
<i>modelString</i> [IN]	<ul style="list-style-type: none"> optional parameter, host device model
<i>swVerString</i> [IN]	<ul style="list-style-type: none"> optional parameter, host device software version
<i>plasmaID</i> ↔ <i>String</i> [IN]	<ul style="list-style-type: none"> optional parameter, host device plasma ID

8.501.2 Field Documentation

8.501.2.1 char pack_dms_SLQSSwiSetHostDevInfo_t::manString[255]

8.501.2.2 char pack_dms_SLQSSwiSetHostDevInfo_t::modelString[255]

8.501.2.3 char pack_dms_SLQSSwiSetHostDevInfo_t::plasmaIDString[255]

8.501.2.4 char pack_dms_SLQSSwiSetHostDevInfo_t::swVerString[255]

8.502 pack_dms_SLQSSwiSetOSInfo_t Struct Reference

Data Fields

- char [nameString](#) [255]
- char [versionString](#) [255]

8.502.1 Detailed Description

This structure contains SWI set host OS info pack information

Parameters

<i>nameString</i> [IN]	<ul style="list-style-type: none">• optional parameter, host device manufacture
<i>VersionString</i> [IN]	<ul style="list-style-type: none">• optional parameter, host device model

8.502.2 Field Documentation

8.502.2.1 char pack_dms_SLQSSwiSetOSInfo_t::nameString[255]

8.502.2.2 char pack_dms_SLQSSwiSetOSInfo_t::versionString[255]

8.503 pack_dms_UIMChangePIN_t Struct Reference

Data Fields

- uint8_t [id](#)
- uint8_t [oldValue](#) [255]
- uint8_t [newValue](#) [255]

8.503.1 Detailed Description

This structure contains UIM Unblock PIN Information

Parameters

<i>id</i> [IN]	<ul style="list-style-type: none"> • PIN ID <ul style="list-style-type: none"> – 1 (PIN1 / CHV1) – 2 (PIN2 / CHV2)
<i>oldValue</i> [IN]	<ul style="list-style-type: none"> • Old PIN value of PIN to change
<i>newValue</i> [IN]	<ul style="list-style-type: none"> • New PIN value of PIN to change

8.503.2 Field Documentation

8.503.2.1 `uint8_t pack_dms_UIMChangePIN_t::id`

8.503.2.2 `uint8_t pack_dms_UIMChangePIN_t::newValue[255]`

8.503.2.3 `uint8_t pack_dms_UIMChangePIN_t::oldValue[255]`

8.504 `pack_dms_UIMGetControlKeyStatus_t` Struct Reference

Data Fields

- `uint8_t facility`

8.504.1 Detailed Description

This structure contains UIM get control key status information

Parameters

<i>facility</i> [IN]	<ul style="list-style-type: none"> • MT or network facility <ul style="list-style-type: none"> – 0 - Network Personalization (PN) – 1 - Network Subset Personalization (PU) – 2 - Service Provider Personalization (PP) – 3 - Corporate Personalization (PC) – 4 - UIM Personalization (PF)
----------------------	--

8.504.2 Field Documentation

8.504.2.1 `uint8_t pack_dms_UIMGetControlKeyStatus_t::facility`

8.505 pack_dms_UIMGetICCID_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.505.1 Detailed Description

This structure contains UIM Get ICCID pack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Pack result.
------------------	--

8.505.2 Field Documentation

8.505.2.1 uint16_t pack_dms_UIMGetICCID_t::Tlvresult

8.506 pack_dms_UIMSetControlKeyProtection_t Struct Reference

Data Fields

- uint8_t [facility](#)
- uint8_t [facilityState](#)
- uint8_t [facilityCk](#) [8]

8.506.1 Detailed Description

This structure contains UIM Set control key protection information

Parameters

<i>facility</i> [IN]	<ul style="list-style-type: none">• MT or network facility<ul style="list-style-type: none">– 0 - Network Personalization (PN)– 1 - Network Subset Personalization (PU)– 2 - Service Provider Personalization (PP)– 3 - Corporate Personalization (PC)– 4 - UIM Personalization (PF)
<i>facilityState</i> [IN]	<ul style="list-style-type: none">• UIM facility state
<i>facilityCk</i> [IN]	<ul style="list-style-type: none">• Facility depersonalization control key, string in ASCII text (maximum 8 bytes)

8.506.2 Field Documentation

8.506.2.1 uint8_t pack_dms_UIMSetControlKeyProtection_t::facility

8.506.2.2 uint8_t pack_dms_UIMSetControlKeyProtection_t::facilityCk[8]

8.506.2.3 uint8_t pack_dms_UIMSetControlKeyProtection_t::facilityState

8.507 pack_dms_UIMSetPINProtection_t Struct Reference

Data Fields

- uint8_t [id](#)
- uint8_t [bEnable](#)
- uint8_t [value](#) [255]

8.507.1 Detailed Description

Set activation status pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter This structure contains PIN Protection Information
	<i>id</i> [IN]	<ul style="list-style-type: none"> • PIN ID <ul style="list-style-type: none"> – 1 (PIN1 / CHV1) – 2 (PIN2 / CHV2)
	<i>bEnable</i> [IN]	<ul style="list-style-type: none"> • Enable/disable PIN protection, 0 = Disable
	<i>pValue</i> [IN]	<ul style="list-style-type: none"> • PIN value of the PIN to be enabled/disabled

8.507.2 Field Documentation

8.507.2.1 uint8_t pack_dms_UIMSetPINProtection_t::bEnable

8.507.2.2 uint8_t pack_dms_UIMSetPINProtection_t::id

8.507.2.3 uint8_t pack_dms_UIMSetPINProtection_t::value[255]

8.508 pack_dms_UIMUnlockControlKey_t Struct Reference

Data Fields

- uint8_t [facility](#)
- uint8_t [facilityCk](#) [8]

8.508.1 Detailed Description

This structure contains UIM unblock Control Key information

Parameters

<i>facility</i> [IN]	<ul style="list-style-type: none">• MT or network facility<ul style="list-style-type: none">– 0 - Network Personalization (PN)– 1 - Network Subset Personalization (PU)– 2 - Service Provider Personalization (PP)– 3 - Corporate Personalization (PC)– 4 - UIM Personalization (PF)
<i>facilityCk</i> [IN]	<ul style="list-style-type: none">• Facility depersonalization control key, string in ASCII text (maximum 8 bytes)

8.508.2 Field Documentation

8.508.2.1 uint8_t pack_dms_UIMUnlockControlKey_t::facility

8.508.2.2 uint8_t pack_dms_UIMUnlockControlKey_t::facilityCk[8]

8.509 pack_dms_UIMUnlockPIN_t Struct Reference

Data Fields

- uint8_t [id](#)
- uint8_t [pukValue](#) [255]
- uint8_t [newPin](#) [255]

8.509.1 Detailed Description

This structure contains UIM Unblock PIN Information

Parameters

<i>id</i> [IN]	<ul style="list-style-type: none"> PIN ID <ul style="list-style-type: none"> 1 (PIN1 / CHV1) 2 (PIN2 / CHV2)
<i>pukValue</i> [IN]	<ul style="list-style-type: none"> PUK value of PIN to be unblocked
<i>newPin</i> [IN]	<ul style="list-style-type: none"> New PIN value for the PIN to be unblocked

8.509.2 Field Documentation

8.509.2.1 uint8_t pack_dms_UIMUnblockPIN_t::id

8.509.2.2 uint8_t pack_dms_UIMUnblockPIN_t::newPin[255]

8.509.2.3 uint8_t pack_dms_UIMUnblockPIN_t::pukValue[255]

8.510 pack_dms_UIMVerifyPIN_t Struct Reference

Data Fields

- uint8_t [id](#)
- uint8_t [value](#) [255]

8.510.1 Detailed Description

This structure contains PIN Value Information

Parameters

<i>id</i> [IN]	<ul style="list-style-type: none"> PIN ID <ul style="list-style-type: none"> 1 (PIN1 / CHV1) 2 (PIN2 / CHV2)
<i>pValue</i> [IN]	<ul style="list-style-type: none"> PIN value of the PIN to be enabled/disabled

8.510.2 Field Documentation

8.510.2.1 uint8_t pack_dms_UIMVerifyPIN_t::id

8.510.2.2 uint8_t pack_dms_UIMVerifyPIN_t::value[255]

8.511 pack_fms_GetImagesPreference_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.511.1 Detailed Description

This structure contains the Get Image Preference information pack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Pack result
------------------	---

8.511.2 Field Documentation

8.511.2.1 uint16_t pack_fms_GetImagesPreference_t::Tlvresult

8.512 pack_fms_GetStoredImages_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.512.1 Detailed Description

This structure contains the Get Stored Images pack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Pack result
------------------	---

8.512.2 Field Documentation

8.512.2.1 uint16_t pack_fms_GetStoredImages_t::Tlvresult

8.513 pack_fms_SetImagesPreference_t Struct Reference

Data Fields

- uint32_t [imageListSize](#)
- FMSPrefImageList * [pImageList](#)
- uint32_t [bForceDownload](#)
- uint8_t [modemindex](#)
- uint16_t [Tlvresult](#)

8.513.1 Detailed Description

This structure contains the Set Images Preference pack

Parameters

<i>imageListSize</i>	<ul style="list-style-type: none"> • Image List Size
<i>pImageList</i>	<ul style="list-style-type: none"> • Image List • See FMSPrefImageList
<i>bForceDownload</i>	<ul style="list-style-type: none"> • 0 - Not Force Download. • 1 - Force Download.
<i>modemindex</i>	<ul style="list-style-type: none"> • Modem Index.
<i>Tlvresult</i>	<ul style="list-style-type: none"> • Unpack result

8.513.2 Field Documentation

8.513.2.1 uint32_t pack_fms_SetImagesPreference_t::bForceDownload

8.513.2.2 uint32_t pack_fms_SetImagesPreference_t::imageListSize

8.513.2.3 uint8_t pack_fms_SetImagesPreference_t::modemindex

8.513.2.4 FMSPrefImageList* pack_fms_SetImagesPreference_t::pImageList

8.513.2.5 uint16_t pack_fms_SetImagesPreference_t::Tlvresult

8.514 pack_loc_Delete_Assist_Data_t Struct Reference

Data Fields

- [loc_SVInfo](#) * [pSVInfo](#)

- [loc_GnssData](#) * [pGnssData](#)
- [loc_CellDb](#) * [pCellDb](#)
- [loc_ClkInfo](#) * [pClkInfo](#)
- [loc_BdsSVInfo](#) * [pBdsSVInfo](#)
- [uint16_t](#) [Tlvresult](#)

8.514.1 Detailed Description

This structure contains LOC delete assist data pack

Parameters

<i>pSVInfo</i>	<ul style="list-style-type: none"> • Pointer to struct loc_SVInfo. See loc_SVInfo for more information
<i>pGnssData</i>	<ul style="list-style-type: none"> • Pointer to struct loc_GnssData. See loc_GnssData for more information
<i>pCellDb</i>	<ul style="list-style-type: none"> • Pointer to struct loc_CellDb. See loc_CellDb for more information
<i>pClkInfo</i>	<ul style="list-style-type: none"> • Pointer to struct loc_ClkInfo. See loc_ClkInfo for more information
<i>pBdsSVInfo</i>	<ul style="list-style-type: none"> • Pointer to struct loc_BdsSVInfo. See loc_BdsSVInfo for more information
<i>Tlvresult</i>	<ul style="list-style-type: none"> • Pack delete assist data request result.

8.514.2 Field Documentation

8.514.2.1 [loc_BdsSVInfo](#)* [pack_loc_Delete_Assist_Data_t::pBdsSVInfo](#)

8.514.2.2 [loc_CellDb](#)* [pack_loc_Delete_Assist_Data_t::pCellDb](#)

8.514.2.3 [loc_ClkInfo](#)* [pack_loc_Delete_Assist_Data_t::pClkInfo](#)

8.514.2.4 [loc_GnssData](#)* [pack_loc_Delete_Assist_Data_t::pGnssData](#)

8.514.2.5 [loc_SVInfo](#)* [pack_loc_Delete_Assist_Data_t::pSVInfo](#)

8.514.2.6 [uint16_t](#) [pack_loc_Delete_Assist_Data_t::Tlvresult](#)

8.515 pack_loc_EventRegister_t Struct Reference

Data Fields

- [uint64_t](#) [eventRegister](#)
- [uint16_t](#) [Tlvresult](#)

8.515.1 Detailed Description

This structure contains the Parameter for RegisterEvents

Parameters

<i>eventRegister</i>	<ul style="list-style-type: none"> • Specifies the events that the control point is interested in receiving. -Values <ul style="list-style-type: none"> – 0x00000001 - to receive position report event indications – 0x00000002 - to receive satellite report event indications. These reports are sent at a 1 Hz rate. – 0x00000004 - to receive NMEA reports for position and satellites in view. The report is at a 1 Hz rate. – 0x00000008 - to receive NI Notify/Verify request event indications – 0x00000010 - to receive time injection request event indications. – 0x00000020 - to receive predicted orbits request event indications. – 0x00000040 - to receive position injection request event indications. – 0x00000080 - to receive engine state report event indications. – 0x00000100 - to receive fix session status report event indications. – 0x00000200 - to receive Wi-Fi position request event indications. – 0x00000400 - to receive notifications from the location engine indicating its readiness to accept data from the sensors (accelerometer, gyroscope, etc.). – 0x00000800 - to receive time sync requests from the GPS engine. Time sync enables the GPS engine to synchronize its clock with the sensor processor's clock. – 0x00001000 - to receive Stationary Position Indicator (SPI) streaming report indications. – 0x00002000 - to receive location server requests. These requests are generated when the service wishes to establish a connection with a location server. – 0x00004000 - to receive notifications related to network-initiated Geofences. These events notify the client when a network-initiated Geofence is added, deleted, or edited. – 0x00008000 - to receive Geofence alerts. These alerts are generated to inform the client of the changes that may affect a Geofence, e.g., if GPS is turned off or if the network is unavailable. – 0x00010000 - to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach report is for a single Geofence. – 0x00020000 - to register for pedometer control requests from the location engine. The location engine sends this event to control the injection of pedometer reports. – 0x00040000 - to register for motion data control requests from the location engine. The location engine sends this event to control the injection of motion data. – 0x00080000 - to receive notification when a batch is full. The location engine sends this event to notify of Batch Full for ongoing batching session. – 0x00100000 - to receive position report indications along with an ongoing batching session. The location engine sends this event to notify the batched position report while a batching session is ongoing. – 0x00200000 - to receive Wi-Fi Access Point (AP) data inject request event indications. – 0x00400000 - to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach notification is for multiple Geofences. Breaches from multiple Geofences are all batched and sent in the same notification. – 0x00800000 - to receive notifications from the location engine indicating its readiness to accept vehicle data (vehicle accelerometer, vehicle angular rate, vehicle odometry, etc.). – 0x01000000 - to receive system clock and satellite measurement report events (system clock, SV time, Doppler, etc.). – 0x02000000 - to receive satellite position reports as polynomials. Reports are generated only for the GNSS satellite constellations that are enabled using QMI_LOC_SET_GNSS←_CONSTELL_REPORT_CONFIG.
----------------------	---

Note

Multiple events can be registered by OR the individual masks and sending them in this TLV. All unused bits in this mask must be set to 0.

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none"> • Pack result.
------------------	--

8.515.2 Field Documentation

8.515.2.1 `uint64_t pack_loc_EventRegister_t::eventRegister`

8.515.2.2 `uint16_t pack_loc_EventRegister_t::Tlvresult`

8.516 pack_loc_SetExtPowerState_t Struct Reference**Data Fields**

- `uint32_t extPowerState`
- `uint16_t Tlvresult`

8.516.1 Detailed Description

This structure contains the Parameter External Power Source State pack.

Parameters

<i>extPowerState</i>	<ul style="list-style-type: none"> • Specifies the Power state; injected by the control point. • Values <ul style="list-style-type: none"> – 0 - Device is not connected to an external power source – 1 - Device is connected to an external power source – 2 - Unknown external power state
<i>Tlvresult</i>	<ul style="list-style-type: none"> • Pack result.

8.516.2 Field Documentation

8.516.2.1 `uint32_t pack_loc_SetExtPowerState_t::extPowerState`

8.516.2.2 `uint16_t pack_loc_SetExtPowerState_t::Tlvresult`

8.517 pack_loc_SetOperationMode_t Struct Reference

Data Fields

- uint32_t [mode](#)
- uint16_t [Tlvresult](#)

8.517.1 Detailed Description

This structure contains Set Operation Mode pack

Parameters

<i>mode</i>	<ul style="list-style-type: none">• 0 - Default Mode.
<i>Tlvresult</i>	<ul style="list-style-type: none">• Pack result.

8.517.2 Field Documentation

8.517.2.1 uint32_t pack_loc_SetOperationMode_t::mode

8.517.2.2 uint16_t pack_loc_SetOperationMode_t::Tlvresult

8.518 pack_loc_SLQSLOCGetBestAvailPos_t Struct Reference

Data Fields

- uint32_t [xid](#)
- uint16_t [Tlvresult](#)

8.518.1 Detailed Description

This structure contains Set Operation Mode pack

Parameters

<i>xid</i>	<ul style="list-style-type: none">• Identifies the transaction.• The transaction ID is returned in the Get Best Available Position indication.
<i>Tlvresult</i>	<ul style="list-style-type: none">• Pack result.

8.518.2 Field Documentation

8.518.2.1 `uint16_t pack_loc_SLQSLOCGetBestAvailPos_t::Tlvresult`

8.518.2.2 `uint32_t pack_loc_SLQSLOCGetBestAvailPos_t::xid`

8.519 `pack_loc_SLQSLOCInjectPosition_t` Struct Reference

Data Fields

- double `latitude`
- int `has_latitude`
- double `longitude`
- int `has_longitude`
- float `horUncCircular`
- int `has_horUncCircular`
- `uint8_t` `horConfidence`
- int `has_horConfidence`
- `uint32_t` `horReliability`
- int `has_horReliability`
- float `altitudeWrtEllipsoid`
- int `has_altitudeWrtEllipsoid`
- float `altitudeWrtMeanSeaLevel`
- int `has_altitudeWrtMeanSeaLevel`
- float `vertUnc`
- int `has_vertUnc`
- `uint8_t` `vertConfidence`
- int `has_vertConfidence`
- `uint32_t` `vertReliability`
- int `has_vertReliability`
- `altSrcInfo_t` `altitudeSrcInfo`
- int `has_altitudeSrcInfo`
- `uint64_t` `timestampUtc`
- int `has_timestampUtc`
- `uint32_t` `timestampAge`
- int `has_timestampAge`
- `uint32_t` `positionSrc`
- int `has_positionSrc`
- float `rawHorUncCircular`
- int `has_rawHorUncCircular`
- `uint8_t` `rawHorConfidence`
- int `has_rawHorConfidence`

8.519.1 Detailed Description

This structure contains LOC Inject Position parameters Please check has_<Param_Name> field for presence of optional parameters

Parameters

<i>latitude</i>	<ul style="list-style-type: none"> • Optional parameter • Type - Floating point • Units - Degrees • Range - -90.0 to 90.0 • Positive values indicate northern latitude • Negative values indicate southern latitude • Note - This field must be specified together with pLongitude and pHorUncCircular.
<i>longitude</i>	<ul style="list-style-type: none"> • Optional parameter • Type - Floating point • Units - Degrees • Range - -180.0 to 180.0 • Positive values indicate eastern latitude • Negative values indicate western latitude • Note - This field must be specified together with pLatitude and pHorUncCircular.
<i>horUncCircular</i>	<ul style="list-style-type: none"> • Optional parameter • Horizontal position uncertainty. • Units - Meters • Note - This field must be specified together with pLatitude and pLongitude.
<i>horConfidence</i>	<ul style="list-style-type: none"> • Optional parameter • Horizontal confidence. • Units - Percent • Values <ul style="list-style-type: none"> – Valid Values - 1 to 99 – Invalid Values - 0, 101 to 255 – If 100 is received, reinterpret to 99 • Note - This field must be specified together with horizontal uncertainty. If not specified when pHorUncCircular is set, the default value is 50.
<i>horReliability</i>	<ul style="list-style-type: none"> • Optional parameter

- Values
 - 0 - Location reliability is not set.
 - 1 - Location reliability is very low; use it at your own risk
 - 2 - Location reliability is low; little or no cross-checking is possible.
 - 3 - Location reliability is medium; limited cross-check passed

- 4 - Location reliability is high; strong cross-check passed

Parameters

<i>altitudeWrt_↔ Ellipsoid</i>	<ul style="list-style-type: none"> • Optional parameter • Altitude With Respect to Ellipsoid. • Units - Meters • Values <ul style="list-style-type: none"> – Positive - height – Negative = depth
<i>altitudeWrt_↔ MeanSeaLevel</i>	<ul style="list-style-type: none"> • Optional parameter • Altitude With Respect to Sea Level. • Units - Meters
<i>vertUnc</i>	<ul style="list-style-type: none"> • Optional parameter • Vertical uncertainty. • Units - Meters • Note - This is mandatory if either pAltitudeWrtEllipsoid or pAltitudeWrtMeanSeaLevel is specified.
<i>vertConfidence</i>	<ul style="list-style-type: none"> • Optional parameter • Vertical confidence. • Units - Percentage • Values <ul style="list-style-type: none"> – Valid Values - 0 to 99 – Invalid Values - 0, 100-256 – If 100 is received, reinterpret to 99 • Note - This field must be specified together with the vertical uncertainty. If not specified, the default value will be 50.
<i>vertReliability</i>	<ul style="list-style-type: none"> • Optional parameter

- Values
 - 0 - Location reliability is not set.
 - 1 - Location reliability is very low; use it at your own risk.
 - 2 - Location reliability is low; little or no cross-checking is possible
 - 3 - Location reliability is medium; limited cross-check passed
 - 4 - Location reliability is high; strong cross-check passed

Parameters

<i>altitudeSrcInfo</i>	<ul style="list-style-type: none"> • Optional parameter
------------------------	--

- Pointer to struct [altitudeSrcInfo](#). See [altitudeSrcInfo](#) for more information

Parameters

<i>timestampUtc</i>	<ul style="list-style-type: none"> • Optional parameter • UTC timestamp • Units - Milliseconds since Jan. 1, 1970
<i>timestampAge</i>	<ul style="list-style-type: none"> • Optional parameter • Position age, which is an estimate of how long ago this fix was made. • Units - Milliseconds
<i>positionSrc</i>	<ul style="list-style-type: none"> • Optional parameter • Source from which this position was obtained • Valid values <ul style="list-style-type: none"> – 0 - Position source is GNSS – 1 - Position source is Cell ID – 2 - Position source is Enhanced Cell ID – 3 - Position source is Wi-Fi – 4 - Position source is Terrestrial – 5 - Position source is GNSS Terrestrial Hybrid – 6 - Other sources • Note - If altitude is specified and the altitude source is not specified, the engine assumes that the altitude was obtained using the specified position source. <ul style="list-style-type: none"> – If both altitude and altitude source are specified, the engine assumes that only latitude and longitude were obtained using the specified position source.
<i>rawHorUnc</i> ↔ <i>Circular</i>	<ul style="list-style-type: none"> • Optional parameter • Horizontal position uncertainty (circular) without any optimization. • Units - Meters
<i>rawHor</i> ↔ <i>Confidence</i>	<ul style="list-style-type: none"> • Optional parameter • Horizontal confidence associated with raw horizontal uncertainty • Units: Percent • Values <ul style="list-style-type: none"> – Valid values - 1 to 99 – Invalid values - 0, 101 to 255 – If 100 is received, reinterpret to 99 • Note - This field must be specified together with raw horizontal uncertainty. If not specified when rawHorUncCircular is set, the default value is 50.

8.519.2 Field Documentation

8.519.2.1 altSrcInfo_t pack_loc_SLQSLOCInjectPosition_t::altitudeSrcInfo

- 8.519.2.2 float pack_loc_SLQSLOCInjectPosition_t::altitudeWrtEllipsoid
- 8.519.2.3 float pack_loc_SLQSLOCInjectPosition_t::altitudeWrtMeanSeaLevel
- 8.519.2.4 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeSrcInfo
- 8.519.2.5 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeWrtEllipsoid
- 8.519.2.6 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeWrtMeanSeaLevel
- 8.519.2.7 int pack_loc_SLQSLOCInjectPosition_t::has_horConfidence
- 8.519.2.8 int pack_loc_SLQSLOCInjectPosition_t::has_horReliability
- 8.519.2.9 int pack_loc_SLQSLOCInjectPosition_t::has_horUncCircular
- 8.519.2.10 int pack_loc_SLQSLOCInjectPosition_t::has_latitude
- 8.519.2.11 int pack_loc_SLQSLOCInjectPosition_t::has_longitude
- 8.519.2.12 int pack_loc_SLQSLOCInjectPosition_t::has_positionSrc
- 8.519.2.13 int pack_loc_SLQSLOCInjectPosition_t::has_rawHorConfidence
- 8.519.2.14 int pack_loc_SLQSLOCInjectPosition_t::has_rawHorUncCircular
- 8.519.2.15 int pack_loc_SLQSLOCInjectPosition_t::has_timestampAge
- 8.519.2.16 int pack_loc_SLQSLOCInjectPosition_t::has_timestampUtc
- 8.519.2.17 int pack_loc_SLQSLOCInjectPosition_t::has_vertConfidence
- 8.519.2.18 int pack_loc_SLQSLOCInjectPosition_t::has_vertReliability
- 8.519.2.19 int pack_loc_SLQSLOCInjectPosition_t::has_vertUnc
- 8.519.2.20 uint8_t pack_loc_SLQSLOCInjectPosition_t::horConfidence
- 8.519.2.21 uint32_t pack_loc_SLQSLOCInjectPosition_t::horReliability
- 8.519.2.22 float pack_loc_SLQSLOCInjectPosition_t::horUncCircular
- 8.519.2.23 double pack_loc_SLQSLOCInjectPosition_t::latitude
- 8.519.2.24 double pack_loc_SLQSLOCInjectPosition_t::longitude

- 8.519.2.25 uint32_t pack_loc_SLQSLOCInjectPosition_t::positionSrc
- 8.519.2.26 uint8_t pack_loc_SLQSLOCInjectPosition_t::rawHorConfidence
- 8.519.2.27 float pack_loc_SLQSLOCInjectPosition_t::rawHorUncCircular
- 8.519.2.28 uint32_t pack_loc_SLQSLOCInjectPosition_t::timestampAge
- 8.519.2.29 uint64_t pack_loc_SLQSLOCInjectPosition_t::timestampUtc
- 8.519.2.30 uint8_t pack_loc_SLQSLOCInjectPosition_t::vertConfidence
- 8.519.2.31 uint32_t pack_loc_SLQSLOCInjectPosition_t::vertReliability
- 8.519.2.32 float pack_loc_SLQSLOCInjectPosition_t::vertUnc

8.520 pack_loc_SLQSLOCInjectSensorData_t Struct Reference

Data Fields

- int [has_opaqueId](#)
- uint32_t [opaqueId](#)
- int [has_acceleroData](#)
- [sensorData_t](#) [acceleroData](#)
- int [has_gyroData](#)
- [sensorData_t](#) [gyroData](#)
- int [has_acceleroTimeSrc](#)
- uint32_t [acceleroTimeSrc](#)
- int [has_gyroTimeSrc](#)
- uint32_t [gyroTimeSrc](#)
- int [has_accelTemp](#)
- [tempData_t](#) [accelTemp](#)
- int [has_gyroTemp](#)
- [tempData_t](#) [gyroTemp](#)

8.520.1 Detailed Description

This structure contains parameters to inject sensor data into the GNSS location engine Please check [has_<↔ Param_Name>](#) field for presence of optional parameters

Parameters

<i>opaqueId</i>	<ul style="list-style-type: none"> • Opaque Identifier (Optional parameter) • An opaque identifier that is sent in by the client that will be echoed in the indication so the client can relate the indication to the request.
<i>acceleroData</i>	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct sensorData. See sensorData for more information
<i>gyroData</i>	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct sensorData. See sensorData for more information
<i>acceleroTimeSrc</i>	<ul style="list-style-type: none"> • 3-Axis Accelerometer Data Time Source (Optional parameter) • The location service uses this field to identify the time reference used in the accelerometer data time stamps. • If not specified, the location service assumes that the time source for the accelerometer data is unknown. • Valid values <ul style="list-style-type: none"> – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine
<i>gyroTimeSrc</i>	<ul style="list-style-type: none"> • 3-Axis Gyroscope Data Time Source (Optional) • The location service uses this field to identify the time reference used in the gyroscope data time stamps. • If not specified, the location service assumes that the time source for the gyroscope data is unknown. • Valid values <ul style="list-style-type: none"> – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine
<i>accelTemp</i>	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct temperatureData. See temperatureData for more information
<i>gyroTemp</i>	<ul style="list-style-type: none"> • Optional parameter • Pointer to struct temperatureData. See temperatureData for more information

8.520.2 Field Documentation

8.520.2.1 `sensorData_t pack_loc_SLQSLOCInjectSensorData_t::acceleroData`8.520.2.2 `uint32_t pack_loc_SLQSLOCInjectSensorData_t::acceleroTimeSrc`8.520.2.3 `tempData_t pack_loc_SLQSLOCInjectSensorData_t::accelTemp`

8.520.2.4 `sensorData_t` `pack_loc_SLQSLOCInjectSensorData_t::gyroData`

8.520.2.5 `tempData_t` `pack_loc_SLQSLOCInjectSensorData_t::gyroTemp`

8.520.2.6 `uint32_t` `pack_loc_SLQSLOCInjectSensorData_t::gyroTimeSrc`

8.520.2.7 `int` `pack_loc_SLQSLOCInjectSensorData_t::has_acceleroTimeSrc`

8.520.2.8 `int` `pack_loc_SLQSLOCInjectSensorData_t::has_accelTemp`

8.520.2.9 `int` `pack_loc_SLQSLOCInjectSensorData_t::has_accleroData`

8.520.2.10 `int` `pack_loc_SLQSLOCInjectSensorData_t::has_gyroData`

8.520.2.11 `int` `pack_loc_SLQSLOCInjectSensorData_t::has_gyroTemp`

8.520.2.12 `int` `pack_loc_SLQSLOCInjectSensorData_t::has_gyroTimeSrc`

8.520.2.13 `int` `pack_loc_SLQSLOCInjectSensorData_t::has_opaqueld`

8.520.2.14 `uint32_t` `pack_loc_SLQSLOCInjectSensorData_t::opaqueld`

8.521 `pack_loc_SLQSLOCInjectUTCTime_t` Struct Reference

Data Fields

- `uint64_t` [timeMsec](#)
- `uint32_t` [timeUncMsec](#)

8.521.1 Detailed Description

Parameters

<i>timeMsec</i> [IN]	<ul style="list-style-type: none"> • The UTC time since Jan. 1, 1970
<i>timeUncMsec</i> [↔ IN]	<ul style="list-style-type: none"> • The time Uncertainty

8.521.2 Field Documentation

8.521.2.1 `uint64_t` `pack_loc_SLQSLOCInjectUTCTime_t::timeMsec`

8.521.2.2 `uint32_t` `pack_loc_SLQSLOCInjectUTCTime_t::timeUncMsec`

8.522 pack_loc_SLQSLOCSetCradleMountConfig_t Struct Reference

Data Fields

- uint32_t [state](#)
- int [has_confidence](#)
- uint8_t [confidence](#)

8.522.1 Detailed Description

This structure contains parameters to set current cradle mount configuration Please check has_<Param_Name> field for presence of optional parameters

Parameters

<i>state</i>	<ul style="list-style-type: none"> • Cradle Mount State • Valid values: <ul style="list-style-type: none"> – 0 - Device is mounted on the cradle – 1 - Device is not mounted on the cradle – 2 - Unknown cradle mount state
<i>confidence</i>	<ul style="list-style-type: none"> • Cradle Mount Confidence (Optional) • Confidence in the Cradle Mount state expressed as a percentage. • Range - 0 to 100

8.522.2 Field Documentation

8.522.2.1 uint8_t pack_loc_SLQSLOCSetCradleMountConfig_t::confidence

8.522.2.2 int pack_loc_SLQSLOCSetCradleMountConfig_t::has_confidence

8.522.2.3 uint32_t pack_loc_SLQSLOCSetCradleMountConfig_t::state

8.523 pack_loc_Start_t Struct Reference

Data Fields

- uint8_t [SessionId](#)
- uint32_t * [pRecurrenceType](#)
- uint32_t * [pHorizontalAccuracyLvl](#)
- uint32_t * [pIntermediateReportState](#)
- uint32_t * [pMinIntervalTime](#)
- [loc_LocApplicationInfo](#) * [pApplicationInfo](#)
- uint32_t * [pConfigAltitudeAssumed](#)
- uint16_t [Tlvresult](#)

8.523.1 Detailed Description

This structure contains the LOC Start pack

Parameters

<i>SessionId</i>	<ul style="list-style-type: none"> • ID of the session as identified by the control point. • Range: 0 to 255
<i>pRecurrence↔ Type</i>	<ul style="list-style-type: none"> • Specifies the type of session in which the control point is interested. • Defaults to SINGLE. -Values <ul style="list-style-type: none"> – 1 - Request periodic position fixes – 2 - Request a single position fix
<i>pHorizontal↔ AccuracyLvl</i>	<ul style="list-style-type: none"> • Specifies the horizontal accuracy level required by the control point. • Defaults to LOW • Values <ul style="list-style-type: none"> – 1 - Low accuracy – 2 - Medium accuracy – 3 - High accuracy
<i>pIntermediate↔ ReportState</i>	<ul style="list-style-type: none"> • Specifies if the control point is interested in receiving intermediate reports. • ON by default. • Values <ul style="list-style-type: none"> – 1 - Intermediate reports are turned on – 2 - Intermediate reports are turned off
<i>appVersionValid</i>	<ul style="list-style-type: none"> • Specifies whether the application version string contains a valid value • 0x00 (FALSE) – Application version string is invalid • 0x01 (TRUE) – Application version string is valid
<i>LocApplication↔ Info</i>	<ul style="list-style-type: none"> • LOC Application Parameters • See loc_LocApplicationInfo for more information
<i>pConfig↔ AltitudeAssumed</i>	<ul style="list-style-type: none"> • Configuration for Altitude Assumed Info in GNSS SV Info Event • Defaults to ENABLED. • Values <ul style="list-style-type: none"> – 1 - Enable Altitude Assumed information in GNSS SV Info Event – 2 - Disable Altitude Assumed information in GNSS SV Info Event

8.523.2 Field Documentation

8.523.2.1 loc_LocApplicationInfo* pack_loc_Start_t::pApplicationInfo

8.523.2.2 uint32_t* pack_loc_Start_t::pConfigAltitudeAssumed

8.523.2.3 uint32_t* pack_loc_Start_t::pHorizontalAccuracyLvl

8.523.2.4 uint32_t* pack_loc_Start_t::pIntermediateReportState

8.523.2.5 uint32_t* pack_loc_Start_t::pMinIntervalTime

8.523.2.6 uint32_t* pack_loc_Start_t::pRecurrenceType

8.523.2.7 uint8_t pack_loc_Start_t::SessionId

8.523.2.8 uint16_t pack_loc_Start_t::Tlvresult

8.524 pack_loc_Stop_t Struct Reference

Data Fields

- uint8_t [SessionId](#)
- uint16_t [Tlvresult](#)

8.524.1 Detailed Description

This structure contains Stop LOC pack

Parameters

<i>sessionId</i>	<ul style="list-style-type: none"> • ID of the session as identified by the control point. • Range: 0 to 255
<i>Tlvresult</i>	<ul style="list-style-type: none"> • Unpack result.

8.524.2 Field Documentation

8.524.2.1 uint8_t pack_loc_Stop_t::SessionId

8.524.2.2 uint16_t pack_loc_Stop_t::Tlvresult

8.525 pack_nas_SetACCOLC_t Struct Reference

Data Fields

- int8_t [spc](#) [6]
- uint8_t [accolc](#)

8.525.1 Detailed Description

Parameters

<i>spc</i>	servcie programming code
<i>accolc</i>	accolc

8.525.2 Field Documentation

8.525.2.1 uint8_t pack_nas_SetACCOLC_t::accolc

8.525.2.2 int8_t pack_nas_SetACCOLC_t::spc[6]

8.526 pack_nas_SetNetworkPreference_t Struct Reference

Data Fields

- uint32_t [TechnologyPref](#)
- uint32_t [Duration](#)
- uint16_t [Tlvresult](#)

8.526.1 Detailed Description

Parameters

<i>TechnologyPref</i> [IN]	<ul style="list-style-type: none"> • Bitmask representing the radio technology preference set. • No bits set indicates to the device to automatically determine the technology to use • Values: <ul style="list-style-type: none"> – Bit 0 - Technology is 3GPP2 – Bit 1 - Technology is 3GPP • Any combination of the following may be returned: <ul style="list-style-type: none"> – Bit 2 - Analog - AMPS if 3GPP2, GSM if 3GPP – Bit 3 - Digital - CDMA if 3GPP2, WCDMA if 3GPP – Bit 4 - HDR – Bit 5 - LTE – Bits 6 to 15 - Reserved
<i>Duration</i> [IN]	<ul style="list-style-type: none"> • Duration of active preference <ul style="list-style-type: none"> – 0 - Permanent – 1 - Power cycle – 2 - Until the end of the next call or a power cycle – 3 - Until the end of the next call, a specified time, or a power cycle – 4 to 6 - Until the end of the next call
<i>Tlvresult</i>	<ul style="list-style-type: none"> • pack result

8.526.2 Field Documentation

8.526.2.1 uint32_t pack_nas_SetNetworkPreference_t::Duration

8.526.2.2 uint32_t pack_nas_SetNetworkPreference_t::TechnologyPref

8.526.2.3 uint16_t pack_nas_SetNetworkPreference_t::Tlvresult

8.527 pack_nas_SLQSGetPLMNName_t Struct Reference

Data Fields

- uint16_t [mcc](#)
- uint16_t [mnc](#)
- uint8_t * [pMncPcsStatus](#)

8.527.1 Detailed Description

Parameters

<i>mcc</i>	<ul style="list-style-type: none"> • A 16-bit integer representation of MCC. Range: 0 to 999
<i>mnc</i>	<ul style="list-style-type: none"> • A 16-bit integer representation of MNC. Range: 0 to 999
<i>pMncPcsStatus</i>	<ul style="list-style-type: none"> • MNC PCS Digit Include Status • Used to interpret the length of the corresponding MNC reported in the PLMN TLV(0x01). • Values <ul style="list-style-type: none"> – TRUE - MNC is a three-digit value. e.g. a reported value of 90 corresponds to an MNC value of 090 – FALSE - MNC is a two-digit value. e.g. a reported value of 90 corresponds to an MNC value of 90

Note

If pMncPcsStatus is not present, an MNC smaller than 100 is assumed to be a two-digit value, and an MNC greater than or equal to 100 is assumed to be a three digit value.

8.527.2 Field Documentation

8.527.2.1 uint16_t pack_nas_SLQSGetPLMNName_t::mcc

8.527.2.2 uint16_t pack_nas_SLQSGetPLMNName_t::mnc

8.527.2.3 uint8_t* pack_nas_SLQSGetPLMNName_t::pMncPcsStatus

8.528 pack_nas_SLQSIInitiateNetworkRegistration_t Struct Reference

Data Fields

- uint32_t [regAction](#)
- nas_MNRInfo * [pMNRInfo](#)
- uint32_t * [pChangeDuration](#)
- uint8_t * [pMncPcsDigitStatus](#)

8.528.1 Detailed Description

This structure contains Initiate Network Registration request parameters

Parameters

<i>regAction</i>	<ul style="list-style-type: none"> • Specifies one of the following register actions : <ul style="list-style-type: none"> – AUTO_REGISTER - Device registers according to its provisioning and optional parameters supplied with the command are ignored. – MANUAL_REGISTER - Device registers to a specified network and the optional Manual Network Register Information parameter pMNRInfo must also be included for the command to process successfully and supported only for 3GPP.
<i>pMNRInfo</i>	[Optional] <ul style="list-style-type: none"> • Pointer to structure MNRInfo <ul style="list-style-type: none"> – See nas_MNRInfo for more information
<i>pChange↔ Duration</i>	[Optional] <ul style="list-style-type: none"> • Duration of the change. <ul style="list-style-type: none"> – 0x00 - Power cycle - Remains active until the next device power cycle – 0x01 - Permanent - Remains active through power cycles until changed by the client
<i>pMncPcsDigit↔ Status</i>	[Optional] <ul style="list-style-type: none"> • MNC PCS Digit Include Status <ul style="list-style-type: none"> – True - MNC is a 3-digit value. – False - MNC is a 2-digit value.

8.528.2 Field Documentation

8.528.2.1 uint32_t* pack_nas_SLQSIInitiateNetworkRegistration_t::pChangeDuration

8.528.2.2 uint8_t* pack_nas_SLQSIInitiateNetworkRegistration_t::pMncPcsDigitStatus

8.528.2.3 nas_MNRInfo* pack_nas_SLQSIInitiateNetworkRegistration_t::pMNRInfo

8.528.2.4 uint32_t pack_nas_SLQSIInitiateNetworkRegistration_t::regAction

8.529 pack_nas_SLQSNasConfigSigInfo2_t Struct Reference

Data Fields

- nas_CDMARSSIThresh * pCDMARSSIThresh
- uint16_t * pCDMARSSIDelta
- nas_CDMAECIOThresh * pCDMAECIOThresh
- uint16_t * pCDMAECIODelta
- nas_HDRRSSIThresh * pHDRRSSIThresh
- uint16_t * pHDRRSSIDelta
- nas_HDRECIOThresh * pHDRECIOThresh
- uint16_t * pHDRECIODelta
- nas_HDRSINRThreshold * pHDRSINRThresh
- uint16_t * pHDRSINRDelta
- nas_HDRIOThresh * pHDRIOThresh
- uint16_t * pHDRIODelta
- nas_GSMRSSIThresh * pGSMRSSIThresh
- uint16_t * pGSMRSSIDelta
- nas_WCDMARSSIThresh * pWCDMARSSIThresh
- uint16_t * pWCDMARSSIDelta
- nas_WCDMAECIOThresh * pWCDMAECIOThresh
- uint16_t * pWCDMAECIODelta
- nas_LTERSSIThresh * pLTERSSIThresh
- uint16_t * pLTERSSIDelta
- nas_LTESNRThreshold * pLTESNRThresh
- uint16_t * pLTESNRDelta
- nas_LTERSRQThresh * pLTERSRQThresh
- uint16_t * pLTERSRQDelta
- nas_LTERSRPThresh * pLTERSRPThresh
- uint16_t * pLTERSRPDelta
- nas_LTESigRptConfig * pLTESigRptConfig
- nas_TDSCDMARSCPTThresh * pTDSCDMARSCPTThresh
- uint16_t * pTDSCDMARSCPDelta
- nas_TDSCDMARSSIThresh * pTDSCDMARSSIThresh
- float * pTDSCDMARSSIDelta
- nas_TDSCDMAECIOThresh * pTDSCDMAECIOThresh
- float * pTDSCDMAECIODelta
- nas_TDSCDMASINRThresh * pTDSCDMASINRThresh
- float * pTDSCDMASINRDelta

8.529.1 Detailed Description

Parameters

<i>pCDMARSSI</i> <i>Thresh</i>	<ul style="list-style-type: none"> CDMA RSSI threshold List
<i>pCDMARSSI</i> <i>Delta</i>	<ul style="list-style-type: none"> RSSI delta (in units of 0.1 dBm). A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pCDMAECIO</i> <i>Thresh</i>	<ul style="list-style-type: none"> CDMA ECIO Threshold List
<i>pCDMAECIO</i> <i>Delta</i>	<ul style="list-style-type: none"> ECIO delta (in units of 0.1 dB). A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pHDRRSSI</i> <i>Thresh</i>	<ul style="list-style-type: none"> HDR RSSI Threshold List
<i>pHDRRSSIDelta</i>	<ul style="list-style-type: none"> RSSI delta (in units of 0.1 dBm) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pHDRECI</i> <i>Thresh</i>	<ul style="list-style-type: none"> HDR ECIO Threshold List
<i>pHDRECI</i> <i>Delta</i>	<ul style="list-style-type: none"> ECIO delta (in units of 0.1 dB) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pHDRSINR</i> <i>Thresh</i>	<ul style="list-style-type: none"> HDR SINR Threshold List
<i>pHDRSINR</i> <i>Delta</i>	<ul style="list-style-type: none"> SINR delta (in units of 1 SINR level) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pHDRIOTresh</i>	<ul style="list-style-type: none"> HDR IO Threshold List
<i>pHDRIODelta</i>	<ul style="list-style-type: none"> IO delta (in units of 0.1 dBm) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pGSMRSSI</i> <i>Thresh</i>	<ul style="list-style-type: none"> GSM RSSI Threshold List See GSMRSSIThresh for more details
<i>pGSMRSSIDelta</i>	<ul style="list-style-type: none"> RSSI delta (in units of 0.1 dBm) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.

<i>pWCDMARSS</i> ↔ <i>IThresh</i>	<ul style="list-style-type: none"> WCDMA RSSI Threshold List See WCDMARSSIThresh for more details
<i>pWCDMARSS</i> ↔ <i>IDelta</i>	<ul style="list-style-type: none"> RSSI delta (in units of 0.1 dBm). A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pWCDMAECI</i> ↔ <i>OThresh</i>	<ul style="list-style-type: none"> WCDMA ECIO Threshold List
<i>pWCDMAECI</i> ↔ <i>ODelta</i>	<ul style="list-style-type: none"> ECIO delta (in units of 0.1 dB) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pLTERSSI</i> ↔ <i>Thresh</i>	<ul style="list-style-type: none"> LTE RSSI Threshold List
<i>pLTERSSIDelta</i>	<ul style="list-style-type: none"> RSSI delta (in units of 0.1 dBm) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pLTERSNR</i> ↔ <i>Thresh</i>	<ul style="list-style-type: none"> LTE SNR Threshold List
<i>pLTERSNRDelta</i>	<ul style="list-style-type: none"> SNR delta (in units of 0.1 dBm) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pLTERSRQ</i> ↔ <i>Thresh</i>	<ul style="list-style-type: none"> LTE RSRQ Threshold List
<i>pLTERSRQ</i> ↔ <i>Delta</i>	<ul style="list-style-type: none"> RSRQ delta (in units of 0.1 dBm) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pLTERSRP</i> ↔ <i>Thresh</i>	<ul style="list-style-type: none"> LTE RSRP Threshold List
<i>pLTERSRPDelta</i>	<ul style="list-style-type: none"> RSRP delta (in units of 0.1 dBm). A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pLTERSigRpt</i> ↔ <i>Config</i>	<ul style="list-style-type: none"> LTE Signal Report Config
<i>pTDSCDMAR</i> ↔ <i>SCPThresh</i>	<ul style="list-style-type: none"> TDSCDMA RSCP Threshold List
<i>pTDSCDMAR</i> ↔ <i>SCPDelta</i>	<ul style="list-style-type: none"> RSCP delta (in units of 0.1 dBm) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.

<i>pTDSCDMARSSIThresh</i>	<ul style="list-style-type: none"> TDSCDMA RSSI Threshold List
<i>pTDSCDMARSSIDelta</i>	<ul style="list-style-type: none"> RSSI delta (in dBm) used by TD-SCDMA.
<i>pTDSCDMAECIOThresh</i>	<ul style="list-style-type: none"> TDSCDMA ECIO Threshold List
<i>pTDSCDMAECIODelta</i>	<ul style="list-style-type: none"> ECIO delta (in dB) used by TD-SCDMA
<i>pTDSCDMASINRThresh</i>	<ul style="list-style-type: none"> TDSCDMA SINR Threshold List
<i>pTDSCDMASINRDelta</i>	<ul style="list-style-type: none"> SINR delta (in dB) used by TD-SCDMA.

8.529.2 Field Documentation

8.529.2.1 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pCDMAECIODelta`

8.529.2.2 `nas_CDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pCDMAECIOThresh`

8.529.2.3 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pCDMARSSIDelta`

8.529.2.4 `nas_CDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pCDMARSSIThresh`

8.529.2.5 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pGSMRSSIDelta`

8.529.2.6 `nas_GSMRSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pGSMRSSIThresh`

8.529.2.7 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRECIODelta`

8.529.2.8 `nas_HDRECIOTThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHDRECIOTThresh`

8.529.2.9 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRIODelta`

8.529.2.10 `nas_HDRIOTThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHDRIOTThresh`

8.529.2.11 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHRRSSIDelta`

8.529.2.12 `nas_HDRRSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHRRSSIThresh`

8.529.2.13 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHRSINRDelta`

- 8.529.2.14 `nas_HDRSINRThreshold* pack_nas_SLQSNasConfigSigInfo2_t::pHDRSINRThresh`
- 8.529.2.15 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRPDelta`
- 8.529.2.16 `nas_LTERSRPThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRPThresh`
- 8.529.2.17 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRQDelta`
- 8.529.2.18 `nas_LTERSRQThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRQThresh`
- 8.529.2.19 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSSIDelta`
- 8.529.2.20 `nas_LTERSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSSIThresh`
- 8.529.2.21 `nas_LTESigRptConfig* pack_nas_SLQSNasConfigSigInfo2_t::pLTESigRptConfig`
- 8.529.2.22 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTESNRDelta`
- 8.529.2.23 `nas_LTESNRThreshold* pack_nas_SLQSNasConfigSigInfo2_t::pLTESNRThresh`
- 8.529.2.24 `float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMAECIODelta`
- 8.529.2.25 `nas_TDSCDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMAECIOThresh`
- 8.529.2.26 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSCPDelta`
- 8.529.2.27 `nas_TDSCDMARSCPThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSCPThresh`
- 8.529.2.28 `float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSSIDelta`
- 8.529.2.29 `nas_TDSCDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSSIThresh`
- 8.529.2.30 `float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMASINRDelta`
- 8.529.2.31 `nas_TDSCDMASINRThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMASINRThresh`
- 8.529.2.32 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMAECIODelta`
- 8.529.2.33 `nas_WCDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMAECIOThresh`
- 8.529.2.34 `uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMARSSIDelta`
- 8.529.2.35 `nas_WCDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMARSSIThresh`

8.530 `pack_nas_SLQSNasIndicationRegisterExt_t` Struct Reference

Data Fields

- `uint8_t * pSystemSelectionInd`

- uint8_t * [pDDTMInd](#)
- uint8_t * [pServingSystemInd](#)
- uint8_t * [pDualStandByPrefInd](#)
- uint8_t * [pSubscriptionInfoInd](#)
- uint8_t * [pNetworkTimeInd](#)
- uint8_t * [pSysInfoInd](#)
- uint8_t * [pSignalStrengthInd](#)
- uint8_t * [pErrorRateInd](#)
- uint8_t * [pHDRNewUATIAssInd](#)
- uint8_t * [pHDRSessionCloseInd](#)
- uint8_t * [pManagedRoamingInd](#)
- uint8_t * [pLTECphyCa](#)

8.530.1 Detailed Description

Parameters

<i>pSystemSelectionInd</i>	[Optional] <ul style="list-style-type: none"> • System Selection Preference indication registration. The following callbacks would not be invoked if the indication is disabled. tFNRoamingIndicator tFNDataCapabilities and tFNServingSystem <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pDDTMInd</i>	[Optional] <ul style="list-style-type: none"> • DDTM (Data Dedicated Transmission Mode) indication registration. The following callbacks would not be invoked if the indication is disabled. tFNDDTM <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pServingSystemInd</i>	[Optional] <ul style="list-style-type: none"> • Serving System indication registration. The following callbacks would not be invoked if the indication is disabled. tFNBandPreference <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pDualStandByPrefInd</i>	[Optional] <ul style="list-style-type: none"> • Dual Standby Preference indication registration. The following callbacks would not be invoked if the indication is disabled. tFNDualStandByPref <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pSubscriptionInfoInd</i>	[Optional] <ul style="list-style-type: none"> • Subscription Information indication registration. The following callbacks would not be invoked if the indication is disabled. tFNSubscriptionInfo <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable

<i>pNetworkTime</i> ↔ <i>Ind</i>	[Optional] <ul style="list-style-type: none"> Network Time indication registration. The following callbacks would not be invoked if the indication is disabled. tFNNetworkTime <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pSysInfoInd</i>	[Optional] <ul style="list-style-type: none"> System Information indication registration. The following callbacks would not be invoked if the indication is disabled. tFNSysInfo <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pSignal</i> ↔ <i>StrengthInd</i>	[Optional] <ul style="list-style-type: none"> Signal Strength indication registration. The following callbacks would not be invoked if the indication is disabled. tFNSigInfo <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pErrorRateInd</i>	[Optional] <ul style="list-style-type: none"> Error Rate indication registration. The following callbacks would not be invoked if the indication is disabled. tFNErrRate <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pHDRNewUA</i> ↔ <i>TIAssInd</i>	[Optional] <ul style="list-style-type: none"> HDR New UATI Assigned indication registration. The following callbacks would not be invoked if the indication is disabled. tFNHDRUATIUpdate <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pHDRSession</i> ↔ <i>CloseInd</i>	[Optional] <ul style="list-style-type: none"> HDR Session Closed indication registration. The following callbacks would not be invoked if the indication is disabled. tFNHDRSessionClose <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable
<i>pManaged</i> ↔ <i>RoamingInd</i>	[Optional] <ul style="list-style-type: none"> Managed Roaming indication registration. The following callbacks would not be invoked if the indication is disabled. tFNManagedRoaming <ul style="list-style-type: none"> 0x00 - Disable 0x01 - Enable

<i>pLTECphyCa</i>	[Optional] <ul style="list-style-type: none"> • LTE Physical Carrier Aggregation Information. The following callbacks would not be invoked if the indication is disabled. tFNManagedRoaming <ul style="list-style-type: none"> – 0x00 - Disable (default value) – 0x01 - Enable
-------------------	--

8.530.2 Field Documentation

8.530.2.1 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pDDTMInd

8.530.2.2 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pDualStandByPrefInd

8.530.2.3 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pErrorRateInd

8.530.2.4 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pHDRNewUATIAssInd

8.530.2.5 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pHDRSessionCloseInd

8.530.2.6 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pLTECphyCa

8.530.2.7 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pManagedRoamingInd

8.530.2.8 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pNetworkTimeInd

8.530.2.9 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pServingSystemInd

8.530.2.10 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSignalStrengthInd

8.530.2.11 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSubscriptionInfoInd

8.530.2.12 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSysInfoInd

8.530.2.13 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSystemSelectionInd

8.531 pack_nas_SLQSNasSwlIndicationRegister_t Struct Reference

Data Fields

- uint8_t lteEsmUI
- uint8_t lteEsmDI
- uint8_t lteEmmUI
- uint8_t lteEmmDI
- uint8_t gsmUmtsUI
- uint8_t gsmUmtsDI
- uint8_t * pRankIndicatorInd
- uint8_t * pTimer

8.531.1 Detailed Description

This structure contains the OTA message indication.

Parameters

<i>lteEsmUI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report LTE ESM uplink messages
<i>lteEsmDI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report LTE ESM downlink messages
<i>lteEmmUI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report LTE EMM uplink messages
<i>lteEmmDI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report GSM/UMTS uplink messages
<i>gsmUmtsUI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report GSM/UMTS uplink messages
<i>gsmUmtsDI</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report GSM/UMTS downlink messages
<i>pRank↔ IndicatorInd</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report Rank Indicator messages
<i>pTimer</i>	<ul style="list-style-type: none"> • 0 - do not report • 1 - report Timer Indicator messages

8.531.2 Field Documentation

8.531.2.1 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::gsmUmtsDI

8.531.2.2 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::gsmUmtsUI

8.531.2.3 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEmmDI

8.531.2.4 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEmmUI

8.531.2.5 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEsmDI

8.531.2.6 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEsmUI

8.531.2.7 uint8_t* pack_nas_SLQSNasSwiIndicationRegister_t::pRankIndicatorInd

8.531.2.8 uint8_t* pack_nas_SLQSNasSwiIndicationRegister_t::pTimer

8.532 pack_nas_SLQSSetSignalStrengthsCallback_t Struct Reference

Data Fields

- uint8_t [bEnable](#)
- [nas_SLQSSignalStrengthsIndReq](#) * [pSigIndReq](#)

8.532.1 Detailed Description

Parameters

<i>bEnable</i>	0/1 to disable/enable RSSI signal strength indication
<i>pSigIndReq</i>	parameters to control signal strength indication

8.532.2 Field Documentation

8.532.2.1 uint8_t pack_nas_SLQSSetSignalStrengthsCallback_t::bEnable

8.532.2.2 [nas_SLQSSignalStrengthsIndReq](#)* pack_nas_SLQSSetSignalStrengthsCallback_t::pSigIndReq

8.533 pack_nas_SLQSSetSysSelectionPref_t Struct Reference

Data Fields

- uint8_t * [pEmerMode](#)
- uint16_t * [pModePref](#)
- uint64_t * [pBandPref](#)
- uint16_t * [pPRLPref](#)
- uint16_t * [pRoamPref](#)
- uint64_t * [pLTEBandPref](#)
- struct [nas_netSelectionPref](#) * [pNetSelPref](#)
- uint8_t * [pChgDuration](#)
- uint8_t * [pMNCIncPCSDigStat](#)
- uint32_t * [pSrvDomainPref](#)
- uint32_t * [pGWAcqOrderPref](#)
- uint64_t * [pTdsdmaBandPref](#)
- struct [nas_acqOrderPref](#) * [pAcqOrderPref](#)
- uint32_t * [pSrvRegRestriction](#)
- struct [nas_CSGID](#) * [pCSGID](#)
- unsigned char * [pRAT](#)

8.533.1 Detailed Description

Contain the system selection preferences.

Parameters

<i>pEmerMode</i>	<ul style="list-style-type: none">• Optional parameter specifying the emergency Mode• Values:<ul style="list-style-type: none">– 0 - OFF (normal)– 1 - ON (Emergency)
<i>pModePref</i>	<ul style="list-style-type: none">• Optional parameter• Bit Mask indicating the radio technology mode preference• Bit values:<ul style="list-style-type: none">– Bit 0 - cdma2000 1x– Bit 1 - cdma2000 HRPD(1xEV-DO)– Bit 2 - GSM– Bit 3 - UMTS– Bit 4 - LTE

<i>pBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the band preference • Bit values: <ul style="list-style-type: none"> – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved
------------------	--

<i>pPRLPref</i>	<ul style="list-style-type: none"> • Optional parameter indicating the CDMA PRL Preference • Values: <ul style="list-style-type: none"> – 0x0001 - Acquire available system only on the A side – 0x0002 - Acquire available system only on the B side – 0x3FFF - Acquire any available systems
<i>pRoamPref</i>	<ul style="list-style-type: none"> • Optional parameter indicating the roaming Preference • Values: <ul style="list-style-type: none"> – 0x01 - Acquire only systems for which the roaming indicator is off – 0x02 - Acquire a system as long as its roaming indicator is not off – 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only – 0xFF - Acquire systems, regardless of their roaming indicator
<i>pLTEBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the LTE band preference • Bit Values <ul style="list-style-type: none"> – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – All other bits are reserved

<i>pNetSelPref</i>	<ul style="list-style-type: none"> - netSelectionPref • Optional parameter for specifying Network Selection Preference • Modem selects networks based on this parameter(if present). • see netSelectionPref for more information
<i>pChgDuration</i>	<ul style="list-style-type: none"> • Optional parameter specifying the duration of the change • Values: <ul style="list-style-type: none"> – 0x00 - Power cycle - Remains active until the next device power cycle – 0x01 - Permanent - Remains active through power cycles until changed by client – Device will use "0x01 - permanent" as default if this parameter is omitted
<i>pMNCIncPCS↔ DigStat</i>	<ul style="list-style-type: none"> • Optional parameter indicating if MNC includes PCS digit • Values: <ul style="list-style-type: none"> – TRUE - MNC is a 3 digit value; e.g., a reported value of 90 corresponds to an MNC value of 090 – FALSE - MNC is a 2-digit value; e.g., a reported value of 90 corresponds to an MNC value of 90
<i>pSrvDomainPref</i>	<ul style="list-style-type: none"> • Optional parameter indicating Service domain preference • Values: <ul style="list-style-type: none"> – 0x00 - Circuit switched only – 0x01 - Packet switched only – 0x02 - Circuit switched and packet switched – 0x03 - Packet switched attach – 0x04 - Packet switched detach
<i>pGWAcqOrder↔ Pref</i>	<ul style="list-style-type: none"> • Optional parameter indicating GSM/WCDMA Acquisition order Preference • Values: <ul style="list-style-type: none"> – 0x00 - Automatic – 0x01 - GSM then WCDMA – 0x02 - WCDMA then GSM

8.533.2 Field Documentation

8.533.2.1 struct nas_acqOrderPref* pack_nas_SLQSSetSysSelectionPref_t::pAcqOrderPref

8.533.2.2 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pBandPref

8.533.2.3 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pChgDuration

8.533.2.4 struct nas_CSGID* pack_nas_SLQSSetSysSelectionPref_t::pCSGID

8.533.2.5 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pEmerMode

- 8.533.2.6 `uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pGWAcqOrderPref`
- 8.533.2.7 `uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pLTEBandPref`
- 8.533.2.8 `uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pMNCIncPCSDigStat`
- 8.533.2.9 `uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pModePref`
- 8.533.2.10 `struct nas_netSelectionPref* pack_nas_SLQSSetSysSelectionPref_t::pNetSelPref`
- 8.533.2.11 `uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pPRLPref`
- 8.533.2.12 `unsigned char* pack_nas_SLQSSetSysSelectionPref_t::pRAT`
- 8.533.2.13 `uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pRoamPref`
- 8.533.2.14 `uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pSrvDomainPref`
- 8.533.2.15 `uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pSrvRegRestriction`
- 8.533.2.16 `uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pTdsdmaBandPref`

8.534 pack_qmi_t Struct Reference

Data Fields

- `uint16_t xid`
- `int timeout`
- `uint16_t msgid`
- `uint8_t svc`

8.534.1 Detailed Description

qmi request context

Parameters

in	<i>xid</i>	transaction id
out	<i>timeout</i>	timeout recommended in seconds
out	<i>msgid</i>	message id
out	<i>svc</i>	qmi service

8.534.2 Field Documentation

- 8.534.2.1 `uint16_t pack_qmi_t::msgid`

8.534.2.2 `uint8_t` `pack_qmi_t::svc`

8.534.2.3 `int` `pack_qmi_t::timeout`

8.534.2.4 `uint16_t` `pack_qmi_t::xid`

8.535 `pack_qos_SLQSQosSwiReadApnExtraParams_t` Struct Reference

Data Fields

- `uint32_t` [apnId](#)

8.535.1 Detailed Description

Structure that contains the APN ID to obtain extra APN parameters

Parameters

<i>apnId</i> [IN]	<ul style="list-style-type: none">• APN id
-------------------	--

8.535.2 Field Documentation

8.535.2.1 `uint32_t` `pack_qos_SLQSQosSwiReadApnExtraParams_t::apnId`

8.536 `pack_qos_SLQSQosSwiReadDataStats_t` Struct Reference

Data Fields

- `uint32_t` [apnId](#)

8.536.1 Detailed Description

Structure that contains the APN ID to obtain data statistics

Parameters

<i>apnId</i> [IN]	<ul style="list-style-type: none">• APN id
-------------------	--

8.536.2 Field Documentation

8.536.2.1 `uint32_t pack_qos_SLQSQosSwiReadDataStats_t::apnId`

8.537 `pack_qos_SLQSSetQosEventCallback_t` Struct Reference

Data Fields

- `uint8_t enable`

8.537.1 Detailed Description

Structure that contains the APN ID to obtain data statistics

Parameters

<i>enable</i> [IN]	<ul style="list-style-type: none"> • 1 - Enable QoS event reporting • 0 - Disable QoS event reporting
--------------------	---

8.537.2 Field Documentation

8.537.2.1 `uint8_t pack_qos_SLQSSetQosEventCallback_t::enable`

8.538 `pack_sms_SendSMS_t` Struct Reference

Data Fields

- `uint32_t messageFormat`
- `uint32_t messageSize`
- `uint8_t * pMessage`
- `uint8_t * pLinktimer`

8.538.1 Detailed Description

Parameters

<i>messageFormat</i>	<ul style="list-style-type: none"> • Message format <ul style="list-style-type: none"> – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
<i>messageSize</i>	<ul style="list-style-type: none"> • The length of the message contents in bytes
<i>pLinktimer</i>	<ul style="list-style-type: none"> • GW SMS link open for the specified number of second
<i>pMessage</i>	<div>Generated by Doxygen</div> <ul style="list-style-type: none"> • The message contents in PDU format contains SMS header and payload message

8.538.2 Field Documentation

8.538.2.1 uint32_t pack_sms_SendSMS_t::messageFormat

8.538.2.2 uint32_t pack_sms_SendSMS_t::messageSize

8.538.2.3 uint8_t* pack_sms_SendSMS_t::pLinktimer

8.538.2.4 uint8_t* pack_sms_SendSMS_t::pMessage

8.539 pack_sms_SetNewSMSCallback_t Struct Reference

Data Fields

- enum [eqmiCbkJSetStatus status](#)

8.539.1 Detailed Description

Parameters

<i>status</i>	callback parameter
---------------	--------------------

8.539.2 Field Documentation

8.539.2.1 enum [eqmiCbkJSetStatus](#) pack_sms_SetNewSMSCallback_t::status

8.540 pack_sms_SLQSDeleteSMS_t Struct Reference

Data Fields

- uint32_t [storageType](#)
- uint32_t * [pMessageIndex](#)
- uint32_t * [pMessageTag](#)
- uint8_t * [pMessageMode](#)

8.540.1 Detailed Description

Parameters

<i>storageType</i>	<ul style="list-style-type: none">SMS message storage type<ul style="list-style-type: none">0 - UIM - Invalid in case of CDMA device that does not require SIM1 - NV
--------------------	---

<i>pMessageIndex</i>	<ul style="list-style-type: none"> • (Optional) message index
<i>pMessageTag</i>	<ul style="list-style-type: none"> • (Optional) message tag <ul style="list-style-type: none"> – 0 - Read – 1 - Not read – 2 - Mobile originated and sent – 3 - Mobile originated but not yet sent
<i>pMessageMode</i>	<ul style="list-style-type: none"> • (Optional) message mode • this must be included if the device is capable of supporting more than one protocol • e.g. CDMA and GW <ul style="list-style-type: none"> – 0x00 - CDMA, LTE (if network type is CDMA) – 0x01 - GW, LTE (if network type is UMTS)

8.540.2 Field Documentation

8.540.2.1 uint32_t* pack_sms_SLQSDelateSMS_t::pMessageIndex

8.540.2.2 uint8_t* pack_sms_SLQSDelateSMS_t::pMessageMode

8.540.2.3 uint32_t* pack_sms_SLQSDelateSMS_t::pMessageTag

8.540.2.4 uint32_t pack_sms_SLQSDelateSMS_t::storageType

8.541 pack_sms_SLQSGetSMS_t Struct Reference

Data Fields

- uint32_t [storageType](#)
- uint32_t [messageIndex](#)
- uint8_t * [pMessageMode](#)

8.541.1 Detailed Description

Parameters

<i>storageType</i>	<ul style="list-style-type: none"> • SMS message storage type <ul style="list-style-type: none"> – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
<i>messageIndex</i>	<ul style="list-style-type: none"> • Message index
<i>pMessageMode</i>	<ul style="list-style-type: none"> • 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

8.541.2 Field Documentation

8.541.2.1 uint32_t pack_sms_SLQSGetSMS_t::messageIndex

8.541.2.2 uint8_t* pack_sms_SLQSGetSMS_t::pMessageMode

8.541.2.3 uint32_t pack_sms_SLQSGetSMS_t::storageType

8.542 pack_sms_SLQSGetSMSList_t Struct Reference

Data Fields

- uint32_t [storageType](#)
- uint32_t * [pRequestedTag](#)
- uint8_t * [pMessageMode](#)

8.542.1 Detailed Description

Parameters

<i>storageType</i>	<ul style="list-style-type: none"> • SMS message storage type <ul style="list-style-type: none"> – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
<i>requestedTag</i>	<ul style="list-style-type: none"> • (Optional) Message tag <ul style="list-style-type: none"> – 0 - Read – 1 - Not read – 2 - Mobile originated and sent – 3 - Mobile originated but not yet sent
<i>messageMode</i>	<ul style="list-style-type: none"> • 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

8.542.2 Field Documentation

8.542.2.1 uint8_t* pack_sms_SLQSGetSMSList_t::pMessageMode

8.542.2.2 uint32_t* pack_sms_SLQSGetSMSList_t::pRequestedTag

8.542.2.3 uint32_t pack_sms_SLQSGetSMSList_t::storageType

8.543 pack_sms_SLQSMModifySMSStatus_t Struct Reference

Data Fields

- uint32_t [storageType](#)
- uint32_t [messageIndex](#)
- uint32_t [messageTag](#)
- uint8_t * [pMessageMode](#)

8.543.1 Detailed Description

Parameters

<i>storageType</i>	<ul style="list-style-type: none"> • SMS message storage type <ul style="list-style-type: none"> – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
<i>messageIndex</i>	<ul style="list-style-type: none"> • Message index
<i>messageTag</i>	<ul style="list-style-type: none"> • Message tag <ul style="list-style-type: none"> – 0 - Read – 1 - Not read
<i>pMessageMode</i>	<ul style="list-style-type: none"> • 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

8.543.2 Field Documentation

8.543.2.1 uint32_t pack_sms_SLQSMModifySMSStatus_t::messageIndex

8.543.2.2 uint32_t pack_sms_SLQSMModifySMSStatus_t::messageTag

8.543.2.3 uint8_t* pack_sms_SLQSMModifySMSStatus_t::pMessageMode

8.543.2.4 uint32_t pack_sms_SLQSMModifySMSStatus_t::storageType

8.544 pack_swiloc_SwiLocSetAutoStart_t Struct Reference

Data Fields

- uint8_t [function](#)
- int [set_function](#)
- uint8_t [fix_type](#)
- int [set_fix_type](#)
- uint8_t [max_time](#)
- int [set_max_time](#)
- uint32_t [max_dist](#)
- int [set_max_dist](#)
- uint32_t [fix_rate](#)
- int [set_fix_rate](#)

8.544.1 Detailed Description

This structure contains SWI LOC Get Auto Start setting

Parameters

<i>function</i>	<ul style="list-style-type: none"> Setting to indicate when modem should start an automatic GNSS fix <ul style="list-style-type: none"> 0 - disabled 1 - At bootup 2 - When NMEA port is opened
<i>set_function</i>	<ul style="list-style-type: none"> 0 - do not set to modem 1 - set to modem
<i>fix_type</i>	<ul style="list-style-type: none"> Type of GNSS fix: <ul style="list-style-type: none"> 1 - Default Engine mode 2 - MS-Based 3 - MS-Assisted 4 - Standalone
<i>set_fix_type</i>	<ul style="list-style-type: none"> 0 - do not set to modem 1 - set to modem
<i>max_time</i>	<ul style="list-style-type: none"> Maximum time allowed for the receiver to get a fix in seconds Valid range: 1-255
<i>set_max_time</i>	<ul style="list-style-type: none"> 0 - do not set to modem 1 - set to modem
<i>max_dist</i>	<ul style="list-style-type: none"> Maximum uncertainty of a fix measured by distance in meters Valid range: 1 - 4294967280
<i>set_max_dist</i>	<ul style="list-style-type: none"> 0 - do not set to modem 1 - set to modem
<i>fix_rate</i>	<ul style="list-style-type: none"> Time between fixes in seconds Valid range: 1–65535
<i>set_fix_rate</i>	<ul style="list-style-type: none"> 0 - do not set to modem 1 - set to modem

8.544.2 Field Documentation

- 8.544.2.1 `uint32_t pack_swiloc_SwiLocSetAutoStart_t::fix_rate`
- 8.544.2.2 `uint8_t pack_swiloc_SwiLocSetAutoStart_t::fix_type`
- 8.544.2.3 `uint8_t pack_swiloc_SwiLocSetAutoStart_t::function`
- 8.544.2.4 `uint32_t pack_swiloc_SwiLocSetAutoStart_t::max_dist`
- 8.544.2.5 `uint8_t pack_swiloc_SwiLocSetAutoStart_t::max_time`
- 8.544.2.6 `int pack_swiloc_SwiLocSetAutoStart_t::set_fix_rate`
- 8.544.2.7 `int pack_swiloc_SwiLocSetAutoStart_t::set_fix_type`
- 8.544.2.8 `int pack_swiloc_SwiLocSetAutoStart_t::set_function`
- 8.544.2.9 `int pack_swiloc_SwiLocSetAutoStart_t::set_max_dist`
- 8.544.2.10 `int pack_swiloc_SwiLocSetAutoStart_t::set_max_time`

8.545 `pack_swioama_SLQSOMADMCancelSession_t` Struct Reference

Data Fields

- `uint32_t` [sessionType](#)

8.545.1 Detailed Description

Structure that contains the session type for OMA cancel session command

Parameters

<i>sessionType</i> [IN]	<ul style="list-style-type: none">• Session type<ul style="list-style-type: none">– 0x01 - FOTA, to check availability of FW Update– 0xFF - Cancel any active OMADM session
-------------------------	--

8.545.2 Field Documentation

- 8.545.2.1 `uint32_t pack_swioama_SLQSOMADMCancelSession_t::sessionType`

8.546 pack_swisma_SLQSOMADMGetSessionInfo_t Struct Reference

Data Fields

- uint32_t [SessionType](#)

8.546.1 Detailed Description

Structure that contains the session type for OMA get session info command

Parameters

<i>SessionType</i> [IN]	<ul style="list-style-type: none"> • Session type <ul style="list-style-type: none"> – 0x01 - FOTA – 0xFF - Any active OMADM session. If no active sessions are available, then previous O↔MADM session info is returned
-------------------------	--

8.546.2 Field Documentation

8.546.2.1 uint32_t pack_swisma_SLQSOMADMGetSessionInfo_t::SessionType

8.547 pack_swisma_SLQSOMADMSendSelection_t Struct Reference

Data Fields

- uint32_t [selection](#)
- uint32_t * [pDeferTime](#)
- uint32_t * [pRejectReason](#)

8.547.1 Detailed Description

Structure containing the OMA DM selection

Parameters

<i>selection</i> [IN]	<ul style="list-style-type: none"> • OMA-DM NIA Selection <ul style="list-style-type: none"> – 0x01 - Accept – 0x02 - Reject – 0x03 - Defer
<i>pDeferTime</i> [IN]	<ul style="list-style-type: none"> • Defer time in minutes. A value of 0 will cause the prompt to be resent immediately. • This TLV is mandatory if selection is set to 0x03.
<i>pRejectReason</i> [IN]	<ul style="list-style-type: none"> • Reject Reason • This TLV is processed if selection is set to 0x02. If it is not present, the reject reason 0 is used as default.

8.547.2 Field Documentation

8.547.2.1 uint32_t* pack_swima_SLQSOMADMSendSelection_t::pDeferTime

8.547.2.2 uint32_t* pack_swima_SLQSOMADMSendSelection_t::pRejectReason

8.547.2.3 uint32_t pack_swima_SLQSOMADMSendSelection_t::selection

8.548 pack_swima_SLQSOMADMSetSettings_t Struct Reference

Data Fields

- uint8_t [FOTAdownload](#)
- uint8_t [FOTAUpdate](#)
- uint8_t * [pAutosdm](#)
- uint8_t * [pFwAutoCheck](#)

8.548.1 Detailed Description

Structure containing the OMA DM settings to be set on the device This maps to structure SLQSOMADMSettings↔ ReqParams3

Parameters

<i>FOTAdownload</i>	<ul style="list-style-type: none"> • 1 Byte parameter indicating support for FOTA Automatic download <ul style="list-style-type: none"> – 0x00 - Firmware autodownload FALSE – 0x01 - Firmware autodownload TRUE
<i>FOTAUpdate</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating FOTA Automatic update <ul style="list-style-type: none"> – 0x00 - Firmware autoupdate FALSE – 0x01 - Firmware autoupdate TRUE
<i>pAutosdm[IN]</i>	<ul style="list-style-type: none"> • Optional 1 byte parameter indicating OMA Automatic UI Alert Response <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled Accept – 0x02 - Enabled Reject
<i>pFwAuto↔ Check[IN]</i>	<ul style="list-style-type: none"> • Optional 1 byte parameter indicating OMA Automatic Check for Firmware Update on Power-Up Response <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled

8.548.2 Field Documentation

8.548.2.1 uint8_t pack_swima_SLQSOMADMSetSettings_t::FOTAdownload

8.548.2.2 uint8_t pack_swima_SLQSOMADMSetSettings_t::FOTAUpdate

8.548.2.3 uint8_t* pack_swima_SLQSOMADMSetSettings_t::pAutosdm

8.548.2.4 uint8_t* pack_swima_SLQSOMADMSetSettings_t::pFwAutoCheck

8.549 pack_swima_SLQSOMADMStartSession_t Struct Reference

Data Fields

- uint32_t [sessionType](#)

8.549.1 Detailed Description

Structure that contains the session type for OMA start session command

Parameters

<i>sessionType</i> [IN]	<ul style="list-style-type: none">Session type<ul style="list-style-type: none">0x01 - FOTA, to check availability of FW Update0x02 - DM, to check availability of DM Update0x03 - PRL, to check availability of PRL Update
-------------------------	---

8.549.2 Field Documentation

8.549.2.1 uint32_t pack_swima_SLQSOMADMStartSession_t::sessionType

8.550 pack_uim_ChangePin_t Struct Reference

Data Fields

- [uim_encryptedPIN1](#) EncryptedPIN1
- uint32_t * [pIndicationToken](#)
- uint8_t * [pKeyReferenceID](#)
- [uim_sessionInformation](#) sessionInfo
- [uim_changeUIMPIN](#) changePIN
- uint16_t [Tlvresult](#)

8.550.1 Detailed Description

This structure contains information of the request parameters associated with a Change PIN API.

Parameters

<i>sessionInfo</i>	<ul style="list-style-type: none"> • See UIMSessionInformation for more information.
<i>changePIN</i>	<ul style="list-style-type: none"> • See changeUIMPIN for more information.
<i>pKey</i> ↔ <i>Reference</i> ↔ <i>D(optional)</i>	<ul style="list-style-type: none"> • Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
<i>pIndication</i> ↔ <i>Token(optional)</i>	<ul style="list-style-type: none"> • Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values <ul style="list-style-type: none"> – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.550.2 Field Documentation

8.550.2.1 `uim_changeUIMPIN` `pack_uim_ChangePin_t::changePIN`

8.550.2.2 `uim_encryptedPIN1` `pack_uim_ChangePin_t::EncryptedPIN1`

8.550.2.3 `uint32_t*` `pack_uim_ChangePin_t::pIndicationToken`

8.550.2.4 `uint8_t*` `pack_uim_ChangePin_t::pKeyReferenceID`

8.550.2.5 `uim_sessionInformation` `pack_uim_ChangePin_t::sessionInfo`

8.550.2.6 `uint16_t` `pack_uim_ChangePin_t::Tlvresult`

8.551 `pack_uim_ReadTransparent_t` Struct Reference**Data Fields**

- [uim_sessionInformation](#) `sessionInfo`
- [uim_fileInfo](#) `fileIndex`
- [uim_readTransparentInfo](#) `readTransparent`
- `uint32_t *` `pIndicationToken`
- `uint8_t *` `pEncryptData`
- `uint16_t` `Tlvresult`

8.551.1 Detailed Description

This structure contains information of the request parameters associated with a Read Transparent API.

Parameters

<i>sessionInfo</i>	<ul style="list-style-type: none"> See UIMSessionInformation for more information.
<i>fileIndex</i>	<ul style="list-style-type: none"> See fileInfo for more information.
<i>readTransparent</i>	<ul style="list-style-type: none"> See readTransparentInfo for more information.
<i>pIndication</i> \leftrightarrow <i>Token(optional)</i>	<ul style="list-style-type: none"> Response in Indication. When this TLV is present, it indicates that the result must be provided in a subsequent indication. Valid Values <ul style="list-style-type: none"> 0 - Result of operation in response. Indication will not be generated by the modem Any other positive number - Result of operation in indication. Indication will have same token value set by this function
<i>pEncrypt</i> \leftrightarrow <i>Data(optional)</i>	<ul style="list-style-type: none"> Encrypt Data. Indicates whether the data read from the card is to be encrypted.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.551.2 Field Documentation

8.551.2.1 `uim_fileInfo pack_uim_ReadTransparent_t::fileIndex`

8.551.2.2 `uint8_t* pack_uim_ReadTransparent_t::pEncryptData`

8.551.2.3 `uint32_t* pack_uim_ReadTransparent_t::pIndicationToken`

8.551.2.4 `uim_readTransparentInfo pack_uim_ReadTransparent_t::readTransparent`

8.551.2.5 `uim_sessionInformation pack_uim_ReadTransparent_t::sessionInfo`

8.551.2.6 `uint16_t pack_uim_ReadTransparent_t::Tlvresult`

8.552 pack_uim_SetPinProtection_t Struct Reference

Data Fields

- [uim_encryptedPIN1](#) EncryptedPIN1
- `uint32_t *` [pIndicationToken](#)
- `uint8_t *` [pKeyReferenceID](#)
- [uim_sessionInformation](#) [sessionInfo](#)
- [uim_setPINProtection](#) [pinProtection](#)
- `uint16_t` [Tlvresult](#)

8.552.1 Detailed Description

This structure contains information of the request parameters associated with a set pin protection API.

Parameters

<i>sessionInfo</i>	<ul style="list-style-type: none"> See uim_sessionInformation for more information.
<i>pinProtection</i>	<ul style="list-style-type: none"> See uim_setPINProtection for more information.
<i>pKeyReferenceID(optional)</i>	<ul style="list-style-type: none"> Indicates the PIN key reference ID. Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
<i>pIndicationToken(optional)</i>	<ul style="list-style-type: none"> Response in Indication. When this TLV is present, it indicates that the result must be provided in a subsequent indication. Valid Values <ul style="list-style-type: none"> 0 - Result of operation in response. Indication will not be generated by the modem Any other positive number - Result of operation in indication. Indication will have same token value set by this function

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.552.2 Field Documentation

8.552.2.1 `uim_encryptedPIN1` `pack_uim_SetPinProtection_t::EncryptedPIN1`

8.552.2.2 `uint32_t*` `pack_uim_SetPinProtection_t::pIndicationToken`

8.552.2.3 `uim_setPINProtection` `pack_uim_SetPinProtection_t::pinProtection`

8.552.2.4 `uint8_t*` `pack_uim_SetPinProtection_t::pKeyReferenceID`

8.552.2.5 `uim_sessionInformation` `pack_uim_SetPinProtection_t::sessionInfo`

8.552.2.6 `uint16_t` `pack_uim_SetPinProtection_t::Tlvresult`

8.553 `pack_uim_SLQSUIEventRegister_t` Struct Reference

Data Fields

- `uint32_t` [eventMask](#)

8.553.1 Detailed Description

Parameters

<i>eventMask</i>	<ul style="list-style-type: none">- bit 1 - card status• bit 4 - physical slot status
------------------	--

8.553.2 Field Documentation

8.553.2.1 `uint32_t pack_uim_SLQSUIMEventRegister_t::eventMask`

8.554 pack_uim_SLQSUIMPowerDown_t Struct Reference

Data Fields

- `uint8_t slot`

8.554.1 Detailed Description

This structure contains information of the request parameters associated with a Power Down.

Parameters

<i>slot</i>	<ul style="list-style-type: none">• Indicates the slot to be used.<ul style="list-style-type: none">– 1 - Slot 1– 2 - Slot 2
-------------	---

8.554.2 Field Documentation

8.554.2.1 `uint8_t pack_uim_SLQSUIMPowerDown_t::slot`

8.555 pack_uim_SLQSUIMPowerUp_t Struct Reference

Data Fields

- `uint8_t slot`
- `uint8_t * plgnoreHotSwapSwitch`

8.555.1 Detailed Description

This structure contains information of the request parameters associated with a Power Down.

Parameters

<i>slot</i>	<ul style="list-style-type: none"> Indicates the slot to be used. <ul style="list-style-type: none"> 1 - Slot 1 2 - Slot 2
<i>plgnoreHot↔ Swap↔ Switch(optional)</i>	<ul style="list-style-type: none"> Hot-swap switch status. <ul style="list-style-type: none"> 0 - Checks the hot-swap switch status 1 - Ignores the hot-swap switch status

8.555.2 Field Documentation

8.555.2.1 `uint8_t* pack_uim_SLQSUIMPowerUp_t::plgnoreHotSwapSwitch`

8.555.2.2 `uint8_t pack_uim_SLQSUIMPowerUp_t::slot`

8.556 `pack_uim_SLQSUIMSwitchSlot_t` Struct Reference

Data Fields

- `uint8_t` [bLogicalSlot](#)
- `uint32_t` [ulPhysicalSlot](#)

8.556.1 Detailed Description

This structure contains information of the request parameters associated with a Switch Slot.

Parameters

<i>bLogicalSlot</i>	<ul style="list-style-type: none"> Indicates the slot to be used. <ul style="list-style-type: none"> 1 - Slot 1 2 - Slot 2 3 - Slot 3 4 - Slot 4 5 - Slot 5
<i>ulPhysicalSlot</i>	<ul style="list-style-type: none"> 1 - Slot 1 2 - Slot 2 3 - Slot 3 4 - Slot 4 5 - Slot 5

8.556.2 Field Documentation

8.556.2.1 uint8_t pack_uim_SLQSUIMSwitchSlot_t::bLogicalSlot

8.556.2.2 uint32_t pack_uim_SLQSUIMSwitchSlot_t::ulPhysicalSlot

8.557 pack_uim_UnblockPin_t Struct Reference

Data Fields

- [uim_encryptedPIN1](#) EncryptedPIN1
- uint32_t * [pIndicationToken](#)
- uint8_t * [pKeyReferenceID](#)
- [uim_sessionInformation](#) sessionInfo
- [uim_unblockUIMPIN](#) pinProtection
- uint16_t [Tlvresult](#)

8.557.1 Detailed Description

This structure contains information of the request parameters associated with a Unblock PIN API.

Parameters

<i>EncryptedPIN1</i>	<ul style="list-style-type: none"> • See uim_encryptedPIN1 for more information.
<i>sessionInfo</i>	<ul style="list-style-type: none"> • See uim_sessionInformation for more information.
<i>pinProtection</i>	<ul style="list-style-type: none"> • See uim_unblockUIMPIN for more information.
<i>pKeyReferenceID(optional)</i>	<ul style="list-style-type: none"> • Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
<i>pIndicationToken(optional)</i>	<ul style="list-style-type: none"> • Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values <ul style="list-style-type: none"> – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function

8.557.2 Field Documentation

8.557.2.1 uim_encryptedPIN1 pack_uim_UnblockPin_t::EncryptedPIN1

8.557.2.2 uint32_t* pack_uim_UnblockPin_t::pIndicationToken

8.557.2.3 uim_unblockUIMPIN pack_uim_UnblockPin_t::pinProtection

8.557.2.4 uint8_t* pack_uim_UnblockPin_t::pKeyReferenceID

8.557.2.5 uim_sessionInformation pack_uim_UnblockPin_t::sessionInfo

8.557.2.6 uint16_t pack_uim_UnblockPin_t::Tlvresult

8.558 pack_uim_VerifyPin_t Struct Reference

Data Fields

- [uim_encryptedPIN1](#) * [pEncryptedPIN1](#)
- [uint32_t](#) * [pIndicationToken](#)
- [uint8_t](#) * [pKeyReferenceID](#)
- [uim_sessionInformation](#) [sessionInfo](#)
- [uim_verifyUIMPIN](#) [verifyPIN](#)
- [uint16_t](#) [Tlvresult](#)

8.558.1 Detailed Description

This structure contains information of the request parameters associated with a verify PIN API.

Parameters

sessionInfo	<ul style="list-style-type: none"> • See UIMSessionInformation for more information.
verifyPIN	<ul style="list-style-type: none"> • See verifyUIMPIN for more information.
pEncryptedPIN1 <i>N1(optional)</i>	<ul style="list-style-type: none"> • See encryptedPIN1 for more information.
pKeyReferenceID <i>D(optional)</i>	<ul style="list-style-type: none"> • Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
pIndicationToken <i>Token(optional)</i>	<ul style="list-style-type: none"> • Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values <ul style="list-style-type: none"> – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function
	Generated by Doxygen

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.558.2 Field Documentation

8.558.2.1 uim_encryptedPIN1* pack_uim_VerifyPin_t::pEncryptedPIN1

8.558.2.2 uint32_t* pack_uim_VerifyPin_t::pIndicationToken

8.558.2.3 uint8_t* pack_uim_VerifyPin_t::pKeyReferenceID

8.558.2.4 uim_sessionInformation pack_uim_VerifyPin_t::sessionInfo

8.558.2.5 uint16_t pack_uim_VerifyPin_t::Tlvresult

8.558.2.6 uim_verifyUIMPIN pack_uim_VerifyPin_t::verifyPIN

8.559 pack_wds_DHCPv4ClientLeaseChange_t Struct Reference

Data Fields

- uint8_t * [pEnableNotification](#)

8.559.1 Detailed Description

WDS SWI DHCPv4 Client Lease Change Structure

Parameters

<i>pEnableNotification</i>	<ul style="list-style-type: none"> • Enable Notification or not
----------------------------	--

8.559.2 Field Documentation

8.559.2.1 uint8_t* pack_wds_DHCPv4ClientLeaseChange_t::pEnableNotification

8.560 pack_wds_GetDefaultProfile_t Struct Reference

Data Fields

- uint32_t [profiletype](#)

8.560.1 Detailed Description

Parameters

<i>profiletype</i>	profile type
--------------------	--------------

8.560.2 Field Documentation

8.560.2.1 uint32_t pack_wds_GetDefaultProfile_t::profiletype

8.561 pack_wds_GetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t [type](#)
- uint8_t [family](#)

8.561.1 Detailed Description

Parameters

<i>type</i>	profile type <ul style="list-style-type: none">• 0 - 3GPP• 1 - 3GPP2
<i>type</i>	profile family <ul style="list-style-type: none">• 0 - Embedded• 1 - Tethered

8.561.2 Field Documentation

8.561.2.1 uint8_t pack_wds_GetDefaultProfileNum_t::family

8.561.2.2 uint8_t pack_wds_GetDefaultProfileNum_t::type

8.562 pack_wds_GetDormancyState_t Struct Reference

8.563 pack_wds_GetLastMobileIPError_t Struct Reference

8.564 pack_wds_GetMobileIP_t Struct Reference

8.565 pack_wds_GetMobileIPProfile_t Struct Reference

Data Fields

- uint8_t [index](#)

8.565.1 Detailed Description

Parameters

<i>index</i>	mobile ip profile identifier
--------------	------------------------------

8.565.2 Field Documentation

8.565.2.1 uint8_t pack_wds_GetMobileIPProfile_t::index

8.566 pack_wds_GetPacketStatistics_t Struct Reference

Data Fields

- uint32_t * [pStatMask](#)

8.566.1 Detailed Description

Parameters

<i>pStatMask</i>	<ul style="list-style-type: none">Packet Statistics Mask 0x00000001 - Tx packets OK 0x00000002 - Rx packets OK 0x00000004 - Tx packet errors 0x00000008 - Rx packet errors 0x00000010 - Tx overflows 0x00000020 - Rx overflows 0x00000040 - Tx bytes OK 0x00000080 - Rx bytes OK
------------------	--

8.566.2 Field Documentation

8.566.2.1 uint32_t* pack_wds_GetPacketStatistics_t::pStatMask

8.567 pack_wds_GetPacketStatus_t Struct Reference

Data Fields

- uint32_t [statmask](#)

8.567.1 Detailed Description

Parameters

<i>statmask</i>	packet statistics mask
-----------------	------------------------

8.567.2 Field Documentation

8.567.2.1 uint32_t pack_wds_GetPacketStatus_t::statmask

8.568 pack_wds_GetSessionDuration_t Struct Reference

8.569 pack_wds_RMSetTransferStatistics_t Struct Reference

Data Fields

- [rmTrasnferStaticsReq](#) [RmTrasnferStaticsReq](#)

8.569.1 Detailed Description

Parameters

rmTrasnferStaticsReq	RM Transfer Statistics Indicator
--------------------------------------	----------------------------------

8.569.2 Field Documentation

8.569.2.1 [rmTrasnferStaticsReq](#) [pack_wds_RMSetTransferStatistics_t::RmTrasnferStaticsReq](#)

8.570 pack_wds_SetAutoconnect_t Struct Reference

Data Fields

- [uint8_t](#) [acsetting](#)
- [uint8_t](#) [acroamsetting](#)

8.570.1 Detailed Description

auto connect data session parameters.

Parameters

<i>setting</i>	<ul style="list-style-type: none"> • NDIS autoconnect setting <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled
----------------	---

8.570.2 Field Documentation

8.570.2.1 [uint8_t](#) [pack_wds_SetAutoconnect_t::acroamsetting](#)

8.570.2.2 [uint8_t](#) [pack_wds_SetAutoconnect_t::acsetting](#)

8.571 pack_wds_SetDefaultProfile_t Struct Reference

Data Fields

- uint32_t [profileType](#)
- uint32_t [pdpType](#)
- uint32_t [ipAddress](#)
- uint32_t [primaryDNS](#)
- uint32_t [secondaryDNS](#)
- uint32_t [authentication](#)
- uint8_t * [pName](#)
- uint8_t * [pUsername](#)
- uint8_t * [pApnname](#)
- uint8_t * [pPassword](#)

8.571.1 Detailed Description

Parameters

<i>profileType</i>	profile type
<i>pdpType</i>	Packet Data Protocol (PDP) type
<i>ipAddress</i>	ip address
<i>primaryDNS</i>	primary dns
<i>secondaryDNS</i>	secondry dns
<i>authentication</i>	authentication type
<i>name</i>	name of the profile
<i>apnname</i>	apn name
<i>username</i>	username of the profile
<i>password</i>	password of profile

8.571.2 Field Documentation

8.571.2.1 uint32_t pack_wds_SetDefaultProfile_t::authentication

8.571.2.2 uint32_t pack_wds_SetDefaultProfile_t::ipAddress

8.571.2.3 uint8_t* pack_wds_SetDefaultProfile_t::pApnname

8.571.2.4 uint32_t pack_wds_SetDefaultProfile_t::pdpType

8.571.2.5 uint8_t* pack_wds_SetDefaultProfile_t::pName

8.571.2.6 uint8_t* pack_wds_SetDefaultProfile_t::pPassword

8.571.2.7 uint32_t pack_wds_SetDefaultProfile_t::primaryDNS

8.571.2.8 uint32_t pack_wds_SetDefaultProfile_t::profileType

8.571.2.9 `uint8_t*` `pack_wds_SetDefaultProfile_t::pUsername`

8.571.2.10 `uint32_t` `pack_wds_SetDefaultProfile_t::secondaryDNS`

8.572 `pack_wds_SetDefaultProfileNum_t` Struct Reference

Data Fields

- `uint8_t` [type](#)
- `uint8_t` [family](#)
- `uint8_t` [index](#)

8.572.1 Field Documentation

8.572.1.1 `uint8_t` `pack_wds_SetDefaultProfileNum_t::family`

8.572.1.2 `uint8_t` `pack_wds_SetDefaultProfileNum_t::index`

8.572.1.3 `uint8_t` `pack_wds_SetDefaultProfileNum_t::type`

8.573 `pack_wds_SetMobileIP_t` Struct Reference

Data Fields

- `uint32_t` [mode](#)

8.573.1 Detailed Description

Parameters

<i>mode</i>	<ul style="list-style-type: none">• Mobile IP setting<ul style="list-style-type: none">– 0 - Mobile IP off (simple IP only)– 1 - Mobile IP preferred– 2 - Mobile IP only
-------------	--

8.573.2 Field Documentation

8.573.2.1 `uint32_t` `pack_wds_SetMobileIP_t::mode`

8.574 `pack_wds_SetMobileIPParameters_t` Struct Reference

Data Fields

- `char *` [pSPC](#)

- uint32_t * [pMode](#)
- uint8_t * [pRetryLimit](#)
- uint8_t * [pRetryInterval](#)
- uint8_t * [pReRegPeriod](#)
- uint8_t * [pReRegTraffic](#)
- uint8_t * [pHAAuthenticator](#)
- uint8_t * [pHA2002bis](#)

8.574.1 Detailed Description

Mobile IP parameters information.

Parameters

<i>pSPC</i>	<ul style="list-style-type: none"> • NULL-terminated string representing six digit service programming code.
<i>pMode</i>	<ul style="list-style-type: none"> • Mode to be set (optional) <ul style="list-style-type: none"> – 0 - Mobile IP off (simple IP only) – 1 - Mobile IP preferred – 2 - Mobile IP only
<i>pRetryLimit</i>	<ul style="list-style-type: none"> • Registration retry attempt limit (optional)
<i>pRetryInterval</i>	<ul style="list-style-type: none"> • Registration retry attempt interval used to determine the time between registration attempts (optional)
<i>pReRegPeriod</i>	<ul style="list-style-type: none"> • Period (in minutes) to attempt re-registration before current registration expires (optional)
<i>pReRegTraffic</i>	<ul style="list-style-type: none"> • Re-registration only if traffic since last attempt (optional) <ul style="list-style-type: none"> – Zero - Disabled – NonZero - Enabled
<i>pHA\leftrightarrow</i> <i>Authenticator</i>	<ul style="list-style-type: none"> • MH-HA authenticator calculator (optional) <ul style="list-style-type: none"> – Zero - Disabled – NonZero - Enabled
<i>pHA2002bis</i>	<ul style="list-style-type: none"> • MH-HA RFC 2002bis authentication instead of RFC2002 (optional) <ul style="list-style-type: none"> – Zero - Disabled – NonZero - Enabled

8.574.2 Field Documentation

- 8.574.2.1 `uint8_t* pack_wds_SetMobileIPParameters_t::pHA2002bis`
- 8.574.2.2 `uint8_t* pack_wds_SetMobileIPParameters_t::pHAAuthenticator`
- 8.574.2.3 `uint32_t* pack_wds_SetMobileIPParameters_t::pMode`
- 8.574.2.4 `uint8_t* pack_wds_SetMobileIPParameters_t::pReRegPeriod`
- 8.574.2.5 `uint8_t* pack_wds_SetMobileIPParameters_t::pReRegTraffic`
- 8.574.2.6 `uint8_t* pack_wds_SetMobileIPParameters_t::pRetryInterval`
- 8.574.2.7 `uint8_t* pack_wds_SetMobileIPParameters_t::pRetryLimit`
- 8.574.2.8 `char* pack_wds_SetMobileIPParameters_t::pSPC`

8.575 `pack_wds_SetMobileIPProfile_t` Struct Reference

Data Fields

- `int8_t spc` [10]
- `uint8_t index`
- `uint8_t * pEnabled`
- `uint32_t * pAddress`
- `uint32_t * pPrimaryHA`
- `uint32_t * pSecondaryHA`
- `uint8_t * pRevTunneling`
- `int8_t * pNAI`
- `uint32_t * pHASPI`
- `uint32_t * pAAASPI`
- `int8_t * pMNHA`
- `int8_t * pMNAAA`

8.575.1 Detailed Description

Parameters

<i>spc</i>	service programming code string
<i>index</i>	Index of the profile to modify
<i>pEnabled</i>	Enable profile 0-disable nonzero enable
<i>pAddress</i>	Home IPv4 address
<i>pPrimaryHA</i>	Primary home agent IPv4 address
<i>pSecondaryHA</i>	secondary home agent IPv4 address
<i>pRevTunneling</i>	Enable reverse tunneling 0-disable nonzero enable
<i>pNAI</i>	Network access identifier string
<i>pHASPI</i>	Home agent security parameter index
<i>pAAASPI</i>	AAA server security parameter index
<i>pMNHA</i>	MN-HA key string
<i>pMNAAA</i>	MN-AAA key string

8.575.2 Field Documentation

8.575.2.1 uint8_t pack_wds_SetMobileIPProfile_t::index

8.575.2.2 uint32_t* pack_wds_SetMobileIPProfile_t::pAAASPI

8.575.2.3 uint32_t* pack_wds_SetMobileIPProfile_t::pAddress

8.575.2.4 uint8_t* pack_wds_SetMobileIPProfile_t::pEnabled

8.575.2.5 uint32_t* pack_wds_SetMobileIPProfile_t::pHASPI

8.575.2.6 int8_t* pack_wds_SetMobileIPProfile_t::pMNAAA

8.575.2.7 int8_t* pack_wds_SetMobileIPProfile_t::pMNHA

8.575.2.8 int8_t* pack_wds_SetMobileIPProfile_t::pNAI

8.575.2.9 uint32_t* pack_wds_SetMobileIPProfile_t::pPrimaryHA

8.575.2.10 uint8_t* pack_wds_SetMobileIPProfile_t::pRevTunneling

8.575.2.11 uint32_t* pack_wds_SetMobileIPProfile_t::pSecondaryHA

8.575.2.12 int8_t pack_wds_SetMobileIPProfile_t::spc[10]

8.576 pack_wds_SLQSCreateProfile_t Struct Reference

Data Fields

- uint8_t * [pProfileId](#)
- uint8_t * [pProfileType](#)
- [wds_profileInfo](#) * [pCurProfile](#)

8.576.1 Detailed Description

Parameters

<i>ProfileID</i>	<ul style="list-style-type: none"> • 1 to 16 for 3GPP profile (EM/MC73xx or earlier) • 1 to 24 for 3GPP profile (EM/MC74xx onwards) • 101 to 106 for 3GPP2 profile
<i>ProfileType</i>	<ul style="list-style-type: none"> • Identifies the technology type of the profile <ul style="list-style-type: none"> – 0x00 - 3GPP – 0x01 - 3GPP2
Generated by Doxygen	<ul style="list-style-type: none"> – NULL is not allowed
<i>curProfile</i>	<ul style="list-style-type: none"> • union of 3GPP and 3GPP2 profile

Note

- If profileID is NULL, 3GPP profile will be created and index will be assigned based on availability in device.
- If profileID is not NULL depending on pProfileType 3GPP/3GPP2 relevant profile will be created

8.576.2 Field Documentation

8.576.2.1 wds_profileInfo* pack_wds_SLQSCreateProfile_t::pCurProfile

8.576.2.2 uint8_t* pack_wds_SLQSCreateProfile_t::pProfileId

8.576.2.3 uint8_t* pack_wds_SLQSCreateProfile_t::pProfileType

8.577 pack_wds_SLQSDeleteProfile_t Struct Reference

Data Fields

- uint8_t [profileType](#)
- uint8_t [profileIndex](#)

8.577.1 Detailed Description

Parameters

<i>profileType</i>	profile type
<i>profileIndex</i>	profile index

8.577.2 Field Documentation

8.577.2.1 uint8_t pack_wds_SLQSDeleteProfile_t::profileIndex

8.577.2.2 uint8_t pack_wds_SLQSDeleteProfile_t::profileType

8.578 pack_wds_SLQSGetCurrDataSystemStat_t Struct Reference

8.579 pack_wds_SLQSGetDataBearerTechnology_t Struct Reference

8.580 pack_wds_SLQSGetDUNCallInfo_t Struct Reference

Data Fields

- uint32_t [Mask](#)
- uint8_t * [pReportConnStatus](#)
- [transferStatInd](#) * [pTransferStatInd](#)
- uint8_t * [pReportDormStatus](#)
- uint8_t * [pReportDataBearerTech](#)
- uint8_t * [pReportChannelRate](#)

8.580.1 Detailed Description

Parameters

<i>Mask</i>	most bits corresponding to the information requested to 1
<i>pReportConn↔ Status</i>	Connect Status Indicator
<i>pTransferStatInd</i>	Transfer Statistics Indicator
<i>pReportDorm↔ Status</i>	Dormancy Status Indicator
<i>pReportData↔ BearerTech</i>	Current Data Bearer Technology Indicator
<i>pReport↔ ChannelRate</i>	Channel Rate Indicator

8.580.2 Field Documentation

8.580.2.1 uint32_t pack_wds_SLQSGetDUNCallInfo_t::Mask

8.580.2.2 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportChannelRate

8.580.2.3 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportConnStatus

8.580.2.4 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportDataBearerTech

8.580.2.5 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportDormStatus

8.580.2.6 transferStatInd* pack_wds_SLQSGetDUNCallInfo_t::pTransferStatInd

8.581 pack_wds_SLQSGetProfileSettings_t Struct Reference

Data Fields

- uint8_t [ProfileID](#)
- uint8_t [ProfileType](#)

8.581.1 Detailed Description

Parameters

<i>ProfileID</i>	<ul style="list-style-type: none"> • 1 to 16 for 3GPP profile (EM/MC73xx or earlier) • 1 to 24 for 3GPP profile (EM/MC74xx onwards) • 101 to 106 for 3GPP2 profile
<i>ProfileType</i>	<ul style="list-style-type: none"> • Identifies the technology type of the profile <ul style="list-style-type: none"> – 0x00 - 3GPP – 0x01 - 3GPP2

Note

- If profileID is NULL, 3GPP profile will be fetched and index will be assigned based on availability in device.
- If profileID is not NULL depending on pProfileType 3GPP/3GPP2 relevant profile will be fetched

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

8.581.2 Field Documentation

8.581.2.1 uint8_t pack_wds_SLQSGetProfileSettings_t::ProfileId

8.581.2.2 uint8_t pack_wds_SLQSGetProfileSettings_t::ProfileType

8.582 pack_wds_SLQSGetRuntimeSettings_t Struct Reference**Data Fields**

- uint32_t * [pReqSettings](#)

8.582.1 Detailed Description

Parameters

<i>pReqSettings</i>	<p>Requested Settings (Optional Parameter)</p> <ul style="list-style-type: none"> • Set bits to 1, corresponding to requested information. All other bits must be set to 0. • If the values are not available, the corresponding TLVs are not returned in the response. • Absence of this mask TLV results in the device returning all of the available information corresponding to bits 0 through 12. • In cases where the information from bit 13 or greater is required, this TLV with all the necessary bits set must be present in the request. • Values <ul style="list-style-type: none"> – Bit 0 - Profile identifier – Bit 1 - Profile name – Bit 2 - PDP type – Bit 3 - APN name – Bit 4 - DNS address – Bit 5 - UMTS/GPRS granted QoS – Bit 6 - Username – Bit 7 - Authentication Protocol – Bit 8 - IP address – Bit 9 - Gateway info (address and subnet mask) – Bit 10 - PCSCF address using PCO flag – Bit 11 - PCSCF server address list – Bit 12 - PCSCF domain name list – Bit 13 - MTU – Bit 14 - domain name list – Bit 15 - IP family – Bit 16 - IM_CM flag – Bit 17 - Technology name – Bit 18 - Operator reserved PCO
---------------------	--

8.582.2 Field Documentation

8.582.2.1 uint32_t* pack_wds_SLQSGetRuntimeSettings_t::pReqSettings

8.583 pack_wds_SLQSModifyProfile_t Struct Reference

Data Fields

- uint8_t * [pProfileId](#)
- uint8_t * [pProfileType](#)
- [wds_profileInfo](#) curProfile

8.583.1 Detailed Description

Parameters

<i>ProfileID</i>	<ul style="list-style-type: none"> • 1 to 16 for 3GPP profile (EM/MC73xx or earlier) • 1 to 24 for 3GPP profile (EM/MC74xx onwards) • 101 to 106 for 3GPP2 profile
<i>ProfileType</i>	<ul style="list-style-type: none"> • Identifies the technology type of the profile <ul style="list-style-type: none"> – 0x00 - 3GPP – 0x01 - 3GPP2 – NULL is not allowed
<i>curProfile</i>	<ul style="list-style-type: none"> • union of 3GPP and 3GPP2 profile

Note

- If profileID is NULL, 3GPP profile will be created and index will be assigned based on availability in device.
- If profileID is not NULL depending on pProfileType 3GPP/3GPP2 relevant profile will be created

8.583.2 Field Documentation

8.583.2.1 wds_profileInfo pack_wds_SLQSMModifyProfile_t::curProfile

8.583.2.2 uint8_t* pack_wds_SLQSMModifyProfile_t::pProfileId

8.583.2.3 uint8_t* pack_wds_SLQSMModifyProfile_t::pProfileType

8.584 pack_wds_SLQSSet3GPPConfigItem_t Struct Reference

Data Fields

- uint16_t * [pLTEAttachProfile](#)
- uint16_t * [pProfileList](#)
- uint8_t * [pDefaultPDNEnabled](#)
- uint8_t * [p3gppRelease](#)
- uint16_t * [pLTEAttachProfileList](#)
- uint16_t [LTEAttachProfileListLen](#)

8.584.1 Detailed Description

Parameters

<i>pLTEAttachProfile</i>	<ul style="list-style-type: none"> Optional parameter LTE Attach Profile <ul style="list-style-type: none"> points to a single WORD Value indicating the attached LTE Profile Optional parameter with possible values 1-16 (EM/MC73xx or earlier) This setting is deprecated on MC/EM74xx
<i>ProfileList</i>	<p>Profile List</p> <ul style="list-style-type: none"> an array of 4 profile configurations Each element points to a single WORD value indicating profile Optional parameter with possible values <ul style="list-style-type: none"> 1 - 16 (MC/EM73xx and before) 1 - 24 (MC/EM74xx and onwards) function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present
<i>pDefaultPDNEnabled</i>	<ul style="list-style-type: none"> Optional parameter <ul style="list-style-type: none"> 0 - disabled 1 - enabled
<i>p3gppRelease</i>	<p>3GPP release</p> <ul style="list-style-type: none"> Optional parameter <ul style="list-style-type: none"> 0 - Release_99 1 - Release_5 2 - Release_6 3 - Release_7 4 - Release_8 In 9x30 and onwards <ul style="list-style-type: none"> 5 - Release 9 6 - Release 10 7 - Release 11
<i>pLTEAttachProfileList</i>	<ul style="list-style-type: none"> pointer to WORD array indicating LTE Attach Profile List <ul style="list-style-type: none"> Optional parameter possible values: 1-24 This setting is only supported for MC/EM74xx onwards Please provide attach profiles in order of decreasing priority in this list.
<i>LTEAttachProfileListLen</i>	<ul style="list-style-type: none"> Number of element in pLTEAttachProfileList <ul style="list-style-type: none"> valid range: 1-24 This setting is only supported for MC/EM74xx onwards

8.584.2 Field Documentation

8.584.2.1 uint16_t pack_wds_SLQSSet3GPPConfigItem_t::LTEAttachProfileListLen

8.584.2.2 `uint8_t*` `pack_wds_SLQSSet3GPPConfigItem_t::p3gppRelease`

8.584.2.3 `uint8_t*` `pack_wds_SLQSSet3GPPConfigItem_t::pDefaultPDNEnabled`

8.584.2.4 `uint16_t*` `pack_wds_SLQSSet3GPPConfigItem_t::pLTEAttachProfile`

8.584.2.5 `uint16_t*` `pack_wds_SLQSSet3GPPConfigItem_t::pLTEAttachProfileList`

8.584.2.6 `uint16_t*` `pack_wds_SLQSSet3GPPConfigItem_t::pProfileList`

8.585 `pack_wds_SLQSSetIPFamilyPreference_t` Struct Reference

Data Fields

- `uint8_t` [IPFamilyPreference](#)

8.585.1 Detailed Description

Parameters

<i>IPFamilyPreference</i>	IP Family preference <ul style="list-style-type: none"> • <code>PACK_WDS_IPV4</code> IP Version 4 • <code>PACK_WDS_IPV6</code> IP Version 6
---------------------------	---

8.585.2 Field Documentation

8.585.2.1 `uint8_t` `pack_wds_SLQSSetIPFamilyPreference_t::IPFamilyPreference`

8.586 `pack_wds_SLQSSetWdsEventCallback_t` Struct Reference

Data Fields

- `uint8_t` [dataBearer](#)
- `uint8_t` [dormancyStatus](#)
- `uint8_t` [mobileIP](#)
- `uint8_t` [transferStats](#)
- `uint8_t` [currentDataBearer](#)
- `uint8_t` [dataSystemStatus](#)
- `uint8_t` [interval](#)

8.586.1 Detailed Description

Parameters

<i>dataBearer</i>	data bearer
<i>dormancyStatus</i>	dormancy status
<i>mobileIP</i>	mobile IP
<i>currentData↔ Bearer</i>	current data bearer
<i>dataSystem↔ Status</i>	data system status
<i>interval</i>	interval

8.586.2 Field Documentation

8.586.2.1 `uint8_t pack_wds_SLQSSetWdsEventCallback_t::currentDataBearer`

8.586.2.2 `uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataBearer`

8.586.2.3 `uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataSystemStatus`

8.586.2.4 `uint8_t pack_wds_SLQSSetWdsEventCallback_t::dormancyStatus`

8.586.2.5 `uint8_t pack_wds_SLQSSetWdsEventCallback_t::interval`

8.586.2.6 `uint8_t pack_wds_SLQSSetWdsEventCallback_t::mobileIP`

8.586.2.7 `uint8_t pack_wds_SLQSSetWdsEventCallback_t::transferStats`

8.587 pack_wds_SLQSSetDHCPv4ClientConfig_t Struct Reference

Data Fields

- [wdsDhcpv4ProfileId](#) * [pProfileId](#)

8.587.1 Detailed Description

Parameters

<i>pProfileId</i>	pointer to Profile Id structure
-------------------	---------------------------------

8.587.2 Field Documentation

8.587.2.1 `wdsDhcpv4ProfileId* pack_wds_SLQSSetDHCPv4ClientConfig_t::pProfileId`

8.588 pack_wds_SLQSSetDHCPv4ClientConfig_t Struct Reference

Data Fields

- [wds_DHCPv4ProfileId](#) * [pProfileId](#)
- [wds_DHCPv4HWConfig](#) * [pHwConfig](#)
- [wds_DHCPv4OptionList](#) * [pRequestOptionList](#)

8.588.1 Detailed Description

WDS SWI DHCPv4 Config Structure

Parameters

<i>pProfileId</i>	<ul style="list-style-type: none"> • pointer to Profile Id structure
<i>pHWConfig</i>	<ul style="list-style-type: none"> • pointer to HW Config structure
<i>pRequest↔ OptionList</i>	<ul style="list-style-type: none"> • pointer to Option List structure to be sent in DHCP request

8.588.2 Field Documentation

8.588.2.1 [wds_DHCPv4HWConfig](#)* [pack_wds_SLQSSSetDHCPv4ClientConfig_t::pHwConfig](#)

8.588.2.2 [wds_DHCPv4ProfileId](#)* [pack_wds_SLQSSSetDHCPv4ClientConfig_t::pProfileId](#)

8.588.2.3 [wds_DHCPv4OptionList](#)* [pack_wds_SLQSSSetDHCPv4ClientConfig_t::pRequestOptionList](#)

8.589 [pack_wds_SLQSSSetLoopback_t](#) Struct Reference

Data Fields

- [uint8_t](#) [loopbackMode](#)
- [uint8_t](#) [loopbackMultiplier](#)

8.589.1 Detailed Description

Parameters

<i>loopbackMode</i>	<ul style="list-style-type: none"> • Loopback Mode. <ul style="list-style-type: none"> – 0 - Disable – 1 - Enable
<i>loopback↔ Multiplier</i>	<ul style="list-style-type: none"> • Loopback multiplier. Number of downlink bytes to send for each uplink byte.

8.589.2 Field Documentation

8.589.2.1 uint8_t pack_wds_SLQSSSetLoopback_t::loopbackMode

8.589.2.2 uint8_t pack_wds_SLQSSSetLoopback_t::loopbackMultiplier

8.590 pack_wds_SLQSSStartDataSession_t Struct Reference

Data Fields

- uint8_t * [pTech](#)
- uint32_t * [pprofileid3gpp](#)
- uint32_t * [pprofileid3gpp2](#)
- uint32_t * [pAuth](#)
- char * [pUser](#)
- char * [pPass](#)

8.590.1 Detailed Description

Parameters

<i>pTech</i>	<ul style="list-style-type: none"> • Indicates the technology preference <ul style="list-style-type: none"> – 1 - UMTS – 2 - CDMA – 3 - eMBMS – 4 - Modem Link Label. Modem Link is an interface for transferring data between entities on AP and modem. • optional
<i>pprofileid3gpp</i>	<ul style="list-style-type: none"> • pointer to 3GPP profile id • optional
<i>pprofileid3gpp2</i>	<ul style="list-style-type: none"> • pointer to 3GPPs profile id • optional
<i>pAuth</i>	<ul style="list-style-type: none"> • Authentication type, it can be PAP or CHAP • optional
<i>pUser</i>	<ul style="list-style-type: none"> • username for authentication process • optional
<i>pPass</i>	<ul style="list-style-type: none"> • password for authentication process • optional

8.590.2 Field Documentation

8.590.2.1 `uint32_t*` `pack_wds_SLQSSstartDataSession_t::pAuth`

8.590.2.2 `char*` `pack_wds_SLQSSstartDataSession_t::pPass`

8.590.2.3 `uint32_t*` `pack_wds_SLQSSstartDataSession_t::pprofileid3gpp`

8.590.2.4 `uint32_t*` `pack_wds_SLQSSstartDataSession_t::pprofileid3gpp2`

8.590.2.5 `uint8_t*` `pack_wds_SLQSSstartDataSession_t::pTech`

8.590.2.6 `char*` `pack_wds_SLQSSstartDataSession_t::pUser`

8.591 `pack_wds_SLQSSstopDataSession_t` Struct Reference

Data Fields

- `uint32_t *` `psid`

8.591.1 Detailed Description

Parameters

<code>sid</code>	session id
------------------	------------

8.591.2 Field Documentation

8.591.2.1 `uint32_t*` `pack_wds_SLQSSstopDataSession_t::psid`

8.592 `pack_wds_SLQSWdsSetEventReport_t` Struct Reference

Data Fields

- `uint8_t *` `pCurrChannelRateInd`
- `wds_TrStatInd *` `pTransferStatInd`
- `uint8_t *` `pDataBearerTechInd`
- `uint8_t *` `pDormancyStatusInd`
- `uint8_t *` `pMIPStatusInd`
- `uint8_t *` `pCurrDataBearerTechInd`
- `uint8_t *` `pDataCallStatusChangeInd`
- `uint8_t *` `pCurrPrefDataSysInd`
- `uint8_t *` `pEVDOPageMonPerChangeInd`
- `uint8_t *` `pDataSystemStatusChangeInd`

8.592.1 Detailed Description

This structure contains the information about the Set Event Report Request parameters.

Parameters

<i>pCurrChannel↔ RateInd</i>	(optional) <ul style="list-style-type: none"> Current Channel Rate Indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report channel rate when it changes
<i>pTransferStatInd</i>	(optional) <ul style="list-style-type: none"> See TrStatInd for more information.
<i>pDataBearer↔ TechInd</i>	(optional) <ul style="list-style-type: none"> Data Bearer Technology Indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report radio interface used for data transfer when it changes
<i>pDormancy↔ StatusInd</i>	(optional) <ul style="list-style-type: none"> Dormancy Status indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report traffic channel state of interface used for data connection
<i>pMIPStatusInd</i>	(optional) <ul style="list-style-type: none"> MIP Status Indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report MIP status
<i>pCurrData↔ BearerTechInd</i>	(optional) <ul style="list-style-type: none"> Current Data Bearer Technology Indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report current data bearer technology when it changes
<i>pDataCall↔ StatusChange↔ Ind</i>	(optional) <ul style="list-style-type: none"> Data Call Status Change Indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report data call status change when it changes
<i>pCurrPrefData↔ SysInd</i>	(optional) <ul style="list-style-type: none"> Current Preferred Data System Indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report preferred data system when it changes
<i>pEVDOPage↔ MonPer↔ ChangeInd</i>	(optional) <ul style="list-style-type: none"> EV-DO Page Monitor Period Change Indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report EV-DO page monitor period change event
<i>pDataSystem↔ StatusChange↔ Ind</i>	(optional) <ul style="list-style-type: none"> Data System Status Change Indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report data system status change event

Note

At least one parameter should be present.

8.592.2 Field Documentation

8.592.2.1 `uint8_t*` `pack_wds_SLQSWdsSetEventReport_t::pCurrChannelRateInd`

8.592.2.2 `uint8_t*` `pack_wds_SLQSWdsSetEventReport_t::pCurrDataBearerTechInd`

8.592.2.3 `uint8_t*` `pack_wds_SLQSWdsSetEventReport_t::pCurrPrefDataSysInd`

8.592.2.4 `uint8_t*` `pack_wds_SLQSWdsSetEventReport_t::pDataBearerTechInd`

8.592.2.5 `uint8_t*` `pack_wds_SLQSWdsSetEventReport_t::pDataCallStatusChangeInd`

8.592.2.6 `uint8_t*` `pack_wds_SLQSWdsSetEventReport_t::pDataSystemStatusChangeInd`

8.592.2.7 `uint8_t*` `pack_wds_SLQSWdsSetEventReport_t::pDormancyStatusInd`

8.592.2.8 `uint8_t*` `pack_wds_SLQSWdsSetEventReport_t::pEVDOPageMonPerChangeInd`

8.592.2.9 `uint8_t*` `pack_wds_SLQSWdsSetEventReport_t::pMIPStatusInd`

8.592.2.10 `wds_TrStatInd*` `pack_wds_SLQSWdsSetEventReport_t::pTransferStatInd`

8.593 pack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference**Data Fields**

- `uint8_t` `contextId`
- `uint8_t` `contextType`

8.593.1 Detailed Description**Parameters**

<code>contextId</code>	Context Identifier
<code>contextType</code>	Context Type 0-3GPP 1-3GPP2

8.593.2 Field Documentation

8.593.2.1 `uint8_t` `pack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextId`

8.593.2.2 `uint8_t` `pack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextType`

8.594 PackCreateProfileOut Struct Reference

Data Fields

- uint8_t [ProfileType](#)
- uint8_t [ProfileIndex](#)
- uint16_t [ExtErrorCode](#)

8.594.1 Field Documentation

8.594.1.1 uint16_t PackCreateProfileOut::ExtErrorCode

8.594.1.2 uint8_t PackCreateProfileOut::ProfileIndex

8.594.1.3 uint8_t PackCreateProfileOut::ProfileType

8.595 packgetDyingGaspCfg Struct Reference

Data Fields

- uint8_t * [pDestSMSNum](#)
- uint8_t * [pDestSMSContent](#)

8.595.1 Detailed Description

Parameters

<i>pDestSMS↔ Num[IN]</i>	<ul style="list-style-type: none"> • SMS Destination Number as string of 8 bit ASCII Characters Max 20 chars. • Optional parameter.
<i>pDestSMS↔ Content[IN]</i>	<ul style="list-style-type: none"> • SMS CContent as a string of 8 bit ASCII text characters Max 160 chars. • Optional parameter.

8.595.2 Field Documentation

8.595.2.1 uint8_t* packgetDyingGaspCfg::pDestSMSContent

8.595.2.2 uint8_t* packgetDyingGaspCfg::pDestSMSNum

8.596 packgetDyingGaspStatistics Struct Reference

Data Fields

- uint32_t * [pTimeStamp](#)
- uint8_t * [pSMSAttemptedFlag](#)

8.596.1 Detailed Description

Parameters

<i>TimeStamp</i> [O↔ UT]	<ul style="list-style-type: none"> Time Stamp.
<i>SMS</i> ↔ <i>Attempted</i> ↔ <i>Flag</i> [OUT]	<ul style="list-style-type: none"> SMS Attempted Flag.

8.596.2 Field Documentation

8.596.2.1 uint8_t* packgetDyingGaspStatistics::pSMSAttemptedFlag

8.596.2.2 uint32_t* packgetDyingGaspStatistics::pTimeStamp

8.597 PCMparams Struct Reference

Data Fields

- [BYTE iFaceTabLen](#)
- [BYTE iFaceTab](#) [255]

8.597.1 Detailed Description

This structure contains the PCM parameters.

Parameters

<i>iFaceTabLen</i>	<ul style="list-style-type: none"> Number of sets of iface table
<i>iFaceTab</i>	<ul style="list-style-type: none"> Physical Interface Parameters See qaGobiApiTableSwiAudio.h for more information on physical interface parameters

8.597.2 Field Documentation

8.597.2.1 BYTE PCMparams::iFaceTab[255]

8.597.2.2 BYTE PCMparams::iFaceTabLen

8.598 PCSCFFQDNAddress Struct Reference

Data Fields

- [WORD fqdnLen](#)
- [CHAR fqdnAddr](#) [256]

8.598.1 Detailed Description

This structure contains the [PCSCFFQDNAddress](#) Information

Parameters

<i>fqdnLen</i>	<ul style="list-style-type: none">• length of the received FQDN address
<i>fqdnAddr</i>	<ul style="list-style-type: none">• FQDN address(Max 256 characters)

8.598.2 Field Documentation

8.598.2.1 [CHAR PCSCFFQDNAddress::fqdnAddr](#)[256]

8.598.2.2 [WORD PCSCFFQDNAddress::fqdnLen](#)

8.599 PCSCFFQDNAddressList Struct Reference

Data Fields

- [BYTE numInstances](#)
- struct [PCSCFFQDNAddress pcsfFQDNAddress](#) [10]

8.599.1 Detailed Description

This structure contains the [PCSCFFQDNAddressList](#) Information

Parameters

<i>numInstances</i>	<ul style="list-style-type: none">• Number of FQDN addresses received
<i>pcsfFQDNAddress</i>	<ul style="list-style-type: none">• FQDN address information(Max 10 addresses)

8.599.2 Field Documentation

8.599.2.1 **BYTE** PCSCFFQDNAddressList::numInstances

8.599.2.2 **struct** PCSCFFQDNAddress PCSCFFQDNAddressList::pcsfQDNAddress[10]

8.600 PCSCFIPv4ServerAddressList Struct Reference

Data Fields

- [BYTE](#) numInstances
- [ULONG](#) pscsfIPv4Addr [64]

8.600.1 Detailed Description

This structure contains the [PCSCFIPv4ServerAddressList](#) Information

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • number of address following
<i>pscsfIPv4Addr</i>	<ul style="list-style-type: none"> • P-CSCF IPv4 server addresses(Max 16 address, 4 bytes each)

8.600.2 Field Documentation

8.600.2.1 **BYTE** PCSCFIPv4ServerAddressList::numInstances

8.600.2.2 **ULONG** PCSCFIPv4ServerAddressList::pscsfIPv4Addr[64]

8.601 PDSPositionData Struct Reference

Data Fields

- [ULONGLONG](#) * pTimeStamp
- [ULONGLONG](#) * pLatitude
- [ULONGLONG](#) * pLongitude
- [ULONG](#) * pAltitudeWrtEllipsoid
- [ULONG](#) * pAltitudeWrtSealevel
- [ULONG](#) * pHorizontalUncCircular
- [ULONG](#) * pVerticalUnc
- [BYTE](#) * pHorizontalConfidence
- [BYTE](#) * pVerticalConfidence
- [BYTE](#) * pPositionSource
- [BYTE](#) * pTimeType

8.601.1 Detailed Description

Position Data Parameters from the external source to be injected to PDS engine.

Parameters

<i>pTimeStamp</i>	<ul style="list-style-type: none"> Timestamp of the injected position in msec. The time can be of type UTC, GPS, or Age and is defined in the pTimeType parameter. If the pTimeType is not present, the timestamp shall be assumed to be UTC time
<i>pLatitude</i>	<ul style="list-style-type: none"> Latitude position referenced to the WGS-84 reference ellipsoid, counting positive angles north of the equator and negative angles south of the equator. Value (in decimal degrees) in the range from -90 degrees to +90 degrees. Value in double float format (refer to IEEE Std 754-1985)
<i>pLongitude</i>	<ul style="list-style-type: none"> Longitude position referenced to the WGS-84 reference ellipsoid, counting positive angles east of the Greenwich Meridian and negative angles west of Greenwich meridian. Value (in decimal degrees) in the range from -180 degrees to +180 degrees.
<i>pAltitudeWrt↔ Ellipsoid</i>	<ul style="list-style-type: none"> Height above the WGS-84 reference ellipsoid. Value conveys height (in meters). When injecting altitude information, the control point should include either this parameter or the pAltitudeWrt↔Sealevel parameter. Value in single float format (refer to IEEE Std 754-1985)
<i>pAltitudeWrt↔ Sealevel</i>	<ul style="list-style-type: none"> Height of MS above the mean sea level in units (in meters). When injecting altitude information, the control point should include either this parameter or the pAltitudeWrtEllipsoid parameter. Value in single float format (refer to IEEE Std 754-1985)
<i>pHorizontal↔ UncCircular</i>	<ul style="list-style-type: none"> Circular horizontal uncertainty (in meters). This parameter must be included if the latitude and longitude parameters are specified. Value in single float format (refer to IEEE Std 754-1985)
<i>pVerticalUnc</i>	<ul style="list-style-type: none"> Vertical uncertainty (in meters). This parameter must be included if one of the altitude parameter are specified. Value in single float format (refer to IEEE Std 754-1985)
<i>pHorizontal↔ Confidence</i>	<ul style="list-style-type: none"> Confidence value of the location horizontal uncertainty, specified as percentage, 1 to 100. This parameter must be included if the latitude and longitude parameters are specified.
<i>pVertical↔ Confidence</i>	<ul style="list-style-type: none"> Confidence value of the location vertical uncertainty, specified as percentage, 1 to 100. This parameter must be included if one of the altitude paramters are specified.

<i>pPositionSource</i>	<ul style="list-style-type: none"> • Source of injected position: <ul style="list-style-type: none"> – 0x00 - Unknown – 0x01 - GPS – 0x02 - Cell ID – 0x03 - Enhanced cell ID – 0x04 - WiFi – 0x05 - Terrestrial – 0x06 - Terrestrial hybrid – 0x07 - Other
<i>pTimeType</i>	<ul style="list-style-type: none"> • Defines the time value set in the pTimeStamp parameter. <ul style="list-style-type: none"> – 0x00 - UTC Time: starting Jan 1, 1970 – 0x01 - GPS Time: starting Jan 6, 1980 – 0x02 - Age: Age of position information

8.601.2 Field Documentation

8.601.2.1 **ULONG*** PDSPositionData::pAltitudeWrtEllipsoid

8.601.2.2 **ULONG*** PDSPositionData::pAltitudeWrtSealevel

8.601.2.3 **BYTE*** PDSPositionData::pHorizontalConfidence

8.601.2.4 **ULONG*** PDSPositionData::pHorizontalUncCircular

8.601.2.5 **ULONGLONG*** PDSPositionData::pLatitude

8.601.2.6 **ULONGLONG*** PDSPositionData::pLongitude

8.601.2.7 **BYTE*** PDSPositionData::pPositionSource

8.601.2.8 **ULONGLONG*** PDSPositionData::pTimeStamp

8.601.2.9 **BYTE*** PDSPositionData::pTimeType

8.601.2.10 **BYTE*** PDSPositionData::pVerticalConfidence

8.601.2.11 **ULONG*** PDSPositionData::pVerticalUnc

8.602 PDSPosMethodStateReq Struct Reference

Data Fields

- **BYTE *** pXtraTimeState
- **BYTE *** pXtraDataState
- **BYTE *** pWifiState

8.602.1 Detailed Description

Parameters to Set state of positioning method for a device.

Parameters

<i>pXtraTimeState</i>	<ul style="list-style-type: none"> • XTRA Time Position Method State. • Values: <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pXtraDataState</i>	<ul style="list-style-type: none"> • XTRA Data Position Method State. • Values: <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>Latitude</i>	<ul style="list-style-type: none"> • WiFi Position Method State • Values: <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable

8.602.2 Field Documentation

8.602.2.1 **BYTE*** PDSPosMethodStateReq::pWifiState

8.602.2.2 **BYTE*** PDSPosMethodStateReq::pXtraDataState

8.602.2.3 **BYTE*** PDSPosMethodStateReq::pXtraTimeState

8.603 peerNumberInfo Struct Reference

Data Fields

- [BYTE](#) callID
- [BYTE](#) numPI
- [BYTE](#) numSI
- [BYTE](#) numType
- [BYTE](#) numPlan
- [BYTE](#) numLen
- [BYTE](#) number [81]

8.603.1 Detailed Description

This structure contains information for Connected Peer Numbers.

Parameters

<i>callID</i>	<ul style="list-style-type: none"> • Unique call identifier for the call.
<i>numPI</i>	<ul style="list-style-type: none"> • Number presentation indicator. <ul style="list-style-type: none"> – 0x00 - PRESENTATION_ALLOWED - Allowed presentation – 0x01 - PRESENTATION_RESTRICTED - Restricted presentation – 0x02 - PRESENTATION_NUM_UNAVAILABLE - Unavailable presentation – 0x04 - PRESENTATION_PAYPHONE - Payphone presentation (GSM/UMTS specific) – 0xFF - Not Available
<i>numSI</i>	<ul style="list-style-type: none"> • Number screening indicator. <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network
<i>numType</i>	<ul style="list-style-type: none"> • Number type. <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension
<i>numPlan</i>	<ul style="list-style-type: none"> • Number plan. <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension
<i>numLen</i>	<ul style="list-style-type: none"> • Provides the length of number which follow.
<i>number</i> [MAX_↔ CALL_NO_LEN]	<ul style="list-style-type: none"> • number of numLen length, NULL terminated.

8.603.2 Field Documentation

8.603.2.1 **BYTE** peerNumberInfo::callID

8.603.2.2 **BYTE** peerNumberInfo::number[81]

8.603.2.3 **BYTE** peerNumberInfo::numLen

8.603.2.4 **BYTE** peerNumberInfo::numPI

8.603.2.5 **BYTE** peerNumberInfo::numPlan

8.603.2.6 **BYTE** peerNumberInfo::numSI

8.603.2.7 **BYTE** peerNumberInfo::numType

8.604 personalizationStatus Struct Reference

Data Fields

- [BYTE numFeatures](#)
- [BYTE feature](#) [12]
- [BYTE verifyLeft](#) [12]
- [BYTE unblockLeft](#) [12]

8.604.1 Detailed Description

This structure contains the information about the card result.

Parameters

<i>numFeatures</i>	<ul style="list-style-type: none"> • Number of active personalization features. The following block is repeated for each feature.
<i>feature</i>	<ul style="list-style-type: none"> • Indicates the personalization feature to deactivate or unblock. Valid values: <ul style="list-style-type: none"> – 0 - GW network personalization – 1 - GW network subset personalization – 2 - GW service provider personalization – 3 - GW corporate personalization – 4 - GW UIM personalization – 5 - 1X network type 1 personalization – 6 - 1X network type 2 personalization – 7 - 1X HRPD personalization – 8 - 1X service provider personalization – 9 - 1X corporate personalization – 10 - 1X RUIM personalization
<i>verifyLeft</i> Generated by Doxygen	<ul style="list-style-type: none"> • Number of the remaining attempts to verify the personalization feature.
<i>unblockLeft</i>	<ul style="list-style-type: none"> • Number of the remaining attempts to unblock the personalization feature.

8.604.2 Field Documentation

8.604.2.1 **BYTE** `personalizationStatus::feature[12]`

8.604.2.2 **BYTE** `personalizationStatus::numFeatures`

8.604.2.3 **BYTE** `personalizationStatus::unblockLeft[12]`

8.604.2.4 **BYTE** `personalizationStatus::verifyLeft[12]`

8.605 PhyCaAggPcellInfo Struct Reference

Data Fields

- int `pci`
- int `freq`
- [NAS_LTE_CPHY_CA_BW_NRB](#) `dl_bw_value`
- int `iLTEbandValue`
- **BYTE** `TlvPresent`

8.605.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Pcell Information.

Parameters

<i>pci</i>	<ul style="list-style-type: none"> • Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
<i>freq</i>	<ul style="list-style-type: none"> • Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
<i>dl_bw_value</i>	<ul style="list-style-type: none"> • Downlink Bandwidth Values. • See NAS_LTE_CPHY_CA_BW_NRB for more information.
<i>scell_state</i>	<ul style="list-style-type: none"> • SCell state Values. • See NAS_LTE_CPHY_SCELL_STATE for more information.
<i>TlvPresent</i>	<ul style="list-style-type: none"> • Tlv Present.

8.605.2 Field Documentation

8.605.2.1 `NAS_LTE_CPHY_CA_BW_NRB` `PhyCaAggPcellInfo::dl_bw_value`

8.605.2.2 `int` `PhyCaAggPcellInfo::freq`

8.605.2.3 `int` `PhyCaAggPcellInfo::ltebandValue`

8.605.2.4 `int` `PhyCaAggPcellInfo::pci`

8.605.2.5 `BYTE` `PhyCaAggPcellInfo::TlvPresent`

8.606 PhyCaAggScellIDIBw Struct Reference

Data Fields

- [NAS_LTE_CPHY_CA_BW_NRB](#) `dl_bw_value`
- `BYTE` `TlvPresent`

8.606.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation Downlink Bandwidth of Scell.

Parameters

<code>dl_bw_value</code>	<ul style="list-style-type: none">• Downlink Bandwidth Values.• See NAS_LTE_CPHY_CA_BW_NRB for more information.
--------------------------	---

8.606.2 Field Documentation

8.606.2.1 `NAS_LTE_CPHY_CA_BW_NRB` `PhyCaAggScellIDIBw::dl_bw_value`

8.606.2.2 `BYTE` `PhyCaAggScellIDIBw::TlvPresent`

8.607 PhyCaAggScellIndex Struct Reference

Data Fields

- `BYTE` `scell_idx`
- `BYTE` `TlvPresent`

8.607.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Index.

Parameters

<i>scell_idx</i>	<ul style="list-style-type: none"> Physical cell ID of the SCell Range. Range for ID values: 0 to 503.
<i>TlvPresent</i>	<ul style="list-style-type: none"> Tlv Present.

8.607.2 Field Documentation

8.607.2.1 BYTE PhyCaAggScellIndex::scell_idx

8.607.2.2 BYTE PhyCaAggScellIndex::TlvPresent

8.608 PhyCaAggScellIndType Struct Reference

Data Fields

- int [pci](#)
- int [freq](#)
- [NAS_LTE_CPHY_SCELL_STATE](#) [scell_state](#)
- BYTE [TlvPresent](#)

8.608.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Indicator Type.

Parameters

<i>pci</i>	<ul style="list-style-type: none"> Physical cell ID of the SCell Range. Range for ID values: 0 to 503.
<i>freq</i>	<ul style="list-style-type: none"> Frequency of the absolute cell Range. Range for ID values: 0 to 65535.
<i>scell_state</i>	<ul style="list-style-type: none"> Scell state Values. See NAS_LTE_CPHY_SCELL_STATE for more information.
<i>TlvPresent</i>	<ul style="list-style-type: none"> Tlv Present.

8.608.2 Field Documentation

8.608.2.1 int PhyCaAggScellIndType::freq

8.608.2.2 int PhyCaAggScellIndType::pci

8.608.2.3 NAS_LTE_CPHY_SCELL_STATE PhyCaAggScellIndType::scell_state

8.608.2.4 BYTE PhyCaAggScellIndType::TlvPresent

8.609 PhyCaAggScellInfo Struct Reference

Data Fields

- int [pci](#)
- int [freq](#)
- [NAS_LTE_CPHY_CA_BW_NRB](#) dl_bw_value
- int [iLTEbandValue](#)
- [NAS_LTE_CPHY_SCELL_STATE](#) scell_state
- [BYTE](#) TlvPresent

8.609.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Information.

Parameters

<i>pci</i>	<ul style="list-style-type: none">• Physical cell ID of the SCell Range.• Range for ID values: 0 to 503.
<i>freq</i>	<ul style="list-style-type: none">• Frequency of the absolute cell Range.• Range for ID values: 0 to 65535.
<i>dl_bw_value</i>	<ul style="list-style-type: none">• Downlink Bandwidth Values.• See NAS_LTE_CPHY_CA_BW_NRB for more information.

<i>ltebandValue</i>	<ul style="list-style-type: none"> • Band value. • Range for LTE Band class 120 to 160. <ul style="list-style-type: none"> – 120 - LTE E-UTRA Operating Band 1 – 121 - LTE E-UTRA Operating Band 2 – 122 - LTE E-UTRA Operating Band 3 – 123 - LTE E-UTRA Operating Band 4 – 124 - LTE E-UTRA Operating Band 5 – 125 - LTE E-UTRA Operating Band 6 – 126 - LTE E-UTRA Operating Band 7 – 127 - LTE E-UTRA Operating Band 8 – 128 - LTE E-UTRA Operating Band 9 – 129 - LTE E-UTRA Operating Band 10 – 130 - LTE E-UTRA Operating Band 11 – 131 - LTE E-UTRA Operating Band 12 – 132 - LTE E-UTRA Operating Band 13 – 133 - LTE E-UTRA Operating Band 14 – 134 - LTE E-UTRA Operating Band 17 – 135 - LTE E-UTRA Operating Band 33 – 136 - LTE E-UTRA Operating Band 34 – 137 - LTE E-UTRA Operating Band 35 – 138 - LTE E-UTRA Operating Band 36 – 139 - LTE E-UTRA Operating Band 37 – 140 - LTE E-UTRA Operating Band 38 – 141 - LTE E-UTRA Operating Band 39 – 142 - LTE E-UTRA Operating Band 40 – 143 - LTE E-UTRA Operating Band 18 – 144 - LTE E-UTRA Operating Band 19 – 145 - LTE E-UTRA Operating Band 20 – 146 - LTE E-UTRA Operating Band 21 – 147 - LTE E-UTRA Operating Band 24 – 148 - LTE E-UTRA Operating Band 25 – 149 - LTE E-UTRA Operating Band 41 – 150 - LTE E-UTRA Operating Band 42 – 151 - LTE E-UTRA Operating Band 43 – 152 - LTE E-UTRA Operating Band 23 – 153 - LTE E-UTRA Operating Band 26 – 154 - LTE E-UTRA Operating Band 32 – 155 - LTE E-UTRA Operating Band 125 – 156 - LTE E-UTRA Operating Band 126 – 157 - LTE E-UTRA Operating Band 127 – 158 - LTE E-UTRA Operating Band 28 – 159 - LTE E-UTRA Operating Band 29 – 160 - LTE E-UTRA Operating Band 30
<i>cell_state</i>	<ul style="list-style-type: none"> • Scell state Values. • See NAS_LTE_CPHY_SCELL_STATE for more information.

<i>TlvPresent</i>	<ul style="list-style-type: none">• Tlv Present.
-------------------	--

8.609.2 Field Documentation

8.609.2.1 **NAS_LTE_CPHY_CA_BW_NRB** PhyCaAggScellInfo::dl_bw_value

8.609.2.2 **int** PhyCaAggScellInfo::freq

8.609.2.3 **int** PhyCaAggScellInfo::ltebandValue

8.609.2.4 **int** PhyCaAggScellInfo::pci

8.609.2.5 **NAS_LTE_CPHY_SCELL_STATE** PhyCaAggScellInfo::scell_state

8.609.2.6 **BYTE** PhyCaAggScellInfo::TlvPresent

8.610 PilotSetData Struct Reference

Data Fields

- [BYTE](#) NumPilots
- [PilotSetParams](#) * [pPilotSetInfo](#)

8.610.1 Detailed Description

This structure contains Pilot Set Data

Parameters

<i>NumPilots</i> (IN↔ OUT)	<ul style="list-style-type: none">• Number of Pilot Sets• As input specifies number of sets of parameter pPilotSetInfo for which memory has been assigned• As output specifies the actual number of sets of parameter pPilotSetInfo returned by device
<i>pPilotSetInfo</i>	<ul style="list-style-type: none">• Pilot Set Parameters• See PilotSetParams for more information.

note A buffer under sized error is returned if the number of sets of [pPilotSetInfo](#) returned by the device is greater than the value in NumPilots input parameter.

8.610.2 Field Documentation

8.610.2.1 BYTE PilotSetData::NumPilots

8.610.2.2 PilotSetParams* PilotSetData::pPilotSetInfo

8.611 PilotSetParams Struct Reference

Data Fields

- [ULONG PilotType](#)
- [WORD PilotPN](#)
- [WORD PilotStrength](#)

8.611.1 Detailed Description

This structure contains Pilot Set parameters

Parameters

<i>PilotType</i>	<ul style="list-style-type: none">• 0x00 - NAS_HRPD_PILOT_CURR_ACT_PLT Current Active Pilot• 0x01 - NAS_HRPD_PILOT_NEIGHBOR_PLT Neighbor pilot information
<i>PilotPN</i>	<ul style="list-style-type: none">• Pilot PN sequence offset index
<i>PilotStrength</i>	<ul style="list-style-type: none">• Strength of the pilot (in dB)

8.611.2 Field Documentation

8.611.2.1 WORD PilotSetParams::PilotPN

8.611.2.2 WORD PilotSetParams::PilotStrength

8.611.2.3 ULONG PilotSetParams::PilotType

8.612 pktErrRate Struct Reference

Data Fields

- [WORD multiplier](#)
- [WORD exponent](#)

8.612.1 Detailed Description

This structure contains the IP flow packet error rate

Parameters

<i>multiplier</i>	Factor m in calculating packet error rate: $E = m \cdot 10^{-(p)}$
<i>exponent</i>	Factor p in calculating packet error rate (see above)

8.612.2 Field Documentation

8.612.2.1 WORD pktErrRate::exponent

8.612.2.2 WORD pktErrRate::multiplier

8.613 PLMNNetworkName Struct Reference

Data Fields

- [BYTE numInstance](#)
- [PLMNNetworkNameData PLMNNetName](#) [255]

8.613.1 Detailed Description

This structure contains PLMN Network Name as defined in 3GPP TS 24.008 (Section 10.5.3.5a) from multiple sources.

Parameters

<i>numInstance</i>	<ul style="list-style-type: none">• Number of sets of the elements.
<i>PLMNNetName</i>	<ul style="list-style-type: none">• Refer PLMNNetworkNameData for details (Optional).

8.613.2 Field Documentation

8.613.2.1 BYTE PLMNNetworkName::numInstance

8.613.2.2 PLMNNetworkNameData PLMNNetworkName::PLMNNetName[255]

8.614 PLMNNetworkNameData Struct Reference

Data Fields

- [BYTE codingScheme](#)

- [BYTE countryInitials](#)
- [BYTE longNameSpareBits](#)
- [BYTE shortNameSpareBits](#)
- [BYTE longNameLen](#)
- [BYTE longName](#) [255]
- [BYTE shortNameLen](#)
- [BYTE shortName](#) [255]

8.614.1 Detailed Description

This structure contains PLMN Network Name Data from multiple sources.

Parameters

<i>codingScheme</i>	<ul style="list-style-type: none"> • Coding scheme: <ul style="list-style-type: none"> – 0 - CODING_SCHEME_CELL_BROADCAST_GSM - Cell broadcast data coding scheme, GSM default alphabet, language unspecified;defined in 3GPP TS 23.038. – 1 - CODING_SCHEME_UCS2 - UCS2 (16 bit);defined in ISO/IEC 10646
<i>countryInitials</i>	<ul style="list-style-type: none"> • Country's initials: <ul style="list-style-type: none"> – 0 - COUNTRY_INITIALS_DO_NOT_ADD - MS should not add the letters for the country's initials to the text string. – 1 - COUNTRY_INITIALS_ADD - MS should add the letters for the country's initials and a separator, e.g., a space, to the text string.
<i>longName</i> ↔ <i>SpareBits</i>	<ul style="list-style-type: none"> • Long Name Spare Bits: <ul style="list-style-type: none"> – 1 - SPARE_BITS_8 - Bit 8 is spare and set to 0 in octet n – 2 - SPARE_BITS_7_TO_8 - Bits 7 and 8 are spare and set to 0 in octet n. – 3 - SPARE_BITS_6_TO_8 - Bits 6 to 8 (inclusive) are spare and set to 0 in octet n. – 4 - SPARE_BITS_5_TO_8 - Bits 5 to 8 (inclusive) are spare and set to 0 in octet n. – 5 - SPARE_BITS_4_TO_8 - Bits 4 to 8 (inclusive) are spare and set to 0 in octet n. – 6 - SPARE_BITS_3_TO_8 - Bits 3 to 8 (inclusive) are spare and set to 0 in octet n. – 7 - SPARE_BITS_2_TO_8 - Bits 2 to 8 (inclusive) are spare and set to 0 in octet n. – 0 - SPARE_BITS_UNKNOWN - Carries no information about the number of spare bits in octet n.
<i>shortName</i> ↔ <i>SpareBits</i>	<ul style="list-style-type: none"> • Short Name Spare Bits: <ul style="list-style-type: none"> – 1 - SPARE_BITS_8 - Bit 8 is spare and set to 0 in octet n. – 2 - SPARE_BITS_7_TO_8 - Bits 7 and 8 are spare and set to 0 in octet n. – 3 - SPARE_BITS_6_TO_8 - Bits 6 to 8 (inclusive) are spare and set to 0 in octet n. – 4 - SPARE_BITS_5_TO_8 - Bits 5 to 8 (inclusive) are spare and set to 0 in octet n. – 5 - SPARE_BITS_4_TO_8 - Bits 4 to 8 (inclusive) are spare and set to 0 in octet n. – 6 - SPARE_BITS_3_TO_8 - Bits 3 to 8 (inclusive) are spare and set to 0 in octet n. – 7 - SPARE_BITS_2_TO_8 - Bits 2 to 8 (inclusive) are spare and set to 0 in octet n. – 0 - SPARE_BITS_UNKNOWN - Carries no information about the number of spare bits in octet n.

<i>longNameLen</i>	<ul style="list-style-type: none">• It provides the length of long name.
<i>longName</i>	<ul style="list-style-type: none">• Long name string in coding_scheme.
<i>shortNameLen</i>	<ul style="list-style-type: none">• It provides the length of short name.
<i>shortName</i>	<ul style="list-style-type: none">• Short name string in coding_scheme.

8.614.2 Field Documentation

8.614.2.1 **BYTE** PLMNNetworkNameData::codingScheme

8.614.2.2 **BYTE** PLMNNetworkNameData::countryInitials

8.614.2.3 **BYTE** PLMNNetworkNameData::longName[255]

8.614.2.4 **BYTE** PLMNNetworkNameData::longNameLen

8.614.2.5 **BYTE** PLMNNetworkNameData::longNameSpareBits

8.614.2.6 **BYTE** PLMNNetworkNameData::shortName[255]

8.614.2.7 **BYTE** PLMNNetworkNameData::shortNameLen

8.614.2.8 **BYTE** PLMNNetworkNameData::shortNameSpareBits

8.615 Port Struct Reference

Data Fields

- [WORD](#) port
- [WORD](#) range

8.615.1 Detailed Description

This structure contains the [Port](#) Filter

Parameters

<i>port</i>	port value of the filter
<i>range</i>	range specifies the number of ports to be included in the filter starting from port; filter will match if port in the IP packet lies between port and (port + range) Range value of 0 implies that only one value of the port is valid, as specified by the port
Generated by Doxygen	

8.615.2 Field Documentation

8.615.2.1 WORD Port::port

8.615.2.2 WORD Port::range

8.616 precisionDilution_s Struct Reference

Data Fields

- [ULONG PDOP](#)
- [ULONG HDOP](#)
- [ULONG VDOP](#)

8.616.1 Detailed Description

This structure contains Dilution of precision associated with this position.

Parameters

<i>PDOP</i>	<ul style="list-style-type: none"> • Position dilution of precision. • Range - 1 (highest accuracy) to 50 (lowest accuracy) • PDOP = square root of (Square of HDOP + Square of VDOP2)
<i>HDOP</i>	<ul style="list-style-type: none"> • Horizontal dilution of precision. • Range - 1 (highest accuracy) to 50 (lowest accuracy)
<i>VDOP</i>	<ul style="list-style-type: none"> • Vertical dilution of precision. • Range- 1 (highest accuracy) to 50 (lowest accuracy)

8.616.2 Field Documentation

8.616.2.1 ULONG precisionDilution_s::HDOP

8.616.2.2 ULONG precisionDilution_s::PDOP

8.616.2.3 ULONG precisionDilution_s::VDOP

8.617 PrefImageList Struct Reference

Data Fields

- [BYTE listSize](#)
- struct [ImageElement listEntries](#) [2]

8.617.1 Detailed Description

This structure contains the Preference Image List information

Parameters

<i>listSize</i>	<ul style="list-style-type: none">• The number of elements in the image list
<i>listEntries</i>	<ul style="list-style-type: none">• Array of Image entries(Max array size 2)• See ImageElement

8.617.2 Field Documentation

8.617.2.1 struct ImageElement PrefImageList::listEntries[2]

8.617.2.2 BYTE PrefImageList::listSize

8.618 prefVoiceSO Struct Reference

Data Fields

- [BYTE](#) *namID*
- [BYTE](#) *evrcCapability*
- [WORD](#) *homePageVoiceSO*
- [WORD](#) *homeOrigVoiceSO*
- [WORD](#) *roamOrigVoiceSO*

8.618.1 Detailed Description

This structure contains information about the Preferred Voice Service Options.

Parameters

<i>namID</i>	<ul style="list-style-type: none">• Index of the NAM(Number Assignment Module) to be configured.• Range 0 to 3.• Some modems support only 1 or 2 NAMs.• 0xFF,if not available.
<i>evrcCapability</i>	<ul style="list-style-type: none">• EVRC capability.• Values:<ul style="list-style-type: none">– 0x00 - Disable– 0x01 - Enable– 0xFF - Not Available

<i>homePage</i> ↔ <i>VoiceSO</i>	<ul style="list-style-type: none"> • Home page voice SO; most preferred CDMA SO to be requested from the network when receiving an incoming (MT) voice call within the home network. • Values: <ul style="list-style-type: none"> – 0x0000 - VOICE_SO_WILD - Any service option – 0x0001 - VOICE_SO_IS_96A - IS-96A – 0x0003 - VOICE_SO_EVRC - EVRC – 0x0011 - VOICE_SO_13K_IS733 - 13K_IS733 – 0x0038 - VOICE_SO_SELECTABLE_MODE_VOCODER - Selectable mode vocoder – 0x0044 - VOICE_SO_4GV_NARROW_BAND - 4GV narrowband – 0x0046 - VOICE_SO_4GV_WIDE_BAND - 4GV wideband – 0x8000 - VOICE_SO_13K - 13K – 0x8001 - VOICE_SO_IS_96 - IS-96 – 0x8023 - VOICE_SO_WVRC - WVRC – 0xFFFF - Not Available
<i>homeOrig</i> ↔ <i>VoiceSO</i>	<ul style="list-style-type: none"> • Home origination voice SO; most preferred CDMA SO to be requested from the network when receiving an incoming (MT) voice call within the home network. • Values: <ul style="list-style-type: none"> – 0x0000 - VOICE_SO_WILD - Any service option – 0x0001 - VOICE_SO_IS_96A - IS-96A – 0x0003 - VOICE_SO_EVRC - EVRC – 0x0011 - VOICE_SO_13K_IS733 - 13K_IS733 – 0x0038 - VOICE_SO_SELECTABLE_MODE_VOCODER - Selectable mode vocoder – 0x0044 - VOICE_SO_4GV_NARROW_BAND - 4GV narrowband – 0x0046 - VOICE_SO_4GV_WIDE_BAND - 4GV wideband – 0x8000 - VOICE_SO_13K - 13K – 0x8001 - VOICE_SO_IS_96 - IS-96 – 0x8023 - VOICE_SO_WVRC - WVRC – 0xFFFF - Not Available
<i>roamOrig</i> ↔ <i>VoiceSO</i>	<ul style="list-style-type: none"> • Roaming origination voice SO; most preferred CDMA SO to be requested from the network when receiving an incoming (MT) voice call within the home network. • Values: <ul style="list-style-type: none"> – 0x0000 - VOICE_SO_WILD - Any service option – 0x0001 - VOICE_SO_IS_96A - IS-96A – 0x0003 - VOICE_SO_EVRC - EVRC – 0x0011 - VOICE_SO_13K_IS733 - 13K_IS733 – 0x0038 - VOICE_SO_SELECTABLE_MODE_VOCODER - Selectable mode vocoder – 0x0044 - VOICE_SO_4GV_NARROW_BAND - 4GV narrowband – 0x0046 - VOICE_SO_4GV_WIDE_BAND - 4GV wideband – 0x8000 - VOICE_SO_13K - 13K – 0x8001 - VOICE_SO_IS_96 - IS-96 – 0x8023 - VOICE_SO_WVRC - WVRC – 0xFFFF - Not Available

8.618.2 Field Documentation

8.618.2.1 **BYTE** prefVoiceSO::evrcCapability

8.618.2.2 **WORD** prefVoiceSO::homeOrigVoiceSO

8.618.2.3 **WORD** prefVoiceSO::homePageVoiceSO

8.618.2.4 **BYTE** prefVoiceSO::namID

8.618.2.5 **WORD** prefVoiceSO::roamOrigVoiceSO

8.619 Profile3GPP Struct Reference

Data Fields

- **CHAR** * pProfilename
- **WORD** * pProfilenameSize
- **BYTE** * pPDType
- **BYTE** * pPdpHdrCompType
- **BYTE** * pPdpDataCompType
- **CHAR** * pAPNName
- **WORD** * pAPNNameSize
- **ULONG** * pPriDNSIPv4AddPref
- **ULONG** * pSecDNSIPv4AddPref
- **struct** UMTSQoS * pUMTSReqQoS
- **struct** UMTSQoS * pUMTSMinQoS
- **struct** GPRSRequestedQoS * pGPRSRequestedQoS
- **struct** GPRSRequestedQoS * pGPRSMinimumQoS
- **CHAR** * pUsername
- **WORD** * pUsernameSize
- **CHAR** * pPassword
- **WORD** * pPasswordSize
- **BYTE** * pAuthenticationPref
- **ULONG** * pIPv4AddrPref
- **BYTE** * pPcscfAddrUsingPCO
- **BYTE** * pPdpAccessConFlag
- **BYTE** * pPcscfAddrUsingDhcp
- **BYTE** * plmCnFlag
- **struct** TFTIDParams * pTFTID1Params
- **struct** TFTIDParams * pTFTID2Params
- **BYTE** * pPdpContext
- **BYTE** * pSecondaryFlag
- **BYTE** * pPrimaryID
- **USHORT** * pIPv6AddPref
- **struct** UMTSReqQoSSigInd * pUMTSReqQoSSigInd
- **struct** UMTSReqQoSSigInd * pUMTSMinQoSsigInd
- **USHORT** * pPriDNSIPv6addpref
- **USHORT** * pSecDNSIPv6addpref
- **BYTE** * pAddrAllocPref
- **struct** QoSClassID * pQoSClassID
- **BYTE** * pAPNDisabledFlag
- **ULONG** * pPDNInactivTimeout
- **BYTE** * pAPNClass

8.619.1 Detailed Description

This structure contains Input parameters of SLQSCreateProfile and SLQSModifyProfile and output parameters of SLQSGetProfileSettings

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>pProfileName</i>	<ul style="list-style-type: none"> • One or more bytes describing the profile
<i>pProfilename</i> ↔ <i>Size;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pProfileName field. Size of this parameter is 2 bytes.
<i>pPDPTType</i>	<ul style="list-style-type: none"> • Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile <ul style="list-style-type: none"> – 0x00 - PDP-IP (IPv4) – 0x01 - PDP-PPP – 0x02 - PDP-IPV6 – 0x03 - PDP-IPV4V6
<i>pPdpHdr</i> ↔ <i>CompType</i>	<ul style="list-style-type: none"> • PDP header compression type <ul style="list-style-type: none"> – 0 - PDP header compression is OFF – 1 - Manufacturer preferred compression – 2 - PDP header compression based on RFC 1144 – 3 - PDP header compression based on RFC 25074 PDP header compression based on RFC 3095
<i>pPdpData</i> ↔ <i>CompType</i>	<ul style="list-style-type: none"> • PDP data compression type <ul style="list-style-type: none"> – 0 - PDP data compression is OFF – 1 - Manufacturer preferred compression – 2 - V.42BIS data compression – 3 - V.44 data compression
<i>pAPNName</i>	<ul style="list-style-type: none"> • Access point name
<i>pAPNnameSize;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pAPNName field. Size of this parameter is 2 bytes.
<i>pPriDNSIPv4</i> ↔ <i>AddPref</i>	<ul style="list-style-type: none"> • Primary DNS IPv4 Address Preference
<i>pSecDNSIPv4</i> ↔ <i>AddPref</i>	<ul style="list-style-type: none"> • Secondary DNS IPv4 Address Preference

<i>pUMTSReqQoS</i>	<ul style="list-style-type: none"> • UMTS Requested QoS
<i>pUMTSMinQoS</i>	<ul style="list-style-type: none"> • UMTS Minimum QoS
<i>pGPRS↔ RequestedQoS</i>	<ul style="list-style-type: none"> • GPRS Minimum QoS
<i>pUsername</i>	<ul style="list-style-type: none"> • User name
<i>pUsernameSize;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pUsername field. Size of this parameter is 2 bytes.
<i>pPassword</i>	<ul style="list-style-type: none"> • Password
<i>pPasswordSize;</i>	<ul style="list-style-type: none"> • This parameter is an input parameter and should be initialised to the size of pPassword field. Size of this parameter is 2 bytes.
<i>p↔ Authentication↔ Pref</i>	<ul style="list-style-type: none"> • Authentication Preference <ul style="list-style-type: none"> – Bit map that indicates the authentication algorithm preference <ul style="list-style-type: none"> * Bit 0 - PAP preference <ul style="list-style-type: none"> • 0 - PAP is never performed • 1 - PAP may be performed * Bit 1 - CHAP preference <ul style="list-style-type: none"> • 0 - CHAP is never performed • 1 - CHAP may be performed * If more than one bit is set, then the device decides which authentication procedure is performed while setting up the data session. For example, the device may have a policy to select the most secure authentication mechanism.
<i>pIPv4AddrPref</i>	<ul style="list-style-type: none"> • IPv4 Address Preference
<i>pPcscfAddr↔ UsingPCO</i>	<ul style="list-style-type: none"> • P-CSCF Address using PCO Flag <ul style="list-style-type: none"> – 1 - (TRUE) implies request PCSCF address using PCO – 0 - (FALSE) implies do not request By default, this value is 0
<i>pPdpAccess↔ ConFlag</i>	<ul style="list-style-type: none"> • PDP access control flag <ul style="list-style-type: none"> – 0 - PDP access control none – 1 - PDP access control reject – 2 - PDP access control permission

<i>pPcscfAddr</i> ↔ <i>UsingDhcp</i>	<ul style="list-style-type: none"> • P-CSCF address using DHCP <ul style="list-style-type: none"> – 1 - (TRUE) implies Request PCSCF address using DHCP – 0 - (FALSE) implies do not request By default, value is 0
<i>pImCnFlag</i>	<ul style="list-style-type: none"> • IM CN flag <ul style="list-style-type: none"> – 1 - (TRUE) implies request IM CN flag for this profile – 0 - (FALSE) implies do not request IM CN flag for this profile
<i>pTFTID1Params</i>	<ul style="list-style-type: none"> • Traffic Flow Template
<i>pTFTID2Params</i>	<ul style="list-style-type: none"> • Traffic Flow Template
<i>pPdpContext</i>	<ul style="list-style-type: none"> • PDP context number
<i>pSecondaryFlag</i>	<ul style="list-style-type: none"> • PDP context secondary flag <ul style="list-style-type: none"> – 1 - (TRUE) implies this is secondary profile – 0 - (FALSE) implies this is not secondary profile
<i>pPrimaryID</i>	<ul style="list-style-type: none"> • PDP context primary ID • function SLQSGetProfileSettings() returns a default value 0xFF if this parameter is not returned by the device
<i>pIPv6AddPref</i>	<ul style="list-style-type: none"> • IPv6 address preference Preferred IPv6 address to be assigned to the TE; actual assigned address is negotiated with the network and may differ from this value; if not specified, the IPv6 address is obtained automatically from the network
<i>pUMTSReqQo</i> ↔ <i>SSigInd</i>	<ul style="list-style-type: none"> • UMTS requested QoS with Signalling Indication flag
<i>pUMTSMinQo</i> ↔ <i>SSigInd</i>	<ul style="list-style-type: none"> • UMTS minimum QoS with Signalling Indication flag
<i>pPrimaryDNSI</i> ↔ <i>Pv6addpref</i>	<ul style="list-style-type: none"> • Primary DNS IPv6 address preference <ul style="list-style-type: none"> – The value may be used as a preference during negotiation with the network; if not specified, the wireless device will attempt to obtain the DNS address automatically from the network; the negotiated value is provided to the host via DHCP
<i>pSecondaryD</i> ↔ <i>NSIPv6addpref</i>	<ul style="list-style-type: none"> • Secondary DNS IPv6 address preference

<i>pAddr</i> ↔ <i>AllocationPref</i>	<ul style="list-style-type: none"> DHCP/NAS preference <ul style="list-style-type: none"> This enumerated value may be used to indicate the address allocation preference <ul style="list-style-type: none"> * 0 - NAS signaling is used for address allocation * 1 - DHCP is used for address allocation
<i>pQosClassID</i>	<ul style="list-style-type: none"> 3GPP LTE QoS parameters
<i>pAPNDisabled</i> ↔ <i>Flag</i>	<ul style="list-style-type: none"> Optional 1 Byte Flag indicating if the APN is disabled/enabled If set, the profile can not be used for making data calls Any data call is failed locally Values: <ul style="list-style-type: none"> 0 - FALSE(default) 1 - True This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().
<i>pPDNInactiv</i> ↔ <i>Timeout</i>	<ul style="list-style-type: none"> Optional 4 Bytes indicating the duration of inactivity timer in seconds If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected Default value of zero indicates infinite value This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().
<i>pAPNClass</i>	<ul style="list-style-type: none"> Optional 1 Byte numeric identifier representing the APN in profile Can be set and queried but is not used by the modem This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().

8.619.2 Field Documentation

8.619.2.1 **BYTE*** Profile3GPP::pAddrAllocPref

8.619.2.2 **BYTE*** Profile3GPP::pAPNClass

8.619.2.3 **BYTE*** Profile3GPP::pAPNDisabledFlag

8.619.2.4 **CHAR*** Profile3GPP::pAPNName

8.619.2.5 **WORD*** Profile3GPP::pAPNnameSize

8.619.2.6 **BYTE*** Profile3GPP::pAuthenticationPref

8.619.2.7 struct GPRSRequestedQoS* Profile3GPP::pGPRSMinimumQoS

8.619.2.8 struct GPRSRequestedQoS* Profile3GPP::pGPRSRequestedQoS

8.619.2.9 BYTE* Profile3GPP::plmCnFlag

8.619.2.10 ULONG* Profile3GPP::plPv4AddrPref

8.619.2.11 USHORT* Profile3GPP::plPv6AddPref

8.619.2.12 CHAR* Profile3GPP::pPassword

8.619.2.13 WORD* Profile3GPP::pPasswordSize

8.619.2.14 BYTE* Profile3GPP::pPcscfAddrUsingDhcp

8.619.2.15 BYTE* Profile3GPP::pPcscfAddrUsingPCO

8.619.2.16 ULONG* Profile3GPP::pPDNInactivTimeout

8.619.2.17 BYTE* Profile3GPP::pPdpAccessConFlag

8.619.2.18 BYTE* Profile3GPP::pPdpContext

8.619.2.19 BYTE* Profile3GPP::pPdpDataCompType

8.619.2.20 BYTE* Profile3GPP::pPdpHdrCompType

8.619.2.21 BYTE* Profile3GPP::pPDType

8.619.2.22 ULONG* Profile3GPP::pPriDNSIPv4AddPref

8.619.2.23 USHORT* Profile3GPP::pPriDNSIPv6addpref

8.619.2.24 BYTE* Profile3GPP::pPrimaryID

8.619.2.25 CHAR* Profile3GPP::pProfilename

8.619.2.26 WORD* Profile3GPP::pProfilenameSize

8.619.2.27 struct QoSClassID* Profile3GPP::pQoSClassID

8.619.2.28 ULONG* Profile3GPP::pSecDNSIPv4AddPref

8.619.2.29 USHORT* Profile3GPP::pSecDNSIPv6addpref

8.619.2.30 **BYTE*** Profile3GPP::pSecondaryFlag

8.619.2.31 **struct TFTIDParams*** Profile3GPP::pTFTID1Params

8.619.2.32 **struct TFTIDParams*** Profile3GPP::pTFTID2Params

8.619.2.33 **struct UMTSQoS*** Profile3GPP::pUMTSMinQoS

8.619.2.34 **struct UMTSReqQoSSigInd*** Profile3GPP::pUMTSMinQoSsigInd

8.619.2.35 **struct UMTSQoS*** Profile3GPP::pUMTSReqQoS

8.619.2.36 **struct UMTSReqQoSSigInd*** Profile3GPP::pUMTSReqQoSsigInd

8.619.2.37 **CHAR*** Profile3GPP::pUsername

8.619.2.38 **WORD*** Profile3GPP::pUsernameSize

8.620 Profile3GPP2 Struct Reference

Data Fields

- **BYTE *** pNegoDnsSrvrPref
- **ULONG *** pPppSessCloseTimerDO
- **ULONG *** pPppSessCloseTimer1x
- **BYTE *** pAllowLinger
- **USHORT *** pLcpAckTimeout
- **USHORT *** plpcpAckTimeout
- **USHORT *** pAuthTimeout
- **BYTE *** pLcpCreqRetryCount
- **BYTE *** plpcpCreqRetryCount
- **BYTE *** pAuthRetryCount
- **BYTE *** pAuthProtocol
- **CHAR *** pUserId
- **WORD *** pUserIdSize
- **CHAR *** pAuthPassword
- **WORD *** pAuthPasswordSize
- **BYTE *** pDataRate
- **ULONG *** pAppType
- **BYTE *** pDataMode
- **BYTE *** pAppPriority
- **CHAR *** pApnString
- **WORD *** pApnStringSize
- **BYTE *** pPdnType
- **BYTE *** plsPcscfAddressNedded
- **ULONG *** pPrimaryV4DnsAddress
- **ULONG *** pSecondaryV4DnsAddress
- **USHORT *** pPriV6DnsAddress
- **USHORT *** pSecV6DnsAddress
- **BYTE *** pRATType
- **BYTE *** pAPNEnabled3GPP2
- **ULONG *** pPDNInactivTimeout3GPP2
- **BYTE *** pAPNClass3GPP2

8.620.1 Detailed Description

This structure contains the 3GPP2 profile parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>pNegoDnsSrvr</i> ↔ <i>Pref</i>	<ul style="list-style-type: none"> • Negotiate DNS Server Preference <ul style="list-style-type: none"> – 1 - (TRUE) implies request DNS addresses from the PDSN – 0 - (FALSE) implies do not request DNS addresses from the PDSN – Default value is 1 (TRUE)
<i>pPppSess</i> ↔ <i>CloseTimerDO</i>	<ul style="list-style-type: none"> • PPP Session Close Timer for DO <ul style="list-style-type: none"> – Timer value (in seconds) on DO indicating how long the PPP Session should linger before closing down
<i>pPppSess</i> ↔ <i>CloseTimer1x</i>	<ul style="list-style-type: none"> • PPP Session Close Timer for 1X <ul style="list-style-type: none"> – Timer value (in seconds) on 1X indicating how long the PPP session should linger before closing down
<i>pAllowLinger</i>	<ul style="list-style-type: none"> • Allow/disallow lingering of interface <ul style="list-style-type: none"> – 1 -(TRUE) implies allow lingering – 0 -(FALSE) implies do not allow lingering
<i>pLcpAckTimeout</i>	<ul style="list-style-type: none"> • LCP ACK Timeout <ul style="list-style-type: none"> – Value of LCP ACK Timeout in milliseconds
<i>pIpcpAck</i> ↔ <i>Timeout</i>	<ul style="list-style-type: none"> • IPCP ACK Timeout <ul style="list-style-type: none"> – Value of IPCP ACK Timeout in milliseconds
<i>pAuthTimeout</i>	<ul style="list-style-type: none"> • AUTH Timeout <ul style="list-style-type: none"> – Value of Authentication Timeout in milliseconds
<i>pLcpCreq</i> ↔ <i>RetryCount</i>	<ul style="list-style-type: none"> • LCP Configuration Request Retry Count
<i>pIpcpCreq</i> ↔ <i>RetryCount</i>	<ul style="list-style-type: none"> • IPCP Configuration Request Retry Count
<i>pAuthRetry</i> ↔ <i>Count</i>	<ul style="list-style-type: none"> • Authentication Retry Count value
<i>pAuthProtocol</i>	<ul style="list-style-type: none"> • Authentication Protocol <ul style="list-style-type: none"> – 1 - PAP – 2 - CHAP – 3 - PAP or CHAP
	Generated by Doxygen

<i>pUserId</i>	<ul style="list-style-type: none"> User ID to be used during data network authentication maximum length allowed is 127 bytes; QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.
<i>pUserIdSize;</i>	<ul style="list-style-type: none"> This parameter is an input parameter and should be initialised to the size of pUserId field. Size of this parameter is 2 bytes.
<i>pAuthPassword</i>	<ul style="list-style-type: none"> Password to be used during data network authentication; maximum length allowed is 127 bytes QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.
<i>pAuthPasswordSize;</i>	<ul style="list-style-type: none"> This parameter is an input parameter and should be initialised to the size of pAuthPassword field. Size of this parameter is 2 bytes.
<i>pDataRate</i>	<ul style="list-style-type: none"> Data Rate Requested <ul style="list-style-type: none"> 0 - Low (Low speed Service Options (SO15) only) 1 - Medium (SO33 + low R-SCH) 2 - High (SO33 + high R-SCH) Default is 2
<i>pAppType</i>	<ul style="list-style-type: none"> Application Type: <ul style="list-style-type: none"> 0x00000001 - Default Application Type 0x00000020 - LBS Application Type 0x00000040 - Tethered Application Type This parameter is not used while creating/modifying a profile
<i>pDataMode</i>	<ul style="list-style-type: none"> Data Mode to use: <ul style="list-style-type: none"> 0 - CDMA or HDR (Hybrid 1X/1xEV-DO) 1 - CDMA Only (1X only) 2 - HDR Only (1xEV-DO only) Default is 0
<i>pAppPriority</i>	<ul style="list-style-type: none"> Application Priority <ul style="list-style-type: none"> Numerical 1 byte value defining the application priority; higher value implies higher priority This parameter is not used while creating/modifying a profile
<i>pApnString</i>	<ul style="list-style-type: none"> String representing the Access Point Name maximum length allowed is 100 bytes QMI_ERR_ARG_TOO_LONG will be returned if the APN name is too long.

<i>pApnStringSize;</i>	<ul style="list-style-type: none"> This parameter is an input parameter and should be initialised to the size of pApnString field. Size of this parameter is 2 bytes.
<i>pPdnType</i>	<ul style="list-style-type: none"> Packed Data Network Type Requested: <ul style="list-style-type: none"> 0 - IPv4 PDN Type 1 - IPv6 PDN Type 2 - IPv4 or IPv6 PDN Type 3 - Unspecified PDN Type (implying no preference)
<i>pIsPcscf↔ AddressNedded</i>	<ul style="list-style-type: none"> This boolean value is used to control if PCSCF address is requested from PDSN <ul style="list-style-type: none"> 1 -(TRUE) implies request for PCSCF value from the PDSN 0 -(FALSE) implies do not request for PCSCF value from the PDSN
<i>pPrimaryV4↔ DnsAddress</i>	<ul style="list-style-type: none"> IPv4 Primary DNS address <ul style="list-style-type: none"> The Primary IPv4 DNS address that can be statically assigned to the UE
<i>pSecondaryV4↔ DnsAddress</i>	<ul style="list-style-type: none"> IPv4 Secondary DNS address <ul style="list-style-type: none"> The Secondary IPv4 DNS address that can be statically assigned to the UE
<i>pPriV6Dns↔ Address</i>	<ul style="list-style-type: none"> Primary IPv6 DNS address <ul style="list-style-type: none"> The Primary IPv6 DNS address that can be statically assigned to the UE
<i>pSecV6Dns↔ Address</i>	<ul style="list-style-type: none"> Secondary IPv6 DNS address <ul style="list-style-type: none"> The Secondary IPv6 DNS address that can be statically assigned to the UE
<i>pRATType</i>	<ul style="list-style-type: none"> Optional 1 Byte Flag indicating RAT Type Values: <ul style="list-style-type: none"> 1 - HRPD 2 - EHRPD 3 - HRPD_EHRPD This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().
<i>pAPN↔ Enabled3GPP2</i>	<ul style="list-style-type: none"> Optional 1 Byte Flag indicating if the APN is disabled/enabled If disabled, the profile can not be used for making data calls Values: <ul style="list-style-type: none"> 0 - Disabled 1 - Enabled(default value) This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().

<i>pPDNInactiv↔ Timeout3GPP2</i>	<ul style="list-style-type: none"> • Optional 4 Bytes indicating the duration of inactivity timer in seconds • If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected • Default value of zero indicates infinite value • This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().
<i>pAPNClass3G↔ PP2</i>	<ul style="list-style-type: none"> • Optional 1 Byte numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • This parameter is currently read only and can be read by using the function SLQSGetProfile↔Settings().

8.620.2 Field Documentation

8.620.2.1 **BYTE*** Profile3GPP2::pAllowLinger

8.620.2.2 **BYTE*** Profile3GPP2::pAPNClass3GPP2

8.620.2.3 **BYTE*** Profile3GPP2::pAPNEnabled3GPP2

8.620.2.4 **CHAR*** Profile3GPP2::pApnString

8.620.2.5 **WORD*** Profile3GPP2::pApnStringSize

8.620.2.6 **BYTE*** Profile3GPP2::pAppPriority

8.620.2.7 **ULONG*** Profile3GPP2::pAppType

8.620.2.8 **CHAR*** Profile3GPP2::pAuthPassword

8.620.2.9 **WORD*** Profile3GPP2::pAuthPasswordSize

8.620.2.10 **BYTE*** Profile3GPP2::pAuthProtocol

8.620.2.11 **BYTE*** Profile3GPP2::pAuthRetryCount

8.620.2.12 **USHORT*** Profile3GPP2::pAuthTimeout

8.620.2.13 **BYTE*** Profile3GPP2::pDataMode

8.620.2.14 **BYTE*** Profile3GPP2::pDataRate

8.620.2.15 **USHORT*** Profile3GPP2::pIpcpAckTimeout

- 8.620.2.16 **BYTE*** Profile3GPP2::plpcpCreqRetryCount
- 8.620.2.17 **BYTE*** Profile3GPP2::plsPcscfAddressNedded
- 8.620.2.18 **USHORT*** Profile3GPP2::pLcpAckTimeout
- 8.620.2.19 **BYTE*** Profile3GPP2::pLcpCreqRetryCount
- 8.620.2.20 **BYTE*** Profile3GPP2::pNegoDnsSrvrPref
- 8.620.2.21 **ULONG*** Profile3GPP2::pPDNInactivTimeout3GPP2
- 8.620.2.22 **BYTE*** Profile3GPP2::pPdnType
- 8.620.2.23 **ULONG*** Profile3GPP2::pPppSessCloseTimer1x
- 8.620.2.24 **ULONG*** Profile3GPP2::pPppSessCloseTimerDO
- 8.620.2.25 **ULONG*** Profile3GPP2::pPrimaryV4DnsAddress
- 8.620.2.26 **USHORT*** Profile3GPP2::pPriV6DnsAddress
- 8.620.2.27 **BYTE*** Profile3GPP2::pRATType
- 8.620.2.28 **ULONG*** Profile3GPP2::pSecondaryV4DnsAddress
- 8.620.2.29 **USHORT*** Profile3GPP2::pSecV6DnsAddress
- 8.620.2.30 **CHAR*** Profile3GPP2::pUserId
- 8.620.2.31 **WORD*** Profile3GPP2::pUserIdSize

8.621 ProfileIdentifier Struct Reference

Data Fields

- [BYTE](#) `profileType`
- [BYTE](#) `profileIndex`

8.621.1 Detailed Description

This structure contains the Profile Identifier Information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>profileType</i>	<ul style="list-style-type: none">Identifies the type of profile 0x00 = 3GPP
<i>profileIndex</i>	<ul style="list-style-type: none">Index of profile whose settings were loaded prior to session parameter negotiation for the current call. If this TLV is not present, data call parameters are based on device default settings for each parameter

8.621.2 Field Documentation

8.621.2.1 BYTE ProfileIdentifier::profileIndex

8.621.2.2 BYTE ProfileIdentifier::profileType

8.622 protocolSubtypeElement Struct Reference

Data Fields

- [WORD PhysicalLayer](#)
- [WORD ControlMac](#)
- [WORD AccessMac](#)
- [WORD ForwardMac](#)
- [WORD ReverseMac](#)
- [WORD KeyExchange](#)
- [WORD AuthProt](#)
- [WORD EncryptProt](#)
- [WORD SecProt](#)
- [WORD IdleState](#)
- [WORD MultDisc](#)
- [WORD VirtStream](#)

8.622.1 Detailed Description

This structure contains Protocol Subtype Elements for Protocol Subtype List

Parameters

<i>PhysicalLayer</i>	<ul style="list-style-type: none">Specifies Physical Layer Protocol subtype
<i>ControlMac</i>	<ul style="list-style-type: none">Specifies Control Channel MAC Protocol subtype
<i>AccessMac</i>	<ul style="list-style-type: none">Specifies Access Channel MAC Protocol subtype

<i>ForwardMac</i>	<ul style="list-style-type: none"> • Specifies Forward Traffic Channel MAC Protocol subtype
<i>ReverseMac</i>	<ul style="list-style-type: none"> • Specifies Reverse Traffic Channel MAC Protocol subtype
<i>KeyExchange</i>	<ul style="list-style-type: none"> • Specifies Key exchange Protocol subtype
<i>AuthProt</i>	<ul style="list-style-type: none"> • Specifies Authentication Protocol subtype
<i>EncryptProt</i>	<ul style="list-style-type: none"> • Specifies Encryption Protocol subtype
<i>SecProt</i>	<ul style="list-style-type: none"> • Specifies Security Protocol subtype
<i>IdleState</i>	<ul style="list-style-type: none"> • Specifies Idle state Protocol subtype
<i>MultDisc</i>	<ul style="list-style-type: none"> • Specifies Generic multimode capability discovery Protocol subtype
<i>VirtStream</i>	<ul style="list-style-type: none"> • Specifies Generic Virtual Stream Protocol subtype

8.622.2 Field Documentation

8.622.2.1 WORD protocolSubtypeElement::AccessMac

8.622.2.2 WORD protocolSubtypeElement::AuthProt

8.622.2.3 WORD protocolSubtypeElement::ControlMac

8.622.2.4 WORD protocolSubtypeElement::EncryptProt

8.622.2.5 WORD protocolSubtypeElement::ForwardMac

8.622.2.6 WORD protocolSubtypeElement::IdleState

8.622.2.7 WORD protocolSubtypeElement::KeyExchange

8.622.2.8 WORD protocolSubtypeElement::MultDisc

8.622.2.9 WORD protocolSubtypeElement::PhysicalLayer

8.622.2.10 WORD protocolSubtypeElement::ReverseMac

8.622.2.11 WORD protocolSubtypeElement::SecProt

8.622.2.12 WORD protocolSubtypeElement::VirtStream

8.623 PSDetachReq Struct Reference

Data Fields

- [BYTE](#) * [pDetachAction](#)

8.623.1 Detailed Description

This structure contains information about the SLQSSwiPSDetach request parameters.

Parameters

<i>pDetach↔ Action[IN]</i>	<ul style="list-style-type: none">• Values<ul style="list-style-type: none">– 2- Initiates an immediate packet domain detach.
--------------------------------	---

8.623.2 Field Documentation

8.623.2.1 [BYTE](#)* PSDetachReq::pDetachAction

8.624 qaQmi3Gpp2TimeZone Struct Reference

Data Fields

- [BYTE](#) [leapSeconds](#)
- [BYTE](#) [localTimeOffset](#)
- [BYTE](#) [daylightSavings](#)

8.624.1 Detailed Description

This structure contains the 3GPP2TimeZone parameters

Parameters

<i>leapSeconds</i>	<ul style="list-style-type: none"> • leap seconds - Number of leap seconds since the start of CDMA system time.
<i>localTimeOffset</i>	<ul style="list-style-type: none"> • Local Time Offset - Offset of system time in units of 30 minutes; the value in this field conveys as 8 bit 2's compliment number.
<i>daylightSavings</i>	<ul style="list-style-type: none"> • Day Light Savings Indicator <ul style="list-style-type: none"> – 0x00 - OFF (daylight savings not in effect) – 0x01 - ON (daylight savings in effect)

8.624.2 Field Documentation

8.624.2.1 BYTE qaQmi3Gpp2TimeZone::daylightSavings

8.624.2.2 BYTE qaQmi3Gpp2TimeZone::leapSeconds

8.624.2.3 BYTE qaQmi3Gpp2TimeZone::localTimeOffset

8.625 qaQmiInterfaceInfo Struct Reference

Data Fields

- [BYTE qaQmiinstanceid](#)
- [eQaQMIService qaQmisvctype](#)
- [ULONG v4sessionId](#)
- [ULONG v6sessionId](#)

8.625.1 Detailed Description

Structure used to store the service, interface and session information

Parameters

<i>qaQmiinstanceid</i>	<ul style="list-style-type: none"> • The interface instance ID <ul style="list-style-type: none"> – 0x00 - PDP instance ID 0 – 0x01 - PDP instance ID 1 – 0x02 - PDP instance ID 2
<i>qaQmisvctype</i>	<ul style="list-style-type: none"> • The service type information. See eQaQMIService for more information
<i>v4sessionId</i>	<ul style="list-style-type: none"> • IPv4 QMI client session handle
<i>v6sessionId</i>	<ul style="list-style-type: none"> • IPv6 QMI client session handle

8.625.2 Field Documentation

8.625.2.1 **BYTE** qaQmiInterfaceInfo::qaQmiinstanceid

8.625.2.2 **eQaQMIService** qaQmiInterfaceInfo::qaQmisvctype

8.625.2.3 **ULONG** qaQmiInterfaceInfo::v4sessionid

8.625.2.4 **ULONG** qaQmiInterfaceInfo::v6sessionid

8.626 qaQmiServingSystemParam Struct Reference

Data Fields

- [servSystem](#) [ServingSystem](#)
- [BYTE](#) [roamIndicatorVal](#)
- [dataSrvCapabilities](#) [DataSrvCapabilities](#)
- [currentPLMN](#) [CurrentPLMN](#)
- [WORD](#) [SystemID](#)
- [WORD](#) [NetworkID](#)
- [WORD](#) [BasestationID](#)
- [ULONG](#) [BasestationLatitude](#)
- [ULONG](#) [BasestationLongitude](#)
- [roamIndList](#) [RoamingIndicatorList](#)
- [BYTE](#) [defaultRoamInd](#)
- [qaQmi3Gpp2TimeZone](#) [Gpp2TimeZone](#)
- [BYTE](#) [CDMA_P_Rev](#)
- [BYTE](#) [GppTimeZone](#)
- [BYTE](#) [GppNetworkDSTAdjustment](#)
- [WORD](#) [Lac](#)
- [ULONG](#) [CellID](#)
- [BYTE](#) [concSvcInfo](#)
- [BYTE](#) [PRLInd](#)
- [BYTE](#) [DTMInd](#)
- [detailSvcInfo](#) [DetailedSvcInfo](#)
- [CDMASysInfoExt](#) [CDMASystemInfoExt](#)
- [BYTE](#) [hdrPersonality](#)
- [WORD](#) [trackAreaCode](#)
- [callBarStatus](#) [CallBarStatus](#)

8.626.1 Detailed Description

This structure contains the Serving System parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>ServingSystem</i>	<ul style="list-style-type: none"> • Serving System • See servSystem for more information
<i>roamIndicatorVal</i>	<ul style="list-style-type: none"> • Optional parameter indicating Roaming Indicator value • Values: <ul style="list-style-type: none"> – 0x00 - Roaming – 0x01 - Home – 0x02 - Flashing – 0x03 and above - Operator defined values
<i>DataSrv↔ Capabilities</i>	<ul style="list-style-type: none"> • Optional parameter indicating Data services capability • See dataSrvCapabilities for more information
<i>CurrentPLMN</i>	<ul style="list-style-type: none"> • Optional parameter indicating Current PLMN • See currentPLMN for more information
<i>SystemID</i>	<ul style="list-style-type: none"> • Optional parameter indicating System ID
<i>NetworkID</i>	<ul style="list-style-type: none"> • Optional parameter indicating Network ID
<i>BaseStationID</i>	<ul style="list-style-type: none"> • Optional parameter indicating Base Station Identification Number
<i>BaseStation↔ Latitude</i>	<ul style="list-style-type: none"> • Optional parameter indicating Base station latitude in units of 0.25 sec, expressed as a two's complement signed number with positive numbers signifying North latitude
<i>Basestation↔ Longitude</i>	<ul style="list-style-type: none"> • Optional parameter indicating Base station longitude in units of 0.25 sec, expressed as a Two's complement signed number with positive numbers signifying East longitude
<i>Roaming↔ IndicatorList</i>	<ul style="list-style-type: none"> • Optional parameter indicating Roaming Indicator List • See roamIndList for more information
<i>defaultRoamInd</i>	<ul style="list-style-type: none"> • Optional parameter indicating Default Roaming Indicator • Values: <ul style="list-style-type: none"> – 0x00 - Roaming – 0x01 - Home
<i>Gpp2TimeZone</i>	<ul style="list-style-type: none"> • Optional parameter indicating 3GPP2 Time Zone • See qaQmi3Gpp2TimeZone for more information

<i>CDMA_P_Rev</i>	<ul style="list-style-type: none"> Optional parameter indicating CDMA P_Rev in use
<i>GppTimeZone</i>	<ul style="list-style-type: none"> Optional parameter indicating Offset from Universal time, i.e., difference between local time and Universal time, in increments of 15 min. (signed value).
<i>GppNetworkD↔STAdjustment</i>	<ul style="list-style-type: none"> Optional parameter indicating 3GPP network daylight saving adjustment Values: <ul style="list-style-type: none"> 0x00 - No adjustment for Daylight Saving Time 0x01 - 1 hr adjustment for Daylight Saving Time 0x02 - 2 hr adjustment for Daylight Saving Time
<i>Lac</i>	<ul style="list-style-type: none"> Optional parameter indicating 3GPP Location Area Code
<i>CellID</i>	<ul style="list-style-type: none"> Optional parameter indicating 3GPP Cell ID
<i>concSvcInfo</i>	<ul style="list-style-type: none"> Optional parameter indicating 3GPP2 concurrent service Info Values: <ul style="list-style-type: none"> 0x00 - Concurrent service not available 0x01 - Concurrent service available
<i>PRLInd</i>	<ul style="list-style-type: none"> Optional parameter indicating 3GPP2 PRL Indicator Values: <ul style="list-style-type: none"> 0x00 - System not in PRL 0x01 - System is in PRL
<i>DTMInd</i>	<ul style="list-style-type: none"> Optional parameter indicating Dual Transfer Mode Indication(GSM Only) Values: <ul style="list-style-type: none"> 0x00 - DTM not supported 0x01 - DTM supported
<i>DetailedSvcInfo</i>	<ul style="list-style-type: none"> Optional parameter indicating Detailed service information See detailSvcInfo for more information
<i>CDMASystem↔InfoExt</i>	<ul style="list-style-type: none"> Optional parameter indicating CDMA System Info Ext See CDMASysInfoExt for more information

<i>hdrPersonality</i>	<ul style="list-style-type: none"> • Optional parameter indicating HDR Personality Information • Values: <ul style="list-style-type: none"> – 0x00 - Unknown – 0x01 - HRPD – 0x02 - eHRPD
<i>trackAreaCode</i>	<ul style="list-style-type: none"> • Optional parameter indicating Tracking area code information for LTE
<i>CallBarStatus</i>	<ul style="list-style-type: none"> • Optional parameter indicating Call Barring Status • See callBarStatus for more information

8.626.2 Field Documentation

8.626.2.1 **WORD** `qaQmiServingSystemParam::BasestationID`

8.626.2.2 **ULONG** `qaQmiServingSystemParam::BasestationLatitude`

8.626.2.3 **ULONG** `qaQmiServingSystemParam::BasestationLongitude`

8.626.2.4 **callBarStatus** `qaQmiServingSystemParam::CallBarStatus`

8.626.2.5 **BYTE** `qaQmiServingSystemParam::CDMA_P_Rev`

8.626.2.6 **CDMASysInfoExt** `qaQmiServingSystemParam::CDMASystemInfoExt`

8.626.2.7 **ULONG** `qaQmiServingSystemParam::CellID`

8.626.2.8 **BYTE** `qaQmiServingSystemParam::concSvcInfo`

8.626.2.9 **currentPLMN** `qaQmiServingSystemParam::CurrentPLMN`

8.626.2.10 **dataSrvCapabilities** `qaQmiServingSystemParam::DataSrvCapabilities`

8.626.2.11 **BYTE** `qaQmiServingSystemParam::defaultRoamInd`

8.626.2.12 **detailSvcInfo** `qaQmiServingSystemParam::DetailedSvcInfo`

8.626.2.13 **BYTE** `qaQmiServingSystemParam::DTMInd`

8.626.2.14 **qaQmi3Gpp2TimeZone** `qaQmiServingSystemParam::Gpp2TimeZone`

8.626.2.15 **BYTE** `qaQmiServingSystemParam::GppNetworkDSTAdjustment`

- 8.626.2.16 **BYTE** qaQmiServingSystemParam::GppTimeZone
- 8.626.2.17 **BYTE** qaQmiServingSystemParam::hdrPersonality
- 8.626.2.18 **WORD** qaQmiServingSystemParam::Lac
- 8.626.2.19 **WORD** qaQmiServingSystemParam::NetworkID
- 8.626.2.20 **BYTE** qaQmiServingSystemParam::PRLInd
- 8.626.2.21 **BYTE** qaQmiServingSystemParam::roamIndicatorVal
- 8.626.2.22 **roamIndList** qaQmiServingSystemParam::RoamingIndicatorList
- 8.626.2.23 **servSystem** qaQmiServingSystemParam::ServingSystem
- 8.626.2.24 **WORD** qaQmiServingSystemParam::SystemID
- 8.626.2.25 **WORD** qaQmiServingSystemParam::trackAreaCode

8.627 QmiCbkCatEventStatusReportInd Struct Reference

Data Fields

- [BYTE event_Index](#)
- struct [CatCommonEventTlv CCETlv](#) [11]

8.627.1 Field Documentation

- 8.627.1.1 struct [CatCommonEventTlv](#) QmiCbkCatEventStatusReportInd::CCETlv[11]
- 8.627.1.2 **BYTE** QmiCbkCatEventStatusReportInd::event_Index

8.628 QmiCbkLocBestAvailPosInd Struct Reference

Data Fields

- [ULONG status](#)
- [ULONG * pXid](#)
- [ULONGLONG * pLatitude](#)
- [ULONGLONG * pLongitude](#)
- [ULONG * pHorUncCircular](#)
- [ULONG * pAltitudeWrtEllipsoid](#)
- [ULONG * pVertUnc](#)
- [ULONGLONG * pTimestampUtc](#)
- [ULONG * pTimeUnc](#)

- ULONG * pHorUncEllipseSemiMinor
- ULONG * pHorUncEllipseSemiMajor
- ULONG * pHorUncEllipseOrientAzimuth
- BYTE * pHorCirConf
- BYTE * pHorEllpConf
- ULONG * pHorReliability
- ULONG * pSpeedHorizontal
- ULONG * pSpeedUnc
- ULONG * pAltitudeWrtMeanSeaLevel
- BYTE * pVertConfidence
- ULONG * pVertReliability
- ULONG * pSpeedVertical
- ULONG * pSpeedVerticalUnc
- ULONG * pHeading
- ULONG * pHeadingUnc
- ULONG * pMagneticDeviation
- ULONG * pTechnologyMask
- precisionDilution * pPrecisionDilution
- gpsTime * pGpsTime
- ULONG * pTimeSrc
- sensorDataUsage * pSensorDataUsage
- svUsedforFix * pSvUsedforFix

8.628.1 Detailed Description

This structure contains Best Available Position

Parameters

<i>status</i>	<ul style="list-style-type: none"> • Valid values: <ul style="list-style-type: none"> – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure
<i>xid</i>	Transaction ID that was specified in the Get Best Available Position request.

<i>pLatitude</i>	<ul style="list-style-type: none"> • Latitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -90.0 to 90.0 • Positive values indicate northern latitude • Negative values indicate southern latitude
<i>pLongitude</i>	<ul style="list-style-type: none"> • Longitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -180.0 to 180.0 • Positive values indicate eastern latitude • Negative values indicate western latitude
<i>pHorUncCircular</i>	<ul style="list-style-type: none"> • Horizontal position uncertainty. • Units - Meters
<i>pAltitudeWrt↔ Ellipsoid</i>	<ul style="list-style-type: none"> • Altitude With Respect to WGS84 Ellipsoid. • Units - Meters • Range -500 to 15883
<i>pVertUnc</i>	<ul style="list-style-type: none"> • Vertical uncertainty. • Units - Meters
<i>pTimestampUtc</i>	<ul style="list-style-type: none"> • UTC timestamp • Units - Milliseconds since Jan. 1, 1970
<i>pTimeUnc</i>	<ul style="list-style-type: none"> • Time uncertainty. • Units - Milliseconds
<i>pHorUnc↔ EllipseSemi↔ Minor</i>	<ul style="list-style-type: none"> • Semi-minor axis of horizontal elliptical uncertainty. • Units - Meters
<i>pHorUnc↔ EllipseSemi↔ Major</i>	<ul style="list-style-type: none"> • Semi-major axis of horizontal elliptical uncertainty. • Units: Meters
<i>pHorUnc↔ EllipseOrient↔ Azimuth</i>	<ul style="list-style-type: none"> • Elliptical horizontal uncertainty azimuth of orientation. • Units - Decimal degrees • Range - 0 to 180

<i>pHorCirConf</i>	<ul style="list-style-type: none"> • Horizontal circular uncertainty confidence • Units: Precent • Range: 0 to 99
<i>pHorEllpConf</i>	<ul style="list-style-type: none"> • Horizontal elliptical uncertainty confidence • Units: Precent • Range: 0 to 99
<i>pHorReliability</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk – 2 - Location reliability is low; little or no cross-checking is possible. – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed
<i>pSpeed</i> ↔ <i>Horizontal</i>	<ul style="list-style-type: none"> • Horizontal speed. • Units - Meters/second
<i>pSpeedUnc</i>	<ul style="list-style-type: none"> • 3-D Speed uncertainty. • Units - Meters/second.
<i>pAltitudeWrt</i> ↔ <i>MeanSeaLevel</i>	<ul style="list-style-type: none"> • Altitude With Respect to Sea Level. • Units - Meters
<i>pVertConfidence</i>	<ul style="list-style-type: none"> • Vertical uncertainty confidence. • Units - Percentage • Range 0 to 99
<i>pVertReliability</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk. – 2 - Location reliability is low; little or no cross-checking is possible – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed
<i>pSpeedVertical</i>	<ul style="list-style-type: none"> • Vertical speed. • Units - Meters/second
<i>pSpeed</i> ↔ <i>VerticalUnc</i>	<ul style="list-style-type: none"> • Vertical speed • Units: Meters/second

<i>pHeading</i>	<ul style="list-style-type: none"> • Heading. • Units - Degree • Range 0 to 359.999
<i>pHeadingUnc</i>	<ul style="list-style-type: none"> • Heading uncertainty. • Units - Degree • Range 0 to 359.999
<i>pMagnetic↔ Deviation</i>	<ul style="list-style-type: none"> • Difference between the bearing to true north and the bearing shown on a magnetic compass. The deviation is positive when the magnetic north is east of true north.
<i>pTechnology↔ Mask</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0x00000001 - Satellites were used to generate the fix – 0x00000002 - Cell towers were used to generate the fix – 0x00000004 - Wi-Fi access points were used to generate the fix – 0x00000008 - Sensors were used to generate the fix – 0x00000010 - Reference Location was used to generate the fix – 0x00000020 - Coarse position injected into the location engine was used to generate the fix – 0x00000040 - AFLT was used to generate the fix – 0x00000080 - GNSS and network-provided measurements were used to generate the fix
<i>-pPrecision↔ Dilution</i>	<ul style="list-style-type: none"> • See precisionDilution for more information
<i>-pGpsTime</i>	<ul style="list-style-type: none"> • See gpsTime for more information

<i>pTimeSrc</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Invalid time. – 1 - Time is set by the 1X system. – 2 - Time is set by WCDMA/GSM time tagging. – 3 - Time is set by an external injection. – 4 - Time is set after decoding over-the-air GPS navigation data from one GPS satellite. – 5 - Time is set after decoding over-the-air GPS navigation data from multiple satellites. – 6 - Both time of the week and the GPS week number are known. – 7 - Time is set by the position engine after the fix is obtained – 8 - Time is set by the position engine after performing SFT, this is done when the clock time uncertainty is large. – 9 - Time is set after decoding GLO satellites. – 10- Time is set after transforming the GPS to GLO time – 11- Time is set by the sleep time tag provided by the WCDMA network. – 12- Time is set by the sleep time tag provided by the GSM network – 13- Source of the time is unknown – 14- Time is derived from the system clock (better known as the slow clock); GNSS time is maintained irrespective of the GNSS receiver state – 15- Time is set after decoding QZSS satellites. – 16- Time is set after decoding BDS satellites.
<i>-pSensorDataUsage</i>	<ul style="list-style-type: none"> • See sensorDataUsage for more information
<i>-pSvUsedforFix</i>	<ul style="list-style-type: none"> • See svUsedforFix for more information

8.628.2 Field Documentation

8.628.2.1 **ULONG*** QmiCbkLocBestAvailPosInd::pAltitudeWrtEllipsoid

8.628.2.2 **ULONG*** QmiCbkLocBestAvailPosInd::pAltitudeWrtMeanSeaLevel

8.628.2.3 **gpsTime*** QmiCbkLocBestAvailPosInd::pGpsTime

8.628.2.4 **ULONG*** QmiCbkLocBestAvailPosInd::pHeading

8.628.2.5 **ULONG*** QmiCbkLocBestAvailPosInd::pHeadingUnc

8.628.2.6 **BYTE*** QmiCbkLocBestAvailPosInd::pHorCirConf

8.628.2.7 **BYTE*** QmiCbkLocBestAvailPosInd::pHorEllpConf

8.628.2.8 **ULONG*** QmiCbkLocBestAvailPosInd::pHorReliability

- 8.628.2.9 **ULONG*** QmiCbkLocBestAvailPosInd::pHorUncCircular
- 8.628.2.10 **ULONG*** QmiCbkLocBestAvailPosInd::pHorUncEllipseOrientAzimuth
- 8.628.2.11 **ULONG*** QmiCbkLocBestAvailPosInd::pHorUncEllipseSemiMajor
- 8.628.2.12 **ULONG*** QmiCbkLocBestAvailPosInd::pHorUncEllipseSemiMinor
- 8.628.2.13 **ULONGLONG*** QmiCbkLocBestAvailPosInd::pLatitude
- 8.628.2.14 **ULONGLONG*** QmiCbkLocBestAvailPosInd::pLongitude
- 8.628.2.15 **ULONG*** QmiCbkLocBestAvailPosInd::pMagneticDeviation
- 8.628.2.16 **precisionDilution*** QmiCbkLocBestAvailPosInd::pPrecisionDilution
- 8.628.2.17 **sensorDataUsage*** QmiCbkLocBestAvailPosInd::pSensorDataUsage
- 8.628.2.18 **ULONG*** QmiCbkLocBestAvailPosInd::pSpeedHorizontal
- 8.628.2.19 **ULONG*** QmiCbkLocBestAvailPosInd::pSpeedUnc
- 8.628.2.20 **ULONG*** QmiCbkLocBestAvailPosInd::pSpeedVertical
- 8.628.2.21 **ULONG*** QmiCbkLocBestAvailPosInd::pSpeedVerticalUnc
- 8.628.2.22 **svUsedforFix*** QmiCbkLocBestAvailPosInd::pSvUsedforFix
- 8.628.2.23 **ULONG*** QmiCbkLocBestAvailPosInd::pTechnologyMask
- 8.628.2.24 **ULONG*** QmiCbkLocBestAvailPosInd::pTimeSrc
- 8.628.2.25 **ULONGLONG*** QmiCbkLocBestAvailPosInd::pTimestampUtc
- 8.628.2.26 **ULONG*** QmiCbkLocBestAvailPosInd::pTimeUnc
- 8.628.2.27 **BYTE*** QmiCbkLocBestAvailPosInd::pVertConfidence
- 8.628.2.28 **ULONG*** QmiCbkLocBestAvailPosInd::pVertReliability
- 8.628.2.29 **ULONG*** QmiCbkLocBestAvailPosInd::pVertUnc
- 8.628.2.30 **ULONG*** QmiCbkLocBestAvailPosInd::pXid
- 8.628.2.31 **ULONG** QmiCbkLocBestAvailPosInd::status

8.629 QmiCbkLocCradleMountInd Struct Reference

Data Fields

- [ULONG](#) `cradleMountConfigStatus`

8.629.1 Detailed Description

This structure contains LOC Cradle Mount Config Status

Parameters

<i>cradleMount↔ ConfigStatus</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Request was completed successfully – 1 - Request failed because of a general failure. – 2 - Request failed because it is not supported. – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - engine could not allocate sufficient memory – 9 - Request failed because the maximum number of Geofences are already programmed – 10 -Location service failed because of an XTRA version-based file format check failure
--------------------------------------	---

8.629.2 Field Documentation

8.629.2.1 **ULONG** QmiCbkLocCradleMountInd::cradleMountConfigStatus

8.630 QmiCbkLocEngineStateInd Struct Reference

Data Fields

- [ULONG engineState](#)

8.630.1 Detailed Description

This structure contains LOC Engine State field.

Parameters

<i>engineState</i>	<ul style="list-style-type: none"> • Location engine state. • Valid values <ul style="list-style-type: none"> – 1 - Location engine is on – 2 - Location engine is off
--------------------	---

8.630.2 Field Documentation

8.630.2.1 **ULONG** QmiCbkLocEngineStateInd::engineState

8.631 QmiCbkLocEventTimeSyncInd Struct Reference

Data Fields

- [ULONG timeSyncRefCounter](#)

8.631.1 Detailed Description

This structure contains LOC Event Time Sync Reference COUNTER

Parameters

<i>timeSyncRefCounter</i>	<ul style="list-style-type: none">• Sent by the location engine when it needs to synchronize location engine and control point (sensor processor) times.
---------------------------	--

8.631.2 Field Documentation

8.631.2.1 [ULONG QmiCbkLocEventTimeSyncInd::timeSyncRefCounter](#)

8.632 QmiCbkLocInjectPositionInd Struct Reference

Data Fields

- [ULONG status](#)

8.632.1 Detailed Description

Contain the parameters passed for SetLocInjectPositionCallback by the device.

Parameters

<i>status</i>	<ul style="list-style-type: none"> • UTC Position Injection Status • Valid values: <ul style="list-style-type: none"> – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure
---------------	--

Note

None

8.632.2 Field Documentation

8.632.2.1 ULONG QmiCbkLocInjectPositionInd::status

8.633 QmiCbkLocInjectSensorDataInd Struct Reference

Data Fields

- [ULONG injectSensorDataStatus](#)
- [ULONG * pOpaqueIdentifier](#)
- [BYTE * pAccelSamplesAccepted](#)
- [BYTE * pGyroSamplesAccepted](#)
- [BYTE * pAccelTempSamplesAccepted](#)
- [BYTE * pGyroTempSamplesAccepted](#)

8.633.1 Detailed Description

This structure contains LOC Inject Sensor Data

Parameters

<i>injectSensor↔ DataStatus</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Request was completed successfully – 1 - Request failed because of a general failure. – 2 - Request failed because it is not supported. – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - engine could not allocate sufficient memory – 9 - Request failed because the maximum number of Geofences are already programmed – 10 -Location service failed because of an XTRA version-based file format check failure
<i>pOpaque↔ Identifier</i>	<ul style="list-style-type: none"> • Sent in by the client echoed so the client can relate the indication to the request.
<i>pAccel↔ Samples↔ Accepted</i>	<ul style="list-style-type: none"> • Lets the client know how many 3-axis accelerometer samples were accepted. • This field is present only if the accelerometer samples were sent in the request.
<i>pGyro↔ Samples↔ Accepted</i>	<ul style="list-style-type: none"> • Lets the client know how many 3-axis gyroscope samples were accepted. • This field is present only if the gyroscope samples were sent in the request.
<i>pAccelTemp↔ Samples↔ Accepted</i>	<ul style="list-style-type: none"> • Lets the client know how many accelerometer temperature samples were accepted. • This field is present only if the accelerometer temperature samples were sent in the request.
<i>pGyroTemp↔ Samples↔ Accepted</i>	<ul style="list-style-type: none"> • Lets the client know how many gyroscope temperature samples were accepted. • This field is present only if the gyroscope temperature samples were sent in the request.

8.633.2 Field Documentation

8.633.2.1 **ULONG** QmiCbkLocInjectSensorDataInd::injectSensorDataStatus8.633.2.2 **BYTE*** QmiCbkLocInjectSensorDataInd::pAccelSamplesAccepted8.633.2.3 **BYTE*** QmiCbkLocInjectSensorDataInd::pAccelTempSamplesAccepted8.633.2.4 **BYTE*** QmiCbkLocInjectSensorDataInd::pGyroSamplesAccepted8.633.2.5 **BYTE*** QmiCbkLocInjectSensorDataInd::pGyroTempSamplesAccepted8.633.2.6 **ULONG*** QmiCbkLocInjectSensorDataInd::pOpaqueIdentifier

8.634 QmiCbkLocInjectTimeInd Struct Reference

Data Fields

- [ULONG injectTimeSyncStatus](#)

8.634.1 Detailed Description

This structure contains LOC Inject Time Sync Data Status

Parameters

<i>injectTime</i> ↔ <i>SyncStatus</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Request was completed successfully – 1 - Request failed because of a general failure. – 2 - Request failed because it is not supported. – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - engine could not allocate sufficient memory – 9 - Request failed because the maximum number of Geofences are already programmed – 10 -Location service failed because of an XTRA version-based file format check failure
--	---

8.634.2 Field Documentation

8.634.2.1 **ULONG** QmiCbkLocInjectTimeInd::injectTimeSyncStatus

8.635 QmiCbkLocInjectUTCTimeInd Struct Reference

Data Fields

- [ULONG status](#)

8.635.1 Detailed Description

Contain the parameters passed for SetLocInjectUTCTimeCallback by the device.

Parameters

<i>status</i>	<ul style="list-style-type: none"> • Status of the UTC Time Injection request • Valid values: <ul style="list-style-type: none"> – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out
---------------	--

Note

None

8.635.2 Field Documentation

8.635.2.1 ULONG QmiCbkLocInjectUTCTimeInd::status

8.636 QmiCbkLocPositionReportInd Struct Reference

Data Fields

- ULONG sessionStatus
- BYTE sessionId
- ULONGLONG * pLatitude
- ULONGLONG * pLongitude
- ULONG * pHorUncCircular
- ULONG * pHorUncEllipseSemiMinor
- ULONG * pHorUncEllipseSemiMajor
- ULONG * pHorUncEllipseOrientAzimuth
- BYTE * pHorConfidence
- ULONG * pHorReliability
- ULONG * pSpeedHorizontal
- ULONG * pSpeedUnc
- ULONG * pAltitudeWrtEllipsoid
- ULONG * pAltitudeWrtMeanSeaLevel
- ULONG * pVertUnc
- BYTE * pVertConfidence
- ULONG * pVertReliability
- ULONG * pSpeedVertical
- ULONG * pHeading
- ULONG * pHeadingUnc
- ULONG * pMagneticDeviation
- ULONG * pTechnologyMask
- precisionDilution * pPrecisionDilution
- ULONGLONG * pTimestampUtc
- BYTE * pLeapSeconds
- gpsTime * pGpsTime
- ULONG * pTimeUnc
- ULONG * pTimeSrc
- sensorDataUsage * pSensorDataUsage
- ULONG * pFixId
- svUsedforFix * pSvUsedforFix
- BYTE * pAltitudeAssumed

8.636.1 Detailed Description

This structure contains Event Position Report

Parameters

<i>sessionStatus</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Session was successful – 1 - Session is still in progress; further position reports will be generated until either the fix criteria specified by the client are met or the client response timeout occurs. – 2 - Session failed.. – 3 - Fix request failed because the session timed out. – 4 - Fix request failed because the session was ended by the user. – 5 - Fix request failed due to bad parameters in the request. – 6 - Fix request failed because the phone is offline. – 7 - Fix request failed because the engine is locked
<i>sessionId</i>	<ul style="list-style-type: none"> • ID of the session that was specified in the Start request • Range - 0 to 255
<i>pLatitude</i>	<ul style="list-style-type: none"> • Latitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -90.0 to 90.0 • Positive values indicate northern latitude • Negative values indicate southern latitude
<i>pLongitude</i>	<ul style="list-style-type: none"> • Longitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -180.0 to 180.0 • Positive values indicate eastern latitude • Negative values indicate western latitude
<i>pHorUncCircular</i>	<ul style="list-style-type: none"> • Horizontal position uncertainty. • Units - Meters
<i>pHorUnc↔ EllipseSemi↔ Minor</i>	<ul style="list-style-type: none"> • Semi-minor axis of horizontal elliptical uncertainty. • Units - Meters
<i>pHorUnc↔ EllipseSemi↔ Major</i>	<ul style="list-style-type: none"> • Semi-major axis of horizontal elliptical uncertainty. • Units: Meters
<i>pHorUnc↔ EllipseOrient↔ Azimuth</i>	<ul style="list-style-type: none"> • Elliptical horizontal uncertainty azimuth of orientation. • Units - Decimal degrees • Range - 0 to 180

<i>pHorConfidence</i>	<ul style="list-style-type: none"> • Horizontal uncertainty confidence. • If both elliptical and horizontal uncertainties are specified in this message, the confidence corresponds to the elliptical uncertainty. • Units - Percentage • Range 0-99
<i>pHorReliability</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk – 2 - Location reliability is low; little or no cross-checking is possible. – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed
<i>pSpeed</i> ↔ <i>Horizontal</i>	<ul style="list-style-type: none"> • Horizontal speed. • Units - Meters/second
<i>pSpeedUnc</i>	<ul style="list-style-type: none"> • 3-D Speed uncertainty. • Units - Meters/second.
<i>pAltitudeWrt</i> ↔ <i>Ellipsoid</i>	<ul style="list-style-type: none"> • Altitude With Respect to WGS84 Ellipsoid. • Units - Meters • Range -500 to 15883
<i>pAltitudeWrt</i> ↔ <i>MeanSeaLevel</i>	<ul style="list-style-type: none"> • Altitude With Respect to Sea Level. • Units - Meters
<i>pVertUnc</i>	<ul style="list-style-type: none"> • Vertical uncertainty. • Units - Meters
<i>pVertConfidence</i>	<ul style="list-style-type: none"> • Vertical uncertainty confidence. • Units - Percentage • Range 0 to 99
<i>pVertReliability</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk. – 2 - Location reliability is low; little or no cross-checking is possible – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed

<i>pSpeedVertical</i>	<ul style="list-style-type: none"> • Vertical speed. • Units - Meters/second
<i>pHeading</i>	<ul style="list-style-type: none"> • Heading. • Units - Degree • Range 0 to 359.999
<i>pHeadingUnc</i>	<ul style="list-style-type: none"> • Heading uncertainty. • Units - Degree • Range 0 to 359.999
<i>pMagnetic↔ Deviation</i>	<ul style="list-style-type: none"> • Difference between the bearing to true north and the bearing shown on a magnetic compass. The deviation is positive when the magnetic north is east of true north.
<i>pTechnology↔ Mask</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0x00000001 - Satellites were used to generate the fix – 0x00000002 - Cell towers were used to generate the fix – 0x00000004 - Wi-Fi access points were used to generate the fix – 0x00000008 - Sensors were used to generate the fix – 0x00000010 - Reference Location was used to generate the fix – 0x00000020 - Coarse position injected into the location engine was used to generate the fix – 0x00000040 - AFLT was used to generate the fix – 0x00000080 - GNSS and network-provided measurements were used to generate the fix
<i>-pPrecision↔ Dilution</i>	<ul style="list-style-type: none"> • See precisionDilution for more information
<i>pTimestampUtc</i>	<ul style="list-style-type: none"> • UTC timestamp • Units - Milliseconds since Jan. 1, 1970
<i>pLeapSeconds</i>	<ul style="list-style-type: none"> • Leap second information. If leapSeconds is not available, timestampUtc is calculated based on a hard-coded value for leap seconds. • Units - Seconds
<i>-pGpsTime</i>	<ul style="list-style-type: none"> • See gpsTime for more information
<i>pTimeUnc</i>	<ul style="list-style-type: none"> • Time uncertainty. • Units - Milliseconds

<i>pTimeSrc</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Invalid time. – 1 - Time is set by the 1X system. – 2 - Time is set by WCDMA/GSM time tagging. – 3 - Time is set by an external injection. – 4 - Time is set after decoding over-the-air GPS navigation data from one GPS satellite. – 5 - Time is set after decoding over-the-air GPS navigation data from multiple satellites. – 6 - Both time of the week and the GPS week number are known. – 7 - Time is set by the position engine after the fix is obtained – 8 - Time is set by the position engine after performing SFT, this is done when the clock time uncertainty is large. – 9 - Time is set after decoding GLO satellites. – 10- Time is set after transforming the GPS to GLO time – 11- Time is set by the sleep time tag provided by the WCDMA network. – 12- Time is set by the sleep time tag provided by the GSM network – 13- Source of the time is unknown – 14- Time is derived from the system clock (better known as the slow clock); GNSS time is maintained irrespective of the GNSS receiver state – 15- Time is set after decoding QZSS satellites. – 16- Time is set after decoding BDS satellites.
<i>-pSensorDataUsage</i>	<ul style="list-style-type: none"> • See sensorDataUsage for more information
<i>pFixId</i>	<ul style="list-style-type: none"> • Fix count for the session. Starts with 0 and increments by one for each successive position report for a particular session.
<i>-pSvUsedforFix</i>	<ul style="list-style-type: none"> • See svUsedforFix for more information
<i>pAltitudeAssumed</i>	<ul style="list-style-type: none"> • Indicates whether altitude is assumed or calculated.

- Value
 - 0x00 - Altitude is calculated
 - 0x01 - Altitude is assumed

8.636.2 Field Documentation

8.636.2.1 **BYTE*** QmiCbkLocPositionReportInd::pAltitudeAssumed

8.636.2.2 **ULONG*** QmiCbkLocPositionReportInd::pAltitudeWrtEllipsoid

8.636.2.3 **ULONG*** QmiCbkLocPositionReportInd::pAltitudeWrtMeanSeaLevel

- 8.636.2.4 **ULONG*** QmiCbkLocPositionReportInd::pFixId
- 8.636.2.5 **gpsTime*** QmiCbkLocPositionReportInd::pGpsTime
- 8.636.2.6 **ULONG*** QmiCbkLocPositionReportInd::pHeading
- 8.636.2.7 **ULONG*** QmiCbkLocPositionReportInd::pHeadingUnc
- 8.636.2.8 **BYTE*** QmiCbkLocPositionReportInd::pHorConfidence
- 8.636.2.9 **ULONG*** QmiCbkLocPositionReportInd::pHorReliability
- 8.636.2.10 **ULONG*** QmiCbkLocPositionReportInd::pHorUncCircular
- 8.636.2.11 **ULONG*** QmiCbkLocPositionReportInd::pHorUncEllipseOrientAzimuth
- 8.636.2.12 **ULONG*** QmiCbkLocPositionReportInd::pHorUncEllipseSemiMajor
- 8.636.2.13 **ULONG*** QmiCbkLocPositionReportInd::pHorUncEllipseSemiMinor
- 8.636.2.14 **ULONGLONG*** QmiCbkLocPositionReportInd::pLatitude
- 8.636.2.15 **BYTE*** QmiCbkLocPositionReportInd::pLeapSeconds
- 8.636.2.16 **ULONGLONG*** QmiCbkLocPositionReportInd::pLongitude
- 8.636.2.17 **ULONG*** QmiCbkLocPositionReportInd::pMagneticDeviation
- 8.636.2.18 **precisionDilution*** QmiCbkLocPositionReportInd::pPrecisionDilution
- 8.636.2.19 **sensorDataUsage*** QmiCbkLocPositionReportInd::pSensorDataUsage
- 8.636.2.20 **ULONG*** QmiCbkLocPositionReportInd::pSpeedHorizontal
- 8.636.2.21 **ULONG*** QmiCbkLocPositionReportInd::pSpeedUnc
- 8.636.2.22 **ULONG*** QmiCbkLocPositionReportInd::pSpeedVertical
- 8.636.2.23 **svUsedforFix*** QmiCbkLocPositionReportInd::pSvUsedforFix
- 8.636.2.24 **ULONG*** QmiCbkLocPositionReportInd::pTechnologyMask
- 8.636.2.25 **ULONG*** QmiCbkLocPositionReportInd::pTimeSrc
- 8.636.2.26 **ULONGLONG*** QmiCbkLocPositionReportInd::pTimestampUtc

8.636.2.27 **ULONG*** QmiCbkLocPositionReportInd::pTimeUnc

8.636.2.28 **BYTE*** QmiCbkLocPositionReportInd::pVertConfidence

8.636.2.29 **ULONG*** QmiCbkLocPositionReportInd::pVertReliability

8.636.2.30 **ULONG*** QmiCbkLocPositionReportInd::pVertUnc

8.636.2.31 **BYTE** QmiCbkLocPositionReportInd::sessionId

8.636.2.32 **ULONG** QmiCbkLocPositionReportInd::sessionStatus

8.637 QmiCbkLocSensorStreamingInd Struct Reference

Data Fields

- [accelAcceptReady](#) * [pAccelAcceptReady](#)
- [gyroAcceptReady](#) * [pGyroAcceptReady](#)
- [accelTempAcceptReady](#) * [pAccelTempAcceptReady](#)
- [gyroTempAcceptReady](#) * [pGyroTempAcceptReady](#)

8.637.1 Detailed Description

This structure contains LOC Event Sensor Streaming Ready Status

Parameters

<i>-pAccelAcceptReady</i>	<ul style="list-style-type: none"> • See accelAcceptReady for more information
<i>-pGyroAcceptReady</i>	<ul style="list-style-type: none"> • See gyroAcceptReady for more information
<i>-pAccelTempAcceptReady</i>	<ul style="list-style-type: none"> • See accelTempAcceptReady for more information
<i>-pGyroTempAcceptReady</i>	<ul style="list-style-type: none"> • See gyroTempAcceptReady for more information

8.637.2 Field Documentation

8.637.2.1 **accelAcceptReady*** QmiCbkLocSensorStreamingInd::pAccelAcceptReady

8.637.2.2 **accelTempAcceptReady*** QmiCbkLocSensorStreamingInd::pAccelTempAcceptReady

8.637.2.3 `gyroAcceptReady*` `QmiCbkLocSensorStreamingInd::pGyroAcceptReady`

8.637.2.4 `gyroTempAcceptReady*` `QmiCbkLocSensorStreamingInd::pGyroTempAcceptReady`

8.638 QmiCbkLocSetExtPowerConfigInd Struct Reference

Data Fields

- [ULONG status](#)

8.638.1 Detailed Description

This structure contains LOC Set External Power Config Status

Parameters

<i>status</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Request was completed successfully – 1 - Request failed because of a general failure. – 2 - Request failed because it is not supported. – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - engine could not allocate sufficient memory – 9 - Request failed because the maximum number of Geofences are already programmed – 10 -Location service failed because of an XTRA version-based file format check failure
---------------	---

8.638.2 Field Documentation

8.638.2.1 `ULONG QmiCbkLocSetExtPowerConfigInd::status`

8.639 QmiCbkNasLTECphyCalInfo Struct Reference

Data Fields

- [PhyCaAggScellIndType sPhyCaAggScellIndType](#)
- [PhyCaAggScellIDBw sPhyCaAggScellIDBw](#)
- [PhyCaAggScellInfo sPhyCaAggScellInfo](#)
- [PhyCaAggPcellInfo sPhyCaAggPcellInfo](#)
- [PhyCaAggScellIndex sPhyCaAggScellIndex](#)

8.639.1 Detailed Description

Structure for storing the LTEC PHY CA indication parameters.

Parameters

<i>pPhyCaAgg</i> ↔ <i>ScellIndType</i>	<ul style="list-style-type: none"> See PhyCaAggScellIndType for more information.
<i>sPhyCaAgg</i> ↔ <i>ScellDIBw</i>	<ul style="list-style-type: none"> See PhyCaAggScellDIBw for more information.
<i>sPhyCaAgg</i> ↔ <i>ScellInfo</i>	<ul style="list-style-type: none"> See PhyCaAggScellInfo for more information.
<i>sPhyCaAgg</i> ↔ <i>PcellInfo</i>	<ul style="list-style-type: none"> See PhyCaAggPcellInfo for more information.
<i>sPhyCaAgg</i> ↔ <i>ScellIndex</i>	<ul style="list-style-type: none"> See PhyCaAggScellIndex for more information.

8.639.2 Field Documentation

8.639.2.1 **PhyCaAggPcellInfo** QmiCbkNasLTECphyCalInfo::sPhyCaAggPcellInfo

8.639.2.2 **PhyCaAggScellDIBw** QmiCbkNasLTECphyCalInfo::sPhyCaAggScellDIBw

8.639.2.3 **PhyCaAggScellIndex** QmiCbkNasLTECphyCalInfo::sPhyCaAggScellIndex

8.639.2.4 **PhyCaAggScellIndType** QmiCbkNasLTECphyCalInfo::sPhyCaAggScellIndType

8.639.2.5 **PhyCaAggScellInfo** QmiCbkNasLTECphyCalInfo::sPhyCaAggScellInfo

8.640 QmiCbkSwiOmaDmEventStatusReportInd Struct Reference

Data Fields

- struct [sessionInfoTlv](#) SITlv

8.640.1 Field Documentation

8.640.1.1 struct **sessionInfoTlv** QmiCbkSwiOmaDmEventStatusReportInd::SITlv

8.641 QmiCbkSwiOmaDmEventStatusReportIndExt Struct Reference

Data Fields

- struct [sessionInfoTlvExt](#) SITlv

8.641.1 Field Documentation

8.641.1.1 struct sessionInfoTlvExt QmiCbkSwiOmaDmEventStatusReportIndExt::SITlv

8.642 QmiCbkTmdMitiLvlRptInd Struct Reference

Data Fields

- [_MitigationDevInfo](#) [MitigationDevInfo](#)
- [BYTE](#) [currentMitigationLvl](#)

8.642.1 Detailed Description

This structure contains LOC Cradle Mount Config Status

Parameters

<i>MitigationDev↔ Info</i>	<ul style="list-style-type: none"> • See MitigationDevInfo for more information.
<i>current↔ MitigationLvl</i>	<ul style="list-style-type: none"> • Current Thermal Mitigation Level

8.642.2 Field Documentation

8.642.2.1 [BYTE](#) [QmiCbkTmdMitiLvlRptInd::currentMitigationLvl](#)

8.642.2.2 [_MitigationDevInfo](#) [QmiCbkTmdMitiLvlRptInd::MitigationDevInfo](#)

8.643 QmiCbkWdsStatisticsIndState Struct Reference

Data Fields

- [DataUlongTlv](#) [TxOkConutTlv](#)
- [DataUlongTlv](#) [RxOkConutTlv](#)
- [DataUlongLongTlv](#) [TxOkByteCountTlv](#)
- [DataUlongLongTlv](#) [RxOkByteCountTlv](#)
- [DataUlongTlv](#) [TxDropConutTlv](#)
- [DataUlongTlv](#) [RxDropConutTlv](#)

8.643.1 Detailed Description

WDS Pkt RM Transfer Statistics data structure for individual session

Parameters

<i>TxOkConutTlv</i>	<ul style="list-style-type: none"> • Tx Ok Packet Tlv Value.
<i>RxOkConutTlv</i>	<ul style="list-style-type: none"> • Rx Ok Packet Tlv Value.
<i>TxOkByte↔CountTlv</i>	<ul style="list-style-type: none"> • Tx Ok Byte Count Packet Tlv Value.
<i>RxOkByte↔CountTlv</i>	<ul style="list-style-type: none"> • Rx Ok Byte Count Packet Tlv Value.
<i>TxDropConutTlv</i>	<ul style="list-style-type: none"> • Tx Drop Count Packet Tlv Value.
<i>RxDropConutTlv</i>	<ul style="list-style-type: none"> • Rx Drop Count Packet Tlv Value.

8.643.2 Field Documentation

8.643.2.1 **DataULongTlv** QmiCbkWdsStatisticsIndState::RxDropConutTlv

8.643.2.2 **DataULongLongTlv** QmiCbkWdsStatisticsIndState::RxOkByteCountTlv

8.643.2.3 **DataULongTlv** QmiCbkWdsStatisticsIndState::RxOkConutTlv

8.643.2.4 **DataULongTlv** QmiCbkWdsStatisticsIndState::TxDropConutTlv

8.643.2.5 **DataULongLongTlv** QmiCbkWdsStatisticsIndState::TxOkByteCountTlv

8.643.2.6 **DataULongTlv** QmiCbkWdsStatisticsIndState::TxOkConutTlv

8.644 qmifwinfo_s Struct Reference

Data Fields

- union {
 - struct [fwinfo_s](#) g
 - struct [slqsfwinfo_s](#) s
- } [dev](#)

8.644.1 Detailed Description

Top level structure for storing information about firmware images. union of structures depending on device type, MC77xx or MC83xx

Parameters

<i>g</i>	- structure for MC83xx devices
<i>s</i>	- structure for devices with SPKG CWE file support

- List of various Firmware Images Supported

Technology	Initials	Carrier	Region	Network Technology
D3600	S	eGOBI_IMG_CAR_SPRINT	eGOBI_IMG_REG_NA	eGOBI_IMG_TECH_CDMA
D3600	V	eGOBI_IMG_CAR_VERIZON	eGOBI_IMG_REG_NA	eGOBI_IMG_TECH_CDMA
D3600	C	eGOBI_IMG_CAR_CHINA_TELECOM	eGOBI_IMG_REG_ASIA	eGOBI_IMG_TECH_CDMA
D3600	G	eGOBI_IMG_CAR_GENERIC_CDMA	eGOBI_IMG_REG_GLOBAL	eGOBI_IMG_TECH_CDMA (item for Generic
D3600	H	eGOBI_IMG_CAR_GENERIC_CDMA	eGOBI_IMG_REG_GLOBAL	eGOBI_IMG_TECH_CDMA (item for Generic
D3200	V	eGOBI_IMG_CAR_VODAFONE	eGOBI_IMG_REG_NA	eGOBI_IMG_TECH_UMTS
D3200	A	eGOBI_IMG_CAR_ATT	eGOBI_IMG_REG_NA	eGOBI_IMG_TECH_UMTS
D3200	L	eGOBI_IMG_CAR_TMOBILE	eGOBI_IMG_REG_EU	eGOBI_IMG_TECH_UMTS
D3200	G	eGOBI_IMG_CAR_GENERIC	eGOBI_IMG_REG_GLOBAL	eGOBI_IMG_TECH_UMTS
D3200	H	eGOBI_IMG_CAR_TELEFONICA	eGOBI_IMG_REG_EU	eGOBI_IMG_TECH_UMTS
D3200	I	eGOBI_IMG_CAR_TELCOM_ITALIA	eGOBI_IMG_REG_NA	eGOBI_IMG_TECH_UMTS
D3200	O	eGOBI_IMG_CAR_ORANGE	eGOBI_IMG_REG_NA	eGOBI_IMG_TECH_UMTS
D3200	U	eGOBI_IMG_CAR_GENERIC	eGOBI_IMG_REG_GLOBAL	eGOBI_IMG_TECH_UMTS
D3200	R	eGOBI_IMG_CAR_ROGERS	eGOBI_IMG_REG_NA	eGOBI_IMG_TECH_UMTS
D3600	A	eGOBI_IMG_CAR_AERIS	eGOBI_IMG_REG_NA	eGOBI_IMG_TECH_CDMA

See also

[fwinfo_s](#)
[slqsfwinfo_s](#)

8.644.2 Field Documentation

8.644.2.1 `union { ... } qmifwinfo_s::dev`

8.644.2.2 `struct fwinfo_s qmifwinfo_s::g`

8.644.2.3 `struct slqsfwinfo_s qmifwinfo_s::s`

8.645 QmiNas3GppNetworkInfo Struct Reference

Data Fields

- [WORD pMCC](#)
- [WORD pMNC](#)
- [ULONG plnUse](#)
- [ULONG pRoaming](#)
- [ULONG pForbidden](#)
- [ULONG pPreferred](#)
- [CHAR pDescription](#) [255]

8.645.1 Detailed Description

This structure contains the PerformNetworkScan response parameters. This structure will hold the array of the network scan information.

Parameters

<i>pMCC</i>	<ul style="list-style-type: none"> • Mobile Country Code
<i>pMNC</i>	<ul style="list-style-type: none"> • Mobile Networ Code
<i>pInUse</i>	<ul style="list-style-type: none"> • Is the Network the current serving Network <ul style="list-style-type: none"> – 0 – Unknown – 1 – Current serving network – 2 – Not current serving network, available
<i>pRoaming</i>	<ul style="list-style-type: none"> • Home/Roam Status of the Network <ul style="list-style-type: none"> – 0 – Unknown – 1 – Home – 2 – Roam
<i>pForbidden</i>	<ul style="list-style-type: none"> • Is the Network in the forbidden network list <ul style="list-style-type: none"> – 0 – Unknown – 1 – Forbidden – 2 – Not Forbidden
<i>pPreferred</i>	<ul style="list-style-type: none"> • Is the Network in the Preferred network list <ul style="list-style-type: none"> – 0 – Unknown – 1 – Preferred – 2 – Not Preferred
<i>pDescription</i>	<ul style="list-style-type: none"> • Network Name/Description

8.645.2 Field Documentation

8.645.2.1 **CHAR** QmiNas3GppNetworkInfo::pDescription[255]

8.645.2.2 **ULONG** QmiNas3GppNetworkInfo::pForbidden

8.645.2.3 **ULONG** QmiNas3GppNetworkInfo::pInUse

8.645.2.4 **WORD** QmiNas3GppNetworkInfo::pMCC

8.645.2.5 **WORD** QmiNas3GppNetworkInfo::pMNC

8.645.2.6 **ULONG** QmiNas3GppNetworkInfo::pPreferred

8.645.2.7 **ULONG** QmiNas3GppNetworkInfo::pRoaming

8.646 QmiNasGetRFBandInfoResp Struct Reference

Data Fields

- struct qmTlvResult [results](#)
- [BYTE](#) * [pInstancesSize](#)
- struct [RFBandInfoElements](#) * [pRFBandInfoElements](#)

8.646.1 Field Documentation

8.646.1.1 [BYTE](#)* [QmiNasGetRFBandInfoResp::pInstancesSize](#)

8.646.1.2 struct [RFBandInfoElements](#)* [QmiNasGetRFBandInfoResp::pRFBandInfoElements](#)

8.646.1.3 struct qmTlvResult [QmiNasGetRFBandInfoResp::results](#)

8.647 QmiNasPerformNetworkScanResp Struct Reference

Data Fields

- struct qmTlvResult [results](#)
- [BYTE](#) * [pInstanceSize](#)
- struct [QmiNas3GppNetworkInfo](#) * [pInstances](#)

8.647.1 Field Documentation

8.647.1.1 struct [QmiNas3GppNetworkInfo](#)* [QmiNasPerformNetworkScanResp::pInstances](#)

8.647.1.2 [BYTE](#)* [QmiNasPerformNetworkScanResp::pInstanceSize](#)

8.647.1.3 struct qmTlvResult [QmiNasPerformNetworkScanResp::results](#)

8.648 qmiSmsMessageList Struct Reference

Data Fields

- uint32_t [messageIndex](#)
- uint32_t [messageTag](#)

8.648.1 Detailed Description

Parameters

<i>messageIndex</i>	<ul style="list-style-type: none">• Message index of each matched message
	Generated by Doxygen
<i>messageTag</i>	<ul style="list-style-type: none">• Messagetag

8.648.2 Field Documentation

8.648.2.1 uint32_t qmiSmsMessageList::messageIndex

8.648.2.2 uint32_t qmiSmsMessageList::messageTag

8.649 qmiWSDSDataBearerTechnology Struct Reference

Data Fields

- uint8_t [currentNetwork](#)
- uint32_t [ratMask](#)
- uint32_t [soMask](#)

8.649.1 Detailed Description

Parameters

<i>currentNetwork</i>	current selected network
<i>Radio</i>	Access Technology (RAT) mask
<i>soMask</i>	Service Option (SO) mask

8.649.2 Field Documentation

8.649.2.1 uint8_t qmiWSDSDataBearerTechnology::currentNetwork

8.649.2.2 uint32_t qmiWSDSDataBearerTechnology::ratMask

8.649.2.3 uint32_t qmiWSDSDataBearerTechnology::soMask

8.650 QmiWdsIpAddressInfo Struct Reference

Data Fields

- ULONG * [pIPAddressV4](#)
- USHORT * [pIPAddressV6](#)
- BYTE * [pIPv6prefixlen](#)

8.650.1 Detailed Description

Parameters

<i>pIPAddress</i> ↔ <i>V4[OUT]</i>	<ul style="list-style-type: none"> Current IPv4 address default value of 0 if not reported by the device.
<i>pIPAddress</i> ↔ <i>V6[OUT]</i>	<ul style="list-style-type: none"> Current IPv6 address Space for storing the 8 element array of type USHORT for the IPv6 address is allocated by the application. The IP Address is stored in the user supplied buffer as follows: User buffer: [<U0>..<<U7>] IPv6 address from the network: 1234:2A01:.....:5678 User buffer contents: U0 corresponds to 1234 U1 corresponds to 2A01 ----- ----- U7 corresponds to 5678
<i>pl</i> ↔ <i>Pv6prefixlen</i> [O↔ <i>UT]</i>	<ul style="list-style-type: none"> IPv6 prefix length in number of bits

8.650.2 Field Documentation

8.650.2.1 **ULONG*** *QmiWdsIpAddressInfo::pIPAddressV4*8.650.2.2 **USHORT*** *QmiWdsIpAddressInfo::pIPAddressV6*8.650.2.3 **BYTE*** *QmiWdsIpAddressInfo::pIPv6prefixlen*8.651 *qmiWdsRunTimeSettings* Struct Reference

Data Fields

- **CHAR *** *pProfileName*
- **ULONG *** *pPDPTType*
- **CHAR *** *pAPNName*
- **ULONG *** *pPrimaryDNSV4*
- **ULONG *** *pSecondaryDNSV4*
- **struct UMTSQoS *** *pUMTSGrantedQoS*
- **struct GPRSQoS *** *pGPRSGrantedQoS*
- **CHAR *** *pUsername*
- **ULONG *** *pAuthentication*
- **ULONG *** *pIPAddressV4*
- **struct ProfileIdentifier *** *pProfileID*
- **ULONG *** *pGWAddressV4*
- **ULONG *** *pSubnetMaskV4*
- **BYTE *** *pPCSCFAddrPCO*

- struct [PCSCFIPv4ServerAddressList](#) * [pServerAddrList](#)
- struct [PCSCFFQDNAddressList](#) * [pPCSCFFQDNAddrList](#)
- [USHORT](#) * [pPrimaryDNSV6](#)
- [USHORT](#) * [pSecondaryDNSV6](#)
- [ULONG](#) * [pMtu](#)
- struct [DomainNameList](#) * [pDomainList](#)
- [BYTE](#) * [pIPFamilyPreference](#)
- [BYTE](#) * [pIMCNflag](#)
- [WORD](#) * [pTechnology](#)
- struct [IPv6AddressInfo](#) * [pIPv6AddrInfo](#)
- struct [IPv6GWAddressInfo](#) * [pIPv6GWAddrInfo](#)

8.651.1 Detailed Description

This structure contains the [WdsRunTimeSettings](#) Information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>pProfileName</i>	<ul style="list-style-type: none"> • Profile name One or more bytes describing the profile. Description may be a user-defined name for the profile. QMI_ERR_ARG_TOO_LONG is returned if profile_name is too long.
<i>pPDPTType</i>	<ul style="list-style-type: none"> • PDP type <ul style="list-style-type: none"> – 0 – PDP-IP (IPv4) – 1 - PDP-PPP – 2 - PDP-IPv6 – 3 - PDP-IPv4v6 – 0xffffffff - invalid
<i>pAPNName</i>	<ul style="list-style-type: none"> • Access point name String parameter that is a logical name used to select the GGSN and external packet data network. If value is NULL or omitted, then the subscription default value is requested. QMI_ERR_ARG_TOO_LONG is returned if the APN name is too long.
<i>pPrimaryDNSV4</i>	<ul style="list-style-type: none"> • Primary DNS IPv4 Address
<i>pSecondaryDNSV4</i>	<ul style="list-style-type: none"> • Secondary DNS IPv4 Address
<i>pUMTSGrantedQoS</i>	<ul style="list-style-type: none"> • UMTS Granted QoS
<i>pGPRSGrantedQoS</i>	<ul style="list-style-type: none"> • GPRS Granted QoS

<i>pUsername</i>	<ul style="list-style-type: none"> User name used during data network authentication
<i>pAuthentication</i>	<ul style="list-style-type: none"> Authentication preference <ul style="list-style-type: none"> Bit 0 – PAP preference <ul style="list-style-type: none"> 0 – PAP is never performed 1 – PAP may be performed Bit 1 – CHAP preference <ul style="list-style-type: none"> 0 – CHAP is never performed 1 – CHAP may be performed
<i>pIPAddressV4</i>	<ul style="list-style-type: none"> IPv4 Address assigned to the TE
<i>pProfileID</i>	<ul style="list-style-type: none"> Profile Identifier
<i>pGWAddressV4</i>	<ul style="list-style-type: none"> IPv4 Gateway Address
<i>pSubnetMaskV4</i>	<ul style="list-style-type: none"> IPv4 Subnet Mask
<i>pPCSCFAddr</i> ↔ <i>PCO</i>	<ul style="list-style-type: none"> PCSCF address using PCO values <ul style="list-style-type: none"> 1 – (TRUE) implies request PCSCF address using PCO 0 – (FALSE) implies do not request. This is the default value.
<i>pServerAddrList</i>	<ul style="list-style-type: none"> P-CSCF IPv4 Server Address List
<i>pPCSCFFQD</i> ↔ <i>NAddrList</i>	<ul style="list-style-type: none"> P-CSCF FQDN Address List
<i>pPrimaryDNSV6</i>	<ul style="list-style-type: none"> Primary DNS IPv6 Address
<i>pSecondaryD</i> ↔ <i>NSV6</i>	<ul style="list-style-type: none"> Secondary DNS IPv6 Address
<i>mtu</i>	<ul style="list-style-type: none"> MTU
<i>pDomainList</i>	<ul style="list-style-type: none"> Domain-Name List
<i>pIPFamily</i> ↔ <i>Preference</i>	<ul style="list-style-type: none"> IP family <ul style="list-style-type: none"> 0x04 – IPV4 ADDR 0x06 – IPV6 ADDR

<i>pIMCNflag</i>	<ul style="list-style-type: none"> IM CN Flag <ul style="list-style-type: none"> 0x00 – FALSE 0x01 – TRUE
<i>pTechnology</i>	<ul style="list-style-type: none"> Technology <ul style="list-style-type: none"> CDMA – 0x8001 UMTS – 0x8004
<i>pIPv6AddressInfo</i>	<ul style="list-style-type: none"> IPv6 Address Information
<i>pIPv6GWAddressInfo</i>	<ul style="list-style-type: none"> IPv6 Gateway Address Information

8.651.2 Field Documentation

8.651.2.1 **CHAR*** qmiWdsRunTimeSettings::pAPNName

8.651.2.2 **ULONG*** qmiWdsRunTimeSettings::pAuthentication

8.651.2.3 **struct DomainNameList*** qmiWdsRunTimeSettings::pDomainList

8.651.2.4 **struct GPRSQoS*** qmiWdsRunTimeSettings::pGPRSGrantedQoS

8.651.2.5 **ULONG*** qmiWdsRunTimeSettings::pGWAddressV4

8.651.2.6 **BYTE*** qmiWdsRunTimeSettings::pIMCNflag

8.651.2.7 **ULONG*** qmiWdsRunTimeSettings::pIPAddressV4

8.651.2.8 **BYTE*** qmiWdsRunTimeSettings::pIPFamilyPreference

8.651.2.9 **struct IPV6AddressInfo*** qmiWdsRunTimeSettings::pIPv6AddrInfo

8.651.2.10 **struct IPV6GWAddressInfo*** qmiWdsRunTimeSettings::pIPv6GWAddrInfo

8.651.2.11 **ULONG*** qmiWdsRunTimeSettings::pMtu

8.651.2.12 **BYTE*** qmiWdsRunTimeSettings::pPCSCFAddrPCO

8.651.2.13 **struct PCSCFFQDNAddressList*** qmiWdsRunTimeSettings::pPCSCFFQDNAddrList

8.651.2.14 **ULONG*** qmiWdsRunTimeSettings::pPDPTType

- 8.651.2.15 **ULONG*** qmiWdsRunTimeSettings::pPrimaryDNSV4
- 8.651.2.16 **USHORT*** qmiWdsRunTimeSettings::pPrimaryDNSV6
- 8.651.2.17 **struct ProfileIdentifier*** qmiWdsRunTimeSettings::pProfileID
- 8.651.2.18 **CHAR*** qmiWdsRunTimeSettings::pProfileName
- 8.651.2.19 **ULONG*** qmiWdsRunTimeSettings::pSecondaryDNSV4
- 8.651.2.20 **USHORT*** qmiWdsRunTimeSettings::pSecondaryDNSV6
- 8.651.2.21 **struct PCSCFIPv4ServerAddressList*** qmiWdsRunTimeSettings::pServerAddrList
- 8.651.2.22 **ULONG*** qmiWdsRunTimeSettings::pSubnetMaskV4
- 8.651.2.23 **WORD*** qmiWdsRunTimeSettings::pTechnology
- 8.651.2.24 **struct UMTSQoS*** qmiWdsRunTimeSettings::pUMTSGrantedQoS
- 8.651.2.25 **CHAR*** qmiWdsRunTimeSettings::pUsername

8.652 QosClassID Struct Reference

Data Fields

- [BYTE QCI](#)
- [ULONG gDIBitRate](#)
- [ULONG maxDIBitRate](#)
- [ULONG gUIBitRate](#)
- [ULONG maxUIBitRate](#)

8.652.1 Detailed Description

structure contains 3GPP LTE QoS parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>QCI</i>	<ul style="list-style-type: none"> QOS specified using the QOS Class Identifier (QOS) values QCI value 0 - Requests the network to assign the appropriate QCI value QCI values 1-4 - Associated with guaranteed bit rates QCI values 5-9 - Associated with non-guaranteed bit rates
<i>gDIBitRate</i>	<ul style="list-style-type: none"> Guaranteed DL bit rate
<i>maxDIBitRate</i>	<ul style="list-style-type: none"> maxDIBitRate

8.652.2 Field Documentation

8.652.2.1 **ULONG** QosClassID::gDIBitRate

8.652.2.2 **ULONG** QosClassID::gUIBitRate

8.652.2.3 **ULONG** QosClassID::maxDIBitRate

8.652.2.4 **ULONG** QosClassID::maxUIBitRate

8.652.2.5 **BYTE** QosClassID::QCI

8.653 QosEventInfo Struct Reference

Data Fields

- [ULONG * pDataBearer](#)
- [ULONG * pPacketsCountTX](#)
- [ULONG * pPacketsCountRX](#)
- [ULONGLONG * pTotalBytesTX](#)
- [ULONGLONG * pTotalBytesRX](#)

8.653.1 Detailed Description

Contains the WDS event information and information about the interface

Parameters

<i>pQmiInterfaceInfo</i>	<ul style="list-style-type: none"> • See qaQmiInterfaceInfo for more information
<i>pDataBearer</i>	<ul style="list-style-type: none"> • Data bearer technology (NULL if not present) <ul style="list-style-type: none"> – 0x00 - Indicates that this field is ignored – 0x01 - CDMA 1X – 0x02 - EV-DO Rev 0 – 0x03 - GPRS – 0x04 - WCDMA – 0x05 - EV-DO Rev A – 0x06 - EDGE – 0x07 - HSDPA and WCDMA – 0x08 - WCDMA and HSUPA – 0x09 - HSDPA and HSUPA – 0x0A - LTE – 0x0B - EV-DO Rev A EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA
Generated by Doxygen	<ul style="list-style-type: none"> – 0x8000 - NULL Bearer – 0xFF - Unknown Technology

<i>pDormancy</i> ↔ <i>Status</i>	<ul style="list-style-type: none"> • Dormancy status (NULL if not present) <ul style="list-style-type: none"> – 1 - traffic channel dormant – 2 - traffic channel active
<i>pPackets</i> ↔ <i>CountTX</i>	<ul style="list-style-type: none"> • Packets transmitted without error (NULL if not present)
<i>pPackets</i> ↔ <i>CountRX</i>	<ul style="list-style-type: none"> • Packets received without error (NULL if not present)
<i>pTotalBytesTX</i>	<ul style="list-style-type: none"> • Bytes transmitted without error (NULL if not present)
<i>pTotalBytesRX</i>	<ul style="list-style-type: none"> • Bytes received without error (NULL if not present)

8.653.2 Field Documentation

8.653.2.1 **ULONG*** QosEventInfo::pDataBearer

8.653.2.2 **ULONG*** QosEventInfo::pPacketsCountRX

8.653.2.3 **ULONG*** QosEventInfo::pPacketsCountTX

8.653.2.4 **ULONGLONG*** QosEventInfo::pTotalBytesRX

8.653.2.5 **ULONGLONG*** QosEventInfo::pTotalBytesTX

8.654 QosFlowInfo Struct Reference

Data Fields

- [QosFlowInfoState](#) * pQFlowState
- [swiQosFlow](#) * pTxQFlowGranted
- [swiQosFlow](#) * pRxQFlowGranted
- [swiQosFilter](#) * pTxQFilter [[MAX_QOS_FILTER_TLV](#)]
- [swiQosFilter](#) * pRxQFilter [[MAX_QOS_FILTER_TLV](#)]
- [BYTE](#) * pBearerID

8.654.1 Detailed Description

This structure contains QoS flow info

Parameters

<i>pQFlowState</i>	<ul style="list-style-type: none"> QoS flow state information, please check QosFlowInfoState for more information
<i>pTxQFlowGranted</i>	<ul style="list-style-type: none"> pointer to the Tx Qos flow granted, please check swiQosFlow for more information
<i>pRxQFlowGranted</i>	<ul style="list-style-type: none"> pointer to the Rx Qos flow granted
<i>pTxQFilter</i>	<ul style="list-style-type: none"> pointer to the Tx Qos filter
<i>pRxQFilter</i>	<ul style="list-style-type: none"> pointer to the Rx Qos flow
<i>pBearerID</i>	<ul style="list-style-type: none"> pointer to the bearer ID Bearer ID or Radio Link Protocol (RLP) ID of the activated flow. Valid Values - 0 to 16 0xFF - Invalid value.

8.654.2 Field Documentation

8.654.2.1 **BYTE*** QosFlowInfo::pBearerID

8.654.2.2 **QosFlowInfoState*** QosFlowInfo::pQFlowState

8.654.2.3 **swiQosFilter*** QosFlowInfo::pRxQFilter[MAX_QOS_FILTER_TLV]

8.654.2.4 **swiQosFlow*** QosFlowInfo::pRxQFlowGranted

8.654.2.5 **swiQosFilter*** QosFlowInfo::pTxQFilter[MAX_QOS_FILTER_TLV]

8.654.2.6 **swiQosFlow*** QosFlowInfo::pTxQFlowGranted

8.655 QosFlowInfoState Struct Reference

Data Fields

- [ULONG](#) id
- [BYTE](#) isNewFlow
- [BYTE](#) state

8.655.1 Detailed Description

This structure contains QoS flow state

Parameters

<i>id</i>	QoS identifier
<i>isNewFlow</i>	<ul style="list-style-type: none"> • 1 – Newly added flow • 0 – Existing flow
<i>state</i>	This indicates that the flow that was added/modified/deleted: <ul style="list-style-type: none"> • 0x01 – Flow activated • 0x02 – Flow modified • 0x03 – Flow deleted • 0x04 – Flow suspended • 0x05 – Flow enabled • 0x06 – Flow disabled

8.655.2 Field Documentation

8.655.2.1 ULONG QosFlowInfoState::id

8.655.2.2 BYTE QosFlowInfoState::isNewFlow

8.655.2.3 BYTE QosFlowInfoState::state

8.656 QosMap Struct Reference

Data Fields

- [BYTE dscp](#)
- [ULONG qos_id](#)
- [BYTE state](#)

8.656.1 Detailed Description

This structure contains the SLQSQosDumpMap Information

Parameters

<i>dscp</i>	<ul style="list-style-type: none"> • Differential Service Code Point(DSCP) value
<i>qos_id</i>	<ul style="list-style-type: none"> • QoS identifier
<i>state</i>	<ul style="list-style-type: none"> • QoS Flow state

8.656.2 Field Documentation

8.656.2.1 BYTE QosMap::dscp

8.656.2.2 ULONG QosMap::qos_id

8.656.2.3 BYTE QosMap::state

8.657 RankIndicatorInd Struct Reference

Data Fields

- [WORD Count1](#)
- [WORD Count2](#)

8.657.1 Field Documentation

8.657.1.1 WORD RankIndicatorInd::Count1

8.657.1.2 WORD RankIndicatorInd::Count2

8.658 readResult Struct Reference

Data Fields

- [WORD contentLen](#)
- [BYTE content](#) [255+1]

8.658.1 Detailed Description

This structure contains the information for write operation.

Parameters

<i>contentLen</i>	<ul style="list-style-type: none">• Number of sets of content.
<i>content</i> [<i>MAX_DESCRIPTOR_LENGTH</i>]	<ul style="list-style-type: none">• Read content.• The content is the sequence of bytes as read from the card.

8.658.2 Field Documentation

8.658.2.1 **BYTE** readResult::content[255+1]

8.658.2.2 **WORD** readResult::contentLen

8.659 readTransparentInfo Struct Reference

Data Fields

- [WORD](#) offset
- [WORD](#) length

8.659.1 Detailed Description

This structure contains the information for read operation.

Parameters

<i>offset</i>	<ul style="list-style-type: none">• Offset for the read operation.
<i>length</i>	<ul style="list-style-type: none">• Length of the content to be read.• The value 0 is used to read the complete file.

8.659.2 Field Documentation

8.659.2.1 **WORD** readTransparentInfo::length

8.659.2.2 **WORD** readTransparentInfo::offset

8.660 redirNumInfo Struct Reference

Data Fields

- [BYTE](#) PI
- [BYTE](#) SI
- [BYTE](#) numType
- [BYTE](#) numPlan
- [BYTE](#) reason
- [BYTE](#) numLen
- [BYTE](#) number [255]

8.660.1 Detailed Description

This structure contains Redirecting Number Information

Parameters

<i>PI</i>	<ul style="list-style-type: none"> • Presentation indicator; refer to [S1, Table 2.7.4.4-1] for valid values.
<i>SI</i>	<ul style="list-style-type: none"> • Number of sets of following elements <ul style="list-style-type: none"> – Caller Id
<i>SI</i>	<ul style="list-style-type: none"> • Number screening indicator. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network
<i>numType</i>	<ul style="list-style-type: none"> • Number type. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension
<i>numPlan</i>	<ul style="list-style-type: none"> • Number plan. • Values: <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension
<i>reason</i>	-Redirecting reason; refer to [S1, Table 3.7.5.11-1] for valid values
<i>numLen</i>	<ul style="list-style-type: none"> • Provides the length of number which follow.
<i>number[255]</i>	<ul style="list-style-type: none"> • number of numLen length, NULL terminated.

8.660.2 Field Documentation

8.660.2.1 **BYTE** `redirNumInfo::number[255]`

8.660.2.2 **BYTE** `redirNumInfo::numLen`

8.660.2.3 **BYTE** `redirNumInfo::numPlan`

8.660.2.4 **BYTE** `redirNumInfo::numType`

8.660.2.5 **BYTE** `redirNumInfo::PI`

8.660.2.6 **BYTE** `redirNumInfo::reason`

8.660.2.7 **BYTE** `redirNumInfo::SI`

8.661 registerRefresh Struct Reference

Data Fields

- [BYTE](#) `registerFlag`
- [BYTE](#) `voteForInit`
- [WORD](#) `numFiles`
- [fileInfo](#) `arrfileInfo` [255]

8.661.1 Detailed Description

This structure contains paramaters of refresh Information

Parameters

<i>registerFlag</i>	<ul style="list-style-type: none"> • Flag that indicates whether to register or deregister for refresh indications. Valid values: <ul style="list-style-type: none"> – 0 - Deregister – 1 - Register
<i>voteForInit</i>	<ul style="list-style-type: none"> • Flag that indicates whether to vote for the init when there is a refresh. Valid values: <ul style="list-style-type: none"> – 0 - Client does not vote for initialization – 1 - Client votes for initialization
<i>numFiles</i>	<ul style="list-style-type: none"> • Number of sets of the following elements: <ul style="list-style-type: none"> – <code>file_id</code> – <code>path_len</code> – <code>path</code>
<i>arrfileInfo</i>	<ul style="list-style-type: none"> • Array of file Information structure.
	<ul style="list-style-type: none"> • See /ref fileInfo for more information

8.661.2 Field Documentation

8.661.2.1 `fileInfo` `registerRefresh::arrfileInfo[255]`

8.661.2.2 `WORD` `registerRefresh::numFiles`

8.661.2.3 `BYTE` `registerRefresh::registerFlag`

8.661.2.4 `BYTE` `registerRefresh::voteForInit`

8.662 remainingRetries Struct Reference

Data Fields

- [BYTE](#) `verifyLeft`
- [BYTE](#) `unblockLeft`

8.662.1 Detailed Description

This structure contains the information about the retries remaining.

Parameters

<i>verifyLeft</i>	<ul style="list-style-type: none">• Number of remaining attempts to verify the PIN.• 0xFF, if unavailable.
<i>unblockLeft</i>	<ul style="list-style-type: none">• Number of remaining attempts to unblock the PIN.• 0xFF, if unavailable.

Note

This value is returned only when the enable/disable operation has failed. This information is not sent for a hidden key PIN type.

8.662.2 Field Documentation

8.662.2.1 `BYTE` `remainingRetries::unblockLeft`

8.662.2.2 `BYTE` `remainingRetries::verifyLeft`

8.663 remotePartyName Struct Reference

Data Fields

- [BYTE](#) `namePI`

- [BYTE codingScheme](#)
- [BYTE nameLen](#)
- [BYTE callerName](#) [255]

8.663.1 Detailed Description

This structure contains information about the names that are dialed from the device or from which a call is received on the device.

Parameters

<i>namePI</i>	<ul style="list-style-type: none"> • Name presentation indicator. <ul style="list-style-type: none"> – 0x00 - PRESENTATION_NAME_PRESENTATION_ALLOWED - Allowed presentation – 0x01 - PRESENTATION_NAME_PRESENTATION_RESTRICTED - Restricted presentation – 0x02 - PRESENTATION_NAME_UNAVAILABLE - Unavailable presentation – 0x03 - PRESENTATION_NAME_NAME_PRESENTATION_RESTRICTED - Restricted name presentation – 0xFF - Not Available
<i>codingScheme</i>	<ul style="list-style-type: none"> • Refer to Table10 qaGobiApiTableCodingScheme.h for coding schemes • 0xFF - Not Available
<i>nameLen</i>	<ul style="list-style-type: none"> • Provides the length of name which follow. • If zero(0) then no further information exists.
<i>callerName</i> [<i>M↔AX_DESCRIPTOR_LENGTH</i>]	<ul style="list-style-type: none"> • Name in ASCII, NULL ending.

8.663.2 Field Documentation

8.663.2.1 **BYTE** remotePartyName::callerName[255]

8.663.2.2 **BYTE** remotePartyName::codingScheme

8.663.2.3 **BYTE** remotePartyName::nameLen

8.663.2.4 **BYTE** remotePartyName::namePI

8.664 remotePartyNum Struct Reference

Data Fields

- [BYTE presentationInd](#)
- [BYTE numLen](#)
- [BYTE remPartyNumber](#) [81]

8.664.1 Detailed Description

This structure contains information about the numbers that are dialed from the device or from which a call is received on the device.

Parameters

<i>presentationInd</i>	<ul style="list-style-type: none"> Presentation indicator. <ul style="list-style-type: none"> 0x00 - PRESENTATION_ALLOWED - Allowed presentation 0x01 - PRESENTATION_RESTRICTED - Restricted presentation 0x02 - PRESENTATION_NUM_UNAVAILABLE - Unavailable presentation 0x04 - PRESENTATION_PAYPHONE - Payphone presentation (GSM/UMTS specific) 0xFF - Not Available
<i>numLen</i>	<ul style="list-style-type: none"> Provides the length of number which follow. If zero(0) then no further information exists.
<i>remParty</i> <i>Number</i> [<i>MAX_</i> <i>CALL_NO_LEN</i>]	<ul style="list-style-type: none"> Array of numbers in ASCII, NULL ending.

8.664.2 Field Documentation

8.664.2.1 BYTE remotePartyNum::numLen

8.664.2.2 BYTE remotePartyNum::presentationInd

8.664.2.3 BYTE remotePartyNum::remPartyNumber[81]

8.665 ReqFieldsList Struct Reference

Data Fields

- [BYTE requestFieldsLen](#)
- [BYTE requestFields](#) [256]

8.665.1 Detailed Description

This structure contains the Supported Request Fields List Information

Parameters

<i>requestFields</i> <i>Len</i>	<ul style="list-style-type: none"> Number of sets of the request fields.
<i>requestFields</i>	<ul style="list-style-type: none"> Describes which optional field IDs are supported in QMI Request.
Generated by Doxygen	<ul style="list-style-type: none"> Array of uint8 is a bitmask where each bit represents a field ID. Field 0-15 are mandatory, First Bit represents field ID 16, Starting with the LSB, bit 0 represents Field ID 16, bit 1 represents ID 17.

8.665.2 Field Documentation

8.665.2.1 **BYTE** ReqFieldsList::requestFields[256]

8.665.2.2 **BYTE** ReqFieldsList::requestFieldsLen

8.666 RespFieldsList Struct Reference

Data Fields

- [BYTE responseFieldsLen](#)
- [BYTE responseFields](#) [256]

8.666.1 Detailed Description

This structure contains the Supported Response Fields List Information

Parameters

<i>responseFieldsLen</i>	<ul style="list-style-type: none"> • Number of sets of the response fields.
<i>responseFields</i>	<ul style="list-style-type: none"> • Describes which optional field IDs are supported in QMI Response. • Format is same as request field.

8.666.2 Field Documentation

8.666.2.1 **BYTE** RespFieldsList::responseFields[256]

8.666.2.2 **BYTE** RespFieldsList::responseFieldsLen

8.667 RFBandInfoElements Struct Reference

Data Fields

- [BYTE radiolInterface](#)
- [WORD activeBandClass](#)
- [WORD activeChannel](#)
- [uint8_t radiolInterface](#)
- [uint16_t activeBandClass](#)
- [uint16_t activeChannel](#)

8.667.1 Detailed Description

This structure contains the RFBandInfo response parameters.

Parameters

<i>radioInterface</i>	<ul style="list-style-type: none"> Radio interface technology <ul style="list-style-type: none"> See Tables for Radio Interface
<i>activeBandClass</i>	<ul style="list-style-type: none"> Active Band Class <ul style="list-style-type: none"> See Tables for Band Classes
<i>activeChannel</i>	<ul style="list-style-type: none"> Active channel (0 if channel is not relevant to the reported technology)
<i>radioInterface</i>	radio interface technology
<i>activeBandClass</i>	active band class
<i>activeChannel</i>	active channel

8.667.2 Field Documentation

8.667.2.1 WORD RFBandInfoElements::activeBandClass

8.667.2.2 uint16_t RFBandInfoElements::activeBandClass

8.667.2.3 WORD RFBandInfoElements::activeChannel

8.667.2.4 uint16_t RFBandInfoElements::activeChannel

8.667.2.5 BYTE RFBandInfoElements::radioInterface

8.667.2.6 uint8_t RFBandInfoElements::radioInterface

8.668 rmTrasnferStaticsReq Struct Reference

Data Fields

- uint8_t [bResetStatistics](#)
- uint32_t [ulMask](#)

8.668.1 Detailed Description

Parameters

<i>bResetStatistics</i>	Clear RM statistics
<i>ulMask</i>	Requested statistic bit mask

8.668.2 Field Documentation

8.668.2.1 `uint8_t rmTrasnferStaticsReq::bResetStatistics`

8.668.2.2 `uint32_t rmTrasnferStaticsReq::ulMask`

8.669 roamIndList Struct Reference

Data Fields

- [BYTE numInstances](#)
- [BYTE radiolInterface](#) [0x0A]
- [BYTE roamIndicator](#) [0x0A]

8.669.1 Detailed Description

This structure contains the Roaming Indicator List

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>numInstances</i>	<ul style="list-style-type: none"> • number of sets of radio interface currently in use and roaming indicator <ul style="list-style-type: none"> – defaults to zero
<i>radiolInterface</i>	<ul style="list-style-type: none"> • Radio Interface currently in use • Values: <ul style="list-style-type: none"> – 0x01 - RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - RADIO_IF_CDMA_1XEVD0 - cdma2000 HRPD (1xEV-DO) – 0x03 - RADIO_IF_AMPS - AMPS – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS – 0x08 - RADIO_IF_LTE - LTE
<i>roamIndicator</i>	<ul style="list-style-type: none"> • Roaming Indicator • Values: <ul style="list-style-type: none"> – 0x00 - Roaming – 0x01 - Home

8.669.2 Field Documentation

8.669.2.1 `BYTE roamIndList::numInstances`

8.669.2.2 **BYTE** roamIndList::radioInterface[0x0A]

8.669.2.3 **BYTE** roamIndList::roamIndicator[0x0A]

8.670 RoamingInfo Struct Reference

Data Fields

- [BYTE TlvPresent](#)
- [BYTE roaming_ind](#)

8.670.1 Field Documentation

8.670.1.1 **BYTE** RoamingInfo::roaming_ind

8.670.1.2 **BYTE** RoamingInfo::TlvPresent

8.671 roamTimer Struct Reference

Data Fields

- [BYTE namID](#)
- [ULONG roamTimerValue](#)

8.671.1 Detailed Description

This structure contains information about the Roam Timer.

Parameters

<i>namID</i>	<ul style="list-style-type: none">• Index of the NAM(Number Assignment Module) to be configured.• Range 0 to 3.• Some modems support only 1 or 2 NAMs.• 0xFF,if not available.
<i>roamTimerValue</i>	<ul style="list-style-type: none">• Time in minutes.• Cumulative air time is slammed.• 0xFFFFFFFF,if not available.

8.671.2 Field Documentation

8.671.2.1 **BYTE** roamTimer::namID

8.671.2.2 **ULONG** roamTimer::roamTimerValue

8.672 RSRPThresh Struct Reference

Data Fields

- [BYTE](#) RSRPThresListLen
- [SHORT](#) * pRSRPThresList

8.672.1 Detailed Description

This structure contains RSRP threshold related parameters.

Parameters

<i>RSRPThresListLen</i>	<ul style="list-style-type: none"> • Length of the LTE RSRP threshold list parameter to follow
<i>pRSRPThresList</i>	<ul style="list-style-type: none"> • Sequence of thresholds delimiting current RSRP event reporting bands • Every time a new RSRP value crosses a specified threshold value, an event report indication message with the new RSRQ value is sent to the requesting control point. For this field <ul style="list-style-type: none"> – RSRP values are applicable only for LTE – RSRP values are measured in dBm, with a range of -44 dBm to -140 dBm – Each RSRP threshold value is a signed byte value – Maximum number of threshold values is 16 – At least one value must be specified

8.672.2 Field Documentation

8.672.2.1 **SHORT*** RSRPThresh::pRSRPThresList

8.672.2.2 **BYTE** RSRPThresh::RSRPThresListLen

8.673 rsrqInformation Struct Reference

Data Fields

- [INT8](#) rsrq
- [BYTE](#) radiolf

8.673.1 Detailed Description

This structure contains the RSRQ Information

Parameters

<i>rsrq</i>	<ul style="list-style-type: none"> • RSRQ value in dB (signed integer value); valid range is -3 to -20 (-3 means -3 dB, -20 means -20 dB)
<i>radiolf</i>	<ul style="list-style-type: none"> • Radio interface technology of the signal being measured <ul style="list-style-type: none"> – 0x08 - LTE

8.673.2 Field Documentation

8.673.2.1 BYTE rsrqInformation::radiolf

8.673.2.2 INT8 rsrqInformation::rsrq

8.674 RSRQThresh Struct Reference

Data Fields

- [BYTE RSRQThresListLen](#)
- [INT8 * pRSRQThresList](#)

8.674.1 Detailed Description

This structure contains RSRQ threshold related parameters.

Parameters

<i>RSRQThresListLen</i>	<ul style="list-style-type: none"> • Length of the LTE RSRQ threshold list parameter to follow
<i>pRSRQThresList</i>	<ul style="list-style-type: none"> • Sequence of thresholds delimiting current RSRQ event reporting bands • Every time a new RSRQ value crosses a threshold value, an event report indication message with the new RSRQ value is sent to the requesting control point. For this field <ul style="list-style-type: none"> – RSRQ values are applicable only for LTE – RSRQ values are measured in dBm, with a range of -20 dBm to -3 dBm – Each RSRQ threshold value is a signed byte value – Maximum number of threshold values is 16 – At least one value must be specified

8.674.2 Field Documentation

8.674.2.1 INT8* RSRQThresh::pRSRQThresList

8.674.2.2 BYTE RSRQThresh::RSRQThresListLen

8.675 RSSIThresh Struct Reference

Data Fields

- [BYTE RSSIThresListLen](#)
- [INT8 * pRSSIThresList](#)

8.675.1 Detailed Description

This structure contains RSSI threshold related parameters.

Parameters

<i>RSSIThresListLen</i>	<ul style="list-style-type: none"> • Length of the RSSI threshold list parameter to follow
<i>pRSSIThresList</i>	<ul style="list-style-type: none"> • RSSI in dBm(signed bytes) • A value of -125 dBm or lower is used to indicate No Signal • RSSI values have the following ranges (in dBm) <ul style="list-style-type: none"> – CDMA is -105 to -21 – HDR is -118 to -13 – GSM is -111 to -48 – WCDMA is -121 to 0 – LTE is -120 to 0 • Threshold values specified above are used for all RATs • The maximum number of threshold values is 16, each a signed byte value.

8.675.2 Field Documentation

8.675.2.1 INT8* RSSIThresh::pRSSIThresList

8.675.2.2 BYTE RSSIThresh::RSSIThresListLen

8.676 RXAGCList Struct Reference

Data Fields

- [WORD * pRXStaticGain](#)
- [WORD * pRXAIG](#)
- [WORD * pRXExpThres](#)
- [WORD * pRXExpSlope](#)
- [WORD * pRXComprThres](#)
- [WORD * pRXComprSlope](#)

8.676.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig parameters related to AV_RXAGCLIST.

Parameters

<i>pRXStaticGain</i>	<ul style="list-style-type: none"> • RX pre-compressor static gain
<i>pRXAIG</i>	<ul style="list-style-type: none"> • RX pre-compressor gain selection flag
<i>pRXExpThres</i>	<ul style="list-style-type: none"> • RX expansion threshold
<i>pRXExpSlope</i>	<ul style="list-style-type: none"> • RX expansion slope
<i>pRXComprThres</i>	<ul style="list-style-type: none"> • RX compression threshold
<i>pRXComprSlope</i>	<ul style="list-style-type: none"> • RX compression slope

8.676.2 Field Documentation

8.676.2.1 **WORD*** RXAGCList::pRXAIG

8.676.2.2 **WORD*** RXAGCList::pRXComprSlope

8.676.2.3 **WORD*** RXAGCList::pRXComprThres

8.676.2.4 **WORD*** RXAGCList::pRXExpSlope

8.676.2.5 **WORD*** RXAGCList::pRXExpThres

8.676.2.6 **WORD*** RXAGCList::pRXStaticGain

8.677 RXAVCList Struct Reference

Data Fields

- **WORD *** pAVRXAVCSens
- **WORD *** pAVRXAVCHheadroom

8.677.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig parameters related to AV_RXAVCLIST.

Parameters

<i>pAVRXAVC_↔Sens</i>	<ul style="list-style-type: none"> • AVC variation from nominal sensitivity
<i>pAVRXAVC_↔Headroom</i>	<ul style="list-style-type: none"> • AVC headroom

8.677.2 Field Documentation

8.677.2.1 WORD* RXAVCList::pAVRXAVCHeadroom

8.677.2.2 WORD* RXAVCList::pAVRXAVCSens

8.678 rxInfo Struct Reference

Data Fields

- [BYTE isRadioTuned](#)
- [INT32 rxPower](#)
- [INT32 ecio](#)
- [INT32 rscp](#)
- [INT32 rsrp](#)
- [ULONG phase](#)

8.678.1 Detailed Description

This structure contains the Rx Information.

Parameters

<i>isRadioTuned</i>	<ul style="list-style-type: none"> • Whether Rx is tuned to a channel: <ul style="list-style-type: none"> – 0x00 - Not tuned – 0x01 - Tuned – 0xFF - Not Available • If the radio is tuned, instantaneous values are set for the signal information fields below. • If the radio is not tuned, or is delayed or invalid, the values are set depending on each technology.
<i>rx_pwr</i>	<ul style="list-style-type: none"> • Rx power value in 1/10 dbm resolution.
<i>ecio</i>	<ul style="list-style-type: none"> • ECIO in 1/10 dbm; valid for CDMA, HDR, GSM, WCDMA, and LTE.
<i>rscp</i>	<ul style="list-style-type: none"> • Received signal code power in 1/10 dbm. • Valid for WCDMA.

<i>rsrp</i>	<ul style="list-style-type: none">• Current reference signal received power in 1/10 dbm valid for LTE.
<i>phase</i>	<ul style="list-style-type: none">• Phase in 1/100 degrees; valid for LTE.• When the phase is unknown, 0xFFFFFFFF is used.

8.678.2 Field Documentation

8.678.2.1 INT32 rxInfo::ecio

8.678.2.2 BYTE rxInfo::isRadioTuned

8.678.2.3 ULONG rxInfo::phase

8.678.2.4 INT32 rxInfo::rscp

8.678.2.5 INT32 rxInfo::rsrp

8.678.2.6 INT32 rxInfo::rxPower

8.679 RXPCMIIRFitr Struct Reference

Data Fields

- WORD * pFlag
- WORD * pStageCnt
- BYTE * pStage0Val
- BYTE * pStage1Val
- BYTE * pStage2Val
- BYTE * pStage3Val
- BYTE * pStage4Val

8.679.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig parameters related to AV_RXPCMIIRFLTR.

Parameters

<i>pFlag</i>	<ul style="list-style-type: none"> • Flag <ul style="list-style-type: none"> – 0x0000 - IIR filter disable – 0xffff - IIR filter enable
<i>pStageCnt</i>	<ul style="list-style-type: none"> • Stage Count <ul style="list-style-type: none"> – 0-4
<i>pStage0Val</i>	<ul style="list-style-type: none"> • A 20 BYTE sized parameter indicating Stage 0 value <ul style="list-style-type: none"> – A1 – A2 – B0 – B1 – B2
<i>pStage1Val</i>	<ul style="list-style-type: none"> • A 20 BYTE sized parameter indicating Stage 1 value <ul style="list-style-type: none"> – A1 – A2 – B0 – B1 – B2
<i>pStage2Val</i>	<ul style="list-style-type: none"> • A 20 BYTE sized parameter indicating Stage 2 value <ul style="list-style-type: none"> – A1 – A2 – B0 – B1 – B2
<i>pStage3Val</i>	<ul style="list-style-type: none"> • A 20 BYTE sized parameter indicating Stage 3 value <ul style="list-style-type: none"> – A1 – A2 – B0 – B1 – B2
<i>pStage4Val</i>	<ul style="list-style-type: none"> • A 20 BYTE sized parameter indicating Stage 4 value <ul style="list-style-type: none"> – A1 – A2 – B0 – B1 – B2

8.679.2 Field Documentation

8.679.2.1 WORD* RXPCMIIRFiltr::pFlag

8.679.2.2 BYTE* RXPCMIIRFiltr::pStage0Val

8.679.2.3 BYTE* RXPCMIIRFiltr::pStage1Val

8.679.2.4 BYTE* RXPCMIIRFiltr::pStage2Val

8.679.2.5 BYTE* RXPCMIIRFiltr::pStage3Val

8.679.2.6 BYTE* RXPCMIIRFiltr::pStage4Val

8.679.2.7 WORD* RXPCMIIRFiltr::pStageCnt

8.680 RxSigInfo Struct Reference

Data Fields

- [BYTE rxChainIndex](#)
- [BYTE isRadioTuned](#)
- [INT32 rxPower](#)
- [INT32 rsrp](#)

8.680.1 Detailed Description

This structure contains the parameters for Rx Signal Info.

Parameters

<i>rxChainIndex</i>	<ul style="list-style-type: none">• Rx antenna path• Valid Values<ul style="list-style-type: none">– 0 - Primary Rx– 1 - Diversity Rx
<i>isRadioTuned</i>	<ul style="list-style-type: none">• Rx path is tuned to a channel or Not• Values<ul style="list-style-type: none">– 0x00 - Not tuned– 0x01 - Tuned

Note

If the radio is tuned, the instantaneous values are set for the fields below. If the radio is not tuned, the values set below may be invalid.

Parameters

<i>rxPower</i>	<ul style="list-style-type: none"> Rx power value in 1/10 dBm resolution
<i>rsrp</i>	<ul style="list-style-type: none"> Current reference signal received power in 1/10 dBm resolution

8.680.2 Field Documentation

8.680.2.1 BYTE RxSigInfo::isRadioTuned

8.680.2.2 INT32 RxSigInfo::rsrp

8.680.2.3 BYTE RxSigInfo::rxChainIndex

8.680.2.4 INT32 RxSigInfo::rxPower

8.681 rxSignalStrengthListElement Struct Reference

Data Fields

- [SHORT rxSignalStrength](#)
- [BYTE radiolf](#)

8.681.1 Detailed Description

This structure contains the Received Signal Strength Information

Parameters

<i>rxSignalStrength</i>	<ul style="list-style-type: none"> Received signal strength in dBm <ul style="list-style-type: none"> For CDMA and UMTS, this indicates forward link pilotEc. For GSM, the received signal strength. For LTE, this indicates the total received wideband power observed by UE.
<i>radiolf</i>	<ul style="list-style-type: none"> Radio interface technology of the signal being radio_if measured <ul style="list-style-type: none"> 0x00 - RADIO_IF_NO_SVC - None (no service) 0x01 - RADIO_IF_CDMA_1X - cdma2000 1X 0x02 - RADIO_IF_CDMA_1XEVD0 - cdma2000 HRPD (1xEV-DO) 0x03 - RADIO_IF_AMPS - AMPS 0x04 - RADIO_IF_GSM - GSM 0x05 - RADIO_IF_UMTS - UMTS 0x08 - RADIO_IF_LTE - LTE

Note

First element of the RSSI list always contains the current Signal strength and Radio Interface.

8.681.2 Field Documentation

8.681.2.1 **BYTE** rxSignalStrengthListElement::radioIrf

8.681.2.2 **SHORT** rxSignalStrengthListElement::rxSignalStrength

8.682 sApnExtraParams Struct Reference

Data Fields

- [ULONG](#) apnId
- [BYTE](#) ambr_ul
- [BYTE](#) ambr_dl
- [BYTE](#) ambr_ul_ext
- [BYTE](#) ambr_dl_ext
- [BYTE](#) ambr_ul_ext2
- [BYTE](#) ambr_dl_ext2

8.682.1 Detailed Description

This structure contains the Extra Apn Params

Parameters

<i>apnId</i>	<ul style="list-style-type: none"> • APN id • ID identifying the APN that the client would like to query the AMBR params
<i>ambr_ul</i>	<ul style="list-style-type: none"> • APN AMBR uplink • APN AMBR uplink values from 1 kbps to 8640 kbps
<i>ambr_dl</i>	<ul style="list-style-type: none"> • APN AMBR downlink • APN AMBR downlink values from 1 kbps to 8640 kbps
<i>ambr_ul_ext</i>	<ul style="list-style-type: none"> • Extended APN AMBR uplink • APN AMBR uplink values from 8700 kbps to 256 Mbps
<i>ambr_dl_ext</i>	<ul style="list-style-type: none"> • Extended APN AMBR downlink • APN AMBR downlink values from 8700 kbps to 256 Mbps
<i>ambr_ul_ext2</i>	<ul style="list-style-type: none"> • Second extended APN AMBR uplink • APN AMBR uplink values from 256 Mbps to 65280 Mbps
Generated by Doxygen	
<i>ambr_dl_ext2</i>	<ul style="list-style-type: none"> • Second extended APN AMBR downlink • APN AMBR downlink values from 256 Mbps to 65280 Mbps

8.682.2 Field Documentation

8.682.2.1 **BYTE** sApnExtraParams::ambr_dl

8.682.2.2 **BYTE** sApnExtraParams::ambr_dl_ext

8.682.2.3 **BYTE** sApnExtraParams::ambr_dl_ext2

8.682.2.4 **BYTE** sApnExtraParams::ambr_ul

8.682.2.5 **BYTE** sApnExtraParams::ambr_ul_ext

8.682.2.6 **BYTE** sApnExtraParams::ambr_ul_ext2

8.682.2.7 **ULONG** sApnExtraParams::apnId

8.683 satelliteInfo Struct Reference

Data Fields

- [BYTE](#) svListLen
- [ULONG](#) validMask
- [ULONG](#) system
- [WORD](#) gnssSvid
- [BYTE](#) healthStatus
- [ULONG](#) svStatus
- [BYTE](#) svInfoMask
- [FLOAT](#) elevation
- [FLOAT](#) azimuth
- [FLOAT](#) snr

8.683.1 Detailed Description

Contain fields in struct [satelliteInfo](#)

Parameters

<i>svListLen</i>	<ul style="list-style-type: none"> • number of sets of the following elements: <ul style="list-style-type: none"> – validMask – system – gnssSvid – healthStatus – svStatus – svInfoMask – elevation – azimuth – snr

<i>validMask</i>	<ul style="list-style-type: none"> • Bitmask indicating which of the fields in this TLV are valid. Valid bitmasks: <ul style="list-style-type: none"> – 0x00000001 - VALID_SYSTEM – 0x00000002 - VALID_GNSS_SVID – 0x00000004 - VALID_HEALTH_STATUS – 0x00000008 - VALID_PROCESS_STATUS – 0x00000010 - VALID_SVINFO_MASK – 0x00000020 - VALID_ELEVATION – 0x00000040 - VALID_AZIMUTH – 0x00000080 - VALID_SNR
<i>system</i>	<ul style="list-style-type: none"> • Indicates to which constellation this SV belongs. Valid values: <ul style="list-style-type: none"> – eQMI_LOC_SV_SYSTEM_GPS (1) - GPS satellite – eQMI_LOC_SV_SYSTEM_GALILEO (2) - GALILEO satellite – eQMI_LOC_SV_SYSTEM_SBAS (3) - SBAS satellite – eQMI_LOC_SV_SYSTEM_COMPASS (4) - COMPASS satellite – eQMI_LOC_SV_SYSTEM_GLONASS (5) - GLONASS satellite – eQMI_LOC_SV_SYSTEM_BDS (6) - BDS satellite
<i>gnssSvId</i>	<ul style="list-style-type: none"> • GNSS SV ID. The GPS and GLONASS SVs can be disambiguated using the system field. Range: <ul style="list-style-type: none"> – FOR GPS: 1 to 32 – FOR GLONASS: 1 to 32 – FOR SBAS: 120 to 151 – for BDS: 201 to 237
<i>healthStatus</i>	<ul style="list-style-type: none"> • health status. Range: 0 - 1 <ul style="list-style-type: none"> – 0 - unhealthy – 1 - healthy
<i>svStatus</i>	<ul style="list-style-type: none"> • SV process status. Valid values: <ul style="list-style-type: none"> – eQMI_LOC_SV_STATUS_IDLE (1) - SV is not being actively processed – eQMI_LOC_SV_STATUS_SEARCH (2) - The system is searching for this SV – eQMI_LOC_SV_STATUS_TRACK (3) - SV is being tracked
<i>svInfoMask</i>	<ul style="list-style-type: none"> • Indicates whether almanac and ephemeris information is available. Valid bitmasks: <ul style="list-style-type: none"> – 0x01 - SVINFO_HAS_EPHEMERIS – 0x02 - SVINFO_HAS_ALMANAC
<i>elevation</i>	<ul style="list-style-type: none"> • SV elevation angle. <ul style="list-style-type: none"> – Units: Degrees – Range: 0 to 90

<i>azimuth</i>	<ul style="list-style-type: none"> • SV azimuth angle. <ul style="list-style-type: none"> – Units: Degrees – Range: 0 to 360
<i>snr</i>	<ul style="list-style-type: none"> • SV signal-to-noise ratio <ul style="list-style-type: none"> – Units: dB-Hz

Note

None

8.683.2 Field Documentation8.683.2.1 **FLOAT** `satelliteInfo::azimuth`8.683.2.2 **FLOAT** `satelliteInfo::elevation`8.683.2.3 **WORD** `satelliteInfo::gnssSvId`8.683.2.4 **BYTE** `satelliteInfo::healthStatus`8.683.2.5 **FLOAT** `satelliteInfo::snr`8.683.2.6 **BYTE** `satelliteInfo::svInfoMask`8.683.2.7 **BYTE** `satelliteInfo::svListLen`8.683.2.8 **ULONG** `satelliteInfo::svStatus`8.683.2.9 **ULONG** `satelliteInfo::system`8.683.2.10 **ULONG** `satelliteInfo::validMask`**8.684 SccRxInfo Struct Reference****Data Fields**

- [INT32](#) `rsrq`
- [SHORT](#) `snr`
- [BYTE](#) `numInstances`
- [RxSigInfo](#) `sigInfo` [255]
- [BYTE](#) `TlvPresent`

8.684.1 Detailed Description

This structure contains information about the [SccRxInfo](#) parameters.

Parameters

<i>rsrq</i>	<ul style="list-style-type: none"> • Current reference signal • Receive quality in 1/10 dB resolution
<i>snr</i>	<ul style="list-style-type: none"> • Reference signal signal-to-noise ratio in dB. • Range -10 to 30
<i>numInstances</i>	<ul style="list-style-type: none"> • Number of sets of the following <ul style="list-style-type: none"> – rxChainIndex – isRadioTuned – rxPower – rsrp
<i>sigInfo</i>	<ul style="list-style-type: none"> • See RxSigInfo for more information
<i>TlvPresent</i>	<ul style="list-style-type: none"> • Tlv Present.

8.684.2 Field Documentation

8.684.2.1 **BYTE** SccRxInfo::numInstances

8.684.2.2 **INT32** SccRxInfo::rsrq

8.684.2.3 **RxSigInfo** SccRxInfo::sigInfo[255]

8.684.2.4 **SHORT** SccRxInfo::snr

8.684.2.5 **BYTE** SccRxInfo::TlvPresent

8.685 sensorData Struct Reference

Data Fields

- [ULONG](#) timeOfFirstSample
- [BYTE](#) flags
- [BYTE](#) sensorDataLen
- [WORD](#) timeOffset [64]
- [ULONG](#) xAxis [64]
- [ULONG](#) yAxis [64]
- [ULONG](#) zAxis [64]

8.685.1 Detailed Description

This structure specifies information regarding the 3-Axis Sensor Data.

Parameters

<i>timeOfFirstSample</i>	<ul style="list-style-type: none"> • Denotes a full 32-bit time stamp of the first (oldest) sample in this message. • The time stamp is in the time reference scale that is used by the sensor time source. • Units - Milliseconds
<i>flags</i>	<ul style="list-style-type: none"> • Flags to indicate any deviation from the default measurement assumptions. • All unused bits in this field must be set to 0. • Valid bitmasks <ul style="list-style-type: none"> – 0x01 - Bitmask to specify that a sign reversal is required while interpreting the sensor data; only applies to the accelerometer samples – 0x02 - Bitmask to specify that the sensor time stamp is the same as the modem time stamp
<i>sensorDataLen</i>	<ul style="list-style-type: none"> • Number of sets of the following elements <ul style="list-style-type: none"> – timeOffset – xAxis – yAxis – zAxis
<i>timeOffset</i>	<ul style="list-style-type: none"> • Sample time offset • Units - Milliseconds
<i>xAxis</i>	<ul style="list-style-type: none"> • Sensor x-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second
<i>yAxis</i>	<ul style="list-style-type: none"> • Sensor Y-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second
<i>zAxis</i>	<ul style="list-style-type: none"> • Sensor Z-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second

8.685.2 Field Documentation

8.685.2.1 BYTE sensorData::flags

8.685.2.2 BYTE sensorData::sensorDataLen

8.685.2.3 ULONG sensorData::timeOfFirstSample

8.685.2.4 WORD sensorData::timeOffset[64]

8.685.2.5 **ULONG** sensorData::xAxis[64]8.685.2.6 **ULONG** sensorData::yAxis[64]8.685.2.7 **ULONG** sensorData::zAxis[64]

8.686 sensorData_t Struct Reference

Data Fields

- uint32_t [timeOfFirstSample](#)
- uint8_t [flags](#)
- uint8_t [sensorDataLen](#)
- uint16_t [timeOffset](#) [64]
- uint32_t [xAxis](#) [64]
- uint32_t [yAxis](#) [64]
- uint32_t [zAxis](#) [64]

8.686.1 Detailed Description

This structure specifies information regarding the 3-Axis Sensor Data. Please check has_<Param_Name> field for presence of optional parameters

Parameters

<i>timeOfFirstSample</i>	<ul style="list-style-type: none"> • Denotes a full 32-bit time stamp of the first (oldest) sample in this message. • The time stamp is in the time reference scale that is used by the sensor time source. • Units - Milliseconds
<i>flags</i>	<ul style="list-style-type: none"> • Flags to indicate any deviation from the default measurement assumptions. • All unused bits in this field must be set to 0. • Valid bitmasks <ul style="list-style-type: none"> – 0x01 - Bitmask to specify that a sign reversal is required while interpreting the sensor data; only applies to the accelerometer samples – 0x02 - Bitmask to specify that the sensor time stamp is the same as the modem time stamp
<i>sensorDataLen</i>	<ul style="list-style-type: none"> • Number of sets of the following elements <ul style="list-style-type: none"> – timeOffset – xAxis – yAxis – zAxis
<i>timeOffset</i>	<ul style="list-style-type: none"> • Sample time offset • Units - Milliseconds

<i>xAxis</i>	<ul style="list-style-type: none"> • Sensor x-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second
<i>yAxis</i>	<ul style="list-style-type: none"> • Sensor Y-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second
<i>zAxis</i>	<ul style="list-style-type: none"> • Sensor Z-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second

8.686.2 Field Documentation

8.686.2.1 `uint8_t sensorData_t::flags`

8.686.2.2 `uint8_t sensorData_t::sensorDataLen`

8.686.2.3 `uint32_t sensorData_t::timeOfFirstSample`

8.686.2.4 `uint16_t sensorData_t::timeOffset[64]`

8.686.2.5 `uint32_t sensorData_t::xAxis[64]`

8.686.2.6 `uint32_t sensorData_t::yAxis[64]`

8.686.2.7 `uint32_t sensorData_t::zAxis[64]`

8.687 sensorDataUsage_s Struct Reference

Data Fields

- [ULONG usageMask](#)
- [ULONG aidingIndicatorMask](#)

8.687.1 Detailed Description

This structure contains Sensor Data Usage info.

Parameters

<i>usageMask</i>	
	<ul style="list-style-type: none"> • Specifies which sensors were used in calculating the position in the position report.

- Value
 - 0x00000001 - Accelerometer used
 - 0x00000002 - Gyroscope used

Parameters

<i>aidingIndicatorMask</i>	
----------------------------	--

- Specifies which results were aided by sensors.

- Value
 - 0x00000001 - AIDED_HEADING
 - 0x00000002 - AIDED_SPEED
 - 0x00000004 - AIDED_POSITION
 - 0x00000008 - AIDED_VELOCITY

8.687.2 Field Documentation

8.687.2.1 **ULONG** sensorDataUsage_s::aidingIndicatorMask

8.687.2.2 **ULONG** sensorDataUsage_s::usageMask

8.688 serialNumbersInfo Struct Reference

Data Fields

- [BYTE](#) esnSize
- [CHAR](#) * pESNString
- [BYTE](#) imeiSize
- [CHAR](#) * pIMEIString
- [BYTE](#) meidSize
- [CHAR](#) * pMEIDString
- [BYTE](#) imeiSvnSize
- [CHAR](#) * plmeiSvnString

8.688.1 Detailed Description

Returns all the serial numbers assigned to the device. These serial numbers include the ESN (Electronic serial number of the device), the IMEI (International Mobile Equipment Identity) and MEID (Mobile Equipment Identifier).

Parameters

<i>esnSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the ESN string array can contain
<i>pESNString</i> [O↔ UT]	<ul style="list-style-type: none"> NULL-terminated ESN string. Empty string is returned when ESN is not supported/programmed
<i>imeiSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the IMEI string array can contain
<i>pIMEIString</i> [O↔ UT]	<ul style="list-style-type: none"> NULL terminated IMEI string. Empty string is returned when IMEI is not supported/programmed
<i>meidSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the MEID string array can contain
<i>pMEIDString</i> [↔ OUT]	<ul style="list-style-type: none"> NULL-terminated MEID string. Empty string is returned when MEID is not supported/programmed
<i>imeiSvnSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the IMEI SVN string array can contain.
<i>plmeiSvn</i> ↔ String[OUT]	<ul style="list-style-type: none"> NULL-terminated IMEI SVN string. Empty string is returned when IMEI SVN is not supported/programmed.

8.688.2 Field Documentation

8.688.2.1 BYTE serialNumbersInfo::esnSize

8.688.2.2 BYTE serialNumbersInfo::imeiSize

8.688.2.3 BYTE serialNumbersInfo::imeiSvnSize

8.688.2.4 BYTE serialNumbersInfo::meidSize

8.688.2.5 CHAR* serialNumbersInfo::pESNString

8.688.2.6 CHAR* serialNumbersInfo::pIMEIString

8.688.2.7 CHAR* serialNumbersInfo::plmeiSvnString

8.688.2.8 CHAR* serialNumbersInfo::pMEIDString

8.689 serviceProviderName Struct Reference

Data Fields

- [BYTE displayCondition](#)
- [BYTE spnLength](#)
- [BYTE spn](#) [255]

8.689.1 Detailed Description

This structure contains Service Provider Name as defined in 3GPP TS 31.102 (Section 4.2.12) from multiple sources.

Parameters

<i>displayCondition</i>	<ul style="list-style-type: none">• Display condition.
<i>spnLength</i>	<ul style="list-style-type: none">• It provides length of spn.
<i>spn</i>	<ul style="list-style-type: none">• Service provider name string must use: The SMS default 7-bit coded alphabet as defined in 3GPP TS 23.038 with bit 8 set to 9.

8.689.2 Field Documentation

8.689.2.1 **BYTE** serviceProviderName::displayCondition

8.689.2.2 **BYTE** serviceProviderName::spn[255]

8.689.2.3 **BYTE** serviceProviderName::spnLength

8.690 ServingSystemInfo Struct Reference

Data Fields

- [BYTE registrationState](#)
- [BYTE csAttachState](#)
- [BYTE psAttachState](#)
- [BYTE selectedNetwork](#)
- [BYTE radiolInterfaceNo](#)
- [BYTE radiolInterfaceList](#) [255]
- [BYTE hdrPersonality](#)

8.690.1 Detailed Description

This structure will hold the serving system parameters information

Parameters

<i>registrationState</i>	<ul style="list-style-type: none"> - Registration state of the mobile <ul style="list-style-type: none"> • 0 - QMI_NAS_NOT_REGISTERED Not registered;mobile is not currently searching for a new network to provide service • 1 - QMI_NAS_REGISTERED Registered with a network • 2 - QMI_NAS_NOT_REGISTERED_SEARCHING Not registered, but mobile is currently searching for a new network to provide service • 3 - QMI_NAS_REGISTRATION_DENIED Registration denied by the visible network • 4 - QMI_NAS_REGISTRATION_UNKNOWN Registration state is unknown
<i>csAttachState</i>	<ul style="list-style-type: none"> - Circuit Switch domain attach state of the mobile <ul style="list-style-type: none"> • 0 - Unknown or not applicable • 1 - Attached • 2 - Detached
<i>psAttachState</i>	<ul style="list-style-type: none"> - Packet domain attach state of the mobile <ul style="list-style-type: none"> • 0 - Unknown or not applicable • 1 - Attached • 2 - Detached
<i>selectedNetwork</i>	<ul style="list-style-type: none"> - Type of selected radio access network <ul style="list-style-type: none"> • 0x00 - Unknown • 0x01 - 3GPP2 network • 0x02 - 3GPP network
<i>radioInterfaceNo</i>	<ul style="list-style-type: none"> - Number of radio interfaces currently in use; this indicates how many radio_if identifiers follow this field
<i>radioInterfaceList</i>	<ul style="list-style-type: none"> - Radio interface currently in use (each is 1 byte) <ul style="list-style-type: none"> • 0x00 - None (no service) • 0x01 - cdma2000 1X • 0x02 - cdma2000 HRPD (1xEV-DO) • 0x03 - AMPS • 0x04 - GSM • 0x05 - UMTS • 0x08 - LTE
<i>hdrPersonality</i>	<ul style="list-style-type: none"> - HDR personality information (valid only for EVDO) <ul style="list-style-type: none"> • 0x00 - Unknown • 0x01 - HRPD • 0x02 - eHRPD

Note: None

8.690.2 Field Documentation

8.690.2.1 **BYTE** ServingSystemInfo::csAttachState

8.690.2.2 **BYTE** ServingSystemInfo::hdrPersonality

8.690.2.3 **BYTE** ServingSystemInfo::psAttachState

8.690.2.4 **BYTE** ServingSystemInfo::radiolInterfaceList[255]

8.690.2.5 **BYTE** ServingSystemInfo::radiolInterfaceNo

8.690.2.6 **BYTE** ServingSystemInfo::registrationState

8.690.2.7 **BYTE** ServingSystemInfo::selectedNetwork

8.691 servSystem Struct Reference

Data Fields

- [BYTE regState](#)
- [BYTE csAttachState](#)
- [BYTE psAttachState](#)
- [BYTE selNetwork](#)
- [BYTE numRadioInterfaces](#)
- [BYTE radiolInterface \[0x0A\]](#)

8.691.1 Detailed Description

This structure contains the Serving System parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>regState</i>	<ul style="list-style-type: none"> • Registration state - Registration state of the mobile • Values: <ul style="list-style-type: none"> – 0 - Not Registered; mobile is not currently searching for a new network to provide service – 1 - Registered with a network – 2 - Not registered, but mobile is currently searching for a new network to provide service – 3 - Registration denied by visible network – 4 - Registration state is unknown
<i>csAttachState</i>	<ul style="list-style-type: none"> • CS Attach State - Circuit-switched domain attach state of the mobile • Values: <ul style="list-style-type: none"> – 0 - Unknown or not applicable – 1 - Attached – 2 - Detached
Generated by Doxygen	

<i>psAttachState</i>	<ul style="list-style-type: none"> • PS Attach State - Packet-switched domain attach state of the mobile • Values: <ul style="list-style-type: none"> – 0 - Unknown or not applicable – 1 - Attached – 2 - Detached
<i>selNetwork</i>	<ul style="list-style-type: none"> • Selected Network - Type of selected radio access network • Values: <ul style="list-style-type: none"> – 0 - Unknown – 1 - 3GPP2 network – 2 - 3GPP network
<i>numRadio↔ Interfaces</i>	<ul style="list-style-type: none"> • In Use Radio Interfaces Number <ul style="list-style-type: none"> – Number of radio interfaces currently in use – defaults to zero
<i>radioInterface</i>	<ul style="list-style-type: none"> • Radio Interface(s) modem discovered • Values: <ul style="list-style-type: none"> – 0x00 - RADIO_IF_NO_SVC - None(no service) – 0x01 - RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - RADIO_IF_CDMA_1xEVDO - cdma2000 HRPD (1xEV-DO) – 0x03 - RADIO_IF_AMPS - AMPS – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS – 0x08 - RADIO_IF_LTE - LTE

8.691.2 Field Documentation

8.691.2.1 **BYTE** `servSystem::csAttachState`

8.691.2.2 **BYTE** `servSystem::numRadioInterfaces`

8.691.2.3 **BYTE** `servSystem::psAttachState`

8.691.2.4 **BYTE** `servSystem::radioInterface[0x0A]`

8.691.2.5 **BYTE** `servSystem::regState`

8.691.2.6 **BYTE** `servSystem::selNetwork`

8.692 sessionInfo Union Reference

Data Fields

- struct [omaDmFotaTlv](#) `omaDmFota`
- struct [omaDmConfigTlv](#) `omaDmConfig`
- struct [omaDmNotificationsTlv](#) `omaDmNotifications`

8.692.1 Detailed Description

This union [sessionInfo](#) consist of [omaDmFotaTlv](#), [omaDmConfigTlv](#) and [omaDmNotificationsTlv](#), out of which one will be unpacked against pEventFields.

8.692.2 Field Documentation

8.692.2.1 struct [omaDmConfigTlv](#) sessionInfo::omaDmConfig

8.692.2.2 struct [omaDmFotaTlv](#) sessionInfo::omaDmFota

8.692.2.3 struct [omaDmNotificationsTlv](#) sessionInfo::omaDmNotifications

8.693 sessionInfoExt Union Reference

Data Fields

- struct [omaDmFotaTlvExt](#) omaDmFota
- struct [omaDmConfigTlvExt](#) omaDmConfig

8.693.1 Detailed Description

This union [sessionInfo](#) consist of [omaDmFotaTlv](#) and [omaDmConfigTlv](#), out of which one will be unpacked against pEventFields.

8.693.2 Field Documentation

8.693.2.1 struct [omaDmConfigTlvExt](#) sessionInfoExt::omaDmConfig

8.693.2.2 struct [omaDmFotaTlvExt](#) sessionInfoExt::omaDmFota

8.694 sessionInfoTlv Struct Reference

Data Fields

- [BYTE](#) TlvPresent
- [ULONG](#) sessionType
- [sessionInformation](#) sessionInfo

8.694.1 Detailed Description

Structure used to store all [sessionInfo\(Union\)](#)TLV Value.

Parameters

<i>TlvPresent</i>	- Boolean indicating the presence of the TLV in the QMI response
<i>sessionType</i>	- sessiontype Value

8.694.2 Field Documentation

8.694.2.1 **sessionInformation** sessionInfoTlv::sessionInfo

8.694.2.2 **ULONG** sessionInfoTlv::sessionType

8.694.2.3 **BYTE** sessionInfoTlv::TlvPresent

8.695 sessionInfoTlvExt Struct Reference

Data Fields

- [BYTE TlvPresent](#)
- [ULONG sessionType](#)
- [sessionInformationExt sessionInfo](#)

8.695.1 Detailed Description

Structure used to store all [sessionInfo\(Union\)](#)TLV Value.

Parameters

<i>TlvPresent</i>	- Boolean indicating the presence of the TLV in the QMI response
<i>sessionType</i>	- sessiontype Value

8.695.2 Field Documentation

8.695.2.1 **sessionInformationExt** sessionInfoTlvExt::sessionInfo

8.695.2.2 **ULONG** sessionInfoTlvExt::sessionType

8.695.2.3 **BYTE** sessionInfoTlvExt::TlvPresent

8.696 SetAudioPathConfigReq Struct Reference

Data Fields

- [BYTE Profile](#)
- [BYTE * pECMode](#)
- [BYTE * pNSEnable](#)

- WORD * pTXGain
- WORD * pDTMFTXGain
- WORD * pCodecSTGain
- TXPCMIIRFiltr * pTXPCMIIRFiltr
- RXPCMIIRFiltr * pRXPCMIIRFiltr
- BYTE * pRXAVCAGCSwitch
- BYTE * pTXAVCSwitch
- RXAGCList * pRXAGCList
- RXAVCList * pRXAVCList
- TXAGCList * pTXAGCList

8.696.1 Detailed Description

This structure contains the SLQSSetAudioPathConfig request parameters.

Parameters

<i>Profile</i>	[Mandatory] <ul style="list-style-type: none"> • Audio Profile <ul style="list-style-type: none"> – 0-9
<i>pECMode</i>	[Optional] <ul style="list-style-type: none"> • AV_EC <ul style="list-style-type: none"> – 0 - Echo cancellation off – 1 - Handset echo mode – 2 - Headset mode – 3 - Car kit mode – 4 - Speaker Mode
<i>pNSEnable</i>	[Optional] <ul style="list-style-type: none"> • Noise Suppression <ul style="list-style-type: none"> – 0 - Noise suppression off – 1 - Noise suppression on
<i>pTXGain</i>	[Optional] <ul style="list-style-type: none"> • TX Voice volume <ul style="list-style-type: none"> – 0x0000 - 0xffff
<i>pDTMFTXGain</i>	[Optional] <ul style="list-style-type: none"> • AV_DTMFTXG <ul style="list-style-type: none"> – 0x0000 - 0xffff
<i>pCodecSTGain</i>	[Optional] <ul style="list-style-type: none"> • AV_CODECASTG <ul style="list-style-type: none"> – 0x0000 - 0xffff
<i>pTXPCMIIRFiltr</i>	[Optional] <ul style="list-style-type: none"> • See TXPCMIIRFiltr for more information
<i>pRXPCMIIRFiltr</i>	[Optional] <ul style="list-style-type: none"> • See RXPCMIIRFiltr for more information

<i>pRXAVCAGC↔ Switch</i>	[Optional] • RX AVC/AGC Switch
<i>pTXAVCSwitch</i>	[Optional] • TX AVC Switch
<i>pRXAGCList</i>	[Optional] • See RXAGCList for more information
<i>pRXAVCList</i>	[Optional] • See RXAVCList for more information
<i>pTXAGCList</i>	[Optional] • See TXAGCList for more information

8.696.2 Field Documentation

8.696.2.1 **WORD*** `SetAudioPathConfigReq::pCodecSTGain`

8.696.2.2 **WORD*** `SetAudioPathConfigReq::pDTMFTXGain`

8.696.2.3 **BYTE*** `SetAudioPathConfigReq::pECMode`

8.696.2.4 **BYTE*** `SetAudioPathConfigReq::pNSEnable`

8.696.2.5 **BYTE** `SetAudioPathConfigReq::Profile`

8.696.2.6 **RXAGCList*** `SetAudioPathConfigReq::pRXAGCList`

8.696.2.7 **BYTE*** `SetAudioPathConfigReq::pRXAVCAGCSwitch`

8.696.2.8 **RXAVCList*** `SetAudioPathConfigReq::pRXAVCList`

8.696.2.9 **RXPCMIIRFiltr*** `SetAudioPathConfigReq::pRXPCMIIRFiltr`

8.696.2.10 **TXAGCList*** `SetAudioPathConfigReq::pTXAGCList`

8.696.2.11 **BYTE*** `SetAudioPathConfigReq::pTXAVCSwitch`

8.696.2.12 **WORD*** `SetAudioPathConfigReq::pTXGain`

8.696.2.13 **TXPCMIIRFiltr*** `SetAudioPathConfigReq::pTXPCMIIRFiltr`

8.697 SetAudioProfileReq Struct Reference

Data Fields

- [BYTE](#) `Profile`
- [BYTE](#) `EarMute`
- [BYTE](#) `MicMute`
- [BYTE](#) `Generator`
- [BYTE](#) `Volume`

8.697.1 Detailed Description

This structure contains the SLQSSetAudioProfile request parameters.

Parameters

<i>Profile</i>	<ul style="list-style-type: none">• Audio Profile<ul style="list-style-type: none">– 0 - Handset– 1 - Headset– 2 - Car Kit– 3 - Speaker phone– 4 - Auxiliary– 5 - TTY– 6 - Auxiliary external PCM– 7 - Primary external PCM– 8 - External slave PCM– 9 - I2S
<i>EarMute</i>	<ul style="list-style-type: none">• Ear Mute Setting<ul style="list-style-type: none">– 0 - unmuted– 1 - muted
<i>MicMute</i>	<ul style="list-style-type: none">• MIC Mute Setting<ul style="list-style-type: none">– 0 - unmuted– 1 - muted
<i>Generator</i>	<ul style="list-style-type: none">• Audio Generator<ul style="list-style-type: none">– 0 - Voice– 1 - Key Beep– 2 - MIDI
<i>Volume</i>	<ul style="list-style-type: none">• Audio Volume Level<ul style="list-style-type: none">– 0 to 7

8.697.2 Field Documentation

8.697.2.1 **BYTE** SetAudioProfileReq::EarMute

8.697.2.2 **BYTE** SetAudioProfileReq::Generator

8.697.2.3 **BYTE** SetAudioProfileReq::MicMute

8.697.2.4 **BYTE** SetAudioProfileReq::Profile

8.697.2.5 **BYTE** SetAudioProfileReq::Volume

8.698 SetAudioVolTLBConfigReq Struct Reference

Data Fields

- [BYTE](#) Profile
- [BYTE](#) Generator
- [BYTE](#) Volume
- [BYTE](#) Item
- [WORD](#) VolValue

8.698.1 Detailed Description

This structure contains the SLQSSetAudioVolTLBConfig request parameters

Parameters

<i>Profile</i>	<ul style="list-style-type: none">• Audio Profile<ul style="list-style-type: none">– 0-9
<i>Generator</i>	<ul style="list-style-type: none">• Audio Generator<ul style="list-style-type: none">– 0-2
<i>Volume</i>	<ul style="list-style-type: none">• Audio Volume Level<ul style="list-style-type: none">– 0-7
<i>Item</i>	<ul style="list-style-type: none">• Item<ul style="list-style-type: none">– 13 - AV_RXVOLDB– 14 - AV_DTMFVOLDB– 15 - AV_PAD
<i>Value</i>	<ul style="list-style-type: none">• Value to be set to the volume table

8.698.2 Field Documentation

8.698.2.1 **BYTE** SetAudioVolTLBConfigReq::Generator

8.698.2.2 **BYTE** SetAudioVolTLBConfigReq::Item

8.698.2.3 **BYTE** SetAudioVolTLBConfigReq::Profile

8.698.2.4 **BYTE** SetAudioVolTLBConfigReq::Volume

8.698.2.5 **WORD** SetAudioVolTLBConfigReq::VolValue

8.699 SetAudioVolTLBConfigResp Struct Reference

Data Fields

- [WORD ResCode](#)

8.699.1 Detailed Description

This structure contains the SLQSSetAudioVolTLBConfig response parameters.

Parameters

<i>ResCode</i>	<ul style="list-style-type: none">• Result of requested item
----------------	--

8.699.2 Field Documentation

8.699.2.1 **WORD** SetAudioVolTLBConfigResp::ResCode

8.700 setCustomSettingV2 Struct Reference

Data Fields

- [CHAR cust_id](#) [64+1]
- [WORD value_length](#)
- [BYTE cust_value](#) [8+1]

8.700.1 Detailed Description

This structure contains customization settings set to modem

Parameters

<i>cust_id</i> [IN]	<ul style="list-style-type: none">• Customization ID (Maximum 64 bytes)
<i>value_length</i> [IN]	<ul style="list-style-type: none">• length of cust_value field
<i>cust_value</i> [IN]	<ul style="list-style-type: none">• Customization Setting Value (Maximum 8 bytes)

8.700.2 Field Documentation

8.700.2.1 CHAR setCustomSettingV2::cust_id[64+1]

8.700.2.2 BYTE setCustomSettingV2::cust_value[8+1]

8.700.2.3 WORD setCustomSettingV2::value_length

8.701 setDyingGaspCfg Struct Reference

Data Fields

- [BYTE](#) * [pDestSMSNum](#)
- [BYTE](#) * [pDestSMSContent](#)

8.701.1 Detailed Description

This struture contains the TLV required to get the Dysing GASP Config.

Parameters

--	--

8.701.2 Field Documentation

8.701.2.1 [BYTE](#)* setDyingGaspCfg::pDestSMSContent

8.701.2.2 [BYTE](#)* setDyingGaspCfg::pDestSMSNum

8.702 SetIMSSMSConfigReq Struct Reference

Data Fields

- [BYTE](#) * [pSMSFormat](#)
- [BYTE](#) * [pSMSOverIPNwInd](#)
- [BYTE](#) * [pPhoneCtxtURLen](#)
- [BYTE](#) * [pPhoneCtxtURI](#)

8.702.1 Detailed Description

This structure contains the SLQSSetIMSSMSConfig request parameters.

Parameters

<i>pSMSFormat</i>	<ul style="list-style-type: none"> SMS format <ul style="list-style-type: none"> 0 - 3GPP 1 - 3GPP2
<i>pSMSOverIP↔ NwInd</i>	<ul style="list-style-type: none"> SMS over IP Network Indication Flag <ul style="list-style-type: none"> TRUE - Turn on mobile-originated SMS FALSE - Turn off mobile-originated SMS
<i>pPhoneCtxtU↔ RILen</i>	<ul style="list-style-type: none"> Length of Phone context Universal Resource Identifier to follow
<i>pPhoneCtxtURI</i>	<ul style="list-style-type: none"> Phone context universal resource identifier Length of this string must be specified in pPhoneCtxtURILen parameter

8.702.2 Field Documentation

8.702.2.1 **BYTE*** SetIMSSMSConfigReq::pPhoneCtxtURI8.702.2.2 **BYTE*** SetIMSSMSConfigReq::pPhoneCtxtURILen8.702.2.3 **BYTE*** SetIMSSMSConfigReq::pSMSFormat8.702.2.4 **BYTE*** SetIMSSMSConfigReq::pSMSOverIPNwInd

8.703 SetIMSSMSConfigResp Struct Reference

Data Fields

- BYTE *** [pSettingResp](#)

8.703.1 Detailed Description

This structure contains the SLQSSetIMSSMSConfig response parameters.

Parameters

<i>pSettingResp</i>	<ul style="list-style-type: none"> Settings standard response type. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE
---------------------	---

8.703.2 Field Documentation

8.703.2.1 BYTE* SetIMSSMSConfigResp::pSettingResp

8.704 SetIMSUserConfigReq Struct Reference

Data Fields

- [BYTE * pIMSDomainLen](#)
- [BYTE * pIMSDomain](#)

8.704.1 Detailed Description

This structure contains the SLQSSetIMSUserConfig request parameters.

Parameters

<i>pIMSDomainLen</i>	<ul style="list-style-type: none">• Length of IMS Domain Name to follow
<i>pIMSDomain</i>	<ul style="list-style-type: none">• IMS domain name

8.704.2 Field Documentation

8.704.2.1 BYTE* SetIMSUserConfigReq::pIMSDomain

8.704.2.2 BYTE* SetIMSUserConfigReq::pIMSDomainLen

8.705 SetIMSUserConfigResp Struct Reference

Data Fields

- [BYTE * pSettingResp](#)

8.705.1 Detailed Description

This structure contains the SLQSSetIMSUserConfig response parameters.

Parameters

<i>pSettingResp</i>	<ul style="list-style-type: none">• Settings standard response type. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE
---------------------	---

8.705.2 Field Documentation

8.705.2.1 BYTE* SetIMSUserConfigResp::pSettingResp

8.706 SetIMSVoIPConfigReq Struct Reference

Data Fields

- WORD * pSessionExpiryTimer
- WORD * pMinSessionExpiryTimer
- BYTE * pAmrWbEnable
- BYTE * pScrAmrEnable
- BYTE * pScrAmrWbEnable
- BYTE * pAmrMode
- WORD * pAmrWBMode
- BYTE * pAmrOctetAligned
- BYTE * pAmrWBOctetAligned
- WORD * pRingingTimer
- WORD * pRingBackTimer
- WORD * pRTPRTCPInactTimer

8.706.1 Detailed Description

This structure contains the SLQSSetIMSVoIPConfig request parameters.

Parameters

<i>pSession↔ ExpiryTimer</i>	<ul style="list-style-type: none"> • Session duration, in seconds
<i>pMinSession↔ ExpiryTimer</i>	<ul style="list-style-type: none"> • Minimum allowed value for session expiry timer, in seconds
<i>pAmrWbEnable</i>	<ul style="list-style-type: none"> • Flag to enable/disable Adaptive Multirate Codec(AMR) WideBand(WB) audio • Values: <ul style="list-style-type: none"> – True - Enable – False - Disable
<i>pScrAmrEnable</i>	<ul style="list-style-type: none"> • Flag to enable/disable Source Control Rate(SCR) for AMR NarrowBand (NB) • Values: <ul style="list-style-type: none"> – True - Enable – False - Disable
<i>pScrAmrWb↔ Enable</i>	<ul style="list-style-type: none"> • Flag to enable/disable SCR for AMR WB Audio • Values: <ul style="list-style-type: none"> – True - Enable – False - Disable

<i>pAmrMode</i>	<ul style="list-style-type: none"> • BitMask for AMR NB modes allowed • Values: <ul style="list-style-type: none"> – 0x1 - 4.75 kbps – 0x2 - 5.15 kbps – 0x4 - 5.9 kbps – 0x8 - 6.17 kbps – 0x10 - 7.4 kbps – 0x20 - 7.95 kbps – 0x40 - 10.2 kbps – 0x80 - 12.2 kbps
<i>pAmrWBMode</i>	<ul style="list-style-type: none"> • BitMask for AMR WB modes allowed • Values: <ul style="list-style-type: none"> – 0x1 - 6.60 kbps – 0x2 - 8.85 kbps – 0x4 - 12.65 kbps – 0x8 - 14.25 kbps – 0x10 - 15.85 kbps – 0x20 - 18.25 kbps – 0x40 - 19.85 kbps – 0x80 - 23.05 kbps – 0x100 - 23.85 kbps
<i>pAmrOctet</i> ↔ <i>Aligned</i>	<ul style="list-style-type: none"> • Flag to indicate if the octet is aligned for AMR NB Audio • Values: <ul style="list-style-type: none"> – True - Aligned – False - Not aligned, Bandwidth Efficient mode
<i>pAmrWBOctet</i> ↔ <i>Aligned</i>	<ul style="list-style-type: none"> • Flag to indicate if the octet is aligned for AMR WB Audio • Values: <ul style="list-style-type: none"> – True - Aligned – False - Not aligned, Bandwidth Efficient mode
<i>pRingingTimer</i>	<ul style="list-style-type: none"> • Duration of ringing timer, in seconds. The ringing timer starts on the ringing event. If the call is not answered within the duration of this timer, the call is disconnected.
<i>pRingBackTimer</i>	<ul style="list-style-type: none"> • Duration of ringback timer, in seconds. The ringback timer starts on the ringback event. If the call is not answered within the duration of this timer, the call is disconnected.
<i>pRTPRTCP</i> ↔ <i>InactTimer</i>	<ul style="list-style-type: none"> • Duration of RTP/RTCP inactivity timer, in seconds. If no RTP/RTCP packet is received prior to the expiry of this timer, the call is disconnected.

8.706.2 Field Documentation

- 8.706.2.1 **BYTE*** SetIMSVoIPConfigReq::pAmrMode
- 8.706.2.2 **BYTE*** SetIMSVoIPConfigReq::pAmrOctetAligned
- 8.706.2.3 **BYTE*** SetIMSVoIPConfigReq::pAmrWbEnable
- 8.706.2.4 **WORD*** SetIMSVoIPConfigReq::pAmrWBMode
- 8.706.2.5 **BYTE*** SetIMSVoIPConfigReq::pAmrWBOctetAligned
- 8.706.2.6 **WORD*** SetIMSVoIPConfigReq::pMinSessionExpiryTimer
- 8.706.2.7 **WORD*** SetIMSVoIPConfigReq::pRingBackTimer
- 8.706.2.8 **WORD*** SetIMSVoIPConfigReq::pRingingTimer
- 8.706.2.9 **WORD*** SetIMSVoIPConfigReq::pRTPRTCPInactTimer
- 8.706.2.10 **BYTE*** SetIMSVoIPConfigReq::pScrAmrEnable
- 8.706.2.11 **BYTE*** SetIMSVoIPConfigReq::pScrAmrWbEnable
- 8.706.2.12 **WORD*** SetIMSVoIPConfigReq::pSessionExpiryTimer

8.707 SetIMSVoIPConfigResp Struct Reference

Data Fields

- **BYTE *** [pSettingResp](#)

8.707.1 Detailed Description

This structure contains the SLQSSetIMSVoIPConfig response parameters.

Parameters

<i>pSettingResp</i>	<ul style="list-style-type: none">• Settings standard response type. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE
---------------------	---

8.707.2 Field Documentation

- 8.707.2.1 **BYTE*** SetIMSVoIPConfigResp::pSettingResp

8.708 SetM2MAudioAVCFGReq Struct Reference

Data Fields

- [BYTE Profile](#)
- [BYTE Device](#)
- [BYTE PIFACEId](#)
- [PCMparams](#) * [pPCMPParams](#)

8.708.1 Detailed Description

This structure contains the SLQSSetM2MAudioAVCFG request parameters.

Parameters

<i>Profile</i>	<ul style="list-style-type: none"> • Audio Profile <ul style="list-style-type: none"> – 0-5
<i>Device</i>	<ul style="list-style-type: none"> • ACDB Device • See qaGobiApiTableSwiAudio.h for more information on ACDB Device
<i>PIFACEId</i>	<ul style="list-style-type: none"> • Physical Interface • See qaGobiApiTableSwiAudio.h for more information on physical interface
<i>pPCMPParams</i>	<ul style="list-style-type: none"> • PCM parameters • See PCMparams for more information

8.708.2 Field Documentation

8.708.2.1 **BYTE** SetM2MAudioAVCFGReq::Device

8.708.2.2 **BYTE** SetM2MAudioAVCFGReq::PIFACEId

8.708.2.3 **PCMparams*** SetM2MAudioAVCFGReq::pPCMPParams

8.708.2.4 **BYTE** SetM2MAudioAVCFGReq::Profile

8.709 SetM2MAudioLPBKReq Struct Reference

Data Fields

- [BYTE Enable](#)

8.709.1 Detailed Description

This structure contains the SLQSSetM2MAudioLPBK request parameters.

Parameters

<i>Enable</i>	<ul style="list-style-type: none">• Operation to be performed<ul style="list-style-type: none">– 0 - stop– 1 - VOCODER loop– 2 - internal codec loop
---------------	--

8.709.2 Field Documentation

8.709.2.1 BYTE SetM2MAudioLPBKReq::Enable

8.710 SetM2MAudioProfileReq Struct Reference

Data Fields

- [BYTE Profile](#)
- [BYTE * pEarMute](#)
- [BYTE * pMicMute](#)
- [BYTE * pGenerator](#)
- [BYTE * pVolume](#)
- [BYTE * pCwtMute](#)

8.710.1 Detailed Description

This structure contains the SLQSSetM2MAudioProfile request parameters.

Parameters

<i>Profile</i>	<ul style="list-style-type: none">• Audio Profile Number<ul style="list-style-type: none">– 0-5
<i>pEarMute</i>	<ul style="list-style-type: none">• Ear Mute<ul style="list-style-type: none">– 0 - mute– 1 - unmute
<i>pMicMute</i>	<ul style="list-style-type: none">• Mic Mute<ul style="list-style-type: none">– 0 - mute– 1 - unmute

<i>pGenerator</i>	<ul style="list-style-type: none"> • Generator <ul style="list-style-type: none"> – 0 - voice
<i>pVolume</i>	<ul style="list-style-type: none"> • Set RX Volume level <ul style="list-style-type: none"> – 0-5
<i>pCwtMute</i>	<ul style="list-style-type: none"> • Call Waiting Tone Mute <ul style="list-style-type: none"> – 0 - Mute – 1 - UnMute

8.710.2 Field Documentation

8.710.2.1 **BYTE*** SetM2MAudioProfileReq::pCwtMute

8.710.2.2 **BYTE*** SetM2MAudioProfileReq::pEarMute

8.710.2.3 **BYTE*** SetM2MAudioProfileReq::pGenerator

8.710.2.4 **BYTE*** SetM2MAudioProfileReq::pMicMute

8.710.2.5 **BYTE** SetM2MAudioProfileReq::Profile

8.710.2.6 **BYTE*** SetM2MAudioProfileReq::pVolume

8.711 SetM2MAudioVolumeReq Struct Reference

Data Fields

- [BYTE Profile](#)
- [BYTE Generator](#)
- [BYTE Level](#)

8.711.1 Detailed Description

This structure contains the SLQSSetM2MAudioProfile request parameters.

Parameters

<i>Profile</i>	<ul style="list-style-type: none"> • Audio Profile Number <ul style="list-style-type: none"> – 0-5
<i>Generator</i>	
	<ul style="list-style-type: none"> • Generator <ul style="list-style-type: none"> – 0 - voice
<i>Level</i>	

8.711.2 Field Documentation

8.711.2.1 **BYTE** SetM2MAudioVolumeReq::Generator

8.711.2.2 **BYTE** SetM2MAudioVolumeReq::Level

8.711.2.3 **BYTE** SetM2MAudioVolumeReq::Profile

8.712 SetM2MAVMuteReq Struct Reference

Data Fields

- [BYTE Profile](#)
- [BYTE EarMute](#)
- [BYTE MicMute](#)
- [BYTE * pCwtMute](#)

8.712.1 Detailed Description

This structure contains the SLQSSetM2MAVMute request parameters.

Parameters

<i>Profile</i>	<ul style="list-style-type: none">• Audio Profile Number<ul style="list-style-type: none">– 0-5
<i>EarMute</i>	<ul style="list-style-type: none">• Ear Mute<ul style="list-style-type: none">– 0-1
<i>MicMute</i>	<ul style="list-style-type: none">• Mic Mute<ul style="list-style-type: none">– 0-1
<i>pCwtMute</i>	[Optional] <ul style="list-style-type: none">• Call Waiting Tone Mute<ul style="list-style-type: none">– 0-1

8.712.2 Field Documentation

8.712.2.1 **BYTE** SetM2MAVMuteReq::EarMute

8.712.2.2 **BYTE** SetM2MAVMuteReq::MicMute

8.712.2.3 **BYTE*** SetM2MAVMuteReq::pCwtMute

8.712.2.4 BYTE SetM2MAVMuteReq::Profile

8.713 SetM2MSpkrGainReq Struct Reference

Data Fields

- [BYTE Profile](#)
- [WORD Value](#)

8.713.1 Detailed Description

This structure contains the SLQSSetM2MSpkrGain request parameters.

Parameters

<i>Profile</i>	<ul style="list-style-type: none">• Audio Profile Number<ul style="list-style-type: none">– 0-5
<i>Value</i>	<ul style="list-style-type: none">• RX speakerphone gain<ul style="list-style-type: none">– 0x0 - 0x7fff

8.713.2 Field Documentation

8.713.2.1 BYTE SetM2MSpkrGainReq::Profile

8.713.2.2 WORD SetM2MSpkrGainReq::Value

8.714 setPINProtection Struct Reference

Data Fields

- [BYTE pinID](#)
- [BYTE pinOperation](#)
- [BYTE pinLength](#)
- [BYTE pinValue](#) [255]

8.714.1 Detailed Description

This structure contains the information about the pin protection parameters that need to be set.

Parameters

<i>pinID</i>	<ul style="list-style-type: none">Indicates the PIN ID to be enabled or disabled.<ul style="list-style-type: none">1 - PIN1 (also called PIN)2 - PIN23 - Universal PIN4 - Hidden key
<i>pinOperation</i>	<ul style="list-style-type: none">Indicates whether the PIN is enabled or disabled.<ul style="list-style-type: none">0 - Disable the PIN1 - Enable the PIN
<i>pinLength</i>	<ul style="list-style-type: none">Length of the following elements i.e. pin value.
<i>pinValue</i> [MAX↔ _DESCRIPTI↔ ON_LENGTH]	<ul style="list-style-type: none">PIN value.This value is a sequence of ASCII characters.

8.714.2 Field Documentation

8.714.2.1 **BYTE** setPINProtection::pinID8.714.2.2 **BYTE** setPINProtection::pinLength8.714.2.3 **BYTE** setPINProtection::pinOperation8.714.2.4 **BYTE** setPINProtection::pinValue[255]

8.715 SetRegMgrConfigReq Struct Reference

Data Fields

- WORD** * pPriCSCFPort
- BYTE** * pCSCFPortNameLen
- BYTE** * pCSCFPortName
- BYTE** * pIMSTestMode

8.715.1 Detailed Description

This structure contains the SLQSSetRegMgrConfig request parameters.

Parameters

<i>pPriCSCFPort</i>	<ul style="list-style-type: none"> Primary call session control function port
<i>pCSCFPort</i> ↔ <i>NameLen</i>	<ul style="list-style-type: none"> Length of the CSCF Port name parameter to follow
<i>pCSCFPort</i> ↔ <i>Name</i>	<ul style="list-style-type: none"> Call Session control port, fully qualified domain name Length of this string must be specified in <i>pCSCFPortNameLen</i> parameter
<i>pIMSTestMode</i>	<ul style="list-style-type: none"> IMS Test mode Enabled. <ul style="list-style-type: none"> TRUE - Enable, no IMS registration FALSE - Disable, IMS registration is initiated

8.715.2 Field Documentation

8.715.2.1 **BYTE*** SetRegMgrConfigReq::pCSCFPortName8.715.2.2 **BYTE*** SetRegMgrConfigReq::pCSCFPortNameLen8.715.2.3 **BYTE*** SetRegMgrConfigReq::pIMSTestMode8.715.2.4 **WORD*** SetRegMgrConfigReq::pPriCSCFPort

8.716 SetRegMgrConfigResp Struct Reference

Data Fields

- BYTE *** [pSettingResp](#)

8.716.1 Detailed Description

This structure contains the SLQSSetRegMgrConfig response parameters.

Parameters

<i>pSettingResp</i>	<ul style="list-style-type: none"> Settings standard response type. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE
---------------------	---

8.716.2 Field Documentation

8.716.2.1 BYTE* SetRegMgrConfigResp::pSettingResp

8.717 setSignalStrengthInfo Struct Reference

Data Fields

- CDMARSSIThresh * pCDMARSSIThresh
- WORD * pCDMARSSIDelta
- CDMAECIOThresh * pCDMAECIOThresh
- WORD * pCDMAECIODelta
- HDRRSSIThresh * pHDRRSSIThresh
- WORD * pHDRRSSIDelta
- HDRECIOThresh * pHDRECIOThresh
- WORD * pHDRECIODelta
- HDRSINRThreshold * pHDRSINRThresh
- WORD * pHDRSINRDelta
- HDRIOThresh * pHDRIOThresh
- WORD * pHDRIODelta
- GSMRSSIThresh * pGSMRSSIThresh
- WORD * pGSMRSSIDelta
- WCDMARSSIThresh * pWCDMARSSIThresh
- WORD * pWCDMARSSIDelta
- WCDMAECIOThresh * pWCDMAECIOThresh
- WORD * pWCDMAECIODelta
- LTERSSIThresh * pLTERSSIThresh
- WORD * pLTERSSIDelta
- LTESNRThreshold * pLTESNRThresh
- WORD * pLTESNRDelta
- LTERSRQThresh * pLTERSRQThresh
- WORD * pLTERSRQDelta
- LTERSRPThresh * pLTERSRPThresh
- WORD * pLTERSRPDelta
- LTESigRptConfig * pLTESigRptConfig
- TDSCDMARSCPTThresh * pTDSCDMARSCPTThresh
- WORD * pTDSCDMARSCPDelta
- TDSCDMARSSIThresh * pTDSCDMARSSIThresh
- ULONG * pTDSCDMARSSIDelta
- TDSCDMAECIOThresh * pTDSCDMAECIOThresh
- ULONG * pTDSCDMAECIODelta
- TDSCDMASINRThresh * pTDSCDMASINRThresh
- ULONG * pTDSCDMASINRDelta

8.717.1 Detailed Description

This structure contains the Signal Strength reporting thresholds Item information.

Parameters

<i>pCDMARSSI</i> <i>Thresh</i>	<ul style="list-style-type: none"> CDMA RSSI threshold List See CDMARSSIThresh for more details
<i>pCDMARSSI</i> <i>Delta</i>	<ul style="list-style-type: none"> RSSI delta (in units of 0.1 dBm). A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pCDMAECIO</i> <i>Thresh</i>	<ul style="list-style-type: none"> CDMA ECIO Threshold List See CDMAECIOThresh for more details
<i>pCDMAECIO</i> <i>Delta</i>	<ul style="list-style-type: none"> ECIO delta (in units of 0.1 dB). A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pHRRSSI</i> <i>Thresh</i>	<ul style="list-style-type: none"> HDR RSSI Threshold List See HRRSSIThresh for more details
<i>pHRRSSIDelta</i>	<ul style="list-style-type: none"> RSSI delta (in units of 0.1 dBm) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pHDRECIOThresh</i>	<ul style="list-style-type: none"> HDR ECIO Threshold List See HDRECIOThresh for more details
<i>pHDRECIODelta</i>	<ul style="list-style-type: none"> ECIO delta (in units of 0.1 dB) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pHRSINR</i> <i>Thresh</i>	<ul style="list-style-type: none"> HDR SINR Threshold List See HRSINRThreshold for more details
<i>pHRSINRDelta</i>	<ul style="list-style-type: none"> SINR delta (in units of 1 SINR level) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pHDRIOThresh</i>	<ul style="list-style-type: none"> HDR IO Threshold List See HDRIOThresh for more details
<i>pHDRIODelta</i>	<ul style="list-style-type: none"> IO delta (in units of 0.1 dBm) A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pGSMRSSI</i> <i>Thresh</i>	<ul style="list-style-type: none"> GSM RSSI Threshold List See GSMRSSIThresh for more details

<i>pGSMRSSIDelta</i>	<ul style="list-style-type: none"> • RSSI delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pWCDMARSS</i> \leftrightarrow <i>IThresh</i>	<ul style="list-style-type: none"> • WCDMA RSSI Threshold List • See WCDMARSSIThresh for more details
<i>pWCDMARSS</i> \leftrightarrow <i>IDelta</i>	<ul style="list-style-type: none"> • RSSI delta (in units of 0.1 dBm). • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pWCDMAECI</i> \leftrightarrow <i>OThresh</i>	<ul style="list-style-type: none"> • WCDMA ECIO Threshold List • See WCDMAECIOThresh for more details
<i>pWCDMAECI</i> \leftrightarrow <i>ODelta</i>	<ul style="list-style-type: none"> • ECIO delta (in units of 0.1 dB) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pLTERSSI</i> \leftrightarrow <i>Thresh</i>	<ul style="list-style-type: none"> • LTE RSSI Threshold List • See LTERSSIThresh for more details
<i>pLTERSSIDelta</i>	<ul style="list-style-type: none"> • RSSI delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pLTESNR</i> \leftrightarrow <i>Thresh</i>	<ul style="list-style-type: none"> • LTE SNR Threshold List • See LTESNRThreshold for more details
<i>pLTESNRDelta</i>	<ul style="list-style-type: none"> • SNR delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pLTERSRQ</i> \leftrightarrow <i>Thresh</i>	<ul style="list-style-type: none"> • LTE RSRQ Threshold List • See LTERSRQThresh for more details
<i>pLTERSRQ</i> \leftrightarrow <i>Delta</i>	<ul style="list-style-type: none"> • RSRQ delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pLTERSRP</i> \leftrightarrow <i>Thresh</i>	<ul style="list-style-type: none"> • LTE RSRP Threshold List • See LTERSRPThresh for more details
<i>pLTERSRPDelta</i>	<ul style="list-style-type: none"> • RSRP delta (in units of 0.1 dBm). • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.

<i>pLTESigRpt</i> ↔ <i>Config</i>	<ul style="list-style-type: none"> • LTE Signal Report Config • See LTESigRptConfig for more details
<i>pTDSCDMAR</i> ↔ <i>SCPThresh</i>	<ul style="list-style-type: none"> • TDSCDMA RSCP Threshold List • See TDSCDMARSCPThresh for more details
<i>pTDSCDMAR</i> ↔ <i>SCPDelta</i>	<ul style="list-style-type: none"> • RSCP delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
<i>pTDSCDMAR</i> ↔ <i>SSIThresh</i>	<ul style="list-style-type: none"> • TDSCDMA RSSI Threshold List • See TDSCDMARSSIThresh for more details
<i>pTDSCDMAR</i> ↔ <i>SSIDelta</i>	<ul style="list-style-type: none"> • RSSI delta (in dBm) used by TD-SCDMA.
<i>pTDSCDMAE</i> ↔ <i>CIOThresh</i>	<ul style="list-style-type: none"> • TDSCDMA ECIO Threshold List • See TDSCDMAECIOThresh for more details
<i>pTDSCDMAE</i> ↔ <i>CIODelta</i>	<ul style="list-style-type: none"> • ECIO delta (in dB) used by TD-SCDMA
<i>pTDSCDMASI</i> ↔ <i>NRThresh</i>	<ul style="list-style-type: none"> • TDSCDMA SINR Threshold List • See TDSCDMASINRThresh for more details
<i>pTDSCDMASI</i> ↔ <i>NRDelta</i>	<ul style="list-style-type: none"> • SINR delta (in dB) used by TD-SCDMA.

8.717.2 Field Documentation

8.717.2.1 **WORD*** `setSignalStrengthInfo::pCDMAECIODelta`

8.717.2.2 **CDMAECIOThresh*** `setSignalStrengthInfo::pCDMAECIOThresh`

8.717.2.3 **WORD*** `setSignalStrengthInfo::pCDMARSSIDelta`

8.717.2.4 **CDMARSSIThresh*** `setSignalStrengthInfo::pCDMARSSIThresh`

8.717.2.5 **WORD*** `setSignalStrengthInfo::pGSMRSSIDelta`

8.717.2.6 **GSMRSSIThresh*** `setSignalStrengthInfo::pGSMRSSIThresh`

8.717.2.7 **WORD*** `setSignalStrengthInfo::pHDRECIODelta`

- 8.717.2.8 HDRECIOTresh* setSignalStrengthInfo::pHDRECIOTresh
- 8.717.2.9 WORD* setSignalStrengthInfo::pHDRIODelta
- 8.717.2.10 HDRIOTresh* setSignalStrengthInfo::pHDRIOTresh
- 8.717.2.11 WORD* setSignalStrengthInfo::pHRRSSIDelta
- 8.717.2.12 HRRSSITresh* setSignalStrengthInfo::pHRRSSITresh
- 8.717.2.13 WORD* setSignalStrengthInfo::pHRSINRDelta
- 8.717.2.14 HRSINRThreshold* setSignalStrengthInfo::pHRSINRThresh
- 8.717.2.15 WORD* setSignalStrengthInfo::pLTERSRPDelta
- 8.717.2.16 LTERSRPThresh* setSignalStrengthInfo::pLTERSRPThresh
- 8.717.2.17 WORD* setSignalStrengthInfo::pLTERSRQDelta
- 8.717.2.18 LTERSRQThresh* setSignalStrengthInfo::pLTERSRQThresh
- 8.717.2.19 WORD* setSignalStrengthInfo::pLTERSSIDelta
- 8.717.2.20 LTERSSITresh* setSignalStrengthInfo::pLTERSSITresh
- 8.717.2.21 LTESigRptConfig* setSignalStrengthInfo::pLTESigRptConfig
- 8.717.2.22 WORD* setSignalStrengthInfo::pLTESNRDelta
- 8.717.2.23 LTESNRThreshold* setSignalStrengthInfo::pLTESNRThresh
- 8.717.2.24 ULONG* setSignalStrengthInfo::pTDSCDMAECIODelta
- 8.717.2.25 TDSCDMAECIOTresh* setSignalStrengthInfo::pTDSCDMAECIOTresh
- 8.717.2.26 WORD* setSignalStrengthInfo::pTDSCDMARSCPDelta
- 8.717.2.27 TDSCDMARSCPThresh* setSignalStrengthInfo::pTDSCDMARSCPThresh
- 8.717.2.28 ULONG* setSignalStrengthInfo::pTDSCDMARSSIDelta
- 8.717.2.29 TDSCDMARSSITresh* setSignalStrengthInfo::pTDSCDMARSSITresh
- 8.717.2.30 ULONG* setSignalStrengthInfo::pTDSCDMASINRDelta

8.717.2.31 **TDSCDMASINRThresh*** **setSignalStrengthInfo::pTDSCDMASINRThresh**

8.717.2.32 **WORD*** **setSignalStrengthInfo::pWCMAECIODelta**

8.717.2.33 **WCMAECIOThresh*** **setSignalStrengthInfo::pWCMAECIOThresh**

8.717.2.34 **WORD*** **setSignalStrengthInfo::pWCDMARSSIDelta**

8.717.2.35 **WCDMARSSIThresh*** **setSignalStrengthInfo::pWCDMARSSIThresh**

8.718 SetSIPConfigReq Struct Reference

Data Fields

- **WORD** * **pSIPLocalPort**
- **ULONG** * **pTimerSIPReg**
- **ULONG** * **pSubscribeTimer**
- **ULONG** * **pTimerT1**
- **ULONG** * **pTimerT2**
- **ULONG** * **pTimerTf**
- **BYTE** * **pSigCompEnabled**

8.718.1 Detailed Description

This structure contains the SLQSSetSIPConfig request parameters.

Parameters

<i>pSIPLocalPort</i>	<ul style="list-style-type: none"> Primary call session control function SIP port number
<i>pTimerSIPReg</i>	<ul style="list-style-type: none"> Initial SIP registration duration from the User equipment, in seconds
<i>pSubscribeTimer</i>	<ul style="list-style-type: none"> Duration of the subscription by the UE for IMS registration notifications, in seconds
<i>pTimerT1</i>	<ul style="list-style-type: none"> RTT estimate, in milliseconds
<i>pTimerT2</i>	<ul style="list-style-type: none"> The maximum retransmit interval for non-invite requests and invite responses, in milliseconds
<i>pTimerTf</i>	<ul style="list-style-type: none"> Non-invite transaction timeout timer, in milliseconds
<i>pSigCompEnabled</i>	<ul style="list-style-type: none"> Sig Comp Status <ul style="list-style-type: none"> TRUE - Enable FALSE - Disable

8.718.2 Field Documentation

8.718.2.1 **BYTE*** SetSIPConfigReq::pSigCompEnabled

8.718.2.2 **WORD*** SetSIPConfigReq::pSIPLocalPort

8.718.2.3 **ULONG*** SetSIPConfigReq::pSubscribeTimer

8.718.2.4 **ULONG*** SetSIPConfigReq::pTimerSIPReg

8.718.2.5 **ULONG*** SetSIPConfigReq::pTimerT1

8.718.2.6 **ULONG*** SetSIPConfigReq::pTimerT2

8.718.2.7 **ULONG*** SetSIPConfigReq::pTimerTf

8.719 SetSIPConfigResp Struct Reference

Data Fields

- **BYTE *** [pSettingResp](#)

8.719.1 Detailed Description

This structure contains the SLQSSetSIPConfig response parameters.

Parameters

<i>pSettingResp</i>	<ul style="list-style-type: none">• Settings standard response type. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE
---------------------	---

8.719.2 Field Documentation

8.719.2.1 **BYTE*** SetSIPConfigResp::pSettingResp

8.720 sGetDeviceSeriesResult Struct Reference

Data Fields

- enum [eGobiDeviceSeries](#) [eDevice](#)
- **ULONG** [uResult](#)

8.720.1 Detailed Description

This structure contains the Device Series

Parameters

<i>eGobiDeviceSeries</i>	<ul style="list-style-type: none">The number of device in the device series
<i>uResult</i>	-eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

8.720.2 Field Documentation

8.720.2.1 enum eGobiDeviceSeries sGetDeviceSeriesResult::eDevice

8.720.2.2 ULONG sGetDeviceSeriesResult::uResult

8.721 sidNid Struct Reference

Data Fields

- [WORD nid](#)
- [WORD sid](#)

8.721.1 Detailed Description

This structure contains the parameters for SidNid Information

Parameters

<i>nid</i>	<ul style="list-style-type: none">Network ID
<i>sid</i>	<ul style="list-style-type: none">System ID

8.721.2 Field Documentation

8.721.2.1 WORD sidNid::nid

8.721.2.2 WORD sidNid::sid

8.722 sigInfo Struct Reference

Data Fields

- [RSSIThresh](#) * [pRSSIThresh](#)

- [ECIOThresh](#) * [pECIOThresh](#)
- [HDRSINRThresh](#) * [pHDRSINRThresh](#)
- [LTESNRThresh](#) * [pLTESNRThresh](#)
- [IOThresh](#) * [pIOThresh](#)
- [RSRQThresh](#) * [pRSRQThresh](#)
- [RSRPThresh](#) * [pRSRPThresh](#)
- [LTESigRptCfg](#) * [pLTESigRptCfg](#)
- [TDSCDMASINRCONFTThresh](#) * [pTDSCDMASINRCONFTThresh](#)

8.722.1 Detailed Description

This structure contains the 3gpp Configuration Item information.

Parameters

<i>pRSSIThresh</i>	<ul style="list-style-type: none"> • RSSI threshold List • See RSSIThresh for more details
<i>pECIOThresh</i>	<ul style="list-style-type: none"> • ECIO Threshold List • See ECIOThresh for more details
<i>pHDRSINR_↔Thresh</i>	<ul style="list-style-type: none"> • HDR SINR Threshold List • See HDRSINRThresh for more details
<i>pLTESNR_↔Thresh</i>	<ul style="list-style-type: none"> • LTE SNR Threshold List • See LTESNRThresh for more details
<i>pIOThresh</i>	<ul style="list-style-type: none"> • IO Threshold List • See IOThresh for more details
<i>pRSRQThresh</i>	<ul style="list-style-type: none"> • RSRQ Threshold List • See RSRQThresh for more details
<i>pRSRPThresh</i>	<ul style="list-style-type: none"> • RSRP Threshold List • See RSRPThresh for more details
<i>pLTESigRptCfg</i>	<ul style="list-style-type: none"> • LTE signal report config • See LTESigRptCfg for more details
<i>pTDSCDMASINR_↔NRCONFTThresh</i>	<ul style="list-style-type: none"> • TD-SCDMA SINR Threshold List • See TDSCDMASINRCONFTThresh for more details

8.722.2 Field Documentation

8.722.2.1 **ECIOThresh*** sigInfo::pECIOThresh

8.722.2.2 **HDRSINRThresh*** sigInfo::pHDRSINRThresh

8.722.2.3 **IOThresh*** sigInfo::pIOThresh

8.722.2.4 **LTESigRptCfg*** sigInfo::pLTESigRptCfg

8.722.2.5 **LTESNRThresh*** sigInfo::pLTESNRThresh

8.722.2.6 **RSRPThresh*** sigInfo::pRSRPThresh

8.722.2.7 **RSRQThresh*** sigInfo::pRSRQThresh

8.722.2.8 **RSSIThresh*** sigInfo::pRSSIThresh

8.722.2.9 **TDSCDMASINRCONFTthresh*** sigInfo::pTDSCDMASINRCONFTthresh

8.723 signalInfo Struct Reference

Data Fields

- [BYTE signalType](#)
- [BYTE alertPitch](#)
- [BYTE signal](#)

8.723.1 Detailed Description

This structure contains Signal Information

Parameters

<i>signalType</i>	<ul style="list-style-type: none"> • Call identifier for the call.
<i>alertPitch</i>	<ul style="list-style-type: none"> • Signal Information
<i>signal</i>	<ul style="list-style-type: none"> • Caller ID Information

8.723.2 Field Documentation

8.723.2.1 BYTE signalInfo::alertPitch

8.723.2.2 BYTE signalInfo::signal

8.723.2.3 BYTE signalInfo::signalType

8.724 SignalStrengthDataType Struct Reference

Data Fields

- [BYTE thresholdsSize](#)
- [INT8 thresholds](#) [5]

8.724.1 Field Documentation

8.724.1.1 INT8 SignalStrengthDataType::thresholds[5]

8.724.1.2 BYTE SignalStrengthDataType::thresholdsSize

8.725 slot_t Struct Reference

Data Fields

- [uint32_t uPhyCardStatus](#)
- [uint32_t uPhySlotStatus](#)
- [uint8_t bLogicalSlot](#)
- [uint8_t bICCIDLength](#)
- [uint8_t bICCID](#) [255]

8.725.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

<i>uPhyCardStatus</i>	<ul style="list-style-type: none">• State of the card in the Pyhsical Slot Status.<ul style="list-style-type: none">– 0x00 - Unknown.– 0x01 - Absent.– 0x02 - Present.
<i>uPhySlotStatus</i>	<ul style="list-style-type: none">• State of the Physical Slot status.<ul style="list-style-type: none">– 0x00 Inactive.– 0x01 Activate.

<i>bLogicalSlot</i>	<ul style="list-style-type: none"> • Logical Slot associated with this physical slot. This is valid if the physical slot is active. <ul style="list-style-type: none"> – 1 - Slot 1. – 2 - Slot 2. – 3 - Slot 3. – 4 - Slot 4. – 5 - Slot 5.
<i>bLogicalSlot</i>	<ul style="list-style-type: none"> • Number of sets the sets of ICCID
<i>bICCID[MAX_ICCID_LENGTH]</i>	<ul style="list-style-type: none"> • Contains the ICCID of the card in the physical slot.

8.725.2 Field Documentation

8.725.2.1 `uint8_t slot_t::bICCID[255]`

8.725.2.2 `uint8_t slot_t::bICCIDLength`

8.725.2.3 `uint8_t slot_t::bLogicalSlot`

8.725.2.4 `uint32_t slot_t::uPhyCardStatus`

8.725.2.5 `uint32_t slot_t::uPhySlotStatus`

8.726 slotInf Struct Reference

Data Fields

- `uint8_t cardState`
- `uint8_t upinState`
- `uint8_t upinRetries`
- `uint8_t upukRetries`
- `uint8_t errorState`
- `uint8_t numApp`
- `appStats AppStatus [10]`

8.726.1 Detailed Description

This structure contains information about the SLOTS present.

Parameters

<i>cardState</i>	<ul style="list-style-type: none"> Indicates the state of the card for each slot. <ul style="list-style-type: none"> 0 - Absent 1 - Present 2 - Error
<i>upinState</i>	<ul style="list-style-type: none"> Indicates the state of UPIN. <ul style="list-style-type: none"> 0 - Unknown 1 - Enabled and not verified 2 - Enabled and verified 3 - Disabled 4 - Blocked 5 - Permanently blocked 0xFF - Not Available
<i>upinRetries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to verify the UPIN. If 0xFF, information not available.
<i>upukRetries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to unblock the UPIN. If 0xFF, information not available.
<i>errorState</i>	<ul style="list-style-type: none"> Indicates the reason for the card error, and is valid only when the card state is Error <ul style="list-style-type: none"> 0 - Unknown 1 - Power down 2 - Poll error 3 - No ATR received 4 - Volt mismatch 5 - Parity error 6 - Unknown; possibly removed 7 - Card returned technical problems 0xFF - Not Available Other values are possible and reserved for future use. When an unknown value is received, it is to be handled as "Unknown".
<i>numApp</i>	<ul style="list-style-type: none"> Indicates the number of applications available on the card. The following block is repeated for each application. i.e. AppStatus. If zero(0) then no AppStatus information exists.
<i>AppStatus[MA↔ X_NO_OF_AP↔ PLICATIONS]</i>	<ul style="list-style-type: none"> See appStats for more information.

8.726.2 Field Documentation

8.726.2.1 `appStats slotInf::AppStatus[10]`

8.726.2.2 `uint8_t slotInf::cardState`

8.726.2.3 `uint8_t slotInf::errorState`

8.726.2.4 `uint8_t slotInf::numApp`

8.726.2.5 `uint8_t slotInf::upinRetries`

8.726.2.6 `uint8_t slotInf::upinState`

8.726.2.7 `uint8_t slotInf::upukRetries`

8.727 slotInfo Struct Reference

Data Fields

- [BYTE cardState](#)
- [BYTE upinState](#)
- [BYTE upinRetries](#)
- [BYTE upukRetries](#)
- [BYTE errorState](#)
- [BYTE numApp](#)
- [appStatus AppStatus \[10\]](#)

8.727.1 Detailed Description

This structure contains information about the SLOTS present.

Parameters

<i>cardState</i>	<ul style="list-style-type: none"> • Indicates the state of the card for each slot. <ul style="list-style-type: none"> – 0 - Absent – 1 - Present – 2 - Error
<i>upinState</i>	<ul style="list-style-type: none"> • Indicates the state of UPIN. <ul style="list-style-type: none"> – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked
	<ul style="list-style-type: none"> – 0xFF - Not Available

<i>upinRetries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to verify the UPIN. If 0xFF, information not available.
<i>upukRetries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to unblock the UPIN. If 0xFF, information not available.
<i>errorState</i>	<ul style="list-style-type: none"> Indicates the reason for the card error, and is valid only when the card state is Error <ul style="list-style-type: none"> 0 - Unknown 1 - Power down 2 - Poll error 3 - No ATR received 4 - Volt mismatch 5 - Parity error 6 - Unknown; possibly removed 7 - Card returned technical problems 0xFF - Not Available Other values are possible and reserved for future use. When an unknown value is received, it is to be handled as "Unknown".
<i>numApp</i>	<ul style="list-style-type: none"> Indicates the number of applications available on the card. The following block is repeated for each application. i.e. AppStatus. If zero(0) then no AppStatus information exists.
<i>AppStatus[MA↔X_NO_OF_AP↔PLICATIONS]</i>	<ul style="list-style-type: none"> See appStatus for more information.

8.727.2 Field Documentation

8.727.2.1 **appStatus** slotInfo::AppStatus[10]

8.727.2.2 **BYTE** slotInfo::cardState

8.727.2.3 **BYTE** slotInfo::errorState

8.727.2.4 **BYTE** slotInfo::numApp

8.727.2.5 **BYTE** slotInfo::upinRetries

8.727.2.6 **BYTE** slotInfo::upinState

8.727.2.7 **BYTE** slotInfo::upukRetries

8.728 slots_t Struct Reference

Data Fields

- [slot_t uimSlotStatus](#) [255]

8.728.1 Field Documentation

8.728.1.1 `slot_t slots_t::uimSlotStatus[255]`

8.729 slqsautoconnect Struct Reference

Data Fields

- [BOOL action](#)
- [BYTE acsetting](#)
- [BYTE acroamsetting](#)

8.729.1 Detailed Description

structure contains autoconnect settings parameters

Parameters

<i>action</i>	<ul style="list-style-type: none">• 0 - get autoconnect settings• 1 - set autoconnect settings
<i>acsetting</i>	<ul style="list-style-type: none">• Current autoconnect setting:<ul style="list-style-type: none">– 0x00 - Autoconnect disabled– 0x01 - Autoconnect enabled– 0x02 - Autoconnect paused (resume on powercycle)
<i>acroamsetting</i>	<ul style="list-style-type: none">• Current autoconnect roaming status<ul style="list-style-type: none">– 0x00 - Autoconnect always allowed– 0x01 - Autoconnect while in home service area only

8.729.2 Field Documentation

8.729.2.1 **BYTE** `slqsautoconnect::acroamsetting`

8.729.2.2 **BYTE** `slqsautoconnect::acsetting`

8.729.2.3 **BOOL** `slqsautoconnect::action`

8.730 SLQSDeleteProfileParams Struct Reference

Data Fields

- [BYTE profileType](#)
- [BYTE profileIndex](#)

8.730.1 Detailed Description

This structure contains the information about the profile to be deleted.

Parameters

<i>profileType</i>	<ul style="list-style-type: none">Identifies the type of profile<ul style="list-style-type: none">0x00 – 3GPPNote: Deletion of 3GPP2 profiles is not supported.
<i>profileIndex</i>	<ul style="list-style-type: none">Index of the configured profile to be deleted<ul style="list-style-type: none">Value between 1 - 16 (EM/MC73xx or earlier)Value between 1 - 24 (EM/MC74xx onwards)

8.730.2 Field Documentation

8.730.2.1 BYTE SLQSDeleteProfileParams::profileIndex

8.730.2.2 BYTE SLQSDeleteProfileParams::profileType

8.731 slqsfwinfo_s Struct Reference

Data Fields

- CHAR [modelid_str](#) [20]
- CHAR [bootversion_str](#) [85]
- CHAR [appversion_str](#) [85]
- CHAR [sku_str](#) [15]
- CHAR [packageid_str](#) [85]
- CHAR [carrier_str](#) [20]
- CHAR [priversion_str](#) [16]
- CHAR [cur_carr_name](#) [17]
- CHAR [cur_carr_rev](#) [13]

8.731.1 Detailed Description

SPKG CWE firmware image info structure

Parameters

<i>modelid_str</i>	<ul style="list-style-type: none"> device model identifier string
<i>bootversion_str</i>	<ul style="list-style-type: none"> firmware boot version string
<i>appversion_str</i>	<ul style="list-style-type: none"> firmware application version string
<i>sku_str</i>	<ul style="list-style-type: none"> SKU(PRI) string
<i>packageid_str</i>	<ul style="list-style-type: none"> package identifier string deprecated on EM/MC74xx(9x30) devices
<i>carrier_str</i>	<ul style="list-style-type: none"> carrier string See qaGobiApiTableCarrierCodes.h for carrier codes
<i>priversion_str</i>	<ul style="list-style-type: none"> PRI version string
<i>cur_carr_name</i>	<ul style="list-style-type: none"> Current PRI Carrier Name
<i>cur_carr_rev</i>	<ul style="list-style-type: none"> Current PRI Carrier Revision

8.731.2 Field Documentation

8.731.2.1 CHAR slqsfwinfo_s::appversion_str[85]

8.731.2.2 CHAR slqsfwinfo_s::bootversion_str[85]

8.731.2.3 CHAR slqsfwinfo_s::carrier_str[20]

8.731.2.4 CHAR slqsfwinfo_s::cur_carr_name[17]

8.731.2.5 CHAR slqsfwinfo_s::cur_carr_rev[13]

8.731.2.6 CHAR slqsfwinfo_s::modelid_str[20]

8.731.2.7 CHAR slqsfwinfo_s::packageid_str[85]

8.731.2.8 CHAR slqsfwinfo_s::priversion_str[16]

8.731.2.9 CHAR slqsfwinfo_s::sku_str[15]

8.732 SlqsNas3GppNetworkInfo Struct Reference

Data Fields

- WORD MCC
- WORD MNC
- ULONG InUse
- ULONG Roaming
- ULONG Forbidden
- ULONG Preferred
- CHAR Description [255]

8.732.1 Detailed Description

Contain the 3GPP network information.

Parameters

<i>MCC</i>	<ul style="list-style-type: none"> • Mobile Country Code
<i>MNC</i>	<ul style="list-style-type: none"> • Mobile Network Code
<i>InUse</i>	<ul style="list-style-type: none"> • Is the Network the current serving Network <ul style="list-style-type: none"> – 0 - Unknown – 1 - Current serving network – 2 - Not current serving network, available
<i>Roaming</i>	<ul style="list-style-type: none"> • Home/Roam Status of the Network <ul style="list-style-type: none"> – 0 - Unknown – 1 - Home – 2 - Roam
<i>Forbidden</i>	<ul style="list-style-type: none"> • Is the Network in the forbidden network list <ul style="list-style-type: none"> – 0 - Unknown – 1 - Forbidden – 2 - Not Forbidden
<i>Preferred</i>	<ul style="list-style-type: none"> • Is the Network in the Preferred network list <ul style="list-style-type: none"> – 0 - Unknown – 1 - Preferred – 2 - Not Preferred
<i>Description</i>	<ul style="list-style-type: none"> • Network Name/Description
Generated by Doxygen	<ul style="list-style-type: none"> • This is a NULL terminated string.

8.732.2 Field Documentation

8.732.2.1 CHAR SIqsNas3GppNetworkInfo::Description[255]

8.732.2.2 ULONG SIqsNas3GppNetworkInfo::Forbidden

8.732.2.3 ULONG SIqsNas3GppNetworkInfo::InUse

8.732.2.4 WORD SIqsNas3GppNetworkInfo::MCC

8.732.2.5 WORD SIqsNas3GppNetworkInfo::MNC

8.732.2.6 ULONG SIqsNas3GppNetworkInfo::Preferred

8.732.2.7 ULONG SIqsNas3GppNetworkInfo::Roaming

8.733 SIqsNasPcsDigit Struct Reference

Data Fields

- [WORD MCC](#)
- [WORD MNC](#)
- [BYTE includes_pcs_digit](#)

8.733.1 Detailed Description

Contain the PCS Digit information

Parameters

<i>MCC</i>	<ul style="list-style-type: none"> • Mobile Country Code
<i>MNC</i>	<ul style="list-style-type: none"> • Mobile Network Code
<i>includes_pcs_↔ digit</i>	<ul style="list-style-type: none"> • this field is use to interpret the length of corresponding MNC reported • 0x01 - MNC is a three-digit value • 0x00 - MNC is a two-digit value

8.733.2 Field Documentation

8.733.2.1 BYTE SIqsNasPcsDigit::includes_pcs_digit

8.733.2.2 WORD SlqsNasPcsDigit::MCC

8.733.2.3 WORD SlqsNasPcsDigit::MNC

8.734 slqssendasyncsmsparams_s Struct Reference

Data Fields

- ULONG messageFormat
- ULONG messageSize
- BYTE * pMessage
- BYTE * pForceOnDC
- BYTE * pServiceOption
- BYTE * pFollowOnDC
- BYTE * pLinktimer
- BYTE * pSmsOnIms
- BYTE * pRetryMessage
- ULONG * pRetryMessageld
- ULONG * pUserData

8.734.1 Detailed Description

This structure contains SMS parameters

Parameters

<i>messageFormat</i>	<ul style="list-style-type: none"> • Message format • Values: <ul style="list-style-type: none"> – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
<i>messageSize</i>	<ul style="list-style-type: none"> • The length of the message contents in bytes
<i>pMessage</i>	<ul style="list-style-type: none"> • The message contents
<i>pForceOnDC</i>	<ul style="list-style-type: none"> • Force the message to be sent on the CDMA dedicated channel. • Values: <ul style="list-style-type: none"> – 0x00 - Do not care about the channel on which the message is sent – 0x01 - Request to send the message over the dedicated channel
<i>pServiceOption</i>	<ul style="list-style-type: none"> • Service option: • Values: <ul style="list-style-type: none"> – 0x00 - SO_AUTO - AUTO (choose the best service option) – 0x06 - SO_6 - Service option 6 – 0x0E - SO_14 - Service option 14

<i>pFollowOnDC</i>	<ul style="list-style-type: none"> • Flag to request not to disconnect the CDMA dedicated channel after the send operation is complete. • This TLV can be included if more messages are expected to follow. • Values: <ul style="list-style-type: none"> – 0x01 - FOLLOW_ON_DC_ON - On (don't disconnect after send operation) Any value other than 0x01 is treated as an absence of this TLV.
<i>pLinktimer</i>	<ul style="list-style-type: none"> • Keeps the GW SMS link open for the specified number of seconds; can be enabled if more messages are expected to follow
<i>pSmsOnIms</i>	<ul style="list-style-type: none"> • Indicates whether the message is to be sent on IMS. • Values: <ul style="list-style-type: none"> – 0x00 - Message is not to be sent on IMS – 0x01 - Message is to be sent on IMS – 0x02 to 0xFF - Reserved
<i>pRetryMessage</i>	<ul style="list-style-type: none"> • Indicates this message is a retry message. • Values: <ul style="list-style-type: none"> – 0x01 - WMS_MESSAGE_IS_A_RETRY - Message is a retry message Note: Any value other than 0x01 in this field is treated as an absence of this TLV.
<i>pRetry↔ MessageId</i>	<ul style="list-style-type: none"> • Message ID to be used in the retry message. • The message ID specified here is used instead of the message ID encoded in the raw message.
<i>pUserData</i>	<ul style="list-style-type: none"> • Enables the control point to associate the request with the corresponding indication. • The control point might send numerous requests. • This TLV will help the control point to identify the request for which the received indication belongs.

8.734.2 Field Documentation

8.734.2.1 **ULONG** `slqssendasyncsmsparams_s::messageFormat`

8.734.2.2 **ULONG** `slqssendasyncsmsparams_s::messageSize`

8.734.2.3 **BYTE*** `slqssendasyncsmsparams_s::pFollowOnDC`

8.734.2.4 **BYTE*** `slqssendasyncsmsparams_s::pForceOnDC`

8.734.2.5 **BYTE*** `slqssendasyncsmsparams_s::pLinktimer`

8.734.2.6 **BYTE*** `slqssendasyncsmsparams_s::pMessage`

8.734.2.7 **BYTE*** slqssendasyncsmsparams_s::pRetryMessage

8.734.2.8 **ULONG*** slqssendasyncsmsparams_s::pRetryMessageId

8.734.2.9 **BYTE*** slqssendasyncsmsparams_s::pServiceOption

8.734.2.10 **BYTE*** slqssendasyncsmsparams_s::pSmsOnlms

8.734.2.11 **ULONG*** slqssendasyncsmsparams_s::pUserData

8.735 slqssendsmsparams_s Struct Reference

Data Fields

- [ULONG messageFormat](#)
- [ULONG messageSize](#)
- [BYTE * pMessage](#)
- [USHORT messageID](#)
- [ULONG messageFailureCode](#)
- [BYTE * pLinktimer](#)
- [BYTE * pSmsOnlms](#)

8.735.1 Detailed Description

This structure contains SMS parameters

Parameters

<i>messageFormat</i> <i>Format</i> [IN]	<ul style="list-style-type: none"> • Message format <ul style="list-style-type: none"> – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
<i>messageSize</i> [IN]	<ul style="list-style-type: none"> • The length of the message contents in bytes
<i>pMessage</i> [IN]	<ul style="list-style-type: none"> • The message contents in PDU format contains SMS header and payload message
<i>pMessageID</i> <i>OUT</i>	<ul style="list-style-type: none"> • message reference ID
<i>pMessageFailureCode</i> <i>OUT</i>	<ul style="list-style-type: none"> • (Optional) Message Failure Code • If cause code is not provided, then value will be 0xFFFFFFFF

<i>pSmsOnIms</i> [IN]	<ul style="list-style-type: none"> • (Optional) SMS on IMS • Indicates whether the message is to be sent on IMS. • Values: <ul style="list-style-type: none"> – 0x00 - Message is not to be sent on IMS – 0x01 - Message is to be sent on IMS – 0x02 to 0xFF - Reserved
<i>pLinktimer</i> [IN]	<ul style="list-style-type: none"> • (Optional) Link Timer • Keeps the GW SMS link open for the specified number of seconds; can be enabled if more messages are expected to follow

8.735.2 Field Documentation

8.735.2.1 **ULONG** `slqssendsmsparams_s::messageFailureCode`

8.735.2.2 **ULONG** `slqssendsmsparams_s::messageFormat`

8.735.2.3 **USHORT** `slqssendsmsparams_s::messageID`

8.735.2.4 **ULONG** `slqssendsmsparams_s::messageSize`

8.735.2.5 **BYTE*** `slqssendsmsparams_s::pLinktimer`

8.735.2.6 **BYTE*** `slqssendsmsparams_s::pMessage`

8.735.2.7 **BYTE*** `slqssendsmsparams_s::pSmsOnIms`

8.736 slqsSessionStateInfo Struct Reference

Data Fields

- [qaQmiInterfaceInfo](#) * [pQmiInterfaceInfo](#)
- **ULONG** `reconfiguration_required`
- **ULONG** `state`
- **ULONG** `sessionEndReason`

8.736.1 Detailed Description

Contains the session state information and information about the interface

Parameters

<i>pQmiInterface↔ Info</i>	<ul style="list-style-type: none"> • See qaQmiInterfaceInfo for more information
<i>state</i>	<ul style="list-style-type: none"> • Current Link Status <ul style="list-style-type: none"> – 1 Disconnected – 2 Connected – 3 Suspended (Unsupported) – 4 Authenticating
<i>reconfiguration↔ _required</i>	<ul style="list-style-type: none"> • Indicates if host needs to be reconfigured <ul style="list-style-type: none"> – 0 No need to reconfigure – 1 Reconfiguration required
<i>sessionEnd↔ Reason</i>	<ul style="list-style-type: none"> • See qaGobiApiTableCallEndReasons.h for Call End Reason

8.736.2 Field Documentation

8.736.2.1 `qaQmiInterfaceInfo* slqsSessionStateInfo::pQmiInterfaceInfo`

8.736.2.2 `ULONG slqsSessionStateInfo::reconfiguration_required`

8.736.2.3 `ULONG slqsSessionStateInfo::sessionEndReason`

8.736.2.4 `ULONG slqsSessionStateInfo::state`

8.737 slqsSignalStrengthInfo Struct Reference

Data Fields

- `USHORT signalStrengthReqMask`
- `USHORT rxSignalStrengthListLen`
- `struct rxSignalStrengthListElement rxSignalStrengthList [18]`
- `USHORT ecioListLen`
- `struct ecioListElement ecioList [18]`
- `INT32 lo`
- `BYTE sinr`
- `USHORT errorRateListLen`
- `struct errorRateListElement errorRateList [18]`
- `struct rsrqInformation rsrqInfo`
- `SHORT ltesnr`
- `SHORT ltersrp`

8.737.1 Detailed Description

This structure contains the Signal Strength Information

Parameters

<i>signalStrength</i> ↔ <i>ReqMask</i> [IN]	<ul style="list-style-type: none"> Request Mask <ul style="list-style-type: none"> Request additional signal information for: Bit 0 - RSSI Information bit Valid values are: 0 - Do Not Request Additional Info for RSSI 1 - Request Additional Info for RSSI Bit 1 - ECIO Information bit Valid values are: 0 - Do Not Request Additional Info for ECIO 1 - Request Additional Info for ECIO Bit 2 - IO Information bit Valid values are: 0 - Do Not Request Additional Info for IO 1 - Request Additional Info for IO Bit 3 - SINR Information bit Valid values are: 0 - Do Not Request Additional Info for SINR 1 - Request Additional Info for SINR Bit 4 - ERROR RATE Information bit Valid values are: 0 - Do Not Request Additional Info for Error Rate 1 - Request Additional Info for Error Rate Bit 5 - RSRQ Information bit Valid values are: 0 - Do Not Request Additional Info for RSRQ 1 - Request Additional Info for RSRQ Bit 6 - LTE SNR information bit Valid values are: 0 - Do not request additional information for LTE SNR 1 - Request additional information for LTE SNR Bit 7 - LTE RSRP Information bit Valid values are: 0 - Do not request additional information for LTE RSRP 1 - Request additional information for LTE RSRP
<i>rxSignal</i> ↔ <i>StrengthList</i> ↔ <i>Len</i> [OUT]	<ul style="list-style-type: none"> Number of elements in Receive Signal Strength List
<i>rxSignal</i> ↔ <i>StrengthList</i> [O↔ <i>UT</i>]	<ul style="list-style-type: none"> See rxSignalStrengthListElement for more information
<i>ecioListLen</i> [O↔ <i>UT</i>]	<ul style="list-style-type: none"> Number of elements in ECIO List
<i>ecioList</i> [OUT]	<ul style="list-style-type: none"> See ecioListElement for more information
<i>Io</i> [OUT]	<ul style="list-style-type: none"> Received Io in dBm; IO is only applicable for 1xEV-DO

<i>sinr</i> [OUT]	<ul style="list-style-type: none"> SINR level <ul style="list-style-type: none"> SINR is only applicable for 1xEV-DO; valid levels are 0 to 8 where maximum value for 0 <ul style="list-style-type: none"> SINR_LEVEL_0 is -9 dB 1 - SINR_LEVEL_1 is -6 dB 2 - SINR_LEVEL_2 is -4.5 dB 3 SINR_LEVEL_3 is -3 dB 4 - SINR_LEVEL_4 is -2 dB 5 - SINR_LEVEL_5 is +1 dB 6 - SINR_LEVEL_6 is +3 dB 7 - SINR_LEVEL_7 is +6 dB 8 - SINR_LEVEL_8 is +9 dB
<i>errorRateList</i> _↔ <i>Len</i> [OUT]	<ul style="list-style-type: none"> Number of elements in Error Rate List
<i>errorRateList</i> _↔ <i>OUT</i>	<ul style="list-style-type: none"> See errorRateListElement for more information
<i>rsrqInfo</i> [OUT]	<ul style="list-style-type: none"> See rsrqInformation for more information
<i>ltesnr</i> [OUT]	<ul style="list-style-type: none"> LTE SNR level as a scaled integer in units of 0.1 dB; e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246. LTE SNR is included only when the current serving system is LTE
<i>ltersrp</i> [OUT]	<ul style="list-style-type: none"> LTE SNR level as a scaled integer in units of 0.1 dB; e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246. LTE SNR is included only when the current serving system is LTE

8.737.2 Field Documentation

8.737.2.1 struct `ecioListElement` `slqsSignalStrengthInfo::ecioList`[18]

8.737.2.2 USHORT `slqsSignalStrengthInfo::ecioListLen`

8.737.2.3 struct `errorRateListElement` `slqsSignalStrengthInfo::errorRateList`[18]

8.737.2.4 USHORT `slqsSignalStrengthInfo::errorRateListLen`

8.737.2.5 INT32 `slqsSignalStrengthInfo::lo`

8.737.2.6 SHORT `slqsSignalStrengthInfo::ltersrp`

8.737.2.7 SHORT `slqsSignalStrengthInfo::ltesnr`

8.737.2.8 struct `rsrqInformation` `slqsSignalStrengthInfo::rsrqInfo`

8.737.2.9 struct `rxSignalStrengthListElement` `slqsSignalStrengthInfo::rxSignalStrengthList`[18]

8.737.2.10 USHORT `slqsSignalStrengthInfo::rxSignalStrengthListLen`

8.737.2.11 USHORT `slqsSignalStrengthInfo::signalStrengthReqMask`

8.737.2.12 BYTE slqsSignalStrengthInfo::sinr

8.738 SLQSSignalStrengthsIndReq Struct Reference

Data Fields

- BYTE rxSignalStrengthDelta
- BYTE ecioDelta
- BYTE ioDelta
- BYTE sinrDelta
- BYTE rsrqDelta
- BYTE ecioThresholdListLen
- SHORT ecioThresholdList [10]
- BYTE sinrThresholdListLen
- BYTE sinrThresholdList [5]
- WORD lteSnrDelta
- BYTE lteRsrpDelta

8.738.1 Detailed Description

Structure for storing the input parameters passed for SLQSSetSignalStrengthsCallback by the user.

Parameters

<i>rxSignal↔ StrengthDelta</i>	<ul style="list-style-type: none"> • RSSI delta(in dBm) at which an event report indication, including the current RSSI, will be sent to the requesting control point.
<i>ecioDelta</i>	<ul style="list-style-type: none"> • ECIO delta at which an event report indication, ecioDelta including the current ECIO, will be sent to the requesting control point. • ECIO delta is an unsigned 1 byte value that increments in negative 0.5 dBm, e.g., ecio_delta of 2 means a change of -1 dBm.
<i>ioDelta</i>	<ul style="list-style-type: none"> • IO delta (in dBm) at which an event report indication, ioDelta including the current IO, will be sent to the requesting control point.
<i>sinrDelta</i>	<ul style="list-style-type: none"> • SINR delta level at which an event report indication, sinrDelta including the current SINR, will be sent to the requesting control point.
<i>rsrqDelta</i>	<ul style="list-style-type: none"> • RSRQ delta level at which an event report indication, including the current RSRQ, will be sent to the requesting control point.
<i>ecioThreshold↔ ListLen</i>	<ul style="list-style-type: none"> • Number of elements in the ECIO threshold list.

<i>ecioThreshold</i> ↔ <i>List</i>	<ul style="list-style-type: none"> • A sequence of thresholds delimiting Ecio event reporting bands. Every time a new Ecio value crosses a threshold value, an event report indication message with the new ECIO value is sent to the requesting control point. For this field: <ul style="list-style-type: none"> – Maximum number of threshold values is 10 – At least one value must be specified.
<i>sinrThreshold</i> ↔ <i>ListLen</i>	<ul style="list-style-type: none"> • Number of elements in the SINR threshold list.
<i>sinrThreshold</i> ↔ <i>List</i>	<ul style="list-style-type: none"> • A sequence of thresholds delimiting SINR event reporting bands. Every time a new SINR value crosses a threshold value, an event report indication message with the new sinr value is sent to the requesting control point. For this field: <ul style="list-style-type: none"> – Maximum number of threshold values is 5 – At least one value must be specified.
<i>ltesnrdelta</i>	<ul style="list-style-type: none"> • LTE SNR delta level at which an event report indication, including the current SNR, will be sent to the requesting control point. LTE SNR delta level is an unsigned 2 byte value, representing the delta in units of 0.1 dB, e.g., lte_snr_delta of 3 means a change 0.3dB.
<i>lteresrpdelta</i>	<ul style="list-style-type: none"> • -LTE RSRP delta level at which an event report -indication, including the current RSRP, will be sent -to the requesting control point. LTE RSRP delta -level is an unsigned 1 byte value, representing the -delta in dB.

Note

None

8.738.2 Field Documentation**8.738.2.1** BYTE SLQSSignalStrengthsIndReq::ecioDelta**8.738.2.2** SHORT SLQSSignalStrengthsIndReq::ecioThresholdList[10]**8.738.2.3** BYTE SLQSSignalStrengthsIndReq::ecioThresholdListLen**8.738.2.4** BYTE SLQSSignalStrengthsIndReq::ioDelta**8.738.2.5** BYTE SLQSSignalStrengthsIndReq::lteRsrpDelta**8.738.2.6** WORD SLQSSignalStrengthsIndReq::lteSnrDelta**8.738.2.7** BYTE SLQSSignalStrengthsIndReq::rsrqDelta**8.738.2.8** BYTE SLQSSignalStrengthsIndReq::rxSignalStrengthDelta

8.738.2.9 **BYTE** SLQSSignalStrengthsIndReq::sinrDelta

8.738.2.10 **BYTE** SLQSSignalStrengthsIndReq::sinrThresholdList[5]

8.738.2.11 **BYTE** SLQSSignalStrengthsIndReq::sinrThresholdListLen

8.739 SLQSSignalStrengthsInformation Struct Reference

Data Fields

- struct [rxSignalStrengthListElement](#) rxSignalStrengthInfo
- struct [ecioListElement](#) ecioInfo
- [ULONG](#) io
- [BYTE](#) sinr
- struct [errorRateListElement](#) errorRateInfo
- struct [rsrqInformation](#) rsrqInfo
- struct [lteSnrinformation](#) lteSnrinfo
- struct [lteRsrpinformation](#) lteRsrpinfo

8.739.1 Detailed Description

Structure for Received Signal Strength Information.

Parameters

<i>rxSignalStrengthInfo</i>	<ul style="list-style-type: none"> • See rxSignalStrengthListElement for more information.
<i>ecioInfo</i>	<ul style="list-style-type: none"> • See ecioListElement for more information.
<i>io</i>	<ul style="list-style-type: none"> • Received IO in dBm; IO is only applicable for 1xEV-DO.
<i>sinr</i>	<ul style="list-style-type: none"> • SINR level <ul style="list-style-type: none"> – SINR is only applicable for 1xEV-DO; valid levels are 0 to 8 where maximum value for 0 - SINR_LEVEL_0 is -9 dB 1 - SINR_LEVEL_1 is -6 dB 2 - SINR_LEVEL_2 is -4.5 dB 3 - SINR_LEVEL_3 is -3 dB 4 - SINR_LEVEL_4 is -2 dB 5 - SINR_LEVEL_5 is +1 dB 6 - SINR_LEVEL_6 is +3 dB 7 - SINR_LEVEL_7 is +6 dB 8 - SINR_LEVEL_8 is +9 dB
<i>errorRateInfo</i>	<ul style="list-style-type: none"> • See errorRateListElement for more information.
<i>rsrqInfo</i>	<ul style="list-style-type: none"> • See rsrqInformation for more information.
<i>lteSnrinfo</i>	<ul style="list-style-type: none"> • See lteSnrinformation for more information.
<i>lteRsrpinfo</i>	<ul style="list-style-type: none"> • See lteRsrpinformation for more information.
	Generated by Doxygen

Note

None

8.739.2 Field Documentation

8.739.2.1 struct ecioListElement SLQSSignalStrengthsInformation::ecioInfo

8.739.2.2 struct errorRateListElement SLQSSignalStrengthsInformation::errorRateInfo

8.739.2.3 ULONG SLQSSignalStrengthsInformation::io

8.739.2.4 struct lteRsrpInformation SLQSSignalStrengthsInformation::lteRsrpInfo

8.739.2.5 struct lteSnrInformation SLQSSignalStrengthsInformation::lteSnrInfo

8.739.2.6 struct rsrqInformation SLQSSignalStrengthsInformation::rsrqInfo

8.739.2.7 struct rxSignalStrengthListElement SLQSSignalStrengthsInformation::rxSignalStrengthInfo

8.739.2.8 BYTE SLQSSignalStrengthsInformation::sinr

8.740 slqsWdsEventInfo Struct Reference

Data Fields

- [qaQmiInterfaceInfo](#) * [pQmiInterfaceInfo](#)
- ULONG * [pDormancyStatus](#)
- ULONG * [pDataBearer](#)
- ULONG * [pPacketsCountTX](#)
- ULONG * [pPacketsCountRX](#)
- ULONGLONG * [pTotalBytesTX](#)
- ULONGLONG * [pTotalBytesRX](#)

8.740.1 Detailed Description

Contains the WDS event information and information about the interface

Parameters

<i>pQmiInterface</i> ↔ <i>Info</i>	<ul style="list-style-type: none"> See qaQmiInterfaceInfo for more information
<i>pDataBearer</i> , -	Data bearer technology (NULL if not present) <ul style="list-style-type: none"> 0x00 - Indicates that this field is ignored 0x01 - CDMA 1X 0x02 - EV-DO Rev 0 0x03 - GPRS 0x04 - WCDMA 0x05 - EV-DO Rev A 0x06 - EDGE 0x07 - HSDPA and WCDMA 0x08 - WCDMA and HSUPA 0x09 - HSDPA and HSUPA 0x0A - LTE 0x0B - EV-DO Rev A EHRPD 0x0C - HSDPA+ and WCDMA 0x0D - HSDPA+ and HSUPA 0x0E - DC_HSDPA+ and WCDMA 0x0F - DC_HSDPA+ and HSUPA 0x8000 - NULL Bearer 0xFF - Unknown Technology
<i>pDormancy</i> ↔ <i>Status</i>	<ul style="list-style-type: none"> Dormancy status (NULL if not present) <ul style="list-style-type: none"> 1 - traffic channel dormant 2 - traffic channel active
<i>pPackets</i> ↔ <i>CountTX</i>	<ul style="list-style-type: none"> Packets transmitted without error (NULL if not present)
<i>pPackets</i> ↔ <i>CountRX</i>	<ul style="list-style-type: none"> Packets received without error (NULL if not present)
<i>pTotalBytesTX</i>	<ul style="list-style-type: none"> Bytes transmitted without error (NULL if not present)
<i>pTotalBytesRX</i>	<ul style="list-style-type: none"> Bytes received without error (NULL if not present)

8.740.2 Field Documentation

8.740.2.1 **ULONG*** `slqsWdsEventInfo::pDataBearer`8.740.2.2 **ULONG*** `slqsWdsEventInfo::pDormancyStatus`8.740.2.3 **ULONG*** `slqsWdsEventInfo::pPacketsCountRX`

8.740.2.4 **ULONG*** slqsWdsEventInfo::pPacketsCountTX

8.740.2.5 **qaQmiInterfaceInfo*** slqsWdsEventInfo::pQmiInterfaceInfo

8.740.2.6 **ULONGLONG*** slqsWdsEventInfo::pTotalBytesRX

8.740.2.7 **ULONGLONG*** slqsWdsEventInfo::pTotalBytesTX

8.741 SMSAsyncRawSend_s Struct Reference

Data Fields

- [WORD](#) sendStatus
- [WORD](#) messageID
- [WORD](#) causeCode
- [BYTE](#) errorClass
- [WORD](#) RPCause
- [BYTE](#) TPCause
- [BYTE](#) msgDelFailureType
- [BYTE](#) msgDelFailureCause
- [BYTE](#) alphaIDLen
- [BYTE *](#) pAlphaID
- [ULONG](#) userData

8.741.1 Detailed Description

This structure contains SMS parameters

Parameters

<i>sendStatus</i>	<ul style="list-style-type: none">• Send Status• Values:<ul style="list-style-type: none">– QMI_ERR_NONE – No error in the request– QMI_ERR_CAUSE_CODE - SMS cause code– QMI_ERR_MESSAGE_DELIVERY_FAILURE - Message could not be delivered– QMI_ERR_NO_MEMORY - Device could not allocate memory to formulate a response
<i>messageID</i>	<ul style="list-style-type: none">• Unique ID assigned by WMS for non-retry messages.
<i>causeCode</i>	<ul style="list-style-type: none">• WMS cause code
<i>errorClass</i>	<ul style="list-style-type: none">• Error Class• Values:<ul style="list-style-type: none">– 0x00 - ERROR_CLASS_TEMPORARY– 0x01 - ERROR_CLASS_PERMANENT

<i>RPCause</i>	<ul style="list-style-type: none"> • GW RP cause
<i>TPCause</i>	<ul style="list-style-type: none"> • GW TP Cause
<i>msgDelFailure</i> ↔ <i>Type</i>	<ul style="list-style-type: none"> • Message delivery failure type • Values: <ul style="list-style-type: none"> – 0x00 - WMS_MESSAGE_DELIVERY_FAILURE_TEMPORARY – 0x01 - WMS_MESSAGE_DELIVERY_FAILURE_PERMANENT
<i>msgDelFailure</i> ↔ <i>Cause</i>	<ul style="list-style-type: none"> • Message delivery failure cause • Values: <ul style="list-style-type: none"> – 0x00 - WMS_MESSAGE_BLOCKED_DUE_TO_CALL_CONTROL
<i>alphaIDLen</i>	<ul style="list-style-type: none"> • Number of sets of the pAlphaID
<i>pAlphaID</i>	<ul style="list-style-type: none"> • Alpha ID
<i>userData</i>	<ul style="list-style-type: none"> • Identifies the request associated with this indication.

8.741.2 Field Documentation

8.741.2.1 BYTE SMSAsyncRawSend_s::alphaIDLen

8.741.2.2 WORD SMSAsyncRawSend_s::causeCode

8.741.2.3 BYTE SMSAsyncRawSend_s::errorClass

8.741.2.4 WORD SMSAsyncRawSend_s::messageID

8.741.2.5 BYTE SMSAsyncRawSend_s::msgDelFailureCause

8.741.2.6 BYTE SMSAsyncRawSend_s::msgDelFailureType

8.741.2.7 BYTE* SMSAsyncRawSend_s::pAlphaID

8.741.2.8 WORD SMSAsyncRawSend_s::RPCause

8.741.2.9 WORD SMSAsyncRawSend_s::sendStatus

8.741.2.10 BYTE SMSAsyncRawSend_s::TPCause

8.741.2.11 ULONG SMSAsyncRawSend_s::userData

8.742 sMSCAddress Struct Reference

Data Fields

- uint8_t [length](#)
- uint8_t [data](#) [256]

8.742.1 Detailed Description

Parameters

<i>length</i>	<ul style="list-style-type: none">• Number of sets of following element
<i>data</i>	<ul style="list-style-type: none">• SMSC address

8.742.2 Field Documentation

8.742.2.1 uint8_t sMSCAddress::data[256]

8.742.2.2 uint8_t sMSCAddress::length

8.743 SMSCAddress Struct Reference

Data Fields

- BYTE [length](#)
- BYTE [data](#) [256]

8.743.1 Detailed Description

This structure holds SMSC information

Parameters

<i>length</i>	<ul style="list-style-type: none">• Number of sets of following element
<i>data</i>	<ul style="list-style-type: none">• SMSC address

8.743.2 Field Documentation

8.743.2.1 `BYTE SMSCAddress::data[256]`

8.743.2.2 `BYTE SMSCAddress::length`

8.744 sMSCAddressTlv Struct Reference

Data Fields

- `uint8_t TlvPresent`
- `sMSCAddressInfo SMSCInfo`

8.744.1 Detailed Description

Parameters

<i>TlvPresent</i>	<ul style="list-style-type: none">• Boolean indicating the presence of the TLV in the QMI response
<i>SMSCInfo</i>	<ul style="list-style-type: none">• SMSC Address• See sMSCAddressInfo for more information

8.744.2 Field Documentation

8.744.2.1 `sMSCAddressInfo sMSCAddressTlv::SMSCInfo`

8.744.2.2 `uint8_t sMSCAddressTlv::TlvPresent`

8.745 sMSEtwsMessage Struct Reference

Data Fields

- `uint8_t notificationType`
- `uint16_t length`
- `uint8_t data [1254]`

8.745.1 Detailed Description

Parameters

<i>notificationType</i>	<ul style="list-style-type: none">• Message mode 0x00 - Primary 0x01 - Secondary GSM 0x02 - Secondary UMTS
<i>length</i>	<ul style="list-style-type: none">• Number of sets of following elements
<i>data</i>	<ul style="list-style-type: none">• Raw message data

8.745.2 Field Documentation

8.745.2.1 `uint8_t sMSEtwsMessage::data[1254]`8.745.2.2 `uint16_t sMSEtwsMessage::length`8.745.2.3 `uint8_t sMSEtwsMessage::notificationType`

8.746 SMSEtwsMessage Struct Reference

Data Fields

- [BYTE notificationType](#)
- [WORD length](#)
- [BYTE data](#) [1254]

8.746.1 Detailed Description

This structure holds information related earthquake and Tsunami warning system

Parameters

<i>notificationType</i>	<ul style="list-style-type: none">• Message mode 0x00 - Primary 0x01 - Secondary GSM 0x02 - Secondary UMTS
<i>length</i>	<ul style="list-style-type: none">• Number of sets of following elements
<i>data</i>	<ul style="list-style-type: none">• Raw message data

8.746.2 Field Documentation

8.746.2.1 `BYTE sMSEtwsMessage::data[1254]`

8.746.2.2 WORD SMSEtwsMessage::length

8.746.2.3 BYTE SMSEtwsMessage::notificationType

8.747 sMSEtwsMessageTlv Struct Reference

Data Fields

- uint8_t [TlvPresent](#)
- [sMSEtwsMessageInfo](#) [EtwsMessageInfo](#)

8.747.1 Detailed Description

Parameters

<i>TlvPresent</i>	<ul style="list-style-type: none"> • Boolean indicating the presence of the TLV in the QMI response
<i>EtwsMessage↔ Info</i>	<ul style="list-style-type: none"> • ETWS Message • See sMSEtwsMessageInfo for more information

8.747.2 Field Documentation

8.747.2.1 sMSEtwsMessageInfo sMSEtwsMessageTlv::EtwsMessageInfo

8.747.2.2 uint8_t sMSEtwsMessageTlv::TlvPresent

8.748 sMSEtwsPlmn Struct Reference

Data Fields

- uint16_t [mobileCountryCode](#)
- uint16_t [mobileNetworkCode](#)

8.748.1 Detailed Description

Parameters

<i>mobileCountry↔ Code</i>	<ul style="list-style-type: none"> • 16 bit representation of MCC value range : 0 -999
<i>mobile↔ NetworkCode</i>	<ul style="list-style-type: none"> • 16 bit representation of MNC value range : 0 -999

8.748.2 Field Documentation

8.748.2.1 uint16_t sMSEtwsPlmn::mobileCountryCode

8.748.2.2 uint16_t sMSEtwsPlmn::mobileNetworkCode

8.749 SMSEtwsPlmn Struct Reference

Data Fields

- [WORD mobileCountryCode](#)
- [WORD mobileNetworkCode](#)

8.749.1 Detailed Description

This structure holds information related ETWS PLMN

Parameters

<i>mobileCountry↔ Code</i>	<ul style="list-style-type: none">• 16 bit representation of MCC value range : 0 -999
<i>mobile↔ NetworkCode</i>	<ul style="list-style-type: none">• 16 bit representation of MNC value range : 0 -999

8.749.2 Field Documentation

8.749.2.1 WORD SMSEtwsPlmn::mobileCountryCode

8.749.2.2 WORD SMSEtwsPlmn::mobileNetworkCode

8.750 SMSEventInfo_s Struct Reference

Data Fields

- [BYTE smsEventType](#)
- [SMSMTMessageInfo * pMTMessageInfo](#)
- [SMSTransferRouteMTMessageInfo * pTransferRouteMTMessageInfo](#)
- [SMSMessageModelInfo * pMessageModelInfo](#)
- [SMSEtwsMessageInfo * pEtwsMessageInfo](#)
- [SMSEtwsPlmnInfo * pEtwsPlmnInfo](#)
- [SMSCAddressInfo * pSMSCAddressInfo](#)
- [SMSOnIMSInfo * pSMSOnIMSInfo](#)

8.750.1 Detailed Description

This structure will hold the information related to received SMS events

Parameters

<i>smsEventType</i>	<ul style="list-style-type: none"> Type of the SMS events that are received. This is a bit map of SMSEventType. Only the parameters (which follows) related to the events received would be filled, and the rest of the parameters would be NULL
<i>pMTMessageInfo</i>	<ul style="list-style-type: none"> pointer to the SMSMTMessageInfo structure NULL, if this event is not present in the smsEventType parameter
<i>pTransferRouteMTMessageInfo</i>	<ul style="list-style-type: none"> pointer to the SMSTransferRouteMTMessageInfo structure . NULL, if this event is not present in the smsEventType parameter
<i>pMessageModelInfo</i>	<ul style="list-style-type: none"> pointer to the SMSMessageModelInfo structure NULL, if this event is not present in the smsEventType parameter
<i>pEtwsMessageInfo</i>	<ul style="list-style-type: none"> pointer to the SMSEtwsMessageInfo structure NULL, if this event is not present in the smsEventType parameter
<i>pEtwsPlmnInfo</i>	<ul style="list-style-type: none"> pointer to the SMSEtwsPlmnInfo structure NULL, if this event is not present in the smsEventType parameter
<i>pSMSCAddressInfo</i>	<ul style="list-style-type: none"> pointer to the SMSCAddressInfo structure NULL, if this event is not present in the smsEventType parameter
<i>pSMSOnIMSInfo</i>	<ul style="list-style-type: none"> pointer to the SMSOnIMSInfo structure NULL, if this event is not present in the smsEventType parameter Note: None

8.750.2 Field Documentation

8.750.2.1 [SMSEtwsMessageInfo](#)* [SMSEventInfo_s::pEtwsMessageInfo](#)

8.750.2.2 [SMSEtwsPlmnInfo](#)* [SMSEventInfo_s::pEtwsPlmnInfo](#)

8.750.2.3 [SMSMessageModelInfo](#)* [SMSEventInfo_s::pMessageModelInfo](#)

8.750.2.4 [SMSMTMessageInfo](#)* [SMSEventInfo_s::pMTMessageInfo](#)

8.750.2.5 [SMSCAddressInfo](#)* [SMSEventInfo_s::pSMSCAddressInfo](#)

8.750.2.6 SMSOnIMSInfo* SMSEventInfo_s::pSMSOnIMSInfo

8.750.2.7 SMSTransferRouteMTMessageInfo* SMSEventInfo_s::pTransferRouteMTMessageInfo

8.750.2.8 BYTE SMSEventInfo_s::smsEventType

8.751 smsMaxStorageSizeReq Struct Reference

Data Fields

- [BYTE storageType](#)
- [BYTE * pMessageMode](#)

8.751.1 Detailed Description

This structure contains get store max size resquest parameters

Parameters

<i>storageType</i>	<ul style="list-style-type: none"> • SMS message storage type <ul style="list-style-type: none"> – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
<i>pMessage↔ Mode(optional)</i>	parameter) <ul style="list-style-type: none"> • 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

Note

The Message Mode TLV must be included if the device is capable of supporting more than one protocol

8.751.2 Field Documentation

8.751.2.1 BYTE* smsMaxStorageSizeReq::pMessageMode

8.751.2.2 BYTE smsMaxStorageSizeReq::storageType

8.752 smsMaxStorageSizeResp Struct Reference

Data Fields

- [ULONG maxStorageSize](#)
- [ULONG freeSlots](#)

8.752.1 Detailed Description

This structure contains get store max size response parameters

Parameters

<i>maxStorageSize</i>	- <ul style="list-style-type: none"> Memory Store Size
<i>freeSlots</i>	- <ul style="list-style-type: none"> Optional parameter indicating how much Memory is available function SLQSSmsGetMaxStorageSize() returns a default value 0xFFFFFFFF for parameter values if no response is received from the device.

8.752.2 Field Documentation

8.752.2.1 **ULONG** smsMaxStorageSizeResp::freeSlots

8.752.2.2 **ULONG** smsMaxStorageSizeResp::maxStorageSize

8.753 SMSMemoryInfo Struct Reference

Data Fields

- [BYTE](#) storageType
- [BYTE](#) messageMode

8.753.1 Detailed Description

This structure holds information related to memory

Parameters

<i>storageType</i>	<ul style="list-style-type: none"> Indicates the type of memory storage 0x00 - STORAGE_TYPE_UIM 0x01 - STORAGE_TYPE_NV
<i>messageMode</i>	<ul style="list-style-type: none"> Indicates the type of memory mode 0x00 - MESSAGE_MODE_CDMA - CDMA 0x01 - MESSAGE_MODE_GW - GW

8.753.2 Field Documentation

8.753.2.1 **BYTE** SMSMemoryInfo::messageMode

8.753.2.2 BYTE SMSMemoryInfo::storageType

8.754 sMSMessageMode Struct Reference

Data Fields

- uint8_t [messageMode](#)

8.754.1 Detailed Description

Parameters

<i>messageMode</i>	Message Mode
--------------------	--------------

8.754.2 Field Documentation

8.754.2.1 uint8_t sMSMessageMode::messageMode

8.755 SMSMessageMode Struct Reference

Data Fields

- [BYTE messageMode](#)

8.755.1 Detailed Description

This structure holds information related to message mode

Parameters

<i>messageMode</i>	<ul style="list-style-type: none">• Message mode 0x00 - CDMA 0x01 - GW
--------------------	--

8.755.2 Field Documentation

8.755.2.1 [BYTE SMSMessageMode::messageMode](#)

8.756 smsMsgprotocolResp Struct Reference

Data Fields

- [BYTE msgProtocol](#)

8.756.1 Detailed Description

This structure contains get message protocol response parameters

Parameters

<i>msgProtocol</i>	- <ul style="list-style-type: none"> • Message Protocol • Values: <ul style="list-style-type: none"> – 0x00 - MESSAGE_PROTOCOL_CDMA – 0x01 - MESSAGE_PROTOCOL_WCDMA
--------------------	--

8.756.2 Field Documentation

8.756.2.1 BYTE smsMsgprotocolResp::msgProtocol

8.757 sMSMTMessage Struct Reference

Data Fields

- uint32_t [storageType](#)
- uint32_t [messageIndex](#)

8.757.1 Detailed Description

Parameters

<i>storageType</i>	memory storage 0x00-UIIM 0x01-NV
<i>messageIndex</i>	MT Message index

8.757.2 Field Documentation

8.757.2.1 uint32_t sMSMTMessage::messageIndex

8.757.2.2 uint32_t sMSMTMessage::storageType

8.758 SMSMTMessage Struct Reference

Data Fields

- ULONG [storageType](#)
- ULONG [messageIndex](#)

8.758.1 Detailed Description

This structure holds information related to MT SMS

Parameters

<i>storageType</i>	<ul style="list-style-type: none">SMS message storage type for the new message 0 - UIM 1 - NV
<i>messageIndex</i>	<ul style="list-style-type: none">Index of the new message

8.758.2 Field Documentation

8.758.2.1 **ULONG** SMSMTMessage::messageIndex

8.758.2.2 **ULONG** SMSMTMessage::storageType

8.759 sMSOnIMS Struct Reference

Data Fields

- BYTE** [sMSOnIMS](#)

8.759.1 Detailed Description

This structure holds information related to message mode

Parameters

<i>sMSOnIMS</i>	<ul style="list-style-type: none">Indicates whether the message is received from IMS 0x00 - Message is not received from IMS 0x01 - Message is received from IMS 0x02-0xFF - Reserved Note: In multiple modem solutions, this TLV may be used to help the client determine with which modem to communicate. This TLV may not be supported on all implementations.
-----------------	---

8.759.2 Field Documentation

8.759.2.1 **BYTE** sMSOnIMS::sMSOnIMS

8.760 sMSOnIMS Struct Reference

Data Fields

- uint8_t** [sMSOnIMS](#)

8.760.1 Detailed Description

Parameters

<i>smsOnIMS</i>	SMS on IMS
-----------------	------------

8.760.2 Field Documentation

8.760.2.1 `uint8_t sMSOnIMS::smsOnIMS`

8.761 sMSOnIMSTlv Struct Reference

Data Fields

- `uint8_t TlvPresent`
- `sMSOnIMSInfo IMSInfo`

8.761.1 Detailed Description

Parameters

<i>TlvPresent</i>	<ul style="list-style-type: none">• Boolean indicating the presence of the TLV in the QMI response
<i>IMSInfo</i>	<ul style="list-style-type: none">• SMS on IMS• See <code>sMSOnIMSInfo</code> for more information

8.761.2 Field Documentation

8.761.2.1 `sMSOnIMSInfo sMSOnIMSTlv::IMSInfo`

8.761.2.2 `uint8_t sMSOnIMSTlv::TlvPresent`

8.762 smsRouteEntry Struct Reference

Data Fields

- `BYTE messageType`
- `BYTE messageClass`
- `BYTE routeStorage`
- `BYTE receiptAction`

8.762.1 Detailed Description

This structure contains SMS route entry details

Parameters

<i>messageType</i>	- <ul style="list-style-type: none"> • Message type matching this route • Values: <ul style="list-style-type: none"> – 0x00 - MESSAGE_TYPE_POINT_TO_POINT
<i>messageClass</i>	- <ul style="list-style-type: none"> • Message Class • Values: <ul style="list-style-type: none"> – 0x00 - MESSAGE_CLASS_0 – 0x01 - MESSAGE_CLASS_1 – 0x02 - MESSAGE_CLASS_2 – 0x03 - MESSAGE_CLASS_3 – 0x04 - MESSAGE_CLASS_NONE – 0x05 - MESSAGE_CLASS_CDMA
<i>routeStorage</i>	- <ul style="list-style-type: none"> • If the receiptAction is store where to store the message • Values: <ul style="list-style-type: none"> – 0x00 - STORAGE_TYPE_UIM – 0x01 - STORAGE_TYPE_NV – 0xFF - STORAGE_TYPE_NONE
<i>receiptAction</i>	- <ul style="list-style-type: none"> • Action to be taken on receipt of a message matching the specified type and class for this route • Values: <ul style="list-style-type: none"> – 0x00 - DISCARD (discarded without notification) – 0x01 - STORE AND NOTIFY (stored and notified to the registered clients) – 0x02 - TRANSFER ONLY (transferred to the client, client expected to send the ACK) – 0x03 - TRANSFER AND ACK (transferred to the client, device expected to send the ACK)

8.762.2 Field Documentation

8.762.2.1 **BYTE** smsRouteEntry::messageClass

8.762.2.2 **BYTE** smsRouteEntry::messageType

8.762.2.3 **BYTE** smsRouteEntry::receiptAction

8.762.2.4 **BYTE** smsRouteEntry::routeStorage

8.763 smsSetRoutesReq Struct Reference

Data Fields

- [WORD](#) numOfRoutes

- [smsRouteEntry routeList](#) [0x0A]
- [BYTE * pTransferStatusReport](#)

8.763.1 Detailed Description

This structure contains SMS route request parameters

Parameters

<i>numOfRoutes</i>	- <ul style="list-style-type: none"> • Number of sets of the following element
<i>routeList</i>	- <ul style="list-style-type: none"> • Array containing the set of smsRouteEntry
<i>pTransferStatusReport</i>	- <ul style="list-style-type: none"> • 0x01 - Status report are transferred to the client (optional)

8.763.2 Field Documentation

8.763.2.1 **WORD** [smsSetRoutesReq::numOfRoutes](#)

8.763.2.2 **BYTE*** [smsSetRoutesReq::pTransferStatusReport](#)

8.763.2.3 [smsRouteEntry](#) [smsSetRoutesReq::routeList](#)[0x0A]

8.764 sMSTransferRouteMTMessage Struct Reference

Data Fields

- [uint8_t ackIndicator](#)
- [uint32_t transactionID](#)
- [uint8_t format](#)
- [uint16_t length](#)
- [uint8_t data](#) [256]

8.764.1 Detailed Description

Parameters

<i>ackIndicator</i>	<ul style="list-style-type: none"> • Parameter to indicate if ACK must be sent by the control point 0x00 - Send ACK 0x01 - Do not send ACK
<i>transactionID</i>	<ul style="list-style-type: none"> • Transaction ID of the message

<i>format</i>	<ul style="list-style-type: none">Message format 0x00 - CDMA 0x02 - 0x05 - Reserved 0x06 - GW_PP 0x07 - GW_BC
<i>length</i>	<ul style="list-style-type: none">Length of the raw message. This length should not exceed the maximum WMS payload length of 256 bytes
<i>data</i>	<ul style="list-style-type: none">Raw message data

8.764.2 Field Documentation

8.764.2.1 `uint8_t sMSTransferRouteMTMessage::ackIndicator`

8.764.2.2 `uint8_t sMSTransferRouteMTMessage::data[256]`

8.764.2.3 `uint8_t sMSTransferRouteMTMessage::format`

8.764.2.4 `uint16_t sMSTransferRouteMTMessage::length`

8.764.2.5 `uint32_t sMSTransferRouteMTMessage::transactionID`

8.765 SMSTransferRouteMTMessage Struct Reference

Data Fields

- [BYTE](#) `ackIndicator`
- [ULONG](#) `transactionID`
- [BYTE](#) `format`
- [WORD](#) `length`
- [BYTE](#) `data` [256]

8.765.1 Detailed Description

This structure holds information related to transfer route MT SMS

Parameters

<i>ackIndicator</i>	<ul style="list-style-type: none">Parameter to indicate if ACK must be sent by the control point 0x00 - Send ACK 0x01 - Do not send ACK
<i>transactionID</i>	<ul style="list-style-type: none">Transaction ID of the message
<i>format</i>	<ul style="list-style-type: none">Message format 0x00 - CDMA 0x02 - 0x05 - Reserved 0x06 - GW_PP 0x07 - GW_BC
Generated by Doxygen	
<i>length</i>	<ul style="list-style-type: none">Length of the raw message. This length should not exceed the maximum WMS payload length of 256 bytes

8.765.2 Field Documentation

8.765.2.1 **BYTE** SMSTransferRouteMTMessage::ackIndicator

8.765.2.2 **BYTE** SMSTransferRouteMTMessage::data[256]

8.765.2.3 **BYTE** SMSTransferRouteMTMessage::format

8.765.2.4 **WORD** SMSTransferRouteMTMessage::length

8.765.2.5 **ULONG** SMSTransferRouteMTMessage::transactionID

8.766 sQosFlowStat Struct Reference

Data Fields

- [ULONG](#) bearerId
- [ULONG](#) tx_pkt
- [ULONG](#) tx_pkt_drp
- [ULONGLONG](#) tx_bytes
- [ULONGLONG](#) tx_bytes_drp

8.766.1 Detailed Description

This structure contains the Data statistic per QoS flow

Parameters

<i>bearerId</i>	<ul style="list-style-type: none"> • Bearer ID
<i>tx_pkt</i>	<ul style="list-style-type: none"> • number of sent packets for the QoS flow ID
<i>tx_pkt_drp</i>	<ul style="list-style-type: none"> • number of dropped(TX) packets for the QoS flow ID
<i>tx_bytes</i>	<ul style="list-style-type: none"> • number of sent bytes for the QoS flow ID
<i>tx_bytes_drp</i>	<ul style="list-style-type: none"> • number of dropped(TX) bytes for the QoS flow ID

8.766.2 Field Documentation

8.766.2.1 **ULONG** sQosFlowStat::bearerId

8.766.2.2 **ULONGLONG** sQosFlowStat::tx_bytes

8.766.2.3 **ULONGLONG** sQosFlowStat::tx_bytes_drp

8.766.2.4 **ULONG** sQosFlowStat::tx_pkt

8.766.2.5 **ULONG** sQosFlowStat::tx_pkt_drp

8.767 sQosStat Struct Reference

Data Fields

- [ULONG](#) apnId
- [ULONG](#) total_tx_pkt
- [ULONG](#) total_tx_pkt_drp
- [ULONG](#) total_rx_pkt
- [ULONGLONG](#) total_tx_bytes
- [ULONGLONG](#) total_tx_bytes_drp
- [ULONGLONG](#) total_rx_bytes
- [ULONG](#) numQosFlow
- [sQosFlowStat](#) qosFlow [(10)]

8.767.1 Detailed Description

This structure contains the Data statistic per QoS flow

Parameters

<i>apnId</i>	<ul style="list-style-type: none"> • APN id • ID identifying the connected APN that the client would like to query the data statistic for
<i>total_tx_pkt</i>	<ul style="list-style-type: none"> • sum of all packets sent
<i>total_tx_pkt_drp</i>	<ul style="list-style-type: none"> • sum of all(TX) packets dropped
<i>total_rx_pkt</i>	<ul style="list-style-type: none"> • sum of all packets received
<i>total_tx_bytes</i>	<ul style="list-style-type: none"> • sum of all bytes sent
<i>total_tx_bytes_drp</i>	<ul style="list-style-type: none"> • sum of all(TX) bytes dropped
<i>total_rx_bytes</i>	<ul style="list-style-type: none"> • number of received bytes for the QoS flow ID

<i>numQosFlow</i>	<ul style="list-style-type: none"> • pointer to number of QoS flow Stat
<i>qosFlow</i> [<i>MAX_QOS_SPEC_PER_APN</i>]	<ul style="list-style-type: none"> • Data statistic per QoS flow • See sQosFlowStat for more information • See MAX_QOS_SPEC_PER_APN for more information

8.767.2 Field Documentation

8.767.2.1 **ULONG** sQosStat::apnId

8.767.2.2 **ULONG** sQosStat::numQosFlow

8.767.2.3 **sQosFlowStat** sQosStat::qosFlow[(10)]

8.767.2.4 **ULONGLONG** sQosStat::total_rx_bytes

8.767.2.5 **ULONG** sQosStat::total_rx_pkt

8.767.2.6 **ULONGLONG** sQosStat::total_tx_bytes

8.767.2.7 **ULONGLONG** sQosStat::total_tx_bytes_drp

8.767.2.8 **ULONG** sQosStat::total_tx_pkt

8.767.2.9 **ULONG** sQosStat::total_tx_pkt_drp

8.768 SrvStatusInfo Struct Reference

Data Fields

- [BYTE](#) srvStatus
- [BYTE](#) isPrefDataPath

8.768.1 Detailed Description

Structure for storing the service status information for CDMA and HDR networks.

Parameters

<i>srvStatus</i>	<ul style="list-style-type: none"> • Service status of the system. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save – 0xFF - Not Available
<i>isPrefDataPath</i>	<ul style="list-style-type: none"> • Whether the RAT is the preferred data path. <ul style="list-style-type: none"> – 0x00 - Not preferred – 0x01 - Preferred – 0xFF - Not Available

8.768.2 Field Documentation

8.768.2.1 **BYTE** SrvStatusInfo::isPrefDataPath

8.768.2.2 **BYTE** SrvStatusInfo::srvStatus

8.769 ssdatasession_params Struct Reference

Data Fields

- [BOOL](#) action
- [BYTE](#) instancelId
- [ULONG](#) * pTechnology
- [ULONG](#) * pProfileId3GPP
- [ULONG](#) * pProfileId3GPP2
- [ULONG](#) sessionId
- [ULONG](#) failureReason
- [ULONG](#) failureReasonv4
- [ULONG](#) failureReasonv6
- [ULONG](#) rcv4
- [ULONG](#) rcv6
- [ULONG](#) v4sessionId
- [ULONG](#) v6sessionId
- [BYTE](#) ipfamily
- [ULONG](#) * pAuthentication
- [CHAR](#) * pUsername
- [CHAR](#) * pPassword
- [ULONG](#) verbFailReasonType
- [ULONG](#) verbFailReason

8.769.1 Detailed Description

This structure contains the start/stop data session params Information

Parameters

<i>action</i>	<ul style="list-style-type: none"> • 1 - Start Session • 0 - Stop Session
<i>instanceId</i>	<ul style="list-style-type: none"> • PDP Instance. • Instance ID corresponding to the session ID
<i>pTechnology</i>	<ul style="list-style-type: none"> • Indicates the technology preference (optional) <ul style="list-style-type: none"> – 1 - UMTS – 2 - CDMA – 3 - eMBMS – 4 - Modem Link Label. Modem Link is an interface for transferring data between entities on AP and modem.
<i>pProfileId3GPP</i>	<ul style="list-style-type: none"> • configured 3GPP profile identifier
<i>pProfileId3GPP2</i>	<ul style="list-style-type: none"> • configured 3GPP2 profile identifier
<i>sessionId</i> [IN]	

8.769.2 Field Documentation

8.769.2.1 **BOOL** ssdatasession_params::action

8.769.2.2 **ULONG** ssdatasession_params::failureReason

8.769.2.3 **ULONG** ssdatasession_params::failureReasonv4

8.769.2.4 **ULONG** ssdatasession_params::failureReasonv6

8.769.2.5 **BYTE** ssdatasession_params::instanceId

8.769.2.6 **BYTE** ssdatasession_params::ipfamily

8.769.2.7 **ULONG*** ssdatasession_params::pAuthentication

8.769.2.8 **CHAR*** ssdatasession_params::pPassword

8.769.2.9 **ULONG*** ssdatasession_params::pProfileId3GPP

- 8.769.2.10 **ULONG*** ssdatasession_params::pProfileId3GPP2
- 8.769.2.11 **ULONG*** ssdatasession_params::pTechnology
- 8.769.2.12 **CHAR*** ssdatasession_params::pUsername
- 8.769.2.13 **ULONG** ssdatasession_params::rcv4
- 8.769.2.14 **ULONG** ssdatasession_params::rcv6
- 8.769.2.15 **ULONG** ssdatasession_params::sessionId
- 8.769.2.16 **ULONG** ssdatasession_params::v4sessionId
- 8.769.2.17 **ULONG** ssdatasession_params::v6sessionId
- 8.769.2.18 **ULONG** ssdatasession_params::verbFailReason
- 8.769.2.19 **ULONG** ssdatasession_params::verbFailReasonType

8.770 SupportedMsgList Struct Reference

Data Fields

- [WORD](#) supportedMsgLen
- [BYTE](#) supportedMsgs [256]

8.770.1 Detailed Description

This structure contains the Supported Messages List Information

Parameters

<i>supportedMsgLen</i>	<ul style="list-style-type: none">• Number of sets of the supported messages
<i>supportedMsgs</i>	<ul style="list-style-type: none">• Array of uint8 is a bitmask where each bit represents a message ID.• Starting with the LSB, bit 0 represents message ID 0, bit 1 represents message ID 1.

8.770.2 Field Documentation

8.770.2.1 WORD SupportedMsgList::supportedMsgLen

8.770.2.2 BYTE SupportedMsgList::supportedMsgs[256]

8.771 SUPSInfo Struct Reference

Data Fields

- [BYTE svcType](#)
- [BYTE isModByCC](#)

8.771.1 Detailed Description

This structure contains information about the Supplementary Services.

Parameters

<i>svcType</i>	<ul style="list-style-type: none"> • Service type. <ul style="list-style-type: none"> – 0x01 - SERVICE_TYPE_ACTIVATE - Activate – 0x02 - SERVICE_TYPE_DEACTIVATE - Deactivate – 0x03 - SERVICE_TYPE_REGISTER - Register – 0x04 - SERVICE_TYPE_ERASE - Erase – 0x05 - SERVICE_TYPE_INTERROGATE - Interrogate – 0x06 - SERVICE_TYPE_REGISTER_PASSWORD - Register password – 0x07 - SERVICE_TYPE_USSD - USSD
<i>isModByCC</i>	<ul style="list-style-type: none"> • Indicates whether the supplementary service data is modified by the card (SIM/USIM) as part of the call control: <ul style="list-style-type: none"> – 0 - False – 1 - True

8.771.2 Field Documentation

8.771.2.1 BYTE SUPSInfo::isModByCC

8.771.2.2 BYTE SUPSInfo::svcType

8.772 SV Struct Reference

Data Fields

- [WORD id](#)
- [ULONG system](#)
- [BYTE mask](#)

8.772.1 Detailed Description

This structure contains the Delete [SV](#) Info

Parameters

<i>id</i>	<ul style="list-style-type: none">• SV ID of the satellite whose data is to be deleted• Range:<ul style="list-style-type: none">– For GPS: 1 to 32– For SBAS: 33 to 64– For GLONASS: 65 to 96
<i>system</i>	<ul style="list-style-type: none">• Indicates to which constellation this SV belongs• Valid values:<ul style="list-style-type: none">– eQMI_LOC_SV_SYSTEM_GPS (1) - GPS satellite– eQMI_LOC_SV_SYSTEM_GALILEO (2) - GALILEO satellite– eQMI_LOC_SV_SYSTEM_SBAS (3) - SBAS satellite– eQMI_LOC_SV_SYSTEM_COMPASS (4) - COMPASS satellite– eQMI_LOC_SV_SYSTEM_GLONASS (5) - GLONASS satellite– eQMI_LOC_SV_SYSTEM_BDS (6) - BDS satellite
<i>mask</i>	<ul style="list-style-type: none">• Indicates if the ephemeris or almanac for a satellite is to be deleted• Valid values:<ul style="list-style-type: none">– 0x01 - DELETE_EPHEMERIS– 0x02 - DELETE_ALMANAC

8.772.2 Field Documentation

8.772.2.1 WORD SV::id

8.772.2.2 BYTE SV::mask

8.772.2.3 ULONG SV::system

8.773 SVInfo Struct Reference

Data Fields

- [BYTE](#) len
- [SV](#) * pSV

8.773.1 Detailed Description

This structure contains the elements of Delete [SV](#) Info

Parameters

<i>len</i>	<ul style="list-style-type: none"> Number of sets of the following elements in struct SV: <ul style="list-style-type: none"> gnssSvId system deleteSvInfoMask
<i>pSV</i>	<ul style="list-style-type: none"> Pointer to struct SV. See SV for more information

8.773.2 Field Documentation

8.773.2.1 BYTE SVInfo::len

8.773.2.2 SV* SVInfo::pSV

8.774 svUsedforFix_s Struct Reference

Data Fields

- [BYTE gnssSvUsedList_len](#)
- [WORD gnssSvUsedList \[255\]](#)

8.774.1 Detailed Description

This structure contains SVs Used to Calculate the Fix.

Parameters

<i>gnssSvUsedList_len</i>	<ul style="list-style-type: none"> Number of sets of gnssSvUsedList
<i>pGnssSvUsedList</i>	<ul style="list-style-type: none"> Entry in the list contains the SV ID of a satellite used for calculating this position report. Following information is associated with each SV ID: <ul style="list-style-type: none"> GPS - 1 to 32 SBAS - 33 to 64 GLONASS - 65 to 96 QZSS - 193 to 197 BDS - 201 to 237

8.774.2 Field Documentation

8.774.2.1 WORD svUsedforFix_s::gnssSvUsedList[255]

8.774.2.2 BYTE svUsedforFix_s::gnssSvUsedList_len

8.775 SWI_STRUCT_CarrierImage Struct Reference

Data Fields

- ULONG m_nCarrierId
- ULONG m_nFolderId
- ULONG m_nStorage
- BYTE m_FwImageId [16]
- BYTE m_FwBuildId [100]
- BYTE m_PriImageId [16]
- BYTE m_PriBuildId [100]

8.775.1 Detailed Description

This structure contains the Carrier Image parameters.

Parameters

<i>m_nCarrierId</i>	<ul style="list-style-type: none"> • Unique numeric carrier ID indicating the carrier that the following images belong to
<i>m_nFolderId</i>	<ul style="list-style-type: none"> • Unique numeric folder ID indicating the folder where the images should reside on the host storage.
<i>m_nStorage</i>	<ul style="list-style-type: none"> • Information of storage type • Values <ul style="list-style-type: none"> – 0 - Device – 1 - Host
<i>m_FwImageId</i>	<ul style="list-style-type: none"> • Firmware image ID
<i>m_FwBuildId</i>	<ul style="list-style-type: none"> • Firmware build ID
<i>m_PriImageId</i>	<ul style="list-style-type: none"> • PRI image ID
<i>m_PriBuildId</i>	<ul style="list-style-type: none"> • PRI build ID

8.775.2 Field Documentation

8.775.2.1 BYTE SWI_STRUCT_CarrierImage::m_FwBuildId[100]

8.775.2.2 **BYTE** SWI_STRUCT_CarrierImage::m_FwImageld[16]

8.775.2.3 **ULONG** SWI_STRUCT_CarrierImage::m_nCarrierId

8.775.2.4 **ULONG** SWI_STRUCT_CarrierImage::m_nFolderId

8.775.2.5 **ULONG** SWI_STRUCT_CarrierImage::m_nStorage

8.775.2.6 **BYTE** SWI_STRUCT_CarrierImage::m_PriBuildId[100]

8.775.2.7 **BYTE** SWI_STRUCT_CarrierImage::m_PrImageld[16]

8.776 SwiLocGetAutoStartResp Struct Reference

Data Fields

- [BYTE function](#)
- [BOOL function_reported](#)
- [BYTE fix_type](#)
- [BOOL fix_type_reported](#)
- [BYTE max_time](#)
- [BOOL max_time_reported](#)
- [ULONG max_dist](#)
- [BOOL max_dist_reported](#)
- [ULONG fix_rate](#)
- [BOOL fix_rate_reported](#)

8.776.1 Detailed Description

This structure contains SWI LOC Get Auto Start setting

Parameters

<i>function</i>	<ul style="list-style-type: none"> • Setting to indicate when modem should start an automatic GNSS fix <ul style="list-style-type: none"> – 0 - disabled – 1 - At bootup – 2 - When NMEA port is opened
<i>function</i> ↔ <i>reported</i>	<ul style="list-style-type: none"> • 0 - not reported by modem • 1 - reported by modem
<i>fix_type</i>	<ul style="list-style-type: none"> • Type of GNSS fix: <ul style="list-style-type: none"> – 1 - Default Engine mode – 2 - MS-Based – 3 - MS-Assisted – 4 - Standalone

<i>fix_type_↔ reported</i>	<ul style="list-style-type: none"> • 0 - not reported by modem • 1 - reported by modem
<i>max_time</i>	<ul style="list-style-type: none"> • Maximum time allowed for the receiver to get a fix in seconds • Valid range: 1-255
<i>max_time_↔ reported</i>	<ul style="list-style-type: none"> • 0 - not reported by modem • 1 - reported by modem
<i>max_dist</i>	<ul style="list-style-type: none"> • Maximum uncertainty of a fix measured by distance in meters • Valid range: 1 - 4294967280
<i>max_dist_↔ reported</i>	<ul style="list-style-type: none"> • 0 - not reported by modem • 1 - reported by modem
<i>fix_rate</i>	<ul style="list-style-type: none"> • Time between fixes in seconds • Valid range: 1–65535
<i>fix_rate_↔ reported</i>	<ul style="list-style-type: none"> • 0 - not reported by modem • 1 - reported by modem

8.776.2 Field Documentation

8.776.2.1 **ULONG** SwiLocGetAutoStartResp::fix_rate

8.776.2.2 **BOOL** SwiLocGetAutoStartResp::fix_rate_reported

8.776.2.3 **BYTE** SwiLocGetAutoStartResp::fix_type

8.776.2.4 **BOOL** SwiLocGetAutoStartResp::fix_type_reported

8.776.2.5 **BYTE** SwiLocGetAutoStartResp::function

8.776.2.6 **BOOL** SwiLocGetAutoStartResp::function_reported

8.776.2.7 **ULONG** SwiLocGetAutoStartResp::max_dist

8.776.2.8 **BOOL** SwiLocGetAutoStartResp::max_dist_reported

8.776.2.9 **BYTE** SwiLocGetAutoStartResp::max_time

8.776.2.10 **BOOL** SwiLocGetAutoStartResp::max_time_reported

8.777 SwiLocSetAutoStartReq Struct Reference

Data Fields

- [BYTE](#) *function*
- [BOOL](#) *set_function*
- [BYTE](#) *fix_type*
- [BOOL](#) *set_fix_type*
- [BYTE](#) *max_time*
- [BOOL](#) *set_max_time*
- [ULONG](#) *max_dist*
- [BOOL](#) *set_max_dist*
- [ULONG](#) *fix_rate*
- [BOOL](#) *set_fix_rate*

8.777.1 Detailed Description

This structure contains SWI LOC Get Auto Start setting

Parameters

<i>function</i>	<ul style="list-style-type: none"> • Setting to indicate when modem should start an automatic GNSS fix <ul style="list-style-type: none"> – 0 - disabled – 1 - At bootup – 2 - When NMEA port is opened
<i>set_function</i>	<ul style="list-style-type: none"> • 0 - do not set to modem • 1 - set to modem
<i>fix_type</i>	<ul style="list-style-type: none"> • Type of GNSS fix: <ul style="list-style-type: none"> – 1 - Default Engine mode – 2 - MS-Based – 3 - MS-Assisted – 4 - Standalone
<i>set_fix_type</i>	<ul style="list-style-type: none"> • 0 - do not set to modem • 1 - set to modem
<i>max_time</i>	<ul style="list-style-type: none"> • Maximum time allowed for the receiver to get a fix in seconds • Valid range: 1-255
<i>set_max_time</i>	<ul style="list-style-type: none"> • 0 - do not set to modem • 1 - set to modem

<i>max_dist</i>	<ul style="list-style-type: none"> • Maximum uncertainty of a fix measured by distance in meters • Valid range: 1 - 4294967280
<i>set_max_dist</i>	<ul style="list-style-type: none"> • 0 - do not set to modem • 1 - set to modem
<i>fix_rate</i>	<ul style="list-style-type: none"> • Time between fixes in seconds • Valid range: 1–65535
<i>set_fix_rate</i>	<ul style="list-style-type: none"> • 0 - do not set to modem • 1 - set to modem

8.777.2 Field Documentation

8.777.2.1 **ULONG** SwiLocSetAutoStartReq::fix_rate

8.777.2.2 **BYTE** SwiLocSetAutoStartReq::fix_type

8.777.2.3 **BYTE** SwiLocSetAutoStartReq::function

8.777.2.4 **ULONG** SwiLocSetAutoStartReq::max_dist

8.777.2.5 **BYTE** SwiLocSetAutoStartReq::max_time

8.777.2.6 **BOOL** SwiLocSetAutoStartReq::set_fix_rate

8.777.2.7 **BOOL** SwiLocSetAutoStartReq::set_fix_type

8.777.2.8 **BOOL** SwiLocSetAutoStartReq::set_function

8.777.2.9 **BOOL** SwiLocSetAutoStartReq::set_max_dist

8.777.2.10 **BOOL** SwiLocSetAutoStartReq::set_max_time

8.778 swiModemStatusResp Struct Reference

Data Fields

- [CommInfo](#) [commonInfo](#)
- [LTEInfo](#) * [pLTEInfo](#)

8.778.1 Detailed Description

Structure for storing the SLQS Nas Swi Modem Status response parameters.

Parameters

<i>commonInfo</i>	(mandatory) <ul style="list-style-type: none"> See CommInfo for more information
<i>pLTEInfo</i>	(optional) <ul style="list-style-type: none"> See LTEInfo for more information

8.778.2 Field Documentation

8.778.2.1 CommInfo swiModemStatusResp::commonInfo

8.778.2.2 LTEInfo* swiModemStatusResp::pLTEInfo

8.779 SwiOTAMsg_s Struct Reference

Data Fields

- [ULONG](#) type
- [WORD](#) data_len
- [BYTE](#) data [2048]
- [LteNasReleaseInfo](#) * [pLteNasRelInfo](#)
- [ULONGLONG](#) * [pTime](#)

8.779.1 Detailed Description

This structure contains OTA message

Parameters

<i>type</i>	<ul style="list-style-type: none"> message type <ul style="list-style-type: none"> 0 - LTE ESM uplink 1 - LTE ESM downlink 2 - LTE EMM uplink 3 - LTE EMM downlink 4 - GSM/UMTS uplink 5 - GSM/UMTS downlink
<i>data_len</i>	<ul style="list-style-type: none"> OTA Message Content Length
<i>data</i>	<ul style="list-style-type: none"> OTA Message Content
<i>pLteNasRelInfo</i>	<ul style="list-style-type: none"> LTE NAS Release Info see LteNasReleaseInfo for details
<i>pTime</i>	<ul style="list-style-type: none"> Seconds in local time since Jan. 6th 1980 00:00:00 UTC

8.779.2 Field Documentation

8.779.2.1 **BYTE** SwiOTAMsg_s::data[2048]

8.779.2.2 **WORD** SwiOTAMsg_s::data_len

8.779.2.3 **LteNasReleaseInfo*** SwiOTAMsg_s::pLteNasRelInfo

8.779.2.4 **ULONGLONG*** SwiOTAMsg_s::pTime

8.779.2.5 **ULONG** SwiOTAMsg_s::type

8.780 swiPDPRuntimeSettingsReq Struct Reference

Data Fields

- [BYTE contextId](#)
- [BYTE contextType](#)

8.780.1 Detailed Description

This structure contains the PDP Runtime Settings Request parameters.

Parameters

<i>contextId</i>	<ul style="list-style-type: none">• Context Identifier
<i>v4sessionId</i>	<ul style="list-style-type: none">• The v4 session ID for which the runtime settings are to be retrieved• provide a NULL pointer if not applicable
<i>v6sessionId</i>	<ul style="list-style-type: none">• The v6 session ID for which the runtime settings are to be retrieved• provide a NULL pointer if not applicable

8.780.2 Field Documentation

8.780.2.1 **BYTE** swiPDPRuntimeSettingsReq::contextId

8.780.2.2 **BYTE** swiPDPRuntimeSettingsReq::contextType

8.781 swiPDPRuntimeSettingsResp Struct Reference

Data Fields

- BYTE * *pContextId*
- BYTE * *pBearerId*
- CHAR * *pAPNName*
- ULONG * *pIPv4Address*
- ULONG * *pIPv4GWAddress*
- ULONG * *pPrDNSIPv4Address*
- ULONG * *pSeDNSIPv4Address*
- struct *IPv6AddressInfo* * *pIPv6Address*
- struct *IPv6AddressInfo* * *pIPv6GWAddress*
- WORD * *pPrDNSIPv6Address*
- WORD * *pSeDNSIPv6Address*
- ULONG * *pPrPCSCFIPv4Address*
- ULONG * *pSePCSCFIPv4Address*
- WORD * *pPrPCSCFIPv6Address*
- WORD * *pSePCSCFIPv6Address*

8.781.1 Detailed Description

This structure contains the response parameters retrieved by the API `SLQSWdsSwiPDPRuntimeSettings`

Parameters

<i>pContextId</i>	(optional) <ul style="list-style-type: none"> Context Identifier <ul style="list-style-type: none"> 0xFF - Not Available
<i>pBearerId</i>	(optional) <ul style="list-style-type: none"> Bearer Identity An EPS bearer identity uniquely identifies an EPS bearer for one UE accessing via E-UTRAN. The EPS Bearer Identity is allocated by the MME. <ul style="list-style-type: none"> 0xFF - Not Available
<i>pAPNName</i>	(optional) <ul style="list-style-type: none"> APN name associated with the context id <ul style="list-style-type: none"> NULL terminated by default.
<i>pIPv4Address</i>	(optional) <ul style="list-style-type: none"> IPv4 Address <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>pIPv4GW↵Address</i>	(optional) <ul style="list-style-type: none"> IPv4 Gateway Address <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>pPrDNSIPv4↵Address</i>	(optional) <ul style="list-style-type: none"> Primary DNS IPv4 Address <ul style="list-style-type: none"> 0xFFFF - Not Available
<i>pSeDNSIPv4↵Address</i>	(optional) <ul style="list-style-type: none"> Secondary DNS IPv4 Address <ul style="list-style-type: none"> 0xFFFF - Not Available

<i>pIPv6Address</i>	(optional) <ul style="list-style-type: none"> IPv6 Address See IPv6AddressInfo for more information
<i>pIPv6GW↔ Address</i>	(optional) <ul style="list-style-type: none"> IPv6 Gateway Address See IPv6AddressInfo for more information
<i>pPrDNSIPv6↔ Address</i>	(optional) <ul style="list-style-type: none"> Primary IPv6 DNS Address(in network byte order) This is an 8-element array of 16-bit numbers, each of which is in big-endian format
<i>pSeDNSIPv6↔ Address</i>	(optional) <ul style="list-style-type: none"> Secondary IPv6 DNS Address(in network byte order) This is an 8-element array of 16-bit numbers, each of which is in big-endian format
<i>pPrPCSCFI↔ Pv4Address</i>	(optional) <ul style="list-style-type: none"> Primary PCSCF IPv4 Address
<i>pSePCSCFI↔ Pv4Address</i>	(optional) <ul style="list-style-type: none"> Secondary PCSCF IPv4 Address
<i>pPrPCSCFI↔ Pv6Address</i>	(optional) <ul style="list-style-type: none"> Primary PCSCF IPv6 Address This is an 8-element array of 16-bit numbers, each of which is in big-endian format
<i>pSePCSCFI↔ Pv6Address</i>	(optional) <ul style="list-style-type: none"> Secondary PCSCF IPv6 Address This is an 8-element array of 16-bit numbers, each of which is in big-endian format

Note

Parameters which are mentioned as NULL will be ignored.

8.781.2 Field Documentation

8.781.2.1 CHAR* swiPDPRuntimeSettingsResp::pAPNName

8.781.2.2 BYTE* swiPDPRuntimeSettingsResp::pBearerId

8.781.2.3 BYTE* swiPDPRuntimeSettingsResp::pContextId

8.781.2.4 ULONG* swiPDPRuntimeSettingsResp::pIPv4Address

8.781.2.5 ULONG* swiPDPRuntimeSettingsResp::pIPv4GWAddress

8.781.2.6 struct [IPv6AddressInfo](#)* swiPDPRuntimeSettingsResp::pIPv6Address

8.781.2.7 struct **IPv6AddressInfo*** swiPDPRuntimeSettingsResp::pIPv6GWAddress

8.781.2.8 ULONG* swiPDPRuntimeSettingsResp::pPrDNSIPv4Address

8.781.2.9 WORD* swiPDPRuntimeSettingsResp::pPrDNSIPv6Address

8.781.2.10 ULONG* swiPDPRuntimeSettingsResp::pPrPCSCFIPv4Address

8.781.2.11 WORD* swiPDPRuntimeSettingsResp::pPrPCSCFIPv6Address

8.781.2.12 ULONG* swiPDPRuntimeSettingsResp::pSeDNSIPv4Address

8.781.2.13 WORD* swiPDPRuntimeSettingsResp::pSeDNSIPv6Address

8.781.2.14 ULONG* swiPDPRuntimeSettingsResp::pSePCSCFIPv4Address

8.781.2.15 WORD* swiPDPRuntimeSettingsResp::pSePCSCFIPv6Address

8.782 swiQosFilter Struct Reference

Data Fields

- [BYTE](#) index
- [BYTE](#) version
- [IPv4Addr](#) * [pIPv4SrcAddr](#)
- [IPv4Addr](#) * [pIPv4DstAddr](#)
- [BYTE](#) * [pNxtHdrProto](#)
- [Tos](#) * [pTos](#)
- [IPv6Addr](#) * [pIPv6SrcAddr](#)
- [IPv6Addr](#) * [pIPv6DstAddr](#)
- [IPv6TrafCls](#) * [pIPv6TrafCls](#)
- [ULONG](#) * [pIPv6Label](#)
- [Port](#) * [pTCPSrcPort](#)
- [Port](#) * [pTCPDstPort](#)
- [Port](#) * [pUDPSrcPort](#)
- [Port](#) * [pUDPDstPort](#)
- [ULONG](#) * [pEspSpi](#)
- [WORD](#) * [pPrecedence](#)
- [WORD](#) * [pId](#)
- [Port](#) * [pTranSrcPort](#)
- [Port](#) * [pTranDstPort](#)

8.782.1 Detailed Description

This structure contains the QoS Filter Request

Parameters

<i>index</i>	IP filter index Integer that uniquely identifies each filter instance This TLV must be present in the request
<i>version</i>	IP filter version Identifies whether the filter is associated with IPv4 or IPv6; value specified also implies that only TLVs defined for that IP version, i.e., TLVs with IPv4 or IPv6 in the name, can be specified <ul style="list-style-type: none"> • 0x04 – IPv4 • 0x06 – Ipv6
<i>pIPv4SrcAddr</i>	IPv4 filter soruce address See IPv4Addr for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>pIPv4DstAddr</i>	IPv4 filter destination address See IPv4Addr for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>pNxtHdrProto</i>	IP filter next header protocol This TLV must be present if any non-IP filter TLV(s) are provided If this field is specified, only IP packets belonging to specified higher layer protocol are considered when filtering The following protocols may be specified: <ul style="list-style-type: none"> • 0x01 = ICMP • 0x06 = TCP • 0x11 = UDP • 0x32 = ESP Note: The next header protocol field will be set to 0xFD (TCP & UDP) if a TFT is received specifying a source or destination port number, but IP next header type is not specified.
<i>pTos</i>	IPv4 filter type of service See Tos for more information
<i>pIPv6SrcAddr</i>	IPv6 filter soruce address See IPv6Addr for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>pIPv6DstAddr</i>	IPv6 filter destination address See IPv6Addr for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>pIPv6TrafCls</i>	IPv6 filter traffic class See IPv6TrafCls for more information
<i>pIPv6Label</i>	IPv6 flow label Packet matches the IPv6 flow label filter if: (*pIPv6Label == flow label in the IPv6 header) <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>pTCPSrcPort</i>	TCP filter source port filter See Port for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>pTCPDstPort</i>	TCP filter destination port filter See Port for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>pUDPSrcPort</i>	UDP filter source port filter See Port for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>pUDPDstPort</i>	UDP filter destination port filter See Port for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>pEspSpi</i>	ESP filter security policy index Security policy index to uniquely identify each IP flow for filtering encrypted packets for encapsulating security payload <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>pPrecedence</i>	Filter Precedence Specifies the order in which filters are applied; lower numerical value has higher precedence Note: This TLV only applies to network-initiated QoS; QoS requests containing this TLV from control points will be ignored

<i>pId</i>	Filter ID Unique identifier for each filter;filter ID is assigned by the modem Note: This TLV only applies to network-initiated QoS; QoS requests containing this TLV from control points will be ignored
<i>pTranSrcPort</i>	Transport protocol filter source port See Port for more information <ul style="list-style-type: none"> Implemented only for unsolicited indication
<i>pUDPDstPort</i>	Transport protocol filter destination port See Port for more information <ul style="list-style-type: none"> Implemented only for unsolicited indication

8.782.2 Field Documentation

8.782.2.1 **BYTE** swiQosFilter::index

8.782.2.2 **ULONG*** swiQosFilter::pEspSpi

8.782.2.3 **WORD*** swiQosFilter::pId

8.782.2.4 **IPv4Addr*** swiQosFilter::pIPv4DstAddr

8.782.2.5 **IPv4Addr*** swiQosFilter::pIPv4SrcAddr

8.782.2.6 **IPv6Addr*** swiQosFilter::pIPv6DstAddr

8.782.2.7 **ULONG*** swiQosFilter::pIPv6Label

8.782.2.8 **IPv6Addr*** swiQosFilter::pIPv6SrcAddr

8.782.2.9 **IPv6TrafCls*** swiQosFilter::pIPv6TrafCls

8.782.2.10 **BYTE*** swiQosFilter::pNextHdrProto

8.782.2.11 **WORD*** swiQosFilter::pPrecedence

8.782.2.12 **Port*** swiQosFilter::pTCPDstPort

8.782.2.13 **Port*** swiQosFilter::pTCPsrcPort

8.782.2.14 **Tos*** swiQosFilter::pTos

8.782.2.15 **Port*** swiQosFilter::pTranDstPort

8.782.2.16 **Port*** swiQosFilter::pTranSrcPort

8.782.2.17 **Port*** swiQosFilter::pUDPDstPort

8.782.2.18 **Port*** swiQosFilter::pUDPsrcPort

8.782.2.19 **BYTE** swiQosFilter::version

8.783 swiQosFlow Struct Reference

Data Fields

- [BYTE](#) index

- WORD * pProfileId3GPP2
- BYTE * p3GPP2Pri
- BYTE * pTrafficClass
- dataRate * pDataRate
- tokenBucket * pTokenBucket
- ULONG * pLatency
- ULONG * pJitter
- pktErrRate * pPktErrRate
- ULONG * pMinPolicedPktSz
- ULONG * pMaxAllowedPktSz
- WORD * p3GPPResResidualBER
- BYTE * p3GPPTraHdlPri
- BYTE * p3GPPImCn
- BYTE * p3GPPSigInd
- BYTE * pLteQci

8.783.1 Detailed Description

This structure contains the QoS Flow Request

Parameters

<i>index</i>	<ul style="list-style-type: none"> • IP flow index • Integer that uniquely identifies each flow instance • Unique index must be assigned by the control point to every flow_spec instance
<i>pProfileId3GPP2</i>	<ul style="list-style-type: none"> • IP flow 3GPP2 profile ID • A profile ID is shorthand for a defined set of QoS flow parameters specified by the network; to be present while requesting QoS for a CDMA device
<i>p3GPP2Pri</i>	<ul style="list-style-type: none"> • IP flow 3GPP2 flow priority • Flow priority used by the network in case of contention between flows with same QoS; this parameter applies for CDMA devices
<i>pTrafficClass</i>	<ul style="list-style-type: none"> • IP flow traffic class • Integer that designates the requested traffic class: <ul style="list-style-type: none"> • 0 – Conversational • 1 – Streaming • 2 – Interactive • 3 – Background
<i>pDataRate</i>	<ul style="list-style-type: none"> • IP flow data rate min max • See dataRate for more information
<i>pTokenBucket</i>	<ul style="list-style-type: none"> • IP flow data rate token bucket • See tokenBucket for more information

<i>pLatency</i>	<ul style="list-style-type: none"> • IP flow latency • Maximum delay (in milliseconds) that can be tolerated by an IP packet during transfer through the wireless link
<i>pJitter</i>	<ul style="list-style-type: none"> • IP flow jitter • Difference between the maximum and minimum latency (in milliseconds) that can be tolerated by an IP packet during the transfer through the wireless link
<i>pPktErrRate</i>	<ul style="list-style-type: none"> • IP flow packet error rate • See pktErrRate for more information
<i>pMinPoliced</i> ↔ <i>PktSz</i>	<ul style="list-style-type: none"> • IP flow minimum policed packet size • Integer that defines the minimum packet size (in bytes) that will be policed for QoS guarantees; any IP packets that are smaller than the minimum specified policed size may not receive requested QoS
<i>pMaxAllowed</i> ↔ <i>PktSz</i>	<ul style="list-style-type: none"> • IP flow maximum allowed packet size • Integer that defines the maximum packet size (in bytes) allowed in the IP flow; any IP packets greater in size than the maximum allowed packet size are not queued for transmission
<i>p3GPPRes</i> ↔ <i>ResidualBER</i>	<ul style="list-style-type: none"> • IP flow 3GPP residual bit error rate • residual_bit_error_rate • 0 = 5*10⁻² residual BER • 1 = 1*10⁻² residual BER • 2 = 5*10⁻³ residual BER • 3 = 4*10⁻³ residual BER • 4 = 1*10⁻³ residual BER • 5 = 1*10⁻⁴ residual BER • 6 = 1*10⁻⁵ residual BER • 7 = 1*10⁻⁶ residual BER • 8 = 6*10⁻⁸ residual BER • Integer that indicates the undetected BER for each IP flow in the delivered packets; Applies only to 3GPP networks
<i>p3GPPTraHdlPri</i>	<ul style="list-style-type: none"> • 3GPP traffic handling priority • 0 – Relative traffic handling priority 1 • 1 – Relative traffic handling priority 2 • 2 – Relative traffic handling priority 3 • Defines the relative priority of the flow; applies only to 3GPP networks

<i>p3GPPImCn</i>	<ul style="list-style-type: none"> • IP flow 3GPP IM CN flag • IM CN subsystem signaling flag: • 0x00 – FALSE • 0x01 – TRUE • This parameter applies only to 3GPP networks
<i>p3GPPSigInd</i>	<ul style="list-style-type: none"> • IP flow 3GPP signaling indication • 0x00 – FALSE • 0x01 – TRUE • This parameter applies only to 3GPP networks
<i>pLteQci</i>	<ul style="list-style-type: none"> • LTE QoS Class Identifier • QoS Class Identifier(QCI) is a required parameter to request QoS in LTE • QCI values: <ul style="list-style-type: none"> – QCI value 0 requests the network to assign the appropriate QCI value – QCI values 1-4 are associated with guaranteed bitrates – QCI values 5-9 are associated with nonguaranteed bitrates, so the values specified as guaranteed and maximum bitrates are ignored

8.783.2 Field Documentation

8.783.2.1 **BYTE** swiQosFlow::index

8.783.2.2 **BYTE*** swiQosFlow::p3GPP2Pri

8.783.2.3 **BYTE*** swiQosFlow::p3GPPImCn

8.783.2.4 **WORD*** swiQosFlow::p3GPPResResidualBER

8.783.2.5 **BYTE*** swiQosFlow::p3GPPSigInd

8.783.2.6 **BYTE*** swiQosFlow::p3GPPTraHdlPri

8.783.2.7 **dataRate*** swiQosFlow::pDataRate

8.783.2.8 **ULONG*** swiQosFlow::pJitter

8.783.2.9 **ULONG*** swiQosFlow::pLatency

8.783.2.10 **BYTE*** swiQosFlow::pLteQci

8.783.2.11 **ULONG*** swiQosFlow::pMaxAllowedPktSz

8.783.2.12 **ULONG*** `swiQosFlow::pMinPolicedPktSz`

8.783.2.13 **pkErrRate*** `swiQosFlow::pPktErrRate`

8.783.2.14 **WORD*** `swiQosFlow::pProfileId3GPP2`

8.783.2.15 **tokenBucket*** `swiQosFlow::pTokenBucket`

8.783.2.16 **BYTE*** `swiQosFlow::pTrafficClass`

8.784 swiQosGranted Struct Reference

Data Fields

- [swiQosFlow](#) * `pTxFlow`
- [swiQosFlow](#) * `pRxFlow`

8.784.1 Detailed Description

This structure contains the QoS granted flow

Parameters

<i>pTxFlow</i>	See swiQosFlow for more information
<i>pRxFlow</i>	See swiQosFlow for more information

8.784.2 Field Documentation

8.784.2.1 **swiQosFlow*** `swiQosGranted::pRxFlow`

8.784.2.2 **swiQosFlow*** `swiQosGranted::pTxFlow`

8.785 swiQosIds Struct Reference

Data Fields

- **BYTE** `sz`
- **ULONG** * `pIds`

8.785.1 Detailed Description

This structure contains the QoS Response parameters.

Parameters

<i>sz</i>	Number of QoS identifiers
<i>plds</i>	Identifier for the QoS flow requested; number of QoS identifiers present will be equal to number of QoS specs requested in the QoS Request message

8.785.2 Field Documentation

8.785.2.1 **ULONG*** swiQoslds::plds

8.785.2.2 **BYTE** swiQoslds::sz

8.786 swiQosModifyReq Struct Reference

Data Fields

- [ULONG](#) id
- [swiQosFlow](#) * pTxFlow
- [swiQosFlow](#) * pRxFlow
- [swiQosFilter](#) * pTxFilter
- [swiQosFilter](#) * pRxFilter

8.786.1 Detailed Description

This structure contains the QoS Request parameters.

Parameters

<i>id</i>	Identifier for the QoS flow/instance that has been negotiated and that needs to be modified The QoS identifier is used to reference the actual flow/filter specifications that are in effect as a result of the negotiation triggered by QMI_QOS_REQUEST_QOS_REQ
<i>pTxFlow</i>	See swiQosFlow for more information
<i>pRxFlow</i>	See swiQosFlow for more information
<i>pTxFilter</i>	See swiQosFilter for more information
<i>pRxFilter</i>	See swiQosFilter for more information

8.786.2 Field Documentation

8.786.2.1 **ULONG** swiQosModifyReq::id

8.786.2.2 **swiQosFilter*** swiQosModifyReq::pRxFilter

8.786.2.3 **swiQosFlow*** swiQosModifyReq::pRxFlow

8.786.2.4 **swiQosFilter*** swiQosModifyReq::pTxFilter

8.786.2.5 **swiQosFlow*** swiQosModifyReq::pTxFlow

8.787 swiQosReq Struct Reference

Data Fields

- [BYTE index](#)
- [swiQosFlow](#) * [pTxFlow](#)
- [swiQosFlow](#) * [pRxFlow](#)
- [swiQosFilter](#) * [pTxFilter](#)
- [swiQosFilter](#) * [pRxFilter](#)

8.787.1 Detailed Description

This structure contains the QoS Request parameters.

Parameters

<i>index</i>	<ul style="list-style-type: none"> • An integer that uniquely identifies each QoS spec included in the QoS request message
<i>pTxFlow</i>	<ul style="list-style-type: none"> • See swiQosFlow for more information
<i>pRxFlow</i>	<ul style="list-style-type: none"> • See swiQosFlow for more information
<i>pTxFilter</i>	<ul style="list-style-type: none"> • See swiQosFilter for more information
<i>pRxFilter</i>	<ul style="list-style-type: none"> • See swiQosFilter for more information

8.787.2 Field Documentation

8.787.2.1 **BYTE** `swiQosReq::index`

8.787.2.2 **swiQosFilter*** `swiQosReq::pRxFilter`

8.787.2.3 **swiQosFlow*** `swiQosReq::pRxFlow`

8.787.2.4 **swiQosFilter*** `swiQosReq::pTxFilter`

8.787.2.5 **swiQosFlow*** `swiQosReq::pTxFlow`

8.788 swiRMTrasferStaticsReq Struct Reference

Data Fields

- [BYTE bResetStatistics](#)
- [ULONG ulMask](#)

8.788.1 Detailed Description

RM Transfer Statistics Structure

Parameters

<i>bResetStatistics</i>	<ul style="list-style-type: none">• Reset Statistics• Values:• 0 - Not Reset• Other - Reset
<i>ulMask</i>	<ul style="list-style-type: none">• Enable/Disable RM Transfer Statistics Indiscation Mask• Bit 0: Tx Packet Ok• Bit 1: Rx Packet Ok• Bit 2: Tx Bytes Ok• Bit 3: Rx Bytes Ok• Bit 4: Tx Packets Dropped• Bit 5: Rx Packets Dropped• Value: -0 - Disable -1 - Enable

8.788.2 Field Documentation

8.788.2.1 **BYTE** swiRMTrasnferStaticsReq::bResetStatistics

8.788.2.2 **ULONG** swiRMTrasnferStaticsReq::ulMask

8.789 sysInfoCommon Struct Reference

Data Fields

- [BYTE](#) `srvDomainValid`
- [BYTE](#) `srvDomain`
- [BYTE](#) `srvCapabilityValid`
- [BYTE](#) `srvCapability`
- [BYTE](#) `roamStatusValid`
- [BYTE](#) `roamStatus`
- [BYTE](#) `isSysForbiddenValid`
- [BYTE](#) `isSysForbidden`

8.789.1 Detailed Description

Structure for storing the System Information common to CDMA, HDR, GSM, WCDMA and LTE networks.

Parameters

<i>srvDomainValid</i>	<ul style="list-style-type: none"> Indicates whether the service domain is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>srvDomain</i>	<ul style="list-style-type: none"> Service domain registered on the system. <ul style="list-style-type: none"> 0x00 - No service 0x01 - Circuit-switched only 0x02 - Packet-switched only 0x03 - Circuit-switched and packet-switched 0x04 - Camped 0xFF - Not Available
<i>srvCapability</i> ↔ <i>Valid</i>	<ul style="list-style-type: none"> Indicates whether the service capability is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>srvCapability</i>	<ul style="list-style-type: none"> Current system's service capability. <ul style="list-style-type: none"> 0x00 - No service 0x01 - Circuit-switched only 0x02 - Packet-switched only 0x03 - Circuit-switched and packet-switched 0x04 - Camped 0xFF - Not Available
<i>roamStatusValid</i>	<ul style="list-style-type: none"> Indicates whether the roaming status is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available

<i>roamStatus</i>	<ul style="list-style-type: none"> • Current roaming status. <ul style="list-style-type: none"> – 0x00 - Off – 0x01 - On – 0x02 - Blinking – 0x03 - Out of the neighborhood – 0x04 - Out of the building – 0x05 - Preferred system – 0x06 - Available system – 0x07 - Alliance partner – 0x08 - Premium partner – 0x09 - Full service – 0x0A - Partial service – 0x0B - Banner is on – 0x0C - Banner is off – 0x0D to 0x3F - Reserved for Standard Enhanced Roaming Indicator Numbers – 0x40 to 0x7F - Reserved for Non-Standard Enhanced Roaming Indicator Numbers – 0x40 to 0xFF - Reserved. – 0xFF - Not Available • Values from 0x02 onward are only applicable for 3GPP2
<i>isSysForbiddenValid</i>	<ul style="list-style-type: none"> • Indicates whether the forbidden system is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>isSysForbidden</i>	<ul style="list-style-type: none"> • Whether the system is forbidden. <ul style="list-style-type: none"> – 0x00 - Not forbidden – 0x01 - Forbidden – 0xFF - Not Available

8.789.2 Field Documentation

8.789.2.1 **BYTE** sysInfoCommon::isSysForbidden

8.789.2.2 **BYTE** sysInfoCommon::isSysForbiddenValid

8.789.2.3 **BYTE** sysInfoCommon::roamStatus

8.789.2.4 **BYTE** sysInfoCommon::roamStatusValid

8.789.2.5 **BYTE** sysInfoCommon::srvCapability

8.789.2.6 **BYTE** sysInfoCommon::srvCapabilityValid

8.789.2.7 **BYTE** sysInfoCommon::srvDomain

8.789.2.8 **BYTE** sysInfoCommon::srvDomainValid

8.790 t_gpsTime Struct Reference

Data Fields

- [USHORT](#) gpsWeek
- [ULONG](#) gpsTimeOfWeekMs

8.790.1 Field Documentation

8.790.1.1 **ULONG** t_gpsTime::gpsTimeOfWeekMs

8.790.1.2 **USHORT** t_gpsTime::gpsWeek

8.791 t_sensor Struct Reference

Data Fields

- [ULONG](#) usageMask
- [ULONG](#) aidingIndicatorMask

8.791.1 Field Documentation

8.791.1.1 **ULONG** t_sensor::aidingIndicatorMask

8.791.1.2 **ULONG** t_sensor::usageMask

8.792 t_Sv Struct Reference

Data Fields

- [BYTE](#) len
- [USHORT](#) entries [255]

8.792.1 Field Documentation

8.792.1.1 **USHORT** t_Sv::entries[255]

8.792.1.2 **BYTE** t_Sv::len

8.793 TDSCDMAECIOThresh Struct Reference

Data Fields

- [BYTE](#) TDSCDMAECIOThreshListLen
- [ULONG](#) * pTDSCDMAECIOThreshList

8.793.1 Detailed Description

This structure contains TDSCDMA ECIO threshold related parameters.

Parameters

<i>TDSCDMAECIOThreshListLen</i>	<ul style="list-style-type: none"> Length of the TDSCDMA ECIO threshold list parameter to follow
<i>pTDSCDMAECIOThreshList</i>	<ul style="list-style-type: none"> Array of ECIO thresholds (in dB) used by TD-SCDMA Maximum of 32 values.

8.793.2 Field Documentation

8.793.2.1 **ULONG*** TDSCDMAECIOThresh::pTDSCDMAECIOThreshList

8.793.2.2 **BYTE** TDSCDMAECIOThresh::TDSCDMAECIOThreshListLen

8.794 TDSCDMARSCPThresh Struct Reference

Data Fields

- [BYTE](#) TDSCDMARSCPThreshListLen
- [WORD *](#) pTDSCDMARSCPThreshList

8.794.1 Detailed Description

This structure contains TDSCDMA RSCP threshold related parameters.

Parameters

<i>TDSCDMARSCPThreshListLen</i>	<ul style="list-style-type: none"> Length of the TDSCDMA RSCP threshold list parameter to follow
<i>pTDSCDMARSCPThreshList</i>	<ul style="list-style-type: none"> Array of RSCP thresholds (in units of 0.1 dBm) Maximum of 32 values Range for RSCP values: -120 to -25 (in dBm).

8.794.2 Field Documentation

8.794.2.1 **WORD*** TDSCDMARSCPThresh::pTDSCDMARSCPThreshList

8.794.2.2 **BYTE** TDSCDMARSCPThresh::TDSCDMARSCPThreshListLen

8.795 TDSCDMARSSIThresh Struct Reference

Data Fields

- [BYTE](#) TDSCDMARSSIThreshListLen
- [ULONG](#) * pTDSCDMARSSIThreshList

8.795.1 Detailed Description

This structure contains TDSCDMA RSSI threshold related parameters.

Parameters

<i>TDSCDMARSSIThreshListLen</i>	<ul style="list-style-type: none"> • Length of the TDSCDMA RSSI threshold list parameter to follow
<i>pTDSCDMARSSIThreshList</i>	<ul style="list-style-type: none"> • Array of RSSI thresholds (in dBm) used by TD-SCDMA • Maximum of 32 values.

8.795.2 Field Documentation

8.795.2.1 **ULONG*** TDSCDMARSSIThresh::pTDSCDMARSSIThreshList

8.795.2.2 **BYTE** TDSCDMARSSIThresh::TDSCDMARSSIThreshListLen

8.796 TDSCDMASigInfoExt Struct Reference

Data Fields

- [FLOAT](#) rssi
- [FLOAT](#) rscp
- [FLOAT](#) ecio
- [FLOAT](#) sinr

8.796.1 Detailed Description

This structure contains the TDSCDMA Signal Strength Info Extended

Parameters

<i>rssI</i>	<ul style="list-style-type: none">Measured RSSI in dB
<i>rscp</i> [Optional]	<ul style="list-style-type: none">Measured RSCP in dBm
<i>ecio</i> [Optional]	<ul style="list-style-type: none">Measured ECIO in dBm.
<i>sinr</i> [Optional]	<ul style="list-style-type: none">Measured SINR in dB. -15 dB is sent to clients if the actual SINR is less than -15 dB

8.796.2 Field Documentation

8.796.2.1 FLOAT TDSCDMASigInfoExt::ecio

8.796.2.2 FLOAT TDSCDMASigInfoExt::rscp

8.796.2.3 FLOAT TDSCDMASigInfoExt::rssI

8.796.2.4 FLOAT TDSCDMASigInfoExt::sinr

8.797 tdscdmaSigInfoExt Struct Reference

Data Fields

- float [rssI](#)
- float [rscp](#)
- float [ecio](#)
- float [sinr](#)

8.797.1 Detailed Description

Parameters

<i>rssI</i>	RSSI in dBm.
<i>rsrq</i>	RSRQ value in dB
<i>rsrp</i>	Current RSRP in dBm as measured by L1.
<i>snr</i>	SNR level as a scaled integer in units of 0.1 dB.

8.797.2 Field Documentation

8.797.2.1 float tdscdmaSigInfoExt::ecio

8.797.2.2 float tdscdmaSigInfoExt::rscp

8.797.2.3 float tdscdmaSigInfoExt::rssi

8.797.2.4 float tdscdmaSigInfoExt::sinr

8.798 TDSCDMASINRCONFThresh Struct Reference

Data Fields

- [BYTE TDSCDMASINRCONFThreshListLen](#)
- [FLOAT * pTDSCDMASINRCONFThreshList](#)

8.798.1 Detailed Description

This structure contains TDSCDMA SINR threshold related parameters.

Parameters

<i>TDSCDMASINRCONFThreshListLen</i>	<ul style="list-style-type: none"> • Length of the TDSCDMA SINR threshold list parameter to follow
<i>pTDSCDMASINRCONFThreshList</i>	<ul style="list-style-type: none"> • Array of SINR thresholds (in dB) used by TD-SCDMA • Maximum of 32 values

8.798.2 Field Documentation

8.798.2.1 **FLOAT*** TDSCDMASINRCONFThresh::pTDSCDMASINRCONFThreshList

8.798.2.2 **BYTE** TDSCDMASINRCONFThresh::TDSCDMASINRCONFThreshListLen

8.799 TDSCDMASINRThresh Struct Reference

Data Fields

- [BYTE TDSCDMASINRThreshListLen](#)
- [ULONG * pTDSCDMASINRThreshList](#)

8.799.1 Detailed Description

This structure contains TDSCDMA SINR threshold related parameters.

Parameters

<i>TDSCDMASINRThreshListLen</i>	<ul style="list-style-type: none"> • Length of the TDSCDMA SINR threshold list parameter to follow
<i>pTDSCDMASINRThreshList</i>	<ul style="list-style-type: none"> • Array of SINR thresholds (in dB) used by TD-SCDMA • Maximum of 32 values

8.799.2 Field Documentation

8.799.2.1 ULONG* TDSCDMASINRThresh::pTDSCDMASINRThreshList

8.799.2.2 BYTE TDSCDMASINRThresh::TDSCDMASINRThreshListLen

8.800 tempData_t Struct Reference

Data Fields

- uint32_t [timeSource](#)
- uint32_t [timeOfFirstSample](#)
- uint8_t [temperatureDataLen](#)
- uint16_t [timeOffset](#) [64]
- uint32_t [temperature](#) [64]

8.800.1 Detailed Description

This structure specifies information regarding the Temperature Data. Please check has_<Param_Name> field for presence of optional parameters

Parameters

<i>timeSource</i>	<ul style="list-style-type: none"> • Time source of the sensor data • Valid values <ul style="list-style-type: none"> – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine
<i>timeOfFirstSample</i>	<ul style="list-style-type: none"> • Denotes a full 32-bit time stamp of the first (oldest) sample in this message. • The time stamp is in the time reference scale that is used by the sensor time source. • Units - Milliseconds
<i>temperatureDataLen</i>	<ul style="list-style-type: none"> • Number of sets of the following elements <ul style="list-style-type: none"> – timeOffset – temperature
<i>timeOffset</i>	<ul style="list-style-type: none"> • Sample time offset • Units - Milliseconds
<i>temperature</i>	<ul style="list-style-type: none"> • Sensor temperature. • Type - Floating point • Units - Degrees Celsius • Range -50 to +100.00

8.800.2 Field Documentation

8.800.2.1 uint32_t tempData_t::temperature[64]

8.800.2.2 uint8_t tempData_t::temperatureDataLen

8.800.2.3 uint32_t tempData_t::timeOfFirstSample

8.800.2.4 uint16_t tempData_t::timeOffset[64]

8.800.2.5 uint32_t tempData_t::timeSource

8.801 tempratureData Struct Reference

Data Fields

- [ULONG timeSource](#)
- [ULONG timeOfFirstSample](#)
- [BYTE temperatureDataLen](#)
- [WORD timeOffset \[64\]](#)
- [ULONG temperature \[64\]](#)

8.801.1 Detailed Description

This structure specifies information regarding the Temperature Data.

Parameters

<i>timeSource</i>	<ul style="list-style-type: none"> • Time source of the sensor data • Valid values <ul style="list-style-type: none"> – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine
<i>timeOfFirstSample</i>	<ul style="list-style-type: none"> • Denotes a full 32-bit time stamp of the first (oldest) sample in this message. • The time stamp is in the time reference scale that is used by the sensor time source. • Units - Milliseconds
<i>temperatureDataLen</i>	<ul style="list-style-type: none"> • Number of sets of the following elements <ul style="list-style-type: none"> – timeOffset – temperature
<i>timeOffset</i>	<ul style="list-style-type: none"> • Sample time offset • Units - Milliseconds
<i>temperature</i>	<ul style="list-style-type: none"> • Sensor temperature.
	<ul style="list-style-type: none"> • Type - Floating point • Units - Degrees Celsius • Range -50 to +100.00

8.801.2 Field Documentation

8.801.2.1 **ULONG** tempratureData::temperature[64]

8.801.2.2 **BYTE** tempratureData::temperatureDataLen

8.801.2.3 **ULONG** tempratureData::timeOfFirstSample

8.801.2.4 **WORD** tempratureData::timeOffset[64]

8.801.2.5 **ULONG** tempratureData::timeSource

8.802 TFTIDParams Struct Reference

Data Fields

- [BYTE](#) filterId
- [BYTE](#) eValid
- [BYTE](#) ipVersion
- [WORD](#) * pSourceIP
- [BYTE](#) sourceIPMask
- [BYTE](#) nextHeader
- [WORD](#) destPortRangeStart
- [WORD](#) destPortRangeEnd
- [WORD](#) srcPortRangeStart
- [WORD](#) srcPortRangeEnd
- [ULONG](#) IPSECSPi
- [WORD](#) tosMask
- [ULONG](#) flowLabel

8.802.1 Detailed Description

structure contains traffic flow template parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>filterId</i>	<ul style="list-style-type: none">• Filter identifier
<i>eValid</i>	<ul style="list-style-type: none">• Evaluation precedence index
<i>pVersion</i>	<ul style="list-style-type: none">• IP version number<ul style="list-style-type: none">– 4 - IPv4– 6 - IPv6

<i>sourceIP</i>	<ul style="list-style-type: none"> • Source IP address <ul style="list-style-type: none"> – IPv4 - Fill the first 4 bytes – IPv6 - Fill all the 16 bytes
<i>sourceIPMask</i>	<ul style="list-style-type: none"> • Mask value for the source address
<i>nextHeader</i>	<ul style="list-style-type: none"> • Next header/protocol value
<i>destPort↔ RangeStart</i>	<ul style="list-style-type: none"> • Start value of the destination port range
<i>destPort↔ RangeEnd</i>	<ul style="list-style-type: none"> • End value of the destination port range
<i>srcPortRange↔ Start</i>	<ul style="list-style-type: none"> • Start value of the source port range
<i>srcPortRange↔ End</i>	<ul style="list-style-type: none"> • End value of the source port range
<i>IPSECSPi</i>	<ul style="list-style-type: none"> • IPSEC security parameter index
<i>tosMask</i>	<ul style="list-style-type: none"> • TOS mask (Traffic class for IPv6)
<i>flowLabel</i>	<ul style="list-style-type: none"> • Flow label

8.802.2 Field Documentation

8.802.2.1 WORD TFTIDParams::destPortRangeEnd

8.802.2.2 WORD TFTIDParams::destPortRangeStart

8.802.2.3 BYTE TFTIDParams::eValid

8.802.2.4 BYTE TFTIDParams::filterId

8.802.2.5 ULONG TFTIDParams::flowLabel

8.802.2.6 ULONG TFTIDParams::IPSECSPi

8.802.2.7 BYTE TFTIDParams::ipVersion

8.802.2.8 **BYTE** TFTIDParams::nextHeader

8.802.2.9 **WORD*** TFTIDParams::pSourceIP

8.802.2.10 **BYTE** TFTIDParams::sourceIPMask

8.802.2.11 **WORD** TFTIDParams::srcPortRangeEnd

8.802.2.12 **WORD** TFTIDParams::srcPortRangeStart

8.802.2.13 **WORD** TFTIDParams::tosMask

8.803 timeInfo Struct Reference

Data Fields

- [WORD](#) year
- [BYTE](#) month
- [BYTE](#) day
- [BYTE](#) hour
- [BYTE](#) minute
- [BYTE](#) second
- [BYTE](#) dayOfWeek
- [INT8](#) timeZone
- [BYTE](#) dayLtSavingAdj
- [BYTE](#) radioInterface
- [BYTE](#) TlvPresent

8.803.1 Detailed Description

This structure contains the parameters for Network Time.

Parameters

<i>year</i>	<ul style="list-style-type: none">• Year
<i>month</i>	<ul style="list-style-type: none">• Month• 1 is January and 12 is December
<i>day</i>	<ul style="list-style-type: none">• Day• Range - 1 to 31
<i>hour</i>	<ul style="list-style-type: none">• Hour• Range - 0 to 59

<i>minute</i>	<ul style="list-style-type: none"> • Minute • Range - 0 to 59
<i>second</i>	<ul style="list-style-type: none"> • Second • Range - 0 to 59
<i>dayOfWeek</i>	<ul style="list-style-type: none"> • Day of the week • 0 is Monday and 6 is Sunday
<i>timeZone</i>	<ul style="list-style-type: none"> • Offset from Universal time • The difference between local time and Universal time, in increments of 15 min • Signed Value
<i>dayLtSavingAdj</i>	<ul style="list-style-type: none"> • Daylight saving adjustment in hours • Possible values - 0, 1, and 2. • This field is ignored if radio_if is NAS_RADIO_IF_CDMA_1XEVDO
<i>radioInterface</i>	<ul style="list-style-type: none"> • Radio interface from which the information comes • Values <ul style="list-style-type: none"> – 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - NAS_RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA - TD-SCDMA
<i>TlvPresent</i>	<ul style="list-style-type: none"> • Tlv Present.

8.803.2 Field Documentation

8.803.2.1 BYTE timeInfo::day

8.803.2.2 BYTE timeInfo::dayLtSavingAdj

8.803.2.3 BYTE timeInfo::dayOfWeek

8.803.2.4 BYTE timeInfo::hour

8.803.2.5 BYTE timeInfo::minute

8.803.2.6 BYTE timeInfo::month

8.803.2.7 **BYTE** timeInfo::radioInterface

8.803.2.8 **BYTE** timeInfo::second

8.803.2.9 **INT8** timeInfo::timeZone

8.803.2.10 **BYTE** timeInfo::TlvPresent

8.803.2.11 **WORD** timeInfo::year

8.804 TmdDeRegNotMitigationLvlReq Struct Reference

Data Fields

- [BYTE](#) mitigationDevIDLen
- [CHAR](#) mitigationDevID [255]

8.804.1 Detailed Description

This structure contains mitigation devices Level deregister request parameters

Parameters

<i>mitigationDevIDLen</i>	<ul style="list-style-type: none">• Number of sets of the following elements<ul style="list-style-type: none">– mitigation_dev_id
<i>mitigationDevID</i>	<ul style="list-style-type: none">• Mitigation device ID

8.804.2 Field Documentation

8.804.2.1 **CHAR** TmdDeRegNotMitigationLvlReq::mitigationDevID[255]

8.804.2.2 **BYTE** TmdDeRegNotMitigationLvlReq::mitigationDevIDLen

8.805 TmdGetMitigationDevListResp Struct Reference

Data Fields

- **BYTE** * pMitigationDevListLen
- mitigationDevList * pMitigationDevList

8.805.1 Detailed Description

This structure contains mitigation devices list from the remote endpoint

Parameters

<i>pMitigationDev</i> ↔ <i>ListLen</i>	<ul style="list-style-type: none"> Mitigation Device List Length (Optional) Number of sets of the following elements <i>pMitigationDevList</i>
<i>pMitigationDev</i> ↔ <i>List</i>	<ul style="list-style-type: none"> Mitigation Device List (Optional) See mitigationDevList for more information.

8.805.2 Field Documentation

8.805.2.1 `mitigationDevList*` `TmdGetMitigationDevListResp::pMitigationDevList`

8.805.2.2 `BYTE*` `TmdGetMitigationDevListResp::pMitigationDevListLen`

8.806 TmdGetMitigationLvlReq Struct Reference

Data Fields

- `BYTE` [mitigationDevIDLen](#)
- `CHAR` [mitigationDevID](#) [255]

8.806.1 Detailed Description

This structure contains mitigation devices Level request parameters

Parameters

<i>mitigationDev</i> ↔ <i>IDLen</i>	<ul style="list-style-type: none"> Number of sets of the following elements <ul style="list-style-type: none"> <code>mitigation_dev_id</code>
<i>mitigationDevID</i>	<ul style="list-style-type: none"> Mitigation device ID

8.806.2 Field Documentation

8.806.2.1 `CHAR` `TmdGetMitigationLvlReq::mitigationDevID[255]`

8.806.2.2 **BYTE** TmdGetMitigationLvlReq::mitigationDevIDLen

8.807 TmdGetMitigationLvlResp Struct Reference

Data Fields

- **BYTE** * pCurrentmitigationLvl
- **BYTE** * pReqMitigationLvl

8.807.1 Detailed Description

This structure contains mitigation devices Level request parameters

Parameters

<i>pCurrentmitigationLvl</i>	<ul style="list-style-type: none"> • Current thermal mitigation level (Optional)
<i>pReqMitigationLvl</i>	<ul style="list-style-type: none"> • Requested Thermal Mitigation Level (Optional) • The requested thermal mitigation level from the client. The default is zero if the client has not previously set the mitigation level.

8.807.2 Field Documentation

8.807.2.1 **BYTE*** TmdGetMitigationLvlResp::pCurrentmitigationLvl

8.807.2.2 **BYTE*** TmdGetMitigationLvlResp::pReqMitigationLvl

8.808 TmdMitigationLvlIndReq Struct Reference

Data Fields

- **BYTE** mitigationDevIDLen
- **CHAR** mitigationDevID [255]

8.808.1 Detailed Description

This structure contains mitigation Level Indication request parameters

Parameters

<i>mitigationDevIDLen</i>	<ul style="list-style-type: none"> • Number of sets of the following elements <ul style="list-style-type: none"> – mitigation_dev_id
<i>mitigationDevID</i>	<ul style="list-style-type: none"> • Mitigation device ID

8.808.2 Field Documentation

8.808.2.1 CHAR TmdMitigationLvlIndReq::mitigationDevID[255]

8.808.2.2 BYTE TmdMitigationLvlIndReq::mitigationDevIDLen

8.809 TmdRegNotMitigationLvlReq Struct Reference

Data Fields

- [BYTE mitigationDevIDLen](#)
- [CHAR mitigationDevID](#) [255]

8.809.1 Detailed Description

This structure contains mitigation devices Level register request parameters

Parameters

<i>mitigationDevIDLen</i>	<ul style="list-style-type: none"> • Number of sets of the following elements <ul style="list-style-type: none"> – mitigation_dev_id
<i>mitigationDevID</i>	<ul style="list-style-type: none"> • Mitigation device ID

8.809.2 Field Documentation

8.809.2.1 CHAR TmdRegNotMitigationLvlReq::mitigationDevID[255]

8.809.2.2 BYTE TmdRegNotMitigationLvlReq::mitigationDevIDLen

8.810 tokenBucket Struct Reference

Data Fields

- [ULONG peakRate](#)
- [ULONG tokenRate](#)
- [ULONG bucketSz](#)

8.810.1 Detailed Description

This structure contains the TP flow data rate token bucket

Parameters

<i>peakRate</i>	Maximum rate at which data can be transmitted when the token bucket is full (bits per second)
<i>tokenRate</i>	Rate at which tokens will be put in the token bucket (bits per second); a token is required to be present in the bucket to send a byte of data
<i>bucketSz</i>	Maximum number of tokens that can be accumulated at any instance (bytes); controls the size of the burst that is allowed at any given time

8.810.2 Field Documentation

8.810.2.1 **ULONG** tokenBucket::bucketSz

8.810.2.2 **ULONG** tokenBucket::peakRate

8.810.2.3 **ULONG** tokenBucket::tokenRate

8.811 Tos Struct Reference

Data Fields

- [BYTE](#) val
- [BYTE](#) mask

8.811.1 Detailed Description

This structure contains the IPv4 filter type of service

Parameters

<i>val</i>	Type of service value
<i>mask</i>	Packet matches the TOS filter if: (IPv4_filter_tos_val and IPv4_filter_tos_mask) == (TOS value in the IP packet & IPv4_filter_tos_mask) Example: <ul style="list-style-type: none"> • IPv4_filter_tos_val = 00101000 • IPv4_filter_tos_mask = 11111100 The filter will compare only the first 6 bits in the IPv4_filter↵_type_of_service with the first 6 bits in the TOS field of the IP packet. The first 6 bits in the TOS field of the IP packet must be 001010 to match the filter. The last 2 bits can be anything since they are ignored by filtering.

8.811.2 Field Documentation

8.811.2.1 **BYTE** Tos::mask

8.811.2.2 **BYTE** Tos::val

8.812 transferRouteMessageTlv Struct Reference

Data Fields

- [uint8_t TlvPresent](#)
- [sMSTransferRouteMTMessageInfo TransferRouteMTMessageInfo](#)

8.812.1 Detailed Description

Parameters

<i>TlvPresent</i>	<ul style="list-style-type: none"> • Boolean indicating the presence of the TLV in the QMI response
<i>TransferRoute↔MTMessageInfo</i>	<ul style="list-style-type: none"> • Transfer Route MT Message • See sMSTransferRouteMTMessageInfo for more information

8.812.2 Field Documentation

8.812.2.1 `uint8_t transferRouteMessageTlv::TlvPresent`

8.812.2.2 `sMSTransferRouteMTMessageInfo transferRouteMessageTlv::TransferRouteMTMessageInfo`

8.813 TransferStatInd Struct Reference

Data Fields

- [BYTE StatsPeriod](#)
- [ULONG StatsMask](#)

8.813.1 Detailed Description

This structure contains Transfer Statistics Indicator

Parameters

<i>StatsPeriod</i>	<ul style="list-style-type: none"> • Period between transfer statistics reports <ul style="list-style-type: none"> – 0 - Do not report – Other - Period between reports (seconds)
<i>StatsMask</i>	<ul style="list-style-type: none"> • Requested statistic bit mask. Each bit set causes the corresponding optional TLV to be sent in the indication. All unlisted bits are reserved for future use and must be set to zero. <ul style="list-style-type: none"> – 0x00000040 - Tx bytes OK – 0x00000080 - Rx bytes OK

8.813.2 Field Documentation

8.813.2.1 **ULONG** TransferStatInd::StatsMask

8.813.2.2 **BYTE** TransferStatInd::StatsPeriod

8.814 transferStatInd Struct Reference

Data Fields

- `uint8_t` [StatsPeriod](#)
- `uint32_t` [StatsMask](#)

8.814.1 Detailed Description

Parameters

<i>StatsPeriod</i>	Field Period between transfer statistic reports.
<i>StatsMask</i>	requested statistic bit mask.

8.814.2 Field Documentation

8.814.2.1 `uint32_t` transferStatInd::StatsMask

8.814.2.2 `uint8_t` transferStatInd::StatsPeriod

8.815 TransferStatsDataType Struct Reference

Data Fields

- [BYTE](#) `interval`

8.815.1 Field Documentation

8.815.1.1 **BYTE** TransferStatsDataType::interval

8.816 TrStatInd Struct Reference

Data Fields

- [BYTE](#) `statsPeriod`
- [ULONG](#) `statsMask`

8.816.1 Detailed Description

This structure contains the information about the Transfer Statistics Indicator parameters.

Parameters

<i>statsPeriod</i>	<ul style="list-style-type: none"> Period between transfer statistics reports. <ul style="list-style-type: none"> 0 - Do not report Other - Period between reports (seconds)
<i>statsMask</i>	<ul style="list-style-type: none"> Requested statistic bit mask. <ul style="list-style-type: none"> 0x00000001 - Tx packets OK 0x00000002 - Rx packets OK 0x00000004 - Tx packet errors 0x00000008 - Rx packet errors 0x00000010 - Tx overflows 0x00000020 - Rx overflows 0x00000040 - Tx bytes OK 0x00000080 - Rx bytes OK Each bit set causes the corresponding optional information to be sent in SLQSWdsEvent↔ ReportCallBack. All unlisted bits are reserved for future use and must be set to zero.

8.816.2 Field Documentation

8.816.2.1 ULONG TrStatInd::statsMask

8.816.2.2 BYTE TrStatInd::statsPeriod

8.817 trueIMSI Struct Reference

Data Fields

- [BYTE mccT](#) [3]
- [WORD imsiT1112](#)
- [BYTE imsiTS1](#) [7]
- [BYTE imsiTS2](#) [3]
- [BYTE imsiTaddrNum](#)

8.817.1 Detailed Description

This structure contains the parameters for True IMSI Information

Parameters

<i>mccT</i>	<ul style="list-style-type: none"> • ASCII character representation of MCC_T
<i>imsiT1112</i>	<ul style="list-style-type: none"> • ASCII character representation of IMSI_T_11_12 value <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>imsiTS1</i>	<ul style="list-style-type: none"> • ASCII character representation of IMSI_T_S1 value
<i>imsiTS2</i>	<ul style="list-style-type: none"> • ASCII character representation of IMSI_T_S2 value
<i>imsiTaddrNum</i>	<ul style="list-style-type: none"> • Value of IMSI_T_ADDR_NUM <ul style="list-style-type: none"> – 0xFF - Not Available

8.817.2 Field Documentation

8.817.2.1 WORD trueIMSI::imsiT1112

8.817.2.2 BYTE trueIMSI::imsiTaddrNum

8.817.2.3 BYTE trueIMSI::imsiTS1[7]

8.817.2.4 BYTE trueIMSI::imsiTS2[3]

8.817.2.5 BYTE trueIMSI::mccT[3]

8.818 TXAGCList Struct Reference

Data Fields

- WORD * pTXStaticGain
- WORD * pTXAIG
- WORD * pTXExpThres
- WORD * pTXExpSlope
- WORD * pTXComprThres
- WORD * pTXComprSlope

8.818.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig parameters related to AV_TXAGCLIST.

Parameters

<i>pTXStaticGain</i>	<ul style="list-style-type: none"> • TX pre-compressor static gain
<i>pTXAIG</i>	<ul style="list-style-type: none"> • TX pre-compressor gain selection flag
<i>pTXExpThres</i>	<ul style="list-style-type: none"> • TX expansion threshold
<i>pTXExpSlope</i>	<ul style="list-style-type: none"> • TX expansion slope
<i>pTXComprThres</i>	<ul style="list-style-type: none"> • TX compression threshold
<i>pTXComprSlope</i>	<ul style="list-style-type: none"> • TX compression slope

8.818.2 Field Documentation

8.818.2.1 WORD* TXAGCList::pTXAIG

8.818.2.2 WORD* TXAGCList::pTXComprSlope

8.818.2.3 WORD* TXAGCList::pTXComprThres

8.818.2.4 WORD* TXAGCList::pTXExpSlope

8.818.2.5 WORD* TXAGCList::pTXExpThres

8.818.2.6 WORD* TXAGCList::pTXStaticGain

8.819 txInfo Struct Reference

Data Fields

- [BYTE isInTraffic](#)
- [INT32 txPower](#)

8.819.1 Detailed Description

This structure contains the Tx Information.

Parameters

<i>isInTraffic</i>	<ul style="list-style-type: none"> • Whether the device is in traffic. <ul style="list-style-type: none"> – 0x00 - not in traffic – 0x01 - in traffic • The txPower field is only meaningful when in the device is in traffic.
--------------------	---

8.819.2 Field Documentation

8.819.2.1 BYTE txInfo::isInTraffic

8.819.2.2 INT32 txInfo::txPower

8.820 TXPCMIIRFiltr Struct Reference

Data Fields

- WORD * pFlag
- WORD * pStageCnt
- BYTE * pStage0Val
- BYTE * pStage1Val
- BYTE * pStage2Val
- BYTE * pStage3Val
- BYTE * pStage4Val

8.820.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig parameters related to AV_TXPCMIIRFLTR.

Parameters

<i>pFlag</i>	<ul style="list-style-type: none">• Flag<ul style="list-style-type: none">– 0x0000 - IIR filter disable– 0xffff - IIR filter enable
<i>pStageCnt</i>	<ul style="list-style-type: none">• Stage Count<ul style="list-style-type: none">– 0-4
<i>pStage0Val</i>	<ul style="list-style-type: none">• A 20 BYTE sized parameter indicating Stage 0 value<ul style="list-style-type: none">– A1– A2– B0– B1– B2
<i>pStage1Val</i>	<ul style="list-style-type: none">• A 20 BYTE sized parameter indicating Stage 1 value<ul style="list-style-type: none">– A1– A2– B0– B1– B2

<i>pStage2Val</i>	<ul style="list-style-type: none"> • A 20 BYTE sized parameter indicating Stage 2 value <ul style="list-style-type: none"> – A1 – A2 – B0 – B1 – B2
<i>pStage3Val</i>	<ul style="list-style-type: none"> • A 20 BYTE sized parameter indicating Stage 3 value <ul style="list-style-type: none"> – A1 – A2 – B0 – B1 – B2
<i>pStage4Val</i>	<ul style="list-style-type: none"> • A 20 BYTE sized parameter indicating Stage 4 value <ul style="list-style-type: none"> – A1 – A2 – B0 – B1 – B2

8.820.2 Field Documentation

8.820.2.1 WORD* TXPCMIIRFitr::pFlag

8.820.2.2 BYTE* TXPCMIIRFitr::pStage0Val

8.820.2.3 BYTE* TXPCMIIRFitr::pStage1Val

8.820.2.4 BYTE* TXPCMIIRFitr::pStage2Val

8.820.2.5 BYTE* TXPCMIIRFitr::pStage3Val

8.820.2.6 BYTE* TXPCMIIRFitr::pStage4Val

8.820.2.7 WORD* TXPCMIIRFitr::pStageCnt

8.821 uim_appStatus Struct Reference

Data Fields

- uint8_t [appType](#)
- uint8_t [appState](#)

- uint8_t [persoState](#)
- uint8_t [persoFeature](#)
- uint8_t [persoRetries](#)
- uint8_t [persoUnblockRetries](#)
- uint8_t [aidLength](#)
- uint8_t [aidVal](#) [255]
- uint8_t [univPin](#)
- uint8_t [pin1State](#)
- uint8_t [pin1Retries](#)
- uint8_t [puk1Retries](#)
- uint8_t [pin2State](#)
- uint8_t [pin2Retries](#)
- uint8_t [puk2Retries](#)

8.821.1 Detailed Description

This structure contains Application Status Information loaded on the card.

Parameters

<i>appType</i>	<ul style="list-style-type: none"> • Indicates the type of the application. <ul style="list-style-type: none"> – 0 - Unknown – 1 - SIM card – 2 - USIM application – 3 - RUIM card – 4 - CSIM application – 5 - ISIM application • Other values are reserved for the future and are to be handled as "Unknown".
<i>appState</i>	<ul style="list-style-type: none"> • Indicates the state of the application. <ul style="list-style-type: none"> – 0 - Unknown – 1 - Detected – 2 - PIN1 or UPIN is required – 3 - PUK1 or PUK for UPIN is required – 4 - Personalization state must be checked – 5 - PIN1 is blocked – 6 - Illegal – 7 - Ready
<i>persoState</i>	<ul style="list-style-type: none"> • Indicates the state of the personalization for the application. <ul style="list-style-type: none"> – 0 - Unknown – 1 - Personalization operation is in progress – 2 - Ready – 3 - Personalization code is required – 4 - PUK for personalization code is required – 5 - Permanently blocked

<i>persoFeature</i>	<ul style="list-style-type: none"> • Indicates the personalization feature. • This applies only when a personalization code is required to deactivate or unblock personalization. <ul style="list-style-type: none"> – 0 - GW network personalization – 1 - GW network subset personalization – 2 - GW service provider personalization – 3 - GW corporate personalization – 4 - GW UIM personalization – 5 - 1X network type 1 personalization – 6 - 1X network type 2 personalization – 7 - 1X HRPD personalization – 8 - 1X service provider personalization – 9 - 1X corporate personalization – 10 - 1X RUIM personalization – 11 - Unknown
<i>persoRetries</i>	<ul style="list-style-type: none"> • Indicates the number of retries remaining to disable the personalization.
<i>persoUnblockRetries</i>	<ul style="list-style-type: none"> • Indicates the number of retries remaining to unblock the personalization.
<i>aidLength</i>	<ul style="list-style-type: none"> • Number of sets of the following elements. i.e. aidVal • If zero(0) then no aidVal information exists.
<i>aidVal[<small>MAX_DESCRIPTOR_LENGTH</small>]</i>	<ul style="list-style-type: none"> • Application identifier value.
<i>univPin</i>	<ul style="list-style-type: none"> • Indicates whether UPIN replaces PIN1. <ul style="list-style-type: none"> – 0 - PIN1 is used – 1 - UPIN replaces PIN1
<i>pin1State</i>	<ul style="list-style-type: none"> • Indicates the state of PIN1. <ul style="list-style-type: none"> – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked
<i>pin1Retries</i>	<ul style="list-style-type: none"> • Indicates the number of retries remaining to verify PIN1.
<i>puk1Retries</i>	<ul style="list-style-type: none"> • Indicates the number of retries remaining to unblock PIN1.

<i>pin2State</i>	<ul style="list-style-type: none"> Indicates the state of PIN2. <ul style="list-style-type: none"> 0 - Unknown 1 - Enabled and not verified 2 - Enabled and verified 3 - Disabled 4 - Blocked 5 - Permanently blocked
<i>pin2Retries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to verify PIN2.
<i>puk2Retries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to unblock PIN2.

8.821.2 Field Documentation

8.821.2.1 `uint8_t uim_appStatus::aidLength`

8.821.2.2 `uint8_t uim_appStatus::aidVal[255]`

8.821.2.3 `uint8_t uim_appStatus::appState`

8.821.2.4 `uint8_t uim_appStatus::appType`

8.821.2.5 `uint8_t uim_appStatus::persoFeature`

8.821.2.6 `uint8_t uim_appStatus::persoRetries`

8.821.2.7 `uint8_t uim_appStatus::persoState`

8.821.2.8 `uint8_t uim_appStatus::persoUnblockRetries`

8.821.2.9 `uint8_t uim_appStatus::pin1Retries`

8.821.2.10 `uint8_t uim_appStatus::pin1State`

8.821.2.11 `uint8_t uim_appStatus::pin2Retries`

8.821.2.12 `uint8_t uim_appStatus::pin2State`

8.821.2.13 `uint8_t uim_appStatus::puk1Retries`

8.821.2.14 `uint8_t uim_appStatus::puk2Retries`

8.821.2.15 `uint8_t uim_appStatus::univPin`

8.822 uim_cardResult Struct Reference

Data Fields

- `uint8_t` [sw1](#)
- `uint8_t` [sw2](#)

8.822.1 Detailed Description

This structure contains the information about the card result.

Parameters

<i>sw1</i>	<ul style="list-style-type: none"> • SW1 received from the card.
<i>sw2</i>	<ul style="list-style-type: none"> • SW2 received from the card.

8.822.2 Field Documentation

8.822.2.1 `uint8_t uim_cardResult::sw1`

8.822.2.2 `uint8_t uim_cardResult::sw2`

8.823 uim_cardStatus Struct Reference

Data Fields

- `uint16_t indexGwPri`
- `uint16_t index1xPri`
- `uint16_t indexGwSec`
- `uint16_t index1xSec`
- `uint8_t numSlot`
- `uim_slotInfo SlotInfo` [5]

8.823.1 Detailed Description

This structure contains Card Status Information.

Parameters

<i>indexGwPri</i>	<ul style="list-style-type: none"> • Index of the primary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
<i>index1xPri</i>	<ul style="list-style-type: none"> • Index of the primary 1X provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.

<i>indexGwSec</i>	<ul style="list-style-type: none"> • Index of the secondary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
<i>index1xSec</i>	<ul style="list-style-type: none"> • Index of the secondary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
<i>numSlot</i>	<ul style="list-style-type: none"> • Indicates the number of slots available on the device. • The following block is repeated for each slot. i.e. cardState • If zero(0) then no cardState information exists.
<i>SlotInfo</i>	<ul style="list-style-type: none"> • See uim_slotInfo for more information.

8.823.2 Field Documentation

8.823.2.1 `uint16_t uim_cardStatus::index1xPri`

8.823.2.2 `uint16_t uim_cardStatus::index1xSec`

8.823.2.3 `uint16_t uim_cardStatus::indexGwPri`

8.823.2.4 `uint16_t uim_cardStatus::indexGwSec`

8.823.2.5 `uint8_t uim_cardStatus::numSlot`

8.823.2.6 `uim_slotInfo uim_cardStatus::SlotInfo[5]`

8.824 uim_changeUIMPIN Struct Reference

Data Fields

- `uint8_t pinID`
- `uint8_t oldPINLen`
- `uint8_t oldPINVal [255]`
- `uint8_t pinLen`
- `uint8_t pinVal [255]`

8.824.1 Detailed Description

This structure contains the information about the pin parameters that need to be verified.

Parameters

<i>pinID</i>	<ul style="list-style-type: none"> Indicates the PIN ID to be changed. <ul style="list-style-type: none"> 1 - PIN1 (also called PIN) 2 - PIN2 3 - Universal PIN 4 - Hidden key
<i>oldPINLen</i>	<ul style="list-style-type: none"> Length of the following elements i.e. old pin value.
<i>oldPINVal</i> [<i>MAX_DESCRIPTOR_LENGTH</i>]	<ul style="list-style-type: none"> Old PIN value. This value is a sequence of ASCII characters.
<i>pinLen</i>	<ul style="list-style-type: none"> Length of the following elements i.e. new pin value.
<i>pinVal</i> [<i>MAX_DESCRIPTOR_LENGTH</i>]	<ul style="list-style-type: none"> New PIN value. This value is a sequence of ASCII characters.

8.824.2 Field Documentation

8.824.2.1 `uint8_t uim_changeUIMPIN::oldPINLen`

8.824.2.2 `uint8_t uim_changeUIMPIN::oldPINVal[255]`

8.824.2.3 `uint8_t uim_changeUIMPIN::pinID`

8.824.2.4 `uint8_t uim_changeUIMPIN::pinLen`

8.824.2.5 `uint8_t uim_changeUIMPIN::pinVal[255]`

8.825 uim_encryptedPIN1 Struct Reference

Data Fields

- `uint8_t pin1Len`
- `uint8_t pin1Val [255]`

8.825.1 Detailed Description

This structure contains the encrypted PIN1 Information.

Parameters

<i>pin1Len</i>	<ul style="list-style-type: none">• Number of sets of the following elements ie encrypted PIN1 value.• If zero(0), no information follows.
<i>pin1Val</i>	<ul style="list-style-type: none">• Encrypted PIN1 value.

Note

This value is returned only when PIN1 is enabled successfully and the feature is supported.

8.825.2 Field Documentation

8.825.2.1 `uint8_t uim_encryptedPIN1::pin1Len`

8.825.2.2 `uint8_t uim_encryptedPIN1::pin1Val[255]`

8.826 uim_fileInfo Struct Reference**Data Fields**

- `uint16_t fileID`
- `uint8_t pathLen`
- `uint16_t path [255]`

8.826.1 Detailed Description

This structure contains paramaters for file Information

Parameters

<i>fileID</i>	<ul style="list-style-type: none">• This is Identifier to SIM files; e.g. in UIM "6F07" is Identifier of IMSI File
<i>pathLen</i>	<ul style="list-style-type: none">• Length of file Path
<i>path</i>	<ul style="list-style-type: none">• Path value. This value must be the complete path of the file, which is a sequence block of 2 bytes (e.g., 0x3F00 0x7FFF).

8.826.2 Field Documentation

8.826.2.1 `uint16_t uim_fileInfo::fileID`

8.826.2.2 `uint16_t uim_fileInfo::path[255]`

8.826.2.3 `uint8_t uim_fileInfo::pathLen`

8.827 `uim_hotSwapStatus` Struct Reference

Data Fields

- `uint8_t hotSwapLength`
- `uint8_t hotSwap [255]`

8.827.1 Detailed Description

This structure contains Hot Swap Status Information.

Parameters

<i>hotSwapLength</i>	<ul style="list-style-type: none"> • Number of sets of the following elements. i.e. <code>hot_swap</code>
<i>hotSwap</i>	<ul style="list-style-type: none"> • Indicates the status of the hot-swap switch. <ul style="list-style-type: none"> – 0 - Hot-swap is not supported – 1 - Hot-swap is supported, but the status of the switch is not supported – 2 - Switch indicates that the card is present – 3 - Switch indicates that the card is not present

8.827.2 Field Documentation

8.827.2.1 `uint8_t uim_hotSwapStatus::hotSwap[255]`

8.827.2.2 `uint8_t uim_hotSwapStatus::hotSwapLength`

8.828 `uim_readResult` Struct Reference

Data Fields

- `uint16_t contentLen`
- `uint8_t content [255]`

8.828.1 Detailed Description

This structure contains the information for write operation.

Parameters

<i>contentLen</i>	<ul style="list-style-type: none"> • Number of sets of content.
<i>content[255]</i>	<ul style="list-style-type: none"> • Read content. • The content is the sequence of bytes as read from the card.

8.828.2 Field Documentation

8.828.2.1 `uint8_t uim_readResult::content[255]`

8.828.2.2 `uint16_t uim_readResult::contentLen`

8.829 uim_readTransparentInfo Struct Reference

Data Fields

- `uint16_t offset`
- `uint16_t length`

8.829.1 Detailed Description

This structure contains the information for read operation.

Parameters

<i>offset</i>	<ul style="list-style-type: none"> • Offset for the read operation.
<i>length</i>	<ul style="list-style-type: none"> • Length of the content to be read. • The value 0 is used to read the complete file.

8.829.2 Field Documentation

8.829.2.1 `uint16_t uim_readTransparentInfo::length`

8.829.2.2 `uint16_t uim_readTransparentInfo::offset`

8.830 uim_remainingRetries Struct Reference

Data Fields

- `uint8_t verifyLeft`
- `uint8_t unblockLeft`

8.830.1 Detailed Description

This structure contains the information about the retries remaining.

Parameters

<i>verifyLeft</i>	<ul style="list-style-type: none"> • Number of remaining attempts to verify the PIN. • 0xFF, if unavailable.
<i>unlockLeft</i>	<ul style="list-style-type: none"> • Number of remaining attempts to unlock the PIN. • 0xFF, if unavailable.

Note

This value is returned only when the enable/disable operation has failed. This information is not sent for a hidden key PIN type.

8.830.2 Field Documentation

8.830.2.1 `uint8_t uim_remainingRetries::unlockLeft`

8.830.2.2 `uint8_t uim_remainingRetries::verifyLeft`

8.831 uim_sessionInformation Struct Reference

Data Fields

- `uint8_t sessionType`
- `uint8_t aidLength`
- `uint8_t aid [255]`

8.831.1 Detailed Description

This structure contains the Session Information.

Parameters

<i>sessionType</i>	<ul style="list-style-type: none"> • Indicates the session type. <ul style="list-style-type: none"> – 0 - Primary GW provisioning – 1 - Primary 1X provisioning – 2 - Secondary GW provisioning – 3 - Secondary 1X provisioning – 4 - Non-provisioning on slot 1 – 5 - Non-provisioning on slot 2 – 6 - Card on slot 1 – 7 - Card on slot 2 – 8 - Logical channel on slot 1 – 9 - Logical channel on slot 2
	Generated by Doxygen

<i>aidLength</i>	<ul style="list-style-type: none"> Length of the following elements i.e. Application Identifier.
<i>aid</i>	<ul style="list-style-type: none"> Application identifier value or channel ID. This value is required for non-provisioning and for logical channel session types. It is ignored in all other cases.

8.831.2 Field Documentation

8.831.2.1 `uint8_t uim_sessionInformation::aid[255]`

8.831.2.2 `uint8_t uim_sessionInformation::aidLength`

8.831.2.3 `uint8_t uim_sessionInformation::sessionType`

8.832 uim_setPINProtection Struct Reference

Data Fields

- `uint8_t pinID`
- `uint8_t pinOperation`
- `uint8_t pinLength`
- `uint8_t pinValue [255]`

8.832.1 Detailed Description

This structure contains the information about the pin protection parameters that need to be set.

Parameters

<i>pinID</i>	<ul style="list-style-type: none"> Indicates the PIN ID to be enabled or disabled. <ul style="list-style-type: none"> 1 - PIN1 (also called PIN) 2 - PIN2 3 - Universal PIN 4 - Hidden key
<i>pinOperation</i>	<ul style="list-style-type: none"> Indicates whether the PIN is enabled or disabled. <ul style="list-style-type: none"> 0 - Disable the PIN 1 - Enable the PIN
<i>pinLength</i>	<ul style="list-style-type: none"> Length of the following elements i.e. pin value.
<i>pinValue</i> [MAX_PIN_LENGTH]	<ul style="list-style-type: none"> PIN value. This value is a sequence of ASCII characters.

8.832.2 Field Documentation

8.832.2.1 `uint8_t uim_setPINProtection::pinID`

8.832.2.2 `uint8_t uim_setPINProtection::pinLength`

8.832.2.3 `uint8_t uim_setPINProtection::pinOperation`

8.832.2.4 `uint8_t uim_setPINProtection::pinValue[255]`

8.833 `uim_slotInfo` Struct Reference

Data Fields

- `uint8_t cardState`
- `uint8_t upinState`
- `uint8_t upinRetries`
- `uint8_t upukRetries`
- `uint8_t errorState`
- `uint8_t numApp`
- `uim_appStatus AppStatus` [10]

8.833.1 Detailed Description

This structure contains information about the SLOTS present.

Parameters

<i>cardState</i>	<ul style="list-style-type: none"> • Indicates the state of the card for each slot. <ul style="list-style-type: none"> – 0 - Absent – 1 - Present – 2 - Error
<i>upinState</i>	<ul style="list-style-type: none"> • Indicates the state of UPIN. <ul style="list-style-type: none"> – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked – 0xFF - Not Available
<i>upinRetries</i>	<ul style="list-style-type: none"> • Indicates the number of retries remaining to verify the UPIN. • If 0xFF, information not available.

<i>upukRetries</i>	<ul style="list-style-type: none"> Indicates the number of retries remaining to unblock the UPIN. If 0xFF, information not available.
<i>errorState</i>	<ul style="list-style-type: none"> Indicates the reason for the card error, and is valid only when the card state is Error <ul style="list-style-type: none"> 0 - Unknown 1 - Power down 2 - Poll error 3 - No ATR received 4 - Volt mismatch 5 - Parity error 6 - Unknown; possibly removed 7 - Card returned technical problems 0xFF - Not Available Other values are possible and reserved for future use. When an unknown value is received, it is to be handled as "Unknown".
<i>numApp</i>	<ul style="list-style-type: none"> Indicates the number of applications available on the card. The following block is repeated for each application. i.e. AppStatus. If zero(0) then no AppStatus information exists.
<i>AppStatus</i>	<ul style="list-style-type: none"> See uim_appStatus for more information.

8.833.2 Field Documentation

8.833.2.1 `uim_appStatus uim_slotInfo::AppStatus[10]`

8.833.2.2 `uint8_t uim_slotInfo::cardState`

8.833.2.3 `uint8_t uim_slotInfo::errorState`

8.833.2.4 `uint8_t uim_slotInfo::numApp`

8.833.2.5 `uint8_t uim_slotInfo::upinRetries`

8.833.2.6 `uint8_t uim_slotInfo::upinState`

8.833.2.7 `uint8_t uim_slotInfo::upukRetries`

8.834 uim_UIMSessionInformation Struct Reference

Data Fields

- `uint8_t sessionType`
- `uint8_t aidLength`
- `uint8_t aid [255]`

8.834.1 Detailed Description

This structure contains the Session Information.

Parameters

<i>sessionType</i>	<ul style="list-style-type: none"> Indicates the session type. <ul style="list-style-type: none"> 0 - Primary GW provisioning 1 - Primary 1X provisioning 2 - Secondary GW provisioning 3 - Secondary 1X provisioning 4 - Non-provisioning on slot 1 5 - Non-provisioning on slot 2 6 - Card on slot 1 7 - Card on slot 2 8 - Logical channel on slot 1 9 - Logical channel on slot 2
<i>aidLength</i>	<ul style="list-style-type: none"> Length of the following elements i.e. Application Identifier.
<i>aid</i>	<ul style="list-style-type: none"> Application identifier value or channel ID. This value is required for non-provisioning and for logical channel session types. It is ignored in all other cases.

8.834.2 Field Documentation

8.834.2.1 `uint8_t uim_UIMSessionInformation::aid[255]`

8.834.2.2 `uint8_t uim_UIMSessionInformation::aidLength`

8.834.2.3 `uint8_t uim_UIMSessionInformation::sessionType`

8.835 uim_unblockUIMPIN Struct Reference

Data Fields

- `uint8_t pinID`
- `uint8_t pukLen`
- `uint8_t pukVal [255]`
- `uint8_t newPINLen`
- `uint8_t newPINVal [255]`

8.835.1 Detailed Description

This structure contains the information about the unblock pin parameters.

Parameters

<i>pinID</i>	<ul style="list-style-type: none"> Indicates the PIN ID to be changed. <ul style="list-style-type: none"> 1 - PIN1 (also called PIN) 2 - PIN2 3 - Universal PIN
<i>pukLen</i>	<ul style="list-style-type: none"> Length of the following elements i.e. puk value.
<i>pukVal</i> [UIM_MAX_DESCRIPTION_LENGTH]	<ul style="list-style-type: none"> PIN Unlock Key value. This value is a sequence of ASCII characters.
<i>newPINLen</i>	<ul style="list-style-type: none"> Length of the following elements i.e. new pin value.
<i>newPINVal</i> [UIM_MAX_DESCRIPTION_LENGTH]	<ul style="list-style-type: none"> New PIN value. This value is a sequence of ASCII characters.

8.835.2 Field Documentation

8.835.2.1 `uint8_t uim_unblockUIMPIN::newPINLen`

8.835.2.2 `uint8_t uim_unblockUIMPIN::newPINVal[255]`

8.835.2.3 `uint8_t uim_unblockUIMPIN::pinID`

8.835.2.4 `uint8_t uim_unblockUIMPIN::pukLen`

8.835.2.5 `uint8_t uim_unblockUIMPIN::pukVal[255]`

8.836 uim_verifyUIMPIN Struct Reference

Data Fields

- `uint8_t pinID`
- `uint8_t pinLen`
- `uint8_t pinVal [255]`

8.836.1 Detailed Description

This structure contains the information about the pin parameters that need to be verified.

Parameters

<i>pinID</i>	<ul style="list-style-type: none"> Indicates the PIN ID to be verified. <ul style="list-style-type: none"> 1 - PIN1 (also called PIN) 2 - PIN2 3 - Universal PIN 4 - Hidden key
<i>pinLen</i>	<ul style="list-style-type: none"> Length of the following elements i.e. pin value.
<i>pinVal</i> [<i>MAX_DESCRIPTOR_LENGTH</i>]	<ul style="list-style-type: none"> PIN value. This value is a sequence of ASCII characters.

8.836.2 Field Documentation

8.836.2.1 `uint8_t uim_verifyUIMPIN::pinID`

8.836.2.2 `uint8_t uim_verifyUIMPIN::pinLen`

8.836.2.3 `uint8_t uim_verifyUIMPIN::pinVal[255]`

8.837 UIMAuthenticateReq Struct Reference

Data Fields

- [UIMSessionInformation sessionInfo](#)
- [authenticationData authData](#)
- [ULONG * pIndicationToken](#)

8.837.1 Detailed Description

This structure contains information of the request parameters associated with a Authenticate API.

Parameters

<i>sessionInfo</i>	<ul style="list-style-type: none"> See UIMSessionInformation for more information.
<i>authData</i>	<ul style="list-style-type: none"> See authenticationData for more information.
<i>pIndicationToken(optional)</i>	<ul style="list-style-type: none"> Response in Indication. When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.837.2 Field Documentation

8.837.2.1 authenticationData UIMAuthenticateReq::authData

8.837.2.2 ULONG* UIMAuthenticateReq::pIndicationToken

8.837.2.3 UIMSessionInformation UIMAuthenticateReq::sessionInfo

8.838 UIMAuthenticateResp Struct Reference

Data Fields

- [cardResult](#) * [pCardResult](#)
- [authenticateResult](#) * [pAuthenticateResult](#)
- [ULONG](#) * [pIndicationToken](#)

8.838.1 Detailed Description

This structure contains information of the response parameters associated with a Authenticate API.

Parameters

<i>pCard</i> <i>Result(optional)</i>	<ul style="list-style-type: none">• See cardResult for more information.
<i>pAuthenticate</i> <i>Result(optional)</i>	<ul style="list-style-type: none">• See authenticateResult for more information.
<i>pIndication</i> <i>Token(optional)</i>	<ul style="list-style-type: none">• Response in Indication.• When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.838.2 Field Documentation

8.838.2.1 authenticateResult* UIMAuthenticateResp::pAuthenticateResult

8.838.2.2 cardResult* UIMAuthenticateResp::pCardResult

8.838.2.3 **ULONG*** `UIMAuthenticateResp::pIndicationToken`

8.839 UIMChangePinReq Struct Reference

Data Fields

- [UIMSessionInformation sessionInfo](#)
- [changeUIMPIN changePIN](#)
- **BYTE *** `pKeyReferenceID`
- **ULONG *** `pIndicationToken`

8.839.1 Detailed Description

This structure contains information of the request parameters associated with a Change PIN API.

Parameters

sessionInfo	<ul style="list-style-type: none"> • See UIMSessionInformation for more information.
<i>changePIN</i>	<ul style="list-style-type: none"> • See changeUIMPIN for more information.
<i>pKey</i> ↔ <i>Reference</i> ↔ <i>D(optional)</i>	<ul style="list-style-type: none"> • Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
<i>pIndication</i> ↔ <i>Token(optional)</i>	<ul style="list-style-type: none"> • Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.839.2 Field Documentation

8.839.2.1 **changeUIMPIN** `UIMChangePinReq::changePIN`

8.839.2.2 **ULONG*** `UIMChangePinReq::pIndicationToken`

8.839.2.3 **BYTE*** `UIMChangePinReq::pKeyReferenceID`

8.839.2.4 **UIMSessionInformation** `UIMChangePinReq::sessionInfo`

8.840 UIMDepersonalizationReq Struct Reference

Data Fields

- [depersonalizationInformation](#) [depersonalisationInfo](#)

8.840.1 Detailed Description

This structure contains information of the request parameters associated with a Depersonalization API.

Parameters

<i>depersonalisationInfo</i>	<ul style="list-style-type: none">• See depersonalizationInformation for more information.
------------------------------	--

8.840.2 Field Documentation

8.840.2.1 [depersonalizationInformation](#) UIMDepersonalizationReq::depersonalisationInfo

8.841 UIMDepersonalizationResp Struct Reference

Data Fields

- [remainingRetries](#) * [pRemainingRetries](#)

8.841.1 Detailed Description

This structure contains information of the response parameters associated with a Depersonalization API.

Parameters

<i>pRemainingRetries(optional)</i>	<ul style="list-style-type: none">• See remainingRetries for more information.
------------------------------------	--

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.841.2 Field Documentation

8.841.2.1 [remainingRetries](#)* UIMDepersonalizationResp::pRemainingRetries

8.842 UIMEventRegisterReqResp Struct Reference

Data Fields

- [ULONG eventMask](#)

8.842.1 Detailed Description

This structure contains information of the request parameters associated with a SLQSUIMEventRegister.

Parameters

<i>eventMask</i> (↔ Mandatory)	<ul style="list-style-type: none"> • Bitmask of the events that were successfully enabled. This result can be different from the mask used in the request when notifications are not supported. Additional bits are reserved for future use. <ul style="list-style-type: none"> – Bit 0 - Card status – Bit 1 - SAP connection – Bit 4 - Physical Slot Status
-----------------------------------	--

8.842.2 Field Documentation

8.842.2.1 ULONG UIMEventRegisterReqResp::eventMask

8.843 UIMGetCardStatusResp Struct Reference

Data Fields

- [cardStatus](#) * [pCardStatus](#)
- [hotSwapStatus](#) * [pHotSwapStatus](#)

8.843.1 Detailed Description

This structure contains information of the response parameters associated with a Get Card Status API.

Parameters

<i>pCard</i> (↔ Status(optional))	<ul style="list-style-type: none"> • See cardStatus for more information.
<i>pHotSwap</i> (↔ Status(optional))	<ul style="list-style-type: none"> • See hotSwapStatus for more information.

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.843.2 Field Documentation

8.843.2.1 `cardStatus*` UIMGetCardStatusResp::pCardStatus

8.843.2.2 `hotSwapStatus*` UIMGetCardStatusResp::pHotSwapStatus

8.844 UIMGetConfigurationReq Struct Reference

Data Fields

- [ULONG](#) * [pConfigurationMask](#)

8.844.1 Detailed Description

This structure contains information of the request parameters associated with to gets the modem configuration for the UIM module API.

Parameters

<i>pConfiguration↔ Mask(optional)</i>	<ul style="list-style-type: none">• Requested configurations<ul style="list-style-type: none">– Bit 0 - Automatic selection– Bit 1 - Personalization status– Bit 2 - Halt subscription– All other bits are reserved for future use
---	---

Note

- if the TLV is missing, the service returns all configuration items in the response.

8.844.2 Field Documentation

8.844.2.1 `ULONG*` UIMGetConfigurationReq::pConfigurationMask

8.845 UIMGetConfigurationResp Struct Reference

Data Fields

- [BYTE](#) * [pAutoSelection](#)
- [personalizationStatus](#) * [pPersonalizationStatus](#)
- [BYTE](#) * [pHaltSubscription](#)

8.845.1 Detailed Description

This structure contains information of the response parameters associated with a Read Transparent API.

Parameters

<i>pAuto</i> <i>Selection(optional)</i>	<ul style="list-style-type: none"> Indicates whether the modem is configured to automatically select the provisioning sessions at powerup. Valid values <ul style="list-style-type: none"> 0 - Automatic provisioning is off 1 - Automatic provisioning is on
<i>p</i> <i>Personalization</i> <i>Status(optional)</i>	<ul style="list-style-type: none"> See personalizationStatus for more information.
<i>pHalt</i> <i>Subscription(optional)</i>	<ul style="list-style-type: none"> Indicates if the modem is configured to publish the subscription after successful initialization. Valid values <ul style="list-style-type: none"> 0 - Modem proceeds with publishing the subscription 1 - Modem does not publish the subscription

8.845.2 Field Documentation

8.845.2.1 **BYTE*** `UIMGetConfigurationResp::pAutoSelection`

8.845.2.2 **BYTE*** `UIMGetConfigurationResp::pHaltSubscription`

8.845.2.3 **personalizationStatus*** `UIMGetConfigurationResp::pPersonalizationStatus`

8.846 UIMGetFileAttributesReq Struct Reference

Data Fields

- [UIMSessionInformation](#) `sessionInfo`
- [fileInfo](#) `fileIndex`
- ULONG *** `pIndicationToken`

8.846.1 Detailed Description

This structure contains information of the request parameters associated with a Get File Attributes API.

Parameters

<i>sessionInfo</i>	<ul style="list-style-type: none"> See UIMSessionInformation for more information.
<i>fileIndex</i>	
	<ul style="list-style-type: none"> See fileInfo for more information.
<i>pIndication</i> <i>Token(optional)</i>	<ul style="list-style-type: none"> Response in Indication. When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.846.2 Field Documentation

8.846.2.1 `fileInfo` `UIMGetFileAttributesReq::fileIndex`

8.846.2.2 `ULONG*` `UIMGetFileAttributesReq::pIndicationToken`

8.846.2.3 `UIMSessionInformation` `UIMGetFileAttributesReq::sessionInfo`

8.847 UIMGetFileAttributesResp Struct Reference

Data Fields

- `cardResult` * `pCardResult`
- `fileAttributes` * `pFileAttributes`
- `ULONG` * `pIndicationToken`

8.847.1 Detailed Description

This structure contains information of the response parameters associated with a Get File Attributes API.

Parameters

<i><code>pCard</code>↔ <code>Result(optional)</code></i>	<ul style="list-style-type: none"> • See cardResult for more information.
<i><code>pFile</code>↔ <code>Attributes(optional)</code></i>	<ul style="list-style-type: none"> • See fileAttributes for more information.
<i><code>pIndication</code>↔ <code>Token(optional)</code></i>	<ul style="list-style-type: none"> • Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.847.2 Field Documentation

8.847.2.1 `cardResult*` `UIMGetFileAttributesResp::pCardResult`

8.847.2.2 `fileAttributes*` `UIMGetFileAttributesResp::pFileAttributes`

8.847.2.3 **ULONG*** `UIMGetFileAttributesResp::pIndicationToken`

8.848 UIMGetSlotsStatusResp Struct Reference

Data Fields

- **BYTE*** `pNumberOfPhySlot`
- **UIMSlotsStatus*** `pUimSlotsStatus`

8.848.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

<i>pNumberOfPhySlot</i>	<ul style="list-style-type: none"> • Number of sets of the Slot Status.
<i>pUimSlotsStatus</i>	<ul style="list-style-type: none"> • Slots Status See UIMSlotsStatus for more information..

8.848.2 Field Documentation

8.848.2.1 **BYTE*** `UIMGetSlotsStatusResp::pNumberOfPhySlot`

8.848.2.2 **UIMSlotsStatus*** `UIMGetSlotsStatusResp::pUimSlotsStatus`

8.849 UIMPinResp Struct Reference

Data Fields

- **remainingRetries*** `pRemainingRetries`
- **encryptedPIN1*** `pEncryptedPIN1`
- **ULONG*** `pIndicationToken`

8.849.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

<i>pRemainingRetries</i> (optional)	<ul style="list-style-type: none">See remainingRetries for more information.
<i>pEncryptedPIN1</i> (optional)	<ul style="list-style-type: none">See encryptedPIN1 for more information.
<i>pIndicationToken</i> (optional)	<ul style="list-style-type: none">Response in Indication.When this TLV is present, it indicates that the result is provided in a subsequent indication.0xFFFFFFFF, if unavailable

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.849.2 Field Documentation

8.849.2.1 `encryptedPIN1`* UIMPinResp::pEncryptedPIN1

8.849.2.2 `ULONG`* UIMPinResp::pIndicationToken

8.849.2.3 `remainingRetries`* UIMPinResp::pRemainingRetries

8.850 UIMPowerDownReq Struct Reference

Data Fields

- [BYTE](#) slot

8.850.1 Detailed Description

This structure contains information of the request parameters associated with a Power Down.

Parameters

<i>slot</i>	<ul style="list-style-type: none">Indicates the slot to be used.<ul style="list-style-type: none">1 - Slot 12 - Slot 2
-------------	---

8.850.2 Field Documentation

8.850.2.1 `BYTE` UIMPowerDownReq::slot

8.851 UIMPowerUpReq Struct Reference

Data Fields

- [BYTE slot](#)
- [BYTE * plgnoreHotSwapSwitch](#)

8.851.1 Detailed Description

This structure contains information of the request parameters associated with a Power Down.

Parameters

<i>slot</i>	<ul style="list-style-type: none"> Indicates the slot to be used. <ul style="list-style-type: none"> – 1 - Slot 1 – 2 - Slot 2
<i>plgnoreHotSwapSwitch(optional)</i>	<ul style="list-style-type: none"> Hot-swap switch status. <ul style="list-style-type: none"> – 0 - Checks the hot-swap switch status – 1 - Ignores the hot-swap switch status

8.851.2 Field Documentation

8.851.2.1 [BYTE*](#) UIMPowerUpReq::plgnoreHotSwapSwitch

8.851.2.2 [BYTE](#) UIMPowerUpReq::slot

8.852 UIMReadTransparentReq Struct Reference

Data Fields

- [UIMSessionInformation sessionInfo](#)
- [fileInfo fileIndex](#)
- [readTransparentInfo readTransparent](#)
- [ULONG * pIndicationToken](#)
- [BYTE * pEncryptData](#)

8.852.1 Detailed Description

This structure contains information of the request parameters associated with a Read Transparent API.

Parameters

<i>sessionInfo</i>	<ul style="list-style-type: none"> • See UIMSessionInformation for more information.
<i>fileIndex</i>	<ul style="list-style-type: none"> • See fileInfo for more information.
<i>readTransparent</i>	<ul style="list-style-type: none"> • See readTransparentInfo for more information.
<i>pIndication</i> ↔ <i>Token(optional)</i>	<ul style="list-style-type: none"> • Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.
<i>pEncrypt</i> ↔ <i>Data(optional)</i>	<ul style="list-style-type: none"> • Encrypt Data. • Indicates whether the data read from the card is to be encrypted.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.852.2 Field Documentation

8.852.2.1 [fileInfo](#) UIMReadTransparentReq::fileIndex

8.852.2.2 [BYTE*](#) UIMReadTransparentReq::pEncryptData

8.852.2.3 [ULONG*](#) UIMReadTransparentReq::pIndicationToken

8.852.2.4 [readTransparentInfo](#) UIMReadTransparentReq::readTransparent

8.852.2.5 [UIMSessionInformation](#) UIMReadTransparentReq::sessionInfo

8.853 UIMReadTransparentResp Struct Reference

Data Fields

- [cardResult](#) * [pCardResult](#)
- [readResult](#) * [pReadResult](#)
- [ULONG](#) * [pIndicationToken](#)
- [BYTE](#) * [pEncryptedData](#)

8.853.1 Detailed Description

This structure contains information of the response parameters associated with a Read Transparent API.

Parameters

<i>pCardResult</i>	<ul style="list-style-type: none"> • See cardResult for more information.
<i>pReadResult</i>	<ul style="list-style-type: none"> • See readResult for more information.
<i>pIndication</i> ↔ <i>Token(optional)</i>	<ul style="list-style-type: none"> • Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.
<i>pEncrypted</i> ↔ <i>Data(optional)</i>	<ul style="list-style-type: none"> • Encrypted Data. • Indicates whether the data from the card passed in read_result is encrypted.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.853.2 Field Documentation

8.853.2.1 **cardResult*** UIMReadTransparentResp::pCardResult

8.853.2.2 **BYTE*** UIMReadTransparentResp::pEncryptedData

8.853.2.3 **ULONG*** UIMReadTransparentResp::pIndicationToken

8.853.2.4 **readResult*** UIMReadTransparentResp::pReadResult

8.854 UIMRefreshCompleteReq Struct Reference**Data Fields**

- [UIMSessionInformation sessionInfo](#)
- [BYTE refreshComplete](#)

8.854.1 Detailed Description

This structure contains information of the request parameters associated with a SLQSUIMRefreshComplete.

Parameters

<i>sessionInfo</i> ↔ <i>Mandatory</i>	<ul style="list-style-type: none"> • See UIMSessionInformation for more information.
<i>refresh</i> ↔ <i>Complete</i> ↔ <i>Mandatory</i>	<ul style="list-style-type: none"> • Indicates whether the refresh was successful. Valid values: <ul style="list-style-type: none"> – 0 - Refresh was not completed successfully – 1 - Refresh was completed successfully

8.854.2 Field Documentation

8.854.2.1 **BYTE** UIMRefreshCompleteReq::refreshComplete

8.854.2.2 **UIMSessionInformation** UIMRefreshCompleteReq::sessionInfo

8.855 UIMRefreshEvent Struct Reference

Data Fields

- [BYTE](#) stage
- [BYTE](#) mode
- [BYTE](#) sessionType
- [BYTE](#) aidLength
- [BYTE](#) aid [255]
- [WORD](#) numOfFiles
- [fileInfo](#) arrfileInfo [255]

8.855.1 Detailed Description

This structure contains information of parameters associated with the Refresh Event.

Parameters

<i>stage</i>	<ul style="list-style-type: none"> • Indicates the stage of the Refresh procedure. <ul style="list-style-type: none"> – 0 - Waiting for OK to refresh – 1 - Refresh started – 2 - Refresh ended successfully – 3 - Refresh failed
<i>mode</i>	<ul style="list-style-type: none"> • Indicates the Refresh mode. <ul style="list-style-type: none"> – 0 - Reset – 1 - Init – 2 - Init and FCN – 3 - FCN – 4 - Init and Full FCN – 5 - Application reset – 6 - 3G session reset
<i>sessionType</i>	<ul style="list-style-type: none"> • Indicates the session type. <ul style="list-style-type: none"> – 0 - Primary GW provisioning – 1 - Primary 1X provisioning – 2 - Secondary GW provisioning – 3 - Secondary 1X provisioning – 4 - Nonprovisioning on slot 1 – 5 - Nonprovisioning on slot 2 – 6 - Card on slot 1
Generated by Doxygen	<ul style="list-style-type: none"> – 7 - Card on slot 2 – 8 - Logical channel on slot 1 – 9 - Logical channel on slot 2

<i>aidLength</i>	<ul style="list-style-type: none"> Number of sets of the following elements <ul style="list-style-type: none"> Application Identifier
<i>aid</i>	<ul style="list-style-type: none"> Application identifier value or channel ID. This value is required for non-provisioning and for logical channel session types. It is ignored in all other cases
<i>numFiles</i>	<ul style="list-style-type: none"> Number of sets of the following elements: <ul style="list-style-type: none"> file_id path_len path
<i>arrfileInfo</i>	<ul style="list-style-type: none"> Array of file Information struct

8.855.2 Field Documentation

8.855.2.1 **BYTE** UIMRefreshEvent::aid[255]

8.855.2.2 **BYTE** UIMRefreshEvent::aidLength

8.855.2.3 **fileInfo** UIMRefreshEvent::arrfileInfo[255]

8.855.2.4 **BYTE** UIMRefreshEvent::mode

8.855.2.5 **WORD** UIMRefreshEvent::numOfFiles

8.855.2.6 **BYTE** UIMRefreshEvent::sessionType

8.855.2.7 **BYTE** UIMRefreshEvent::stage

8.856 UIMRefreshGetLastEventReq Struct Reference

Data Fields

- [UIMSessionInformation sessionInfo](#)

8.856.1 Detailed Description

This structure contains information of the request parameters associated with a SLQSUIMRefreshGetLastEvent.

Parameters

<i>sessionInfo</i> (↔ Mandatory)	<ul style="list-style-type: none">• See UIMSessionInformation for more information.
-------------------------------------	---

8.856.2 Field Documentation

8.856.2.1 [UIMSessionInformation](#) [UIMRefreshGetLastEventReq::sessionInfo](#)

8.857 UIMRefreshGetLastEventResp Struct Reference

Data Fields

- [UIMRefreshEvent](#) * [pRefreshEvent](#)

8.857.1 Detailed Description

This structure contains information of the response parameters associated with a [SLQSUIMRefreshGetLastEvent](#).

Parameters

<i>refreshEvent</i> (↔ Optional)	<ul style="list-style-type: none">• See UIMRefreshEvent for more information.
-------------------------------------	---

8.857.2 Field Documentation

8.857.2.1 [UIMRefreshEvent](#)* [UIMRefreshGetLastEventResp::pRefreshEvent](#)

8.858 UIMRefreshOKReq Struct Reference

Data Fields

- [UIMSessionInformation](#) [sessionInfo](#)
- [BYTE](#) [OKtoRefresh](#)

8.858.1 Detailed Description

This structure contains Parameters of the Session Information

Parameters

<i>sessionInfo</i>	<ul style="list-style-type: none"> • Session Information • See UIMSessionInformation for more information
<i>OKtoRefresh</i>	<ul style="list-style-type: none"> • Indicates whether a refresh is OK. Valid values: <ul style="list-style-type: none"> – 0 - Not OK to refresh – 1 - OK to refresh

8.858.2 Field Documentation

8.858.2.1 **BYTE** UIMRefreshOKReq::OKtoRefresh8.858.2.2 **UIMSessionInformation** UIMRefreshOKReq::sessionInfo

8.859 UIMRefreshRegisterReq Struct Reference

Data Fields

- [UIMSessionInformation](#) sessionInfo
- [registerRefresh](#) regRefresh

8.859.1 Detailed Description

This structure contains information of the request parameters associated with a Refresh Register.

Parameters

<i>sessionInfo</i>	<ul style="list-style-type: none"> • Session Information params • See UIMSessionInformation for more information
<i>regRefresh</i>	<ul style="list-style-type: none"> • Register Refresh parameters • See registerRefresh for more information

8.859.2 Field Documentation

8.859.2.1 **registerRefresh** UIMRefreshRegisterReq::regRefresh8.859.2.2 **UIMSessionInformation** UIMRefreshRegisterReq::sessionInfo

8.860 UIMSessionInformation Struct Reference

Data Fields

- [BYTE sessionType](#)
- [BYTE aidLength](#)
- [BYTE aid](#) [255]

8.860.1 Detailed Description

This structure contains the Session Information.

Parameters

<i>sessionType</i>	<ul style="list-style-type: none">• Indicates the session type.<ul style="list-style-type: none">– 0 - Primary GW provisioning– 1 - Primary 1X provisioning– 2 - Secondary GW provisioning– 3 - Secondary 1X provisioning– 4 - Non-provisioning on slot 1– 5 - Non-provisioning on slot 2– 6 - Card on slot 1– 7 - Card on slot 2– 8 - Logical channel on slot 1– 9 - Logical channel on slot 2
<i>aidLength</i>	<ul style="list-style-type: none">• Length of the following elements i.e. Application Identifier.
<i>aid</i>	<ul style="list-style-type: none">• Application identifier value or channel ID.• This value is required for non-provisioning and for logical channel session types. It is ignored in all other cases.

8.860.2 Field Documentation

8.860.2.1 **BYTE** UIMSessionInformation::aid[255]

8.860.2.2 **BYTE** UIMSessionInformation::aidLength

8.860.2.3 **BYTE** UIMSessionInformation::sessionType

8.861 UIMSetPinProtectionReq Struct Reference

Data Fields

- [UIMSessionInformation sessionInfo](#)
- [setPINProtection pinProtection](#)
- [BYTE *](#) pKeyReferenceID
- [ULONG *](#) pIndicationToken

8.861.1 Detailed Description

This structure contains information of the request parameters associated with a set pin protection API.

Parameters

<i>sessionInfo</i>	<ul style="list-style-type: none"> See UIMSessionInformation for more information.
<i>pinProtection</i>	<ul style="list-style-type: none"> See setPINProtection for more information.
<i>pKeyReferenceID(optional)</i>	<ul style="list-style-type: none"> Indicates the PIN key reference ID. Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
<i>pIndicationToken(optional)</i>	<ul style="list-style-type: none"> Response in Indication. When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.861.2 Field Documentation

8.861.2.1 **ULONG*** [UIMSetPinProtectionReq::pIndicationToken](#)

8.861.2.2 **setPINProtection** [UIMSetPinProtectionReq::pinProtection](#)

8.861.2.3 **BYTE*** [UIMSetPinProtectionReq::pKeyReferenceID](#)

8.861.2.4 **UIMSessionInformation** [UIMSetPinProtectionReq::sessionInfo](#)

8.862 UIMSlotsStatus Struct Reference

Data Fields

- [UIMSlotStatus](#) [uimSlotStatus](#) [255]

8.862.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

<i>uimSlotStatus</i> [↔ <i>MAX_SLOTS</i> ↔ <i>STATUS</i>]	<ul style="list-style-type: none"> Contain all slots status.
--	---

8.862.2 Field Documentation

8.862.2.1 UIMSlotStatus UIMSlotsStatus::uimSlotStatus[255]

8.863 UIMSlotStatus Struct Reference

Data Fields

- [ULONG uPhyCardStatus](#)
- [ULONG uPhySlotStatus](#)
- [BYTE bLogicalSlot](#)
- [BYTE bICCIDLength](#)
- [BYTE bICCID](#) [255]

8.863.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

<i>uPhyCardStatus</i>	<ul style="list-style-type: none"> State of the card in the Pyhsical Slot Status. <ul style="list-style-type: none"> 0x00 - Unknown. 0x01 - Absent. 0x02 - Present.
<i>uPhySlotStatus</i>	<ul style="list-style-type: none"> State of the Physical Slot status. <ul style="list-style-type: none"> 0x00 Inactive. 0x01 Activate.
<i>bLogicalSlot</i>	<ul style="list-style-type: none"> Logical Slot associated with this physical slot. THis is valid if the physical slot is active. <ul style="list-style-type: none"> 1 - Slot 1. 2 - Slot 2. 3 - Slot 3. 4 - Slot 4. 5 - Slot 5.
<i>bLogicalSlot</i>	<ul style="list-style-type: none"> Number of sets the sets of ICCID
<i>bICCID</i> [<i>MAX_I</i> ↔ <i>CCID_LENGTH</i>]	<ul style="list-style-type: none"> Contains the ICCID of the card in the physical slot.

8.863.2 Field Documentation

8.863.2.1 **BYTE** UIMSlotStatus::bICCID[255]

8.863.2.2 **BYTE** UIMSlotStatus::bICCIDLength

8.863.2.3 **BYTE** UIMSlotStatus::bLogicalSlot

8.863.2.4 **ULONG** UIMSlotStatus::uPhyCardStatus

8.863.2.5 **ULONG** UIMSlotStatus::uPhySlotStatus

8.864 UIMSlotStatusChangeInfo Struct Reference

Data Fields

- [UIMSlotsStatus slotsstatusChange](#)
- [BYTE bNumberOfPhySlots](#)

8.864.1 Detailed Description

Structure consist of cardstatus params

Parameters

<i>slotstatus</i> ↔ <i>Change</i>	<ul style="list-style-type: none">• See UIMSlotStatus for more information
<i>bNumberOf</i> ↔ <i>PhySlots</i>	<ul style="list-style-type: none">• Number of Physical Slot(s)

8.864.2 Field Documentation

8.864.2.1 **BYTE** UIMSlotStatusChangeInfo::bNumberOfPhySlots

8.864.2.2 **UIMSlotsStatus** UIMSlotStatusChangeInfo::slotsstatusChange

8.865 UIMStatusChangeInfo Struct Reference

Data Fields

- [cardStatus statusChange](#)

8.865.1 Detailed Description

Structure consist of cardstatus params

Parameters

<i>statusChange</i>	<ul style="list-style-type: none">• See cardStatus for more information
---------------------	---

8.865.2 Field Documentation

8.865.2.1 cardStatus UIMStatusChangeInfo::statusChange

8.866 UIMSwitchSlotReq Struct Reference

Data Fields

- [BYTE bLogicalSlot](#)
- [ULONG ulPhysicalSlot](#)

8.866.1 Detailed Description

This structure contains information of the request parameters associated with a Switch Slot.

Parameters

<i>bLogicalSlot</i>	<ul style="list-style-type: none">• Indicates the slot to be used.<ul style="list-style-type: none">– 1 - Slot 1– 2 - Slot 2– 3 - Slot 3– 4 - Slot 4– 5 - Slot 5
<i>bPhysicalSlot</i>	<ul style="list-style-type: none">• 1 - Slot 1• 2 - Slot 2• 3 - Slot 3• 4 - Slot 4• 5 - Slot 5

8.866.2 Field Documentation

8.866.2.1 BYTE UIMSwitchSlotReq::bLogicalSlot

8.866.2.2 **ULONG** UIMSwitchSlotReq::ulPhysicalSlot

8.867 UIMUnblockPinReq Struct Reference

Data Fields

- [UIMSessionInformation sessionInfo](#)
- [unblockUIMPIN unblockPIN](#)
- **BYTE** * [pKeyReferenceID](#)
- **ULONG** * [pIndicationToken](#)

8.867.1 Detailed Description

This structure contains information of the request parameters associated with a Unblock PIN API.

Parameters

sessionInfo	<ul style="list-style-type: none"> • See UIMSessionInformation for more information.
unblockPIN	<ul style="list-style-type: none"> • See unblockUIMPIN for more information.
pKeyReferenceID <i>(optional)</i>	<ul style="list-style-type: none"> • Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
pIndicationToken <i>(optional)</i>	<ul style="list-style-type: none"> • Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.867.2 Field Documentation

8.867.2.1 **ULONG*** UIMUnblockPinReq::pIndicationToken

8.867.2.2 **BYTE*** UIMUnblockPinReq::pKeyReferenceID

8.867.2.3 **UIMSessionInformation** UIMUnblockPinReq::sessionInfo

8.867.2.4 **unblockUIMPIN** UIMUnblockPinReq::unblockPIN

8.868 UIMVerifyPinReq Struct Reference

Data Fields

- [UIMSessionInformation sessionInfo](#)
- [verifyUIMPIN verifyPIN](#)
- [encryptedPIN1 * pEncryptedPIN1](#)
- [BYTE * pKeyReferenceID](#)
- [ULONG * pIndicationToken](#)

8.868.1 Detailed Description

This structure contains information of the request parameters associated with a verify PIN API.

Parameters

sessionInfo	<ul style="list-style-type: none"> • See UIMSessionInformation for more information.
verifyPIN	<ul style="list-style-type: none"> • See verifyUIMPIN for more information.
pEncryptedPIN1 <i>N1(optional)</i>	<ul style="list-style-type: none"> • See encryptedPIN1 for more information.
pKeyReferenceID <i>D(optional)</i>	<ul style="list-style-type: none"> • Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
pIndicationToken <i>Token(optional)</i>	<ul style="list-style-type: none"> • Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.868.2 Field Documentation

8.868.2.1 [encryptedPIN1 * UIMVerifyPinReq::pEncryptedPIN1](#)

8.868.2.2 [ULONG * UIMVerifyPinReq::pIndicationToken](#)

8.868.2.3 [BYTE * UIMVerifyPinReq::pKeyReferenceID](#)

8.868.2.4 [UIMSessionInformation UIMVerifyPinReq::sessionInfo](#)

8.868.2.5 verifyUIMPIN UIMVerifyPinReq::verifyPIN

8.869 UMTSInfo Struct Reference

Data Fields

- [WORD cellID](#)
- [BYTE plmn](#) [3]
- [WORD lac](#)
- [WORD uarfcn](#)
- [WORD psc](#)
- [SHORT rscp](#)
- [SHORT ecio](#)
- [BYTE umtsInst](#)
- [UMTSInstInfo UMTSInstInfo](#) [255]
- [BYTE geranInst](#)
- [geranInstInfo GeranInstInfo](#) [255]

8.869.1 Detailed Description

This structure contains information about the UMTS Network.

Parameters

<i>cellID</i>	<ul style="list-style-type: none"> • Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.
<i>plmn</i> [<i>PLMN_LENGTH</i>]	<ul style="list-style-type: none"> • MCC/MNC information coded as octet 3, 4, and 5. • This field is ignored when nmrCellID is not present.
<i>lac</i>	<ul style="list-style-type: none"> • Location area code. • This field is ignored when nmrCellID is not present. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>uarfcn</i>	<ul style="list-style-type: none"> • UTRA absolute RF channel number. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>psc</i>	<ul style="list-style-type: none"> • Primary scrambling code. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>rscp</i>	<ul style="list-style-type: none"> • Received signal code power. <ul style="list-style-type: none"> – 0xFFFF - Not Available

<i>ecio</i>	<ul style="list-style-type: none"> • ECIO(Signal-to-Interference-ratio). <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>umtsInst</i>	<ul style="list-style-type: none"> • Provides the number of set of UMTS info instances. • If 0(zero), then no information follows it.
<i>UMTSInstInfo</i> [↔ <i>MAX_DESCR</i> ↔ <i>PTION_LEN</i> ↔ <i>TH</i>]	<ul style="list-style-type: none"> • See UMTSInstInfo for more information.
<i>geranInst</i>	<ul style="list-style-type: none"> • Provides the number of set of GERAN info instances. • If 0(zero), then no information follows it.
<i>GeranInstInfo</i> [↔ <i>MAX_DESCR</i> ↔ <i>PTION_LEN</i> ↔ <i>TH</i>]	<ul style="list-style-type: none"> • See geranInstInfo for more information.

8.869.2 Field Documentation

8.869.2.1 WORD UMTSInfo::cellID

8.869.2.2 SHORT UMTSInfo::ecio

8.869.2.3 BYTE UMTSInfo::geranInst

8.869.2.4 [geranInstInfo](#) UMTSInfo::GeranInstInfo[255]

8.869.2.5 WORD UMTSInfo::lac

8.869.2.6 BYTE UMTSInfo::plmn[3]

8.869.2.7 WORD UMTSInfo::psc

8.869.2.8 SHORT UMTSInfo::rscp

8.869.2.9 WORD UMTSInfo::uarfcn

8.869.2.10 BYTE UMTSInfo::umtsInst

8.869.2.11 [UMTSInstInfo](#) UMTSInfo::UMTSInstInfo[255]

8.870 UMTSInstInfo Struct Reference

Data Fields

- [WORD umtsUarfcn](#)
- [WORD umtsPsc](#)
- [SHORT umtsRscp](#)
- [SHORT umtsEcio](#)

8.870.1 Detailed Description

This structure contains information about the UMTS Instances in UMTS Network.

Parameters

<i>umtsUarfcn</i>	<ul style="list-style-type: none">• UTRA absolute RF channel number.
<i>umtsPsc</i>	<ul style="list-style-type: none">• Primary scrambling code.
<i>umtsRscp</i>	<ul style="list-style-type: none">• Received signal code power.
<i>umtsEcio</i>	<ul style="list-style-type: none">• ECIO(Signal-to-Interference-ratio).

8.870.2 Field Documentation

8.870.2.1 **SHORT** UMTSinstInfo::umtsEcio

8.870.2.2 **WORD** UMTSinstInfo::umtsPsc

8.870.2.3 **SHORT** UMTSinstInfo::umtsRscp

8.870.2.4 **WORD** UMTSinstInfo::umtsUarfcn

8.871 umtsLTENbrCell Struct Reference

Data Fields

- [WORD](#) earfcn
- [WORD](#) pci
- [ULONG](#) rsrp
- [ULONG](#) rsrq
- [SHORT](#) srxlev
- [BYTE](#) cellsTDD

8.871.1 Detailed Description

This structure contains information about the UMTS LTE neighbour Cell.

Parameters

<i>earfcn</i>	<ul style="list-style-type: none"> E-UTRA absolute RF channel number of the detected cell.
<i>pci</i>	<ul style="list-style-type: none"> Physical cell ID of the detected cell. Range is defined in 3GPP TS 36.211
<i>rsrp</i>	<ul style="list-style-type: none"> Current received signal strength indication (in dBm) of the detected cell.
<i>rsrq</i>	<ul style="list-style-type: none"> Current reference signal received quality (in dB) of the detected cell.
<i>srxlev</i>	<ul style="list-style-type: none"> Cell selection Rx level (Srxlev) value of the detected cell in linear scale. This field is only valid when <code>wcdma_rrc_state</code> is not <code>NAS_WCDMA_RRC_STATE_CEL_FACH</code> or <code>NAS_WCDMA_RRC_STATE_CELL_DCH</code>.
<i>cellsTDD</i>	<ul style="list-style-type: none"> TRUE if the cell is TDD; FALSE if the cell is FDD.

8.871.2 Field Documentation

8.871.2.1 BYTE umtsLTENbrCell::cellsTDD

8.871.2.2 WORD umtsLTENbrCell::earfcn

8.871.2.3 WORD umtsLTENbrCell::pci

8.871.2.4 ULONG umtsLTENbrCell::rsrp

8.871.2.5 ULONG umtsLTENbrCell::rsrq

8.871.2.6 SHORT umtsLTENbrCell::srxlev

8.872 UMTSMinQoS Struct Reference

Data Fields

- [BYTE trafficClass](#)
- [ULONG maxUplinkBitrate](#)
- [ULONG maxDownlinkBitrate](#)
- [ULONG grntUplinkBitrate](#)
- [ULONG grntDownlinkBitrate](#)
- [BYTE qosDeliveryOrder](#)
- [ULONG maxSDUSize](#)
- [BYTE sduErrorRatio](#)
- [BYTE resBerRatio](#)
- [BYTE deliveryErrSDU](#)
- [ULONG transferDelay](#)
- [ULONG trafficPriority](#)

8.872.1 Detailed Description

This structure contains the UMTS Quality Of Service Information

Parameters

<i>trafficClass</i>	<ul style="list-style-type: none"> • 0x00 - Subscribed • 0x01 - Conversational • 0x02 - Streaming • 0x03 - Interactive • 0x04 - Background
<i>maxUplinkBitrate</i>	<ul style="list-style-type: none"> • Maximum uplink bit rate in bits/sec
<i>maxDownlink↔ Bitrate</i>	<ul style="list-style-type: none"> • Maximum downlink bit rate in bits/sec
<i>grntUplinkBitrate</i>	<ul style="list-style-type: none"> • Guaranteed uplink bit rate in bits/sec
<i>grntDownlink↔ Bitrate</i>	<ul style="list-style-type: none"> • Guaranteed downlink bit rate in bits/sec
<i>qosDelivery↔ Order</i>	<ul style="list-style-type: none"> - Qos delivery order • 0x00 - Subscribe • 0x01 - Delivery order on • 0x02 - Delivery order off
<i>maxSDUSize</i>	<ul style="list-style-type: none"> • Maximum SDU size
<i>sduErrorRatio</i>	<ul style="list-style-type: none"> - SDU error ratio • Target value for fraction of SDUs lost or detected as erroneous. • 0x00 - Subscribe • 0x01 - $1 \cdot 10^{-2}$ • 0x02 - $7 \cdot 10^{-3}$ • 0x03 - $1 \cdot 10^{-3}$ • 0x04 - $1 \cdot 10^{-4}$ • 0x05 - $1 \cdot 10^{-5}$ • 0x06 - $1 \cdot 10^{-6}$ • 0x07 - $1 \cdot 10^{-1}$

<i>resBerRatio</i>	<ul style="list-style-type: none"> - Residual bit error ratio • Target value for undetected bit error ratio in the delivered SDUs. • 0x00 - Subscribe • 0x01 - $5 \cdot 10^{-2}$ • 0x02 - $1 \cdot 10^{-2}$ • 0x03 - $5 \cdot 10^{-3}$ • 0x04 - $4 \cdot 10^{-3}$ • 0x05 - $1 \cdot 10^{-3}$ • 0x06 - $1 \cdot 10^{-4}$ • 0x07 - $1 \cdot 10^{-5}$ • 0x08 - $1 \cdot 10^{-6}$ • 0x09 - $1 \cdot 10^{-8}$
<i>deliveryErrSDU</i>	<ul style="list-style-type: none"> - delivery of erroneous SDUs • Indicates whether SDUs detected as erroneous shall be delivered or not. • 0x00 - Subscribe • 0x01 - $5 \cdot 10^{-2}$ • 0x02 - $1 \cdot 10^{-2}$ • 0x03 - $5 \cdot 10^{-3}$ • 0x04 - $4 \cdot 10^{-3}$ • 0x05 - $1 \cdot 10^{-3}$ • 0x06 - $1 \cdot 10^{-4}$ • 0x07 - $1 \cdot 10^{-5}$ • 0x08 - $1 \cdot 10^{-6}$ • 0x09 - $1 \cdot 10^{-8}$
<i>transferDelay</i>	<ul style="list-style-type: none"> - Transfer delay (ms) • Indicates the targeted time between a request to transfer an SDU at one SAP to its delivery at the other SAP in milliseconds.
<i>trafficPriority</i>	<ul style="list-style-type: none"> - Transfer handling priority • Specifies the relative importance for handling of SDUs that belong to the UMTS bearer, compared to the SDUs of other bearers.

8.872.2 Field Documentation

8.872.2.1 **BYTE** UMTSMinQoS::deliveryErrSDU

8.872.2.2 **ULONG** UMTSMinQoS::grntDownlinkBitrate

8.872.2.3 **ULONG** UMTSMinQoS::grntUplinkBitrate

8.872.2.4 **ULONG** UMTSMinQoS::maxDownlinkBitrate

8.872.2.5 **ULONG** UMTSMinQoS::maxSDUSize

8.872.2.6 **ULONG** UMTSMinQoS::maxUplinkBitrate

8.872.2.7 **BYTE** UMTSMinQoS::qosDeliveryOrder

8.872.2.8 **BYTE** UMTSMinQoS::resBerRatio

8.872.2.9 **BYTE** UMTSMinQoS::sduErrorRatio

8.872.2.10 **BYTE** UMTSMinQoS::trafficClass

8.872.2.11 **ULONG** UMTSMinQoS::trafficPriority

8.872.2.12 **ULONG** UMTSMinQoS::transferDelay

8.873 UMTSQoS Struct Reference

Data Fields

- [BYTE](#) trafficClass
- [ULONG](#) maxUplinkBitrate
- [ULONG](#) maxDownlinkBitrate
- [ULONG](#) grntUplinkBitrate
- [ULONG](#) grntDownlinkBitrate
- [BYTE](#) qosDeliveryOrder
- [ULONG](#) maxSDUSize
- [BYTE](#) sduErrorRatio
- [BYTE](#) resBerRatio
- [BYTE](#) deliveryErrSDU
- [ULONG](#) transferDelay
- [ULONG](#) trafficPriority

8.873.1 Detailed Description

This structure contains the UMTS Quality Of Service Information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>trafficClass</i>	<ul style="list-style-type: none">• 0x00 - Subscribed• 0x01 - Conversational• 0x02 - Streaming• 0x03 - Interactive• 0x04 - Background
<i>maxUplinkBitrate</i>	<ul style="list-style-type: none">• Maximum uplink bit rate in bits/sec

<i>maxDownlink</i> ↔ <i>Bitrate</i>	<ul style="list-style-type: none"> • Maximum downlink bit rate in bits/sec
<i>grntUplinkBitrate</i>	<ul style="list-style-type: none"> • Guaranteed uplink bit rate in bits/sec
<i>grntDownlink</i> ↔ <i>Bitrate</i>	<ul style="list-style-type: none"> • Guranteed downlink bit rate in bits/sec
<i>qosDelivery</i> ↔ <i>Order</i>	<ul style="list-style-type: none"> - Qos delivery order • 0x00 - Subscribe • 0x01 - delivery order on • 0x02 - delivery order off
<i>maxSDUSize</i>	<ul style="list-style-type: none"> • Maximum SDU size
<i>sduErrorRatio</i>	<ul style="list-style-type: none"> - SDU error ratio • Target value for fraction of SDUs lost or detected as erroneous. • 0x00 - Subscribe • 0x01 - $1 \cdot 10^{-2}$ • 0x02 - $7 \cdot 10^{-3}$ • 0x03 - $1 \cdot 10^{-3}$ • 0x04 - $1 \cdot 10^{-4}$ • 0x05 - $1 \cdot 10^{-5}$ • 0x06 - $1 \cdot 10^{-6}$ • 0x07 - $1 \cdot 10^{-1}$
<i>resBerRatio</i>	<ul style="list-style-type: none"> - Residual bit error ratio • Target value for undetected bit error ratio in in the delivered SDUs. • 0x00 - Subscribe • 0x01 - $5 \cdot 10^{-2}$ • 0x02 - $1 \cdot 10^{-2}$ • 0x03 - $5 \cdot 10^{-3}$ • 0x04 - $4 \cdot 10^{-3}$ • 0x05 - $1 \cdot 10^{-3}$ • 0x06 - $1 \cdot 10^{-4}$ • 0x07 - $1 \cdot 10^{-5}$ • 0x08 - $1 \cdot 10^{-6}$ • 0x09 - $1 \cdot 10^{-8}$

<i>deliveryErrSDU</i>	- Delivery of erroneous SDUs <ul style="list-style-type: none"> • Indicates whether SDUs detected as erroneous shall be delivered or not. • 0x00 - Subscribe • 0x01 - $5 \cdot 10^{-2}$ • 0x02 - $1 \cdot 10^{-2}$ • 0x03 - $5 \cdot 10^{-3}$ • 0x04 - $4 \cdot 10^{-3}$ • 0x05 - $1 \cdot 10^{-3}$ • 0x06 - $1 \cdot 10^{-4}$ • 0x07 - $1 \cdot 10^{-5}$ • 0x08 - $1 \cdot 10^{-6}$ • 0x09 - $1 \cdot 10^{-8}$
<i>transferDelay</i>	- Transfer delay (ms) <ul style="list-style-type: none"> • Indicates the targeted time between a request to transfer an SDU at one SAP to its delivery at the other SAP in milliseconds.
<i>trafficPriority</i>	- Transfer handling priority <ul style="list-style-type: none"> • Specifies the relative importance for handling of SDUs that belong to the UMTS bearer, compared to the SDUs of other bearers.

8.873.2 Field Documentation

8.873.2.1 **BYTE** UMTSQoS::deliveryErrSDU

8.873.2.2 **ULONG** UMTSQoS::grntDownlinkBitrate

8.873.2.3 **ULONG** UMTSQoS::grntUplinkBitrate

8.873.2.4 **ULONG** UMTSQoS::maxDownlinkBitrate

8.873.2.5 **ULONG** UMTSQoS::maxSDUSize

8.873.2.6 **ULONG** UMTSQoS::maxUplinkBitrate

8.873.2.7 **BYTE** UMTSQoS::qosDeliveryOrder

8.873.2.8 **BYTE** UMTSQoS::resBerRatio

8.873.2.9 **BYTE** UMTSQoS::sduErrorRatio

8.873.2.10 **BYTE** UMTSQoS::trafficClass

8.873.2.11 **ULONG** UMTSQoS::trafficPriority

8.873.2.12 **ULONG** UMTSQoS::transferDelay

8.874 UMTSReqQoSsigInd Struct Reference

Data Fields

- struct [UMTSQoS](#) [UMTSReqQoS](#)
- [BYTE](#) sigInd

8.874.1 Detailed Description

structure contains UMTS requested QoS with Signaling Indication flag

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

<i>UMTSReqQoS</i>	<ul style="list-style-type: none">• Contains the UMTS Quality Of Service Information
<i>SigInd</i>	<ul style="list-style-type: none">- Signaling Indication flag<ul style="list-style-type: none">• TRUE - Signaling indication ON• FALSE - Signaling indication OFF

8.874.2 Field Documentation

8.874.2.1 **BYTE** UMTSReqQoSSigInd::SigInd

8.874.2.2 **struct** UMTSQoS UMTSReqQoSSigInd::UMTSReqQoS

8.875 unblockUIMPIN Struct Reference

Data Fields

- [BYTE](#) pinID
- [BYTE](#) pukLen
- [BYTE](#) pukVal [255]
- [BYTE](#) newPINLen
- [BYTE](#) newPINVal [255]

8.875.1 Detailed Description

This structure contains the information about the unblock pin parameters.

Parameters

<i>pinID</i>	<ul style="list-style-type: none">• Indicates the PIN ID to be changed.<ul style="list-style-type: none">– 1 - PIN1 (also called PIN)– 2 - PIN2– 3 - Universal PIN
<i>pukLen</i>	<ul style="list-style-type: none">• Length of the following elements i.e. puk value.

<i>pukVal</i> [<i>MAX_PUK_LENGTH</i>]	<ul style="list-style-type: none"> PIN Unlock Key value. This value is a sequence of ASCII characters.
<i>pinLen</i>	<ul style="list-style-type: none"> Length of the following elements i.e. new pin value.
<i>pinVal</i> [<i>MAX_DESCRIPTION_LENGTH</i>]	<ul style="list-style-type: none"> New PIN value. This value is a sequence of ASCII characters.

8.875.2 Field Documentation

8.875.2.1 **BYTE** unblockUIMPIN::newPINLen

8.875.2.2 **BYTE** unblockUIMPIN::newPINVal[255]

8.875.2.3 **BYTE** unblockUIMPIN::pinID

8.875.2.4 **BYTE** unblockUIMPIN::pukLen

8.875.2.5 **BYTE** unblockUIMPIN::pukVal[255]

8.876 UniversalTime Struct Reference

Data Fields

- [WORD](#) year
- [BYTE](#) month
- [BYTE](#) day
- [BYTE](#) hour
- [BYTE](#) minute
- [BYTE](#) second
- [BYTE](#) dayOfWeek

8.876.1 Detailed Description

This structure contains the parameters for Universal Time Information.

Parameters

<i>year</i>	<ul style="list-style-type: none"> • Year.
<i>month</i>	<ul style="list-style-type: none"> • Month. <ul style="list-style-type: none"> – 1 is January and 12 is December.
<i>day</i>	<ul style="list-style-type: none"> • Day. <ul style="list-style-type: none"> – Range 1 to 31.
<i>hour</i>	<ul style="list-style-type: none"> • Hour. <ul style="list-style-type: none"> – Range 0 to 59.
<i>minute</i>	<ul style="list-style-type: none"> • Minute. <ul style="list-style-type: none"> – Range 0 to 59.
<i>second</i>	<ul style="list-style-type: none"> • Second. <ul style="list-style-type: none"> – Range 0 to 59.
<i>dayOfWeek</i>	<ul style="list-style-type: none"> • Day of the Week. <ul style="list-style-type: none"> – 0 is Monday and 6 is Sunday.

8.876.2 Field Documentation

8.876.2.1 BYTE UniversalTime::day

8.876.2.2 BYTE UniversalTime::dayOfWeek

8.876.2.3 BYTE UniversalTime::hour

8.876.2.4 BYTE UniversalTime::minute

8.876.2.5 BYTE UniversalTime::month

8.876.2.6 BYTE UniversalTime::second

8.876.2.7 WORD UniversalTime::year

8.877 unpack_dms_GetActivationState_t Struct Reference

Data Fields

- uint8_t [state](#)

8.877.1 Detailed Description

Parameters

<i>pActivation</i> ↔ <i>State[OUT]</i>	<ul style="list-style-type: none"> • Service Activation Code <ul style="list-style-type: none"> 0 - Service not activated 1 - Service activated 2 - Activation connecting 3 - Activation connected 4 - OTASP security authenticated 5 - OTASP NAM downloaded 6 - OTASP MDN downloaded 7 - OTASP IMSI downloaded 8 - OTASP PRL downloaded 9 - OTASP SPC downloaded 10 - OTASP settings committed
---	--

8.877.2 Field Documentation

8.877.2.1 `uint8_t unpack_dms_GetActivationState_t::state`

8.878 `unpack_dms_GetBandCapability_t` Struct Reference

Data Fields

- `uint32_t` [BandCapability](#)
- `uint16_t` [Tlvresult](#)

8.878.1 Field Documentation

8.878.1.1 `uint32_t unpack_dms_GetBandCapability_t::BandCapability`

8.878.1.2 `uint16_t unpack_dms_GetBandCapability_t::Tlvresult`

8.879 `unpack_dms_GetCrashAction_t` Struct Reference

Data Fields

- `uint8_t` [DevCrashState](#)
- `uint16_t` [Tlvresult](#)

8.879.1 Field Documentation

8.879.1.1 uint8_t unpack_dms_GetCrashAction_t::DevCrashState

8.879.1.2 uint16_t unpack_dms_GetCrashAction_t::Tlvresult

8.880 unpack_dms_GetCustFeature_t Struct Reference

Data Fields

- uint32_t [GpsEnable](#)
- uint8_t [DisableIMSI](#)
- uint16_t [IPFamSupport](#)
- uint8_t [RMAutoConnect](#)
- uint8_t [GPSSel](#)
- uint8_t [SMSSupport](#)
- uint8_t [IsVoiceEnabled](#)
- uint8_t [DHCPRelayEnabled](#)
- uint8_t [GPSLPM](#)
- uint16_t [Tlvresult](#)

8.880.1 Field Documentation

8.880.1.1 uint8_t unpack_dms_GetCustFeature_t::DHCPRelayEnabled

8.880.1.2 uint8_t unpack_dms_GetCustFeature_t::DisableIMSI

8.880.1.3 uint32_t unpack_dms_GetCustFeature_t::GpsEnable

8.880.1.4 uint8_t unpack_dms_GetCustFeature_t::GPSLPM

8.880.1.5 uint8_t unpack_dms_GetCustFeature_t::GPSSel

8.880.1.6 uint16_t unpack_dms_GetCustFeature_t::IPFamSupport

8.880.1.7 uint8_t unpack_dms_GetCustFeature_t::IsVoiceEnabled

8.880.1.8 uint8_t unpack_dms_GetCustFeature_t::RMAutoConnect

8.880.1.9 uint8_t unpack_dms_GetCustFeature_t::SMSSupport

8.880.1.10 uint16_t unpack_dms_GetCustFeature_t::Tlvresult

8.881 unpack_dms_GetCustFeaturesV2_t Struct Reference

Data Fields

- [DMSgetCustomFeatureV2](#) [GetCustomFeatureV2](#)
- uint16_t [Tlvresult](#)

8.881.1 Detailed Description

This structure contains customization settings set to modem unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none"> • Unpack Result
------------------	---

8.881.2 Field Documentation

8.881.2.1 **DMSgetCustomFeatureV2** `unpack_dms_GetCustFeaturesV2_t::GetCustomFeatureV2`

8.881.2.2 `uint16_t` `unpack_dms_GetCustFeaturesV2_t::Tlvresult`

8.882 `unpack_dms_GetDeviceCap_t` Struct Reference

Data Fields

- `uint32_t` [MaxTXChannelRate](#)
- `uint32_t` [MaxRXChannelRate](#)
- `uint32_t` [DataServiceCapability](#)
- `uint32_t` [SimCapability](#)
- `uint32_t` [RadiofacesSize](#)
- `uint8_t` [Radiofaces](#) [64]
- `uint16_t` [Tlvresult](#)

8.882.1 Field Documentation

8.882.1.1 `uint32_t` `unpack_dms_GetDeviceCap_t::DataServiceCapability`

8.882.1.2 `uint32_t` `unpack_dms_GetDeviceCap_t::MaxRXChannelRate`

8.882.1.3 `uint32_t` `unpack_dms_GetDeviceCap_t::MaxTXChannelRate`

8.882.1.4 `uint8_t` `unpack_dms_GetDeviceCap_t::Radiofaces`[64]

8.882.1.5 `uint32_t` `unpack_dms_GetDeviceCap_t::RadiofacesSize`

8.882.1.6 `uint32_t` `unpack_dms_GetDeviceCap_t::SimCapability`

8.882.1.7 `uint16_t` `unpack_dms_GetDeviceCap_t::Tlvresult`

8.883 `unpack_dms_GetDeviceCapabilities_t` Struct Reference

Data Fields

- `uint32_t` [maxTxChannelRate](#)
- `uint32_t` [maxRxChannelRate](#)
- `uint32_t` [dataServiceCaCapability](#)
- `uint32_t` [simCapability](#)
- `uint32_t` [radiofacesSize](#)
- `uint8_t` [Radiofaces](#) [255]

8.883.1 Detailed Description

Parameters

<i>maxTx↔ ChannelRate</i>	Maximum Tx transmission rate in bits per second.
<i>maxRx↔ ChannelRate</i>	Maximum Rx transmission rate in bits per second
<i>dataServiceCa↔ Capability</i>	data service capability
<i>simCapability</i>	SIM Capability
<i>radiofacesSize</i>	radio interface length
<i>Radiofaces</i>	radio interfaces

8.883.2 Field Documentation

8.883.2.1 `uint32_t unpack_dms_GetDeviceCapabilities_t::dataServiceCaCapability`

8.883.2.2 `uint32_t unpack_dms_GetDeviceCapabilities_t::maxRxChannelRate`

8.883.2.3 `uint32_t unpack_dms_GetDeviceCapabilities_t::maxTxChannelRate`

8.883.2.4 `uint8_t unpack_dms_GetDeviceCapabilities_t::Radiofaces[255]`

8.883.2.5 `uint32_t unpack_dms_GetDeviceCapabilities_t::radiofacesSize`

8.883.2.6 `uint32_t unpack_dms_GetDeviceCapabilities_t::simCapability`

8.884 unpack_dms_GetDeviceHardwareRev_t Struct Reference

Data Fields

- `uint8_t stringSize`
- `char String [255]`
- `uint16_t Tlvresult`

8.884.1 Field Documentation

8.884.1.1 `char unpack_dms_GetDeviceHardwareRev_t::String[255]`

8.884.1.2 `uint8_t unpack_dms_GetDeviceHardwareRev_t::stringSize`

8.884.1.3 `uint16_t unpack_dms_GetDeviceHardwareRev_t::Tlvresult`

8.885 unpack_dms_GetDeviceMfr_t Struct Reference

Data Fields

- `uint8_t stringSize`
- `char String [255]`
- `uint16_t Tlvresult`

8.885.1 Field Documentation

8.885.1.1 char unpack_dms_GetDeviceMfr_t::String[255]

8.885.1.2 uint8_t unpack_dms_GetDeviceMfr_t::stringSize

8.885.1.3 uint16_t unpack_dms_GetDeviceMfr_t::Tlvresult

8.886 unpack_dms_GetDeviceSerialNumbers_t Struct Reference

Data Fields

- uint8_t [esnSize](#)
- char [ESNString](#) [255]
- uint8_t [imeiSize](#)
- char [IMEIString](#) [255]
- uint8_t [meidSize](#)
- char [MEIDString](#) [255]
- uint8_t [imeiSvnSize](#)
- char [ImeiSvnString](#) [255]
- uint16_t [Tlvresult](#)

8.886.1 Field Documentation

8.886.1.1 uint8_t unpack_dms_GetDeviceSerialNumbers_t::esnSize

8.886.1.2 char unpack_dms_GetDeviceSerialNumbers_t::ESNString[255]

8.886.1.3 uint8_t unpack_dms_GetDeviceSerialNumbers_t::imeiSize

8.886.1.4 char unpack_dms_GetDeviceSerialNumbers_t::IMEIString[255]

8.886.1.5 uint8_t unpack_dms_GetDeviceSerialNumbers_t::imeiSvnSize

8.886.1.6 char unpack_dms_GetDeviceSerialNumbers_t::ImeiSvnString[255]

8.886.1.7 uint8_t unpack_dms_GetDeviceSerialNumbers_t::meidSize

8.886.1.8 char unpack_dms_GetDeviceSerialNumbers_t::MEIDString[255]

8.886.1.9 uint16_t unpack_dms_GetDeviceSerialNumbers_t::Tlvresult

8.887 unpack_dms_GetFirmwareInfo_t Struct Reference

Data Fields

- char [modelid_str](#) [20]
- char [bootversion_str](#) [85]
- char [appversion_str](#) [85]
- char [sku_str](#) [15]
- char [packageid_str](#) [85]
- char [carrier_str](#) [20]
- char [priversion_str](#) [16]
- char [cur_carr_name](#) [17]
- char [cur_carr_rev](#) [13]
- uint16_t [Tlvresult](#)

8.887.1 Detailed Description

Parameters

<i>modelid_str</i>	Mode ID String.
<i>bootversion_str</i>	Boot Version.
<i>appversion_str</i>	Application Version String.
<i>sku_str</i>	SKU String.
<i>packageid_str</i>	<ul style="list-style-type: none"> Package ID String. deprecated on EM/MC74xx(9x30) devices
<i>carrier_str</i>	Carrier String.
<i>priversion_str</i>	PRI Version String.
<i>priversion_str</i>	PRI Version String.
<i>cur_carr_name</i>	Current Carrier Name String.
<i>cur_carr_rev</i>	Current Carrier Revision String.
<i>Tlvresult</i>	Tlv Result.

8.887.2 Field Documentation

8.887.2.1 char unpack_dms_GetFirmwareInfo_t::appversion_str[85]

8.887.2.2 char unpack_dms_GetFirmwareInfo_t::bootversion_str[85]

8.887.2.3 char unpack_dms_GetFirmwareInfo_t::carrier_str[20]

8.887.2.4 char unpack_dms_GetFirmwareInfo_t::cur_carr_name[17]

8.887.2.5 char unpack_dms_GetFirmwareInfo_t::cur_carr_rev[13]

8.887.2.6 char unpack_dms_GetFirmwareInfo_t::modelid_str[20]

8.887.2.7 char unpack_dms_GetFirmwareInfo_t::packageid_str[85]

8.887.2.8 char unpack_dms_GetFirmwareInfo_t::priversion_str[16]

8.887.2.9 char unpack_dms_GetFirmwareInfo_t::sku_str[15]

8.887.2.10 uint16_t unpack_dms_GetFirmwareInfo_t::Tlvresult

8.888 unpack_dms_GetFirmwareRevision_t Struct Reference

Data Fields

- uint8_t [amssSize](#)
- char [AMSSString](#) [255]
- char [PRIString](#) [255]
- uint16_t [Tlvresult](#)

8.888.1 Field Documentation

8.888.1.1 `uint8_t unpack_dms_GetFirmwareRevision_t::amssSize`

8.888.1.2 `char unpack_dms_GetFirmwareRevision_t::AMSSString[255]`

8.888.1.3 `char unpack_dms_GetFirmwareRevision_t::PRIString[255]`

8.888.1.4 `uint16_t unpack_dms_GetFirmwareRevision_t::Tlvresult`

8.889 unpack_dms_GetFirmwareRevisions_t Struct Reference

Data Fields

- `uint8_t amssSize`
- `char AMSSString [255]`
- `uint8_t bootSize`
- `char BootString [255]`
- `uint8_t priSize`
- `char PRIString [255]`
- `uint16_t Tlvresult`

8.889.1 Detailed Description

Parameters

<i>amssstring</i>	AMSS revision string
<i>bootstring</i>	boot code revision string
<i>pristring</i>	PRI revision string

8.889.2 Field Documentation

8.889.2.1 `uint8_t unpack_dms_GetFirmwareRevisions_t::amssSize`

8.889.2.2 `char unpack_dms_GetFirmwareRevisions_t::AMSSString[255]`

8.889.2.3 `uint8_t unpack_dms_GetFirmwareRevisions_t::bootSize`

8.889.2.4 `char unpack_dms_GetFirmwareRevisions_t::BootString[255]`

8.889.2.5 `uint8_t unpack_dms_GetFirmwareRevisions_t::priSize`

8.889.2.6 `char unpack_dms_GetFirmwareRevisions_t::PRIString[255]`

8.889.2.7 `uint16_t unpack_dms_GetFirmwareRevisions_t::Tlvresult`

8.890 unpack_dms_GetFSN_t Struct Reference

Data Fields

- char [String](#) [255]
- uint16_t [Tlvresult](#)

8.890.1 Field Documentation

8.890.1.1 char unpack_dms_GetFSN_t::String[255]

8.890.1.2 uint16_t unpack_dms_GetFSN_t::Tlvresult

8.891 unpack_dms_GetHardwareRevision_t Struct Reference

Data Fields

- char [hwVer](#) [255]

8.891.1 Detailed Description

Parameters

<i>hwVer</i>	hardware vesion
--------------	-----------------

8.891.2 Field Documentation

8.891.2.1 char unpack_dms_GetHardwareRevision_t::hwVer[255]

8.892 unpack_dms_GetIMSI_t Struct Reference

Data Fields

- char [imsi](#) [255]
- uint16_t [Tlvresult](#)

8.892.1 Field Documentation

8.892.1.1 char unpack_dms_GetIMSI_t::imsi[255]

8.892.1.2 uint16_t unpack_dms_GetIMSI_t::Tlvresult

8.893 unpack_dms_GetManufacturer_t Struct Reference

Data Fields

- char [manufacturer](#) [255]
- uint16_t [Tlvresult](#)

8.893.1 Detailed Description

This structure is used to store device manufacturer information.

Parameters

<i>manufacturer</i> <i>[↔</i> <i>OUT]</i>	<ul style="list-style-type: none">• NULL terminated string
<i>Tlvresult</i>	<ul style="list-style-type: none">• Unpack Result

8.893.2 Field Documentation

8.893.2.1 char unpack_dms_GetManufacturer_t::manufacturer[255]

8.893.2.2 uint16_t unpack_dms_GetManufacturer_t::Tlvresult

8.894 unpack_dms_GetModelID_t Struct Reference

Data Fields

- char [modelid](#) [255]
- uint16_t [Tlvresult](#)

8.894.1 Detailed Description

Parameters

<i>modelid</i>	device model id
----------------	-----------------

8.894.2 Field Documentation

8.894.2.1 char unpack_dms_GetModelID_t::modelid[255]

8.894.2.2 uint16_t unpack_dms_GetModelID_t::Tlvresult

8.895 unpack_dms_GetNetworkTime_t Struct Reference

Data Fields

- uint16_t [source](#)
- uint64_t [timestamp](#)
- uint16_t [Tlvresult](#)

8.895.1 Detailed Description

Parameters

<i>source</i>	<ul style="list-style-type: none">• Source of timestamp 0 - 32 kHz device clock 1 - CDMA network 2 - cdma2000 1xEV-DO network
<i>timestamp</i>	<ul style="list-style-type: none">• Count of 1.25 ms that have elapsed from the start of GPS time (Jan 6, 1980)
<i>Tlvresult</i>	<ul style="list-style-type: none">• Unpack Result

Note

The source of the timestamp provided specifies how the timestamp was determined. The first network time that is available will be returned. If no network time is available, the timestamp is taken from the 32 kHz slow-clock of the device.

8.895.2 Field Documentation

8.895.2.1 `uint16_t unpack_dms_GetNetworkTime_t::source`

8.895.2.2 `uint64_t unpack_dms_GetNetworkTime_t::timestamp`

8.895.2.3 `uint16_t unpack_dms_GetNetworkTime_t::Tlvresult`

8.896 unpack_dms_GetOfflineReason_t Struct Reference

Data Fields

- `uint32_t * pReasonMask`
- `uint32_t * pbPlatform`
- `uint16_t Tlvresult`

8.896.1 Detailed Description

This structure is used to store reason why the operating mode of the device is currently offline.

Parameters

<i>pReasonMask</i> [OUT]	<ul style="list-style-type: none"> Optional parameter Bitmask of offline reasons <ul style="list-style-type: none"> 0x00000001 - Host image configuration issue 0x00000002 - PRI image configuration issue 0x00000004 - PRI version incompatible 0x00000008 - PRI copy issue All others - Reserved
<i>pbPlatform</i> [OUT]	<ul style="list-style-type: none"> Optional parameter Is the device offline due to a platform restriction? <ul style="list-style-type: none"> 0 - No 1 - Yes
<i>Tlvresult</i>	<ul style="list-style-type: none"> Unpack Result

8.896.2 Field Documentation

8.896.2.1 uint32_t* unpack_dms_GetOfflineReason_t::pbPlatform

8.896.2.2 uint32_t* unpack_dms_GetOfflineReason_t::pReasonMask

8.896.2.3 uint16_t unpack_dms_GetOfflineReason_t::Tlvresult

8.897 unpack_dms_GetPower_t Struct Reference

Data Fields

- uint32_t [OperationMode](#)
- uint32_t [OfflineReason](#)
- uint32_t [HardwareControlledMode](#)
- uint16_t [Tlvresult](#)

8.897.1 Detailed Description

Parameters

<i>OperationMode</i>	operating mode
<i>OfflineReason</i>	offline reason
<i>HardwareControlledMode</i>	hardware restricted mode

8.897.2 Field Documentation

8.897.2.1 uint32_t unpack_dms_GetPower_t::HardwareControlledMode

8.897.2.2 uint32_t unpack_dms_GetPower_t::OfflineReason

8.897.2.3 uint32_t unpack_dms_GetPower_t::OperationMode

8.897.2.4 uint16_t unpack_dms_GetPower_t::Tlvresult

8.898 unpack_dms_GetPRLVersion_t Struct Reference

Data Fields

- uint8_t [u8PRLPreference](#)
- uint16_t [u16PRLVersion](#)
- uint16_t [Tlvresult](#)

8.898.1 Field Documentation

8.898.1.1 uint16_t unpack_dms_GetPRLVersion_t::Tlvresult

8.898.1.2 uint16_t unpack_dms_GetPRLVersion_t::u16PRLVersion

8.898.1.3 uint8_t unpack_dms_GetPRLVersion_t::u8PRLPreference

8.899 unpack_dms_GetSerialNumbers_t Struct Reference

Data Fields

- char [esn](#) [255]
- char [imei_no](#) [255]
- char [meid](#) [255]
- char [imeisv_svn](#) [255]

8.899.1 Detailed Description

Parameters

<i>esn</i>	Electronic Serial Number of the device
<i>imei_no</i>	International Mobile Equipment Identity of the device.
<i>meid</i>	Mobile Equipment Identifier of the device.
<i>imeisv_svn</i>	imei software version revision

8.899.2 Field Documentation

8.899.2.1 char unpack_dms_GetSerialNumbers_t::esn[255]

8.899.2.2 char unpack_dms_GetSerialNumbers_t::imei_no[255]

8.899.2.3 char unpack_dms_GetSerialNumbers_t::imeisv_svn[255]

8.899.2.4 char unpack_dms_GetSerialNumbers_t::meid[255]

8.900 unpack_dms_GetUSBComp_t Struct Reference

Data Fields

- uint8_t [USBComp](#) [255]
- uint8_t [NumSupUSBComps](#)
- uint8_t [SupUSBComps](#)
- uint16_t [Tlvresult](#)

8.900.1 Field Documentation

8.900.1.1 uint8_t unpack_dms_GetUSBComp_t::NumSupUSBComps

8.900.1.2 uint8_t unpack_dms_GetUSBComp_t::SupUSBComps

8.900.1.3 uint16_t unpack_dms_GetUSBComp_t::Tlvresult

8.900.1.4 uint8_t unpack_dms_GetUSBComp_t::USBComp[255]

8.901 unpack_dms_GetVoiceNumber_t Struct Reference

Data Fields

- uint8_t [voiceNumberSize](#)
- char [VoiceNumber](#) [255]
- uint8_t [minSize](#)
- char [MIN](#) [255]
- uint16_t [Tlvresult](#)

8.901.1 Field Documentation

8.901.1.1 char unpack_dms_GetVoiceNumber_t::MIN[255]

8.901.1.2 uint8_t unpack_dms_GetVoiceNumber_t::minSize

8.901.1.3 uint16_t unpack_dms_GetVoiceNumber_t::Tlvresult

8.901.1.4 char unpack_dms_GetVoiceNumber_t::VoiceNumber[255]

8.901.1.5 uint8_t unpack_dms_GetVoiceNumber_t::voiceNumberSize

8.902 unpack_dms_ResetToFactoryDefaults_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.902.1 Detailed Description

This structure contains reset to factory default unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">Unpack Result
------------------	---

8.902.2 Field Documentation

8.902.2.1 uint16_t unpack_dms_ResetToFactoryDefaults_t::Tlvresult

8.903 unpack_dms_SetActivationStatusCallback_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.903.1 Detailed Description

unpack structure for Activation status callback

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">Unpack Result
------------------	---

8.903.2 Field Documentation

8.903.2.1 uint16_t unpack_dms_SetActivationStatusCallback_t::Tlvresult

8.904 unpack_dms_SetCrashAction_t Struct Reference

Data Fields

- uint8_t [notused](#)

8.904.1 Detailed Description

Modem response. Not used

8.904.2 Field Documentation

8.904.2.1 `uint8_t unpack_dms_SetCrashAction_t::notused`

8.905 `unpack_dms_SetCustFeature_t` Struct Reference

Data Fields

- `uint16_t Tlvresult`

8.905.1 Field Documentation

8.905.1.1 `uint16_t unpack_dms_SetCustFeature_t::Tlvresult`

8.906 `unpack_dms_SetCustFeaturesV2_t` Struct Reference

Data Fields

- `uint16_t Tlvresult`

8.906.1 Detailed Description

This structure contains customization settings set to modem unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Unpack Result
------------------	---

8.906.2 Field Documentation

8.906.2.1 `uint16_t unpack_dms_SetCustFeaturesV2_t::Tlvresult`

8.907 `unpack_dms_SetEventReport_ind_t` Struct Reference

Data Fields

- `dms_ActivationStatusTlv ActivationStatusTlv`
- `dms_OperatingModeTlv OperatingModeTlv`
- `uint16_t Tlvresult`

8.907.1 Detailed Description

DMS Event Report indication structure

Parameters

<i>Activation↔ StatusTlv</i>	<ul style="list-style-type: none"> See dms_ActivationStatusTlv
<i>Operating↔ ModeTlv</i>	<ul style="list-style-type: none"> See dms_OperatingModeTlv
<i>Tlvresult</i>	<ul style="list-style-type: none"> Unpack Result

8.907.2 Field Documentation

8.907.2.1 `dms_ActivationStatusTlv unpack_dms_SetEventReport_ind_t::ActivationStatusTlv`

8.907.2.2 `dms_OperatingModeTlv unpack_dms_SetEventReport_ind_t::OperatingModeTlv`

8.907.2.3 `uint16_t unpack_dms_SetEventReport_ind_t::Tlvresult`

8.908 unpack_dms_SetEventReport_t Struct Reference

Data Fields

- `uint16_t` [Tlvresult](#)

8.908.1 Field Documentation

8.908.1.1 `uint16_t unpack_dms_SetEventReport_t::Tlvresult`

8.909 unpack_dms_SetFirmwarePreference_t Struct Reference

Data Fields

- `uint16_t` [Tlvresult](#)

8.909.1 Field Documentation

8.909.1.1 `uint16_t unpack_dms_SetFirmwarePreference_t::Tlvresult`

8.910 unpack_dms_SetPower_t Struct Reference

Data Fields

- `uint16_t` [Tlvresult](#)

8.910.1 Field Documentation

8.910.1.1 uint16_t unpack_dms_SetPower_t::Tlvresult

8.911 unpack_dms_SetUSBComp_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.911.1 Field Documentation

8.911.1.1 uint16_t unpack_dms_SetUSBComp_t::Tlvresult

8.912 unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t Struct Reference

Data Fields

- uint8_t [type](#)
- uint8_t [source](#)
- uint16_t [Tlvresult](#)

8.912.1 Detailed Description

This structure contains the TLV required to Get Reset Info.

Parameters

--	--

8.912.2 Field Documentation

8.912.2.1 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::source

8.912.2.2 uint16_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::Tlvresult

8.912.2.3 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::type

8.913 unpack_dms_SLQSDmsSwiGetResetInfo_t Struct Reference

Data Fields

- uint8_t [type](#)
- uint8_t [source](#)
- uint16_t [Tlvresult](#)

8.913.1 Detailed Description

This structure contains the TLV required to Get Reset Info.

Parameters

--	--

8.913.2 Field Documentation

8.913.2.1 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_t::source

8.913.2.2 uint16_t unpack_dms_SLQSDmsSwiGetResetInfo_t::Tlvresult

8.913.2.3 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_t::type

8.914 unpack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.914.1 Detailed Description

This structure contains set registration state for different indication unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Unpack Result
------------------	---

8.914.2 Field Documentation

8.914.2.1 uint16_t unpack_dms_SLQSDmsSwiIndicationRegister_t::Tlvresult

8.915 unpack_dms_SLQSGetBandCapability_t Struct Reference

Data Fields

- uint64_t [bandCapability](#)
- int [is_LteBandCapability_Available](#)
- uint64_t [LteBandCapability](#)
- int [is_TdsBandCapability_Available](#)
- uint64_t [TdsBandCapability](#)

8.915.1 Detailed Description

This structure contains the Band Capabilities response.

Please check is_<Param_Name>_Available field for presence of optional parameters

Parameters

<i>band</i> ↔ <i>Capability[OUT]</i>	Bitmask of bands supported by the device <ul style="list-style-type: none"> • Bit 0 - Band class 0, A-system • Bit 1 - Band class 0, B-system • Bit 2 - Band class 1, all blocks • Bit 3 - Band class 2 • Bit 4 - Band class 3, A-system • Bit 5 - Band class 4, all blocks • Bit 6 - Band class 5, all blocks • Bit 7 - GSM DCS band (1800) • Bit 8 - GSM Extended GSM (E-GSM) band (900) • Bit 9 - GSM Primary GSM (P-GSM) band (900) • Bit 10 - Band class 6 • Bit 11 - Band class 7 • Bit 12 - Band class 8 • Bit 13 - Band class 9 • Bit 14 - Band class 10 • Bit 15 - Band class 11 • Bit 16 - GSM 450 band • Bit 17 - GSM 480 band • Bit 18 - GSM 750 band • Bit 19 - GSM 850 band • Bit 20 - GSM railways GSM band (900) • Bit 21 - GSM PCS band (1900) • Bit 22 - WCDMA (Europe, Japan, and China) 2100 band • Bit 23 - WCDMA US PCS 1900 band • Bit 24 - WCDMA (Europe and China) DCS 1800 band • Bit 25 - WCDMA US 1700 band • Bit 26 - WCDMA US 850 band • Bit 27 - WCDMA Japan 800 band • Bit 28 - Band class 12 • Bit 29 - Band class 14 • Bit 30 - Reserved • Bit 31 - Band class 15 • Bits 32 through 47 - Reserved • Bit 48 - WCDMA Europe 2600 band • Bit 49 - WCDMA Europe and Japan 900 band • Bit 50 - WCDMA Japan 1700 band • Bits 51 through 55 - Reserved • Bit 56 - Band class 16 • Bit 57 - Band class 17 • Bit 58 - Band class 18 • Bit 59 - Band class 19
---	--

<i>LteBand</i> ↔ <i>Capability[OUT]</i>	Bitmask of LTE bands supported by the device <ul style="list-style-type: none"> • Bit 0 - LTE EUTRAN Band 1 UL:1920-1980; DL: 2110-2170 • Bit 1 - LTE EUTRAN Band 2 UL:1850-1910; DL: 1930-1990 • Bit 2 - LTE EUTRAN Band 3 UL:1710-1785; DL: 1805-1880 • Bit 3 - LTE EUTRAN Band 4 UL:1710-1755; DL: 2110-2155 • Bit 4 - LTE EUTRAN Band 5 UL: 824-849; DL: 869-894 • Bit 5 - LTE EUTRAN Band 6 UL: 830-840; DL: 875-885 • Bit 6 - LTE EUTRAN Band 7 UL:2500-2570; DL: 2620-2690 • Bit 7 - LTE EUTRAN Band 8 UL: 880-915; DL: 925-960 • Bit 8 - LTE EUTRAN Band 9 UL:1749.9-1784.9; DL: 1844.9-1879.9 • Bit 9 - LTE EUTRAN Band 10 UL:1710-1770; DL: 2110-2170 • Bit 10 - LTE EUTRAN Band 11 UL:1427.9-1452.9; DL: 1475.9-1500.9 • Bit 11 - LTE EUTRAN Band 12 UL:698-716; DL: 728-746 • Bit 12 - LTE EUTRAN Band 13 UL: 777-787; DL: 746-756 • Bit 13 - LTE EUTRAN Band 14 UL: 788-798; DL: 758-768 • Bits 14 and 15 - Reserved • Bit 16 - LTE EUTRAN Band 17 UL: 704-716; DL: 734-746 • Bit 17 - LTE EUTRAN Band 18 UL: 815-830; DL: 860-875 • Bit 18 - LTE EUTRAN Band 19 UL: 830-845; DL: 875-890 • Bit 19 - LTE EUTRAN Band 20 UL: 832-862; DL: 791-821 • Bit 20 - LTE EUTRAN Band 21 UL: 1447.9-1462.9; DL: 1495.9-1510.9 • Bit 21 - Reserved • Bit 22 - LTE EUTRAN Band 23 UL: 2000-2020; DL: 2180-2200 • Bit 23 - LTE EUTRAN Band 24 UL: 1626.5-1660.5; DL: 1525-1559 • Bit 24 - LTE EUTRAN Band 25 UL: 1850-1915; DL: 1930-1995 • Bit 25 - LTE EUTRAN Band 26 UL: 814-849; DL: 859-894 • Bit 26 - Reserved • Bit 27 - LTE EUTRAN Band 28 UL: 703-748; DL: 758-803 • Bit 28 - LTE EUTRAN Band 29 UL: 1850-1910 or 1710-1755; DL: 716-728 • Bits 29 through 31 - Reserved • Bit 32 - LTE EUTRAN Band 33 UL: 1900-1920; DL: 1900-1920 • Bit 33 - LTE EUTRAN Band 34 UL: 2010-2025; DL: 2010-2025 • Bit 34 - LTE EUTRAN Band 35 UL: 1850-1910; DL: 1850-1910 • Bit 35 - LTE EUTRAN Band 36 UL: 1930-1990; DL: 1930-1990 • Bit 36 - LTE EUTRAN Band 37 UL: 1910-1930; DL: 1910-1930 • Bit 37 - LTE EUTRAN Band 38 UL: 2570-2620; DL: 2570-2620 • Bit 38 - LTE EUTRAN Band 39 UL: 1880-1920; DL: 1880-1920 • Bit 39 - LTE EUTRAN Band 40 UL: 2300-2400; DL: 2300-2400 • Bit 40 - LTE EUTRAN Band 41 UL: 2496-2690; DL: 2496-2690 • Bit 41 - LTE EUTRAN Band 42 UL: 3400-3600; DL: 3400-3600 • Bit 42 - LTE EUTRAN Band 43 UL: 3600-3800; DL: 3600-3800 • Bits 43 through 64 - Reserved
--	---

<i>TdsBand</i> ↔ <i>Capability[OUT]</i>	Bitmask of TDS bands supported by the device. <ul style="list-style-type: none"> • Bit 0 - TDS Band A 1900 to 1920 MHz, 2010 to 2020 MHz • Bit 1 - TDS Band B 1850 to 1910 MHz, 1930 to 1990 MHz • Bit 2 - TDS Band C 1910 to 1930 MHz • Bit 3 - TDS Band D 2570 to 2620 MHz • Bit 4 - TDS Band E 2300 to 2400 MHz • Bit 5 - TDS Band F 1880 to 1920 MHz
--	--

8.915.2 Field Documentation

8.915.2.1 `uint64_t unpack_dms_SLQSGetBandCapability_t::bandCapability`

8.915.2.2 `int unpack_dms_SLQSGetBandCapability_t::is_LteBandCapability_Available`

8.915.2.3 `int unpack_dms_SLQSGetBandCapability_t::is_TdsBandCapability_Available`

8.915.2.4 `uint64_t unpack_dms_SLQSGetBandCapability_t::LteBandCapability`

8.915.2.5 `uint64_t unpack_dms_SLQSGetBandCapability_t::TdsBandCapability`

8.916 `unpack_dms_SLQSGetERIFile_t` Struct Reference

Data Fields

- `uint16_t` [Tlvresult](#)
- [eriDataparams](#) `eriFile`

8.916.1 Detailed Description

This structure contains Get ERI file unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none"> • Unpack Result
------------------	---

8.916.2 Field Documentation

8.916.2.1 `eriDataparams unpack_dms_SLQSGetERIFile_t::eriFile`

8.916.2.2 `uint16_t unpack_dms_SLQSGetERIFile_t::Tlvresult`

8.917 unpack_dms_SLQSSwiClearDyingGaspStatistics_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.917.1 Detailed Description

This structure contains Clear Dying GASP unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Unpack Result
------------------	---

8.917.2 Field Documentation

8.917.2.1 uint16_t unpack_dms_SLQSSwiClearDyingGaspStatistics_t::Tlvresult

8.918 unpack_dms_SLQSSwiGetCrashInfo_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)
- [crashInfoParams](#) [crashInfoParam](#)

8.918.1 Detailed Description

This structure contains SWI get crash information unpack information

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Unpack Result
<i>crashInfoParam</i> <i>Param[OUT]</i>	<ul style="list-style-type: none">• See crashInfoParams

8.918.2 Field Documentation

8.918.2.1 [crashInfoParams](#) unpack_dms_SLQSSwiGetCrashInfo_t::crashInfoParam

8.918.2.2 uint16_t unpack_dms_SLQSSwiGetCrashInfo_t::Tlvresult

8.919 unpack_dms_SLQSSwiGetDyingGaspCfg_t Struct Reference

Data Fields

- [packgetDyingGaspCfg](#) * [pGetDyingGaspCfg](#)
- [uint16_t](#) [Tlvresult](#)

8.919.1 Detailed Description

This structure contains Get Dying GASP Config unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Unpack Result
------------------	---

8.919.2 Field Documentation

8.919.2.1 [packgetDyingGaspCfg](#)* [unpack_dms_SLQSSwiGetDyingGaspCfg_t::pGetDyingGaspCfg](#)

8.919.2.2 [uint16_t](#) [unpack_dms_SLQSSwiGetDyingGaspCfg_t::Tlvresult](#)

8.920 unpack_dms_SLQSSwiGetDyingGaspStatistics_t Struct Reference

Data Fields

- [packgetDyingGaspStatistics](#) * [pGetDyingGaspStatistics](#)
- [uint16_t](#) [Tlvresult](#)

8.920.1 Detailed Description

This structure contains Get Dying GASP Statistics.

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Unpack Result
------------------	---

8.920.2 Field Documentation

8.920.2.1 [packgetDyingGaspStatistics](#)* [unpack_dms_SLQSSwiGetDyingGaspStatistics_t::pGetDyingGaspStatistics](#)

8.920.2.2 uint16_t unpack_dms_SLQSSwiGetDyingGaspStatistics_t::Tlvresult

8.921 unpack_dms_SLQSSwiGetFirmwareCurr_t Struct Reference

Data Fields

- uint8_t numEntries
- image_info_t * pCurrImgInfo
- char priver [16]
- char pkgver [16]
- char fwvers [16]
- char carrier [16]

8.921.1 Detailed Description

Parameters

<i>numEntries</i> [↔ N/OUT]	<ul style="list-style-type: none"> • Number of entries in the image list to follow • The size of the list pCurrImgInfo must be specified when calling the API
<i>pCurrImgInfo</i> [↔ OUT]	<ul style="list-style-type: none"> • Currently Active Image List
<i>priver</i> [OUT]	<ul style="list-style-type: none"> • PRI version of the currently running firmware
<i>pkgver</i> [OUT]	<ul style="list-style-type: none"> • Package version of the currently running firmware
<i>fwvers</i> [OUT]	<ul style="list-style-type: none"> • firmware version of the currently running firmware
<i>carrier</i> [OUT]	<ul style="list-style-type: none"> • Carrier string of the currently running firmware

8.921.2 Field Documentation

8.921.2.1 char unpack_dms_SLQSSwiGetFirmwareCurr_t::carrier[16]

8.921.2.2 char unpack_dms_SLQSSwiGetFirmwareCurr_t::fwvers[16]

8.921.2.3 uint8_t unpack_dms_SLQSSwiGetFirmwareCurr_t::numEntries

8.921.2.4 image_info_t* unpack_dms_SLQSSwiGetFirmwareCurr_t::pCurrImgInfo

8.921.2.5 char unpack_dms_SLQSSwiGetFirmwareCurr_t::pkgver[16]

8.921.2.6 `char unpack_dms_SLQSSwiGetFirmwareCurr_t::priver[16]`

8.922 `unpack_dms_SLQSSwiGetFwUpdateStatus_t` Struct Reference

Data Fields

- `uint32_t` [ResCode](#)
- `uint8_t` [imgType](#)
- `uint32_t` [refData](#)
- `uint8_t` [refString](#) [15]
- `uint8_t` [logString](#) [255]
- `uint16_t` [Tlvresult](#)

8.922.1 Detailed Description

This structure is used to store Firmware Update Status

Parameters

<i>ResCode</i>	<ul style="list-style-type: none"> • FW Update Result Code • Values: <ul style="list-style-type: none"> – 0x00000001 - Successful – 0xFFFFFFFF - Unknown (due to power off reset after firmware update) – 0x100000nn - File update errors while nn will be the exact error number: <ul style="list-style-type: none"> * 00 - General error – 0x200000nn - NVUP update errors while nn will be the exact error number: <ul style="list-style-type: none"> * 00 - General error – 0x40000nnn - FOTA update agent errors while nnn will be the exact error number: <ul style="list-style-type: none"> * 000 ~ 0FF - Insignia defined error code * 100 ~ 1FF - Sierra defined error code * See <code>qaGobiApiTableFwDldErrorCodes.h</code> for more detailed information – 0x800000nn - FDT/SSDP reported errors while nn will be the exact error number <ul style="list-style-type: none"> * See <code>qaGobiApiTableFwDldErrorCodes.h</code> for more detailed information
<i>imgType</i>	<ul style="list-style-type: none"> • Optional parameter • Firmware image type that failed the update
<i>refData</i>	<ul style="list-style-type: none"> • Optional parameter • Failed image reference data • This is normally the offset of the image that caused the failure
<i>refString</i>	<ul style="list-style-type: none"> • Optional parameter • Failed image reference string. This is normally the partition name of the image that caused the failure if applicable.

<i>logString</i>	<ul style="list-style-type: none"> Optional parameter Failed image reference string. This is normally the partition name of the image that caused the failure if applicable.
<i>Tlvresult</i>	<ul style="list-style-type: none"> Unpack Result

8.922.2 Field Documentation

8.922.2.1 `uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::imgType`

8.922.2.2 `uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::logString[255]`

8.922.2.3 `uint32_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::refData`

8.922.2.4 `uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::refString[15]`

8.922.2.5 `uint32_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::ResCode`

8.922.2.6 `uint16_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::Tlvresult`

8.923 unpack_dms_SLQSSwiGetHostDevInfo_t Struct Reference

Data Fields

- `uint16_t Tlvresult`
- `char manString [255]`
- `char modelString [255]`
- `char swVerString [255]`
- `char plasmaIDString [255]`

8.923.1 Detailed Description

This structure contains SWI get host device info unpack information

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none"> unpack Result
<i>manString[OUT]</i>	<ul style="list-style-type: none"> optional parameter, host device manufacture
<i>modelString[OUT]</i>	<ul style="list-style-type: none"> optional parameter, host device model
<i>swVerString[OUT]</i>	<ul style="list-style-type: none"> optional parameter, host device software version
<i>plasmaID</i>	

8.923.2 Field Documentation

8.923.2.1 char unpack_dms_SLQSSwiGetHostDevInfo_t::manString[255]

8.923.2.2 char unpack_dms_SLQSSwiGetHostDevInfo_t::modelString[255]

8.923.2.3 char unpack_dms_SLQSSwiGetHostDevInfo_t::plasmaIDString[255]

8.923.2.4 char unpack_dms_SLQSSwiGetHostDevInfo_t::swVerString[255]

8.923.2.5 uint16_t unpack_dms_SLQSSwiGetHostDevInfo_t::Tlvresult

8.924 unpack_dms_SLQSSwiGetOSInfo_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)
- char [nameString](#) [255]
- char [versionString](#) [255]

8.924.1 Detailed Description

This structure contains SWI get host os info unpack information

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none"> • pack Result
<i>nameString</i> [<i>O↔UT</i>]	<ul style="list-style-type: none"> • optional parameter, host operating system name
<i>VersionString</i> [<i>↔OUT</i>]	<ul style="list-style-type: none"> • optional parameter, host operating system version

8.924.2 Field Documentation

8.924.2.1 char unpack_dms_SLQSSwiGetOSInfo_t::nameString[255]

8.924.2.2 uint16_t unpack_dms_SLQSSwiGetOSInfo_t::Tlvresult

8.924.2.3 char unpack_dms_SLQSSwiGetOSInfo_t::versionString[255]

8.925 unpack_dms_SLQSSwiGetSerialNoExt_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)
- char [meidString](#) [8]

8.925.1 Detailed Description

This structure contains SWI get device serial number extension unpack information

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• pack Result
<i>meidString[OUT]</i>	<ul style="list-style-type: none">• optional parameter, mobile equipment identifier

8.925.2 Field Documentation

8.925.2.1 char unpack_dms_SLQSSwiGetSerialNoExt_t::meidString[8]

8.925.2.2 uint16_t unpack_dms_SLQSSwiGetSerialNoExt_t::Tlvresult

8.926 unpack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.926.1 Detailed Description

This structure contains set Dying GASP Config unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Unpack Result
------------------	---

8.926.2 Field Documentation

8.926.2.1 uint16_t unpack_dms_SLQSSwiSetDyingGaspCfg_t::Tlvresult

8.927 unpack_dms_SLQSSwiSetHostDevInfo_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.927.1 Detailed Description

This structure contains SWI set host dev info unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Unpack Result
------------------	---

8.927.2 Field Documentation

8.927.2.1 uint16_t unpack_dms_SLQSSwiSetHostDevInfo_t::Tlvresult

8.928 unpack_dms_SLQSSwiSetOSInfo_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.928.1 Detailed Description

This structure contains SWI set host OS info unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">• Unpack Result
------------------	---

8.928.2 Field Documentation

8.928.2.1 uint16_t unpack_dms_SLQSSwiSetOSInfo_t::Tlvresult

8.929 unpack_dms_SLQSUIMGetState_t Struct Reference

Data Fields

- uint8_t [state](#)
- uint16_t [Tlvresult](#)

8.929.1 Detailed Description

This structure contains UIM get state unpack information

Parameters

<i>state[OUT]</i>	<ul style="list-style-type: none"> UIM state
<i>Tlvresult</i>	<ul style="list-style-type: none"> Unpack Result

8.929.2 Field Documentation

8.929.2.1 uint8_t unpack_dms_SLQSUIMGetState_t::state

8.929.2.2 uint16_t unpack_dms_SLQSUIMGetState_t::Tlvresult

8.930 unpack_dms_UIMGetControlKeyStatus_t Struct Reference

Data Fields

- uint8_t [facilityState](#)
- uint8_t [verifyRetriesLeft](#)
- uint8_t [unblockRetriesLeft](#)
- uint16_t [Tlvresult](#)

8.930.1 Detailed Description

This structure contains PIN retries status

Parameters

<i>facilityState[Optional UT]</i>	<ul style="list-style-type: none"> Control key status <ul style="list-style-type: none"> 0 - Deactivated 1 - Activated 2 - Blocked
<i>verifyRetriesLeft[Optional Left[OUT]]</i>	<ul style="list-style-type: none"> Optional parameter Upon operational failure, this will indicate number of retries left, after which PIN will be blocked. <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
<i>unblockRetriesLeft[Optional Left[OUT]]</i>	<ul style="list-style-type: none"> Optional parameter Upon operational failure, this will indicate number of unblock retries left, after which the PIN will be permanently blocked i.e. UIM is unusable. <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
Generated by Doxygen	
<i>Tlvresult</i>	<ul style="list-style-type: none"> Unpack Result

8.930.2 Field Documentation

8.930.2.1 uint8_t unpack_dms_UIMGetControlKeyStatus_t::facilityState

8.930.2.2 uint16_t unpack_dms_UIMGetControlKeyStatus_t::Tlvresult

8.930.2.3 uint8_t unpack_dms_UIMGetControlKeyStatus_t::unblockRetriesLeft

8.930.2.4 uint8_t unpack_dms_UIMGetControlKeyStatus_t::verifyRetriesLeft

8.931 unpack_dms_UIMGetICCID_t Struct Reference

Data Fields

- uint8_t [stringSize](#)
- uint8_t [String](#) [255]
- uint16_t [Tlvresult](#)

8.931.1 Detailed Description

This structure contains Get ICCID pack

Parameters

<i>stringSize</i>	<ul style="list-style-type: none">• Size of String.
<i>String</i>	<ul style="list-style-type: none">• ICCID String.
<i>Tlvresult</i>	<ul style="list-style-type: none">• Pack result.

8.931.2 Field Documentation

8.931.2.1 uint8_t unpack_dms_UIMGetICCID_t::String[255]

8.931.2.2 uint8_t unpack_dms_UIMGetICCID_t::stringSize

8.931.2.3 uint16_t unpack_dms_UIMGetICCID_t::Tlvresult

8.932 unpack_dms_UIMGetPINStatus_t Struct Reference

Data Fields

- uint8_t [p1Status](#)

- uint8_t [p1VerifyRetriesLeft](#)
- uint8_t [p1UnblockRetriesLeft](#)
- uint8_t [p2Status](#)
- uint8_t [p2VerifyRetriesLeft](#)
- uint8_t [p2UnblockRetriesLeft](#)
- uint16_t [Tlvresult](#)

8.932.1 Detailed Description

This structure contains Get PIN Status unpack information

Parameters

<i>p1Status[OUT]</i>	<ul style="list-style-type: none"> • PIN1 status(0xFFFFFFFF - Unknown) <ul style="list-style-type: none"> – 0 - PIN not initialized – 1 - PIN enabled, not verified – 2 - PIN enabled, verified – 3 - PIN disabled – 4 - PIN blocked – 5 - PIN permanently blocked
<i>p1Verify↔RetriesLeft[Optional UT]</i>	<ul style="list-style-type: none"> • Optional parameter • Upon operational failure, this will indicate number of retries left of PIN1, after which PIN will be blocked. <ul style="list-style-type: none"> – 0xFFFFFFFF - Unknown
<i>p1Unblock↔RetriesLeft[Optional UT]</i>	<ul style="list-style-type: none"> • Optional parameter • Upon operational failure, this will indicate number of unblock retries left of PIN1, after which the PIN will be permanently blocked i.e. UIM is unusable. <ul style="list-style-type: none"> – 0xFFFFFFFF - Unknown
<i>p2Status[OUT]</i>	<ul style="list-style-type: none"> • PIN2 status(0xFFFFFFFF - Unknown) <ul style="list-style-type: none"> – 0 - PIN not initialized – 1 - PIN enabled, not verified – 2 - PIN enabled, verified – 3 - PIN disabled – 4 - PIN blocked – 5 - PIN permanently blocked
<i>p2Verify↔RetriesLeft[Optional UT]</i>	<ul style="list-style-type: none"> • Optional parameter • Upon operational failure, this will indicate number of retries left of PIN2, after which PIN will be blocked. <ul style="list-style-type: none"> – 0xFFFFFFFF - Unknown

<i>p2UnblockRetriesLeft[OUT]</i>	<ul style="list-style-type: none"> • Optional parameter • Upon operational failure, this will indicate number of unblock retries left of PIN2, after which the PIN will be permanently blocked i.e. UIM is unusable. <ul style="list-style-type: none"> – 0xFFFFFFFF - Unknown
<i>Tlvresult</i>	<ul style="list-style-type: none"> • Unpack Result

8.932.2 Field Documentation

8.932.2.1 `uint8_t unpack_dms_UIMGetPINStatus_t::p1Status`

8.932.2.2 `uint8_t unpack_dms_UIMGetPINStatus_t::p1UnblockRetriesLeft`

8.932.2.3 `uint8_t unpack_dms_UIMGetPINStatus_t::p1VerifyRetriesLeft`

8.932.2.4 `uint8_t unpack_dms_UIMGetPINStatus_t::p2Status`

8.932.2.5 `uint8_t unpack_dms_UIMGetPINStatus_t::p2UnblockRetriesLeft`

8.932.2.6 `uint8_t unpack_dms_UIMGetPINStatus_t::p2VerifyRetriesLeft`

8.932.2.7 `uint16_t unpack_dms_UIMGetPINStatus_t::Tlvresult`

8.933 `unpack_dms_UIMSetControlKeyProtection_t` Struct Reference

Data Fields

- `uint8_t verifyRetriesLeft`
- `uint16_t Tlvresult`

8.933.1 Detailed Description

This structure contains UIM Set control key protection unpack information

Parameters

<i>verifyRetriesLeft[OUT]</i>	<ul style="list-style-type: none"> • Optional parameter • number of retries left after which the control key is blocked
<i>Tlvresult</i>	<ul style="list-style-type: none"> • Unpack Result

8.933.2 Field Documentation

8.933.2.1 uint16_t unpack_dms_UIMSetControlKeyProtection_t::Tlvresult

8.933.2.2 uint8_t unpack_dms_UIMSetControlKeyProtection_t::verifyRetriesLeft

8.934 unpack_dms_UIMSetPINProtection_t Struct Reference

Data Fields

- uint8_t [verifyRetriesLeft](#)
- uint8_t [unblockRetriesLeft](#)
- uint16_t [Tlvresult](#)

8.934.1 Detailed Description

This structure contains PIN retries status

Parameters

<i>verifyRetriesLeft</i> <i>Left[OUT]</i>	<ul style="list-style-type: none"> • Optional parameter • Upon operational failure, this will indicate number of retries left, after which PIN will be blocked. <ul style="list-style-type: none"> – 0xFFFFFFFF - Unknown
<i>unblockRetriesLeft</i> <i>Left[OUT]</i>	<ul style="list-style-type: none"> • Optional parameter • Upon operational failure, this will indicate number of unblock retries left, after which the PIN will be permanently blocked i.e. UIM is unusable. <ul style="list-style-type: none"> – 0xFFFFFFFF - Unknown
<i>Tlvresult</i>	<ul style="list-style-type: none"> • Unpack Result

8.934.2 Field Documentation

8.934.2.1 uint16_t unpack_dms_UIMSetPINProtection_t::Tlvresult

8.934.2.2 uint8_t unpack_dms_UIMSetPINProtection_t::unblockRetriesLeft

8.934.2.3 uint8_t unpack_dms_UIMSetPINProtection_t::verifyRetriesLeft

8.935 unpack_dms_UIMUnlockControlKey_t Struct Reference

Data Fields

- uint8_t [unblockRetriesLeft](#)
- uint16_t [Tlvresult](#)

8.935.1 Detailed Description

This structure contains UIM Set control key protection unpack information

Parameters

<i>unlockRetriesLeft</i> <i>OUT</i>	<ul style="list-style-type: none"> Optional parameter number of unlock retries left after which the control key is permanently blocked
<i>Tlvresult</i>	<ul style="list-style-type: none"> Unpack Result

8.935.2 Field Documentation

8.935.2.1 uint16_t unpack_dms_UIMUnlockControlKey_t::Tlvresult

8.935.2.2 uint8_t unpack_dms_UIMUnlockControlKey_t::unlockRetriesLeft

8.936 unpack_fms_GetImagesPreference_t Struct Reference

Data Fields

- uint32_t [ImageListSize](#)
- FMSPrefImageList * [pImageList](#)
- uint16_t [Tlvresult](#)

8.936.1 Detailed Description

This structure contains the Get Image Preference information unpack

Parameters

<i>listSize</i>	<ul style="list-style-type: none"> The number of elements in the image list
<i>pListEntries</i>	<ul style="list-style-type: none"> Array of Image entries with size provided by previous field See FMSImageElement
<i>Tlvresult</i>	<ul style="list-style-type: none"> Unpack result

8.936.2 Field Documentation

8.936.2.1 `uint32_t` unpack_fms_GetImagesPreference_t::ImageListSize

8.936.2.2 `FMSPrefImageList*` unpack_fms_GetImagesPreference_t::pImageList

8.936.2.3 `uint16_t` unpack_fms_GetImagesPreference_t::Tlvresult

8.937 unpack_fms_GetStoredImages_t Struct Reference

Data Fields

- `uint32_t` [imagelistSize](#)
- `FMSImageList` [imageList](#)
- `uint16_t` [Tlvresult](#)

8.937.1 Detailed Description

This structure contains the Get Stored Images unpack

Parameters

<i>listSize</i>	<ul style="list-style-type: none"> • The number of elements in the image list
<i>imageList</i>	<ul style="list-style-type: none"> • Array of Image entries with size provided by previous field • See FMSImageElement
<i>Tlvresult</i>	<ul style="list-style-type: none"> • Unpack result

8.937.2 Field Documentation

8.937.2.1 `FMSImageList` unpack_fms_GetStoredImages_t::imageList

8.937.2.2 `uint32_t` unpack_fms_GetStoredImages_t::imagelistSize

8.937.2.3 `uint16_t` unpack_fms_GetStoredImages_t::Tlvresult

8.938 unpack_fms_SetImagesPreference_t Struct Reference

Data Fields

- `uint32_t` [ImageTypesSize](#)
- `uint8_t` [ImageTypes](#) [255]
- `uint16_t` [Tlvresult](#)

8.938.1 Detailed Description

This structure contains the Set Images Preference unpack

Parameters

<i>ImageTypesSize</i>	<ul style="list-style-type: none"> Image Type Size
<i>ImageTypes</i>	<ul style="list-style-type: none"> Image Type
<i>Tlvresult</i>	<ul style="list-style-type: none"> Unpack result

8.938.2 Field Documentation

8.938.2.1 `uint8_t unpack_fms_SetImagesPreference_t::ImageTypes[255]`

8.938.2.2 `uint32_t unpack_fms_SetImagesPreference_t::ImageTypesSize`

8.938.2.3 `uint16_t unpack_fms_SetImagesPreference_t::Tlvresult`

8.939 `unpack_loc_BestAvailPos_Ind_t` Struct Reference

Data Fields

- `uint32_t status`
- `uint32_t * pXid`
- `uint64_t * pLatitude`
- `uint64_t * pLongitude`
- `uint32_t * pHorUncCircular`
- `uint32_t * pAltitudeWrtEllipsoid`
- `uint32_t * pVertUnc`
- `uint64_t * pTimestampUtc`
- `uint32_t * pTimeUnc`
- `uint32_t * pHorUncEllipseSemiMinor`
- `uint32_t * pHorUncEllipseSemiMajor`
- `uint32_t * pHorUncEllipseOrientAzimuth`
- `uint8_t * pHorCirConf`
- `uint8_t * pHorEllpConf`
- `uint32_t * pHorReliability`
- `uint32_t * pSpeedHorizontal`
- `uint32_t * pSpeedUnc`
- `uint32_t * pAltitudeWrtMeanSeaLevel`
- `uint8_t * pVertConfidence`
- `uint32_t * pVertReliability`
- `uint32_t * pSpeedVertical`
- `uint32_t * pSpeedVerticalUnc`

- uint32_t * [pHeading](#)
- uint32_t * [pHeadingUnc](#)
- uint32_t * [pMagneticDeviation](#)
- uint32_t * [pTechnologyMask](#)
- [loc_precisionDilution](#) * [pPrecisionDilution](#)
- [loc_gpsTime](#) * [pGpsTime](#)
- uint32_t * [pTimeSrc](#)
- [loc_sensorDataUsage](#) * [pSensorDataUsage](#)
- [loc_svUsedforFix](#) * [pSvUsedforFix](#)
- uint16_t [Tlvresult](#)

8.939.1 Detailed Description

This structure contains Best Available Position

Parameters

<i>status</i>	<ul style="list-style-type: none"> Valid values: <ul style="list-style-type: none"> eQMI_LOC_SUCCESS (0) - Request was completed successfully eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline eQMI_LOC_TIMEOUT (6) - Request failed because it timed out eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure
<i>xid</i>	Transaction ID that was specified in the Get Best Available Position request.
<i>pLatitude</i>	<ul style="list-style-type: none"> Latitude (specified in WGS84 datum) Type - Floating point Units - Degrees Range - -90.0 to 90.0 Positive values indicate northern latitude Negative values indicate southern latitude
<i>pLongitude</i>	<ul style="list-style-type: none"> Longitude (specified in WGS84 datum) Type - Floating point Units - Degrees Range - -180.0 to 180.0 Positive values indicate eastern latitude Negative values indicate western latitude

<i>pHorUncCircular</i>	<ul style="list-style-type: none"> • Horizontal position uncertainty. • Units - Meters
<i>pAltitudeWrt↔ Ellipsoid</i>	<ul style="list-style-type: none"> • Altitude With Respect to WGS84 Ellipsoid. • Units - Meters • Range -500 to 15883
<i>pVertUnc</i>	<ul style="list-style-type: none"> • Vertical uncertainty. • Units - Meters
<i>pTimestampUtc</i>	<ul style="list-style-type: none"> • UTC timestamp • Units - Milliseconds since Jan. 1, 1970
<i>pTimeUnc</i>	<ul style="list-style-type: none"> • Time uncertainty. • Units - Milliseconds
<i>pHorUnc↔ EllipseSemi↔ Minor</i>	<ul style="list-style-type: none"> • Semi-minor axis of horizontal elliptical uncertainty. • Units - Meters
<i>pHorUnc↔ EllipseSemi↔ Major</i>	<ul style="list-style-type: none"> • Semi-major axis of horizontal elliptical uncertainty. • Units: Meters
<i>pHorUnc↔ EllipseOrient↔ Azimuth</i>	<ul style="list-style-type: none"> • Elliptical horizontal uncertainty azimuth of orientation. • Units - Decimal degrees • Range - 0 to 180
<i>pHorCirConf</i>	<ul style="list-style-type: none"> • Horizontal circular uncertainty confidence • Units: Precent • Range: 0 to 99
<i>pHorEllpConf</i>	<ul style="list-style-type: none"> • Horizontal elliptical uncertainty confidence • Units: Precent • Range: 0 to 99
<i>pHorReliability</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk – 2 - Location reliability is low; little or no cross-checking is possible. – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed

<i>pSpeed</i> ↔ <i>Horizontal</i>	<ul style="list-style-type: none"> • Horizontal speed. • Units - Meters/second
<i>pSpeedUnc</i>	<ul style="list-style-type: none"> • 3-D Speed uncertainty. • Units - Meters/second.
<i>pAltitudeWrt</i> ↔ <i>MeanSeaLevel</i>	<ul style="list-style-type: none"> • Altitude With Respect to Sea Level. • Units - Meters
<i>pVertConfidence</i>	<ul style="list-style-type: none"> • Vertical uncertainty confidence. • Units - Percentage • Range 0 to 99
<i>pVertReliability</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk. – 2 - Location reliability is low; little or no cross-checking is possible – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed
<i>pSpeedVertical</i>	<ul style="list-style-type: none"> • Vertical speed. • Units - Meters/second
<i>pSpeed</i> ↔ <i>VerticalUnc</i>	<ul style="list-style-type: none"> • Vertical speed • Units: Meters/second
<i>pHeading</i>	<ul style="list-style-type: none"> • Heading. • Units - Degree • Range 0 to 359.999
<i>pHeadingUnc</i>	<ul style="list-style-type: none"> • Heading uncertainty. • Units - Degree • Range 0 to 359.999
<i>pMagnetic</i> ↔ <i>Deviation</i>	<ul style="list-style-type: none"> • Difference between the bearing to true north and the bearing shown on a magnetic compass. The deviation is positive when the magnetic north is east of true north.

<i>pTechnology↔ Mask</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0x00000001 - Satellites were used to generate the fix – 0x00000002 - Cell towers were used to generate the fix – 0x00000004 - Wi-Fi access points were used to generate the fix – 0x00000008 - Sensors were used to generate the fix – 0x00000010 - Reference Location was used to generate the fix – 0x00000020 - Coarse position injected into the location engine was used to generate the fix – 0x00000040 - AFLT was used to generate the fix – 0x00000080 - GNSS and network-provided measurements were used to generate the fix
<i>-pPrecision↔ Dilution</i>	<ul style="list-style-type: none"> • See loc_precisionDilution for more information
<i>-pGpsTime</i>	<ul style="list-style-type: none"> • See loc_gpsTime for more information
<i>pTimeSrc</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Invalid time. – 1 - Time is set by the 1X system. – 2 - Time is set by WCDMA/GSM time tagging. – 3 - Time is set by an external injection. – 4 - Time is set after decoding over-the-air GPS navigation data from one GPS satellite. – 5 - Time is set after decoding over-the-air GPS navigation data from multiple satellites. – 6 - Both time of the week and the GPS week number are known. – 7 - Time is set by the position engine after the fix is obtained – 8 - Time is set by the position engine after performing SFT, this is done when the clock time uncertainty is large. – 9 - Time is set after decoding GLO satellites. – 10- Time is set after transforming the GPS to GLO time – 11- Time is set by the sleep time tag provided by the WCDMA network. – 12- Time is set by the sleep time tag provided by the GSM network – 13- Source of the time is unknown – 14- Time is derived from the system clock (better known as the slow clock); GNSS time is maintained irrespective of the GNSS receiver state – 15- Time is set after decoding QZSS satellites. – 16- Time is set after decoding BDS satellites.
<i>-pSensorData↔ Usage</i>	<ul style="list-style-type: none"> • See loc_sensorDataUsage for more information
<i>-pSvUsedforFix</i>	<ul style="list-style-type: none"> • See loc_svUsedforFix for more information

8.939.2 Field Documentation

8.939.2.1 uint32_t* unpack_loc_BestAvailPos_Ind_t::pAltitudeWrtEllipsoid

- 8.939.2.2 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pAltitudeWrtMeanSeaLevel`
- 8.939.2.3 `loc_gpsTime*` `unpack_loc_BestAvailPos_Ind_t::pGpsTime`
- 8.939.2.4 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pHeading`
- 8.939.2.5 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pHeadingUnc`
- 8.939.2.6 `uint8_t*` `unpack_loc_BestAvailPos_Ind_t::pHorCirConf`
- 8.939.2.7 `uint8_t*` `unpack_loc_BestAvailPos_Ind_t::pHorEllpConf`
- 8.939.2.8 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pHorReliability`
- 8.939.2.9 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pHorUncCircular`
- 8.939.2.10 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseOrientAzimuth`
- 8.939.2.11 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseSemiMajor`
- 8.939.2.12 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseSemiMinor`
- 8.939.2.13 `uint64_t*` `unpack_loc_BestAvailPos_Ind_t::pLatitude`
- 8.939.2.14 `uint64_t*` `unpack_loc_BestAvailPos_Ind_t::pLongitude`
- 8.939.2.15 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pMagneticDeviation`
- 8.939.2.16 `loc_precisionDilution*` `unpack_loc_BestAvailPos_Ind_t::pPrecisionDilution`
- 8.939.2.17 `loc_sensorDataUsage*` `unpack_loc_BestAvailPos_Ind_t::pSensorDataUsage`
- 8.939.2.18 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pSpeedHorizontal`
- 8.939.2.19 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pSpeedUnc`
- 8.939.2.20 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pSpeedVertical`
- 8.939.2.21 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pSpeedVerticalUnc`
- 8.939.2.22 `loc_svUsedforFix*` `unpack_loc_BestAvailPos_Ind_t::pSvUsedforFix`
- 8.939.2.23 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pTechnologyMask`
- 8.939.2.24 `uint32_t*` `unpack_loc_BestAvailPos_Ind_t::pTimeSrc`

8.939.2.25 uint64_t* unpack_loc_BestAvailPos_Ind_t::pTimestampUtc

8.939.2.26 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTimeUnc

8.939.2.27 uint8_t* unpack_loc_BestAvailPos_Ind_t::pVertConfidence

8.939.2.28 uint32_t* unpack_loc_BestAvailPos_Ind_t::pVertReliability

8.939.2.29 uint32_t* unpack_loc_BestAvailPos_Ind_t::pVertUnc

8.939.2.30 uint32_t* unpack_loc_BestAvailPos_Ind_t::pXid

8.939.2.31 uint32_t unpack_loc_BestAvailPos_Ind_t::status

8.939.2.32 uint16_t unpack_loc_BestAvailPos_Ind_t::Tlvresult

8.940 unpack_loc_Delete_Assist_Data_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.940.1 Detailed Description

This structure contains LOC delete assist data unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none"> • Unpack result.
------------------	--

8.940.2 Field Documentation

8.940.2.1 uint16_t unpack_loc_Delete_Assist_Data_t::Tlvresult

8.941 unpack_loc_DeleteAssistData_Ind_t Struct Reference

Data Fields

- uint32_t [status](#)
- uint16_t [Tlvresult](#)

8.941.1 Detailed Description

Contain the parameters passed for SetLocDeleteAssistDataCallback by the device.

Parameters

<i>status</i>	<ul style="list-style-type: none"> • Status of the Delete Assist Data request • Valid values: <ul style="list-style-type: none"> – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure
<i>Tlvresult</i>	<ul style="list-style-type: none"> • unpack result

8.941.2 Field Documentation

8.941.2.1 uint32_t unpack_loc_DeleteAssistData_Ind_t::status

8.941.2.2 uint16_t unpack_loc_DeleteAssistData_Ind_t::Tlvresult

8.942 unpack_loc_EngineState_Ind_t Struct Reference

Data Fields

- uint32_t [engineState](#)
- uint16_t [Tlvresult](#)

8.942.1 Detailed Description

This structure contains LOC Engine State field.

Parameters

<i>engineState</i>	<ul style="list-style-type: none"> • Location engine state. • Valid values <ul style="list-style-type: none"> – 1 - Location engine is on – 2 - Location engine is off
<i>Tlvresult</i>	<ul style="list-style-type: none"> • unpack result

8.942.2 Field Documentation

8.942.2.1 uint32_t unpack_loc_EngineState_Ind_t::engineState

8.942.2.2 uint16_t unpack_loc_EngineState_Ind_t::Tlvresult

8.943 unpack_loc_EventRegister_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.943.1 Detailed Description

This structure contains Event Register unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none"> • Unpack result.
------------------	--

8.943.2 Field Documentation

8.943.2.1 uint16_t unpack_loc_EventRegister_t::Tlvresult

8.944 unpack_loc_GnssSvInfo_Ind_t Struct Reference

Data Fields

- uint8_t [altitudeAssumed](#)
- [loc_satelliteInfo](#) * [pSatelliteInfo](#)
- uint16_t [Tlvresult](#)

8.944.1 Detailed Description

Contain the parameters passed for SetLocGnssSvInfoCallback by the device.

Parameters

<i>altitudeAssumed</i>	<ul style="list-style-type: none"> Indicates whether altitude is assumed or calculated <ul style="list-style-type: none"> 0x00 (FALSE) - Valid altitude is calculated 0x01 (TRUE) - Valid altitude is assumed; there may not be enough satellites to determine precise altitude
<i>pSatelliteInfo</i>	<ul style="list-style-type: none"> See loc_satelliteInfo for more information.
<i>Tlvresult</i>	<ul style="list-style-type: none"> unpack result

8.944.2 Field Documentation

8.944.2.1 `uint8_t unpack_loc_GnssSvInfo_Ind_t::altitudeAssumed`

8.944.2.2 `loc_satelliteInfo* unpack_loc_GnssSvInfo_Ind_t::pSatelliteInfo`

8.944.2.3 `uint16_t unpack_loc_GnssSvInfo_Ind_t::Tlvresult`

8.945 unpack_loc_PositionRpt_Ind_t Struct Reference

Data Fields

- `uint32_t sessionStatus`
- `uint8_t sessionId`
- `uint64_t * pLatitude`
- `uint64_t * pLongitude`
- `uint32_t * pHorUncCircular`
- `uint32_t * pHorUncEllipseSemiMinor`
- `uint32_t * pHorUncEllipseSemiMajor`
- `uint32_t * pHorUncEllipseOrientAzimuth`
- `uint8_t * pHorConfidence`
- `uint32_t * pHorReliability`
- `uint32_t * pSpeedHorizontal`
- `uint32_t * pSpeedUnc`
- `uint32_t * pAltitudeWrtEllipsoid`
- `uint32_t * pAltitudeWrtMeanSeaLevel`
- `uint32_t * pVertUnc`
- `uint8_t * pVertConfidence`
- `uint32_t * pVertReliability`
- `uint32_t * pSpeedVertical`
- `uint32_t * pHeading`

- uint32_t * [pHeadingUnc](#)
- uint32_t * [pMagneticDeviation](#)
- uint32_t * [pTechnologyMask](#)
- [loc_precisionDilution](#) * [pPrecisionDilution](#)
- uint64_t * [pTimestampUtc](#)
- uint8_t * [pLeapSeconds](#)
- [loc_gpsTime](#) * [pGpsTime](#)
- uint32_t * [pTimeUnc](#)
- uint32_t * [pTimeSrc](#)
- [loc_sensorDataUsage](#) * [pSensorDataUsage](#)
- uint32_t * [pFixId](#)
- [loc_svUsedforFix](#) * [pSvUsedforFix](#)
- uint8_t * [pAltitudeAssumed](#)
- uint16_t [Tlvresult](#)

8.945.1 Detailed Description

This structure contains Event Position Report Indication unpack

Parameters

<i>sessionStatus</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Session was successful – 1 - Session is still in progress; further position reports will be generated until either the fix criteria specified by the client are met or the client response timeout occurs. – 2 - Session failed.. – 3 - Fix request failed because the session timed out. – 4 - Fix request failed because the session was ended by the user. – 5 - Fix request failed due to bad parameters in the request. – 6 - Fix request failed because the phone is offline. – 7 - Fix request failed because the engine is locked
<i>sessionId</i>	<ul style="list-style-type: none"> • ID of the session that was specified in the Start request • Range - 0 to 255
<i>pLatitude</i>	<ul style="list-style-type: none"> • Latitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -90.0 to 90.0 • Positive values indicate northern latitude • Negative values indicate southern latitude
<i>pLongitude</i>	<ul style="list-style-type: none"> • Longitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -180.0 to 180.0 • Positive values indicate eastern latitude • Negative values indicate western latitude

<i>pHorUncCircular</i>	<ul style="list-style-type: none"> • Horizontal position uncertainty. • Units - Meters
<i>pHorUnc</i> ↔ <i>EllipseSemi</i> ↔ <i>Minor</i>	<ul style="list-style-type: none"> • Semi-minor axis of horizontal elliptical uncertainty. • Units - Meters
<i>pHorUnc</i> ↔ <i>EllipseSemi</i> ↔ <i>Major</i>	<ul style="list-style-type: none"> • Semi-major axis of horizontal elliptical uncertainty. • Units: Meters
<i>pHorUnc</i> ↔ <i>EllipseOrient</i> ↔ <i>Azimuth</i>	<ul style="list-style-type: none"> • Elliptical horizontal uncertainty azimuth of orientation. • Units - Decimal degrees • Range - 0 to 180
<i>pHorConfidence</i>	<ul style="list-style-type: none"> • Horizontal uncertainty confidence. • If both elliptical and horizontal uncertainties are specified in this message, the confidence corresponds to the elliptical uncertainty. • Units - Percentage • Range 0-99
<i>pHorReliability</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk – 2 - Location reliability is low; little or no cross-checking is possible. – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed
<i>pSpeed</i> ↔ <i>Horizontal</i>	<ul style="list-style-type: none"> • Horizontal speed. • Units - Meters/second
<i>pSpeedUnc</i>	<ul style="list-style-type: none"> • 3-D Speed uncertainty. • Units - Meters/second.
<i>pAltitudeWrt</i> ↔ <i>Ellipsoid</i>	<ul style="list-style-type: none"> • Altitude With Respect to WGS84 Ellipsoid. • Units - Meters • Range -500 to 15883
<i>pAltitudeWrt</i> ↔ <i>MeanSeaLevel</i>	<ul style="list-style-type: none"> • Altitude With Respect to Sea Level. • Units - Meters

<i>pVertUnc</i>	<ul style="list-style-type: none"> • Vertical uncertainty. • Units - Meters
<i>pVertConfidence</i>	<ul style="list-style-type: none"> • Vertical uncertainty confidence. • Units - Percentage • Range 0 to 99
<i>pVertReliability</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk. – 2 - Location reliability is low; little or no cross-checking is possible – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed
<i>pSpeedVertical</i>	<ul style="list-style-type: none"> • Vertical speed. • Units - Meters/second
<i>pHeading</i>	<ul style="list-style-type: none"> • Heading. • Units - Degree • Range 0 to 359.999
<i>pHeadingUnc</i>	<ul style="list-style-type: none"> • Heading uncertainty. • Units - Degree • Range 0 to 359.999
<i>pMagnetic↔ Deviation</i>	<ul style="list-style-type: none"> • Difference between the bearing to true north and the bearing shown on a magnetic compass. The deviation is positive when the magnetic north is east of true north.
<i>pTechnology↔ Mask</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0x00000001 - Satellites were used to generate the fix – 0x00000002 - Cell towers were used to generate the fix – 0x00000004 - Wi-Fi access points were used to generate the fix – 0x00000008 - Sensors were used to generate the fix – 0x00000010 - Reference Location was used to generate the fix – 0x00000020 - Coarse position injected into the location engine was used to generate the fix – 0x00000040 - AFLT was used to generate the fix – 0x00000080 - GNSS and network-provided measurements were used to generate the fix
<i>-pPrecision↔ Dilution</i>	<ul style="list-style-type: none"> • See loc_precisionDilution for more information

<i>pTimestampUtc</i>	<ul style="list-style-type: none"> • UTC timestamp • Units - Milliseconds since Jan. 1, 1970
<i>pLeapSeconds</i>	<ul style="list-style-type: none"> • Leap second information. If leapSeconds is not available, timestampUtc is calculated based on a hard-coded value for leap seconds. • Units - Seconds
<i>-pGpsTime</i>	<ul style="list-style-type: none"> • See loc_gpsTime for more information
<i>pTimeUnc</i>	<ul style="list-style-type: none"> • Time uncertainty. • Units - Milliseconds
<i>pTimeSrc</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - Invalid time. – 1 - Time is set by the 1X system. – 2 - Time is set by WCDMA/GSM time tagging. – 3 - Time is set by an external injection. – 4 - Time is set after decoding over-the-air GPS navigation data from one GPS satellite. – 5 - Time is set after decoding over-the-air GPS navigation data from multiple satellites. – 6 - Both time of the week and the GPS week number are known. – 7 - Time is set by the position engine after the fix is obtained – 8 - Time is set by the position engine after performing SFT, this is done when the clock time uncertainty is large. – 9 - Time is set after decoding GLO satellites. – 10- Time is set after transforming the GPS to GLO time – 11- Time is set by the sleep time tag provided by the WCDMA network. – 12- Time is set by the sleep time tag provided by the GSM network – 13- Source of the time is unknown – 14- Time is derived from the system clock (better known as the slow clock); GNSS time is maintained irrespective of the GNSS receiver state – 15- Time is set after decoding QZSS satellites. – 16- Time is set after decoding BDS satellites.
<i>-pSensorData↔ Usage</i>	<ul style="list-style-type: none"> • See loc_sensorDataUsage for more information
<i>pFixId</i>	<ul style="list-style-type: none"> • Fix count for the session. Starts with 0 and increments by one for each successive position report for a particular session.
<i>-pSvUsedforFix</i>	<ul style="list-style-type: none"> • See loc_svUsedforFix for more information
<i>pAltitude↔ Assumed</i>	<ul style="list-style-type: none"> • Indicates whether altitude is assumed or calculated.

- Value
 - 0x00 - Altitude is calculated
 - 0x01 - Altitude is assumed

8.945.2 Field Documentation

8.945.2.1 `uint8_t* unpack_loc_PositionRpt_Ind_t::pAltitudeAssumed`

8.945.2.2 `uint32_t* unpack_loc_PositionRpt_Ind_t::pAltitudeWrtEllipsoid`

8.945.2.3 `uint32_t* unpack_loc_PositionRpt_Ind_t::pAltitudeWrtMeanSeaLevel`

8.945.2.4 `uint32_t* unpack_loc_PositionRpt_Ind_t::pFixId`

8.945.2.5 `loc_gpsTime* unpack_loc_PositionRpt_Ind_t::pGpsTime`

8.945.2.6 `uint32_t* unpack_loc_PositionRpt_Ind_t::pHeading`

8.945.2.7 `uint32_t* unpack_loc_PositionRpt_Ind_t::pHeadingUnc`

8.945.2.8 `uint8_t* unpack_loc_PositionRpt_Ind_t::pHorConfidence`

8.945.2.9 `uint32_t* unpack_loc_PositionRpt_Ind_t::pHorReliability`

8.945.2.10 `uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncCircular`

8.945.2.11 `uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseOrientAzimuth`

8.945.2.12 `uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseSemiMajor`

8.945.2.13 `uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseSemiMinor`

8.945.2.14 `uint64_t* unpack_loc_PositionRpt_Ind_t::pLatitude`

8.945.2.15 `uint8_t* unpack_loc_PositionRpt_Ind_t::pLeapSeconds`

8.945.2.16 `uint64_t* unpack_loc_PositionRpt_Ind_t::pLongitude`

8.945.2.17 `uint32_t* unpack_loc_PositionRpt_Ind_t::pMagneticDeviation`

8.945.2.18 `loc_precisionDilution* unpack_loc_PositionRpt_Ind_t::pPrecisionDilution`

8.945.2.19 `loc_sensorDataUsage* unpack_loc_PositionRpt_Ind_t::pSensorDataUsage`

- 8.945.2.20 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedHorizontal
- 8.945.2.21 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedUnc
- 8.945.2.22 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedVertical
- 8.945.2.23 loc_svUsedforFix* unpack_loc_PositionRpt_Ind_t::pSvUsedforFix
- 8.945.2.24 uint32_t* unpack_loc_PositionRpt_Ind_t::pTechnologyMask
- 8.945.2.25 uint32_t* unpack_loc_PositionRpt_Ind_t::pTimeSrc
- 8.945.2.26 uint64_t* unpack_loc_PositionRpt_Ind_t::pTimestampUtc
- 8.945.2.27 uint32_t* unpack_loc_PositionRpt_Ind_t::pTimeUnc
- 8.945.2.28 uint8_t* unpack_loc_PositionRpt_Ind_t::pVertConfidence
- 8.945.2.29 uint32_t* unpack_loc_PositionRpt_Ind_t::pVertReliability
- 8.945.2.30 uint32_t* unpack_loc_PositionRpt_Ind_t::pVertUnc
- 8.945.2.31 uint8_t unpack_loc_PositionRpt_Ind_t::sessionId
- 8.945.2.32 uint32_t unpack_loc_PositionRpt_Ind_t::sessionStatus
- 8.945.2.33 uint16_t unpack_loc_PositionRpt_Ind_t::Tlvresult

8.946 unpack_loc_SetExtPowerConfig_Ind_t Struct Reference

Data Fields

- uint32_t [status](#)
- uint16_t [Tlvresult](#)

8.946.1 Detailed Description

This structure contains LOC Set External Power Configure status field.

Parameters

<i>status</i>	<ul style="list-style-type: none"> Valid values <ul style="list-style-type: none"> 0 - Request was completed successfully 1 - Request failed because of a general failure. 2 - Request failed because it is not supported. 3 - Request failed because it contained invalid parameters 4 - Request failed because the engine is busy 5 - Request failed because the phone is offline 6 - Request failed because it timed out 7 - Request failed because an undefined configuration was requested 8 - engine could not allocate sufficient memory 9 - Request failed because the maximum number of Geofences are already programmed 10 - Location service failed because of an XTRA version-based file format check failure
<i>Tlvresult</i>	<ul style="list-style-type: none"> unpack result

8.946.2 Field Documentation

8.946.2.1 uint32_t unpack_loc_SetExtPowerConfig_Ind_t::status

8.946.2.2 uint16_t unpack_loc_SetExtPowerConfig_Ind_t::Tlvresult

8.947 unpack_loc_SetExtPowerState_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.947.1 Detailed Description

This structure contains Set Ext Power State unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none"> Unpack result.
------------------	--

8.947.2 Field Documentation

8.947.2.1 uint16_t unpack_loc_SetExtPowerState_t::Tlvresult

8.948 unpack_loc_SetOperationMode_Ind_t Struct Reference

Data Fields

- uint32_t [status](#)
- uint16_t [Tlvresult](#)

8.948.1 Detailed Description

This structure contains LOC Set External Power Configure status field.

Parameters

<i>status</i>	<ul style="list-style-type: none">• Status of the Set Operation Mode request.• Valid values:<ul style="list-style-type: none">– 0 - Request was completed successfully– 1 - Request failed because of a general failure– 2 - Request failed because it is not supported– 3 - Request failed because it contained invalid parameters– 4 - Request failed because the engine is busy– 5 - Request failed because the phone is offline– 6 - Request failed because it timed out– 7 - Request failed because an undefined configuration was requested– 8 - Request failed because the engine could not allocate sufficient memory for the request– 9 - Request failed because the maximum number of Geofences are already programmed– 10 - Location service failed because of an XTRA version-based file format check failure
<i>Tlvresult</i>	<ul style="list-style-type: none">• unpack result

8.948.2 Field Documentation

8.948.2.1 uint32_t unpack_loc_SetOperationMode_Ind_t::status

8.948.2.2 uint16_t unpack_loc_SetOperationMode_Ind_t::Tlvresult

8.949 unpack_loc_SetOperationMode_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.949.1 Detailed Description

This structure contains Set Operation Mode unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">Unpack result.
------------------	--

8.949.2 Field Documentation

8.949.2.1 uint16_t unpack_loc_SetOperationMode_t::Tlvresult

8.950 unpack_loc_SLQSLOCGetBestAvailPos_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.950.1 Detailed Description

This structure contains Set Operation Mode unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">Unpack result.
------------------	--

8.950.2 Field Documentation

8.950.2.1 uint16_t unpack_loc_SLQSLOCGetBestAvailPos_t::Tlvresult

8.951 unpack_loc_Start_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.951.1 Detailed Description

This structure contains Start LOC unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">Unpack result.
------------------	--

8.951.2 Field Documentation

8.951.2.1 uint16_t unpack_loc_Start_t::Tlvresult

8.952 unpack_loc_Stop_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.952.1 Detailed Description

This structure contains Stop LOC unpack

Parameters

<i>Tlvresult</i>	<ul style="list-style-type: none">Unpack result.
------------------	--

8.952.2 Field Documentation

8.952.2.1 uint16_t unpack_loc_Stop_t::Tlvresult

8.953 unpack_nas_GetCDMANetworkParameters_t Struct Reference

Data Fields

- uint8_t [SCI](#)
- uint8_t [SCM](#)
- uint8_t [RegHomeSID](#)
- uint8_t [RegForeignSID](#)
- uint8_t [RegForeignNID](#)
- uint8_t [ForceRev0](#)
- uint8_t [CustomSCP](#)
- uint32_t [Protocol](#)
- uint32_t [Broadcast](#)
- uint32_t [Application](#)
- uint32_t [Roaming](#)

8.953.1 Detailed Description

Parameters

<i>SCI</i>	slot cycle index
<i>SCM</i>	station class mark
<i>RegHomeSID</i>	register on home sid
<i>RegForeignSID</i>	register on foreign sid
<i>RegForeignNID</i>	register on foreign nid
<i>ForceRev0</i>	force header revision
<i>CustomSCP</i>	custom SCP
<i>Protocol</i>	protocol
<i>Booadcast</i>	broadcast
<i>Application</i>	application
<i>Roaming</i>	roaming

8.953.2 Field Documentation

8.953.2.1 `uint32_t unpack_nas_GetCDMANetworkParameters_t::Application`

8.953.2.2 `uint32_t unpack_nas_GetCDMANetworkParameters_t::Broadcast`

8.953.2.3 `uint8_t unpack_nas_GetCDMANetworkParameters_t::CustomSCP`

8.953.2.4 `uint8_t unpack_nas_GetCDMANetworkParameters_t::ForceRev0`

8.953.2.5 `uint32_t unpack_nas_GetCDMANetworkParameters_t::Protocol`

8.953.2.6 `uint8_t unpack_nas_GetCDMANetworkParameters_t::RegForeignNID`

8.953.2.7 `uint8_t unpack_nas_GetCDMANetworkParameters_t::RegForeignSID`

8.953.2.8 `uint8_t unpack_nas_GetCDMANetworkParameters_t::RegHomeSID`

8.953.2.9 `uint32_t unpack_nas_GetCDMANetworkParameters_t::Roaming`

8.953.2.10 `uint8_t unpack_nas_GetCDMANetworkParameters_t::SCI`

8.953.2.11 `uint8_t unpack_nas_GetCDMANetworkParameters_t::SCM`

8.954 `unpack_nas_GetHomeNetwork_t` Struct Reference

Data Fields

- `uint16_t mcc`
- `uint16_t mnc`
- `char name [255]`
- `uint16_t sid`
- `uint16_t nid`

8.954.1 Detailed Description

Parameters

<i>mcc</i>	mobile country code
<i>mnc</i>	mobile network code
<i>name</i>	network name or description
<i>sid</i>	home network system id only applies to cdma2000
<i>nid</i>	home network id Only applies to cdma2000

8.954.2 Field Documentation

8.954.2.1 uint16_t unpack_nas_GetHomeNetwork_t::mcc

8.954.2.2 uint16_t unpack_nas_GetHomeNetwork_t::mnc

8.954.2.3 char unpack_nas_GetHomeNetwork_t::name[255]

8.954.2.4 uint16_t unpack_nas_GetHomeNetwork_t::nid

8.954.2.5 uint16_t unpack_nas_GetHomeNetwork_t::sid

8.955 unpack_nas_GetNetworkPreference_t Struct Reference

Data Fields

- uint32_t [ActiveTechPref](#)
- uint32_t [Duration](#)
- uint32_t [PersistentTechPref](#)
- uint16_t [Tlvresult](#)

8.955.1 Detailed Description

Parameters

<i>Technology</i> ↔ <i>Pref[OUT]</i>	<ul style="list-style-type: none">• Bitmask representing the radio technology preference set.• No bits set indicates to the device to automatically determine the technology to use• Values:<ul style="list-style-type: none">– Bit 0 - Technology is 3GPP2– Bit 1 - Technology is 3GPP• Any combination of the following may be returned:<ul style="list-style-type: none">– Bit 2 - Analog - AMPS if 3GPP2, GSM if 3GPP– Bit 3 - Digital - CDMA if 3GPP2, WCDMA if 3GPP– Bit 4 - HDR– Bit 5 - LTE– Bits 6 to 15 - Reserved
---	--

<i>Duration</i> [OUT]	<ul style="list-style-type: none"> Duration of active preference <ul style="list-style-type: none"> 0 - Permanent 1 - Power cycle 2 - Until the end of the next call or a power cycle 3 - Until the end of the next call, a specified time, or a power cycle 4 to 6 - Until the end of the next call
<i>Persistent</i> ↔ <i>Technology</i> ↔ <i>Pref</i> [OUT]	<ul style="list-style-type: none"> Bit field representing persistent radio technology preference <ul style="list-style-type: none"> Same representation as the pTechnologyPref parameter
<i>Tlvresult</i>	<ul style="list-style-type: none"> unpack result

8.955.2 Field Documentation

8.955.2.1 uint32_t unpack_nas_GetNetworkPreference_t::ActiveTechPref

8.955.2.2 uint32_t unpack_nas_GetNetworkPreference_t::Duration

8.955.2.3 uint32_t unpack_nas_GetNetworkPreference_t::PersistentTechPref

8.955.2.4 uint16_t unpack_nas_GetNetworkPreference_t::Tlvresult

8.956 unpack_nas_GetRFInfo_t Struct Reference

Data Fields

- uint8_t [instancesSize](#)
- [RFBandInfo](#)↔
[Elements](#) [RFBandInfoElements](#) [255]

8.956.1 Detailed Description

Parameters

<i>instancesSize</i>	number of elements in RF info instances array.
RFBandInfo ↔ Elements	RF info instances array

8.956.2 Field Documentation

8.956.2.1 uint8_t unpack_nas_GetRFInfo_t::instancesSize

8.956.2.2 RFBandInfoElements unpack_nas_GetRFInfo_t::RFBandInfoElements[255]

8.957 unpack_nas_GetServingNetwork_t Struct Reference

Data Fields

- uint32_t [RegistrationState](#)
- uint32_t [CSDomain](#)
- uint32_t [PSDomain](#)
- uint32_t [RAN](#)
- uint8_t [RadiolfacesSize](#)
- uint8_t [Radiolfaces](#) [255]
- uint32_t [Roaming](#)
- uint16_t [MCC](#)
- uint16_t [MNC](#)
- uint8_t [nameSize](#)
- uint8_t [Name](#) [255]
- uint8_t [DataCapsLen](#)
- uint8_t [DataCaps](#) [255]

8.957.1 Detailed Description

Parameters

<i>Registration↔ State</i>	registration state
<i>CSDomain</i>	CS domain
<i>PSDomain</i>	PS domain
<i>RAN</i>	radio access network
<i>RadiolfacesSize</i>	radio interface size
<i>Radiolfaces</i>	radio interface list
<i>Roaming</i>	romaing indicator
<i>MCC</i>	Mobile country code
<i>MNC</i>	Mobile network code
<i>nameSize</i>	network name size
<i>Name</i>	network name
<i>DataCapsLen</i>	data capabilities len
<i>DataCap</i>	data capabilities

8.957.2 Field Documentation

8.957.2.1 uint32_t unpack_nas_GetServingNetwork_t::CSDomain

8.957.2.2 uint8_t unpack_nas_GetServingNetwork_t::DataCaps[255]

8.957.2.3 uint8_t unpack_nas_GetServingNetwork_t::DataCapsLen

8.957.2.4 uint16_t unpack_nas_GetServingNetwork_t::MCC

- 8.957.2.5 `uint16_t unpack_nas_GetServingNetwork_t::MNC`
- 8.957.2.6 `uint8_t unpack_nas_GetServingNetwork_t::Name[255]`
- 8.957.2.7 `uint8_t unpack_nas_GetServingNetwork_t::nameSize`
- 8.957.2.8 `uint32_t unpack_nas_GetServingNetwork_t::PSDomain`
- 8.957.2.9 `uint8_t unpack_nas_GetServingNetwork_t::Radiolfaces[255]`
- 8.957.2.10 `uint8_t unpack_nas_GetServingNetwork_t::RadiolfacesSize`
- 8.957.2.11 `uint32_t unpack_nas_GetServingNetwork_t::RAN`
- 8.957.2.12 `uint32_t unpack_nas_GetServingNetwork_t::RegistrationState`
- 8.957.2.13 `uint32_t unpack_nas_GetServingNetwork_t::Roaming`

8.958 `unpack_nas_GetServingNetworkCapabilities_t` Struct Reference

Data Fields

- `uint8_t DataCapsLen`
- `uint8_t DataCaps [255]`

8.958.1 Detailed Description

Parameters

<i>DataCapsLen</i>	data capabilities len
<i>DataCap</i>	data capabilities

8.958.2 Field Documentation

- 8.958.2.1 `uint8_t unpack_nas_GetServingNetworkCapabilities_t::DataCaps[255]`
- 8.958.2.2 `uint8_t unpack_nas_GetServingNetworkCapabilities_t::DataCapsLen`

8.959 `unpack_nas_GetSignalStrengths_t` Struct Reference

Data Fields

- `uint32_t len`
- signed char `rssi` [8]
- `uint32_t radio` [8]

8.959.1 Detailed Description

Parameters

<i>len</i>	number of rssi & radio items following
<i>rssi</i>	signal strength array
<i>radio</i>	radio interface array

8.959.2 Field Documentation

8.959.2.1 `uint32_t unpack_nas_GetSignalStrengths_t::len`

8.959.2.2 `uint32_t unpack_nas_GetSignalStrengths_t::radio[8]`

8.959.2.3 `signed char unpack_nas_GetSignalStrengths_t::rssi[8]`

8.960 unpack_nas_PerformNetworkScan_t Struct Reference

Data Fields

- `uint8_t * p3GppNetworkInstanceSize`
- `nas_QmiNas3GppNetworkInfo * p3GppNetworkInfoInstances`
- `uint8_t * pRATInstanceSize`
- `nas_QmiNas3GppNetworkRAT * pRATInstance`
- `uint8_t * pPCSInstanceSize`
- `nas_QmisNasPcsDigit * pPCSInstance`
- `uint32_t * pScanResult`

8.960.1 Detailed Description

Parameters

<i>InstanceSize</i>	total instances
<i>Instances</i>	info for instances

8.960.2 Field Documentation

8.960.2.1 `nas_QmiNas3GppNetworkInfo* unpack_nas_PerformNetworkScan_t::p3GppNetworkInfoInstances`

8.960.2.2 `uint8_t* unpack_nas_PerformNetworkScan_t::p3GppNetworkInstanceSize`

8.960.2.3 `nas_QmisNasPcsDigit* unpack_nas_PerformNetworkScan_t::pPCSInstance`

8.960.2.4 `uint8_t* unpack_nas_PerformNetworkScan_t::pPCSInstanceSize`

8.960.2.5 `nas_QmiNas3GppNetworkRAT* unpack_nas_PerformNetworkScan_t::pRATInstance`

8.960.2.6 `uint8_t*` `unpack_nas_PerformNetworkScan_t::pRATInstanceSize`

8.960.2.7 `uint32_t*` `unpack_nas_PerformNetworkScan_t::pScanResult`

8.961 `unpack_nas_SetDataCapabilitiesCallback_ind_t` Struct Reference

Data Fields

- `uint8_t` [dataCapsSize](#)
- `uint8_t` [dataCaps](#) [255]

8.961.1 Detailed Description

Parameters

<i>dataCapsSize</i>	Number of Data Capabilities
<i>dataCaps</i>	Data Capabilities

8.961.2 Field Documentation

8.961.2.1 `uint8_t` `unpack_nas_SetDataCapabilitiesCallback_ind_t::dataCaps[255]`

8.961.2.2 `uint8_t` `unpack_nas_SetDataCapabilitiesCallback_ind_t::dataCapsSize`

8.962 `unpack_nas_SetEventReportInd_t` Struct Reference

Data Fields

- [nas_SignalStrengthTlv](#) SSTlv
- [nas_RFInfoTlv](#) RFTlv
- [nas_RejectReasonTlv](#) RRTlv
- [nas_SLQSSignalStrengthsTlv](#) SLQSSSTlv

8.962.1 Detailed Description

Parameters

<i>SSTlv</i>	signal strength tlv
<i>RFTlv</i>	RF tlv
<i>RRTlv</i>	RR tlv
<i>SLQSSSTlv</i>	signal strength complete info tlv

8.962.2 Field Documentation

8.962.2.1 `nas_RFInfoTlv` `unpack_nas_SetEventReportInd_t::RFTlv`

8.962.2.2 nas_RejectReasonTlv unpack_nas_SetEventReportInd_t::RRTlv

8.962.2.3 nas_SLQSSignalStrengthsTlv unpack_nas_SetEventReportInd_t::SLQSSSTlv

8.962.2.4 nas_SignalStrengthTlv unpack_nas_SetEventReportInd_t::SSTlv

8.963 unpack_nas_SetNasLTECphyCalndCallback_ind_t Struct Reference

Data Fields

- [nas_PhyCaAggScellIndType](#) sPhyCaAggScellIndType
- [nas_PhyCaAggScellDIBw](#) sPhyCaAggScellDIBw
- [nas_PhyCaAggScellInfo](#) sPhyCaAggScellInfo
- [nas_PhyCaAggPcellInfo](#) sPhyCaAggPcellInfo
- [nas_PhyCaAggScellIndex](#) sPhyCaAggScellIndex

8.963.1 Detailed Description

Structure for storing the LTEC PHY CA indication parameters.

Parameters

<i>pPhyCaAggScellIndType</i>	<ul style="list-style-type: none"> • See nas_PhyCaAggScellIndType for more information.
<i>sPhyCaAggScellDIBw</i>	<ul style="list-style-type: none"> • See nas_PhyCaAggScellDIBw for more information.
<i>sPhyCaAggScellInfo</i>	<ul style="list-style-type: none"> • See nas_PhyCaAggScellInfo for more information.
<i>sPhyCaAggPcellInfo</i>	<ul style="list-style-type: none"> • See nas_PhyCaAggPcellInfo for more information.
<i>sPhyCaAggScellIndex</i>	<ul style="list-style-type: none"> • See nas_PhyCaAggScellIndex for more information.

8.963.2 Field Documentation

8.963.2.1 nas_PhyCaAggPcellInfo unpack_nas_SetNasLTECphyCalndCallback_ind_t::sPhyCaAggPcellInfo

8.963.2.2 nas_PhyCaAggScellDIBw unpack_nas_SetNasLTECphyCalndCallback_ind_t::sPhyCaAggScellDIBw

8.963.2.3 nas_PhyCaAggScellIndex unpack_nas_SetNasLTECphyCalndCallback_ind_t::sPhyCaAggScellIndex

8.963.2.4 nas_PhyCaAggScellIndType unpack_nas_SetNasLTECphyCalndCallback_ind_t::sPhyCaAggScellIndType

8.963.2.5 nas_PhyCaAggScellInfo unpack_nas_SetNasLTECphyCalndCallback_ind_t::sPhyCaAggScellInfo

8.964 unpack_nas_SetNetworkPreference_t Struct Reference

Data Fields

- uint16_t [Tlvresult](#)

8.964.1 Detailed Description

Parameters

<i>Technology</i> ↔ <i>Pref[OUT]</i>	<ul style="list-style-type: none"> • Bitmask representing the radio technology preference set. • No bits set indicates to the device to automatically determine the technology to use • Values: <ul style="list-style-type: none"> – Bit 0 - Technology is 3GPP2 – Bit 1 - Technology is 3GPP • Any combination of the following may be returned: <ul style="list-style-type: none"> – Bit 2 - Analog - AMPS if 3GPP2, GSM if 3GPP – Bit 3 - Digital - CDMA if 3GPP2, WCDMA if 3GPP – Bit 4 - HDR – Bit 5 - LTE – Bits 6 to 15 - Reserved
<i>Duration[OUT]</i>	<ul style="list-style-type: none"> • Duration of active preference <ul style="list-style-type: none"> – 0 - Permanent – 1 - Power cycle – 2 - Until the end of the next call or a power cycle – 3 - Until the end of the next call, a specified time, or a power cycle – 4 to 6 - Until the end of the next call
<i>Persistent</i> ↔ <i>Technology</i> ↔ <i>Pref[OUT]</i>	<ul style="list-style-type: none"> • Bit field representing persistent radio technology preference <ul style="list-style-type: none"> – Same representation as the pTechnologyPref parameter
<i>Tlvresult</i>	<ul style="list-style-type: none"> • unpack result

8.964.2 Field Documentation

8.964.2.1 uint16_t unpack_nas_SetNetworkPreference_t::Tlvresult

8.965 unpack_nas_SetRoamingIndicatorCallback_ind_t Struct Reference

Data Fields

- uint8_t [roaming](#)

8.965.1 Detailed Description

Parameters

<i>roaming</i>	<ul style="list-style-type: none">• Roaming Indication<ul style="list-style-type: none">– 0 - Roaming– 1 - Home– 2 - Roaming partner– >2 - Operator defined values
----------------	--

8.965.2 Field Documentation

8.965.2.1 uint8_t unpack_nas_SetRoamingIndicatorCallback_ind_t::roaming

8.966 unpack_nas_SetServingSystemCallback_ind_t Struct Reference

Data Fields

- [NAServingSystemInfo](#) SSInfo
- uint16_t Tlvresult

8.966.1 Detailed Description

Parameters

<i>SSInfo</i>	<ul style="list-style-type: none">• Serving system parameters information<ul style="list-style-type: none">– See NAServingSystemInfo for more details
<i>Tlvresult</i>	<ul style="list-style-type: none">• unpack result

8.966.2 Field Documentation

8.966.2.1 NAServingSystemInfo unpack_nas_SetServingSystemCallback_ind_t::SSInfo

8.966.2.2 uint16_t unpack_nas_SetServingSystemCallback_ind_t::Tlvresult

8.967 unpack_nas_SlqsGetLTECphyCAInfo_t Struct Reference

Data Fields

- [NasGetLTECphyCaInfo](#) [LTECphyCaInfo](#)
- [uint16_t](#) [Tlvresult](#)

8.967.1 Detailed Description

Parameters

<i>LTECphyCa</i>	<ul style="list-style-type: none"> • Carrier aggregation event information <ul style="list-style-type: none"> – See NasGetLTECphyCaInfo for more details
<i>Tlvresult</i>	<ul style="list-style-type: none"> • unpack result

8.967.2 Field Documentation

8.967.2.1 [NasGetLTECphyCaInfo](#) [unpack_nas_SlqsGetLTECphyCAInfo_t::LTECphyCaInfo](#)

8.967.2.2 [uint16_t](#) [unpack_nas_SlqsGetLTECphyCAInfo_t::Tlvresult](#)

8.968 [unpack_nas_SLQSGetNetworkTime_t](#) Struct Reference

Data Fields

- [nas_timeInfo](#) * [p3GPP2TimeInfo](#)
- [nas_timeInfo](#) * [p3GPPTimeInfo](#)

8.968.1 Detailed Description

This structure contains information about the GetNetworkTime response parameters.

Parameters

<i>p3GPP2Time↔ Info</i>	[Optional] <ul style="list-style-type: none"> • See nas_timeInfo for more information
<i>p3GPPTimeInfo</i>	[Optional] <ul style="list-style-type: none"> • See nas_timeInfo for more information

8.968.2 Field Documentation

8.968.2.1 [nas_timeInfo](#)* [unpack_nas_SLQSGetNetworkTime_t::p3GPP2TimeInfo](#)

8.968.2.2 nas_timeInfo* unpack_nas_SLQSGetNetworkTime_t::p3GPPTIMEInfo

8.969 unpack_nas_SLQSGetPLMNName_t Struct Reference

Data Fields

- uint8_t [spnEncoding](#)
- uint8_t [spnLength](#)
- char [spn](#) [255]
- uint8_t [shortNameEn](#)
- uint8_t [shortNameCI](#)
- uint8_t [shortNameSB](#)
- char [shortNameLen](#)
- uint8_t [shortName](#) [255]
- uint8_t [longNameEn](#)
- uint8_t [longNameCI](#)
- uint8_t [longNameSB](#)
- uint8_t [longNameLen](#)
- char [longName](#) [255]

8.969.1 Field Documentation

8.969.1.1 char unpack_nas_SLQSGetPLMNName_t::longName[255]

8.969.1.2 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameCI

8.969.1.3 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameEn

8.969.1.4 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameLen

8.969.1.5 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameSB

8.969.1.6 uint8_t unpack_nas_SLQSGetPLMNName_t::shortName[255]

8.969.1.7 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameCI

8.969.1.8 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameEn

8.969.1.9 char unpack_nas_SLQSGetPLMNName_t::shortNameLen

8.969.1.10 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameSB

8.969.1.11 char unpack_nas_SLQSGetPLMNName_t::spn[255]

8.969.1.12 uint8_t unpack_nas_SLQSGetPLMNName_t::spnEncoding

8.969.1.13 uint8_t unpack_nas_SLQSGetPLMNName_t::spnLength

8.970 unpack_nas_SLQSGetServingSystem_t Struct Reference

Data Fields

- [nas_servSystem](#) ServingSystem

- uint8_t [RoamIndicatorVal](#)
- nas_dataSrvCapabilities [DataSrvCapabilities](#)
- nas_currentPLMN [CurrentPLMN](#)
- uint16_t [SystemID](#)
- uint16_t [NetworkID](#)
- uint16_t [BasestationID](#)
- uint32_t [BasestationLatitude](#)
- uint32_t [BasestationLongitude](#)
- nas_roamIndList [RoamingIndicatorList](#)
- uint8_t [DefaultRoamInd](#)
- nas_qaQmi3Gpp2TimeZone [Gpp2TimeZone](#)
- uint8_t [CDMA_P_Rev](#)
- uint8_t [GppTimeZone](#)
- uint8_t [GppNetworkDSTAdjustment](#)
- uint16_t [Lac](#)
- uint32_t [CellID](#)
- uint8_t [ConcSvcInfo](#)
- uint8_t [PRLInd](#)
- uint8_t [DTMInd](#)
- nas_detailSvcInfo [DetailedSvcInfo](#)
- nas_CDMA SysInfoExt [CDMASystemInfoExt](#)
- uint8_t [HdrPersonality](#)
- uint16_t [TrackAreaCode](#)
- nas_callBarStatus [CallBarStatus](#)

8.970.1 Detailed Description

Parameters

<i>ServingSystem</i>	serving system info
<i>RoamIndicator</i> ↔ <i>Val</i>	roaming indicator value
<i>DataSrv</i> ↔ <i>Capabilities</i>	data servcie capabilities
<i>CurrentPLMN</i>	current PLMN info
<i>SystemID</i>	system id
<i>NetworkID</i>	network id
<i>BasestationID</i>	base station id
<i>Basestation</i> ↔ <i>Latitude</i>	base station latitude
<i>Basestation</i> ↔ <i>Longitude</i>	base station longitude
<i>Roaming</i> ↔ <i>IndicatorList</i>	roaming indicator list
<i>DefaultRoamInd</i>	default roaming indicator
<i>3Gpp2TimeZone</i>	3Gpp2 time zone
<i>pCDMA_P_Rev</i>	cdma P_Rev in use
<i>3GppTimeZone</i>	3Gpp time zone
<i>GppNetworkD</i> ↔ <i>STAdjustment</i>	3GPP network daylight saving adjustment
<i>Lac</i>	location area code
<i>CellID</i>	3GPP cell id
<i>ConcSvcInfo</i>	3GPP2 concurrent servcie info
<i>PRLInd</i>	3GPP2 PRL indicator
<i>DTMInd</i>	DTM indicator(GSM)

<i>DetailedSvcInfo</i>	detail service info
<i>CDMASystemInfoExt</i>	extra cdma system info
<i>HdrPersonality</i>	hdr personality
<i>TrackAreaCode</i>	track area code
<i>CallBarStatus</i>	call barring status

8.970.2 Field Documentation

8.970.2.1 `uint16_t unpack_nas_SLQSGetServingSystem_t::BasestationID`

8.970.2.2 `uint32_t unpack_nas_SLQSGetServingSystem_t::BasestationLatitude`

8.970.2.3 `uint32_t unpack_nas_SLQSGetServingSystem_t::BasestationLongitude`

8.970.2.4 `nas_callBarStatus unpack_nas_SLQSGetServingSystem_t::CallBarStatus`

8.970.2.5 `uint8_t unpack_nas_SLQSGetServingSystem_t::CDMA_P_Rev`

8.970.2.6 `nas_CDMASysInfoExt unpack_nas_SLQSGetServingSystem_t::CDMASystemInfoExt`

8.970.2.7 `uint32_t unpack_nas_SLQSGetServingSystem_t::CellID`

8.970.2.8 `uint8_t unpack_nas_SLQSGetServingSystem_t::ConcSvcInfo`

8.970.2.9 `nas_currentPLMN unpack_nas_SLQSGetServingSystem_t::CurrentPLMN`

8.970.2.10 `nas_dataSrvCapabilities unpack_nas_SLQSGetServingSystem_t::DataSrvCapabilities`

8.970.2.11 `uint8_t unpack_nas_SLQSGetServingSystem_t::DefaultRoamInd`

8.970.2.12 `nas_detailSvcInfo unpack_nas_SLQSGetServingSystem_t::DetailedSvcInfo`

8.970.2.13 `uint8_t unpack_nas_SLQSGetServingSystem_t::DTMInd`

8.970.2.14 `nas_qaQmi3Gpp2TimeZone unpack_nas_SLQSGetServingSystem_t::Gpp2TimeZone`

8.970.2.15 `uint8_t unpack_nas_SLQSGetServingSystem_t::GppNetworkDSTAdjustment`

8.970.2.16 `uint8_t unpack_nas_SLQSGetServingSystem_t::GppTimeZone`

8.970.2.17 `uint8_t unpack_nas_SLQSGetServingSystem_t::HdrPersonality`

8.970.2.18 `uint16_t unpack_nas_SLQSGetServingSystem_t::Lac`

8.970.2.19 uint16_t unpack_nas_SLQSGetservingSystem_t::NetworkID

8.970.2.20 uint8_t unpack_nas_SLQSGetservingSystem_t::PRLInd

8.970.2.21 uint8_t unpack_nas_SLQSGetservingSystem_t::RoamIndicatorVal

8.970.2.22 nas_roamIndList unpack_nas_SLQSGetservingSystem_t::RoamingIndicatorList

8.970.2.23 nas_servSystem unpack_nas_SLQSGetservingSystem_t::ServingSystem

8.970.2.24 uint16_t unpack_nas_SLQSGetservingSystem_t::SystemID

8.970.2.25 uint16_t unpack_nas_SLQSGetservingSystem_t::TrackAreaCode

8.971 unpack_nas_SLQSGetSignalStrength_t Struct Reference

Data Fields

- uint16_t [signalStrengthReqMask](#)
- uint16_t [rxSignalStrengthListLen](#)
- [nas_rxSignalStrengthListElement](#) [rxSignalStrengthList](#) [18]
- uint16_t [ecioListLen](#)
- [nas_ecioListElement](#) [ecioList](#) [18]
- int32_t [Io](#)
- uint8_t [sinr](#)
- uint16_t [errorRateListLen](#)
- [nas_errorRateListElement](#) [errorRateList](#) [18]
- [nas_rsrqInformation](#) [rsrqInfo](#)
- int16_t [lteSnr](#)
- int16_t [lteSrp](#)

8.971.1 Detailed Description

Parameters

<i>rxSignalStrengthListLen</i>	number of elements in Receive Signal Strength List
<i>rxSignalStrengthList</i>	signal strength list
<i>ecioListLen</i>	number of elements in ECIO List
<i>ecioList</i>	ecio list
<i>Io</i>	received Io in dBm; IO is only applicable for 1xEV-DO
<i>sinr</i>	SINR level; SINR is only applicable for 1xEV-DO
<i>errorRateListLen</i>	number of elements in Error Rate List
<i>errorRateList</i>	error rate list
<i>rsrqInfo</i>	rsrq info
<i>lteSnr</i>	lte snr info
<i>lteSrp</i>	lte srp info

8.971.2 Field Documentation

- 8.971.2.1 nas_ecioListElement unpack_nas_SLQSGetSignalStrength_t::ecioList[18]
- 8.971.2.2 uint16_t unpack_nas_SLQSGetSignalStrength_t::ecioListLen
- 8.971.2.3 nas_errorRateListElement unpack_nas_SLQSGetSignalStrength_t::errorRateList[18]
- 8.971.2.4 uint16_t unpack_nas_SLQSGetSignalStrength_t::errorRateListLen
- 8.971.2.5 int32_t unpack_nas_SLQSGetSignalStrength_t::lo
- 8.971.2.6 int16_t unpack_nas_SLQSGetSignalStrength_t::ltsrpr
- 8.971.2.7 int16_t unpack_nas_SLQSGetSignalStrength_t::ltsnr
- 8.971.2.8 nas_rsrqInformation unpack_nas_SLQSGetSignalStrength_t::rsrqInfo
- 8.971.2.9 nas_rxSignalStrengthListElement unpack_nas_SLQSGetSignalStrength_t::rxSignalStrengthList[18]
- 8.971.2.10 uint16_t unpack_nas_SLQSGetSignalStrength_t::rxSignalStrengthListLen
- 8.971.2.11 uint16_t unpack_nas_SLQSGetSignalStrength_t::signalStrengthReqMask
- 8.971.2.12 uint8_t unpack_nas_SLQSGetSignalStrength_t::sinr

8.972 unpack_nas_SLQSGetSysInfo_t Struct Reference

Data Fields

- nas_SrvStatusInfo * pCDMASrvStatusInfo
- nas_SrvStatusInfo * pHDRSrvStatusInfo
- nas_GSMsSrvStatusInfo * pGSMsSrvStatusInfo
- nas_GSMsSrvStatusInfo * pWCDMASrvStatusInfo
- nas_GSMsSrvStatusInfo * pLTESrvStatusInfo
- nas_CDMASysInfo * pCDMASysInfo
- nas_HDRSysInfo * pHDRSysInfo
- nas_GSMsSysInfo * pGSMsSysInfo
- nas_WCDMASysInfo * pWCDMASysInfo
- nas_LTESysInfo * pLTESysInfo
- nas_AddCDMASysInfo * pAddCDMASysInfo
- uint16_t * pAddHDRSysInfo
- nas_AddSysInfo * pAddGSMsSysInfo
- nas_AddSysInfo * pAddWCDMASysInfo
- uint16_t * pAddLTESysInfo
- nas_CallBarringSysInfo * pGSMCallBarringSysInfo
- nas_CallBarringSysInfo * pWCDMACallBarringSysInfo
- uint8_t * pLTEVoiceSupportSysInfo
- uint8_t * pGSMCipherDomainSysInfo
- uint8_t * pWCDMACipherDomainSysInfo

8.972.1 Detailed Description

Parameters

<i>pCDMASrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See SrvStatusInfo for more information.
<i>pHDRSrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See SrvStatusInfo for more information.
<i>pGSMSrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See GSMSrvStatusInfo for more information.
<i>pWCDMASrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See GSMSrvStatusInfo for more information.
<i>pLTESrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See GSMSrvStatusInfo for more information.
<i>pCDMASysInfo</i>	<ul style="list-style-type: none"> • See CDMASysInfo for more information.
<i>pHDRSysInfo</i>	<ul style="list-style-type: none"> • See HDRSysInfo for more information.
<i>pGSMSysInfo</i>	<ul style="list-style-type: none"> • See GSMSysInfo for more information.
<i>pWCDMASys↔ Info</i>	<ul style="list-style-type: none"> • See WCDMASysInfo for more information.
<i>pLTESysInfo</i>	<ul style="list-style-type: none"> • See LTESysInfo for more information.
<i>pAddCDMA↔ SysInfo</i>	<ul style="list-style-type: none"> • See AddCDMASysInfo for more information.
<i>pAddHDRSys↔ Info</i>	<ul style="list-style-type: none"> • System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
<i>pAddGSMSys↔ Info</i>	<ul style="list-style-type: none"> • See AddSysInfo for more information.
<i>pAddWCDMA↔ SysInfo</i>	<ul style="list-style-type: none"> • See AddSysInfo for more information.
<i>pAddLTESysInfo</i>	<ul style="list-style-type: none"> • System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
<i>pGSMCall↔ BarringSysInfo</i>	<ul style="list-style-type: none"> • See CallBarringSysInfo for more information.

<i>pWCDMA</i> <i>CallBarringSysInfo</i>	<ul style="list-style-type: none"> • See CallBarringSysInfo for more information.
<i>pLTE</i> <i>VoiceSupportSysInfo</i>	<ul style="list-style-type: none"> • Indicates voice support status on LTE. <ul style="list-style-type: none"> – 0x00 - Voice is not supported – 0x01 - Voice is supported
<i>pGSM</i> <i>CipherDomainSysInfo</i>	<ul style="list-style-type: none"> • Ciphering on the service domain. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched
<i>pWCDMA</i> <i>CipherDomainSysInfo</i>	<ul style="list-style-type: none"> • Ciphering on the service domain. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched

8.972.2 Field Documentation

8.972.2.1 **nas_AddCDMASysInfo*** `unpack_nas_SLQSGetSysInfo_t::pAddCDMASysInfo`

8.972.2.2 **nas_AddSysInfo*** `unpack_nas_SLQSGetSysInfo_t::pAddGSMSysInfo`

8.972.2.3 **uint16_t*** `unpack_nas_SLQSGetSysInfo_t::pAddHDRSysInfo`

8.972.2.4 **uint16_t*** `unpack_nas_SLQSGetSysInfo_t::pAddLTESysInfo`

8.972.2.5 **nas_AddSysInfo*** `unpack_nas_SLQSGetSysInfo_t::pAddWCDMASysInfo`

8.972.2.6 **nas_SrvStatusInfo*** `unpack_nas_SLQSGetSysInfo_t::pCDMASrvStatusInfo`

8.972.2.7 **nas_CDMASysInfo*** `unpack_nas_SLQSGetSysInfo_t::pCDMASysInfo`

8.972.2.8 **nas_CallBarringSysInfo*** `unpack_nas_SLQSGetSysInfo_t::pGSMCallBarringSysInfo`

8.972.2.9 **uint8_t*** `unpack_nas_SLQSGetSysInfo_t::pGSMCipherDomainSysInfo`

8.972.2.10 **nas_GSMSrvStatusInfo*** `unpack_nas_SLQSGetSysInfo_t::pGSMSrvStatusInfo`

8.972.2.11 **nas_GSMSysInfo*** `unpack_nas_SLQSGetSysInfo_t::pGSMSysInfo`

8.972.2.12 `nas_SrvStatusInfo*` `unpack_nas_SLQSGetSysInfo_t::pHDRSrvStatusInfo`

8.972.2.13 `nas_HDRSysInfo*` `unpack_nas_SLQSGetSysInfo_t::pHDRSysInfo`

8.972.2.14 `nas_GSMSrvStatusInfo*` `unpack_nas_SLQSGetSysInfo_t::pLTESrvStatusInfo`

8.972.2.15 `nas_LTESysInfo*` `unpack_nas_SLQSGetSysInfo_t::pLTESysInfo`

8.972.2.16 `uint8_t*` `unpack_nas_SLQSGetSysInfo_t::pLTEVoiceSupportSysInfo`

8.972.2.17 `nas_CallBarringSysInfo*` `unpack_nas_SLQSGetSysInfo_t::pWCDMACallBarringSysInfo`

8.972.2.18 `uint8_t*` `unpack_nas_SLQSGetSysInfo_t::pWCDMACipherDomainSysInfo`

8.972.2.19 `nas_GSMSrvStatusInfo*` `unpack_nas_SLQSGetSysInfo_t::pWCDMASrvStatusInfo`

8.972.2.20 `nas_WCDMASysInfo*` `unpack_nas_SLQSGetSysInfo_t::pWCDMASysInfo`

8.973 `unpack_nas_SLQSGetSysSelectionPref_t` Struct Reference

Data Fields

- `uint8_t *` [pEmerMode](#)
- `uint16_t *` [pModePref](#)
- `uint64_t *` [pBandPref](#)
- `uint16_t *` [pPRLPref](#)
- `uint16_t *` [pRoamPref](#)
- `uint64_t *` [pLTEBandPref](#)
- `uint8_t *` [pNetSelPref](#)
- `uint32_t *` [pSrvDomainPref](#)
- `uint32_t *` [pGWAcqOrderPref](#)

8.973.1 Detailed Description

Parameters

<i>pEmerMode</i>	<ul style="list-style-type: none"> • Optional parameter specifying the emergency Mode • Values: <ul style="list-style-type: none"> – 0 - OFF (normal) – 1 - ON (Emergency)
<i>pModePref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit Mask indicating the radio technology mode preference • Bit values: <ul style="list-style-type: none"> – Bit 0 - cdma2000 1x – Bit 1 - cdma2000 HRPD(1xEV-DO) – Bit 2 - GSM – Bit 3 - UMTS – Bit 4 - LTE
	Generated by Doxygen

<i>pBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the band preference • Bit values: <ul style="list-style-type: none"> – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved
------------------	--

<i>pPRLPref</i>	<ul style="list-style-type: none"> • Optional parameter indicating the CDMA PRL Preference • Values: <ul style="list-style-type: none"> – 0x0001 - Acquire available system only on the A side – 0x0002 - Acquire available system only on the B side – 0x3FFF - Acquire any available systems
<i>pRoamPref</i>	<ul style="list-style-type: none"> • Optional parameter indicating the roaming Preference • Values: <ul style="list-style-type: none"> – 0x01 - Acquire only systems for which the roaming indicator is off – 0x02 - Acquire a system as long as its roaming indicator is not off – 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only – 0xFF - Acquire systems, regardless of their roaming indicator
<i>pLTEBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the LTE band preference • Bit Values <ul style="list-style-type: none"> – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – All other bits are reserved

<i>pNetSelPref</i>	<ul style="list-style-type: none"> Optional parameter indicating network selection preference Values: <ul style="list-style-type: none"> 0x00 - Automatic network selection 0x01 - Manual network selection.
<i>pSrvDomainPref</i>	<ul style="list-style-type: none"> Optional parameter indicating Service domain preference Values: <ul style="list-style-type: none"> 0x00 - Circuit switched only 0x01 - Packet switched only 0x02 - Circuit switched and packet switched 0x03 - Packet switched attach 0x04 - Packet switched detach
<i>pGWAcqOrderPref</i>	<ul style="list-style-type: none"> Optional parameter indicating GSM/WCDMA Acquisition order Preference Values: <ul style="list-style-type: none"> 0x00 - Automatic 0x01 - GSM then WCDMA 0x02 - WCDMA then GSM

8.973.2 Field Documentation

8.973.2.1 uint64_t* unpack_nas_SLQSGetSysSelectionPref_t::pBandPref

8.973.2.2 uint8_t* unpack_nas_SLQSGetSysSelectionPref_t::pEmerMode

8.973.2.3 uint32_t* unpack_nas_SLQSGetSysSelectionPref_t::pGWAcqOrderPref

8.973.2.4 uint64_t* unpack_nas_SLQSGetSysSelectionPref_t::pLTEBandPref

8.973.2.5 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pModePref

8.973.2.6 uint8_t* unpack_nas_SLQSGetSysSelectionPref_t::pNetSelPref

8.973.2.7 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pPRLPref

8.973.2.8 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pRoamPref

8.973.2.9 uint32_t* unpack_nas_SLQSGetSysSelectionPref_t::pSrvDomainPref

8.974 unpack_nas_SLQSNasGetCellLocationInfo_t Struct Reference

Data Fields

- [nas_GERANInfo](#) * [pGERANInfo](#)

- [nas_UMTSInfo](#) * [pUMTSInfo](#)
- [nas_CDMAInfo](#) * [pCDMAInfo](#)
- [nas_LTEInfoIntrafreq](#) * [pLTEInfoIntrafreq](#)
- [nas_LTEInfoInterfreq](#) * [pLTEInfoInterfreq](#)
- [nas_LTEInfoNeighboringGSM](#) * [pLTEInfoNeighboringGSM](#)
- [nas_LTEInfoNeighboringWCDMA](#) * [pLTEInfoNeighboringWCDMA](#)
- [uint32_t](#) * [pUMTSCellID](#)
- [nas_WCDMAInfoLTENeighborCell](#) * [pWCDMAInfoLTENeighborCell](#)

8.974.1 Detailed Description

This structure contains information about the Get Cell Location response parameters.

Parameters

<i>pGERANInfo</i>	<ul style="list-style-type: none"> • See nas_GERANInfo for more information.
<i>pUMTSInfo</i>	<ul style="list-style-type: none"> • See nas_UMTSInfo for more information.
<i>pCDMAInfo</i>	<ul style="list-style-type: none"> • See nas_CDMAInfo for more information.
<i>pLTEInfo↔ Intrafreq</i>	<ul style="list-style-type: none"> • See nas_LTEInfoIntrafreq for more information.
<i>pLTEInfo↔ Interfreq</i>	<ul style="list-style-type: none"> • See nas_LTEInfoInterfreq for more information.
<i>pLTEInfo↔ NeighboringG↔ SM</i>	<ul style="list-style-type: none"> • See nas_LTEInfoNeighboringGSM for more information.
<i>pLTEInfo↔ NeighboringW↔ CDMA</i>	<ul style="list-style-type: none"> • See nas_LTEInfoNeighboringWCDMA for more information.
<i>pUMTSCellID</i>	<ul style="list-style-type: none"> • Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.
<i>pWCDMAInfo↔ LTENeighbor↔ Cell</i>	<ul style="list-style-type: none"> • See nas_WCDMAInfoLTENeighborCell for more information.

8.974.2 Field Documentation

8.974.2.1 [nas_CDMAInfo](#)* [unpack_nas_SLQSNasGetCellLocationInfo_t::pCDMAInfo](#)

8.974.2.2 [nas_GERANInfo](#)* [unpack_nas_SLQSNasGetCellLocationInfo_t::pGERANInfo](#)

8.974.2.3 nas_LTEInfoInterfreq* unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoInterfreq

8.974.2.4 nas_LTEInfoIntrafreq* unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoIntrafreq

8.974.2.5 nas_LTEInfoNeighboringGSM* unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoNeighboringGSM

8.974.2.6 nas_LTEInfoNeighboringWCDMA* unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoNeighboringWCDMA

8.974.2.7 uint32_t* unpack_nas_SLQSNasGetCellLocationInfo_t::pUMTSCellID

8.974.2.8 nas_UMTSInfo* unpack_nas_SLQSNasGetCellLocationInfo_t::pUMTSInfo

8.974.2.9 nas_WCDMAInfoLTENeighborCell* unpack_nas_SLQSNasGetCellLocationInfo_t::pWCDMAInfoLTENeighborCell

8.975 unpack_nas_SLQSNasGetSigInfo_t Struct Reference

Data Fields

- [cdmaSSInfo](#) [CDMASSInfo](#)
- [hdrSSInfo](#) [HDRSSInfo](#)
- [int8_t](#) [GSMSSInfo](#)
- [cdmaSSInfo](#) [WCDMASSInfo](#)
- [lteSSInfo](#) [LTESSInfo](#)

8.975.1 Detailed Description

Parameters

CDMASSInfo	CDMA Signal Strength Information
HDRSSInfo	HDR Signal Strength Information
GSMSSInfo	GSM signal strength is the RSSI in dBm.
WCDMASSInfo	WCDMA Signal Strength Information
LTESSInfo	LTE Signal Strength Information

8.975.2 Field Documentation

8.975.2.1 cdmaSSInfo unpack_nas_SLQSNasGetSigInfo_t::CDMASSInfo

8.975.2.2 int8_t unpack_nas_SLQSNasGetSigInfo_t::GSMSSInfo

8.975.2.3 hdrSSInfo unpack_nas_SLQSNasGetSigInfo_t::HDRSSInfo

8.975.2.4 lteSSInfo unpack_nas_SLQSNasGetSigInfo_t::LTESSInfo

8.975.2.5 cdmaSSInfo unpack_nas_SLQSNasGetSigInfo_t::WCDMASSInfo

8.976 unpack_nas_SLQSNasNetworkTimeCallBack_ind_t Struct Reference

Data Fields

- [nas_UniversalTime](#) universalTime
- uint8_t * [pTimeZone](#)
- uint8_t * [pDayltSavAdj](#)
- uint8_t * [pRadioInterface](#)

8.976.1 Detailed Description

Structure for storing the NAS Network Time indication parameters.

Parameters

<i>universalTime</i>	<ul style="list-style-type: none"> • See nas_UniversalTime for more information.
<i>pTimeZone</i>	<ul style="list-style-type: none"> • Time Zone. • Offset from Universal time, i.e., the difference between local time and Universal time, in increments of 15 min (signed value).
<i>pDayltSavAdj</i>	<ul style="list-style-type: none"> • Daylight Saving Adjustment. • Daylight saving adjustment in hr. <ul style="list-style-type: none"> – Possible values: 0, 1, and 2.
<i>pRadioInterface</i>	<ul style="list-style-type: none"> • Radio interface from which the information comes • Values <ul style="list-style-type: none"> – 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - NAS_RADIO_IF_CDMA_1XEVD0 - cdma2000 HRPD (1xEV-DO) – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA - TD-SCDMA

8.976.2 Field Documentation

8.976.2.1 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pDayltSavAdj

8.976.2.2 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pRadioInterface

8.976.2.3 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pTimeZone

8.976.2.4 nas_UniversalTime unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::universalTime

8.977 unpack_nas_SLQSNasSigInfoCallback_ind_t Struct Reference

Data Fields

- [cdmaSSInfo](#) * [pCDMASigInfo](#)
- [hdrSSInfo](#) * [pHDRSigInfo](#)
- [int8_t](#) * [pGSMSigInfo](#)
- [cdmaSSInfo](#) * [pWCDMASigInfo](#)
- [lteSSInfo](#) * [pLTESigInfo](#)
- [int8_t](#) * [pRscp](#)
- [tdscdmaSigInfoExt](#) * [pTDSCDMASigInfoExt](#)

8.977.1 Detailed Description

Parameters

pCDMASigInfo	CDMA SS info
pHDRSigInfo	HDR SS info
pGSMSigInfo	GSM signal info
pWCDMASigInfo	WCDMA signal info
pLTESigInfo	LTE signal info
pRscp	RSCP of the Primary Common Control Physical Channel
pTDSCDMASigInfoExt	extra CDMA sig info

8.977.2 Field Documentation

8.977.2.1 [cdmaSSInfo](#)* [unpack_nas_SLQSNasSigInfoCallback_ind_t::pCDMASigInfo](#)

8.977.2.2 [int8_t](#)* [unpack_nas_SLQSNasSigInfoCallback_ind_t::pGSMSigInfo](#)

8.977.2.3 [hdrSSInfo](#)* [unpack_nas_SLQSNasSigInfoCallback_ind_t::pHDRSigInfo](#)

8.977.2.4 [lteSSInfo](#)* [unpack_nas_SLQSNasSigInfoCallback_ind_t::pLTESigInfo](#)

8.977.2.5 [int8_t](#)* [unpack_nas_SLQSNasSigInfoCallback_ind_t::pRscp](#)

8.977.2.6 [tdscdmaSigInfoExt](#)* [unpack_nas_SLQSNasSigInfoCallback_ind_t::pTDSCDMASigInfoExt](#)

8.977.2.7 [cdmaSSInfo](#)* [unpack_nas_SLQSNasSigInfoCallback_ind_t::pWCDMASigInfo](#)

8.978 unpack_nas_SLQSNasSwtModemStatus_t Struct Reference

Data Fields

- [nas_CommInfo](#) [commonInfo](#)
- [nas_LTEInfo](#) * [pLTEInfo](#)

8.978.1 Detailed Description

Structure for storing the SLQS Nas Swi Modem Status response parameters.

Parameters

<i>commonInfo</i>	(mandatory) <ul style="list-style-type: none"> See CommInfo for more information
<i>pLTEInfo</i>	(optional) <ul style="list-style-type: none"> See LTEInfo for more information

8.978.2 Field Documentation

8.978.2.1 `nas_CommInfo` `unpack_nas_SLQSNasSwiModemStatus_t::commonInfo`

8.978.2.2 `nas_LTEInfo*` `unpack_nas_SLQSNasSwiModemStatus_t::pLTEInfo`

8.979 `unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t` Struct Reference

Data Fields

- [NASQmiCbkNasSwiOTAMessageInd](#) Info
- `uint16_t` [Tlvresult](#)

8.979.1 Detailed Description

Parameters

<i>Info</i>	<ul style="list-style-type: none"> Structure used to store all QMI Notification Info. <ul style="list-style-type: none"> See NASQmiCbkNasSwiOTAMessageInd for more details
<i>Tlvresult</i>	<ul style="list-style-type: none"> unpack result

8.979.2 Field Documentation

8.979.2.1 `NASQmiCbkNasSwiOTAMessageInd` `unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::Info`

8.979.2.2 `uint16_t` `unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::Tlvresult`

8.980 `unpack_nas_SLQSNasTimerCallback_ind_t` Struct Reference

Data Fields

- char [t3396_apn](#) [101]
- uint8_t [t3396_plmn_id](#) [3]
- uint32_t [t3396_val](#)

8.980.1 Detailed Description

Structure for Network Timer indication parameters.

Parameters

<i>t3396_apn</i>	<ul style="list-style-type: none">• apn
<i>t3396_plmn_id</i>	<ul style="list-style-type: none">• plmn id
<i>t3396_val</i>	<ul style="list-style-type: none">• timer value

8.980.2 Field Documentation

8.980.2.1 char unpack_nas_SLQSNasTimerCallback_ind_t::t3396_apn[101]

8.980.2.2 uint8_t unpack_nas_SLQSNasTimerCallback_ind_t::t3396_plmn_id[3]

8.980.2.3 uint32_t unpack_nas_SLQSNasTimerCallback_ind_t::t3396_val

8.981 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t Struct Reference

Data Fields

- [NASQmiCbKnasSystemSelPrefInd](#) Info
- uint16_t [Tlvresult](#)

8.981.1 Detailed Description

Parameters

<i>Info</i>	<ul style="list-style-type: none">• Structure used to store all QMI Notification Info.<ul style="list-style-type: none">– See NASQmiCbKnasSystemSelPrefInd for more details
<i>Tlvresult</i>	<ul style="list-style-type: none">• unpack result

8.981.2 Field Documentation

8.981.2.1 NASQmiCbkNasSystemSelPrefInd unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::Info

8.981.2.2 uint16_t unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::Tlvresult

8.982 unpack_nas_SLQSSwiGetLteCQI_t Struct Reference

Data Fields

- uint8_t [ValidityCW0](#)
- uint8_t [CQIValueCW0](#)
- uint8_t [ValidityCW1](#)
- uint8_t [CQIValueCW1](#)

8.982.1 Detailed Description

Parameters

<i>ValidityCW0</i> [O \leftrightarrow UT]	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0- Invalid. – 1- Valid.
<i>CQIValueC\leftrightarrowW0</i> [OUT]	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – Range 0~15
<i>ValidityCW1</i> [O \leftrightarrow UT]	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0- Invalid. – 1- Valid.
<i>CQIValueC\leftrightarrowW1</i> [OUT]	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – Range 0~15

8.982.2 Field Documentation

8.982.2.1 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::CQIValueCW0

8.982.2.2 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::CQIValueCW1

8.982.2.3 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::ValidityCW0

8.982.2.4 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::ValidityCW1

8.983 unpack_nas_SLQSSwiGetLteSccRxInfo_t Struct Reference

Data Fields

- [nas_SccRxInfo](#) * [pSccRxInfo](#)

8.983.1 Detailed Description

Parameters

<i>pSccRxInfo</i>	Secondary carrier Rx signal level info
-------------------	--

8.983.2 Field Documentation

8.983.2.1 [nas_SccRxInfo](#)* [unpack_nas_SLQSSwiGetLteSccRxInfo_t::pSccRxInfo](#)

8.984 unpack_nas_SLQSSysInfoCallback_ind_t Struct Reference

Data Fields

- [nas_SrvStatusInfo](#) * [pCDMASrvStatusInfo](#)
- [nas_SrvStatusInfo](#) * [pHDRSrvStatusInfo](#)
- [nas_GSMSrvStatusInfo](#) * [pGSMSrvStatusInfo](#)
- [nas_GSMSrvStatusInfo](#) * [pWCDMASrvStatusInfo](#)
- [nas_GSMSrvStatusInfo](#) * [pLTESrvStatusInfo](#)
- [nas_CDMASysInfo](#) * [pCDMASysInfo](#)
- [nas_HDRSysInfo](#) * [pHDRSysInfo](#)
- [nas_GSMSysInfo](#) * [pGSMSysInfo](#)
- [nas_WCDMASysInfo](#) * [pWCDMASysInfo](#)
- [nas_LTESysInfo](#) * [pLTESysInfo](#)
- [nas_AddCDMASysInfo](#) * [pAddCDMASysInfo](#)
- [uint16_t](#) * [pAddHDRSysInfo](#)
- [nas_AddSysInfo](#) * [pAddGSMSysInfo](#)
- [nas_AddSysInfo](#) * [pAddWCDMASysInfo](#)
- [uint16_t](#) * [pAddLTESysInfo](#)
- [nas_CallBarringSysInfo](#) * [pGSMCallBarringSysInfo](#)
- [nas_CallBarringSysInfo](#) * [pWCDMACallBarringSysInfo](#)
- [uint8_t](#) * [pLTEVoiceSupportSysInfo](#)
- [uint8_t](#) * [pGSMCipherDomainSysInfo](#)
- [uint8_t](#) * [pWCDMACipherDomainSysInfo](#)
- [uint8_t](#) * [pSysInfoNoChange](#)

8.984.1 Detailed Description

Parameters

<i>pCDMASrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See SrvStatusInfo for more information.
<i>pHDRSrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See SrvStatusInfo for more information.
<i>pGSMSrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See GSMSrvStatusInfo for more information.
<i>pWCDMASrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See GSMSrvStatusInfo for more information.
<i>pLTESrv↔ StatusInfo</i>	<ul style="list-style-type: none"> • See GSMSrvStatusInfo for more information.
<i>pCDMASysInfo</i>	<ul style="list-style-type: none"> • See CDMASysInfo for more information.
<i>pHDRSysInfo</i>	<ul style="list-style-type: none"> • See HDRSysInfo for more information.
<i>pGSMSysInfo</i>	<ul style="list-style-type: none"> • See GSMSysInfo for more information.
<i>pWCDMASys↔ Info</i>	<ul style="list-style-type: none"> • See WCDMASysInfo for more information.
<i>pLTESysInfo</i>	<ul style="list-style-type: none"> • See LTESysInfo for more information.
<i>pAddCDMA↔ SysInfo</i>	<ul style="list-style-type: none"> • See AddCDMASysInfo for more information.
<i>pAddHDRSys↔ Info</i>	<ul style="list-style-type: none"> • System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
<i>pAddGSMSys↔ Info</i>	<ul style="list-style-type: none"> • See AddSysInfo for more information.
<i>pAddWCDMA↔ SysInfo</i>	<ul style="list-style-type: none"> • See AddSysInfo for more information.
<i>pAddLTESysInfo</i>	<ul style="list-style-type: none"> • System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
<i>pGSMCall↔ BarringSysInfo</i>	<ul style="list-style-type: none"> • See CallBarringSysInfo for more information.

<i>pWCDMACallBarringSysInfo</i>	<ul style="list-style-type: none"> • See CallBarringSysInfo for more information.
<i>pLTEVoiceSupportSysInfo</i>	<ul style="list-style-type: none"> • Indicates voice support status on LTE. <ul style="list-style-type: none"> – 0x00 - Voice is not supported – 0x01 - Voice is supported
<i>pGSMCipherDomainSysInfo</i>	<ul style="list-style-type: none"> • Ciphering on the service domain. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched
<i>pWCDMACipherDomainSysInfo</i>	<ul style="list-style-type: none"> • Ciphering on the service domain. <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched
<i>pSysInfoNoChange</i>	<ul style="list-style-type: none"> • System Info No Change. • Flag used to notify clients that a request to select a network ended with no change in the PLMN. <ul style="list-style-type: none"> – 0x01 - No change in system information

8.984.2 Field Documentation

8.984.2.1 **nas_AddCDMASysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pAddCDMASysInfo

8.984.2.2 **nas_AddSysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pAddGSMSysInfo

8.984.2.3 **uint16_t*** unpack_nas_SLQSSysInfoCallback_ind_t::pAddHDRSysInfo

8.984.2.4 **uint16_t*** unpack_nas_SLQSSysInfoCallback_ind_t::pAddLTESysInfo

8.984.2.5 **nas_AddSysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pAddWCDMASysInfo

8.984.2.6 **nas_SrvStatusInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pCDMASrvStatusInfo

8.984.2.7 **nas_CDMASysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pCDMASysInfo

8.984.2.8 **nas_CallBarringSysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pGSMCallBarringSysInfo

8.984.2.9 **uint8_t*** unpack_nas_SLQSSysInfoCallback_ind_t::pGSMCipherDomainSysInfo

- 8.984.2.10 `nas_GSMsRvStatusInfo*` `unpack_nas_SLQSSysInfoCallback_ind_t::pGSMsRvStatusInfo`
- 8.984.2.11 `nas_GSMsSysInfo*` `unpack_nas_SLQSSysInfoCallback_ind_t::pGSMsSysInfo`
- 8.984.2.12 `nas_SrvStatusInfo*` `unpack_nas_SLQSSysInfoCallback_ind_t::pHDRsRvStatusInfo`
- 8.984.2.13 `nas_HDRsSysInfo*` `unpack_nas_SLQSSysInfoCallback_ind_t::pHDRsSysInfo`
- 8.984.2.14 `nas_GSMsRvStatusInfo*` `unpack_nas_SLQSSysInfoCallback_ind_t::pLTESrvStatusInfo`
- 8.984.2.15 `nas_LTEsSysInfo*` `unpack_nas_SLQSSysInfoCallback_ind_t::pLTESysInfo`
- 8.984.2.16 `uint8_t*` `unpack_nas_SLQSSysInfoCallback_ind_t::pLTEVoiceSupportSysInfo`
- 8.984.2.17 `uint8_t*` `unpack_nas_SLQSSysInfoCallback_ind_t::pSysInfoNoChange`
- 8.984.2.18 `nas_CallBarringSysInfo*` `unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMA_CallBarringSysInfo`
- 8.984.2.19 `uint8_t*` `unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMACipherDomainSysInfo`
- 8.984.2.20 `nas_GSMsRvStatusInfo*` `unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMASrvStatusInfo`
- 8.984.2.21 `nas_WCDMAsSysInfo*` `unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMAsSysInfo`

8.985 `unpack_omaDmConfigTlv_t` Struct Reference

Data Fields

- `uint8_t` [state](#)
- `uint8_t` [userInputReq](#)
- `uint16_t` [userInputTimeout](#)
- `uint16_t` [alertmsglength](#)
- `uint8_t` [alertmsg](#) [256]

8.985.1 Detailed Description

This structure will hold the `SwiOmaDmConfig` session parameters information.

Parameters

<i>state</i>	<ul style="list-style-type: none"> • 0x01 - OMA-DM Read Request • 0x02 - OMA-DM Change Request • 0x03 - OMA-DM Config Complete
<i>user_input_req</i>	<ul style="list-style-type: none"> - Bit mask of available user inputs • 0x00 - No user input required. Informational indication • 0x01 - Accept • 0x02 - Reject
<i>user_input_timeout</i>	<ul style="list-style-type: none"> • Timeout for user input in minutes. A value of 0 means no time-out
<i>alertmsglength</i>	<ul style="list-style-type: none"> • Length of Alert message string in bytes
<i>alertmsg</i>	<ul style="list-style-type: none"> • Alert message in UCS2 (Max 256 characters)

8.985.2 Field Documentation

8.985.2.1 uint8_t unpack_omaDmConfigTlv_t::alertmsg[256]

8.985.2.2 uint16_t unpack_omaDmConfigTlv_t::alertmsglength

8.985.2.3 uint8_t unpack_omaDmConfigTlv_t::state

8.985.2.4 uint8_t unpack_omaDmConfigTlv_t::userInputReq

8.985.2.5 uint16_t unpack_omaDmConfigTlv_t::userInputTimeout

8.986 unpack_omaDmFotaTlv_t Struct Reference

Data Fields

- uint8_t [state](#)
- uint8_t [userInputReq](#)
- uint16_t [userInputTimeout](#)
- uint32_t [fwdloadsize](#)
- uint32_t [fwloadComplete](#)
- uint16_t [updateCompleteStatus](#)
- uint8_t [severity](#)
- uint16_t [versionlength](#)
- uint8_t [version](#) [256]
- uint16_t [namelength](#)
- uint8_t [package_name](#) [256]
- uint16_t [descriptionlength](#)
- uint8_t [description](#) [256]
- uint8_t [sessionType](#)

8.986.1 Detailed Description

This structure will hold the SwiOmaDmFota session parameters information.

Parameters

<i>state</i>	<ul style="list-style-type: none"> • 0x01 - No Firmware available • 0x02 - Query Firmware Download • 0x03 - Firmware Downloading • 0x04 - Firmware downloaded • 0x05 - Query Firmware Update • 0x06 - Firmware updating • 0x07 - Firmware updated
<i>user_input_req</i>	<ul style="list-style-type: none"> - Bit mask of available user inputs • 0x00 - No user input required. Informational indication • 0x01 - Accept • 0x02 - Reject
<i>user_input</i> ↔ <i>timeout</i>	<ul style="list-style-type: none"> • Timeout for user input in minutes. A value of 0 means no time-out
<i>fw_dload_size</i>	<ul style="list-style-type: none"> • The size (in bytes) of the firmware update package
<i>fw_dload</i> ↔ <i>complete</i>	<ul style="list-style-type: none"> • The number of bytes downloaded. Need to determine how often to send this message for progress bar notification. Every 500ms or 5% increment.
<i>update</i> ↔ <i>complete_status</i>	<ul style="list-style-type: none"> • See table below.
<i>severity</i>	<ul style="list-style-type: none"> • 0x01 - Mandatory • 0x02 - Optional
<i>versionlength</i>	<ul style="list-style-type: none"> • Length of FW Version string in bytes
<i>version</i>	<ul style="list-style-type: none"> • FW Version string in ASCII (Max 256 characters)
<i>namelength</i>	<ul style="list-style-type: none"> • Length Package Name string in bytes
<i>package_name</i>	<ul style="list-style-type: none"> • Package Name in UCS2 (Max 256 characters)
<i>descriptionlength</i>	<ul style="list-style-type: none"> • Length of description in bytes
<i>description</i>	<ul style="list-style-type: none"> • Description of Update Package in USC2 (Max 256 characters)

<i>sessionType</i>	<ul style="list-style-type: none">• 0x00 - Client initiated• 0x01 - Network initiated
--------------------	--

8.986.2 Field Documentation

8.986.2.1 uint8_t unpack_omaDmFotaTlv_t::description[256]

8.986.2.2 uint16_t unpack_omaDmFotaTlv_t::descriptionlength

8.986.2.3 uint32_t unpack_omaDmFotaTlv_t::fwdloadsize

8.986.2.4 uint32_t unpack_omaDmFotaTlv_t::fwloadComplete

8.986.2.5 uint16_t unpack_omaDmFotaTlv_t::namelength

8.986.2.6 uint8_t unpack_omaDmFotaTlv_t::package_name[256]

8.986.2.7 uint8_t unpack_omaDmFotaTlv_t::sessionType

8.986.2.8 uint8_t unpack_omaDmFotaTlv_t::severity

8.986.2.9 uint8_t unpack_omaDmFotaTlv_t::state

8.986.2.10 uint16_t unpack_omaDmFotaTlv_t::updateCompleteStatus

8.986.2.11 uint8_t unpack_omaDmFotaTlv_t::userInputReq

8.986.2.12 uint16_t unpack_omaDmFotaTlv_t::userInputTimeout

8.986.2.13 uint8_t unpack_omaDmFotaTlv_t::version[256]

8.986.2.14 uint16_t unpack_omaDmFotaTlv_t::versionlength

8.987 unpack_omaDmNotificationsTlv_t Struct Reference

Data Fields

- uint8_t [notification](#)
- uint16_t [sessionStatus](#)

8.987.1 Field Documentation

8.987.1.1 `uint8_t unpack_omaDmNotificationsTlv_t::notification`

8.987.1.2 `uint16_t unpack_omaDmNotificationsTlv_t::sessionStatus`

8.988 `unpack_qmi_t` Struct Reference

Data Fields

- enum [msgtype](#) `type`
- `uint16_t` [msgid](#)
- `uint16_t` [xid](#)

8.988.1 Detailed Description

qmi response context

Parameters

<code>out</code>	<i>type</i>	message type
<code>out</code>	<i>msgid</i>	message id
<code>out</code>	<i>xid</i>	transaction id

8.988.2 Field Documentation

8.988.2.1 `uint16_t unpack_qmi_t::msgid`

8.988.2.2 `enum msgtype unpack_qmi_t::type`

8.988.2.3 `uint16_t unpack_qmi_t::xid`

8.989 `unpack_qos_dataRate_t` Struct Reference

Data Fields

- `uint32_t` [dataRateMax](#)
- `uint32_t` [guaranteedRate](#)

8.989.1 Detailed Description

This structure contains the IP flow data rate min max

Parameters

<i>dataRateMax</i>	Maximum required data rate (bits per second)
<i>guaranteedRate</i>	Minimum guaranteed data rate (bits per second)

8.989.2 Field Documentation

8.989.2.1 `uint32_t unpack_qos_dataRate_t::dataRateMax`

8.989.2.2 `uint32_t unpack_qos_dataRate_t::guaranteedRate`

8.990 unpack_qos_IPv4Addr_t Struct Reference

Data Fields

- `uint32_t` [addr](#)
- `uint32_t` [subnetMask](#)

8.990.1 Detailed Description

This structure contains the IPv4 filter address

Parameters

<i>addr</i>	IPv4 address
<i>subnetMask</i>	A packet matches if: <ul style="list-style-type: none">• (addr and subnetMask) == (IP pkt addr & subnetMask) Callers to set up a filter with a range of source addresses, if needed; subnet mask of all 1s (255.255.255.255) specifies a single address value

8.990.2 Field Documentation

8.990.2.1 `uint32_t unpack_qos_IPv4Addr_t::addr`

8.990.2.2 `uint32_t unpack_qos_IPv4Addr_t::subnetMask`

8.991 unpack_qos_IPv6Addr_t Struct Reference

Data Fields

- `uint8_t` [addr](#) [16]
- `uint8_t` [prefixLen](#)

8.991.1 Detailed Description

This structure contains the IPv6 filter address

Parameters

<i>addr</i>	IPv6 address (in network byte order); this is a 16-byte byte array (in Big-endian format)
<i>prefixLen</i>	IPv6 filter prefix length; can take a value between 0 and 128 Note: A packet matches if the IPv6 source address bytes until the prefix lengths are equal. Therefore prefix length can be used to set a filter with a range of source addresses. A prefix length of 128 specifies a single address value.

8.991.2 Field Documentation

8.991.2.1 `uint8_t unpack_qos_IPv6Addr_t::addr[16]`

8.991.2.2 `uint8_t unpack_qos_IPv6Addr_t::prefixLen`

8.992 unpack_qos_IPv6TrafCls_t Struct Reference

Data Fields

- `uint8_t val`
- `uint8_t mask`

8.992.1 Detailed Description

This structure contains the IPv6 filter traffic class

Parameters

<i>val</i>	The traffic class value
<i>mask</i>	The packet matches the traffic class filter if: $(IPv6_filter_traffic_class_val \text{ and } IPv6_filter_traffic_class_mask) == (Traffic\ class\ value\ in\ the\ IP\ packet \& IPv6_filter_traffic_class_mask)$ Example: <ul style="list-style-type: none"> • <code>IPv6_filter_tc_val = 00101000</code> • <code>IPv6_filter_tc_mask = 11111100</code> Filter will compare only the first 6 bits in <code>IPv6_filter_traffic_↵</code> class with the first 6 bits in the traffic class field of the IP packet; first 6 bits in the traffic class field of the IP packet must be 001010 to match filter; last 2 bits can be anything, since they are ignored by filtering

8.992.2 Field Documentation

8.992.2.1 `uint8_t unpack_qos_IPv6TrafCls_t::mask`

8.992.2.2 `uint8_t unpack_qos_IPv6TrafCls_t::val`

8.993 unpack_qos_pktErrRate_t Struct Reference

Data Fields

- uint16_t [multiplier](#)
- uint16_t [exponent](#)

8.993.1 Detailed Description

This structure contains the IP flow packet error rate

Parameters

<i>multiplier</i>	Factor m in calculating packet error rate: $E = m * 10^{(-p)}$
<i>exponent</i>	Factor p in calculating packet error rate (see above)

8.993.2 Field Documentation

8.993.2.1 uint16_t unpack_qos_pktErrRate_t::exponent

8.993.2.2 uint16_t unpack_qos_pktErrRate_t::multiplier

8.994 unpack_qos_Port_t Struct Reference

Data Fields

- uint16_t [port](#)
- uint16_t [range](#)

8.994.1 Detailed Description

This structure contains the [Port](#) Filter

Parameters

<i>port</i>	port value of the filter
<i>range</i>	range specifies the number of ports to be included in the filter starting from port; filter will match if port in the IP packet lies between port and (port + range) Range value of 0 implies that only one value of the port is valid, as specified by the port

8.994.2 Field Documentation

8.994.2.1 uint16_t unpack_qos_Port_t::port

8.994.2.2 uint16_t unpack_qos_Port_t::range

8.995 unpack_qos_QosFlowInfo_t Struct Reference

Data Fields

- [unpack_qos_QosFlowInfoState_t](#) QFlowState
- [uint8_t is_TxQFlowGranted_Available](#)
- [unpack_qos_swiQosFlow_t](#) TxQFlowGranted
- [uint8_t is_RxQFlowGranted_Available](#)
- [unpack_qos_swiQosFlow_t](#) RxQFlowGranted
- [uint8_t](#) NumTxFilters
- [unpack_qos_swiQosFilter_t](#) TxQFilter [25]
- [uint8_t](#) NumRxFilters
- [unpack_qos_swiQosFilter_t](#) RxQFilter [25]
- [uint8_t](#) BearerID

8.995.1 Detailed Description

Structure with QoS flow details.

Please check `is_<Param_Name>_Available` field for presence of optional parameters

Parameters

<i>QFlowState</i>	<ul style="list-style-type: none"> • QoS flow state information, please check unpack_qos_QosFlowInfoState_t for more information
<i>is_TxQFlowGranted_Available</i>	<ul style="list-style-type: none"> • TRUE if optional TxQFlowGranted is available
<i>TxQFlowGranted</i>	<ul style="list-style-type: none"> • The Tx Qos flow granted, please check unpack_qos_swiQosFlow_t for more information
<i>is_RxQFlowGranted_Available</i>	<ul style="list-style-type: none"> • TRUE if optional RxQFlowGranted is available
<i>RxQFlowGranted</i>	<ul style="list-style-type: none"> • The Rx Qos flow granted, please check unpack_qos_swiQosFlow_t for more information
<i>NumTxFilters</i>	<ul style="list-style-type: none"> • Number of Tx filters available
<i>TxQFilter</i>	<ul style="list-style-type: none"> • The Tx Qos filter, please check unpack_qos_swiQosFilter_t for more information • See LIBPACK_MAX_QOS_FILTERS for more information
<i>NumRxFilters</i>	<ul style="list-style-type: none"> • Number of Tx filters available
<i>RxQFilter</i>	<ul style="list-style-type: none"> • The Rx Qos filter, please check unpack_qos_swiQosFilter_t for more information • See LIBPACK_MAX_QOS_FILTERS for more information

<i>BearerID</i>	<ul style="list-style-type: none">• The bearer ID• Bearer ID or Radio Link Protocol (RLP) ID of the activated flow.• Valid Values - 0 to 16• 0xFF - Invalid value.
-----------------	---

8.995.2 Field Documentation

8.995.2.1 `uint8_t unpack_qos_QosFlowInfo_t::BearerID`

8.995.2.2 `uint8_t unpack_qos_QosFlowInfo_t::is_RxQFlowGranted_Available`

8.995.2.3 `uint8_t unpack_qos_QosFlowInfo_t::is_TxQFlowGranted_Available`

8.995.2.4 `uint8_t unpack_qos_QosFlowInfo_t::NumRxFilters`

8.995.2.5 `uint8_t unpack_qos_QosFlowInfo_t::NumTxFilters`

8.995.2.6 `unpack_qos_QosFlowInfoState_t unpack_qos_QosFlowInfo_t::QFlowState`

8.995.2.7 `unpack_qos_swiQosFilter_t unpack_qos_QosFlowInfo_t::RxQFilter[25]`

8.995.2.8 `unpack_qos_swiQosFlow_t unpack_qos_QosFlowInfo_t::RxQFlowGranted`

8.995.2.9 `unpack_qos_swiQosFilter_t unpack_qos_QosFlowInfo_t::TxQFilter[25]`

8.995.2.10 `unpack_qos_swiQosFlow_t unpack_qos_QosFlowInfo_t::TxQFlowGranted`

8.996 unpack_qos_QosFlowInfoState_t Struct Reference

Data Fields

- `uint32_t id`
- `uint8_t isNewFlow`
- `uint8_t state`

8.996.1 Detailed Description

This structure contains QoS flow state

Parameters

<i>id</i>	QoS identifier
<i>isNewFlow</i>	<ul style="list-style-type: none"> • 1 – Newly added flow • 0 – Existing flow
<i>state</i>	This indicates that the flow that was added/modified/deleted: <ul style="list-style-type: none"> • 0x01 – Flow activated • 0x02 – Flow modified • 0x03 – Flow deleted • 0x04 – Flow suspended • 0x05 – Flow enabled • 0x06 – Flow disabled

8.996.2 Field Documentation

8.996.2.1 `uint32_t unpack_qos_QosFlowInfoState_t::id`

8.996.2.2 `uint8_t unpack_qos_QosFlowInfoState_t::isNewFlow`

8.996.2.3 `uint8_t unpack_qos_QosFlowInfoState_t::state`

8.997 `unpack_qos_SLQSQosGetNetworkStatus_t` Struct Reference

Data Fields

- `uint8_t` [NWQoSStatus](#)

8.997.1 Detailed Description

Structure that contains the response to get NW QoS status command

Parameters

<i>NWQoSStatus</i>	Network QoS support status <ul style="list-style-type: none"> • 0 – No QoS support in network • 1 – Network supports QoS
--------------------	--

8.997.2 Field Documentation

8.997.2.1 uint8_t unpack_qos_SLQSQosGetNetworkStatus_t::NWQoSStatus

8.998 unpack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference

Data Fields

- uint32_t [apnId](#)
- uint8_t [ambr_ul](#)
- uint8_t [ambr_dl](#)
- uint8_t [ambr_ul_ext](#)
- uint8_t [ambr_dl_ext](#)
- uint8_t [ambr_ul_ext2](#)
- uint8_t [ambr_dl_ext2](#)

8.998.1 Detailed Description

Structure that contains extra APN parameters

Parameters

<i>apnId</i>	<ul style="list-style-type: none"> • APN id • ID identifying the APN that the client would like to query the AMBR params
<i>ambr_ul</i>	<ul style="list-style-type: none"> • APN AMBR uplink • APN AMBR uplink values from 1 kbps to 8640 kbps
<i>ambr_dl</i>	<ul style="list-style-type: none"> • APN AMBR downlink • APN AMBR downlink values from 1 kbps to 8640 kbps
<i>ambr_ul_ext</i>	<ul style="list-style-type: none"> • Extended APN AMBR uplink • APN AMBR uplink values from 8700 kbps to 256 Mbps
<i>ambr_dl_ext</i>	<ul style="list-style-type: none"> • Extended APN AMBR downlink • APN AMBR downlink values from 8700 kbps to 256 Mbps
<i>ambr_ul_ext2</i>	<ul style="list-style-type: none"> • Second extended APN AMBR uplink • APN AMBR uplink values from 256 Mbps to 65280 Mbps
<i>ambr_dl_ext2</i>	<ul style="list-style-type: none"> • Second extended APN AMBR downlink • APN AMBR downlink values from 256 Mbps to 65280 Mbps

8.998.2 Field Documentation

8.998.2.1 `uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl`

8.998.2.2 `uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl_ext`

8.998.2.3 `uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl_ext2`

8.998.2.4 `uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul`

8.998.2.5 `uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul_ext`

8.998.2.6 `uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul_ext2`

8.998.2.7 `uint32_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::apnId`

8.999 `unpack_qos_SLQSQosSwiReadDataStats_t` Struct Reference

Data Fields

- `uint32_t apnId`
- `uint32_t total_tx_pkt`
- `uint32_t total_tx_pkt_drp`
- `uint32_t total_rx_pkt`
- `uint64_t total_tx_bytes`
- `uint64_t total_tx_bytes_drp`
- `uint64_t total_rx_bytes`
- `uint32_t numQosFlow`
- `unpack_QosFlowStat_t qosFlow` [10]

8.999.1 Detailed Description

Structure that contains APN data statistics

Parameters

<i>apnId</i>	<ul style="list-style-type: none"> • APN id • ID identifying the connected APN that the client would like to query the data statistic for
<i>total_tx_pkt</i>	<ul style="list-style-type: none"> • sum of all packets sent
<i>total_tx_pkt_drp</i>	<ul style="list-style-type: none"> • sum of all(TX) packets dropped
<i>total_rx_pkt</i>	<ul style="list-style-type: none"> • sum of all packets received

<i>total_tx_bytes</i>	<ul style="list-style-type: none"> • sum of all bytes sent
<i>total_tx_bytes_drp</i>	<ul style="list-style-type: none"> • sum of all(TX) bytes dropped
<i>total_rx_bytes</i>	<ul style="list-style-type: none"> • number of received bytes for the QoS flow ID
<i>numQosFlow</i>	<ul style="list-style-type: none"> • pointer to number of QoS flow Stat
<i>qosFlow[LIBPACK_MAX_QOS_FLOW_PER_APN_STATS]</i>	<ul style="list-style-type: none"> • Data statistic per QoS flow • See unpack_QosFlowStat_t for more information • See LIBPACK_MAX_QOS_FLOW_PER_APN_STATS for more information

8.999.2 Field Documentation

8.999.2.1 `uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::apnId`

8.999.2.2 `uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::numQosFlow`

8.999.2.3 `unpack_QosFlowStat_t unpack_qos_SLQSQosSwiReadDataStats_t::qosFlow[10]`

8.999.2.4 `uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_rx_bytes`

8.999.2.5 `uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_rx_pkt`

8.999.2.6 `uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_bytes`

8.999.2.7 `uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_bytes_drp`

8.999.2.8 `uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_pkt`

8.999.2.9 `uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_pkt_drp`

8.1000 unpack_qos_SLQSSetQosEventCallback_ind_t Struct Reference

Data Fields

- `uint8_t NumFlows`
- `unpack_qos_QosFlowInfo_t QosFlowInfo [8]`

8.1000.1 Detailed Description

Structure with QoS event details

Parameters

<i>NumFlows</i>	<ul style="list-style-type: none"> • Number of QoS flows available
<i>QosFlowInfo</i>	<ul style="list-style-type: none"> • The Qos flow details, please check unpack_qos_QosFlowInfo_t for more information • See LIBPACK_MAX_QOS_FLOWS for more information

8.1000.2 Field Documentation

8.1000.2.1 `uint8_t unpack_qos_SLQSSetQosEventCallback_ind_t::NumFlows`

8.1000.2.2 `unpack_qos_QosFlowInfo_t unpack_qos_SLQSSetQosEventCallback_ind_t::QosFlowInfo[8]`

8.1001 `unpack_qos_SLQSSetQosNWStatusCallback_ind_t` Struct Reference

Data Fields

- `uint8_t status`

8.1001.1 Detailed Description

Structure with network's QoS status

Parameters

<i>status</i>	Network QoS support status <ul style="list-style-type: none"> • 0x00 – Current network does not support QoS • 0x01 – Current network supports QoS
---------------	---

Note

- Technology Supported: CDMA

8.1001.2 Field Documentation

8.1001.2.1 `uint8_t unpack_qos_SLQSSetQosNWStatusCallback_ind_t::status`

8.1002 `unpack_qos_SLQSSetQosPriEventCallback_ind_t` Struct Reference

Data Fields

- `uint16_t event`

8.1002.1 Detailed Description

Structure with QoS primary flow events

Parameters

<i>event</i>	Event which causes this indication: <ul style="list-style-type: none">• 0x0001 – Primary flow QoS modify operation success• 0x0002 – Primary flow QoS modify operation failure
--------------	---

8.1002.2 Field Documentation

8.1002.2.1 uint16_t unpack_qos_SLQSSetQosPriEventCallback_ind_t::event

8.1003 unpack_qos_SLQSSetQosStatusCallback_ind_t Struct Reference

Data Fields

- uint32_t [id](#)
- uint8_t [status](#)
- uint8_t [event](#)
- uint8_t [reason](#)

8.1003.1 Detailed Description

Structure with QoS status indication details

Parameters

<i>id</i>	<ul style="list-style-type: none">• Index identifying the QoS flow whose status is being reported
<i>status</i>	Current QoS flow status: <ul style="list-style-type: none">• 0x01 – QMI_QOS_STATUS_ACTIVATED• 0x02 – QMI_QOS_STATUS_SUSPENDED• 0x03 – QMI_QOS_STATUS_GONE
<i>event</i>	<ul style="list-style-type: none">• 0x01 – QMI_QOS_ACTIVATED_EV• 0x02 – QMI_QOS_SUSPENDED_EV• 0x03 – QMI_QOS_GONE_EV• 0x04 – QMI_QOS_MODIFY_ACCEPTED_EV• 0x05 – QMI_QOS_MODIFY_REJECTED_EV• 0x06 – QMI_QOS_INFO_CODE_UPDATED_EV

<i>reason</i>	<ul style="list-style-type: none"> • 0x01 - QMI_QOS_INVALID_PARAMS • 0x02 - QMI_QOS_INTERNAL_CALL_ENDED • 0x03 - QMI_QOS_INTERNAL_ERROR • 0x04 - QMI_QOS_INSUFFICIENT_LOCAL_Resources • 0x05 - QMI_QOS_TIMED_OUT_OPERATION • 0x06 - QMI_QOS_INTERNAL_UNKNOWN_CAUSE_CODE • 0x07 - QMI_QOS_INTERNAL_MODIFY_IN_PROGRESS • 0x08 - QMI_QOS_NOT_SUPPORTED • 0x09 - QMI_QOS_NOT_AVAILABLE • 0x0A - QMI_QOS_NOT_GUARANTEED • 0x0B - QMI_QOS_INSUFFICIENT_NETWORK_RESOURCES • 0x0C - QMI_QOS_AWARE_SYSTEM • 0x0D - QMI_QOS_UNAWARE_SYSTEM • 0x0E - QOS_REJECTED_OPERATION • 0x0F - QMI_QOS_WILL_GRANT_WHEN_QOS_RESUMED • 0x10 - QMI_QOS_NETWORK_CALL_ENDED • 0x11 - QMI_QOS_NETWORK_SERVICE_NOT_AVAILABLE • 0x12 - QMI_QOS_NETWORK_L2_LINK_RELEASED • 0x13 - QMI_QOS_NETWORK_L2_LINK_REESTAB_REJ • 0x14 - QMI_QOS_NETWORK_L2_LINK_REESTAB_IND • 0x15 - QMI_QOS_NETWORK_UNKNOWN_CAUSE_CODE • 0x16 - QMI_NETWORK_BUSY
---------------	--

8.1003.2 Field Documentation

8.1003.2.1 `uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::event`

8.1003.2.2 `uint32_t unpack_qos_SLQSSetQosStatusCallback_ind_t::id`

8.1003.2.3 `uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::reason`

8.1003.2.4 `uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::status`

8.1004 `unpack_qos_swiQosFilter_t` Struct Reference

Data Fields

- `uint8_t index`
- `uint8_t version`
- `uint8_t is_IPv4SrcAddr_Available`
- `unpack_qos_IPv4Addr_t IPv4SrcAddr`
- `uint8_t is_IPv4DstAddr_Available`
- `unpack_qos_IPv4Addr_t IPv4DstAddr`
- `uint8_t is_NxtHdrProto_Available`
- `uint8_t NxtHdrProto`

- [uint8_t is_IPv4Tos_Available](#)
- [unpack_qos_Tos_t](#) IPv4Tos
- [uint8_t is_IPv6SrcAddr_Available](#)
- [unpack_qos_IPv6Addr_t](#) IPv6SrcAddr
- [uint8_t is_IPv6DstAddr_Available](#)
- [unpack_qos_IPv6Addr_t](#) IPv6DstAddr
- [uint8_t is_IPv6TrafCls_Available](#)
- [unpack_qos_IPv6TrafCls_t](#) IPv6TrafCls
- [uint8_t is_IPv6Label_Available](#)
- [uint32_t](#) IPv6Label
- [uint8_t is_TCPSrcPort_Available](#)
- [unpack_qos_Port_t](#) TCPSrcPort
- [uint8_t is_TCPDstPort_Available](#)
- [unpack_qos_Port_t](#) TCPDstPort
- [uint8_t is_UDPSrcPort_Available](#)
- [unpack_qos_Port_t](#) UDPSrcPort
- [uint8_t is_UDPDstPort_Available](#)
- [unpack_qos_Port_t](#) UDPDstPort
- [uint8_t is_EspSpi_Available](#)
- [uint32_t](#) EspSpi
- [uint8_t is_Precedence_Available](#)
- [uint16_t](#) Precedence
- [uint8_t is_Id_Available](#)
- [uint16_t](#) Id
- [uint8_t is_TransrcPort_Available](#)
- [unpack_qos_Port_t](#) TransrcPort
- [uint8_t is_TransdstPort_Available](#)
- [unpack_qos_Port_t](#) TransdstPort

8.1004.1 Detailed Description

This structure contains the QoS Filter Request.

Please check `is_<Param_Name>_Available` field for presence of optional parameters

Parameters

<i>index</i>	Mandatory parameter IP filter index Integer that uniquely identifies each filter instance This TLV must be present in the request
<i>version</i>	Mandatory parameter IP filter version Identifies whether the filter is associated with IPv4 or IPv6; value specified also implies that only TLVs defined for that IP version, i.e., TLVs with IPv4 or IPv6 in the name, can be specified <ul style="list-style-type: none"> • 0x04 – IPv4 • 0x06 – Ipv6
<i>IPv4SrcAddr</i>	IPv4 filter source address See unpack_qos_IPv4Addr_t for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>IPv4DstAddr</i>	IPv4 filter destination address See unpack_qos_IPv4Addr_t for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication

<i>NxtHdrProto</i>	IP filter next header protocol This TLV must be present if any non-IP filter TLV(s) are provided If this field is specified, only IP packets belonging to specified higher layer protocol are considered when filtering The following protocols may be specified: <ul style="list-style-type: none"> • 0x01 = ICMP • 0x06 = TCP • 0x11 = UDP • 0x32 = ESP Note: The next header protocol field will be set to 0xFD (TCP & UDP) if a TFT is received specifying a source or destination port number, but IP next header type is not specified.
<i>IPv4Tos</i>	IPv4 filter type of service See unpack_qos_Tos_t for more information
<i>IPv6SrcAddr</i>	IPv6 filter source address See unpack_qos_IPv6Addr_t for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>IPv6DstAddr</i>	IPv6 filter destination address See unpack_qos_IPv6Addr_t for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>IPv6TrafCls</i>	IPv6 filter traffic class See unpack_qos_IPv6TrafCls_t for more information
<i>IPv6Label</i>	IPv6 flow label Packet matches the IPv6 flow label filter if: (*pIPv6Label == flow label in the IPv6 header) <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>TCPsrcPort</i>	TCP filter source port filter See unpack_qos_Port_t for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>TCPdstPort</i>	TCP filter destination port filter See unpack_qos_Port_t for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>UDPSrcPort</i>	UDP filter source port filter See unpack_qos_Port_t for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>UDPdstPort</i>	UDP filter destination port filter See unpack_qos_Port_t for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>EspSpi</i>	ESP filter security policy index Security policy index to uniquely identify each IP flow for filtering encrypted packets for encapsulating security payload <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>Precedence</i>	Filter Precedence Specifies the order in which filters are applied; lower numerical value has higher precedence Note: This TLV only applies to network-initiated QoS; QoS requests containing this TLV from control points will be ignored
<i>Id</i>	Filter ID Unique identifier for each filter; filter ID is assigned by the modem Note: This TLV only applies to network-initiated QoS; QoS requests containing this TLV from control points will be ignored
<i>TranSrcPort</i>	Transport protocol filter source port See unpack_qos_Port_t for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication
<i>UDPdstPort</i>	Transport protocol filter destination port See unpack_qos_Port_t for more information <ul style="list-style-type: none"> • Implemented only for unsolicited indication

8.1004.2 Field Documentation

8.1004.2.1 uint32_t unpack_qos_swiQosFilter_t::EspSpi

- 8.1004.2.2 `uint16_t unpack_qos_swiQosFilter_t::ld`
- 8.1004.2.3 `uint8_t unpack_qos_swiQosFilter_t::index`
- 8.1004.2.4 `unpack_qos_IPv4Addr_t unpack_qos_swiQosFilter_t::IPv4DstAddr`
- 8.1004.2.5 `unpack_qos_IPv4Addr_t unpack_qos_swiQosFilter_t::IPv4SrcAddr`
- 8.1004.2.6 `unpack_qos_Tos_t unpack_qos_swiQosFilter_t::IPv4Tos`
- 8.1004.2.7 `unpack_qos_IPv6Addr_t unpack_qos_swiQosFilter_t::IPv6DstAddr`
- 8.1004.2.8 `uint32_t unpack_qos_swiQosFilter_t::IPv6Label`
- 8.1004.2.9 `unpack_qos_IPv6Addr_t unpack_qos_swiQosFilter_t::IPv6SrcAddr`
- 8.1004.2.10 `unpack_qos_IPv6TrafCls_t unpack_qos_swiQosFilter_t::IPv6TrafCls`
- 8.1004.2.11 `uint8_t unpack_qos_swiQosFilter_t::is_EspSpi_Available`
- 8.1004.2.12 `uint8_t unpack_qos_swiQosFilter_t::is_Id_Available`
- 8.1004.2.13 `uint8_t unpack_qos_swiQosFilter_t::is_IPv4DstAddr_Available`
- 8.1004.2.14 `uint8_t unpack_qos_swiQosFilter_t::is_IPv4SrcAddr_Available`
- 8.1004.2.15 `uint8_t unpack_qos_swiQosFilter_t::is_IPv4Tos_Available`
- 8.1004.2.16 `uint8_t unpack_qos_swiQosFilter_t::is_IPv6DstAddr_Available`
- 8.1004.2.17 `uint8_t unpack_qos_swiQosFilter_t::is_IPv6Label_Available`
- 8.1004.2.18 `uint8_t unpack_qos_swiQosFilter_t::is_IPv6SrcAddr_Available`
- 8.1004.2.19 `uint8_t unpack_qos_swiQosFilter_t::is_IPv6TrafCls_Available`
- 8.1004.2.20 `uint8_t unpack_qos_swiQosFilter_t::is_NxtHdrProto_Available`
- 8.1004.2.21 `uint8_t unpack_qos_swiQosFilter_t::is_Precedence_Available`
- 8.1004.2.22 `uint8_t unpack_qos_swiQosFilter_t::is_TCPDstPort_Available`
- 8.1004.2.23 `uint8_t unpack_qos_swiQosFilter_t::is_TCPSrcPort_Available`
- 8.1004.2.24 `uint8_t unpack_qos_swiQosFilter_t::is_TranDstPort_Available`

8.1004.2.25 `uint8_t unpack_qos_swiQosFilter_t::is_TransrcPort_Available`

8.1004.2.26 `uint8_t unpack_qos_swiQosFilter_t::is_UDPdstPort_Available`

8.1004.2.27 `uint8_t unpack_qos_swiQosFilter_t::is_UDPSrcPort_Available`

8.1004.2.28 `uint8_t unpack_qos_swiQosFilter_t::NxtHdrProto`

8.1004.2.29 `uint16_t unpack_qos_swiQosFilter_t::Precedence`

8.1004.2.30 `unpack_qos_Port_t unpack_qos_swiQosFilter_t::TCPdstPort`

8.1004.2.31 `unpack_qos_Port_t unpack_qos_swiQosFilter_t::TCPSrcPort`

8.1004.2.32 `unpack_qos_Port_t unpack_qos_swiQosFilter_t::TranDstPort`

8.1004.2.33 `unpack_qos_Port_t unpack_qos_swiQosFilter_t::TranSrcPort`

8.1004.2.34 `unpack_qos_Port_t unpack_qos_swiQosFilter_t::UDPdstPort`

8.1004.2.35 `unpack_qos_Port_t unpack_qos_swiQosFilter_t::UDPSrcPort`

8.1004.2.36 `uint8_t unpack_qos_swiQosFilter_t::version`

8.1005 `unpack_qos_swiQosFlow_t` Struct Reference

Data Fields

- `uint8_t index`
- `uint8_t is_ProfileId3GPP2_Available`
- `uint16_t ProfileId3GPP2`
- `uint8_t is_val_3GPP2Pri_Available`
- `uint8_t val_3GPP2Pri`
- `uint8_t is_TrafficClass_Available`
- `uint8_t TrafficClass`
- `uint8_t is_DataRate_Available`
- `unpack_qos_dataRate_t DataRate`
- `uint8_t is_TokenBucket_Available`
- `unpack_qos_tokenBucket_t TokenBucket`
- `uint8_t is_Latency_Available`
- `uint32_t Latency`
- `uint8_t is_Jitter_Available`
- `uint32_t Jitter`
- `uint8_t is_PktErrRate_Available`
- `unpack_qos_pktErrRate_t PktErrRate`
- `uint8_t is_MinPolicedPktSz_Available`
- `uint32_t MinPolicedPktSz`
- `uint8_t is_MaxAllowedPktSz_Available`
- `uint32_t MaxAllowedPktSz`
- `uint8_t is_val_3GPPResResidualBER_Available`
- `uint16_t val_3GPPResResidualBER`
- `uint8_t is_val_3GPPTraHdlPri_Available`
- `uint8_t val_3GPPTraHdlPri`
- `uint8_t is_val_3GPPImCn_Available`
- `uint8_t val_3GPPImCn`
- `uint8_t is_val_3GPPSigInd_Available`
- `uint8_t val_3GPPSigInd`
- `uint8_t is_LteQci_Available`
- `uint8_t LteQci`

8.1005.1 Detailed Description

This structure contains the QoS Flow Request.

Please check is_<Param_Name>_Available field for presence of optional parameters

Parameters

<i>index</i>	<ul style="list-style-type: none"> • Mandatory parameter • IP flow index • Integer that uniquely identifies each flow instance • Unique index must be assigned by the control point to every flow_spec instance
<i>ProfileId3GPP2</i>	<ul style="list-style-type: none"> • IP flow 3GPP2 profile ID • A profile ID is shorthand for a defined set of QoS flow parameters specified by the network; to be present while requesting QoS for a CDMA device
<i>val_3GPP2Pri</i>	<ul style="list-style-type: none"> • IP flow 3GPP2 flow priority • Flow priority used by the network in case of contention between flows with same QoS; this parameter applies for CDMA devices
<i>TrafficClass</i>	<ul style="list-style-type: none"> • IP flow traffic class • Integer that designates the requested traffic class: <ul style="list-style-type: none"> • 0 – Conversational • 1 – Streaming • 2 – Interactive • 3 – Background
<i>DataRate</i>	<ul style="list-style-type: none"> • IP flow data rate min max • See unpack_qos_dataRate_t for more information
<i>TokenBucket</i>	<ul style="list-style-type: none"> • IP flow data rate token bucket • See unpack_qos_tokenBucket_t for more information
<i>Latency</i>	<ul style="list-style-type: none"> • IP flow latency • Maximum delay (in milliseconds) that can be tolerated by an IP packet during transfer through the wireless link
<i>Jitter</i>	<ul style="list-style-type: none"> • IP flow jitter • Difference between the maximum and minimum latency (in milliseconds) that can be tolerated by an IP packet during the transfer through the wireless link
<i>PktErrRate</i>	<ul style="list-style-type: none"> • IP flow packet error rate • See unpack_qos_pktErrRate_t for more information

<i>MinPolicedPktSz</i>	<ul style="list-style-type: none"> • IP flow minimum policed packet size • Integer that defines the minimum packet size (in bytes) that will be policed for QoS guarantees; any IP packets that are smaller than the minimum specified policed size may not receive requested QoS
<i>MaxAllowedPktSz</i>	<ul style="list-style-type: none"> • IP flow maximum allowed packet size • Integer that defines the maximum packet size (in bytes) allowed in the IP flow; any IP packets greater in size than the maximum allowed packet size are not queued for transmission
<i>val_3GPPResidualBER</i>	<ul style="list-style-type: none"> • IP flow 3GPP residual bit error rate • residual_bit_error_rate • 0 = 5×10^{-2} residual BER • 1 = 1×10^{-2} residual BER • 2 = 5×10^{-3} residual BER • 3 = 4×10^{-3} residual BER • 4 = 1×10^{-3} residual BER • 5 = 1×10^{-4} residual BER • 6 = 1×10^{-5} residual BER • 7 = 1×10^{-6} residual BER • 8 = 6×10^{-8} residual BER • Integer that indicates the undetected BER for each IP flow in the delivered packets; Applies only to 3GPP networks
<i>val_3GPPTrafficHandlingPriority</i>	<ul style="list-style-type: none"> • 3GPP traffic handling priority • 0 – Relative traffic handling priority 1 • 1 – Relative traffic handling priority 2 • 2 – Relative traffic handling priority 3 • Defines the relative priority of the flow; applies only to 3GPP networks
<i>val_3GPPIMCn</i>	<ul style="list-style-type: none"> • IP flow 3GPP IM CN flag • IM CN subsystem signaling flag: • 0x00 – FALSE • 0x01 – TRUE • This parameter applies only to 3GPP networks
<i>val_3GPPSigInd</i>	<ul style="list-style-type: none"> • IP flow 3GPP signaling indication • 0x00 – FALSE • 0x01 – TRUE • This parameter applies only to 3GPP networks

<i>LteQci</i>	<ul style="list-style-type: none"> • LTE QoS Class Identifier • QoS Class Identifier(QCI) is a required parameter to request QoS in LTE • QCI values: <ul style="list-style-type: none"> – QCI value 0 requests the network to assign the appropriate QCI value – QCI values 1-4 are associated with guaranteed bitrates – QCI values 5-9 are associated with nonguaranteed bitrates, so the values specified as guaranteed and maximum bitrates are ignored
---------------	---

8.1005.2 Field Documentation

8.1005.2.1 `unpack_qos_dataRate_t unpack_qos_swiQosFlow_t::DataRate`

8.1005.2.2 `uint8_t unpack_qos_swiQosFlow_t::index`

8.1005.2.3 `uint8_t unpack_qos_swiQosFlow_t::is_DataRate_Available`

8.1005.2.4 `uint8_t unpack_qos_swiQosFlow_t::is_Jitter_Available`

8.1005.2.5 `uint8_t unpack_qos_swiQosFlow_t::is_Latency_Available`

8.1005.2.6 `uint8_t unpack_qos_swiQosFlow_t::is_LteQci_Available`

8.1005.2.7 `uint8_t unpack_qos_swiQosFlow_t::is_MaxAllowedPktSz_Available`

8.1005.2.8 `uint8_t unpack_qos_swiQosFlow_t::is_MinPolicedPktSz_Available`

8.1005.2.9 `uint8_t unpack_qos_swiQosFlow_t::is_PktErrRate_Available`

8.1005.2.10 `uint8_t unpack_qos_swiQosFlow_t::is_ProfileId3GPP2_Available`

8.1005.2.11 `uint8_t unpack_qos_swiQosFlow_t::is-TokenBucket_Available`

8.1005.2.12 `uint8_t unpack_qos_swiQosFlow_t::is_TrafficClass_Available`

8.1005.2.13 `uint8_t unpack_qos_swiQosFlow_t::is_val_3GPP2Pri_Available`

8.1005.2.14 `uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPImCn_Available`

8.1005.2.15 `uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPResResidualBER_Available`

8.1005.2.16 `uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPSigInd_Available`

8.1005.2.17 `uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPTraHdIPri_Available`

- 8.1005.2.18 `uint32_t unpack_qos_swiQosFlow_t::Jitter`
- 8.1005.2.19 `uint32_t unpack_qos_swiQosFlow_t::Latency`
- 8.1005.2.20 `uint8_t unpack_qos_swiQosFlow_t::LteQci`
- 8.1005.2.21 `uint32_t unpack_qos_swiQosFlow_t::MaxAllowedPktSz`
- 8.1005.2.22 `uint32_t unpack_qos_swiQosFlow_t::MinPolicedPktSz`
- 8.1005.2.23 `unpack_qos_pktErrRate_t unpack_qos_swiQosFlow_t::PktErrRate`
- 8.1005.2.24 `uint16_t unpack_qos_swiQosFlow_t::ProfileId3GPP2`
- 8.1005.2.25 `unpack_qos_tokenBucket_t unpack_qos_swiQosFlow_t::TokenBucket`
- 8.1005.2.26 `uint8_t unpack_qos_swiQosFlow_t::TrafficClass`
- 8.1005.2.27 `uint8_t unpack_qos_swiQosFlow_t::val_3GPP2Pri`
- 8.1005.2.28 `uint8_t unpack_qos_swiQosFlow_t::val_3GPPImCn`
- 8.1005.2.29 `uint16_t unpack_qos_swiQosFlow_t::val_3GPPResResidualBER`
- 8.1005.2.30 `uint8_t unpack_qos_swiQosFlow_t::val_3GPPSigInd`
- 8.1005.2.31 `uint8_t unpack_qos_swiQosFlow_t::val_3GPPTraHdlPri`

8.1006 `unpack_qos_tokenBucket_t` Struct Reference

Data Fields

- `uint32_t` [peakRate](#)
- `uint32_t` [tokenRate](#)
- `uint32_t` [bucketSz](#)

8.1006.1 Detailed Description

This structure contains the TP flow data rate token bucket

Parameters

<i>peakRate</i>	Maximum rate at which data can be transmitted when the token bucket is full (bits per second)
<i>tokenRate</i>	Rate at which tokens will be put in the token bucket (bits per second); a token is required to be present in the bucket to send a byte of data
<i>bucketSz</i>	Maximum number of tokens that can be accumulated at any instance (bytes); controls the size of the burst that is allowed at any given time

8.1006.2 Field Documentation

8.1006.2.1 uint32_t unpack_qos_tokenBucket_t::bucketSz

8.1006.2.2 uint32_t unpack_qos_tokenBucket_t::peakRate

8.1006.2.3 uint32_t unpack_qos_tokenBucket_t::tokenRate

8.1007 unpack_qos_Tos_t Struct Reference

Data Fields

- uint8_t [val](#)
- uint8_t [mask](#)

8.1007.1 Detailed Description

This structure contains the IPv4 filter type of service

Parameters

<i>val</i>	Type of service value
<i>mask</i>	Packet matches the TOS filter if: (IPv4_filter_tos_val and IPv4_filter_tos_mask) == (TOS value in the IP packet & IPv4_filter_tos_mask) Example: <ul style="list-style-type: none"> • IPv4_filter_tos_val = 00101000 • IPv4_filter_tos_mask = 11111100 The filter will compare only the first 6 bits in the IPv4_filter↔_type_of_service with the first 6 bits in the TOS field of the IP packet. The first 6 bits in the TOS field of the IP packet must be 001010 to match the filter. The last 2 bits can be anything since they are ignored by filtering.

8.1007.2 Field Documentation

8.1007.2.1 uint8_t unpack_qos_Tos_t::mask

8.1007.2.2 uint8_t unpack_qos_Tos_t::val

8.1008 unpack_QosFlowStat_t Struct Reference

Data Fields

- uint32_t [bearerId](#)
- uint32_t [tx_pkt](#)
- uint32_t [tx_pkt_drp](#)
- uint64_t [tx_bytes](#)
- uint64_t [tx_bytes_drp](#)

8.1008.1 Detailed Description

This structure contains the Data statistic per QoS flow

Parameters

<i>bearerId</i>	<ul style="list-style-type: none"> • Bearer ID
<i>tx_pkt</i>	<ul style="list-style-type: none"> • number of sent packets for the QoS flow ID
<i>tx_pkt_drp</i>	<ul style="list-style-type: none"> • number of dropped(TX) packets for the QoS flow ID
<i>tx_bytes</i>	<ul style="list-style-type: none"> • number of sent bytes for the QoS flow ID
<i>tx_bytes_drp</i>	<ul style="list-style-type: none"> • number of dropped(TX) bytes for the QoS flow ID

8.1008.2 Field Documentation

8.1008.2.1 `uint32_t unpack_QosFlowStat_t::bearerId`

8.1008.2.2 `uint64_t unpack_QosFlowStat_t::tx_bytes`

8.1008.2.3 `uint64_t unpack_QosFlowStat_t::tx_bytes_drp`

8.1008.2.4 `uint32_t unpack_QosFlowStat_t::tx_pkt`

8.1008.2.5 `uint32_t unpack_QosFlowStat_t::tx_pkt_drp`

8.1009 `unpack_RMTransferStatistics_ind_t` Struct Reference

Data Fields

- [wds_DataUlongTlv TxOkConutTlv](#)
- [wds_DataUlongTlv RxOkConutTlv](#)
- [wds_DataUlongLongTlv TxOkByteCountTlv](#)
- [wds_DataUlongLongTlv RxOkByteCountTlv](#)
- [wds_DataUlongTlv TxDropConutTlv](#)
- [wds_DataUlongTlv RxDropConutTlv](#)

8.1009.1 Detailed Description

WDS Pkt RM Transfer Statistics data structure for individual session

Parameters

<i>TxOkConutTlv</i>	<ul style="list-style-type: none"> Tx Ok Packet Tlv Value.
<i>RxOkConutTlv</i>	<ul style="list-style-type: none"> Rx Ok Packet Tlv Value.
<i>TxOkByte↔CountTlv</i>	<ul style="list-style-type: none"> Tx Ok Byte Count Packet Tlv Value.
<i>RxOkByte↔CountTlv</i>	<ul style="list-style-type: none"> Rx Ok Byte Count Packet Tlv Value.
<i>TxDropConutTlv</i>	<ul style="list-style-type: none"> Tx Drop Count Packet Tlv Value.
<i>RxDropConutTlv</i>	<ul style="list-style-type: none"> Rx Drop Count Packet Tlv Value.

8.1009.2 Field Documentation

8.1009.2.1 `wds_DataULongTlv unpack_RMTransferStatistics_ind_t::RxDropConutTlv`

8.1009.2.2 `wds_DataULongLongTlv unpack_RMTransferStatistics_ind_t::RxOkByteCountTlv`

8.1009.2.3 `wds_DataULongTlv unpack_RMTransferStatistics_ind_t::RxOkConutTlv`

8.1009.2.4 `wds_DataULongTlv unpack_RMTransferStatistics_ind_t::TxDropConutTlv`

8.1009.2.5 `wds_DataULongLongTlv unpack_RMTransferStatistics_ind_t::TxOkByteCountTlv`

8.1009.2.6 `wds_DataULongTlv unpack_RMTransferStatistics_ind_t::TxOkConutTlv`

8.1010 unpack_sms_SendSMS_t Struct Reference

Data Fields

- uint16_t [messageID](#)
- uint32_t [messageFailureCode](#)

8.1010.1 Detailed Description

Parameters

<i>messageID</i>	<ul style="list-style-type: none"> WMS message ID
<i>message↔FailureCode</i>	<ul style="list-style-type: none"> pointer to message failure code. If cause code is not provided, then value will be 0xFFFFFFFF

8.1010.2 Field Documentation

8.1010.2.1 `uint32_t unpack_sms_SendSMS_t::messageFailureCode`

8.1010.2.2 `uint16_t unpack_sms_SendSMS_t::messageID`

8.1011 `unpack_sms_SetNewSMSCallback_ind_t` Struct Reference

Data Fields

- struct [newMTMessageTlv](#) `NewMMTlv`
- struct [transferRouteMessageTlv](#) `TRMessageTlv`
- struct [messageModeTlv](#) `MMTlv`
- struct [sMSEtwSMessageTlv](#) `ETWSTlv`
- struct [eTWSPLMNInfoTlv](#) `ETWSPLMNTlv`
- struct [sMSCAddressTlv](#) `SMSCtlv`
- struct [sMSOnIMSTlv](#) `IMSTlv`

8.1011.1 Detailed Description

Parameters

<i>NewMMTlv</i>	<ul style="list-style-type: none"> • MT message
<i>TRMessageTlv</i>	<ul style="list-style-type: none"> • Transfer Route MT Message • See transferRouteMessageTlv for more information
<i>MMTlv</i>	<ul style="list-style-type: none"> • Message mode • See messageModeTlv for more information
<i>ETWSTlv</i>	<ul style="list-style-type: none"> • ETWS Message • See sMSEtwSMessageTlv for more information
<i>ETWSPLMNTlv</i>	<ul style="list-style-type: none"> • ETWS PLMN Information • See eTWSPLMNInfoTlv for more information
<i>SMSCtlv</i>	<ul style="list-style-type: none"> • SMSC Address • See sMSCAddressTlv for more information
<i>IMSTlv</i>	<ul style="list-style-type: none"> • SMS on IMS • See sMSOnIMSTlv for more information

8.1011.2 Field Documentation

8.1011.2.1 struct eTWSPLMNInfoTlv unpack_sms_SetNewSMSCallback_ind_t::ETWSPLMNTlv

8.1011.2.2 struct sMSEtwsMessageTlv unpack_sms_SetNewSMSCallback_ind_t::ETWSTlv

8.1011.2.3 struct sMSOnIMSTlv unpack_sms_SetNewSMSCallback_ind_t::IMSTlv

8.1011.2.4 struct messageModeTlv unpack_sms_SetNewSMSCallback_ind_t::MMTlv

8.1011.2.5 struct newMTMessageTlv unpack_sms_SetNewSMSCallback_ind_t::NewMMTlv

8.1011.2.6 struct sMSCAddressTlv unpack_sms_SetNewSMSCallback_ind_t::SMSCSTlv

8.1011.2.7 struct transferRouteMessageTlv unpack_sms_SetNewSMSCallback_ind_t::TRMessageTlv

8.1012 unpack_sms_SetNewSMSCallback_t Struct Reference

8.1013 unpack_sms_SLQSDeleteSMS_t Struct Reference

8.1014 unpack_sms_SLQSGetSMS_t Struct Reference

Data Fields

- uint32_t [messageTag](#)
- uint32_t [messageFormat](#)
- uint32_t [messageSize](#)
- uint8_t [message](#) [2048]

8.1014.1 Detailed Description

Parameters

<i>messageTag</i>	<ul style="list-style-type: none"> • Message tag <ul style="list-style-type: none"> – 0 - Read – 1 - Not read – 2 - Mobile originated and sent – 3 - Mobile originated but not yet sent
<i>messageFormat</i>	<ul style="list-style-type: none"> • Message format <ul style="list-style-type: none"> – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
<i>messageSize</i>	
Generated by Doxygen	<ul style="list-style-type: none"> • Upon input the maximum number of bytes that can be written to the message array.

- Upon successful output the actual number of bytes written to the message array.

Parameters

<i>message</i>	<ul style="list-style-type: none"> • The message contents array
----------------	--

8.1014.2 Field Documentation

8.1014.2.1 uint8_t unpack_sms_SLQSGetSMS_t::message[2048]

8.1014.2.2 uint32_t unpack_sms_SLQSGetSMS_t::messageFormat

8.1014.2.3 uint32_t unpack_sms_SLQSGetSMS_t::messageSize

8.1014.2.4 uint32_t unpack_sms_SLQSGetSMS_t::messageTag

8.1015 unpack_sms_SLQSGetSMSList_t Struct Reference

Data Fields

- uint32_t [messageListSize](#)
- [qmiSmsMessageList](#) [messageList](#) [255]

8.1015.1 Detailed Description

Parameters

<i>messageListSize</i>	<ul style="list-style-type: none"> • Upon input the maximum number of elements that the message list array can contain. • Upon successful output the actual number of elements in the message list array.
<i>messageList</i>	<ul style="list-style-type: none"> • Message List • See qmiSmsMessageList for more information

8.1015.2 Field Documentation

8.1015.2.1 qmiSmsMessageList unpack_sms_SLQSGetSMSList_t::messageList[255]

8.1015.2.2 uint32_t unpack_sms_SLQSGetSMSList_t::messageListSize

8.1016 unpack_sms_SLQSMModifySMSStatus_t Struct Reference

8.1017 unpack_sms_SLQSWmsMemoryFullCallBack_ind_t Struct Reference

Data Fields

- [uint8_t storageType](#)
- [uint8_t messageMode](#)

8.1017.1 Detailed Description

Parameters

<i>storageType</i>	<ul style="list-style-type: none">• SMS message storage type<ul style="list-style-type: none">– 0 - UIM - Invalid in case of CDMA device that does not require SIM– 1 - NV
<i>messageMode</i>	<ul style="list-style-type: none">• 0x00 - CDMA, LTE (if network type is CDMA)• 0x01 - GW, LTE (if network type is UMTS)

8.1017.2 Field Documentation

8.1017.2.1 [uint8_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::messageMode](#)

8.1017.2.2 [uint8_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::storageType](#)

8.1018 unpack_swiloc_SwiLocGetAutoStart_t Struct Reference

Data Fields

- [uint8_t function](#)
- [int function_reported](#)
- [uint8_t fix_type](#)
- [int fix_type_reported](#)
- [uint8_t max_time](#)
- [int max_time_reported](#)
- [uint32_t max_dist](#)
- [int max_dist_reported](#)
- [uint32_t fix_rate](#)
- [int fix_rate_reported](#)

8.1018.1 Detailed Description

This structure contains SWI LOC Get Auto Start setting

Parameters

<i>function</i>	<ul style="list-style-type: none"> • Setting to indicate when modem should start an automatic GNSS fix <ul style="list-style-type: none"> – 0 - disabled – 1 - At bootup – 2 - When NMEA port is opened
<i>function_↔ reported</i>	<ul style="list-style-type: none"> • 0 - not reported by modem • 1 - reported by modem
<i>fix_type</i>	<ul style="list-style-type: none"> • Type of GNSS fix: <ul style="list-style-type: none"> – 1 - Default Engine mode – 2 - MS-Based – 3 - MS-Assisted – 4 - Standalone
<i>fix_type_↔ reported</i>	<ul style="list-style-type: none"> • 0 - not reported by modem • 1 - reported by modem
<i>max_time</i>	<ul style="list-style-type: none"> • Maximum time allowed for the receiver to get a fix in seconds • Valid range: 1-255
<i>max_time_↔ reported</i>	<ul style="list-style-type: none"> • 0 - not reported by modem • 1 - reported by modem
<i>max_dist</i>	<ul style="list-style-type: none"> • Maximum uncertainty of a fix measured by distance in meters • Valid range: 1 - 4294967280
<i>max_dist_↔ reported</i>	<ul style="list-style-type: none"> • 0 - not reported by modem • 1 - reported by modem
<i>fix_rate</i>	<ul style="list-style-type: none"> • Time between fixes in seconds • Valid range: 1–65535
<i>fix_rate_↔ reported</i>	<ul style="list-style-type: none"> • 0 - not reported by modem • 1 - reported by modem

8.1018.2 Field Documentation

8.1018.2.1 `uint32_t unpack_swiloc_SwiLocGetAutoStart_t::fix_rate`

8.1018.2.2 `int unpack_swiloc_SwiLocGetAutoStart_t::fix_rate_reported`

8.1018.2.3 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::fix_type

8.1018.2.4 int unpack_swiloc_SwiLocGetAutoStart_t::fix_type_reported

8.1018.2.5 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::function

8.1018.2.6 int unpack_swiloc_SwiLocGetAutoStart_t::function_reported

8.1018.2.7 uint32_t unpack_swiloc_SwiLocGetAutoStart_t::max_dist

8.1018.2.8 int unpack_swiloc_SwiLocGetAutoStart_t::max_dist_reported

8.1018.2.9 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::max_time

8.1018.2.10 int unpack_swiloc_SwiLocGetAutoStart_t::max_time_reported

8.1019 unpack_swima_SLQSOMADMAAlertCallback_ind_t Struct Reference

Data Fields

- uint32_t [eventType](#)
- [unpack_omaDmFotaTlv_t](#) SessionInfoFota
- [unpack_omaDmConfigTlv_t](#) SessionInfoConfig
- [unpack_omaDmNotificationsTlv_t](#) SessionInfoNotification

8.1019.1 Detailed Description

Structure that contains OMA indication information based on eventType Structures for which the event is not valid will have values set to 0

Parameters

<i>eventType</i>	<ul style="list-style-type: none"> • 0x00 - SWIOMA-DM FOTA • 0x01 - SWIOMA-DM Config • 0x02 - SWIOMA-DM Notification
<i>SessionInfo</i> ↔ <i>Fota</i> [OUT]	<ul style="list-style-type: none"> • See unpack_omaDmFotaTlv_t for more information
<i>SessionInfo</i> ↔ <i>Config</i> [OUT]	<ul style="list-style-type: none"> • See unpack_omaDmConfigTlv_t for more information
<i>SessionInfo</i> ↔ <i>Notification</i> [OUT]	<ul style="list-style-type: none"> • See unpack_omaDmNotificationsTlv_t for more information

8.1019.2 Field Documentation

8.1019.2.1 `uint32_t unpack_swima_SLQSOMADMAAlertCallback_ind_t::eventType`

8.1019.2.2 `unpack_omaDmConfigTlv_t unpack_swima_SLQSOMADMAAlertCallback_ind_t::SessionInfoConfig`

8.1019.2.3 `unpack_omaDmFotaTlv_t unpack_swima_SLQSOMADMAAlertCallback_ind_t::SessionInfoFota`

8.1019.2.4 `unpack_omaDmNotificationsTlv_t unpack_swima_SLQSOMADMAAlertCallback_ind_t::SessionInfo←
Notification`

8.1020 `unpack_swima_SLQSOMADMGetSessionInfo_t` Struct Reference

Data Fields

- `uint8_t Status`
- `uint16_t UpdateCompleteStatus`
- `uint8_t Severity`
- `uint16_t SourceLength`
- `uint8_t Source` [255]
- `uint16_t PkgNameLength`
- `uint8_t PkgName` [255]
- `uint16_t PkgDescLength`
- `uint8_t PkgDescription` [255]
- `uint16_t DateLength`
- `uint8_t Date` [255]
- `uint16_t TimeLength`
- `uint8_t Time` [255]
- `uint8_t SessionType`
- `uint8_t SessionState`
- `uint16_t RetryCount`

8.1020.1 Detailed Description

Structure that contains the session type for OMA get session info unpack command Also used as input parameter to specify the size of variable parameters. (ref. notes)

Parameters

<i>Status</i>	<ul style="list-style-type: none"> • 1 Byte parameter indicating status <ul style="list-style-type: none"> – 0x01 - No Firmware available – 0x02 - Query Firmware Download – 0x03 - Firmware Downloading – 0x04 - Firmware Downloaded – 0x05 - Query Firmware Update – 0x06 - Firmware Updating – 0x07 - Firmware Updated
---------------	--

<i>Update↔ CompleteStatus</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Update Complete Status <ul style="list-style-type: none"> – See qaGobiApiTableSwiOMADMUpdateCompleteStatus.h Update Complete Status
<i>Severity</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating severity <ul style="list-style-type: none"> – 0x01 - Mandatory – 0x02 - Optional
<i>SourceLength</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Vendor Name String in Bytes.
<i>Source</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Vendor Name in ASCII • See LIBPACK_MAX_SWIOMA_STR_LEN for more information
<i>PkgNameLength</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Package Name String in Bytes.
<i>PkgName</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Package Name in ASCII • See LIBPACK_MAX_SWIOMA_STR_LEN for more information
<i>PkgDescLength</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Package Description String in Bytes.
<i>PkgDescription</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Package Description in ASCII • See LIBPACK_MAX_SWIOMA_STR_LEN for more information
<i>DateLength</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Package Description String in Bytes.
<i>Date</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Package Description in ASCII • See LIBPACK_MAX_SWIOMA_STR_LEN for more information
<i>TimeLength</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Time String in Bytes.
<i>Time</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Time String in ASCII • See LIBPACK_MAX_SWIOMA_STR_LEN for more information

<i>SessionType</i>	<ul style="list-style-type: none"> • 1 byte parameter reflects the last session started for Sprint <ul style="list-style-type: none"> – 0x00 - No session since boot – 0x01 - Sprint CI-DC Session – 0x02 - Sprint CI-PRL Session – 0x03 - Sprint CI-FUMO Session – 0x04 - Sprint HFA-DC Session – 0x05 - Sprint HFA-PRL Session – 0x06 - Sprint HFA-FUMO Session – 0x07 - Sprint NI Session
<i>SessionState</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating session state <ul style="list-style-type: none"> – 0x01 - idle – 0x02 - active – 0x03 - pending
<i>RetryCount</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating retries left count <ul style="list-style-type: none"> – valid values 0 to 6

8.1020.2 Field Documentation

8.1020.2.1 `uint8_t unpack_swioama_SLQSOMADMGetSessionInfo_t::Date[255]`

8.1020.2.2 `uint16_t unpack_swioama_SLQSOMADMGetSessionInfo_t::DateLength`

8.1020.2.3 `uint16_t unpack_swioama_SLQSOMADMGetSessionInfo_t::PkgDescLength`

8.1020.2.4 `uint8_t unpack_swioama_SLQSOMADMGetSessionInfo_t::PkgDescription[255]`

8.1020.2.5 `uint8_t unpack_swioama_SLQSOMADMGetSessionInfo_t::PkgName[255]`

8.1020.2.6 `uint16_t unpack_swioama_SLQSOMADMGetSessionInfo_t::PkgNameLength`

8.1020.2.7 `uint16_t unpack_swioama_SLQSOMADMGetSessionInfo_t::RetryCount`

8.1020.2.8 `uint8_t unpack_swioama_SLQSOMADMGetSessionInfo_t::SessionState`

8.1020.2.9 `uint8_t unpack_swioama_SLQSOMADMGetSessionInfo_t::SessionType`

8.1020.2.10 `uint8_t unpack_swioama_SLQSOMADMGetSessionInfo_t::Severity`

8.1020.2.11 `uint8_t unpack_swioama_SLQSOMADMGetSessionInfo_t::Source[255]`

8.1020.2.12 uint16_t unpack_swima_SLQSOMADMGetSessionInfo_t::SourceLength

8.1020.2.13 uint8_t unpack_swima_SLQSOMADMGetSessionInfo_t::Status

8.1020.2.14 uint8_t unpack_swima_SLQSOMADMGetSessionInfo_t::Time[255]

8.1020.2.15 uint16_t unpack_swima_SLQSOMADMGetSessionInfo_t::TimeLength

8.1020.2.16 uint16_t unpack_swima_SLQSOMADMGetSessionInfo_t::UpdateCompleteStatus

8.1021 unpack_swima_SLQSOMADMGetSettings_t Struct Reference

Data Fields

- uint32_t [OMADMEEnabled](#)
- uint8_t [FOTAdownload](#)
- uint8_t [FOTAUpdate](#)
- uint8_t [Autosdm](#)
- uint8_t [FwAutoCheck](#)

8.1021.1 Detailed Description

Structure containing the OMA DM settings retrieved from the device

Parameters

<i>OMADM</i> ↔ <i>Enabled</i> [OUT]	<ul style="list-style-type: none"> • Optional 4 byte parameter indicating OMADM service enabled <ul style="list-style-type: none"> – 0x00000001 - Client-initiated device configuration – 0x00000002 - Network-initiated device configuration – 0x00000010 - Client-initiated FUMO – 0x00000020 - Network-initiated FUMO • function SLQSOMADMGetSettings2() returns a default value 0xFFFFFFFF in case this parameter is not returned by the modem.
<i>FOTA</i> ↔ <i>Adownload</i> [O↔ UT]	<ul style="list-style-type: none"> • Optional 1 Byte parameter indicating support for FOTA Automatic download <ul style="list-style-type: none"> – 0x00 - Host permission required before downloading – 0x01 - Automatically start downloading, no host permission required – 0x02 - Automatically start downloading, while not roaming – 0x03 - Automatically reject download – 0x04 - Automatically reject download with “Enterprise Reject Policy” • function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.

<i>FOTAUpdate</i> [OUT]	<ul style="list-style-type: none"> Optional 1 byte parameter indicating FOTA Automatic update <ul style="list-style-type: none"> 0x00 - User permission required before updating firmware 0x01 - No user permission required before updating firmware 0x02 - User permission required, auto update on power up function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.
<i>Autosdm</i> [OUT]	<ul style="list-style-type: none"> Optional 1 byte parameter indicating OMA Automatic UI Alert Response <ul style="list-style-type: none"> 0x00 - Disabled 0x01 - Enabled Accept 0x02 - Enabled Reject function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.
<i>FwAutoCheck</i> [OUT]	<ul style="list-style-type: none"> Optional 1 byte parameter indicating OMA Automatic Check for Firmware Update on Power-Up Response <ul style="list-style-type: none"> 0x00 - Disabled 0x01 - Enabled function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.

8.1021.2 Field Documentation

8.1021.2.1 `uint8_t unpack_swioama_SLQSOMADMGetSettings_t::Autosdm`

8.1021.2.2 `uint8_t unpack_swioama_SLQSOMADMGetSettings_t::FOTAdownload`

8.1021.2.3 `uint8_t unpack_swioama_SLQSOMADMGetSettings_t::FOTAUpdate`

8.1021.2.4 `uint8_t unpack_swioama_SLQSOMADMGetSettings_t::FwAutoCheck`

8.1021.2.5 `uint32_t unpack_swioama_SLQSOMADMGetSettings_t::OMADMEEnabled`

8.1022 unpack_swioama_SLQSOMADMStartSession_t Struct Reference

Data Fields

- `uint32_t` [FwAvailability](#)

8.1022.1 Detailed Description

Structure that contains the responses for OMA start session command

Parameters

<i>pFw</i> <i>Availability[OUT]</i>	<ul style="list-style-type: none"> OMA-DM CHECK FW Available <ul style="list-style-type: none"> 0x00000001 - FW Available. For CIDC and CIPRL, this value will be returned by the modem. CIDC and CIPRL are asynchronous OMADM sessions. 0x00000002 - FW Not Available 0x00000003 - FW Check Timed Out
--	---

8.1022.2 Field Documentation

8.1022.2.1 uint32_t unpack_swoma_SLQSOMADMStartSession_t::FwAvailability

8.1023 unpack_uim_ChangePin_t Struct Reference

Data Fields

- [uim_remainingRetries](#) * [pRemainingRetries](#)
- [uim_encryptedPIN1](#) * [pEncryptedPIN1](#)
- uint32_t * [pIndicationToken](#)
- uint16_t [Tlvresult](#)

8.1023.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

<i>pRemainingRetries(optional)</i>	<ul style="list-style-type: none"> See uim_remainingRetries for more information.
<i>pEncryptedPIN1(optional)</i>	<ul style="list-style-type: none"> See uim_encryptedPIN1 for more information.
<i>pIndicationToken(optional)</i>	<ul style="list-style-type: none"> Response in Indication. When this TLV is present, it indicates that the result is provided in a subsequent indication. 0xFFFFFFFF, if unavailable

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.1023.2 Field Documentation

8.1023.2.1 [uim_encryptedPIN1](#) * [unpack_uim_ChangePin_t::pEncryptedPIN1](#)

8.1023.2.2 `uint32_t* unpack_uim_ChangePin_t::pIndicationToken`

8.1023.2.3 `uim_remainingRetries* unpack_uim_ChangePin_t::pRemainingRetries`

8.1023.2.4 `uint16_t unpack_uim_ChangePin_t::Tlvresult`

8.1024 unpack_uim_GetCardStatus_t Struct Reference

Data Fields

- [uim_cardStatus](#) * [pCardStatus](#)
- [uim_hotSwapStatus](#) * [pHotSwapStatus](#)
- `uint16_t` [Tlvresult](#)

8.1024.1 Detailed Description

This structure contains information of the response parameters associated with a Get Card Status API.

Parameters

<i>pCard↔ Status(optional)</i>	<ul style="list-style-type: none"> • See uim_cardStatus for more information.
<i>pHotSwap↔ Status(optional)</i>	<ul style="list-style-type: none"> • See uim_hotSwapStatus for more information.

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.1024.2 Field Documentation

8.1024.2.1 `uim_cardStatus* unpack_uim_GetCardStatus_t::pCardStatus`

8.1024.2.2 `uim_hotSwapStatus* unpack_uim_GetCardStatus_t::pHotSwapStatus`

8.1024.2.3 `uint16_t unpack_uim_GetCardStatus_t::Tlvresult`

8.1025 unpack_uim_ReadTransparent_t Struct Reference

Data Fields

- [uim_cardResult](#) * [pCardResult](#)
- [uim_readResult](#) * [pReadResult](#)
- `uint32_t` * [pIndicationToken](#)
- `uint8_t` * [pEncryptedData](#)
- `uint16_t` [Tlvresult](#)

8.1025.1 Detailed Description

This structure contains information of the response parameters associated with a Read Transparent API.

Parameters

<i>pCardResult</i>	<ul style="list-style-type: none"> See cardResult for more information.
<i>pReadResult</i>	<ul style="list-style-type: none"> See readResult for more information.
<i>pIndication</i> ↔ <i>Token(optional)</i>	<ul style="list-style-type: none"> Response in Indication. When this TLV is present, it indicates that the result must be provided in a subsequent indication.
<i>pEncrypted</i> ↔ <i>Data(optional)</i>	<ul style="list-style-type: none"> Encrypted Data. Indicates whether the data from the card passed in read_result is encrypted.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.1025.2 Field Documentation

8.1025.2.1 `uim_cardResult*` `unpack_uim_ReadTransparent_t::pCardResult`

8.1025.2.2 `uint8_t*` `unpack_uim_ReadTransparent_t::pEncryptedData`

8.1025.2.3 `uint32_t*` `unpack_uim_ReadTransparent_t::pIndicationToken`

8.1025.2.4 `uim_readResult*` `unpack_uim_ReadTransparent_t::pReadResult`

8.1025.2.5 `uint16_t` `unpack_uim_ReadTransparent_t::Tlvresult`

8.1026 unpack_uim_SetPinProtection_t Struct Reference

Data Fields

- `uim_remainingRetries` * `pRemainingRetries`
- `uim_encryptedPIN1` * `pEncryptedPIN1`
- `uint32_t` * `pIndicationToken`
- `uint16_t` `Tlvresult`

8.1026.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

<i>pRemainingRetries</i> (optional)	<ul style="list-style-type: none"> See uim_remainingRetries for more information.
<i>pEncryptedPIN1</i> (optional)	<ul style="list-style-type: none"> See uim_encryptedPIN1 for more information.
<i>pIndicationToken</i> (optional)	<ul style="list-style-type: none"> Response in Indication. When this TLV is present, it indicates that the result is provided in a subsequent indication. 0xFFFFFFFF, if unavailable

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.1026.2 Field Documentation

8.1026.2.1 `uim_encryptedPIN1* unpack_uim_SetPinProtection_t::pEncryptedPIN1`

8.1026.2.2 `uint32_t* unpack_uim_SetPinProtection_t::pIndicationToken`

8.1026.2.3 `uim_remainingRetries* unpack_uim_SetPinProtection_t::pRemainingRetries`

8.1026.2.4 `uint16_t unpack_uim_SetPinProtection_t::Tlvresult`

8.1027 unpack_uim_SetUimSlotStatusChangeCallback_ind_t Struct Reference**Data Fields**

- [slots_t slotsstatusChange](#)
- `uint8_t` [bNumberOfPhySlots](#)

8.1027.1 Detailed Description

Structure consist of card status params

Parameters

<i>slotsstatusChange</i>	<ul style="list-style-type: none"> See slot_t for more information
<i>bNumberOfPhySlots</i>	<ul style="list-style-type: none"> Number of Physical Slot(s)

8.1027.2 Field Documentation

8.1027.2.1 `uint8_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::bNumberOfPhySlots`

8.1027.2.2 `slots_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::slotsstatusChange`

8.1028 unpack_uim_SLQSUIEventRegister_t Struct Reference

Data Fields

- `uint32_t eventMask`

8.1028.1 Detailed Description

Parameters

<i>eventMask</i>	<ul style="list-style-type: none">- bit 0 - card status• bit 1 - SAP connection• bit 4 - physical slot status
------------------	---

8.1028.2 Field Documentation

8.1028.2.1 `uint32_t unpack_uim_SLQSUIEventRegister_t::eventMask`

8.1029 unpack_uim_SLQSUIGetSlotsStatus_t Struct Reference

Data Fields

- `uint8_t * pNumberOfPhySlot`
- `slots_t * pUimSlotsStatus`

8.1029.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

<i>pNumberOfPhySlot</i>	<ul style="list-style-type: none">• Number of sets of the Slot Status.
<i>pUimSlotsStatus</i>	<ul style="list-style-type: none">• Slots Status See slots_t for more information..

8.1029.2 Field Documentation

8.1029.2.1 `uint8_t*` `unpack_uim_SLQSUIMGetSlotsStatus_t::pNumberOfPhySlot`

8.1029.2.2 `slots_t*` `unpack_uim_SLQSUIMGetSlotsStatus_t::pUimSlotsStatus`

8.1030 `unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t` Struct Reference

Data Fields

- [uim_cardStatus](#) * [pCardStatus](#)

8.1030.1 Detailed Description

This structure contains information about Status change callback.

Parameters

<i>pCardStatus</i>	Card Status <ul style="list-style-type: none">• See uim_cardStatus for more information.
--------------------	--

8.1030.2 Field Documentation

8.1030.2.1 `uim_cardStatus*` `unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t::pCardStatus`

8.1031 `unpack_uim_UnblockPin_t` Struct Reference

Data Fields

- [uim_remainingRetries](#) * [pRemainingRetries](#)
- [uim_encryptedPIN1](#) * [pEncryptedPIN1](#)
- `uint32_t` * [pIndicationToken](#)
- `uint16_t` [Tlvresult](#)

8.1031.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

<i>pRemainingRetries</i> (optional)	<ul style="list-style-type: none"> See uim_remainingRetries for more information.
<i>pEncryptedPIN1</i> (optional)	<ul style="list-style-type: none"> See uim_encryptedPIN1 for more information.
<i>pIndicationToken</i> (optional)	<ul style="list-style-type: none"> Response in Indication. When this TLV is present, it indicates that the result is provided in a subsequent indication. 0xFFFFFFFF, if unavailable

8.1031.2 Field Documentation

8.1031.2.1 [uim_encryptedPIN1](#)* [unpack_uim_UnblockPin_t::pEncryptedPIN1](#)8.1031.2.2 [uint32_t](#)* [unpack_uim_UnblockPin_t::pIndicationToken](#)8.1031.2.3 [uim_remainingRetries](#)* [unpack_uim_UnblockPin_t::pRemainingRetries](#)8.1031.2.4 [uint16_t](#) [unpack_uim_UnblockPin_t::Tlvresult](#)

8.1032 unpack_uim_VerifyPin_t Struct Reference

Data Fields

- [uim_remainingRetries](#) * [pRemainingRetries](#)
- [uim_encryptedPIN1](#) * [pEncryptedPIN1](#)
- [uint32_t](#) * [pIndicationToken](#)
- [uint16_t](#) [Tlvresult](#)

8.1032.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

<i>pRemainingRetries</i> (optional)	<ul style="list-style-type: none"> See uim_remainingRetries for more information.
<i>pEncryptedPIN1</i> (optional)	<ul style="list-style-type: none"> See uim_encryptedPIN1 for more information.
<i>pIndicationToken</i> (optional)	<ul style="list-style-type: none"> Response in Indication. When this TLV is present, it indicates that the result is provided in a subsequent indication. 0xFFFFFFFF, if unavailable

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.1032.2 Field Documentation

8.1032.2.1 `uim_encryptedPIN1*` `unpack_uim_VerifyPin_t::pEncryptedPIN1`

8.1032.2.2 `uint32_t*` `unpack_uim_VerifyPin_t::pIndicationToken`

8.1032.2.3 `uim_remainingRetries*` `unpack_uim_VerifyPin_t::pRemainingRetries`

8.1032.2.4 `uint16_t` `unpack_uim_VerifyPin_t::Tlvresult`

8.1033 `unpack_wds_DHCPv4ClientLease_ind_t` Struct Reference

Data Fields

- [wds_DHCPProfileIdTlv](#) `ProfileIdTlv`
- [wds_DHCPLeaseStateTlv](#) `DHCPv4LeaseStateTlv`
- [wds_IPv4AdTlv](#) `IPv4AddrTlv`
- [wds_DHCPLeaseOptTlv](#) `DHCPv4LeaseOptTlv`

8.1033.1 Field Documentation

8.1033.1.1 `wds_DHCPLeaseOptTlv` `unpack_wds_DHCPv4ClientLease_ind_t::DHCPv4LeaseOptTlv`

8.1033.1.2 `wds_DHCPLeaseStateTlv` `unpack_wds_DHCPv4ClientLease_ind_t::DHCPv4LeaseStateTlv`

8.1033.1.3 `wds_IPv4AdTlv` `unpack_wds_DHCPv4ClientLease_ind_t::IPv4AddrTlv`

8.1033.1.4 `wds_DHCPProfileIdTlv` `unpack_wds_DHCPv4ClientLease_ind_t::ProfileIdTlv`

8.1034 `unpack_wds_GetAutoconnect_t` Struct Reference

Data Fields

- `uint32_t *` `psetting`

8.1034.1 Detailed Description

auto connect data session setting parameter.

Parameters

<i>pSetting</i>	<ul style="list-style-type: none"> • NDIS auto connect setting <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled
-----------------	--

8.1034.2 Field Documentation

8.1034.2.1 `uint32_t*` `unpack_wds_GetAutoconnect_t::psetting`

8.1035 unpack_wds_GetByteTotals_t Struct Reference

Data Fields

- `uint64_t` * [pTXTotalBytes](#)
- `uint64_t` * [pRXTotalBytes](#)

8.1035.1 Detailed Description

Parameters

<i>pTXTotalBytes</i>	<ul style="list-style-type: none"> • Bytes transmitted without error
<i>pRXTotalBytes</i>	<ul style="list-style-type: none"> • Bytes received without error

8.1035.2 Field Documentation

8.1035.2.1 `uint64_t*` `unpack_wds_GetByteTotals_t::pRXTotalBytes`

8.1035.2.2 `uint64_t*` `unpack_wds_GetByteTotals_t::pTXTotalBytes`

8.1036 unpack_wds_GetConnectionRate_t Struct Reference

Data Fields

- `uint32_t` [currentChannelTXRate](#)
- `uint32_t` [currentChannelRXRate](#)
- `uint32_t` [maxChannelTXRate](#)
- `uint32_t` [maxChannelRXRate](#)

8.1036.1 Detailed Description

Parameters

<i>current</i> ↔ <i>ChannelTXRate</i>	Instantaneous channel Tx rate
<i>current</i> ↔ <i>ChannelRXRate</i>	Instantaneous channel Rx rate
<i>maxChannelT</i> ↔ <i>XRate</i>	Maximum Tx rate
<i>maxChannelR</i> ↔ <i>XRate</i>	Maximum Rx rate

8.1036.2 Field Documentation

8.1036.2.1 `uint32_t unpack_wds_GetConnectionRate_t::currentChannelRXRate`

8.1036.2.2 `uint32_t unpack_wds_GetConnectionRate_t::currentChannelTXRate`

8.1036.2.3 `uint32_t unpack_wds_GetConnectionRate_t::maxChannelRXRate`

8.1036.2.4 `uint32_t unpack_wds_GetConnectionRate_t::maxChannelTXRate`

8.1037 `unpack_wds_GetDataBearerTechnology_t` Struct Reference

Data Fields

- `uint32_t * pDataBearer`

8.1037.1 Detailed Description

Parameters

<i>pDataBearer</i> ↔ <i>OUT</i>	<ul style="list-style-type: none"> • Data bearer technology <ul style="list-style-type: none"> – 0x01 - CDMA2000 1x – 0x02 - CDMA 1xEV-DO Rev 0 – 0x03 - GSM – 0x04 - UMTS – 0x05 - CDMA2000 HRPD (1xEV-DO Rev A) – 0x06 - EDGE – 0x07 - HSDPA AND WCDMA – 0x08 - WCDMA AND HSUPA – 0x09 - HSDPA AND HSUPA – 0x0A - LTE – 0x0B - CDMA2000 EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA – 0x10 - HSDPA+ and 64QAM
	<ul style="list-style-type: none"> – 0x11 - HSDPA+, 64QAM and HSUPA – 0x12 - TDSCDMA – 0x13 - TDSCDMA and HSDPA – 0xFF - Unknown

8.1037.2 Field Documentation

8.1037.2.1 uint32_t* unpack_wds_GetDataBearerTechnology_t::pDataBearer

8.1038 unpack_wds_GetDefaultProfile_t Struct Reference

Data Fields

- uint32_t [pdptype](#)
- uint32_t [ipaddr](#)
- uint32_t [pridns](#)
- uint32_t [secdns](#)
- uint16_t [ipaddrv6](#)
- uint16_t [pridnsv6](#)
- uint16_t [secdnsv6](#)
- uint32_t [auth](#)
- uint8_t [namesize](#)
- int8_t [name](#) [255]
- uint8_t [apnsize](#)
- int8_t [apnname](#) [255]
- uint8_t [usersize](#)
- int8_t [username](#) [255]

8.1038.1 Detailed Description

Parameters

<i>pdptype</i>	pdp type
<i>ipaddr</i>	ip address
<i>pridns</i>	primary dns
<i>secdns</i>	secondry dns
<i>ipaddrv6</i>	ip address v6
<i>pridnsv6</i>	primary dns v6
<i>secdnsv6</i>	secondry dns v6
<i>namesize</i>	profile name size
<i>name</i>	profile name
<i>apnsize</i>	apn size
<i>apnname</i>	apn name
<i>usersize</i>	username size
<i>username</i>	username

8.1038.2 Field Documentation

8.1038.2.1 int8_t unpack_wds_GetDefaultProfile_t::apnname[255]

8.1038.2.2 uint8_t unpack_wds_GetDefaultProfile_t::apnsize

8.1038.2.3 uint32_t unpack_wds_GetDefaultProfile_t::auth

- 8.1038.2.4 `uint32_t unpack_wds_GetDefaultProfile_t::ipaddr`
- 8.1038.2.5 `uint16_t unpack_wds_GetDefaultProfile_t::ipaddrv6`
- 8.1038.2.6 `int8_t unpack_wds_GetDefaultProfile_t::name[255]`
- 8.1038.2.7 `uint8_t unpack_wds_GetDefaultProfile_t::namesize`
- 8.1038.2.8 `uint32_t unpack_wds_GetDefaultProfile_t::pdptype`
- 8.1038.2.9 `uint32_t unpack_wds_GetDefaultProfile_t::pridns`
- 8.1038.2.10 `uint16_t unpack_wds_GetDefaultProfile_t::pridnsv6`
- 8.1038.2.11 `uint32_t unpack_wds_GetDefaultProfile_t::secdns`
- 8.1038.2.12 `uint16_t unpack_wds_GetDefaultProfile_t::secdnsv6`
- 8.1038.2.13 `int8_t unpack_wds_GetDefaultProfile_t::username[255]`
- 8.1038.2.14 `uint8_t unpack_wds_GetDefaultProfile_t::usersize`

8.1039 `unpack_wds_GetDefaultProfileNum_t` Struct Reference

Data Fields

- `uint8_t` [index](#)

8.1039.1 Detailed Description

Parameters

<i>index</i>	profile index
--------------	---------------

8.1039.2 Field Documentation

- 8.1039.2.1 `uint8_t unpack_wds_GetDefaultProfileNum_t::index`

8.1040 `unpack_wds_GetDormancyState_t` Struct Reference

Data Fields

- `uint32_t` [dormancyState](#)

8.1040.1 Detailed Description

Parameters

<i>dormancyState</i>	dormancy status
----------------------	-----------------

8.1040.2 Field Documentation

8.1040.2.1 `uint32_t unpack_wds_GetDormancyState_t::dormancyState`

8.1041 unpack_wds_GetLastMobileIPError_t Struct Reference

Data Fields

- `uint32_t error`

8.1041.1 Detailed Description

Parameters

<i>error</i>	last mip status 0-success >0- error code
--------------	--

8.1041.2 Field Documentation

8.1041.2.1 `uint32_t unpack_wds_GetLastMobileIPError_t::error`

8.1042 unpack_wds_GetMobileIP_t Struct Reference

Data Fields

- `uint32_t mipMode`

8.1042.1 Detailed Description

Parameters

<i>mipMode</i>	mobile IP mode
----------------	----------------

8.1042.2 Field Documentation

8.1042.2.1 `uint32_t unpack_wds_GetMobileIP_t::mipMode`

8.1043 unpack_wds_GetMobileIPProfile_t Struct Reference

Data Fields

- uint8_t [enabled](#)
- uint32_t [address](#)
- uint32_t [primaryHA](#)
- uint32_t [secondaryHA](#)
- uint8_t [revTunneling](#)
- uint8_t [naiSize](#)
- int8_t [NAI](#) [255]
- uint32_t [HASPI](#)
- uint32_t [AAASPI](#)
- uint32_t [HASState](#)
- uint32_t [AAASState](#)

8.1043.1 Detailed Description

Parameters

<i>enabled</i>	mobile ip profile state enabled/disabled
<i>address</i>	mobile ip profile home address
<i>primaryHA</i>	mobile ip profile home agent primary
<i>secondaryHA</i>	mobile ip profile secondary home agent address
<i>revTunneling</i>	mobile ip profile rev tunneling
<i>naiSize</i>	mobile ip profile NAI size
<i>NAI</i>	NAI string in ASCII text.
<i>HASPI</i>	HA security parameter index.
<i>AAASPI</i>	AAA server security parameter index.
<i>HASState</i>	Mobile IP Profile HA Key State
<i>AAASState</i>	Mobile IP Profile AAA Key State

8.1043.2 Field Documentation

8.1043.2.1 uint32_t unpack_wds_GetMobileIPProfile_t::AAASPI

8.1043.2.2 uint32_t unpack_wds_GetMobileIPProfile_t::AAASState

8.1043.2.3 uint32_t unpack_wds_GetMobileIPProfile_t::address

8.1043.2.4 uint8_t unpack_wds_GetMobileIPProfile_t::enabled

8.1043.2.5 uint32_t unpack_wds_GetMobileIPProfile_t::HASPI

8.1043.2.6 uint32_t unpack_wds_GetMobileIPProfile_t::HASState

8.1043.2.7 int8_t unpack_wds_GetMobileIPProfile_t::NAI[255]

8.1043.2.8 uint8_t unpack_wds_GetMobileIPProfile_t::naiSize

8.1043.2.9 uint32_t unpack_wds_GetMobileIPProfile_t::primaryHA

8.1043.2.10 uint8_t unpack_wds_GetMobileIPProfile_t::revTunneling

8.1043.2.11 uint32_t unpack_wds_GetMobileIPProfile_t::secondaryHA

8.1044 unpack_wds_GetPacketStatistics_t Struct Reference

Data Fields

- uint32_t * [pTXPacketSuccesses](#)
- uint32_t * [pRXPacketSuccesses](#)
- uint32_t * [pTXPacketErrors](#)
- uint32_t * [pRXPacketErrors](#)
- uint32_t * [pTXPacketOverflows](#)
- uint32_t * [pRXPacketOverflows](#)
- uint64_t * [pTXOkBytesCount](#)
- uint64_t * [pRXOkBytesCount](#)
- uint64_t * [pTXOKBytesLastCall](#)
- uint64_t * [pRXOKBytesLastCall](#)
- uint32_t * [pTXDroppedCount](#)
- uint32_t * [pRXDroppedCount](#)

8.1044.1 Detailed Description

Parameters

<i>pTXPacket↔ Successes</i>	<ul style="list-style-type: none"> • No of transmitted Packets without error.
<i>pRXPacket↔ Successes</i>	<ul style="list-style-type: none"> • No of received Packets without error.
<i>pTXPacketErrors</i>	<ul style="list-style-type: none"> • Number of outgoing packets with framing errors.
<i>pRXPacket↔ Errors</i>	<ul style="list-style-type: none"> • Number of incoming packets with framing errors.
<i>pTXPacket↔ Overflows</i>	<ul style="list-style-type: none"> • Number of packets dropped because Tx buffer overflowed (out of memory).
<i>pRXPacket↔ Overflows</i>	<ul style="list-style-type: none"> • Number of packets dropped because Rx buffer overflowed (out of memory).
<i>pTXOkBytes↔ Count</i>	<ul style="list-style-type: none"> • No of bytes transmitted without error.
<i>pRXOkBytes↔ Count</i>	<ul style="list-style-type: none"> • No of bytes received without error.

<i>pTXOKBytes</i> ↔ <i>LastCall</i>	<ul style="list-style-type: none"> No of bytes transmitted without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support
<i>pRXOKBytes</i> ↔ <i>LastCall</i>	<ul style="list-style-type: none"> Number of bytes received without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support
<i>pTXDropped</i> ↔ <i>Count</i>	<ul style="list-style-type: none"> Number of outgoing packets dropped.
<i>pRXDropped</i> ↔ <i>Count</i>	<ul style="list-style-type: none"> Number of incoming packets dropped.

8.1044.2 Field Documentation

8.1044.2.1 `uint32_t* unpack_wds_GetPacketStatistics_t::pRXDroppedCount`

8.1044.2.2 `uint64_t* unpack_wds_GetPacketStatistics_t::pRXOkBytesCount`

8.1044.2.3 `uint64_t* unpack_wds_GetPacketStatistics_t::pRXOKBytesLastCall`

8.1044.2.4 `uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketErrors`

8.1044.2.5 `uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketOverflows`

8.1044.2.6 `uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketSuccesses`

8.1044.2.7 `uint32_t* unpack_wds_GetPacketStatistics_t::pTXDroppedCount`

8.1044.2.8 `uint64_t* unpack_wds_GetPacketStatistics_t::pTXOkBytesCount`

8.1044.2.9 `uint64_t* unpack_wds_GetPacketStatistics_t::pTXOKBytesLastCall`

8.1044.2.10 `uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketErrors`

8.1044.2.11 `uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketOverflows`

8.1044.2.12 `uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketSuccesses`

8.1045 `unpack_wds_GetPacketStatus_t` Struct Reference

Data Fields

- `uint32_t tXPacketSuccesses`

- [uint32_t rXPacketSuccesses](#)
- [uint32_t tXPacketErrors](#)
- [uint32_t rXPacketErrors](#)
- [uint32_t tXPacketOverflows](#)
- [uint32_t rXPacketOverflows](#)
- [uint64_t tXOkBytesCount](#)
- [uint64_t rXOkBytesCount](#)
- [uint64_t tXOKBytesLastCall](#)
- [uint64_t rXOKBytesLastCall](#)
- [uint32_t tXDroppedCount](#)
- [uint32_t rXDroppedCount](#)

8.1045.1 Detailed Description

Parameters

<i>tXPacket↔ Successes</i>	Tx Packets OK
<i>rXPacket↔ Successes</i>	Rx Packets OK
<i>tXPacketErrors</i>	Tx Packet Errors
<i>rXPacketErrors</i>	Rx Packet Errors
<i>tXPacket↔ Overflows</i>	Tx Overflows
<i>rXPacket↔ Overflows</i>	Rx Overflows
<i>tXOkBytesCount</i>	Tx Bytes OK
<i>rXOkBytesCount</i>	Rx Bytes OK
<i>tXOKBytes↔ LastCall</i>	Last call Tx Bytes OK
<i>rXOKBytes↔ LastCall</i>	Last call Rx Bytes OK
<i>tXDroppedCount</i>	Tx Packets Dropped
<i>rXDroppedCount</i>	Rx Packets Dropped

8.1045.2 Field Documentation

8.1045.2.1 `uint32_t unpack_wds_GetPacketStatus_t::rXDroppedCount`

8.1045.2.2 `uint64_t unpack_wds_GetPacketStatus_t::rXOkBytesCount`

8.1045.2.3 `uint64_t unpack_wds_GetPacketStatus_t::rXOKBytesLastCall`

8.1045.2.4 `uint32_t unpack_wds_GetPacketStatus_t::rXPacketErrors`

8.1045.2.5 `uint32_t unpack_wds_GetPacketStatus_t::rXPacketOverflows`

8.1045.2.6 `uint32_t unpack_wds_GetPacketStatus_t::rXPacketSuccesses`

8.1045.2.7 `uint32_t unpack_wds_GetPacketStatus_t::tXDroppedCount`

8.1045.2.8 `uint64_t unpack_wds_GetPacketStatus_t::tXOkBytesCount`

8.1045.2.9 `uint64_t unpack_wds_GetPacketStatus_t::tXOkBytesLastCall`

8.1045.2.10 `uint32_t unpack_wds_GetPacketStatus_t::tXPacketErrors`

8.1045.2.11 `uint32_t unpack_wds_GetPacketStatus_t::tXPacketOverflows`

8.1045.2.12 `uint32_t unpack_wds_GetPacketStatus_t::tXPacketSuccesses`

8.1046 `unpack_wds_GetSessionDuration_t` Struct Reference

Data Fields

- `uint64_t` [callDuration](#)

8.1046.1 Detailed Description

Parameters

<i>callDuration</i>	call duration in milliseconds
---------------------	-------------------------------

8.1046.2 Field Documentation

8.1046.2.1 `uint64_t unpack_wds_GetSessionDuration_t::callDuration`

8.1047 `unpack_wds_GetSessionState_t` Struct Reference

Data Fields

- `uint32_t` [connectionStatus](#)

8.1047.1 Detailed Description

Parameters

<i>connection↔ Status</i>	state of the current packet data session
---	--

8.1047.2 Field Documentation

8.1047.2.1 `uint32_t unpack_wds_GetSessionState_t::connectionStatus`

8.1048 unpack_wds_RMSetTransferStatistics_t Struct Reference

8.1049 unpack_wds_SetMobileIPProfile_t Struct Reference

8.1050 unpack_wds_SLQSCreateProfile_t Struct Reference

Data Fields

- [PackCreateProfileOut](#) * [pCreateProfileOut](#)
- [uint8_t](#) * [pProfileID](#)
- [uint16_t](#) [Tlvresult](#)

8.1050.1 Detailed Description

Parameters

<i>profile</i>	type
<i>profile</i>	index
<i>extended</i>	error

8.1050.2 Field Documentation

8.1050.2.1 [PackCreateProfileOut](#)* [unpack_wds_SLQSCreateProfile_t::pCreateProfileOut](#)

8.1050.2.2 [uint8_t](#)* [unpack_wds_SLQSCreateProfile_t::pProfileID](#)

8.1050.2.3 [uint16_t](#) [unpack_wds_SLQSCreateProfile_t::Tlvresult](#)

8.1051 unpack_wds_SLQSDeleteProfile_t Struct Reference

Data Fields

- [uint16_t](#) [extendedErrorCode](#)

8.1051.1 Detailed Description

Parameters

<i>extendedError↔ Code</i>	extended error code
--------------------------------	---------------------

8.1051.2 Field Documentation

8.1051.2.1 [uint16_t](#) [unpack_wds_SLQSDeleteProfile_t::extendedErrorCode](#)

8.1052 unpack_wds_SLQSGet3GPPConfigItem_t Struct Reference

Data Fields

- uint16_t [LTEAttachProfile](#)
- uint16_t [profileList](#) [5]
- uint8_t [defaultPDNEnabled](#)
- uint8_t [_3gppRelease](#)
- uint16_t [LTEAttachProfileList](#) [24]
- uint16_t [LTEAttachProfileListLen](#)

8.1052.1 Detailed Description

Parameters

	<i>pLTEAttachProfile</i>	<ul style="list-style-type: none"> • Optional parameter • LTE Attach Profile <ul style="list-style-type: none"> – points to a single WORD Value indicating the attached LTE Profile – Optional parameter with possible values 1-16 (EM/MC73xx or earlier) • This setting is deprecated on MC/EM74xx
	<i>profileList</i>	<p>Profile List</p> <ul style="list-style-type: none"> • an array of 4 profile configurations • Each element points to a single WORD value indicating profile • Optional parameter with possible values <ul style="list-style-type: none"> – 1 - 16 (MC/EM73xx and before) – 1 - 24 (MC/EM74xx and onwards) • function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present • Note: the 5th entry is currently ignored, please set it to zero
out	<i>defaultPDNEnabled</i>	<ul style="list-style-type: none"> • 0 - disabled • 1 - enabled
out	<i>_3gppRelease</i>	<p>3GPP release</p> <ul style="list-style-type: none"> • 0 - Release_99 • 1 - Release_5 • 2 - Release_6 • 3 - Release_7 • 4 - Release_8 • 5 - Release_9 (In 9x30 and toward) • 6 - Release_10 (In 9x30 and toward) • 7 - Release_11 (In 9x30 and toward)
out	<i>LTEAttachProfileList</i>	<ul style="list-style-type: none"> • pointer to WORD array indicating LTE Attach Profile List <ul style="list-style-type: none"> – Optional parameter – possible values: 1-24 – This setting is only supported for MC/EM74xx onwards – Please provide attach profiles in order of decreasing priority in this list.

in, out	<i>LTEAttachProfileListLen</i>	<ul style="list-style-type: none"> Number of element in pLTEAttachProfileList <ul style="list-style-type: none"> valid range: 1-24 This setting is only supported for MC/EM74xx onwards
---------	--------------------------------	---

8.1052.2 Field Documentation

8.1052.2.1 `uint8_t unpack_wds_SLQSGet3GPPConfigItem_t::_3gppRelease`

8.1052.2.2 `uint8_t unpack_wds_SLQSGet3GPPConfigItem_t::defaultPDNEnabled`

8.1052.2.3 `uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfile`

8.1052.2.4 `uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfileList[24]`

8.1052.2.5 `uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfileListLen`

8.1052.2.6 `uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::profileList[5]`

8.1053 unpack_wds_SLQSGetCurrDataSystemStat_t Struct Reference

Data Fields

- `uint8_t prefNetwork`
- `uint8_t networkInfoLen`
- `currNetworkInfo currNetworkInfo [255]`

8.1053.1 Detailed Description

Parameters

<i>prefNetwork</i>	preferred network
<i>networkInfoLen</i>	number of set of <code>currNetworkInfo</code> elements
<i>currNetworkInfo</i>	current network information.

8.1053.2 Field Documentation

8.1053.2.1 `currNetworkInfo unpack_wds_SLQSGetCurrDataSystemStat_t::currNetworkInfo[255]`

8.1053.2.2 `uint8_t unpack_wds_SLQSGetCurrDataSystemStat_t::networkInfoLen`

8.1053.2.3 `uint8_t unpack_wds_SLQSGetCurrDataSystemStat_t::prefNetwork`

8.1054 unpack_wds_SLQSGetCurrentChannelRate_t Struct Reference

Data Fields

- uint32_t [current_channel_tx_rate](#)
- uint32_t [current_channel_rx_rate](#)
- uint32_t [max_channel_tx_rate](#)
- uint32_t [max_channel_rx_rate](#)

8.1054.1 Detailed Description

Parameters

<i>current</i> ↔ <i>channel_tx_rate</i>	<ul style="list-style-type: none"> • Current Channel Tx Rate. • Instantaneous channel Tx rate in bits per second. • In 9x15, this is the total current channel rate for all PDNs combined. • In 9x30 and later, this is the channel rate for a specific PDN.
<i>current</i> ↔ <i>channel_rx_rate</i>	<ul style="list-style-type: none"> • Current Channel Rx Rate. • Instantaneous channel Rx rate in bits per second. • In 9x15, this is the total current channel rate for all PDNs combined. • In 9x30 and later, this is the channel rate for a specific PDN
<i>max_channel</i> ↔ <i>tx_rate</i>	<ul style="list-style-type: none"> • Max Channel Tx Rate. • Maximum total Tx rate that modem is able to support in current serving system in bits per second. • In 9x15, this is a default hard coded value for the current serving system.
<i>max_channel</i> ↔ <i>rx_rate</i>	<ul style="list-style-type: none"> • Max Channel Rx Rate. • Maximum total Rx rate that modem is able to support in current serving system in bits per second. • In 9x15, this is a default hard coded value for the current serving system.

8.1054.2 Field Documentation

8.1054.2.1 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::current_channel_rx_rate

8.1054.2.2 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::current_channel_tx_rate

8.1054.2.3 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::max_channel_rx_rate

8.1054.2.4 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::max_channel_tx_rate

8.1055 unpack_wds_SLQSGetDataBearerTechnology_t Struct Reference

Data Fields

- [uint8_t dataBearerMask](#)
- [qmiWSDDataBearerTechnology curDataBearerTechnology](#)
- [qmiWSDDataBearerTechnology lastCallDataBearerTechnology](#)

8.1055.1 Detailed Description

Parameters

<i>dataBearerMask</i>	bit mask indicates bearer info is for current and/or last call
<i>curDataBearerTechnology</i>	current data bearer technology value
<i>lastCallDataBearerTechnology</i>	last call data bearer technology value

8.1055.2 Field Documentation

8.1055.2.1 [qmiWSDDataBearerTechnology unpack_wds_SLQSGetDataBearerTechnology_t::curDataBearerTechnology](#)

8.1055.2.2 [uint8_t unpack_wds_SLQSGetDataBearerTechnology_t::dataBearerMask](#)

8.1055.2.3 [qmiWSDDataBearerTechnology unpack_wds_SLQSGetDataBearerTechnology_t::lastCallDataBearerTechnology](#)

8.1056 unpack_wds_SLQSGetDUNCallInfo_t Struct Reference

Data Fields

- [connectionStatus connectionStatus](#)
- [uint16_t callEndReason](#)
- [uint64_t txOKBytesCount](#)
- [uint64_t rxOKBytesCount](#)
- [uint8_t dormancyStatus](#)
- [uint8_t dataBearerTech](#)
- [dunchannelRate channelRate](#)
- [uint64_t lastCallTXOKBytesCnt](#)
- [uint64_t lastCallRXOKBytesCnt](#)
- [uint64_t mdmCallDurationActive](#)
- [uint8_t lastCallDataBearerTech](#)

8.1056.1 Detailed Description

Parameters

<i>connection↔ Status</i>	Connection Status
<i>callEndReason</i>	Last Modem Call End Reason
<i>txOKBytesCount</i>	Tx Bytes OK
<i>rxOKBytesCount</i>	Rx Bytes OK
<i>dormancyStatus</i>	Dormancy Status
<i>dataBearerTech</i>	data bearer technology
<i>channelRate</i>	data Channel Rate
<i>lastCallTXOK↔ BytesCnt</i>	Last Call Tx Bytes OK
<i>lastCallRXOK↔ BytesCnt</i>	Last Call Rx Bytes OK
<i>mdmCall↔ DurationActive</i>	Call active duration
<i>lastCallData↔ BearerTech</i>	Last Call Data Bearer Technology

8.1056.2 Field Documentation

8.1056.2.1 `uint16_t unpack_wds_SLQSGetDUNCallInfo_t::callEndReason`

8.1056.2.2 `dunchannelRate unpack_wds_SLQSGetDUNCallInfo_t::channelRate`

8.1056.2.3 `connectionStatus unpack_wds_SLQSGetDUNCallInfo_t::connectionStatus`

8.1056.2.4 `uint8_t unpack_wds_SLQSGetDUNCallInfo_t::dataBearerTech`

8.1056.2.5 `uint8_t unpack_wds_SLQSGetDUNCallInfo_t::dormancyStatus`

8.1056.2.6 `uint8_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallDataBearerTech`

8.1056.2.7 `uint64_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallRXOKBytesCnt`

8.1056.2.8 `uint64_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallTXOKBytesCnt`

8.1056.2.9 `uint64_t unpack_wds_SLQSGetDUNCallInfo_t::mdmCallDurationActive`

8.1056.2.10 `uint64_t unpack_wds_SLQSGetDUNCallInfo_t::rxOKBytesCount`

8.1056.2.11 `uint64_t unpack_wds_SLQSGetDUNCallInfo_t::txOKBytesCount`

8.1057 `unpack_wds_SLQSGetProfileSettings_t` Struct Reference

Data Fields

- [`UnPackGetProfileSettingOut * pProfileSettings`](#)
- `uint8_t ProfileType`
- `uint16_t Tlvresult`

8.1057.1 Field Documentation

8.1057.1.1 **UnPackGetProfileSettingOut*** unpack_wds_SLQSGetProfileSettings_t::pProfileSettings

8.1057.1.2 uint8_t unpack_wds_SLQSGetProfileSettings_t::ProfileType

8.1057.1.3 uint16_t unpack_wds_SLQSGetProfileSettings_t::Tlvresult

8.1058 unpack_wds_SLQSGetRuntimeSettings_t Struct Reference

Data Fields

- uint32_t [IPv4](#)
- uint8_t [ProfileName](#) [128]
- uint32_t [PDPTType](#)
- uint8_t [APNName](#) [128]
- uint32_t [PrimaryDNSV4](#)
- uint32_t [SecondaryDNSV4](#)
- [LibPackUMTSQoS](#) UMTSGrantedQoS
- struct [wds_GPRSQoS](#) GPRSGrantedQoS
- uint8_t [Username](#) [128]
- uint32_t [Authentication](#)
- struct [wds_ProfileIdentifier](#) ProfileID
- uint32_t [GWAddressV4](#)
- uint32_t [SubnetMaskV4](#)
- uint8_t [PCSCFAddrPCO](#)
- struct [wds_PCSCFIPv4ServerAddressList](#) ServerAddrList
- struct [wds_PCSCFFQDNAddressList](#) PCSCFFQDNAddrList
- uint16_t [PrimaryDNSV6](#) [8]
- uint16_t [SecondaryDNSV6](#) [8]
- uint32_t [Mtu](#)
- struct [wds_DomainNameList](#) DomainList
- uint8_t [IPFamilyPreference](#)
- uint8_t [IMCNflag](#)
- uint16_t [Technology](#)
- struct [wds_IPV6AddressInfo](#) IPV6AddrInfo
- struct [wds_IPV6GWAddressInfo](#) IPV6GWAddrInfo

8.1058.1 Detailed Description

Parameters

<i>IPv4</i>	ipv4 address
<i>ProfileName</i>	profile name
<i>PDPTType</i>	PDP type
<i>APNName</i>	APN name
<i>PrimaryDNSV4</i>	
<i>SecondaryDNSV4</i>	
<i>UMTSGrantedQoS</i>	UMTS Granted Qos

<i>GPRSGrantedQoS</i>	GPRS Granted QoS
<i>Username</i>	
<i>Authentication</i>	
<i>ProfileID</i>	
<i>GWAddressV4</i>	Gateway IPv4
<i>SubnetMaskV4</i>	Subnet mask IPV4
<i>PCSCFAddrPCO</i>	
<i>PrimaryDNSV6</i>	Primary DNS IPV6
<i>SecondaryDNSV6</i>	Secondary DNS IPV6
<i>UMTSGrantedQoS</i>	UMTS Granted QoS
<i>SecondaryDNSV4</i>	
<i>Mtu</i>	Maximum Transfer Unit
<i>DomainList</i>	
<i>IPFamilyPreference</i>	

8.1058.2 Field Documentation

8.1058.2.1 `uint8_t unpack_wds_SLQSGetRuntimeSettings_t::APNName[128]`

8.1058.2.2 `uint32_t unpack_wds_SLQSGetRuntimeSettings_t::Authentication`

8.1058.2.3 `struct wds_DomainNameList unpack_wds_SLQSGetRuntimeSettings_t::DomainList`

8.1058.2.4 `struct wds_GPRSQoS unpack_wds_SLQSGetRuntimeSettings_t::GPRSGrantedQoS`

8.1058.2.5 `uint32_t unpack_wds_SLQSGetRuntimeSettings_t::GWAddressV4`

8.1058.2.6 `uint8_t unpack_wds_SLQSGetRuntimeSettings_t::IMCNflag`

8.1058.2.7 `uint8_t unpack_wds_SLQSGetRuntimeSettings_t::IPFamilyPreference`

8.1058.2.8 `uint32_t unpack_wds_SLQSGetRuntimeSettings_t::IPv4`

8.1058.2.9 `struct wds_IPV6AddressInfo unpack_wds_SLQSGetRuntimeSettings_t::IPv6AddrInfo`

8.1058.2.10 `struct wds_IPV6GWAddressInfo unpack_wds_SLQSGetRuntimeSettings_t::IPv6GWAddrInfo`

8.1058.2.11 `uint32_t unpack_wds_SLQSGetRuntimeSettings_t::Mtu`

8.1058.2.12 `uint8_t unpack_wds_SLQSGetRuntimeSettings_t::PCSCFAddrPCO`

8.1058.2.13 `struct wds_PCSCFFQDNAddressList unpack_wds_SLQSGetRuntimeSettings_t::PCSCFFQDNAddrList`

- 8.1058.2.14 `uint32_t` unpack_wds_SLQSGetRuntimeSettings_t::PDPTType
- 8.1058.2.15 `uint32_t` unpack_wds_SLQSGetRuntimeSettings_t::PrimaryDNSV4
- 8.1058.2.16 `uint16_t` unpack_wds_SLQSGetRuntimeSettings_t::PrimaryDNSV6[8]
- 8.1058.2.17 `struct wds_ProfileIdentifier` unpack_wds_SLQSGetRuntimeSettings_t::ProfileID
- 8.1058.2.18 `uint8_t` unpack_wds_SLQSGetRuntimeSettings_t::ProfileName[128]
- 8.1058.2.19 `uint32_t` unpack_wds_SLQSGetRuntimeSettings_t::SecondaryDNSV4
- 8.1058.2.20 `uint16_t` unpack_wds_SLQSGetRuntimeSettings_t::SecondaryDNSV6[8]
- 8.1058.2.21 `struct wds_PCSCFIPv4ServerAddressList` unpack_wds_SLQSGetRuntimeSettings_t::ServerAddrList
- 8.1058.2.22 `uint32_t` unpack_wds_SLQSGetRuntimeSettings_t::SubnetMaskV4
- 8.1058.2.23 `uint16_t` unpack_wds_SLQSGetRuntimeSettings_t::Technology
- 8.1058.2.24 `LibPackUMTSQoS` unpack_wds_SLQSGetRuntimeSettings_t::UMTSGrantedQoS
- 8.1058.2.25 `uint8_t` unpack_wds_SLQSGetRuntimeSettings_t::Username[128]

8.1059 unpack_wds_SLQSModifyProfile_t Struct Reference

Data Fields

- `uint16_t` * [pExtErrorCode](#)

8.1059.1 Detailed Description

Parameters

<i>extended</i>	error
-----------------	-------

8.1059.2 Field Documentation

- 8.1059.2.1 `uint16_t`* unpack_wds_SLQSModifyProfile_t::pExtErrorCode

8.1060 unpack_wds_SLQSSetIPFamilyPreference_t Struct Reference

Data Fields

- `uint16_t` [Tlvresult](#)

8.1060.1 Detailed Description

Parameters

<i>Tlvresult</i>	unpack result
------------------	---------------

8.1060.2 Field Documentation

8.1060.2.1 uint16_t unpack_wds_SLQSSetIPFamilyPreference_t::Tlvresult

8.1061 unpack_wds_SLQSSetPacketSrvStatusCallback_t Struct Reference

Data Fields

- uint8_t [conn_status](#)
- uint8_t [reconfigReqd](#)
- uint16_t [sessionEndReason](#)
- uint16_t [verboseSessnEndReasonType](#)
- uint16_t [verboseSessnEndReason](#)
- uint8_t [ipFamily](#)
- uint16_t [techName](#)
- uint8_t [bearerID](#)

8.1061.1 Detailed Description

Parameters

<i>conn_status</i>	connection status
<i>reconfigReqd</i>	Indicates whether the network interface on the host needs to be reconfigured.
<i>sessionEndReason</i>	Call End Reason
<i>verboseSessnEndReasonType</i>	Verbose call end reason type
<i>verboseSessnEndReason</i>	Reason the call ended (verbose)
<i>ipFamily</i>	IP family of the packet data connection.
<i>techName</i>	Technology name of the packet data connection.
<i>bearerID</i>	<ul style="list-style-type: none"> • bearer ID (3GPP) or RLP ID (3GPP2) of the packet data connection. • Valid Values - 0 to 16 • 0xFF - Invalid value.

8.1061.2 Field Documentation

8.1061.2.1 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::bearerID

8.1061.2.2 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::conn_status

8.1061.2.3 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::ipFamily

8.1061.2.4 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::reconfigReqd

8.1061.2.5 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::sessionEndReason

8.1061.2.6 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::techName

8.1061.2.7 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::verboseSessnEndReason

8.1061.2.8 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::verboseSessnEndReasonType

8.1062 unpack_wds_SLQSSetWdsEventCallback_ind_t Struct Reference

Data Fields

- uint8_t [xferStatAvail](#)
- uint64_t [tx_bytes](#)
- uint64_t [rx_bytes](#)
- uint64_t [tx_pkts](#)
- uint64_t [rx_pkts](#)
- uint8_t [mipstatAvail](#)
- uint32_t [mipStatus](#)
- uint8_t [dBTechAvail](#)
- uint32_t [dBTechnology](#)
- uint8_t [dormancyStatAvail](#)
- uint32_t [dormancyStatus](#)
- uint8_t [currDBTechAvail](#)
- uint32_t [ratMask](#)
- uint32_t [soMask](#)
- uint8_t [dataSysStatAvail](#)
- uint8_t [prefNetwork](#)
- uint8_t [netInfoLen](#)
- [wds_currNetworkInfo](#) [currNWInfo](#) [255]

8.1062.1 Detailed Description

Parameters

<i>xferStatAvail</i>	transfer statistic available
<i>tx_bytes</i>	transmit bytes
<i>rx_bytes</i>	received bytes
<i>tx_pkts</i>	transmit packets
<i>rx_pkts</i>	received packets
<i>mipstatAvail</i>	Mobile IP status available
<i>mipStatus</i>	Mobile IP status
<i>dBTechAvail</i>	Data Bearer technology available
<i>dBTechnology</i>	Data Bearer technology
<i>dormancyStatAvail</i>	Dormancy status available
<i>dormancyStatus</i>	Dormancy status
<i>currDBTechAvail</i>	Current Data Bearer technology available
<i>ratMask</i>	RAT mask to indicate type of technology
<i>soMask</i>	SO mask to indicate the service type
<i>dataSysStatAvail</i>	Data System Status available
<i>prefNetwork</i>	preferred network

8.1062.2 Field Documentation

- 8.1062.2.1 `uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::currDBTechAvail`
- 8.1062.2.2 `wds_currNetworkInfo unpack_wds_SLQSSetWdsEventCallback_ind_t::currNWInfo[255]`
- 8.1062.2.3 `uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dataSysStatAvail`
- 8.1062.2.4 `uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dbTechAvail`
- 8.1062.2.5 `uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dbTechnology`
- 8.1062.2.6 `uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dormancyStatAvail`
- 8.1062.2.7 `uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dormancyStatus`
- 8.1062.2.8 `uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::mipstatAvail`
- 8.1062.2.9 `uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::mipStatus`
- 8.1062.2.10 `uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::netInfoLen`
- 8.1062.2.11 `uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::prefNetwork`
- 8.1062.2.12 `uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::ratMask`
- 8.1062.2.13 `uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::rx_bytes`
- 8.1062.2.14 `uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::rx_pkts`
- 8.1062.2.15 `uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::soMask`
- 8.1062.2.16 `uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::tx_bytes`
- 8.1062.2.17 `uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::tx_pkts`
- 8.1062.2.18 `uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::xferStatAvail`

8.1063 `unpack_wds_SLQSSetDHCPv4ClientConfig_t` Struct Reference

Data Fields

- [wdsDhcpv4HwConfig](#) * [pHwConfig](#)
- [wdsDhcpv4OptionList](#) * [pRequestOptionList](#)

8.1063.1 Detailed Description

Parameters

<i>pHwConfig</i>	pointer to HW Config structure
<i>pRequestOptionList</i>	pointer to Option List structure to be sent in DHCP request

8.1063.2 Field Documentation

8.1063.2.1 wdsDhcpv4HwConfig* unpack_wds_SLQSSGetDHCpv4ClientConfig_t::pHwConfig

8.1063.2.2 wdsDhcpv4OptionList* unpack_wds_SLQSSGetDHCpv4ClientConfig_t::pRequestOptionList

8.1064 unpack_wds_SLQSSGetLoopback_t Struct Reference

Data Fields

- uint8_t [ByteLoopbackMode](#)
- uint8_t [ByteLoopbackMultiplier](#)

8.1064.1 Detailed Description

Parameters

<i>ByteLoopbackMode</i>	<ul style="list-style-type: none"> • Loopback Mode. <ul style="list-style-type: none"> – 0 - Disable – 1 - Enable
<i>ByteLoopbackMultiplier</i>	<ul style="list-style-type: none"> • Loopback multiplier. Number of downlink bytes to send for each uplink byte.

8.1064.2 Field Documentation

8.1064.2.1 uint8_t unpack_wds_SLQSSGetLoopback_t::ByteLoopbackMode

8.1064.2.2 uint8_t unpack_wds_SLQSSGetLoopback_t::ByteLoopbackMultiplier

8.1065 unpack_wds_SLQSSStartDataSession_t Struct Reference

Data Fields

- uint32_t * [psid](#)
- uint32_t * [pFailureReason](#)
- uint32_t * [pVerboseFailReasonType](#)
- uint32_t * [pVerboseFailureReason](#)

8.1065.1 Detailed Description

Parameters

<i>psid</i>	<ul style="list-style-type: none"> Assigned session ID when starting a data session
<i>pFailureReason</i>	<ul style="list-style-type: none"> Reason data session failed to be established See qaGobiApiTableCallEndReasons.h for Call End Reason
<i>pVerboseFailReasonType</i>	<ul style="list-style-type: none"> Parameter describing type of verbose failure reason See qaGobiApiTableCallEndReasons.h for Call End Reason Type
<i>pVerboseFailureReason</i>	<ul style="list-style-type: none"> Verbose reason explaining why call failed. Depends on verbFailReasonType parameter See qaGobiApiTableCallEndReasons.h for Call End Reason

8.1065.2 Field Documentation

8.1065.2.1 uint32_t* unpack_wds_SLQSSstartDataSession_t::pFailureReason

8.1065.2.2 uint32_t* unpack_wds_SLQSSstartDataSession_t::psid

8.1065.2.3 uint32_t* unpack_wds_SLQSSstartDataSession_t::pVerboseFailReasonType

8.1065.2.4 uint32_t* unpack_wds_SLQSSstartDataSession_t::pVerboseFailureReason

8.1066 unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference

Data Fields

- uint8_t contextId
- uint8_t bearerId
- int8_t apnName [100]
- uint32_t ipv4Address
- uint32_t ipv4GWAddress
- uint32_t prDNSIPv4Address
- uint32_t seDNSIPv4Address
- struct ipv6AddressInfo ipv6Address
- struct ipv6AddressInfo ipv6GWAddress
- uint16_t prDNSIPv6Address [8]
- uint16_t seDNSIPv6Address [8]
- uint32_t prPCSCFIPv4Address
- uint32_t sePCSCFIPv4Address
- uint16_t prPCSCFIPv6Address [8]
- uint16_t sePCSCFIPv6Address [8]

8.1066.1 Detailed Description

Parameters

<i>contextId</i>	Context Identifier
<i>bearerId</i>	Bearer Identity
<i>apnName</i>	APN name associated with the context id
<i>ipv4Address</i>	IPv4 Address
<i>ipv4GWAddress</i>	IPv4 Gateway Address
<i>prDNSIPv4↔ Address</i>	Primary DNS IPv4 Address
<i>seDNSIPv4↔ Address</i>	Secondary DNS IPv4 Address
<i>ipv6Address</i>	IPv6 Address
<i>ipv6GWAddress</i>	IPv6 Gateway Address
<i>prDNSIPv6↔ Address</i>	Primary IPv6 DNS Address
<i>seDNSIPv6↔ Address</i>	Secondary IPv6 DNS Address
<i>prPCSCFIPv4↔ Address</i>	Primary PCSCF IPv4 Address
<i>sePCSCFIPv4↔ Address</i>	Secondary PCSCF IPv4 Address
<i>prPCSCFIPv6↔ Address</i>	Primary PCSCF IPv6 Address
<i>sePCSCFIPv6↔ Address</i>	Secondary PCSCF IPv6 Address

8.1066.2 Field Documentation

8.1066.2.1 `int8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::apnName[100]`

8.1066.2.2 `uint8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::bearerId`

8.1066.2.3 `uint8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextId`

8.1066.2.4 `uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv4Address`

8.1066.2.5 `uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv4GWAddress`

8.1066.2.6 `struct ipv6AddressInfo unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv6Address`

8.1066.2.7 `struct ipv6AddressInfo unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv6GWAddress`

8.1066.2.8 `uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prDNSIPv4Address`

8.1066.2.9 `uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prDNSIPv6Address[8]`

8.1066.2.10 `uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prPCSCFIPv4Address`

8.1066.2.11 `uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prPCSCFIPv6Address[8]`

8.1066.2.12 `uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::seDNSIPv4Address`

8.1066.2.13 `uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::seDNSIPv6Address[8]`

8.1066.2.14 `uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::sePCSCFIPv4Address`

8.1066.2.15 `uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::sePCSCFIPv6Address[8]`

8.1067 UnPackGetProfileSettingOut Struct Reference

Data Fields

- [UnpackQmiProfileInfo curProfile](#)
- `uint16_t *` [pExtErrCode](#)

8.1067.1 Field Documentation

8.1067.1.1 `UnpackQmiProfileInfo UnPackGetProfileSettingOut::curProfile`

8.1067.1.2 `uint16_t* UnPackGetProfileSettingOut::pExtErrCode`

8.1068 unpackWdsProfileParam Union Reference

Data Fields

- [LibpackProfile3GPP SlqsProfile3GPP](#)
- [LibpackProfile3GPP2 SlqsProfile3GPP2](#)

8.1068.1 Field Documentation

8.1068.1.1 `LibpackProfile3GPP unpackWdsProfileParam::SlqsProfile3GPP`

8.1068.1.2 `LibpackProfile3GPP2 unpackWdsProfileParam::SlqsProfile3GPP2`

8.1069 USBCompConfig Struct Reference

Data Fields

- `BYTE *` [pUSBComp](#)

8.1069.1 Detailed Description

This structure is used to store USB composition information

Parameters

<i>pUSBComp[IN]</i>	<ul style="list-style-type: none"> • Current USB Composition • Values: <ul style="list-style-type: none"> – 0..5 - Reserved (non-QMI) – 6 - DM NMEA AT QMI – 7 - DM NMEA AT QMI1 QMI2 QMI3 – 8 - DM NMEA AT MBIM – 9 - MBIM – 10 - NMEA MBIM – 11 - DM MBIM – 12 - DM NMEA MBIM – 13-22 are combined compositions. One is for Win8 MBIM interfaces, another is for legacy QMI interfaces – 13 - 6 for QMI, 8 for MBIM – 14 - 6 for QMI, 9 for MBIM – 15 - 6 for QMI, 10 for MBIM – 16 - 6 for QMI, 11 for MBIM – 17 - 6 for QMI, 12 for MBIM – 18 - 7 for QMI, 8 for MBIM – 19 - 7 for QMI, 9 for MBIM – 20 - 7 for QMI, 10 for MBIM – 21 - 7 for QMI, 11 for MBIM – 22 - 7 for QMI, 12 for MBIM
---------------------	--

8.1069.2 Field Documentation

8.1069.2.1 BYTE* USBCompConfig::pUSBComp

8.1070 USBCompParams Struct Reference

Data Fields

- [BYTE * pUSBComp](#)
- [BYTE * pNumSupUSBComps](#)
- [BYTE * pSupUSBComps](#)

8.1070.1 Detailed Description

This structure is used to store retrieved USB Composition

Parameters

<p><i>pUSBComp</i>[O↔ UT]</p>	<ul style="list-style-type: none"> • Current USB Composition(optional parameter) • Values: <ul style="list-style-type: none"> – 0..5 - Reserved (non-QMI) – 6 - DM NMEA AT QMI – 7 - DM NMEA AT QMI1 QMI2 QMI3 – 8 - DM NMEA AT MBIM – 9 - MBIM – 10 - NMEA MBIM – 11 - DM MBIM – 12 - DM NMEA MBIM 13-22 are combined compositions. One is for Win8 MBIM interfaces, another is for legacy QMI interfaces – 13 - 6 for QMI, 8 for MBIM – 14 - 6 for QMI, 9 for MBIM – 15 - 6 for QMI, 10 for MBIM – 16 - 6 for QMI, 11 for MBIM – 17 - 6 for QMI, 12 for MBIM – 18 - 7 for QMI, 8 for MBIM – 19 - 7 for QMI, 9 for MBIM – 20 - 7 for QMI, 10 for MBIM – 21 - 7 for QMI, 11 for MBIM – 22 - 7 for QMI, 12 for MBIM
<p><i>pNumSupUSB</i>↔ <i>Comps</i>[OUT]</p>	<ul style="list-style-type: none"> • Number of supported USB compositions in the parameter to follow • Range - 0-255

<p><i>pSupUSB↔</i> <i>Comps[OUT]</i></p>	<ul style="list-style-type: none"> • Optional parameter • List of supported USB compositions(1 Byte each - Max 255) • Total length is defined by pNumSupUSBComps parameter • Values: <ul style="list-style-type: none"> – 0..5 - Reserved (non-QMI) – 6 - DM NMEA AT QMI – 7 - DM NMEA AT QMI1 QMI2 QMI3 – 8 - DM NMEA AT MBIM – 9 - MBIM – 10 - NMEA MBIM – 11 - DM MBIM – 12 - DM NMEA MBIM – 13-22 are combined compositions. One is for Win8 MBIM interfaces, another is for legacy QMI interfaces – 13 - 6 for QMI, 8 for MBIM – 14 - 6 for QMI, 9 for MBIM – 15 - 6 for QMI, 10 for MBIM – 16 - 6 for QMI, 11 for MBIM – 17 - 6 for QMI, 12 for MBIM – 18 - 7 for QMI, 8 for MBIM – 19 - 7 for QMI, 9 for MBIM – 20 - 7 for QMI, 10 for MBIM – 21 - 7 for QMI, 11 for MBIM – 22 - 7 for QMI, 12 for MBIM
--	--

8.1070.2 Field Documentation

8.1070.2.1 **BYTE*** USBCompParams::pNumSupUSBComps

8.1070.2.2 **BYTE*** USBCompParams::pSupUSBComps

8.1070.2.3 **BYTE*** USBCompParams::pUSBComp

8.1071 USSDNoWaitIndicationInfo Struct Reference

Data Fields

- **BYTE *** pError
- **BYTE *** pFailureCause
- **struct USSInfo *** pUSSDData
- **alphaIDInfo *** pAlphaIdentifier

8.1071.1 Detailed Description

Contains the parameters passed for USSDNoWaitIndicationCallback by the device.

Parameters

<i>pError</i>	<ul style="list-style-type: none"> Type of Error (if any)
<i>pFailureCause</i>	<ul style="list-style-type: none"> Supplementary services failure cause
<i>pUSSDData</i>	<ul style="list-style-type: none"> USS Data from Network. See USSInfo for more details.

8.1071.2 Field Documentation

8.1071.2.1 `alphaIDInfo* USSDNoWaitIndicationInfo::pAlphaIdentifier`

8.1071.2.2 `BYTE* USSDNoWaitIndicationInfo::pError`

8.1071.2.3 `BYTE* USSDNoWaitIndicationInfo::pFailureCause`

8.1071.2.4 `struct USSInfo* USSDNoWaitIndicationInfo::pUSSDData`

8.1072 USSDRespFNetwork Struct Reference

Data Fields

- char * [pTypeCode](#)
- char * [pRespData](#)

8.1072.1 Detailed Description

This structure contains the response from the network

Parameters

<i>pTypeCode</i>	"0" USSD-Notify – text in pRespData "1" USSD-Request – text in pRespData "2" Session terminated by network "3" other local client (eg, SIM Toolkit) has responded "4" Operation not supported "5" Network timeout
<i>pRespData</i>	<ul style="list-style-type: none"> points to a message string received from the network

8.1072.2 Field Documentation

8.1072.2.1 `char* USSDRespFNetwork::pRespData`

8.1072.2.2 `char*` USSDRespFNetwork::pTypeCode

8.1073 USSInfo Struct Reference

Data Fields

- [BYTE](#) `ussDCS`
- [BYTE](#) `ussLen`
- [BYTE](#) `ussData` [182]

8.1073.1 Detailed Description

This structure contains USS Information

Parameters

<i>ussDCS</i>	<ul style="list-style-type: none">• 1 - ASCII coding scheme• 2 - 8-BIT coding scheme• 3 - UCS2
<i>ussLen</i>	<ul style="list-style-type: none">• Range 1 to 182
<i>ussData</i>	<ul style="list-style-type: none">• Data encoded as per the DCS

8.1073.2 Field Documentation

8.1073.2.1 [BYTE](#) `USSInfo::ussData`[182]

8.1073.2.2 [BYTE](#) `USSInfo::ussDCS`

8.1073.2.3 [BYTE](#) `USSInfo::ussLen`

8.1074 USSResp Struct Reference

Data Fields

- [WORD](#) * `pfailureCause`
- [alphaIDInfo](#) * `pAlphaIDInfo`
- [struct](#) `USSInfo` * `pUSSDInfo`
- [BYTE](#) * `pCcResultType`
- [BYTE](#) * `pCallId`
- `ccSUPSType` * `pCCSuppsType`

8.1074.1 Field Documentation

8.1074.1.1 **alphaIDInfo*** USSResp::pAlphaIDInfo

8.1074.1.2 **BYTE*** USSResp::pCallId

8.1074.1.3 **BYTE*** USSResp::pCcResultType

8.1074.1.4 **ccSUPSType*** USSResp::pCCSuppsType

8.1074.1.5 **WORD*** USSResp::pFailureCause

8.1074.1.6 **struct USSInfo*** USSResp::pUSSDInfo

8.1075 UUSInfo Struct Reference

Data Fields

- [BYTE UUSType](#)
- [BYTE UUSDcs](#)
- [BYTE UUSDatalen](#)
- [BYTE UUSData](#) [255]

8.1075.1 Detailed Description

This structure contains User to User Signaling Service Information.

Parameters

<i>UUSType</i>	<ul style="list-style-type: none"> • UUS type values are: <ul style="list-style-type: none"> – 0x00 - UUS_DATA – 0x01 - UUS_TYPE1_IMPLICIT – 0x02 - UUS_TYPE1_REQUIRED – 0x03 - UUS_TYPE1_NOT_REQUIRED – 0x04 - UUS_TYPE2_REQUIRED – 0x05 - UUS_TYPE2_NOT_REQUIRED – 0x06 - UUS_TYPE3_REQUIRED – 0x07 - UUS_TYPE3_NOT_REQUIRED – 0xFF - Not Available
<i>UUSDcs</i>	<ul style="list-style-type: none"> • UUS data coding scheme values are: <ul style="list-style-type: none"> – 0x01 - UUS_DCS_USP – 0x02 - UUS_DCS_OHLP – 0x03 - UUS_DCS_X244 – 0x04 - UUS_DCS_SMCf – 0x05 - UUS_DCS_IA5
	<ul style="list-style-type: none"> – 0x06 - UUS_DCS_RV12RD – 0x07 - UUS_DCS_Q931UNCCM – 0xFF - Not Available

<i>UUSDatalen</i>	<ul style="list-style-type: none"> Number of sets of the following elements. <ul style="list-style-type: none"> UUSData If zero(0) then no further information exists.
<i>UUSData[MA↔X_DESCRIPTOR↔ON_LENGTH]</i>	<ul style="list-style-type: none"> UUS data encoded as per coding scheme

8.1075.2 Field Documentation

8.1075.2.1 **BYTE** UUSInfo::UUSData[255]

8.1075.2.2 **BYTE** UUSInfo::UUSDatalen

8.1075.2.3 **BYTE** UUSInfo::UUSDcs

8.1075.2.4 **BYTE** UUSInfo::UUSType

8.1076 verifyUIMPIN Struct Reference

Data Fields

- [BYTE](#) pinID
- [BYTE](#) pinLen
- [BYTE](#) pinVal [255]

8.1076.1 Detailed Description

This structure contains the information about the pin parameters that need to be verified.

Parameters

<i>pinID</i>	<ul style="list-style-type: none"> Indicates the PIN ID to be verified. <ul style="list-style-type: none"> 1 - PIN1 (also called PIN) 2 - PIN2 3 - Universal PIN 4 - Hidden key
<i>pinLen</i>	<ul style="list-style-type: none"> Length of the following elements i.e. pin value.
<i>pinVal[MAX_DESCRIPTOR↔_LENGTH]</i>	<ul style="list-style-type: none"> PIN value. This value is a sequence of ASCII characters.

8.1076.2 Field Documentation

8.1076.2.1 BYTE verifyUIMPIN::pinID

8.1076.2.2 BYTE verifyUIMPIN::pinLen

8.1076.2.3 BYTE verifyUIMPIN::pinVal[255]

8.1077 voiceALSSelectLineInfo Struct Reference

Data Fields

- [BYTE lineValue](#)

8.1077.1 Detailed Description

This structure contains ALS Select Line Information Parameters.

Parameters

<i>lineValue</i>	<ul style="list-style-type: none">• ALS Line Value.<ul style="list-style-type: none">– 0x00 - ALS_LINE1 - Line 1 (default)– 0x01 - ALS_LINE2 - Line 2
------------------	--

8.1077.2 Field Documentation

8.1077.2.1 BYTE voiceALSSelectLineInfo::lineValue

8.1078 voiceALSSetLineSwitchInfo Struct Reference

Data Fields

- [BYTE switchOption](#)

8.1078.1 Detailed Description

This structure contains ALS Set Line Switching Information Parameters.

Parameters

<i>switchOption</i>	<ul style="list-style-type: none">• Switch Option.<ul style="list-style-type: none">– 0x00 - VOICE_LINE_SWITCHING_NOT_ALLOWED - Line switching is not allowed– 0x01 - VOICE_LINE_SWITCHING_ALLOWED - Line switching is allowed
---------------------	---

8.1078.2 Field Documentation

8.1078.2.1 BYTE voiceALSSetLineSwitchInfo::switchOption

8.1079 voiceAnswerCall Struct Reference

Data Fields

- [BYTE](#) * [pCallId](#)

8.1079.1 Detailed Description

Contains the parameters passed for SLQSVoiceAnswerCall.

Parameters

<i>pCallId</i> [IN/OUT]	<ul style="list-style-type: none">• Unique call identifier for the call that must be answered.
-------------------------	--

8.1079.2 Field Documentation

8.1079.2.1 BYTE* voiceAnswerCall::pCallId

8.1080 voiceBindSubscriptionInfo Struct Reference

Data Fields

- [BYTE](#) [subsType](#)

8.1080.1 Detailed Description

This structure contains Bind Subscription Information Parameters.

Parameters

<i>subsType</i>	<ul style="list-style-type: none">• Subscription Type.<ul style="list-style-type: none">– 0x00 - VOICE_SUBS_TYPE_PRIMARY - Primary– 0x01 - VOICE_SUBS_TYPE_SECONDARY - Secondary
-----------------	---

8.1080.2 Field Documentation

8.1080.2.1 **BYTE** `voiceBindSubscriptionInfo::subType`

8.1081 `voiceBurstDTMFInfo` Struct Reference

Data Fields

- [burstDTMFInfo](#) `BurstDTMFInfo`
- [DTMFLengths](#) * `pBurstDTMFLengths`

8.1081.1 Detailed Description

This structure contains parameters of burst Dual-Tone Multifrequency (DTMF)

Parameters

<i>BurstDTMFInfo</i>	<ul style="list-style-type: none">• Burst DTMF Information<ul style="list-style-type: none">– See burstDTMFInfo for more information
<i>pBurstDTMF↔ Lengths</i>	[optional] <ul style="list-style-type: none">• DTMF Lengths<ul style="list-style-type: none">– See DTMFLengths for more information

8.1081.2 Field Documentation

8.1081.2.1 **burstDTMFInfo** `voiceBurstDTMFInfo::BurstDTMFInfo`

8.1081.2.2 **DTMFLengths*** `voiceBurstDTMFInfo::pBurstDTMFLengths`

8.1082 `voiceCallInfoReq` Struct Reference

Data Fields

- **BYTE** `callID`

8.1082.1 Detailed Description

This structure contains information of the request parameters associated with a call.

Parameters

<i>callID</i>	<ul style="list-style-type: none">• Call identifier for the call queried for information.
---------------	---

8.1082.2 Field Documentation

8.1082.2.1 BYTE voiceCallInfoReq::callID

8.1083 voiceCallInfoResp Struct Reference

Data Fields

- [callInfo](#) * [pCallInfo](#)
- [remotePartyNum](#) * [pRemotePartyNum](#)
- [WORD](#) * [pSrvOpt](#)
- [BYTE](#) * [pVoicePrivacy](#)
- [BYTE](#) * [pOTASPStatus](#)
- [remotePartyName](#) * [pRemotePartyName](#)
- [UUSInfo](#) * [pUUSInfo](#)
- [BYTE](#) * [pAlertType](#)
- [alphaIDInfo](#) * [pAlphaIDInfo](#)
- [connectNumInfo](#) * [pConnectNumInfo](#)
- [diagInfo](#) * [pDiagInfo](#)
- [ULONG](#) * [pAlertingPattern](#)

8.1083.1 Detailed Description

This structure contains information of the response parameters associated with a call.

Parameters

<i>pCallInfo(optional)</i>	<ul style="list-style-type: none">• See callInfo for more information.
<i>pRemotePartyNum(optional)</i>	<ul style="list-style-type: none">• See remotePartyNum for more information.
<i>pSrvOpt</i>	<ul style="list-style-type: none">• Service option(optional)• Applicable only for 3GPP2 devices.• See Table8 qaGobiApiTableServiceOptions.h for standard service option number assignments.
<i>pVoicePrivacy</i>	<ul style="list-style-type: none">• Voice Privacy.(optional)• Applicable only for 3GPP2 devices.• Values.<ul style="list-style-type: none">– 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy– 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy– 0xFF - Not Available

<i>pOTASPStatus</i>	<ul style="list-style-type: none"> • OTASP status for the OTASP call.(optional) • Applicable only for 3GPP2 devices. <ul style="list-style-type: none"> – 0x00 - OTASP_STATUS_SPL_UNLOCKED - SPL unlocked; only for user-initiated OTASP – 0x01 - OTASP_STATUS_SPRC_RETRIES_EXCEEDED - SPC retries exceeded; only for user-initiated OTASP – 0x02 - OTASP_STATUS_AKEY_EXCHANGED - A-key exchanged; only for user-initiated OTASP – 0x03 - OTASP_STATUS_SSD_UPDATED - SSD updated; for both user-initiated OTASP and network-initiated OTASP (OTAPA) – 0x04 - OTASP_STATUS_NAM_DOWNLOADED - NAM downloaded; only for user-initiated OTASP – 0x05 - OTASP_STATUS_MDN_DOWNLOADED - MDN downloaded; only for user-initiated OTASP – 0x06 - OTASP_STATUS_IMSI_DOWNLOADED - IMSI downloaded; only for user-initiated OTASP – 0x07 - OTASP_STATUS_PRL_DOWNLOADED - PRL downloaded; only for user-initiated OTASP – 0x08 - OTASP_STATUS_COMMITTED - Commit successful; only for user-initiated OTASP – 0x09 - OTASP_STATUS_OTAPA_STARTED - OTAPA started; only for network-initiated OTASP (OTAPA) – 0x0A - OTASP_STATUS_OTAPA_STOPPED - OTAPA stopped; only for network-initiated OTASP (OTAPA) – 0x0B - OTASP_STATUS_OTAPA_ABORTED - OTAPA aborted; only for network-initiated OTASP (OTAPA) – 0x0C - OTASP_STATUS_OTAPA_COMMITTED - OTAPA committed; only for network-initiated OTASP (OTAPA) – 0xFF - Not Available
<i>pRemoteParty</i> ↔ <i>Name(optional)</i>	<ul style="list-style-type: none"> • Applicable only for 3GPP devices. • See remotePartyName for more information.
<i>pUUS</i> ↔ <i>Info(optional)</i>	<ul style="list-style-type: none"> • Applicable only for 3GPP devices. • See UUSInfo for more information.
<i>pAlert</i> ↔ <i>Type(optional)</i>	<ul style="list-style-type: none"> • Alerting type. • Applicable only for 3GPP devices. <ul style="list-style-type: none"> – 0x00 - ALERTING_LOCAL - Local – 0x01 - ALERTING_REMOTE - Remote – 0xFF - Not Available
<i>pAlphaID</i> ↔ <i>Info(optional)</i>	<ul style="list-style-type: none"> • Applicable only for 3GPP devices. • See alphaIDInfo for more information.
<i>pConnectNum</i> ↔ <i>Info(optional)</i>	<ul style="list-style-type: none"> • See connectNumInfo for more information.

<i>pDiag</i> ↔ <i>Info(optional)</i>	<ul style="list-style-type: none"> • See diagInfo for more information.
<i>pAlertingPattern</i>	<ul style="list-style-type: none"> • Alerting pattern.(optional) <ul style="list-style-type: none"> – 0x00 - QMI_VOICE_ALERTING_PATTERN_1 - Pattern 1 – 0x01 - QMI_VOICE_ALERTING_PATTERN_2 - Pattern 2 – 0x02 - QMI_VOICE_ALERTING_PATTERN_3 - Pattern 3 – 0x04 - QMI_VOICE_ALERTING_PATTERN_5 - Pattern 5 – 0x05 - QMI_VOICE_ALERTING_PATTERN_6 - Pattern 6 – 0x06 - QMI_VOICE_ALERTING_PATTERN_7 - Pattern 7 – 0x07 - QMI_VOICE_ALERTING_PATTERN_8 - Pattern 8 – 0x08 - QMI_VOICE_ALERTING_PATTERN_9 - Pattern 9 – 0xFF - Not Available

8.1083.2 Field Documentation

8.1083.2.1 **ULONG*** voiceCallInfoResp::pAlertingPattern

8.1083.2.2 **BYTE*** voiceCallInfoResp::pAlertType

8.1083.2.3 **alphaIDInfo*** voiceCallInfoResp::pAlphaIDInfo

8.1083.2.4 **callInfo*** voiceCallInfoResp::pCallInfo

8.1083.2.5 **connectNumInfo*** voiceCallInfoResp::pConnectNumInfo

8.1083.2.6 **diagInfo*** voiceCallInfoResp::pDiagInfo

8.1083.2.7 **BYTE*** voiceCallInfoResp::pOTASPStatus

8.1083.2.8 **remotePartyName*** voiceCallInfoResp::pRemotePartyName

8.1083.2.9 **remotePartyNum*** voiceCallInfoResp::pRemotePartyNum

8.1083.2.10 **WORD*** voiceCallInfoResp::pSrvOpt

8.1083.2.11 **UUSInfo*** voiceCallInfoResp::pUUSInfo

8.1083.2.12 **BYTE*** voiceCallInfoResp::pVoicePrivacy

8.1084 voiceCallRequestParams Struct Reference

Data Fields

- **BYTE** callNumber [81]
- **BYTE *** pCallType
- **BYTE *** pCLIRType
- **UUSInfo *** pUUSInfo
- **CUGInfo *** pCUGInfo
- **BYTE *** pEmergencyCategory
- **calledPartySubAdd *** pCallPartySubAdd
- **ULONG *** pSvcType

8.1084.1 Detailed Description

This structure contains Voice Call Request Parameters

Parameters

<i>callNumber[81]</i>	<ul style="list-style-type: none"> • Number to be dialed in ASCII string, NULL terminated. • Length Range [1 to 81]
<i>pCallType(optional)</i>	<ul style="list-style-type: none"> • the type of call to be dialed. CALL_TYPE_VOICE is automatically selected if this parameter is not provided. When CALL_TYPE_NON_STD_OTASP is selected, the call is sent as a nonstandard OTASP call regardless of the digit string Call type values are: <ul style="list-style-type: none"> – 0x00 - CALL_TYPE_VOICE - Voice (automatic selection) – 0x01 - CALL_TYPE_VOICE_FORCED - Avoid modem call classification – 0x08 - CALL_TYPE_NON_STD_OTASP - Nonstandard OTASP* – 0x09 - CALL_TYPE_EMERGENCY - Emergency
<i>pCLIRType(optional)</i>	<ul style="list-style-type: none"> • CLIR type values are: <ul style="list-style-type: none"> – 0x01 - CLIR_SUPPRESSION - Suppression – 0x02 - CLIR_INVOCATION - Invocation
<i>pUUSInfoFo(optional)</i>	<ul style="list-style-type: none"> • Pointer to structure of UUSInfo <ul style="list-style-type: none"> – See UUSInfo for more information
<i>pCUGInfo(optional)</i>	<ul style="list-style-type: none"> • Pointer to structure of CUGInfo <ul style="list-style-type: none"> – See CUGInfo for more information
<i>pEmergencyCategory(optional)</i>	<ul style="list-style-type: none"> • Bit mask of emergency number categories. This is only applicable when the call type is set to Emergency. <ul style="list-style-type: none"> – Bit 0 - VOICE_EMER_CAT_POLICE_BIT - Police – Bit 1 - VOICE_EMER_CAT_AMBULANCE_BIT - Ambulance – Bit 2 - VOICE_EMER_CAT_FIRE_BRIGADE_BIT - Fire brigade – Bit 3 - VOICE_EMER_CAT_MARINE_GUARD_BIT - Marine guard – Bit 4 - VOICE_EMER_CAT_MOUNTAIN_RESCUE_BIT - Mountain rescue – Bit 5 - VOICE_EMER_CAT_MANUAL_ECALL_BIT - Manual emergency call – Bit 6 - VOICE_EMER_CAT_AUTO_ECALL_BIT - Automatic emergency call – Bit 7 - VOICE_EMER_CAT_SPARE_BIT - Spare bit
<i>pCallPartySubAdd(optional)</i>	<ul style="list-style-type: none"> • Pointer to structure of calledPartySubAdd <ul style="list-style-type: none"> – See calledPartySubAdd for more information

<p><i>pSvc</i>↔ <i>Type(optional)</i></p>	<ul style="list-style-type: none"> • Service Type. <ul style="list-style-type: none"> – 0x01 - VOICE_DIAL_CALL_SRV_TYPE_AUTOMATIC - Automatic – 0x02 - VOICE_DIAL_CALL_SRV_TYPE_GSM - GSM – 0x03 - VOICE_DIAL_CALL_SRV_TYPE_WCDMA - WCDMA – 0x04 - VOICE_DIAL_CALL_SRV_TYPE_CDMA_AUTOMATIC - CDMA automatic – 0x05 - VOICE_DIAL_CALL_SRV_TYPE_GSM_WCDMA - GSM or WCDMA – 0x06 - VOICE_DIAL_CALL_SRV_TYPE_LTE -LTE
---	---

8.1084.2 Field Documentation

8.1084.2.1 **BYTE** voiceCallRequestParams::callNumber[81]

8.1084.2.2 **calledPartySubAdd*** voiceCallRequestParams::pCallPartySubAdd

8.1084.2.3 **BYTE*** voiceCallRequestParams::pCallType

8.1084.2.4 **BYTE*** voiceCallRequestParams::pCLIRType

8.1084.2.5 **CUGInfo*** voiceCallRequestParams::pCUGInfo

8.1084.2.6 **BYTE*** voiceCallRequestParams::pEmergencyCategory

8.1084.2.7 **ULONG*** voiceCallRequestParams::pSvcType

8.1084.2.8 **UUSInfo*** voiceCallRequestParams::pUUSInFo

8.1085 voiceCallResponseParams Struct Reference

Data Fields

- **BYTE *** pCallID
- **alphaIDInfo *** pAlphaIDInfo
- **BYTE *** pCCResultType
- **ccSUPSType *** pCCSUPSType

8.1085.1 Detailed Description

This structure contains Voice Call Response Parameters

Parameters

<i>pCallID(optional)</i>	<ul style="list-style-type: none"> • Unique call identifier for the dialed call
<i>pAlphaID↔Info(optional)</i>	<ul style="list-style-type: none"> • Pointer to structure of alphaIDInfo <ul style="list-style-type: none"> – See alphaIDInfo for more information
<i>pCCResult↔Type(optional)</i>	<ul style="list-style-type: none"> • Call Control Result Type. <ul style="list-style-type: none"> – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service
<i>pCCSUPS↔Type(optional)</i>	<ul style="list-style-type: none"> • Pointer to structure of ccSUPSType • Data is present when pCCResultType is present and is other than Voice. <ul style="list-style-type: none"> – See ccSUPSType for more information

8.1085.2 Field Documentation

8.1085.2.1 [alphaIDInfo](#)* `voiceCallResponseParams::pAlphaIDInfo`

8.1085.2.2 `BYTE`* `voiceCallResponseParams::pCallID`

8.1085.2.3 `BYTE`* `voiceCallResponseParams::pCCResultType`

8.1085.2.4 [ccSUPSType](#)* `voiceCallResponseParams::pCCSUPSType`

8.1086 voiceContDTMFinfo Struct Reference

Data Fields

- `BYTE` * [pCallID](#)
- `BYTE` [DTMFdigit](#)

8.1086.1 Detailed Description

This structure contains parameters of continuous DTMF

Parameters

<i>pCallID</i> [IN/OUT]	<ul style="list-style-type: none"> • Call ID associated with call on which the DTMF information has to be sent. Start continuous DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device
<i>DTMFdigit</i> [IN]	<ul style="list-style-type: none"> • DTMF digit in ASCII.

8.1086.2 Field Documentation

8.1086.2.1 **BYTE** voiceContDTMFInfo::DTMFdigit

8.1086.2.2 **BYTE*** voiceContDTMFInfo::pCallID

8.1087 voiceDTMFEventInfo Struct Reference

Data Fields

- [DTMFInfo DTMFInformation](#)
- **BYTE** * pOnLength
- **BYTE** * pOffLength

8.1087.1 Detailed Description

This structure contains the parameters passed for SLQSVoiceSetDTMFEventCallBack by the device.

Parameters

<i>DTMF</i> <i>Information(mandatory)</i>	See DTMFInfo for more information.
<i>pOn</i> <i>Length(optional)</i>	<ul style="list-style-type: none"> • DTMF Pulse Width <ul style="list-style-type: none"> – 0x00 - DTMF_ONLENGTH_95MS - 95 ms – 0x01 - DTMF_ONLENGTH_150MS - 150 ms – 0x02 - DTMF_ONLENGTH_200MS - 200 ms – 0x03 - DTMF_ONLENGTH_250MS - 250 ms – 0x04 - DTMF_ONLENGTH_300MS - 300 ms – 0x05 - DTMF_ONLENGTH_350MS - 350 ms – 0x06 - DTMF_ONLENGTH_SMS - SMS Tx special pulse width
<i>pOff</i> <i>Length(optional)</i>	<ul style="list-style-type: none"> • DTMF Interdigit Interval <ul style="list-style-type: none"> – 0x00 - DTMF_OFFLENGTH_60MS - 60 ms – 0x01 - DTMF_OFFLENGTH_100MS - 100 ms – 0x02 - DTMF_OFFLENGTH_150MS - 150 ms – 0x03 - DTMF_OFFLENGTH_200MS - 200 ms
Generated by Doxygen	

Note

None

8.1087.2 Field Documentation

8.1087.2.1 DTMFInfo voiceDTMFEventInfo::DTMFInformation

8.1087.2.2 BYTE* voiceDTMFEventInfo::pOffLength

8.1087.2.3 BYTE* voiceDTMFEventInfo::pOnLength

8.1088 voiceFlashInfo Struct Reference

Data Fields

- [BYTE * pCallID](#)
- [BYTE * pFlashPayLd](#)
- [BYTE * pFlashType](#)

8.1088.1 Detailed Description

This structure contains the flash information associated with a call.

Parameters

<i>pCallID</i> [IN/OUT]	<ul style="list-style-type: none"> • Unique call identifier associated with the current call.
<i>pFlashPayLd</i> [↔ N](optional)	<ul style="list-style-type: none"> • Payload in ASCII to be sent in Flash. • Variable Length, NULL terminated.
<i>pFlashType</i> [↔ N](optional)	<ul style="list-style-type: none"> • Flash type. <ul style="list-style-type: none"> – 0 - Simple Flash (default) – 1 - Activate answer hold – 2 - Deactivate answer hold

8.1088.2 Field Documentation

8.1088.2.1 BYTE* voiceFlashInfo::pCallID

8.1088.2.2 BYTE* voiceFlashInfo::pFlashPayLd

8.1088.2.3 BYTE* voiceFlashInfo::pFlashType

8.1089 voiceGetAllCallInfo Struct Reference

Data Fields

- [arrCallInfo](#) * [pArrCallInfo](#)
- [arrRemotePartyNum](#) * [pArrRemotePartyNum](#)
- [arrRemotePartyName](#) * [pArrRemotePartyName](#)
- [arrAlertingType](#) * [pArrAlertingType](#)
- [arrUUSInfo](#) * [pArrUUSInfo](#)
- [arrSvcOption](#) * [pArrSvcOption](#)
- [BYTE](#) * [pOTASPStatus](#)
- [BYTE](#) * [pVoicePrivacy](#)
- [arrCallEndReason](#) * [pArrCallEndReason](#)
- [arrAlphaID](#) * [pArrAlphaID](#)
- [arrConnectPartyNum](#) * [pArrConnectPartyNum](#)
- [arrDiagInfo](#) * [pArrDiagInfo](#)
- [arrCalledPartyNum](#) * [pArrCalledPartyNum](#)
- [arrRedirPartyNum](#) * [pArrRedirPartyNum](#)
- [arrAlertingPattern](#) * [pArrAlertingPattern](#)

8.1089.1 Detailed Description

This structure contains information about the response parameters with all the calls originating or terminating from a particular device.

Parameters

<i>pArrCallInfo(optional)</i>	• See arrCallInfo for more information.
<i>pArrRemotePartyNum(optional)</i>	• See arrRemotePartyNum for more information.
<i>pArrRemotePartyName(optional)</i>	• See arrRemotePartyName for more information.
<i>pArrAlertingType(optional)</i>	• See arrAlertingType for more information.
<i>pArrUUSInfo(optional)</i>	• See arrUUSInfo for more information.
<i>pArrSvcOption(optional)</i>	• See arrSvcOption for more information.

<p><i>pOTASP</i>↔ <i>Status(optional)</i></p>	<ul style="list-style-type: none"> • OTASP status for the OTASP call. • Applicable only for 3GPP2 devices. <ul style="list-style-type: none"> – 0x00 - OTASP_STATUS_SPL_UNLOCKED - SPL unlocked; only for user-initiated OTASP – 0x01 - OTASP_STATUS_SPRC_RETRIES_EXCEEDED - SPC retries exceeded; only for user-initiated OTASP – 0x02 - OTASP_STATUS_AKEY_EXCHANGED - A-key exchanged; only for user-initiated OTASP – 0x03 - OTASP_STATUS_SSD_UPDATED - SSD updated; for both user-initiated OTASP and network-initiated OTASP (OTAPA) – 0x04 - OTASP_STATUS_NAM_DOWNLOADED - NAM downloaded; only for user-initiated OTASP – 0x05 - OTASP_STATUS_MDN_DOWNLOADED - MDN downloaded; only for user-initiated OTASP – 0x06 - OTASP_STATUS_IMSI_DOWNLOADED - IMSI downloaded; only for user-initiated OTASP – 0x07 - OTASP_STATUS_PRL_DOWNLOADED - PRL downloaded; only for user-initiated OTASP – 0x08 - OTASP_STATUS_COMMITTED - Commit successful; only for user-initiated OTASP – 0x09 - OTASP_STATUS_OTAPA_STARTED - OTAPA started; only for network-initiated OTASP (OTAPA) – 0x0A - OTASP_STATUS_OTAPA_STOPPED - OTAPA stopped; only for network-initiated OTASP (OTAPA) – 0x0B - OTASP_STATUS_OTAPA_ABORTED - OTAPA aborted; only for network-initiated OTASP (OTAPA) – 0x0C - OTASP_STATUS_OTAPA_COMMITTED - OTAPA committed; only for network-initiated OTASP (OTAPA) – 0xFF - Not Available
<p><i>pVoice</i>↔ <i>Privacy(optional)</i></p>	<ul style="list-style-type: none"> • Voice Privacy. • Values. <ul style="list-style-type: none"> – 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy – 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy – 0xFF - Not Available
<p><i>pArrCallEnd</i>↔ <i>Reason(optional)</i></p>	<ul style="list-style-type: none"> • See arrCallEndReason for more information.
<p><i>pArrAlpha</i>↔ <i>D(optional)</i></p>	<ul style="list-style-type: none"> • See arrAlphaID for more information.
<p><i>pArrConnect</i>↔ <i>Party</i>↔ <i>Num(optional)</i></p>	<ul style="list-style-type: none"> • See arrConnectPartyNum for more information.
<p><i>pArrDiag</i>↔ <i>Info(optional)</i></p>	<ul style="list-style-type: none"> • See arrDiagInfo for more information.
<p><i>pArrCalled</i>↔ <i>Party</i>↔ <i>Num(optional)</i></p>	<ul style="list-style-type: none"> • See arrCalledPartyNum for more information.

<i>pArrRedir↵ Party↵ Num(optional)</i>	<ul style="list-style-type: none"> • See arrRedirPartyNum for more information.
<i>pArrAlerting↵ Pattern(optional)</i>	<ul style="list-style-type: none"> • See arrAlertingPattern for more information.

8.1089.2 Field Documentation

- 8.1089.2.1 **arrAlertingPattern*** voiceGetAllCallInfo::pArrAlertingPattern
- 8.1089.2.2 **arrAlertingType*** voiceGetAllCallInfo::pArrAlertingType
- 8.1089.2.3 **arrAlphaID*** voiceGetAllCallInfo::pArrAlphaID
- 8.1089.2.4 **arrCalledPartyNum*** voiceGetAllCallInfo::pArrCalledPartyNum
- 8.1089.2.5 **arrCallEndReason*** voiceGetAllCallInfo::pArrCallEndReason
- 8.1089.2.6 **arrCallInfo*** voiceGetAllCallInfo::pArrCallInfo
- 8.1089.2.7 **arrConnectPartyNum*** voiceGetAllCallInfo::pArrConnectPartyNum
- 8.1089.2.8 **arrDiagInfo*** voiceGetAllCallInfo::pArrDiagInfo
- 8.1089.2.9 **arrRedirPartyNum*** voiceGetAllCallInfo::pArrRedirPartyNum
- 8.1089.2.10 **arrRemotePartyName*** voiceGetAllCallInfo::pArrRemotePartyName
- 8.1089.2.11 **arrRemotePartyNum*** voiceGetAllCallInfo::pArrRemotePartyNum
- 8.1089.2.12 **arrSvcOption*** voiceGetAllCallInfo::pArrSvcOption
- 8.1089.2.13 **arrUUSInfo*** voiceGetAllCallInfo::pArrUUSInfo
- 8.1089.2.14 **BYTE*** voiceGetAllCallInfo::pOTASPStatus
- 8.1089.2.15 **BYTE*** voiceGetAllCallInfo::pVoicePrivacy

8.1090 voiceGetCallBarringReq Struct Reference

Data Fields

- [BYTE reason](#)
- [BYTE * pSvcClass](#)

8.1090.1 Detailed Description

This structure contains Voice Get Call Barring Request Parameters

Parameters

<i>reason</i>	<ul style="list-style-type: none"> • Call Barring Reason • Values: <ul style="list-style-type: none"> – 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING - All outgoing – 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT - Outgoing internal – 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTOHOM - Outgoing external to home – 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING - All incoming – 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING - Roaming incoming – 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING - All calls are barred – 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING - All outgoing calls are barred – 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING - All incoming calls are barred
<i>pSvcClass</i> [IN↔OUT]	<ul style="list-style-type: none"> • Service class is a combination (sum) of information class constants (optional) • See qaGobiApiTableSupServiceInfoClasses.h for service classes. • Service Class is set to 0 if call waiting is not active for any of the information classes. • 0xFF, if Not Available

8.1090.2 Field Documentation

8.1090.2.1 **BYTE*** voiceGetCallBarringReq::pSvcClass

8.1090.2.2 **BYTE** voiceGetCallBarringReq::reason

8.1091 voiceGetCallBarringResp Struct Reference

Data Fields

- **BYTE** * pSvcClass
- **WORD** * pFailCause
- **alphaIDInfo** * pAlphaIDInfo
- **BYTE** * pCCResType
- **BYTE** * pCallID
- **ccSUPSType** * pCCSUPSType

8.1091.1 Detailed Description

This structure contains Voice Get Call Barring Response Parameters

Parameters

<i>pSvcClass</i> [IN/OUT]	<ul style="list-style-type: none"> Service class is a combination (sum) of information class constants (optional) See qaGobiApiTableSupServiceInfoClasses.h for service classes. Service Class is set to 0 if call waiting is not active for any of the information classes. 0xFF, if Not Available
<i>pFailCause</i>	<ul style="list-style-type: none"> Supplementary services failure cause (optional) see qaGobiApiTableVoiceCallEndReasons.h for more information. 0xFFFF, if Not Available
<i>pAlphaIDInfo</i>	<ul style="list-style-type: none"> Pointer to structure of alphaIDInfo (optional) <ul style="list-style-type: none"> See alphaIDInfo for more information
<i>pCCResType</i>	<ul style="list-style-type: none"> Call Control Result Type (optional) <ul style="list-style-type: none"> 0x00 - CC_RESULT_TYPE_VOICE - Voice 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service 0xFF - Not Available
<i>pCallID</i>	<ul style="list-style-type: none"> Call ID of the voice call that resulted from call control. (optional) It is present when pCCResType is present and is Voice. If zero(0) then invalid.
<i>pCCSUPSType</i>	<ul style="list-style-type: none"> Supplementary service data that resulted from call control (optional) Data is present when pCCResType is present and is other than Voice. <ul style="list-style-type: none"> See ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.1091.2 Field Documentation

8.1091.2.1 **alphaIDInfo*** voiceGetCallBarringResp::pAlphaIDInfo

8.1091.2.2 **BYTE*** voiceGetCallBarringResp::pCallID

8.1091.2.3 **BYTE*** voiceGetCallBarringResp::pCCResType

8.1091.2.4 **ccSUPSType*** voiceGetCallBarringResp::pCCSUPSType

8.1091.2.5 **WORD*** voiceGetCallBarringResp::pFailCause

8.1091.2.6 **BYTE*** voiceGetCallBarringResp::pSvcClass

8.1092 voiceGetCallFWReq Struct Reference

Data Fields

- [BYTE Reason](#)
- [BYTE * pSvcClass](#)

8.1092.1 Detailed Description

This structure contains Voice Get Call Forwarding Status Request Parameters

Parameters

<i>Reason</i>	<ul style="list-style-type: none"> • Call Forwarding Reason • Values: <ul style="list-style-type: none"> – 0x01 - QMI_VOICE_REASON_FWDREASON_UNCONDITIONAL - Unconditional call forwarding – 0x02 - QMI_VOICE_REASON_FWDREASON_MOBILEBUSY - Forward when the mobile is busy – 0x03 - QMI_VOICE_REASON_FWDREASON_NOREPLY - Forward when there is no reply – 0x04 - QMI_VOICE_REASON_FWDREASON_UNREACHABLE - Forward when the call is unreachable – 0x05 - QMI_VOICE_REASON_FWDREASON_ALLFORWARDING - All forwarding – 0x06 - QMI_VOICE_REASON_FWDREASON_ALLCONDITIONAL - All conditional forwarding
<i>pSvc↔ Class(optional)</i>	<ul style="list-style-type: none"> • Service Class is a combination (sum) of information class constants • See qaGobiApiTableSupServiceInfoClasses.h for service classes.

8.1092.2 Field Documentation

8.1092.2.1 **BYTE*** voiceGetCallFWReq::pSvcClass

8.1092.2.2 **BYTE** voiceGetCallFWReq::Reason

8.1093 voiceGetCallFWResp Struct Reference

Data Fields

- [getCallFWInfo * pGetCallFWInfo](#)
- [WORD * pFailCause](#)
- [alphaIDInfo * pAlphaIDInfo](#)
- [BYTE * pCCResType](#)
- [BYTE * pCallID](#)
- [ccSUPSType * pCCSUPSType](#)
- [getCallFWExtInfo * pGetCallFWExtInfo](#)

8.1093.1 Detailed Description

This structure contains Voice Get Call Forwarding Status Response Parameters

Parameters

<i>pGetCallFWInfo</i>	<ul style="list-style-type: none"> • Pointer to structure of getCallFWInfo (optional) <ul style="list-style-type: none"> – See getCallFWInfo for more information
<i>pFailCause</i>	<ul style="list-style-type: none"> • Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
<i>pAlphaIDInfo</i>	<ul style="list-style-type: none"> • Pointer to structure of alphaIDInfo (optional) <ul style="list-style-type: none"> – See alphaIDInfo for more information
<i>pCCResType</i>	<ul style="list-style-type: none"> • Call Control Result Type (optional) <ul style="list-style-type: none"> – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
<i>pCallID</i>	<ul style="list-style-type: none"> • Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
<i>pCCSUPSType</i>	<ul style="list-style-type: none"> • Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. <ul style="list-style-type: none"> – See ccSUPSType for more information
<i>pGetCallFW↵ ExtInfo</i>	<ul style="list-style-type: none"> • Pointer to structure of getCallFWExtInfo (optional) <ul style="list-style-type: none"> – See getCallFWExtInfo for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.1093.2 Field Documentation

8.1093.2.1 alphaIDInfo* voiceGetCallFWResp::pAlphaIDInfo

8.1093.2.2 BYTE* voiceGetCallFWResp::pCallID

8.1093.2.3 **BYTE*** voiceGetCallFWResp::pCCResType

8.1093.2.4 **ccSUPSType*** voiceGetCallFWResp::pCCSUPSType

8.1093.2.5 **WORD*** voiceGetCallFWResp::pFailCause

8.1093.2.6 **getCallFWExtInfo*** voiceGetCallFWResp::pGetCallFWExtInfo

8.1093.2.7 **getCallFWInfo*** voiceGetCallFWResp::pGetCallFWInfo

8.1094 voiceGetCallWaitInfo Struct Reference

Data Fields

- **BYTE** * pSvcClass
- **WORD** * pFailCause
- **alphaIDInfo** * pAlphaIDInfo
- **BYTE** * pCCResType
- **BYTE** * pCallID
- **ccSUPSType** * pCCSUPSType

8.1094.1 Detailed Description

This structure contains Voice Get Call Waiting Response Parameters

Parameters

<i>pSvcClass</i> [IN↔ OUT]	<ul style="list-style-type: none"> • Service class is a combination (sum) of information class constants (optional) • See qaGobiApiTableSupServiceInfoClasses.h for service classes. • Service Class is set to 0 if call waiting is not active for any of the information classes. • 0xFF,if Not Available
<i>pFailCause</i> [O↔ UT]	<ul style="list-style-type: none"> • Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
<i>pAlphaIDInfo</i> [↔ OUT]	<ul style="list-style-type: none"> • Pointer to structure of alphaIDInfo (optional) <ul style="list-style-type: none"> – See alphaIDInfo for more information
<i>pCCResType</i> [↔ OUT]	<ul style="list-style-type: none"> • Call Control Result Type (optional) <ul style="list-style-type: none"> – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available

<i>pCallID</i> [OUT]	<ul style="list-style-type: none"> • Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
<i>pCCSUPS</i> Type[OUT]	<ul style="list-style-type: none"> • Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. <ul style="list-style-type: none"> – See ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.1094.2 Field Documentation

8.1094.2.1 **alphaIDInfo*** voiceGetCallWaitInfo::pAlphaIDInfo

8.1094.2.2 **BYTE*** voiceGetCallWaitInfo::pCallID

8.1094.2.3 **BYTE*** voiceGetCallWaitInfo::pCCResType

8.1094.2.4 **ccSUPSType*** voiceGetCallWaitInfo::pCCSUPSType

8.1094.2.5 **WORD*** voiceGetCallWaitInfo::pFailCause

8.1094.2.6 **BYTE*** voiceGetCallWaitInfo::pSvcClass

8.1095 voiceGetCLIPResp Struct Reference**Data Fields**

- [CLIPResp](#) * [pCLIPResp](#)
- [WORD](#) * [pFailCause](#)
- [alphaIDInfo](#) * [pAlphaIDInfo](#)
- [BYTE](#) * [pCCResType](#)
- [BYTE](#) * [pCallID](#)
- [ccSUPSType](#) * [pCCSUPSType](#)

8.1095.1 Detailed Description

This structure contains Voice Get Calling Line Identification Presentation(CLIP) Response Parameters

Parameters

<i>pCLIPResp</i>	<ul style="list-style-type: none"> • Pointer to structure of CLIPResp (optional) <ul style="list-style-type: none"> – See CLIPResp for more information
<i>pFailCause</i>	<ul style="list-style-type: none"> • Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
<i>pAlphaIDInfo</i>	<ul style="list-style-type: none"> • Pointer to structure of alphaIDInfo (optional) <ul style="list-style-type: none"> – See alphaIDInfo for more information
<i>pCCResType</i>	<ul style="list-style-type: none"> • Call Control Result Type (optional) <ul style="list-style-type: none"> – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
<i>pCallID</i>	<ul style="list-style-type: none"> • Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
<i>pCCSUPSType</i>	<ul style="list-style-type: none"> • Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. <ul style="list-style-type: none"> – See ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.1095.2 Field Documentation

8.1095.2.1 **alphaIDInfo*** voiceGetCLIPResp::pAlphaIDInfo

8.1095.2.2 **BYTE*** voiceGetCLIPResp::pCallID

8.1095.2.3 **BYTE*** voiceGetCLIPResp::pCCResType

8.1095.2.4 **ccSUPSType*** voiceGetCLIPResp::pCCSUPSType

8.1095.2.5 **CLIPResp*** voiceGetCLIPResp::pCLIPResp

8.1095.2.6 **WORD*** voiceGetCLIPResp::pFailCause

8.1096 voiceGetCLIRResp Struct Reference

Data Fields

- [CLIRResp](#) * [pCLIRResp](#)
- [WORD](#) * [pFailCause](#)
- [alphaIDInfo](#) * [pAlphaIDInfo](#)
- [BYTE](#) * [pCCResType](#)
- [BYTE](#) * [pCallID](#)
- [ccSUPSType](#) * [pCCSUPSType](#)

8.1096.1 Detailed Description

This structure contains Voice Get Calling Line Identification Restriction (CLIR) Response Parameters

Parameters

<i>pCLIRResp</i>	<ul style="list-style-type: none"> • Pointer to structure of CLIRResp (optional) <ul style="list-style-type: none"> – See CLIRResp for more information
<i>pFailCause</i>	<ul style="list-style-type: none"> • Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
<i>pAlphaIDInfo</i>	<ul style="list-style-type: none"> • Pointer to structure of alphaIDInfo (optional) <ul style="list-style-type: none"> – See alphaIDInfo for more information
<i>pCCResType</i>	<ul style="list-style-type: none"> • Call Control Result Type (optional) <ul style="list-style-type: none"> – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
<i>pCallID</i>	<ul style="list-style-type: none"> • Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
<i>pCCSUPSType</i>	<ul style="list-style-type: none"> • Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. <ul style="list-style-type: none"> – See ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.1096.2 Field Documentation

8.1096.2.1 **alphaIDInfo*** voiceGetCLIRResp::pAlphaIDInfo

8.1096.2.2 **BYTE*** voiceGetCLIRResp::pCallID

8.1096.2.3 **BYTE*** voiceGetCLIRResp::pCCResType

8.1096.2.4 **ccSUPSType*** voiceGetCLIRResp::pCCSUPSType

8.1096.2.5 **CLIRResp*** voiceGetCLIRResp::pCLIRResp

8.1096.2.6 **WORD*** voiceGetCLIRResp::pFailCause

8.1097 voiceGetCNAPResp Struct Reference

Data Fields

- [CNAPResp](#) * [pCNAPResp](#)
- [WORD](#) * [pFailCause](#)
- [alphaIDInfo](#) * [pAlphaIDInfo](#)
- [BYTE](#) * [pCCResType](#)
- [BYTE](#) * [pCallID](#)
- [ccSUPSType](#) * [pCCSUPSType](#)

8.1097.1 Detailed Description

This structure contains Voice Get Calling Name Presentation(CNAP) Response Parameters

Parameters

<i>pCNAPResp</i>	<ul style="list-style-type: none"> • Pointer to structure of CNAPResp (optional) <ul style="list-style-type: none"> – See CNAPResp for more information
<i>pFailCause</i>	<ul style="list-style-type: none"> • Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
<i>pAlphaIDInfo</i>	<ul style="list-style-type: none"> • Pointer to structure of alphaIDInfo (optional) <ul style="list-style-type: none"> – See alphaIDInfo for more information
<i>pCCResType</i>	<ul style="list-style-type: none"> • Call Control Result Type (optional) <ul style="list-style-type: none"> – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service
	<ul style="list-style-type: none"> – 0xFF - Not Available

<i>pCallID</i>	<ul style="list-style-type: none"> • Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
<i>pCCSUPSType</i>	<ul style="list-style-type: none"> • Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. <ul style="list-style-type: none"> – See ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.1097.2 Field Documentation

8.1097.2.1 **alphaIDInfo*** voiceGetCNAPResp::pAlphaIDInfo

8.1097.2.2 **BYTE*** voiceGetCNAPResp::pCallID

8.1097.2.3 **BYTE*** voiceGetCNAPResp::pCCResType

8.1097.2.4 **ccSUPSType*** voiceGetCNAPResp::pCCSUPSType

8.1097.2.5 **CNAPResp*** voiceGetCNAPResp::pCNAPResp

8.1097.2.6 **WORD*** voiceGetCNAPResp::pFailCause

8.1098 voiceGetCOLPResp Struct Reference**Data Fields**

- [COLPResp](#) * pCOLPResp
- [WORD](#) * pFailCause
- [alphaIDInfo](#) * pAlphaIDInfo
- [BYTE](#) * pCCResType
- [BYTE](#) * pCallID
- [ccSUPSType](#) * pCCSUPSType

8.1098.1 Detailed Description

This structure contains Voice Get Connected Line Identification Presentation(COLP) Response Parameters

Parameters

<i>pCOLPResp</i>	<ul style="list-style-type: none"> • Pointer to structure of COLPResp (optional) <ul style="list-style-type: none"> – See COLPResp for more information
<i>pFailCause</i>	<ul style="list-style-type: none"> • Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
<i>pAlphaIDInfo</i>	<ul style="list-style-type: none"> • Pointer to structure of alphaIDInfo (optional) <ul style="list-style-type: none"> – See alphaIDInfo for more information
<i>pCCResType</i>	<ul style="list-style-type: none"> • Call Control Result Type (optional) <ul style="list-style-type: none"> – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
<i>pCallID</i>	<ul style="list-style-type: none"> • Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
<i>pCCSUPSType</i>	<ul style="list-style-type: none"> • Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. <ul style="list-style-type: none"> – See ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.1098.2 Field Documentation

8.1098.2.1 **alphaIDInfo*** voiceGetCOLPResp::pAlphaIDInfo

8.1098.2.2 **BYTE*** voiceGetCOLPResp::pCallID

8.1098.2.3 **BYTE*** voiceGetCOLPResp::pCCResType

8.1098.2.4 **ccSUPSType*** voiceGetCOLPResp::pCCSUPSType

8.1098.2.5 **COLPResp*** voiceGetCOLPResp::pCOLPResp

8.1098.2.6 **WORD*** voiceGetCOLPResp::pFailCause

8.1099 voiceGetCOLRResp Struct Reference

Data Fields

- [COLRResp](#) * [pCOLRResp](#)
- [WORD](#) * [pFailCause](#)
- [alphaIDInfo](#) * [pAlphaIDInfo](#)
- [BYTE](#) * [pCCResType](#)
- [BYTE](#) * [pCallID](#)
- [ccSUPSType](#) * [pCCSUPSType](#)

8.1099.1 Detailed Description

This structure contains Voice Get Connected Line Identification Restriction(COLR) Response Parameters

Parameters

<i>pCOLRResp</i>	<ul style="list-style-type: none"> • Pointer to structure of COLRResp (optional) <ul style="list-style-type: none"> – See COLRResp for more information
<i>pFailCause</i>	<ul style="list-style-type: none"> • Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
<i>pAlphaIDInfo</i>	<ul style="list-style-type: none"> • Pointer to structure of alphaIDInfo (optional) <ul style="list-style-type: none"> – See alphaIDInfo for more information
<i>pCCResType</i>	<ul style="list-style-type: none"> • Call Control Result Type (optional) <ul style="list-style-type: none"> – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
<i>pCallID</i>	<ul style="list-style-type: none"> • Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
<i>pCCSUPSType</i>	<ul style="list-style-type: none"> • Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. <ul style="list-style-type: none"> – See ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.1099.2 Field Documentation

8.1099.2.1 **alphaIDInfo*** voiceGetCOLRResp::pAlphaIDInfo

8.1099.2.2 **BYTE*** voiceGetCOLRResp::pCallID

8.1099.2.3 **BYTE*** voiceGetCOLRResp::pCCResType

8.1099.2.4 **ccSUPSType*** voiceGetCOLRResp::pCCSUPSType

8.1099.2.5 **COLRResp*** voiceGetCOLRResp::pCOLRResp

8.1099.2.6 **WORD*** voiceGetCOLRResp::pFailCause

8.1100 voiceGetConfigReq Struct Reference

Data Fields

- **BYTE *** pAutoAnswer
- **BYTE *** pAirTimer
- **BYTE *** pRoamTimer
- **BYTE *** pTTYMode
- **BYTE *** pPrefVoiceSO
- **BYTE *** pAMRStatus
- **BYTE *** pPrefVoicePrivacy
- **BYTE *** pNamID
- **BYTE *** pVoiceDomainPref

8.1100.1 Detailed Description

This structure contains Voice Get Configuration Request Parameters

Parameters

<i>pAuto</i> ↔ <i>Answer(optional)</i>	<ul style="list-style-type: none"> • Indicator to retrieve the Auto Answer Information. <ul style="list-style-type: none"> – 0x01 - Include auto answer information
<i>pAir</i> ↔ <i>Timer(optional)</i>	<ul style="list-style-type: none"> • Indicator to retrieve the Air Timer Information. <ul style="list-style-type: none"> – 0x01 - Include air calls timer count information • Currently Not Supported.
<i>pRoam</i> ↔ <i>Timer(optional)</i>	<ul style="list-style-type: none"> • Indicator to retrieve the Roam Timer Information. <ul style="list-style-type: none"> – 0x01 - Include roam calls timer information • Currently Not Supported.

<i>pTTY</i> ↔ <i>Mode(optional)</i>	<ul style="list-style-type: none"> Indicator to retrieve the TTY Mode Information. <ul style="list-style-type: none"> 0x01 - Include TTY configuration status information
<i>pPrefVoiceSO</i> ↔ <i>O(optional)</i>	<ul style="list-style-type: none"> Indicator to retrieve the Preferred Voice SO Information. <ul style="list-style-type: none"> 0x01 - Include preferred voice configuration status information Currently Not Supported.
<i>pAMR</i> ↔ <i>Status(optional)</i>	<ul style="list-style-type: none"> Indicator to retrieve the AMR Status Information. <ul style="list-style-type: none"> 0x01 - Include AMR status information
<i>pPrefVoice</i> ↔ <i>Privacy(optional)</i>	<ul style="list-style-type: none"> Indicator to retrieve the Preferred Voice Privacy Information. <ul style="list-style-type: none"> 0x01 - Include preferred voice privacy status information
<i>pNamID</i> ↔ <i>D(optional)</i>	<ul style="list-style-type: none"> Index of the Number Assignment Module Index (CDMA subscription) to be configured Range: 0 to 3. Some modems support only 1 or 2 NAMs. The NAM Index is valid only when the request contains at least one of Air Timer, Roam Timer, and Preferred Voice SO. If no nam_id value is specified in the request, the default value is 0.
<i>pVoiceDomain</i> ↔ <i>Pref(optional)</i>	<ul style="list-style-type: none"> Indicator to retrieve the Preferred Voice Domain Information. <ul style="list-style-type: none"> 0x01 - Include voice domain preference information

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.1100.2 Field Documentation

8.1100.2.1 **BYTE*** voiceGetConfigReq::pAirTimer

8.1100.2.2 **BYTE*** voiceGetConfigReq::pAMRStatus

8.1100.2.3 **BYTE*** voiceGetConfigReq::pAutoAnswer

8.1100.2.4 **BYTE*** voiceGetConfigReq::pNamID

8.1100.2.5 **BYTE*** voiceGetConfigReq::pPrefVoicePrivacy

8.1100.2.6 **BYTE*** voiceGetConfigReq::pPrefVoiceSO

8.1100.2.7 **BYTE*** voiceGetConfigReq::pRoamTimer

8.1100.2.8 **BYTE*** voiceGetConfigReq::pTTYMode

8.1100.2.9 **BYTE*** voiceGetConfigReq::pVoiceDomainPref

8.1101 voiceGetConfigResp Struct Reference

Data Fields

- [BYTE *](#) [pAutoAnswerStat](#)
- [airTimer](#) * [pAirTimerCnt](#)
- [roamTimer](#) * [pRoamTimerCnt](#)
- **BYTE *** [pCurrTTYMode](#)
- [prefVoiceSO](#) * [pCurPrefVoiceSO](#)
- [curAMRConfig](#) * [pCurAMRConfig](#)
- **BYTE *** [pCurVoicePrivacyPref](#)
- **BYTE *** [pCurVoiceDomainPref](#)

8.1101.1 Detailed Description

This structure contains Voice Get Configuration Response Parameters.

Parameters

<i>pAutoAnswer</i> <i>Stat(optional)</i>	<ul style="list-style-type: none"> • Auto Answer Status • Value returned is read from NV_AUTO_ANSWER_I. <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled – 0xFF - Not Available
<i>pAirTimer</i> <i>Cnt(optional)</i>	<ul style="list-style-type: none"> • Air Timer Count • Value returned is read from NV_AIR_CNT_I. • See airTimer for more information
<i>pRoamTimer</i> <i>Cnt(optional)</i>	<ul style="list-style-type: none"> • Roam Timer Count • Value returned is read from NV_ROAM_CNT_I. • See roamTimer for more information
<i>pCurrTTY</i> <i>Mode(optional)</i>	<ul style="list-style-type: none"> • Current TTY Mode • Value returned is read from NV_TTY_I. <ul style="list-style-type: none"> – 0x00 - TTY_MODE_FULL - Full – 0x01 - TTY_MODE_VCO - Voice carry over – 0x02 - TTY_MODE_HCO - Hearing carry over – 0x03 - TTY_MODE_OFF - Off – 0xFF - Not Available

<i>pCurPrefVoiceSO(optional)</i>	<ul style="list-style-type: none"> • Current Preferred Voice SO • Value returned is read from NV_PREF_VOICE_SO_I. • See prefVoiceSO for more information
<i>pCurAMRConfig(optional)</i>	<ul style="list-style-type: none"> • Current Adaptive Multi-Rate Configuration. • Values returned are read from NV_GSM_ARM_CALL_CONFIG_I and NV_UMTS_AMR_CONFIG_I. • See curAMRConfig for more information
<i>pCurVoicePrivacyPref(optional)</i>	<ul style="list-style-type: none"> • Current Voice Privacy Preference • Value returned is read from NV_VOICE_PRIV_I. <ul style="list-style-type: none"> – 0x00 - Standard privacy – 0x01 - Enhanced privacy – 0xFF - Not Available
<i>pCurVoiceDomainPref(optional)</i>	<ul style="list-style-type: none"> • Current Voice Domain Preference. <ul style="list-style-type: none"> – 0x00 - Circuit-switched (CS) only – 0x01 - Packet-switched (PS) only – 0x02 - CS is preferred; PS is secondary – 0x03 - PS is preferred; CS is secondary – 0xFF - Not Available

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.1101.2 Field Documentation

8.1101.2.1 `airTimer*` `voiceGetConfigResp::pAirTimerCnt`

8.1101.2.2 `BYTE*` `voiceGetConfigResp::pAutoAnswerStat`

8.1101.2.3 `curAMRConfig*` `voiceGetConfigResp::pCurAMRConfig`

8.1101.2.4 `prefVoiceSO*` `voiceGetConfigResp::pCurPrefVoiceSO`

8.1101.2.5 `BYTE*` `voiceGetConfigResp::pCurrTTYMode`

8.1101.2.6 `BYTE*` `voiceGetConfigResp::pCurVoiceDomainPref`

8.1101.2.7 `BYTE*` `voiceGetConfigResp::pCurVoicePrivacyPref`

8.1101.2.8 `roamTimer* voiceGetConfigResp::pRoamTimerCnt`

8.1102 `voiceIndicationRegisterInfo` Struct Reference

Data Fields

- `BYTE * pRegDTMFEvents`
- `BYTE * pRegVoicePrivacyEvents`
- `BYTE * pSuppsNotifEvents`

8.1102.1 Detailed Description

This structure contains parameters of Indication Register Information

Parameters

<i><code>pRegDTMFEvents(optional)</code></i>	<ul style="list-style-type: none"> • Registration Indication For DTMF Events. • When this registration is enabled, the device learns of DTMF events via the <code>QMI_VOICE_DTMF_IND</code> indication. <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i><code>pRegVoicePrivacyEvents(optional)</code></i>	<ul style="list-style-type: none"> • Registration Indication For Voice Privacy Events. • When this registration is enabled, the device learns of DTMF events via the <code>QMI_VOICE_PRIVACY_IND</code> indication. <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i><code>pSuppsNotifEvents(optional)</code></i>	<ul style="list-style-type: none"> • Registration Indication For Supplementary Service Notification Events. • When this registration is enabled, the device learns of DTMF events via the <code>QMI_VOICE_SUPPS_NOTIFICATION_IND</code> indication. <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable

Note

One of the optional parameter is mandatory to be present in the request.

8.1102.2 Field Documentation

8.1102.2.1 `BYTE* voiceIndicationRegisterInfo::pRegDTMFEvents`

8.1102.2.2 `BYTE* voiceIndicationRegisterInfo::pRegVoicePrivacyEvents`

8.1102.2.3 BYTE* voicelIndicationRegisterInfo::pSuppsNotifEvents

8.1103 voicelInfoRec Struct Reference

Data Fields

- [BYTE callID](#)
- [signalInfo * pSignalInfo](#)
- [callerIDInfo * pCallerIDInfo](#)
- [BYTE * pDispInfo](#)
- [BYTE * pExtDispInfo](#)
- [BYTE * pCallerNameInfo](#)
- [BYTE * pCallWaitInd](#)
- [connectNumInfo * pConnectNumInfo](#)
- [connectNumInfo * pCallingPartyInfo](#)
- [calledPartyInfo * pCalledPartyInfo](#)
- [redirNumInfo * pRedirNumInfo](#)
- [BYTE * pCLIRCause](#)
- [NSSAudioCtrl * pNSSAudioCtrl](#)
- [BYTE * pNSSRelease](#)
- [lineCtrlInfo * pLineCtrlInfo](#)
- [extDispRecInfo * pExtDispRecInfo](#)

8.1103.1 Detailed Description

This structure contains Voice record Information

Parameters

<i>callID</i>	[Mandatory] • Call identifier for the call.
<i>pSignalInfo</i> [Optional]	• Signal Information • See signalInfo for more information
<i>pCallerIDInfo</i> [Optional]	• Caller ID Information • See callerIDInfo for more information
<i>pDispInfo</i> [Optional]	• Display Information
<i>pExtDispInfo</i> [Optional]	• Extended Display Information
<i>pCallerNameInfo</i> [Optional]	• Caller Name Information
<i>pCallWaitInd</i> [Optional]	• Call Waiting Indicator

<i>pConnectNum</i> <i>Info[Optional]</i>	<ul style="list-style-type: none"> • Connected Number Information • see connectNumInfo for more information
<i>pCallingParty</i> <i>Info[Optional]</i>	<ul style="list-style-type: none"> • Calling Party Number Information • This structure is having exactly same elements as connectNumInfo • see connectNumInfo for more information
<i>pCalledParty</i> <i>Info[Optional]</i>	<ul style="list-style-type: none"> • Called Party Number Information • see calledPartyInfo for more information
<i>pRedirNum</i> <i>Info[Optional]</i>	<ul style="list-style-type: none"> • Redirecting Number Information • see redirNumInfo for more information
<i>pCLIRCause</i> <i>Optional]</i>	<ul style="list-style-type: none"> • National Supplementary Services - CLIR • see NSSAudioCtrl for more information
<i>pNSSAudio</i> <i>Ctrl[Optional]</i>	<ul style="list-style-type: none"> • National Supplementary Services - Audio Control
<i>pNSSRelease</i> <i>Optional]</i>	<ul style="list-style-type: none"> • National Supplementary Services - Release
<i>pLineCtrlInfo</i> <i>Optional]</i>	<ul style="list-style-type: none"> • Line Control Information • see lineCtrlInfo for more information
<i>pExtDispRec</i> <i>Info[Optional]</i>	<ul style="list-style-type: none"> • Extended Display Record Information • see extDispRecInfo for more information

8.1103.2 Field Documentation

8.1103.2.1 **BYTE** `voicInfoRec::callID`

8.1103.2.2 **calledPartyInfo*** `voicInfoRec::pCalledPartyInfo`

8.1103.2.3 **callerIDInfo*** `voicInfoRec::pCallerIDInfo`

8.1103.2.4 **BYTE*** `voicInfoRec::pCallerNameInfo`

8.1103.2.5 **connectNumInfo*** `voicInfoRec::pCallingPartyInfo`

8.1103.2.6 **BYTE*** `voicInfoRec::pCallWaitInd`

- 8.1103.2.7 **BYTE*** voiceInfoRec::pCLIRCause
- 8.1103.2.8 **connectNumInfo*** voiceInfoRec::pConnectNumInfo
- 8.1103.2.9 **BYTE*** voiceInfoRec::pDisplInfo
- 8.1103.2.10 **BYTE*** voiceInfoRec::pExtDisplInfo
- 8.1103.2.11 **extDispRecInfo*** voiceInfoRec::pExtDispRecInfo
- 8.1103.2.12 **lineCtrlInfo*** voiceInfoRec::pLineCtrlInfo
- 8.1103.2.13 **NSSAudioCtrl*** voiceInfoRec::pNSSAudioCtrl
- 8.1103.2.14 **BYTE*** voiceInfoRec::pNSSRelease
- 8.1103.2.15 **redirNumInfo*** voiceInfoRec::pRedirNumInfo
- 8.1103.2.16 **signalInfo*** voiceInfoRec::pSignalInfo

8.1104 voiceManageCallsReq Struct Reference

Data Fields

- [BYTE SUPSType](#)
- [BYTE * pCallID](#)

8.1104.1 Detailed Description

This structure contains Manage Calls Information.

Parameters

<i>SUPSType</i>	<ul style="list-style-type: none"> • Supplementary service type during the call. <ul style="list-style-type: none"> – 0x01 - SUPS_TYPE_RELEASE_HELD_OR_WAITING <ul style="list-style-type: none"> * Release is held or waiting – 0x02 - SUPS_TYPE_RELEASE_ACTIVE_ACCEPT_HELD_OR_WAITING <ul style="list-style-type: none"> * Release is active and accepting held or waiting – 0x03 - SUPS_TYPE_HOLD_ACTIVE_ACCEPT_WAITING_OR_HELD <ul style="list-style-type: none"> * Hold is active and accepting waiting or held – 0x04 - SUPS_TYPE_HOLD_ALL_EXCEPT_SPECIFIED_CALL <ul style="list-style-type: none"> * Hold all calls except a specified one – 0x05 - SUPS_TYPE_MAKE_CONFERENCE_CALL <ul style="list-style-type: none"> * Make a conference call – 0x06 - SUPS_TYPE_EXPLICIT_CALL_TRANSFER <ul style="list-style-type: none"> * Explicit call transfer
Generated by Doxygen	<ul style="list-style-type: none"> – 0x07 - SUPS_TYPE_CCBS_ACTIVATION <ul style="list-style-type: none"> * Activate completion of calls to busy subscriber – 0x08 - SUPS_TYPE_END_ALL_CALLS <ul style="list-style-type: none"> * End all calls

<i>pCallID[Optional]</i>	<ul style="list-style-type: none">• Applicable only for SUPSType 0x04, 0x07, and 0x09
--------------------------	---

8.1104.2 Field Documentation

8.1104.2.1 **BYTE*** voiceManageCallsReq::pCallID

8.1104.2.2 **BYTE** voiceManageCallsReq::SUPSType

8.1105 voiceManageCallsResp Struct Reference

Data Fields

- **WORD*** pFailCause

8.1105.1 Detailed Description

This structure contains Failure cause Information. Populated when API Fails.

Parameters

<i>pFailCause</i>	<ul style="list-style-type: none">• Supplementary service failure causes (optional, supply NULL if not required).• See Table8 qaGobiApiTableVoiceCallEndReasons.h for supplementary services failure cause<ul style="list-style-type: none">– 0xFFFF is the value when the information is not received from device
-------------------	---

8.1105.2 Field Documentation

8.1105.2.1 **WORD*** voiceManageCallsResp::pFailCause

8.1106 voiceOrigUSSDNoWaitInfo Struct Reference

Data Fields

- struct [USSInfo](#) USSInformation

8.1106.1 Detailed Description

This structure contains Orig USSD No Wait Information Parameters.

Parameters

<i>USSInformation</i>	<ul style="list-style-type: none"> • See USSInfo for more information.
-----------------------	---

8.1106.2 Field Documentation

8.1106.2.1 struct USSInfo voiceOrigUSSDNoWaitInfo::USSInformation

8.1107 voiceOTASPStatusInfo Struct Reference

Data Fields

- [BYTE callID](#)
- [BYTE OTASPStatus](#)

8.1107.1 Detailed Description

This structure consist of OTASP or OTAPA event params

Parameters

<i>callID</i>	<ul style="list-style-type: none"> • Call identifier for the call.
<i>OTASPStatus</i>	<ul style="list-style-type: none"> • OTASP status for the OTASP call. Values: <ul style="list-style-type: none"> – 0x00 - OTASP_STATUS_SPL_UNLOCKED. SPL unlocked; only for user-initiated OTASP – 0x01 - OTASP_STATUS_SPRC_RETRIES_EXCEEDED. SPC retries exceeded; only for user-initiated OTASP – 0x02 - OTASP_STATUS_AKEY_EXCHANGED. A-key exchanged; only for user-initiated OTASP – 0x03 - OTASP_STATUS_SSD_UPDATED. SSD updated; for both user-initiated OTASP and network-initiated OTASP (OTAPA) – 0x04 - OTASP_STATUS_NAM_DOWNLOADED - NAM downloaded; only for user-initiated OTASP – 0x05 - OTASP_STATUS_MDN_DOWNLOADED - MDN downloaded; only for user-initiated OTASP – 0x06 - OTASP_STATUS_IMSI_DOWNLOADED - IMSI downloaded; only for user-initiated OTASP – 0x07 - OTASP_STATUS_PRL_DOWNLOADED - PRL downloaded; only for user-initiated OTASP – 0x08 - OTASP_STATUS_COMMITTED - Commit successful; only for user-initiated OTASP – 0x09 - OTASP_STATUS_OTAPA_STARTED - OTAPA started; only for network-initiated OTASP(OTAPA) – 0x0A - OTASP_STATUS_OTAPA_STOPPED - OTAPA stopped; only for network-initiated OTASP(OTAPA) – 0x0B - OTASP_STATUS_OTAPA_ABORTED - OTAPA aborted; only for network-initiated OTASP(OTAPA) – 0x0C - OTASP_STATUS_OTAPA_COMMITTED - OTAPA committed; only for network-initiated OTASP(OTAPA)
Generated by Doxygen	

8.1107.2 Field Documentation

8.1107.2.1 **BYTE** voiceOTASPStatusInfo::callID

8.1107.2.2 **BYTE** voiceOTASPStatusInfo::OTASPStatus

8.1108 voicePrivacyInfo Struct Reference

Data Fields

- [BYTE callID](#)
- [BYTE voicePrivacy](#)

8.1108.1 Detailed Description

Contains the parameters passed for SLQSVoiceSetPrivacyChangeCallBack by the device.

Parameters

<i>callID</i>	<ul style="list-style-type: none">• Unique identifier of the call for which the voice privacy is applicable. (mandatory)
<i>voicePrivacy</i>	<ul style="list-style-type: none">• Voice Privacy (mandatory)<ul style="list-style-type: none">– 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy– 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy

Note

None

8.1108.2 Field Documentation

8.1108.2.1 **BYTE** voicePrivacyInfo::callID

8.1108.2.2 **BYTE** voicePrivacyInfo::voicePrivacy

8.1109 voiceSetAllCallStatusCbkJInfo Struct Reference

Data Fields

- [arrCallInfo arrCallInfomation](#)
- [arrRemotePartyNum * pArrRemotePartyNum](#)
- [arrRemotePartyName * pArrRemotePartyName](#)
- [arrAlertingType * pArrAlertingType](#)

- [arrSvcOption](#) * [pArrSvcOption](#)
- [arrCallEndReason](#) * [pArrCallEndReason](#)
- [arrAlphaID](#) * [pArrAlphaID](#)
- [arrConnectPartyNum](#) * [pArrConnectPartyNum](#)
- [arrDiagInfo](#) * [pArrDiagInfo](#)
- [arrCalledPartyNum](#) * [pArrCalledPartyNum](#)
- [arrRedirPartyNum](#) * [pArrRedirPartyNum](#)
- [arrAlertingPattern](#) * [pArrAlertingPattern](#)

8.1109.1 Detailed Description

This structure contains VoiceCall Information parameters. [arrCallInformation](#) will be populated in case of change in the call information. Other paramters are optional therefore are populated based on device and technology type being used.

Parameters

arrCallInformation	[mandatory] <ul style="list-style-type: none"> • Array of Call Information This must be populated if Indication is received See arrCallInfo for more information. <ul style="list-style-type: none"> – Applicable for both "3GPP/3GPP2"
pArrRemotePartyNum	[optional] <ul style="list-style-type: none"> • Array of Remote Party Name.(NULL when not present) See arrRemotePartyNum for more information. <ul style="list-style-type: none"> – Applicable only for "3GPP/3GPP2"
pArrRemotePartyName	[optional] <ul style="list-style-type: none"> • Array of Alerting Type.(NULL when not present) See arrRemotePartyName for more information. <ul style="list-style-type: none"> – Applicable only for "3GPP"
pArrAlertingType	[optional] <ul style="list-style-type: none"> • Array of Alerting Type(NULL when not present) See arrAlertingType for more information. <ul style="list-style-type: none"> – Applicable only for "3GPP"
pArrSvcOption	[optional] <ul style="list-style-type: none"> • Array of Service Option.(NULL when not present) See arrSvcOption for more information. <ul style="list-style-type: none"> – Applicable only for "3GPP"
pArrCallEndReason	[optional] <ul style="list-style-type: none"> • Array of Call End Reason.(NULL when not present) See arrCallEndReason for more information. <ul style="list-style-type: none"> – Applicable only for "3GPP"
pArrAlphaID	[optional] <ul style="list-style-type: none"> • Array of Alpha Identifier(NULL when not present) See arrAlphaID for more information. <ul style="list-style-type: none"> – Applicable only for "3GPP"

<i>pArrConnect↔ PartyNum</i>	[optional] <ul style="list-style-type: none"> • Array of Connected Party Number.(NULL when not present) See arrConnectPartyNum for more information. <ul style="list-style-type: none"> – Applicable for both "3GPP/3GPP2"
<i>pArrDiagInfo</i>	[optional] <ul style="list-style-type: none"> • Array of Diagnostic Information.(NULL when not present) See arrDiagInfo for more information. <ul style="list-style-type: none"> – Applicable only for "3GPP"
<i>pArrCalled↔ PartyNum</i>	[optional] <ul style="list-style-type: none"> • Array of Called Party Number.(NULL when not present) See arrCalledPartyNum for more information. <ul style="list-style-type: none"> – Applicable only for "3GPP"
<i>pArrRedir↔ PartyNum</i>	[optional] <ul style="list-style-type: none"> • Array of Redirecting Party Number.(NULL when not present) See arrRedirPartyNum for more information. <ul style="list-style-type: none"> – Applicable only for "3GPP"
<i>pArrAlerting↔ Pattern</i>	[optional] <ul style="list-style-type: none"> • Array of Alerting Pattern.(NULL when not present) See arrAlertingPattern for more information. <ul style="list-style-type: none"> – Applicable only for "3GPP"

Note

Optional paramters would be NULL, if not received from the device.

8.1109.2 Field Documentation

8.1109.2.1 **arrCallInfo** voiceSetAllCallStatusCbkInfo::arrCallInfomation

8.1109.2.2 **arrAlertingPattern*** voiceSetAllCallStatusCbkInfo::pArrAlertingPattern

8.1109.2.3 **arrAlertingType*** voiceSetAllCallStatusCbkInfo::pArrAlertingType

8.1109.2.4 **arrAlphaID*** voiceSetAllCallStatusCbkInfo::pArrAlphaID

8.1109.2.5 **arrCalledPartyNum*** voiceSetAllCallStatusCbkInfo::pArrCalledPartyNum

8.1109.2.6 **arrCallEndReason*** voiceSetAllCallStatusCbkInfo::pArrCallEndReason

8.1109.2.7 **arrConnectPartyNum*** voiceSetAllCallStatusCbkInfo::pArrConnectPartyNum

8.1109.2.8 **arrDiagInfo*** voiceSetAllCallStatusCbkInfo::pArrDiagInfo

8.1109.2.9 **arrRedirPartyNum*** voiceSetAllCallStatusCbkInfo::pArrRedirPartyNum

8.1109.2.10 `arrRemotePartyName*` `voiceSetAllCallStatusCbklInfo::pArrRemotePartyName`

8.1109.2.11 `arrRemotePartyNum*` `voiceSetAllCallStatusCbklInfo::pArrRemotePartyNum`

8.1109.2.12 `arrSvcOption*` `voiceSetAllCallStatusCbklInfo::pArrSvcOption`

8.1110 voiceSetCallBarringPwdInfo Struct Reference

Data Fields

- [BYTE Reason](#)
- [BYTE oldPasswd](#) [4]
- [BYTE newPasswd](#) [4]
- [BYTE newPasswdAgain](#) [4]

8.1110.1 Detailed Description

This structure contains Voice Set Call Barring Password Request Parameters

Parameters

<i>Reason</i>	<ul style="list-style-type: none"> • Call Barring Reason • Values: <ul style="list-style-type: none"> – 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING - All outgoing – 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT - Outgoing internal – 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTOHOM - Outgoing external to home – 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING - All incoming – 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING - Roaming incoming – 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING - All calls are barred – 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING - All outgoing calls are barred – 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING - All incoming calls are barred
<i>oldPasswd</i> [PASSWORD_LENGTH]	<ul style="list-style-type: none"> • Old password. <ul style="list-style-type: none"> – Password consists of 4 ASCII digits. – Range: 0000 to 9999.
<i>newPasswd</i> [PASSWORD_LENGTH]	<ul style="list-style-type: none"> • New password. <ul style="list-style-type: none"> – Password consists of 4 ASCII digits. – Range: 0000 to 9999.
<i>newPasswdAgain</i> [PASSWORD_LENGTH]	<ul style="list-style-type: none"> • New password Again. <ul style="list-style-type: none"> – Password consists of 4 ASCII digits. – Range: 0000 to 9999.

8.1110.2 Field Documentation

8.1110.2.1 **BYTE** voiceSetCallBarringPwdInfo::newPasswd[4]

8.1110.2.2 **BYTE** voiceSetCallBarringPwdInfo::newPasswdAgain[4]

8.1110.2.3 **BYTE** voiceSetCallBarringPwdInfo::oldPasswd[4]

8.1110.2.4 **BYTE** voiceSetCallBarringPwdInfo::Reason

8.1111 voiceSetCallBarringPwdResp Struct Reference

Data Fields

- **WORD** * pFailCause
- **alphaIDInfo** * pAlphaIDInfo
- **BYTE** * pCCResType
- **BYTE** * pCallID
- **ccSUPSType** * pCCSUPSType

8.1111.1 Detailed Description

This structure contains Voice Set Call Barring Password Response Parameters

Parameters

<i>pFailCause</i>	<ul style="list-style-type: none"> • Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
<i>pAlphaIDInfo</i>	<ul style="list-style-type: none"> • Pointer to structure of alphaIDInfo (optional) <ul style="list-style-type: none"> – See alphaIDInfo for more information
<i>pCCResType</i>	<ul style="list-style-type: none"> • Call Control Result Type (optional) <ul style="list-style-type: none"> – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
<i>pCallID</i>	<ul style="list-style-type: none"> • Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
<i>pCCSUPSType</i>	<ul style="list-style-type: none"> • Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice.
	<ul style="list-style-type: none"> – See ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.1111.2 Field Documentation

8.1111.2.1 **alphaIDInfo*** voiceSetCallBarringPwdResp::pAlphaIDInfo

8.1111.2.2 **BYTE*** voiceSetCallBarringPwdResp::pCallID

8.1111.2.3 **BYTE*** voiceSetCallBarringPwdResp::pCCResType

8.1111.2.4 **ccSUPSType*** voiceSetCallBarringPwdResp::pCCSUPSType

8.1111.2.5 **WORD*** voiceSetCallBarringPwdResp::pFailCause

8.1112 voiceSetConfigReq Struct Reference

Data Fields

- [BYTE *](#) [pAutoAnswer](#)
- [airTimer *](#) [pAirTimerConfig](#)
- [roamTimer *](#) [pRoamTimerConfig](#)
- [BYTE *](#) [pTTYMode](#)
- [prefVoiceSO *](#) [pPrefVoiceSO](#)
- [BYTE *](#) [pPrefVoiceDomain](#)

8.1112.1 Detailed Description

This structure contains information about the Set Configuration Request Parameters.

Parameters

<i>pAutoAnswer</i>	<ul style="list-style-type: none"> • Value specified is written to NV_AUTO_ANSWER_I. (optional) • Values: <ul style="list-style-type: none"> – 0x00 - Disable – 0x01 - Enable
<i>pAirTimerConfig</i>	<ul style="list-style-type: none"> • Value specified is written to NV_AIR_CNT_I. (optional) • See airTimer for more information
<i>pRoamTimer↔ Config</i>	<ul style="list-style-type: none"> • Value specified is written to NV_ROAM_CNT_I. (optional) • See roamTimer for more information

<i>pTTYMode</i>	<ul style="list-style-type: none"> • Value specified is written to NV_TTY_I. (optional) • Values: <ul style="list-style-type: none"> – 0x00 - TTY_MODE_FULL - Full – 0x01 - TTY_MODE_VCO - Voice carry over – 0x02 - TTY_MODE_HCO - Hearing carry over – 0x03 - TTY_MODE_OFF - Off
<i>pPrefVoiceSO</i>	<ul style="list-style-type: none"> • Value specified is written to NV_PREF_VOICE_SO_I. (optional) • See prefVoiceSO for more information
<i>pPrefVoiceDomain</i>	<ul style="list-style-type: none"> • Preferred Voice-Domain. (optional) • Values: <ul style="list-style-type: none"> – 0x00 - VOICE_DOMAIN_PREF_CS_ONLY - Circuit-switched (CS) only – 0x01 - VOICE_DOMAIN_PREF_PS_ONLY - Packet-switched (PS) only – 0x02 - VOICE_DOMAIN_PREF_CS_PREF - CS is preferred, PS is secondary – 0x03 - VOICE_DOMAIN_PREF_PS_PREF - PS is preferred, CS is secondary

Note

One of the optional parameters must be present in the request.

8.1112.2 Field Documentation

8.1112.2.1 **airTimer*** voiceSetConfigReq::pAirTimerConfig

8.1112.2.2 **BYTE*** voiceSetConfigReq::pAutoAnswer

8.1112.2.3 **BYTE*** voiceSetConfigReq::pPrefVoiceDomain

8.1112.2.4 **prefVoiceSO*** voiceSetConfigReq::pPrefVoiceSO

8.1112.2.5 **roamTimer*** voiceSetConfigReq::pRoamTimerConfig

8.1112.2.6 **BYTE*** voiceSetConfigReq::pTTYMode

8.1113 voiceSetConfigResp Struct Reference**Data Fields**

- **BYTE *** [pAutoAnsStatus](#)
- **BYTE *** [pAirTimerStatus](#)
- **BYTE *** [pRoamTimerStatus](#)
- **BYTE *** [pTTYConfigStatus](#)
- **BYTE *** [pPrefVoiceSOStatus](#)
- **BYTE *** [pVoiceDomainPrefStatus](#)

8.1113.1 Detailed Description

This structure contains information about the Set Configuration Response Parameters.

Parameters

<i>pAutoAnsStatus</i>	<ul style="list-style-type: none"> • Auto Answer Status. (optional) • Values: <ul style="list-style-type: none"> – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.
<i>pAirTimerStatus</i>	<ul style="list-style-type: none"> • Air Timer Status. (optional) • Values: <ul style="list-style-type: none"> – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.
<i>pRoamTimer↔ Status</i>	<ul style="list-style-type: none"> • Roam Timer Status. (optional) • Values: <ul style="list-style-type: none"> – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.
<i>pTTYConfig↔ Status</i>	<ul style="list-style-type: none"> • TTY Config Status. (optional) • Values: <ul style="list-style-type: none"> – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.
<i>pPrefVoiceSO↔ Status</i>	<ul style="list-style-type: none"> • Preferred Voice SO Status. (optional) • Values: <ul style="list-style-type: none"> – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.
<i>pVoiceDomain↔ PrefStatus</i>	<ul style="list-style-type: none"> • Voice-Domain Preference Status. (optional) • Values: <ul style="list-style-type: none"> – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.

Note

Parameters which are mentioned as NULL will be ignored.

8.1113.2 Field Documentation

8.1113.2.1 **BYTE*** voiceSetConfigResp::pAirTimerStatus

8.1113.2.2 **BYTE*** voiceSetConfigResp::pAutoAnsStatus

8.1113.2.3 **BYTE*** voiceSetConfigResp::pPrefVoiceSOStatus

8.1113.2.4 **BYTE*** voiceSetConfigResp::pRoamTimerStatus

8.1113.2.5 **BYTE*** voiceSetConfigResp::pTTYConfigStatus

8.1113.2.6 **BYTE*** voiceSetConfigResp::pVoiceDomainPrefStatus

8.1114 voiceSetPrefPrivacy Struct Reference

Data Fields

- [BYTE](#) [privacyPref](#)

8.1114.1 Detailed Description

This structure contains the preferred voice privacy values.

Parameters

<i>privacyPref</i>	<ul style="list-style-type: none">• Voice Privacy Preference<ul style="list-style-type: none">– 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy– 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy
--------------------	---

8.1114.2 Field Documentation

8.1114.2.1 **BYTE** voiceSetPrefPrivacy::privacyPref

8.1115 voiceSetSUPSServiceReq Struct Reference

Data Fields

- [BYTE](#) [voiceSvc](#)
- [BYTE](#) [reason](#)
- [BYTE](#) * [pServiceClass](#)
- [BYTE](#) * [pCallBarringPasswd](#)
- [BYTE](#) * [pCallForwardingNumber](#)
- [BYTE](#) * [pTimerVal](#)
- [callFwdTypeAndPlan](#) * [pCallFwdTypeAndPlan](#)

8.1115.1 Detailed Description

This structure contains Supplementary Service request parameters related to different features and their activation, deactivation, registration and erasure (applicable only for 3GPP)

Parameters

<i>voiceSvc</i>	<ul style="list-style-type: none"> Manages all call-independent supplementary services, such as activation, deactivation, registration, and erasure (mandatory) <ul style="list-style-type: none"> 0x01 - VOICE_SERVICE_ACTIVATE 0x02 - VOICE_SERVICE_DEACTIVATE 0x03 - VOICE_SERVICE_REGISTER 0x04 - VOICE_SERVICE_ERASE
<i>reason</i>	<ul style="list-style-type: none"> supplementary service reason values (mandatory) <ul style="list-style-type: none"> 0x01 - QMI_VOICE_REASON_FWD_UNCONDITIONAL Unconditional call forwarding 0x02 - QMI_VOICE_REASON_FWD_MOBILEBUSY Forward when the mobile is busy 0x03 - QMI_VOICE_REASON_FWD_NOREPLY Forward when there is no reply 0x04 - QMI_VOICE_REASON_FWD_UNREACHABLE Forward when the call is unreachable 0x05 - QMI_VOICE_REASON_FWD_ALLFORWARDING All forwarding 0x06 - QMI_VOICE_REASON_FWD_ALLCONDITIONAL All conditional forwarding 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING All outgoing calls are barred 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT Outgoing internal calls are barred 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTTOHOME Outgoing calls external to home are barred 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING All incoming calls are barred 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING Roaming incoming calls are barred 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING All calls are barred 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING All outgoing calls are barred 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING All incoming calls are barred 0x0F - QMI_VOICE_REASON_CALLWAITING Call waiting
<i>pServiceClass</i>	<ul style="list-style-type: none"> Service class is a combination (sum) of information class constants (optional) <ul style="list-style-type: none"> See serviceClassInformation for more information
<i>pCallBarring</i> <i>Passwd</i>	<ul style="list-style-type: none"> Password is required if call barring is provisioned using a password. Password consists of 4 ASCII digits. Range: 0000 to 9999 (optional)

<i>pCallForwardingNumber</i>	<ul style="list-style-type: none"> Call forwarding number to be registered with the network. This has to be included in the request only when the service is set to VOICE_SERVICE_REGISTER. NULL terminated ASCII string. (optional)
<i>pTimerVal</i>	<ul style="list-style-type: none"> Call forwarding no reply timer value in seconds. This has to be included in the request only when the service is set to VOICE_SERVICE_REGISTER and the reason is QMI_VOICE_REASON_FWD_NOREPLY. (optional) <ul style="list-style-type: none"> Range: 5 to 30 in steps of 5
<i>pCallFwdTypeAndPlan</i>	<ul style="list-style-type: none"> Information about call forwarding type and plan. This parameter is ignored when the Call Forwarding Number is not included (optional) <ul style="list-style-type: none"> See callFwdTypeAndPlan for more information

8.1115.2 Field Documentation

8.1115.2.1 **BYTE*** voiceSetSUPSServiceReq::pCallBarringPasswd

8.1115.2.2 **BYTE*** voiceSetSUPSServiceReq::pCallForwardingNumber

8.1115.2.3 **callFwdTypeAndPlan*** voiceSetSUPSServiceReq::pCallFwdTypeAndPlan

8.1115.2.4 **BYTE*** voiceSetSUPSServiceReq::pServiceClass

8.1115.2.5 **BYTE*** voiceSetSUPSServiceReq::pTimerVal

8.1115.2.6 **BYTE** voiceSetSUPSServiceReq::reason

8.1115.2.7 **BYTE** voiceSetSUPSServiceReq::voiceSvc

8.1116 voiceSetSUPSServiceResp Struct Reference

Data Fields

- WORD*** pFailCause
- alphaIDInfo*** pAlphaIDInfo
- BYTE*** pCCResultType
- BYTE*** pCallID
- ccSUPSType*** pCCSUPSType

8.1116.1 Detailed Description

This structure contains Supplementary Service response parameters related to different features and their activation, deactivation, registration and erasure (applicable only for 3GPP)

Parameters

<i>pFailCause</i>	<ul style="list-style-type: none"> Supplementary service failure causes (optional, supply NULL if not required). <ul style="list-style-type: none"> 0xFFFF is the value when the information is not received from device
<i>pAlphaIDInfo</i>	<ul style="list-style-type: none"> Pointer to structure of alphaIDInfo. The parameter used to pass the alpha (if any) given by the SIM/R-UIM after call control (optional, supply NULL if not required) <ul style="list-style-type: none"> See alphaIDInfo for more information
<i>pCCResultType</i>	<ul style="list-style-type: none"> Call control result types (optional, supply NULL if not required) <ul style="list-style-type: none"> 0x00 - CC_RESULT_TYPE_VOICE - Voice 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service 0xFF - if the device does not provide this information
<i>pCallID</i>	<ul style="list-style-type: none"> Unique call identifier for the dialed call (optional, supply NULL if not required) <ul style="list-style-type: none"> 0x00 - if the device does not provide this information
<i>pCCSUPSType</i>	<ul style="list-style-type: none"> Data is present when pCCResultType is present and is other than Voice. (optional, supply NULL if not required) <ul style="list-style-type: none"> See ccSUPSType for more information

8.1116.2 Field Documentation

8.1116.2.1 **alphaIDInfo*** voiceSetSUPSServiceResp::pAlphaIDInfo

8.1116.2.2 **BYTE*** voiceSetSUPSServiceResp::pCallID

8.1116.2.3 **BYTE*** voiceSetSUPSServiceResp::pCCResultType

8.1116.2.4 **ccSUPSType*** voiceSetSUPSServiceResp::pCCSUPSType

8.1116.2.5 **WORD*** voiceSetSUPSServiceResp::pFailCause

8.1117 voiceStopContDTMFinfo Struct Reference

Data Fields

- BYTE** callID

8.1117.1 Detailed Description

This structure contains parameters of stop continuous DTMF

Parameters

<i>pCallID</i> [IN/OUT]	<ul style="list-style-type: none"> • Call ID associated with call on which the DTMF information has to be sent. Stop continuous DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device
-------------------------	--

8.1117.2 Field Documentation

8.1117.2.1 BYTE voiceStopContDTMFInfo::callID

8.1118 voiceSUPSInfo Struct Reference

Data Fields

- [SUPSInfo SUPSInformation](#)
- [BYTE * pSvcClass](#)
- [BYTE * pReason](#)
- [BYTE * pCallFWNum](#)
- [BYTE * pCallFWTimerVal](#)
- [struct USSInfo * pUSSInfo](#)
- [BYTE * pCallID](#)
- [alphaIDInfo * pAlphaIDInfo](#)
- [BYTE * pCallBarPasswd](#)
- [newPwdData * pNewPwdData](#)
- [BYTE * pDataSrc](#)
- [WORD * pFailCause](#)
- [getCallFWInfo * pCallFwdInfo](#)
- [CLIRResp * pCLIRstatus](#)
- [CLIPResp * pCLIPstatus](#)
- [COLPResp * pCOLPstatus](#)
- [COLRResp * pCOLRstatus](#)
- [CNAPResp * pCNAPstatus](#)

8.1118.1 Detailed Description

This structure contains the parameters passed for SLQSVoiceSetSUPSCallBack by the device.

Parameters

<i>SUPS</i> <i>Information(mandatory)</i>	See SUPSInfo for more information.
<i>pSvc</i> <i>Class(optional)</i>	<ul style="list-style-type: none"> Service class is a combination (sum) of information class constants (optional) See qaGobiApiTableSupServiceInfoClasses.h for service classes.
<i>p</i> <i>Reason(optional)</i>	<ul style="list-style-type: none"> See qaGobiApiTableCallControlReturnReasons.h for return reasons.
<i>pCallFW</i> <i>Num(optional)</i>	<ul style="list-style-type: none"> Call forwarding number to be registered with the network. ASCII String, NULL terminated.
<i>pCallFWTimer</i> <i>Val(optional)</i>	<ul style="list-style-type: none"> Call Forwarding No Reply Timer. <ul style="list-style-type: none"> Range: 5 to 30 in steps of 5.
<i>pUSS</i> <i>Info(optional)</i>	<ul style="list-style-type: none"> See USSInfo for more information.
<i>pCallID(optional)</i>	<ul style="list-style-type: none"> Call identifier of the voice call that has been modified to a supplementary service as a result of call control.
<i>pAlphaID</i> <i>Info(optional)</i>	<ul style="list-style-type: none"> See alphaIDInfo for more information.
<i>pCallBar</i> <i>Passwd(optional)</i>	<ul style="list-style-type: none"> Password is required if call barring is provisioned using a password. <ul style="list-style-type: none"> Password consists of 4 ASCII digits. Range: 0000 to 9999. This also serves as the old password in the register password scenario.
<i>pNewPwd</i> <i>Data(optional)</i>	<ul style="list-style-type: none"> See newPwdData for more information.
<i>pData</i> <i>Src(optional)</i>	<ul style="list-style-type: none"> Sups Data Source. Used to distinguish between the supplementary service data sent to the network and the response received from the network. If absent, the supplementary service data in this indication can be assumed as a request sent to the network.
<i>pFail</i> <i>Cause(optional)</i>	<ul style="list-style-type: none"> Supplementary services failure cause. See qaGobiApiTableVoiceCallEndReasons.h for more information.
<i>pCallFwd</i> <i>Info(optional)</i>	<ul style="list-style-type: none"> See getCallFWInfo for more information.

<i>pCLI</i> ↔ <i>Rstatus(optional)</i>	<ul style="list-style-type: none"> • See CLIRResp for more information.
<i>pCLI</i> ↔ <i>Pstatus(optional)</i>	<ul style="list-style-type: none"> • See CLIPResp for more information.
<i>pCOL</i> ↔ <i>Pstatus(optional)</i>	<ul style="list-style-type: none"> • See COLPResp for more information.
<i>pCOL</i> ↔ <i>Rstatus(optional)</i>	<ul style="list-style-type: none"> • See COLRResp for more information.
<i>pCNA</i> ↔ <i>Pstatus(optional)</i>	<ul style="list-style-type: none"> • See CNAPResp for more information.

Note

None

8.1118.2 Field Documentation

8.1118.2.1 **alphaIDInfo*** voiceSUPSInfo::pAlphaIDInfo8.1118.2.2 **BYTE*** voiceSUPSInfo::pCallBarPasswd8.1118.2.3 **getCallFWInfo*** voiceSUPSInfo::pCallFwdInfo8.1118.2.4 **BYTE*** voiceSUPSInfo::pCallFWNum8.1118.2.5 **BYTE*** voiceSUPSInfo::pCallFWTimerVal8.1118.2.6 **BYTE*** voiceSUPSInfo::pCallIID8.1118.2.7 **CLIPResp*** voiceSUPSInfo::pCLIPstatus8.1118.2.8 **CLIRResp*** voiceSUPSInfo::pCLIRstatus8.1118.2.9 **CNAPResp*** voiceSUPSInfo::pCNAPstatus8.1118.2.10 **COLPResp*** voiceSUPSInfo::pCOLPstatus8.1118.2.11 **COLRResp*** voiceSUPSInfo::pCOLRstatus8.1118.2.12 **BYTE*** voiceSUPSInfo::pDataSrc

8.1118.2.13 **WORD*** voiceSUPSInfo::pFailCause

8.1118.2.14 **newPwdData*** voiceSUPSInfo::pNewPwdData

8.1118.2.15 **BYTE*** voiceSUPSInfo::pReason

8.1118.2.16 **BYTE*** voiceSUPSInfo::pSvcClass

8.1118.2.17 **struct USSInfo*** voiceSUPSInfo::pUSSInfo

8.1118.2.18 **SUPInfo** voiceSUPSInfo::SUPSInformation

8.1119 voiceSUPSNotification Struct Reference

Data Fields

- [BYTE](#) callID
- [BYTE](#) notifType
- [WORD](#) * pCUGIndex
- [ECTNum](#) * pECTNum

8.1119.1 Detailed Description

Contains the parameters passed for SLQSVoiceSetSUPSNotificationCallback by the device.

Parameters

<i>callID</i>	<ul style="list-style-type: none">• Unique identifier of the call for which the notification is applicable. (mandatory)
---------------	---

<i>notifType</i>	<ul style="list-style-type: none"> • Notification type parameter (mandatory) <ul style="list-style-type: none"> – 0x01 - NOTIFICATION_TYPE_OUTGOING_CALL_IS_FORWARDED Originated MO call is being forwarded to another user – 0x02 - NOTIFICATION_TYPE_OUTGOING_CALL_IS_WAITING Originated MO call is waiting at the called user – 0x03 - NOTIFICATION_TYPE_OUTGOING_CUG_CALL Outgoing call is a CUG call – 0x04 - NOTIFICATION_TYPE_OUTGOING_CALLS_BARRED Outgoing calls are barred – 0x05 - NOTIFICATION_TYPE_OUTGOING_CALL_IS_DEFLECTED Outgoing call is deflected – 0x06 - NOTIFICATION_TYPE_INCOMING_CUG_CALL Incoming call is a CUG call – 0x07 - NOTIFICATION_TYPE_INCOMING_CALLS_BARRED Incoming calls are barred – 0x08 - NOTIFICATION_TYPE_INCOMING_FORWARDED_CALL Incoming call received is a forwarded call – 0x09 - NOTIFICATION_TYPE_INCOMING_DEFLECTED_CALL Incoming call is a deflected call – 0x0A - NOTIFICATION_TYPE_INCOMING_CALL_IS_FORWARDED Incoming call is forwarded to another user – 0x0B - NOTIFICATION_TYPE_UNCOND_CALL_FORWARD_ACTIVE Unconditional call forwarding is active – 0x0C - NOTIFICATION_TYPE_COND_CALL_FORWARD_ACTIVE Conditional call forwarding is active – 0x0D - NOTIFICATION_TYPE_CLIR_SUPPRESSION_REJECTED CLIR suppression is rejected – 0x0E - NOTIFICATION_TYPE_CALL_IS_ON_HOLD Call is put on hold at the remote party – 0x0F - NOTIFICATION_TYPE_CALL_IS_RETRIEVED Call is retrieved at the remote party from the hold state – 0x10 - NOTIFICATION_TYPE_CALL_IS_IN_MPTY Call is in a conference – 0x11 - NOTIFICATION_TYPE_INCOMING_CALL_IS_ECT Incoming call is an explicit call transfer
<i>pCUGIndex</i>	<ul style="list-style-type: none"> • The CUG Index used to indicate that the incoming/outgoing call is a CUG call. (optional, NULL when not present) Range: 0x00 to 0x7FFF.
<i>pECTNum</i>	<ul style="list-style-type: none"> • The ECT Number is used to indicate that the incoming call is an explicitly transferred call. (optional, NULL when not present) Refer ECTNum for details.

Note

None

8.1119.2 Field Documentation

8.1119.2.1 **BYTE** voiceSUPSNotification::callID

8.1119.2.2 **BYTE** voiceSUPSNotification::notifType

8.1119.2.3 **WORD*** voiceSUPSNotification::pCUGIndex

8.1119.2.4 **ECTNum*** voiceSUPSNotification::pECTNum

8.1120 wcdmaCellInfo Struct Reference

Data Fields

- [WORD](#) psc
- [SHORT](#) cpich_rscp
- [SHORT](#) cpich_ecno
- [SHORT](#) srxlev

8.1120.1 Detailed Description

This structure contains information about the WCDMA Cell.

Parameters

<i>psc</i>	<ul style="list-style-type: none">• Primary scrambling code.• Range: 0 to 511.
<i>cpich_rscp</i>	<ul style="list-style-type: none">• Absolute power level (in 1/10 dBm) of the common pilot channel as received by the UE.• Range: -120.0 dBm to -25.0 dBm
<i>cpich_ecno</i>	<ul style="list-style-type: none">• CPICH Ec/No; ratio (in 1/10 dB) of the received energy per PN chip for the CPICH to the total received power spectral density at the UE antenna connector.• Range: -50.0 dB to 0.
<i>srxlev</i>	<ul style="list-style-type: none">• Cell selection Rx level (Srxlev) value.• Range: -128 to 128.• This field is only valid when ue_in_idle is TRUE.

8.1120.2 Field Documentation

8.1120.2.1 **SHORT** wcdmaCellInfo::cpich_ecno

8.1120.2.2 **SHORT** wcdmaCellInfo::cpich_rscp

8.1120.2.3 **WORD** wcdmaCellInfo::psc

8.1120.2.4 **SHORT** wcdmaCellInfo::srxlev

8.1121 WCDMAECIOThresh Struct Reference

Data Fields

- [BYTE](#) WCDMAECIOThreshListLen
- [WORD](#) * pWCDMAECIOThreshList

8.1121.1 Detailed Description

This structure contains WCDMA ECIO threshold related parameters.

Parameters

<i>WCDMAECIO</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> • Length of the WCDMA ECIO threshold list parameter to follow
<i>pWCDMAECIO</i> ↔ <i>ThreshList</i>	<ul style="list-style-type: none"> • Array of ECIO thresholds (in units of 0.1 dB) • Maximum of 32 values • Range for ECIO values: -31.5 to 0 (in dB)

8.1121.2 Field Documentation

8.1121.2.1 **WORD*** WCDMAECIOThresh::pWCDMAECIOThreshList

8.1121.2.2 **BYTE** WCDMAECIOThresh::WCDMAECIOThreshListLen

8.1122 WCDMAInfoLTENeighborCell Struct Reference

Data Fields

- [ULONG](#) wcdmaRRCTest
- [BYTE](#) umtsLTENbrCellLen
- [umtsLTENbrCell](#) UMTSLTENbrCell [255]

8.1122.1 Detailed Description

This structure contains information about the WCDMA - LTE Neighboring Cell Info Set.

Parameters

<i>wcdmaRRC</i> <i>State</i>	<ul style="list-style-type: none"> • WCDMA RRC states. • Defined in 3GPP TS 25.331 • Values: <ul style="list-style-type: none"> – 0x00 - NAS_WCDMA_RRC_STATE_DISCONNECTED * WCDMA RRC State is IDLE – 0x01 - NAS_WCDMA_RRC_STATE_CELL_PCH * WCDMA RRC state is CELL_PCH – 0x02 - NAS_WCDMA_RRC_STATE_URA_PCH * WCDMA RRC state is URA_PCH – 0x03 - NAS_WCDMA_RRC_STATE_CELL_FACH * WCDMA RRC state is CELL_FACH – 0x04 - NAS_WCDMA_RRC_STATE_CELL_DCH * WCDMA RRC state is CELL_DCH
<i>umtsLTENbr</i> <i>CellLen</i>	<ul style="list-style-type: none"> • Number of sets of UMTS LTE Neighbors.
<i>UMTSLTENbr</i> <i>Cell</i>	<ul style="list-style-type: none"> • See umtsLTENbrCell for more information.

8.1122.2 Field Documentation

8.1122.2.1 `umtsLTENbrCell WCDMAInfoLTENeighborCell::UMTSLTENbrCell[255]`

8.1122.2.2 `BYTE WCDMAInfoLTENeighborCell::umtsLTENbrCellLen`

8.1122.2.3 `ULONG WCDMAInfoLTENeighborCell::wcdmaRRCState`

8.1123 wcdmaLongMsgDecodingParams Struct Reference

Data Fields

- `BYTE * pMessage`
- `BYTE * pSenderAddrLength`
- `CHAR * pSenderAddr`
- `BYTE * pTextMsgLength`
- `CHAR * pTextMsg`
- `BYTE * pScAddrLength`
- `CHAR * pScAddr`
- `BYTE Time [0x09]`
- `BYTE Date [0x09]`
- `BYTE * pReferenceNum`
- `BYTE * pTotalNum`
- `BYTE * pPartNum`
- `BOOL * plsUDHPresent`

8.1123.1 Detailed Description

Structure contains parameters which need to be decoded from message

Parameters

<i>pMessage</i> [IN]	<ul style="list-style-type: none"> Message read off the device via SLQSGetSMS
<i>pSenderAddr</i> ↔ <i>Length</i> [IN/OUT]	<ul style="list-style-type: none"> Upon input, indicates the maximum number of ASCII characters (including NULL termination) that the pSenderAddr buffer can accommodate. A length with 24 will be much safe since this address field can be up to 12 octets (24 bytes) Upon successful output, returns the length of destination address string.
<i>pSenderAddr</i> ↔ <i>OUT</i>	<ul style="list-style-type: none"> Note that a length with 24 bytes will be much safe. Returns NULL-terminated ASCII String containing destination address
<i>pTextMsg</i> ↔ <i>Length</i> [IN/OUT]	<ul style="list-style-type: none"> Upon input, specifies the number of characters the given text message buffer can accommodate. Upon successful output, returns the number of characters returns in the given text message buffer.
<i>pTextMsg</i> [OUT]	<ul style="list-style-type: none"> Encoded PDU message
<i>pScAddr</i> ↔ <i>Length</i> [IN/OUT]	<ul style="list-style-type: none"> A length with 24 will be much safe since this address field can be up to 12 octets (24 bytes) Returns NULL-terminated ASCII String containing destination address
<i>pScAddr</i> [OUT]	<ul style="list-style-type: none"> Note that a length with 24 bytes will be much safe. Returns NULL-terminated ASCII String containing service center address. This SMSC field contains the Type of Address. To get the exact SMSC address, skip the first two bytes. e.g, 9085290100334, 90 is the Type of Address, indicates international format of phone number
<i>pTime</i> [OUT]	<ul style="list-style-type: none"> Time fetched from message
<i>pReference</i> ↔ <i>Num</i> [OUT]	<ul style="list-style-type: none"> Reference number of the sms
<i>pTotalNum</i> [OUT]	<ul style="list-style-type: none"> Total number of the concatenated message
<i>pPartNum</i> [OUT]	<ul style="list-style-type: none"> Sequence number of the current message
<i>pIsUDHPresent</i>	<ul style="list-style-type: none"> Is User Data Header Present in the PDU? If yes, it means it is a concatenated SMS.

8.1123.2 Field Documentation

- 8.1123.2.1 **BYTE** wcdmaLongMsgDecodingParams::Date[0x09]
- 8.1123.2.2 **BOOL*** wcdmaLongMsgDecodingParams::plsUDHPresent
- 8.1123.2.3 **BYTE*** wcdmaLongMsgDecodingParams::pMessage
- 8.1123.2.4 **BYTE*** wcdmaLongMsgDecodingParams::pPartNum
- 8.1123.2.5 **BYTE*** wcdmaLongMsgDecodingParams::pReferenceNum
- 8.1123.2.6 **CHAR*** wcdmaLongMsgDecodingParams::pScAddr
- 8.1123.2.7 **BYTE*** wcdmaLongMsgDecodingParams::pScAddrLength
- 8.1123.2.8 **CHAR*** wcdmaLongMsgDecodingParams::pSenderAddr
- 8.1123.2.9 **BYTE*** wcdmaLongMsgDecodingParams::pSenderAddrLength
- 8.1123.2.10 **CHAR*** wcdmaLongMsgDecodingParams::pTextMsg
- 8.1123.2.11 **BYTE*** wcdmaLongMsgDecodingParams::pTextMsgLength
- 8.1123.2.12 **BYTE*** wcdmaLongMsgDecodingParams::pTotalNum
- 8.1123.2.13 **BYTE** wcdmaLongMsgDecodingParams::Time[0x09]

8.1124 wcdmaMsgDecodingParams Struct Reference

Data Fields

- **BYTE *** pMessage
- **BYTE *** pSenderAddrLength
- **CHAR *** pSenderAddr
- **BYTE *** pTextMsgLength
- **CHAR *** pTextMsg
- **BYTE *** pScAddrLength
- **CHAR *** pScAddr
- **BYTE** Time [0x09]
- **BYTE** Date [0x09]

8.1124.1 Detailed Description

Structure contains parameters which need to be decoded from message

Parameters

<i>pMessage</i> [IN]	<ul style="list-style-type: none"> • Message read off the device via SLQSGetSMS
<i>pSenderAddr</i> ↔ <i>Length</i> [IN/OUT]	<ul style="list-style-type: none"> • Upon input, indicates the maximum number of ASCII characters (including NULL termination) that the pSenderAddr buffer can accommodate. A length with 24 will be much safe since this address field can be up to 12 octets (24 bytes) Upon successful output, returns the length of destination address string.
<i>pSenderAddr</i> ↔ <i>OUT</i>	<ul style="list-style-type: none"> • Note that a length with 24 bytes will be much safe. Returns NULL-terminated ASCII String containing destination address
<i>pTextMsg</i> ↔ <i>Length</i> [IN/OUT]	<ul style="list-style-type: none"> • Upon input, specifies the number of characters the given text message buffer can accommodate. Upon successful output, returns the number of characters returns in the given text message buffer.
<i>pTextMsg</i> [OUT]	<ul style="list-style-type: none"> • Encoded PDU message
<i>pScAddr</i> ↔ <i>Length</i> [IN/OUT]	<ul style="list-style-type: none"> • A length with 24 will be much safe since this address field can be up to 12 octets (24 bytes) Returns NULL-terminated ASCII String containing destination address
<i>pScAddr</i> [OUT]	<ul style="list-style-type: none"> • Note that a length with 24 bytes will be much safe. Returns NULL-terminated ASCII String containing service center address. This SMSC field contains the Type of Address. To get the exact SMSC address, skip the first two bytes. e.g, 9085290100334, 90 is the Type of Address, indicates international format of phone number
<i>pTime</i> [OUT]	<ul style="list-style-type: none"> • Time fetched from message
<i>pDate</i>	<ul style="list-style-type: none"> • Date fetched from message

8.1124.2 Field Documentation

8.1124.2.1 **BYTE** wcdmaMsgDecodingParams::Date[0x09]

8.1124.2.2 **BYTE*** wcdmaMsgDecodingParams::pMessage

8.1124.2.3 **CHAR*** wcdmaMsgDecodingParams::pScAddr

8.1124.2.4 **BYTE*** wcdmaMsgDecodingParams::pScAddrLength

8.1124.2.5 **CHAR*** wcdmaMsgDecodingParams::pSenderAddr

8.1124.2.6 **BYTE*** wcdmaMsgDecodingParams::pSenderAddrLength

8.1124.2.7 CHAR* wcdmaMsgDecodingParams::pTextMsg

8.1124.2.8 BYTE* wcdmaMsgDecodingParams::pTextMsgLength

8.1124.2.9 BYTE wcdmaMsgDecodingParams::Time[0x09]

8.1125 wcdmaMsgEncodingParams Struct Reference

Data Fields

- ULONG messageSize
- CHAR * pDestAddr
- CHAR * pTextMsg
- CHAR * pPDUMessage
- BYTE alphabet

8.1125.1 Detailed Description

Structure contains parameters which need to encoded with message

Parameters

<i>messageSize</i>	<ul style="list-style-type: none"> • The length of the message contents in bytes
<i>pDestAddr[IN]</i>	<ul style="list-style-type: none"> • Gives NULL-terminated ASCII String containing destination address
<i>pTextMsg[IN]</i>	<ul style="list-style-type: none"> • Text message to be encoded, maximum limit is 160 charaters
<i>pPDU↔ Message[OUT]</i>	<ul style="list-style-type: none"> • Encoded PDU message
<i>alphabet[IN]</i>	<ul style="list-style-type: none"> • Encoding method to generate the PDU <ul style="list-style-type: none"> – 0 - 7 bit encoding – 4 - 8 bit encoding – 8 - 16 bit UCS2 encoding – others value will be treated as default 7 bit encoding

8.1125.2 Field Documentation

8.1125.2.1 BYTE wcdmaMsgEncodingParams::alphabet

8.1125.2.2 ULONG wcdmaMsgEncodingParams::messageSize

8.1125.2.3 **CHAR*** wcdmaMsgEncodingParams::pDestAddr

8.1125.2.4 **CHAR*** wcdmaMsgEncodingParams::pPDUMessage

8.1125.2.5 **CHAR*** wcdmaMsgEncodingParams::pTextMsg

8.1126 WCDMARSSIThresh Struct Reference

Data Fields

- [BYTE](#) WCDMARSSIThreshListLen
- [WORD](#) * pWCDMARSSIThreshList

8.1126.1 Detailed Description

This structure contains WCDMA RSSI threshold related parameters.

Parameters

<i>WCDMARSSI</i> ↔ <i>ThreshListLen</i>	<ul style="list-style-type: none"> • Length of the WCDMA RSSI threshold list parameter to follow
<i>pWCDMARSSI</i> ↔ <i>IThreshList</i>	<ul style="list-style-type: none"> • Array of RSSI thresholds (in units of 0.1 dBm) • Maximum of 32 values. • Range for RSSI values: -121 to 0 (in dBm)

8.1126.2 Field Documentation

8.1126.2.1 **WORD*** WCDMARSSIThresh::pWCDMARSSIThreshList

8.1126.2.2 **BYTE** WCDMARSSIThresh::WCDMARSSIThreshListLen

8.1127 WCDMASysInfo Struct Reference

Data Fields

- [sysInfoCommon](#) sysInfoWCDMA
- [BYTE](#) lacValid
- [WORD](#) lac
- [BYTE](#) cellIdValid
- [ULONG](#) cellId
- [BYTE](#) regRejectInfoValid
- [BYTE](#) rejectSrvDomain
- [BYTE](#) rejCause

- [BYTE networkIdValid](#)
- [BYTE MCC \[3\]](#)
- [BYTE MNC \[3\]](#)
- [BYTE hsCallStatusValid](#)
- [BYTE hsCallStatus](#)
- [BYTE hsIndValid](#)
- [BYTE hsInd](#)
- [BYTE pscValid](#)
- [WORD psc](#)

8.1127.1 Detailed Description

Structure for storing the WCDMA System Information.

Parameters

<i>sysInfoWCDMA</i>	<ul style="list-style-type: none"> • See sysInfoCommon for more information.
<i>lacValid</i>	<ul style="list-style-type: none"> • Indicates whether the location area code is valid.. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>lac</i>	<ul style="list-style-type: none"> • Location area code. • Only applies to 3GPP. <ul style="list-style-type: none"> – 0xFFFF - Not Available
<i>cellIdValid</i>	<ul style="list-style-type: none"> • Indicates whether the cell ID is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>cellId</i>	<ul style="list-style-type: none"> • Cell ID. <ul style="list-style-type: none"> – 0xFFFFFFFF - Not Available
<i>regRejectInfoValid</i>	<ul style="list-style-type: none"> • Indicates whether the registration reject information is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>rejectSrvDomain</i>	<ul style="list-style-type: none"> • Type of service domain in which the registration is rejected. <ul style="list-style-type: none"> – 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped
Generated by Doxygen	<ul style="list-style-type: none"> – 0xFF - Not Available

<i>rejCause</i>	<ul style="list-style-type: none"> Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6]. <ul style="list-style-type: none"> 0xFF - Not Available
<i>networkIdValid</i>	<ul style="list-style-type: none"> Indicates whether the network ID is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>MCC[PLMN_L↔ ENGTH]</i>	<ul style="list-style-type: none"> Mobile Country Code. MCC digits in ASCII characters
<i>MNC[PLMN_L↔ ENGTH]</i>	<ul style="list-style-type: none"> Mobile Network Code. MNC digits in ASCII characters An unused byte is set to 0xFF. In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.
<i>hsCallStatus↔ Valid</i>	<ul style="list-style-type: none"> Indicates whether the high-speed call status is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available
<i>hsCallStatus</i>	<ul style="list-style-type: none"> Call status on high speed. Only applicable for WCDMA. <ul style="list-style-type: none"> 0x00 - HSDPA and HSUPA are unsupported 0x01 - HSDPA is supported 0x02 - HSUPA is supported 0x03 - HSDPA and HSUPA are supported 0x04 - HSDPA+ is supported 0x05 - HSDPA+ and HSUPA are supported 0x06 - Dual-cell HSDPA+ is supported 0x07 - Dual-cell HSDPA+ and HSUPA are supported 0xFF - Not Available
<i>hsIndValid</i>	<ul style="list-style-type: none"> Indicates whether high-speed service indication is valid. <ul style="list-style-type: none"> 0x00 - Invalid 0x01 - Valid 0xFF - Not Available

<i>hsInd</i>	<ul style="list-style-type: none"> • High-speed service indication • Only applicable for WCDMA. <ul style="list-style-type: none"> – 0x00 - HSDPA and HSUPA are unsupported – 0x01 - HSDPA is supported – 0x02 - HSUPA is supported – 0x03 - HSDPA and HSUPA are supported – 0x04 - HSDPA+ is supported – 0x05 - HSDPA+ and HSUPA are supported – 0x06 - Dual-cell HSDPA+ is supported – 0x07 - Dual-cell HSDPA+ and HSUPA are supported – 0xFF - Not Available
<i>pscValid</i>	<ul style="list-style-type: none"> • Indicates whether primary scrambling code is valid. <ul style="list-style-type: none"> – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
<i>psc</i>	<ul style="list-style-type: none"> • Primary scrambling code. <ul style="list-style-type: none"> – 0xFFFF - Not Available

8.1127.2 Field Documentation

8.1127.2.1 **ULONG** WCDMASysInfo::cellId

8.1127.2.2 **BYTE** WCDMASysInfo::cellIdValid

8.1127.2.3 **BYTE** WCDMASysInfo::hsCallStatus

8.1127.2.4 **BYTE** WCDMASysInfo::hsCallStatusValid

8.1127.2.5 **BYTE** WCDMASysInfo::hsInd

8.1127.2.6 **BYTE** WCDMASysInfo::hsIndValid

8.1127.2.7 **WORD** WCDMASysInfo::lac

8.1127.2.8 **BYTE** WCDMASysInfo::lacValid

8.1127.2.9 **BYTE** WCDMASysInfo::MCC[3]

8.1127.2.10 **BYTE** WCDMASysInfo::MNC[3]

8.1127.2.11 **BYTE** WCDMASysInfo::networkIdValid

8.1127.2.12 **WORD** WCDMASysInfo::psc

8.1127.2.13 **BYTE** WCDMASysInfo::pscValid

8.1127.2.14 **BYTE** WCDMASysInfo::regRejectInfoValid

8.1127.2.15 **BYTE** WCDMASysInfo::rejCause

8.1127.2.16 **BYTE** WCDMASysInfo::rejectSrvDomain

8.1127.2.17 **sysInfoCommon** WCDMASysInfo::sysInfoWCDMA

8.1128 wcdmaUARFCN Struct Reference

Data Fields

- [BYTE](#) *status*
- [ULONG](#) *uarfcn*

8.1128.1 Detailed Description

This structure contains the parameters for WCDMA UARFCN.

Parameters

<i>status</i>	<ul style="list-style-type: none">• 0 - Disable• 1 - Enable
<i>uarfcn</i>	<ul style="list-style-type: none">• UARFCN to which UE is locked

8.1128.2 Field Documentation

8.1128.2.1 **BYTE** wcdmaUARFCN::status

8.1128.2.2 **ULONG** wcdmaUARFCN::uarfcn

8.1129 wds_currNetworkInfo Struct Reference

Data Fields

- [uint8_t](#) *NetworkType*
- [uint32_t](#) *RATMask*
- [uint32_t](#) *SOMask*

8.1129.1 Detailed Description

Network information structure

Parameters

<i>NetworkType</i>	<ul style="list-style-type: none">• Values:<ul style="list-style-type: none">– 0 - 3GPP– 1 - 3GPP2
<i>RATMask</i>	<ul style="list-style-type: none">• Radio Access Technology (RAT) mask to indicate the type of technology.• Values:<ul style="list-style-type: none">– 0 - Don't Care– 0x8000 - NULL Bearer• CDMA RAT mask values:<ul style="list-style-type: none">– 0x01 - CDMA_1x– 0x02 - EVDO_REV0– 0x04 - EVDO_REVA– 0x08 - EVDO_REVB– 0x10 - EHRPD– 0x20 - FMC• UMTS RAT mask values:<ul style="list-style-type: none">– 0x01 - WCDMA– 0x02 - GPRS– 0x04 - HSDPA– 0x08 - HSUPA– 0x10 - EDGE– 0x20 - LTE– 0x40 - HSDPA+– 0x80 - DC_HSDPA+– 0x100 - 64_QAM– 0x200 - TDSCDMA

<i>SOMask</i>	<ul style="list-style-type: none"> • Service Option (SO) mask to indicate the service option or type of application. • Values: <ul style="list-style-type: none"> – 0 - Don't Care • CDMA 1x SO mask values: <ul style="list-style-type: none"> – 0x01 - CDMA_1X_IS95 – 0x02 - CDMA_1X_IS2000 – 0x04 - CDMA_1X_IS2000_REL_A • CDMA EV-DO Rev 0 SO mask values: <ul style="list-style-type: none"> – 0x01 - DPA • CDMA EV-DO Rev A SO mask values: <ul style="list-style-type: none"> – 0x01 - DPA – 0x02 - MFPA – 0x04 - EMPA – 0x08 - EMPA_EHRPD • CDMA EV-DO Rev B SO mask values: <ul style="list-style-type: none"> – 0x01 - DPA – 0x02 - MFPA – 0x04 - EMPA – 0x08 - EMPA_EHRPD – 0x10 - MMPA – 0x20 - MMPA_EHRPD
---------------	--

8.1129.2 Field Documentation

8.1129.2.1 `uint8_t wds_currNetworkInfo::NetworkType`

8.1129.2.2 `uint32_t wds_currNetworkInfo::RATMask`

8.1129.2.3 `uint32_t wds_currNetworkInfo::SOMask`

8.1130 wds_DataULongLongTlv Struct Reference

Data Fields

- `uint8_t TlvPresent`
- `uint64_t ulldata`

8.1130.1 Field Documentation

8.1130.1.1 `uint8_t wds_DataULongLongTlv::TlvPresent`

8.1130.1.2 `uint64_t wds_DataULongLongTlv::ulldata`

8.1131 wds_DataULongTlv Struct Reference

Data Fields

- `uint8_t TlvPresent`
- `uint32_t ulldata`

8.1131.1 Field Documentation

8.1131.1.1 `uint8_t wds_DataULongTlv::TlvPresent`

8.1131.1.2 `uint32_t wds_DataULongTlv::ulData`

8.1132 wds_DHCPLeaseOptTlv Struct Reference

Data Fields

- `uint8_t TlvPresent`
- `uint8_t numOpt`
- `wds_DHCPOpt optList [30]`
- `uint8_t optListData [2048]`

8.1132.1 Field Documentation

8.1132.1.1 `uint8_t wds_DHCPLeaseOptTlv::numOpt`

8.1132.1.2 `wds_DHCPOpt wds_DHCPLeaseOptTlv::optList[30]`

8.1132.1.3 `uint8_t wds_DHCPLeaseOptTlv::optListData[2048]`

8.1132.1.4 `uint8_t wds_DHCPLeaseOptTlv::TlvPresent`

8.1133 wds_DHCPLeaseStateTlv Struct Reference

Data Fields

- `uint8_t TlvPresent`
- `uint8_t leaseState`

8.1133.1 Field Documentation

8.1133.1.1 `uint8_t wds_DHCPLeaseStateTlv::leaseState`

8.1133.1.2 `uint8_t wds_DHCPLeaseStateTlv::TlvPresent`

8.1134 wds_DHCPOpt Struct Reference

Data Fields

- `uint8_t optCode`
- `uint8_t optValLen`
- `uint8_t * pOptVal`

8.1134.1 Field Documentation

8.1134.1.1 `uint8_t wds_DHCPOpt::optCode`

8.1134.1.2 `uint8_t wds_DHCPOpt::optValLen`

8.1134.1.3 `uint8_t* wds_DHCPOpt::pOptVal`

8.1135 wds_DHCPProfileIdTlv Struct Reference

Data Fields

- `uint8_t TlvPresent`
- `uint8_t profileType`
- `uint8_t profileId`

8.1135.1 Field Documentation

8.1135.1.1 `uint8_t wds_DHCPProfileIdTlv::profileId`

8.1135.1.2 `uint8_t wds_DHCPProfileIdTlv::profileType`

8.1135.1.3 `uint8_t wds_DHCPProfileIdTlv::TlvPresent`

8.1136 wds_DHCPv4HWConfig Struct Reference

Data Fields

- `uint8_t hwType`
- `uint8_t chaddrLen`
- `uint8_t chaddr [16]`

8.1136.1 Detailed Description

WDS SWI DHCPv4 HW Config Structure.

Parameters

<i>hwType</i>	<ul style="list-style-type: none"> • HW Type 1 - Ethernet 20 - Serial
<i>chaddrLen</i>	<ul style="list-style-type: none"> • chaddrLen
<i>chaddr</i>	<ul style="list-style-type: none"> • chaddr. Max size 16 bytes

8.1136.2 Field Documentation

8.1136.2.1 `uint8_t wds_DHCPv4HWConfig::chaddr[16]`

8.1136.2.2 `uint8_t wds_DHCPv4HWConfig::chaddrLen`

8.1136.2.3 `uint8_t wds_DHCPv4HWConfig::hwType`

8.1137 wds_DHCPv4Option Struct Reference

Data Fields

- `uint8_t optCode`
- `uint8_t optValLen`
- `uint8_t optVal [255]`

8.1137.1 Detailed Description

WDS SWI DHCPv4 Option Structure

Parameters

<i>optCode</i>	<ul style="list-style-type: none">• Option code<ul style="list-style-type: none">– 0 - 255
<i>optValLen</i>	<ul style="list-style-type: none">• Option value length<ul style="list-style-type: none">– 0 - 255
<i>optVal</i>	<ul style="list-style-type: none">• Option value

8.1137.2 Field Documentation

8.1137.2.1 `uint8_t wds_DHCPv4Option::optCode`

8.1137.2.2 `uint8_t wds_DHCPv4Option::optVal[255]`

8.1137.2.3 `uint8_t wds_DHCPv4Option::optValLen`

8.1138 wds_DHCPv4OptionList Struct Reference

Data Fields

- `uint8_t numOpt`
- `wds_DHCPv4Option * pOptList`

8.1138.1 Detailed Description

WDS SWI DHCPv4 Option List Structure

Parameters

<i>numOpt</i>	<ul style="list-style-type: none"> number of options <ul style="list-style-type: none"> – 0 - 255
<i>pOptList</i>	<ul style="list-style-type: none"> pointer to list of DHCP Options

8.1138.2 Field Documentation

8.1138.2.1 `uint8_t wds_DHCPv4OptionList::numOpt`

8.1138.2.2 `wds_DHCPv4Option* wds_DHCPv4OptionList::pOptList`

8.1139 wds_DHCPv4ProfileId Struct Reference

Data Fields

- `uint8_t profileType`
- `uint8_t profileId`

8.1139.1 Detailed Description

WDS SWI DHCPv4 Profile Identifier Structure

Parameters

<i>profileType</i>	<ul style="list-style-type: none"> 0 for 3GPP
<i>profileId</i>	<ul style="list-style-type: none"> 1 to 24 for 3GPP profile

8.1139.2 Field Documentation

8.1139.2.1 `uint8_t wds_DHCPv4ProfileId::profileId`

8.1139.2.2 `uint8_t wds_DHCPv4ProfileId::profileType`

8.1140 wds_Domain Struct Reference

Data Fields

- uint16_t [domainLen](#)
- uint8_t [domainName](#) [256]

8.1140.1 Detailed Description

This structure contains the DomainName Information

Parameters

<i>domainLen</i>	<ul style="list-style-type: none">• length of the recieved Domain name
<i>domainName</i>	<ul style="list-style-type: none">• Domain name(Max 256 characters)

8.1140.2 Field Documentation

8.1140.2.1 uint16_t wds_Domain::domainLen

8.1140.2.2 uint8_t wds_Domain::domainName[256]

8.1141 wds_DomainNameList Struct Reference

Data Fields

- uint8_t [numInstances](#)
- struct [wds_Domain domain](#) [10]

8.1141.1 Detailed Description

This structure contains the [DomainNameList](#) Information

Parameters

<i>numInstances</i>	<ul style="list-style-type: none">• Number of Domain name received
<i>domain</i>	<ul style="list-style-type: none">• Domain name information(Max 10 Domain names)

8.1141.2 Field Documentation

8.1141.2.1 struct wds_Domain wds_DomainNameList::domain[10]

8.1141.2.2 uint8_t wds_DomainNameList::numInstances

8.1142 wds_GPRSQoS Struct Reference

Data Fields

- uint32_t [precedenceClass](#)
- uint32_t [delayClass](#)
- uint32_t [reliabilityClass](#)
- uint32_t [peakThroughputClass](#)
- uint32_t [meanThroughputClass](#)

8.1142.1 Detailed Description

This structure contains the GPRS Quality Of Service Information

Parameters

<i>precedence</i> ↔ <i>Class</i>	<ul style="list-style-type: none"> • Precedence class
<i>delayClass</i>	<ul style="list-style-type: none"> • Delay class
<i>reliabilityClass</i>	<ul style="list-style-type: none"> • Reliability class
<i>peak</i> ↔ <i>Throughput</i> ↔ <i>Class</i>	<ul style="list-style-type: none"> • Peak throughput class
<i>mean</i> ↔ <i>Throughput</i> ↔ <i>Class</i>	<ul style="list-style-type: none"> • Mean throughput class

8.1142.2 Field Documentation

8.1142.2.1 uint32_t wds_GPRSQoS::delayClass

8.1142.2.2 uint32_t wds_GPRSQoS::meanThroughputClass

8.1142.2.3 uint32_t wds_GPRSQoS::peakThroughputClass

8.1142.2.4 uint32_t wds_GPRSQoS::precedenceClass

8.1142.2.5 `uint32_t wds_GPRSQoS::reliabilityClass`

8.1143 wds_IPv4AdTlv Struct Reference

Data Fields

- `uint8_t TlvPresent`
- `uint32_t IPv4Addr`

8.1143.1 Field Documentation

8.1143.1.1 `uint32_t wds_IPv4AdTlv::IPv4Addr`

8.1143.1.2 `uint8_t wds_IPv4AdTlv::TlvPresent`

8.1144 wds_IPV6AddressInfo Struct Reference

Data Fields

- `uint8_t IPV6PrefixLen`
- `uint16_t IPAddressV6 [8]`

8.1144.1 Detailed Description

This structure contains the IPV6 Address Information

Parameters

<i>IPV6PrefixLen</i>	<ul style="list-style-type: none">• Length of the received IPv6 address in no. of bits; can take value between 0 and 128<ul style="list-style-type: none">– 0xFF - Not Available
<i>IPAddressV6</i>	<ul style="list-style-type: none">• IPv6 address(in network byte order); This is an 8-element array of 16 bit numbers, each of which is in big endian format.

8.1144.2 Field Documentation

8.1144.2.1 `uint16_t wds_IPV6AddressInfo::IPAddressV6[8]`

8.1144.2.2 `uint8_t wds_IPV6AddressInfo::IPV6PrefixLen`

8.1145 wds_IPV6GWAddressInfo Struct Reference

Data Fields

- uint8_t [gwV6PrefixLen](#)
- uint16_t [gwAddressV6](#) [8]

8.1145.1 Detailed Description

This structure contains the IPV6 Gateway Address Information

Parameters

<i>gwV6PrefixLen</i>	<ul style="list-style-type: none"> • Length of the received IPV6 Gateway address in no. of bits; can take value between 0 and 128
<i>IPAddressV6</i>	<ul style="list-style-type: none"> • IPV6 Gateway address(in network byte order); This is an 8-element array of 16 bit numbers, each of which is in big endian format.

8.1145.2 Field Documentation

8.1145.2.1 uint16_t wds_IPV6GWAddressInfo::gwAddressV6[8]

8.1145.2.2 uint8_t wds_IPV6GWAddressInfo::gwV6PrefixLen

8.1146 wds_PCSCFFQDNAddress Struct Reference

Data Fields

- uint16_t [fqdnLen](#)
- uint8_t [fqdnAddr](#) [256]

8.1146.1 Detailed Description

This structure contains the [PCSCFFQDNAddress](#) Information

Parameters

<i>fqdnLen</i>	<ul style="list-style-type: none"> • length of the received FQDN address
<i>fqdnAddr</i>	<ul style="list-style-type: none"> • FQDN address(Max 256 characters)

8.1146.2 Field Documentation

8.1146.2.1 `uint8_t wds_PCSCFFQDNAddress::fqdnAddr[256]`

8.1146.2.2 `uint16_t wds_PCSCFFQDNAddress::fqdnLen`

8.1147 wds_PCSCFFQDNAddressList Struct Reference

Data Fields

- `uint8_t numInstances`
- struct `wds_PCSCFFQDNAddress pcsfQDNAddress` [10]

8.1147.1 Detailed Description

This structure contains the [PCSCFFQDNAddressList](#) Information

Parameters

<i>numInstances</i>	<ul style="list-style-type: none">• Number of FQDN addresses received
<i>pcsfQDN↔ Address</i>	<ul style="list-style-type: none">• FQDN address information(Max 10 addresses)

8.1147.2 Field Documentation

8.1147.2.1 `uint8_t wds_PCSCFFQDNAddressList::numInstances`

8.1147.2.2 `struct wds_PCSCFFQDNAddress wds_PCSCFFQDNAddressList::pcsfQDNAddress[10]`

8.1148 wds_PCSCFIPv4ServerAddressList Struct Reference

Data Fields

- `uint8_t numInstances`
- `uint32_t pcsfIPv4Addr` [64]

8.1148.1 Detailed Description

This structure contains the [PCSCFIPv4ServerAddressList](#) Information

Parameters

<i>numInstances</i>	<ul style="list-style-type: none">• number of address following
<i>pcsfIPv4Addr</i> Generated by Doxygen	<ul style="list-style-type: none">• P-CSCF IPv4 server addresses(Max 16 address, 4 bytes each)

8.1148.2 Field Documentation

8.1148.2.1 `uint8_t wds_PCSCFIPv4ServerAddressList::numInstances`

8.1148.2.2 `uint32_t wds_PCSCFIPv4ServerAddressList::pcscfIPv4Addr[64]`

8.1149 wds_ProfileIdentifier Struct Reference

Data Fields

- `uint8_t` [profileType](#)
- `uint8_t` [profileIndex](#)

8.1149.1 Detailed Description

This structure contains the Profile Identifier Information

Parameters

<i>profileType</i>	<ul style="list-style-type: none"> Identifies the type of profile 0x00 = 3GPP
<i>profileIndex</i>	<ul style="list-style-type: none"> Index of profile whose settings were loaded prior to session parameter negotiation for the current call. If this TLV is not present, data call parameters are based on device default settings for each parameter

8.1149.2 Field Documentation

8.1149.2.1 `uint8_t wds_ProfileIdentifier::profileIndex`

8.1149.2.2 `uint8_t wds_ProfileIdentifier::profileType`

8.1150 wds_profileInfo Union Reference

Data Fields

- [LibPackprofile_3GPP](#) `SlqsProfile3GPP`
- [LibPackprofile_3GPP2](#) `SlqsProfile3GPP2`

8.1150.1 Detailed Description

This union consist of `profile_3GPP` and `profile_3GPP2` out of which one will be used to create profile.

8.1150.2 Field Documentation

8.1150.2.1 LibPackprofile_3GPP wds_profileInfo::SlqsProfile3GPP

8.1150.2.2 LibPackprofile_3GPP2 wds_profileInfo::SlqsProfile3GPP2

8.1151 wds_TrStatInd Struct Reference

Data Fields

- uint8_t statsPeriod
- uint32_t statsMask

8.1151.1 Detailed Description

This structure contains the information about the Transfer Statistics Indicator parameters.

Parameters

<i>statsPeriod</i>	<ul style="list-style-type: none">• Period between transfer statistics reports.<ul style="list-style-type: none">– 0 - Do not report– Other - Period between reports (seconds)
<i>statsMask</i>	<ul style="list-style-type: none">• Requested statistic bit mask.<ul style="list-style-type: none">– 0x00000001 - Tx packets OK– 0x00000002 - Rx packets OK– 0x00000004 - Tx packet errors– 0x00000008 - Rx packet errors– 0x00000010 - Tx overflows– 0x00000020 - Rx overflows– 0x00000040 - Tx bytes OK– 0x00000080 - Rx bytes OK• Each bit set causes the corresponding optional information to be sent in SLQSWdsEvent↵ ReportCallBack.• All unlisted bits are reserved for future use and must be set to zero.

8.1151.2 Field Documentation

8.1151.2.1 uint32_t wds_TrStatInd::statsMask

8.1151.2.2 uint8_t wds_TrStatInd::statsPeriod

8.1152 wds_UMTSMinQoS Struct Reference

Data Fields

- uint8_t [trafficClass](#)
- uint32_t [maxUplinkBitrate](#)
- uint32_t [maxDownlinkBitrate](#)
- uint32_t [grntUplinkBitrate](#)
- uint32_t [grntDownlinkBitrate](#)
- uint8_t [qosDeliveryOrder](#)
- uint32_t [maxSDUSize](#)
- uint8_t [sduErrorRatio](#)
- uint8_t [resBerRatio](#)
- uint8_t [deliveryErrSDU](#)
- uint32_t [transferDelay](#)
- uint32_t [trafficPriority](#)

8.1152.1 Detailed Description

This structure contains the UMTS Quality Of Service Information

Parameters

<i>trafficClass</i>	<ul style="list-style-type: none"> • 0x00 - Subscribed • 0x01 - Conversational • 0x02 - Streaming • 0x03 - Interactive • 0x04 - Background
<i>maxUplinkBitrate</i>	<ul style="list-style-type: none"> • Maximum uplink bit rate in bits/sec
<i>maxDownlink↔ Bitrate</i>	<ul style="list-style-type: none"> • Maximum downlink bit rate in bits/sec
<i>grntUplinkBitrate</i>	<ul style="list-style-type: none"> • Guaranteed uplink bit rate in bits/sec
<i>grntDownlink↔ Bitrate</i>	<ul style="list-style-type: none"> • Guaranteed downlink bit rate in bits/sec
<i>qosDelivery↔ Order</i>	<ul style="list-style-type: none"> - Qos delivery order • 0x00 - Subscribe • 0x01 - Delivery order on • 0x02 - Delivery order off
<i>maxSDUSize</i>	<ul style="list-style-type: none"> • Maximum SDU size

<i>sduErrorRatio</i>	<ul style="list-style-type: none"> - SDU error ratio • Target value for fraction of SDUs lost or detected as erroneous. • 0x00 - Subscribe • 0x01 - $1 \cdot 10^{-2}$ • 0x02 - $7 \cdot 10^{-3}$ • 0x03 - $1 \cdot 10^{-3}$ • 0x04 - $1 \cdot 10^{-4}$ • 0x05 - $1 \cdot 10^{-5}$ • 0x06 - $1 \cdot 10^{-6}$ • 0x07 - $1 \cdot 10^{-1}$
<i>resBerRatio</i>	<ul style="list-style-type: none"> - Residual bit error ratio • Target value for undetected bit error ratio in the delivered SDUs. • 0x00 - Subscribe • 0x01 - $5 \cdot 10^{-2}$ • 0x02 - $1 \cdot 10^{-2}$ • 0x03 - $5 \cdot 10^{-3}$ • 0x04 - $4 \cdot 10^{-3}$ • 0x05 - $1 \cdot 10^{-3}$ • 0x06 - $1 \cdot 10^{-4}$ • 0x07 - $1 \cdot 10^{-5}$ • 0x08 - $1 \cdot 10^{-6}$ • 0x09 - $1 \cdot 10^{-8}$
<i>deliveryErrSDU</i>	<ul style="list-style-type: none"> - delivery of erroneous SDUs • Indicates whether SDUs detected as erroneous shall be delivered or not. • 0x00 - Subscribe • 0x01 - $5 \cdot 10^{-2}$ • 0x02 - $1 \cdot 10^{-2}$ • 0x03 - $5 \cdot 10^{-3}$ • 0x04 - $4 \cdot 10^{-3}$ • 0x05 - $1 \cdot 10^{-3}$ • 0x06 - $1 \cdot 10^{-4}$ • 0x07 - $1 \cdot 10^{-5}$ • 0x08 - $1 \cdot 10^{-6}$ • 0x09 - $1 \cdot 10^{-8}$
<i>transferDelay</i>	<ul style="list-style-type: none"> - Transfer delay (ms) • Indicates the targeted time between a request to transfer an SDU at one SAP to its delivery at the other SAP in milliseconds.
<i>trafficPriority</i>	<ul style="list-style-type: none"> - Transfer handling priority • Specifies the relative importance for handling of SDUs that belong to the UMTS bearer, compared to the SDUs of other bearers.

8.1152.2 Field Documentation

8.1152.2.1 uint8_t wds_UMTSMinQoS::deliveryErrSDU

8.1152.2.2 `uint32_t wds_UMTSMInQoS::grntDownlinkBitrate`

8.1152.2.3 `uint32_t wds_UMTSMInQoS::grntUplinkBitrate`

8.1152.2.4 `uint32_t wds_UMTSMInQoS::maxDownlinkBitrate`

8.1152.2.5 `uint32_t wds_UMTSMInQoS::maxSDUSize`

8.1152.2.6 `uint32_t wds_UMTSMInQoS::maxUplinkBitrate`

8.1152.2.7 `uint8_t wds_UMTSMInQoS::qosDeliveryOrder`

8.1152.2.8 `uint8_t wds_UMTSMInQoS::resBerRatio`

8.1152.2.9 `uint8_t wds_UMTSMInQoS::sduErrorRatio`

8.1152.2.10 `uint8_t wds_UMTSMInQoS::trafficClass`

8.1152.2.11 `uint32_t wds_UMTSMInQoS::trafficPriority`

8.1152.2.12 `uint32_t wds_UMTSMInQoS::transferDelay`

8.1153 WdsByteTotals Struct Reference

Data Fields

- [ULONG](#) * `pV4sessionId`
- [ULONG](#) * `pV6sessionId`
- struct [WdsByteTotalsElmnts](#) `ByteTotalsElmntsV4`
- struct [WdsByteTotalsElmnts](#) `ByteTotalsElmntsV6`

8.1153.1 Detailed Description

WDS ByteTotals request data structure

Parameters

<i>pV4sessionId</i>	<ul style="list-style-type: none"> • The v4 session ID for which the byte totals are to be retrieved • provide a NULL pointer if not applicable
<i>pV6sessionId</i>	<ul style="list-style-type: none"> • The v6 session ID for which the byte totals are to be retrieved • provide a NULL pointer if not applicable
<i>ByteTotals</i> ↔ <i>ElmntsV4</i>	<ul style="list-style-type: none"> • data structure to be populated with the byte totals for V4 session
<i>ByteTotals</i> ↔ <i>ElmntsV6</i>	<ul style="list-style-type: none"> • data structure to be populated with the byte totals for V6 session

Note

At least one of pV4sessionId and pV6sessionId must point to a valid session ID.

8.1153.2 Field Documentation

8.1153.2.1 struct WdsByteTotalsElmnts WdsByteTotals::ByteTotalsElmntsV4

8.1153.2.2 struct WdsByteTotalsElmnts WdsByteTotals::ByteTotalsElmntsV6

8.1153.2.3 ULONG* WdsByteTotals::pV4sessionId

8.1153.2.4 ULONG* WdsByteTotals::pV6sessionId

8.1154 WdsByteTotalsElmnts Struct Reference

Data Fields

- [ULONGLONG](#) * [pTXTotalBytes](#)
- [ULONGLONG](#) * [pRXTotalBytes](#)

8.1154.1 Detailed Description

WDS Bytes Totals request data structure for individual session

Parameters

<i>pTXTotalBytes</i>	<ul style="list-style-type: none">• No of transmitted bytes without error.
<i>pRXTotalBytes</i>	<ul style="list-style-type: none">• No of received bytes without error.

8.1154.2 Field Documentation

8.1154.2.1 ULONGLONG* WdsByteTotalsElmnts::pRXTotalBytes

8.1154.2.2 ULONGLONG* WdsByteTotalsElmnts::pTXTotalBytes

8.1155 WdsClientLeaseChange Struct Reference

Data Fields

- [BYTE](#) * [pEnableNotification](#)

8.1155.1 Detailed Description

WDS SWI DHCPv4 Client Lease Change Structure

Parameters

<i>pEnable↔ Notification</i>	[IN] <ul style="list-style-type: none"> • Enable Notification or not
----------------------------------	--

8.1155.2 Field Documentation

8.1155.2.1 **BYTE*** WdsClientLeaseChange::pEnableNotification

8.1156 WdsConnectionRate Struct Reference

Data Fields

- [ULONG](#) * pV4sessionId
- [ULONG](#) * pV6sessionId
- struct [WdsConnectionRateElmnts ConnRateElmntsV4](#)
- struct [WdsConnectionRateElmnts ConnRateElmntsV6](#)

8.1156.1 Detailed Description

WDS ConnectionRate request data structure

Parameters

<i>pV4sessionId</i>	<ul style="list-style-type: none"> • The v4 session ID for which the connection rate are to be retrieved • provide a NULL pointer if not applicable
<i>pV6sessionId</i>	<ul style="list-style-type: none"> • The v6 session ID for which the connection rate are to be retrieved • provide a NULL pointer if not applicable
<i>ConnRate↔ ElmntsV4</i>	<ul style="list-style-type: none"> • data structure to be populated with the connection rate for V4 session
<i>ConnRate↔ ElmntsV6</i>	<ul style="list-style-type: none"> • data structure to be populated with the connection rate for V6 session

Note

At least one of pV4sessionId and pV6sessionId must point to a valid session ID.

8.1156.2 Field Documentation

8.1156.2.1 struct WdsConnectionRateElmnts WdsConnectionRate::ConnRateElmntsV4

8.1156.2.2 struct WdsConnectionRateElmnts WdsConnectionRate::ConnRateElmntsV6

8.1156.2.3 ULONG* WdsConnectionRate::pV4sessionId

8.1156.2.4 ULONG* WdsConnectionRate::pV6sessionId

8.1157 WdsConnectionRateElmnts Struct Reference

Data Fields

- ULONG * pCurrentChannelTXRate
- ULONG * pCurrentChannelRXRate
- ULONG * pMaxChannelTXRate
- ULONG * pMaxChannelRXRate

8.1157.1 Detailed Description

WDS Connection rates request data structure for individual session

Parameters

<i>pCurrent↔ ChannelTX↔ Rate[OUT]</i>	<ul style="list-style-type: none"> • Instantaneous channel Tx rate in bits per second.
<i>pCurrent↔ ChannelRX↔ Rate[OUT]</i>	<ul style="list-style-type: none"> • Instantaneous channel Rx rate in bits per second.
<i>pMaxChannel↔ TXRate[OUT]</i>	<ul style="list-style-type: none"> • Maximum Tx rate that can be assigned to the device by the • serving system in bits per second
<i>pMaxChannel↔ RXRate[OUT]</i>	<ul style="list-style-type: none"> • Maximum Rx rate that can be assigned to the device by the • serving system in bits per second

8.1157.2 Field Documentation

8.1157.2.1 ULONG* WdsConnectionRateElmnts::pCurrentChannelRXRate

8.1157.2.2 ULONG* WdsConnectionRateElmnts::pCurrentChannelTXRate

8.1157.2.3 ULONG* WdsConnectionRateElmnts::pMaxChannelRXRate

8.1157.2.4 **ULONG*** WdsConnectionRateElmnts::pMaxChannelTXRate

8.1158 WdsDHCPv4ClientLeaseInd Struct Reference

Data Fields

- [WdsDHCPv4ProfileId](#) * [pProfileId](#)
- [BYTE](#) * [pLeaseState](#)
- [ULONG](#) * [pIPv4Addr](#)
- [DHCPOptionList](#) * [pOptList](#)

8.1158.1 Detailed Description

This structure contains DHCPv4 client lease status

Parameters

<i>pProfileId</i>	<ul style="list-style-type: none"> • Profile Type and Id
<i>pLeaseState</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – 0 - active, newly acquired – 1 - active, renewed – 2 - active, renewing – 3 - active, rebinding – 4 - inactive, expired – 5 - inactive, renew refused – 6 - inactive, rebind refused – 7 - inactive, other
<i>pIPv4Addr</i>	<ul style="list-style-type: none"> • Values <ul style="list-style-type: none"> – IPv4 Address
<i>pOptList</i>	<ul style="list-style-type: none"> • Option list

8.1158.2 Field Documentation

8.1158.2.1 **ULONG*** WdsDHCPv4ClientLeaseInd::pIPv4Addr

8.1158.2.2 **BYTE*** WdsDHCPv4ClientLeaseInd::pLeaseState

8.1158.2.3 **DHCPOptionList*** WdsDHCPv4ClientLeaseInd::pOptList

8.1158.2.4 **WdsDHCPv4ProfileId*** WdsDHCPv4ClientLeaseInd::pProfileId

8.1159 WdsDHCPv4Config Struct Reference

Data Fields

- [WdsDHCPv4ProfileId](#) * [pProfileId](#)
- [WdsDHCPv4HWConfig](#) * [pHwConfig](#)
- [WdsDHCPv4OptionList](#) * [pRequestOptionList](#)

8.1159.1 Detailed Description

WDS SWI DHCPv4 Config Structure

Parameters

<i>pProfileId</i>	[IN] • pointer to Profile Id structure
<i>pHWConfig</i>	[IN/OUT] • pointer to HW Config structure
<i>pRequest↔ OptionList</i>	[IN/OUT] • pointer to Option List structure to be sent in DHCP request

8.1159.2 Field Documentation

8.1159.2.1 [WdsDHCPv4HWConfig](#)* [WdsDHCPv4Config::pHwConfig](#)

8.1159.2.2 [WdsDHCPv4ProfileId](#)* [WdsDHCPv4Config::pProfileId](#)

8.1159.2.3 [WdsDHCPv4OptionList](#)* [WdsDHCPv4Config::pRequestOptionList](#)

8.1160 WdsDHCPv4HWConfig Struct Reference

Data Fields

- [BYTE](#) [hwType](#)
- [BYTE](#) [chaddrLen](#)
- [BYTE](#) [chaddr](#) [16]

8.1160.1 Detailed Description

WDS SWI DHCPv4 HW Config Structure.

Parameters

<i>hwType</i>	<ul style="list-style-type: none"> • HW Type 1 - Ethernet 20 - Serial
<i>chaddrlen</i>	<ul style="list-style-type: none"> • chaddrlen
<i>chaddr</i>	<ul style="list-style-type: none"> • chaddr. Max size 16 bytes

8.1160.2 Field Documentation

8.1160.2.1 BYTE WdsDhcpv4HWConfig::chaddr[16]

8.1160.2.2 BYTE WdsDhcpv4HWConfig::chaddrLen

8.1160.2.3 BYTE WdsDhcpv4HWConfig::hwType

8.1161 wdsDhcpv4HwConfig Struct Reference

Data Fields

- uint8_t [hwType](#)
- uint8_t [chaddrLen](#)
- uint8_t [chaddr](#) [16]

8.1161.1 Detailed Description

Parameters

<i>hwType</i>	DHCP HW Type, examples: <ul style="list-style-type: none"> • 0 - Ethernet • 20 - Serial
<i>chaddrLen</i>	Length of chaddr field, examples: <ul style="list-style-type: none"> • 6 for Ethernet MAC address
<i>chaddr</i>	Client hardware address

8.1161.2 Field Documentation

8.1161.2.1 uint8_t wdsDhcpv4HwConfig::chaddr[16]

8.1161.2.2 uint8_t wdsDhcpv4HwConfig::chaddrLen

8.1161.2.3 `uint8_t wdsDhcpv4HwConfig::hwType`

8.1162 wdsDhcpv4Option Struct Reference

Data Fields

- `uint8_t optCode`
- `uint8_t optValLen`
- `uint8_t optVal [255]`

8.1162.1 Detailed Description

Parameters

<i>optCode</i>	Option code <ul style="list-style-type: none">• 0 - 255
<i>optValLen</i>	Option value length <ul style="list-style-type: none">• 0 - 255
<i>optVal</i>	Option Value

8.1162.2 Field Documentation

8.1162.2.1 `uint8_t wdsDhcpv4Option::optCode`

8.1162.2.2 `uint8_t wdsDhcpv4Option::optVal[255]`

8.1162.2.3 `uint8_t wdsDhcpv4Option::optValLen`

8.1163 WdsDHCPv4Option Struct Reference

Data Fields

- `BYTE optCode`
- `BYTE optValLen`
- `BYTE optVal [255]`

8.1163.1 Detailed Description

WDS SWI DHCPv4 Option Structure

Parameters

<i>optCode</i>	<ul style="list-style-type: none"> Option code <ul style="list-style-type: none"> – 0 - 255
<i>optValLen</i>	<ul style="list-style-type: none"> Option value length <ul style="list-style-type: none"> – 0 - 255
<i>optVal</i>	<ul style="list-style-type: none"> Option value

8.1163.2 Field Documentation

8.1163.2.1 BYTE WdsDHCPv4Option::optCode

8.1163.2.2 BYTE WdsDHCPv4Option::optVal[255]

8.1163.2.3 BYTE WdsDHCPv4Option::optValLen

8.1164 WdsDHCPv4OptionList Struct Reference

Data Fields

- [BYTE numOpt](#)
- [WdsDHCPv4Option * pOptList](#)

8.1164.1 Detailed Description

WDS SWI DHCPv4 Option List Structure

Parameters

<i>numOpt</i>	<ul style="list-style-type: none"> number of options <ul style="list-style-type: none"> – 0 - 255
<i>pOptList</i>	<ul style="list-style-type: none"> pointer to list of DHCP Options

8.1164.2 Field Documentation

8.1164.2.1 BYTE WdsDHCPv4OptionList::numOpt

8.1164.2.2 WdsDHCPv4Option* WdsDHCPv4OptionList::pOptList

8.1165 wdsDhcpv4OptionList Struct Reference

Data Fields

- [uint8_t numOpt](#)
- [wdsDhcpv4Option * pOptList](#)

8.1165.1 Detailed Description

Parameters

<i>numOpt</i>	number of options <ul style="list-style-type: none">• 0 - 255
<i>pOptList</i>	pointer to list of DHCP Options

8.1165.2 Field Documentation

8.1165.2.1 [uint8_t wdsDhcpv4OptionList::numOpt](#)

8.1165.2.2 [wdsDhcpv4Option* wdsDhcpv4OptionList::pOptList](#)

8.1166 WdsDHCPv4ProfileId Struct Reference

Data Fields

- [BYTE profileType](#)
- [BYTE profileId](#)

8.1166.1 Detailed Description

WDS SWI DHCPv4 Profile Identifier Structure

Parameters

<i>profileType</i>	<ul style="list-style-type: none">• 0 for 3GPP
<i>profileId</i>	<ul style="list-style-type: none">• 1 to 24 for 3GPP profile

8.1166.2 Field Documentation

8.1166.2.1 **BYTE** WdsDHCPv4ProfileId::profileId

8.1166.2.2 **BYTE** WdsDHCPv4ProfileId::profileType

8.1167 wdsDhcpv4ProfileId Struct Reference

Data Fields

- [uint8_t profileType](#)
- [uint8_t profileId](#)

8.1167.1 Detailed Description

Parameters

<i>profileType</i>	profile type <ul style="list-style-type: none">• 0 - 3GPP
<i>profileId</i>	profile index <ul style="list-style-type: none">• index identifying the profile 1-24 valid for 3GPP profile type (EM74xx and onwards)

8.1167.2 Field Documentation

8.1167.2.1 **uint8_t** wdsDhcpv4ProfileId::profileId

8.1167.2.2 **uint8_t** wdsDhcpv4ProfileId::profileType

8.1168 WDSGetLoopbackData Struct Reference

Data Fields

- [BYTE ByteLoopbackMode](#)
- [BYTE ByteLoopbackMultiplier](#)

8.1168.1 Detailed Description

This API to Queries Enable/disable Data Loopback Mode and set the value of loopback multiplier.

Parameters

<i>pReq</i>	[IN] <ul style="list-style-type: none">• See WDSSetLoopbackData for more information
-------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 2 seconds\n

WDS SWI Get Loopback Structure of Packet Data Connection Information.

Parameters

<i>ByteLoopback↔ Mode</i>	<ul style="list-style-type: none"> • Loopback Mode. <ul style="list-style-type: none"> – 0 - Disable – 1 - Enable
<i>ByteLoopback↔ Multiplier</i>	<ul style="list-style-type: none"> • Loopback multiplier. Number of downlink bytes to send for each uplink byte.

8.1168.2 Field Documentation

8.1168.2.1 BYTE WDSGetLoopbackData::ByteLoopbackMode

8.1168.2.2 BYTE WDSGetLoopbackData::ByteLoopbackMultiplier

8.1169 WdsIpAddressInfoReq Struct Reference

Data Fields

- [ULONG](#) * [pv4sessionId](#)
- [ULONG](#) * [pv6sessionId](#)
- [QmiWdsIpAddressInfo](#) ip

8.1169.1 Field Documentation

8.1169.1.1 [QmiWdsIpAddressInfo](#) WdsIpAddressInfoReq::ip

8.1169.1.2 [ULONG](#)* WdsIpAddressInfoReq::pv4sessionId

8.1169.1.3 [ULONG](#)* WdsIpAddressInfoReq::pv6sessionId

8.1170 WdsPktStatisticsElmnts Struct Reference

Data Fields

- [ULONG](#) * [pTXPacketSuccesses](#)

- [ULONG * pRXPacketSuccesses](#)
- [ULONG * pTXPacketErrors](#)
- [ULONG * pRXPacketErrors](#)
- [ULONG * pTXPacketOverflows](#)
- [ULONG * pRXPacketOverflows](#)
- [ULONGLONG * pTXOkBytesCount](#)
- [ULONGLONG * pRXOkBytesCount](#)
- [ULONGLONG * pTXOKBytesLastCall](#)
- [ULONGLONG * pRXOKBytesLastCall](#)
- [ULONG * pTXDroppedCount](#)
- [ULONG * pRXDroppedCount](#)

8.1170.1 Detailed Description

WDS Pkt Statistics request data structure for individual session

Parameters

pTXPacket ↔ <i>Successes</i>	<ul style="list-style-type: none"> • No of transmitted Packets without error.
pRXPacket ↔ <i>Successes</i>	<ul style="list-style-type: none"> • No of received Packets without error.
pTXPacketErrors	<ul style="list-style-type: none"> • Number of outgoing packets with framing errors.
pRXPacket ↔ <i>Errors</i>	<ul style="list-style-type: none"> • Number of incoming packets with framing errors.
pTXPacket ↔ <i>Overflows</i>	<ul style="list-style-type: none"> • Number of packets dropped because Tx buffer overflowed (out of memory).
pRXPacket ↔ <i>Overflows</i>	<ul style="list-style-type: none"> • Number of packets dropped because Rx buffer overflowed (out of memory).
pTXOkBytes ↔ <i>Count</i>	<ul style="list-style-type: none"> • No of bytes transmitted without error.
pRXOkBytes ↔ <i>Count</i>	<ul style="list-style-type: none"> • No of bytes received without error.
pTXOKBytes ↔ <i>LastCall</i>	<ul style="list-style-type: none"> • No of bytes transmitted without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support
pRXOKBytes ↔ <i>LastCall</i>	<ul style="list-style-type: none"> • Number of bytes received without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support

<i>pTXDroppedCount</i>	<ul style="list-style-type: none"> Number of outgoing packets dropped.
<i>pRXDroppedCount</i>	<ul style="list-style-type: none"> Number of incoming packets dropped.

8.1170.2 Field Documentation

8.1170.2.1 **ULONG*** WdsPktStatisticsElmnts::pRXDroppedCount

8.1170.2.2 **ULONGLONG*** WdsPktStatisticsElmnts::pRXOkBytesCount

8.1170.2.3 **ULONGLONG*** WdsPktStatisticsElmnts::pRXOKBytesLastCall

8.1170.2.4 **ULONG*** WdsPktStatisticsElmnts::pRXPacketErrors

8.1170.2.5 **ULONG*** WdsPktStatisticsElmnts::pRXPacketOverflows

8.1170.2.6 **ULONG*** WdsPktStatisticsElmnts::pRXPacketSuccesses

8.1170.2.7 **ULONG*** WdsPktStatisticsElmnts::pTXDroppedCount

8.1170.2.8 **ULONGLONG*** WdsPktStatisticsElmnts::pTXOkBytesCount

8.1170.2.9 **ULONGLONG*** WdsPktStatisticsElmnts::pTXOKBytesLastCall

8.1170.2.10 **ULONG*** WdsPktStatisticsElmnts::pTXPacketErrors

8.1170.2.11 **ULONG*** WdsPktStatisticsElmnts::pTXPacketOverflows

8.1170.2.12 **ULONG*** WdsPktStatisticsElmnts::pTXPacketSuccesses

8.1171 WdsPktStatisticsReq Struct Reference

Data Fields

- ULONG *** pStatMask

8.1171.1 Detailed Description

WDS PktStatistics request data structure

Parameters

<i>pStatMask</i>	<ul style="list-style-type: none"> Packet Statistics Mask 0x00000001 - Tx packets OK 0x00000002 - Rx packets OK 0x00000004 - Tx packet errors 0x00000008 - Rx packet errors 0x00000010 - Tx overflows 0x00000020 - Rx overflows 0x00000040 - Tx bytes OK 0x00000080 - Rx bytes OK
------------------	--

8.1171.2 Field Documentation

8.1171.2.1 ULONG* WdsPktStatisticsReq::pStatMask

8.1172 WdsPktStatisticsResp Struct Reference

Data Fields

- [ULONG * pV4sessionId](#)
- [ULONG * pV6sessionId](#)
- struct [WdsPktStatisticsElmnts PktStatElmntsV4](#)
- struct [WdsPktStatisticsElmnts PktStatElmntsV6](#)

8.1172.1 Detailed Description

WDS PktStatistics response data structure

Parameters

<i>pV4sessionId</i>	<ul style="list-style-type: none"> The v4 session ID for which the byte totals are to be retrieved provide a NULL pointer if not applicable
<i>pV6sessionId</i>	<ul style="list-style-type: none"> The v6 session ID for which the byte totals are to be retrieved provide a NULL pointer if not applicable
<i>PktStatElmntsV4</i>	<ul style="list-style-type: none"> data structure to be populated with the Pkt Statistics for V4 session
<i>PktStatElmntsV6</i>	<ul style="list-style-type: none"> data structure to be populated with the Pkt Statistics for V6 session

Note

At least one of pV4sessionId and pV6sessionId must point to a valid session ID.

8.1172.2 Field Documentation

8.1172.2.1 struct WdsPktStatisticsElmnts WdsPktStatisticsResp::PktStatElmntsV4

8.1172.2.2 struct WdsPktStatisticsElmnts WdsPktStatisticsResp::PktStatElmntsV6

8.1172.2.3 ULONG* WdsPktStatisticsResp::pV4sessionId

8.1172.2.4 ULONG* WdsPktStatisticsResp::pV6sessionId

8.1173 WdsProfileParam Union Reference

Data Fields

- struct [Profile3GPP](#) SlqsProfile3GPP
- struct [Profile3GPP2](#) SlqsProfile3GPP2

8.1173.1 Detailed Description

This union [WdsProfileParam](#) consist of [Profile3GPP](#) and [Profile3GPP2](#) out of which one will be used to create profile.

8.1173.2 Field Documentation

8.1173.2.1 struct [Profile3GPP](#) WdsProfileParam::SlqsProfile3GPP

8.1173.2.2 struct [Profile3GPP2](#) WdsProfileParam::SlqsProfile3GPP2

8.1174 WdsRunTimeSettings Struct Reference

Data Fields

- ULONG * [v4sessionId](#)
- ULONG * [v6sessionId](#)
- struct [qmiWdsRunTimeSettings](#) rts

8.1174.1 Detailed Description

WDS runtime settings request data structure

Parameters

<i>v4sessionId</i>	<ul style="list-style-type: none"> • The v4 session ID for which the runtime settings are to be retrieved • provide a NULL pointer if not applicable
<i>v6sessionId</i>	<ul style="list-style-type: none"> • The v6 session ID for which the runtime settings are to be retrieved • provide a NULL pointer if not applicable
Generated by Doxygen qmiWdsRunTimeSettings	<ul style="list-style-type: none"> • data structure to be populated with the runtime settings

Note

At least one of v4sessionId and v6sessionId must point to a valid session ID.

8.1174.2 Field Documentation

8.1174.2.1 struct qmiWdsRunTimeSettings WdsRunTimeSettings::rts

8.1174.2.2 ULONG* WdsRunTimeSettings::v4sessionId

8.1174.2.3 ULONG* WdsRunTimeSettings::v6sessionId

8.1175 wdsSetEventReportReq Struct Reference

Data Fields

- BYTE * pCurrChannelRateInd
- TrStatInd * pTransferStatInd
- BYTE * pDataBearerTechInd
- BYTE * pDormancyStatusInd
- BYTE * pMIPStatusInd
- BYTE * pCurrDataBearerTechInd
- BYTE * pDataCallStatusChangeInd
- BYTE * pCurrPrefDataSysInd
- BYTE * pEVDOPageMonPerChangeInd
- BYTE * pDataSystemStatusChangeInd

8.1175.1 Detailed Description

This structure contains the information about the Set Event Report Request parameters.

Parameters

<i>pCurrChannel↔ RateInd</i>	(optional) <ul style="list-style-type: none"> Current Channel Rate Indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report channel rate when it changes
<i>pTransferStatInd</i>	(optional) <ul style="list-style-type: none"> See TrStatInd for more information.
<i>pDataBearer↔ TechInd</i>	(optional) <ul style="list-style-type: none"> Data Bearer Technology Indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report radio interface used for data transfer when it changes
<i>pDormancy↔ StatusInd</i>	(optional) <ul style="list-style-type: none"> Dormancy Status indicator. <ul style="list-style-type: none"> 0 - Do not report 1 - Report traffic channel state of interface used for data connection

<i>pMIPStatusInd</i>	(optional) <ul style="list-style-type: none"> • MIP Status Indicator. <ul style="list-style-type: none"> – 0 - Do not report – 1 - Report MIP status
<i>pCurrData↔ BearerTechInd</i>	(optional) <ul style="list-style-type: none"> • Current Data Bearer Technology Indicator. <ul style="list-style-type: none"> – 0 - Do not report – 1 - Report current data bearer technology when it changes
<i>pDataCall↔ StatusChange↔ Ind</i>	(optional) <ul style="list-style-type: none"> • Data Call Status Change Indicator. <ul style="list-style-type: none"> – 0 - Do not report – 1 - Report data call status change when it changes
<i>pCurrPrefData↔ SysInd</i>	(optional) <ul style="list-style-type: none"> • Current Preferred Data System Indicator. <ul style="list-style-type: none"> – 0 - Do not report – 1 - Report preferred data system when it changes
<i>pEVDOPage↔ MonPer↔ ChangeInd</i>	(optional) <ul style="list-style-type: none"> • EV-DO Page Monitor Period Change Indicator. <ul style="list-style-type: none"> – 0 - Do not report – 1 - Report EV-DO page monitor period change event
<i>pDataSystem↔ StatusChange↔ Ind</i>	(optional) <ul style="list-style-type: none"> • Data System Status Change Indicator. <ul style="list-style-type: none"> – 0 - Do not report – 1 - Report data system status change event

Note

At least one parameter should be present.

8.1175.2 Field Documentation

8.1175.2.1 **BYTE*** wdsSetEventReportReq::pCurrChannelRateInd

8.1175.2.2 **BYTE*** wdsSetEventReportReq::pCurrDataBearerTechInd

8.1175.2.3 **BYTE*** wdsSetEventReportReq::pCurrPrefDataSysInd

8.1175.2.4 **BYTE*** wdsSetEventReportReq::pDataBearerTechInd

8.1175.2.5 **BYTE*** wdsSetEventReportReq::pDataCallStatusChangeInd

8.1175.2.6 **BYTE*** wdsSetEventReportReq::pDataSystemStatusChangeInd

8.1175.2.7 **BYTE*** wdsSetEventReportReq::pDormancyStatusInd

8.1175.2.8 **BYTE*** wdsSetEventReportReq::pEVDOPageMonPerChangeInd

8.1175.2.9 **BYTE*** wdsSetEventReportReq::pMIPStatusInd

8.1175.2.10 **TrStatInd*** wdsSetEventReportReq::pTransferStatInd

8.1176 WDSSetLoopbackData Struct Reference

Data Fields

- **BYTE *** [pLoopbackMode](#)
- **BYTE *** [pLoopbackMultiplier](#)

8.1176.1 Detailed Description

WDS SWI Set Loopback Structure of Set Loopback Information.

Parameters

<i>pLoopbackMode</i>	<ul style="list-style-type: none"> • Loopback Mode. <ul style="list-style-type: none"> – 0 - Disable – 1 - Enable
<i>pLoopback↔ Multiplier</i>	<ul style="list-style-type: none"> • Loopback multiplier. Number of downlink bytes to send for each uplink byte.

8.1176.2 Field Documentation

8.1176.2.1 **BYTE*** WDSSetLoopbackData::pLoopbackMode

8.1176.2.2 **BYTE*** WDSSetLoopbackData::pLoopbackMultiplier

8.1177 WDSSWICurrentChannelRates Struct Reference

Data Fields

- unsigned long [current_channel_tx_rate](#)
- unsigned long [current_channel_rx_rate](#)
- unsigned long [max_channel_tx_rate](#)
- unsigned long [max_channel_rx_rate](#)

8.1177.1 Detailed Description

WDS SWI Current Channel Rates Structure of Packet Data Connection Information.

Parameters

<i>current_↔ channel_tx_rate</i>	<ul style="list-style-type: none">• Current Channel Tx Rate.
<i>current_↔ channel_rx_rate</i>	<ul style="list-style-type: none">• Current Channel Rx Rate.
<i>max_channel_↔ tx_rate</i>	<ul style="list-style-type: none">• Max Channel Tx Rate.
<i>max_channel_↔ rx_rate</i>	<ul style="list-style-type: none">• Max Channel Rx Rate.

8.1177.2 Field Documentation

8.1177.2.1 unsigned long WDSSWICurrentChannelRates::current_channel_rx_rate

8.1177.2.2 unsigned long WDSSWICurrentChannelRates::current_channel_tx_rate

8.1177.2.3 unsigned long WDSSWICurrentChannelRates::max_channel_rx_rate

8.1177.2.4 unsigned long WDSSWICurrentChannelRates::max_channel_tx_rate

Chapter 9

File Documentation

9.1 apdoxypages.c File Reference

Contains the module declaration for the Doxygen output. Also contains the content of the main page and related pages.

Namespaces

- [Tables](#)

9.1.1 Detailed Description

Contains the module declaration for the Doxygen output. Also contains the content of the main page and related pages.

9.2 common.h File Reference

Data Structures

- struct [pack_qmi_t](#)
- struct [unpack_qmi_t](#)

Macros

- #define [SDU_HDR_LEN](#) (3)
- #define [MINREQBKLEN](#) (2048)
- #define [MSGID_AND_LEN](#) (4)
- #define [MSGID_DONT_CARE](#) (0xffff)
- #define [UNUSEDPARAM](#)(x) (void)x
- #define [DEAULT_LOC_TIMEOUT_IN_SEC](#) 2
- #define [SDK_VALIDATE_INPUT_PACK_PARAM](#)(pCtx, pBuf, pLen)
- #define [SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL_XID](#)(pCtx, pBuf, pLen)

Typedefs

- typedef void(* [logger](#)) (uint8_t lvl, const char *buff)

Enumerations

- enum [eLOG_LEVEL](#) {
[eLOG_INFO](#),
[eLOG_DEBUG](#),
[eLOG_WARN](#),
[eLOG_FATAL](#) }
- enum [eTimeout](#) {
[eTIMEOUT_2_S](#) = 2000,
[eTIMEOUT_5_S](#) = 5000,
[eTIMEOUT_8_S](#) = 8000,
[eTIMEOUT_10_S](#) = 10000,
[eTIMEOUT_20_S](#) = 20000,
[eTIMEOUT_30_S](#) = 30000,
[eTIMEOUT_60_S](#) = 60000,
[eTIMEOUT_300_S](#) = 300000,
[eTIMEOUT_DEFAULT](#) = [eTIMEOUT_8_S](#) }
- enum [eQMI_SVC](#) {
[eCTL](#),
[eWDS](#),
[eDMS](#),
[eNAS](#) =3,
[eQOS](#),
[eSMS](#) =5,
[eCAT](#) =0x0A,
[eUIM](#) =0x0B,
[eLOC](#) =0x10,
[eTMD](#) =0x18,
[eSWIOMA](#) =240,
[eSWILOC](#) =246 }
- enum [msgtype](#) {
[eREQ](#) =0,
[eRSP](#) =2,
[eIND](#) =4 }

Functions

- uint16_t [helper_get_xid](#) (uint8_t *qmi_resp)
- const char * [helper_get_resp_ctx](#) (uint8_t svc, uint8_t *pbuf, uint32_t len, [unpack_qmi_t](#) *pCtx)
- unsigned [unpack_result_code_only](#) (uint8_t *pMdmResp)
- int [helper_set_log_func](#) (logger func)
- void [libpack_log](#) (uint8_t lvl, const char *fmt,...)
- int [helper_set_log_lvl](#) (uint8_t lvl)
- void [fill_sdu_hdr](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf)
- void [fill_pack_ctx](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t svc, int timeout)
- char * [get_version](#) ()
- char * [libpack_GetVersion](#) ()

Variables

- [logger](#) glog
- uint8_t [gloglvl](#)

9.2.1 Macro Definition Documentation

9.2.1.1 `#define DEault_LOC_TIMEOUT_IN_SEC 2`

9.2.1.2 `#define MINREQBKLEN (2048)`

9.2.1.3 `#define MSGID_AND_LEN (4)`

9.2.1.4 `#define MSGID_DONT_CARE (0xffff)`

9.2.1.5 `#define SDK_VALIDATE_INPUT_PACK_PARAM(pCtx, pBuf, pLen)`

Value:

```
if ((pCtx == NULL) || (pBuf == NULL) || (pLen == NULL) ) \
{ \
    libpack_log(eLOG_DEBUG, "[ pack] %s parameter NULL\n", __func__); \
    return eQCWWAN_ERR_INVALID_ARG; \
}
```

9.2.1.6 `#define SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL_XID(pCtx, pBuf, pLen)`

Value:

```
if ((pCtx == NULL) || (pBuf == NULL) || (pLen == NULL) ) \
{ \
    libpack_log(eLOG_DEBUG, "[ pack] %s parameter NULL\n", __func__); \
    return eQCWWAN_ERR_INVALID_ARG; \
} \
if (pCtx->xid == 0) \
    return eQCWWAN_ERR_INVALID_XID; \
pBuf[0] = eREQ; \
pBuf[1] = pCtx->xid & 0xff; \
pBuf[2] = pCtx->xid >> 8;
```

9.2.1.7 `#define SDU_HDR_LEN (3)`

9.2.1.8 `#define UNUSEDPARAM(x)(void)x`

9.2.2 Typedef Documentation

9.2.2.1 `typedef void(* logger)(uint8_t lvl, const char *buff)`

9.2.3 Enumeration Type Documentation

9.2.3.1 `enum eLOG_LEVEL`

log levels

Enumerator

eLOG_INFO
eLOG_DEBUG
eLOG_WARN
eLOG_FATAL

9.2.3.2 enum eQMI_SVC

qmi service

Enumerator

eCTL
eWDS
eDMS
eNAS
eQOS
eSMS
eCAT
eUIM
eLOC
eTMD
eSWIOMA
eSWILOC

9.2.3.3 enum eTimeout

eTimeout

Enumerator

eTIMEOUT_2_S
eTIMEOUT_5_S
eTIMEOUT_8_S
eTIMEOUT_10_S
eTIMEOUT_20_S
eTIMEOUT_30_S
eTIMEOUT_60_S
eTIMEOUT_300_S
eTIMEOUT_DEFAULT

9.2.3.4 enum msgtype

qmi message type

Enumerator

eREQ
eRSP
eIND

9.2.4 Function Documentation

9.2.4.1 void fill_pack_ctx (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, uint8_t svc, int timeout)

9.2.4.2 void fill_sdu_hdr (pack_qmi_t * pCtx, uint8_t * pReqBuf)

9.2.4.3 char* get_version ()

Returns

version string

9.2.4.4 const char* helper_get_resp_ctx (uint8_t svc, uint8_t * pbuf, uint32_t len, unpack_qmi_t * pCtx)

extract msgid/xid/type from modem reply

Parameters

in	svc	qmi service
in	pbuf	qmi response/indication
in	len	response/indication length
out	pCtx	unpacked context

Returns

qmi message string

9.2.4.5 uint16_t helper_get_xid (uint8_t * qmi_resp)

9.2.4.6 int helper_set_log_func (logger func)

set log function

9.2.4.7 int helper_set_log_lvl (uint8_t lvl)

set log level

9.2.4.8 char* libpack_GetVersion ()

Returns

version string

9.2.4.9 void libpack_log (uint8_t *lvl*, const char * *fmt*, ...)

9.2.4.10 unsigned unpack_result_code_only (uint8_t * *pMdmResp*)

common handler for unpacking response with TLV type 0x02 only

9.2.5 Variable Documentation

9.2.5.1 logger glog

9.2.5.2 uint8_t gloglvl

9.3 dms.h File Reference

Data Structures

- struct [unpack_dms_GetModelID_t](#)
- struct [unpack_dms_GetIMSI_t](#)
- struct [unpack_dms_GetFirmwareInfo_t](#)
- struct [unpack_dms_GetPower_t](#)
- struct [unpack_dms_GetSerialNumbers_t](#)
- struct [unpack_dms_GetHardwareRevision_t](#)
- struct [unpack_dms_SLQSGetBandCapability_t](#)
- struct [unpack_dms_GetDeviceCapabilities_t](#)
- struct [unpack_dms_GetFirmwareRevisions_t](#)
- struct [unpack_dms_GetFirmwareRevision_t](#)
- struct [unpack_dms_GetDeviceSerialNumbers_t](#)
- struct [unpack_dms_GetPRLVersion_t](#)
- struct [unpack_dms_GetNetworkTime_t](#)
- struct [unpack_dms_GetVoiceNumber_t](#)
- struct [unpack_dms_GetDeviceHardwareRev_t](#)
- struct [unpack_dms_GetFSN_t](#)
- struct [unpack_dms_GetDeviceCap_t](#)
- struct [pack_dms_SetPower_t](#)
- struct [unpack_dms_SetPower_t](#)
- struct [unpack_dms_GetBandCapability_t](#)
- struct [unpack_dms_GetUSBComp_t](#)
- struct [pack_dms_SetUSBComp_t](#)
- struct [unpack_dms_SetUSBComp_t](#)
- struct [pack_dms_SetCustFeature_t](#)
- struct [unpack_dms_SetCustFeature_t](#)
- struct [unpack_dms_GetCustFeature_t](#)
- struct [unpack_dms_SetFirmwarePreference_t](#)
- struct [unpack_dms_GetCrashAction_t](#)
- struct [pack_dms_SetCrashAction_t](#)
- struct [unpack_dms_SetCrashAction_t](#)
- struct [unpack_dms_GetDeviceMfr_t](#)
- struct [pack_dms_SetEventReport_t](#)
- struct [unpack_dms_SetEventReport_t](#)
- struct [dms_OperatingModeTlv](#)

- struct [dms_ActivationStatusTlv](#)
- struct [unpack_dms_SetEventReport_ind_t](#)
- struct [pack_dms_UIMGetICCID_t](#)
- struct [unpack_dms_UIMGetICCID_t](#)
- struct [pack_dms_SetCustFeaturesV2_t](#)
- struct [unpack_dms_SetCustFeaturesV2_t](#)
- struct [pack_dms_GetCustFeaturesV2_t](#)
- struct [DMSgetCustomInput](#)
- struct [DMScustSettingInfo](#)
- struct [DMScustSettingList](#)
- struct [DMSgetCustomFeatureV2](#)
- struct [unpack_dms_GetCustFeaturesV2_t](#)
- struct [unpack_dms_GetActivationState_t](#)
- struct [image_info_t](#)
- struct [unpack_dms_SLQSSwiGetFirmwareCurr_t](#)
- struct [pack_dms_SLQSSwiSetDyingGaspCfg_t](#)
- struct [unpack_dms_SLQSSwiSetDyingGaspCfg_t](#)
- struct [unpack_dms_SLQSSwiClearDyingGaspStatistics_t](#)
- struct [packgetDyingGaspStatistics](#)
- struct [unpack_dms_SLQSSwiGetDyingGaspStatistics_t](#)
- struct [packgetDyingGaspCfg](#)
- struct [unpack_dms_SLQSSwiGetDyingGaspCfg_t](#)
- struct [unpack_dms_SLQSDmsSwiGetResetInfo_t](#)
- struct [unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t](#)
- struct [pack_dms_SLQSDmsSwiIndicationRegister_t](#)
- struct [unpack_dms_SLQSDmsSwiIndicationRegister_t](#)
- struct [unpack_dms_SLQSSwiGetFwUpdateStatus_t](#)
- struct [unpack_dms_GetManufacturer_t](#)
- struct [unpack_dms_GetOfflineReason_t](#)
- struct [pack_dms_SetActivationStatusCallback_t](#)
- struct [unpack_dms_SetActivationStatusCallback_t](#)
- struct [pack_dms_UIMSetPINProtection_t](#)
- struct [unpack_dms_UIMSetPINProtection_t](#)
- struct [pack_dms_UIMUnblockPIN_t](#)
- struct [pack_dms_UIMVerifyPIN_t](#)
- struct [pack_dms_UIMChangePIN_t](#)
- struct [pack_dms_UIMGetControlKeyStatus_t](#)
- struct [unpack_dms_UIMGetControlKeyStatus_t](#)
- struct [unpack_dms_UIMGetPINStatus_t](#)
- struct [pack_dms_UIMSetControlKeyProtection_t](#)
- struct [unpack_dms_UIMSetControlKeyProtection_t](#)
- struct [pack_dms_UIMUnblockControlKey_t](#)
- struct [unpack_dms_UIMUnblockControlKey_t](#)
- struct [pack_dms_ResetToFactoryDefaults_t](#)
- struct [unpack_dms_ResetToFactoryDefaults_t](#)
- struct [pack_dms_ActivateAutomatic_t](#)
- struct [eriDataparams](#)
- struct [unpack_dms_SLQSGetERIFile_t](#)
- struct [unpack_dms_SLQSUIMGetState_t](#)
- struct [pack_dms_SLQSSwiGetCrashInfo_t](#)
- struct [crashInformation](#)
- struct [crashInfoParams](#)
- struct [unpack_dms_SLQSSwiGetCrashInfo_t](#)
- struct [unpack_dms_SLQSSwiGetHostDevInfo_t](#)
- struct [pack_dms_SLQSSwiSetHostDevInfo_t](#)

- struct [unpack_dms_SLQSSwiSetHostDevInfo_t](#)
- struct [unpack_dms_SLQSSwiGetOSInfo_t](#)
- struct [pack_dms_SLQSSwiSetOSInfo_t](#)
- struct [unpack_dms_SLQSSwiSetOSInfo_t](#)
- struct [unpack_dms_SLQSSwiGetSerialNoExt_t](#)

Macros

- #define [DMS_UINT8_MAX_STRING_SZ](#) 255
- #define [DMS_MAX_CUST_ID_LEN](#) 64
- #define [DMS_MAX_CUST_VALUE_LEN](#) 8
- #define [DMS_IMGDETAILS_LEN](#) 16
- #define [DMS_MAX_FWUPDATE_LOG_STR_SZ](#) 255
- #define [DMS_MAX_FWUPDATE_REF_STR_SZ](#) 15
- #define [DMS_SLQSFWINFO_MODELID_SZ](#) 20
- #define [DMS_SLQSFWINFO_BOOTVERSION_SZ](#) 85
- #define [DMS_SLQSFWINFO_APPVERSION_SZ](#) 85
- #define [DMS_SLQSFWINFO_SKU_SZ](#) 15
- #define [DMS_SLQSFWINFO_PACKAGEID_SZ](#) 85
- #define [DMS_SLQSFWINFO_CARRIER_SZ](#) 20
- #define [DMS_SLQSFWINFO_PRIVERSION_SZ](#) 16
- #define [DMS_SLQSFWINFO_CUR_CARR_NAME](#) 17
- #define [DMS_SLQSFWINFO_CUR_CARR_REV](#) 13
- #define [MAX_BUILD_ID_LEN](#) 255
- #define [UNIQUE_ID_LEN](#) 16
- #define [SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH](#) 160
- #define [SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH](#) 20
- #define [SPC_SIZE](#) 6
- #define [CK_MAX_SIZE](#) 8
- #define [ACT_CODE_MAX_SIZE](#) 81
- #define [ERI_DATA_MAX_SIZE](#) 1024
- #define [MEID_MAX_SIZE](#) 8
- #define [DMS_PM_ONLINE](#) 0x00 /* Online */
- #define [DMS_PM_LOW](#) 0x01 /* Low Power */
- #define [DMS_PM_FACTORY](#) 0x02 /* Factory Test Mode */
- #define [DMS_PM_OFFLINE](#) 0x03 /* Offline */
- #define [DMS_PM_RESET](#) 0x04 /* Reset */
- #define [DMS_PM_SHUT_DOWN](#) 0x05 /* Shut Down */
- #define [DMS_PM_PERSISTENT_LOW](#) 0x06 /* Persistent Low Power */
- #define [DMS_SET_REPORT_ENABLE](#) 1
- #define [DMS_SET_REPORT_DISABLE](#) 0
- #define [DMS_SWI_SET_IND_ENABLE](#) 1
- #define [DMS_SWI_SET_IND_DISABLE](#) 0

Functions

- int [pack_dms_GetIMSI](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetIMSI](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetIMSI_t](#) *pOutput)
- int [pack_dms_GetModelID](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetModelID](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetModelID_t](#) *pOutput)
- int [pack_dms_GetFirmwareInfo](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetFirmwareInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetFirmwareInfo_t](#) *pOutput)
- int [pack_dms_GetPower](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetPower](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetPower_t](#) *pOutput)
- int [pack_dms_GetSerialNumbers](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetSerialNumbers](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetSerialNumbers_t](#) *pOutput)
- int [pack_dms_GetHardwareRevision](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetHardwareRevision](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetHardwareRevision_t](#) *pOutput)
- int [pack_dms_SLQSGetBandCapability](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_SLQSGetBandCapability](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSGetBandCapability_t](#) *pOutput)
- int [pack_dms_GetDeviceCapabilities](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetDeviceCapabilities](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetDeviceCapabilities_t](#) *pOutput)
- int [pack_dms_GetFirmwareRevisions](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetFirmwareRevisions](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetFirmwareRevisions_t](#) *pOutput)
- int [pack_dms_GetFirmwareRevision](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetFirmwareRevision](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetFirmwareRevision_t](#) *pOutput)
- int [pack_dms_GetDeviceSerialNumbers](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetDeviceSerialNumbers](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetDeviceSerialNumbers_t](#) *pOutput)
- int [pack_dms_GetPRLVersion](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetPRLVersion](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetPRLVersion_t](#) *pOutput)
- int [pack_dms_GetNetworkTime](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetNetworkTime](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetNetworkTime_t](#) *pOutput)
- int [pack_dms_GetVoiceNumber](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetVoiceNumber](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetVoiceNumber_t](#) *pOutput)
- int [pack_dms_GetDeviceHardwareRev](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetDeviceHardwareRev](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetDeviceHardwareRev_t](#) *pOutput)
- int [pack_dms_GetFSN](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetFSN](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetFSN_t](#) *pOutput)
- int [pack_dms_GetDeviceCap](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetDeviceCap](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetDeviceCap_t](#) *pOutput)
- int [pack_dms_SetPower](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SetPower_t](#) *reqArg)
- int [unpack_dms_SetPower](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SetPower_t](#) *pOutput)
- int [pack_dms_GetBandCapability](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetBandCapability](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetBandCapability_t](#) *pOutput)
- int [pack_dms_GetUSBComp](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetUSBComp](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetUSBComp_t](#) *pOutput)

- int [pack_dms_SetUSBComp](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SetUSBComp_t](#) *reqArg)
- int [unpack_dms_SetUSBComp](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SetUSBComp_t](#) *pOutput)
- int [pack_dms_SetCustFeature](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SetCustFeature_t](#) *reqArg)
- int [unpack_dms_SetCustFeature](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SetCustFeature_t](#) *pOutput)
- int [pack_dms_GetCustFeature](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetCustFeature](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetCustFeature_t](#) *pOutput)
- int [pack_dms_SetFirmwarePreference](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_SetFirmwarePreference](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SetFirmwarePreference_t](#) *pOutput)
- int [pack_dms_GetCrashAction](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetCrashAction](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetCrashAction_t](#) *pOutput)
- int [pack_dms_SetCrashAction](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SetCrashAction_t](#) reqArg)
- int [unpack_dms_SetCrashAction](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SetCrashAction_t](#) *pOutput)
- int [pack_dms_GetDeviceMfr](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_dms_GetDeviceMfr](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetDeviceMfr_t](#) *pOutput)
- int [pack_dms_SetEventReport](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SetEventReport_t](#) *reqArg)
- int [unpack_dms_SetEventReport](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SetEventReport_t](#) *pOutput)
- int [unpack_dms_SetEventReport_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SetEventReport_ind_t](#) *pOutput)
- int [pack_dms_UIMGetICCID](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_UIMGetICCID_t](#) *reqArg)
- int [unpack_dms_UIMGetICCID](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_UIMGetICCID_t](#) *pOutput)
- int [pack_dms_SetCustFeaturesV2](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SetCustFeaturesV2_t](#) *reqArg)
- int [unpack_dms_SetCustFeaturesV2](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SetCustFeaturesV2_t](#) *pOutput)
- int [pack_dms_GetCustFeaturesV2](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_GetCustFeaturesV2_t](#) *reqArg)
- int [unpack_dms_GetCustFeaturesV2](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetCustFeaturesV2_t](#) *pOutput)
- int [pack_dms_GetActivationState](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_GetActivationState](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetActivationState_t](#) *pOutput)
- int [pack_dms_SLQSSwiGetFirmwareCurr](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_SLQSSwiGetFirmwareCurr](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiGetFirmwareCurr_t](#) *pOutput)
- int [pack_dms_SLQSSwiSetDyingGaspCfg](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SLQSSwiSetDyingGaspCfg_t](#) *reqArg)
- int [unpack_dms_SLQSSwiSetDyingGaspCfg](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiSetDyingGaspCfg_t](#) *pOutput)
- int [pack_dms_SLQSSwiClearDyingGaspStatistics](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_SLQSSwiClearDyingGaspStatistics](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiClearDyingGaspStatistics_t](#) *pOutput)
- int [pack_dms_SLQSSwiGetDyingGaspStatistics](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_SLQSSwiGetDyingGaspStatistics](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiGetDyingGaspStatistics_t](#) *pOutput)
- int [pack_dms_SLQSSwiGetDyingGaspCfg](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)

- int [unpack_dms_SLQSSwiGetDyingGaspCfg](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiGetDyingGaspCfg_t](#) *pOutput)
- int [pack_dms_SLQSDmsSwiGetResetInfo](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_SLQSDmsSwiGetResetInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSDmsSwiGetResetInfo_t](#) *pOutput)
- int [unpack_dms_SLQSDmsSwiGetResetInfo_Ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t](#) *pOutput)
- int [pack_dms_SLQSDmsSwiIndicationRegister](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SLQSDmsSwiIndicationRegister_t](#) *reqArg)
- int [unpack_dms_SLQSDmsSwiIndicationRegister](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSDmsSwiIndicationRegister_t](#) *pOutput)
- int [pack_dms_SLQSSwiGetFwUpdateStatus](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_SLQSSwiGetFwUpdateStatus](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiGetFwUpdateStatus_t](#) *pOutput)
- int [pack_dms_GetManufacturer](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_GetManufacturer](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetManufacturer_t](#) *pOutput)
- int [pack_dms_GetOfflineReason](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_GetOfflineReason](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_GetOfflineReason_t](#) *pOutput)
- int [pack_dms_SetActivationStatusCallback](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SetActivationStatusCallback_t](#) *reqArg)
- int [unpack_dms_SetActivationStatusCallback](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SetActivationStatusCallback_t](#) *pOutput)
- int [pack_dms_UIMSetPINProtection](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_UIMSetPINProtection_t](#) *pReq)
- int [unpack_dms_UIMSetPINProtection](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_UIMSetPINProtection_t](#) *pOutput)
- int [pack_dms_UIMUnblockPIN](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_UIMUnblockPIN_t](#) *pReq)
- int [unpack_dms_UIMUnblockPIN](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_UIMSetPINProtection_t](#) *pOutput)
- int [pack_dms_UIMVerifyPIN](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_UIMVerifyPIN_t](#) *pReq)
- int [unpack_dms_UIMVerifyPIN](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_UIMSetPINProtection_t](#) *pOutput)
- int [pack_dms_UIMChangePIN](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_UIMChangePIN_t](#) *pReq)
- int [unpack_dms_UIMChangePIN](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_UIMSetPINProtection_t](#) *pOutput)
- int [pack_dms_UIMGetControlKeyStatus](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_UIMGetControlKeyStatus_t](#) *pReq)
- int [unpack_dms_UIMGetControlKeyStatus](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_UIMGetControlKeyStatus_t](#) *pOutput)
- int [pack_dms_UIMGetPINStatus](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_UIMGetPINStatus](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_UIMGetPINStatus_t](#) *pOutput)
- int [pack_dms_UIMSetControlKeyProtection](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_UIMSetControlKeyProtection_t](#) *pReq)
- int [unpack_dms_UIMSetControlKeyProtection](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_UIMSetControlKeyProtection_t](#) *pOutput)
- int [pack_dms_UIMUnblockControlKey](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_UIMUnblockControlKey_t](#) *pReq)
- int [unpack_dms_UIMUnblockControlKey](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_UIMUnblockControlKey_t](#) *pOutput)

- int [pack_dms_ResetToFactoryDefaults](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_ResetToFactoryDefaults_t](#) *pReq)
- int [unpack_dms_ResetToFactoryDefaults](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_ResetToFactoryDefaults_t](#) *pOutput)
- int [pack_dms_ValidateSPC](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_ResetToFactoryDefaults_t](#) *pReq)
- int [unpack_dms_ValidateSPC](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_ResetToFactoryDefaults_t](#) *pOutput)
- int [pack_dms_ActivateAutomatic](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_ActivateAutomatic_t](#) *pReq)
- int [unpack_dms_ActivateAutomatic](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_ResetToFactoryDefaults_t](#) *pOutput)
- int [pack_dms_SLQSSwiGetERIFile](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_SLQSSwiGetERIFile](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiGetERIFile_t](#) *pOutput)
- int [pack_dms_SLQSSwiGetState](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_SLQSSwiGetState](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiGetState_t](#) *pOutput)
- int [pack_dms_SLQSSwiGetCrashInfo](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SLQSSwiGetCrashInfo_t](#) *pReq)
- int [unpack_dms_SLQSSwiGetCrashInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiGetCrashInfo_t](#) *pOutput)
- int [pack_dms_SLQSSwiGetHostDevInfo](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_SLQSSwiGetHostDevInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiGetHostDevInfo_t](#) *pOutput)
- int [pack_dms_SLQSSwiSetHostDevInfo](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SLQSSwiSetHostDevInfo_t](#) *pReq)
- int [unpack_dms_SLQSSwiSetHostDevInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiSetHostDevInfo_t](#) *pOutput)
- int [pack_dms_SLQSSwiGetOSInfo](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_SLQSSwiGetOSInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiGetOSInfo_t](#) *pOutput)
- int [pack_dms_SLQSSwiSetOSInfo](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_dms_SLQSSwiSetOSInfo_t](#) *pReq)
- int [unpack_dms_SLQSSwiSetOSInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiSetOSInfo_t](#) *pOutput)
- int [pack_dms_SLQSSwiGetSerialNoExt](#) (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_dms_SLQSSwiGetSerialNoExt](#) (uint8_t *pResp, uint16_t respLen, [unpack_dms_SLQSSwiGetSerialNoExt_t](#) *pOutput)

9.3.1 Macro Definition Documentation

9.3.1.1 `#define ACT_CODE_MAX_SIZE 81`

9.3.1.2 `#define CK_MAX_SIZE 8`

9.3.1.3 `#define DMS_IMGDETAILS_LEN 16`

9.3.1.4 `#define DMS_MAX_CUST_ID_LEN 64`

9.3.1.5 `#define DMS_MAX_CUST_VALUE_LEN 8`

- 9.3.1.6 `#define DMS_MAX_FWUPDATE_LOG_STR_SZ 255`
- 9.3.1.7 `#define DMS_MAX_FWUPDATE_REF_STR_SZ 15`
- 9.3.1.8 `#define DMS_PM_FACTORY 0x02 /* Factory Test Mode */`
- 9.3.1.9 `#define DMS_PM_LOW 0x01 /* Low Power */`
- 9.3.1.10 `#define DMS_PM_OFFLINE 0x03 /* Offline */`
- 9.3.1.11 `#define DMS_PM_ONLINE 0x00 /* Online */`
- 9.3.1.12 `#define DMS_PM_PERSISTENT_LOW 0x06 /* Persistent Low Power */`
- 9.3.1.13 `#define DMS_PM_RESET 0x04 /* Reset */`
- 9.3.1.14 `#define DMS_PM_SHUT_DOWN 0x05 /* Shut Down */`
- 9.3.1.15 `#define DMS_SET_REPORT_DISABLE 0`
- 9.3.1.16 `#define DMS_SET_REPORT_ENABLE 1`
- 9.3.1.17 `#define DMS_SLQSFWINFO_APPVERSION_SZ 85`
- 9.3.1.18 `#define DMS_SLQSFWINFO_BOOTVERSION_SZ 85`
- 9.3.1.19 `#define DMS_SLQSFWINFO_CARRIER_SZ 20`
- 9.3.1.20 `#define DMS_SLQSFWINFO_CUR_CARR_NAME 17`
- 9.3.1.21 `#define DMS_SLQSFWINFO_CUR_CARR_REV 13`
- 9.3.1.22 `#define DMS_SLQSFWINFO_MODELID_SZ 20`
- 9.3.1.23 `#define DMS_SLQSFWINFO_PACKAGEID_SZ 85`
- 9.3.1.24 `#define DMS_SLQSFWINFO_PRIVERSION_SZ 16`
- 9.3.1.25 `#define DMS_SLQSFWINFO_SKU_SZ 15`
- 9.3.1.26 `#define DMS_SWI_SET_IND_DISABLE 0`
- 9.3.1.27 `#define DMS_SWI_SET_IND_ENABLE 1`
- 9.3.1.28 `#define DMS_UINT8_MAX_STRING_SZ 255`

9.3.1.29 `#define ERI_DATA_MAX_SIZE 1024`

9.3.1.30 `#define MAX_BUILD_ID_LEN 255`

9.3.1.31 `#define MEID_MAX_SIZE 8`

9.3.1.32 `#define SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH 160`

9.3.1.33 `#define SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH 20`

9.3.1.34 `#define SPC_SIZE 6`

9.3.1.35 `#define UNIQUE_ID_LEN 16`

9.3.2 Function Documentation

9.3.2.1 `int pack_dms_ActivateAutomatic (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_ActivateAutomatic_t * pReq)`

To set UIM Unblock Control Key pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.2 `int pack_dms_GetActivationState (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Get Activation State pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.3 int pack_dms_GetBandCapability (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)

Get Band Capability pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.4 int pack_dms_GetCrashAction (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)

Get Crash Action pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.5 `int pack_dms_GetCustFeature (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

Get Custom Feature pack. This API is deprecated for EM74xx/MC74xx, please use [pack_dms_GetCustFeaturesV2\(\)](#) instead for EM74xx/MC74xx.

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.6 `int pack_dms_GetCustFeaturesV2 (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_GetCustFeaturesV2_t * reqArg)`

9.3.2.7 `int pack_dms_GetDeviceCap (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

Get Device Capabilities pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.8 `int pack_dms_GetDeviceCapabilities (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

get device capability pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.9 int pack_dms_GetDeviceHardwareRev (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)

Get Hardware Revision pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.10 int pack_dms_GetDeviceMfr (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)

Get Manufacture pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.11 `int pack_dms_GetDeviceSerialNumbers (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

get Device Serial Number pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.12 `int pack_dms_GetFirmwareInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

get firmware info pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.13 `int pack_dms_GetFirmwareRevision (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

get Firmware Revision pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.14 int pack_dms_GetFirmwareRevisions (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

get Firmware Revisions pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.15 int pack_dms_GetFSN (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get FSN pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.16 `int pack_dms_GetHardwareRevision (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

get hardware revision pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.17 `int pack_dms_GetIMSI (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

get IMSI pack. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_↔
xx_xx and all EM74xx firmware versions. Please use [pack_uim_ReadTransparent\(\)](#)(EF ID: 3F00 7F20 6F07 for 2G
card and 3F00 7FFF 6F07 for 3G card) instead for new firmware versions and new modules.

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.18 `int pack_dms_GetManufacturer (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

To get device manufacturer information.pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.19 int pack_dms_GetModelID (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

get model id pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.20 int pack_dms_GetNetworkTime (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Network Time pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.21 `int pack_dms_GetOfflineReason (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

To get operating mode offline reason pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.22 `int pack_dms_GetPower (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

get power pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.23 `int pack_dms_GetPRLVersion (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

Get PRL Versions pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.24 int pack_dms_GetSerialNumbers (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

get serial numbers pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.25 int pack_dms_GetUSBComp (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get USB Comp pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.26 `int pack_dms_GetVoiceNumber (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

Get Voice Number pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.27 `int pack_dms_ResetToFactoryDefaults (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_ResetToFactoryDefaults_t * pReq)`

To set UIM Unblock Control Key pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.28 `int pack_dms_SetActivationStatusCallback (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SetActivationStatusCallback_t * reqArg)`

Set activation status pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>pespLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.29 int pack_dms_SetCrashAction (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SetCrashAction_t reqArg)

Set Crash Action pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>pespLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Set Crash Action unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>pespLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.30 `int pack_dms_SetCustFeature (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SetCustFeature_t * reqArg)`

Set Custom Feature pack. This API is deprecated for EM74xx/MC74xx, please use [pack_dms_SetCustFeaturesV2\(\)](#) instead for EM74xx/MC74xx.

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.3.2.31 `int pack_dms_SetCustFeaturesV2 (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SetCustFeaturesV2_t * reqArg)`

Set Cust Features pack.

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.3.2.32 `int pack_dms_SetEventReport (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SetEventReport_t * reqArg)`

Set Event Report pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>pespLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.33 int pack_dms_SetFirmwarePreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Set Firmware Preference pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>pespLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.34 int pack_dms_SetPower (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetPower_t * *reqArg*)

Set Power pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>pespLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.3.2.35 `int pack_dms_SetUSBComp (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SetUSBComp_t * reqArg)`

Set USB Comp pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.3.2.36 `int pack_dms_SLQSDmsSwiGetResetInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

To get reset info pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.3.2.37 `int pack_dms_SLQSDmsSwiIndicationRegister (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SLQSDmsSwiIndicationRegister_t * reqArg)`

Set the registration state for different indication pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

support EM/MC74xx onwards

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.38 int pack_dms_SLQSGetBandCapability (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

get band capability pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.39 int pack_dms_SLQSGetERIFile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To set UIM Get ERI file pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.40 `int pack_dms_SLQSSwiClearDyingGaspStatistics (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Clear Dying GASP Statistics pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.41 `int pack_dms_SLQSSwiGetCrashInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SLQSSwiGetCrashInfo_t * pReq)`

To set Get Crash Info pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.42 `int pack_dms_SLQSSwiGetDyingGaspCfg (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Get Dying GASP Config pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.43 `int pack_dms_SLQSSwiGetDyingGaspStatistics (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Get Dying GASP Statistics pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.44 `int pack_dms_SLQSSwiGetFirmwareCurr (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

get currently active image pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.45 `int pack_dms_SLQSSwiGetFwUpdateStatus (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

To get Firmware Update status pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.46 `int pack_dms_SLQSSwiGetHostDevInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

To set Host Dev Info pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.47 `int pack_dms_SLQSSwiGetOSInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

To set Host OS Info pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.48 int pack_dms_SLQSSwiGetSerialNoExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To set serial number extension pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.49 int pack_dms_SLQSSwiSetDyingGaspCfg (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SLQSSwiSetDyingGaspCfg_t * *reqArg*)

Set Dying GASP Config pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.50 `int pack_dms_SLQSSwiSetHostDevInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SLQSSwiSetHostDevInfo_t * pReq)`

To set Host Dev Info pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.51 `int pack_dms_SLQSSwiSetOSInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SLQSSwiSetOSInfo_t * pReq)`

To set Host OS Info pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.52 `int pack_dms_SLQSUIGetState (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

To set UIM Get State pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.53 int pack_dms_UIMChangePIN (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_dms_UIMChangePIN_t * *pReq*)

To set UIM change PIN pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.54 int pack_dms_UIMGetControlKeyStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_dms_UIMGetControlKeyStatus_t * *pReq*)

To set UIM Get Control Key Status pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.55 `int pack_dms_UIMGetICCID (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_UIMGetICCID_t * reqArg)`

Packs the UIMGetICCID response message to a user-provided response structure. This API is deprecated on MC73xx/EM73xx modules. Since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use [pack_uim_ReadTransparent\(\)](#)(EF ID: 3F00 2FE2) instead for new firmware versions and new modules.

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.56 `int pack_dms_UIMGetPINStatus (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

To set UIM Get PIN Status pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.57 `int pack_dms_UIMSetControlKeyProtection (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_UIMSetControlKeyProtection_t * pReq)`

To set UIM Set Control Key Protection pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.58 `int pack_dms_UIMSetPINProtection (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_UIMSetPINProtection_t * pReq)`

To set UIM PIN protection pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.59 `int pack_dms_UIMUnblockControlKey (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_UIMUnblockControlKey_t * pReq)`

To set UIM Unblock Control Key pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.60 `int pack_dms_UIMUnblockPIN (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_UIMUnblockPIN_t * pReq)`

To set UIM unblock PIN pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.61 `int pack_dms_UIMVerifyPIN (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_UIMVerifyPIN_t * pReq)`

To set UIM verify PIN pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.62 `int pack_dms_ValidateSPC (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_ResetToFactoryDefaults_t * pReq)`

To set UIM Unblock Control Key pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>pReq</i>	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.63 `int unpack_dms_ActivateAutomatic (uint8_t * pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t * pOutput)`

To get UIM Unblock Control Key unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.64 `int unpack_dms_GetActivationState (uint8_t * pResp, uint16_t respLen, unpack_dms_GetActivationState_t * pOutput)`

Get Activation State unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.65 `int unpack_dms_GetBandCapability (uint8_t * pResp, uint16_t respLen, unpack_dms_GetBandCapability_t * pOutput)`

Get Band Capabilities unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.66 `int unpack_dms_GetCrashAction (uint8_t * pResp, uint16_t respLen, unpack_dms_GetCrashAction_t * pOutput)`

Get Crash Action unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.67 `int unpack_dms_GetCustFeature (uint8_t * pResp, uint16_t respLen, unpack_dms_GetCustFeature_t * pOutput)`

Get Custom Feature unpack. This API is deprecated for EM74xx/MC74xx, please use [unpack_dms_GetCustFeaturesV2\(\)](#) instead for EM74xx/MC74xx.

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.3.2.68 `int unpack_dms_GetCustFeaturesV2 (uint8_t * pResp, uint16_t respLen, unpack_dms_GetCustFeaturesV2_t * pOutput)`

9.3.2.69 `int unpack_dms_GetDeviceCap (uint8_t * pResp, uint16_t respLen, unpack_dms_GetDeviceCap_t * pOutput)`

Get Device Capabilities unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.3.2.70 `int unpack_dms_GetDeviceCapabilities (uint8_t * pResp, uint16_t respLen, unpack_dms_GetDeviceCapabilities_t * pOutput)`

get device capability unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.71 int unpack_dms_GetDeviceHardwareRev (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceHardwareRev_t * *pOutput*)

Get Hardware Revision unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.72 int unpack_dms_GetDeviceMfr (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceMfr_t * *pOutput*)

Get Manufacture unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.73 `int unpack_dms_GetDeviceSerialNumbers (uint8_t * pResp, uint16_t respLen, unpack_dms_GetDeviceSerialNumbers_t * pOutput)`

get Device Serial Number unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.74 `int unpack_dms_GetFirmwareInfo (uint8_t * pResp, uint16_t respLen, unpack_dms_GetFirmwareInfo_t * pOutput)`

get firmware info unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.75 `int unpack_dms_GetFirmwareRevision (uint8_t * pResp, uint16_t respLen, unpack_dms_GetFirmwareRevision_t * pOutput)`

get Firmware Revision unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.76 int unpack_dms_GetFirmwareRevisions (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFirmwareRevisions_t * *pOutput*)

get Firmware Revisions unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.77 int unpack_dms_GetFSN (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFSN_t * *pOutput*)

Get FSN unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.78 `int unpack_dms_GetHardwareRevision (uint8_t * pResp, uint16_t respLen, unpack_dms_GetHardwareRevision_t * pOutput)`

get hardware revision unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.79 `int unpack_dms_GetIMSI (uint8_t * pResp, uint16_t respLen, unpack_dms_GetIMSI_t * pOutput)`

get IMSI unpack. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx↔_xx_xx and all EM74xx firmware versions. Please use [unpack_uim_ReadTransparent\(\)](#)(EF ID: 3F00 7F20 6F07 for 2G card and 3F00 7FFF 6F07 for 3G card) instead for new firmware versions and new modules.

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.80 `int unpack_dms_GetManufacturer (uint8_t * pResp, uint16_t respLen, unpack_dms_GetManufacturer_t * pOutput)`

To get device manufacturer information unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.81 int unpack_dms_GetModelID (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetModelID_t * *pOutput*)

get model id unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.82 int unpack_dms_GetNetworkTime (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetNetworkTime_t * *pOutput*)

Get Network Time unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.83 `int unpack_dms_GetOfflineReason (uint8_t * pResp, uint16_t respLen, unpack_dms_GetOfflineReason_t * pOutput)`

To get operating mode offline reason unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.84 `int unpack_dms_GetPower (uint8_t * pResp, uint16_t respLen, unpack_dms_GetPower_t * pOutput)`

get power unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.85 `int unpack_dms_GetPRLVersion (uint8_t * pResp, uint16_t respLen, unpack_dms_GetPRLVersion_t * pOutput)`

Get PRL Versions unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.86 int unpack_dms_GetSerialNumbers (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetSerialNumbers_t * *pOutput*)

get serial numbers unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.87 int unpack_dms_GetUSBComp (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetUSBComp_t * *pOutput*)

Get USB Comp unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.3.2.88 `int unpack_dms_GetVoiceNumber (uint8_t * pResp, uint16_t respLen, unpack_dms_GetVoiceNumber_t * pOutput)`

Get Voice Number unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.3.2.89 `int unpack_dms_ResetToFactoryDefaults (uint8_t * pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t * pOutput)`

To get UIM Unblock Control Key unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.3.2.90 `int unpack_dms_SetActivationStatusCallback (uint8_t * pResp, uint16_t respLen, unpack_dms_SetActivationStatusCallback_t * pOutput)`

Set Activation status unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.91 `int unpack_dms_SetCrashAction (uint8_t * pResp, uint16_t respLen, unpack_dms_SetCrashAction_t * pOutput)`

Set Crash Action unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response. Not used

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.92 `int unpack_dms_SetCustFeature (uint8_t * pResp, uint16_t respLen, unpack_dms_SetCustFeature_t * pOutput)`

Set Custom Feature unpack. This API is deprecated for EM74xx/MC74xx, please use [unpack_dms_SetCustFeaturesV2\(\)](#) instead for EM74xx/MC74xx.

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.93 `int unpack_dms_SetCustFeaturesV2 (uint8_t * pResp, uint16_t respLen, unpack_dms_SetCustFeaturesV2_t * pOutput)`

Set Cust features unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.94 `int unpack_dms_SetEventReport (uint8_t * pResp, uint16_t respLen, unpack_dms_SetEventReport_t * pOutput)`

Set Event Report unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.95 `int unpack_dms_SetEventReport_ind (uint8_t * pResp, uint16_t respLen, unpack_dms_SetEventReport_t * pOutput)`

Event Report Indication unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.96 int unpack_dms_SetFirmwarePreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetFirmwarePreference_t * *pOutput*)

Set Firmware Preference unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.97 int unpack_dms_SetPower (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetPower_t * *pOutput*)

Set Power unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.98 `int unpack_dms_SetUSBComp (uint8_t * pResp, uint16_t respLen, unpack_dms_SetUSBComp_t * pOutput)`

Set USB Comp unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.99 `int unpack_dms_SLQSDmsSwiGetResetInfo (uint8_t * pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiGetResetInfo_t * pOutput)`

To get reset info unpack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.100 `int unpack_dms_SLQSDmsSwiGetResetInfo_Ind (uint8_t * pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t * pOutput)`

DMS reset info Indication unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

support EM/MC74xx onwards

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.101 int unpack_dms_SLQSDmsSwilIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*,
unpack_dms_SLQSDmsSwilIndicationRegister_t * *pOutput*)

Set the registration state for different indication unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

support EM/MC74xx onwards

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.102 int unpack_dms_SLQSGetBandCapability (uint8_t * *pResp*, uint16_t *respLen*,
unpack_dms_SLQSGetBandCapability_t * *pOutput*)

get band capability unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.103 int unpack_dms_SLQSGetERIFile (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSGetERIFile_t * *pOutput*)

To get UIM Get ERI file unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.104 int unpack_dms_SLQSSwiClearDyingGaspStatistics (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiClearDyingGaspStatistics_t * *pOutput*)

Clear Dying GASP Statistics unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.105 `int unpack_dms_SLQSSwiGetCrashInfo (uint8_t * pResp, uint16_t respLen, unpack_dms_SLQSSwiGetCrashInfo_t * pOutput)`

To get Crash Info unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.106 `int unpack_dms_SLQSSwiGetDyingGaspCfg (uint8_t * pResp, uint16_t respLen, unpack_dms_SLQSSwiGetDyingGaspCfg_t * pOutput)`

Get Dying GASP Config unpack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.107 `int unpack_dms_SLQSSwiGetDyingGaspStatistics (uint8_t * pResp, uint16_t respLen, unpack_dms_SLQSSwiGetDyingGaspStatistics_t * pOutput)`

Get Dying GASP Statistics unpack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.108 int unpack_dms_SLQSSwiGetFirmwareCurr (uint8_t * *pResp*, uint16_t *respLen*,
unpack_dms_SLQSSwiGetFirmwareCurr_t * *pOutput*)

get currently active image unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.109 int unpack_dms_SLQSSwiGetFwUpdateStatus (uint8_t * *pResp*, uint16_t *respLen*,
unpack_dms_SLQSSwiGetFwUpdateStatus_t * *pOutput*)

To get Firmware Update status unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.110 `int unpack_dms_SLQSSwiGetHostDevInfo (uint8_t * pResp, uint16_t respLen,
unpack_dms_SLQSSwiGetHostDevInfo_t * pOutput)`

To get Host Dev Info unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.111 `int unpack_dms_SLQSSwiGetOSInfo (uint8_t * pResp, uint16_t respLen, unpack_dms_SLQSSwiGetOSInfo_t * pOutput)`

To get Host OS Info unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.112 `int unpack_dms_SLQSSwiGetSerialNoExt (uint8_t * pResp, uint16_t respLen,
unpack_dms_SLQSSwiGetSerialNoExt_t * pOutput)`

To get serial number extension unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.113 int unpack_dms_SLQSSwiSetDyingGaspCfg (uint8_t * *pResp*, uint16_t *respLen*,
unpack_dms_SLQSSwiSetDyingGaspCfg_t * *pOutput*)

Set Dying GASP Config unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.114 int unpack_dms_SLQSSwiSetHostDevInfo (uint8_t * *pResp*, uint16_t *respLen*,
unpack_dms_SLQSSwiSetHostDevInfo_t * *pOutput*)

To get Host Dev Info unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.115 `int unpack_dms_SLQSSwiSetOSInfo (uint8_t * pResp, uint16_t respLen, unpack_dms_SLQSSwiSetOSInfo_t * pOutput)`

To get Host OS Info unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.116 `int unpack_dms_SLQSUIGetState (uint8_t * pResp, uint16_t respLen, unpack_dms_SLQSUIGetState_t * pOutput)`

To get UIM Get State unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.117 `int unpack_dms_UIMChangePIN (uint8_t * pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t * pOutput)`

To get UIM change PIN unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.118 int unpack_dms_UIMGetControlKeyStatus (uint8_t * *pResp*, uint16_t *respLen*,
unpack_dms_UIMGetControlKeyStatus_t * *pOutput*)

To get UIM Get ControlKeyStatus unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.119 int unpack_dms_UIMGetICCID (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMGetICCID_t * *pOutput*)

Unpacks the UIMGetICCID response message to a user-provided response structure.

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.120 `int unpack_dms_UIMGetPINStatus (uint8_t * pResp, uint16_t respLen, unpack_dms_UIMGetPINStatus_t * pOutput)`

To get UIM Get PIN Status unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.121 `int unpack_dms_UIMSetControlKeyProtection (uint8_t * pResp, uint16_t respLen, unpack_dms_UIMSetControlKeyProtection_t * pOutput)`

To get UIM Set Control Key Protection unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.122 `int unpack_dms_UIMSetPINProtection (uint8_t * pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t * pOutput)`

To get UIM Set PIN protection unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.123 int unpack_dms_UIMUnblockControlKey (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMUnblockControlKey_t * *pOutput*)

To get UIM Unblock Control Key unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.124 int unpack_dms_UIMUnblockPIN (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetPINProtection_t * *pOutput*)

To get UIM unblock PIN unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.125 `int unpack_dms_UIMVerifyPIN (uint8_t * pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t * pOutput)`

To get UIM verify PIN unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.3.2.126 `int unpack_dms_ValidateSPC (uint8_t * pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t * pOutput)`

To get UIM Unblock Control Key unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.4 fms.h File Reference

Data Structures

- struct [CarrierImage_t](#)

- struct [pack_fms_GetImagesPreference_t](#)
- struct [FMSImageElement](#)
- struct [FMSPrefImageList](#)
- struct [unpack_fms_GetImagesPreference_t](#)
- struct [pack_fms_GetStoredImages_t](#)
- struct [FMSImageElement](#)
- struct [FMSImageIDEntries](#)
- struct [FMSImageList](#)
- struct [unpack_fms_GetStoredImages_t](#)
- struct [pack_fms_SetImagesPreference_t](#)
- struct [unpack_fms_SetImagesPreference_t](#)

Macros

- `#define FMS_GOBI_MBN_IMG_ID_STR_LEN 16`
- `#define FMS_GOBI_MBN_BUILD_ID_STR_LEN 100`
- `#define FMS_GOBI_LISTENTRIES_MAX 2`
- `#define FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE 255`
- `#define FMS_MAX_IMAGE_ID_ELEMENT 50`
- `#define FMS_IMAGE_ID_MAX_ENTRIES 2`
- `#define FMS_FW_PRI_BUILD_MATCH_LEN 11`
- `#define FMS_IMAGE_ID_IMG_ID_LEN 16`
- `#define FMS_IMAGE_ID_BUILD_ID_LEN 32`
- `#define FMS_IMAGE_ID_PRI_IMGTYPE 0x01`

Functions

- `int pack_fms_GetImagesPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_GetImagesPreference_t *reqArg)`
- `int unpack_fms_GetImagesPreference (uint8_t *pResp, uint16_t respLen, unpack_fms_GetImagesPreference_t *pOutput)`
- `int pack_fms_GetStoredImages (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_GetStoredImages_t *reqArg)`
- `int unpack_fms_GetStoredImages (uint8_t *pResp, uint16_t respLen, unpack_fms_GetStoredImages_t *pOutput)`
- `int pack_fms_SetImagesPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_SetImagesPreference_t *reqArg)`
- `int unpack_fms_SetImagesPreference (uint8_t *pResp, uint16_t respLen, unpack_fms_SetImagesPreference_t *pOutput)`
- `uint32_t GetValidFwPriCombinations (FMSImageList *pStoredImageList, uint32_t *pValidCombinationSize, CarrierImage_t *pValidCombinations)`

9.4.1 Macro Definition Documentation

9.4.1.1 `#define FMS_FW_PRI_BUILD_MATCH_LEN 11`

9.4.1.2 `#define FMS_GOBI_LISTENTRIES_MAX 2`

9.4.1.3 `#define FMS_GOBI_MBN_BUILD_ID_STR_LEN 100`

9.4.1.4 `#define FMS_GOB_I_MBN_IMG_ID_STR_LEN 16`

9.4.1.5 `#define FMS_IMAGE_ID_BUILD_ID_LEN 32`

9.4.1.6 `#define FMS_IMAGE_ID_IMG_ID_LEN 16`

9.4.1.7 `#define FMS_IMAGE_ID_MAX_ENTRIES 2`

9.4.1.8 `#define FMS_IMAGE_ID_PRI_IMGTYPE 0x01`

9.4.1.9 `#define FMS_MAX_IMAGE_ID_ELEMENT 50`

9.4.1.10 `#define FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE 255`

9.4.2 Function Documentation

9.4.2.1 `uint32_t GetValidFwPriCombinations (FMSImageList * pStoredImageList, uint32_t * pValidCombinationSize, CarrierImage_t * pValidCombinations)`

This API distills valid Firmware/PRI combinations from GetStoredImages result

Parameters

in	<i>pStoredImageList</i>	<ul style="list-style-type: none"> image list returned from GetStoredImages See FMSImageList
in, out	<i>pValidCombinationSize</i>	<ul style="list-style-type: none"> number of combination passed in and returned
out	<i>pValidCombinations</i>	<ul style="list-style-type: none"> valid combinations returned See CarrierImage_t

Returns

- `eQCWWAN_ERR_INVALID_ARG` - Invalid parameters
- `eQCWWAN_ERR_BUFFER_SZ` - No enough element to store combinatons returned

See also

See [qmerrno.h](#) for `eQCWWAN_`xxx error values

9.4.2.2 `int pack_fms_GetImagesPreference (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_fms_GetImagesPreference_t * reqArg)`

Get Images Preference pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

9.4.2.3 int pack_fms_GetStoredImages (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_fms_GetStoredImages_t * reqArg)

Get Images Preference pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

9.4.2.4 int pack_fms_SetImagesPreference (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_fms_SetImagesPreference_t * reqArg)

Set Images Preference pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

9.4.2.5 int unpack_fms_GetImagesPreference (uint8_t * pResp, uint16_t respLen, unpack_fms_GetImagesPreference_t * pOutput)

Get Images Preference unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

9.4.2.6 int unpack_fms_GetStoredImages (uint8_t * pResp, uint16_t respLen, unpack_fms_GetStoredImages_t * pOutput)

Get Images Preference unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

9.4.2.7 `int unpack_fms_SetImagesPreference (uint8_t * pResp, uint16_t respLen, unpack_fms_SetImagesPreference_t * pOutput)`

Set Images Preference unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

9.5 loc.h File Reference

Data Structures

- struct [loc_LocApplicationInfo](#)
- struct [loc_SV](#)
- struct [loc_SVInfo](#)
- struct [loc_GnssData](#)
- struct [loc_CellDb](#)
- struct [loc_ClkInfo](#)
- struct [loc_BdsSV](#)
- struct [loc_BdsSVInfo](#)
- struct [pack_loc_EventRegister_t](#)
- struct [unpack_loc_EventRegister_t](#)
- struct [pack_loc_SetExtPowerState_t](#)
- struct [unpack_loc_SetExtPowerState_t](#)
- struct [pack_loc_Start_t](#)
- struct [unpack_loc_Start_t](#)
- struct [pack_loc_Stop_t](#)
- struct [unpack_loc_Stop_t](#)
- struct [pack_loc_SetOperationMode_t](#)
- struct [unpack_loc_SetOperationMode_t](#)
- struct [pack_loc_Delete_Assist_Data_t](#)
- struct [unpack_loc_Delete_Assist_Data_t](#)
- struct [loc_precisionDilution](#)
- struct [loc_sensorDataUsage](#)
- struct [loc_svUsedforFix](#)
- struct [loc_gpsTime](#)
- struct [unpack_loc_PositionRpt_Ind_t](#)
- struct [unpack_loc_EngineState_Ind_t](#)
- struct [unpack_loc_SetExtPowerConfig_Ind_t](#)
- struct [unpack_loc_SLQSLOCGetBestAvailPos_t](#)
- struct [pack_loc_SLQSLOCGetBestAvailPos_t](#)
- struct [unpack_loc_BestAvailPos_Ind_t](#)

- struct [unpack_loc_SetOperationMode_Ind_t](#)
- struct [unpack_loc_DeleteAssistData_Ind_t](#)
- struct [loc_satelliteInfo](#)
- struct [unpack_loc_GnssSvInfo_Ind_t](#)
- struct [pack_loc_SLQSLOCInjectUTCtime_t](#)
- struct [altSrcInfo_t](#)
- struct [pack_loc_SLQSLOCInjectPosition_t](#)
- struct [pack_loc_SLQSLOCSetCradleMountConfig_t](#)
- struct [sensorData_t](#)
- struct [tempData_t](#)
- struct [pack_loc_SLQSLOCInjectSensorData_t](#)

Macros

- #define [LOC_UINT8_MAX_STRING_SZ](#) 255
- #define [LOCEVENTMASKPOSITIONREPORT](#) 0x00000001
- #define [LOCEVENTMASKGNSSSVINFO](#) 0x00000002
- #define [LOCEVENTMASKNMEA](#) 0x00000004
- #define [LOCEVENTMASKNINOTIFYVERIFYREQ](#) 0x00000008
- #define [LOCEVENTMASKINJECTTIMERREQ](#) 0x00000010
- #define [LOCEVENTMASKINJECTPREDICTEDORBITSREQ](#) 0x00000020
- #define [LOCEVENTMASKINJECTPOSITIONREQ](#) 0x00000040
- #define [LOCEVENTMASKENGINESTATE](#) 0x00000080
- #define [LOCEVENTMASKFIXSESSIONSTATE](#) 0x00000100
- #define [LOCEVENTMASKWIFIREQ](#) 0x00000200
- #define [LOCEVENTMASKSENSORSTREAMINGREADYSTATUS](#) 0x00000400
- #define [LOCEVENTMASKTIMESYNCREQ](#) 0x00000800
- #define [LOCEVENTMASKSETSPSTREAMINGREPORT](#) 0x00001000
- #define [LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ](#) 0x00002000
- #define [LOCEVENTMASKNIGEOFENCENOTIFICATION](#) 0x00004000
- #define [LOCEVENTMASKGEOFENCEGENALERT](#) 0x00008000
- #define [LOCEVENTMASKGEOFENCEBREACHNOTIFICATION](#) 0x00010000
- #define [LOCEVENTMASKPEDOMETERCONTROL](#) 0x00020000
- #define [LOCEVENTMASKMOTIONDATACONTROL](#) 0x00040000
- #define [LOCEVENTMASKBATCHFULLNOTIFICATION](#) 0x00080000
- #define [LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT](#) 0x00100000
- #define [LOCEVENTMASKINJECTWIFIAPDATAREQ](#) 0x00200000
- #define [LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION](#) 0x00400000
- #define [LOCEVENTMASKVEHICLEDATAREADYSTATUS](#) 0x00800000
- #define [LOCEVENTMASKGNSSMEASUREMENTREPORT](#) 0x01000000
- #define [LOCEVENTMASKINVALIDVALUE](#) 0xFFFFFFFF
- #define [MAX_SENSOR_DATA_LEN](#) 64
- #define [MAX_TEMP_DATA_LEN](#) 64

Enumerations

- enum {
[eQMI_LOC_SESS_STATUS_SUCCESS](#) =0,
[eQMI_LOC_SESS_STATUS_IN_PROGRESS](#) =1,
[eQMI_LOC_SESS_STATUS_FAILURE](#) =2,
[eQMI_LOC_SESS_STATUS_TIMEOUT](#) =3 }

Functions

- int [pack_loc_EventRegister](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_loc_EventRegister_t](#) *reqArg)
- int [unpack_loc_EventRegister](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_EventRegister_t](#) *pOutput)
- int [pack_loc_SetExtPowerState](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_loc_SetExtPowerState_t](#) *reqArg)
- int [unpack_loc_SetExtPowerState](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_SetExtPowerState_t](#) *pOutput)
- int [pack_loc_Start](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_loc_Start_t](#) *reqArg)
- int [unpack_loc_Start](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_Start_t](#) *pOutput)
- int [pack_loc_Stop](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_loc_Stop_t](#) *reqArg)
- int [unpack_loc_Stop](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_Stop_t](#) *pOutput)
- int [pack_loc_SetOperationMode](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_loc_SetOperationMode_t](#) *reqArg)
- int [unpack_loc_SetOperationMode](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_SetOperationMode_t](#) *pOutput)
- int [pack_loc_DeleteAssistData](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_loc_DeleteAssistData_t](#) *reqArg)
- int [unpack_loc_DeleteAssistData](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_DeleteAssistData_t](#) *pOutput)
- int [unpack_loc_PositionRpt_Ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_PositionRpt_Ind_t](#) *pOutput)
- int [unpack_loc_EngineState_Ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_EngineState_Ind_t](#) *pOutput)
- int [unpack_loc_SetExtPowerConfig_Ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_SetExtPowerConfig_Ind_t](#) *pOutput)
- int [pack_loc_SLQSLOCGetBestAvailPos](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_loc_SLQSLOCGetBestAvailPos_t](#) *reqArg)
- int [unpack_loc_SLQSLOCGetBestAvailPos](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_SLQSLOCGetBestAvailPos_t](#) *pOutput)
- int [unpack_loc_BestAvailPos_Ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_BestAvailPos_Ind_t](#) *pOutput)
- int [unpack_loc_SetOperationMode_Ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_SetOperationMode_Ind_t](#) *pOutput)
- int [unpack_loc_DeleteAssistData_Ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_DeleteAssistData_Ind_t](#) *pOutput)
- int [unpack_loc_GnssSvInfo_Ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_loc_GnssSvInfo_Ind_t](#) *pOutput)
- int [pack_loc_SLQSLOCInjectUTCtime](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_loc_SLQSLOCInjectUTCtime_t](#) *reqArg)
- int [unpack_loc_SLQSLOCInjectUTCtime](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_loc_SLQSLOCInjectPosition](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_loc_SLQSLOCInjectPosition_t](#) *reqArg)
- int [unpack_loc_SLQSLOCInjectPosition](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_loc_SLQSLOCSetCradleMountConfig](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_loc_SLQSLOCSetCradleMountConfig_t](#) *reqArg)
- int [unpack_loc_SLQSLOCSetCradleMountConfig](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_loc_SLQSLOCInjectSensorData](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_loc_SLQSLOCInjectSensorData_t](#) *reqArg)
- int [unpack_loc_SLQSLOCInjectSensorData](#) (uint8_t *pResp, uint16_t respLen)

9.5.1 Macro Definition Documentation

9.5.1.1 `#define LOC_UINT8_MAX_STRING_SZ 255`

9.5.1.2 `#define LOCEVENTMASKBATCHFULLNOTIFICATION 0x00080000`

The control point must enable this mask to receive notification when a batch is full. The location engine sends this event to notify of Batch Full for ongoing batching session.

9.5.1.3 #define LOCEVENTMASKENGINESTATE 0x00000080

The control point must enable this mask to receive engine state report event indications.

9.5.1.4 #define LOCEVENTMASKFIXSESSIONSTATE 0x00000100

The control point must enable this mask to receive fix session status report event indications.

9.5.1.5 #define LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION 0x00400000

The control point must enable this mask to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach notification is for multiple Geofences. Breaches from multiple Geofences are all batched and sent in the same notification.

9.5.1.6 #define LOCEVENTMASKGEOFENCEBREACHNOTIFICATION 0x00010000

The control point must enable this mask to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach report is for a single Geofence.

9.5.1.7 #define LOCEVENTMASKGEOFENCEGENALERT 0x00008000

The control point must enable this mask to receive Geofence alerts. These alerts are generated to inform the client of the changes that may affect a Geofence, for example, if GPS is turned off or if the network is unavailable.

9.5.1.8 #define LOCEVENTMASKGNSSMEASUREMENTREPORT 0x01000000

The control point must enable this mask to receive system clock and satellite measurement report events (system clock, [SV](#) time, Doppler, etc.). Reports are generated only for the GNSS satellite constellations that are enabled using QMI_LOC_SET_GNSS_CONSTELL_REPORT_CONFIG(Not yet supported).

9.5.1.9 #define LOCEVENTMASKGNSSSVINFO 0x00000002

The control point must enable this mask to receive satellite report event indications. These reports are sent at a 1 Hz rate.

9.5.1.10 #define LOCEVENTMASKINJECTPOSITIONREQ 0x00000040

The control point must enable this mask to receive position injection request event indications.

9.5.1.11 #define LOCEVENTMASKINJECTPREDICTEDORBITSREQ 0x00000020

The control point must enable this mask to receive predicted orbits request event indications.

9.5.1.12 #define LOCEVENTMASKINJECTTIMEREQ 0x00000010

The control point must enable this mask to receive time injection request event indications.

9.5.1.13 #define LOCEVENTMASKINJECTWIFIAPDATAREQ 0x00200000

The control point must enable this mask to receive Wi-Fi Access Point (AP) data inject request event indications.

9.5.1.14 #define LOCEVENTMASKINVALIDVALUE 0xFFFFFFFF

Invalid Event Mask

9.5.1.15 #define LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT 0x00100000

The control point must enable this mask to receive position report indications along with an ongoing batching session. The location engine sends this event to notify the batched position report while a batching session is ongoing.

9.5.1.16 #define LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ 0x00002000

The control point must enable this mask to receive location server requests. These requests are generated when the service wishes to establish a connection with a location server.

9.5.1.17 #define LOCEVENTMASKMOTIONDATACONTROL 0x00040000

The control point must enable this mask to register for motion data control requests from the location engine. The location engine sends this event to control the injection of motion data.

9.5.1.18 #define LOCEVENTMASKNIGEOFENCENOTIFICATION 0x00004000

The control point must enable this mask to receive notifications related to network-initiated Geofences. These events notify the client when a network-initiated Geofence is added, deleted, or edited.

9.5.1.19 #define LOCEVENTMASKNINOTIFYVERIFYREQ 0x00000008

The control point must enable this mask to receive NI Notify/Verify request event indications.

9.5.1.20 #define LOCEVENTMASKNMEA 0x00000004

The control point must enable this mask to receive NMEA reports for position and satellites in view. The report is at a 1 Hz rate.

9.5.1.21 #define LOCEVENTMASKPEDOMETERCONTROL 0x00020000

The control point must enable this mask to register for pedometer control requests from the location engine. The location engine sends this event to control the injection of pedometer reports.

9.5.1.22 #define LOCEVENTMASKPOSITIONREPORT 0x00000001

The control point must enable this mask to receive position report event indications.

9.5.1.23 #define LOCEVENTMASKSENSORSTREAMINGREADYSTATUS 0x00000400

The control point must enable this mask to receive notifications from the location engine indicating its readiness to accept data from the sensors (accelerometer, gyroscope, etc.).

9.5.1.24 #define LOCEVENTMASKSETSPISTREAMINGREPORT 0x00001000

The control point must enable this mask to receive Stationary Position Indicator (SPI) streaming report indications.

9.5.1.25 #define LOCEVENTMASKTIMESYNCREQ 0x00000800

The control point must enable this mask to receive time sync requests from the GPS engine. Time sync enables the GPS engine to synchronize its clock with the sensor processor's clock.

9.5.1.26 #define LOCEVENTMASKVEHICLEDATAREADYSTATUS 0x00800000

The control point must enable this mask to receive notifications from the location engine indicating its readiness to accept vehicle data (vehicle accelerometer, vehicle angular rate, vehicle odometry, etc.).

9.5.1.27 #define LOCEVENTMASKWIFIREQ 0x00000200

The control point must enable this mask to receive Wi-Fi position request event indications.

9.5.1.28 #define MAX_SENSOR_DATA_LEN 64**9.5.1.29 #define MAX_TEMP_DATA_LEN 64****9.5.2 Enumeration Type Documentation****9.5.2.1 anonymous enum**

Enumerator

eQMI_LOC_SESS_STATUS_SUCCESS
eQMI_LOC_SESS_STATUS_IN_PROGRESS
eQMI_LOC_SESS_STATUS_FAILURE
eQMI_LOC_SESS_STATUS_TIMEOUT

9.5.3 Function Documentation

9.5.3.1 `int pack_loc_DeleteAssistData (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_Delete_Assist_Data_t * reqArg)`

Delete Assistant Data pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.2 `int pack_loc_EventRegister (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_EventRegister_t * reqArg)`

Event Register pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.3 `int pack_loc_SetExtPowerState (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_SetExtPowerState_t * reqArg)`

Set Ext Power State pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.4 int pack_loc_SetOperationMode (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen,
pack_loc_SetOperationMode_t * reqArg)

Set Operation Mode pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.5 int pack_loc_SLQSLOCGetBestAvailPos (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen,
pack_loc_SLQSLOCGetBestAvailPos_t * reqArg)

Get Best Avail position pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.6 `int pack_loc_SLQSLOCInjectPosition (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_SLQSLOCInjectPosition_t * reqArg)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.7 `int pack_loc_SLQSLOCInjectSensorData (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_SLQSLOCInjectSensorData_t * reqArg)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.8 `int pack_loc_SLQSLOCInjectUTCTime (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_SLQSLOCInjectUTCTime_t * reqArg)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.9 `int pack_loc_SLQSLOCSetCradleMountConfig (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_SLQSLOCSetCradleMountConfig_t * reqArg)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.10 `int pack_loc_Start (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_Start_t * reqArg)`

LOC Start pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.11 `int pack_loc_Stop (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_Stop_t * reqArg)`

Loc Stop pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Generated by Doxygen

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.12 `int unpack_loc_BestAvailPos_Ind (uint8_t * pResp, uint16_t respLen, unpack_loc_BestAvailPos_Ind_t * pOutput)`

Loc Best Avial position Indication unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.13 `int unpack_loc_DeleteAssistData (uint8_t * pResp, uint16_t respLen, unpack_loc_Delete_Assist_Data_t * pOutput)`

Delete Assistant Data unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.14 `int unpack_loc_DeleteAssistData_Ind (uint8_t * pResp, uint16_t respLen, unpack_loc_DeleteAssistData_Ind_t * pOutput)`

Unpack the status of delete the location engine assistance data

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.15 `int unpack_loc_EngineState_Ind (uint8_t * pResp, uint16_t respLen, unpack_loc_EngineState_Ind_t * pOutput)`

Loc Engine State Indication unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.16 `int unpack_loc_EventRegister (uint8_t * pResp, uint16_t respLen, unpack_loc_EventRegister_t * pOutput)`

Event Register unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

```
9.5.3.17 int unpack_loc_GnssSvInfo_Ind ( uint8_t * pResp, uint16_t respLen, unpack_loc_GnssSvInfo_Ind_t * pOutput )
```

Unpack the GNSS [SV](#) Info Indication.

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

```
9.5.3.18 int unpack_loc_PositionRpt_Ind ( uint8_t * pResp, uint16_t respLen, unpack_loc_PositionRpt_Ind_t * pOutput )
```

Loc Position Report Indication unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.19 `int unpack_loc_SetExtPowerConfig_Ind (uint8_t * pResp, uint16_t respLen, unpack_loc_SetExtPowerConfig_Ind_t * pOutput)`

Loc Set External Power Configure Indication unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.20 `int unpack_loc_SetExtPowerState (uint8_t * pResp, uint16_t respLen, unpack_loc_SetExtPowerState_t * pOutput)`

Set Ext Power State unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.21 `int unpack_loc_SetOperationMode (uint8_t * pResp, uint16_t respLen, unpack_loc_SetOperationMode_t * pOutput)`

Set Operation Mode unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.22 `int unpack_loc_SetOperationMode_Ind (uint8_t * pResp, uint16_t respLen, unpack_loc_SetOperationMode_Ind_t * pOutput)`

Unpack the engine to use the specified operation mode.

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.23 `int unpack_loc_SLQSLOCGetBestAvailPos (uint8_t * pResp, uint16_t respLen, unpack_loc_SLQSLOCGetBestAvailPos_t * pOutput)`

Get Best Avail position unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.24 `int unpack_loc_SLQSLOCInjectPosition (uint8_t * pResp, uint16_t respLen)`**Parameters**

<i>in</i>	<i>pResp</i>	qmi response
<i>in</i>	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.25 `int unpack_loc_SLQSLOCInjectSensorData (uint8_t * pResp, uint16_t respLen)`**Parameters**

<i>in</i>	<i>pResp</i>	qmi response
<i>in</i>	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.26 `int unpack_loc_SLQSLOCInjectUTCTime (uint8_t * pResp, uint16_t respLen)`**Parameters**

<i>in</i>	<i>pResp</i>	qmi response
<i>in</i>	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.27 `int unpack_loc_SLQSLOCSetCradleMountConfig (uint8_t * pResp, uint16_t respLen)`**Parameters**

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.28 int unpack_loc_Start (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_Start_t * *pOutput*)

Loc Start unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.5.3.29 int unpack_loc_Stop (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_Stop_t * *pOutput*)

Loc Stop unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6 nas.h File Reference

Data Structures

- struct [unpack_nas_GetSignalStrengths_t](#)
- struct [unpack_nas_SLQSGetSysSelectionPref_t](#)
- struct [nas_netSelectionPref](#)
- struct [nas_acqOrderPref](#)
- struct [nas_CSGID](#)
- struct [pack_nas_SLQSSetSysSelectionPref_t](#)
- struct [pack_nas_SLQSNasIndicationRegisterExt_t](#)
- struct [RFBandInfoElements](#)
- struct [unpack_nas_GetRFInfo_t](#)
- struct [cdmaSSInfo](#)
- struct [hdrSSInfo](#)
- struct [lteSSInfo](#)
- struct [tdscdmaSigInfoExt](#)
- struct [unpack_nas_SLQSNasGetSigInfo_t](#)
- struct [unpack_nas_SLQSNasSigInfoCallback_ind_t](#)
- struct [unpack_nas_GetHomeNetwork_t](#)
- struct [nas_SrvStatusInfo](#)
- struct [nas_GSMSrvStatusInfo](#)
- struct [nas_sysInfoCommon](#)
- struct [nas_CDMASysInfo](#)
- struct [nas_HDRSysInfo](#)
- struct [nas_GSMSysInfo](#)
- struct [nas_WCDMASysInfo](#)
- struct [nas_LTESysInfo](#)
- struct [nas_AddCDMASysInfo](#)
- struct [nas_AddSysInfo](#)
- struct [nas_CallBarringSysInfo](#)
- struct [unpack_nas_SLQSGetSysInfo_t](#)
- struct [unpack_nas_SLQSSysInfoCallback_ind_t](#)
- struct [unpack_nas_GetServingNetwork_t](#)
- struct [unpack_nas_GetServingNetworkCapabilities_t](#)
- struct [nas_QmiNas3GppNetworkInfo](#)
- struct [nas_QmiNas3GppNetworkRAT](#)
- struct [nas_QmisNasPcsDigit](#)
- struct [unpack_nas_PerformNetworkScan_t](#)
- struct [unpack_nas_SLQSSwiGetLteCQI_t](#)
- struct [nas_CommInfo](#)
- struct [nas_LTEInfo](#)
- struct [unpack_nas_SLQSNasSwiModemStatus_t](#)
- struct [nas_servSystem](#)
- struct [nas_dataSrvCapabilities](#)
- struct [nas_currentPLMN](#)
- struct [nas_roamIndList](#)
- struct [nas_qaQmi3Gpp2TimeZone](#)
- struct [nas_detailSvcInfo](#)
- struct [nas_CDMASysInfoExt](#)
- struct [nas_callBarStatus](#)
- struct [unpack_nas_SLQSGetServingSystem_t](#)
- struct [nas_rxSignalStrengthListElement](#)
- struct [nas_ecioListElement](#)

- struct [nas_errorRateListElement](#)
- struct [nas_rsrqInformation](#)
- struct [nas_lteSnrinformation](#)
- struct [nas_lteRsrpinformation](#)
- struct [unpack_nas_SLQSGetSignalStrength_t](#)
- struct [nas_SLQSSignalStrengthsIndReq](#)
- struct [pack_nas_SLQSSetSignalStrengthsCallback_t](#)
- struct [nas_SLQSSignalStrengthsInformation](#)
- struct [nas_RejectReasonTlv](#)
- struct [nas_SignalStrengthTlv](#)
- struct [nas_RFInfoTlv](#)
- struct [nas_SLQSSignalStrengthsTlv](#)
- struct [unpack_nas_SetEventReportInd_t](#)
- struct [unpack_nas_GetCDMANetworkParameters_t](#)
- struct [pack_nas_SetACCOLC_t](#)
- struct [nas_CDMARSSIThresh](#)
- struct [nas_CDMAECIOThresh](#)
- struct [nas_HDRRSSIThresh](#)
- struct [nas_HDRECIOThresh](#)
- struct [nas_HDRSINRThreshold](#)
- struct [nas_HDRIOThresh](#)
- struct [nas_GSMRSSIThresh](#)
- struct [nas_WCDMARSSIThresh](#)
- struct [nas_WCDMAECIOThresh](#)
- struct [nas_LTERSSIThresh](#)
- struct [nas_LTESNRThreshold](#)
- struct [nas_LTERSRQThresh](#)
- struct [nas_LTERSRPThresh](#)
- struct [nas_LTESigRptConfig](#)
- struct [nas_TDSCDMARSCPTthresh](#)
- struct [nas_TDSCDMARSSIThresh](#)
- struct [nas_TDSCDMAECIOThresh](#)
- struct [nas_TDSCDMASINRThresh](#)
- struct [pack_nas_SLQSNasConfigSigInfo2_t](#)
- struct [unpack_nas_SetDataCapabilitiesCallback_ind_t](#)
- struct [unpack_nas_GetNetworkPreference_t](#)
- struct [pack_nas_SetNetworkPreference_t](#)
- struct [unpack_nas_SetNetworkPreference_t](#)
- struct [unpack_nas_SetRoamingIndicatorCallback_ind_t](#)
- struct [NAServingSystemInfo](#)
- struct [unpack_nas_SetServingSystemCallback_ind_t](#)
- struct [NASPhyCaAggScellIndType](#)
- struct [NASPhyCaAggScellIDBw](#)
- struct [NASPhyCaAggScellInfo](#)
- struct [NASPhyCaAggPcellInfo](#)
- struct [NASPhyCaAggScellIndex](#)
- struct [NasGetLTCphyCAInfo](#)
- struct [unpack_nas_SlqsGetLTCphyCAInfo_t](#)
- struct [NASEmergencyModeTlv](#)
- struct [NASModePreferenceTlv](#)
- struct [NASBandPreferenceTlv](#)
- struct [NASPRLPreferenceTlv](#)
- struct [NASRoamPreferenceTlv](#)
- struct [NASLTEBandPreferenceTlv](#)
- struct [NASNetSelPreferenceTlv](#)

- struct [NASServDomainPrefTlv](#)
- struct [NASGWAcqOrderPrefTlv](#)
- struct [NASQmiCbkNasSystemSelPrefInd](#)
- struct [unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t](#)
- struct [NASOTAMessageTlv](#)
- struct [NASLteNasReleaseInfoTlv](#)
- struct [NASTimeInfoTlv](#)
- struct [NASQmiCbkNasSwiOTAMessageInd](#)
- struct [unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t](#)
- struct [nas_MNRInfo](#)
- struct [pack_nas_SLQSInitiateNetworkRegistration_t](#)
- struct [pack_nas_SLQSNasSwiIndicationRegister_t](#)
- struct [pack_nas_SLQSGetPLMNName_t](#)
- struct [unpack_nas_SLQSGetPLMNName_t](#)
- struct [nas_nmrCellInfo](#)
- struct [nas_GERANInfo](#)
- struct [nas_geranInstInfo](#)
- struct [nas_UMTSinstInfo](#)
- struct [nas_UMTSInfo](#)
- struct [nas_CDMAInfo](#)
- struct [nas_cellParams](#)
- struct [nas_LTEInfoIntraFreq](#)
- struct [nas_infoInterFreq](#)
- struct [nas_LTEInfoInterFreq](#)
- struct [nas_gsmCellInfo](#)
- struct [nas_lteGsmCellInfo](#)
- struct [nas_LTEInfoNeighboringGSM](#)
- struct [nas_wcdmaCellInfo](#)
- struct [nas_lteWcdmaCellInfo](#)
- struct [nas_LTEInfoNeighboringWCDMA](#)
- struct [nas_umtsLTENbrCell](#)
- struct [nas_WCDMAInfoLTENeighborCell](#)
- struct [unpack_nas_SLQSNasGetCellLocationInfo_t](#)
- struct [nas_timeInfo](#)
- struct [unpack_nas_SLQSGetNetworkTime_t](#)
- struct [nas_UniversalTime](#)
- struct [unpack_nas_SLQSNasNetworkTimeCallBack_ind_t](#)
- struct [nas_PhyCaAggScellIndType](#)
- struct [nas_PhyCaAggScellIDBw](#)
- struct [nas_PhyCaAggScellInfo](#)
- struct [nas_PhyCaAggPcellInfo](#)
- struct [nas_PhyCaAggScellIndex](#)
- struct [unpack_nas_SetNasLTECphyCaIndCallback_ind_t](#)
- struct [nas_RxSigInfo](#)
- struct [nas_SccRxInfo](#)
- struct [unpack_nas_SLQSSwiGetLteSccRxInfo_t](#)
- struct [unpack_nas_SLQSNasTimerCallback_ind_t](#)

Macros

- [#define NAS_OTA_MESSAGE_MAX_BUF_SIZE](#) 2048
- [#define NAS_MAX_NUM_NETWORKS](#) 30
- [#define NAS_MAX_DESCRIPTION_LENGTH](#) 255
- [#define NAS_PLMN_LENGTH](#) 3
- [#define NAS_MAX_SCC_RX_INFO_INSTANCES](#) 255
- [#define NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST](#) 255

Enumerations

- enum LIBPACK_NAS_LTE_CPHY_SCELL_STATE {
eLIBPACK_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED =0x00,
eLIBPACK_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED =0x01,
eLIBPACK_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED =0x02 }
- enum LIBPACK_NAS_LTE_CPHY_CA_BW_NRB {
eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_6 =0x00,
eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_15 =0x01,
eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_25 =0x02,
eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_50 =0x03,
eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_75 =0x04,
eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_100 =0x05 }
- enum NAS_LTE_CPHY_CA_BW_NRB_LITE {
eNAS_LTE_CPHY_CA_BW_NRB_LITE_6 =0x00,
eNAS_LTE_CPHY_CA_BW_NRB_LITE_15 =0x01,
eNAS_LTE_CPHY_CA_BW_NRB_LITE_25 =0x02,
eNAS_LTE_CPHY_CA_BW_NRB_LITE_50 =0x03,
eNAS_LTE_CPHY_CA_BW_NRB_LITE_75 =0x04,
eNAS_LTE_CPHY_CA_BW_NRB_LITE_100 =0x05 }
- enum NAS_LTE_CPHY_SCELL_STATE_LITE {
eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE =0x00,
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE =0x01,
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE =0x02 }

Functions

- int unpack_nas_GetSignalStrengths (uint8_t *pResp, uint16_t respLen, unpack_nas_GetSignalStrengths_t *pOutput)
- int pack_nas_GetSignalStrengths (pack_qmi_t *pCtx, uint8_t *pReq, uint16_t *pLen)
- int pack_nas_SLQSGetSysSelectionPref (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetSysSelectionPref (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSysSelectionPref_t *pOutput)
- int pack_nas_SLQSSetSysSelectionPref (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSetSysSelectionPref_t *pReqParam)
- int unpack_nas_SLQSSetSysSelectionPref (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SLQSSetBandPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint64_t bandPref)
- int unpack_nas_SLQSSetBandPreference (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SLQSNasIndicationRegisterExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasIndicationRegisterExt_t *pReqParam)
- int unpack_nas_SLQSNasIndicationRegisterExt (uint8_t *pResp, uint16_t respLen)
- int pack_nas_GetRFInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetRFInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_GetRFInfo_t *pOutput)
- int pack_nas_SLQSNasGetSigInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNasGetSigInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetSigInfo_t *pOutput)
- int unpack_nas_SLQSNasSigInfoCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasSigInfoCallback_ind_t *pOutput)
- int unpack_nas_GetHomeNetwork (uint8_t *pResp, uint16_t respLen, unpack_nas_GetHomeNetwork_t *pOutput)
- int pack_nas_GetHomeNetwork (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int pack_nas_SLQSGetSysInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetSysInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSysInfo_t *pOutput)

- int [unpack_nas_SLQSNasSysInfoCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSSysInfoCallback_ind_t](#) *pOutput)
- int [pack_nas_GetServingNetwork](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_GetServingNetwork](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_GetServingNetwork_t](#) *pOutput)
- int [pack_nas_GetServingNetworkCapabilities](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_GetServingNetworkCapabilities](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_GetServingNetworkCapabilities_t](#) *pOutput)
- int [pack_nas_PerformNetworkScan](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_PerformNetworkScan](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_PerformNetworkScan_t](#) *pOutput)
- int [pack_nas_SLQSSwiGetLteCQI](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_SLQSSwiGetLteCQI](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSSwiGetLteCQI_t](#) *pOutput)
- int [pack_nas_SLQSNasSwiModemStatus](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_SLQSNasSwiModemStatus](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSNasSwiModemStatus_t](#) *pOutput)
- int [pack_nas_SLQSGetServingSystem](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_SLQSGetServingSystem](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSGetServingSystem_t](#) *pOutput)
- int [pack_nas_SLQSGetSignalStrength](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint16_t reqMask)
- int [unpack_nas_SLQSGetSignalStrength](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSGetSignalStrength_t](#) *pOutput)
- int [pack_nas_SLQSSetSignalStrengthsCallback](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_nas_SLQSSetSignalStrengthsCallback_t](#) *pReqParam)
- int [unpack_nas_SLQSSetSignalStrengthsCallback](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_nas_SetRFInfoCallback](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t *pBenable)
- int [unpack_nas_SetRFInfoCallback](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_nas_SetLURRejectCallback](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t *pBenable)
- int [unpack_nas_SetLURRejectCallback](#) (uint8_t *pResp, uint16_t respLen)
- int [unpack_nas_SetEventReportInd](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SetEventReportInd_t](#) *pOutput)
- int [pack_nas_GetCDMANetworkParameters](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_GetCDMANetworkParameters](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_GetCDMANetworkParameters_t](#) *pOutput)
- int [pack_nas_GetANAAAAuthenticationStatus](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_GetANAAAAuthenticationStatus](#) (uint8_t *pResp, uint16_t respLen, uint32_t *pAuthStatus)
- int [pack_nas_GetACCOLC](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_GetACCOLC](#) (uint8_t *pResp, uint16_t respLen, uint8_t *pAccolc)
- int [pack_nas_SetACCOLC](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_nas_SetACCOLC_t](#) reqParam)
- int [unpack_nas_SetACCOLC](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_nas_SLQSNasConfigSigInfo2](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_nas_SLQSNasConfigSigInfo2_t](#) *pReqParam)
- int [unpack_nas_SLQSNasConfigSigInfo2](#) (uint8_t *pResp, uint16_t respLen)
- int [unpack_nas_SetDataCapabilitiesCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SetDataCapabilitiesCallback_ind_t](#) *pOutput)
- int [pack_nas_GetNetworkPreference](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_GetNetworkPreference](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_GetNetworkPreference_t](#) *pOutput)
- int [pack_nas_SetNetworkPreference](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_nas_SetNetworkPreference_t](#) *reqArg)
- int [unpack_nas_SetNetworkPreference](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SetNetworkPreference_t](#) *pOutput)

- int [unpack_nas_SetRoamingIndicatorCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SetRoamingIndicatorCallback_ind_t](#) *pOutput)
- int [unpack_nas_SetServingSystemCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SetServingSystemCallback_ind_t](#) *pOutput)
- int [pack_nas_SLqsGetLTECphyCAInfo](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_SLqsGetLTECphyCAInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLqsGetLTECphyCAInfo_t](#) *pOutput)
- int [unpack_nas_SLQSSetSysSelectionPrefCallBack_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t](#) *pOutput)
- int [unpack_nas_SLQSNasSwiOTAMessageCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t](#) *pOutput)
- int [pack_nas_SLQSInitiateNetworkRegistration](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_nas_SLQSInitiateNetworkRegistration_t](#) *pReqParam)
- int [unpack_nas_SLQSInitiateNetworkRegistration](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_nas_SLQSNasSwiIndicationRegister](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_nas_SLQSNasSwiIndicationRegister_t](#) *pReqParam)
- int [unpack_nas_SLQSNasSwiIndicationRegister](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_nas_SLQSGetPLMNName](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_nas_SLQSGetPLMNName_t](#) *reqArg)
- int [unpack_nas_SLQSGetPLMNName](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSGetPLMNName_t](#) *pOutput)
- int [pack_nas_SLQSNasGetCellLocationInfo](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_SLQSNasGetCellLocationInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSNasGetCellLocationInfo_t](#) *pOutput)
- int [pack_nas_SLQSGetNetworkTime](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_nas_SLQSGetNetworkTime](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSGetNetworkTime_t](#) *pOutput)
- int [unpack_nas_SLQSNasNetworkTimeCallBack_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSNasNetworkTimeCallBack_ind_t](#) *pOutput)
- int [unpack_nas_SetNasLTECphyCaIndCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SetNasLTECphyCaIndCallback_ind_t](#) *pOutput)
- int [pack_nas_SLQSSwiGetLteSccRxInfo](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReq, uint16_t *pLen)
- int [unpack_nas_SLQSSwiGetLteSccRxInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSSwiGetLteSccRxInfo_t](#) *pOutput)
- int [unpack_nas_SLQSNasTimerCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_nas_SLQSNasTimerCallback_ind_t](#) *pOutput)

9.6.1 Macro Definition Documentation

9.6.1.1 `#define NAS_MAX_DESCRIPTION_LENGTH 255`

9.6.1.2 `#define NAS_MAX_NUM_NETWORKS 30`

9.6.1.3 `#define NAS_MAX_SCC_RX_INFO_INSTANCES 255`

9.6.1.4 `#define NAS_OTA_MESSAGE_MAX_BUF_SIZE 2048`

9.6.1.5 `#define NAS_PLMN_LENGTH 3`

9.6.1.6 `#define NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST 255`

9.6.2 Enumeration Type Documentation

9.6.2.1 enum LIBPACK_NAS_LTE_CPHY_CA_BW_NRB

Enumerator

eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_6
eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_15
eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_25
eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_50
eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_75
eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_100

9.6.2.2 enum LIBPACK_NAS_LTE_CPHY_SCELL_STATE

Enumerator

eLIBPACK_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED
eLIBPACK_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED
eLIBPACK_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED

9.6.2.3 enum NAS_LTE_CPHY_CA_BW_NRB_LITE

Enumerator

eNAS_LTE_CPHY_CA_BW_NRB_LITE_6
eNAS_LTE_CPHY_CA_BW_NRB_LITE_15
eNAS_LTE_CPHY_CA_BW_NRB_LITE_25
eNAS_LTE_CPHY_CA_BW_NRB_LITE_50
eNAS_LTE_CPHY_CA_BW_NRB_LITE_75
eNAS_LTE_CPHY_CA_BW_NRB_LITE_100

9.6.2.4 enum NAS_LTE_CPHY_SCELL_STATE_LITE

Enumerator

eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE

9.6.3 Function Documentation

9.6.3.1 int pack_nas_GetACCOLC (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.2 `int pack_nas_GetANAAAuthenticationStatus (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.3 `int pack_nas_GetCDMANetworkParameters (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.4 `int pack_nas_GetHomeNetwork (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

get home network pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.5 int pack_nas_GetNetworkPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

9.6.3.6 int pack_nas_GetRFInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get rf info pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.7 int pack_nas_GetServingNetwork (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.8 int pack_nas_GetServingNetworkCapabilities (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.9 int pack_nas_GetSignalStrengths (pack_qmi_t * *pCtx*, uint8_t * *pReq*, uint16_t * *pLen*)

get signal strengths pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.10 int pack_nas_PerformNetworkScan (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.11 `int pack_nas_SetACCOLC (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SetACCOLC_t reqParam)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.12 `int pack_nas_SetLURjectCallback (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, uint8_t * pBenable)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>bEnable</i>	0/1 value to disable/enable indication respectively

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.13 `int pack_nas_SetNetworkPreference (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SetNetworkPreference_t * reqArg)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>pack</i>	default prototype

9.6.3.14 `int pack_nas_SetRFInfoCallback (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, uint8_t * pBenable)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>bEnable</i>	0/1 value to disable/enable indication respectively

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.15 `int pack_nas_SlqsGetLTEphyCAInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

9.6.3.16 `int pack_nas_SLQSGetNetworkTime (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.17 `int pack_nas_SLQSGetPLMNName (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SLQSGetPLMNName_t * reqArg)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.18 int pack_nas_SLQSGetServingSystem (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.19 int pack_nas_SLQSGetSignalStrength (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, uint16_t *reqMask*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqMask</i>	request mask for fetching extra signal info

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.20 int pack_nas_SLQSGetSysInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.21 `int pack_nas_SLQSGetSysSelectionPref (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.22 `int pack_nas_SLQSIInitiateNetworkRegistration (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SLQSIInitiateNetworkRegistration_t * pReqParam)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>pReqParam</i>	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.23 `int pack_nas_SLQSNasConfigSigInfo2 (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SLQSNasConfigSigInfo2_t * pReqParam)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.24 int pack_nas_SLQSNasGetCellLocationInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.25 int pack_nas_SLQSNasGetSigInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get sig info pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.26 int pack_nas_SLQSNasIndicationRegisterExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_nas_SLQSNasIndicationRegisterExt_t * *pReqParam*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
Generated by Doxygen	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.27 int pack_nas_SLQSNasSwiIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_nas_SLQSNasSwiIndicationRegister_t * *pReqParam*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>pReqParam</i>	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.28 int pack_nas_SLQSNasSwiModemStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.29 int pack_nas_SLQSSetBandPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, uint64_t *bandPref*
)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>band</i>	preference

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.30 int pack_nas_SLQSSetSignalStrengthsCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_nas_SLQSSetSignalStrengthsCallback_t * *pReqParam*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>pReqParam</i>	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.31 int pack_nas_SLQSSetSysSelectionPref (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_nas_SLQSSetSysSelectionPref_t * *pReqParam*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.32 int pack_nas_SLQSSwiGetLteCQI (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.33 `int pack_nas_SLQSSwiGetLteScsRxInfo (pack_qmi_t * pCtx, uint8_t * pReq, uint16_t * pLen)`

get LTE Secondary carrier Rx signal level information pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.34 `int unpack_nas_GetACCOLC (uint8_t * pResp, uint16_t respLen, uint8_t * pAccolc)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pAccolc</i>	accolc

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.35 `int unpack_nas_GetANAAAAAuthenticationStatus (uint8_t * pResp, uint16_t respLen, uint32_t * pAuthStatus)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>authStatus</i>	auth status

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.36 int unpack_nas_GetCDMANetworkParameters (uint8_t * *pResp*, uint16_t *respLen*,
unpack_nas_GetCDMANetworkParameters_t * *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	qmi output parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.37 int unpack_nas_GetHomeNetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetHomeNetwork_t *
pOutput)

get home network unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.38 int unpack_nas_GetNetworkPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetNetworkPreference_t * *pOutput*)

9.6.3.39 int unpack_nas_GetRFInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetRFInfo_t * *pOutput*)

get rf info unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.40 int unpack_nas_GetServingNetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetServingNetwork_t * *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.41 int unpack_nas_GetServingNetworkCapabilities (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetServingNetworkCapabilities_t * *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.42 `int unpack_nas_GetSignalStrengths (uint8_t * pResp, uint16_t respLen, unpack_nas_GetSignalStrengths_t * pOutput)`

get signal strengths unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.43 `int unpack_nas_PerformNetworkScan (uint8_t * pResp, uint16_t respLen, unpack_nas_PerformNetworkScan_t * pOutput)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.44 `int unpack_nas_SetACCOLC (uint8_t * pResp, uint16_t respLen)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.45 `int unpack_nas_SetDataCapabilitiesCallback_ind (uint8_t * pResp, uint16_t respLen, unpack_nas_SetDataCapabilitiesCallback_ind_t * pOutput)`

Data Capabilities indication unpack

Parameters

in	<i>pResp</i>	qmi indication from modem
in	<i>respLen</i>	qmi indication length
out	<i>pOutput</i>	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.46 `int unpack_nas_SetEventReportInd (uint8_t * pResp, uint16_t respLen, unpack_nas_SetEventReportInd_t * pOutput)`

9.6.3.47 `int unpack_nas_SetLURjectCallback (uint8_t * pResp, uint16_t respLen)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.48 `int unpack_nas_SetNasLTECphyCalIndCallback_ind (uint8_t * pResp, uint16_t respLen, unpack_nas_SetNasLTECphyCalIndCallback_ind_t * pOutput)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	sig info indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.49 int unpack_nas_SetNetworkPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetNetworkPreference_t * *pOutput*)

9.6.3.50 int unpack_nas_SetRFInfoCallback (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.51 int unpack_nas_SetRoamingIndicatorCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetRoamingIndicatorCallback_ind_t * *pOutput*)

Roaming indication unpack

Parameters

in	<i>pResp</i>	qmi indication from modem
in	<i>respLen</i>	qmi indication length
out	<i>pOutput</i>	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.52 int unpack_nas_SetServingSystemCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetServingSystemCallback_ind_t * *pOutput*)

9.6.3.53 `int unpack_nas_SlqsGetLTECphyCAInfo (uint8_t * pResp, uint16_t respLen, unpack_nas_SlqsGetLTECphyCAInfo_t * pOutput)`

9.6.3.54 `int unpack_nas_SLQSGetNetworkTime (uint8_t * pResp, uint16_t respLen, unpack_nas_SLQSGetNetworkTime_t * pOutput)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.55 `int unpack_nas_SLQSGetPLMNName (uint8_t * pResp, uint16_t respLen, unpack_nas_SLQSGetPLMNName_t * pOutput)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.56 `int unpack_nas_SLQSGetServingSystem (uint8_t * pResp, uint16_t respLen, unpack_nas_SLQSGetServingSystem_t * pOutput)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.57 int unpack_nas_SLQSGetSignalStrength (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSignalStrength_t * *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.58 int unpack_nas_SLQSGetSysInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSysInfo_t * *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.59 int unpack_nas_SLQSGetSysSelectionPref (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSysSelectionPref_t * *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.60 int unpack_nas_SLQSIInitiateNetworkRegistration (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.61 int unpack_nas_SLQSNasConfigSigInfo2 (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.62 int unpack_nas_SLQSNasGetCellLocationInfo (uint8_t * *pResp*, uint16_t *respLen*,
unpack_nas_SLQSNasGetCellLocationInfo_t * *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.63 int unpack_nas_SLQSNasGetSigInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGetSigInfo_t * *pOutput*)

get sig info unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.64 int unpack_nas_SLQSNasIndicationRegisterExt (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.65 int unpack_nas_SLQSNasNetworkTimeCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasNetworkTimeCallBack_ind_t * *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
Generated by Doxygen	<i>pOutput</i>	sig info indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.66 int unpack_nas_SLQSNasSigInfoCallback_ind (uint8_t * *pResp*, uint16_t *respLen*,
unpack_nas_SLQSNasSigInfoCallback_ind_t * *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	sig info indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.67 int unpack_nas_SLQSNasSwiIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.68 int unpack_nas_SLQSNasSwiModemStatus (uint8_t * *pResp*, uint16_t *respLen*,
unpack_nas_SLQSNasSwiModemStatus_t * *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.69 `int unpack_nas_SLQSNasSwiOTAMessageCallback_ind (uint8_t * pResp, uint16_t respLen,
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t * pOutput)`

OTA message indication unpack

Parameters

in	<i>pResp</i>	qmi indication from modem
in	<i>respLen</i>	qmi indication length
out	<i>pOutput</i>	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.70 `int unpack_nas_SLQSNasSysInfoCallback_ind (uint8_t * pResp, uint16_t respLen,
unpack_nas_SLQSSysInfoCallback_ind_t * pOutput)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.71 `int unpack_nas_SLQSNasTimerCallback_ind (uint8_t * pResp, uint16_t respLen,
unpack_nas_SLQSNasTimerCallback_ind_t * pOutput)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	network timer indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.72 int unpack_nas_SLQSSetBandPreference (uint8_t * *pResp*, uint16_t *respLen*)**Parameters**

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

9.6.3.73 int unpack_nas_SLQSSetSignalStrengthsCallback (uint8_t * *pResp*, uint16_t *respLen*)**Parameters**

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.74 int unpack_nas_SLQSSetSysSelectionPref (uint8_t * *pResp*, uint16_t *respLen*)**Parameters**

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.75 `int unpack_nas_SLQSSetSysSelectionPrefCallBack_ind (uint8_t * pResp, uint16_t respLen, unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t * pOutput)`

System Selection Preference indication unpack

Parameters

in	<i>pResp</i>	qmi indication from modem
in	<i>respLen</i>	qmi indication length
out	<i>pOutput</i>	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.76 `int unpack_nas_SLQSSwiGetLteCQI (uint8_t * pResp, uint16_t respLen, unpack_nas_SLQSSwiGetLteCQI_t * pOutput)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.6.3.77 `int unpack_nas_SLQSSwiGetLteScCRxInfo (uint8_t * pResp, uint16_t respLen, unpack_nas_SLQSSwiGetLteScCRxInfo_t * pOutput)`

get LTE Secondary carrier Rx signal level information unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
Generated by Doxygen	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.7 qaCbkCatEventReportInd.h File Reference

Data Structures

- struct [CatEventIDDataTlv](#)
- struct [CatAlphaIdentifierTlv](#)
- struct [CatEventListTlv](#)
- struct [CatRefreshTlv](#)
- struct [CatEndProactiveSessionTlv](#)
- union [currentCatEvent](#)
- struct [CatCommonEventTlv](#)
- struct [QmiCbkCatEventStatusReportInd](#)

Macros

- #define [QMI_MAX_CAT_EVENT_DATA_LENGTH](#) 255
- #define [QMI_CAN_COMMON_EVENT_TLV_NUMBER](#) 11

Enumerations

- enum [eQMI_CAT_EVENT_REPORT_IND_TLV](#) {
[eTLV_CBK_DISPLAY_TEXT](#) = 0x10,
[eTLV_CBK_GET_IN_KEY](#) = 0x11,
[eTLV_CBK_GET_INPUT](#) = 0x12,
[eTLV_CBK_SETUP_MENU](#) = 0x13,
[eTLV_CBK_SELECT_ITEM](#) = 0x14,
[eTLV_CBK_ALPHA_IDENTIFIER](#) = 0x15,
[eTLV_CBK_SETUP_EVENT_LIST](#) = 0x16,
[eTLV_CBK_SETUP_IDLE_MODE_TEXT](#) = 0x17,
[eTLV_CBK_LANGUAGE_NOTIFICATION](#) = 0x18,
[eTLV_CBK_REFRESH](#) = 0x19,
[eTLV_CBK_END_PROACTIVE_SESSION](#) = 0x1A }
- enum [eQMI_CAT_EVENT_REPORT_IND_TLV_LENGTH](#) {
[eTLV_SETUP_EVENT_LIST_LENGTH](#) = 0x04,
[eTLV_REFRESH_LENGTH](#) = 0x03,
[eTLV_END_PROACTIVE_SESSION_LENGTH](#) = 0x01 }

Functions

- enum [eQCWWANError](#) [UpkQmiCbkCatEventReportInd](#) (BYTE *pMdmResp, struct [QmiCbkCatEventStatusReportInd](#) *pAipResp)

9.7.1 Macro Definition Documentation

9.7.1.1 `#define QMI_CAN_COMMON_EVENT_TLV_NUMBER 11`

9.7.1.2 `#define QMI_MAX_CAT_EVENT_DATA_LENGTH 255`

9.7.2 Enumeration Type Documentation

9.7.2.1 `enum eQMI_CAT_EVENT_REPORT_IND_TLV`

Enumerator

eTLV_CBK_DISPLAY_TEXT
eTLV_CBK_GET_IN_KEY
eTLV_CBK_GET_INPUT
eTLV_CBK_SETUP_MENU
eTLV_CBK_SELECT_ITEM
eTLV_CBK_ALPHA_IDENTIFIER
eTLV_CBK_SETUP_EVENT_LIST
eTLV_CBK_SETUP_IDLE_MODE_TEXT
eTLV_CBK_LANGUAGE_NOTIFICATION
eTLV_CBK_REFRESH
eTLV_CBK_END_PROACTIVE_SESSION

9.7.2.2 `enum eQMI_CAT_EVENT_REPORT_IND_TLV_LENGTH`

Enumerator

eTLV_SETUP_EVENT_LIST_LENGTH
eTLV_REFRESH_LENGTH
eTLV_END_PROACTIVE_SESSION_LENGTH

9.7.3 Function Documentation

9.7.3.1 `enum eQCWWANError UpkQmiCbkCatEventReportInd (BYTE * pMdmResp, struct QmiCbkCatEventStatusReportInd * pAipResp)`

9.8 qaCbkSwiOmaDmEventReportInd.h File Reference

Data Structures

- struct [sessionInfoTlv](#)
- struct [sessionInfoTlvExt](#)
- struct [QmiCbkSwiOmaDmEventStatusReportInd](#)
- struct [QmiCbkSwiOmaDmEventStatusReportIndExt](#)

Macros

- #define QMI_SWIOMA_DM_FOTA 0x00
- #define QMI_SWIOMA_DM_CONFIG 0x01
- #define QMI_SWIOMA_DM_NOT 0x02

Enumerations

- enum eQMI_SWIOMA_DM_EVENT_REPORT_IND {
eTLV_IND_OMA_DM_FOTA = 0x10,
eTLV_IND_OMA_DM_CONFIG = 0x11,
eTLV_IND_OMA_DM_NOT = 0x12 }

Functions

- enum eQCWWANError UpkQmiCbkSwiOmaDmEventReportInd (BYTE *pMdmResp, struct QmiCbkSwiOmaDmEventStatusReportInd *pApiResp)
- package enum eQCWWANError UpkQmiCbkSwiOmaDmEventReportIndExt (BYTE *pMdmResp, struct QmiCbkSwiOmaDmEventStatusReportInd *pApiResp)

9.8.1 Macro Definition Documentation

9.8.1.1 #define QMI_SWIOMA_DM_CONFIG 0x01

9.8.1.2 #define QMI_SWIOMA_DM_FOTA 0x00

9.8.1.3 #define QMI_SWIOMA_DM_NOT 0x02

9.8.2 Enumeration Type Documentation

9.8.2.1 enum eQMI_SWIOMA_DM_EVENT_REPORT_IND

Enumerator

eTLV_IND_OMA_DM_FOTA
eTLV_IND_OMA_DM_CONFIG
eTLV_IND_OMA_DM_NOT

9.8.3 Function Documentation

9.8.3.1 enum eQCWWANError UpkQmiCbkSwiOmaDmEventReportInd (BYTE * *pMdmResp*, struct QmiCbkSwiOmaDmEventStatusReportInd * *pApiResp*)

9.8.3.2 package enum eQCWWANError UpkQmiCbkSwiOmaDmEventReportIndExt (BYTE * *pMdmResp*, struct QmiCbkSwiOmaDmEventStatusReportInd * *pApiResp*)

9.9 qaGobiApiAudio.h File Reference

Audio Service API function prototypes.

Data Structures

- struct [GetAudioProfileReq](#)
- struct [GetAudioProfileResp](#)
- struct [SetAudioProfileReq](#)
- struct [GetAudioPathConfigReq](#)
- struct [TXPCMIIRFtr](#)
- struct [RXPCMIIRFtr](#)
- struct [RXAGCList](#)
- struct [RXAVCList](#)
- struct [TXAGCList](#)
- struct [GetAudioPathConfigResp](#)
- struct [SetAudioPathConfigReq](#)
- struct [GetAudioVolTLBConfigReq](#)
- struct [GetAudioVolTLBConfigResp](#)
- struct [SetAudioVolTLBConfigReq](#)
- struct [SetAudioVolTLBConfigResp](#)

Functions

- [ULONG](#) [SLQSGetAudioProfile](#) ([GetAudioProfileReq](#) *pGetAudioProfileReq, [GetAudioProfileResp](#) *pGetAudioProfileResp)
- [ULONG](#) [SLQSSetAudioProfile](#) ([SetAudioProfileReq](#) *pSetAudioProfileReq)
- [ULONG](#) [SLQSGetAudioPathConfig](#) ([GetAudioPathConfigReq](#) *pGetAudioPathConfigReq, [GetAudioPathConfigResp](#) *pGetAudioPathConfigResp)
- [ULONG](#) [SLQSSetAudioPathConfig](#) ([SetAudioPathConfigReq](#) *pSetAudioPathConfigReq)
- [ULONG](#) [SLQSGetAudioVolTLBConfig](#) ([GetAudioVolTLBConfigReq](#) *pGetAudioVolTLBCfgReq, [GetAudioVolTLBConfigResp](#) *pGetAudioVolTLBCfgResp)
- [ULONG](#) [SLQSSetAudioVolTLBConfig](#) ([SetAudioVolTLBConfigReq](#) *pSetAudioVolTLBCfgReq, [SetAudioVolTLBConfigResp](#) *pSetAudioVolTLBCfgResp)

9.9.1 Detailed Description

Audio Service API function prototypes.

9.9.2 Function Documentation

9.9.2.1 [ULONG](#) [SLQSGetAudioPathConfig](#) ([GetAudioPathConfigReq](#) * *pGetAudioPathConfigReq*, [GetAudioPathConfigResp](#) * *pGetAudioPathConfigResp*)

This API gets the audio path configuration parameters.

Parameters

<i>pGetAudioPathConfigReq</i> [IN]	<ul style="list-style-type: none"> • See GetAudioPathConfigReq for more information
<i>pGetAudioPathConfigResp</i> [OUT]	<ul style="list-style-type: none"> • See GetAudioPathConfigResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Device Supported: SL9090
Timeout: 5 seconds

9.9.2.2 ULONG SLQSGetAudioProfile (GetAudioProfileReq * pGetAudioProfileReq, GetAudioProfileResp * pGetAudioProfileResp)

This API get the profile content of the requested audio generator.

Parameters

<i>pGetAudioProfileReq</i> [IN]	<ul style="list-style-type: none"> See GetAudioProfileReq for more information
<i>pGetAudioProfileResp</i> [OUT]	<ul style="list-style-type: none"> See GetAudioProfileResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Device Supported: SL9090
Timeout: 5 seconds

9.9.2.3 ULONG SLQSGetAudioVoTLBConfig (GetAudioVoTLBConfigReq * pGetAudioVoTLBConfigReq, GetAudioVoTLBConfigResp * pGetAudioVoTLBConfigResp)

This API gets the audio path configuration parameters.

Parameters

<i>pGetAudioVol</i> ↔ <i>TLBCfgReq[IN]</i>	<ul style="list-style-type: none">• See GetAudioVolTLBConfigReq for more information
<i>pGetAudioVol</i> ↔ <i>TLBCfgResp</i> ↔ <i>OUT]</i>	<ul style="list-style-type: none">• See GetAudioVolTLBConfigResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Device Supported: SL9090
Timeout: 5 seconds

9.9.2.4 ULONG SLQSSetAudioPathConfig (SetAudioPathConfigReq * pSetAudioPathConfigReq)

This API sets the audio path configuration parameters.

Parameters

<i>pSetAudio</i> ↔ <i>PathConfig</i> ↔ <i>Req[IN]</i>	<ul style="list-style-type: none">• See SetAudioPathConfigReq for more information
---	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Device Supported: SL9090
Timeout: 5 seconds

9.9.2.5 ULONG SLQSSetAudioProfile (SetAudioProfileReq * pSetAudioProfileReq)

This API sets an audio profile.

Parameters

<i>pSetAudioProfileReq</i> [IN]	<ul style="list-style-type: none"> See SetAudioProfileReq for more information
---------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Device Supported: SL9090
Timeout: 5 seconds

9.9.2.6 ULONG SLQSSetAudioVoTLBConfig (SetAudioVoTLBConfigReq * pSetAudioVoTLBCfgReq, SetAudioVoTLBConfigResp * pSetAudioVoTLBCfgResp)

This API sets the audio path configuration parameters.

Parameters

<i>pSetAudioVoTLBCfgReq</i> [IN]	<ul style="list-style-type: none"> See SetAudioVoTLBCfgReq for more information
<i>pSetAudioVoTLBCfgResp</i> [OUT]	<ul style="list-style-type: none"> See SetAudioVoTLBCfgResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Device Supported: SL9090
Timeout: 5 seconds

9.10 qaGobiApiCat.h File Reference

Card Application Toolkit API function headers.

Functions

- [ULONG CATSendEnvelopeCommand](#) ([ULONG](#) cmdID, [ULONG](#) dataLen, [BYTE](#) *pData)
- [ULONG CATSendTerminalResponse](#) ([ULONG](#) refID, [ULONG](#) dataLen, [BYTE](#) *pData)

9.10.1 Detailed Description

Card Application Toolkit API function headers.

9.10.2 Function Documentation

9.10.2.1 [ULONG CATSendEnvelopeCommand](#) ([ULONG](#) cmdID, [ULONG](#) dataLen, [BYTE](#) * pData)

Sends the envelope command to the device.

Parameters

<i>cmdID</i>	<ul style="list-style-type: none">• Envelope command type<ul style="list-style-type: none">– 0x01 - Menu Selection– 0x02 - Event DL User activity– 0x03 - Event DL Idle Screen Available– 0x04 - Event DL Language Selection
<i>dataLen</i>	<ul style="list-style-type: none">• Length of pData in bytes
<i>pData[IN]</i>	<ul style="list-style-type: none">• Encoded envelope data as defined in ETSI TS 102 223, section 7 [Smart Cards: Card Application Toolkit (CAT) – Release 4]

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
Timeout: 2 seconds

9.10.2.2 ULONG CATSendTerminalResponse (ULONG *refID*, ULONG *dataLen*, BYTE * *pData*)

Sends the terminal response to the device.

Parameters

<i>refID</i>	<ul style="list-style-type: none"> Proactive command reference ID. The value should be the same as indicated in the CAT event callback data for the relevant proactive command.
<i>dataLen</i>	<ul style="list-style-type: none"> Terminal response data length
<i>pData</i> [!N]	<ul style="list-style-type: none"> Terminal response for the relevant proactive command encoded as per ETSI TS 102 223, section 6.8 [Smart Cards: Card Application Toolkit (CAT) – Release 4]

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
Timeout: 2 seconds

9.11 qaGobiApiCbK.h File Reference

Callback Service API function prototypes.

Data Structures

- struct [qaQmiInterfaceInfo](#)
- struct [slqsSessionStateInfo](#)
- struct [slqsWdsEventInfo](#)
- struct [TransferStatsDataType](#)
- struct [SignalStrengthDataType](#)
- struct [SMSMTMessage](#)
- struct [SMSTransferRouteMTMessage](#)
- struct [SMSMessageMode](#)
- struct [SMSEtwsMessage](#)
- struct [SMSEtwsPlmn](#)
- struct [SMSCAddress](#)

- struct [SMSONIMS](#)
- struct [SMSEventInfo_s](#)
- struct [CATEventDataType](#)
- struct [ServingSystemInfo](#)
- struct [RoamingInfo](#)
- struct [SLQSSignalStrengthsInformation](#)
- struct [SLQSSignalStrengthsIndReq](#)
- struct [ECTNum](#)
- struct [voiceSUPSNotification](#)
- struct [voiceSetAllCallStatusCbKInfo](#)
- struct [_transLayerInfoNotification](#)
- struct [_transNWRegInfoNotification](#)
- struct [UIMStatusChangeInfo](#)
- struct [voicePrivacyInfo](#)
- struct [voiceDTMFEventInfo](#)
- struct [voiceSUPSInfo](#)
- struct [nasSysInfo](#)
- struct [UniversalTime](#)
- struct [nasNetworkTime](#)
- struct [omaDmFotaTlv](#)
- struct [omaDmFotaTlvExt](#)
- struct [omaDmConfigTlvExt](#)
- struct [omaDmConfigTlv](#)
- struct [omaDmNotificationsTlv](#)
- union [sessionInfo](#)
- union [sessionInfoExt](#)
- struct [SMSMemoryInfo](#)
- struct [voiceOTASPStatusInfo](#)
- struct [signalInfo](#)
- struct [callerIDInfo](#)
- struct [callingPartyInfo](#)
- struct [calledPartyInfo](#)
- struct [redirNumInfo](#)
- struct [NSSAudioCtrl](#)
- struct [lineCtrlInfo](#)
- struct [extDispRecInfo](#)
- struct [voiceInfoRec](#)
- struct [msgWaitingInfo](#)
- struct [QosFlowInfoState](#)
- struct [QosFlowInfo](#)
- struct [QosEventInfo](#)
- struct [nasSigInfo](#)
- struct [_modemTempNotification](#)
- struct [_packetSrvStatus](#)
- struct [HDRPersonalityInd](#)
- struct [imsSIPConfigInfo](#)
- struct [imsRegMgrConfigInfo](#)
- struct [imsSMSConfigInfo](#)
- struct [imsUserConfigInfo](#)
- struct [imsVoIPConfigInfo](#)
- struct [USSDNoWaitIndicationInfo](#)
- struct [channelRate](#)
- struct [DUNCallInfoInd](#)
- struct [SMSAsyncRawSend_s](#)
- struct [LteNasReleaseInfo_s](#)

- struct [SwiOTAMsg_s](#)
- struct [DHCPOption](#)
- struct [DHCPOptionList](#)
- struct [WdsDHCPv4ClientLeaseInd](#)
- struct [QmiCbkLocCradleMountInd](#)
- struct [QmiCbkLocEventTimeSyncInd](#)
- struct [QmiCbkLocInjectTimeInd](#)
- struct [accelAcceptReady_s](#)
- struct [gyroAcceptReady_s](#)
- struct [accelTempAcceptReady_s](#)
- struct [gyroTempAcceptReady_s](#)
- struct [QmiCbkLocSensorStreamingInd](#)
- struct [QmiCbkLocInjectSensorDataInd](#)
- struct [precisionDilution_s](#)
- struct [gpsTime_s](#)
- struct [sensorDataUsage_s](#)
- struct [svUsedforFix_s](#)
- struct [QmiCbkLocPositionReportInd](#)
- struct [t_Sv](#)
- struct [t_sensor](#)
- struct [t_gpsTime](#)
- struct [QmiCbkLocBestAvailPosInd](#)
- struct [imsaRegStatusInfo](#)
- struct [imsaSvcStatusInfo](#)
- struct [imsaRatStatusInfo](#)
- struct [imsaPdpStatusInfo](#)
- struct [satelliteInfo](#)
- struct [gnssSvInfoNotification](#)
- struct [delAssistDataStatus](#)
- struct [QmiCbkNasLTECphyCalInfo](#)
- struct [RankIndicatorInd](#)
- struct [QmiCbkLocInjectUTCTimeInd](#)
- struct [QmiCbkLocInjectPositionInd](#)
- struct [UIMSlotStatusChangeInfo](#)
- struct [QmiCbkLocEngineStateInd](#)
- struct [_getResetInfoNotification](#)
- struct [_MitigationDevInfo](#)
- struct [QmiCbkTmdMitiLvlRptInd](#)
- struct [QmiCbkLocSetExtPowerConfigInd](#)
- struct [nasTimers](#)

Macros

- #define [SIGSTRENGTH_THRESHOLD_ARR_SZ](#) 5
- #define [QMI_WMS_MAX_PAYLOAD_LENGTH](#) 256
- #define [QMI_ETWS_MAX_PAYLOAD_LENGTH](#) 1254 /* Qualcomm defined max */
- #define [QMI_MAX_VOICE_NUMBER_LENGTH](#) 81
- #define [MAX_NO_OF_UUSINFO](#) 20
- #define [MAXUSSDLENGTH](#) 182
- #define [MAX_NO_OF_CALLS](#) 20
- #define [CBK_ENABLE_EVENT](#) 0x01
- #define [CBK_DISABLE_EVENT](#) 0x00
- #define [CBK_NOCHANGE](#) 0xFF
- #define [MAX_NO_OF_APPLICATIONS](#) 10

- #define `MAX_NO_OF_SLOTS` 5
- #define `MAX_NO_OF_FILES` 255
- #define `MAX_PATH_LENGTH` 255
- #define `EVENT_MASK_CARD` 0x00000001
- #define `EVENT_MASK_PHY_SLOT_STATUS` 0x00000010
- #define `EVENT_MASK_DEREGISTER_ALL` 0x00000000
- #define `REGISTER_EVENT` 0x01
- #define `DEREGISTER_EVENT` 0x00
- #define `FIRST_INSTANCE` 0x00
- #define `SECOND_INSTANCE` 0x01
- #define `THIRD_INSTANCE` 0x02
- #define `INVALID_INSTACNE` 0x08
- #define `REGISTER_SRV` 0x01
- #define `DEREGISTER_SRV` 0x00
- #define `WDS_SRV` 0x01
- #define `NAS_SRV` 0x02
- #define `PDS_SRV` 0x04
- #define `VOICE_SRV` 0x08
- #define `NUM_OF_SET` 0xFF
- #define `IPV4` 4
- #define `IPV6` 6
- #define `IPV4V6` 7
- #define `LOC_EVENT_MASK_ENG_STATE` 0x00000080
- #define `LOC_EVENT_MASK_TIME_SYNC` 0x00000800
- #define `LOC_EVENT_MASK_INJECT_TIME` 0x00000010
- #define `LOC_EVENT_MASK_SENSOR_STREAM` 0x00000400
- #define `LOC_EVENT_POSITION_REPORT` 0x00000001
- #define `LOC_EVENT_MASK_GNSS_SV_INFO` 0x00000002
- #define `DHCP_MAX_NUM_OPTIONS` 30
- #define `DHCP_OPTION_DATA_BUF_SIZE` 2048 /* current max size of raw message in SDK process is 2048 */
- #define `MAX_MITIGATION_DEV_ID_LEN` 255
- #define `MAX_RADIO_INTERFACE_LIST` 255
- #define `USSD_DCS_ASCII` 0x01 /* ASCII coding scheme */
- #define `USSD_DCS_8BIT` 0x02 /* 8-bit coding scheme */
- #define `USSD_DCS_UCS2` 0x03 /* UCS2 coding scheme */

Typedefs

- typedef void(* `tFNSLQSSessionState`) (`slqsSessionStateInfo` *pSessionStateInfo)
- typedef void(* `tFNSLQSWDSEvent`) (`slqsWdsEventInfo` *pWdsEventInfo)
- typedef void(* `tFNPower`) (`ULONG` operatingMode)
- typedef void(* `tFNActivationStatus`) (`ULONG` activationStatus)
- typedef void(* `tFNMobileIPStatus`) (`ULONG` mipStatus)
- typedef void(* `tFNRoamingIndicator`) (`ULONG` roaming)
- typedef void(* `tFNDataCapabilities`) (`BYTE` dataCapsSize, `BYTE` *pDataCaps)
- typedef void(* `tFNSignalStrength`) (`INT8` signalStrength, `ULONG` radioInterface)
- typedef void(* `tFNRInfo`) (`ULONG` radioInterface, `ULONG` activeBandClass, `ULONG` activeChannel)
- typedef void(* `tFNLURreject`) (`ULONG` serviceDomain, `ULONG` rejectCause)
- typedef void(* `tFNNewSMS`) (`ULONG` storageType, `ULONG` messageIndex)
- typedef enum `SMSEventType` `eSMSEventType`
- typedef struct `SMSMTMessage` `SMSMTMessageInfo`
- typedef struct `SMSTTransferRouteMTMessage` `SMSTTransferRouteMTMessageInfo`
- typedef struct `SMSMessageMode` `SMSMessageModeInfo`

- typedef struct [SMSEtwMessage](#) [SMSEtwMessageInfo](#)
- typedef struct [SMSEtwPlmn](#) [SMSEtwPlmnInfo](#)
- typedef struct [SMSCAddress](#) [SMSCAddressInfo](#)
- typedef struct [SMSONIMS](#) [SMSONIMSInfo](#)
- typedef struct [SMSEventInfo_s](#) [SMSEventInfo](#)
- typedef void(* [tFNSMSEvents](#)) ([SMSEventInfo](#) *pSMSEventInfo)
- typedef void(* [tFNNewNMEA](#)) ([LPCSTR](#) pNMEA)
- typedef void(* [tFNPDSState](#)) ([ULONG](#) enabledStatus, [ULONG](#) trackingStatus)
- typedef void(* [tFNCATEvent](#)) ([ULONG](#) eventID, [ULONG](#) eventLen, [BYTE](#) *pEventData)
- typedef enum [device_state_enum](#) [eDevState](#)
- typedef void(* [tFNDeviceStateChange](#)) ([eDevState](#) device_state)
- typedef void(* [tFNNet](#)) ([ULONG](#) q_depth, [BYTE](#) isThrottle, [BYTE](#) instanceId)
- typedef void(* [tFNFwdIdCompletion](#)) ([ULONG](#) fwdId_completion_status)
- typedef void(* [tFNSLQSOMADMAAlert](#)) ([ULONG](#) eventType, [BYTE](#) *pEventFields)
- typedef void(* [tFNOMADMState](#)) ([ULONG](#) sessionState, [ULONG](#) failureReason)
- typedef void(* [tFNServingSystem](#)) (struct [ServingSystemInfo](#) *pServingSystem, struct [RoamingInfo](#) *pRoamingInfo)
- typedef void(* [tFNBandPreference](#)) ([ULONGLONG](#) band_pref)
- typedef void(* [tFNUSSDRelease](#)) (void)
- typedef void(* [tFNUSSDNotification](#)) ([ULONG](#) type, [BYTE](#) *pNetworkInfo)
- typedef void(* [tFNSLQSSignalStrengths](#)) (struct [SLQSSignalStrengthsInformation](#) sSLQSSignalStrengthsInfo)
- typedef void(* [tFNSUPSNotification](#)) ([voiceSUPSNotification](#) *pVoiceSUPSNotification)
- typedef void(* [tFNSDKTerminated](#)) ([BYTE](#) *psReason)
- typedef void(* [tFNAllCallStatus](#)) ([voiceSetAllCallStatusCbkJInfo](#) *pVoiceSetAllCallStatusCbkJInfo)
- typedef struct [_transLayerInfoNotification](#) [transLayerNotification](#)
- typedef void(* [tFNtransLayerInfo](#)) ([transLayerNotification](#) *pTransLayerNotification)
- typedef struct [_transNWRegInfoNotification](#) [transNWRegInfoNotification](#)
- typedef void(* [tFNtransNWRegInfo](#)) ([transNWRegInfoNotification](#) *pTransNWRegInfoNotification)
- typedef void(* [tFNSysSelectionPref](#)) ([sysSelectPrefInfo](#) *pSysSelectPrefInfo)
- typedef void(* [tFNUIMRefresh](#)) ([UIMRefreshEvent](#) *pUIMRefreshEvent)
- typedef void(* [tFNUIMStatusChangeInfo](#)) ([UIMStatusChangeInfo](#) *pUIMStatusChangeInfo)
- typedef void(* [tFNPrivacyChange](#)) ([voicePrivacyInfo](#) *pVoicePrivacyInfo)
- typedef void(* [tFNDTMFEvent](#)) ([voiceDTMFEventInfo](#) *pVoiceDTMFEventInfo)
- typedef void(* [tFNSUPSInfo](#)) ([voiceSUPSInfo](#) *pVoiceSUPSInfo)
- typedef void(* [tFNSysInfo](#)) ([nasSysInfo](#) *pNasSysInfo)
- typedef void(* [tFNNetworkTime](#)) ([nasNetworkTime](#) *pNasNetworkTime)
- typedef union [sessionInfo](#) [sessionInformation](#)
- typedef union [sessionInfoExt](#) [sessionInformationExt](#)
- typedef void(* [tFNMemoryFull](#)) ([SMSMemoryInfo](#) *pSMSMemoryFullInfo)
- typedef void(* [tFNOTASPStatus](#)) ([voiceOTASPStatusInfo](#) *pVoiceOTASPStatusInfo)
- typedef void(* [tFNInfoRec](#)) ([voiceInfoRec](#) *pVoiceInfoRec)
- typedef void(* [tFNMessageWaiting](#)) ([msgWaitingInfo](#) *pSMSMessageWaitingInfo)
- typedef void(* [tFNSLQSQOSEvent](#)) ([BYTE](#) instance, [QosFlowInfo](#) *pFlowInfo)
- typedef void(* [tFNQosStatus](#)) ([BYTE](#) instance, [ULONG](#) id, [BYTE](#) status, [BYTE](#) event, [BYTE](#) reason)
- typedef void(* [tFNQosNWStatus](#)) ([BYTE](#) status)
- typedef void(* [tFNQosPriEvent](#)) ([WORD](#) event)
- typedef void(* [tFNSigInfo](#)) ([nasSigInfo](#) *pNasSigInfo)
- typedef struct [_modemTempNotification](#) [modemTempNotification](#)
- typedef void(* [tFNModemTempInfo](#)) ([modemTempNotification](#) *pModemTempNotification)
- typedef struct [_packetSrvStatus](#) [packetSrvStatus](#)
- typedef void(* [tFNPacketSrvState](#)) ([packetSrvStatus](#) *pPacketSrvStatus)
- typedef void(* [tFNHDRPersonality](#)) ([HDRPersonalityInd](#) *pHDRPers)
- typedef void(* [tFNImSIPConfig](#)) ([imsSIPConfigInfo](#) *pImSIPConfigInfo)
- typedef void(* [tFNImSRegMgrConfig](#)) ([imsRegMgrConfigInfo](#) *pImSRegMgrConfigInfo)

- typedef void(* tFNImSMSConfig) (imsSMSConfigInfo *pImSMSConfigInfo)
- typedef void(* tFNImUserConfig) (imsUserConfigInfo *pImUserConfigInfo)
- typedef void(* tFNImVoIPConfig) (imsVoIPConfigInfo *pImVoIPConfigInfo)
- typedef void(* tFNUSSDNoWaitIndication) (USSDNoWaitIndicationInfo *pNetworkInfo)
- typedef void(* tFNDUNCAllInfo) (DUNCAllInfoInd *pDUNCAllInfo)
- typedef void(* tFNDataSysStatus) (CurrDataSysStat *pCurrDataSysStat)
- typedef struct SMSAsyncRawSend_s SMSAsyncRawSend
- typedef void(* tFNAsyncRawSend) (SMSAsyncRawSend *pSMSAsyncRawSend)
- typedef struct LteNasReleaseInfo_s LteNasReleaseInfo
- typedef struct SwiOTAMsg_s SwiOTAMsg
- typedef void(* tFNASwiOTAMsg) (SwiOTAMsg *pSwiOTAMsg)
- typedef void(* tFNNewGPS) (double dLongitude, double dLatitude, BYTE session_status, ULONG pos_src)
- typedef void(* tFNNewRMTransferStatistics) (QmiCbkWdsStatisticsIndState *pMsg)
- typedef void(* tFNDHCPv4ClientLeaseStatus) (BYTE instance, WdsDHCPv4ClientLeaseInd *pMsg)
- typedef void(* tFNSetCradleMount) (QmiCbkLocCradleMountInd *pSetLocCradleMount)
- typedef void(* tFNSetEventTimeSync) (QmiCbkLocEventTimeSyncInd *pSetLocEventTimeSync)
- typedef void(* tFNInjectTimeStatus) (QmiCbkLocInjectTimeInd *pLocInjectTime)
- typedef struct accelAcceptReady_s accelAcceptReady
- typedef struct gyroAcceptReady_s gyroAcceptReady
- typedef struct accelTempAcceptReady_s accelTempAcceptReady
- typedef struct gyroTempAcceptReady_s gyroTempAcceptReady
- typedef void(* tFNSensorStreaming) (QmiCbkLocSensorStreamingInd *pLocSensorStream)
- typedef void(* tFNInjectSensorData) (QmiCbkLocInjectSensorDataInd *pLocInjectSensorData)
- typedef struct precisionDilution_s precisionDilution
- typedef struct gpsTime_s gpsTime
- typedef struct sensorDataUsage_s sensorDataUsage
- typedef struct svUsedforFix_s svUsedforFix
- typedef void(* tFNEventPosition) (QmiCbkLocPositionReportInd *pLocPositionReport)
- typedef void(* tFNBestAvailPos) (QmiCbkLocBestAvailPosInd *pBestAvailPos)
- typedef void(* tFNOpMode) (ULONG mode)
- typedef void(* tFNImsaRegStatus) (imsaRegStatusInfo *pImsaRegStatusInfo)
- typedef void(* tFNImsaSvcStatus) (imsaSvcStatusInfo *pImsaSvcStatusInfo)
- typedef void(* tFNImsaRatStatus) (imsaRatStatusInfo *pImsaRatStatusInfo)
- typedef void(* tFNImsaPdpStatus) (imsaPdpStatusInfo *pImsaPdpStatusInfo)
- typedef void(* tFNGnssSvInfo) (gnssSvInfoNotification *pGnssSvInfoNotification)
- typedef void(* tFNDeIAssistData) (delAssistDataStatus *pAssistDataNotification)
- typedef void(* tFNASwiLTECphyCallInfo) (QmiCbkNasLTECphyCalInfo *pQmiCbkNasLTECphyCalInfo)
- typedef void(* tFNRankIndicator) (RankIndicatorInd *pRankIndicatorInd)
- typedef void(* tFNInjectUTCTime) (QmiCbkLocInjectUTCTimeInd *pInjectUTCTimeNotification)
- typedef void(* tFNInjectPosition) (QmiCbkLocInjectPositionInd *pInjectPositionNotification)
- typedef void(* tFNCbkUimSlotStatusChangeInd) (UIMSlotStatusChangeInfo *pQmiCbkUimSlotStatusChangeInd)
- typedef void(* tFNSetEngineState) (QmiCbkLocEngineStateInd *pSetLocEngineState)
- typedef struct _getResetInfoNotification ResetInfoNotification
- typedef void(* tFNResetInfo) (ResetInfoNotification *pResetInfoNotification)
- typedef void(* tFNMTiLvIRpt) (QmiCbkTmdMTiLvIRptInd *pSetLocCradleMount)
- typedef void(* tFNSetExtPowerConfig) (QmiCbkLocSetExtPowerConfigInd *pSetExtConfigIndStatus)
- typedef void(* tFNNasTimer) (nasTimers *timers)

Enumerations

- enum `eQaQMIService` {
`eQA_QMI_SVC_WDS` = 0x01,
`eQA_QMI_SVC_NAS` = 0x03,
`eQA_QMI_SVC_NA` = 0xFF }
- enum `SMSEventType` {
`SMS_EVENT_MT_MESSAGE` = 0x01,
`SMS_EVENT_TRANSFER_ROUTE_MT_MESSAGE` = 0x02,
`SMS_EVENT_MESSAGE_MODE` = 0x04,
`SMS_EVENT_ETWS` = 0x08,
`SMS_EVENT_ETWS_PLMN` = 0x10,
`SMS_EVENT_SMSC_ADDRESS` = 0x20,
`SMS_EVENT_SMS_ON_IMS` = 0x40 }
- enum `device_state_enum` {
`DEVICE_STATE_DISCONNECTED`,
`DEVICE_STATE_READY`,
`DEVICE_STATE_BOOT` }

Functions

- `ULONG SLQSSetSessionStateCallback` (tFNSLQSSessionState pCallback)
- `ULONG SLQSSetWdsEventCallback` (tFNSLQSWDSEvent pCallback, `BYTE` interval, `BYTE` instanceid, `BYTE` ipfamily)
- `ULONG SLQSSetWdsTransferStatisticCallback` (tFNSLQSWDSEvent pXferStatsCb, `BYTE` interval, `BYTE` instanceid, `BYTE` ipfamily)
- `ULONG iSLQSSetWdsFirstInstEventCallback` (tFNSLQSWDSEvent pCallback)
- `ULONG iSLQSSetWdsSecondInstEventCallback` (tFNSLQSWDSEvent pCallback)
- `ULONG iSLQSSetWdsThirdInstEventCallback` (tFNSLQSWDSEvent pCallback)
- `ULONG iSLQSSetWdsXferStatsFirstInstCallback` (tFNSLQSWDSEvent pCallback)
- `ULONG iSLQSSetWdsXferStatsSecondInstCallback` (tFNSLQSWDSEvent pCallback)
- `ULONG SetPowerCallback` (tFNPower pCallback)
- `ULONG SetActivationStatusCallback` (tFNActivationStatus pCallback)
- `ULONG SetMobileIPStatusCallback` (tFNMobileIPStatus pCallback)
- `ULONG SetRoamingIndicatorCallback` (tFNRoamingIndicator pCallback)
- `ULONG SetDataCapabilitiesCallback` (tFNDataCapabilities pCallback)
- `ULONG SetSignalStrengthCallback` (tFNSignalStrength pCallback, `BYTE` thresholdsSize, `INT8` *pThresholds)
- `ULONG iSetSignalStrengthCallback` (tFNSignalStrength pCallback)
- `ULONG SetRFInfoCallback` (tFNRInfo pCallback)
- `ULONG SetLURRejectCallback` (tFNLURReject pCallback)
- `ULONG SetNewSMSCallback` (tFNNewSMS pCallback)
- `ULONG SLQSSetSMSEventCallback` (tFNSMSEvents pCallback)
- `ULONG SetNMEACallback` (tFNNewNMEA pCallback)
- `ULONG SetPDSSStateCallback` (tFNPDSState pCallback)
- `ULONG SetCATEventCallback` (tFNCATEvent pCallback, `ULONG` eventMask, `ULONG` *pErrorMask)
- `ULONG iSetCATEventCallback` (tFNCATEvent pCallback)
- `ULONG SetDeviceStateChangeCb` (tFNDeviceStateChange pCallback)
- `ULONG SetNetChangeCb` (`BYTE` instance, tFNNet pCallback, `ULONG` loMark, `ULONG` hiMark, `ULONG` period)
- `ULONG SetFwDidCompletionCb` (tFNFwDidCompletion pCallback)
- `ULONG SetSLQSOMADMAAlertCallback` (tFNSLQSOMADMAAlert pCallback)
- `ULONG SetSLQSOMADMAAlertCallbackExt` (tFNSLQSOMADMAAlert pCallback)
- `ULONG SetOMADMStateCallback` (tFNOMADMState pCallback)
- `ULONG SLQSSetServingSystemCallback` (tFNServingSystem pCallback)

- [ULONG SLQSSetBandPreferenceCbK](#) ([tFNBandPreference](#) pCallback)
- [ULONG SetUSSDReleaseCallback](#) ([tFNUSSDRelease](#) pCallback)
- [ULONG SetUSSDNotificationCallback](#) ([tFNUSSDNotification](#) pCallback)
- [ULONG SLQSSetSignalStrengthsCallback](#) ([tFNSLQSSignalStrengths](#) pCallback, struct [SLQSSignalStrengthsIndReq](#) *pSLQSSignalStrengthsIndReq)
- [ULONG iSLQSSetSignalStrengthsCallback](#) ([tFNSLQSSignalStrengths](#) pCallback)
- [ULONG SLQSVoiceSetSUPSNotificationCallback](#) ([tFNSUPSNotification](#) pCallback)
- [ULONG SLQSSetSDKTerminatedCallback](#) ([tFNSDKTerminated](#) pCallback)
- [ULONG SLQSVoiceSetAllCallStatusCallBack](#) ([tFNAIICallStatus](#) pCallback)
- [ULONG SLQSSetTransLayerInfoCallback](#) ([tFNtransLayerInfo](#) pCallback)
- [ULONG SLQSSetTransNWRegInfoCallback](#) ([tFNtransNWRegInfo](#) pCallback)
- [ULONG SLQSSetSysSelectionPrefCallBack](#) ([tFNSysSelectionPref](#) pCallback)
- [ULONG SLQSUIIMSetRefreshCallBack](#) ([tFNUIMRefresh](#) pCallback)
- [ULONG SLQSUIIMSetStatusChangeCallBack](#) ([tFNUIMStatusChangeInfo](#) pCallback)
- [ULONG SLQSVoiceSetPrivacyChangeCallBack](#) ([tFNPrivacyChange](#) pCallback)
- [ULONG SLQSVoiceSetDTMFEventCallBack](#) ([tFNDTMFEvent](#) pCallback)
- [ULONG SLQSVoiceSetSUPSCallBack](#) ([tFNSUPSInfo](#) pCallback)
- [ULONG SLQSNasSysInfoCallBack](#) ([tFNSysInfo](#) pCallback)
- [ULONG SLQSNasNetworkTimeCallBack](#) ([tFNNetworkTime](#) pCallback)
- [ULONG SLQSWmsMemoryFullCallBack](#) ([tFNMemoryFull](#) pCallback)
- [ULONG SLQSVoiceSetOTASPStatusCallBack](#) ([tFNOTASPStatus](#) pCallback)
- [ULONG SLQSVoiceInfoRecCallBack](#) ([tFNInfoRec](#) pCallback)
- [ULONG SLQSWmsMessageWaitingCallBack](#) ([tFNMessageWaiting](#) pCallback)
- [ULONG SLQSSetQosEventCallback](#) ([BYTE](#) instance, [tFNSLQSQOSEvent](#) pCallback)
- [ULONG SLQSSetQosStatusCallback](#) ([BYTE](#) instance, [tFNQosStatus](#) pCallback)
- [ULONG SLQSSetQosNWStatusCallback](#) ([tFNQosNWStatus](#) pCallback)
- [ULONG SLQSSetQosPriEventCallback](#) ([tFNQosPriEvent](#) pCallback)
- [ULONG SLQSNasSigInfoCallBack](#) ([tFNSigInfo](#) pCallback, [sigInfo](#) *pSigInfo)
- [ULONG SLQSSetModemTempCallback](#) ([tFNModemTempInfo](#) pCallback)
- [ULONG SLQSSetPacketSrvStatusCallback](#) ([tFNPacketSrvState](#) pCallback)
- [ULONG SLQSSetSwtHDPersCallback](#) ([tFNHDPersonaity](#) pCallback)
- [ULONG SLQSSetSIPConfigCallback](#) ([tFNImSIPConfig](#) pCallback)
- [ULONG SLQSSetRegMgrConfigCallback](#) ([tFNImRegMgrConfig](#) pCallback)
- [ULONG SLQSSetIMSSMSConfigCallback](#) ([tFNImSMSConfig](#) pCallback)
- [ULONG SLQSSetIMSUserConfigCallback](#) ([tFNImUserConfig](#) pCallback)
- [ULONG SLQSSetIMSVoIPConfigCallback](#) ([tFNImVoIPConfig](#) pCallback)
- [ULONG SetUSSDNoWaitIndicationCallback](#) ([tFNUSSDNoWaitIndication](#) pCallback)
- [ULONG SLQSSetDUNCallInfoCallback](#) ([BYTE](#) StatsPeriod, [tFNDUNCallInfo](#) pCallback)
- [ULONG iSLQSSetDUNCallInfoCallback](#) ([tFNDUNCallInfo](#) pCallback)
- [ULONG SLQSSetDataSystemStatusCallback](#) ([tFNDataSysStatus](#) pCallback)
- [ULONG SLQSWmsAsyncRawSendCallBack](#) ([tFNAsyncRawSend](#) pCallback)
- [ULONG SLQSNasSwtOTAMessageCallback](#) ([NasSwtIndReq](#) *req, [tFNASwtOTAMsg](#) pCallback)
- [ULONG SetGPSCallback](#) ([tFNNewGPS](#) pCallback)
- [ULONG SetRMTransferStatisticsCallback](#) ([tFNNewRMTransferStatistics](#) pCallback)
- [ULONG SLQSSetDHCPv4ClientLeaseStatusCallback](#) ([BYTE](#) instance, [tFNDHCPv4ClientLeaseStatus](#) pCallback)
- [ULONG SetLocCradleMountCallback](#) ([tFNSetCradleMount](#) pCallback)
- [ULONG SetLocEventTimeSyncCallback](#) ([tFNSetEventTimeSync](#) pCallback)
- [ULONG SetLocInjectTimeCallback](#) ([tFNInjectTimeStatus](#) pCallback)
- [ULONG SetLocSensorStreamingCallback](#) ([tFNSensorStreaming](#) pCallback)
- [ULONG SetLocInjectSensorDataCallback](#) ([tFNInjectSensorData](#) pCallback)
- [ULONG SetLocEventPositionCallback](#) ([tFNEventPosition](#) pCallback)
- [ULONG SetLocOpModeCallback](#) ([tFNOpMode](#) pCallback)
- [ULONG SLQSSetIMSAREgStatusCallback](#) ([tFNImsaRegStatus](#) pCallback)
- [ULONG SLQSSetIMSASvcStatusCallback](#) ([tFNImsaSvcStatus](#) pCallback)

- [ULONG SLQSSetIMSAStatusCallback](#) (tFNImsaRatStatus pCallback)
- [ULONG SLQSSetIMSApdpStatusCallback](#) (tFNImsaPdpStatus pCallback)
- [ULONG SLQSNasSigInfo2CallBack](#) (tFNSigInfo pCallback, [setSignalStrengthInfo](#) *pSigInfo2)
- [ULONG SetLocGnssSvInfoCallback](#) (tFNGnssSvInfo pCallback)
- [ULONG SetLocDeleteAssistDataCallback](#) (tFNDeIAssistData pCallback)
- [ULONG SetNasLTECphyCaIndCallback](#) (tFNASwiLTECphyCalInfo pCallback)
- [ULONG SetRankIndicatorCallback](#) (tFNRankIndicator pCallback)
- [ULONG SLQSSetLocInjectUTCTimeCallback](#) (tFNInjectUTCTime pCallback)
- [ULONG SLQSSetLocInjectPositionCallback](#) (tFNInjectPosition pCallback)
- [ULONG SetUimSlotStatusChangeCallback](#) (tFNCbkUimSlotStatusChangeInd pCallback)
- [ULONG SetLocEngineStateCallback](#) (tFNSetEngineState pCallback)
- [ULONG SLQSSetSwiGetResetInfoCallback](#) (tFNResetInfo pCallback)
- [ULONG SLQSTmdMitigationLvIRptCallback](#) (TmdMitigationLvIIndReq *req, tFNMitIvIRpt pCallback)
- [ULONG SetLocSetExtPowerConfigCallback](#) (tFNSetExtPowerConfig pCallback)
- [ULONG SetLocBestAvailPosCallback](#) (tFNBestAvailPos pCallback)
- [ULONG SLQSNasTimerCallback](#) (tFNNasTimer pCallback)

9.11.1 Detailed Description

Callback Service API function prototypes.

9.11.2 Macro Definition Documentation

9.11.2.1 `#define CBK_DISABLE_EVENT 0x00`

9.11.2.2 `#define CBK_ENABLE_EVENT 0x01`

9.11.2.3 `#define CBK_NOCHANGE 0xFF`

9.11.2.4 `#define DEREGISTER_EVENT 0x00`

9.11.2.5 `#define DEREGISTER_SRV 0x00`

9.11.2.6 `#define DHCP_MAX_NUM_OPTIONS 30`

9.11.2.7 `#define DHCP_OPTION_DATA_BUF_SIZE 2048 /* current max size of raw message in SDK process is 2048 */`

9.11.2.8 `#define EVENT_MASK_CARD 0x00000001`

9.11.2.9 `#define EVENT_MASK_DEREGISTER_ALL 0x00000000`

9.11.2.10 `#define EVENT_MASK_PHY_SLOT_STATUS 0x00000010`

9.11.2.11 `#define FIRST_INSTANCE 0x00`

9.11.2.12 `#define INVALID_INSTACNE 0x08`

- 9.11.2.13 `#define IPV4 4`
- 9.11.2.14 `#define IPV4V6 7`
- 9.11.2.15 `#define IPV6 6`
- 9.11.2.16 `#define LOC_EVENT_MASK_ENG_STATE 0x00000080`
- 9.11.2.17 `#define LOC_EVENT_MASK_GNSS_SV_INFO 0x00000002`
- 9.11.2.18 `#define LOC_EVENT_MASK_INJECT_TIME 0x00000010`
- 9.11.2.19 `#define LOC_EVENT_MASK_SENSOR_STREAM 0x00000400`
- 9.11.2.20 `#define LOC_EVENT_MASK_TIME_SYNC 0x00000800`
- 9.11.2.21 `#define LOC_EVENT_POSITION_REPORT 0x00000001`
- 9.11.2.22 `#define MAX_MITIGATION_DEV_ID_LEN 255`
- 9.11.2.23 `#define MAX_NO_OF_APPLICATIONS 10`
- 9.11.2.24 `#define MAX_NO_OF_CALLS 20`
- 9.11.2.25 `#define MAX_NO_OF_FILES 255`
- 9.11.2.26 `#define MAX_NO_OF_SLOTS 5`
- 9.11.2.27 `#define MAX_NO_OF_UUSINFO 20`
- 9.11.2.28 `#define MAX_PATH_LENGTH 255`
- 9.11.2.29 `#define MAX_RADIO_INTERFACE_LIST 255`
- 9.11.2.30 `#define MAXUSSDLENGTH 182`
- 9.11.2.31 `#define NAS_SRV 0x02`
- 9.11.2.32 `#define NUM_OF_SET 0xFF`
- 9.11.2.33 `#define PDS_SRV 0x04`
- 9.11.2.34 `#define QMI_ETWS_MAX_PAYLOAD_LENGTH 1254 /* Qualcomm defined max */`
- 9.11.2.35 `#define QMI_MAX_VOICE_NUMBER_LENGTH 81`

9.11.2.36 `#define QMI_WMS_MAX_PAYLOAD_LENGTH 256`

9.11.2.37 `#define REGISTER_EVENT 0x01`

9.11.2.38 `#define REGISTER_SRV 0x01`

9.11.2.39 `#define SECOND_INSTANCE 0x01`

9.11.2.40 `#define SIGSTRENGTH_THRESHOLD_ARR_SZ 5`

9.11.2.41 `#define THIRD_INSTANCE 0x02`

9.11.2.42 `#define USSD_DCS_8BIT 0x02 /* 8-bit coding scheme */`

9.11.2.43 `#define USSD_DCS_ASCII 0x01 /* ASCII coding scheme */`

9.11.2.44 `#define USSD_DCS_UCS2 0x03 /* UCS2 coding scheme */`

9.11.2.45 `#define VOICE_SRV 0x08`

9.11.2.46 `#define WDS_SRV 0x01`

9.11.3 Typedef Documentation

9.11.3.1 typedef struct accelAcceptReady_s accelAcceptReady

This structure contains Accelerometer Accept Ready Info

Parameters

<i>injectEnable</i>	<ul style="list-style-type: none"> GNSS location engine is ready to accept data from sensor. Values 0x01 - Ready to accept sensor data 0x00 - Not ready to accept sensor data
<i>samplesPerBatch</i>	<ul style="list-style-type: none"> number of samples per batch the GNSS location engine is to receive. samplingFrequency = samplesPerBatch * batchesPerSecond samplesPerBatch must be a nonzero positive value.
<i>batchesPerSec</i>	<ul style="list-style-type: none"> LTE NAS version minor Number of sensor-data batches the GNSS location engine is to receive per second. BatchesPerSecond must be a nonzero positive value.

9.11.3.2 typedef struct accelTempAcceptReady_s accelTempAcceptReady

This structure contains Accelerometer Temperature Accept Ready Info

Parameters

<i>injectEnable</i>	<ul style="list-style-type: none"> GNSS location engine is ready to accept data from sensor. Values 0x01 - Ready to accept sensor data 0x00 - Not ready to accept sensor data
<i>samplesPerBatch</i>	<ul style="list-style-type: none"> number of samples per batch the GNSS location engine is to receive. samplingFrequency = samplesPerBatch * batchesPerSecond samplesPerBatch must be a nonzero positive value.
<i>batchPerSec</i>	<ul style="list-style-type: none"> LTE NAS version minor Number of sensor-data batches the GNSS location engine is to receive per second. BatchesPerSecond must be a nonzero positive value.

9.11.3.3 typedef enum device_state_enum eDevState

Device State enumeration

- See [device_state_enum](#) for more details

9.11.3.4 typedef enum SMSEventType eSMSEventType

This enumeration defines the different type of SMS events that are received

- See [SMSEventType](#) for more details

9.11.3.5 typedef struct gpsTime_s gpsTime

This structure contains GPS Time info.

Parameters

<i>gpsWeek</i>	<ul style="list-style-type: none"> Current GPS week as calculated from midnight, Jan. 6, 1980. Units - Weeks
<i>gpsTimeOfWeekMs</i>	<ul style="list-style-type: none"> Amount of time into the current GPS week.
Generated by Doxygen	<ul style="list-style-type: none"> Units - Milliseconds

9.11.3.6 typedef struct gyroAcceptReady_s gyroAcceptReady

This structure contains Gyroscope Accept Ready Info

Parameters

<i>injectEnable</i>	<ul style="list-style-type: none"> GNSS location engine is ready to accept data from sensor. Values 0x01 - Ready to accept sensor data 0x00 - Not ready to accept sensor data
<i>samplesPer↔ Batch</i>	<ul style="list-style-type: none"> number of samples per batch the GNSS location engine is to receive. samplingFrequency = samplesPerBatch * batchesPerSecond samplesPerBatch must be a nonzero positive value.
<i>batchPerSec</i>	<ul style="list-style-type: none"> LTE NAS version minor Number of sensor-data batches the GNSS location engine is to receive per second. BatchesPerSecond must be a nonzero positive value.

9.11.3.7 typedef struct gyroTempAcceptReady_s gyroTempAcceptReady

This structure contains Gyroscope Temperature Accept Ready Info

Parameters

<i>injectEnable</i>	<ul style="list-style-type: none"> GNSS location engine is ready to accept data from sensor. Values 0x01 - Ready to accept sensor data 0x00 - Not ready to accept sensor data
<i>samplesPer↔ Batch</i>	<ul style="list-style-type: none"> number of samples per batch the GNSS location engine is to receive. samplingFrequency = samplesPerBatch * batchesPerSecond samplesPerBatch must be a nonzero positive value.
<i>batchPerSec</i>	<ul style="list-style-type: none"> LTE NAS version minor Number of sensor-data batches the GNSS location engine is to receive per second. BatchesPerSecond must be a nonzero positive value.

9.11.3.8 typedef struct LteNasReleaseInfo_s LteNasReleaseInfo

This structure contains LTE Nas Release Information

Parameters

<i>nas_release</i>	<ul style="list-style-type: none"> • LTE NAS release
<i>nas_major</i>	<ul style="list-style-type: none"> • LTE NAS version major
<i>nas_minor</i>	<ul style="list-style-type: none"> • LTE NAS version minor

9.11.3.9 typedef struct _modemTempNotification modemTempNotification

Contains the parameters passed for SLQSSetModemTempCallback by the device.

Parameters

<i>ModemTemp↔ State</i>	<ul style="list-style-type: none"> • provides the temperature state of the modem • Values: <ul style="list-style-type: none"> – 0 - unknown – 1 - normal – 2 - high(warning) – 3 - high(critical) – 4 - low(critical)
<i>Modem↔ Temperature</i>	<ul style="list-style-type: none"> • provides the temperature of the modem

Note

None

9.11.3.10 typedef struct _packetSrvStatus packetSrvStatus

Contains the parameters passed for SLQSSetPacketSrvStatusCallback by the device.

Parameters

<i>pQmilInterface↔ Info</i>	<ul style="list-style-type: none"> • See qaQmilInterfaceInfo for more information
<i>connStatus</i>	<ul style="list-style-type: none"> • Current Link Status <ul style="list-style-type: none"> – 1 - Disconnected – 2 - Connected – 3 - Suspended – 4 - Authenticating

<i>reconfigReqd</i>	<ul style="list-style-type: none"> Indicates if the network interface on the host needs to be reconfigured <ul style="list-style-type: none"> 0 - No need to reconfigure 1 - Reconfiguration required
<i>sessionEndReason</i>	<ul style="list-style-type: none"> See qaGobiApiTableCallEndReasons.h for Call End Reason, 0xFFFF means invalid value
<i>verboseSessnEndReasonType</i>	<ul style="list-style-type: none"> Call End Reason Type <ul style="list-style-type: none"> 0 - Unspecified 1 - Mobile IP 2 - Internal 3 - Call Manager defined 6 - 3GPP Specification defined 7 - PPP 8 - EHRPD 9 - IPv6 0xFFFF - invalid value
<i>verboseSessnEndReason</i>	<ul style="list-style-type: none"> See qaGobiApiTableCallEndReasons.h for verbose Call End Reason. The values depend on verboseSessnEndReasonType parameter 0xFFFF means invalid value
<i>ipFamily</i>	<ul style="list-style-type: none"> IP Family of the packet data connection <ul style="list-style-type: none"> 4 - IPv4 6 - IPv6 0xFF - invalid value
<i>techName</i>	<ul style="list-style-type: none"> Technology name of the packet data connection. <ul style="list-style-type: none"> 32767 - CDMA 32764 - UMTS 30592 - EPC 30590 - EMBMS 30584 - Modem Link Local 0xFFFF - invalid value EPC is a logical interface to support LTE/eHRPD handoff. Modem Link is an interface for transferring data between entities on the AP and modem.
<i>bearerID</i>	<ul style="list-style-type: none"> Bearer ID (3GPP) or RLP ID (3GPP2) of the packet data connection 0xFF means invalid value

Note

Any parameter not returned by the device is returned as its maximum unsigned value by the callback.

9.11.3.11 typedef struct precisionDilution_s precisionDilution

This structure contains Dilution of precision associated with this position.

Parameters

<i>PDOP</i>	<ul style="list-style-type: none"> • Position dilution of precision. • Range - 1 (highest accuracy) to 50 (lowest accuracy) • PDOP = square root of (Square of HDOP + Square of VDOP2)
<i>HDOP</i>	<ul style="list-style-type: none"> • Horizontal dilution of precision. • Range - 1 (highest accuracy) to 50 (lowest accuracy)
<i>VDOP</i>	<ul style="list-style-type: none"> • Vertical dilution of precision. • Range- 1 (highest accuracy) to 50 (lowest accuracy)

9.11.3.12 typedef struct **_getResetInfoNotification** **ResetInfoNotification**

Contains the parameters passed for SLQSSetSwiGetResetInfoCallback by the device.

Parameters

<i>type</i>	<ul style="list-style-type: none"> • type of reset or power down, possible values listed below: <ul style="list-style-type: none"> – 0 - unknown – 1 - warm – 2 - hard – 3 - crash – 4 - power down
<i>source</i>	<ul style="list-style-type: none"> • entity which initiated the reset or power down, possible values listed below: <ul style="list-style-type: none"> – 0 - unknown – 1 - user requested – 2 - hardware switch – 3 - temperature critical – 4 - voltage critical – 5 - configuration update – 6 - LWM2M – 7 - OMA-DM – 8 - FOTA

Note

None

9.11.3.13 typedef struct **sensorDataUsage_s** **sensorDataUsage**

This structure contains Sensor Data Usage info.

Parameters

<i>usageMask</i>	<ul style="list-style-type: none"> Specifies which sensors were used in calculating the position in the position report.
------------------	---

- Value
 - 0x00000001 - Accelerometer used
 - 0x00000002 - Gyroscope used

Parameters

<i>aidingIndicator↔ Mask</i>	
----------------------------------	--

- Value
 - 0x00000001 - AIDED_HEADING
 - 0x00000002 - AIDED_SPEED
 - 0x00000004 - AIDED_POSITION
 - 0x00000008 - AIDED_VELOCITY

9.11.3.14 `typedef union sessionInfo sessionInformation`

This union `sessionInfo` consist of `omaDmFotaTlv`, `omaDmConfigTlv` and `omaDmNotificationsTlv`, out of which one will be unpacked against `pEventFields`.

9.11.3.15 `typedef union sessionInfoExt sessionInformationExt`

This union `sessionInfo` consist of `omaDmFotaTlv` and `omaDmConfigTlv`, out of which one will be unpacked against `pEventFields`.

9.11.3.16 `typedef struct SMSAsyncRawSend_s SMSAsyncRawSend`

This structure contains SMS parameters

Parameters

<i>sendStatus</i>	<ul style="list-style-type: none"> Send Status Values: <ul style="list-style-type: none"> QMI_ERR_NONE – No error in the request QMI_ERR_CAUSE_CODE - SMS cause code QMI_ERR_MESSAGE_DELIVERY_FAILURE - Message could not be delivered QMI_ERR_NO_MEMORY - Device could not allocate memory to formulate a response
<i>messageID</i>	<ul style="list-style-type: none"> Unique ID assigned by WMS for non-retry messages.

<i>causeCode</i>	<ul style="list-style-type: none"> WMS cause code
<i>errorClass</i>	<ul style="list-style-type: none"> Error Class Values: <ul style="list-style-type: none"> 0x00 - ERROR_CLASS_TEMPORARY 0x01 - ERROR_CLASS_PERMANENT
<i>RPCause</i>	<ul style="list-style-type: none"> GW RP cause
<i>TPCause</i>	<ul style="list-style-type: none"> GW TP Cause
<i>msgDelFailureType</i>	<ul style="list-style-type: none"> Message delivery failure type Values: <ul style="list-style-type: none"> 0x00 - WMS_MESSAGE_DELIVERY_FAILURE_TEMPORARY 0x01 - WMS_MESSAGE_DELIVERY_FAILURE_PERMANENT
<i>msgDelFailureCause</i>	<ul style="list-style-type: none"> Message delivery failure cause Values: <ul style="list-style-type: none"> 0x00 - WMS_MESSAGE_BLOCKED_DUE_TO_CALL_CONTROL
<i>alphaIDLen</i>	<ul style="list-style-type: none"> Number of sets of the pAlphaID
<i>pAlphaID</i>	<ul style="list-style-type: none"> Alpha ID
<i>userData</i>	<ul style="list-style-type: none"> Identifies the request associated with this indication.

9.11.3.17 typedef struct SMSCAddress SMSCAddressInfo

This structure holds SMSC information

Parameters

<i>length</i>	<ul style="list-style-type: none"> Number of sets of following element
<i>data</i>	<ul style="list-style-type: none"> SMSC address

9.11.3.18 typedef struct **SMSEtwsMessage** **SMSEtwsMessageInfo**

This structure holds information related earthquake and Tsunami warning system

Parameters

<i>notificationType</i>	<ul style="list-style-type: none"> Message mode 0x00 - Primary 0x01 - Secondary GSM 0x02 - Secondary UMTS
<i>length</i>	<ul style="list-style-type: none"> Number of sets of following elements
<i>data</i>	<ul style="list-style-type: none"> Raw message data

9.11.3.19 typedef struct **SMSEtwsPlmn** **SMSEtwsPlmnInfo**

This structure holds information related ETWS PLMN

Parameters

<i>mobileCountry↔ Code</i>	<ul style="list-style-type: none"> 16 bit representation of MCC value range : 0 -999
<i>mobile↔ NetworkCode</i>	<ul style="list-style-type: none"> 16 bit representation of MNC value range : 0 -999

9.11.3.20 typedef struct **SMSEventInfo_s** **SMSEventInfo**

This structure will hold the information related to received SMS events

Parameters

<i>smsEventType</i>	<ul style="list-style-type: none"> Type of the SMS events that are received. This is a bit map of SMSEventType. Only the parameters (which follows) related to the events received would be filled, and the rest of the parameters would be NULL
<i>pMTMessage↔ Info</i>	<ul style="list-style-type: none"> pointer to the SMSMTMessageInfo structure NULL, if this event is not present in the smsEventType parameter
<i>pTransfer↔ RouteMT↔ MessageInfo</i>	<ul style="list-style-type: none"> pointer to the SMSTransferRouteMTMessageInfo structure . NULL, if this event is not present in the smsEventType parameter
<i>pMessage↔ ModelInfo</i>	<ul style="list-style-type: none"> pointer to the SMSMessageModelInfo structure NULL, if this event is not present in the smsEventType parameter

<i>pEtws</i> ↔ <i>MessageInfo</i>	<ul style="list-style-type: none"> pointer to the SMSEtwsMessageInfo structure NULL, if this event is not present in the smsEventType parameter
<i>pEtwsPlmnInfo</i>	<ul style="list-style-type: none"> pointer to the SMSEtwsPlmnInfo structure NULL, if this event is not present in the smsEventType parameter
<i>pSMSC</i> ↔ <i>AddressInfo</i>	<ul style="list-style-type: none"> pointer to the SMSCAddressInfo structure NULL, if this event is not present in the smsEventType parameter
<i>pSMSOnIMSInfo</i>	<ul style="list-style-type: none"> pointer to the SMSOnIMSInfo structure NULL, if this event is not present in the smsEventType parameter Note: None

9.11.3.21 typedef struct **SMSMessageMode** SMSMessageModelInfo

This structure holds information related to message mode

Parameters

<i>messageMode</i>	<ul style="list-style-type: none"> Message mode 0x00 - CDMA 0x01 - GW
--------------------	--

9.11.3.22 typedef struct **SMSMTMessage** SMSMTMessageInfo

This structure holds information related to MT SMS

Parameters

<i>storageType</i>	<ul style="list-style-type: none"> SMS message storage type for the new message 0 - UIM 1 - NV
<i>messageIndex</i>	<ul style="list-style-type: none"> Index of the new message

9.11.3.23 typedef struct **SMSOnIMS** SMSOnIMSInfo

This structure holds information related to message mode

Parameters

<i>smsOnIMS</i>	<ul style="list-style-type: none"> Indicates whether the message is received from IMS 0x00 - Message is not received from IMS 0x01 - Message is received from IMS 0x02-0xFF - Reserved Note: In multiple modem solutions, this TLV may be used to help the client determine with which modem to communicate. This TLV may not be supported on all implementations.
-----------------	---

9.11.3.24 typedef struct **SMSTransferRouteMTMessage** **SMSTransferRouteMTMessageInfo**

This structure holds information related to transfer route MT SMS

Parameters

<i>ackIndicator</i>	<ul style="list-style-type: none"> Parameter to indicate if ACK must be sent by the control point 0x00 - Send ACK 0x01 - Do not send ACK
<i>transactionID</i>	<ul style="list-style-type: none"> Transaction ID of the message
<i>format</i>	<ul style="list-style-type: none"> Message format 0x00 - CDMA 0x02 - 0x05 - Reserved 0x06 - GW_PP 0x07 - GW_BC
<i>length</i>	<ul style="list-style-type: none"> Length of the raw message. This length should not exceed the maximum WMS payload length of 256 bytes
<i>data</i>	<ul style="list-style-type: none"> Raw message data

9.11.3.25 typedef struct **svUsedforFix_s** **svUsedforFix**

This structure contains SVs Used to Calculate the Fix.

Parameters

<i>gnssSvUsed↔ List_len</i>	<ul style="list-style-type: none"> Number of sets of gnssSvUsedList
<i>pGnssSvUsed↔ List</i>	<ul style="list-style-type: none"> Entry in the list contains the SV ID of a satellite used for calculating this position report. Following information is associated with each SV ID: <ul style="list-style-type: none"> GPS - 1 to 32 SBAS - 33 to 64 GLONASS - 65 to 96 QZSS - 193 to 197 BDS - 201 to 237

9.11.3.26 typedef struct SwiOTAMsg_s SwiOTAMsg

This structure contains OTA message

Parameters

<i>type</i>	<ul style="list-style-type: none">• message type<ul style="list-style-type: none">– 0 - LTE ESM uplink– 1 - LTE ESM downlink– 2 - LTE EMM uplink– 3 - LTE EMM downlink– 4 - GSM/UMTS uplink– 5 - GSM/UMTS downlink
<i>data_len</i>	<ul style="list-style-type: none">• OTA Message Content Length
<i>data</i>	<ul style="list-style-type: none">• OTA Message Content
<i>pLteNasRelInfo</i>	<ul style="list-style-type: none">• LTE NAS Release Info• see LteNasReleaseInfo for details
<i>pTime</i>	<ul style="list-style-type: none">• Seconds in local time since Jan. 6th 1980 00:00:00 UTC

9.11.3.27 typedef void(* tFNActivationStatus) (ULONG activationStatus)

Activation status callback function.

Parameters

<i>activationStatus</i>	<ul style="list-style-type: none">• Service Activation Code<ul style="list-style-type: none">– 0 - Service not activated– 1 - Service activated– 2 - Activation connecting– 3 - Activation connected– 4 - OTASP security authenticated– 5 - OTASP NAM downloaded– 6 - OTASP MDN downloaded– 7 - OTASP IMSI downloaded– 8 - OTASP PRL downloaded– 9 - OTASP SPC downloaded– 10 - OTASP settings committed
-------------------------	--

9.11.3.28 `typedef void(* tFNAllCallStatus) (voiceSetAllCallStatusCbInfo *pVoiceSetAllCallStatusCbInfo)`

Voice Call Status Callback function. This function pointer will be executed to process received Indication.

Parameters

<i>pVoiceSetAllCallStatusCbInfo</i>	<ul style="list-style-type: none"> Call back will populated memory pointed by this parameter when a call is originated, connected, or ended. See voiceSetAllCallStatusCbInfo for more information.
-------------------------------------	---

9.11.3.29 `typedef void(* tFNASwiLTECphyCallInfo) (QmiCbkNasLTECphyCaInfo *pQmiCbkNasLTECphyCaInfo)`

LTE CPHY CA message callback function.

Parameters

<i>pQmiCbkNasLTECphyCaInfo[OUT]</i>	<ul style="list-style-type: none"> Events related to NAS, see QmiCbkNasLTECphyCaInfo for details.
-------------------------------------	--

9.11.3.30 `typedef void(* tFNASwiOTAMsg) (SwiOTAMsg *pSwiOTAMsg)`

OTA message callback function.

Parameters

<i>pSwiOTAMsg[OUT]</i>	<ul style="list-style-type: none"> Events related to NAS, see SwiOTAMsg for details
------------------------	--

9.11.3.31 `typedef void(* tFNAsyncRawSend) (SMSAsyncRawSend *pSMSAsyncRawSend)`

SMS event related callback function.

Parameters

<i>pSMSEventInfo[OUT]</i>	<ul style="list-style-type: none"> Events related to SMS, see SMSEventInfo for details
---------------------------	---

9.11.3.32 `typedef void(* tFNBandPreference) (ULONGLONG band_pref)`

Band Preference Callback function

Parameters

<i>pBandPref</i>	<ul style="list-style-type: none"> - Bit mask representing the current band preference Bit position meanings: <ul style="list-style-type: none"> • 0 - BC0_A - Band Class 0, A-System • 1 - BC0_B - Band Class 0, B-System, Band Class 0 AB , GSM 850 Band • 2 - BC1 - Band Class 1, all blocks • 3 - BC2 - Band Class 2 place holder • 4 - BC3 - Band Class 3, A-System • 5 - BC4 - Band Class 4, all blocks • 6 - BC5 - Band Class 5, all blocks • 7 - GSM_DCS_1800 - GSM DCS band • 8 - GSM_EGSM_900 - GSM Extended GSM (E-GSM) band • 9 - GSM_PGSM_900 - GSM Primary GSM (P-GSM) band • 10 - BC6 - Band Class 6 • 11 - BC7 - Band Class 7 • 12 - BC8 - Band Class 8 • 13 - BC9 - Band Class 9 • 14 - BC10 - Band Class 10 • 15 - BC11 - Band Class 11 • 16 - GSM_450 - GSM 450 band • 17 - GSM_480 - GSM 480 band • 18 - GSM_750 - GSM 750 band • 19 - GSM_850 - GSM 850 band • 20 - GSM_RGSM_900 - GSM Railways GSM Band • 21 - GSM_PCS_1900 - GSM PCS band • 22 - WCDMA_I_IMT_2000 - WCDMA EUROPE JAPAN and CHINA IMT 2100 band • 23 - WCDMA_II_PCS_1900 - WCDMA US PCS 1900 band • 24 - WCDMA_III_1700 - WCDMA EUROPE and CHINA DCS 1800 band • 25 - WCDMA_IV_1700 - WCDMA US 1700 band • 26 - WCDMA_V_850 - WCDMA US 850 band • 27 - WCDMA_VI_800 - WCDMA JAPAN 800 band • 28 - BC12 - Band Class 12 • 29 - BC14 - Band Class 14 • 30 - RESERVED_2 - Reserved 2 • 31 - BC15 - Band Class 15 • 32 - 47 - Reserved • 48 - WCDMA_VII_2600 - WCDMA EUROPE 2600 band • 49 - WCDMA_VIII_900 - WCDMA EUROPE and JAPAN 900 band • 50 - WCDMA_IX_1700 - WCDMA JAPAN 1700 band • 51 to 55 - Reserved • 56 - BBC16 - Band Class 16 • 57 - BC17 - Band Class 17 • 58 - BC18 - Band Class 18 • 59 - BC19 - Band Class 19 • 60 to 64 - Reserved
------------------	---

Note

Timeout: NA To set the band preference the API [SLQSSetBandPreference\(\)](#) should be used

9.11.3.33 `typedef void(* tFNBstAvailPos) (QmiCbkLocBestAvailPosInd *pBestAvailPos)`

9.11.3.34 `typedef void(* tFNCATEvent) (ULONG eventID, ULONG eventLen, BYTE *pEventData)`

CAT event callback function.

Parameters

<i>eventID</i>	<ul style="list-style-type: none"> • Event ID <ul style="list-style-type: none"> – 16 - Display Text – 17 - Get In-Key – 18 - Get Input – 19 - Setup Menu – 20 - Select Item – 21 - Send SMS - Alpha Identifier – 22 - Setup Event List – 23 - Setup Idle Mode Text – 24 - Language Notification – 25 - Refresh – 26 - End Proactive Session
<i>eventLen</i>	<ul style="list-style-type: none"> • Length of pData (in bytes)
<i>pEventData</i>	<ul style="list-style-type: none"> • Data specific to the CAT event ID See currentCatEvent for details

Note

Technology Supported: UMTS

9.11.3.35 `typedef void(* tFNCbkUimSlotStatusChangeInd) (UIMSlotStatusChangeInfo *pQmiCbkUimSlotStatusChangeInd)`

Slot Status Change Notification callback.

Parameters

<i>pQmiCbkUimSlotStatusChangeInd</i>	<ul style="list-style-type: none"> • See UIMSlotStatusChangeInfo for more information.
--------------------------------------	---

9.11.3.36 `typedef void(* tFNDataCapabilities) (BYTE dataCapsSize, BYTE *pDataCaps)`

Serving system data capabilities callback function.

Parameters

<i>dataCapsSize</i>	<ul style="list-style-type: none"> • Number of elements the data capability array contains
<i>pDataCaps</i>	<ul style="list-style-type: none"> • Data Capabilities Array. <ul style="list-style-type: none"> – 1 - GPRS – 2 - EDGE – 3 - HSDPA – 4 - HSUPA – 5 - WCDMA – 6 - CDMA 1xRTT – 7 - CDMA 1xEV-DO Rev 0 – 8 - CDMA 1xEV-DO Rev. A – 9 - GSM – 10 - EVDO Rev. B – 11 - LTE – 12 - HSDPA Plus – 13 - Dual Carrier HSDPA Plus

9.11.3.37 `typedef void(* tFNDataSysStatus)(CurrDataSysStat *pCurrDataSysStat)`

Data System Status callback.

Parameters

<i>pCurrDataSysStat</i>	<ul style="list-style-type: none"> • See CurrDataSysStat for more information.
-------------------------	---

9.11.3.38 `typedef void(* tFNDelAssistData)(delAssistDataStatus *pAssistDataNotification)`

Delete Assist Data Notification callback.

Parameters

<i>pAssistDataNotification</i>	<ul style="list-style-type: none"> • See delAssistDataStatus for more information.
--------------------------------	---

9.11.3.39 `typedef void(* tFNDeviceStateChange)(eDevState device_state)`

Device State Change callback function prototype

Parameters

<i>device_state</i>	<ul style="list-style-type: none"> the current state of the device
---------------------	---

Note

Does not require communication with the device

9.11.3.40 `typedef void(* tFNDHCPv4ClientLeaseStatus) (BYTE instance, WdsDHCPv4ClientLeaseInd *pMsg)`

DHCPv4 client lease status message callback function.

Parameters

<i>pMsg[OUT]</i>	<ul style="list-style-type: none"> Events related to DHCPv4 client lease, see WdsDHCPv4ClientLeaseInd for details
------------------	--

9.11.3.41 `typedef void(* tFNDTMFEvent) (voiceDTMFEventInfo *pVoiceDTMFEventInfo)`

Preferred DTMF event indication callback.

Parameters

<i>pVoiceDTMF↔ EventInfo</i>	<ul style="list-style-type: none"> See voiceDTMFEventInfo for more information.
----------------------------------	--

9.11.3.42 `typedef void(* tFNDUNCallInfo) (DUNCallInfoInd *pDUNCallInfo)`

DUN Call Info indication callback.

Parameters

<i>pDUNCallInfo</i>	<ul style="list-style-type: none"> See DUNCallInfoInd for more information.
---------------------	--

9.11.3.43 `typedef void(* tFNEventPosition) (QmiCbkLocPositionReportInd *pLocPositionReport)`

9.11.3.44 `typedef void(* tFNFwDidCompletion) (ULONG fwdld_completion_status)`

Firmware Download Completion callback function prototype

Parameters

<i>error_code</i>	<ul style="list-style-type: none"> error code returned from firmware download operation, the possible return values are listed below: <ul style="list-style-type: none"> eQCWWAN_ERR_NONE - indicates firmware download/switching is successful eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED - indicates no actual download takes place, this is the case of image switching stored on device eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE - indicates modem enters firmware download mode, firmware flashing is going to be started. eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE - indicates firmware flashing was complete, SDK is waiting for modem to reboot (can be more than one time), when modem is ready, SDK will send eQCWWAN_ERR_NONE to the host application.
-------------------	---

Note

Does not require communication with the device

9.11.3.45 `typedef void(* tFNGnssSvInfo) (gnssSvInfoNotification *pGnssSvInfoNotification)`

GNSS SVN Information Notification callback.

Parameters

<i>pGnssSvInfo</i> <i>Notification</i>	<ul style="list-style-type: none"> See gnssSvInfoNotification for more information.
---	--

9.11.3.46 `typedef void(* tFNHDRPersonaity) (HDRPersonalityInd *pHDRPers)`

HDR Personality indication callback.

Parameters

<i>pHDRPers</i>	<ul style="list-style-type: none"> See HDRPersonalityInd for more information.
-----------------	---

Note

Technology Supported: CDMA

9.11.3.47 `typedef void(* tFNImsaPdpStatus) (imsaPdpStatusInfo *plmsaPdpStatusInfo)`

IMSA PDP status indication callback.

Parameters

<i>plmsaPdp↔ StatusInfo</i>	<ul style="list-style-type: none"> See imsaPdpStatusInfo for more information.
---------------------------------	---

9.11.3.48 `typedef void(* tFNImsaRatStatus) (imsaRatStatusInfo *plmsaRatStatusInfo)`

IMSA RAT handover status indication callback.

Parameters

<i>plmsaRat↔ StatusInfo</i>	<ul style="list-style-type: none"> See imsaRatStatusInfo for more information.
---------------------------------	---

9.11.3.49 `typedef void(* tFNImsaRegStatus) (imsaRegStatusInfo *plmsaRegStatusInfo)`

IMSA Registration Status indication callback.

Parameters

<i>plmsaReg↔ StatusInfo</i>	<ul style="list-style-type: none"> See imsaRegStatusInfo for more information.
---------------------------------	---

9.11.3.50 `typedef void(* tFNImsaSvcStatus) (imsaSvcStatusInfo *plmsaSvcStatusInfo)`

IMSA Service Status indication callback.

Parameters

<i>plmsaSvc↔ StatusInfo</i>	<ul style="list-style-type: none"> See imsaSvcStatusInfo for more information.
---------------------------------	---

9.11.3.51 `typedef void(* tFNImRegMgrConfig) (imsRegMgrConfigInfo *plmsRegMgrConfigInfo)`

IMS Reg Mgr Config indication callback.

Parameters

<i>plmsRegMgr↔ ConfigInfo</i>	<ul style="list-style-type: none"> See imsRegMgrConfigInfo for more information.
-----------------------------------	---

9.11.3.52 typedef void(* tFNImSIPConfig) (imsSIPConfigInfo *plmsSIPConfigInfo)

IMS SIP Config indication callback.

Parameters

<i>plmsSIP↔ ConfigInfo</i>	<ul style="list-style-type: none">• See imsSIPConfigInfo for more information.
--------------------------------	--

9.11.3.53 typedef void(* tFNImSMSConfig) (imsSMSConfigInfo *plmsSMSConfigInfo)

IMS SMS Config indication callback.

Parameters

<i>plmsSMS↔ ConfigInfo</i>	<ul style="list-style-type: none">• See imsSMSConfigInfo for more information.
--------------------------------	--

9.11.3.54 typedef void(* tFNImUserConfig) (imsUserConfigInfo *plmsUserConfigInfo)

IMS User Config indication callback.

Parameters

<i>plmsUser↔ ConfigInfo</i>	<ul style="list-style-type: none">• See imsUserConfigInfo for more information.
---------------------------------	---

9.11.3.55 typedef void(* tFNImVoIPConfig) (imsVoIPConfigInfo *plmsVoIPConfigInfo)

IMS VoIP Config indication callback.

Parameters

<i>plmsVoIP↔ ConfigInfo</i>	<ul style="list-style-type: none">• See imsVoIPConfigInfo for more information.
---------------------------------	---

9.11.3.56 typedef void(* tFNInfoRec) (voiceInfoRec *pVoiceInfoRec)

Voice Information Record callback.

Parameters

<i>pVoiceInfoRec</i>	<ul style="list-style-type: none"> • See voiceInfoRec for more information.
----------------------	--

Note

Technology Supported: CDMA

Device Supported: MC7750

9.11.3.57 `typedef void(* tFNInjectPosition) (QmiCbkLocInjectPositionInd *pInjectPositionNotification)`

Inject Position Notification callback.

Parameters

<i>pInjectPosition</i> <i>Notification</i>	<ul style="list-style-type: none"> • See QmiCbkLocInjectPositionInd for more information.
---	--

9.11.3.58 `typedef void(* tFNInjectSensorData) (QmiCbkLocInjectSensorDataInd *pLocInjectSensorData)`

9.11.3.59 `typedef void(* tFNInjectTimeStatus) (QmiCbkLocInjectTimeInd *pLocInjectTime)`

9.11.3.60 `typedef void(* tFNInjectUTCTime) (QmiCbkLocInjectUTCTimeInd *pInjectUTCTimeNotification)`

Inject UTC Time Notification callback.

Parameters

<i>pInjectUTC</i> <i>TimeNotification</i>	<ul style="list-style-type: none"> • See QmiCbkLocInjectUTCTimeInd for more information.
--	---

9.11.3.61 `typedef void(* tFNLURject) (ULONG serviceDomain, ULONG rejectCause)`

LU reject callback function.

Parameters

<i>serviceDomain</i>	<ul style="list-style-type: none"> • Service domain <ul style="list-style-type: none"> – 1 - Circuit Switched – 2 - Packet Switched – 3 - Circuit and Packet Switched
----------------------	--

<i>rejectCause</i>	<ul style="list-style-type: none"> • Reject cause • Valid Values <ul style="list-style-type: none"> – 2 - IMSI unknown in HLR – 3 - Illegal MS – 4 - IMSI unknown in VLR – 5 - IMEI not accepted – 6 - Illegal ME – 11 - PLMN not allowed\ – 12 - Location Area not allowed – 13 - Roaming not allowed in this location area – 15 - No Suitable Cells In Location Area – 17 - Network failure – 20 - MAC failure – 21 - Synch failure – 22 - Congestion – 23 - GSM authentication unacceptable – 25 - Not authorized for this CSG – 32 - Service option not supported – 33 - Requested service option not subscribed – 34 - Service option temporarily out of order – 38 - Call cannot be identified – 48 to 63 - retry upon entry into a new cell – 95 - Semantically incorrect message – 96 - Invalid mandatory information – 97 - Message type non-existent or not implemented – 98 - Message type not compatible with the protocol state – 99 - Information element non-existent or not implemented – 100 - Conditional IE error – 101 - Message not compatible with the protocol state – 111 - Protocol error, unspecified – Note - Any other value received by the mobile station shall be treated as 34, 'Service option temporarily out of order'. • Any other value received by the network shall be treated as 111, 'Protocol error, unspecified'. <p>See 3GPP TS 24.008, Section 4.4.4.7 and Section 10.5.3.6 See qaGobiApiTableCall↔EndReasons.h for Call End reasons</p>
--------------------	--

Note

Technology Supported: UMTS

9.11.3.62 `typedef void(* tFNMemoryFull) (SMSMemoryInfo *pSMSMemoryFullInfo)`

SMS Memory related callback function.

Parameters

<i>pSMSMemoryFullInfo</i> [OUT]	<ul style="list-style-type: none"> • pointer to SMSMemoryInfo. • see SMSMemoryInfo for details.
---------------------------------	---

9.11.3.63 `typedef void(* tFNMessageWaiting) (msgWaitingInfo *pSMSMessageWaitingInfo)`

SMS Memory related callback function.

Parameters

<i>pSMSMessageWaitingInfo</i> [OUT]	<ul style="list-style-type: none"> • pointer to msgWaitingInfo. • see msgWaitingInfo for details.
-------------------------------------	---

9.11.3.64 `typedef void(* tFNMitlLvIRpt) (QmiCbkTmdMitlLvIRptInd *pSetLocCradleMount)`

9.11.3.65 `typedef void(* tFNMobileIPStatus) (ULONG mipStatus)`

Mobile IP status callback function.

Parameters

<i>mipStatus</i>	<ul style="list-style-type: none"> • Mobile IP Status <ul style="list-style-type: none"> – 0 - success – All others error codes as defined in RFC 2002 See qaGobiApiTableCallEndReasons.h for mobile IP error codes
------------------	---

9.11.3.66 `typedef void(* tFNModemTempInfo) (modemTempNotification *pModemTempNotification)`

Modem Temperature Information callback.

Parameters

<i>pModemTempNotification</i>	<ul style="list-style-type: none"> • See modemTempNotification for more information.
-------------------------------	---

9.11.3.67 `typedef void(* tFNNasTimer) (nasTimers *timers)`

9.11.3.68 `typedef void(* tFNNet) (ULONG q_depth, BYTE isThrottle, BYTE instanceld)`

Transmit Queue Length Change callback function prototype

Parameters

<i>q_depth</i>	<ul style="list-style-type: none">transmit queue length
<i>isThrottle</i>	<ul style="list-style-type: none">0: unthrottle1: throttle
<i>instanceld</i>	<ul style="list-style-type: none">qmi instance id

Note

Does not require communication with the device

9.11.3.69 `typedef void(* tFNNetworkTime) (nasNetworkTime *pNasNetworkTime)`

Network Time indication callback.

Parameters

<i>pNasNetworkTime</i>	<ul style="list-style-type: none">See nasNetworkTime for more information.
------------------------	--

9.11.3.70 `typedef void(* tFNNewGPS) (double dLongitude, double dLatitude, BYTE session_status, ULONG pos_src)`

Set Current Location Data.

Parameters

<i>dLongitude[IN]</i>	<ul style="list-style-type: none">Current Longitude Value
<i>dLatitude[IN]</i>	<ul style="list-style-type: none">Current Latitude Value

<i>session</i> ↔ <i>status</i> [IN]	<ul style="list-style-type: none"> • Session Status <ul style="list-style-type: none"> – 0 - Success – 1 - In progress – 2 - General failure – 3 - Timeout – 4 - User ended the session – 5 - Bad parameter – 6 - Phone is offline – 7 - Engine is locked – 8 - E911 session in progress
<i>pos_src</i> [IN]	<ul style="list-style-type: none"> • position source • Bitmasks <ul style="list-style-type: none"> – 0x01 - GPS – 0x02 - Cell ID – 0x04 - GLONASS – 0x08 - Network – 0x10 - External positino injection – Others - unknown

9.11.3.71 typedef void(* tFNNewNMEA) (LPCSTR pNMEA)

New NMEA sentence callback function.

Parameters

<i>pNMEA</i>	<ul style="list-style-type: none"> • NULL-terminated string containing the position data in NMEA sentence format
--------------	---

9.11.3.72 typedef void(* tFNNewRMTransferStatistics) (QmiCbkWdsStatisticsIndState *pMsg)

PDS session state callback function.

Parameters

<i>enabledStatus</i>	<ul style="list-style-type: none"> • GPS enabled status <ul style="list-style-type: none"> – 0 - Disable – 1 - Enable
<i>trackingStatus</i>	<ul style="list-style-type: none"> • GPS tracking status <ul style="list-style-type: none"> – 0 - Unknown – 1 - Inactive – 2 - Active RM Transfer Statistics message callback function.
	Generated by Doxygen

<i>pMsg[OUT]</i>	<ul style="list-style-type: none">Events related to NAS, see QmiCbKWsStatisticsIndState for details
------------------	---

9.11.3.73 typedef void(* tFNNewSMS) (ULONG storageType, ULONG messageIndex)

New SMS message callback function.

Parameters

<i>storageType</i>	<ul style="list-style-type: none">SMS message storage type for the new message 0 - UIM 1 - NV
<i>messageIndex</i>	<ul style="list-style-type: none">Index of the new message

9.11.3.74 typedef void(* tFNOMADMState) (ULONG sessionState, ULONG failureReason)

OMA-DM state callback function

Parameters

<i>sessionState</i>	<ul style="list-style-type: none">Session state<ul style="list-style-type: none">0x00 - Complete, information was updated0x01 - Complete, update information is unavailable0x02 - Failed0x03 - Retrying0x04 - Connecting0x05 - Connected0x06 - Authenticated0x07 - Mobile Directory Number (MDN) downloaded0x08 - Mobile Station Identifier (MSID) downloaded0x09 - PRL downloaded0x0A - Mobile IP (MIP) profile downloaded
<i>failureReason</i>	<ul style="list-style-type: none">Session failure reason (when state indicates failure)<ul style="list-style-type: none">0x00 - Unknown0x01 - Network is unavailable0x02 - Server is unavailable0x03 - Authentication failed0x04 - Maximum retry exceeded0x05 - Session is cancelled

Note

Technology Supported: CDMA

9.11.3.75 `typedef void(* tFNOpMode) (ULONG mode)`

9.11.3.76 `typedef void(* tFNOTASPStatus) (voiceOTASPStatusInfo *pVoiceOTASPStatusInfo)`

OTASP or OTAPA event Indication Callback function

Parameters

<i>pVoiceOTASP↔ StatusInfo</i>	<ul style="list-style-type: none"> OTASP Status Information. See voiceOTASPStatusInfo for more information
------------------------------------	--

Note

Technology Supported: CDMA

9.11.3.77 `typedef void(* tFNPacketSrvState) (packetSrvStatus *pPacketSrvStatus)`

Packet Service state callback function.

Parameters

<i>pPacketSrv↔ Status</i>	<ul style="list-style-type: none"> See packetSrvStatus for more details
-------------------------------	--

9.11.3.78 `typedef void(* tFNPDSSState) (ULONG enabledStatus, ULONG trackingStatus)`

PDS session state callback function.

Parameters

<i>enabledStatus</i>	<ul style="list-style-type: none"> GPS enabled status <ul style="list-style-type: none"> 0 - Disable 1 - Enable
<i>trackingStatus</i>	<ul style="list-style-type: none"> GPS tracking status <ul style="list-style-type: none"> 0 - Unknown 1 - Inactive 2 - Active
Generated by Doxygen	

9.11.3.79 `typedef void(* tFNPower) (ULONG operatingMode)`

Power operating mode callback function.

Parameters

<i>operatingMode</i>	<ul style="list-style-type: none">• Service Operating mode See Tables for Operating Modes
----------------------	---

Note

Technology Supported: UMTS/CDMA
Device Supported: MC83x5, MC7700/50

9.11.3.80 `typedef void(* tFNPrivacyChange) (voicePrivacyInfo *pVoicePrivacyInfo)`

Preferred voice privacy indication callback.

Parameters

<i>pVoicePrivacyInfo</i>	<ul style="list-style-type: none">• See voicePrivacyInfo for more information.
--------------------------	--

Note

Technology Supported: CDMA

9.11.3.81 `typedef void(* tFNQosNWStatus) (BYTE status)`

QOS Network status callback function.

Parameters

<i>status</i>	Network QoS support status <ul style="list-style-type: none">• 0x00 – Current network does not support QoS• 0x01 – Current network supports QoS
---------------	--

9.11.3.82 `typedef void(* tFNQosPriEvent) (WORD event)`

QOS primary flow callback function.

Parameters

<i>event</i>	Event which causes this indication: <ul style="list-style-type: none"> • 0x0001 – Primary flow QoS modify operation success • 0x0002 – Primary flow QoS modify operation failure
--------------	--

9.11.3.83 `typedef void(* tFNQosStatus) (BYTE instance, ULONG id, BYTE status, BYTE event, BYTE reason)`

QOS Status callback function.

Parameters

<i>instance</i>	<ul style="list-style-type: none"> • QMI instance
<i>id</i>	<ul style="list-style-type: none"> • Index identifying the QoS flow whose status is being reported
<i>status</i>	Current QoS flow status: <ul style="list-style-type: none"> • 0x01 – QMI_QOS_STATUS_ACTIVATED • 0x02 – QMI_QOS_STATUS_SUSPENDED • 0x03 – QMI_QOS_STATUS_GONE
<i>event</i>	<ul style="list-style-type: none"> • 0x01 – QMI_QOS_ACTIVATED_EV • 0x02 – QMI_QOS_SUSPENDED_EV • 0x03 – QMI_QOS_GONE_EV • 0x04 – QMI_QOS_MODIFY_ACCEPTED_EV • 0x05 – QMI_QOS_MODIFY_REJECTED_EV • 0x06 – QMI_QOS_INFO_CODE_UPDATED_EV

<i>reason</i>	<ul style="list-style-type: none"> • 0x01 - QMI_QOS_INVALID_PARAMS • 0x02 - QMI_QOS_INTERNAL_CALL_ENDED • 0x03 - QMI_QOS_INTERNAL_ERROR • 0x04 - QMI_QOS_INSUFFICIENT_LOCAL_Resources • 0x05 - QMI_QOS_TIMED_OUT_OPERATION • 0x06 - QMI_QOS_INTERNAL_UNKNOWN_CAUSE_CODE • 0x07 - QMI_QOS_INTERNAL_MODIFY_IN_PROGRESS • 0x08 - QMI_QOS_NOT_SUPPORTED • 0x09 - QMI_QOS_NOT_AVAILABLE • 0x0A - QMI_QOS_NOT_GUARANTEED • 0x0B - QMI_QOS_INSUFFICIENT_NETWORK_RESOURCES • 0x0C - QMI_QOS_AWARE_SYSTEM • 0x0D - QMI_QOS_UNAWARE_SYSTEM • 0x0E - QOS_REJECTED_OPERATION • 0x0F - QMI_QOS_WILL_GRANT_WHEN_QOS_RESUMED • 0x10 - QMI_QOS_NETWORK_CALL_ENDED • 0x11 - QMI_QOS_NETWORK_SERVICE_NOT_AVAILABLE • 0x12 - QMI_QOS_NETWORK_L2_LINK_RELEASED • 0x13 - QMI_QOS_NETWORK_L2_LINK_REESTAB_REJ • 0x14 - QMI_QOS_NETWORK_L2_LINK_REESTAB_IND • 0x15 - QMI_QOS_NETWORK_UNKNOWN_CAUSE_CODE • 0x16 - QMI_NETWORK_BUSY
---------------	--

9.11.3.84 `typedef void(* tFNRankIndicator)(RankIndicatorInd *pRankIndicatorInd)`

9.11.3.85 `typedef void(* tFNResetInfo)(ResetInfoNotification *pResetInfoNotification)`

Get Reset Info Indication callback.

Parameters

<i>pResetInfo</i> ↔ <i>Notification</i>	<ul style="list-style-type: none"> • See ResetInfoNotification for more information.
--	---

9.11.3.86 `typedef void(* tFNRFInfo)(ULONG radioInterface, ULONG activeBandClass, ULONG activeChannel)`

RF information callback function.

Parameters

<i>radioInterface</i>	<ul style="list-style-type: none"> • Radio interface technology of the signal being measured See Tables for Radio Interface
-----------------------	--

<i>activeBandClass</i>	<ul style="list-style-type: none"> Active band class See Tables for Active Band Class
<i>activeChannel</i>	<ul style="list-style-type: none"> Active channel <ul style="list-style-type: none"> 0 - Channel is not relevant to the reported technology

9.11.3.87 typedef void(* tFNRoamingIndicator) (ULONG roaming)

Roaming indicator callback function.

Parameters

<i>roaming</i>	<ul style="list-style-type: none"> Roaming Indication <ul style="list-style-type: none"> 0 - Roaming 1 - Home 2 - Roaming partner >2 - Operator defined values
----------------	---

9.11.3.88 typedef void(* tFNSDKTerminated) (BYTE *psReason)

SDK terminated callback function prototype

Parameters

<i>psReason</i>	<ul style="list-style-type: none"> sdk termination reason string
-----------------	---

Note

Timeout: None

Does not require communication with the device

9.11.3.89 typedef void(* tFNSensorStreaming) (QmiCbkLocSensorStreamingInd *pLocSensorStream)

9.11.3.90 typedef void(* tFNServingSystem) (struct ServingSystemInfo *pServingSystem, struct RoamingInfo *pRoamingInfo)

Serving System callback function

Parameters

<i>pServingSystem</i>	<ul style="list-style-type: none"> • ServingSystemInfo structure
-----------------------	---

9.11.3.91 `typedef void(* tFNSetCradleMount) (QmiCbkLocCradleMountInd *pSetLocCradleMount)`

9.11.3.92 `typedef void(* tFNSetEngineState) (QmiCbkLocEngineStateInd *pSetLocEngineState)`

9.11.3.93 `typedef void(* tFNSetEventTimeSync) (QmiCbkLocEventTimeSyncInd *pSetLocEventTimeSync)`

9.11.3.94 `typedef void(* tFNSetExtPowerConfig) (QmiCbkLocSetExtPowerConfigInd *pSetExtConfigIndStatus)`

9.11.3.95 `typedef void(* tFNSigInfo) (nasSigInfo *pNasSigInfo)`

Signal Strength Information indication callback.

Parameters

<i>pNasSigInfo</i>	<ul style="list-style-type: none"> • See nasSigInfo for more information.
--------------------	--

9.11.3.96 `typedef void(* tFNSignalStrength) (INT8 signalStrength, ULONG radiolInterface)`

Signal strength callback function.

Parameters

<i>signalStrength</i>	<ul style="list-style-type: none"> • Received signal strength (in dBm)
<i>radiolInterface</i>	<ul style="list-style-type: none"> • Radio interface technology of the signal being measured See Tables for Radio Interface

9.11.3.97 `typedef void(* tFNSLQSOMADMAAlert) (ULONG eventType, BYTE *pEventFields)`

SWIOMA-DM network-initiated alert callback function

Parameters

<i>eventType</i>	<ul style="list-style-type: none"> • 0x00 - SWIOMA-DM FOTA • 0x01 - SWIOMA-DM Config • 0x02 - SWIOMA-DM Notification
Generated by Doxygen	

<i>pEventFields</i>	<ul style="list-style-type: none"> • Pointer to structure containing info for that session type • See sessionInfo for more details
---------------------	--

9.11.3.98 `typedef void(* tFNSLQSQOSEvent) (BYTE instance, QosFlowInfo *pFlowInfo)`

QOS Event callback function.

Parameters

<i>instance</i>	<ul style="list-style-type: none"> • QMI instance
<i>pFlowInfo</i>	<ul style="list-style-type: none"> • See QosFlowInfo for more information

9.11.3.99 `typedef void(* tFNSLQSSessionState) (slqsSessionStateInfo *pSessionStateInfo)`

Session state callback function.

Parameters

<i>pSessionState↔ Info</i>	<ul style="list-style-type: none"> • See slqsSessionStateInfo for more details
--------------------------------	---

9.11.3.100 `typedef void(* tFNSLQSSignalStrengths) (struct SLQSSignalStrengthsInformation sSLQSSignalStrengthsInfo)`

Received Signal Strength Information callback function.

Parameters

<i>sSLQSSignal↔ StrengthsInfo</i>	<ul style="list-style-type: none"> • See SLQSSignalStrengthsInformation for more information.
---------------------------------------	--

9.11.3.101 `typedef void(* tFNSLQSWDSEvent) (slqsWdsEventInfo *pWdsEventInfo)`

WDS Event callback function.

Parameters

<i>pWdsEventInfo</i>	<ul style="list-style-type: none"> • See slqsWdsEventInfo for more details
	Generated by Doxygen

9.11.3.102 `typedef void(* tFNSMSEvents) (SMSEventInfo *pSMSEventInfo)`

SMS event related callback function.

Parameters

<i>pSMSEventInfo</i> [OUT]	<ul style="list-style-type: none">Events related to SMS, see SMSEventInfo for details
----------------------------	---

9.11.3.103 `typedef void(* tFNSUPSInfo) (voiceSUPSInfo *pVoiceSUPSInfo)`

Preferred SUPS indication callback.

Parameters

<i>pVoiceSUPSInfo</i>	<ul style="list-style-type: none">See voiceSUPSInfo for more information.
-----------------------	---

Note

Technology Supported: GSM

9.11.3.104 `typedef void(* tFNSUPSNotification) (voiceSUPSNotification *pVoiceSUPSNotification)`

Supplementary service notification callback.

Parameters

<i>pVoiceSUPSNotification</i>	<ul style="list-style-type: none">See voiceSUPSNotification for more information.
-------------------------------	---

9.11.3.105 `typedef void(* tFNSysInfo) (nasSysInfo *pNasSysInfo)`

System Information indication callback.

Parameters

<i>pNasSysInfo</i>	<ul style="list-style-type: none">See nasSysInfo for more information.
--------------------	--

9.11.3.106 `typedef void(* tFNSysSelectionPref) (sysSelectPrefInfo *pSysSelectPrefInfo)`

System Selection Preference Callback function

Parameters

<i>pSysSelectPrefInfo</i>	<ul style="list-style-type: none"> Current System Selection preferences for the device. See sysSelectPrefInfo for more information
---------------------------	--

9.11.3.107 `typedef void(* tFNtransLayerInfo) (transLayerNotification *pTransLayerNotification)`

Transport Layer Information callback.

Parameters

<i>transLayerNotification</i>	<ul style="list-style-type: none"> See transLayerNotification for more information.
-------------------------------	--

9.11.3.108 `typedef void(* tFNtransNWRegInfo) (transNWRegInfoNotification *pTransNWRegInfoNotification)`

Transport Network Registration Information callback.

Parameters

<i>pTransNWRegInfoNotification</i>	<ul style="list-style-type: none"> See transNWRegInfoNotification for more information.
------------------------------------	--

9.11.3.109 `typedef void(* tFNUIMRefresh) (UIMRefreshEvent *pUIMRefreshEvent)`

UIM Refresh Callback function

Parameters

<i>pUIMRefreshEvent</i>	<ul style="list-style-type: none"> Pointer to Refresh Event structure. See UIMRefreshEvent for more information
-------------------------	---

9.11.3.110 `typedef void(* tFNUIMStatusChangeInfo) (UIMStatusChangeInfo *pUIMStatusChangeInfo)`

UIM Status Change Callback function

Parameters

<i>pUIMStatusChangeInfo</i>	<ul style="list-style-type: none"> • Pointer to UIM status change structure. • See UIMStatusChangeInfo for more information
-----------------------------	---

9.11.3.111 `typedef void(* tFNUSSDNotification) (ULONG type, BYTE *pNetworkInfo)`

SetUSSDNotificationCallback function prototype

Parameters

<i>type</i>	<ul style="list-style-type: none"> - Notification type <ul style="list-style-type: none"> • 0x01 - No action required • 0x02 - Action required
<i>pNetworkInfo</i>	<ul style="list-style-type: none"> • USS information from the network (0 indicates that no info was received) <ul style="list-style-type: none"> – See USSInfo for more details

Note

Technology Supported: UMTS

9.11.3.112 `typedef void(* tFNUSSDNoWaitIndication) (USSDNoWaitIndicationInfo *pNetworkInfo)`

9.11.3.113 `typedef void(* tFNUSSDRelease) (void)`

USSD releasecallback function prototype

Note

Technology Supported: UMTS

9.11.3.114 `typedef struct _transLayerInfoNotification transLayerNotification`

Contains the parameters passed for SLQSSetTransLayerInfoCallback by the device.

Parameters

<i>regInd</i>	<ul style="list-style-type: none"> • Indicates whether the transport layer is registered or not • Values: <ul style="list-style-type: none"> – 0x00 - Transport layer is not registered – 0x01 - Transport layer is registered
Generated by Doxygen	
<i>pTransLayerInfo</i>	<ul style="list-style-type: none"> • Optional parameter • See transLayerInfo for more information

Note

None

9.11.3.115 typedef struct _transNWRegInfoNotification transNWRegInfoNotification

Contains the parameters passed for SLQSSetTransNWRegInfoCallback by the device.

Parameters

<i>NWRegStat</i>	<ul style="list-style-type: none"> • provides the transport network registration information • Values: <ul style="list-style-type: none"> – 0x00 - No Service – 0x01 - In Progress – 0x02 - Failed – 0x03 - Limited Service – 0x04 - Full Service
------------------	---

Note

None

9.11.4 Enumeration Type Documentation

9.11.4.1 enum device_state_enum

Device State enumeration

- See [device_state_enum](#) for more details

Enumerator

DEVICE_STATE_DISCONNECTED
DEVICE_STATE_READY
DEVICE_STATE_BOOT

9.11.4.2 enum eQaQMIService

The QMI service information which is exposed to the application, only the services which are relevant to multiple PDP are listed in this enumeration as these are the only required services to be exposed.

Enumerator

eQA_QMI_SVC_WDS
eQA_QMI_SVC_NAS
eQA_QMI_SVC_NA

9.11.4.3 enum SMSEventType

This enumeration defines the different type of SMS events that are received

- See [SMSEventType](#) for more details

Enumerator

SMS_EVENT_MT_MESSAGE
SMS_EVENT_TRANSFER_ROUTE_MT_MESSAGE
SMS_EVENT_MESSAGE_MODE
SMS_EVENT_ETWS
SMS_EVENT_ETWS_PLMN
SMS_EVENT_SMSC_ADDRESS
SMS_EVENT_SMS_ON_IMS

9.11.5 Function Documentation

9.11.5.1 **ULONG** iSetCATEventCallback (**tFNCATEvent** *pCallback*)

9.11.5.2 **ULONG** iSetSignalStrengthCallback (**tFNSignalStrength** *pCallback*)

9.11.5.3 **ULONG** iLQSSetDUNCallInfoCallback (**tFNDUNCallInfo** *pCallback*)

9.11.5.4 **ULONG** iLQSSetSignalStrengthsCallback (**tFNSLQSSignalStrengths** *pCallback*)

9.11.5.5 **ULONG** iLQSSetWdsFirstInstEventCallback (**tFNSLQSWDSEvent** *pCallback*)

9.11.5.6 **ULONG** iLQSSetWdsSecondInstEventCallback (**tFNSLQSWDSEvent** *pCallback*)

9.11.5.7 **ULONG** iLQSSetWdsThirdInstEventCallback (**tFNSLQSWDSEvent** *pCallback*)

9.11.5.8 **ULONG** iLQSSetWdsXferStatsFirstInstCallback (**tFNSLQSWDSEvent** *pCallback*)

9.11.5.9 **ULONG** iLQSSetWdsXferStatsSecondInstCallback (**tFNSLQSWDSEvent** *pCallback*)

9.11.5.10 **ULONG** SetActivationStatusCallback (**tFNActivationStatus** *pCallback*)

Enables/disables the Activation Status callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none"> • Callback function pointer (0 - disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.11.5.11 ULONG SetCATEventCallback (tFNCATEvent pCallback, ULONG eventMask, ULONG * pErrorMask)

Enables/disables the CAT event callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none"> • Callback function pointer (0 - Disable)
<i>eventMask</i>	<ul style="list-style-type: none"> • bitmask of CAT events to register for <ul style="list-style-type: none"> – 0x00000001 - Display Text – 0x00000002 - Get In-Key – 0x00000004 - Get Input – 0x00000008 - Setup Menu – 0x00000010 - Select Item – 0x00000020 - Send SMS - Alpha Identifier – 0x00000040 - Setup Event: User Activity – 0x00000080 - Setup Event: Idle Screen Notify – 0x00000100 - Setup Event: Language Sel Notify – 0x00000200 - Setup Idle Mode Text – 0x00000400 - Language Notification – 0x00000800 - Refresh – 0x00001000 - End Proactive Session
<i>pErrorMask</i> [O↔ UT]	<ul style="list-style-type: none"> • error bitmask. Each bit set indicates the proactive command that caused the error <ul style="list-style-type: none"> – 0x00000001 - Display Text – 0x00000002 - Get In-Key – 0x00000004 - Get Input – 0x00000008 - Setup Menu – 0x00000010 - Select Item – 0x00000020 - Send SMS - Alpha Identifier – 0x00000040 - Setup Event: User Activity – 0x00000080 - Setup Event: Idle Screen Notify – 0x00000100 - Setup Event: Language Sel Notify – 0x00000200 - Setup Idle Mode Text – 0x00000400 - Language Notification – 0x00000800 - Refresh – 0x00001000 - End Proactive Session

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

Technology Supported: UMTS

Timeout: 2 seconds

9.11.5.12 ULONG SetDataCapabilitiesCallback (tFNDataCapabilities pCallback)

Enables/disables the data capabilities callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Does not require communication with the device

9.11.5.13 ULONG SetDeviceStateChangeCb (tFNDeviceStateChange pCallback)

Used by the client application to register a Callback function for Device State Change (DSC) event notifications. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• a valid function pointer to be notified of DSC events• NULL to disable DSC event notification
-----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.11.5.14 **ULONG** SetFwDldCompletionCbk (**tFNFwDldCompletion** *pCallback*)

Used by the client application to register a Callback function for a Firmware Download Completion (FDC) event notification. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• a valid function pointer to enable FDC event notification• NULL to disable FDC event notification
-----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: N/A

9.11.5.15 **ULONG** SetGPSCallback (**tFNNewGPS** *pCallback*)

Enables/disables the NMEA sentence callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.16 **ULONG** SetLocBestAvailPosCallback (**tFNBestAvailPos** *pCallback*)

Enables/disables Best Available Location callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.17 ULONG SetLocCradleMountCallback (tFNSetCradleMount *pCallback*)

Enables/disables the Cradle Mount callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.18 ULONG SetLocDeleteAssistDataCallback (tFNDelAssistData *pCallback*)

Enables/disables Delete Assist Data callback function. This API is used to receive the SUCCESS/FAILURE status of API [SLQSLOCDeAssData\(\)](#).

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

9.11.5.19 ULONG SetLocEngineStateCallback (tFNSetEngineState *pCallback*)

Sends the GPS State Information event to the control point.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.20 ULONG SetLocEventPositionCallback (tFNEventPosition pCallback)

Enables/disables the Event Position Report callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.21 ULONG SetLocEventTimeSyncCallback (tFNSetEventTimeSync pCallback)

Enables/disables the Event Time Sync callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.22 ULONG SetLocGnssSvInfoCallback (tFNGnssSvInfo pCallback)

Enables/disables the GNSS [SV](#) Info callback function. This API is used to send the satellite report to the application. The satellite reports are sent only to the application that invoked API [SLQSLOCStart\(\)](#) that generated the satellite report.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.23 ULONG SetLocInjectSensorDataCallback (tFNInjectSensorData pCallback)

Enables/disables the Inject Sensor Data callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.24 ULONG SetLocInjectTimeCallback (tFNInjectTimeStatus pCallback)

Enables/disables the Inject Time Sync Data callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.25 ULONG SetLocOpModeCallback (tFNOpMode pCallback)

Enables/disables Set Operating Mode callback function. This API is used to receive the SUCCESS/FAILURE status of API [SLQSLOCSetOpMode\(\)](#).

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

9.11.5.26 ULONG SetLocSensorStreamingCallback (tFNSensorStreaming *pCallback*)

Enables/disables the Event Sensor Streaming Ready Status callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.27 ULONG SetLocSetExtPowerConfigCallback (tFNSetExtPowerConfig *pCallback*)

Enables/disables the Set External Power Config Status callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.28 ULONG SetLURejectCallback (tFNLUReject *pCallback*)

Enables/disables the LU reject callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.29 ULONG SetMobileIPStatusCallback (tFNMobileIPStatus pCallback)

Enables/disables the Mobile IP Status callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0 - disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 2 seconds

9.11.5.30 ULONG SetNasLTECphyCalndCallback (tFNASwiLTECphyCalInfo pCallback)

Enables/disables the LTE NAS CA Info callback function.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.31 **ULONG** SetNetChangeCbk (**BYTE** *instance*, **tFNNet** *pCallback*, **ULONG** *loMark*, **ULONG** *hiMark*, **ULONG** *period*)

Used by the client application to register a Callback function for USB Transmit Queue Length Change event notifications. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>instance</i> [IN]	<ul style="list-style-type: none"> PDP instance
<i>pCallback</i> [IN]	<ul style="list-style-type: none"> a valid function pointer to be notified of the event NULL to disable the event notification
<i>loMark</i> [IN]	<ul style="list-style-type: none"> Transmit queue length smaller will trigger unthrottle event notification
<i>hiMark</i> [IN]	<ul style="list-style-type: none"> Transmit queue length larger will trigger throttle event notification
<i>period</i> [IN]	<ul style="list-style-type: none"> monitoring period in seconds, minimum 1 second

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.11.5.32 **ULONG** SetNewSMSCallback (**tFNNewSMS** *pCallback*)

Enables/disables the new SMS callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

Timeout: 2 seconds

9.11.5.33 ULONG SetNMEACallback (tFNNewNMEA pCallback)

Enables/disables the NMEA sentence callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

This API is deprecated on all MC/EM74xx firmware versions. Please use [SetLocEventPositionCallback\(\)](#)

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.34 ULONG SetOMADMStateCallback (tFNOMADMState pCallback)

Enables/disables the OMADM state callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

This API is deprecated on all MC/EM74xx firmware versions. Please use [SetSLQSOMADMAAlertCallback\(\)](#)

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• a valid function pointer to enable OMADMState notification• NULL to disable OMADMState notification
-----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

Technology Supported: CDMA
Timeout: 2 seconds

9.11.5.35 `ULONG SetPDSSStateCallback (tFNPDSSState pCallback)`

Enables/disables the PDS service state callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.36 `ULONG SetPowerCallback (tFNPower pCallback)`

Enables/disables the Operating Mode callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.37 `ULONG SetRankIndicatorCallback (tFNRankIndicator pCallback)`**9.11.5.38 `ULONG SetRFInfoCallback (tFNRFInfo pCallback)`**

Enables/disables the radio frequency information callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.39 ULONG SetRMTransferStatisticsCallback (tFNNewRMTransferStatistics pCallback)

Enables/disables the RM Transfer Statistics callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.40 ULONG SetRoamingIndicatorCallback (tFNRoamingIndicator pCallback)

Enables/disables the Roaming Indicator callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

This API is deprecated on all MC/EM74xx firmware versions. Please use [SLQSNasSysInfoCallBack\(\)](#) instead

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.41 U_LONG SetSignalStrengthCallback (tFNSignalStrength pCallback, BYTE thresholdsSize, INT8 * pThresholds)

Enables/disables the Signal Strength callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs. This API is deprecated on MC73xx/↔ EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSNasIndicationRegisterExt\(\)](#) for new firmware versions and new modules

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none"> • Callback function pointer (0-Disable)
<i>thresholdsSize</i>	<ul style="list-style-type: none"> • Number of elements threshold array contains; a maximum of five thresholds is supported; • This parameter is not used when disabling the callback.
<i>pThresholds[IN]</i>	<ul style="list-style-type: none"> • Signal threshold array for each entry (in dBm). • This parameter is not used when disabling the callback.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

The signal strength callback function is called when a threshold in the threshold array is crossed.

9.11.5.42 ULONG SetSLQSOMADMAAlertCallback (tFNSLQSOMADMAAlert pCallback)

Enables/disables the SWIOMADM network-initiated alert callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• a valid function pointer to enable SLQSOMADMAAlert notification• NULL to disable SLQSOMADMAAlert notification
----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.43 ULONG SetSLQSOMADMAAlertCallbackExt (tFNSLQSOMADMAAlert pCallback)

Enables/disables the SWIOMADM network-initiated alert callback function for SL9090 module. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• a valid function pointer to enable SLQSOMADMAAlert notification• NULL to disable SLQSOMADMAAlert notification
----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

Technology Supported: UMTS/CDMA
Device Supported: SL9090
Timeout: 2 seconds

9.11.5.44 ULONG SetUimSlotStatusChangeCallback (tFNCbkUimSlotStatusChangeInd pCallback)

Enables/disables Slot Status Change callback function.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.45 ULONG SetUSSDNotificationCallback (tFNUSSDNotification pCallback)

Enables/disables the USSDNotification callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• a valid function pointer to enable ServingSystem notification• NULL to disable ServingSystem notification
----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

Technology Supported: UMTS

Timeout: Does not require communication with device

9.11.5.46 ULONG SetUSSDNoWaitIndicationCallback (tFNUSSDNoWaitIndication pCallback)

SetUSSDNoWaitIndicationCallback

Parameters

<i>pNetworkInfo</i>	<ul style="list-style-type: none">• Data from the network.• See USSDNoWaitIndicationInfo for more details.
---------------------	---

Note

Technology Supported: UMTS
Device Supported: MC83x5

9.11.5.47 ULONG SetUSSDReleaseCallback (tFNUSSDRelease *pCallback*)

Enables/disables the USSD release callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• a valid function pointer to enable ServingSystem notification• NULL to disable ServingSystem notification
-----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

Technology Supported: UMTS
Timeout: Does not require communication with the device

9.11.5.48 ULONG SLQSNasNetworkTimeCallBack (tFNNetworkTime *pCallback*)

Enables/disables the Network Time callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds
This callback is sent when the 3GPP or 3GPP2 network sends time information to the User Equipment.

9.11.5.49 **ULONG SLQSNasSigInfo2CallBack (tFNSigInfo pCallback, setSignalStrengthInfo * pSigInfo2)**

Enables/disables the Signal Info callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none"> • Callback function pointer (0-Disable)
<i>pSigInfo2[IN]</i>	<ul style="list-style-type: none"> • Structure containing the threshold values beyond which signal information is to be reported • See setSignalStrengthInfo for more details

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

This callback is sent when the signal strength change occurs

9.11.5.50 **ULONG SLQSNasSigInfoCallBack (tFNSigInfo pCallback, sigInfo * pSigInfo)**

Enables/disables the Signal Info callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs. This callback is deprecated on MC73xx/↔ EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use callback [SLQSNasSigInfo2CallBack\(\)](#) for new firmware versions and new modules

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none"> • Callback function pointer (0-Disable)
<i>pSigInfo[IN]</i>	<ul style="list-style-type: none"> • Structure containing the threshold values beyond which signal information is to be reported • See sigInfo for more details

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

Note

Timeout: 2 seconds

This callback is sent when the signal strength change occurs

9.11.51 ULONG SLQSNasSwiOTAMessageCallback (NasSwiIndReg * req, tFNASwiOTAMsg pCallback)

Enables/disables the SLQSNasSwiOTAMessageCallback callback function. To disable the callback, provide both req and pCallback as NULL pointer to the API

Parameters

<i>req</i> [IN]	<ul style="list-style-type: none">the request to which kind of message type should be enabled, see NasSwiIndReg for details
<i>pCallback</i> [IN]	<ul style="list-style-type: none">Callback function pointer (0-Disable)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

Note

Timeout: 2 seconds

9.11.52 ULONG SLQSNasSysInfoCallBack (tFNSysInfo pCallback)

Enables/disables the Sys Info callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

This callback provides current serving system information, including registration information and system property. The serving system information of the radio interfaces specified in mode_pref are included in the response message. When any value in the sys_info message changes, an indication message is sent. Indications contain all the values for all active RATs.

9.11.5.53 ULONG SLQSNasTimerCallback (tFN NasTimer pCallback)

Enables/disables Network Timer callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i>	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.54 ULONG SLQSSetBandPreferenceCbk (tFN BandPreference pCallback)

Enables/disables the Band Preference callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• a valid function pointer to enable Band Preference Indication notification• NULL to disable Band Preference notification
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

Timeout: NA To set the band preference the API [SLQSSetBandPreference\(\)](#) should be used

9.11.5.55 ULONG SLQSSetDataSystemStatusCallback (tFNDataSysStatus pCallback)

Enables/disables the Data System Status callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.11.5.56 ULONG SLQSSetDHCPv4ClientLeaseStatusCallback (BYTE instance, tFNDHCPv4ClientLeaseStatus pCallback)

Enables/disables the DHCP Client V4 Lease Status callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>instance[IN]</i>	<ul style="list-style-type: none">• QMI instance
<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.57 ULONG SLQSSetDUNCallInfoCallback (BYTE StatsPeriod, tFNDUNCallInfo pCallback)

Enables/disables the DUN Call Info callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>StatsPeriod</i> [IN]	<ul style="list-style-type: none">• Period between reports(seconds)• 0 - Do not report• Only applicable to pTXOKBytesCount and pRXOKBytesCount parameters
<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.11.5.58 ULONG SLQSSetIMSApdpStatusCallback (tFNImsaPdpStatus pCallback)

SLQSSetIMSApdpStatusCallback

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.11.5.59 ULONG SLQSSetIMSAratStatusCallback (tFNImsaRatStatus pCallback)

SLQSSetIMSAratStatusCallback

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.11.5.60 ULONG SLQSSetIMSARegStatusCallback (tFNImsaRegStatus *pCallback*)

SLQSSetIMSARegStatusCallback

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.11.5.61 ULONG SLQSSetIMSASvcStatusCallback (tFNImsaSvcStatus *pCallback*)

SLQSSetIMSASvcStatusCallback

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.11.5.62 ULONG SLQSSetIMSSMSConfigCallback (tFNImSMSConfig *pCallback*)

Enables/disables the SMS Config callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.63 ULONG SLQSSetIMSUserConfigCallback (tFNImUserConfig *pCallback*)

Enables/disables the User Config callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.64 ULONG SLQSSetIMSVoIPConfigCallback (tFNImSVoIPConfig *pCallback*)

Enables/disables the VoIP Config callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.65 ULONG SLQSSetLocInjectPositionCallback (tFNInjectPosition *pCallback*)

Enables/disables Inject Position callback function. This API is used to receive the SUCCESS/FAILURE status of API [SLQSLOCInjectPosition\(\)](#).

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

9.11.5.66 ULONG SLQSSetLocInjectUTCTimeCallback (tFNInjectUTCTime *pCallback*)

Enables/disables Inject UTC Time callback function. This API is used to receive the SUCCESS/FAILURE status of API [SLQSLOCInjectUTCTime\(\)](#).

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

9.11.5.67 ULONG SLQSSetModemTempCallback (tFNModemTempInfo *pCallback*)

Enables/disables the Modem Temperature information callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.11.5.68 ULONG SLQSSetPacketSrvStatusCallback (tFNPacketSrvState *pCallback*)

Enables/disables the session state callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - disable)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: none; does not require communication with the device

9.11.5.69 ULONG SLQSSetQosEventCallback (BYTE *instance*, tFNSLQSQOSEvent *pCallback*)

Enables/disables the QoS event callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs. This callback provide extra information regarding the QoS flow state

Parameters

in	<i>instance</i>	<ul style="list-style-type: none"> QMI instance
in	<i>pCallback</i>	<ul style="list-style-type: none"> Callback function pointer (0 - disable)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

maximum number of tx/rx filters supported is 25 (pTxQFilter/pRxQFilter)

9.11.5.70 ULONG SLQSSetQosNWStatusCallback (tFNQosNWStatus *pCallback*)

Enables/disables the QoS event callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs. This callback provide extra information regarding the QoS Network supports status

Parameters

in	<i>pCallback[IN]</i>	<ul style="list-style-type: none"> Callback function pointer (0 - disable)
----	----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.11.5.71 ULONG SLQSSetQosPriEventCallback (tFNQosPriEvent *pCallback*)

Enables/disables the QoS event callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs. This callback provide extra information regarding the QoS Primary flow event

Parameters

in	<i>pCallback</i>	<ul style="list-style-type: none">• Callback function pointer (0 - disable)
----	------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

9.11.5.72 ULONG SLQSSetQosStatusCallback (BYTE *instance*, tFNQosStatus *pCallback*)

Enables/disables the QoS event callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs. This callback provide extra information regarding the QoS status

Parameters

in	<i>instance</i>	<ul style="list-style-type: none">• QMI instance
in	<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0 - disable)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.11.5.73 ULONG SLQSSetRegMgrConfigCallback (tFNlmsRegMgrConfig pCallback)

Enables/disables the Reg Mgr Config callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.74 ULONG SLQSSetSDKTerminatedCallback (tFNSDKTerminated pCallback)

Used by the client application to register a Callback function for SDK terminated event notifications. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• a valid function pointer to be notified of SWI events• NULL to disable SWI event notification
----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: N/A

Device Supported: N/A

Timeout: N/A

The following signals will trigger this callback:

2 INT	4 ILL	5 TRAP	6 ABRT	7 BUS
8 FPE	11 SEGV	13 PIPE	15 TERM	31 SYS

9.11.5.75 ULONG SLQSSetServingSystemCallback (tFNServingSystem pCallback)

Enables/disables the Serving System callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

This API is deprecated on all MC/EM74xx firmware versions. Please use [SLQSNasSysInfoCallBack\(\)](#)

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• a valid function pointer to enable ServingSystem notification• NULL to disable ServingSystem notification
-----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.76 ULONG SLQSSetSessionStateCallback (tFNSLQSSessionState pCallback)

Enables/disables the session state callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs. This callback provide extra information regarding the multiple PDP interface

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0 - disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: none; does not require communication with the device

9.11.5.77 ULONG SLQSSetSignalStrengthsCallback (tFN~~SLQSS~~SignalStrengths pCallback, struct SLQSSignalStrengthsIndReq * pSLQSSignalStrengthsIndReq)

Enables/disables the Received Signal Strength Information callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs. This API is same as API SetSignalStrengthsCallback() except providing more information of signal such as ECIO, SNR etc. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all E↔M74xx firmware versions. Please use API [SLQSNasIndicationRegisterExt\(\)](#) for new firmware versions and new modules

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none"> • Callback function pointer (0-Disable)
<i>pSLQSSignal↔StrengthsIndReq</i>	<ul style="list-style-type: none"> • See SLQSSignalStrengthsIndReq for more information • This parameter is not used when disabling the callback.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

The signal strength callback function is called when a threshold in the threshold array is crossed.

9.11.5.78 ULONG SLQSSetSIPConfigCallback (tFN~~Im~~sSIPConfig pCallback)

Enables/disables the SIP Config callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none"> • Callback function pointer (0-Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.79 ULONG SLQSSetSMSEventCallback (tFNSMSEvents pCallback)

Enables/disables the events related to SMS callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

9.11.5.80 ULONG SLQSSetSwiGetResetInfoCallback (tFNResetInfo pCallback)

Reset Info callback.

Parameters

<i>pCallback</i>	<ul style="list-style-type: none">• See tFNResetInfo for more information.
------------------	--

9.11.5.81 ULONG SLQSSetSwiHDRPersCallback (tFNHDRPersonaity pCallback)

Enables/disables the HDR Personality callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 5 seconds

9.11.5.82 ULONG SLQSSetSysSelectionPrefCallBack (tFNSysSelectionPref pCallback)

Enables/disables the System Selection Preference callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• a valid function pointer to enable System Selection Preference Indication notification• NULL to disable Band Preference notification
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

Timeout: 2 seconds
To set the system selection preferences the API [SLQSSetSysSelectionPref\(\)](#) should be used

9.11.5.83 ULONG SLQSSetTransLayerInfoCallback (tFNtransLayerInfo pCallback)

Enables/disables the Transport Layer information callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.84 ULONG SLQSSetTransNWRegInfoCallback (tFNtransNWRegInfo pCallback)

Enables/disables the Transport Network Registration information callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.85 ULONG SLQSSetWdsEventCallback (tFNSLQSWDSEvent pCallback, BYTE interval, BYTE instanceid, BYTE ipfamily)

Enables/disables the WDS event callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs. This callback provide extra information regarding the multiple PDP interface. Transfer statistic are reported only when changed.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0 - disable)
<i>interval</i>	<ul style="list-style-type: none">• Interval in seconds.• ignored when disabling, should be non-zero when enabling• period only affect transfer statistic attributes

<i>instanceid</i>	<ul style="list-style-type: none"> • PDP instance id 0 - First PDP instance 1 - Second PDP instance 2 - Third PDP instance
<i>ipfamily</i>	<ul style="list-style-type: none"> • 4 for an IPv4 data session • 6 for an IPv6 data session • 7 for an IPv4v6 data session

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds Currently 3 PDP instances are supported in device. user of this callback can subscribe by passing instanceid of particular instance. All PDP instance can be subscribed by passing instanceid sequentially.

9.11.5.86 **ULONG** SLQSSetWdsTransferStatisticCallback (**tFNSLQSWDSEvent** *pXferStatsCb*, **BYTE** *interval*, **BYTE** *instanceid*, **BYTE** *ipfamily*)

Enables/disables the WDS transfer statistic callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs. This callback provide extra information regarding the multiple PDP interface. Transfer statistic are reported only when changed.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none"> • Callback function pointer (0 - disable)
<i>interval</i>	<ul style="list-style-type: none"> • Interval in seconds. • ignored when disabling, should be non-zero when enabling • period only affect transfer statistic attributes
<i>instanceid</i>	<ul style="list-style-type: none"> • PDP instance id 0 - First PDP instance 1 - Second PDP instance 2 - Third PDP instance
<i>ipfamily</i>	<ul style="list-style-type: none"> • 4 for an IPv4 data session • 6 for an IPv6 data session • 7 for an IPv4v6 data session

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds Currently 3 PDP instances are supported in device. User of this callback can subscribe by passing instance id of particular instance. All PDP instance can be subscribed by passing instance id sequentially.

9.11.5.87 **ULONG SLQSTmdMitigationLvIRptCallback (TmdMitigationLvIRpt * req, tFNMitilvIRpt pCallback)**

Thermal Mitigation callback.

Parameters

<i>req</i>	<ul style="list-style-type: none"> See TmdMitigationLvIRpt for more information.
<i>pCallback</i>	<ul style="list-style-type: none"> See tFNMitilvIRpt for more information.

9.11.5.88 **ULONG SLQSUIMSetRefreshCalBack (tFNUIMRefresh pCallback)**

Enables/disables the UIM refresh callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none"> a valid function pointer to enable UIM Refresh Indication notification NULL to disable Band Preference notification
----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

Timeout: 2 seconds

[SLQSUIMRefreshRegister\(\)](#) API should be invoked prior to the invocation of the callback for the events to be registered.

9.11.5.89 ULONG SLQSUIMSetStatusChangeCallBack (tFNUIMStatusChangeInfo *pCallback*)

Enables/disables the UIM Status Change Callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• a valid function pointer to enable UIM Status Change Indication notification• NULL to disable Band Preference notification
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

Timeout: 2 seconds

9.11.5.90 ULONG SLQSVoiceInfoRecCallback (tFNInfoRec *pCallback*)

Enables/disables the Voice information Record callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs. (Applicable only for 3GPP2)

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA

Timeout: 2 seconds

9.11.5.91 ULONG SLQSVoiceSetAllCallStatusCallBack (tFNAIICallStatus pCallback)

Enables/disables Voice Call Status Callback function. User can subscribe this callback get the call state change notifications. eg:- Call originated, connected, or ended. Whenever there is a change in the call information, there will be a indication with the information.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0 - Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.11.5.92 ULONG SLQSVoiceSetDTMFEventCallBack (tFNDTMFEvent pCallback)

Enables/disables the DTMF Event callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback[IN]</i>	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

This callback communicates that a DTMF event has been received.

9.11.5.93 ULONG SLQSVoiceSetOTASPStatusCallBack (tFNOTASPStatus pCallback)

Enables/disables OTASP(Over-The-Air Service Provisioning) or OTAPA(Over-The-Air Parameter Administration) event CallBack Function (applicable only for 3GPP2). The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• a valid function pointer to enable OTASP or OTAPA event Indication notification• NULL to disable OTASP or OTAPA event, Indication notification
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

Technology Supported: CDMA

Timeout: 10 seconds

This indication communicates the occurrence of an OTASP or OTAPA event. This indication is only applicable for 3GPP2 devices.

9.11.5.94 ULONG SLQSVoiceSetPrivacyChangeCallBack (tFNPrivacyChange pCallback)

Enables/disables the voice privacy change callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA

Timeout: 2 seconds

This callback communicates a change in the voice privacy of a call. This is applicable only in 3GPP2 devices.

9.11.5.95 ULONG SLQSVoiceSetSUPSCallBack (tFNSUPSInfo pCallback)

Enables/disables the SUPS callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: GSM

Timeout: 2 seconds

This callback notifies clients about the modem-originated supplementary service requests and the responses received from the network.

9.11.5.96 ULONG SLQSVoiceSetSUPSNotificationCallback (tFNSUPSNotification pCallback)

Enables/disables the supplementary service notification callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.97 ULONG SLQSWmsAsyncRawSendCallBack (tFNAsyncRawSend pCallback)

Enables/disables the SLQSWmsAsyncRawSendCallBack callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.11.5.98 ULONG SLQSWmsMemoryFullCallBack (tFNMemoryFull pCallback)

Enables/disables the event related to memory full status callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.11.5.99 ULONG SLQSWmsMessageWaitingCallBack (tFNMessageWaiting pCallback)

Enables/disables the event related to message waiting information callback function. The most recent successfully subscribed callback function will be the only function that is invoked when the corresponding event occurs.

Parameters

<i>pCallback</i> [IN]	<ul style="list-style-type: none">• Callback function pointer (0-Disable)
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.12 qaGobiApiDcs.h File Reference

Device Connectivity Service API function prototypes.

Data Structures

- struct [DcsUsbPortNames](#)
- struct [QosMap](#)
- struct [NetStats](#)

Macros

- #define [LEN](#) 10
- #define [PORTNAM_LEN](#) 32

Functions

- [ULONG QCWWAN2kEnumerateDevices](#) ([BYTE](#) *pDevicesSize, [BYTE](#) *pDevices)
- [ULONG QCWWAN2kConnect](#) ([CHAR](#) *pDeviceID, [CHAR](#) *pDeviceKey)
- [ULONG QCWWANDisconnect](#) ()
- [ULONG QCWWAN2kGetConnectedDeviceID](#) ([ULONG](#) deviceIDSize, [CHAR](#) *pDeviceID, [ULONG](#) deviceKeySize, [CHAR](#) *pDeviceKey)
- [ULONG QCWWANEnumerateDevices](#) ([BYTE](#) *pDevicesSize, [BYTE](#) *pDevices)
- [ULONG QCWWANConnect](#) ([CHAR](#) *pDeviceID, [CHAR](#) *pDeviceKey)
- [ULONG SetSDKImagePath](#) ([LPCSTR](#) pPath)
- [ULONG SLQSGetUsbPortNames](#) (struct [DcsUsbPortNames](#) *pUsbPortNames)
- [ULONG SLQSStart_AVAgent](#) ([BYTE](#) modem_index)
- [ULONG SLQSStart](#) ([BYTE](#) modem_index, [CHAR](#) *sn)
- [ULONG SLQSKillSDKProcess](#) ()
- [ULONG SLQSGetDeviceMode](#) ([BYTE](#) *pDeviceMode)
- [ULONG SLQSStartSrv](#) ([BYTE](#) action, [BYTE](#) mask)
- [ULONG SLQSSetLoggingMask](#) ([BYTE](#) mask)
- [ULONG SLQSGetNetStatistic](#) (struct [NetStats](#) *pNetStatistic, [BYTE](#) instance)

9.12.1 Detailed Description

Device Connectivity Service API function prototypes.

9.12.2 Macro Definition Documentation

9.12.2.1 #define LEN 10

9.12.2.2 #define PORTNAM_LEN 32

This structure contains the SLQSGetUsbPortNames Information

Parameters

<i>AtCmdPort</i>	[OUT] <ul style="list-style-type: none">Name of AT command port
<i>NmeaPort</i>	[OUT] <ul style="list-style-type: none">Name of NMEA port
<i>DmPort</i>	[OUT] <ul style="list-style-type: none">Name of DM port

Note

Technology Supported: UMTS/CDMA
Device Supported: MC83x5, MC7700/10/50
Timeout: 2 seconds
[Port](#) names are limited to 32 characters.

9.12.3 Function Documentation**9.12.3.1 ULONG QCWWAN2kConnect (CHAR * *pDeviceID*, CHAR * *pDeviceKey*)**

Connects the Connection Manager API to the first detected QC WWAN device. This function MUST be called after QCWWAN2kEnumerateDevices has been called.

Parameters

<i>pDeviceID</i> [IN]	<ul style="list-style-type: none">Device path pertaining to the device for which the API is being invoked e.g. /dev/qcqm0.
<i>pDeviceKey</i> [IN]	<ul style="list-style-type: none">Device key pertaining to the device for which the API is being invoked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_ERR_NO_DEVICE otherwise

Note

Timeout: 2 seconds

9.12.3.2 ULONG QCWWAN2kEnumerateDevices (BYTE * *pDevicesSize*, BYTE * *pDevices*)

Enumerates the QC WWAN devices currently attached to the host. This API MUST be called before any other API.

Parameters

<i>pDeviceSize</i> [I↔ N/OUT]	<ul style="list-style-type: none"> Upon input, maximum number of elements that the device array can contain. Upon successful output, actual number of elements in the device array.
<i>pDevices</i> [IN/O↔ UT]	<ul style="list-style-type: none"> Device array; array elements are structures with the following elements: CHAR deviceId[256] - Device path (e.g. /dev/qcqmio) CHAR deviceKey[16] - Device key stored in the device (e.g. A1000004B01051)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_ERR_NO_DEVICE on otherwise

Note

Timeout: 2 seconds

9.12.3.3 ULONG QCWWAN2kGetConnectedDeviceID (ULONG deviceIdSize, CHAR * pDeviceID, ULONG deviceKeySize, CHAR * pDeviceKey)

Returns the device ID and device key of the currently connected QC WWAN device.

Parameters

<i>deviceIdSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that the device ID array can contain.
<i>pDeviceID</i> [OUT]	<ul style="list-style-type: none"> Device path string
<i>deviceKeySize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that the device key array can contain.
<i>pDeviceKey</i> [O↔ UT]	<ul style="list-style-type: none"> Device key string

Returns

eQCWWAN_ERR_NONE if device found, eQCWWAN_ERR_NO_DEVICE otherwise

Note

Timeout: 2 seconds

9.12.3.4 ULONG QCWWANConnect (CHAR * *pDeviceID*, CHAR * *pDeviceKey*)

Enumerates the QC WWAN devices currently attached to the host. This API MUST be called before any other API.

Parameters

<i>pDeviceID</i> [IN]	<ul style="list-style-type: none">Device path pertaining to the device for which the API is being invoked e.g. /dev/qcqm10.
<i>pDeviceKey</i> [IN]	<ul style="list-style-type: none">Device key pertaining to the device for which the API is being invoked

Returns

eQCWWAN_ERR_NONE if device found, eQCWWAN_ERR_NO_DEVICE otherwise

Note

Timeout: 2 seconds

This API is deprecated; use [QCWWAN2kConnect\(\)](#) instead

9.12.3.5 ULONG QCWWANDisconnect ()

Disconnects the Connection Manager API from a previously connected QC device. This function de-registers all the callback functions that have been registered.

Parameters

<i>none</i>

Returns

eQCWWAN_ERR_NONE

Note

Timeout: 2 seconds

9.12.3.6 ULONG QCWWANEnumerateDevices (BYTE * *pDevicesSize*, BYTE * *pDevices*)

Enumerates the QC WWAN devices currently attached to the host. This API is deprecated; use [QCWWAN2kEnumerateDevices\(\)](#) instead.

Parameters

<i>pDeviceSize</i> [↔ N/OUT]	<ul style="list-style-type: none"> Upon input, maximum number of elements that the device array can contain. Upon successful output, actual number of elements in the device array.
<i>pDevices</i> [IN/↔ OUT]	<ul style="list-style-type: none"> Device array; array elements are structures with the following elements: CHAR deviceId[256] - Device path (e.g. /dev/qcqmio) CHAR deviceKey[16] - Device key stored in the device

Returns

eQCWWAN_ERR_NONE

Note

Timeout: 2 seconds

This API must be called prior to any other APIs.

9.12.3.7 ULONG SetSDKImagePath (LPCSTR *pPath*)

Set the location of the SLQS executable

Parameters

<i>pPath</i> [IN]	- Pointer to fully qualified path of SLQS executable (includes the executable file's name)
-------------------	--

Returns

eQCWWAN_ERR_NONE

Note

Timeout: None

9.12.3.8 ULONG SLQSGetDeviceMode (BYTE * *pDeviceMode*)

Returns the Device Mode

Parameters

<i>pDeviceMode</i> [↔ OUT]	<ul style="list-style-type: none"> Pointer to SLQS Device Mode of type eDevState
-------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.12.3.9 ULONG SLQSGetNetStatistic (struct NetStats * pNetStatistic, BYTE instance)

Returns the usbnet statistics for a particular PDN.

Parameters

	<i>pNetStatistic</i> [↔ <i>OUT</i>]	<ul style="list-style-type: none">• Pointer to the structure NetStats which the value of every member is to be retrieved
<i>in</i>	<i>instance</i>	<ul style="list-style-type: none">• PDP Instance id

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.12.3.10 ULONG SLQSGetUsbPortNames (struct DcsUsbPortNames * pUsbPortNames)

Returns the Usb [Port](#) Names currently in use.

Parameters

<i>pUsbPort</i> ↔ <i>Names</i> [<i>OUT</i>]	<ul style="list-style-type: none">• Pointer to SLQS USB Port Names Information
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.12.3.11 ULONG SLQSKillSDKProcess ()

Kill the SDK process

Parameters

<i>none</i>	
-------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: N/A

This API useful if the application was started with non-root privileges as subsequent attempt to start any application will fail because the SDK requires root permission to access /dev/qcqm_i device special files.

9.12.3.12 ULONG SLQSSetLoggingMask (BYTE mask)

Limit Syslog messages according to the Mask provided by user

Parameters

<i>mask</i>	<ul style="list-style-type: none">• Mask 0x01: disable all log• Mask 0xFF: enable all log
-------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.12.3.13 ULONG SLQSStart (BYTE *modem_index*, CHAR * *sn*)

Create the SDK process and IPC sockets for the Application and SDK processes to communicate over.

Parameters

in	<i>modem_index</i>	<ul style="list-style-type: none">• 0: first modem detected• 1: second modem detected• 2: third modem detected• ...• 7: seventh modem detected
in	<i>sn</i>	This field is optional, it can be serial number or usb path for multiple modem feature, it can be retrieved from sytem command "dmesg" when specified, the modem_index will be mapping to sn or usb path Please set to NULL when not used

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: N/A

9.12.3.14 ULONG SLQSStart_AVAgent (BYTE *modem_index*)

Create IPC sockets for AirVantage Agent and SDK processes to communicate over

Parameters

<i>none</i>

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: N/A

9.12.3.15 ULONG SLQSSStartSrv (BYTE action, BYTE mask)

Registers/deregisters for service with unsolicited notifications

Parameters

<i>action, 1</i>	for register, 0 for deregister
<i>mask</i>	<ul style="list-style-type: none">• Bit mask for unsolicited notifications<ul style="list-style-type: none">– Bit0 - WDS– Bit1 - NAS– Bit2 - PDS– Bit3 - VOICE

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds API is useful to register for the services which supports unsolicited notifications. Registration/deregistration can be done by using parameter action if action is set then the mask (set bits) will be used for registering service and if action is "0" mask(set bits) will be used to deregister services. For example : bit mask 0x03 - Registers for services WDS and NAS if action is "1" and deregisters WDS and NAS if action is "0".

9.13 qaGobiApiDms.h File Reference

Device Management Service API function prototypes.

Data Structures

- struct [serialNumbersInfo](#)
- struct [ERIFileparams](#)
- struct [custFeaturesInfo](#)
- struct [custFeaturesSetting](#)
- struct [dmsCurrentPRLInfo](#)
- struct [FactorySequenceNumber](#)
- struct [FirmwareUpdatStat](#)
- struct [USBCompParams](#)
- struct [USBCompConfig](#)
- struct [CrashInfo](#)
- struct [CrashInfoParams](#)
- struct [_SLQSSwiGetHostDevInfoParams](#)
- struct [_SLQSSwiSetHostDevInfoParams](#)
- struct [_SLQSSwiGetOSInfoParams](#)
- struct [_SLQSSwiSetOSInfoParams](#)
- struct [_SLQSSwiGetSerialNoExtParams](#)
- struct [setCustomSettingV2](#)
- struct [getCustomInput](#)
- struct [custSettingInfo](#)
- struct [custSettingList](#)
- struct [getCustomFeatureV2](#)
- struct [getDyingGaspCfg](#)
- struct [setDyingGaspCfg](#)
- struct [getDyingGaspStatistics](#)
- struct [dmsIndicationRegisterReq](#)
- struct [dmsSwiGetResetInfo](#)
- struct [BandCapabilityResp](#)

Macros

- #define [MAX_FSN_LENGTH](#) 255
- #define [MAX_CUST_ID_LEN](#) 64
- #define [MAX_CUST_VALUE_LEN](#) 8
- #define [MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH](#) 160
- #define [MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH](#) 20

Typedefs

- typedef struct [serialNumbersInfo](#) serialNumbersInfo
- typedef struct [ERIFileparams](#) ERIFileparams
- typedef struct [custFeaturesInfo](#) custFeaturesInfo
- typedef struct [custFeaturesSetting](#) custFeaturesSetting
- typedef struct [dmsCurrentPRLInfo](#) dmsCurrentPRLInfo
- typedef struct [_SLQSSwiGetHostDevInfoParams](#) SLQSSwiGetHostDevInfoParams
- typedef struct [_SLQSSwiSetHostDevInfoParams](#) SLQSSwiSetHostDevInfoParams
- typedef struct [_SLQSSwiGetOSInfoParams](#) SLQSSwiGetOSInfoParams
- typedef struct [_SLQSSwiSetOSInfoParams](#) SLQSSwiSetOSInfoParams
- typedef struct [_SLQSSwiGetSerialNoExtParams](#) SLQSSwiGetSerialNoExtParams

Functions

- [ULONG GetManufacturer](#) (BYTE stringSize, CHAR *pString)
- [ULONG GetModelID](#) (BYTE stringSize, CHAR *pString)
- [ULONG GetFirmwareRevision](#) (BYTE stringSize, CHAR *pString)
- [ULONG GetFirmwareRevisions](#) (BYTE amssSize, CHAR *pAMSSString, BYTE bootSize, CHAR *pBootString, BYTE priSize, CHAR *pPRIString)
- [ULONG GetPRLVersion](#) (WORD *pPRLVersion)
- [ULONG GetIMSI](#) (BYTE stringSize, CHAR *pString)
- [ULONG GetSerialNumbers](#) (BYTE esnSize, CHAR *pESNString, BYTE imeiSize, CHAR *pIMEIString, BYTE meidSize, CHAR *pMEIDString)
- [ULONG SLQSGetSerialNumbers](#) (serialNumbersInfo *pSerialNumbersInfo)
- [ULONG GetHardwareRevision](#) (BYTE stringSize, CHAR *pString)
- [ULONG GetNetworkTime](#) (ULONGLONG *pTimeStamp, ULONG *pTimeSource)
- [ULONG UIMSetPINProtection](#) (ULONG id, ULONG bEnable, CHAR *pValue, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)
- [ULONG UIMUnblockPIN](#) (ULONG id, CHAR *pPUKValue, CHAR *pNewValue, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)
- [ULONG UIMVerifyPIN](#) (ULONG id, CHAR *pValue, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)
- [ULONG UIMChangePIN](#) (ULONG id, CHAR *pOldValue, CHAR *pNewValue, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)
- [ULONG GetVoiceNumber](#) (BYTE voiceNumberSize, CHAR *pVoiceNumber, BYTE minSize, CHAR *pMIN)
- [ULONG SetPower](#) (ULONG powerMode)
- [ULONG GetPower](#) (ULONG *pPowerMode)
- [ULONG UIMGetControlKeyStatus](#) (ULONG id, ULONG *pStatus, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)
- [ULONG UIMGetICCID](#) (BYTE stringSize, CHAR *pString)
- [ULONG UIMGetPINStatus](#) (ULONG id, ULONG *pStatus, ULONG *pVerifyRetriesLeft, ULONG *pUnblockRetriesLeft)
- [ULONG GetOfflineReason](#) (ULONG *pReasonMask, ULONG *pbPlatform)
- [ULONG UIMSetControlKeyProtection](#) (ULONG id, ULONG status, CHAR *pValue, ULONG *pVerifyRetriesLeft)
- [ULONG UIMUnblockControlKey](#) (ULONG id, CHAR *pValue, ULONG *pUnblockRetriesLeft)
- [ULONG GetDeviceCapabilities](#) (ULONG *pMaxTXChannelRate, ULONG *pMaxRXChannelRate, ULONG *pDataServiceCapability, ULONG *pSimCapability, ULONG *pRadiofacesSize, BYTE *pRadiofaces)
- [ULONG ResetToFactoryDefaults](#) (CHAR *pSPC)
- [ULONG ValidateSPC](#) (CHAR *pSPC)
- [ULONG ActivateAutomatic](#) (CHAR *pActivationCode)
- [ULONG SLQSGetERIFile](#) (ERIFileparams *pERIFileparams)
- [ULONG GetActivationState](#) (ULONG *pActivationState)
- [ULONG SLQSUIMGetState](#) (ULONG *pUIMState)
- [ULONG SLQSGetBandCapability](#) (ULONGLONG *pBandCapability)
- [ULONG SLQSGetCustFeatures](#) (custFeaturesInfo *pCustFeaturesInfo)
- [ULONG SLQSSetCustFeatures](#) (custFeaturesSetting *pCustFeaturesSetting)
- [ULONG SLQSGetCurrentPRLInfo](#) (dmsCurrentPRLInfo *pCurrentPRLInfo)
- [ULONG SLQSSwiGetFSN](#) (FactorySequenceNumber *pFSNumber)
- [ULONG SLQSSwiGetFwUpdateStatus](#) (FirmwareUpdatStat *pFirmwareUpdatStat)
- [ULONG SLQSSwiGetUSBComp](#) (USBCompParams *pUSBCompParams)
- [ULONG SLQSSwiSetUSBComp](#) (USBCompConfig *pUSBCompConfig)
- [ULONG SLQSSwiGetCrashInfo](#) (BYTE *pClear, CrashInfoParams *pCrashInfoParams)
- [ULONG SLQSSwiGetCrashAction](#) (BYTE *pDevCrashState)
- [ULONG SLQSSwiSetCrashAction](#) (BYTE crashActionParams)
- [ULONG SLQSSwiGetHostDevInfo](#) (SLQSSwiGetHostDevInfoParams *pGetHostDevInfoParams)
- [ULONG SLQSSwiSetHostDevInfo](#) (SLQSSwiSetHostDevInfoParams *pSetHostDevInfoParams)

- [ULONG SLQSSwiGetOSInfo](#) ([SLQSSwiGetOSInfoParams](#) *pParams)
- [ULONG SLQSSwiSetOSInfo](#) ([SLQSSwiSetOSInfoParams](#) *pParams)
- [ULONG SLQSSwiGetSerialNoExt](#) ([SLQSSwiGetSerialNoExtParams](#) *pParams)
- [ULONG SLQSSetCustFeaturesV2](#) ([setCustomSettingV2](#) *pSetCustSetting)
- [ULONG SLQSGetCustFeaturesV2](#) ([getCustomFeatureV2](#) *pGetCustomFeatureV2)
- [ULONG SLQSSwiGetDyingGaspCfg](#) ([getDyingGaspCfg](#) *pConfig)
- [ULONG SLQSSwiSetDyingGaspCfg](#) ([setDyingGaspCfg](#) *pConfig)
- [ULONG SLQSSwiGetDyingGaspStatistics](#) ([getDyingGaspStatistics](#) *pStatistics)
- [ULONG SLQSSwiClearDyingGaspStatistics](#) ()
- [ULONG SLQSDmsSwtIndicationRegister](#) ([dmsIndicationRegisterReq](#) *pIndicationRegisterReq)
- [ULONG SLQSDmsSwtGetResetInfo](#) ([dmsSwtGetResetInfo](#) *pGetResetInfoResp)
- [ULONG SLQSGetBandCapabilities](#) ([BandCapabilityResp](#) *pBandCapability)

9.13.1 Detailed Description

Device Management Service API function prototypes.

9.13.2 Macro Definition Documentation

9.13.2.1 `#define MAX_CUST_ID_LEN 64`

9.13.2.2 `#define MAX_CUST_VALUE_LEN 8`

9.13.2.3 `#define MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH 160`

9.13.2.4 `#define MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH 20`

9.13.2.5 `#define MAX_FSN_LENGTH 255`

9.13.3 Typedef Documentation

9.13.3.1 `typedef struct custFeaturesInfo custFeaturesInfo`

This structure contains current settings of custom features

Parameters

<i>GpsEnable</i> [<i>O↔UT</i>]	<ul style="list-style-type: none"> • describes if GPS is enabled or disabled • values: <ul style="list-style-type: none"> – 0x00 - GPS is disabled – 0x01 - GPS is enabled • function SLQSGetCustFeatures() returns a default value FFFFFFFF if no value is returned by the modem
----------------------------------	---

<p><i>pDisableIMSI</i>[↔ OUT]</p>	<ul style="list-style-type: none"> • optional 1 byte parameter • describes if IMSI display is enabled or disabled • values: <ul style="list-style-type: none"> – 0x00 - Allow display of IMSI – 0x01 - Do not display IMSI • function SLQSGetCustFeatures() returns a default value FF if no value is returned by the modem
<p><i>pIPFam</i>↔ Support[OUT]</p>	<ul style="list-style-type: none"> • optional 2 byte BitMask • bitmask representing the IP families supported • values: <ul style="list-style-type: none"> – 0x01 - IPv4 – 0x02 - IPv6 – 0x04 - IPv4v6 • function SLQSGetCustFeatures() returns a default value FFFF if no value is returned by the modem
<p><i>pRMAuto</i>↔ Connect[OUT]</p>	<ul style="list-style-type: none"> • optional 1 byte parameter • QMI Mode RM Net Auto Connect Support • values: <ul style="list-style-type: none"> – 0x00 - Not Supported – 0x01 - Supported • function SLQSGetCustFeatures() returns a default value FF if no value is returned by the modem
<p><i>pGPSSel</i>[OUT]</p>	<ul style="list-style-type: none"> • optional 1 byte parameter • GPS Antenna Select • values: <ul style="list-style-type: none"> – 0x00 - Dedicated GPS Port – 0x01 - GPS Rx over AUX Port – 0x02 - GPS Rx over dedicated GPS port with no bias voltage applied • function SLQSGetCustFeatures() returns a default value FF if no value is returned by the modem
<p><i>pSMSSupport</i>[↔ OUT]</p>	<ul style="list-style-type: none"> • optional 1 byte parameter • SMS support • values: <ul style="list-style-type: none"> – 0x00 - Not supported – 0x01 - supported • Used to determine whether or not to hide SMS from user • function SLQSGetCustFeatures() returns a default value FF if no value is returned by the modem. In this case assume, SMS is supported.

<i>plsVoice</i> ↔ <i>Enabled[OUT]</i>	<ul style="list-style-type: none"> • optional 1 byte parameter • Voice support • values: <ul style="list-style-type: none"> – 0x00 - Enable voice on both AT and QMI interface (default) – 0x01 - Reserved – 0x02 - Disable voice on both AT and QMI interface
<i>pDHCPRelay</i> ↔ <i>Enabled[OUT]</i>	<ul style="list-style-type: none"> • optional 1 byte parameter • DHCP Relay support • values: <ul style="list-style-type: none"> – 0x00 - Disable DHCP relay – 0x01 - Enable DHCP relay
<i>pGPSLPM[OUT]</i>	<ul style="list-style-type: none"> • optional 1 byte parameter • GPSLPM support • values: <ul style="list-style-type: none"> – 0x00 - Enable GPS in Low Power Mode – 0x01 - Disable GPS in Low Power Mode

9.13.3.2 typedef struct custFeaturesSetting custFeaturesSetting

This structure contains settings to be used for custom features

Parameters

<i>pGPSSel</i>	<ul style="list-style-type: none"> • optional 1 byte parameter • GPS Antenna Select • values: <ul style="list-style-type: none"> – 0x00 - Dedicated GPS Port – 0x01 - GPS Rx over AUX Port – 0x02 - GPS Rx over dedicated GPS port with no bias voltage applied
<i>pGPSEnable</i>	<ul style="list-style-type: none"> • optional 4 byte parameter • GPS Enable/Disable • values: The value of 7 least significant bits: <ul style="list-style-type: none"> – 0 - Disabled – 1 - MT & MO enabled – 2 - MO enabled – 3 - MT enabled – 4 - MT & MO enabled if GPS_DISABLE pin is not asserted – 5 - MO GPS enabled if GPS_DISABLE pin is not asserted – 6 - MT GPS enabled if GPS_DISABLE pin is not asserted

Note

Only MC7750 3.5.x firmware supports above 0x04, 0x05 and 0x06 settings. To disable GLONASS, set the most significant bit - 0x80. This setting is only applicable if GPS is not Disabled.

Parameters

<i>plsVoiceEnabled</i>	<ul style="list-style-type: none"> • optional 1 byte parameter • voice enabled/disabled • values: <ul style="list-style-type: none"> – 0 - Enable voice on both AT and QMI interface (default) – 1 - Reserved – 2 - Disable voice on both AT and QMI interface
<i>pDHCPRelay</i> <i>Enabled</i>	<ul style="list-style-type: none"> • optional 1 byte parameter • DHCPRELAYENABLE support • values: <ul style="list-style-type: none"> – 0 - Disable DHCP relay – 1 - Enable DHCP relay
<i>pGPSLPM</i>	<ul style="list-style-type: none"> • optional 1 byte parameter • GPSLPM support • values: <ul style="list-style-type: none"> – 0 - Enable GPS in Low Power Mode – 1 - Disable GPS in Low Power Mode

9.13.3.3 typedef struct dmsCurrentPRLInfo dmsCurrentPRLInfo

This structure contains GetCurrentPRLInfo response parameter

Parameters

<i>pPRLVersion</i> <i>OUT</i>	<ul style="list-style-type: none"> - Optional • PRL version of device.
<i>pPRLPreference</i>	<ul style="list-style-type: none"> [OUT]- Optional • PRL Preference <ul style="list-style-type: none"> – 0 - Unset – 1 - Set

9.13.3.4 typedef struct ERIFileparams ERIFileparams

This structure contains Extended Roaming Indicator(ERI) file parameters

Parameters

<i>pFileSize</i> [IN/OUT]	<ul style="list-style-type: none"> Upon input, the maximum number of bytes that file contents array can contain. Upon successful output, actual number of bytes written to file contents array
<i>pFile</i> [OUT]	<ul style="list-style-type: none"> ERI data read from persistent storage(Max size is 1024)

9.13.3.5 typedef struct serialNumbersInfo serialNumbersInfo

Returns all the serial numbers assigned to the device. These serial numbers include the ESN (Electronic serial number of the device), the IMEI (International Mobile Equipment Identity) and MEID (Mobile Equipment Identifier).

Parameters

<i>esnSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the ESN string array can contain
<i>pESNString</i> [OUT]	<ul style="list-style-type: none"> NULL-terminated ESN string. Empty string is returned when ESN is not supported/programmed
<i>imeiSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the IMEI string array can contain
<i>pIMEIString</i> [OUT]	<ul style="list-style-type: none"> NULL terminated IMEI string. Empty string is returned when IMEI is not supported/programmed
<i>meidSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the MEID string array can contain
<i>pMEIDString</i> [OUT]	<ul style="list-style-type: none"> NULL-terminated MEID string. Empty string is returned when MEID is not supported/programmed
<i>imeiSvnSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the IMEI SVN string array can contain.
<i>pImeiSvnString</i> [OUT]	<ul style="list-style-type: none"> NULL-terminated IMEI SVN string. Empty string is returned when IMEI SVN is not supported/programmed.

9.13.3.6 typedef struct _SLQSSwiGetHostDevInfoParams SLQSSwiGetHostDevInfoParams

This structure is used to Get Host Device Information

Parameters

<i>bManSize</i> [IN↔ OUT]	<ul style="list-style-type: none"> Host Device Manufacturer String Size
<i>pManString</i> [O↔ UT]	<ul style="list-style-type: none"> Host Device Manufacturer Name(Optional parameter) Null terminated ASCII String
<i>bModelSize</i> [I↔ N/OUT]	<ul style="list-style-type: none"> Host Device Model String Size
<i>pModelString</i> [↔ OUT]	<ul style="list-style-type: none"> Host Device Model String(Optional parameter) Null terminated ASCII string.
<i>bSWVerSize</i> [I↔ N/OUT]	<ul style="list-style-type: none"> Host Device Software Version String Size
<i>pSWVerString</i> [↔ OUT]	<ul style="list-style-type: none"> Host Device Software Version String(Optional parameter) Null terminated ASCII string
<i>bPlasmaID</i> ↔ Size[IN/OUT]	<ul style="list-style-type: none"> Host Device Plasma ID String Size
<i>pPlasmaID</i> ↔ String[OUT]	<ul style="list-style-type: none"> Host Device Plasma ID String(Optional parameter) Null terminated alphanumeric ASCII String.

9.13.3.7 typedef struct _SLQSSwiGetOSInfoParams SLQSSwiGetOSInfoParams

This structure is used to Get OS Information

Parameters

<i>bNameSize</i> [I↔ N/OUT]	<ul style="list-style-type: none"> Size of Operating System Name
<i>pNameString</i> [↔ OUT]	<ul style="list-style-type: none"> Operating System Name(Optional parameter) Null terminated ASCII string
<i>bVersionSize</i> [I↔ N/OUT]	<ul style="list-style-type: none"> Operating System Version Size
<i>pVersion</i> ↔ String[OUT]	<ul style="list-style-type: none"> Operating System Version String(Optional parameter) Null terminated ASCII string.

9.13.3.8 typedef struct _SLQSSwiGetSerialNoExtParams SLQSSwiGetSerialNoExtParams

This structure is used to store MEID Information

Parameters

<i>meidLength</i> [O↔ UT]	<ul style="list-style-type: none"> String length of the of MEID received
<i>pMeidString</i> [O↔ UT]	<ul style="list-style-type: none"> Optional parameter Pointer to receive String containing the Mobile Equipment Identifier(MEID) of the device.

9.13.3.9 typedef struct _SLQSSwiSetHostDevInfoParams SLQSSwiSetHostDevInfoParams

This structure is used to Set Host Device Information

Parameters

<i>bManSize</i> [IN]	<ul style="list-style-type: none"> Host Device Manufacturer String Size
<i>pManString</i> [IN]	<ul style="list-style-type: none"> Host Device Manufacturer Name(Optional parameter) Null terminated ASCII String
<i>bModelSize</i> [IN]	<ul style="list-style-type: none"> Host Device Model String Size
<i>pModelString</i> [IN]	<ul style="list-style-type: none"> Host Device Model String(Optional parameter) Null terminated ASCII string.
<i>bSWVerSize</i> [IN]	<ul style="list-style-type: none"> Host Device Software Version String Size
<i>pSWVerString</i> [↔ IN]	<ul style="list-style-type: none"> Host Device Software Version String(Optional parameter) Null terminated ASCII string
<i>bPlasmaID</i> ↔ Size[IN]	<ul style="list-style-type: none"> Host Device Plasma ID String Size
<i>pPlasmaID</i> ↔ String[IN]	<ul style="list-style-type: none"> Host Device Plasma ID String(Optional parameter) Null terminated alphanumeric ASCII String.

9.13.3.10 typedef struct _SLQSSwiSetOSInfoParams SLQSSwiSetOSInfoParams

This structure is used to Set OS Information

Parameters

<i>bNameSize</i> [IN]	<ul style="list-style-type: none"> Size of Operating System Name
<i>pNameString</i> [IN]	<ul style="list-style-type: none"> Operating System Name(Optional parameter) Null terminated ASCII string
<i>bVersionSize</i> [IN]	<ul style="list-style-type: none"> Operating System Version Size
<i>pVersionString</i> [IN]	<ul style="list-style-type: none"> Operating System Version String(Optional parameter) Null terminated ASCII string.

9.13.4 Function Documentation

9.13.4.1 ULONG ActivateAutomatic (CHAR * *pActivationCode*)

Requests the device to perform automatic service activation

Parameters

<i>pActivationCode</i> [IN]	<ul style="list-style-type: none"> NULL-terminated string representing activation code (maximum string length of 12); specific carrier requirements may dictate actual activation code that is applicable, e.g., "*22899"
-----------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Minutes

9.13.4.2 ULONG GetActivationState (ULONG * *pActivationState*)

Returns the device activation state.

Parameters

<i>pActivationState</i> ↔ <i>[OUT]</i>	<ul style="list-style-type: none"> • Service Activation Code 0 - Service not activated 1 - Service activated 2 - Activation connecting 3 - Activation connected 4 - OTASP security authenticated 5 - OTASP NAM downloaded 6 - OTASP MDN downloaded 7 - OTASP IMSI downloaded 8 - OTASP PRL downloaded 9 - OTASP SPC downloaded 10 - OTASP settings committed
---	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 2 Seconds

9.13.4.3 ULONG GetDeviceCapabilities (ULONG * pMaxTXChannelRate, ULONG * pMaxRXChannelRate, ULONG * pDataServiceCapability, ULONG * pSimCapability, ULONG * pRadiolfacesSize, BYTE * pRadiolfaces)

Gets the device capabilities

Parameters

<i>pMaxTXChannelRate</i> ↔ <i>[OUT]</i>	<ul style="list-style-type: none"> • Maximum transmission rate (in bps) supported by the device • In multi-technology devices, this value will be the greatest rate among all supported technologies
<i>pMaxRXChannelRate</i> ↔ <i>[OUT]</i>	<ul style="list-style-type: none"> • Maximum reception rate (in bps) supported by the device • In multi-technology devices, this value will be the greatest rate among all supported technologies
<i>pDataServiceCapability</i> ↔ <i>[OUT]</i>	<ul style="list-style-type: none"> • CS/PS data service capability <ul style="list-style-type: none"> – 0 - No data services supported – 1 - Only Circuit Switched (CS) services supported – 2 - Only Packet Switched (PS) services supported – 3 - Simultaneous CS and PS – 4 - Non-simultaneous CS and PS

<i>pSim</i> ↔ <i>Capability</i> [OUT]	<ul style="list-style-type: none"> • Device SIM capability <ul style="list-style-type: none"> – 0 - SIM not supported – 1 - SIM supported
<i>pRadioIfaces</i> ↔ <i>Size</i> [IN/OUT]	<ul style="list-style-type: none"> • Upon input, the maximum number of elements that the radio interface array can contain • Upon successful output, actual number of elements in the radio interface array
<i>pRadioIfaces</i> [↔ <i>OUT</i>]	<ul style="list-style-type: none"> • Radio interface array. This is a structure of array containing the elements below. ULONG radioInterface <ul style="list-style-type: none"> – See qaGobiApiTableRadioInterfaces.h for Radio Interfaces

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.13.4.4 ULONG GetFirmwareRevision (BYTE *stringSize*, CHAR * *pString*)

Returns the device firmware revision

Parameters

<i>stringSize</i>	<ul style="list-style-type: none"> • The maximum number of characters (including NULL terminator) that the string array can contain
<i>pString</i> [OUT]	<ul style="list-style-type: none"> • NULL terminated string

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.13.4.5 **ULONG** GetFirmwareRevisions (**BYTE** *amssSize*, **CHAR** * *pAMSSString*, **BYTE** *bootSize*, **CHAR** * *pBootString*, **BYTE** *priSize*, **CHAR** * *pPRIString*)

Returns the device firmware revisions (AMSS, boot, and PRI)

Parameters

<i>amssSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that the AMSS string array can contain
<i>pAMSSString</i> [↔ <i>OUT</i>]	<ul style="list-style-type: none"> NULL-terminated AMSS revision string
<i>bootSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that the boot string array can contain
<i>pBootString</i> [↔ <i>OUT</i>]	<ul style="list-style-type: none"> NULL-terminated boot code revision string
<i>priSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that the PRI string array can contain
<i>pPRIString</i> [↔ <i>OUT</i>]	<ul style="list-style-type: none"> NULL-terminated PRI revision string

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.13.4.6 **ULONG** GetHardwareRevision (**BYTE** *stringSize*, **CHAR** * *pString*)

Returns the hardware revision of the device

Parameters

<i>stringSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the string array can contain
<i>pString[OUT]</i>	<ul style="list-style-type: none"> NULL terminated string

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.13.4.7 ULONG GetIMSI (BYTE *stringSize*, CHAR * *pString*)

Returns the device IMSI. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C↔_05_xx_xx_xx and all EM74xx firmware versions. Please use [SLQSUIMReadTransparent\(\)](#) (EF ID: 3F00 7F20 6F07 for 2G card and 3F00 7FFF 6F07 for 3G card) instead for new firmware versions and new modules.

Parameters

<i>stringSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the string array can contain
<i>pString[OUT]</i>	<ul style="list-style-type: none"> NULL terminated string

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.13.4.8 ULONG GetManufacturer (BYTE *stringSize*, CHAR * *pString*)

Returns the device manufacturer name

Parameters

<i>stringSize</i>	<ul style="list-style-type: none">• The maximum number of characters (including NULL terminator) that the string array can contain.
<i>pString</i> [OUT]	<ul style="list-style-type: none">• NULL terminated string

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.13.4.9 ULONG GetModelID (BYTE *stringSize*, CHAR * *pString*)

Returns the device model ID

Parameters

<i>stringSize</i>	<ul style="list-style-type: none">• The maximum number of characters (including NULL terminator) that the string array can contain
<i>pString</i> [OUT]	<ul style="list-style-type: none">• NULL terminated string

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.13.4.10 **ULONG** GetNetworkTime (**ULONGLONG** * *pTimeStamp*, **ULONG** * *pTimeSource*)

Returns the current time of the device based on the value supported by the network.

Parameters

<i>pTimeStamp</i> OUT]	<ul style="list-style-type: none"> Count of 1.25 ms that have elapsed from the start of GPS time (Jan 6, 1980)
<i>pTimeSource</i> OUT]	<ul style="list-style-type: none"> Source of timestamp <ul style="list-style-type: none"> 0 - 32 kHz device clock 1 - CDMA network 2 - cdma2000 1xEV-DO network

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

The source of the timestamp provided specifies how the timestamp was determined. The first network time that is available will be returned. If no network time is available, the timestamp is taken from the 32 kHz slow-clock of the device.

9.13.4.11 **ULONG** GetOfflineReason (**ULONG** * *pReasonMask*, **ULONG** * *pbPlatform*)

Returns reason why the operating mode of the device is currently offline.

Parameters

<i>pReasonMask</i> OUT]	<ul style="list-style-type: none"> Optional parameter Bitmask of offline reasons <ul style="list-style-type: none"> 0x00000001 - Host image configuration issue 0x00000002 - PRI image configuration issue 0x00000004 - PRI version incompatible 0x00000008 - PRI copy issue All others - Reserved
<i>pbPlatform</i> OUT]	<ul style="list-style-type: none"> Optional parameter Is the device offline due to a platform restriction? <ul style="list-style-type: none"> 0 - No 1 - Yes
Generated by Doxygen	

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.13.4.12 ULONG GetPower (ULONG * pPowerMode)

Returns the operating mode of the device.

Parameters

<i>pPowerMode</i> [↔ OUT]	<ul style="list-style-type: none">Selected operating mode<ul style="list-style-type: none">See qaGobiApiTablePowerModes.h for power modes
---------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.13.4.13 ULONG GetPRLVersion (WORD * pPRLVersion)

Returns the version of the active Preferred Roaming List (PRL) in use by the device.

Parameters

<i>pPRLVersion</i> [↔ OUT]	<ul style="list-style-type: none">PRL version number
----------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.13.4.14 ULONG GetSerialNumbers (BYTE *esnSize*, CHAR * *pESNString*, BYTE *imeiSize*, CHAR * *pIMEIString*, BYTE *meidSize*, CHAR * *pMEIDString*)

Returns all the serial numbers assigned to the device. These serial numbers include the ESN (Electronic serial number of the device), the IMEI (International Mobile Equipment Identity) and MEID (Mobile Equipment Identifier).

Parameters

<i>esnSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the ESN string array can contain
<i>pESNString</i> [O↔ UT]	<ul style="list-style-type: none"> NULL-terminated ESN string. Empty string is returned when ESN is not supported/programmed
<i>imeiSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the IMEI string array can contain
<i>pIMEIString</i> [O↔ UT]	<ul style="list-style-type: none"> NULL terminated IMEI string. Empty string is returned when IMEI is not supported/programmed
<i>meidSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the MEID string array can contain
<i>pMEIDString</i> [↔ OUT]	<ul style="list-style-type: none"> NULL-terminated MEID string. Empty string is returned when MEID is not supported/programmed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

For CDMA devices that use a RUIM, the MEID of the RUIM (if present) will be returned. Use [SLQSSwiGetSerialNoExt\(\)](#) to get MEID of CDMA modems. Timeout: 2 seconds

9.13.4.15 ULONG GetVoiceNumber (BYTE *voiceNumberSize*, CHAR * *pVoiceNumber*, BYTE *minSize*, CHAR * *pMIN*)

Returns the voice number in use by the device

Parameters

<i>voiceNumberSize</i>	<ul style="list-style-type: none">Maximum number of characters (including NULL terminator) that the voice number array can contain.
<i>pVoiceNumber</i> [OUT]	<ul style="list-style-type: none">Voice number string: MDN or MS ISDN
<i>minSize</i>	<ul style="list-style-type: none">Maximum number of characters (including NULL terminator) that the MIN array can contain.
<i>pMIN</i> [OUT]	<ul style="list-style-type: none">Optional ParameterMIN string: Empty string returned when MIN is not supported/ programmed.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.13.4.16 ULONG ResetToFactoryDefaults (CHAR * *pSPC*)

Resets to default factory settings of the device

Parameters

<i>pSPC</i> [IN]	<ul style="list-style-type: none">NULL-terminated string representing a six-digit service programming code
------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 minutes

9.13.4.17 ULONG SetPower (ULONG *powerMode*)

Sets the operating mode of the device.

Parameters

<i>powerMode</i> [IN]	<ul style="list-style-type: none">Selected operating mode<ul style="list-style-type: none">See qaGobiApiTablePowerModes.h for power modes
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.13.4.18 ULONG SLQSDmsSwiGetResetInfo (dmsSwiGetResetInfo * *pGetResetInfoResp*)

This function is used to get reset info

Parameters

<i>pGetResetInfo</i> ↔ <i>Resp</i>	<ul style="list-style-type: none">See dmsSwiGetResetInfo for more information of the input structure
---------------------------------------	--

9.13.4.19 ULONG SLQSDmsSwiIndicationRegister (dmsIndicationRegisterReq * pIndicationRegisterReq)

This function used to set Swi Indication Register

Parameters

<i>pConfig</i>	<ul style="list-style-type: none">• See dmsIndicationRegisterReq for more information of the input structure
----------------	--

9.13.4.20 ULONG SLQSGetBandCapabilities (BandCapabilityResp * pBandCapability)

Returns the band capability of the device.

Parameters

<i>pBand↔ Capability[OUT]</i>	See BandCapabilityResp for more information of the input structure
-----------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.13.4.21 ULONG SLQSGetBandCapability (ULONGLONG * pBandCapability)

Returns the band capability of the device.

Parameters

<p><i>pBand</i>↔ <i>Capability[OUT]</i></p>	<p>Bitmask of bands supported by the device</p> <ul style="list-style-type: none"> • Bit 0 - Band class 0, A-system • Bit 1 - Band class 0, B-system • Bit 2 - Band class 1, all blocks • Bit 3 - Band class 2 • Bit 4 - Band class 3, A-system • Bit 5 - Band class 4, all blocks • Bit 6 - Band class 5, all blocks • Bit 7 - GSM DCS band (1800) • Bit 8 - GSM Extended GSM (E-GSM) band (900) • Bit 9 - GSM Primary GSM (P-GSM) band (900) • Bit 10 - Band class 6 • Bit 11 - Band class 7 • Bit 12 - Band class 8 • Bit 13 - Band class 9 • Bit 14 - Band class 10 • Bit 15 - Band class 11 • Bit 16 - GSM 450 band • Bit 17 - GSM 480 band • Bit 18 - GSM 750 band • Bit 19 - GSM 850 band • Bit 20 - GSM railways GSM band (900) • Bit 21 - GSM PCS band (1900) • Bit 22 - WCDMA (Europe, Japan, and China) 2100 band • Bit 23 - WCDMA US PCS 1900 band • Bit 24 - WCDMA (Europe and China) DCS 1800 band • Bit 25 - WCDMA US 1700 band • Bit 26 - WCDMA US 850 band • Bit 27 - WCDMA Japan 800 band • Bit 28 - Band class 12 • Bit 29 - Band class 14 • Bit 30 - Reserved • Bit 31 - Band class 15 • Bits 32 through 47 - Reserved • Bit 48 - WCDMA Europe 2600 band • Bit 49 - WCDMA Europe and Japan 900 band • Bit 50 - WCDMA Japan 1700 band • Bits 51 through 55 - Reserved • Bit 56 - Band class 16 • Bit 57 - Band class 17 • Bit 58 - Band class 18 • Bit 59 - Band class 19
---	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.13.4.22 ULONG SLQSGetCurrentPRLInfo (dmsCurrentPRLInfo * pCurrentPRLInfo)

This API get the currently active PRL information of the device.

Parameters

<i>pCurrentPRLInfo</i>	<ul style="list-style-type: none">• Pointer to structure dmsCurrentPRLInfo• See dmsCurrentPRLInfo for more information
------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 2 Secs

9.13.4.23 ULONG SLQSGetCustFeatures (custFeaturesInfo * pCustFeaturesInfo)

This API fetches the current settings of custom features. This API is deprecated for EM74xx/MC74xx, please use [SLQSGetCustFeaturesV2\(\)](#) instead for EM74xx/MC74xx.

Parameters

<i>pCustFeaturesInfo</i>	<ul style="list-style-type: none">• Structure containing settings of custom features.• See custFeaturesInfo for more information
--------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Secs

9.13.4.24 ULONG SLQSGetCustFeaturesV2 ([getCustomFeatureV2](#) * *pGetCustomFeatureV2*)

This function queries the modem for a list of supported features. This function is for firmware version 2.0 and newer. Currently supported Customization features:

- GPIOSARENABLE
- GPSSEL
- IMSWITCHHIDE
- IPV6ENABLE
- WAKEHOSTEN

Parameters

<i>pGetCustomFeatureV2</i>	<ul style="list-style-type: none">• See getCustomFeatureV2 for more information of the input structure
----------------------------	--

9.13.4.25 ULONG SLQSGetERIFile ([ERIFileparams](#) * *pERIFileparams*)

Returns the Extended Roaming Indicator (ERI) file that is stored in EFS on the device at a predetermined location. See the carrier requirements for specific details.

Parameters

<i>pERIFileparams</i>	<ul style="list-style-type: none">• Pointer to structure ERIFileparams• See ERIFileparams for more information
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 5 Seconds

9.13.4.26 ULONG SLQSGetSerialNumbers (serialNumbersInfo * pSerialNumbersInfo)

Returns all the serial numbers assigned to the device. These serial numbers include the ESN (Electronic serial number of the device), the IMEI (International Mobile Equipment Identity), MEID (Mobile Equipment Identifier) and IMEI SVN (IMEI software version number).

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

For CDMA devices that use a RUIM, the MEID of the RUIM (if present) will be returned. Use [SLQSSwiGetSerialNoExt\(\)](#) to get MEID of CDMA modems. Timeout: 2 seconds

9.13.4.27 ULONG SLQSSetCustFeatures (custFeaturesSetting * pCustFeaturesSetting)

This API changes the settings of custom features, a reset is required for any settings that are changed to take effect. This API is deprecated for EM74xx/MC74xx, please use [SLQSSetCustFeaturesV2\(\)](#) for EM74xx/MC74xx.

Parameters

<i>pCustFeaturesSetting</i> [IN]	<ul style="list-style-type: none"> Structure containing settings of custom features. See custFeaturesSetting for more information
----------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Secs

9.13.4.28 ULONG SLQSSetCustFeaturesV2 (setCustomSettingV2 * pSetCustSetting)

This function sets the modem for a list of supported features. This function is for firmware version 2.0 and newer. Currently supported customization features:

- GPIOSARENABLE
- GPSSEL
- IMSWITCHHIDE
- IPV6ENABLE
- WAKEHOSTEN

Parameters

<i>pSetCustSetting</i>	<ul style="list-style-type: none"> • Optional parameter • See setCustomSettingV2 for more information
------------------------	---

9.13.4.29 **ULONG** SLQSSwiClearDyingGaspStatistics ()

This function Clear Dying GASP Statistics.

9.13.4.30 **ULONG** SLQSSwiGetCrashAction (**BYTE** * *pDevCrashState*)

This API queries the Crash State from the device.

Parameters

<i>pDevCrashState</i> <i>State[OUT]</i>	<ul style="list-style-type: none"> • Device Crash State • Values: <ul style="list-style-type: none"> – 0 - USB Memory Download Modem will reset after a crash and will stay in USB download mode with only ttyUSB0 enumerated. ramdump tool is to be used to recover the crash dump. Modem needs to be reset again to come back in ONLINE mode. – 1 - Reset Modem will reset and come back in ONLINE mode. Minimal crash data will be available and can be extracted with at!gcdump? AT command or SLQSSwiGetCrashInfo() SDK API – 2 - No action
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA

Please free two buffers after get crash report successfully

1. pCrashInfoParams->pCrashInfo->pCrashString
2. pCrashInfoParams->pCrashInfo->pGCDumpString Timeout: 5 Secs

9.13.4.31 ULONG SLQSSwiGetCrashInfo (BYTE * *pClear*, CrashInfoParams * *pCrashInfoParams*)

This API queries the Crash Information from the device.

Parameters

<i>pClear</i> [IN]	<ul style="list-style-type: none"> request parameter Clear(Optional parameter) Values: 0 - Do not clear crash data after response 1 - Clear crash data after response
<i>pCrashInfoParams</i> [Out]	<ul style="list-style-type: none"> Pointer to structure CrashInfoParams See CrashInfoParams for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Timeout: 5 Secs

9.13.4.32 ULONG SLQSSwiGetDyingGaspCfg (getDyingGaspCfg * *pConfig*)

This function queries Dying GASP Config.

Parameters

<i>pGetCustomFeatureV2</i>	<ul style="list-style-type: none"> See getDyingGaspCfg for more information of the input structure *
----------------------------	---

9.13.4.33 ULONG SLQSSwiGetDyingGaspStatistics (getDyingGaspStatistics * *pStatistics*)

This function queries Dying GASP Statistics.

Parameters

<i>pStatistics</i>	<ul style="list-style-type: none"> See getDyingGaspStatistics for more information of the input structure *
--------------------	--

9.13.4.34 ULONG SLQSSwiGetFSN (FactorySequenceNumber * pFSNumber)

This API get the Factory Sequence Number of the device.

Parameters

<i>pFSNumber</i>	<ul style="list-style-type: none"> • Pointer to structure FactorySequenceNumber • See FactorySequenceNumber for more information
------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 5 Secs

9.13.4.35 ULONG SLQSSwiGetFwUpdateStatus (FirmwareUpdatStat * pFirmwareUpdatStat)

This API will be used to query last firmware update status. The firmware status is stored in RAM and can be retained over warm resets but not power off resets.

Parameters

<i>pFirmware↔ UpdatStat</i>	<ul style="list-style-type: none"> • Pointer to structure FirmwareUpdatStat • See FirmwareUpdatStat for more information
---------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Device Supported: MC73xx
Timeout: 5 Secs

9.13.4.36 ULONG SLQSSwiGetHostDevInfo (SLQSSwiGetHostDevInfoParams * *pGetHostDevInfoParams*)

This API Get Host Information from the device.

Parameters

<i>pGetHostDevInfoParams</i>	<ul style="list-style-type: none">• See SLQSSwiGetHostDevInfoParams for more information
------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Timeout: 2 Secs

9.13.4.37 ULONG SLQSSwiGetOSInfo (SLQSSwiGetOSInfoParams * *pParams*)

This API queries the device operating system info configured on the modem for OMA-DM reporting

Parameters

<i>pParams</i>	<ul style="list-style-type: none">• - See SLQSSwiGetOSInfoParams for more information
----------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Timeout: 2 Secs

9.13.4.38 ULONG SLQSSwiGetSerialNoExt (SLQSSwiGetSerialNoExtParams * *pParams*)

This API is used to get the MEID of the modem. For CDMA devices that use a RUIM, the MEID of the modem will always be returned.

Parameters

<i>SLQSSwiGetSerialNoExtParams</i>	
------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Timeout: 5 Secs

9.13.4.39 ULONG SLQSSwiGetUSBComp (USBCompParams * *pUSBCompParams*)

This API queries the modem's USB interface configuration and supported configuration parameters.

Parameters

<i>pUSBCompParams</i>	<ul style="list-style-type: none">• Pointer to structure USBCompParams• See USBCompParams for more information
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Timeout: 5 Secs

9.13.4.40 **ULONG** SLQSSwiSetCrashAction (**BYTE** *crashActionParams*)

This API set the Crash Action to the device.

Parameters

<i>crashActionParams</i> [IN]	<ul style="list-style-type: none"> Crash Action Values: <ul style="list-style-type: none"> 0 - USB Memory Download Modem will reset after a crash and will stay in USB download mode with only ttyUSB0 enumerated. ramdump tool is to be used to recover the crash dump. Modem needs to be reset again to come back in ONLINE mode. 1 - Reset Modem will reset and come back in ONLINE mode. Minimal crash data will be available and can be extracted with at!gcdump? AT command or SLQSSwiGetCrashInfo() SDK API 2 - No action
-------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Timeout: 5 Secs

9.13.4.41 **ULONG** SLQSSwiSetDyingGaspCfg (**setDyingGaspCfg** * *pConfig*)

This function set Dying GASP Config.

Parameters

<i>pConfig</i>	<ul style="list-style-type: none"> See setDyingGaspCfg for more information of the input structure
----------------	---

9.13.4.42 **ULONG** SLQSSwiSetHostDevInfo (**SLQSSwiSetHostDevInfoParams** * *pSetHostDevInfoParams*)

This API Sets the host device info configured on the modem for OMA-DM reporting

Parameters

<i>pSetHostDevInfoParams</i>	<ul style="list-style-type: none"> See SLQSSwiSetHostDevInfoParams for more information
------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Timeout: 2 Secs

9.13.4.43 ULONG SLQSSwiSetOSInfo (SLQSSwiSetOSInfoParams * *pParams*)

This API Set OS Information to the device.

Parameters

<i>pParams</i>	<ul style="list-style-type: none">• See SLQSSwiSetOSInfoParams for more information
----------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Timeout: 2 Secs

9.13.4.44 ULONG SLQSSwiSetUSBComp (USBCompConfig * *pUSBCompConfig*)

This API is used to change the modem's USB interface configuration thus allowing a device to have multiple USB compositions. Devices will, by default, be configured to support a minimal set of interfaces to reduce end user modem installation time. Developers and some customers, however, require access to a custom set of interfaces. A reset is required for any change in the USB composition to take effect.

Parameters

<i>pUSBCompConfig</i>	<ul style="list-style-type: none">• Pointer to structure USBCompConfig• See USBCompConfig for more information
-----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Timeout: 5 Secs

9.13.4.45 ULONG SLQSUIMGetState (ULONG * pUIMState)

Returns the UIM state. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15↵C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSUIMGetCardStatus\(\)](#) for new firmware versions and new modules

Parameters

<i>pUIMState[OUT]</i>	<ul style="list-style-type: none">• UIM state:<ul style="list-style-type: none">– 0x00 - UIM initialization completed– 0x01 - UIM locked or failed– 0x02 - UIM not present– 0x03 - 0xFE - Reserved– 0xFF - UIM state currently unavailable
-----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.13.4.46 **ULONG** UIMChangePIN (**ULONG** *id*, **CHAR** * *pOldValue*, **CHAR** * *pNewValue*, **ULONG** * *pVerifyRetriesLeft*, **ULONG** * *pUnblockRetriesLeft*)

Changes the PIN value for a given PIN. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSUIMChangePin\(\)](#) for new firmware versions and new modules

Parameters

<i>id</i> [IN]	<ul style="list-style-type: none"> PIN ID <ul style="list-style-type: none"> 1 (PIN1 / CHV1) 2 (PIN2 / CHV2)
<i>pOldValue</i> [IN]	<ul style="list-style-type: none"> Old PIN value of PIN to change
<i>pNewValue</i> [IN]	<ul style="list-style-type: none"> New PIN value of PIN to change
<i>pVerifyRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Optional Parameter Upon operational failure, this will indicate number of retries left, after which PIN will be blocked. <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
<i>pUnblockRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Optional Parameter Upon operational failure, this will indicate number of unblock retries left, after which the PIN will be permanently blocked; i.e. UIM is unusable. <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.13.4.47 **ULONG** UIMGetControlKeyStatus (**ULONG** *id*, **ULONG** * *pStatus*, **ULONG** * *pVerifyRetriesLeft*, **ULONG** * *pUnblockRetriesLeft*)

Returns the status of the specified UIM facility control key. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use [SLQSUIMGetConfiguration](#) instead for new firmware versions and new modules.

Parameters

<i>id</i> [IN]	<ul style="list-style-type: none"> Facility ID <ul style="list-style-type: none"> 0 - Network Personalization (PN) 1 - Network Subset Personalization (PU) 2 - Service Provider Personalization (PP) 3 - Corporate Personalization (PC) 4 - UIM Personalization (PF)
<i>pStatus</i> [OUT]	<ul style="list-style-type: none"> Control key status <ul style="list-style-type: none"> 0 - Deactivated 1 - Activated 2 - Blocked
<i>pVerifyRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> The number of retries left, after which the control key will be blocked <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
<i>pUnblockRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> The number of unblock retries left, after which the control key will be permanently blocked <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.13.4.48 ULONG UIMGetICCID (BYTE *stringSize*, CHAR * *pString*)

Returns the UIM ICCID. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use [SLQSUIReadTransparent\(\)](#) (EF ID: 3F00 2FE2) instead for new firmware versions and new modules.

Parameters

<i>stringSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the string array can contain.
<i>pString</i> [OUT]	<ul style="list-style-type: none"> NULL terminated string

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.13.4.49 **ULONG** UIMGetPINStatus (**ULONG** *id*, **ULONG** * *pStatus*, **ULONG** * *pVerifyRetriesLeft*, **ULONG** * *pUnblockRetriesLeft*)

Gets the status of the SIM PINs. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSUIMGetCardStatus\(\)](#) for new firmware versions and new modules

Parameters

<i>id</i>	<ul style="list-style-type: none"> PIN ID <ul style="list-style-type: none"> 1 (PIN1 / CHV1) 2 (PIN2 / CHV2)
<i>pStatus</i> [OUT]	<ul style="list-style-type: none"> PIN status(0xFFFFFFFF - Unknown) <ul style="list-style-type: none"> 0 - PIN not initialized 1 - PIN enabled, not verified 2 - PIN enabled, verified 3 - PIN disabled 4 - PIN blocked 5 - PIN permanently blocked
<i>pVerifyRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Upon operational failure, this will indicate number of retries left, after which PIN will be blocked. <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
<i>pUnblockRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Upon operational failure, this will indicate number of unblock retries left, after which the PIN will be permanently blocked; i.e., UIM is unusable. <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.13.4.50 ULONG UIMSetControlKeyProtection (ULONG id, ULONG status, CHAR * pValue, ULONG * pVerifyRetriesLeft)

Changes the specified UIM facility control key. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSUIMDepersonalization\(\)](#) for new firmware versions and new modules

Parameters

<i>id</i> [IN]	<ul style="list-style-type: none"> Facility ID <ul style="list-style-type: none"> 0 - Network Personalization (PN) 1 - Network Subset Personalization (PU) 2 - Service Provider Personalization (PP) 3 - Corporate Personalization (PC) 4 - UIM Personalization (PF)
<i>status</i> [IN]	<ul style="list-style-type: none"> Control key status <ul style="list-style-type: none"> 0 - Deactivated
<i>pValue</i> [IN]	<ul style="list-style-type: none"> Control key de-personalization string (maximum length of 8 characters)
<i>pVerifyRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Optional parameter Upon operational failure, this will indicate number of retries left, after which the control key will be blocked <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.13.4.51 `ULONG UIMSetPINProtection (ULONG id, ULONG bEnable, CHAR * pValue, ULONG * pVerifyRetriesLeft, ULONG * pUnblockRetriesLeft)`

Enables or disables protection of SIM contents for a given PIN, This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSUIMSetPinProtection\(\)](#) for new firmware versions and new modules

Parameters

<i>id</i> [IN]	<ul style="list-style-type: none"> PIN ID <ul style="list-style-type: none"> 1 (PIN1 / CHV1) 2 (PIN2 / CHV2)
<i>bEnable</i> [IN]	<ul style="list-style-type: none"> Enable/disable PIN protection, 0 = Disable
<i>pValue</i> [IN]	<ul style="list-style-type: none"> PIN value of the PIN to be enabled/disabled
<i>pVerifyRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Optional parameter Upon operational failure, this will indicate number of retries left, after which PIN will be blocked. <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
<i>pUnblockRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Optional parameter Upon operational failure, this will indicate number of unblock retries left, after which the PIN will be permanently blocked i.e. UIM is unusable. <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.13.4.52 `ULONG UIMUnblockControlKey (ULONG id, CHAR * pValue, ULONG * pUnblockRetriesLeft)`

Unlocks the specified UIM facility control key. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSUIMDepersonalization\(\)](#) for new firmware versions and new modules

Parameters

<i>id</i> [IN]	<ul style="list-style-type: none"> Facility ID <ul style="list-style-type: none"> 0 - Network Personalization (PN) 1 - Network Subset Personalization (PU) 2 - Service Provider Personalization (PP) 3 - Corporate Personalization (PC) 4 - UIM Personalization (PF)
<i>pValue</i> [IN]	<ul style="list-style-type: none"> Control key de-personalization string (maximum length of 8 characters)
<i>pUnblockRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Optional parameter Upon operational failure, this will indicate number of unblock retries left, after which the control key will be blocked <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.13.4.53 **ULONG** UIMUnblockPIN (**ULONG** *id*, **CHAR** * *pPUKValue*, **CHAR** * *pNewValue*, **ULONG** * *pVerifyRetriesLeft*, **ULONG** * *pUnblockRetriesLeft*)

Unblocks a blocked SIM. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSUIMUnblockPin\(\)](#) for new firmware versions and new modules

Parameters

<i>id</i> [IN]	<ul style="list-style-type: none"> PIN ID <ul style="list-style-type: none"> 1 (PIN1 / CHV1) 2 (PIN2 / CHV2)
<i>pPUKValue</i> [IN]	<ul style="list-style-type: none"> PUK value of PIN to unblock
<i>pNewValue</i> [IN]	<ul style="list-style-type: none"> New PIN value of PIN to unblock

<i>pVerifyRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Optional Parameter Upon operational failure, this will indicate number of retries left, after which the PIN will be blocked. <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
<i>pUnblockRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Optional Parameter Upon operational failure, this will indicate number of unblock retries left, after which the PIN will be permanently blocked; i.e. UIM is unusable <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.13.4.54 ULONG UIMVerifyPIN (ULONG id, CHAR * pValue, ULONG * pVerifyRetriesLeft, ULONG * pUnblockRetriesLeft)

Verifies a SIM PIN. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSUIMVerifyPin\(\)](#) for new firmware versions and new modules

Parameters

<i>id</i> [IN]	<ul style="list-style-type: none"> PIN ID <ul style="list-style-type: none"> 1 (PIN1 / CHV1) 2 (PIN2 / CHV2)
<i>pValue</i> [IN]	<ul style="list-style-type: none"> Value of PIN to verify
<i>pVerifyRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Optional Parameter Upon operational failure, this will indicate number of retries left, after which the PIN will be blocked. <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
<i>pUnblockRetriesLeft</i> [OUT]	<ul style="list-style-type: none"> Optional Parameter Upon operational failure, this will indicate number of unblock retries left, after which the PIN will be permanently blocked; i.e. UIM is unusable <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.13.4.55 ULONG ValidateSPC (CHAR * pSPC)

This function Validates Service Programming code of the device

Parameters

<i>pSPC</i> [IN]	<ul style="list-style-type: none"> • NULL-terminated string representing a six-digit service programming code
------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Device Supported: MC83x5, MC7750
Timeout: 2 seconds

9.14 qaGobiApiFms.h File Reference

Firmware Management Service API function prototypes.

Data Structures

- struct [fwinfo_s](#)
- struct [slqsfwinfo_s](#)
- struct [qmifwinfo_s](#)
- struct [ImageElement](#)
- struct [PrefImageList](#)
- struct [ImageIdElement](#)
- struct [ImageIDEntries](#)
- struct [ImageList](#)
- struct [sGetDeviceSeriesResult](#)
- struct [SWI_STRUCT_CarrierImage](#)
- struct [CurrImageInfo](#)
- struct [CurrentImgList](#)

Macros

- `#define SLQSFWINFO_MODELID_SZ 20`
- `#define SLQSFWINFO_BOOTVERSION_SZ 85`
- `#define SLQSFWINFO_APPVERSION_SZ 85`
- `#define SLQSFWINFO_SKU_SZ 15`
- `#define SLQSFWINFO_PACKAGEID_SZ 85`
- `#define SLQSFWINFO_CARRIER_SZ 20`
- `#define SLQSFWINFO_PRIVERSION_SZ 16`
- `#define SLQSFWINFO_CUR_CARR_NAME 17`
- `#define SLQSFWINFO_CUR_CARR_REV 13`
- `#define GOBI_MBN_IMG_ID_STR_LEN 16`
- `#define GOBI_MBN_BUILD_ID_STR_LEN 100`
- `#define GOBI_LISTENTRIES_MAX 2`
- `#define GOBI_SET_IMG_PREF_RSLEN 40`
- `#define DEVICE_SHUTDOWN 5`
- `#define DEVICE_RESET 4`
- `#define DEVICE_OFFLINE 3`
- `#define FIRMWARE_UPDATE_SUCCESS 0x01`
- `#define FIRMWARE_UPDATE_FAIL 0x01`
- `#define PRI_UPDATE_FAIL 0x02`
- `#define FIRMWARE_UPGRADE_SUCCESS 0x00`
- `#define IMG_ID_LEN 16`
- `#define BUILD_ID_LEN 100`
- `#define G3K_FIRMWARE_DOWNLOAD 1`
- `#define SPKG_FIRMWARE_DOWNLOAD 2`
- `#define UNIQUE_ID_LEN 16`
- `#define BUILD_ID_MAX_LEN 255`
- `#define IMGDETAILS_LEN 16`
- `#define MAX_IMAGE_IDE_ELEMENTS 50`

Enumerations

- `enum eGobiImageTech {
 eGOBI_IMG_TECH_CDMA = 0,
 eGOBI_IMG_TECH_UMTS }`
- `enum eGobiImageCarrier {`

```
eGOBI_IMG_CAR_GENERIC = 1,  
eGOBI_IMG_CAR_FACTORY,  
eGOBI_IMG_CAR_NORF,  
eGOBI_IMG_CAR_VERIZON = 101,  
eGOBI_IMG_CAR_SPRINT,  
eGOBI_IMG_CAR_ALLTEL,  
eGOBI_IMG_CAR_BELL,  
eGOBI_IMG_CAR_TELUS,  
eGOBI_IMG_CAR_US,  
eGOBI_IMG_CAR_TELSTRA1,  
eGOBI_IMG_CAR_CHINA_UNICOM,  
eGOBI_IMG_CAR_TELCOM_NZ,  
eGOBI_IMG_CAR_SK_TELCOM1,  
eGOBI_IMG_CAR_RELIANCE1,  
eGOBI_IMG_CAR_TATA,  
eGOBI_IMG_CAR_METROPCS,  
eGOBI_IMG_CAR_LEAP,  
eGOBI_IMG_CAR_KDDI,  
eGOBI_IMG_CAR_IUSACELL,  
eGOBI_IMG_CAR_CHINA_TELECOM,  
eGOBI_IMG_CAR_OMH,  
eGOBI_IMG_CAR_GENERIC_CDMA,  
eGOBI_IMG_CAR_ATT = 201,  
eGOBI_IMG_CAR_VODAFONE,  
eGOBI_IMG_CAR_TMOBILE,  
eGOBI_IMG_CAR_ORANGE,  
eGOBI_IMG_CAR_TELEFONICA,  
eGOBI_IMG_CAR_TELCOM_ITALIA,  
eGOBI_IMG_CAR_3,  
eGOBI_IMG_CAR_O2,  
eGOBI_IMG_CAR_SFR,  
eGOBI_IMG_CAR_SWISSCOM,  
eGOBI_IMG_CAR_CHINA_MOBILE,  
eGOBI_IMG_CAR_TELSTRA2,  
eGOBI_IMG_CAR_SINGTEL_OPTUS,  
eGOBI_IMG_CAR_RELIANCE2,  
eGOBI_IMG_CAR_BHARTI,  
eGOBI_IMG_CAR_NTT_DOCOMO,  
eGOBI_IMG_CAR_EMOBILE,  
eGOBI_IMG_CAR_SOFTBANK,  
eGOBI_IMG_CAR_KT_FREETEL,  
eGOBI_IMG_CAR_SK_TELCOM2,  
eGOBI_IMG_CAR_TELENOR,  
eGOBI_IMG_CAR_NETCOM,  
eGOBI_IMG_CAR_TELIASONERA,  
eGOBI_IMG_CAR_AMX_TELCEL,  
eGOBI_IMG_CAR_BRASIL_VIVO,  
eGOBI_IMG_CAR_AERIS,  
eGOBI_IMG_CAR_ROGERS }  
  
• enum eGobiImageRegion {  
    eGOBI_IMG_REG_NA = 0,  
    eGOBI_IMG_REG_LA,  
    eGOBI_IMG_REG_EU,  
    eGOBI_IMG_REG_ASIA,  
    eGOBI_IMG_REG_AUS,  
    eGOBI_IMG_REG_GLOBAL }  
  
• enum eGobiImageGPS {
```

```

eGOBI_IMG_GPS_NONE = 0,
eGOBI_IMG_GPS_STAND_ALONE,
eGOBI_IMG_GPS_ASSISTED,
eGOBI_IMG_GPS_NO_XTRA }
• enum eGobiDeviceSeries {
eGOBI_DEV_SERIES_UNKNOWN = -1,
eGOBI_DEV_SERIES_NON_GOBI = 0,
eGOBI_DEV_SERIES_G3K,
eGOBI_DEV_SERIES_SIERRA_GOBI,
eGOBI_DEV_SERIES_9X15,
eGOBI_DEV_SERIES_9X30,
eGobi_DEV_SERIES_MC83 }

```

Functions

- [ULONG GetImageStore](#) ([WORD](#) imageStorePathSize, [CHAR](#) *pImageStorePath)
- [ULONG SLQSGetFirmwareInfo](#) (struct [qmifwinfo_s](#) *pinfo)
- [ULONG SLQSGetImageInfoMC77xx](#) ([LPCSTR](#) path, struct [qmifwinfo_s](#) *pinfo)
- [ULONG SLQSGetImageInfoMC83xx](#) ([LPCSTR](#) path, struct [qmifwinfo_s](#) *pinfo)
- [ULONG SLQSGetImageInfo](#) ([LPCSTR](#) path, struct [qmifwinfo_s](#) *pinfo)
- [ULONG UpgradeFirmware2k](#) ([CHAR](#) *pDestinationPath)
- [ULONG GetImagesPreference](#) ([ULONG](#) *pImageListSize, struct [PrefImageList](#) *pImageList)
- [ULONG SetImagesPreference](#) ([ULONG](#) imageListSize, [BYTE](#) *pImageList, [ULONG](#) bForceDownload, [BYTE](#) modemIndex, [ULONG](#) *pImageTypesSize, [BYTE](#) *pImageTypes)
- [ULONG GetStoredImages](#) ([ULONG](#) *pImageListSize, [BYTE](#) *pImageList)
- [ULONG DeleteStoredImage](#) ([ULONG](#) imageInfoSize, [BYTE](#) *pImageInfo)
- [ULONG SLQSGetImageInfo_9x15](#) ([LPCSTR](#) path, [BYTE](#) imgType, struct [slqsfwinfo_s](#) *pinfo)
- [ULONG SLQSUpgradeFirmware9x15](#) ([CHAR](#) *pDestinationPath)
- [ULONG SLQSGetBootVersionNumber](#) ([ULONG](#) *bootversion)
- [BOOL SLQSIspkgFormatRequired](#) (void)
- void [SLQSSetSpkgFormatRequired](#) ([BYTE](#) isneeded)
- [ULONG upgrade_mc77xx_fw](#) ([LPCSTR](#) path)
- void [eGetDeviceSeries](#) (struct [sGetDeviceSeriesResult](#) *)
- [ULONG SLQSSwiGetAllCarrierImages](#) ([ULONG](#) *pNumOfItems, struct [SWI_STRUCT_CarrierImage](#) *pCarrierImages, [char](#) *pFolderPath)
- [ULONG DownloadToSlot](#) ([CHAR](#) *path, struct [slqsfwinfo_s](#) fwImgInfo, struct [slqsfwinfo_s](#) priImgInfo, [BYTE](#) slot, [BYTE](#) forceDownload)
- [ULONG SLQSDownloadFirmwareToSlot](#) ([CHAR](#) *pPath, [BYTE](#) slot_index, [BYTE](#) force_download)
- [ULONG SLQSGetValidFwPriCombinations](#) (struct [ImageList](#) *pStoredImageList, [ULONG](#) *pValidCombinationSize, struct [SWI_STRUCT_CarrierImage](#) *pValidCombinations)
- [ULONG SLQSSetSIMBasedImageSwitching](#) (void)
- void [SLQSSetCrashStateCheckIgnore](#) ([BOOL](#) ignore)
- [ULONG SLQSSwiGetFirmwareCurr](#) ([CurrentImgList](#) *pCurrentImgList)

9.14.1 Detailed Description

Firmware Management Service API function prototypes.

9.14.2 Macro Definition Documentation

9.14.2.1 `#define BUILD_ID_LEN 100`

9.14.2.2 `#define BUILD_ID_MAX_LEN 255`

9.14.2.3 `#define DEVICE_OFFLINE 3`

9.14.2.4 `#define DEVICE_RESET 4`

9.14.2.5 `#define DEVICE_SHUTDOWN 5`

9.14.2.6 `#define FIRMWARE_UPDATE_FAIL 0x01`

9.14.2.7 `#define FIRMWARE_UPDATE_SUCCESS 0x01`

9.14.2.8 `#define FIRMWARE_UPGRADE_SUCCESS 0x00`

9.14.2.9 `#define G3K_FIRMWARE_DOWNLOAD 1`

9.14.2.10 `#define GOBI_LISTENTRIES_MAX 2`

9.14.2.11 `#define GOBI_MBN_BUILD_ID_STR_LEN 100`

9.14.2.12 `#define GOBI_MBN_IMG_ID_STR_LEN 16`

9.14.2.13 `#define GOBI_SET_IMG_PREF_RSLEN 40`

9.14.2.14 `#define IMG_ID_LEN 16`

9.14.2.15 `#define IMGDETAILS_LEN 16`

9.14.2.16 `#define MAX_IMAGE_IDE_ELEMENTS 50`

9.14.2.17 `#define PRI_UPDATE_FAIL 0x02`

9.14.2.18 `#define SLQSFWINFO_APPVERSION_SZ 85`

9.14.2.19 `#define SLQSFWINFO_BOOTVERSION_SZ 85`

9.14.2.20 `#define SLQSFWINFO_CARRIER_SZ 20`

9.14.2.21 `#define SLQSFWINFO_CUR_CARR_NAME 17`

9.14.2.22 `#define SLQSFWINFO_CUR_CARR_REV 13`

9.14.2.23 `#define SLQSFWINFO_MODELID_SZ 20`

9.14.2.24 `#define SLQSFWINFO_PACKAGEID_SZ 85`

9.14.2.25 `#define SLQSFWINFO_PRIVERSION_SZ 16`

9.14.2.26 `#define SLQSFWINFO_SKU_SZ 15`

9.14.2.27 `#define SPKG_FIRMWARE_DOWNLOAD 2`

9.14.2.28 `#define UNIQUE_ID_LEN 16`

9.14.3 Enumeration Type Documentation

9.14.3.1 `enum eGobiDeviceSeries`

enumeration which lists the Device Series

Enumerator

`eGOBI_DEV_SERIES_UNKNOWN`
`eGOBI_DEV_SERIES_NON_GOBI`
`eGOBI_DEV_SERIES_G3K`
`eGOBI_DEV_SERIES_SIERRA_GOBI`
`eGOBI_DEV_SERIES_9X15`
`eGOBI_DEV_SERIES_9X30`
`eGobi_DEV_SERIES_MC83`

9.14.3.2 `enum eGobiImageCarrier`

enumeration which lists the carrier supported by the image

Enumerator

`eGOBI_IMG_CAR_GENERIC`
`eGOBI_IMG_CAR_FACTORY`
`eGOBI_IMG_CAR_NORF`
`eGOBI_IMG_CAR_VERIZON`
`eGOBI_IMG_CAR_SPRINT`
`eGOBI_IMG_CAR_ALLTEL`
`eGOBI_IMG_CAR_BELL`
`eGOBI_IMG_CAR_TELUS`
`eGOBI_IMG_CAR_US`
`eGOBI_IMG_CAR_TELSTRA1`
`eGOBI_IMG_CAR_CHINA_UNICOM`
`eGOBI_IMG_CAR_TELCOM_NZ`

eGOBI_IMG_CAR_SK_TELCOM1
eGOBI_IMG_CAR_RELIANCE1
eGOBI_IMG_CAR_TATA
eGOBI_IMG_CAR_METROPCS
eGOBI_IMG_CAR_LEAP
eGOBI_IMG_CAR_KDDI
eGOBI_IMG_CAR_IUSACELL
eGOBI_IMG_CAR_CHINA_TELECOM
eGOBI_IMG_CAR_OMH
eGOBI_IMG_CAR_GENERIC_CDMA
eGOBI_IMG_CAR_ATT
eGOBI_IMG_CAR_VODAFONE
eGOBI_IMG_CAR_TMOBILE
eGOBI_IMG_CAR_ORANGE
eGOBI_IMG_CAR_TELEFONICA
eGOBI_IMG_CAR_TELCOM_ITALIA
eGOBI_IMG_CAR_3
eGOBI_IMG_CAR_O2
eGOBI_IMG_CAR_SFR
eGOBI_IMG_CAR_SWISSCOM
eGOBI_IMG_CAR_CHINA_MOBILE
eGOBI_IMG_CAR_TELSTRA2
eGOBI_IMG_CAR_SINGTEL_OPTUS
eGOBI_IMG_CAR_RELIANCE2
eGOBI_IMG_CAR_BHARTI
eGOBI_IMG_CAR_NTT_DOCOMO
eGOBI_IMG_CAR_EMOBILE
eGOBI_IMG_CAR_SOFTBANK
eGOBI_IMG_CAR_KT_FREETEL
eGOBI_IMG_CAR_SK_TELCOM2
eGOBI_IMG_CAR_TELNOR
eGOBI_IMG_CAR_NETCOM
eGOBI_IMG_CAR_TELIASONERA
eGOBI_IMG_CAR_AMX_TELCEL
eGOBI_IMG_CAR_BRASIL_VIVO
eGOBI_IMG_CAR_AERIS
eGOBI_IMG_CAR_ROGERS

9.14.3.3 enum eGobiImageGPS

enumeration which lists the GPS type supported by the image

Enumerator

eGOBI_IMG_GPS_NONE
eGOBI_IMG_GPS_STAND_ALONE
eGOBI_IMG_GPS_ASSISTED
eGOBI_IMG_GPS_NO_XTRA

9.14.3.4 enum eGobiImageRegion

enumeration which lists the region supported by the image

Enumerator

```
eGOBI_IMG_REG_NA
eGOBI_IMG_REG_LA
eGOBI_IMG_REG_EU
eGOBI_IMG_REG_ASIA
eGOBI_IMG_REG_AUS
eGOBI_IMG_REG_GLOBAL
```

9.14.3.5 enum eGobiImageTech

enumeration which lists the technology supported by the image

Enumerator

```
eGOBI_IMG_TECH_CDMA
eGOBI_IMG_TECH_UMTS
```

9.14.4 Function Documentation

9.14.4.1 ULONG DeleteStoredImage (ULONG *imageInfoSize*, BYTE * *plmageInfo*)

Used to delete the specified image from the device. This API function is only relevant to devices with the ability to store multiple firmware images(see Device Supported section below).

Parameters

<i>imageInfoSize</i> [↔ IN]	<ul style="list-style-type: none"> The size in BYTES of the image info array
<i>plmageInfo</i> [IN]	<ul style="list-style-type: none"> The image info list array containing information about the image to be deleted. See ImageElement

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Device Supported: MC83x5/SL9090
Timeout: 2 Secs

9.14.4.2 **ULONG** DownloadToSlot (**CHAR** * *path*, struct **slqsfwinfo_s** *fwlmglInfo*, struct **slqsfwinfo_s** *prilmglInfo*, **BYTE** *slot*, **BYTE** *forceDownload*)

This API is dedicated for downloading a firmware image into a specific slot. it is used for EM74xx/MC74xx only

Parameters

<i>path</i>	<ul style="list-style-type: none">fully qualified path to firmware image to download
<i>fwlmglInfo</i>	<ul style="list-style-type: none">See struct slqsfwinfo_s
<i>prilmglInfo</i>	<ul style="list-style-type: none">See struct slqsfwinfo_s
<i>slot</i>	<ul style="list-style-type: none">slot id in the modem to store the firmware
<i>forceDownload</i>	<ul style="list-style-type: none">a flag to force download take place.

Returns

See [qmerrno.h](#) for eQCWWAN_XXX error values

Note

only EM74xx/MC74xx is supported

9.14.4.3 **void** eGetDeviceSeries (struct **sGetDeviceSeriesResult** *)

Name : eGetDeviceSeries

Parameters

<i>none</i>	
-------------	--

Returns

[sGetDeviceSeriesResult](#)

Note

Get Devie Series

9.14.4.4 **ULONG** GetImagesPreference (**ULONG** * *plmageListSize*, struct **PrefImageList** * *plmageList*)

restore original alignment from stack Gets the current images preference from the device.

Parameters

<i>plmageList</i> ↔ <i>Size</i> [IN/OUT]	<ul style="list-style-type: none"> • Upon input, the size of structure ImageList ImageList • Upon successful output, the number of BYTES copied to the image list array
<i>plmageList</i> ↔ <i>UT</i>	<ul style="list-style-type: none"> • The caller must supply a pointer to a ImageList structure typecast as a BYTE pointer

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.14.4.5 ULONG GetImageStore (WORD *imageStorePathSize*, CHAR * *plmageStorePath*)

Returns the image store folder, i.e., the folder containing one or more carrier-specific image subfolders compatible with the currently connected QC WWAN device.

Parameters

<i>imageStore</i> ↔ <i>PathSize</i>	<ul style="list-style-type: none"> • Maximum number of characters (including NULL terminator) that can be copied to the image store path array.
<i>plmageStore</i> ↔ <i>Path</i> [OUT]	<ul style="list-style-type: none"> • The path to the image store

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.14.4.6 ULONG GetStoredImages (ULONG * *plmageListSize*, BYTE * *plmageList*)

restore original alignment from stack Gets the list of images stored on the device.

Parameters

<i>plmImageList</i> ↔ <i>Size</i> [IN/OUT]	<ul style="list-style-type: none"> • Upon input, the size of structure ImageList ImageList • Upon successful output, the number of BYTEs copied to the image list array
<i>plmImageList</i> [O↔ <i>UT</i>]	<ul style="list-style-type: none"> • The caller must supply a pointer to a ImageList structure typecast as a BYTE pointer

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Device Supported: MC83x5/SL9090
Timeout: 2 seconds

9.14.4.7 ULONG SetImagesPreference (ULONG *imageListSize*, BYTE * *plmImageList*, ULONG *bForceDownload*, BYTE *modemIndex*, ULONG * *plmImageTypesSize*, BYTE * *plmImageTypes*)

Sets the current images preference on the device. After this function successfully completes, the device must be reset for the selected image preference to be realized. Additionally, when the returned list of image types that require downloading is not empty, the device opens in QDL mode after the reset. At that point, the QDL portion of this API must be used to download the selected image preference to the device.

Parameters

<i>imageListSize</i>	<ul style="list-style-type: none"> • The size in BYTEs of the image list array
<i>plmImageList</i> [IN]	<ul style="list-style-type: none"> • The image info list array containing Image Elements <ul style="list-style-type: none"> – See PrefImageList
<i>bForce</i> ↔ <i>Download</i> [IN]	<ul style="list-style-type: none"> • Force device to download images from host? 0 - No Nonzero - Yes
<i>modemIndex</i>	<ul style="list-style-type: none"> • Desired storage index for downloaded modem image (optional, a value of 0xFF indicates unspecified)
<i>plmImageTypes</i> ↔ <i>Size</i> [IN/OUT]	<ul style="list-style-type: none"> • Upon input, maximum number of elements that download image types array can contain • Upon successful output, number of elements in download image types array
<i>plmImageTypes</i> [↔ <i>OUT</i>]	-The download image types array.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.14.4.8 ULONG SLQSDownloadFirmwareToSlot (CHAR * *pPath*, BYTE *slot_index*, BYTE *force_download*)

This API is used to download firmware to a specific slot id of the modem. It is only applicable for EM74xx variant. This API encapsulates all steps involved in the firmware download process. Hence it is a blocking API call. This API will not return until the entire process has been completed. This API will takes significant amount of time (in order of minutes, normally should be less than 10 minutes).

This API Performs the following steps:

1. Verifies arguments.
 2. Retrieve and store the details of the firmware and the PRI file
 3. Enable device state change callback.
 4. Enable firmware download callback.
 5. Set Image preference on the device and reset the device.
 6. Wait for the firmware to download and device to become ready.
 7. Check the firmware update status. If fail, return an error.
 8. If update status is OK, check if current image preference and preferred image preference(from step 2) match
 9. If match, firmware download is successful. otherwise, report FW_PREFERENCE_MISMATCH
 10. Disable callbacks and exit.
- The call to this API blocks until step 7 or 10. This could be a significant amount of time (in order of minutes). Also note that the device state change callback and firmware download callback are used internally within this API. Hence the user application's instance of these callbacks (if any) are cleared. The user must re-enable these callbacks after a call to this API in order to use them.

Parameters

<i>pPath</i> [IN]	<ul style="list-style-type: none"> fully qualified path to firmware image to download.
<i>slot_index</i> [IN]	<ul style="list-style-type: none"> slot id in the modem to store the firmware
<i>force_↔</i>	

Returns

- eQCWWAN_ERR_NONE - Firmware download/Switch success.
- eQCWWAN_ERR_INVALID_ARG - The path input does not contain any image
- eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL - Firmware download/switch failed
- eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH - Download success but device offline due to image preference mismatch (ref. syslogs for cause)

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: NA

9.14.4.9 ULONG SLQSGetBootVersionNumber (ULONG * bootversion)

Gets the boot loader version number

Parameters

<i>bootversion</i> [OUT]	<ul style="list-style-type: none">• boot loader version presented by a 4 byte integer
--------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Device Supported: MC9090/SL9090
Timeout: 2 seconds

9.14.4.10 ULONG SLQSGetFirmwareInfo (struct qmifwinfo_s * pinfo)

Returns firmware image information from the connected device

Parameters

<i>pinfo</i> [OUT]	<ul style="list-style-type: none">• firmware image information record
--------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values
struct [qmifwinfo_s](#)

Note

Timeout: 2 Seconds.

9.14.4.11 ULONG SLQSGetImageInfo (LPCSTR *path*, struct *qmifwinfo_s* * *pinfo*)

Returns firmware image information from a CWE file or mbn files stored on the host. For CWE, information is returned for the first CWE image found at the specified path. For MBN, the provided path must be located under the image store for the currently connected QC WWAN device. Note that as this API supports multiple firmware image types, it relies on the presence of a supported device. Otherwise, refer to SLQSGetImageInfoMC83xx and SLQSGetImageInfoMC77xx for APIs which do not rely on the presence of a supported device.

Parameters

<i>path</i> [IN]	<ul style="list-style-type: none">fully qualified path to folder containing CWE image or MBN imagesshould use a "/" at the end of the path.
<i>pinfo</i> [OUT]	<ul style="list-style-type: none">firmware image information record

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values
struct [qmifwinfo_s](#)

Note

Timeout: N/A

9.14.4.12 ULONG SLQSGetImageInfo_9x15 (LPCSTR *path*, BYTE *imgType*, struct *slqsfwinfo_s* * *pinfo*)

Returns firmware image information from a CWE file(s) stored on the host. It does not rely on the presence of a supported device.

Parameters

<i>path</i> [IN]	<ul style="list-style-type: none">• fully qualified path to folder containing the image(s)• should use a "/" at the end of the path.
<i>imgType</i> [IN]	<ul style="list-style-type: none">• 2 - Firmware Image(.cwe extension)• 3 - PRI Image (.nvu extension)
<i>pinfo</i> [OUT]	<ul style="list-style-type: none">• firmware image information record

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values
struct [qmifwinfo_s](#)

Note

Device Supported: MC73xx
Timeout: N/A

9.14.4.13 ULONG SLQSGetImageInfoMC77xx (LPCSTR *path*, struct *qmifwinfo_s* * *pinfo*)

Returns firmware image information from a SPKGS CWE file stored on the host. The information is returned for the first SPKGS CWE image found at the specified path. This API executes independent of a MC77xx being connected to the target.

Parameters

<i>path</i> [IN]	<ul style="list-style-type: none">• fully qualified path to folder containing SPKG CWE image• should use a "/" at the end of the path.
<i>pinfo</i> [OUT]	<ul style="list-style-type: none">• firmware image information record

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values
struct [qmifwinfo_s](#)

Note

Device Supported: MC77xx
Timeout: N/A

9.14.4.14 ULONG SLQSGetImageInfoMC83xx (LPCSTR *path*, struct qmifwinfo_s * *pinfo*)

Returns firmware image information from an MBN file located on the host. This API executes independent of a MC83xx being connected to the target.

Parameters

<i>path</i> [IN]	<ul style="list-style-type: none">fully qualified path to folder containing MBN fileshould use a "/" at the end of the path.
<i>pinfo</i> [OUT]	<ul style="list-style-type: none">firmware image information record

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values
struct [qmifwinfo_s](#)

Note

Device Supported: MC83xx/SL9090
Timeout: N/A

9.14.4.15 ULONG SLQSGetValidFwPriCombinations (struct ImageList * *pStoredImageList*, ULONG * *pValidCombinationSize*, struct SWI_STRUCT_CarrierImage * *pValidCombinations*)

This API distills valid Firmware/PRI combinations from GetStoredImages result

Parameters

in	<i>pStoredImage↔ List</i>	<ul style="list-style-type: none"> image list returned from GetStoredImages
in, out	<i>pValid↔ CombinationSize</i>	<ul style="list-style-type: none"> number of combination passed in and returned
out	<i>pValid↔ Combinations</i>	<ul style="list-style-type: none"> valid combinations returned

Returns

- eQCWWAN_ERR_INVALID_ARG - Invalid parameters
- eQCWWAN_ERR_BUFFER_SZ - No enough element to store combinatons returned

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.14.4.16 **BOOL** SLQSlSpkgFormatRequired (void)

Check if SPKG format download is required for SL9090/MC9090, it returns the value whcih was set by API [SLQ↔
SSetSpkgFormatRequired\(\)](#)

Parameters

<i>none</i>

Returns

return TRUE if required, otherwise, return FALSE

Note

Device Supported: MC9090/SL9090

9.14.4.17 **void** SLQSSetCrashStateCheckIgnore (**BOOL** ignore)

This API is used to set whether ignore crash state checking before proceed firmware download using the API [UpgradeFirmware2k\(\)](#).

Parameters

in	<i>ignore</i>	<ul style="list-style-type: none"> 0 - crash state checking applied (default value) 1 - ignore crash state checking
----	---------------	---

Note

Device Supported: EM73xx/MC73xx, EM74xx/MC74xx

9.14.4.18 ULONG SLQSSetSIMBasedImageSwitching (void)

This API is used to enable the SIM-based Image Switching. The modem will reboot automatically to take effect of the enabling

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Device Supported: EM74xx/MC74xx

9.14.4.19 void SLQSSetSpkgFormatRequired (BYTE *isneeded*)

Set if SPKG format download is required for SL9090/MC9090

Parameters

<i>isneeded</i>	<p>[INPUT] user inputs the firmware download method preference</p> <ul style="list-style-type: none">• 1 - Gobi3K download method, use mbn files. This is default value• 2 - SPKG download method, use cwe file
-----------------	--

Returns

None

Note

Device Supported: MC9090/SL9090

9.14.4.20 ULONG SLQSSwiGetAllCarrierImages (ULONG * *pNumOfItems*, struct SWI_STRUCT_CarrierImage * *pCarrierImages*, char * *pFolderPath*)

This API gets a list of all images stored on both the host and the device

Parameters

<i>pNumOfItems</i>	<ul style="list-style-type: none">• Number of Images{IN/OUT}
<i>pCarrier↔ Images[OUT]</i>	<ul style="list-style-type: none">• See SWI_STRUCT_CarrierImage
<i>pFolderPath</i>	<ul style="list-style-type: none">• Path of Input folder [IN]

Returns

TRUE/FALSE

Note

In case pFolderPath is invalid, API does not return invalid path error as SLQSSwiGetAllCarrierImages get carrier images from device also.

9.14.4.21 ULONG SLQSSwiGetFirmwareCurr (CurrentImgList * pCurrentImgList)

This API gets the currently active images on the device.

Parameters

<i>pCurrentImgList</i>	<ul style="list-style-type: none">• Pointer to structure CurrentImgList• See CurrentImgList for more information
------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Device Supported: MC73xx, MC74xx/EM74xx
Timeout: 5 Secs

9.14.4.22 ULONG SLQSUgradeFirmware9x15 (CHAR * *pDestinationPath*)

This API is used to upgrade firmware on a MC73xx device. This API encapsulates all steps involved in the firmware download process. It is an alternative to any firmware download application. Hence it is a blocking API call. This API will not return until the entire process has been completed.

This API Performs the following steps:

1. Verifies arguments.
2. Retrieve and store the details of the firmware and the PRI file
3. Enable device state change callback.
4. Enable firmware download callback.
5. Set Image preference on the device and reset the device.
6. Wait for the firmware to download and device to become ready.
7. Check the firmware update status. If fail, return an error.
8. If update status is OK, check if current image preference and preferred image preference(from step 2) match
9. If match, firmware download is successful.
10. If do not match, repeat from step 5 once more.
11. Disable callbacks and exit.
The call to this API blocks until step 7 or 11. This could be a significant amount of time (in order of minutes). Also note that the device state change callback and firmware download callback are used internally within this API. Hence the user application's instance of these callbacks (if any) are cleared. The user must re-enable these callbacks after a call to this API in order to use them.

Parameters

<i>pDestinationPath</i> <i>Path[IN]</i>	<ul style="list-style-type: none">• fully qualified path to firmware image to download. The path must end with a forward slash.
--	---

Returns

- eQCWWAN_ERR_NONE - Firmware download/Switch success.
- eQCWWAN_ERR_INVALID_ARG - The path input does not contain any image
- eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL - Firmware download/switch failed
- eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH - Download success but device of-line due to image preference mismatch (ref. syslogs for cause)

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: NA

9.14.4.23 ULONG upgrade_mc77xx_fw (LPCSTR path)

9.14.4.24 ULONG UpgradeFirmware2k (CHAR * pDestinationPath)

This API is used to download firmware to a MC77xx or Gobi 3000 device. For SL909/MC9090, [SLQSSetSpkgFormatRequired\(\)](#) needs to be called in advance to specify the download method, Gobi3K or SPKG download, please refer to the API [SLQSSetSpkgFormatRequired\(\)](#) for more details of the input values. If [SLQSSetSpkgFormatRequired\(\)](#) is not called in advance, it will use Gobi3K firmware download method (MBN files) as the default download method.

This API Performs the following steps:

1. Verifies arguments.
2. Verify that device Crash State should be 1 (RESET State).
3. Informs the SDK of the firmware upgrade path
4. Updates the images preference on the currently connected device.
5. Requests the device reset (device will reset after all open handles are released).
Upon successful completion, the above steps will have been completed, however, the actual upgrade of the firmware will necessarily then follow.

Parameters

<i>pDestinationPath</i> [IN]	<ul style="list-style-type: none">• fully qualified path to firmware image to download. The path must end with a forward slash. For a Gobi 3000 device the path should specify the carrier image folder index i.e. "<path>\to\carrier\image>\<carrier index>\" where <carrier index>=""> is a valid sub-directory entry. For 9x30 devices if pDestinationPath is not valid on host, it will use pseudo path for image switching.
------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

Note

Timeout: 12 seconds

9.15 qaGobiApilms.h File Reference

IMS Service API function prototypes.

Data Structures

- struct [SetSIPConfigReq](#)
- struct [SetSIPConfigResp](#)
- struct [SetRegMgrConfigReq](#)
- struct [SetRegMgrConfigResp](#)
- struct [SetIMSSMSConfigReq](#)
- struct [SetIMSSMSConfigResp](#)
- struct [SetIMSUserConfigReq](#)
- struct [SetIMSUserConfigResp](#)
- struct [SetIMSVoIPConfigReq](#)
- struct [SetIMSVoIPConfigResp](#)
- struct [GetSIPConfigResp](#)
- struct [GetRegMgrConfigParams](#)
- struct [GetIMSSMSConfigParams](#)
- struct [GetIMSUserConfigParams](#)
- struct [GetIMSVoIPConfigResp](#)
- struct [imsCfgIndRegisterInfo](#)

Functions

- [ULONG SLQSSetSIPConfig](#) ([SetSIPConfigReq](#) *pSetSIPConfigReq, [SetSIPConfigResp](#) *pSetSIPConfigResp)
- [ULONG SLQSSetRegMgrConfig](#) ([SetRegMgrConfigReq](#) *pSetRegMgrConfigReq, [SetRegMgrConfigResp](#) *pSetRegMgrConfigResp)
- [ULONG SLQSSetIMSSMSConfig](#) ([SetIMSSMSConfigReq](#) *pSetIMSSMSConfigReq, [SetIMSSMSConfigResp](#) *pSetIMSSMSConfigResp)
- [ULONG SLQSSetIMSUserConfig](#) ([SetIMSUserConfigReq](#) *pSetIMSUserConfigReq, [SetIMSUserConfigResp](#) *pSetIMSUserConfigResp)
- [ULONG SLQSSetIMSVoIPConfig](#) ([SetIMSVoIPConfigReq](#) *pSetIMSVoIPConfigReq, [SetIMSVoIPConfigResp](#) *pSetIMSVoIPConfigResp)
- [ULONG SLQSGetSIPConfig](#) ([GetSIPConfigResp](#) *pGetSIPConfigResp)
- [ULONG SLQSGetRegMgrConfig](#) ([GetRegMgrConfigParams](#) *pGetRegMgrConfigParams)
- [ULONG SLQSGetIMSSMSConfig](#) ([GetIMSSMSConfigParams](#) *pGetIMSSMSConfigParams)
- [ULONG SLQSGetIMSUserConfig](#) ([GetIMSUserConfigParams](#) *pGetIMSUserConfigParams)
- [ULONG SLQSGetIMSVoIPConfig](#) ([GetIMSVoIPConfigResp](#) *pGetIMSVoIPConfigResp)
- [ULONG SLQSImsConfigIndicationRegister](#) ([imsCfgIndRegisterInfo](#) *pImsCfgIndRegisterInfo)

9.15.1 Detailed Description

IMS Service API function prototypes.

9.15.2 Function Documentation

9.15.2.1 ULONG SLQSGetIMSSMSConfig (GetIMSSMSConfigParams * pGetIMSSMSConfigParams)

This API retrieves the SMS configuration parameters.

Parameters

<i>pGetIMSSMSConfigParams</i> [↔ IN/OUT]	<ul style="list-style-type: none">See GetIMSSMSConfigParams for more information
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Device Supported: MC73xx, MC74xx and EM74xx
Timeout: 5 seconds

9.15.2.2 ULONG SLQSGetIMSUserConfig (GetIMSUserConfigParams * pGetIMSUserConfigParams)

This API retrieves the IMS User configuration parameters.

Parameters

<i>pGetIMSUserConfigParams</i> [↔ IN/OUT]	<ul style="list-style-type: none">See GetIMSUserConfigParams for more information
---	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Device Supported: MC73xx, MC74xx and EM74xx
Timeout: 5 seconds

9.15.2.3 ULONG SLQSGetIMSVoIPConfig (GetIMSVoIPConfigResp * pGetIMSVoIPConfigResp)

This API retrieves the IMS VoIP configuration parameters.

Parameters

<i>GetIMSVoIPConfigResp</i> [OUT]	<ul style="list-style-type: none">• See GetIMSVoIPConfigResp for more information
-----------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Device Supported: MC73xx, MC74xx and EM74xx
Timeout: 5 seconds

9.15.2.4 ULONG SLQSGetRegMgrConfig (GetRegMgrConfigParams * pGetRegMgrConfigParams)

This API retrieves the registration manager configuration parameters.

Parameters

<i>pGetRegMgrConfigParams</i> [IN/OUT]	<ul style="list-style-type: none">• See GetRegMgrConfigResp for more information
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Device Supported: MC73xx, MC74xx and EM74xx
Timeout: 5 seconds

9.15.2.5 ULONG SLQSGetSIPConfig (GetSIPConfigResp * pGetSIPConfigResp)

This API retrieves the Session Initiation Protocol(SIP) configuration parameters.

Parameters

<i>pGetSIPConfigResp</i> [OUT]	<ul style="list-style-type: none">• See GetSIPConfigResp for more information
--------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
Device Supported: MC73xx, MC74xx and EM74xx
Timeout: 5 seconds

9.15.2.6 ULONG SLQSImsConfigIndicationRegister (imsCfgIndRegisterInfo * plmsCfgIndRegisterInfo)

Sets the registration state for different QMI_IMS indications for the requesting control point

Parameters

<i>plmsCfgIndRegisterInfo</i> [IN]	<ul style="list-style-type: none">• Structure containing Indication Register Information.<ul style="list-style-type: none">– See imsCfgIndRegisterInfo for more information.
------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
 Device Supported: MC73xx, MC74xx and EM74xx
 Timeout: 10 Secs

This API is used by a device to register/deregister for different QMI IMS indications. The device's registration state variables that control registration for indications will be modified to reflect the settings indicated in the request message. At least one optional parameter must be present in the request.

9.15.2.7 `ULONG SLQSSetIMSSMSConfig (SetIMSSMSConfigReq * pSetIMSSMSConfigReq, SetIMSSMSConfigResp * pSetIMSSMSConfigResp)`

This API sets the IMS SMS configuration parameters for the requesting control point.

Parameters

<i>pSetIMSSMSConfigReq</i> [IN]	<ul style="list-style-type: none"> See SetIMSSMSConfigReq for more information
<i>pSetIMSSMSConfigResp</i> [OUT]	<ul style="list-style-type: none"> See SetIMSSMSConfigResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
 Device Supported: MC73xx, MC74xx and EM74xx
 Timeout: 5 seconds

9.15.2.8 `ULONG SLQSSetIMSUserConfig (SetIMSUserConfigReq * pSetIMSUserConfigReq, SetIMSUserConfigResp * pSetIMSUserConfigResp)`

This API sets the IMS user configuration parameters for the requesting control point.

Parameters

<i>pSetIMSUserConfigReq</i> [IN]	<ul style="list-style-type: none"> See SetIMSUserConfigReq for more information
<i>pSetIMSUserConfigResp</i> [OUT]	<ul style="list-style-type: none"> See SetIMSUserConfigResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
 Device Supported: MC73xx, MC74xx and EM74xx
 Timeout: 5 seconds

9.15.2.9 `ULONG SLQSSetIMSVoIPConfig (SetIMSVoIPConfigReq * pSetIMSVoIPConfigReq, SetIMSVoIPConfigResp * pSetIMSVoIPConfigResp)`

This API sets the IMS Voice over Internet Protocol (VoIP) configuration parameters for the requesting control point.

Parameters

<i>pSetIMSVoIPConfigReq</i> [IN]	<ul style="list-style-type: none"> See SetIMSVoIPConfigReq for more information
<i>pSetIMSVoIPConfigResp</i> [OUT]	<ul style="list-style-type: none"> See SetIMSVoIPConfigResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
 Device Supported: MC73xx, MC74xx and EM74xx
 Timeout: 5 seconds

9.15.2.10 `ULONG SLQSSetRegMgrConfig (SetRegMgrConfigReq * pSetRegMgrConfigReq, SetRegMgrConfigResp * pSetRegMgrConfigResp)`

This API sets the IMS registration manager configuration parameters for the requesting control point.

Parameters

<i>pSetRegMgrConfigReq</i> [IN]	<ul style="list-style-type: none"> See SetRegMgrConfigReq for more information
<i>pSetRegMgrConfigResp</i> [OUT]	<ul style="list-style-type: none"> See SetRegMgrConfigResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
 Device Supported: MC73xx, MC74xx and EM74xx
 Timeout: 5 seconds

9.15.2.11 `ULONG SLQSSetSIPConfig (SetSIPConfigReq * pSetSIPConfigReq, SetSIPConfigResp * pSetSIPConfigResp)`

This API sets the IMS Session Initiation Protocol(SIP) configuration parameters for the requesting control point.

Parameters

<i>pSetSIPConfigReq</i> [IN]	<ul style="list-style-type: none"> See SetSIPConfigReq for more information
<i>pSetSIPConfigResp</i> [OUT]	<ul style="list-style-type: none"> See SetSIPConfigResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: NA
 Device Supported: MC73xx, MC74xx and EM74xx
 Timeout: 5 seconds

9.16 qaGobiApilmsa.h File Reference

IMS A Service API function prototypes.

Data Structures

- struct [IMSALndRegisterInfo](#)
- struct [SupportedMsgList](#)
- struct [IMSASupportedMsgInfo](#)
- struct [ReqFieldsList](#)
- struct [RespFieldsList](#)
- struct [LndFieldsList](#)
- struct [IMSASupportedFieldsResp](#)
- struct [IMSARegistrationStatus](#)
- struct [IMSAServiceStatus](#)

Functions

- [ULONG SLQSRegisterIMSIndication](#) ([IMSALndRegisterInfo](#) *pImsaLndRegisterInfo)
- [ULONG SLQSGetIMSASupportedMsg](#) ([IMSASupportedMsgInfo](#) *pIMSASupportedMsgInfo)
- [ULONG SLQSGetIMSASupportedFields](#) ([WORD](#) messageID, [IMSASupportedFieldsResp](#) *pIMSASupportedFieldsResp)
- [ULONG SLQSGetIMSARegStatus](#) ([IMSARegistrationStatus](#) *pIMSARegistrationStatus)
- [ULONG SLQSGetIMSAServiceStatus](#) ([IMSAServiceStatus](#) *pIMSAServiceStatus)

9.16.1 Detailed Description

IMS A Service API function prototypes.

9.16.2 Function Documentation

9.16.2.1 [ULONG SLQSGetIMSARegStatus](#) ([IMSARegistrationStatus](#) * *pIMSARegistrationStatus*)

Queries the set of messages implemented by the currently running software.

Parameters

<i>pIMSARegistrationStatus</i> [OUT]	<ul style="list-style-type: none">• Structure containing response parameters for registration status.<ul style="list-style-type: none">– See IMSARegistrationStatus for more information.
--------------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Secs

This API is used by a device to get the registration status for various IMS services for the requesting control point.

9.16.2.2 ULONG SLQSGetIMSAServiceStatus (IMSAServiceStatus * pIMSAServiceStatus)

Gets the service status for various IMS services for the requesting control point.

Parameters

<i>pIMSAServiceStatus</i> [OUT]	<ul style="list-style-type: none">• Structure containing response parameters for service status.<ul style="list-style-type: none">– See IMSAServiceStatus for more information.
---------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Secs

This API is used by a device to Gets the service status for various IMS services for the requesting control point.

9.16.2.3 ULONG SLQSGetIMSASupportedFields (WORD *messageID*, IMSASupportedFieldsResp * *pIMSASupportedFieldsResp*)

Queries the set of supported fields implemented by the currently running software.

Parameters

<i>messageID</i> [IN]	<ul style="list-style-type: none"> Service Message ID.
<i>pIMSASupportedFieldsResp</i> [OUT]	<ul style="list-style-type: none"> Structure containing Supported Fields Response. <ul style="list-style-type: none"> See IMSASupportedFieldsResp for more information.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Secs

This API is used by a device to query the fields supported for a single command as implemented by the currently running software.

9.16.2.4 ULONG SLQSGetIMSASupportedMsg (IMSASupportedMsgInfo * *pIMSASupportedMsgInfo*)

Queries the set of messages implemented by the currently running software.

Parameters

<i>pIMSASupportedMsgInfo</i> [OUT]	<ul style="list-style-type: none"> Structure containing Supported Messages Information. <ul style="list-style-type: none"> See IMSASupportedMsgInfo for more information.
------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Secs

This API is used by a device to query the set of messages implemented by the currently running software

9.16.2.5 ULONG SLQSRegisterIMSAIndication (IMSAIndRegisterInfo * *plmsalndRegisterInfo*)

Sets the registration state for different QMI_IMSA indications for the requesting control point

Parameters

<i>plmsalndRegisterInfo</i> [IN]	<ul style="list-style-type: none"> Structure containing Indication Register Information. <ul style="list-style-type: none"> See IMSAIndRegisterInfo for more information.
----------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Secs

This API is used by a device to register/deregister for different QMI_IMSA indications.
The device's registration state variables that control registration for indications will be modified to reflect the settings indicated in the request message.
At least one optional parameter must be present in the request.

9.17 qaGobiApiLoc.h File Reference

Location API function prototypes.

Data Structures

- struct [LOCEventRegisterReqResp](#)
- struct [LOCExtPowerStateReqResp](#)
- struct [LocApplicationInfo](#)
- struct [LOCStartReq](#)
- struct [LOCStopReq](#)
- struct [SV](#)
- struct [SVInfo](#)
- struct [GnssData](#)
- struct [CellIdb](#)

- struct [ClkInfo](#)
- struct [BdsSV](#)
- struct [BdsSVInfo](#)
- struct [LocDelAssDataReq](#)
- struct [SwiLocGetAutoStartResp](#)
- struct [SwiLocSetAutoStartReq](#)
- struct [altitudeSrcInfo](#)
- struct [LocInjectPositionReq](#)
- struct [LocSetCradleMountReq](#)
- struct [sensorData](#)
- struct [tempratureData](#)
- struct [LocInjectSensorDataReq](#)

Macros

- `#define MAX_SENSOR_DATA_LEN 64`
- `#define MAX_TEMP_DATA_LEN 64`

Functions

- [ULONG SLQSLOCEventRegister](#) ([LOCEventRegisterReqResp](#) *pLOCEventRegisterReqResp)
- [ULONG SLQSLOCSetExtPowerState](#) ([LOCExtPowerStateReqResp](#) *pLOCExtPowerStateReqResp)
- [ULONG SLQSLOCStart](#) ([LOCStartReq](#) *pLOCStartReq)
- [ULONG SLQSLOCStop](#) ([LOCStopReq](#) *pLOCStopReq)
- [ULONG SLQSLOCSetOpMode](#) ([ULONG](#) mode)
- [ULONG SLQSLOCDelAssData](#) ([LocDelAssDataReq](#) request)
- [ULONG SwiLocGetAutoStart](#) ([SwiLocGetAutoStartResp](#) *resp)
- [ULONG SwiLocSetAutoStart](#) ([SwiLocSetAutoStartReq](#) *req)
- [ULONG SLQSLOCInjectUTCTime](#) ([ULONGLONG](#) timeMsec, [ULONG](#) timeUncMsec)
- [ULONG SLQSLOCInjectPosition](#) ([LocInjectPositionReq](#) *pLocInjectPositionReq)
- [ULONG SLQSLOCSetCradleMountConfig](#) ([LocSetCradleMountReq](#) *pLocSetCradleMountReq)
- [ULONG SLQSLOCInjectSensorData](#) ([LocInjectSensorDataReq](#) *pLocInjectSensorDataReq)
- [ULONG SLQSLOCGetBestAvailPos](#) ([ULONG](#) xid)

9.17.1 Detailed Description

Location API function prototypes.

9.17.2 Macro Definition Documentation

9.17.2.1 `#define MAX_SENSOR_DATA_LEN 64`

9.17.2.2 `#define MAX_TEMP_DATA_LEN 64`

9.17.3 Function Documentation

9.17.3.1 [ULONG SLQSLOCDelAssData](#) ([LocDelAssDataReq](#) request)

Used by the control point to delete the location engine assistance data

Parameters

<i>request</i> [IN]	<ul style="list-style-type: none">request structure parameters should contain all NULL pointers to delete all assistance data. Otherwise, specify optional fields to be deleted. See LocDelAssDataReq for more information
---------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.17.3.2 ULONG SLQSLocEventRegister (LOcEventRegisterReqResp * pLOcEventRegisterReqResp)

Used by the control point to register for events from the location subsystem.

Parameters

<i>pLOcEventRegisterReqResp</i> [IN]	<ul style="list-style-type: none">See LOcEventRegisterReqResp for more information
--------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.17.3.3 ULONG SLQSLocGetBestAvailPos (ULONG xid)

Control point to get the best available position estimate from the location engine.

Parameters

<i>xid</i> [IN]	<ul style="list-style-type: none">Identifies the transaction.
	<ul style="list-style-type: none">The transaction ID is returned in the Get Best Available Position indication.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

- GPS engine should be started to get best available position.

9.17.3.4 ULONG SLQSLOCInjectPosition (LocInjectPositionReq * pLocInjectPositionReq)

Injects a position to the location engine.

Parameters

<i>pLocInjectPositionReq</i> [IN]	<ul style="list-style-type: none">• See LocInjectPositionReq for more information
-----------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.17.3.5 ULONG SLQSLOCInjectSensorData (LocInjectSensorDataReq * pLocInjectSensorDataReq)

Control point to to inject sensor data into the GNSS location engine.

Parameters

<i>pLocInjectSensorDataReq</i> [IN]	<ul style="list-style-type: none">• See LocInjectSensorDataReq for more information
-------------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.17.3.6 ULONG SLQSLOCInjectUTCtime (ULONGLONG timeMsec, ULONG timeUncMsec)

Injects UTC time in the location engine.

Parameters

<i>timeMsec</i> [IN]	<ul style="list-style-type: none">The UTC time since Jan. 1, 1970
<i>timeUncMsec</i> [↔ IN]	<ul style="list-style-type: none">The time Uncertainty

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.17.3.7 ULONG SLQSLOCSetCradleMountConfig (LocSetCradleMountReq * pLocSetCradleMountReq)

Control point to set the current cradle mount configuration.

Parameters

<i>pLocSetCradleMountReq</i> [↔ IN]	<ul style="list-style-type: none">See LocSetCradleMountReq for more information
--	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.17.3.8 ULONG SLQSLOCSetExtPowerState (LOCExtPowerStateReqResp * pLOCExtPowerStateReqResp)

Used by the control point to set the current external power configuration.

Parameters

<i>pLOCExtPowerStateReqResp</i> [IN]	<ul style="list-style-type: none"> See LOCExtPowerStateReqResp for more information
--------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.17.3.9 ULONG SLQSLOCSetOpMode (ULONG mode)

Used by the control point to tells the engine to use the specified operation mode while making the position fixes

Parameters

<i>mode</i> [IN]	<ul style="list-style-type: none"> Valid values: <ul style="list-style-type: none"> eQMI_LOC_OPER_MODE_DEFAULT (1) - Use the default engine mode eQMI_LOC_OPER_MODE_MSB (2) - Use the MS-based mode eQMI_LOC_OPER_MODE_MSA (3) - Use the MS-assisted mode eQMI_LOC_OPER_MODE_STANDALONE (4) - Use Standalone mode eQMI_LOC_OPER_MODE_CELL_ID (5) - Use cell ID; this mode is only valid for GSM/UMTS networks eQMI_LOC_OPER_MODE_WWAN (6) - Use WWAN measurements to calculate the position; if this mode is set, AFLT will be used for 1X networks and OTDOA will be used for LTE networks
Generated by Doxygen	

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.17.3.10 ULONG SLQSLOCStart (LOCStartReq * pLOCStartReq)

Used by the control point to initiate a GPS session.

Parameters

<i>pLOCStartReq</i> <i>ReqReq[IN]</i>	<ul style="list-style-type: none">• See LOCStartReq for more information
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.17.3.11 ULONG SLQSLOCStop (LOCStopReq * pLOCStopReq)

Used by the control point to stop a GPS session.

Parameters

<i>pLOCStopReq</i> <i>Resp[IN]</i>	<ul style="list-style-type: none">• See LOCStopReq for more information
---------------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.17.3.12 **ULONG** SwiLocGetAutoStart (**SwiLocGetAutoStartResp** * *resp*)

Used by the control point to Get Loc Auto Start settings

Parameters

<i>resp</i> [OUT]	<ul style="list-style-type: none">See SwiLocGetAutoStartResp for more information
-------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.17.3.13 **ULONG** SwiLocSetAutoStart (**SwiLocSetAutoStartReq** * *req*)

Used by the control point to Set Loc Auto Start settings

Parameters

<i>req</i> [IN]	<ul style="list-style-type: none">See SwiLocSetAutoStartReq for more information
-----------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.18 qaGobiApiNas.h File Reference

Network Access Service API function prototypes.

Data Structures

- struct [RFBandInfoElements](#)
- struct [servSystem](#)
- struct [dataSrvCapabilities](#)
- struct [currentPLMN](#)
- struct [roamIndList](#)
- struct [qaQmi3Gpp2TimeZone](#)
- struct [detailSvcInfo](#)
- struct [CDMASysInfoExt](#)
- struct [callBarStatus](#)
- struct [qaQmiServingSystemParam](#)
- struct [rxSignalStrengthListElement](#)
- struct [ecioListElement](#)
- struct [errorRateListElement](#)
- struct [rsrqInformation](#)
- struct [lteSnrInformation](#)
- struct [lteRsrpInformation](#)
- struct [slqsSignalStrengthInfo](#)
- struct [SlqsNas3GppNetworkInfo](#)
- struct [_SlqsNas3GppNetworkRAT_](#)
- struct [SlqsNasPcsDigit](#)
- struct [_slqsNetworkScanInfo](#)
- struct [netSelectionPref](#)
- struct [acqOrderPref](#)
- struct [CSGID](#)
- struct [_sysSelectPrefParams](#)
- struct [_sysSelectPrefInfo](#)
- struct [SrvStatusInfo](#)
- struct [GSMSrvStatusInfo](#)
- struct [sysInfoCommon](#)
- struct [CDMASysInfo](#)
- struct [HDRSysInfo](#)
- struct [GSMSysInfo](#)
- struct [WCDMASysInfo](#)
- struct [LTESysInfo](#)
- struct [AddCDMASysInfo](#)
- struct [AddSysInfo](#)
- struct [CallBarringSysInfo](#)
- struct [nasGetSysInfoResp](#)
- struct [CommInfo](#)
- struct [LTEInfo](#)
- struct [swiModemStatusResp](#)
- struct [nasGetHDRColorCodeResp](#)
- struct [nasGetTxRxInfoReq](#)
- struct [rxInfo](#)
- struct [txInfo](#)
- struct [nasGetTxRxInfoResp](#)
- struct [CDMASSInfo](#)
- struct [HDRSSInfo](#)
- struct [LTESInfo](#)
- struct [TDSCDMASigInfoExt](#)
- struct [nasGetSigInfoResp](#)
- struct [nasIndicationRegisterReq](#)
- struct [nasPLMNNameReq](#)

- struct [nasPLMNNameResp](#)
- struct [OperatorPLMNData](#)
- struct [operatorPLMNList](#)
- struct [serviceProviderName](#)
- struct [PLMNNetworkNameData](#)
- struct [PLMNNetworkName](#)
- struct [operatorNameString](#)
- struct [nasOperatorNameResp](#)
- struct [nasGet3GPP2SubscriptionInfoReq](#)
- struct [namName](#)
- struct [dirNum](#)
- struct [sidNid](#)
- struct [homeSIDNID](#)
- struct [minBasedIMSI](#)
- struct [trueIMSI](#)
- struct [CDMAChannel](#)
- struct [nasGet3GPP2SubscriptionInfoResp](#)
- struct [nmrCellInfo](#)
- struct [GERANInfo](#)
- struct [geranInstInfo](#)
- struct [UMTSinstInfo](#)
- struct [UMTSInfo](#)
- struct [CDMAInfo](#)
- struct [cellParams](#)
- struct [LTEInfoIntrafreq](#)
- struct [infoInterFreq](#)
- struct [LTEInfoInterfreq](#)
- struct [gsmCellInfo](#)
- struct [lteGsmCellInfo](#)
- struct [LTEInfoNeighboringGSM](#)
- struct [wcdmaCellInfo](#)
- struct [lteWcdmaCellInfo](#)
- struct [LTEInfoNeighboringWCDMA](#)
- struct [umtsLTENbrCell](#)
- struct [WCDMAInfoLTENeighborCell](#)
- struct [nasCellLocationInfoResp](#)
- struct [MNRInfo](#)
- struct [nasInitNetworkReg](#)
- struct [protocolSubtypeElement](#)
- struct [HDRPersonalityResp](#)
- struct [HDRProtSubtypResp](#)
- struct [PSDetachReq](#)
- struct [GetErrRateResp](#)
- struct [DRCParams](#)
- struct [PilotSetParams](#)
- struct [PilotSetData](#)
- struct [GetHRPDStatsResp](#)
- struct [ActPilotPNElement](#)
- struct [NetworkStat1x](#)
- struct [NetworkStatEVDO](#)
- struct [DeviceConfigDetail](#)
- struct [DataStatusDetail](#)
- struct [NetworkDebugResp](#)
- struct [LteCQIParm](#)
- struct [RSSIThresh](#)

- struct [ECIOThresh](#)
- struct [HDSINRThresh](#)
- struct [LTESNRThresh](#)
- struct [IOTThresh](#)
- struct [RSRQThresh](#)
- struct [RSRPThresh](#)
- struct [LTSigRptCfg](#)
- struct [TDSCDMASINRCONFThresh](#)
- struct [sigInfo](#)
- struct [NasSwiIndReg](#)
- struct [CDMARSSIThresh](#)
- struct [CDMAECIOThresh](#)
- struct [HRRSSIThresh](#)
- struct [HDRECIOThresh](#)
- struct [HDSINRThreshold](#)
- struct [HDRIOThresh](#)
- struct [GSMRSSIThresh](#)
- struct [WCDMARSSIThresh](#)
- struct [WCDMAECIOThresh](#)
- struct [LTERSSIThresh](#)
- struct [LTESNRThreshold](#)
- struct [LTERSRQThresh](#)
- struct [LTERSRPThresh](#)
- struct [LTSigRptConfig](#)
- struct [TDSCDMARSCPThresh](#)
- struct [TDSCDMARSSIThresh](#)
- struct [TDSCDMAECIOThresh](#)
- struct [TDSCDMASINRThresh](#)
- struct [setSignalStrengthInfo](#)
- struct [PhyCaAggScellIndType](#)
- struct [PhyCaAggScellIDBw](#)
- struct [PhyCaAggScellInfo](#)
- struct [PhyCaAggPcellInfo](#)
- struct [PhyCaAggScellIndex](#)
- struct [nasGetLTECphyCaResp](#)
- struct [nasGetLTECphyCa](#)
- struct [wcdmaUARFCN](#)
- struct [lteEARFCN](#)
- struct [ltePCI](#)
- struct [nasSwiGetChannelLockResp](#)
- struct [nasSwiSetChannelLockReq](#)
- struct [timeInfo](#)
- struct [GetNetworkTimeResp](#)
- struct [RxSigInfo](#)
- struct [SccRxInfo](#)
- struct [LteSccRxInfoResp](#)

Macros

- #define [SLQS_SS_INFO_LIST_MAX_ELEMENTS](#) 18
- #define [MAX_DESCRIPTION_LENGTH](#) 255
- #define [SLQS_SYSTEM_ID_SIZE](#) 16
- #define [PLMN_LENGTH](#) 3
- #define [MAX_SERV_SYSTEM_RADIO_INTERFACES](#) 0x0A
- #define [MAX_DATA_SRV_CAPABILITIES](#) 0x20
- #define [NAM_NAME_LENGTH](#) 12
- #define [IMSI_M_S1_LENGTH](#) 7
- #define [IMSI_M_S2_LENGTH](#) 3
- #define [MAX_PILOT_SETS](#) 0xFF
- #define [UATISIZE](#) 16
- #define [NAS_SIG_INFO_MAX_TDSCDMA_THRESHOLDS_LIST_SIZE](#) 16
- #define [NAS_SIG_INFO_MIN_dBm_FLOAT_VALUE](#) -125.0
- #define [NAS_SIG_INFO_MIN_dB_FLOAT_VALUE](#) -10.0
- #define [NAS_MAX_SCC_RX_INFO_INSTANCES](#) 255

Typedefs

- typedef struct [_SlqsNas3GppNetworkRAT](#) [SlqsNas3GppNetworkRAT](#)
- typedef struct [_slqsNetworkScanInfo](#) [slqsNetworkScanInfo](#)
- typedef struct [_sysSelectPrefParams](#) [sysSelectPrefParams](#)
- typedef struct [_sysSelectPrefInfo](#) [sysSelectPrefInfo](#)

Enumerations

- enum [_NAMS_RADIO_IF_TECHNOLOGY](#) {
[eNAS_RADIO_IF_GSM](#) = 0x04,
[eNAS_RADIO_IF_UMTS](#) = 0x05,
[eNAS_RADIO_IF_LTE](#) = 0x08,
[eNAS_RADIO_IF_TDSCDMA](#) = 0x09 }
- enum [NAS_LTE_CPHY_SCELL_STATE](#) {
[eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED](#) =0x00,
[eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED](#) =0x01,
[eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED](#) =0x02 }
- enum [NAS_LTE_CPHY_CA_BW_NRB](#) {
[eNAS_LTE_CPHY_CA_BW_NRB_6](#) =0x00,
[eNAS_LTE_CPHY_CA_BW_NRB_15](#) =0x01,
[eNAS_LTE_CPHY_CA_BW_NRB_25](#) =0x02,
[eNAS_LTE_CPHY_CA_BW_NRB_50](#) =0x03,
[eNAS_LTE_CPHY_CA_BW_NRB_75](#) =0x04,
[eNAS_LTE_CPHY_CA_BW_NRB_100](#) =0x05 }
- enum [eSYS_SRV_DOMAIN](#) {
[eSYS_SRV_DOMAIN_NO_SRV](#) = 0x00,
[eSYS_SRV_DOMAIN_CS_ONLY](#) = 0x01,
[eSYS_SRV_DOMAIN_PS_ONLY](#) = 0x02,
[eSYS_SRV_DOMAIN_CS_PS](#) = 0x03,
[eSYS_SRV_DOMAIN_CAMPED](#) = 0x04,
[eSYS_SRV_DOMAIN_UNKNOWN](#) }

Functions

- [ULONG GetSignalStrengths](#) ([ULONG](#) *pArraySizes, [INT8](#) *pSignalStrength, [ULONG](#) *pRadioInterface)
- [ULONG PerformNetworkScan](#) ([BYTE](#) *pInstanceSize, [BYTE](#) *pInstances)
- [ULONG InitiateNetworkRegistration](#) ([ULONG](#) regType, [WORD](#) mcc, [WORD](#) mnc, [ULONG](#) rat)
- [ULONG GetServingNetwork](#) ([ULONG](#) *pRegistrationState, [ULONG](#) *pCSDomain, [ULONG](#) *pPSDomain, [ULONG](#) *pRAN, [BYTE](#) *pRadiolfacesSize, [BYTE](#) *pRadiolfaces, [ULONG](#) *pRoaming, [WORD](#) *pMCC, [WORD](#) *pMNC, [BYTE](#) nameSize, [CHAR](#) *pName)
- [ULONG GetHomeNetwork](#) ([WORD](#) *pMCC, [WORD](#) *pMNC, [BYTE](#) nameSize, [CHAR](#) *pName, [WORD](#) *pSID, [WORD](#) *pNID)
- [ULONG GetServingNetworkCapabilities](#) ([BYTE](#) *pDataCapsSize, [BYTE](#) *pDataCaps)
- [ULONG SetNetworkPreference](#) ([ULONG](#) technologyPref, [ULONG](#) duration)
- [ULONG GetNetworkPreference](#) ([ULONG](#) *pTechnologyPref, [ULONG](#) *pDuration, [ULONG](#) *pPersistentTechnologyPref)
- [ULONG GetRFInfo](#) ([BYTE](#) *pInstanceSize, struct [RFBandInfoElements](#) *pRFBandInfo)
- [ULONG InitiateDomainAttach](#) ([ULONG](#) action)
- [ULONG GetACCOLC](#) ([BYTE](#) *pACCOLC)
- [ULONG SetACCOLC](#) ([CHAR](#) *spc, [BYTE](#) acccolc)
- [ULONG SetCDMANetworkParameters](#) ([CHAR](#) *pSPC, [BYTE](#) *pForceRev0, [BYTE](#) *pCustomSCP, [ULONG](#) *pProtocol, [ULONG](#) *pBroadcast, [ULONG](#) *pApplication, [ULONG](#) *pRoaming)
- [ULONG GetCDMANetworkParameters](#) ([BYTE](#) *pSCI, [BYTE](#) *pSCM, [BYTE](#) *pRegHomeSID, [BYTE](#) *pRegForeignSID, [BYTE](#) *pRegForeignNID, [BYTE](#) *pForceRev0, [BYTE](#) *pCustomSCP, [ULONG](#) *pProtocol, [ULONG](#) *pBroadcast, [ULONG](#) *pApplication, [ULONG](#) *pRoaming)
- [ULONG GetANAAAAAuthenticationStatus](#) ([ULONG](#) *pStatus)
- [ULONG SLQSGetServingSystem](#) ([qaQmiServingSystemParam](#) *pServingSystem)
- [ULONG SLQSSetBandPreference](#) ([ULONGLONG](#) bandpreference)
- [ULONG SLQSNasIndicationRegister](#) ([BYTE](#) systemSelectionInd, [BYTE](#) DDTMInd, [BYTE](#) servingSystemInd)
- [ULONG SLQSGetSignalStrength](#) (struct [slqsSignalStrengthInfo](#) *pSignalInfo)
- [ULONG SLQSPerformNetworkScan](#) ([slqsNetworkScanInfo](#) *pNetworkInfo)
- [ULONG SLQSSetSysSelectionPref](#) ([sysSelectPrefParams](#) *pSysSelectPrefParams)
- [ULONG SLQSGetSysSelectionPref](#) ([sysSelectPrefInfo](#) *pSysSelectPrefInfo)
- [ULONG SLQSNasGetSysInfo](#) ([nasGetSysInfoResp](#) *pGetSysInfoResp)
- [ULONG SLQSNasSwiModemStatus](#) ([swiModemStatusResp](#) *pModemStatusResp)
- [ULONG SLQSNasGetHDRColorCode](#) ([nasGetHDRColorCodeResp](#) *pGetHDRColorCodeResp)
- [ULONG SLQSNasGetTxRxInfo](#) ([nasGetTxRxInfoReq](#) *pGetTxRxInfoReq, [nasGetTxRxInfoResp](#) *pGetTxRxInfoResp)
- [ULONG SLQSNasGetSigInfo](#) ([nasGetSigInfoResp](#) *pGetSigInfoResp)
- [ULONG SLQSNasIndicationRegisterExt](#) ([nasIndicationRegisterReq](#) *pIndicationRegisterReq)
- [ULONG SLQSGetPLMNName](#) ([nasPLMNNameReq](#) *pPLMNNameReq, [nasPLMNNameResp](#) *pPLMNNameResp)
- [ULONG SLQSGetOperatorNameData](#) ([nasOperatorNameResp](#) *pOperatorNameData)
- [ULONG SLQSNasGet3GPP2Subscription](#) ([nasGet3GPP2SubscriptionInfoReq](#) *pGet3GPP2SubsInfoReq, [nasGet3GPP2SubscriptionInfoResp](#) *pGet3GPP2SubsInfoResp)
- [ULONG SLQSNasGetCellLocationInfo](#) ([nasCellLocationInfoResp](#) *pNasCellLocationInfoResp)
- [ULONG SLQSInitiateNetworkRegistration](#) ([nasInitNetworkReg](#) *pNasInitNetRegistrationReg)
- [ULONG SLQSSwiGetHDRPersonality](#) ([HDRPersonalityResp](#) *pHDRPersonalityResp)
- [ULONG SLQSSwiGetHDRProtSubtype](#) ([HDRProtSubtypResp](#) *pHDRProtSubtypResp)
- [ULONG SLQSSwiPSDetach](#) ([PSDetachReq](#) *pPSDetachReq)
- [ULONG SLQSGetErrorRate](#) ([GetErrRateResp](#) *pGetErrRateResp)
- [ULONG SLQSSwiGetHRPDStats](#) ([GetHRPDStatsResp](#) *pGetHRPDStatsResp)
- [ULONG SLQSSwiNetworkDebug](#) ([NetworkDebugResp](#) *pNetworkDebugResp)
- [ULONG SLQSSwiGetLteCQI](#) ([LteCQIParm](#) *pLteCQIResp)
- [ULONG SLQSConfigSigInfo](#) ([sigInfo](#) *pSigInfo)
- [ULONG SLQSNasSwiIndicationRegister](#) ([NasSwiIndReg](#) *pIndRegReq)

- [ULONG GetHomeNetwork3GPP2](#) ([WORD *pMCC](#), [WORD *pMNC](#), [BYTE nameSize](#), [CHAR *pName](#), [WORD *pSID](#), [WORD *pNID](#), [WORD *pNw2MCC](#), [WORD *pNw2MNC](#), [BYTE *pNw2DescDisp](#), [BYTE *pNw2DescEnc](#), [BYTE *pNw2DescLen](#), [BYTE *pNw2Name](#))
- [ULONG SLQSNasConfigSigInfo2](#) ([setSignalStrengthInfo *pSetSignalStrengthInfo](#))
- [ULONG SLQSNASGetLTECPHYCaInfo](#) ([nasGetLTECphyCa *pLTECPhyCa](#))
- [ULONG SLQSNasIndicationRegisterLTECphyCa](#) ([BYTE *bStatus](#))
- [ULONG SLQSNASSwiGetChannelLock](#) ([nasSwiGetChannelLockResp *pNasSwiGetChannelLockResp](#))
- [ULONG SLQSNASSwiSetChannelLock](#) ([nasSwiSetChannelLockReq *pNasSwiSetChannelLockReq](#))
- [ULONG SLQSGetNetworkTime](#) ([GetNetworkTimeResp *pGetNetworkTimeResp](#))
- [ULONG SLQSSwiGetLteSccRxInfo](#) ([LteSccRxInfoResp *pLteSccRxInfoResp](#))

9.18.1 Detailed Description

Network Access Service API function prototypes.

9.18.2 Macro Definition Documentation

9.18.2.1 `#define IMSI_M_S1_LENGTH 7`

9.18.2.2 `#define IMSI_M_S2_LENGTH 3`

9.18.2.3 `#define MAX_DATA_SRV_CAPABILITIES 0x20`

9.18.2.4 `#define MAX_DESCRIPTION_LENGTH 255`

9.18.2.5 `#define MAX_PILOT_SETS 0xFF`

9.18.2.6 `#define MAX_SERV_SYSTEM_RADIO_INTERFACES 0x0A`

9.18.2.7 `#define NAM_NAME_LENGTH 12`

9.18.2.8 `#define NAS_MAX_SCC_RX_INFO_INSTANCES 255`

9.18.2.9 `#define NAS_SIG_INFO_MAX_TDSCDMA_THRESHOLDS_LIST_SIZE 16`

9.18.2.10 `#define NAS_SIG_INFO_MIN_dB_FLOAT_VALUE -10.0`

9.18.2.11 `#define NAS_SIG_INFO_MIN_dBm_FLOAT_VALUE -125.0`

9.18.2.12 `#define PLMN_LENGTH 3`

9.18.2.13 `#define SLQS_SS_INFO_LIST_MAX_ELEMENTS 18`

9.18.2.14 `#define SLQS_SYSTEM_ID_SIZE 16`

9.18.2.15 `#define UATISIZE 16`

9.18.3 Typedef Documentation

9.18.3.1 `typedef struct _SlqsNas3GppNetworkRAT SlqsNas3GppNetworkRAT`

Contain the 3GPP radio access technology information.

Parameters

<i>MCC</i>	<ul style="list-style-type: none"> • Mobile Country Code
<i>MNC</i>	<ul style="list-style-type: none"> • Mobile Network Code
<i>RAT</i>	<ul style="list-style-type: none"> • Radio Access Technology <ul style="list-style-type: none"> – 0x04 - GERAN – 0x05 - UMTS – 0x08 - LTE – 0x09 - TD-SCDMA

9.18.3.2 typedef struct _slqsNetworkScanInfo slqsNetworkScanInfo

Contain the network scan information.

Parameters

<i>pNetworkInfo</i> ↔ <i>Instances</i> [IN↔ OUT]	<ul style="list-style-type: none"> • Upon input, maximum number of elements that the network info instance array can contain. • Upon successful output, the actual number of elements in the network info instance array.
<i>pNetworkInfo</i> ↔ OUT]	<ul style="list-style-type: none"> • Network info instance array <ul style="list-style-type: none"> – See SlqsNas3GppNetworkInfo for more information
<i>pRAT</i> ↔ <i>Instances</i> [IN↔ OUT]	<ul style="list-style-type: none"> • Upon input, maximum number of elements that the RAT info instance array can contain. • Upon successful output, the actual number of elements in the RAT info instance array.
<i>pRATInfo</i> [OUT]	<ul style="list-style-type: none"> • RAT info instance array <ul style="list-style-type: none"> – See SlqsNas3GppNetworkRAT for more information
<i>pPCSDigit</i> ↔ <i>Instances</i> [IN↔ OUT]	<ul style="list-style-type: none"> • Upon input, maximum number of elements that the PCS Digit info instance array can contain. • Upon successful output, the actual number of elements in the PCS Digit info instance array.
<i>pPCSDigitInfo</i> ↔ OUT]	<ul style="list-style-type: none"> • PCS Digit info instance array <ul style="list-style-type: none"> – See SlqsNasPcsDigit for more information
<i>pScanResult</i> ↔ OUT]	<ul style="list-style-type: none"> • status of network scan • 0x00 - scan successful • 0x01 - scan was aborted • 0x02 - scan did not complete due to a radio link failure recovery in progress

9.18.3.3 typedef struct _sysSelectPrefInfo sysSelectPrefInfo

Structure for storing the current preferred system selection settings for the device.

Parameters

<i>pEmerMode</i>	<ul style="list-style-type: none">• Optional parameter specifying the emergency Mode• Values:<ul style="list-style-type: none">– 0 - OFF (normal)– 1 - ON (Emergency)• function SLQSGetSysSelectionPref() returns a default value FF if no value is returned by the device.
<i>pModePref</i>	<ul style="list-style-type: none">• Optional parameter• Bit Mask indicating the radio technology mode preference• Bit values:<ul style="list-style-type: none">– Bit 0 - cdma2000 1x– Bit 1 - cdma2000 HRPD(1xEV-DO)– Bit 2 - GSM– Bit 3 - UMTS– Bit 4 - LTE• function SLQSGetSysSelectionPref() returns a default value FF if no value is returned by the device.

<i>pBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the band preference • Bit values: <ul style="list-style-type: none"> – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved • function SLQSGetSysSelectionPref() returns a default value FFFFFFFFFFFFFFFF if no value is returned by the device.
------------------	--

<i>pPRLPref</i>	<ul style="list-style-type: none">• Optional parameter indicating the CDMA PRL Preference• Values:<ul style="list-style-type: none">– 0x0001 - Acquire available system only on the A side– 0x0002 - Acquire available system only on the B side– 0x3FFF - Acquire any available systems• function SLQSGetSysSelectionPref() returns a default value FFFF if no value is returned by the device.
<i>pRoamPref</i>	<ul style="list-style-type: none">• Optional parameter indicating the roaming Preference• Values:<ul style="list-style-type: none">– 0x01 - Acquire only systems for which the roaming indicator is off– 0x02 - Acquire a system as long as its roaming indicator is not off– 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only– 0xFF - Acquire systems, regardless of their roaming indicator• function SLQSGetSysSelectionPref() returns a default value FFFF if no value is returned by the device.

<i>pLTEBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the LTE band preference • Bit Values <ul style="list-style-type: none"> – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – All other bits are reserved • function SLQSGetSysSelectionPref() returns a default value FFFFFFFFFFFFFFFF if no value is returned by the device.
---------------------	--

<i>pNetSelPref</i>	<ul style="list-style-type: none"> • Optional parameter indicating network selection preference • Values: <ul style="list-style-type: none"> – 0x00 - Automatic network selection – 0x01 - Manual network selection • function SLQSGetSysSelectionPref() returns a default value FF if no value is returned by the device.
<i>pSrvDomainPref</i>	<ul style="list-style-type: none"> • Optional parameter indicating Service domain preference • Values: <ul style="list-style-type: none"> – 0x00 - Circuit switched only – 0x01 - Packet switched only – 0x02 - Circuit switched and packet switched – 0x03 - Packet switched attach – 0x04 - Packet switched detach • function SLQSGetSysSelectionPref() returns a default value FFFFFFFF if no value is returned by the device.
<i>pGWAcqOrder</i> <i>Pref</i>	<ul style="list-style-type: none"> • Optional parameter indicating GSM/WCDMA Acquisition order Preference • Values: <ul style="list-style-type: none"> – 0x00 - Automatic – 0x01 - GSM then WCDMA – 0x02 - WCDMA then GSM • function SLQSGetSysSelectionPref() returns a default value FFFFFFFF if no value is returned by the device.

Note

None

9.18.3.4 typedef struct _sysSelectPrefParams sysSelectPrefParams

Contain the system selection preferences.

Parameters

<i>pEmerMode</i>	<ul style="list-style-type: none"> • Optional parameter specifying the emergency Mode • Values: <ul style="list-style-type: none"> – 0 - OFF (normal) – 1 - ON (Emergency)
------------------	---

<i>pModePref</i>	<ul style="list-style-type: none">• Optional parameter• Bit Mask indicating the radio technology mode preference• Bit values:<ul style="list-style-type: none">– Bit 0 - cdma2000 1x– Bit 1 - cdma2000 HRPD(1xEV-DO)– Bit 2 - GSM– Bit 3 - UMTS– Bit 4 - LTE
------------------	--

<i>pBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the band preference • Bit values: <ul style="list-style-type: none"> – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved
------------------	--

<i>pPRLPref</i>	<ul style="list-style-type: none">• Optional parameter indicating the CDMA PRL Preference• Values:<ul style="list-style-type: none">– 0x0001 - Acquire available system only on the A side– 0x0002 - Acquire available system only on the B side– 0x3FFF - Acquire any available systems
<i>pRoamPref</i>	<ul style="list-style-type: none">• Optional parameter indicating the roaming Preference• Values:<ul style="list-style-type: none">– 0x01 - Acquire only systems for which the roaming indicator is off– 0x02 - Acquire a system as long as its roaming indicator is not off– 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only– 0xFF - Acquire systems, regardless of their roaming indicator

<i>pLTEBandPref</i>	<ul style="list-style-type: none"> • Optional parameter • Bit mask representing the LTE band preference • Bit Values <ul style="list-style-type: none"> – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – All other bits are reserved
<i>pNetSelPref</i>	<ul style="list-style-type: none"> - netSelectionPref <ul style="list-style-type: none"> • Optional parameter for specifying Network Selection Preference • Modem selects networks based on this parameter(if present). • Either of pNetSelPref or pCSGID can be set. • see netSelectionPref for more information

<i>pChgDuration</i>	<ul style="list-style-type: none"> Optional parameter specifying the duration of the change At least one system selection setting to be set if pChgDuration is populated. Values: <ul style="list-style-type: none"> 0x00 - Power cycle - Remains active until the next device power cycle 0x01 - Permanent - Remains active through power cycles until changed by client Device will use "0x01 - permanent" as default if this parameter is omitted
<i>pMNCIncPCS↔ DigStat</i>	<ul style="list-style-type: none"> Optional parameter indicating if MNC includes PCS digit pNetSelPref is expected if MNC includes PCS digit is set to 1. Values: <ul style="list-style-type: none"> TRUE - MNC is a 3 digit value; e.g., a reported value of 90 corresponds to an MNC value of 090 FALSE - MNC is a 2-digit value; e.g., a reported value of 90 corresponds to an MNC value of 90
<i>pSrvDomainPref</i>	<ul style="list-style-type: none"> Optional parameter indicating Service domain preference Values: <ul style="list-style-type: none"> 0x00 - Circuit switched only 0x01 - Packet switched only 0x02 - Circuit switched and packet switched 0x03 - Packet switched attach 0x04 - Packet switched detach
<i>pGWAcqOrder↔ Pref</i>	<ul style="list-style-type: none"> Optional parameter indicating GSM/WCDMA Acquisition order Preference Values: <ul style="list-style-type: none"> 0x00 - Automatic 0x01 - GSM then WCDMA 0x02 - WCDMA then GSM
<i>pTdscdma↔ BandPref</i>	<ul style="list-style-type: none"> Optional parameter indicating bitmask representing the TD-SCDMA band preference to be set. Values: <ul style="list-style-type: none"> 0x01 - TD-SCDMA Band A 0x02 - TD-SCDMA Band B 0x04 - TD-SCDMA Band C 0x08 - TD-SCDMA Band D 0x10 - TD-SCDMA Band E 0x20 - TD-SCDMA Band F All other bits are reserved
<i>pAcqOrderPref</i>	<ul style="list-style-type: none"> - acqOrderPref Optional parameter for specifying Acquisition Order Preference see acqOrderPref for more information

<i>pSrvReg</i> <i>Restriction</i>	<ul style="list-style-type: none"> Optional parameter indicating Network Selection Registration Restriction Preference Values: <ul style="list-style-type: none"> 0x00 - Device follows the normal registration process 0x01 - Device camps on the network according to its provisioning, but does not register 0x02 - Device selects the network for limited service All other values are reserved.
<i>pCSGID</i>	<ul style="list-style-type: none"> - CSGID Optional parameter for specifying CSG ID Either of pNetSelPref or pCSGID can be set. see CSGID for more information
<i>pRAT</i>	<ul style="list-style-type: none"> Optional parameter Radio Access Technology order Preference Values: <ul style="list-style-type: none"> 0x04 - GSM 0x05 - UMTS 0x08 - LTE 0x09 - TDSCDMA

9.18.4 Enumeration Type Documentation

9.18.4.1 enum _NAMS_RADIO_IF_TECHNOLOGY_

Enumerator

eNAS_RADIO_IF_GSM

eNAS_RADIO_IF_UMTS

eNAS_RADIO_IF_LTE

eNAS_RADIO_IF_TDSCDMA

9.18.4.2 enum eSYS_SRV_DOMAIN

Enumerator

eSYS_SRV_DOMAIN_NO_SRV

eSYS_SRV_DOMAIN_CS_ONLY

eSYS_SRV_DOMAIN_PS_ONLY

eSYS_SRV_DOMAIN_CS_PS

eSYS_SRV_DOMAIN_CAMPED

eSYS_SRV_DOMAIN_UNKNOWN

9.18.4.3 enum NAS_LTE_CPHY_CA_BW_NRB

Enumerator

eNAS_LTE_CPHY_CA_BW_NRB_6
eNAS_LTE_CPHY_CA_BW_NRB_15
eNAS_LTE_CPHY_CA_BW_NRB_25
eNAS_LTE_CPHY_CA_BW_NRB_50
eNAS_LTE_CPHY_CA_BW_NRB_75
eNAS_LTE_CPHY_CA_BW_NRB_100

9.18.4.4 enum NAS_LTE_CPHY_SCELL_STATE

Enumerator

eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED

9.18.5 Function Documentation

9.18.5.1 ULONG GetACCOLC (BYTE * pACCOLC)

Retrieves information about the access overload class (ACCOLC)

Parameters

<i>pACCOLC</i> [OUT]	<ul style="list-style-type: none">• ACCOLC : Valid range is 0 to 15
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 2 seconds

9.18.5.2 ULONG GetANAAAAuthenticationStatus (ULONG * pStatus)

AN-AAA authentication status of the device.

Parameters

<i>pStatus[OUT]</i>	<ul style="list-style-type: none"> • Status of last AN-AAA authentication attempt <ul style="list-style-type: none"> – 0 - Failure – 1 - Success – 2 - Not Requested
---------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 2 seconds

9.18.5.3 `ULONG` GetCDMANetworkParameters (`BYTE * pSCI`, `BYTE * pSCM`, `BYTE * pRegHomeSID`, `BYTE * pRegForeignSID`, `BYTE * pRegForeignNID`, `BYTE * pForceRev0`, `BYTE * pCustomSCP`, `ULONG * pProtocol`, `ULONG * pBroadcast`, `ULONG * pApplication`, `ULONG * pRoaming`)

Gets the current CDMA network parameters

Parameters

<i>pSCI[OUT]</i>	<ul style="list-style-type: none"> • Slot cycle index <ul style="list-style-type: none"> – 0xFF-Unknown
<i>pSCM[OUT]</i>	<ul style="list-style-type: none"> • Station class mark <ul style="list-style-type: none"> – 0xFF-Unknown
<i>pRegHomeSID</i> ↔ <i>D[OUT]</i>	<ul style="list-style-type: none"> • Register on home SID <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled – 0xFF - Unknown

<i>pRegForeignSID</i> <i>ID[OUT]</i>	<ul style="list-style-type: none"> • Register on foreign SID <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled – 0xFF - Unknown
<i>pRegForeignNID</i> <i>ID[OUT]</i>	<ul style="list-style-type: none"> • Register on foreign NID <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled – 0xFF - Unknown
<i>pForceRev0</i> <i>OUT</i>	<ul style="list-style-type: none"> • Force CDMA 1x-EV-DO Rev. 0 mode <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled – 0xFF - Unknown
<i>pCustomSCP</i> <i>OUT</i>	<ul style="list-style-type: none"> • Use a custom config for CDMA 1x-EV-DO SCP <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled – 0xFF - Unknown
<i>pProtocol</i> <i>OUT</i>	<ul style="list-style-type: none"> • Protocol mask for custom SCP config <ul style="list-style-type: none"> – 0x00000001 - Subtype 2 Physical Layer – 0x00000002 - Enhanced CCMAC – 0x00000004 - Enhanced ACMAC – 0x00000008 - Enhanced FTCMAC – 0x00000010 - Subtype 3 RTCMAC – 0x00000020 - Subsystem 1 RTCMAC – 0x00000040 - Enhanced Idle – 0x00000080 - Generic Multimode Capable Disc Port – 0xFFFFFFFF - Unknown
<i>pBroadcast</i> <i>OUT</i>	<ul style="list-style-type: none"> • Broadcast mask for custom SCP config <ul style="list-style-type: none"> – 0x00000001 - Generic broadcast enabled – 0xFFFFFFFF - Unknown

<i>pApplication</i> [O↔ UT]	<ul style="list-style-type: none"> • Application mask for custom SCP config <ul style="list-style-type: none"> – 0x00000001 - SN Multiflow Packet Application – 0x00000002 - SN Enhanced Multiflow Packet Application – 0xFFFFFFFF - Unknown
<i>pRoaming</i> [OUT]	<ul style="list-style-type: none"> • Roaming preference <ul style="list-style-type: none"> – 0 - Automatic – 1 - Home Only – 2 - Affiliated Roaming Only – 3 - Home and Affiliated Roaming – 0xFFFFFFFF - Unknown

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 5 seconds

9.18.5.4 ULONG GetHomeNetwork (WORD * *pMCC*, WORD * *pMNC*, BYTE *nameSize*, CHAR * *pName*, WORD * *pSID*, WORD * *pNID*)

Retrieves information about the home network of the device. For 3GPP2 home network information use GetHome↔
Network3GPP2.

Parameters

<i>pMCC</i> [OUT]	<ul style="list-style-type: none"> • Mobile country code (UMTS only).
<i>pMNC</i> [OUT]	<ul style="list-style-type: none"> • Mobile network code (UMTS only).
<i>nameSize</i>	<ul style="list-style-type: none"> • Maximum number of characters (including NULL terminator) that 8 network name array can contain (UMTS only).

<i>pName[OUT]</i>	<ul style="list-style-type: none"> • Network name or description represented as a NULL terminated string (empty string returned when unknown) (UMTS only).
<i>pSID[OUT]</i>	<ul style="list-style-type: none"> • Home network system ID <ul style="list-style-type: none"> – 0xFFFF - Unknown. – Only applies to cdma2000
<i>pNID[OUT]</i>	<ul style="list-style-type: none"> • Home network ID <ul style="list-style-type: none"> – 0xFFFF - Unknown. – Only applies to cdma2000

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.5 **ULONG** GetHomeNetwork3GPP2 (**WORD** * *pMCC*, **WORD** * *pMNC*, **BYTE** *nameSize*, **CHAR** * *pName*, **WORD** * *pSID*, **WORD** * *pNID*, **WORD** * *pNw2MCC*, **WORD** * *pNw2MNC*, **BYTE** * *pNw2DescDisp*, **BYTE** * *pNw2DescEnc*, **BYTE** * *pNw2DescLen*, **BYTE** * *pNw2Name*)

Retrieves information about the home network of the device. It will extract 3GPP2 Network Information also.

Parameters

<i>pMCC[OUT]</i>	<ul style="list-style-type: none"> • Mobile country code (UMTS only).
<i>pMNC[OUT]</i>	<ul style="list-style-type: none"> • Mobile network code (UMTS only).
<i>nameSize</i>	<ul style="list-style-type: none"> • Maximum number of characters (including NULL terminator) that 8 network name array can contain (UMTS only).
<i>pName[OUT]</i>	<ul style="list-style-type: none"> • Network name or description represented as a NULL terminated string (empty string returned when unknown) (UMTS only).

<i>pSID[OUT]</i>	<ul style="list-style-type: none"> • Home network system ID <ul style="list-style-type: none"> – 0xFFFF - Unknown. – Only applies to cdma2000
<i>pNID[OUT]</i>	<ul style="list-style-type: none"> • Home network ID <ul style="list-style-type: none"> – 0xFFFF - Unknown. – Only applies to cdma2000
<i>pNw2MCC[O↔UT]</i>	<ul style="list-style-type: none"> • Mobile country code (3GPP2 only). • Range : 0 to 999
<i>pNw2MNC[O↔UT]</i>	<ul style="list-style-type: none"> • Mobile network code (3GPP2 only). • Range : 0 to 999
<i>pNw2Desc↔Disp[OUT]</i>	<ul style="list-style-type: none"> • Network Name Display (3GPP2 only). -Valid Value <ul style="list-style-type: none"> – 0x00 - Do not display – 0x01 - Display – 0xFF - Unknown
<i>pNw2Desc↔Disp[OUT]</i>	<ul style="list-style-type: none"> • Encoding of the network description (3GPP2 only). • Valid Value <ul style="list-style-type: none"> – 0x00 - Octet, unspecified – 0x02 - 7-bit ASCII – 0x04 - Unicode – 0x09 - GSM 7-bit default
<i>nw2DescLen[↔OUT]</i>	<ul style="list-style-type: none"> • Network Description Length (3GPP2 only).
<i>pNw2Name[O↔UT]</i>	<ul style="list-style-type: none"> • Network Name (3GPP2 only).

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.6 **ULONG** GetNetworkPreference (**ULONG** * *pTechnologyPref*, **ULONG** * *pDuration*, **ULONG** * *pPersistentTechnologyPref*)

Returns the network registration preference. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSGetSysSelectionPref\(\)](#) for new firmware versions and new modules

Parameters

<i>pTechnologyPref</i> [OUT]	<ul style="list-style-type: none"> • Bitmask representing the radio technology preference set. • No bits set indicates to the device to automatically determine the technology to use • Values: <ul style="list-style-type: none"> – Bit 0 - Technology is 3GPP2 – Bit 1 - Technology is 3GPP • Any combination of the following may be returned: <ul style="list-style-type: none"> – Bit 2 - Analog - AMPS if 3GPP2, GSM if 3GPP – Bit 3 - Digital - CDMA if 3GPP2, WCDMA if 3GPP – Bit 4 - HDR – Bit 5 - LTE – Bits 6 to 15 - Reserved
<i>pDuration</i> [OUT]	<ul style="list-style-type: none"> • Duration of active preference <ul style="list-style-type: none"> – 0 - Permanent – 1 - Power cycle – 2 - Until the end of the next call or a power cycle – 3 - Until the end of the next call, a specified time, or a power cycle – 4 to 6 - Until the end of the next call
<i>pPersistentTechnologyPref</i> [OUT]	<ul style="list-style-type: none"> • Bit field representing persistent radio technology preference <ul style="list-style-type: none"> – Same representation as the <i>pTechnologyPref</i> parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.7 **ULONG** GetRFInfo (**BYTE** * *pInstanceSize*, struct RFBandInfoElements * *pRFBandInfo*)

Sets the RFInfoList

Parameters

<i>pInstanceSize</i> [↔ IN/OUT]	<ul style="list-style-type: none"> • Upon input, maximum number of elements that the RF info instances array can contain. • Upon successful output, actual number of elements in RF info instances array.
<i>pInstances</i> [O↔ UT]	<ul style="list-style-type: none"> • RF info instances array <ul style="list-style-type: none"> – See RFBandInfoElements for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.8 **ULONG** [GetServingNetwork](#) (**ULONG** * *pRegistrationState*, **ULONG** * *pCSDomain*, **ULONG** * *pPSDomain*, **ULONG** * *pRAN*, **BYTE** * *pRadiolfacesSize*, **BYTE** * *pRadiolfaces*, **ULONG** * *pRoaming*, **WORD** * *pMCC*, **WORD** * *pMNC*, **BYTE** *nameSize*, **CHAR** * *pName*)

Provides information about the system that provides service to the device. This API is deprecated on MC73xx/↔ EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSNasGetSysInfo\(\)](#) for new firmware versions and new modules

Parameters

<i>pRegistration</i> ↔ State[OUT]	<ul style="list-style-type: none"> • Registration state: <ul style="list-style-type: none"> – 0 - Not registered – 1 - Registered – 2 - Searching/Not Registered – 3 - Registration Denied – 4 - Unknown
<i>pCSDomain</i> [O↔ UT]	<ul style="list-style-type: none"> • Circuit switch domain status: <ul style="list-style-type: none"> – 0 - Unknown/Not Applicable – 1 - Attached – 2 - Detached
<i>pPSDomain</i> [O↔ UT]	<ul style="list-style-type: none"> • Packet switch domain status <ul style="list-style-type: none"> – 0 - Unknown/Not Applicable – 1 - Attached
Generated by Doxygen	<ul style="list-style-type: none"> – 2 - Detached

<i>pRAN[OUT]</i>	<ul style="list-style-type: none"> Type of radio access network on which mobile is registered: <ul style="list-style-type: none"> 0 - Unknown 1 - cdma2000 network 2 - UMTS network
<i>pRadiofaces</i> _↔ <i>Size[IN/OUT]</i>	<ul style="list-style-type: none"> Upon input, maximum number of elements that the radio interface array contain. Upon successful output, actual number of elements in the radio interface array.
<i>pRadiofaces</i> _↔ <i>OUT</i>	<ul style="list-style-type: none"> An array of Radio Interface Technology <ul style="list-style-type: none"> See qaGobiApiTableRadioInterfaces.h for the Radio Interface Technologies
<i>pRoaming[OUT]</i>	<ul style="list-style-type: none"> Roaming indicator
<i>pMCC[OUT]</i>	<ul style="list-style-type: none"> Mobile country code
<i>pMNC[OUT]</i>	<ul style="list-style-type: none"> Mobile network code
<i>nameSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that network name array can contain; applicable only for UMTS networks
<i>pName[OUT]</i>	<ul style="list-style-type: none"> Network name or description represented as a NULL terminated string; empty string is returned when unknown; applicable only for UMTS networks

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.9 ULONG GetServingNetworkCapabilities (BYTE * *pDataCapsSize*, BYTE * *pDataCaps*)

Returns information regarding the data capabilities of the system that currently provides service to the device.

Parameters

<i>pDataCaps</i> ↔ <i>Size</i> [IN/OUT]	<ul style="list-style-type: none"> • Upon input, the maximum number of elements the data capabilities array can contain. • Upon output, the actual number of elements in the data capabilities array.
<i>pDataCaps</i> [O↔ UT]	<ul style="list-style-type: none"> • Data capabilities array of unsigned long type <ul style="list-style-type: none"> – 1 - GPRS – 2 - EDGE – 3 - HSDPA – 4 - HSUPA – 5 - WCDMA – 6 - CDMA 1xRTT – 7 - CDMA 1xEV-DO Rev 0 – 8 - CDMA 1xEV-DO Rev. A – 9 - GSM – 10 - EVDO Rev. B – 11 - LTE – 12 - HSDPA Plus – 13 - Dual Carrier HSDPA Plus

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.10 ULONG GetSignalStrengths (ULONG * pArraySizes, INT8 * pSignalStrength, ULONG * pRadioInterface)

Returns the available signal strengths (in dBm) as measured by the device in an array. The API also provides the corresponding radio radio interfaces in an array.

Parameters

<i>pArraySizes</i> [<i>I</i> ↔ <i>N/OUT</i>]	<ul style="list-style-type: none"> • Upon input maximum number of elements that each array can contain. • Upon successful output actual number of elements in the array.
<i>pSignal</i> ↔ <i>Strength</i> [<i>OUT</i>]	<ul style="list-style-type: none"> • Received signal strength array (in dBm)
<i>pRadio</i> ↔ <i>Interface</i> [<i>OUT</i>]	<ul style="list-style-type: none"> • Radio interface technology array of the signal being measured <ul style="list-style-type: none"> – See qaGobiApiTableRadioInterfaces.h for Radio Interface info

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.11 ULONG InitiateDomainAttach (ULONG action)

Initiates a domain attach/detach of the device. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSSetSysSelectionPref\(\)](#) for new firmware versions and new modules

Parameters

<i>action</i> [<i>IN</i>]	<ul style="list-style-type: none"> • Domain action to attempt <ul style="list-style-type: none"> 1 - Attach 2 - Detach
-----------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 seconds

9.18.5.12 `ULONG InitiateNetworkRegistration (ULONG regType, WORD mcc, WORD mnc, ULONG rat)`

Initiates the network registration process. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSSetSysSelectionPref\(\)](#) and [SLQSSetBandPreference\(\)](#) for new firmware versions and new modules

Parameters

<i>regType</i>	<ul style="list-style-type: none"> Registration type <ul style="list-style-type: none"> 1 - Automatic 2 - Manual
<i>mcc</i>	<ul style="list-style-type: none"> Mobile country code
<i>mnc</i>	<ul style="list-style-type: none"> Mobile network code
<i>rat</i>	<ul style="list-style-type: none"> Radio access technology <ul style="list-style-type: none"> 4 - GSM 5 - UMTS

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
Timeout: 30 seconds

9.18.5.13 `ULONG PerformNetworkScan (BYTE * pInstanceSize, BYTE * pInstances)`

Performs scan for available networks.

Parameters

<i>pInstanceSize</i> [↔ IN/OUT]	<ul style="list-style-type: none"> Upon input, maximum number of elements that the network info instance array can contain. Upon successful output, the actual number of elements in the network info instance array.
<i>pInstances</i> [↔ OUT] <small>Generated by Doxygen</small>	<ul style="list-style-type: none"> Network info instance array <ul style="list-style-type: none"> See QmiNas3GppNetworkInfo

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
Timeout: 5 minutes

9.18.5.14 ULONG SetACCOLC (CHAR * spc, BYTE accolc)

Sets the access overload class (ACCOLC)

Parameters

<i>spc[IN]</i>	<ul style="list-style-type: none">• service programming code NULL-terminated string of six digit
<i>ACCOLC[IN]</i>	<ul style="list-style-type: none">• ACCOLC : Valid range is 0 to 15

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 5 seconds

9.18.5.15 ULONG SetCDMANetworkParameters (CHAR * pSPC, BYTE * pForceRev0, BYTE * pCustomSCP, ULONG * pProtocol, ULONG * pBroadcast, ULONG * pApplication, ULONG * pRoaming)

Sets the CDMA network parameters. Currently the modified settings will not be utilized until the device has been reset. For this reason, the recommended approach when using SetCDMANetworkParameters is for the application to perform the following steps:

1 - Call [SetCDMANetworkParameters\(\)](#) 2 - Call SetPower(5) 3 - Call [QCWWANDisconnect\(\)](#) 4 - Reconnect after the device power cycles

Parameters

<i>pSPC</i> [IN]	<ul style="list-style-type: none"> • Six digit service programming code (not necessary when only the roaming field is being set)
<i>pForceRev0</i> [IN]	<ul style="list-style-type: none"> • (Optional)Force CDMA 1x-EV-DO Rev. 0 mode <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled Note: Enabled can only be specified if pCustomSCP state is set to Disabled
<i>pCustomSCP</i> [↔IN]	<ul style="list-style-type: none"> • (Optional)Use a custom config for CDMA 1x-EV-DO SCP <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled Note: Enabled can only be specified if pForceRev0 is set to Disabled
<i>pProtocol</i> [IN]	<ul style="list-style-type: none"> • Protocol mask for custom SCP config <ul style="list-style-type: none"> – 0x00000001 - Subtype 2 Physical Layer – 0x00000002 - Enhanced CCMAC – 0x00000004 - Enhanced ACMAC – 0x00000008 - Enhanced FTCMAC – 0x00000010 - Subtype 3 RTCMAC – 0x00000020 - Subsystem 1 RTCMAC – 0x00000040 - Enhanced Idle – 0x00000080 - Generic Multimode Capable Disc Port – 0xFFFFFFFF - Unknown
<i>pBroadcast</i> [IN]	<ul style="list-style-type: none"> • Broadcast mask for custom SCP config <ul style="list-style-type: none"> – 0x00000001 - Generic broadcast enabled – 0xFFFFFFFF - Unknown
<i>pApplication</i> [IN]	<ul style="list-style-type: none"> • Application mask for custom SCP config <ul style="list-style-type: none"> – 0x00000001 - SN Multiflow Packet Application – 0x00000002 - SN Enhanced Multiflow Packet Application – 0xFFFFFFFF - Unknown
<i>pRoaming</i> [IN]	<ul style="list-style-type: none"> • Roaming preference <ul style="list-style-type: none"> – 0 - Automatic – 1 - Home Only – 2 - Affiliated Roaming Only – 3 - Home and Affiliated Roaming – 0xFFFFFFFF - Unknown

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 5 seconds

9.18.5.16 ULONG SetNetworkPreference (ULONG *technologyPref*, ULONG *duration*)

Sets the network registration preference. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSSetSysSelectionPref\(\)](#) for new firmware versions and new modules

Parameters

<i>technologyPref</i> [IN]	<ul style="list-style-type: none"> 2 Byte Bitmask representing radio technology preference <ul style="list-style-type: none"> No bits set indicates device to automatically determine the technology to use. Type of technology <ul style="list-style-type: none"> Bit 0 - Technology is 3GPP2 Bit 1 - Technology is 3GPP Technology-specific protocol bitmask <ul style="list-style-type: none"> Bit 2 - Analog <ul style="list-style-type: none"> AMPS if 3GPP2, GSM if 3GPP Bit 3 - Digital <ul style="list-style-type: none"> CDMA if 3GPP2, WCDMA if 3GPP Bit 4 - HDR Bit 5 - LTE All other bits are reserved.
<i>duration</i> [IN]	<ul style="list-style-type: none"> Duration of active preference <ul style="list-style-type: none"> 0 - Persistent 1 - Power cycle

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.17 ULONG SLQSConfigSigInfo (sigInfo * pSigInfo)

Sets the signal reporting thresholds. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use [SLQSNasConfigSigInfo2\(\)](#) instead for new firmware versions and new modules.

Parameters

<i>pSigInfo[IN]</i>	<ul style="list-style-type: none">• See sigInfo for more information
---------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.18.5.18 ULONG SLQSGetErrorRate (GetErrRateResp * pGetErrRateResp)

This API retrieves current error rate information

Parameters

<i>pGetErrRate↔ Resp[OUT]</i>	<ul style="list-style-type: none">• See GetErrRateResp for more information
-----------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.18.5.19 ULONG SLQSGetNetworkTime (GetNetworkTimeResp * pGetNetworkTimeResp)

This API retrieves the last known network time information from the UE.

Parameters

<i>pGetNetworkTimeResp</i> [OUT]	<ul style="list-style-type: none">See GetNetworkTimeResp for more information
----------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.18.5.20 ULONG SLQSGetOperatorNameData (nasOperatorNameResp * pOperatorNameData)

Get the operator name data from the network. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSGetPLMNName\(\)](#) for new firmware versions and new modules

Parameters

<i>pOperatorNameData</i> [OUT]	<ul style="list-style-type: none">See nasOperatorNameResp for more information
--------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 10 seconds

9.18.5.21 ULONG SLQSGetPLMNName (nasPLMNNameReq * pPLMNNameReq, nasPLMNNameResp * pPLMNNameResp)

Get the operator name data from the network.

Parameters

<i>pPLMNNameReq</i> [IN]	<ul style="list-style-type: none"> See nasPLMNNameReq for more information
<i>pPLMNNameResp</i> [OUT]	<ul style="list-style-type: none"> See nasPLMNNameResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 10 seconds

9.18.5.22 ULONG SLQSGetServingSystem (qaQmiServingSystemParam * pServingSystem)

Provides information about the system that provides service to the device. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSNasGetSysInfo\(\)](#) for new firmware versions and new modules. Also report available radio interface technology. If there are more than one radio interface, please choose the right interface(usually the first pair).

- See [SLQSSetSysSelectionPref](#)

Parameters

<i>pServingSystem</i> [OUT]	<ul style="list-style-type: none"> serving system parameters obtained from the system
-----------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise.

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values.

Note

Timeout: 2 seconds

9.18.5.23 ULONG SLQSGetSignalStrength (struct slqsSignalStrengthInfo * pSignalInfo)

Queries the current signal strength as measured by the device. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSNasGetSigInfo\(\)](#) for new firmware versions and new modules

Parameters

<i>pSignalInfo</i> [IN↔OUT]	<ul style="list-style-type: none">• See slqsSignalStrengthInfo for more information
-----------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.24 ULONG SLQSGetSysSelectionPref (sysSelectPrefInfo * pSysSelectPrefInfo)

Queries the different system selection preferences of the device.

Parameters

<i>pSysSelectPrefInfo</i> [OUT]	<ul style="list-style-type: none">• See sysSelectPrefInfo for more information
---------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.25 ULONG SLQSIInitiateNetworkRegistration (nasInitNetworkReq * pNasInitNetRegistrationReg)

Initiates the network registration process.

Parameters

<i>pNasInitNetRegistrationReq</i> [IN]	<ul style="list-style-type: none"> • Pointer to structure nasInitNetworkReq <ul style="list-style-type: none"> – See nasInitNetworkReq for more information
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
Timeout: 30 seconds

9.18.5.26 ULONG SLQSNasConfigSigInfo2 (setSignalStrengthInfo * pSetSignalStrengthInfo)

Sets the signal strength reporting thresholds

Parameters

<i>pSetSignalStrengthInfo</i> [IN]	<ul style="list-style-type: none"> • See setSignalStrengthInfo for more information
------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

Mixture of threshold and delta values can be provided in the request. But for each type and RAT, only one of threshold list or delta value is to be provided.

9.18.5.27 **ULONG** SLQSNasGet3GPP2Subscription (**nasGet3GPP2SubscriptionInfoReq** * *pGet3GPP2SubsInfoReq*, **nasGet3GPP2SubscriptionInfoResp** * *pGet3GPP2SubsInfoResp*)

This API retrieves 3GPP2 subscription-related information.

Parameters

<i>pGet3GPP2SubsInfoReq</i> [IN]	<ul style="list-style-type: none"> See nasGet3GPP2SubscriptionInfoReq for more information
<i>pGet3GPP2SubsInfoResp</i> [OUT]	<ul style="list-style-type: none"> See nasGet3GPP2SubscriptionInfoResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA

Timeout: 2 seconds

This command retrieves 3GPP2 subscription-related information. The QMI_ERR_INTERNAL error is returned when no information can be retrieved from the modem.

9.18.5.28 **ULONG** SLQSNasGetCellLocationInfo (**nasCellLocationInfoResp** * *pNasCellLocationInfoResp*)

This API retrieves cell location-related information

Parameters

<i>pNasCellLocationInfoResp</i> [OUT]	<ul style="list-style-type: none"> See nasCellLocationInfoResp for more information
---------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

This API retrieves cell location-related information, depending on current serving system.

9.18.5.29 ULONG SLQSNasGetHDRColorCode (nasGetHDRColorCodeResp * pGetHDRColorCodeResp)

This API retrieves the current HDR color code value.

Parameters

<i>pGetHDRColorCodeResp[OUT]</i>	<ul style="list-style-type: none">• See nasGetHDRColorCodeResp for more information
----------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.30 ULONG SLQSNASGetLTECPHYCaInfo (nasGetLTECphyCa * pLTECPhyCa)

This API Get LTE CPHY Carrier Info

Parameters

<i>pLTECPhyCa[IN]</i>	<ul style="list-style-type: none">• See nasGetLTECphyCa for more information.
-----------------------	---

Returns

eQCWWAN_ERR_sNONE on success, eQCWWAN_xxx error value otherwise.

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values.

9.18.5.31 **ULONG** SLQSNasGetSigInfo (**nasGetSigInfoResp** * *pGetSigInfoResp*)

This API queries information regarding the signal strength.

Parameters

<i>pGetSigInfoResp</i> <i>Resp[OUT]</i>	<ul style="list-style-type: none"> See nasGetSigInfoResp for more information
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

This command queries the signal strength information for currently active RATs. Information is reported only if the corresponding RATs have signal strength values to be reported. If no signal strength information is available for any RAT, the response message contains only the mandatory response message

9.18.5.32 **ULONG** SLQSNasGetSysInfo (**nasGetSysInfoResp** * *pGetSysInfoResp*)

Provides the system information. This API is preferred when trying to get the service status info and serving system info. The API [SLQSGetServingSystem\(\)](#) reports similar NAS information, but it is deprecated. Please refer to the header description of API [SLQSGetServingSystem\(\)](#) for more information.

Parameters

<i>pGetSysInfoResp</i> <i>Resp[OUT]</i>	<ul style="list-style-type: none"> See nasGetSysInfoResp for more information
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

This API queries current serving system information, including registration information and system property. The registration information for all RATs specified in the mode capability setting are included regardless of registration status. The RAT-specific system property are included only for RATs that are specified in the mode capability setting and which are not in either No Service or Power Save modes.

9.18.5.33 **ULONG** SLQSNasGetTxRxInfo (**nasGetTxRxInfoReq** * *pGetTxRxInfoReq*, **nasGetTxRxInfoResp** * *pGetTxRxInfoResp*)

This API retrieves the detailed Tx/Rx information.

Parameters

<i>pGetTxRxInfoReq</i> [IN]	<ul style="list-style-type: none"> See nasGetTxRxInfoReq for more information
<i>pGetTxRxInfoResp</i> [OUT]	<ul style="list-style-type: none"> See nasGetTxRxInfoResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

This command retrieves Tx/Rx information for a radio interface. The Rx chain are included in the response message only if they are enabled.

9.18.5.34 **ULONG** SLQSNasIndicationRegister (**BYTE** *systemSelectionInd*, **BYTE** *DDTMInd*, **BYTE** *servingSystemInd*)

Register/De-register from NAS (Network access service) broadcast indications. Some callbacks would not be invoked if the indications are not registered. The details are provided in the parameter description.

Parameters

<i>systemSelectionInd</i> [IN]	<ul style="list-style-type: none"> system selection preference indication registration. The following callbacks would not be invoked if the indication is disabled. <ul style="list-style-type: none"> tFNRoamingIndicator tFNDataCapabilities tFNServingSystem 0x00 - for disable 0x01 - for enable 0xFF - No change - Specifying this parameter indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device
<i>DDTMInd</i> [IN]	<ul style="list-style-type: none"> DDTM (Data Dedicated Transmission Mode) indication registration. <ul style="list-style-type: none"> 0x00 - for disable 0x01 - for enable 0xFF - No change - Specifying this parameter indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device

<i>servingSystem</i> ↔ <i>Ind[IN]</i>	<ul style="list-style-type: none"> • Serving system indication registration. The following callbacks would not be invoked if the indication is disabled. tFNBandPreference <ul style="list-style-type: none"> – 0x00 - for disable – 0x01 - for enable – 0xFF - No change - Specifying this parameter indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device
--	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise.

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values.

Note

Timeout: 2 seconds

9.18.5.35 ULONG SLQSNasIndicationRegisterExt (nasIndicationRegisterReq * pIndicationRegisterReq)

This API Registers/De-registers for different NAS (Network access service) indications.

Parameters

<i>pIndication</i> ↔ <i>RegisterReq[IN]</i>	<ul style="list-style-type: none"> • See nasIndicationRegisterReq for more information
--	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise.

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values.

Note

Timeout: 2 seconds

This API is used by a control point to register/deregister for different QMI_NAS indications. The control point's registration state variables, controlling registration for indications, are modified to reflect the settings indicated in the parameters that are present in the request message. At least one optional parameter must be present in the request.

9.18.5.36 ULONG SLQSNasIndicationRegisterLTECphyCa (BYTE * *bStatus*)

This API Registers/De-registers for NAS CPHY Carrier Info.

Parameters

<i>bStatus</i> [IN]	<ul style="list-style-type: none">• Values<ul style="list-style-type: none">– 0 - De-register.– 1 - Register.
---------------------	--

Returns

eQCWWAN_ERR_SNONE on success, eQCWWAN_xxx error value otherwise.

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values.

9.18.5.37 ULONG SLQSNASSwiGetChannelLock (nasSwiGetChannelLockResp * *pNasSwiGetChannelLockResp*)

This API queries the channel or cell which the UE is locked into.

Parameters

<i>pNasSwiGetChannelLock</i> [↔ OUT]	<ul style="list-style-type: none">• See nasSwiGetChannelLockResp for more information.
---	--

Returns

eQCWWAN_ERR_SNONE on success, eQCWWAN_xxx error value otherwise.

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values.

9.18.5.38 ULONG SLQSNasSwiIndicationRegister (NasSwiIndReg * *pIndRegReq*)

sets the registration state for different QMI_NAS SWI indications

Parameters

<i>pIndRegReq</i> [IN]	<ul style="list-style-type: none">• See NasSwiIndReg for more information
------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.39 ULONG SLQSNasSwiModemStatus (swiModemStatusResp * pModemStatusResp)

This API requests the device to return the current status of modem.

Parameters

<i>pModemStatusResp</i> [Out]	<ul style="list-style-type: none">• See swiModemStatusResp for more information
-------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.40 ULONG SLQSNASSwiSetChannelLock (nasSwiSetChannelLockReq * pNasSwiSetChannelLockReq)

This API allows the host to lock the UE to a specific channel or cell.

Parameters

<i>pNasSwiSetChannelLockReq</i> [In]	<ul style="list-style-type: none">• See nasSwiSetChannelLockReq for more information.
--------------------------------------	---

Returns

eQCWWAN_ERR_sNONE on success, eQCWWAN_xxx error value otherwise.

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values.

Note

The settings are persistent across reboots.

9.18.5.41 ULONG SLQSPerformNetworkScan (slqsNetworkScanInfo * pNetworkInfo)

Performs scan for available networks and scans for RAT info as well.

Parameters

<i>pNetworkInfo</i> [\leftrightarrow N/OUT]	<ul style="list-style-type: none">• See slqsNetworkScanInfo for more information• Valid pointers to the following structure members are mandatory<ul style="list-style-type: none">– pNetworkInfoInstances– pNetworkInfo
---	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

Note

Technology Supported: UMTS
Timeout: 5 minutes

9.18.5.42 ULONG SLQSSetBandPreference (ULONGLONG bandpreference)

Provides information about the band preference.

Parameters

<p><i>bandpreference</i> [↔] IN]</p>	<ul style="list-style-type: none"> • Bit mask representing the band preference to be set. • Bit position meanings: <ul style="list-style-type: none"> – 0 - BC0_A - Band Class 0, A-System – 1 - BC0_B - Band Class 0, B-System, Band Class 0 AB , GSM 850 Band – 2 - BC1 - Band Class 1, all blocks – 3 - BC2 - Band Class 2 place holder – 4 - BC3 - Band Class 3, A-System – 5 - BC4 - Band Class 4, all blocks – 6 - BC5 - Band Class 5, all blocks – 7 - GSM_DCS_1800 - GSM DCS band – 8 - GSM_EGSM_900 - GSM Extended GSM (E-GSM) band – 9 - GSM_PGSM_900 - GSM Primary GSM (P-GSM) band – 10 - BC6 - Band Class 6 – 11 - BC7 - Band Class 7 – 12 - BC8 - Band Class 8 – 13 - BC9 - Band Class 9 – 14 - BC10 - Band Class 10 – 15 - BC11 - Band Class 11 – 16 - GSM_450 - GSM 450 band – 17 - GSM_480 - GSM 480 band – 18 - GSM_750 - GSM 750 band – 19 - GSM_850 - GSM 850 band – 20 - GSM_RGSM_900 - GSM Railways GSM Band – 21 - GSM_PCS_1900 - GSM PCS band – 22 - WCDMA_I_IMT_2000 - WCDMA EUROPE JAPAN and CHINA IMT 2100 band – 23 - WCDMA_II_PCS_1900 - WCDMA US PCS 1900 band – 24 - WCDMA_III_1700 - WCDMA EUROPE and CHINA DCS 1800 band – 25 - WCDMA_IV_1700 - WCDMA US 1700 band – 26 - WCDMA_V_850 - WCDMA US 850 band – 27 - WCDMA_VI_800 - WCDMA JAPAN 800 band – 28 - BC12 - Band Class 12 – 29 - BC14 - Band Class 14 – 30 - RESERVED_2 - Reserved 2 – 31 - BC15 - Band Class 15 – 32 - 47 - Reserved – 48 - WCDMA_VII_2600 - WCDMA EUROPE 2600 band – 49 - WCDMA_VIII_900 - WCDMA EUROPE and JAPAN 900 band – 50 - WCDMA_IX_1700 - WCDMA JAPAN 1700 band – 51 to 55 - Reserved – 56 - BBC16 - Band Class 16 – 57 - BC17 - Band Class 17 – 58 - BC18 - Band Class 18 – 59 - BC19 - Band Class 19 – 60 to 64 - Reserved
--	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise.

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values.

Note

Timeout: 2 seconds

9.18.5.43 ULONG SLQSSetSysSelectionPref (sysSelectPrefParams * pSysSelectPrefParams)

Sets the different system selection preferences of the device.

Parameters

<i>pSysSelect</i> ↔ <i>PrefParams[IN]</i>	<ul style="list-style-type: none">• See sysSelectPrefParams for more information
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.18.5.44 ULONG SLQSSwiGetHDRPersonality (HDRPersonalityResp * pHDRPersonalityResp)

This API retrieves HDR Personality related information

Parameters

<i>pHDR</i> ↔ <i>Personality</i> ↔ <i>Resp[OUT]</i>	<ul style="list-style-type: none">• See HDRPersonalityResp for more information
---	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 5 seconds

9.18.5.45 ULONG SLQSSwiGetHDRProtSubtype (HDRProtSubtypResp * pHDRProtSubtypResp)

This API retrieves HDR Prototype Subtype related information

Parameters

<i>pHDRProtSubtypResp</i> [OUT]	<ul style="list-style-type: none">• See HDRProtSubtypResp for more information
------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 5 seconds

9.18.5.46 ULONG SLQSSwiGetHRPDStats (GetHRPDStatsResp * pGetHRPDStatsResp)

This API retrieves currently acquired HRPD system statistics

Parameters

<i>pGetHRPDStatsResp</i> [OUT]	<ul style="list-style-type: none">• See GetHRPDStatsResp for more information
-----------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.18.5.47 ULONG SLQSSwiGetLteCQI (LteCQIParm * pLteCQIResp)

This API Fetch CQI parameters for LTE data session

Parameters

<i>pLteCQIParm</i> <i>[OUT]</i>	<ul style="list-style-type: none">• See LteCQIParm for more information
------------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.18.5.48 ULONG SLQSSwiGetLteScsRxInfo (LteScsRxInfoResp * pLteScsRxInfoResp)

This API retrieves the LTE Secondary carrier Rx signal level information.

Parameters

<i>pGetLteScsRxInfoResp</i> <i>[OUT]</i>	<ul style="list-style-type: none">• See GetLteScsRxInforesp for more information
---	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.18.5.49 ULONG SLQSSwiNetworkDebug (NetworkDebugResp * pNetworkDebugResp)

This API retrieves device and network status details

Parameters

<i>pNetworkDebugResp</i> [Out] UT]	<ul style="list-style-type: none">• See NetworkDebugResp for more information
---------------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.18.5.50 ULONG SLQSSwiPSDetach (PSDetachReq * pPSDetachReq)

This API detaches PS connection.

Parameters

<i>pPSDetachReq</i> [In]	<ul style="list-style-type: none">• See PSDetachReq for more information
--------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 5 seconds

9.19 qaGobiApiOadm.h File Reference

Open Mobile Alliance Device Management Service API function prototypes.

Functions

- [ULONG OMADMStartSession](#) ([ULONG](#) sessionType)
- [ULONG OMADMCancelSession](#) ()
- [ULONG OMADMGetSessionInfo](#) ([ULONG](#) *pSessionState, [ULONG](#) *pSessionType, [ULONG](#) *pFailureReason, [BYTE](#) *pRetryCount, [WORD](#) *pSessionPause, [WORD](#) *pTimeRemaining)
- [ULONG OMADMGetPendingNIA](#) ([ULONG](#) *pSessionType, [USHORT](#) *pSessionID)

9.19.1 Detailed Description

Open Mobile Alliance Device Management Service API function prototypes.

9.19.2 Function Documentation

9.19.2.1 [ULONG OMADMCancelSession](#) ()

Cancels an ongoing OMA-DM session.

Parameters

<i>None</i>	
-------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA

Timeout: 2 seconds

9.19.2.2 ULONG OMADMGetPendingNIA (ULONG * pSessionType, USHORT * pSessionID)

Returns information about the pending network-initiated alert

Parameters

<i>SessionType</i> [↔ <i>OUT</i>]	<ul style="list-style-type: none">• Session Type<ul style="list-style-type: none">– 0x04 - Network-initiated PRL update– 0x05 - Network-initiated device configure
<i>SessionID</i> [<i>OUT</i>]	<ul style="list-style-type: none">• Session Id<ul style="list-style-type: none">– Unique session ID for NIA request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA

Timeout: 2 seconds

9.19.2.3 ULONG OMADMGetSessionInfo (ULONG * pSessionState, ULONG * pSessionType, ULONG * pFailureReason, BYTE * pRetryCount, WORD * pSessionPause, WORD * pTimeRemaining)

Returns information related to the current (or previous if no session is active) OMA-DM session.

Parameters

<i>SessionState</i> [OUT]	<ul style="list-style-type: none"> • Session state <ul style="list-style-type: none"> – 0x00 - Complete, information was updated – 0x01 - Complete, update information is unavailable – 0x02 - Failed – 0x03 - Retrying – 0x04 - Connecting – 0x05 - Connected – 0x06 - Authenticated – 0x07 - Mobile Directory Number (MDN) downloaded – 0x08 - Mobile Station Identifier (MSID) downloaded – 0x09 - PRL downloaded – 0x0A - Mobile IP (MIP) profile downloaded
<i>sessionType</i> [OUT]	<ul style="list-style-type: none"> • Session State <ul style="list-style-type: none"> – 0x00 - Client-initiated device configure – 0x01 - Client-initiated PRL update – 0x02 - Client-initiated hands-free activation – 0x03 - Device-initiated hands-free activation – 0x04 - Network-initiated PRL update – 0x05 - Network-initiated device configure
<i>FailureReason</i> [OUT]	<ul style="list-style-type: none"> • Session failure reason (when state indicates failure) <ul style="list-style-type: none"> – 0x00 - Unknown – 0x01 - Network is unavailable – 0x02 - Server is unavailable – 0x03 - Authentication failed – 0x04 - Maximum retry exceeded – 0x05 - Session is cancelled
<i>RetryCount</i> [OUT]	<ul style="list-style-type: none"> • Session retry count (when state indicates retrying)
<i>SessionPause</i> [OUT]	<ul style="list-style-type: none"> • Session pause timer (in seconds , when state indicates retrying)
<i>TimeRemaining</i> [OUT]	<ul style="list-style-type: none"> • Pause time remaining (in seconds , when state indicates retrying)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 2 seconds

9.19.2.4 ULONG OMADMStartSession (ULONG *sessionType*)

Starts an OMA-DM session.

Parameters

<i>sessionType</i>	<ul style="list-style-type: none"> • Session type <ul style="list-style-type: none"> – 0x00 - Client-initiated device configure – 0x01 - Client-initiated PRL update – 0x02 - Client-initiated hands-free activation
--------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 2 seconds

9.20 qaGobiApiPds.h File Reference

Position Determination Service API function prototypes.

Data Structures

- struct [PDSPositionData](#)
- struct [GPSSStateInfo](#)
- struct [PDSPosMethodStateReq](#)

Macros

- #define [DEFAULTBYTEVALUE](#) 0xFF
- #define [DEFAULTWORDVALUE](#) 0xFFFF
- #define [DEFAULTLONGVALUE](#) 0xFFFFFFFF

Enumerations

- enum {
[eSetServiceAutomaticTrackingDisable](#) =0,
[eSetServiceAutomaticTrackingEnable](#) =1 }

Functions

- [ULONG GetPDSSState](#) ([ULONG](#) *pEnabledStatus, [ULONG](#) *pTrackingStatus)
- [ULONG SetPDSSState](#) ([ULONG](#) enable)
- [ULONG StartPDSTrackingSessionExt](#) ([BYTE](#) sessionControl, [BYTE](#) sessionType, [BYTE](#) sessionOperation, [BYTE](#) sessionServerOption, [BYTE](#) fixTimeout, [ULONG](#) fixInterval, [ULONG](#) fixCount, [ULONG](#) fixAccuracy)
- [ULONG StopPDSTrackingSession](#) ()
- [ULONG PDSInjectTimeReference](#) ([ULONGLONG](#) systemTime, [USHORT](#) systemDiscontinuities)
- [ULONG GetPDSDefaults](#) ([ULONG](#) *pOperation, [BYTE](#) *pTimeout, [ULONG](#) *pInterval, [ULONG](#) *pAccuracy)
- [ULONG SetPDSDefaults](#) ([ULONG](#) operation, [BYTE](#) timeout, [ULONG](#) interval, [ULONG](#) accuracy)
- [ULONG GetXTRAAutomaticDownload](#) ([ULONG](#) *pbEnabled, [USHORT](#) *pInterval)
- [ULONG SetXTRAAutomaticDownload](#) ([ULONG](#) bEnabled, [USHORT](#) interval)
- [ULONG GetXTRANetwork](#) ([ULONG](#) *pPreference)
- [ULONG SetXTRANetwork](#) ([ULONG](#) preference)
- [ULONG GetXTRAVailidity](#) ([USHORT](#) *pGPSWeek, [USHORT](#) *pGPSWeekOffset, [USHORT](#) *pDuration)
- [ULONG ForceXTRADownload](#) ()
- [ULONG GetServiceAutomaticTracking](#) ([ULONG](#) *pbAuto)
- [ULONG SetServiceAutomaticTracking](#) ([ULONG](#) bAuto)
- [ULONG GetPortAutomaticTracking](#) ([ULONG](#) *pbAuto)
- [ULONG SetPortAutomaticTracking](#) ([ULONG](#) bAuto)
- [ULONG ResetPDSDData](#) ([ULONG](#) *pGPSDataMask, [ULONG](#) *pCellDataMask)
- [ULONG SLQSSetAGPSConfig](#) ([ULONG](#) *pServerAddress, [ULONG](#) *pServerPort, [BYTE](#) *pServerURL, [BYTE](#) *pServerURLLength, [BYTE](#) *pNetworkMode)
- [ULONG SLQSPDSInjectAbsoluteTimeReference](#) ([ULONGLONG](#) timeMsec, [ULONG](#) timeUncMsec, [BYTE](#) timeBase, [BYTE](#) forceFlag)
- [ULONG SLQSGetAGPSConfig](#) ([ULONG](#) *pServerAddress, [ULONG](#) *pServerPort, [BYTE](#) *pServerURL, [BYTE](#) *pServerURLLength, [BYTE](#) *pNetworkMode)
- [ULONG SLQSPDSInjectPositionData](#) (struct [PDSPositionData](#) *pPositionData)
- [ULONG SLQSPDSDeterminePosition](#) ()
- [ULONG SLQSGetGPSSStateInfo](#) ([GPSSStateInfo](#) *pGPSSStateInfo)
- [ULONG SLQSSetPositionMethodState](#) ([PDSPosMethodStateReq](#) *pPDSPosMethodStateReq)

9.20.1 Detailed Description

Position Determination Service API function prototypes.

9.20.2 Macro Definition Documentation

9.20.2.1 `#define DEFAULTBYTEVALUE 0xFF`

9.20.2.2 `#define DEFAULTLONGVALUE 0xFFFFFFFF`

9.20.2.3 `#define DEFAULTWORDVALUE 0xFFFF`

9.20.3 Enumeration Type Documentation

9.20.3.1 anonymous enum

Enumerator

eSetServiceAutomaticTrackingDisable
eSetServiceAutomaticTrackingEnable

9.20.4 Function Documentation

9.20.4.1 ULONG ForceXTRADownload ()

Forces the XTRA database to be downloaded to the device.

Parameters

<i>none</i>	
-------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.2 ULONG GetPDSDefaults (ULONG * pOperation, BYTE * pTimeout, ULONG * pInterval, ULONG * pAccuracy)

Returns the default tracking session configuration. The tracking session configuration is used when a tracking session is automatically started using SetServiceAutomaticTracking or due to the device detecting an application opening the NMEA port.

Parameters

<i>pOperation</i> [O↔ UT]	<ul style="list-style-type: none"> Current session operating mode <ul style="list-style-type: none"> 0 - Standalone 1 - MS based 2 - MS assisted
<i>pTimeout</i> [OUT]	<ul style="list-style-type: none"> Maximum amount of time (seconds) to work on each fix, maximum is 255
<i>pInterval</i> [OUT]	<ul style="list-style-type: none"> Interval (seconds) between fix requests
<i>pAccuracy</i> [OUT]	<ul style="list-style-type: none"> Preferred accuracy threshold (meters)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.3 ULONG GetPDSState (ULONG * pEnabledStatus, ULONG * pTrackingStatus)

Returns the current PDS state.

Parameters

<i>pEnabledStatus</i> [OUT]	<ul style="list-style-type: none">• Current PDS state<ul style="list-style-type: none">– 0 - disable– 1 - enable
<i>pTrackingStatus</i> [OUT]	<ul style="list-style-type: none">• Current PDS tracking session state• Values:<ul style="list-style-type: none">– 0x00 - Unknown– 0x01 - Inactive– 0x02 - Active

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.20.4.4 ULONG GetPortAutomaticTracking (ULONG * pbAuto)

Returns the automatic tracking configuration for the NMEA COM port.

Parameters

<i>pbAuto[OUT]</i>	<ul style="list-style-type: none">• Automatic tracking enabled for NMEA COM port<ul style="list-style-type: none">– 0x00 - Disabled– 0x01 - Enabled
--------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.5 ULONG GetServiceAutomaticTracking (ULONG * pbAuto)

Returns the automatic tracking state for the service.

Parameters

<i>pbAuto[OUT]</i>	<ul style="list-style-type: none">• Automatic tracking session started for service<ul style="list-style-type: none">– 0x00 - Disabled– 0x01 - Enabled
--------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.6 ULONG GetXTRAAutomaticDownload (ULONG * pbEnabled, USHORT * pInterval)

Returns the XTRA automatic database download configuration.

Parameters

<i>pbEnabled</i> [OUT]	<ul style="list-style-type: none"> Automatic XTRA download status <ul style="list-style-type: none"> 0 - Disabled 1 - Enabled
<i>pInterval</i> [OUT]	<ul style="list-style-type: none"> Interval (hours) between XTRA downloads

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.7 ULONG GetXTRANetwork (ULONG * pPreference)

Returns the XTRA WWAN network preference. When automatic XTRA database downloading is enabled this preference determines which WWAN networks will be used to perform the XTRA database download.

Parameters

<i>pPreference</i> [OUT]	<ul style="list-style-type: none"> XTRA WWAN network preference <ul style="list-style-type: none"> 0x00 - None (any available network) 0x01 - Home-only, only when on home systems 0x02 - Roam-only, only when on non-home systems
--------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.8 **ULONG** GetXTRAVValidity (**USHORT** * *pGPSWeek*, **USHORT** * *pGPSWeekOffset*, **USHORT** * *pDuration*)

Returns the XTRA database validity period. When automatic XTRA database downloading is enabled the validity period determines when the XTRA database will be updated through a new download.

Parameters

<i>pGPSWeek</i> [OUT]	<ul style="list-style-type: none"> Starting GPS week of validity period
<i>pGPSWeekOffset</i> [OUT]	<ul style="list-style-type: none"> Starting GPS week offset (minutes) of validity period
<i>pDuration</i> [OUT]	<ul style="list-style-type: none"> Length of validity period (hours)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.9 **ULONG** PDSInjectTimeReference (**ULONGLONG** *systemTime*, **USHORT** *systemDiscontinuities*)

Injects a system time into the PDS engine.

Parameters

<i>systemTime</i>	<ul style="list-style-type: none"> System time(milliseconds)
<i>systemDiscontinuities</i>	<ul style="list-style-type: none"> Number of system time discontinuities

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.10 ULONG ResetPDSData (ULONG * pGPSDataMask, ULONG * pCellDataMask)

Resets the specified PDS data.

Parameters

<p><i>pGPSDataMask</i> [IN]</p>	<ul style="list-style-type: none"> • Bitmask of GPS data to clear (optional) <ul style="list-style-type: none"> – 0x00000001 - EPH – 0x00000002 - ALM – 0x00000004 - POS – 0x00000008 - TIME – 0x00000010 - IONO – 0x00000020 - UTC – 0x00000040 - HEALTH – 0x00000080 - SVDIR – 0x00000100 - SVSTEER – 0x00000200 - SADATA – 0x00000400 - RTI – 0x00000800 - ALM_CORR – 0x00001000 - FREQ_BIAS_EST
<p><i>pCellDataMask</i> [IN]</p>	<ul style="list-style-type: none"> • Bitmask of cell data to clear (optional) <ul style="list-style-type: none"> – 0x00000001 - POS – 0x00000002 - LATEST_GPS_POS – 0x00000004 - OTA_POS – 0x00000008 - EXT_REF_POS – 0x00000010 - TIMETAG – 0x00000020 - CELLID – 0x00000040 - CACHED_CELLID – 0x00000080 - LAST_SRV_CELL – 0x00000100 - CUR_SRV_CELL – 0x00000200 - NEIGHBOR_INFO

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.11 **ULONG** SetPDSDefaults (**ULONG** *operation*, **BYTE** *timeout*, **ULONG** *interval*, **ULONG** *accuracy*)

Sets the default tracking session configuration. The tracking session configuration is used when a tracking session is automatically started using SetServiceAutomaticTracking or due to the device detecting an application opening the NMEA port.

Parameters

<i>operation</i>	<ul style="list-style-type: none"> Current session operating mode <ul style="list-style-type: none"> 0 - Standalone 1 - MS based 2 - MS assisted
<i>timeout</i>	<ul style="list-style-type: none"> Maximum amount of time (seconds) to work on each fix, maximum is 255
<i>interval</i>	<ul style="list-style-type: none"> Interval (seconds) between fix requests
<i>accuracy</i>	<ul style="list-style-type: none"> Preferred accuracy threshold (meters)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.12 **ULONG** SetPDSState (**ULONG** *enable*)

Sets the PDS state.

Parameters

<i>enable</i> [IN]	<ul style="list-style-type: none"> Desired PDS state <ul style="list-style-type: none"> Zero - disable Non-Zero - enable
--------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Seconds

9.20.4.13 ULONG SetPortAutomaticTracking (ULONG bAuto)

Sets the automatic tracking configuration for the NMEA COM port.

Parameters

<i>bAuto</i> [IN]	<ul style="list-style-type: none"> • Enable automatic tracking for NMEA COM port <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled
-------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.14 ULONG SetServiceAutomaticTracking (ULONG bAuto)

Sets the automatic tracking state for the service. Tracking session being started using the default session configuration. Auto-tracking continues to generate fixes indefinitely until requested to be disabled. In StartPDSTracking↔SessionExt a tracking session get started using the specified session control method and input parameters. After completion of requested no. of position fixes or service times out to perform fix, tracking session ends and GPS service deactivates.

Parameters

<i>bAuto</i> [IN]	<ul style="list-style-type: none"> • Automatic tracking session started for service <ul style="list-style-type: none"> – 0x00 - Disabled
Generated by Doxygen	<ul style="list-style-type: none"> – 0x01 - Enabled

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.15 **ULONG** SetXTRAAutomaticDownload (**ULONG** *bEnabled*, **USHORT** *interval*)

Sets the XTRA automatic database download configuration.

Parameters

<i>bEnabled</i> [IN]	<ul style="list-style-type: none"> Automatic XTRA download status <ul style="list-style-type: none"> 0 - Disabled 1 - Enabled
<i>interval</i> [IN]	<ul style="list-style-type: none"> Interval (hours) between XTRA downloads

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.16 **ULONG** SetXTRANetwork (**ULONG** *preference*)

Sets the XTRA WWAN network preference. When automatic XTRA database downloading is enabled this preference determines which WWAN networks will be used to perform the XTRA database download.

Parameters

<i>preference</i> [IN]	<ul style="list-style-type: none"> XTRA WWAN network preference <ul style="list-style-type: none"> 0x00 - None (any available network)
	<ul style="list-style-type: none"> 0x01 - Home-only, only when on home systems 0x02 - Roam-only, only when on non-home systems

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.17 `ULONG SLQSGetAGPSConfig (ULONG * pServerAddress, ULONG * pServerPort, BYTE * pServerURL, BYTE * pServerURLLength, BYTE * pNetworkMode)`

Gets the PDS AGPS (MS-based) configuration.

Parameters

<i>pServerAddress</i> [OUT]	<ul style="list-style-type: none"> IPv4 address of AGPS server. "0" if not set
<i>pServerPort</i> [OUT]	<ul style="list-style-type: none"> Port number of AGPS server. "0" if not set
<i>pServerURL</i> [OUT]	<ul style="list-style-type: none"> URL of the AGPS server. "0" if not set
<i>pServerURLLength</i> [OUT]	<ul style="list-style-type: none"> URL length of AGPS server. "0" if not set
<i>pNetworkMode</i> [IN]	<ul style="list-style-type: none"> Network Mode of AGPS Server [optional - should be present in Multimode Systems] <ul style="list-style-type: none"> 0x00 - UMTS 0x01 - CDMA

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.18 ULONG SLQSGetGPSSStateInfo (GPSSStateInfo * pGPSSStateInfo)

Queries the MSM GPS server for receiver state information

Parameters

<i>pGPSSStateInfo</i> <i>Info[OUT]</i>	<ul style="list-style-type: none">• contains the GPS State Info• See GPSSStateInfo for more information
---	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Seconds

9.20.4.19 ULONG SLQSPDSDeterminePosition ()

Requests the MSM GPS service to obtain the current position for manually controlled tracking sessions.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Seconds

9.20.4.20 **ULONG** SLQSPDSInjectAbsoluteTimeReference (**ULONGLONG** *timeMsec*, **ULONG** *timeUncMsec*, **BYTE** *timeBase*, **BYTE** *forceFlag*)

Injects a absolute time reference into the PDS engine.

Parameters

<i>timeMsec</i> [IN]	<ul style="list-style-type: none">Represents the number of milliseconds elapsed since either a GPS or UTC time base. If the time base is UTC, this value should NOT include leap seconds
<i>timeUncMsec</i> [↔IN]	<ul style="list-style-type: none">Time uncertainty in milliseconds
<i>timeBase</i> [IN]	<ul style="list-style-type: none">Time base<ul style="list-style-type: none">0x00 - GPS (midnight, Jan 6, 1980)0x01 - UTC (midnight, Jan 1, 1970)
<i>forceFlag</i> [IN]	<ul style="list-style-type: none">Force acceptance of data

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.21 **ULONG** SLQSPDSInjectPositionData (**struct** **PDSPositionData** * *pPositionData*)

Injects position data into the PDS engine.

Parameters

<i>pPositionData</i> [↔IN]	<ul style="list-style-type: none">contains the position data to be injected to the PDS engine
----------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.22 `ULONG SLQSSetAGPSConfig (ULONG * pServerAddress, ULONG * pServerPort, BYTE * pServerURL, BYTE * pServerURLLength, BYTE * pNetworkMode)`

Sets the PDS AGPS (MS-based) configuration.

Parameters

<i>pServer↔ Address</i> [IN]	<ul style="list-style-type: none"> IPv4 address of AGPS server [optional]
<i>pServerPort</i> [IN]	<ul style="list-style-type: none"> Port number of AGPS server [optional - should be present when pServerAddress is present]
<i>pServerURL</i> [IN]	<ul style="list-style-type: none"> URL of the AGPS server [optional]
<i>pServerURL↔ Length</i> [IN]	<ul style="list-style-type: none"> URL length of AGPS server [optional - should be present when pServerURL is present]
<i>pNetwork↔ Mode</i> [IN]	<ul style="list-style-type: none"> Network Mode of AGPS Server [optional - should be present in Multimode Systems] <ul style="list-style-type: none"> 0x00 - UMTS 0x01 - CDMA

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.23 `ULONG SLQSSetPositionMethodState (PDSPosMethodStateReq * pPDSPosMethodStateReq)`

Sets the state of positioning methods for the device.

Parameters

<p><i>pDSPos</i>↔ <i>MethodState</i>↔ <i>Req[IN]</i></p>	<ul style="list-style-type: none"> • See PDSPosMethodStateReq for more information
--	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Seconds

9.20.4.24 **ULONG** StartPDSTrackingSessionExt (**BYTE** *sessionControl*, **BYTE** *sessionType*, **BYTE** *sessionOperation*, **BYTE** *sessionServerOption*, **BYTE** *fixTimeout*, **ULONG** *fixInterval*, **ULONG** *fixCount*, **ULONG** *fixAccuracy*)

This function starts a PDS tracking session.

Parameters

<p><i>session</i>↔ <i>Control[IN]</i></p>	<ul style="list-style-type: none"> • Control method: <ul style="list-style-type: none"> – 0x0 - Manual
<p><i>sessionType[IN]</i></p>	<ul style="list-style-type: none"> • Type: <ul style="list-style-type: none"> – 0x0 - New
<p><i>session</i>↔ <i>Operation[IN]</i></p>	<ul style="list-style-type: none"> • Operating mode: <ul style="list-style-type: none"> – 0x00 - Standalone – 0x01 - MS-based
<p><i>sessionServer</i>↔ <i>Option[IN]</i></p>	<ul style="list-style-type: none"> • Location server option: <ul style="list-style-type: none"> – 0x0 - Default
<p><i>fixTimeout[IN]</i></p>	<ul style="list-style-type: none"> • Maximum time to work on each fix (in seconds, max 255)
<p><i>fixCount[IN]</i></p>	<ul style="list-style-type: none"> • Count of position fix requests for this session (must be at least 1)
<p><i>fixInterval[IN]</i></p>	<ul style="list-style-type: none"> • interval between position fix requests (in seconds)
<p><i>fixAccuracy[IN]</i></p>	<ul style="list-style-type: none"> • Preferred accuracy threshold(in meters)
Generated by Doxygen	

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.20.4.25 ULONG StopPDSTrackingSession ()

This function stops a PDS tracking session.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Seconds

9.21 qaGobiApiQos.h File Reference

Quality of Service API function prototypes.

Data Structures

- struct [dataRate](#)
- struct [tokenBucket](#)
- struct [pktErrRate](#)
- struct [swiQosFlow](#)
- struct [IPv4Addr](#)
- struct [Tos](#)
- struct [IPv6Addr](#)
- struct [IPv6TrafCls](#)
- struct [Port](#)
- struct [swiQosFilter](#)
- struct [swiQosReq](#)
- struct [swiQosIds](#)
- struct [swiQosModifyReq](#)
- struct [swiQosGranted](#)
- struct [NWProfile](#)
- struct [sQosFlowStat](#)
- struct [sQosStat](#)
- struct [sApnExtraParams](#)

Macros

- #define [MAX_QOS_SPEC_PER_APN](#) (10)
- #define [MAX_QOS_FILTER_TLV](#) 25

Functions

- [ULONG SLQSQosReset](#) ([BYTE](#) instance)
- [ULONG SLQSQosReq](#) ([BYTE](#) instance, [swiQosReq](#) *pQosReq, [swiQosIds](#) *pQosResp)
- [ULONG SLQSQosRel](#) ([BYTE](#) instance, [swiQosIds](#) *pQosIds)
- [ULONG SLQSQosSuspend](#) ([BYTE](#) instance, [swiQosIds](#) *pQosIds)
- [ULONG SLQSQosResume](#) ([BYTE](#) instance, [swiQosIds](#) *pQosIds)
- [ULONG SLQSQosModify](#) ([BYTE](#) instance, [swiQosModifyReq](#) *pReq)
- [ULONG SLQSQosGetGranted](#) ([BYTE](#) instance, [ULONG](#) id, [swiQosGranted](#) *pGranted)
- [ULONG SLQSQosGetFlowStatus](#) ([BYTE](#) instance, [ULONG](#) id, [BYTE](#) *pStatus)
- [ULONG SLQSQosGetNetworkStatus](#) ([BYTE](#) instance, [BYTE](#) *pStatus)
- [ULONG SLQSQosGetNWProf](#) ([BYTE](#) instance, [BYTE](#) *pSz, [NWProfile](#) *pProfile)
- [ULONG SLQSQosSwiReadDataStats](#) ([BYTE](#) instance, [ULONG](#) apnId, [sQosStat](#) *pQosStat)
- [ULONG SLQSQosSwiReadApnExtraParams](#) ([BYTE](#) instance, [ULONG](#) apnId, [sApnExtraParams](#) *pApn↔ExtraParams)

9.21.1 Detailed Description

Quality of Service API function prototypes.

9.21.2 Macro Definition Documentation

9.21.2.1 #define [MAX_QOS_FILTER_TLV](#) 25

9.21.2.2 #define [MAX_QOS_SPEC_PER_APN](#) (10)

9.21.3 Function Documentation

9.21.3.1 [ULONG SLQSQosGetFlowStatus](#) ([BYTE](#) *instance*, [ULONG](#) *id*, [BYTE](#) * *pStatus*)

Get the status of a QoS flow.

Parameters

in	<i>instance</i>	<ul style="list-style-type: none"> • QMI instance
in	<i>id[IN]</i>	Qos identifier Index identifying the QoS flow that has been negotiated
out	<i>pStatus[OUT]</i>	Qos status Current QoS instance status: <ul style="list-style-type: none"> • 0x01 – QMI_QOS_STATUS_ACTIVATED • 0x02 – QMI_QOS_STATUS_SUSPENDED • 0x03 – QMI_QOS_STATUS_GONE

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

9.21.3.2 ULONG SLQSQosGetGranted (BYTE *instance*, ULONG *id*, swiQosGranted * *pGranted*)

Retrieve the QoS parameters that are in effect for the specified QoS 16 flow as a result of network negotiation

Parameters

in	<i>instance</i>	<ul style="list-style-type: none">• QMI instance
in	<i>id</i> [IN]	<ul style="list-style-type: none">• Qos identifier• Index identifying the QoS flow that has been negotiated
in	<i>pGranted</i> [OUT]	<ul style="list-style-type: none">• Tx/Rx Qos granted flow• See swiQosGranted for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

9.21.3.3 ULONG SLQSQosGetNetworkStatus (BYTE *instance*, BYTE * *pStatus*)

Queries whether the device is currently on a network that supports QoS

Parameters

	<i>instance</i> [IN]	<ul style="list-style-type: none">• QMI instance
out	<i>pStatus</i> [OUT]	Network QoS support status <ul style="list-style-type: none">• 0 – No QoS support in network• 1 – Network supports QoS

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA

9.21.3.4 ULONG SLQSQosGetNWProf (BYTE *instance*, BYTE * *pSz*, NWProfile * *pProfile*)

Get network supported QoS profile information

Parameters

<i>in</i>	<i>instance</i>	<ul style="list-style-type: none">• QMI instance

9.21.3.5 ULONG SLQSQosModify (BYTE *instance*, swiQosModifyReq * *pReq*)

Resume one or more existing QoS flows

Warning

NOT IMPLEMENTED

Parameters

<i>in</i>	<i>instance</i>	<ul style="list-style-type: none">• QMI instance
	<i>pReq[IN]</i>	<ul style="list-style-type: none">• See swiQosModifyReq for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

9.21.3.6 ULONG SLQSQosRel (BYTE *instance*, swiQosIds * *pQosIds*)

Release one or more existing QoS flows

Parameters

<i>in</i>	<i>instance</i>	<ul style="list-style-type: none">• QMI instance
	<i>pQosIds[IN]</i>	<ul style="list-style-type: none">• See swiQosIds for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

9.21.3.7 ULONG SLQSQosReq (BYTE *instance*, swiQosReq * *pQosReq*, swiQosIds * *pQosResp*)

Triggers QoS negotiation by providing QoS parameters

Parameters

<i>instance[IN]</i>	<ul style="list-style-type: none">• QMI instance
<i>pQoSReq[IN]</i>	<ul style="list-style-type: none">• See swiQosReq for more information
<i>pQosResp[OUT]</i>	<ul style="list-style-type: none">• See swiQosIds for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS/CDMA
Device Supported: MC77XX
Timeout: 2 seconds

At least one pair of optional TLVs must be present; Tx QoS Flow Request and Tx QoS Filter Request TLVs must both be present if either one is present; Rx QoS Flow Request and Rx QoS Filter Request TLVs must both be present if either one is present

9.21.3.8 ULONG SLQSQosReset (BYTE *instance*)

Reset the QoS service state variables of the requesting control point

Parameters

<i>in</i>	<i>instance</i>	<ul style="list-style-type: none">• QMI instance
-----------	-----------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA, UMTS & LTE
Device Supported: MC7750
Timeout: 2 seconds

9.21.3.9 ULONG SLQSQosResume (BYTE *instance*, swiQosIds * *pQosIds*)

Resume one or more existing QoS flows

Parameters

<i>in</i>	<i>instance</i>	<ul style="list-style-type: none">• QMI instance
	<i>pQosIds</i> [IN]	<ul style="list-style-type: none">• See swiQosIds for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

9.21.3.10 ULONG SLQSQoSSuspend (BYTE *instance*, swiQosIds * *pQosIds*)

Suspend one or more existing QoS flows

Parameters

in	<i>instance</i>	<ul style="list-style-type: none">• QMI instance
	<i>pQosIds[IN]</i>	<ul style="list-style-type: none">• See swiQosIds for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

9.21.3.11 ULONG SLQSQoSReadApnExtraParams (BYTE *instance*, ULONG *apnId*, sApnExtraParams * *pApnExtraParams*)

Queries extra APN parameters that are not reported by existing QCT QMI service

Parameters

in	<i>instance</i>	<ul style="list-style-type: none">• QMI instance
in	<i>apnId</i>	<ul style="list-style-type: none">• APN id
out	<i>pApnExtra↔ Params</i>	See sApnExtraParams for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.21.3.12 ULONG SLQSQosSwiReadDataStats (BYTE *instance*, ULONG *apnId*, sQosStat * *pQosStat*)

Get the current number of packets and bytes sent, dropped and received for each UL, DL bearer and a sum of them for UL and DL direction in the modem.

Parameters

in	<i>instance</i>	<ul style="list-style-type: none"> QMI instance
in	<i>apnId</i>	<ul style="list-style-type: none"> APN id
out	<i>pQosStat</i>	See sQosStat for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.22 qaGobiApiRms.h File Reference

Remote Management Service API function prototypes.

Functions

- [ULONG GetSMSWake](#) ([ULONG](#) *pEnabled, [ULONG](#) *pWakeMask)
- [ULONG SetSMSWake](#) ([ULONG](#) bEnable, [ULONG](#) wakeMask)

9.22.1 Detailed Description

Remote Management Service API function prototypes.

9.22.2 Function Documentation

9.22.2.1 ULONG GetSMSWake (ULONG * *pEnabled*, ULONG * *pWakeMask*)

Queries the state of the SMS wake functionality. When enabled SMS wake functionality results in incoming messages being searched for the configured mask. Upon detection of the mask the incoming message is deleted (i.e. not stored in memory) and the device attempts to wake the host (requires host platform support).

Parameters

<i>pEnabled</i> [OUT]	<ul style="list-style-type: none"> • SMS wake functionality enabled <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled
<i>pWakeMask</i> [O↔UT]	<ul style="list-style-type: none"> • SMS wake mask to search for incoming messages (only relevant when enabled)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.22.2.2 ULONG SetSMSWake (ULONG *bEnable*, ULONG *wakeMask*)

Configures the SMS wake functionality. When enabled SMS wake functionality results in incoming messages being searched for the configured mask. Upon detection of the mask the incoming message is deleted (i.e. not stored in memory) and the device attempts to wake the host (requires host platform support).

Parameters

<i>bEnable</i>	<ul style="list-style-type: none"> • Enable SMS wake functionality <ul style="list-style-type: none"> – Zero - Disable – Non-Zero - Enable
<i>wakeMask</i>	<ul style="list-style-type: none"> • SMS wake mask to search for incoming messages (only relevant when enabling)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.23 qaGobiApiSar.h File Reference

Specific Absorption Rate API function prototypes.

Enumerations

- enum [eQMISARRFState](#) {
 [QMI_SAR_RF_STATE_DEFAULT](#) = 0,
 [QMI_SAR_RF_STATE_1](#),
 [QMI_SAR_RF_STATE_2](#),
 [QMI_SAR_RF_STATE_3](#),
 [QMI_SAR_RF_STATE_4](#),
 [QMI_SAR_RF_STATE_5](#),
 [QMI_SAR_RF_STATE_6](#),
 [QMI_SAR_RF_STATE_7](#),
 [QMI_SAR_RF_STATE_8](#) }

Functions

- [ULONG SLQSGetRfSarState](#) ([ULONG](#) *pSarRFState)
- [ULONG SLQSSetRfSarState](#) ([ULONG](#) RfSarState)

9.23.1 Detailed Description

Specific Absorption Rate API function prototypes.

9.23.2 Enumeration Type Documentation

9.23.2.1 enum [eQMISARRFState](#)

This enum contains the SAR RF States

Parameters

SAR	RF State
	<ul style="list-style-type: none">• QMI_SAR_RF_STATE_DEFAULT = 0• QMI_SAR_RF_STATE_1• QMI_SAR_RF_STATE_2• QMI_SAR_RF_STATE_3• QMI_SAR_RF_STATE_4• QMI_SAR_RF_STATE_5• QMI_SAR_RF_STATE_6• QMI_SAR_RF_STATE_7• QMI_SAR_RF_STATE_8

Enumerator

[QMI_SAR_RF_STATE_DEFAULT](#)

QMI_SAR_RF_STATE_1
QMI_SAR_RF_STATE_2
QMI_SAR_RF_STATE_3
QMI_SAR_RF_STATE_4
QMI_SAR_RF_STATE_5
QMI_SAR_RF_STATE_6
QMI_SAR_RF_STATE_7
QMI_SAR_RF_STATE_8

9.23.3 Function Documentation

9.23.3.1 ULONG SLQSGetRfSarState (ULONG * *pSarRFState*)

Gets the specified RF SAR state.

Parameters

<i>pSarRFState</i>	<ul style="list-style-type: none"> • SAR RF State <ul style="list-style-type: none"> – QMI_SAR_RF_STATE_DEFAULT – QMI_SAR_RF_STATE_1 – QMI_SAR_RF_STATE_2 – QMI_SAR_RF_STATE_3 – QMI_SAR_RF_STATE_4 – QMI_SAR_RF_STATE_5 – QMI_SAR_RF_STATE_6 – QMI_SAR_RF_STATE_7 – QMI_SAR_RF_STATE_8
--------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.23.3.2 ULONG SLQSSetRfSarState (ULONG *RfSarState*)

Sets the specified RF SAR state.

Parameters

<i>sar_rf_state</i>	<ul style="list-style-type: none"> • SAR RF State <ul style="list-style-type: none"> – QMI_SAR_RF_STATE_DEFAULT – QMI_SAR_RF_STATE_1 – QMI_SAR_RF_STATE_2 – QMI_SAR_RF_STATE_3 – QMI_SAR_RF_STATE_4 – QMI_SAR_RF_STATE_5 – QMI_SAR_RF_STATE_6 – QMI_SAR_RF_STATE_7 – QMI_SAR_RF_STATE_8
---------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.24 qaGobiApiSms.h File Reference

Short Message Service API function prototypes.

Data Structures

- struct [slqssendsmsparams_s](#)
- struct [BroadcastConfig](#)
- struct [_qaQmi3GPPBroadcastCfgInfo](#)
- struct [CDMABroadcastConfig](#)
- struct [_qaQmi3GPP2BroadcastCfgInfo](#)
- struct [cdmaMsgEncodingParams](#)
- struct [cdmaMsgDecodingParams](#)
- struct [wcdmaMsgEncodingParams](#)
- struct [wcdmaMsgDecodingParams](#)
- struct [wcdmaLongMsgDecodingParams](#)
- struct [_transLayerinfo](#)
- struct [_getTransLayerInfoResp](#)
- struct [_getTransNWRegInfoResp](#)
- struct [_getIndicationRegResp](#)
- struct [_setIndicationRegReq](#)
- struct [smsRouteEntry](#)
- struct [smsSetRoutesReq](#)
- struct [smsMsgprotocolResp](#)
- struct [smsMaxStorageSizeReq](#)
- struct [smsMaxStorageSizeResp](#)
- struct [messageWaitingInfoContent](#)
- struct [getMsgWaitingInfo](#)
- struct [slqssendasyncsmsparams_s](#)

Macros

- `#define CONFIG_LEN 0x05`
- `#define TIME_STAMP_BUF 0x08`
- `#define ABSOLUTE_VALIDITY 0x08`
- `#define TIME_DATE_BUF 0x09`
- `#define MAX_SMS_ROUTES 0x0A`
- `#define NUM_OF_SET 0xFF`

Typedefs

- `typedef struct _qaQmi3GPPBroadcastCfgInfo qaQmi3GPPBroadcastCfgInfo`
- `typedef struct _qaQmi3GPP2BroadcastCfgInfo qaQmi3GPP2BroadcastCfgInfo`
- `typedef struct _transLayerInfo transLayerInfo`
- `typedef struct _getTransLayerInfoResp getTransLayerInfoResp`
- `typedef struct _getTransNWRegInfoResp getTransNWRegInfoResp`
- `typedef struct _getIndicationRegResp getIndicationRegResp`
- `typedef struct _setIndicationRegReq setIndicationRegReq`

Functions

- `ULONG SLQSDeleteSMS (ULONG storageType, ULONG *pMessageIndex, ULONG *pMessageTag, BYTE *pMessageMode)`
- `ULONG SLQSGetSMS (ULONG storageType, ULONG messageIndex, ULONG *pMessageTag, ULONG *pMessageFormat, ULONG *pMessageSize, BYTE *pMessage, BYTE *pMessageMode)`
- `ULONG SendSMS (ULONG messageFormat, ULONG messageSize, BYTE *pMessage, ULONG *pMessageFailureCode, BYTE *pSmsOnlms)`
- `ULONG SLQSSendSMS (slqssendsmsparams_s *pSendSmsParams)`
- `ULONG GetSMSCAddress (BYTE addressSize, CHAR *pSMSCAddress, BYTE typeSize, CHAR *pSMSCType)`
- `ULONG SetSMSCAddress (CHAR *pSMSCAddress, CHAR *pSMSCType)`
- `ULONG SaveSMS (ULONG storageType, ULONG messageFormat, ULONG messageSize, BYTE *pMessage, ULONG *pMessageIndex)`
- `ULONG SLQSGetSMSList (ULONG storageType, ULONG *pRequestedTag, ULONG *pMessageListSize, BYTE *pMessageList, BYTE *pMessageMode)`
- `ULONG SLQSMModifySMSStatus (ULONG storageType, ULONG messageIndex, ULONG messageTag, BYTE *pMessageMode)`
- `ULONG SLQSGetSmsBroadcastConfig (BYTE mode, qaQmi3GPPBroadcastCfgInfo *pBroadcastConfig, qaQmi3GPP2BroadcastCfgInfo *pCDMABroadcastConfig)`
- `ULONG SLQSSetSmsBroadcastConfig (BYTE mode, qaQmi3GPPBroadcastCfgInfo *pBroadcastConfig, qaQmi3GPP2BroadcastCfgInfo *pCDMABroadcastConfig)`
- `ULONG SLQSSetSmsBroadcastActivation (BYTE mode, BYTE broadcastActivate)`
- `ULONG SLQSCDMAEncodeMOTextMsg (struct cdmaMsgEncodingParams *pCdmaMsgEncodingParams)`
- `ULONG SLQSCDMADecodeMTTextMsg (struct cdmaMsgDecodingParams *pCdmaMsgDecodingParams)`
- `ULONG SLQSWCDMAEncodeMOTextMsg (struct wcdmaMsgEncodingParams *pWcdmaMsgEncodingParams)`
- `ULONG SLQSWCDMADecodeMTTextMsg (struct wcdmaMsgDecodingParams *pWcdmaMsgDecodingParams)`
- `ULONG SLQSWCDMADecodeLongTextMsg (struct wcdmaLongMsgDecodingParams *pWcdmaLongMsgDecodingParams)`
- `ULONG SLQSGetTransLayerInfo (getTransLayerInfoResp *pGetTransLayerInfoResp)`
- `ULONG SLQSGetTransNWRegInfo (getTransNWRegInfoResp *pGetTransNWRegInfoResp)`
- `ULONG SLQSGetIndicationRegister (getIndicationRegResp *pGetIndicationRegInfo)`

- [ULONG SLQSSetIndicationRegister](#) ([setIndicationRegReq](#) *pSetIndicationRegReq)
- [ULONG SLQSSmsSetRoutes](#) ([smsSetRoutesReq](#) *pSetRoutesReq)
- [ULONG SLQSSmsGetMessageProtocol](#) ([smsMsgprotocolResp](#) *pMessageProtocol)
- [ULONG SLQSSmsGetMaxStorageSize](#) ([smsMaxStorageSizeReq](#) *pMaxStorageSizeReq, [smsMaxStorageSizeResp](#) *pMaxStorageSizeResp)
- [ULONG SLQSGetMessageWaiting](#) ([getMsgWaitingInfo](#) *pGetMsgWaitingInfoResp)
- [ULONG SLQSSendAsyncSMS](#) ([slqssendasyncsmsparams_s](#) *pSendSmsParams)
- [ULONG SLQSSetSmsStorage](#) ([BYTE](#) smsStorage)
- [ULONG SLQSSwiGetSMSStorage](#) ([ULONG](#) *pSmsStorage)
- [ULONG SLQSSendLongSMS](#) ([ULONG](#) messageFormat, [ULONG](#) messageSize, [CHAR](#) *pMessage, [BYTE](#) encodingScheme, [ULONG](#) *pMessageFailureCode, [CHAR](#) *pMobileNum, [BYTE](#) *pSmsOnIMS)

9.24.1 Detailed Description

Short Message Service API function prototypes.

9.24.2 Macro Definition Documentation

9.24.2.1 `#define ABSOLUTE_VALIDITY 0x08`

9.24.2.2 `#define CONFIG_LEN 0x05`

9.24.2.3 `#define MAX_SMS_ROUTES 0x0A`

9.24.2.4 `#define NUM_OF_SET 0xFF`

9.24.2.5 `#define TIME_DATE_BUF 0x09`

9.24.2.6 `#define TIME_STAMP_BUF 0x08`

9.24.3 Typedef Documentation

9.24.3.1 `typedef struct __getIndicationRegResp getIndicationRegResp`

This structure contains Get Indication Register Response parameters

Parameters

<i>pRegTransLayerInfoEvt</i>	- <ul style="list-style-type: none"> • Optional 1 BYTE parameter indicating registration status of transport layer information events • Values: <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled • function SLQSGetIndicationRegister() returns a default value 0xFF if this parameter is allocated memory in the structure and no response is received from the device.
------------------------------	---

<i>pRegTransNW↔ RegInfoEvt</i>	- <ul style="list-style-type: none"> Optional 1 BYTE parameter indicating registration status of transport network registration information events Values: <ul style="list-style-type: none"> 0x00 - Disabled 0x01 - Enabled function SLQSGetIndicationRegister() returns a default value 0xFF if this parameter is allocated memory in the structure and no response is received from the device.
<i>pRegCallStat↔ InfoEvt</i>	- <ul style="list-style-type: none"> Optional 1 BYTE parameter indicating registration status of call status information events Values: <ul style="list-style-type: none"> 0x00 - Disabled 0x01 - Enabled function SLQSGetIndicationRegister() returns a default value 0xFF if this parameter is allocated memory in the structure and no response is received from the device.

9.24.3.2 typedef struct _getTransLayerInfoResp getTransLayerInfoResp

This structure contains Get Transport Layer Info Response parameters

Parameters

<i>pRegInd</i>	- <ul style="list-style-type: none"> Optional parameter indicating if transport layer is registered Values: <ul style="list-style-type: none"> 0x00 - Transport layer is not registered 0x01 - Transport layer is registered function SLQSGetTransLayerInfo() returns a default value 0xFF if no response is received from the device.
<i>pTransLayerInfo</i>	<ul style="list-style-type: none"> Pointer to structure of transLayerInfo. <ul style="list-style-type: none"> Optional parameter See transLayerInfo for more information function SLQSGetTransLayerInfo() returns a default value 0xFF for parameter values if no response is received from the device.

9.24.3.3 typedef struct _getTransNWRegInfoResp getTransNWRegInfoResp

This structure contains transport network registration info parameter

Parameters

<i>pRegStatus</i>	- <ul style="list-style-type: none"> • Optional 1 BYTE parameter indicating transport layer network registration status • Values: <ul style="list-style-type: none"> – 0x00 - No service – 0x01 - In progress – 0x02 - Failed – 0x03 - Limited Service – 0x04 - Full Service • function SLQSGetTransNWRegInfo() returns a default value 0xFF if no response is received from the device.
-------------------	---

9.24.3.4 typedef struct _qaQmi3GPP2BroadcastCfgInfo qaQmi3GPP2BroadcastCfgInfo

This structure contains the 3GPP2 Broadcast Configuration Information parameters

Parameters

<i>activated_ind</i>	<ul style="list-style-type: none"> • Broadcast SMS <ul style="list-style-type: none"> – 0x00 - Deactivated – 0x01 - Activated
<i>num_instances</i>	<ul style="list-style-type: none"> • Number of sets (N) of parameters Following each set describes one entry in the broadcast configuration table. <ul style="list-style-type: none"> – serviceCategory – language – selected
<i>broadcastConfig</i>	<ul style="list-style-type: none"> • A CDMABroadcastConfig structure array. • Further defined by the structure CDMABroadcastConfig

9.24.3.5 typedef struct _qaQmi3GPPBroadcastCfgInfo qaQmi3GPPBroadcastCfgInfo

This structure contains the 3GPP Broadcast Configuration Information parameters

Parameters

<i>activated_ind</i>	<ul style="list-style-type: none"> • Broadcast SMS <ul style="list-style-type: none"> – 0x00 - Deactivated – 0x01 - Activated
----------------------	---

<i>num_instances</i>	<ul style="list-style-type: none"> • Number of sets (N) of parameters Following each set describes one entry in the broadcast configuration table. <ul style="list-style-type: none"> – fromServiceId – toServiceId – selected
<i>broadcastConfig</i>	<ul style="list-style-type: none"> • A BroadcastConfig structure array. • Further defined by the structure BroadcastConfig

9.24.3.6 typedef struct _setIndicationRegReq setIndicationRegReq

This structure contains Indication Register request parameters

Parameters

<i>pRegTrans↔ LayerInfoEvt</i>	- <ul style="list-style-type: none"> • Optional 1 BYTE parameter indicating registration status of transport layer information events • Values: <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled – NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device
<i>pRegTransNW↔ RegInfoEvt</i>	- <ul style="list-style-type: none"> • Optional 1 BYTE parameter indicating registration status of transport network registration information events • Values: <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled – NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device
<i>pRegCallStat↔ InfoEvt</i>	- <ul style="list-style-type: none"> • Optional 1 BYTE parameter indicating registration status of call status information events • Values: <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled – NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device

9.24.3.7 typedef struct _transLayerinfo transLayerInfo

This structure contains Transport Layer Information

Parameters

<i>TransType</i>	<ul style="list-style-type: none">• Transport Type<ul style="list-style-type: none">– 0x00 - IMS
<i>TransCap</i>	<ul style="list-style-type: none">• Transport Capability• Values:<ul style="list-style-type: none">– 0x00 - CDMA– 0x01 - GW

9.24.4 Function Documentation

9.24.4.1 **ULONG** GetSMSCAddress (**BYTE** *addressSize*, **CHAR** * *pSMSCAddress*, **BYTE** *typeSize*, **CHAR** * *pSMSCType*)

Gets the SMS center address.

Parameters

<i>addressSize</i>	<ul style="list-style-type: none">• The maximum number of characters (including NULL terminator) that the SMS center address array can contain.
<i>pSMSCAddress</i> [0↔UT]	<ul style="list-style-type: none">• The SMS center address represented as a NULL terminated string.
<i>typeSize</i>	<ul style="list-style-type: none">• The maximum number of characters (including NULL terminator) that the SMS center address type array can contain.
<i>pSMSCType</i> [0↔UT]	<ul style="list-style-type: none">• The SMS center address type represented as a NULL terminated string.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 2 seconds

9.24.4.2 **ULONG** SaveSMS (**ULONG** *storageType*, **ULONG** *messageFormat*, **ULONG** *messageSize*, **BYTE** * *pMessage*, **ULONG** * *pMessageIndex*)

Saves an SMS message to device memory

Parameters

<i>storageType</i> [IN]	<ul style="list-style-type: none"> SMS message storage type <ul style="list-style-type: none"> 0 - UIM - Invalid in case of CDMA device that does not require SIM 1 - NV
<i>messageFormat</i> [IN]	<ul style="list-style-type: none"> Message format <ul style="list-style-type: none"> 0 - CDMA (IS-637B) 1 - 5 (Reserved) 6 - GSM/WCDMA PP
<i>messageSize</i>	<ul style="list-style-type: none"> The length of the message contents in bytes
<i>pMessage</i> [IN]	<ul style="list-style-type: none"> The message contents
<i>pMessageIndex</i> [OUT]	<ul style="list-style-type: none"> The message index assigned by the device

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 10 seconds

9.24.4.3 **ULONG** SendSMS (**ULONG** *messageFormat*, **ULONG** *messageSize*, **BYTE** * *pMessage*, **ULONG** * *pMessageFailureCode*, **BYTE** * *pSmsOnlms*)

Sends an SMS message for immediate over-the-air transmission

Parameters

<i>message</i> ↔ <i>Format</i> [IN]	<ul style="list-style-type: none"> • Message format <ul style="list-style-type: none"> – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
<i>messageSize</i> [IN]	<ul style="list-style-type: none"> • The length of the message contents in bytes
<i>pMessage</i> [IN]	<ul style="list-style-type: none"> • The message contents in PDU format contains SMS header and payload message
<i>pSmsOnIms</i> [IN]	<ul style="list-style-type: none"> • (Optional) SMS on IMS • The message is to be sent on IMS. <ul style="list-style-type: none"> – 0x00 Message is not to be sent on IMS. – 0x01 Message is to be sent on IMS. – 0x02 to 0xFF Reserved.
<i>pMessage</i> ↔ <i>FailureCode</i> [O↔ UT]	<ul style="list-style-type: none"> • (Optional) Message Failure Code • pointer to message failure code. If cause code is not provided, then value will be 0xFFFFFFFF

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 5 minutes

9.24.4.4 ULONG SetSMSCAddress (CHAR * *pSMSCAddress*, CHAR * *pSMSCType*)

Sets the SMS center address.

Parameters

<i>pSMSC</i> ↔ <i>Address</i> [IN]	<ul style="list-style-type: none"> • The SMS center address represented as a NULL terminated string
<i>pSMSCType</i> [IN]	<ul style="list-style-type: none"> • The SMS center address type represented as a NULL terminated string (optional).

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 5 seconds

9.24.4.5 ULONG SLQSCDMADecodeMTTextMsg (struct cdmaMsgDecodingParams * pCdmaMsgDecodingParams)

Decodes text message to CDMA PDU message

Parameters

<p><i>pMsgToBe↔</i> <i>EncodedCDM↔</i> <i>A[IN/OUT]</i></p>	<ul style="list-style-type: none"> • Pointer to structure containing parameters needed for decoding
---	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: None

9.24.4.6 ULONG SLQSCDMAEncodeMOTextMsg (struct cdmaMsgEncodingParams * pCdmaMsgEncodingParams)

Encodes text message to CDMA PDU message.

Parameters

<p><i>pMsgToBe↔</i> <i>EncodedCDM↔</i> <i>A[IN/OUT]</i></p>	<ul style="list-style-type: none"> • SLQS Runtime Settings Information
---	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: None

9.24.4.7 ULONG SLQSDDeleteSMS (ULONG *storageType*, ULONG * *pMessageIndex*, ULONG * *pMessageTag*, BYTE * *pMessageMode*)

Deletes one or more SMSs from device memory. If both of the optional parameters, *messageIndex* and *messageTag*, are NULL, all messages are deleted from the storage location specified in the mandatory *storageType* parameter. The optional index and tag parameters narrow the range of messages being deleted. If an index is specified, the single message with the index from the specified memory store will be deleted. If a tag is specified, all messages in the specified memory store whose tag matches that specified will be deleted.

There are three ways to use this message:

- Specify *storageType* only
 - Deletes all messages from memory storage
- Specify *storageType* and a tag
 - Deletes all messages from memory storage that match the given message tag
- Specify *storageType* and an index
 - Deletes only the message with the given index from memory storage

Parameters

<i>storageType</i>	<ul style="list-style-type: none"> • SMS message storage type <ul style="list-style-type: none"> – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
<i>pMessageIndex</i> [IN]	<ul style="list-style-type: none"> • (Optional) message index
<i>pMessageTag</i> [IN]	<ul style="list-style-type: none"> • (Optional) message tag <ul style="list-style-type: none"> – 0 - Read – 1 - Not read – 2 - Mobile originated and sent – 3 - Mobile originated but not yet sent
<i>pMessageMode</i> [IN]	<ul style="list-style-type: none"> • (Optional) message mode
Generated by Doxygen	<ul style="list-style-type: none"> • 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 10 seconds

9.24.4.8 ULONG SLQSGetIndicationRegister (getIndicationRegResp * pGetIndicationRegInfo)

This API provides registration state of different WMS indications.

Parameters

<i>pGetIndicationRegInfo</i>	[OUT] <ul style="list-style-type: none">• Pointer to structure of getIndicationRegResp<ul style="list-style-type: none">– See getIndicationRegResp for more information
------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Secs

9.24.4.9 ULONG SLQSGetMessageWaiting (getMsgWaitingInfo * pGetMsgWaitingInfoResp)

This API provides information about the message waiting information.

Parameters

<i>pGetMsgWaitingInfoResp</i>	[OUT] <ul style="list-style-type: none">• Pointer to structure of getMsgWaitingInfoResp<ul style="list-style-type: none">– See getMsgWaitingInfoResp for more information
-------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Secs

9.24.4.10 **ULONG** SLQSGetSMS (**ULONG** *storageType*, **ULONG** *messageIndex*, **ULONG** * *pMessageTag*, **ULONG** * *pMessageFormat*, **ULONG** * *pMessageSize*, **BYTE** * *pMessage*, **BYTE** * *pMessageMode*)

Returns an SMS from device memory.

Parameters

<i>storageType</i>	<ul style="list-style-type: none"> SMS message storage type <ul style="list-style-type: none"> 0 - UIM - Invalid in case of CDMA device that does not require SIM 1 - NV
<i>messageIndex</i>	<ul style="list-style-type: none"> Message index
<i>pMessageTag</i> [↔ OUT]	<ul style="list-style-type: none"> Message tag <ul style="list-style-type: none"> 0 - Read 1 - Not read 2 - Mobile originated and sent 3 - Mobile originated but not yet sent
<i>pMessage</i> ↔ <i>Format</i> [OUT]	<ul style="list-style-type: none"> Message format <ul style="list-style-type: none"> 0 - CDMA (IS-637B) 1 - 5 (Reserved) 6 - GSM/WCDMA PP
<i>pMessage</i> ↔ <i>Size</i> [IN/OUT]	<ul style="list-style-type: none"> Upon input the maximum number of bytes that can be written to the message array. Upon successful output the actual number of bytes written to the message array.
<i>pMessage</i> [OUT]	<ul style="list-style-type: none"> The message contents array
<i>pMessage</i> ↔ <i>Mode</i> [IN]	<ul style="list-style-type: none"> (Optional) Message Mode <ul style="list-style-type: none"> 0x00 - CDMA, LTE (if network type is CDMA) 0x01 - GW, LTE (if network type is UMTS)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 5 seconds

9.24.4.11 **ULONG** SLQSGetSmsBroadcastConfig (**BYTE** *mode*, **qaQmi3GPPBroadcastCfgInfo** * *pBroadcastConfig*, **qaQmi3GPP2BroadcastCfgInfo** * *pCDMABroadcastConfig*)

Provides Information about the SMS BroadcastConfiguration

Parameters

<i>mode</i> [IN]	<ul style="list-style-type: none"> Mode <ul style="list-style-type: none"> 0x00 - CDMA, LTE (if network type is CDMA) 0x01 - GW, LTE (if network type is UMTS)
<i>pBroadcastConfig</i> [OUT]	<ul style="list-style-type: none"> The data for 3GPP Broadcast Information(Optional).
<i>pCDMABroadcastConfig</i> [OUT]	<ul style="list-style-type: none"> The data for 3GPP2 Broadcast Information(Optional).

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 5 seconds

9.24.4.12 **ULONG** SLQSGetSMSList (**ULONG** *storageType*, **ULONG** * *pRequestedTag*, **ULONG** * *pMessageListSize*, **BYTE** * *pMessageList*, **BYTE** * *pMessageMode*)

Returns the list of SMS messages stored on the device.

Parameters

<i>storageType</i> [IN]	<ul style="list-style-type: none"> SMS message storage type <ul style="list-style-type: none"> 0 - UIM - Invalid in case of CDMA device that does not require SIM 1 - NV
<i>pRequestedTag</i> [IN]	<ul style="list-style-type: none"> (Optional) Message tag <ul style="list-style-type: none"> 0 - Read 1 - Not read 2 - Mobile originated and sent 3 - Mobile originated but not yet sent
<i>pMessageListSize</i> [IN/OUT]	<ul style="list-style-type: none"> Upon input the maximum number of elements that the message list array can contain. Upon successful output the actual number of elements in the message list array.
<i>pMessageList</i> [OUT]	<ul style="list-style-type: none"> The message list array
<i>pMessageMode</i> [IN]	<ul style="list-style-type: none"> (Optional) Message Mode <ul style="list-style-type: none"> 0x00 - CDMA, LTE (if network type is CDMA) 0x01 - GW, LTE (if network type is UMTS)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 5 seconds

9.24.4.13 ULONG SLQSGetTransLayerInfo (getTransLayerInfoResp * pGetTransLayerInfoResp)

This API provides information about the transport layer.

Parameters

<i>pGetTransLayerInfoResp</i>	[OUT] <ul style="list-style-type: none"> Pointer to structure of getTransLayerInfoResp <ul style="list-style-type: none"> See getTransLayerInfoResp for more information
-------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Secs

9.24.4.14 ULONG SLQSGetTransNWRegInfo (getTransNWRegInfoResp * pGetTransNWRegInfoResp)

This API provides transport layer network registration info.

Parameters

<i>pGetTransNWRegInfoResp</i>	[OUT] <ul style="list-style-type: none">• Pointer to structure of getTransNWRegInfoResp<ul style="list-style-type: none">– See getTransNWRegInfoResp for more information
-------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Secs

9.24.4.15 ULONG SLQSModifySMSStatus (ULONG storageType, ULONG messageIndex, ULONG messageTag, BYTE * pMessageMode)

Modifies the status of an SMS message saved in storage on the device.

Parameters

<i>storageType</i> [IN]	<ul style="list-style-type: none">• SMS message storage type<ul style="list-style-type: none">– 0 - UIM - Invalid in case of CDMA device that does not require SIM– 1 - NV
-------------------------	---

<i>message</i> ↔ <i>Index[IN]</i>	<ul style="list-style-type: none"> • Message index
<i>messageTag[IN]</i>	<ul style="list-style-type: none"> • Message tag <ul style="list-style-type: none"> – 0 - Read – 1 - Not read
<i>pMessage</i> ↔ <i>Mode[IN]</i>	<ul style="list-style-type: none"> • (Optional) Message Mode • 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 5 seconds

9.24.4.16 ULONG SLQSSendAsyncSMS ([slqssendasyncsmsparams_s](#) * *pSendSmsParams*)

Sends an SMS message for immediate over-the-air transmission

Parameters

<i>pSendSms</i> ↔ <i>Params</i>	<ul style="list-style-type: none"> • structure containing the SMS parameters. Refer slqssendasyncsmsparams_s
------------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 5 minutes

9.24.4.17 **ULONG** SLQSSendLongSMS (**ULONG** *messageFormat*, **ULONG** *messageSize*, **CHAR** * *pMessage*, **BYTE** *encodingScheme*, **ULONG** * *pMessageFailureCode*, **CHAR** * *pMobileNum*, **BYTE** * *pSmsOnIMS*)

Sends a long SMS message for immediate over-the-air transmission, a short SMS can be sent by this API as well, the input message is text string without any encoding

Parameters

<i>messageFormat</i> [IN]	<ul style="list-style-type: none"> Message format <ul style="list-style-type: none"> 0 - CDMA (IS-637B) 1 - 5 (Reserved) 6 - GSM/WCDMA PP
<i>messageSize</i> [IN]	<ul style="list-style-type: none"> Message size of the input message text
<i>pMessage</i> [IN]	<ul style="list-style-type: none"> Original message text
<i>encodingScheme</i> [IN]	<ul style="list-style-type: none"> Encoding method to generate the PDU <ul style="list-style-type: none"> 0 - 7 bit encoding 4 - 8 bit encoding 8 - 16 bit UCS2 encoding others value will be treated as default 7 bit encoding
<i>pMessageFailureCode</i> [OUT]	<ul style="list-style-type: none"> message failure code. If cause code is not provided, then value will be 0xFFFFFFFF
<i>pMobileNum</i> [IN]	<ul style="list-style-type: none"> Mobile number of the receiver
<i>pSmsOnIMS</i> [IN]	<ul style="list-style-type: none"> A flag indicates whether SMS was sent through IMS

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: 3GPP and 3GPP2, but 3GPP2 does not support multiple sms
Timeout: 5 minutes

9.24.4.18 ULONG SLQSSendSMS (*slqssendsmsparams_s* * *pSendSmsParams*)

Sends an SMS message for immediate over-the-air transmission

Parameters

<i>pSendSmsParams</i>	<ul style="list-style-type: none">• structure containing the SMS parameters. Refer slqssendsmsparams_s
-----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 5 minutes

9.24.4.19 ULONG SLQSSetIndicationRegister (*setIndicationRegReq* * *pSetIndicationRegReq*)

This API sets the registration state of different WMS indications.

Parameters

<i>pSetIndicationRegReq</i>	[IN] <ul style="list-style-type: none">• Pointer to structure of indicationRegReqParams<ul style="list-style-type: none">– See setIndicationRegReq for more information
-----------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Secs

9.24.4.20 ULONG SLQSSetSmsBroadcastActivation (BYTE mode, BYTE broadcastActivate)

Enables or disables the reception of broadcast SMS messages.

Parameters

<i>Mode</i> [IN]	<ul style="list-style-type: none"> • Mode • 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)
<i>broadcast↔ Activate</i> [IN]	<ul style="list-style-type: none"> • 0x00 - Disable broadcast • 0x01 - Activate broadcast

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 5 seconds

9.24.4.21 ULONG SLQSSetSmsBroadcastConfig (BYTE mode, qaQmi3GPPBroadcastCfgInfo * pBroadcastConfig, qaQmi3GPP2BroadcastCfgInfo * pCDMABroadcastConfig)

Sets the information about the SMS BroadcastConfiguration

Parameters

<i>mode</i> [IN]	<ul style="list-style-type: none"> • Mode <ul style="list-style-type: none"> – 0x00 - CDMA, LTE (if network type is CDMA) – 0x01 - GW, LTE (if network type is UMTS)
<i>pBroadcast↔ Config</i> [IN]	<ul style="list-style-type: none"> • The data for 3GPP Broadcast Information(Optional).
<i>pCDMA↔ Broadcast↔ Config</i> [IN]	<ul style="list-style-type: none"> • The data for 3GPP2 Broadcast Information(Optional).

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 5 seconds

9.24.4.22 ULONG SLQSSetSmsStorage (BYTE smsStorage)

Sets the SMS Storage on the device

Parameters

<i>smsStorage</i> [IN]	<ul style="list-style-type: none"> • SMS Storage <ul style="list-style-type: none"> – 0x01 - device's permanent memory – 0x02 - UICC
------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: ALL
Timeout: 5 seconds

9.24.4.23 ULONG SLQSSmsGetMaxStorageSize (smsMaxStorageSizeReq * pMaxStorageSizeReq, smsMaxStorageSizeResp * pMaxStorageSizeResp)

This API provides the maximum number of messages that can be stored in the specified memory storage. Also it provides the number of slots currently available

Parameters

<i>pMaxStorageSizeReq</i> [IN]	<ul style="list-style-type: none"> • Request parameters for SmsSLQSSetMaxStorageSize <ul style="list-style-type: none"> – See smsMaxStorageSizeReq for more information
<i>pMaxStorageSizeResp</i> [OUT]	<ul style="list-style-type: none"> • Response parameters for SmsSLQSSetMaxStorageSize <ul style="list-style-type: none"> – See smsMaxStorageSizeResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Secs

9.24.4.24 ULONG SLQSSmsGetMessageProtocol (smsMsgprotocolResp * pMessageProtocol)

This API queries the message protocol currently in use for the WMS client.

Parameters

<i>pMessageProtocol</i>	[OUT] <ul style="list-style-type: none">• Pointer to smsMsgprotocolResp<ul style="list-style-type: none">– See smsMsgprotocolResp for more information
-------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Secs

9.24.4.25 ULONG SLQSSmsSetRoutes (smsSetRoutesReq * pSetRoutesReq)

This API sets the action performed on SMS message receipt for specified message routes. It also specifies the action performed on SMS receipt of status reports.

Parameters

<i>pSetRoutesReq</i>	[IN] <ul style="list-style-type: none">• Pointer to structure of smsSetRoutesReq<ul style="list-style-type: none">– See smsSetRoutesReq for more information
----------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 Secs

9.24.4.26 ULONG SLQSSwiGetSMSStorage (ULONG * pSmsStorage)

This API queries the device to return current SMS configuration that is applied to all incoming and outgoing messages.

Parameters

<i>pSmsStorage</i> <i>OUT</i>	<ul style="list-style-type: none"> Values: <ul style="list-style-type: none"> 0x01 - device's permanent memory 0x02 - UICC
----------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Secs

9.24.4.27 ULONG SLQSWCDMADecodeLongTextMsg (struct wcdmaLongMsgDecodingParams * pWcdmaLongMsgDecodingParams)

Decodes WCDMA Long SMS PDU message, returns structure filled with decoded parameters

Parameters

<i>pwcdmaMsgDecodingParams</i> <i>IN/OUT</i>	<ul style="list-style-type: none"> Pointer to parameters required for decoding
---	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
Timeout: None

9.25 qaGobiApiSwi.h File Reference

SWI API function prototypes.

Functions

- [ULONG SLQSGetSdkVersion](#) ([CHAR](#) **sdkversionpp)
- [ULONG SLQSSendRawQMI](#) ([BYTE](#) *pReqBuf, [USHORT](#) service, [USHORT](#) length, [ULONG](#) timeout, [BYTE](#) **ppInParm, [USHORT](#) *pParamLength)
- [int SLQSGetPidof](#) ([CHAR](#) *pProcName)

9.25.1 Detailed Description

SWI API function prototypes.

9.25.2 Function Documentation

9.25.2.1 int SLQSGetPidof ([CHAR](#) * *pProcName*)

Internal Wrapper function for enabling invocation of SLQS implementation pidof() function

Parameters

<i>pProcName</i> [IN]	<ul style="list-style-type: none">• Process name whose PID is to be retrieved
-----------------------	---

Returns

pid if process exists else 0

See also

NA

Note

NA

9.25.2.2 ULONG SLQSSetSdkVersion (CHAR ** sdkversionpp)

Returns the SDK version string

Parameters

<i>ppString[OUT]</i>	<ul style="list-style-type: none">• pointer to pointer of NULL terminated string
----------------------	--

Returns

eQCWWAN_ERR_NONE success eQCWWAN_ERR_INVALID_ARG provided pointer is NULL

Note

Technology Supported: N/A Timeout: 2 seconds

9.25.2.3 ULONG SLQSSendRawQMI (BYTE * pReqBuf, USHORT service, USHORT length, ULONG timeout, BYTE ** ppInParm, USHORT * pParamLength)

9.26 qaGobiApiSwiAudio.h File Reference

M2M Audio Service API function prototypes.

Data Structures

- struct [GetM2MAudioProfileReq](#)
- struct [GetM2MAudioProfileResp](#)
- struct [SetM2MAudioProfileReq](#)
- struct [GetM2MAudioVolumeReq](#)
- struct [GetM2MAudioVolumeResp](#)
- struct [SetM2MAudioVolumeReq](#)
- struct [PCMparams](#)
- struct [SetM2MAudioAVCFGReq](#)
- struct [SetM2MAudioLPBKReq](#)
- struct [GetM2MSpkrGainReq](#)
- struct [GetM2MSpkrGainResp](#)
- struct [SetM2MSpkrGainReq](#)
- struct [GetM2MAVMuteReq](#)
- struct [GetM2MAVMuteResp](#)
- struct [SetM2MAVMuteReq](#)

Macros

- `#define MAX_LEN_IFACE_TABLE 255`

Functions

- `ULONG SLQSGetM2MAudioProfile (GetM2MAudioProfileReq *pGetM2MAudioProfileReq, GetM2MAudioProfileResp *pGetM2MAudioProfileResp)`
- `ULONG SLQSSetM2MAudioProfile (SetM2MAudioProfileReq *pSetM2MAudioProfileReq)`
- `ULONG SLQSGetM2MAudioVolume (GetM2MAudioVolumeReq *pGetM2MAudioVolumeReq, GetM2MAudioVolumeResp *pGetM2MAudioVolumeResp)`
- `ULONG SLQSSetM2MAudioVolume (SetM2MAudioVolumeReq *pSetM2MAudioVolumeReq)`
- `ULONG SLQSSetM2MAudioAVCFG (SetM2MAudioAVCFGReq *pSetM2MAudioAVCFGReq)`
- `ULONG SLQSSetM2MAudioLPBK (SetM2MAudioLPBKReq *pSetM2MAudioLPBKReq)`
- `ULONG SLQSSetM2MAudioNVDef ()`
- `ULONG SLQSGetM2MSpkrGain (GetM2MSpkrGainReq *pSpkrGainReq, GetM2MSpkrGainResp *pSpkrGainResp)`
- `ULONG SLQSSetM2MSpkrGain (SetM2MSpkrGainReq *pSpkrGainReq)`
- `ULONG SLQSGetM2MAVMute (GetM2MAVMuteReq *pGetM2MAVMuteReq, GetM2MAVMuteResp *pGetM2MAVMuteResp)`
- `ULONG SLQSSetM2MAVMute (SetM2MAVMuteReq *pSetM2MAVMuteReq)`

9.26.1 Detailed Description

M2M Audio Service API function prototypes.

9.26.2 Macro Definition Documentation

9.26.2.1 `#define MAX_LEN_IFACE_TABLE 255`

9.26.3 Function Documentation

9.26.3.1 `ULONG SLQSGetM2MAudioProfile (GetM2MAudioProfileReq * pGetM2MAudioProfileReq, GetM2MAudioProfileResp * pGetM2MAudioProfileResp)`

This API gets the profile content.

Parameters

<i>pGetM2MAudioProfileReq</i> [OUT]	<ul style="list-style-type: none"> • See GetM2MAudioProfileReq for more information
<i>pGetM2MAudioProfileResp</i> [OUT]	<ul style="list-style-type: none"> • See GetM2MAudioProfileResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.26.3.2 ULONG SLQSGetM2MAudioVolume (GetM2MAudioVolumeReq * pGetM2MAudioVolumeReq, GetM2MAudioVolumeResp * pGetM2MAudioVolumeResp)

This API gets the Volume content.

Parameters

<i>pGetM2MAudioVolumeReq[IN]</i>	<ul style="list-style-type: none"> See GetM2MAudioVolumeReq for more information
<i>pGetM2MAudioVolumeResp[OUT]</i>	<ul style="list-style-type: none"> See GetM2MAudioVolumeResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.26.3.3 ULONG SLQSGetM2MAVMute (GetM2MAVMuteReq * pGetM2MAVMuteReq, GetM2MAVMuteResp * pGetM2MAVMuteResp)

This API Gets the AV Mute content.

Parameters

<i>pGetM2MAVMuteReq[IN]</i>	<ul style="list-style-type: none"> See GetM2MAVMuteReq for more information
<i>pGetM2MAVMuteResp[OUT]</i>	<ul style="list-style-type: none"> See GetM2MAVMuteResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.26.3.4 ULONG SLQSGetM2MSpkrGain (GetM2MSpkrGainReq * pSpkrGainReq, GetM2MSpkrGainResp * pSpkrGainResp)

This API Gets the SPKRGAIN content.

Parameters

<i>pSpkrGainReq</i> [IN]	<ul style="list-style-type: none">See GetM2MSpkrGainReq for more information
<i>pSpkrGainResp</i> [OUT]	<ul style="list-style-type: none">See GetM2MSpkrGainResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.26.3.5 ULONG SLQSSetM2MAudioAVCFG (SetM2MAudioAVCFGReq * pSetM2MAudioAVCFGReq)

This API sets the AVCFG content.

Parameters

<i>pSetM2MAudioAVCFGReq</i> [IN]	<ul style="list-style-type: none">See SetM2MAudioAVCFGReq for more information
----------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.26.3.6 ULONG SLQSSetM2MAudioLPBK (SetM2MAudioLPBKReq * pSetM2MAudioLPBKReq)

This API sets the LPBK content.

Parameters

<i>pSetM2MAudioLPBKReq</i> <i>Req[IN]</i>	<ul style="list-style-type: none">• See SetM2MAudioLPBKReq for more information
--	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.26.3.7 ULONG SLQSSetM2MAudioNVDef ()

This API sets the NVDef content.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.26.3.8 ULONG SLQSSetM2MAudioProfile (SetM2MAudioProfileReq * pSetM2MAudioProfileReq)

This API sets the profile content.

Parameters

<i>pSetM2MAudioProfileReq</i> [IN]	<ul style="list-style-type: none">See SetM2MAudioProfileReq for more information
------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.26.3.9 ULONG SLQSSetM2MAudioVolume (SetM2MAudioVolumeReq * pSetM2MAudioVolumeReq)

This API sets the Volume content.

Parameters

<i>pSetM2MAudioVolumeReq</i> [IN]	<ul style="list-style-type: none">See SetM2MAudioVolumeReq for more information
-----------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.26.3.10 ULONG SLQSSetM2MAVMute (SetM2MAVMuteReq * pSetM2MAVMuteReq)

This API Sets the AV Mute content.

Parameters

<i>pSetM2MAVMuteReq</i> [IN]	<ul style="list-style-type: none"> See SetM2MAVMuteReq for more information
------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.26.3.11 ULONG SLQSSetM2MSpkrGain (SetM2MSpkrGainReq * pSpkrGainReq)

This API Sets the SPKRGAIN content.

Parameters

<i>pSpkrGainReq</i> [IN]	<ul style="list-style-type: none"> See GetM2MSpkrGainReq for more information
--------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.27 qaGobiApiSwiOmadms.h File Reference

SWI Open Mobile Alliance Device Management Service API function prototypes SWI OMA-DM QMI Service revision 1.6.

Data Structures

- struct [_SLQSOMADMSessionInfo](#)
- struct [_SLQSOMADMSettingsReqParams](#)
- struct [_SLQSOMADMSettings](#)
- struct [_SLQSOMADMSettingsReqParams3](#)

Typedefs

- typedef struct [_SLQSOMADMSessionInfo](#) [SLQSOMADMSessionInfo](#)
- typedef struct [_SLQSOMADMSettingsReqParams](#) [SLQSOMADMSettingsReqParams](#)
- typedef struct [_SLQSOMADMSettings](#) [SLQSOMADMSettings](#)
- typedef struct [_SLQSOMADMSettingsReqParams3](#) [SLQSOMADMSettingsReqParams3](#)

Functions

- [ULONG SLQSOMADMStartSession](#) ([ULONG](#) sessionType)
- [ULONG SLQSOMADMCancelSession](#) ([ULONG](#) session)
- [ULONG SLQSOMADMGetSessionInfo](#) ([ULONG](#) *pSessionType, [SLQSOMADMSessionInfo](#) *pResp)
- [ULONG SLQSOMADMSendSelection](#) ([ULONG](#) selection)
- [ULONG SLQSOMADMGetSettings](#) ([ULONG](#) *pbOMADMAEnabled, [ULONG](#) *pbFOTAdownload, [ULONG](#) *pbFOTAUpdate)
- [ULONG SLQSOMADMSetSettings](#) ([ULONG](#) bFOTAdownload, [ULONG](#) bFOTAUpdate)
- [ULONG SLQSOMADMSetSettings2](#) ([SLQSOMADMSettingsReqParams](#) *pSLQSOMADMSettingsReqParams)
- [ULONG SLQSOMADMGetSettings2](#) ([SLQSOMADMSettings](#) *pSLQSOMADMSettings)
- [ULONG SLQSOMADMStartSession2](#) ([ULONG](#) sessionType, [ULONG](#) *pFwAvailability)
- [ULONG SLQSOMADMSendSelection2](#) ([ULONG](#) selection, [ULONG](#) *pDeferTime, [ULONG](#) *pRejectReason)
- [ULONG SLQSOMADMSetSettings3](#) ([SLQSOMADMSettingsReqParams3](#) *pSLQSOMADMSettingsReqParams3)

9.27.1 Detailed Description

SWI Open Mobile Alliance Device Management Service API function prototypes SWI OMA-DM QMI Service revision 1.6.

9.27.2 Typedef Documentation

9.27.2.1 typedef struct [_SLQSOMADMSessionInfo](#) [SLQSOMADMSessionInfo](#)

Structure containing the OMA DM Session Info returned by the device. Also used as input parameter to specify the size of variable parameters. (ref. notes)

Parameters

<i>pStatus</i>	<ul style="list-style-type: none"> • 1 Byte parameter indicating status(optional) <ul style="list-style-type: none"> – 0x01 - No Firmware available – 0x02 - Query Firmware Download – 0x03 - Firmware Downloading – 0x04 - Firmware Downloaded – 0x05 - Query Firmware Update – 0x06 - Firmware Updating – 0x07 - Firmware Updated
<i>pUpdate↔</i> <i>CompleteStatus</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Update Complete Status(optional) <ul style="list-style-type: none"> – See qaGobiApiTableSwiOMADMUpdateCompleteStatus.h Update Complete Status
<i>pSeverity</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating severity(optional) <ul style="list-style-type: none"> – 0x01 - Mandatory – 0x02 - Optional
<i>pSourceLength</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Vendor Name String in Bytes.(optional)
<i>pSource</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Vendor Name in ASCII(optional)
<i>pPkgName↔</i> <i>Length</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Package Name String in Bytes.(optional)
<i>pPkgName</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Package Name in ASCII(optional)
<i>pPkgDesc↔</i> <i>Length</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Package Description String in Bytes.(optional)
<i>pPkgDescription</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Package Description in ASCII(optional)
<i>pDateLength</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Package Description String in Bytes.(optional)
<i>pDate</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Package Description in ASCII
<i>pTimeLength</i>	<ul style="list-style-type: none"> • 2 byte parameter indicating Length of Time String in Bytes.(optional)
<i>pTime</i>	<ul style="list-style-type: none"> • Variable length parameter indicating Time String in ASCII(optional)

<i>pSessionType</i>	<ul style="list-style-type: none"> • 1 byte parameter reflects the last session started for Sprint(optional) <ul style="list-style-type: none"> – 0x00 - No session since boot – 0x01 - Sprint CI-DC Session – 0x02 - Sprint CI-PRL Session – 0x03 - Sprint CI-FUMO Session – 0x04 - Sprint HFA-DC Session – 0x05 - Sprint HFA-PRL Session – 0x06 - Sprint HFA-FUMO Session – 0x07 - Sprint NI Session
<i>pSessionState</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating session state(optional) <ul style="list-style-type: none"> – 0x01 - idle – 0x02 - active – 0x03 - pending
<i>pRetryCount</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating retries left count(optional) <ul style="list-style-type: none"> – valid values 0 to 6

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

As input parameter the members pSourceLength, pPkgNameLength pPkgDescLength, pDateLength, pTimeLength have to be specified. These should contain the initialized size of pSource, pPkgName, pPkgDescription, pDate, pTime respectively.

9.27.2.2 typedef struct _SLQSOMADMSettings SLQSOMADMSettings

Structure containing the OMA DM settings retrieved from the device

Parameters

<i>pOMADMEnabled[OUT]</i>	<ul style="list-style-type: none"> • 4 byte parameter indicating OMADM service enabled <ul style="list-style-type: none"> – 0x00000001 - Client-initiated device configuration – 0x00000002 - Network-initiated device configuration – 0x00000010 - Client-initiated FUMO – 0x00000020 - Network-initiated FUMO • function SLQSOMADMGetSettings2() returns a default value 0xFFFFFFFF in case this parameter is not returned by the modem.
Generated by Doxygen	

<p><i>pFOTA↔</i> <i>Adownload[O↔</i> <i>UT]</i></p>	<ul style="list-style-type: none"> • 1 Byte parameter indicating support for FOTA Automatic download <ul style="list-style-type: none"> – 0x00 - Host permission required before downloading – 0x01 - Automatically start downloading, no host permission required – 0x02 - Automatically start downloading, while not roaming – 0x03 - Automatically reject download – 0x04 - Automatically reject download with “Enterprise Reject Policy” • function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.
<p><i>pFOTA↔</i> <i>Update[OUT]</i></p>	<ul style="list-style-type: none"> • 1 byte parameter indicating FOTA Automatic update <ul style="list-style-type: none"> – 0x00 - User permission required before updating firmware – 0x01 - No user permission required before updating firmware – 0x02 - User permission required, auto update on power up • function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.
<p><i>pAutosdm[OUT]</i></p>	<ul style="list-style-type: none"> • 1 byte parameter indicating OMA Automatic UI Alert Response <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled Accept – 0x02 - Enabled Reject • function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.
<p><i>pFwAuto↔</i> <i>Check[OUT]</i></p>	<ul style="list-style-type: none"> • Optional 1 byte parameter indicating OMA Automatic Check for Firmware Update on Power-Up Response <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled • function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.27.2.3 typedef struct _SLQSOMADMSettingsReqParams SLQSOMADMSettingsReqParams

Structure containing the OMA DM settings to be set on the device

Parameters

<i>FOTAdownload</i>	<ul style="list-style-type: none"> • 1 Byte parameter indicating support for FOTA Automatic download <ul style="list-style-type: none"> – 0x00 - Firmware auto download FALSE – 0x01 - Firmware auto download TRUE
<i>FOTAUpdate</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating FOTA Automatic update <ul style="list-style-type: none"> – 0x00 - Firmware auto update FALSE – 0x01 - Firmware auto update TRUE
<i>pAutosdm[IN]</i>	<ul style="list-style-type: none"> • Optional 1 byte parameter indicating OMA Automatic UI Alert Response <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled Accept – 0x02 - Enabled Reject

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.27.2.4 typedef struct _SLQSOMADMSettingsReqParams3 SLQSOMADMSettingsReqParams3

Structure containing the OMA DM settings to be set on the device

Parameters

<i>FOTAdownload</i>	<ul style="list-style-type: none"> • 1 Byte parameter indicating support for FOTA Automatic download <ul style="list-style-type: none"> – 0x00 - Firmware auto download FALSE – 0x01 - Firmware auto download TRUE
<i>FOTAUpdate</i>	<ul style="list-style-type: none"> • 1 byte parameter indicating FOTA Automatic update <ul style="list-style-type: none"> – 0x00 - Firmware auto update FALSE – 0x01 - Firmware auto update TRUE
<i>pAutosdm[IN]</i>	<ul style="list-style-type: none"> • Optional 1 byte parameter indicating OMA Automatic UI Alert Response <ul style="list-style-type: none"> – 0x00 - Disabled – 0x01 - Enabled Accept – 0x02 - Enabled Reject
<i>pFwAuto↔ Check[IN]</i>	<ul style="list-style-type: none"> • Optional 1 byte parameter indicating OMA Automatic Check for Firmware Update on Power-Up Response <ul style="list-style-type: none"> – 0x00 - Disabled
Generated by Doxygen	<ul style="list-style-type: none"> – 0x01 - Enabled

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.27.3 Function Documentation**9.27.3.1 ULONG SLQSOMADMCancelSession (ULONG session)**

Cancels an ongoing OMA-DM session.

Parameters

<i>session</i> [IN]	<ul style="list-style-type: none"> • Session <ul style="list-style-type: none"> – 0x01 - FOTA, to check availability of FW Update – 0xFF - Cancel any active OMADM session
---------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.27.3.2 ULONG SLQSOMADMGetSessionInfo (ULONG * pSessionType, SLQSOMADMSessionInfo * pResp)

Returns information related to the current (or previous if no session is active) OMA-DM session.

Parameters

<i>SessionType</i> [IN]	<ul style="list-style-type: none"> • Session type <ul style="list-style-type: none"> – 0x01 - FOTA – 0xFF - Any active OMADM session. If none active, then previous OMADM session
<i>pResp</i> [IN/OUT]	<ul style="list-style-type: none"> • See SLQSOMADMSessionInfo for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.27.3.3 ULONG SLQSOMADMGetSettings (ULONG * pbOMADMEEnabled, ULONG * pbFOTAdownload, ULONG * pbFOTAUpdate)

Returns the OMA-DM settings.

Parameters

<i>pbOMADM</i> ↔ <i>Enabled[OUT]</i>	<ul style="list-style-type: none"> • Device OMADM service enabled <ul style="list-style-type: none"> – 0x00000001 - Client-initiated device configuration – 0x00000002 - Network-initiated device configuration – 0x00000010 - Client-initiated FUMO – 0x00000020 - Network-initiated FUMO
<i>pbFOTA</i> ↔ <i>Adownload[OUT]</i>	<ul style="list-style-type: none"> • Firmware AutoDownload <ul style="list-style-type: none"> – 0x00 - Firmware auto download FALSE – 0x01 - Firmware autod ownload TRUE – 0x02 - Automatically start downloading, while not roaming – 0x03 - Automatically reject download – 0x04 - Automatically reject download with “Enterprise Reject Policy”
<i>pbFOTA</i> ↔ <i>Update[OUT]</i>	<ul style="list-style-type: none"> • Firmware AutoUpdate <ul style="list-style-type: none"> – 0x00 - Firmware auto update FALSE – 0x01 - Firmware auto update TRUE

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.27.3.4 ULONG SLQSOMADMGetSettings2 (SLQSOMADMSettings * pSLQSOMADMSettings)

Retrieves the OMA-DM settings from the device.

Parameters

<i>SLQSOMADMSettingsReqParams</i>	<ul style="list-style-type: none">• See SLQSOMADMSettings for more information
-----------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.27.3.5 ULONG SLQSOMADMSelectSelection (ULONG selection)

Sends the specified OMA-DM selection for the current network initiated session.

Parameters

<i>selection[IN]</i>	<ul style="list-style-type: none">• OMA-DM NIA Selection<ul style="list-style-type: none">– 0x01 - Accept– 0x02 - Reject– 0x03 - Defer
----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.27.3.6 ULONG SLQSOMADMSendSelection2 (ULONG *selection*, ULONG * *pDeferTime*, ULONG * *pRejectReason*)

Sends the specified OMA-DM selection for the current network initiated session.

Parameters

<i>selection</i> [IN]	<ul style="list-style-type: none"> OMA-DM NIA Selection <ul style="list-style-type: none"> 0x01 - Accept 0x02 - Reject 0x03 - Defer
<i>pDeferTime</i> [IN]	<ul style="list-style-type: none"> Defer time in minutes. A value of 0 will cause the prompt to be resent immediately. This TLV is mandatory if selection is set to 0x03.
<i>pRejectReason</i> [IN]	<ul style="list-style-type: none"> Reject Reason This TLV is processed if selection is set to 0x02. If it is not present, the reject reason 0 is used as default.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.27.3.7 ULONG SLQSOMADMSetSettings (ULONG *bFOTAdownload*, ULONG *bFOTAupdate*)

Sets the OMA-DM settings requested.

Parameters

<i>bFOTAdownload</i> [IN]	<ul style="list-style-type: none"> Firmware Auto Download <ul style="list-style-type: none"> 0x00 - Host permission required before downloading 0x01 - Automatically start downloading, no host permission required 0x02 - Automatically start downloading, while not roaming 0x03 - Automatically reject download 0x04 - Automatically reject download with "Enterprise Reject Policy"
<i>bFOTAupdate</i> [IN]	<ul style="list-style-type: none"> Firmware Auto Update <ul style="list-style-type: none"> 0x00 - User permission required before updating firmware 0x01 - No user permission required before updating firmware 0x02 - User permission required, auto update on power up
Generated by Doxygen	

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.27.3.8 **ULONG** SLQSOMADMSetSettings2 (**SLQSOMADMSettingsReqParams** * *pSLQSOMADMSettingsReqParams*)

Sets the settings related to OMADM. These settings are saved on the modem across power cycles.

Parameters

<i>pSLQSOMADMSettingsReqParams</i> [IN]	<ul style="list-style-type: none"> See SLQSOMADMSettingsReqParams for more information
---	---

Note

Timeout: 20 seconds

9.27.3.9 **ULONG** SLQSOMADMSetSettings3 (**SLQSOMADMSettingsReqParams3** * *pSLQSOMADMSettingsReqParams3*)

Sets the settings related to OMADM. These settings are saved on the modem across power cycles.

Parameters

<i>SLQSOMADMSettingsReqParamsExt</i> [IN]	<ul style="list-style-type: none"> See SLQSOMADMSettingsReqParamsExt for more information
---	--

Note

Timeout: 20 seconds

9.27.3.10 **ULONG** SLQSOMADMStartSession (**ULONG** *sessionType*)

Starts an OMA-DM session.

Parameters

<i>sessionType</i> [IN]	<ul style="list-style-type: none"> • Session type <ul style="list-style-type: none"> – 0x01 - FOTA, to check availability of FW Update – 0x02 - DM, to check availability of DM Update – 0x03 - PRL, to check availability of PRL Update
-------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.27.3.11 ULONG SLQSOMADMStartSession2 (ULONG sessionType, ULONG * pFwAvailability)

Starts an OMA-DM session.

Parameters

<i>sessionType</i> [IN]	<ul style="list-style-type: none"> • Session type <ul style="list-style-type: none"> – 0x01 - FOTA, to check availability of FW Update – 0x02 - DM, to check availability of DM Update – 0x03 - PRL, to check availability of PRL Update
<i>pFwAvailability</i> [OUT]	<ul style="list-style-type: none"> • OMA-DM CHECK FW Available <ul style="list-style-type: none"> – 0x00000001 - FW Available. For CIDC and CIPRL, this value will be returned by the modem. CIDC and CIPRL are asynchronous OMADM sessions. – 0x00000002 - FW Not Available – 0x00000003 - FW Check Timed Out

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.28 qaGobiApiTableBandClasses.h File Reference

Network Access Service API Band Classes table.

9.28.1 Detailed Description

Network Access Service API Band Classes table.

9.28.2 Band Classes (Value - Description)

- 0 - CDMA Band Class 0
- 1 - CDMA Band Class 1
- 3 - CDMA Band Class 3
- 4 - CDMA Band Class 4
- 5 - CDMA Band Class 5
- 6 - CDMA Band Class 6
- 7 - CDMA Band Class 7
- 8 - CDMA Band Class 8
- 9 - CDMA Band Class 9
- 10 - CDMA Band Class 10
- 11 - CDMA Band Class 11
- 12 - CDMA Band Class 12
- 13 - CDMA Band Class 13
- 14 - CDMA Band Class 14
- 15 - CDMA Band Class 15
- 16 - CDMA Band Class 16
- 17 - CDMA Band Class 17
- 18 - CDMA Band Class 18
- 19 - CDMA Band Class 19
- 40 - GSM 450
- 41 - GSM 480
- 42 - GSM 750
- 43 - GSM 850
- 44 - GSM 900 (Extended)
- 45 - GSM 900 (Primary)
- 46 - GSM 900 (Railways)
- 47 - GSM 1800

- 48 - GSM 1900
- 80 - WCDMA 2100
- 81 - WCDMA PCS 1900
- 82 - WCDMA DCS 1800
- 83 - WCDMA 1700 (US)
- 84 - WCDMA 850
- 85 - WCDMA 800
- 86 - WCDMA 2600
- 87 - WCDMA 900
- 88 - WCDMA 1700 (Japan)
- 90 - WCDMA 1500 band (Japan)
- 91 - WCDMA 850 band (Japan)
- < Reserved 89, 92-109 for WCDMA band classes >
- 110 - WLAN US 2400 MHz
- 111 - WLAN JAPAN 2400 MHz
- 112 - WLAN EUROPEAN 2400 MHz
- 113 - WLAN FRANCE 2400 MHz
- 114 - WLAN SPAIN 2400 MHz
- 115 - WLAN US 5000 MHz band
- 116 - WLAN JAPAN 5000 MHz
- 117 - WLAN EUROPEAN 5000 MHz
- 118 - WLAN FRANCE 5000 MHz
- 119 - WLAN SPAIN 5000 MHz

9.28.2.1 LTE Bands

- 28 - LTE Band Class 28
- 39 - LTE Band Class 39
- 40 - LTE Band Class 40
- 41 - LTE Band Class 41
- 120 - FDD UL:1920-1980; DL:2110-2170; E-UTRA Operating Band 1
- 121 - FDD UL:1850-1910; DL:1930-1990; E-UTRA Operating Band 2
- 122 - FDD UL:1710-1785; DL:1805-1880; E-UTRA Operating Band 3
- 123 - FDD UL:1710-1755; DL:2110-2155; E-UTRA Operating Band 4
- 124 - FDD UL: 824- 849; DL: 869- 894; E-UTRA Operating Band 5
- 125 - FDD UL: 830- 840; DL: 875- 885; E-UTRA Operating Band 6

- 126 - FDD UL:2500-2570; DL:2620-2690; E-UTRA Operating Band 7
- 127 - FDD UL: 880- 915; DL: 925- 960; E-UTRA Operating Band 8
- 128 - FDD UL:1749.9-1784.9; DL:1844.9-1879.9; E-UTRA Operating Band 9
- 129 - FDD UL:1710-1770; DL:2110-2170; E-UTRA Operating Band 10
- 130 - FDD UL:1427.9-1452.9; DL:1475.9-1500.9; E-UTRA Operating Band 11
- 131 - FDD UL:698-716; DL:728-746; E-UTRA Operating Band 12
- 132 - FDD UL: 777- 787; DL: 746-756; E-UTRA Operating Band 13
- 133 - FDD UL: 788- 798; DL: 758-768; E-UTRA Operating Band 14
- 134 - FDD UL: 704-716; DL: 734-746; E-UTRA Operating Band 17
- 135 - TDD LTE UL: 1900-1920; DL: 1900-1920; E-UTRA Operating Band 33
- 136 - TDD LTE UL: 2010-2025; DL: 2010-2025; E-UTRA Operating Band 34
- 137 - TDD LTE UL: 1850-1910; DL: 1850-1910; E-UTRA Operating Band 35
- 138 - TDD LTE UL: 1930-1990; DL: 1930-1990; E-UTRA Operating Band 36
- 139 - TDD LTE UL: 1910-1930; DL: 1910-1930; E-UTRA Operating Band 37
- 140 - TDD LTE UL: 2570-2620; DL: 2570-2620; E-UTRA Operating Band 38
- 141 - TDD LTE UL: 1880-1920; DL: 1880-1920; E-UTRA Operating Band 39
- 142 - TDD LTE UL: 2300-2400; DL: 2300-2400; E-UTRA Operating Band 40
- 143 - FDD LTE UL: 815-830; DL: 860-875; E-UTRA Operating Band 18
- 144 - FDD LTE UL: 830-845; DL: 875-890; E-UTRA Operating Band 19
- 145 - FDD LTE UL: 832-862; DL: 791-821; E-UTRA Operating Band 20
- 146 - FDD LTE UL: 1447.9-1462.9; DL: 1495.9-1510.9; E-UTRA Operating Band 21
- 147 - FDD LTE UL: 1626.5-1660.5; DL: 1525-1559; E-UTRA Operating Band 24
- 148 - FDD LTE UL: 1850-1919.5; DL: 1930-1995; E-UTRA Operating Band 25
- 149 - TDD LTE UL: 2496-2690; DL: 2496-2690; E-UTRA Operating Band 41
- 150 - TDD LTE UL: 3400-3600; DL: 3400-3600; E-UTRA Operating Band 42
- 151 - TDD LTE UL: 3600-3800; DL: 3600-3800; E-UTRA Operating Band 43
- 200 - TD-SCDMA Band A
- 201 - TD-SCDMA Band B
- 202 - TD-SCDMA Band C
- 203 - TD-SCDMA Band D
- 204 - TD-SCDMA Band E
- 205 - TD-SCDMA Band F

9.29 qaGobiApiTableCallControlReturnReasons.h File Reference

Call Control Return Reasons table.

9.29.1 Detailed Description

Call Control Return Reasons table.

9.29.2 Coding Group Bits 7..4(0000)

- 0x01 - QMI_VOICE_REASON_FWD_UNCONDITIONAL - Unconditional call forwarding
- 0x02 - QMI_VOICE_REASON_FWD_MOBILEBUSY - Forward when the mobile is busy
- 0x03 - QMI_VOICE_REASON_FWD_NOREPLY - Forward when there is no reply
- 0x04 - QMI_VOICE_REASON_FWD_UNREACHABLE - Forward when the call is unreachable
- 0x05 - QMI_VOICE_REASON_FWD_ALLFORWARDING - All forwarding
- 0x06 - QMI_VOICE_REASON_FWD_ALLCONDITIONAL - All conditional forwarding
- 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING - All outgoing
- 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT - Outgoing internal
- 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTHOME - Outgoing external to home
- 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING - All incoming
- 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING - Roaming incoming
- 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING - All calls are barred
- 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING - All outgoing calls are barred
- 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING - All incoming calls are barred
- 0x0F - QMI_VOICE_REASON_CALLWAITING - Call waiting
- 0x10 - VOICE_CC_SUPS_RESULT_REASON_CLIP - CLIP
- 0x11 - VOICE_CC_SUPS_RESULT_REASON_CLIR - CLIR
- 0x12 - VOICE_CC_SUPS_RESULT_REASON_COLP - COLP
- 0x13 - VOICE_CC_SUPS_RESULT_REASON_COLR - COLR
- 0x14 - VOICE_CC_SUPS_RESULT_REASON_CNAP - CNAP
- 0xFF - Not Available

Copyright: © 2012 Sierra Wireless, Inc. all rights reserved

9.30 qaGobiApiTableCallEndReasons.h File Reference

Wireless Data Service Call End Reasons.

9.30.1 Detailed Description

Wireless Data Service Call End Reasons.

9.30.2 Call end reason codes (Code - Reason)

9.30.2.1 Technology-agnostic call end reasons

- 1 - Reason unspecified, check the verbose call end reason
- 2 - Client ended the call
- 3 - Device has no service
- 4 - Call ended abnormally
- 5 - Received release from base station; no reason given
- 6 - Access attempt already in progress; SD2.0 only
- 7 - Access failure for reason other than the above
- 8 - Call rejected because of redirection or handoff
- 9 - Call failed because close is in progress
- 10 - Authentication failed, 3GPP equivalent ESM(EPS Session Management) cause code value 29, User authentication failed
- 11 - Call ended because of internal call end. This error code is returned when data call is brought down due to some unknown error, such as not specific to any RAT
- 12 - Call ended because of internal error. This error code is returned when data call is brought down due to some unspecified internal error, such as NULL pointer
- 13 - Internal unknown cause code

9.30.2.2 CDMA

- 500 - Device is CDMA-locked until power cycle
- 501 - Received intercept from base station; origination only
- 502 - Received reorder from base station; origination only
- 503 - Received release from base station; service option reject
- 504 - Received incoming call from base station
- 505 - Received alert stop from base station; incoming only
- 506 - Received end activation; OTASP call only
- 507 - Max access probes transmitted
- 508 - Concurrent service is not supported by base station
- 509 - No response received from base station
- 510 - Call rejected by the base station; CDMA only
- 511 - Concurrent services requested were not compatible; CDMA only
- 512 - Corresponds to CM CALL ORIG ERR ALREADY IN TC
- 513 - Used if Call manager subsystem is ending a GPS call in favor of a user call
- 514 - Used if Call manager subsystem is ending a SMS call in favor of a user call
- 515 - CDMA Only; Device has no service

9.30.2.3 WCDMA/GSM call end reasons

- 1000 - Call origination request failed; WCDMA/GSM Only
- 1001 - Client rejected the incoming call; WCDMA/GSM Only
- 1002 - Device has no UMTS service; WCDMA/GSM Only
- 1003 - Network ended the call, look in cc call; WCDMA/GSM Only
- 1004 - LLC(Logical Link Control) or SMDCP(Sub Network Dependent Convergence Protocol) failure
- 1005 - Insufficient resources, 3GPP equivalent ESM(EPS Session Management) cause code value 26, Insufficient resources
- 1006 - Service option temporarily out of order, 3GPP equivalent ESM(EPS Session Management) cause code value 34, Service option temporarily out of order
- 1007 - PTI already used, 3GPP equivalent ESM(EPS Session Management) cause code value 35, PTI(← Procedure Transaction Identity) already in use
- 1008 - Regular PDP context deactivation, 3GPP equivalent ESM(EPS Session Management) cause code value 36, Regular deactivation
- 1009 - Network failure, 3GPP equivalent ESM(EPS Session Management) cause code value 38, Network failure
- 1010 - Reactivation requested, 3GPP equivalent ESM(EPS Session Management) cause code value 39, Reactivation requested
- 1011 - Protocol error, unspecified, 3GPP equivalent ESM(EPS Session Management) cause code value 111, Protocol error, unspecified
- 1012 - Operator determined barring, 3GPP equivalent ESM(EPS Session Management) cause code value 8, Operator Determined Barring
- 1013 - Unknown or missing Access Point Name (APN), 3GPP equivalent ESM(EPS Session Management) cause code value 27, Missing or unknown APN
- 1014 - Unknown PDP address or PDP type, 3GPP equivalent ESM(EPS Session Management) cause code value 28, Unknown PDN type
- 1015 - Activation rejected by GGSN, 3GPP equivalent ESM(EPS Session Management) cause code value 30, Requested rejected by Serving GW or PDN GW
- 1016 - Activation rejected, unspecified, 3GPP equivalent ESM(EPS Session Management) cause code value 31, Request rejected, unspecified
- 1017 - Service option not supported, 3GPP equivalent ESM(EPS Session Management) cause code value 32, Service option not supported
- 1018 - Requested service option not subscribed, 3GPP equivalent ESM(EPS Session Management) cause code value 33, Requested service option not subscribed
- 1019 - EPS Quality of Service (QoS) not accepted, 3GPP equivalent ESM(EPS Session Management) cause code value 37, EPS QoS not accepted
- 1020 - Semantic error in the TFT operation, 3GPP equivalent ESM(EPS Session Management) cause code value 41, Semantic error in the TFT operation
- 1021 - Syntactical error in the TFT operation, 3GPP equivalent ESM(EPS Session Management) cause code value 42, Syntactical error in the TFT operation
- 1022 - Unknown PDP context, 3GPP equivalent ESM(EPS Session Management) cause code value 43, Invalid EPS bearer identity

- 1023 - Semantic errors in packet filter(s), 3GPP equivalent ESM(EPS Session Management) cause code value 44, Semantic errors in packet filter(s)
- 1024 - Syntactical error in packet filter(s), 3GPP equivalent ESM(EPS Session Management) cause code value 45, Syntactical errors in packet filter(s)
- 1025 - PDP context without TFT already activated, 3GPP equivalent ESM(EPS Session Management) cause code value 46, Unused
- 1026 - Invalid transaction identifier value, 3GPP equivalent ESM(EPS Session Management) cause code value 81, Invalid PTI value
- 1027 - Semantically incorrect message, 3GPP equivalent ESM(EPS Session Management) cause code value 95, Semantically incorrect message
- 1028 - Invalid mandatory information, 3GPP equivalent ESM(EPS Session Management) cause code value 96, Invalid mandatory information
- 1029 - Message type non-existent or not implemented, 3GPP equivalent ESM(EPS Session Management) cause code value 97, Message type non-existent or not implemented
- 1030 - Message not compatible with state, 3GPP equivalent ESM(EPS Session Management) cause code value 98, Message type not compatible with the protocol state
- 1031 - Information element nonexistent or not implemented, 3GPP equivalent ESM(EPS Session Management) cause code value 99, Information element non-existent or not implemented
- 1032 - Conditional information element error, 3GPP equivalent ESM(EPS Session Management) cause code value 100, Conditional IE error
- 1033 - Message not compatible with protocol state, 3GPP equivalent ESM(EPS Session Management) cause code value 101, Message not compatible with the protocol state
- 1034 - APN restriction value incompatible with active PDP context, 3GPP equivalent ESM(EPS Session Management) cause code value 112, APN restriction value incompatible with
 - active EPS bearer context
- 1035 - No GPRS context present
- 1036 - Requested feature not supported, 3GPP equivalent ESM(EPS Session Management) cause code value 40, Feature not supported
- 1037 - Illegal MS, 3GPP equivalent EMM(EPS Mobility Management) cause code value 3, Illegal UE (MS)
- 1038 - Illegal ME, 3GPP equivalent EMM(EPS Mobility Management) cause code value 6, Illegal ME. This error code is sent to the MS if the ME used is not acceptable
 - to the network, e.g. blacklisted
- 1039 - GPRS and non GPRS services not allowed
- 1040 - GPRS services not allowed
- 1041 - MS identity not derived by the network, 3GPP equivalent EMM(EPS Mobility Management) cause code value 9, UE (MS) Identify cannot be derived by the network
- 1042 - Implicitly detached, 3GPP equivalent EMM(EPS Mobility Management) cause code value 10, Implicitly Detached
- 1043 - PLMN not allowed, 3GPP equivalent EMM(EPS Mobility Management) cause code value 11, PLMN not allowed
- 1044 - LA not allowed, this cause is sent to the MS if it requests location updating in a location area where the HPLMN determines that the MS, by subscription, is not allowed to operate.
- 1045 - GPRS services not allowed in PLMN

- 1046 - PDP duplicate
- 1047 - UE radio access technology change
- 1048 - app preempted
- 1049 - Congestion, This cause is sent if the service request or LOCATION UPDATING REQUEST message cannot be actioned because of congestion (e.g. congestion of the MSC or SGSN or GGSN or PDN Gateway; no channel; facility busy/congested etc.).
- 1050 - No PDP context activated
- 1051 - Access class DSAC rejection

9.30.2.4 CDMA

- 1500 - Abort connection setup due to the reception of a Connection Deny message with deny code set to either general or network busy.
- 1501 - Abort connection setup due to the reception of a Connection Deny message with deny code set to either billing or authentication failure.
- 1502 - Change HDR system due to redirection or PRL not preferred
- 1503 - Exit HDR due to redirection or PRL not preferred
- 1504 - No HDR session
- 1505 - Used if Call manager is ending an HDR call origination in favor of a GPS fix
- 1506 - Connection setup timeout
- 1507 - Call manager released HDR call so 1x call can continue

9.30.2.5 call end reason type

- 1 - Mobile IP
- 2 - Internal
- 3 - Call Manager defined
- 6 - 3GPP specification defined
- 7 - PPP
- 8 - EHRPD
- 9 - IPv6

9.30.2.6 Mobile IP call end reasons (Type=1)

- 64 - MIP(Mobile IP) FA(Foreign Agent) ERR REASON UNSPECIFIED, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration with unspecified reason
- 65 - MIP(Mobile IP) FA(Foreign Agent) ERR ADMINISTRATIVELY PROHIBITED, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent administratively prohibited MIP registration
- 66 - MIP(Mobile IP) FA(Foreign Agent) ERR INSUFFICIENT RESOURCES, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to insufficient resources
- 67 - MIP(Mobile IP) FA(Foreign Agent) ERR MOBILE NODE AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because MN-AAA authenticator was wrong
- 68 - MIP(Mobile IP) FA(Foreign Agent) ERR HA AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because of home agent authentication failure
- 69 - MIP(Mobile IP) FA(Foreign Agent) ERR REQUESTED LIFETIME TOO LONG, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because requested lifetime is too long
- 70 - MIP(Mobile IP) FA(Foreign Agent) ERR MALFORMED REQUEST, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to malformed request
- 71 - MIP(Mobile IP) FA(Foreign Agent) ERR MALFORMED REPLY, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to malformed reply
- 72 - MIP(Mobile IP) FA(Foreign Agent) ERR ENCAPSULATION UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because requested encapsulation is unavailable
- 73 - MIP(Mobile IP) FA(Foreign Agent) ERR VJHC UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because VJ Header Compression is unavailable
- 74 - MIP(Mobile IP) FA(Foreign Agent) ERR REVERSE TUNNEL UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because reverse tunnel is unavailable
- 75 - MIP(Mobile IP) FA(Foreign Agent) ERR REVERSE TUNNEL IS MANDATORY AND T BIT NOT SET, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because reverse tunnel is mandatory but not requested by device
- 79 - MIP(Mobile IP) FA(Foreign Agent) ERR DELIVERY STYLE NOT SUPPORTED, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because delivery style is not supported
- 97 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING NAI, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing NAI
- 98 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING HA, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing Home Agent
- 99 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING HOME ADDR, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing Home Address

- 104 - MIP(Mobile IP) FA(Foreign Agent) ERR UNKNOWN CHALLENGE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to unknown challenge
- 105 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING CHALLENGE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing challenge
- 106 - MIP(Mobile IP) FA(Foreign Agent) ERR STALE CHALLENGE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to stale challenge
- 128 - MIP(Mobile IP) FA(Home Agent) ERR REASON UNSPECIFIED, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration with unspecified reason
- 129 - MIP(Mobile IP) FA(Home Agent) ERR ADMINISTRATIVELY PROHIBITED, this error code is returned when the data call bring up fails in MIP setup phase since home agent administratively prohibited MIP registration
- 130 - MIP(Mobile IP) FA(Home Agent) ERR INSUFFICIENT RESOURCES, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to insufficient resources
- 131 - MIP(Mobile IP) FA(Home Agent) ERR MOBILE NODE AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since home agent fails authentication because MN-HA authenticator was wrong
- 132 - MIP(Mobile IP) FA(Home Agent) ERR FA AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to FA authentication failure
- 133 - MIP(Mobile IP) FA(Home Agent) ERR REGISTRATION ID MISMATCH, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to registration id mismatch
- 134 - MIP(Mobile IP) FA(Home Agent) ERR MALFORMED REQUEST, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to malformed request
- 136 - MIP(Mobile IP) FA(Home Agent) ERR UNKNOWN HA ADDR, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to unknown Home Agent address. This code is returned by a home agent when the mobile node is performing dynamic home agent address resolution as described in RFC 3220 (IP Mobility Support for IPV4) Sections 3.6.1.1 and 3.6.1.2
- 137 - MIP(Mobile IP) FA(Home Agent) ERR REVERSE TUNNEL UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration because reverse tunnel is unavailable
- 138 - MIP(Mobile IP) FA(Home Agent) ERR REVERSE TUNNEL IS MANDATORY AND T BIT NOT SET, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration because reverse tunnel is mandatory but not requested by device
- 139 - MIP(Mobile IP) FA(Home Agent) ERR ENCAPSULATION UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to encapsulation unavailable
- 65536 - MIP ERR REASON UNKNOWN

9.30.2.7 Internal call end reasons (Type=2)

- 200 - INTERNAL MIN, internal error table offset value, no meaningful message to the error.

- 201 - INTERNAL ERROR, this error code is returned when data call is brought down due to some unspecified internal error
- 202 - CALL ENDED
- 203 - INTERNAL UNKNOWN CAUSE CODE, this error code is returned when data call is brought down due to some unknown error
- 204 - UNKNOWN CAUSE CODE, this error code is returned when data call is brought down due to some unknown error
- 205 - CLOSE IN PROGRESS
- 206 - NETWORK INITIATED TERMINATION
- 207 - APP PREEMPTED
- 208 - ERR PDN IPV4 CALL DISALLOWED, this error code is returned when V4 PDN is in throttled state due to network providing only V6 address during the previous VSNCP bring up (subs_limited_to_v6). The time for which the IPv4 PDN is throttled is determined by the IPv4 throttling timers maintained in the profile
- 209 - ERR PDN IPV4 CALL THROTTLED, this error code is returned when V4 PDN is in throttled state due to previous VSNCP bring up failure(s). The time for which the IPv4 PDN is throttled is determined by the IPv4 throttling timers maintained in the profile
- 210 - ERR PDN IPV6 CALL DISALLOWED, this error code is returned when V6 PDN is in throttled state due to network providing only V4 address during the previous VSNCP bring up (subs_limited_to_v4). The time for which the IPv6 PDN is throttled is determined by the IPv6 throttling timers maintained in the profile
- 211 - ERR PDN IPV6 CALL THROTTLED, this error code is returned when V6 PDN is in throttled state due to previous VSNCP bring up failure(s). The time for which the IPv6 PDN is throttled is determined by the IPv6 throttling timers maintained in the profile
- 212 - MODEM RESTART
- 213 - PDP PPP NOT SUPPORTED
- 214 - UNPREFERRED RAT, this error code is returned when data call is brought down since the RAT on which the data call is attempted/connected is no longer the preferred RAT
- 215 - PHYS LINK CLOSE IN PROGRESS, this error code is returned when data call bring up is rejected because physical link is in the process of cleanup
- 216 - APN PENDING HANDOVER, this error code is returned when interface bring up is attempted for an APN that is yet to be handed over to target RAT
- 217 - PROFILE BEARER INCOMPATIBLE
- 218 - MMGSDI CARD EVT, this error code is returned when data call is brought down because card got refreshed/removed
- 219 - LPM OR PWR DOWN, this error code is returned when data call is brought down because device is going into lower power mode or powering down
- 220 - APN DISABLED, this error code is returned when APN is disabled in card
- 221 - MPIT EXPIRED, this error code is returned when data call is brought down because maximum PPP inactivity timer expired
- 222 - IPV6 ADDR TRANSFER FAILED
- 223 - TRAT SWAP FAILED
- 224 - EHRPD TO HRPD FALLBACK, this error code is returned when data call is brought down because device falls back from eHRPD to HRPD (not because of OOS on eHRPD but due to operator/spec driven eHRPD to HRPD fallback requirements)

- 225 - MANDATORY APN DISABLED, this error code is returned when any mandatory APN is disabled, and MinApnList Disallow call config item is set to TRUE in device
- 226 - MIP CONFIG FAILURE, this error code is returned when UE is in MIP Only config (QCMIP=2) but MIP config fails on call bring up due to incorrect provisioning

9.30.2.8 Call Manager defined call end reasons (Type=3)

- 500 - CDMA LOCK, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to device in CDMA locked state
- 501 - INTERCEPT, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since it received an intercept order from the base station
- 502 - REORDER, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due to receiving a reorder from base station
- 503 - REL SO REJ, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to receiving a release from base station with reason: SO Reject
- 504 - INCOM CALL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since it received an incoming call from base station
- 505 - ALERT STOP, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to RL/FL fade (or) receiving call release from base stations
- 506 - ACTIVATION, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to channel acquisition failures. This indicates that device has failed acquiring all the channels in the PRL
- 507 - MAX ACCESS PROBE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due maximum access probes transmitted
- 508 - CCS NOT SUPPORTED BY BS, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since concurrent service is not supported by base station
- 509 - NO RESPONSE FROM BS, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since there is no response received from base station
- 510 - REJECTED BY BS, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due to base station rejecting the call
- 511 - INCOMPATIBLE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since concurrent services requested were not compatible
- 512 - ALREADY IN TC, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since traffic channel is already up for voice calls
- 513 - USER CAL ORIG DURING GPS
- 514 - USER CAL ORIG DURING SMS, this error code is returned when data call is brought down because traffic channel request got rejected since SMS is ongoing
- 515 - NO CDMA SRV, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device does not have CDMA service
- 516 - MC ABORT, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since MC aborted the origination/conversation
- 517 - PSIST NG, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to persistence test failure

- 518 - UIM NOT PRESENT, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to RUIM not present
- 519 - RETRY ORDER, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due to receiving a retry order from base station
- 520 - ACCESS BLOCK, this error code is returned when data call is brought down because traffic channel rejected/released due to Access blocked by base station
- 521 - ACCESS BLOCK ALL, this error code is returned when data call is brought down because traffic channel rejected due to Access blocked by the base station for all mobile devices
- 522 - IS707B MAX ACC, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due maximum access probes for IS-707B call
- 523 - THERMAL EMERGENCY, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) to put device in thermal emergency
- 524 - CALL ORIG THROTTLED, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since call origination is throttled by DCTM module
- 525 - USER CALL ORIGINATE DURING VOICE CALL, this error code is returned when data call is brought down because traffic channel got released by CM(Call Manager) in favor of voice call or SMS when concurrent voice and data are not supported
- 1000 - CONFERENCE FAILED
- 1001 - INCOMING REJECTED
- 1002 - NO GATEWAY SERVICE
- 1003 - NO GPRS CONTEXT
- 1004 - ILLEGAL MS, This cause is sent to the MS when the network refuses service to the MS either because an identity of the MS is not acceptable to the network or because the MS does not pass the authentication check, i.e. the SRES received from the MS is different from that generated by the network
- 1005 - ILLEGAL ME, This cause is sent to the MS if the ME used is not acceptable to the network, e.g. blacklisted
- 1006 - GPRS SERVICES AND NON GPRS SERVICES NOT ALLOWED
- 1007 - GPRS SERVICES NOT ALLOWED
- 1008 - MS IDENTITY CANNOT BE DERIVED BY THE NETWORK
- 1009 - IMPLICITLY DETACHED, this error code is sent to the MS either if the network has implicitly detached the MS, e.g. some while after the Mobile reachable timer has expired, or if the GMM context data related to the subscription does not exist in the SGSN e.g. because of a SGSN restart.
- 1010 - PLMN NOT ALLOWED, this error code is sent to the MS if it requests location updating in a PLMN where the MS, by subscription or due to operator determined barring is not allowed to operate
- 1011 - LOCAL AREA NOT ALLOWED
- 1012 - GPRS SERVICES NOT ALLOWED IN THIS PLMN
- 1013 - PDP DUPLICATE
- 1014 - USER EQUIPMENT RADIO ACCESS TECHNOLOGY CHANGE
- 1015 - CONGESTION
- 1016 - NO PDP CONEXT ACTIVATED
- 1017 - ACCESS CLASS DSAC REJECTION
- 1018 - PDP ACTIVATE MAX RETRY FAILED

- 1019 - RAB FAILURE
- 1020 - EPS SERVICE NOT ALLOWED
- 1021 - TRACKING AREA NOT ALLOWED
- 1022 - ROAMING NOT ALLOWED IN THIS TRACKING AREA
- 1023 - NO SUITABLE CELLS IN TRACKING AREA
- 1024 - NOT AUTHORIZED FOR THIS CLOSED SUBSCRIBER GROUP
- 1025 - ESM UNKNOWN EPS BEARER CONTEXT
- 1026 - DRB RELEASED AT RRC
- 1027 - NAS SIG CONN RELEASED
- 1028 - EPS MOBILITY MANAGEMENT DETACHED
- 1029 - EPS MOBILITY MANAGEMENT ATTACH FAILED
- 1030 - EPS MOBILITY MANAGEMENT ATTACH STARTED
- 1031 - LTE NAS SERVICE REQ FAILED
- 1032 - ESM(EPS Session Management) ACTIVE DEDICATED BEARER REACTIVATED BY NW
- 1033 - ESM(EPS Session Management) LOWER LAYER FAILURE
- 1034 - ESM(EPS Session Management) SYNC UP WITH NW
- 1035 - ESM(EPS Session Management) NW ACTIVATED DED BEARER WITH ID OF DEF BEARER
- 1036 - ESM(EPS Session Management) BAD OTA MESSAGE
- 1037 - ESM DS REJECTED THE CALL
- 1038 - ESM(EPS Session Management) CONTEXT TRANSFERRED DUE TO IRAT
- 1039 - DS EXPLICIT DEACT
- 1040 - ESM(EPS Session Management) LOCAL CAUSE NONE
- 1041 - LTE NAS SERVICE REQ FAILED NO THROTTLE
- 1042 - ACL FAILURE, This error code should rarely triggered and reported to the application
- 1043 - LTE NAS SERVICE REQ FAILED DS DISALLOW
- 1044 - EMM(EPS Mobility Management) T3417 EXPIRED
- 1045 - EMM(EPS Mobility Management) T3417 EXT EXPIRED
- 1046 - LRR(LTE Radio Resource Control) UL DATA CNF FAILURE TXN - Light Radio Resource Controller Uplink data confirmation failure
- 1047 - LRR(LTE Radio Resource Control) UL DATA CNF FAILURE HO
- 1048 - LRR(LTE Radio Resource Control) UL DATA CNF FAILURE CONN REL
- 1049 - LRR(LTE Radio Resource Control) UL DATA CNF FAILURE RLF
- 1050 - LRR(LTE Radio Resource Control) UL DATA CNF FAILURE CTRL NOT CONN
- 1051 - LRR(LTE Radio Resource Control) CONN EST FAILURE
- 1052 - LRR(LTE Radio Resource Control) CONN EST FAILURE ABORTED
- 1053 - LRR(LTE Radio Resource Control) CONN EST FAILURE ACCESS BARRED

- 1054 - LRRc(LTE Radio Resource Control) CONN EST FAILURE CELL RESEL
- 1055 - LRRc(LTE Radio Resource Control) CONN EST FAILURE CONFIG FAILURE
- 1056 - LRRc(LTE Radio Resource Control) CONN EST FAILURE TIMER EXPIRED
- 1057 - LRRc(LTE Radio Resource Control) CONN EST FAILURE LINK FAILURE
- 1058 - LRRc(LTE Radio Resource Control) CONN EST FAILURE NOT CAMPED
- 1059 - LRRc(LTE Radio Resource Control) CONN EST FAILURE SI FAILURE
- 1060 - LRRc(LTE Radio Resource Control) CONN EST FAILURE CONN REJECT
- 1061 - LRRc(LTE Radio Resource Control) CONN REL NORMAL
- 1062 - LRRc(LTE Radio Resource Control) CONN REL RLF
- 1063 - LRRc(LTE Radio Resource Control) CONN REL CRE FAILURE
- 1064 - LRRc(LTE Radio Resource Control) CONN REL OOS DURING CRE
- 1065 - LRRc(LTE Radio Resource Control) CONN REL ABORTED
- 1066 - LRRc(LTE Radio Resource Control) CONN REL SIB READ ERROR
- 1067 - DETACH WITH REATTACH LTE NW DETACH
- 1068 - DETACH WITH OUT REATTACH LTE NW DETACH
- 1069 - ESM(EPS Session Management) PROC TIME OUT
- 1070 - MESSAGE EXCEED MAX L2 LIMIT
- 1500 - CD GEN OR BUSY, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to the reception of a Connection Deny message with a deny code of general or network busy
- 1501 - CD BILL OR AUTH, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to the reception of a Connection Deny message with a deny code of billing failure or authentication failure
- 1502 - CHG HDR, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since there is a change to HDR system due to redirection or PRL not preferred
- 1503 - EXIT HDR, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device exited HDR due to redirection or PRL not preferred
- 1504 - HDR NO SESSION, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device does not have a HDR session
- 1505 - HDR ORIG DURING GPS FIX, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since it is ending an HDR call origination in favor of a GPS fix
- 1506 - HDR CS TIMEOUT, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since connection setup on HDR system timed out
- 1507 - HDR RELEASED BY CM, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) when it wants to release a HDR call so a 1X call can continue
- 1508 - COLLOC ACQ FAIL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) when device failed to acquire co-located HDR for origination
- 1509 - OTASP COMMIT IN PROG, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since an OTASP commit is in progress

- 1510 - NO HYBR HDR SRV, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device has no Hybrid HDR service
- 1511 - HDR NO LOCK GRANTED, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since HDR module could not get the RF lock
- 1512 - HOLD OTHER IN PROG, this error code is returned when data call is brought down by CM(Call Manager) because DBM or SMS is in progress
- 1513 - HDR FADE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since HDR module released the call due to fade
- 1514 - HDR ACC FAIL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to HDR system Access Failure
- 2000 - CLIENT END, this error code is returned when client ends the data call
- 2001 - NO SRV, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since device has no service
- 2002 - FADE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device lost the system due to fade
- 2003 - REL NORMAL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to receiving a release from base station with no reason
- 2004 - ACC IN PROG, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to Access attempt already in progress
- 2005 - ACC FAIL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to Access Failure
- 2006 - REDIR OR HANDOFF, this error code is returned when data call is brought down because device is in the process of redirecting/handing off to a different target system
- 2500 - OFFLINE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device went offline
- 2501 - EMERGENCY MODE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since device is operating in Emergency mode
- 2502 - PHONE IN USE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device is in use (e.g voice call)
- 2503 - INVALID MODE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since the device's operational mode is different from the mode requested in the traffic channel bring up
- 2504 - INVALID SIM STATE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since the SIM was marked by network as invalid for circuit and/or packet service domain
- 2505 - NO COLLOC HDR, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since there is no collocated HDR
- 2506 - CALL CONTROL REJECTED, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since Call control module rejected the request

9.30.2.9 3GPP specification defined call end reasons (Type=6)

- 8 - OPERATOR DETERMINED BARRING, this reason code is posted by the MME(Mobility Management Entity) to indicate operator has barred the UE
- 25 - LLC SMDCP FAILURE, PDP context deactivation initiated by the MS or by the Network
- 26 - INSUFFICIENT RESOURCES, this reason is posted to indicate that the network cannot provide the requested service due to insufficient resources
- 27 - MISSING OR UNKNOWN APN, the APN was required and not specified or APN could not be resolved. In LTE mode of operation, this is a PDN throttling cause code, meaning the UE may end up throttling further requests to the same APN
- 28 - UNKNOWN PDN TYPE, the reason is posted by the network to indicate that the PDN type was not recognized
- 29 - AUTH FAILED, the reason is posted when authentication fails. In LTE mode of operation, this is a PDN throttling cause code, meaning the UE may end up throttling further requests to the same APN
- 30 - GGSN REJECT, the reason is posted when the request was rejected by Serving GW or PDN GW. In LTE mode of operation, this is a PDN throttling cause code, meaning the UE may end up throttling further requests to the same APN
- 31 - ACTIVATION REJECT, the reason is posted when the request is rejected by the network due to unspecified reasons
- 32 - OPTION NOT SUPPORTED, the reason is posted when UE requested a service not supported by the PLMN
- 33 - OPTION UNSUBSCRIBED, This cause is sent when the MS requests a service option for which it has no subscription
- 34 - OPTION TEMP OOO, service option temporarily out of order, this reason is posted when the network is temporarily out of resources to service the request
- 35 - PTI ALREADY USED, the reason is posted to indicate that PTI (Procedure Transaction Identifier) used in the request is already active via another UE requested procedure
- 36 - REGULAR DEACTIVATION, this reason is posted by the network to initiate a regular release of bearer resources
- 37 - EPS QOS NOT ACCEPTED, this reason is posted by the network to indicate that the QoS requested by the UE could not be accepted
- 38 - NETWORK FAILURE, this reason is posted when an error occurs in the network
- 39 - UMTS REACTIVATION REQ, this reason is posted by the network to request for bearer reactivation. This code may be posted during network congestion
- 40 - FEATURE NOT SUPPORTED, Unsuccessful MBMS context activation requested by the network
- 41 - TFT SEMANTIC ERROR, the reason is posted by the network to indicate semantic error(s) in specifying TFT operation included in the request
- 42 - TFT SYNTAX ERROR, the reason is posted by the network to indicate syntactic error(s) in specifying TFT operation included in the request
- 43 - UNKNOWN PDP CONTEXT, the reason is posted when the bearer identity (or linked bearer identity) in the request is invalid (or inactive)
- 44 - FILTER SEMANTIC ERROR, the reason is posted by the network to indicate semantic error(s) in specifying packet filter(s) associated with a TFT
- 45 - FILTER SYNTAX ERROR, the reason is posted by the network to indicate syntactic error(s) in specifying packet filter(s) associated with a TFT

- 46 - PDP WITHOUT ACTIVE TFT, the reason is posted by the network when UW requested more than one PDP connection without TFT
- 50 - IPV4 ONLY ALLOWED, 3GPP equivalent ESM(EPS Session Management) cause code value 50, PDN type IPv4 only allowed.
- 51 - IPV6 ONLY ALLOWED, 3GPP equivalent ESM(EPS Session Management) cause code value 51, PDN type IPv6 only allowed
- 52 - SINGLE ADDRESS BEARER ONLY, 3GPP equivalent ESM(EPS Session Management) cause code value 52, Single address bearers only allowed. The reason is posted when the network supports single address bearers only, meaning dual IP bearers are not supported
- 53 - ESM INFORMATION NOT RECEIVED, 3GPP equivalent ESM(EPS Session Management) cause code value 53, ESM information not received. The reason is posted by the network to indicate that the PDN connection request was rejected because ESM information was not received
- 54 - PND CONNECTION DOES NOT EXIST, 3GPP equivalent ESM(EPS Session Management) cause code value 54, PDN connection does not exist The reason is posted by the network during handover from a non-3GPP network to indicate that the MME does not have any information regarding the requested PDN connection
- 55 - MULTIPLE CONNECTION TO SAME PDN NOT ALLOWED, 3GPP equivalent ESM(EPS Session Management) cause code value 55, Multiple PDN connections for a given APN not allowed. The reason is posted by the network to indicate that the UE is already connected to the requested APN via another PDN/PDN connection
- 81 - INVALID TRANSACTION ID, the reason is posted by the network to indicate that the PTI used in the request is unassigned or reserved
- 95 - MESSAGE INCORRECT SEMANTIC, the reason is posted by the network to indicate receipt of an invalid message
- 96 - INVALID MANDATORY INFO, the reason is posted by the network to indicate receipt of a message with semantic error in a mandatory information element
- 97 - MESSAGE TYPE UNSUPPORTED, the reason is posted by the network to indicate receipt of a message that is either undefined or defined but not implemented by the equipment sending this ESM cause
- 98 - MSG TYPE NONCOMPATIBLE STATE, the reason is posted by the network to indicate receipt of a message type that cannot be handled in the current network protocol state
- 99 - UNKNOWN INFO ELEMENT, the reason is posted by the network to indicate receipt of a message that includes an information element that is either not defined or defined but not implemented by the equipment sending the ESM cause
- 100 - CONDITIONAL IE ERROR, the reason is posted by the network to indicate receipt of a message that includes a syntactically incorrect information element. This message is ignored by the network.
- 101 - MSG AND PROTOCOL STATE UNCOMPATIBLE, the reason is posted by the network to indicate receipt of a message that cannot be handled in the current network protocol state
- 111 - PROTOCOL ERROR, the reason is posted by the network to indicate a protocol error when no other error applies
- 112 - APN TYPE CONFLICT
- 113 - INVALID PROXY-CALL SESSION CONTROL FUNCTION ADDRESS

9.30.2.10 PPP call end reasons (Type=7)

- 1 - TIMEOUT, this error code is returned when the data call bring up fails in PPP setup due to timeout (For e.g: LCP Conf Ack not received from network)
- 2 - AUTH FAILURE, this error code is returned when the data call bring up fails in PPP setup due to authentication failure
- 3 - OPTION MISMATCH, this error code is returned when the data call bring up fails in PPP setup due option mismatch (e.g: Authentication is required, but not negotiated with network during LCP phase)
- 31 - PAP FAILURE, this error code is returned when the data call bring up fails in PPP setup due to PAP failure
- 32 - CHAP FAILURE, this error code is returned when the data call bring up fails in PPP setup due to CHAP failure
- 33 - CLOSE IN PROGRESS, this error code is returned when the data call bring up fails in PPP setup since PPP is in the process of cleaning the previous PPP session
- -1 - UNKNOWN, this error code is unused

9.30.2.11 EHRPD call end reasons (Type=8)

- 1 - SUBS LIMITED TO V4, this error code is returned when the V6 interface bring up fails because network provided only V4 address for the upcoming PDN
- 2 - SUBS LIMITED TO V6, this error code is returned when the V4 interface bring up fails because network provided only V6 address for the upcoming PDN
- 4 - VSNCP(Vendor Specific Network Control Protocol) TIMEOUT, this error code is returned when the data call bring up fails in VSNCP phase due to VSNCP timeout error
- 5 - VSNCP(Vendor Specific Network Control Protocol) FAILURE, this error code is returned when VSNCP configuration failed during call bring up
- 6 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I GEN ERROR, this error code is returned when the data call bring up fails in VSNCP phase due to general error
- 7 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I UNAUTH APN, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason requested APN is unauthorized
- 8 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN LIMIT EXCEED, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN limit exceeded
- 9 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I NO PDN GW, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason no PDN gateway
- 10 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN GW UNREACH, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN gateway unreachable
- 11 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN GW REJ, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN gateway reject
- 12 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I INSUFF PARAM, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason insufficient parameter

- 13 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I RESOURCE UNAVAIL, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason resource unavailable
- 14 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I ADMIN PROHIBIT, this error code is returned when the data call bring up fails in SNCP phase since network rejected VSNCP config request with reason admin prohibited
- 15 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN ID IN USE, this error code is returned when the data call bring up fails in VSNCP phase because network rejected with reason PDN ID IN USE (or) All existing PDNs are brought down with this end reason because one of the PDN bring up got rejected by network with reason PDN ID IN USE
- 16 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I SUBSCR LIMITATION, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason subscriber limitation
- 17 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN EXISTS FOR THIS APN, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN exists for this APN

9.30.2.12 IPv6 call end reasons (Type=9)

- 1 - PREFIX UNAVAILABLE, this error code is returned when V6 data call is brought down because device failed to get the prefix from network
- 2 - IPV6 ERR HRPD IPV6 DISABLED, this error code is returned when V6 data call bring up is rejected because IPV6 is disabled in 1X/HRPD mode
- 3 - IPV6 DISABLED, this error code is returned when IPv6 data call bring up is rejected because NV1896 (IPV6 enable) is disabled

Copyright: © 2011-2013 Sierra Wireless, Inc. all rights reserved

9.31 qaGobiApiTableCarrierCodes.h File Reference

Carrier Codes table.

9.31.1 Detailed Description

Carrier Codes table.

9.31.2 Carrier Codes (Number - Carrier)

- 0 - no carrier specified
- 1 - Generic
- 2 - Telstra
- 4 - AT&T
- 5 - Verizon
- 11 - Sprint
- 12 - Telefonica
- 101 - Verizon
- 102 - Sprint
- 103 - Alltel
- 104 - Bell Mobility
- 105 - Telus
- 106 - U.S. Cellular
- 107 - Telstra
- 108 - China Unicom
- 109 - Telecom New Zealand
- 110 - SK Telecom
- 111 - Reliance Communications
- 112 - Tata Communications
- 113 - MetroPCS Communications
- 114 - Leap Wireless
- 115 - KDDI
- 116 - Grupo Iusacell
- 117 - China Telecom
- 118 - Open Mobile Handset
- 176 - Rogers
- 177 - NetIndex
- 178 - DNA
- 179 - Big Pond
- 201 - AT&T
- 202 - Vodafone
- 203 - T-Mobile
- 204 - Orange
- 205 - Telefonica

- 206 - Telecom Italia
- 207 - 3
- 208 - O2
- 209 - SFR
- 210 - Swisscom AG
- 211 - China Mobile
- 212 - Telstra
- 213 - Singapore Telecommunications
- 214 - Reliance Telecommunications
- 215 - Bharti Airtel
- 216 - NTT docomo
- 217 - E Mobile
- 218 - Softbank
- 219 - Korea Telecom Freetel
- 220 - SK Telecom
- 221 - Telenor
- 222 - NetCom Norway
- 223 - TeliaSonera
- 224 - América Móvil
- 225 - Brasil Vivo
- 0xFFFFFFFF - Unknown

Copyright: © 2011-2014 Sierra Wireless, Inc. all rights reserved

9.32 qaGobiApiTableCodingScheme.h File Reference

Data Coding Scheme.

Macros

- `#define __GOBI_API_CODING_SCHEME_H__`

9.32.1 Detailed Description

Data Coding Scheme.

9.32.2 Coding Group Bits 7..4(0000)

9.32.2.1 Use of bits 3..0

- Language using the GSM 7 bit default alphabet Bits 3..0 indicate the language:
 - 0000 German
 - 0001 English
 - 0010 Italian
 - 0011 French
 - 0100 Spanish
 - 0101 Dutch
 - 0110 Swedish
 - 0111 Danish
 - 1000 Portuguese
 - 1001 Finnish
 - 1010 Norwegian
 - 1011 Greek
 - 1100 Turkish
 - 1101 Hungarian
 - 1110 Polish
 - 1111 Language unspecified

9.32.3 Coding Group Bits 7..4(0001)

9.32.3.1 use of bits 3..0

- 0000 GSM 7 bit default alphabet; message preceded by language indication. The first 3 characters of the message are a two-character representation of the language encoded according to ISO 639 [12], followed by a CR character. The CR character is then followed by 90 characters of text.
- 0001 UCS2; message preceded by language indication. The message starts with a two GSM 7-bit default alphabet character representation of the language encoded according to ISO 639. This is padded to the octet boundary with two bits set to 0 and then followed by 40 characters of UCS2-encoded message. An MS not supporting UCS2 coding will present the two character language identifier followed by improperly interpreted user data.

9.32.4 Coding Group Bits 7..4(0010)

9.32.4.1 use of bits 3..0

- 0000 Czech
- 0001 Hebrew
- 0010 Arabic
- 0011 Russian
- 0100 Icelandic
- 0101..1111 Reserved for other languages using the GSM 7 bit default alphabet, with unspecified handling at the MS

9.32.5 Coding Group Bits 7..4(0011)

9.32.5.1 use of bits 3..0

- 0000..1111 Reserved for other languages using the GSM 7 bit default alphabet, with unspecified handling at the MS

9.32.6 Coding Group Bits 7..4(01xx)

9.32.6.1 use of bits 3..0

- General Data Coding indication
 - Bits 5..0 indicate the following:
 - Bit 5, if set to 0, indicates the text is uncompressed
 - Bit 5, if set to 1, indicates the text is compressed using the compression algorithm defined in 3GPP TS 23.042
 - Bit 4, if set to 0, indicates that bits 1 to 0 are reserved and have no message class meaning
 - Bit 4, if set to 1, indicates that bits 1 to 0 have a message class meaning: Bit 1 Bit 0 Message Class:
 - 0 0 Class 0
 - 0 1 Class 1 Default meaning: ME-specific.
 - 1 0 Class 2 (U)SIM specific message.
 - 1 1 Class 3 Default meaning: TE-specific (see 3GPP TS 27.005)
- Bits 3 and 2 indicate the character set being used, as follows:
- Bit 3 Bit 2 Character set:
- 0 0 GSM 7 bit default alphabet
 - 0 1 8 bit data
 - 1 0 UCS2 (16 bit) [10]
 - 1 1 Reserved

9.32.7 Coding Group Bits 7..4(1001)

9.32.7.1 Reserved coding groups

- Message with User Data Header (UDH) structure:
 - Bit 1 Bit 0 Message Class:
 - 0 0 Class 0
 - 0 1 Class 1 Default meaning: ME-specific.
 - 1 0 Class 2 (U)SIM specific message.
 - 1 1 Class 3 Default meaning: TE-specific (see 3GPP TS 27.005 [8])
- Bits 3 and 2 indicate the alphabet being used, as follows:
- Bit 3 Bit 2 Alphabet:
- 0 0 GSM 7 bit default alphabet
 - 0 1 8 bit data
 - 1 0 USC2 (16 bit) [10]
 - 1 1 Reserved

9.32.8 Coding Group Bits 7..4(1010..1101)

9.32.8.1 Reserved coding groups

9.32.9 Coding Group Bits 7..4(1110)

9.32.9.1 Defined by the WAP Forum

9.32.10 Coding Group Bits 7..4 (1111)

9.32.10.1 Data coding / message handling

- Bit 3 is reserved, set to 0.
 - Bit 2 Message coding:
 - 0 GSM 7 bit default alphabet
 - 1 8 bit data
 - Bit 1 Bit 0 Message Class:
 - 0 0 No message class.
 - 0 1 Class 1 user defined.
 - 1 0 Class 2 user defined.
 - 1 1 Class 3
- default meaning: TE specific(3GPP TS 27.005)
Copyright: © 2012 Sierra Wireless, Inc. all rights reserved

9.32.11 Macro Definition Documentation

9.32.11.1 #define __GOBI_API_CODING_SCHEME_H__

9.33 qaGobiApiTableGpsCapabilityCodes.h File Reference

Position Determination Service API GPS Capability Codes.

9.33.1 Detailed Description

Position Determination Service API GPS Capability Codes.

9.33.2 GPS capability (Value - Capability)

- 0 - None
- 1 - Standalone
- 2 - Assisted (including XTRA and implying standalone is also supported)
- 3 - Assisted (without XTRA and implying standalone is also supported)
- 0xFFFFFFFF - Unknown

Copyright: © 2011 Sierra Wireless, Inc. all rights reserved

9.34 qaGobiApiTablePowerModes.h File Reference

Device Management Service API Power Modes table.

9.34.1 Detailed Description

Device Management Service API Power Modes table.

9.34.2 Power Modes (Value - Description)

- 0 - Online (default)
- 1 - Low power (airplane) mode
- 2 - Factory test mode
- 3 - Offline
- 4 - Reset
- 5 - Power off
- 6 - Persistent low power (airplane) mode
- 7 - Mode - only low power

Valid transitions for Power Modes

- Online to Low Power, Persistent low power, Factory test, Offline or Shut Down
- Low power to online, Persistent low power, Offline, or Shut Down
- Persistent low power to Online, Low power, Offline or Shut down
- Factory test to online
- Offline to Reset

Copyright: © 2011 Sierra Wireless, Inc. all rights reserved

9.35 qaGobiApiTableRadioInterfaces.h File Reference

Network Access Service API Radio Interfaces table.

9.35.1 Detailed Description

Network Access Service API Radio Interfaces table.

9.35.2 Radio interface

9.35.2.1 Technology (Value - Radio Interface Technology)

- 0 - No service
- 1 - CDMA 1xRTT
- 2 - CDMA 1xEV-DO
- 3 - AMPS (Unsupported)
- 4 - GSM
- 5 - UMTS
- 6 - WLAN
- 7 - GPS
- 8 - LTE

Copyright: © 2011 Sierra Wireless, Inc. all rights reserved

9.36 qaGobiApiTableRegionCodes.h File Reference

Region Codes table.

9.36.1 Detailed Description

Region Codes table.

9.36.2 Region Codes (Code - Region)

- 0 - North America
- 1 - Latin America
- 2 - Europe
- 3 - Asia
- 4 - Australia
- 5 - Global
- 0xFFFFFFFF - Unknown

Copyright: © 2011 Sierra Wireless, Inc. all rights reserved

9.37 qaGobiApiTableServiceOptions.h File Reference

Voice Service Options.

9.37.1 Detailed Description

Voice Service Options.

9.37.2 Service Option codes (Code - Reason)

9.37.2.1 Description

- 0x0001 - Basic variable rate voice service (8 kbps)
- 0x0002 - Mobile station loopback (8 kbps)
- 0x0003 - Enhanced variable rate voice service (8 kbps)
- 0x0004 - Asynchronous data service (9.6 kbps)
- 0x0005 - Group 3 facsimile (9.6 kbps)
- 0x0006 - Short message service (rate set 1)
- 0x0007 - Packet data service: Internet or ISO Protocol stack (9.6 kbps)
- 0x0008 - Packet data service: CDPD Protocol stack (9.6 kbps)
- 0x0009 - Mobile station loopback (13 kbps)
- 0x000A - transparent service
- 0x000B - III nontransparent service
- 0x000C - Asynchronous data service (14.4 or 9.6 kbps)
- 0x000D - Group 3 facsimile (14.4 or 9.6 kbps)
- 0x000E - Short message service (rate set 2)
- 0x000F - Packet data service: Internet or ISO Protocol stack (14.4 kbps)
- 0x0010 - Packet data service: CDPD Protocol stack (14.4 kbps)
- 0x0011 - High-rate voice service (13 kbps)
- 0x0012 - Over-the-air parameter administration (rate set 1)
- 0x0013 - Over-the-air parameter administration (rate set 2)
- 0x0014 - Group 3 analog facsimile (rate set 1)
- 0x0015 - Group 3 analog facsimile (rate set 2)
- 0x0016 - High-speed packet data service: Internet or ISO Protocol stack (RS1 forward, RS1 reverse)
- 0x0017 - High-speed packet data service: Internet or ISO Protocol stack (RS1 forward, RS2 reverse)
- 0x0018 - High-speed packet data service: Internet or ISO Protocol stack (RS2 forward, RS1 reverse)
- 0x0019 - High-speed packet data service: Internet or ISO Protocol stack (RS2 forward, RS2 reverse)
- 0x001A - High-speed packet data service: CDPD Protocol stack (RS1 forward, RS1 reverse)
- 0x001B - High-speed packet data service: CDPD Protocol stack (RS1 forward, RS2 reverse)
- 0x001C - High-speed packet data service: CDPD Protocol stack (RS2 forward, RS1 reverse)
- 0x001D - High-speed packet data service: CDPD Protocol stack (RS2 forward, RS2 reverse)

- 0x001E - RATE_SET_1 Supplemental channel loopback test for rate set 1
- 0x001F - RATE_SET_2 Supplemental channel loopback test for rate set 2
- 0x0020 - Test Data Service Option (TDSO)
- 0x0021 - cdma2000 high-speed packet data service, Internet or ISO Protocol stack
- 0x0022 - cdma2000 high-speed packet data service, CDPD Protocol
- 0x0023 - Location services, rate set 1 (9.6 kbps)
- 0x0024 - Location services, rate set 2 (14.4 kbps)
- 0x0025 - ISDN interworking service (64 kbps)
- 0x0026 - GSM voice
- 0x0027 - GSM circuit data
- 0x0028 - GSM packet data
- 0x0029 - GSM short message service
- 0x0036 - Markov Service Option (MSO)
- 0x0037 - Loopback Service Option (LSO)
- 0x0038 - Selectable mode vocoder
- 0x0039 - 32 kbps circuit video conferencing
- 0x003A - CONFERENCING 64 kbps circuit video conferencing
- 0x003B - HRPD packet data service, which when used in paging over the 1X air interface, a page response is not required
- 0x003C - Link Layer Assisted Robust Header Compression (LLA ROHC) - header removal
- 0x003D - LLA ROHC - Header Compression
- 0x003E - Source-controlled Variable-Rate Multimode Wideband (VMR-WB) speech codec rate set 2
- 0x003F - Source-controlled VMR-WB speech codec rate set 1
- 0x0040 - HRPD auxiliary packet data service instance
- 0x0041 - cdma2000/GPRS interworking
- 0x0042 - ISO_PROTOCOL_SO_66 cdma2000 high-speed packet data service, Internet or ISO Protocol stack
- 0x0043 - HRPD packet data IP service where higher layer protocol is IP or ROHC
- 0x0044 - Enhanced variable rate voice service (EVRC-B)
- 0x0045 - HRPD packet data service, which when used in paging over the 1X air interface, a page response is required
- 0x0046 - Enhanced variable rate voice service (EVRC-WB)
- 0x1004 - Asynchronous data service, Revision 1 (9.6 or 14.4 kbps)
- 0x1005 - Group 3 facsimile, Revision 1 (9.6 or 14.4 kbps)
- 0x1007 - Packet data service: Internet or ISO Protocol stack, Revision 1 (9.6 or 14.4 kbps)
- 0x1008 - Packet data service: CDPD Protocol stack, Revision 1 (9.6 or 14.4 kbps)
- 0x7FF8 - Identifies service reference identifier 0
- 0x7FF9 - Identifies service reference identifier 1

- 0x7FFA - Identifies service reference identifier 2
- 0x7FFB - Identifies service reference identifier 3
- 0x7FFC - Identifies service reference identifier 4
- 0x7FFD - Identifies service reference identifier 5
- 0x7FFE - Identifies service reference identifier 6
- 0x7FFF - Identifies service reference identifier 7

Copyright: © 2012 Sierra Wireless, Inc. all rights reserved

9.38 qaGobiApiTableSupServiceInfoClasses.h File Reference

Voice Supplementary Service Information Classes.

9.38.1 Detailed Description

Voice Supplementary Service Information Classes.

9.38.2 Supplementary Service Information Classes (Value - Service Class)

- 0X00 - CLASS_NONE
- 0X01 - CLASS_VOICE
- 0X02 - CLASS_DATA
- 0X04 - CLASS_FAX
- 0X08 - CLASS_SMS
- 0X10 - CLASS_DATACIRCUITSYNC
- 0X20 - CLASS_DATACIRCUITASYNC
- 0X40 - CLASS_PACKETACCESS
- 0X80 - CLASS_PADACCESS

Copyright: © 2012 Sierra Wireless, Inc. all rights reserved

9.39 qaGobiApiTableSwiAudio.h File Reference

Swi Audio related tables.

9.39.1 Detailed Description

Swi Audio related tables.

9.39.2 ACDB Device (Device ID - description)

- 0 - Vehicle HF
- 1 - Handset
- 2 - TTY
- 3 - USB
- 4 - NA

9.39.3 Physical Interface (Device ID - description - Interface parameters)

- 0 - PCM - Mode: 0-slave, 1-master, 2-Auxiliary PCM; Rate: 0-8k, 1-16k; Format: 0-linear, 1-u-law, 2-A-law; Padding: 0-disable, 1-enable; Bits-frame: 0-8BPF, 1-16BPF, 2-32BPF, 3-64BPF, 4-128BPF, 5-256BPF;
- 1 - I2S - None
- 2 - Analog(with internal codec) - None
- 3 - USB - None

Copyright: © 2013 Sierra Wireless, Inc. all rights reserved

9.40 qaGobiApiTableSwiOMADMUpdateCompleteStatus.h File Reference

Update Complete Status table.

9.40.1 Detailed Description

Update Complete Status table.

9.40.2 OMA DM Update Complete Status (Update Complete Status - Meaning - Usage)

- 200 - Successful - The request has succeeded
- 250-299 - Successful(vendor specified) - successful operation with vendor specified ResultCode
- 400 - Management Client Error - Management Client error - based on User or Device behavior
- 401 - User Cancelled - User chose not to accept the operation when prompted
- 402 - Corrupted Firmware Update Package - Corrupted firmware update package did not store correctly. Detected for example, by mismatch CRCs between actual and expected
- 403 - Firmware UpdatePackage(Device Mismatch) - Wrong firmware update package delivered to device based on current device characteristics
- 404 - Failed Firmware Update Package Validation - Failure to positively validate digital signature of firmware update package
- 405 - Firmware Update Package Not acceptable - firmware update package is not acceptable

- 406 - Alternate Download Authentication Failure - authentication required but authentication failure was encountered when downloading firmware update package
- 407 - Alternate Download Request Timeout - client has encountered a timeout when downloading firmware update package
- 408 - Not Implemented - the device does not support the requested operation
- 409 - Undefined Error - indicates failure not defined by any other error code
- 410 - Firmware Update Failed - firmware update operation failed in device
- 411 - Malformed or Bad URL - the URL provided for alternate download is bad
- 412 - Alternate Download Server Unavailable - the alternate download server is unavailable or does not respond
- 450 - Client Error (OMADM General) - Vendor defined client error
- 451 - Client Error (OMADM SyncML) - Vendor defined client error
- 452 - Client Error (OMADM Auth) - Vendor defined client error
- 453 - Client Error (OMADM Protocol) - Vendor defined client error
- 454 - Client Error (OMADM Tree) - Vendor defined client error
- 455 - Client Error (OMADM DStore) - Vendor defined client error
- 456 - Client Error (OMADM Trigger) - Vendor defined client error
- 457 - Client Error (OMADM Fumo) - Vendor defined client error
- 458 - Client Error (OMADM Comms) - Vendor defined client error
- 459 - Client Error (OMADM Parse) - Vendor defined client error
- 460 - Client Error (OMADM TNDIS) - Vendor defined client error
- 461 - Client Error (OMADM SCM) - Vendor defined client error
- 462 - Client Error (OMADM Impl) - Vendor defined client error
- 463-499 - Client Error (Vendor Specified) - client error encountered for operation with vendor specified result code
- 500 - Alternate Download Server Error - Alternate download server error encountered
- 501 - Download fails due to device out of memory - The download fails due to insufficient memory in the device to save the firmware update package
- 502 - Firmware update fails due to device out of memory - The update fails because there isn't sufficient memory to update the device
- 503 - Download fails due to network issues - The download fails due to network/transport level errors
- 550-599 - Alternate Download Server Error (vendor specified)- Alternate download server error encountered for operation with vendor specified result code

Copyright: © 2013 Sierra Wireless, Inc. all rights reserved

9.41 qaGobiApiTableVoiceCallEndReasons.h File Reference

Voice Service Call and supplementary services end reasons.

9.41.1 Detailed Description

Voice Service Call and supplementary services end reasons.

9.41.2 Voice Call and supplementary services end reason codes (Code - Reason)

9.41.2.1 General

- 0 - Phone is offline
- 20 - Phone is CDMA locked until a power cycle; CDMA only
- 21 - Phone has no service, this is for backward compatibility
- 22 - Call has ended abnormally; CDMA only
- 23 - Received intercept from the base station; originating only; CDMA only
- 24 - Received reorder from the base station; originating only; CDMA only
- 25 - Received release from the base station; no reason was given
- 26 - Received release from the base station; SO reject; CDMA only
- 27 - Received incoming call from the base station
- 28 - Received alert stop from the base station; incoming only; CDMA only
- 29 - Client ended the call
- 30 - Received end activation; OTASP call only; CDMA only
- 31 - MC aborted the origination/conversation; CDMA only
- 32 - Maximum access probes were transmitted; CDMA only
- 33 - Persistence test failure; FEATURE_JCDMA only; CDMA only
- 34 - R-UIM is not present
- 35 - Access attempt is already in progress
- 36 - Access failure for a reason other than the above
- 37 - Received retry order; originating only; IS 2000; CDMA only
- 38 - BYBS Concurrent service is not supported by the base station
- 39 - No response was received from the base station
- 40 - Call was rejected by the base station; CDMA only
- 41 - Concurrent services requested were not compatible; CDMA only
- 42 - Access is blocked by the base station; CDMA only
- 43 - Corresponds to CM_CALL_ORIG_ERR_ALREADY_IN_TC
- 44 - Call is ended because an emergency call is flashed over this call; CDMA only
- 45 - Used if CM is ending a GPS call in preference of a user call
- 46 - Used if CM is ending an SMS call in preference of a user call
- 47 - Used if CM is ending a data call in preference of an emergency call

- 48 - Call was rejected because of a redirection or handoff
- 49 - Access is blocked by the base station for all mobiles; KDDI-specific; CDMA only
- 50 - To support OTASP SPC Error indication
- 51 - Maximum access probes for an IS-707B call; CDMA only
- 52 - Base station reject order
- 53 - Base station retry order
- 54 - Timer T42 is expired
- 55 - Timer T40 is expired
- 56 - Service initialization failure - Traffic Channel Initialization
- 57 - Timer T50m is expired - Traffic Channel Initialization
- 58 - Timer T51m is expired - Traffic Channel Initialization
- 59 - Acknowledgement timeout due to 12 retransmissions
- 60 - Bad forward link or timer T5M is expired
- 61 - Transceiver Resource Manager request failed
- 62 - Timer T41 is expired
- 100 - WCDMA/GSM/TDS only; call end LL cause, Received a reason for ending the call from the lower layer
- 101 - WCDMA/GSM only; Call origination request failed
- 102 - WCDMA/GSM only; client rejected an incoming call
- 103 - WCDMA/GSM only; client rejected a setup indication
- 104 - WCDMA/GSM only; network ended the call
- 105 - WCDMA/GSM only
- 106 - GWM/WCDMA only; phone has no service
- 107 - 1X only; phone has no service
- 108 - Full service is unavailable
- 109 - Indicates resources are not available to handle a new MO/MT PS call

9.41.2.2 service Errors

- 110 - Unknown subscriber
- 111 - Illegal subscriber
- 112 - Bearer service not provisioned
- 113 - Tele service not provisioned
- 114 - Illegal equipment
- 115 - Call barred
- 116 - Illegal ss operation
- 117 - Ss error status

- 118 - Ss not available
- 119 - Ss subscription violation
- 120 - Ss incompatibility
- 121 - Facility not supported
- 122 - Absent subscriber
- 123 - Short term denial
- 124 - Long term denial
- 125 - System failure
- 126 - Data missing
- 127 - Unexpected data value
- 128 - Pwd registration failure
- 129 - Negative pwd check
- 130 - Num of pwd attempts violation
- 131 - Position method failure
- 132 - Unknown alphabet
- 133 - Ussd busy
- 134 - Rejected by user
- 135 - Rejected by network
- 136 - Deflection to served subscriber
- 137 - Special service code
- 138 - Invalid deflected to number
- 139 - Mpty participants exceeded
- 140 - Resources not available

9.41.2.3 control cause values

- 141 - Unassigned number
- 142 - No route to destination
- 143 - Channel unacceptable
- 144 - Operator determined barring
- 145 - Normal call clearing
- 146 - User busy sEE [s3, aNNEX h]
- 147 - No user responding sEE [s3, aNNEX h]
- 148 - User alerting no answer
- 149 - Call rejected sEE [s3, aNNEX h]
- 150 - Number changed sEE [s3, aNNEX h]

- 151 - Preemption sEE [s3, aNNEX h]
- 152 - Destination out of order
- 153 - Invalid number format
- 154 - Facility rejected
- 155 - Resp to status enquiry
- 156 - Normal unspecified
- 157 - No circuit or channel available
- 158 - Network out of order
- 159 - Temporary failure
- 160 - Switching equipment congestion
- 161 - Access information discarded
- 162 - Requested circuit or channel not available
- 163 - Resources unavailable or unspecified
- 164 - Qos unavailable
- 165 - Requested facility not subscribed
- 166 - Incoming calls barred within cug
- 167 - Bearer capability not auth
- 168 - Bearer capability unavailable
- 169 - Service option not available
- 170 - Acn limit exceeded
- 171 - Bearer service not implemented
- 172 - Requested facility not implemented
- 173 - Only digital information bearer available
- 174 - Service or option not implemented
- 175 - Invalid transaction identifier
- 176 - USER NOT MEMBER OF CUG
- 177 - Incompatible destination
- 178 - Invalid transit nw selection
- 179 - Semantically incorrect message
- 180 - Invalid mandatory information
- 181 - Message type non implemented
- 182 - Message type not compatible with protocol statE
- 183 - Information element non existent
- 184 - Conditonal ie error
- 185 - Message not compatible with protocol state
- 186 - Recovery on timer expired

- 187 - Protocol error unspecified
- 188 - Interworking unspecified
- 189 - Outgoing calls barred within cug
- 190 - No cug selection
- 191 - Unknown cug index
- 192 - Cug index incompatible
- 193 - Cug call failure unspecified
- 194 - Clir not subscribed
- 195 - Ccbs possible sEE
- 196 - Ccbs not possible

9.41.2.4 reject causes

- 197 - Imsi unknown in hlr
- 198 - Illegal ms
- 199 - Imsi unknown in vlr
- 200 - Imei not accepted
- 201 - Illegal me sEE
- 202 - Plmn not allowed
- 203 - Location area not allowed
- 204 - Roaming not allowed in this location area
- 205 - No suitable cells in location area
- 206 - Network failure sEE
- 207 - mac failure sEE
- 208 - Synch failure
- 209 - Network congestion
- 210 - GSM authentication unacceptable
- 211 - Service not subscribed
- 212 - Service temporarily out of order
- 213 - Call cannot be identified
- 214 - Incorrect semantics in message
- 215 - Mandatory information invalid
- 216 - Call failed due to other access stratum failures
- 217 - SIM is invalid
- 218 - Invalid call state
- 219 - Access class is blocked
- 220 - No resources are in the protocol stack to allow the call
- 221 - Invalid user data was received

9.41.2.5 reject causes

- 222 - Timer T3230 is expired
- 223 - No cell is available
- 224 - Abort message was received
- 225 - Radio link was lost due to other lower layer causes

9.41.2.6 reject causes

- 226 - Timer T303 is expired
- 227 - CNM MM release is pending

9.41.2.7 stratum reject causes

- 228 - Access stratum RR release indication
- 229 - Access stratum random access failure
- 230 - RRC_REL_IND Access stratum RRC release indication
- 231 - Access stratum close session indication
- 232 - Access stratum open session failure
- 233 - Access stratum low level failure
- 234 - Access stratum low level failure redial is not allowed
- 235 - Access stratum low level immediate retry
- 236 - Access stratum abort radio is unavailable

9.41.2.8 reject causes

- 237 - Service option is not supported

9.41.2.9 IP end reasons

- 300 - Received SIP 400 bad request;waiting for INVITE response
- 301 - Received SIP 400 bad request;waiting for INVITE response
- 302 - Received SIP 404 not found; call failed; called party does not exist
- 303 - Received SIP 415 unsupported media type; call failed; called party does not support media
- 304 - Received SIP 480 temporarily unavailable; call failed; called party is not in the LTE area
- 305 - No network response; call failed
- 306 - No network response; unable to put call on hold
- 307 - Moved to eHRPD; call failed or dropped; not in the LTE area

- 308 - Upgrade/downgrade rejected (200 OK with the current call SDP)
- 309 - Received 403 call forbidden; waiting for INVITE response
- 310 - Generic timeout; did not receive a response from the server or other end
- 311 - Reported on the MO side for generic internal software errors; user can try again if the call still exists
- 312 - Reported on the MT side if the upgrade timer has been cancelled or cannot complete the request for some reason after notifying the user of a re-invite request
- 313 - Call origination is rejected due to a Service-Specific Access Control (SSAC) barring
- 314 - Phone was put in thermal emergency
- 315 - 1XCSFB call ended because of a soft failure
- 316 - 1XCSFB call ended because of a hard failure

Copyright: © 2012 Sierra Wireless, Inc. all rights reserved

9.42 qaGobiApiTmd.h File Reference

Thermal Mitigation Device API function prototypes.

Data Structures

- struct [mitigationDevList](#)
- struct [TmdGetMitigationDevListResp](#)
- struct [TmdGetMitigationLvlReq](#)
- struct [TmdGetMitigationLvlResp](#)
- struct [TmdRegNotMitigationLvlReq](#)
- struct [TmdDeRegNotMitigationLvlReq](#)
- struct [TmdMitigationLvlIndReq](#)

Macros

- #define [MAX_MITIGATION_DEV_LIST_LEN](#) 255
- #define [MAX_MITIGATION_DEV_ID_LEN](#) 255

Functions

- [ULONG](#) [SLQSTmdGetMitigationDevList](#) ([TmdGetMitigationDevListResp](#) *pTmdGetMitigationDevListResp)
- [ULONG](#) [SLQSTmdGetMitigationLvl](#) ([TmdGetMitigationLvlReq](#) *pTmdGetMitigationLvlReq, [TmdGetMitigationLvlResp](#) *pTmdGetMitigationLvlResp)
- [ULONG](#) [SLQSTmdRegNotMitigationLvl](#) ([TmdRegNotMitigationLvlReq](#) *pTmdRegNotMitigationLvlReq)
- [ULONG](#) [SLQSTmdDeRegNotMitigationLvl](#) ([TmdDeRegNotMitigationLvlReq](#) *pTmdDeRegNotMitigationLvlReq)

9.42.1 Detailed Description

Thermal Mitigation Device API function prototypes.

9.42.2 Macro Definition Documentation

9.42.2.1 `#define MAX_MITIGATION_DEV_ID_LEN 255`

9.42.2.2 `#define MAX_MITIGATION_DEV_LIST_LEN 255`

9.42.3 Function Documentation

9.42.3.1 **ULONG** SLQSTmdDeRegNotMitigationLvl (**TmdDeRegNotMitigationLvlReq** * *pTmdDeRegNotMitigationLvlReq*)

Deregisters the notification for mitigation device level changes.

Parameters

<i>pTmdDeRegNotMitigationLvlReq</i> [IN]	<ul style="list-style-type: none">See TmdDeRegNotMitigationLvlReq for more information
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.42.3.2 **ULONG** SLQSTmdGetMitigationDevList (**TmdGetMitigationDevListResp** * *pTmdGetMitigationDevListResp*)

Used by the control point to gets the list of mitigation devices from the remote endpoint.

Parameters

<i>pTmdGetMitigationDevListResp</i> [OUT]	<ul style="list-style-type: none">See TmdGetMitigationDevListResp for more information
---	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.42.3.3 **ULONG** SLQSTmdGetMitigationLvl (**TmdGetMitigationLvlReq** * *pTmdGetMitigationLvlReq*, **TmdGetMitigationLvlResp** * *pTmdGetMitigationLvlResp*)

Gets the thermal mitigation level for the mitigation device

Parameters

<i>pTmdGetMitigationLvlReq</i> [IN]	<ul style="list-style-type: none"> See TmdGetMitigationLvlReq for more information
<i>pTmdGetMitigationLvlResp</i> [OUT]	<ul style="list-style-type: none"> See TmdGetMitigationLvlResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.42.3.4 **ULONG** SLQSTmdRegNotMitigationLvl (**TmdRegNotMitigationLvlReq** * *pTmdRegNotMitigationLvlReq*)

Registers for notification of mitigation device level changes.

Parameters

<i>pTmdRegNotMitigationLvlReq</i> [IN]	<ul style="list-style-type: none"> See TmdRegNotMitigationLvlReq for more information
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.43 qaGobiApiUim.h File Reference

Uim Service API function prototypes.

Data Structures

- struct [UIMPowerDownReq](#)
- struct [fileInfo](#)
- struct [UIMRefreshEvent](#)
- struct [appStatus](#)
- struct [slotInfo](#)
- struct [cardStatus](#)
- struct [hotSwapStatus](#)
- struct [UIMGetCardStatusResp](#)
- struct [UIMSessionInformation](#)
- struct [setPINProtection](#)
- struct [UIMSetPinProtectionReq](#)
- struct [remainingRetries](#)
- struct [encryptedPIN1](#)
- struct [UIMPinResp](#)
- struct [verifyUIMPIN](#)
- struct [UIMVerifyPinReq](#)
- struct [changeUIMPIN](#)
- struct [UIMChangePinReq](#)
- struct [unlockUIMPIN](#)
- struct [UIMUnlockPinReq](#)
- struct [UIMEventRegisterReqResp](#)
- struct [UIMRefreshOKReq](#)
- struct [registerRefresh](#)
- struct [UIMRefreshRegisterReq](#)
- struct [UIMRefreshCompleteReq](#)
- struct [UIMRefreshGetLastEventResp](#)
- struct [UIMRefreshGetLastEventReq](#)
- struct [UIMGetFileAttributesReq](#)
- struct [cardResult](#)
- struct [fileAttributes](#)
- struct [UIMGetFileAttributesResp](#)
- struct [depersonalizationInformation](#)
- struct [UIMDepersonalizationReq](#)
- struct [UIMDepersonalizationResp](#)
- struct [authenticationData](#)
- struct [UIMAuthenticateReq](#)
- struct [authenticateResult](#)
- struct [UIMAuthenticateResp](#)
- struct [readResult](#)
- struct [readTransparentInfo](#)
- struct [UIMReadTransparentReq](#)
- struct [UIMReadTransparentResp](#)
- struct [UIMPowerUpReq](#)
- struct [UIMSlotStatus](#)
- struct [UIMSlotsStatus](#)
- struct [UIMGetSlotsStatusResp](#)
- struct [UIMSwitchSlotReq](#)
- struct [personalizationStatus](#)
- struct [UIMGetConfigurationReq](#)
- struct [UIMGetConfigurationResp](#)

Macros

- `#define MAX_DESCRIPTION_LENGTH 255`
- `#define MAX_CONTENT_LENGTH 1024`
- `#define MAX_NO_OF_APPLICATIONS 10`
- `#define MAX_NO_OF_SLOTS 5`
- `#define MAX_PUK_LENGTH 8`
- `#define MAX_PATH_LENGTH 255`
- `#define MAX_ICCID_LENGTH 255`
- `#define MAX_SLOTS_STATUS 255`
- `#define MAX_ACTIVE_PERS_FEATURES 12`

Functions

- `ULONG SLQSUIReset ()`
- `ULONG SLQSUIPowerDown (UIMPowerDownReq *pUIMPowerDownReq)`
- `ULONG SLQSUIGetCardStatus (UIMGetCardStatusResp *pUIMGetCardStatusResp)`
- `ULONG SLQSUISetPinProtection (UIMSetPinProtectionReq *pUIMSetPinProtectionReq, UIMPinResp *pUIMSetPinProtectionResp)`
- `ULONG SLQSUIVerifyPin (UIMVerifyPinReq *pUIMVerifyPinReq, UIMPinResp *pUIMVerifyPinResp)`
- `ULONG SLQSUIChangePin (UIMChangePinReq *pUIMChangePinReq, UIMPinResp *pUIMChangePin↔Resp)`
- `ULONG SLQSUIUnblockPin (UIMUnblockPinReq *pUIMUnblockPinReq, UIMPinResp *pUIMUnblock↔PinResp)`
- `ULONG SLQSUIEventRegister (UIMEventRegisterReqResp *pUIMEventRegisterReqResp)`
- `ULONG SLQSUIRefreshOK (UIMRefreshOKReq *pUIMRefreshOKReq)`
- `ULONG SLQSUIRefreshRegister (UIMRefreshRegisterReq *pUIMRefreshRegisterReq)`
- `ULONG SLQSUIRefreshComplete (UIMRefreshCompleteReq *pUIMRefreshCompleteReq)`
- `ULONG SLQSUIRefreshGetLastEvent (UIMRefreshGetLastEventReq *pUIMRefreshGetLastEventReq, UIMRefreshGetLastEventResp *pUIMRefreshGetLastEventResp)`
- `ULONG SLQSUIGetFileAttributes (UIMGetFileAttributesReq *pUIMGetFileAttributesReq, UIMGetFile↔AttributesResp *pUIMGetFileAttributesResp)`
- `ULONG SLQSUIDepersonalization (UIMDepersonalizationReq *pUIMDepersonalizationReq, UIM↔DepersonalizationResp *pUIMDepersonalizationResp)`
- `ULONG SLQSUIAuthenticate (UIMAuthenticateReq *pUIMAuthenticateReq, UIMAuthenticateResp *pU↔IMAuthenticateResp)`
- `ULONG SLQSUIReadTransparent (UIMReadTransparentReq *pUIMReadTransparentReq, UIMRead↔TransparentResp *pUIMReadTransparentResp)`
- `ULONG SLQSUIPowerUp (UIMPowerUpReq *pUIMPowerUpReq)`
- `ULONG SLQSUIGetSlotsStatus (UIMGetSlotsStatusResp *pResp)`
- `ULONG SLQSUISwitchSlot (UIMSwitchSlotReq *pReq)`
- `ULONG SLQSUIGetConfiguration (UIMGetConfigurationReq *pUIMGetConfigurationReq, UIMGet↔ConfigurationResp *pUIMGetConfigurationResp)`

9.43.1 Detailed Description

Uim Service API function prototypes.

9.43.2 Macro Definition Documentation

9.43.2.1 `#define MAX_ACTIVE_PERS_FEATURES 12`

9.43.2.2 `#define MAX_CONTENT_LENGTH 1024`

9.43.2.3 `#define MAX_DESCRIPTION_LENGTH 255`

9.43.2.4 `#define MAX_ICCID_LENGTH 255`

9.43.2.5 `#define MAX_NO_OF_APPLICATIONS 10`

9.43.2.6 `#define MAX_NO_OF_SLOTS 5`

9.43.2.7 `#define MAX_PATH_LENGTH 255`

9.43.2.8 `#define MAX_PUK_LENGTH 8`

9.43.2.9 `#define MAX_SLOTS_STATUS 255`

9.43.3 Function Documentation

9.43.3.1 **ULONG** SLQSUIMAuthenticate (**UIMAuthenticateReq** * *pUIMAuthenticateReq*, **UIMAuthenticateResp** * *pUIMAuthenticateResp*)

This API executes the authentication algorithm on the card.

Parameters

<i>pUIMAuthenticateReq</i> [IN]	<ul style="list-style-type: none">See UIMAuthenticateReq for more information.
<i>pUIMAuthenticateResp</i> [OUT]	<ul style="list-style-type: none">See UIMAuthenticateResp for more information.

Returns

`eQCWWAN_ERR_NONE` on success, `eQCWWAN_xxx` error value otherwise

See also

See [qmerrno.h](#) for `eQCWWAN_xxx` error values

Note

Timeout: 30 Secs

This API executes a security command on the card that depends on the card type.
 The response contains the status code received from the card (SW1 and SW2) when the card responded to the read request.
 The client can pass a token in the request to receive the result in a subsequent SLQSUIMAuthenticateCallback

9.43.3.2 ULONG SLQSUIMChangePin (UIMChangePinReq * pUIMChangePinReq, UIMPinResp * pUIMChangePinResp)

This API changes the value of the specified PIN.

Parameters

<i>pUIMChangePinReq</i> [IN]	<ul style="list-style-type: none"> See UIMChangePinReq for more information.
<i>pUIMChangePinResp</i> [OUT]	<ul style="list-style-type: none"> See UIMPinResp for more information.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This API changes the value of the specified PIN.
 The application must pass both the new and the old values of the PIN to complete the operation
 The same PIN can be used by multiple sessions (i.e., the PIN is shared between GSM and RUIM in an ICC card).
 The PIN is automatically set for all the sessions when the API is executed.
 The client can pass a token in the request to receive the result in a subsequent SLQSUIMChangePinCallback.

9.43.3.3 ULONG SLQSUIMDepersonalization (UIMDepersonalizationReq * pUIMDepersonalizationReq, UIMDepersonalizationResp * pUIMDepersonalizationResp)

This API de-activates or unblocks the personalization on the phone.

Parameters

<i>pUIMDepersonalizationReq</i> [IN]	<ul style="list-style-type: none"> See UIMDepersonalizationReq for more information. 	Generated by Doxygen
<i>pUIMDepersonalizationResp</i> [OUT]	<ul style="list-style-type: none"> See UIMDepersonalizationResp for more information. 	

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This API deactivates or unblocks the personalization on the phone.
Each feature can be deactivated/unblocked independently of the
other features.

9.43.3.4 ULONG SLQSUIMEventRegister (UIMEventRegisterReqResp * pUIMEventRegisterReqResp)

This API Registers for event notifications from the card.

Parameters

<i>pUIMEventRegisterReqResp</i> [IN/OUT]	<ul style="list-style-type: none"> See UIMEventRegisterReqResp for more information.
--	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This function registers for event notifications from the card.
The client must verify the mask in the response to determine
which events were registered successfully. Events not supported
correctly are not registered. The client can deregister from
all event notifications by passing "0x00000000" bitmask in the
request.

9.43.3.5 ULONG SLQSUIGetCardStatus (UIMGetCardStatusResp * pUIMGetCardStatusResp)

This API retrieves the current status of the card.

Parameters

<i>pUIMGetCardStatusResp</i>	<ul style="list-style-type: none"> See UIMGetCardStatusResp for more information.
<i>StatusResp</i> [OUT] <small>Generated by Doxygen</small>	

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This function retrieves the current status of the card(activated) and the status of all applications available on the card. The function also returns support information for the hot-swap feature and the status of the switch used to detect a card removal/insertion.

Please use \ref SLQSUIMGetSlotsStatus to retrieves active and inactivate SIMs status.

9.43.3.6 **ULONG SLQSUIMGetConfiguration (UIMGetConfigurationReq * pUIMGetConfigurationReq, UIMGetConfigurationResp * pUIMGetConfigurationResp)**

This API Gets the modem configuration for the UIM module.

Parameters

<i>pUIMGetConfigurationReq[IN]</i>	<ul style="list-style-type: none"> See UIMGetConfigurationReq for more information.
<i>pUIMGetConfigurationResp[OUT]</i>	<ul style="list-style-type: none"> See UIMGetConfigurationResp for more information.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

9.43.3.7 **ULONG SLQSUIMGetFileAttributes (UIMGetFileAttributesReq * pUIMGetFileAttributesReq, UIMGetFileAttributesResp * pUIMGetFileAttributesResp)**

This API retrieves the file attributes for any Elementary File(EF) or Dedicated File(DF) in the card and provides access by the path.

Parameters

<i>pUIMGetFileAttributesReq</i> [IN]	<ul style="list-style-type: none">• See UIMGetFileAttributesReq for more information.
<i>pUIMGetFileAttributesResp</i> [OUT]	<ul style="list-style-type: none">• See UIMGetFileAttributesResp for more information.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This API retrieves the file attributes for any Elementary File(EF) or Dedicated File(DF) in the card and provides access by the path. The response contains the status code received from the card (SW1 and SW2) when the card responded to the select request. The client can pass a token in the request to receive the result in a subsequent SLQSUIMGetFileAttributesCallback.

9.43.3.8 ULONG SLQSUIMGetSlotsStatus (UIMGetSlotsStatusResp * pResp)

This API Retrieves the current of the physical and logical slots.

Parameters

<i>pResp</i> [OUT]	<ul style="list-style-type: none">• See UIMGetSlotsStatusResp for more information.
--------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

9.43.3.9 ULONG SLQSUIMPowerDown (UIMPowerDownReq * pUIMPowerDownReq)

This API powers down the SIM card.

Parameters

<i>pUIMPowerDownReq</i> [IN]	<ul style="list-style-type: none">See UIMPowerDownReq for more information.
------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This function powers down the card.
This is usually performed when the phone is switched off or when it is set to Airplane mode.

9.43.3.10 ULONG SLQSUIMPowerUp (UIMPowerUpReq * pUIMPowerUpReq)

This API powers up the SIM card.

Parameters

<i>pUIMPowerUpReq</i> [IN]	<ul style="list-style-type: none">See UIMPowerUpReq for more information.
----------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This function powers up the card.
This is usually performed when the phone is switched off or when it is set to Airplane mode.

9.43.3.11 **ULONG** SLQSUIReadTransparent (**UIMReadTransparentReq** * *pUIMReadTransparentReq*, **UIMReadTransparentResp** * *pUIMReadTransparentResp*)

This API executes the Read Transparent algorithm on the card.

Parameters

<i>pUIMReadTransparentReq</i> [IN]	<ul style="list-style-type: none"> See UIMReadTransparentReq for more information.
<i>pUIMReadTransparentResp</i> [OUT]	<ul style="list-style-type: none"> See UIMReadTransparentResp for more information.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This API Provides read access to any transparent file in the card and provides access by the path.
The response contains the status code received from the card (SW1 and SW2) when the card responded to the read request.
The client can pass a token in the request to receive the result in a subsequent QMI_UIM_READ_TRANSPARENT_IND indication.

9.43.3.12 **ULONG** SLQSUIRefreshComplete (**UIMRefreshCompleteReq** * *pUIMRefreshCompleteReq*)

This API invoked when the client has finished the Refresh procedure.

Parameters

<i>pUIMRefreshCompleteReq</i> [IN]	<ul style="list-style-type: none"> See UIMRefreshCompleteReq for more information.
------------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This function is invoked when the client has finished the Refresh procedure (has reread all the cached files) and communicates this to the modem. This function enables the terminal response to be sent to the card

9.43.3.13 **ULONG SLQSUIRefreshGetLastEvent (UIMRefreshGetLastEventReq * pUIMRefreshGetLastEventReq, UIMRefreshGetLastEventResp * pUIMRefreshGetLastEventResp)**

This API provides the ability to retrieve the last refresh event.

Parameters

<i>pUIMRefreshGetLastEventReq</i> [IN]	<ul style="list-style-type: none"> See UIMRefreshGetLastEventReq for more information.
<i>pUIMRefreshGetLastEventResp</i> [OUT]	<ul style="list-style-type: none"> See UIMRefreshGetLastEventResp for more information.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This function provides the ability to retrieve the last refresh event. The event information is usually passed in as an indication from the QMI to the application and is saved by the application at that time. If the event information is not saved, the client can retrieve the last refresh event.
Details regarding the Refresh procedure (i.e., the stages and actions that an application must complete) are described in document: 80-VM566-1 (NAA Refresh High Level Guide)

9.43.3.14 **ULONG SLQSUIRefreshOK (UIMRefreshOKReq * pUIMRefreshOKReq)**

This API Enables the client to indicate whether it is OK to start the Refresh procedure.

Parameters

<i>pUIMRefreshOKReq</i> [IN]	<ul style="list-style-type: none"> Consist of parameters for SLQSUIRefreshOK. Please see /ref UIMRefreshOKReq for details.
------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This function enables the client to indicate whether it is OK to start the Refresh procedure. This command is used only after a refresh event is received, which indicates the need to vote.

9.43.3.15 ULONG SLQSUIMRefreshRegister (UIMRefreshRegisterReq * pUIMRefreshRegisterReq)

This API Registers for file change notifications triggered by the card.

Parameters

<i>pUIMRefreshRegisterReq</i> [IN]	<ul style="list-style-type: none"> See UIMRefreshRegisterReq for more information.
------------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This function registers for file change notifications triggered by the card. The client can specify a list of files. The client is notified only when one of the files is modified by the Refresh procedure. This function can be invoked multiple times for each session type. If the function is invoked twice with the same session type, the new values overwrite the previous values. The client can also use this function to stop receiving indications of the refresh. This API should be invoked prior to the invocation of the SLQSUIMSetRefreshCallBack for the events to be registered.

9.43.3.16 ULONG SLQSUIMReset ()

This API resets the issuing control points state kept by the service.

Parameters

None	
------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

The list of events to which the client is registered is emptied.
The client must register again using the SLQSUIEventReg API
to start receiving the events again. This would mean that the
callback registrations would be reset after this API.

9.43.3.17 ULONG SLQSUISetPinProtection (UIMSetPinProtectionReq * pUIMSetPinProtectionReq, UIMPinResp * pUIMSetPinProtectionResp)

This API enables or disables the protection of the UIM contents by a specific PIN.

Parameters

<i>pUIMSetPinProtectionReq</i> [IN]	<ul style="list-style-type: none">See UIMSetPinProtectionReq for more information.
<i>pUIMSetPinProtectionResp</i> [OUT]	<ul style="list-style-type: none">See UIMPinResp for more information.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This API enables or disables the protection of UIM contents by a specific PIN.
The same PIN can be used by multiple sessions (i.e., the PIN is shared between GSM and RUIM in an ICC card).
The PIN is automatically set for all the sessions when the API is executed.
The client can pass a token in the request to receive the result in a subsequent SLQSUISetPinProtectionCallback indication.

9.43.3.18 ULONG SLQSUIMSwitchSlot (UIMSwitchSlotReq * *pReq*)

This API Switches the binding between a logical slot and a physical slot.

Parameters

<i>pReq</i> [IN]	<ul style="list-style-type: none">• See UIMSwitchSlotReq for more information.
------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

Please wait at least one second before this API call again.

9.43.3.19 ULONG SLQSUIMUnblockPin (UIMUnblockPinReq * *pUIMUnblockPinReq*, UIMPinResp * *pUIMUnblockPinResp*)

This API unblocks a blocked PIN using the PUK code.

Parameters

<i>pUIMUnblockPinReq</i> [IN]	<ul style="list-style-type: none">• See UIMUnblockPinReq for more information.
<i>pUIMUnblockPinResp</i> [OUT]	<ul style="list-style-type: none">• See UIMPinResp for more information.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This API unblocks a blocked PIN using the PUK code.
 The client must pass PUK1 to unblock PIN1 or PUK2 to unblock PIN2.
 The same PIN can be used by multiple sessions (i.e., the PIN is shared between GSM and RUIM in an ICC card).
 The PIN is automatically set for all the sessions when the API is executed.
 The client can pass a token in the request to receive the result in a subsequent SLQSUIMUnlockPinCallback.

9.43.3.20 ULONG SLQSUIMVerifyPin (UIMVerifyPinReq * *pUIMVerifyPinReq*, UIMPinResp * *pUIMVerifyPinResp*)

This API verifies the PIN before the card content is accessed.

Parameters

<i>pUIMVerifyPinReq</i> <i>Req[IN]</i>	<ul style="list-style-type: none"> See UIMVerifyPinReq for more information.
<i>pUIMVerifyPinResp</i> <i>Resp[OUT]</i>	<ul style="list-style-type: none"> See UIMPinResp for more information.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This API verifies the PIN before the card content is accessed.
 The same PIN can be used by multiple sessions (i.e., the PIN is shared between GSM and RUIM in an ICC card).
 The PIN is automatically set for all the sessions when the API is executed.
 The client can pass a token in the request to receive the result in a subsequent SLQSUIMVerifyPinCallback.

9.44 qaGobiApiVoice.h File Reference

Voice Service API function prototypes.

Data Structures

- struct [USSInfo](#)
- struct [UUSInfo](#)
- struct [CUGInfo](#)
- struct [calledPartySubAdd](#)
- struct [voiceCallRequestParams](#)
- struct [alphaIDInfo](#)
- struct [ccSUPSType](#)
- struct [voiceCallResponseParams](#)
- struct [callFwdTypeAndPlan](#)
- struct [voiceSetSUPSServiceReq](#)
- struct [voiceSetSUPSServiceResp](#)
- struct [airTimer](#)
- struct [roamTimer](#)
- struct [prefVoiceSO](#)
- struct [voiceSetConfigReq](#)
- struct [voiceSetConfigResp](#)
- struct [voiceAnswerCall](#)
- struct [CLIRResp](#)
- struct [voiceGetCLIRResp](#)
- struct [CLIPResp](#)
- struct [voiceGetCLIPResp](#)
- struct [voiceGetCallWaitInfo](#)
- struct [voiceGetCallBarringReq](#)
- struct [voiceGetCallBarringResp](#)
- struct [voiceGetCallFWReq](#)
- struct [callFWInfo](#)
- struct [callFWExtInfo](#)
- struct [getCallFWInfo](#)
- struct [getCallFWExtInfo](#)
- struct [voiceGetCallFWResp](#)
- struct [voiceSetCallBarringPwdInfo](#)
- struct [voiceSetCallBarringPwdResp](#)
- struct [callInfo](#)
- struct [remotePartyNum](#)
- struct [remotePartyName](#)
- struct [connectNumInfo](#)
- struct [diagInfo](#)
- struct [voiceCallInfoReq](#)
- struct [voiceCallInfoResp](#)
- struct [getAllCallInformation](#)
- struct [getAllCallRmtPtyNum](#)
- struct [getAllCallRmtPtyName](#)
- struct [allCallsUUSInfo](#)
- struct [allCallsAlphaIDInfo](#)
- struct [allCallsDiagInfo](#)
- struct [peerNumberInfo](#)
- struct [arrCallInfo](#)
- struct [arrRemotePartyNum](#)
- struct [arrRemotePartyName](#)
- struct [arrAlertingType](#)
- struct [arrUUSInfo](#)
- struct [arrSvcOption](#)
- struct [arrCallEndReason](#)

- struct [arrAlphaID](#)
- struct [arrConnectPartyNum](#)
- struct [arrDiagInfo](#)
- struct [arrCalledPartyNum](#)
- struct [arrRedirPartyNum](#)
- struct [arrAlertingPattern](#)
- struct [voiceGetAllCallInfo](#)
- struct [voiceManageCallsReq](#)
- struct [voiceManageCallsResp](#)
- struct [burstDTMFInfo](#)
- struct [DTMFLengths](#)
- struct [voiceBurstDTMFInfo](#)
- struct [voiceContDTMFInfo](#)
- struct [voiceStopContDTMFInfo](#)
- struct [voiceFlashInfo](#)
- struct [voiceSetPrefPrivacy](#)
- struct [voiceIndicationRegisterInfo](#)
- struct [DTMFInfo](#)
- struct [SUPSInfo](#)
- struct [newPwdData](#)
- struct [COLPResp](#)
- struct [COLRResp](#)
- struct [CNAPResp](#)
- struct [voiceGetConfigReq](#)
- struct [curAMRConfig](#)
- struct [voiceGetConfigResp](#)
- struct [voiceOrigUSSDNoWaitInfo](#)
- struct [voiceBindSubscriptionInfo](#)
- struct [voiceALSSetLineSwitchInfo](#)
- struct [voiceALSSelectLineInfo](#)
- struct [voiceGetCOLPResp](#)
- struct [voiceGetCOLRResp](#)
- struct [voiceGetCNAPResp](#)
- struct [USSResp](#)
- struct [USSDRespFNetwork](#)

Macros

- #define [MAXUSSDLENGTH](#) 182
- #define [MAX_CALL_NO_LEN](#) 81
- #define [MAX_DESCRIPTION_LENGTH](#) 255
- #define [PASSWORD_LENGTH](#) 4
- #define [MAX_NO_OF_CALLS](#) 20

Enumerations

- enum [serviceClassInformation](#) {
[VOICE_SUPS_SRV_CLASS_NONE](#) = 0x00,
[VOICE_SUPS_SRV_CLASS_VOICE](#) = 0x01,
[VOICE_SUPS_SRV_CLASS_DATA](#) = 0x02,
[VOICE_SUPS_SRV_CLASS_FAX](#) = 0x04,
[VOICE_SUPS_SRV_CLASS_SMS](#) = 0x08,
[VOICE_SUPS_SRV_CLASS_DATACIRCUITSYNC](#) = 0x10,
[VOICE_SUPS_SRV_CLASS_DATACIRCUITASYNC](#) = 0x20,
[VOICE_SUPS_SRV_CLASS_PACKETACCESS](#) = 0x40,
[VOICE_SUPS_SRV_CLASS_PADACCESS](#) = 0x80 }

Functions

- [ULONG OriginateUSSD](#) ([BYTE](#) *pInfo)
- [ULONG AnswerUSSD](#) ([BYTE](#) *pInfo)
- [ULONG CancelUSSD](#) ()
- [ULONG SLQSVoiceDialCall](#) ([voiceCallRequestParams](#) *pCallRequestParams, [voiceCallResponseParams](#) *pCallResponseParams)
- [ULONG SLQSVoiceEndCall](#) ([BYTE](#) *pCallId)
- [ULONG SLQSVoiceSetSUPSService](#) ([voiceSetSUPSServiceReq](#) *pVoiceSetSUPSServiceReq, [voiceSetSUPSServiceResp](#) *pVoiceSetSUPSServiceResp)
- [ULONG SLQSVoiceSetConfig](#) ([voiceSetConfigReq](#) *pVoiceSetConfigReq, [voiceSetConfigResp](#) *pVoiceSetConfigResp)
- [ULONG SLQSVoiceAnswerCall](#) ([voiceAnswerCall](#) *pVoiceAnswerCall)
- [ULONG SLQSVoiceGetCLIR](#) ([voiceGetCLIRResp](#) *pVoiceGetCLIRResp)
- [ULONG SLQSVoiceGetCLIP](#) ([voiceGetCLIPResp](#) *pVoiceGetCLIPResp)
- [ULONG SLQSVoiceGetCallWaiting](#) ([voiceGetCallWaitInfo](#) *pVoiceGetCallWaitInfo)
- [ULONG SLQSVoiceGetCallBarring](#) ([voiceGetCallBarringReq](#) *pVoiceGetCallBarringReq, [voiceGetCallBarringResp](#) *pVoiceGetCallBarringResp)
- [ULONG SLQSVoiceGetCallForwardingStatus](#) ([voiceGetCallFWReq](#) *pVoiceGetCallFWReq, [voiceGetCallFWResp](#) *pVoiceGetCallFWResp)
- [ULONG SLQSVoiceSetCallBarringPassword](#) ([voiceSetCallBarringPwdInfo](#) *pVoiceSetCallBarringPwdInfo, [voiceSetCallBarringPwdResp](#) *pSetCallBarringPwdResp)
- [ULONG SLQSVoiceGetCallInfo](#) ([voiceCallInfoReq](#) *pGetCallInfoReq, [voiceCallInfoResp](#) *pGetCallInfoResp)
- [ULONG SLQSVoiceGetAllCallInfo](#) ([voiceGetAllCallInfo](#) *pGetAllCallInfo)
- [ULONG SLQSVoiceManageCalls](#) ([voiceManageCallsReq](#) *pVoiceManageCallsReq, [voiceManageCallsResp](#) *pVoiceManageCallsResp)
- [ULONG SLQSVoiceBurstDTMF](#) ([voiceBurstDTMFInfo](#) *pBurstDTMFInfo)
- [ULONG SLQSVoiceStartContDTMF](#) ([voiceContDTMFInfo](#) *pContDTMFInfo)
- [ULONG SLQSVoiceStopContDTMF](#) ([voiceStopContDTMFInfo](#) *pVoiceStopContDTMFInfo)
- [ULONG SLQSVoiceSendFlash](#) ([voiceFlashInfo](#) *pFlashInfo)
- [ULONG SLQSVoiceSetPreferredPrivacy](#) ([voiceSetPrefPrivacy](#) *pSetPrefPrivacy)
- [ULONG SLQSVoiceIndicationRegister](#) ([voiceIndicationRegisterInfo](#) *pVoiceIndicationRegisterInfo)
- [ULONG SLQSVoiceGetConfig](#) ([voiceGetConfigReq](#) *pVoiceGetConfigReq, [voiceGetConfigResp](#) *pVoiceGetConfigResp)
- [ULONG SLQSVoiceOrigUSSDNoWait](#) ([voiceOrigUSSDNoWaitInfo](#) *pVoiceOrigUSSDNoWaitInfo)
- [ULONG SLQSVoiceBindSubscription](#) ([voiceBindSubscriptionInfo](#) *pVoiceBindSubscriptionInfo)
- [ULONG SLQSVoiceALSSetLineSwitching](#) ([voiceALSSetLineSwitchInfo](#) *pVoiceALSSetLineSwitchInfo)
- [ULONG SLQSVoiceALSSelectLine](#) ([voiceALSSelectLineInfo](#) *pVoiceALSSelectLineInfo)
- [ULONG SLQSVoiceGetCOLP](#) ([voiceGetCOLPResp](#) *pVoiceGetCOLPResp)
- [ULONG SLQSVoiceGetCOLR](#) ([voiceGetCOLRResp](#) *pVoiceGetCOLRResp)
- [ULONG SLQSVoiceGetCNAP](#) ([voiceGetCNAPResp](#) *pVoiceGetCNAPResp)
- [ULONG SLQSVoiceOriginateUSSD](#) ([struct USSInfo](#) *pReq, [struct USSResp](#) *pResp)

9.44.1 Detailed Description

Voice Service API function prototypes.

9.44.2 Macro Definition Documentation

9.44.2.1 `#define MAX_CALL_NO_LEN 81`

9.44.2.2 `#define MAX_DESCRIPTION_LENGTH 255`

9.44.2.3 `#define MAX_NO_OF_CALLS 20`

9.44.2.4 `#define MAXUSSDLENGTH 182`

9.44.2.5 `#define PASSWORD_LENGTH 4`

9.44.3 Enumeration Type Documentation

9.44.3.1 `enum serviceClassInformation`

Service Class information

Enumerator

`VOICE_SUPS_SRV_CLASS_NONE`
`VOICE_SUPS_SRV_CLASS_VOICE`
`VOICE_SUPS_SRV_CLASS_DATA`
`VOICE_SUPS_SRV_CLASS_FAX`
`VOICE_SUPS_SRV_CLASS_SMS`
`VOICE_SUPS_SRV_CLASS_DATACIRCUITSYNC`
`VOICE_SUPS_SRV_CLASS_DATACIRCUITASYNC`
`VOICE_SUPS_SRV_CLASS_PACKETACCESS`
`VOICE_SUPS_SRV_CLASS_PADACCESS`

9.44.4 Function Documentation

9.44.4.1 `ULONG AnswerUSSD (BYTE * pInfo)`

Responds to a USSD request from the network.

Parameters

<i>pInfo</i> [IN]	<ul style="list-style-type: none">• USS information
-------------------	---

Returns

`eQCWWAN_ERR_NONE` on success, `eQCWWAN_xxx` error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 5 mins

9.44.4.2 ULONG CancelUSSD ()

Cancels an in-progress USSD operation.

Parameters

<i>None</i>	
-------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 30 Secs

9.44.4.3 ULONG OriginateUSSD (BYTE * pInfo)

Initiates a USSD operation.

Parameters

<i>pInfo[IN]</i>	<ul style="list-style-type: none">• USS information• See USSInfo for more details
------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
Timeout: 5 mins

9.44.4.4 ULONG SLQSOriginateUSSD (struct USSInfo * *pReq*, struct USSResp * *pResp*)

Initiates a USSD session.

Parameters

<i>pReq</i>	[IN] <ul style="list-style-type: none">• USS information• See USSInfo for more details
<i>pResp</i>	[OUT] <ul style="list-style-type: none">• USS information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
Device Supported: MC83x5
Timeout: 5 mins

9.44.4.5 ULONG SLQSVoiceALSSelectLine (voiceALSSelectLineInfo * *pVoiceALSSelectLineInfo*)

This API allows the user to select the preferred line.

Parameters

<i>pVoiceALSSelectLineInfo</i> [IN]	<ul style="list-style-type: none">• See voiceALSSelectLineInfo for more information.
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 30 Secs

This API allows the user to select the preferred line, and the status is updated on the card. The API is supported only for specific SIM/USIMs that support alternate line service. This command is applicable only in 3GPP devices. A No Effect error is returned if the update on the card fails.

9.44.4.6 ULONG SLQSVoiceALSSetLineSwitching (voiceALSSetLineSwitchInfo * pVoiceALSSetLineSwitchInfo)

This API sets the line switch setting on the card.

Parameters

<i>pVoiceALSSetLineSwitchInfo</i> [IN]	<ul style="list-style-type: none">See voiceALSSetLineSwitchInfo for more information.
--	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 30 Secs

This API sets a line to be switchable or unswitchable, and the switch status is updated on the card. The API is supported only for specific SIM/USIMs that support alternate line service. This command is applicable only in 3GPP devices. A No Effect error is returned if the update on the card fails.

9.44.4.7 ULONG SLQSVoiceAnswerCall (voiceAnswerCall * pVoiceAnswerCall)

Answers an incoming voice call.

Parameters

<i>pVoiceAnswerCall</i> [IN/OUT]	<ul style="list-style-type: none">Pointer to structure of voiceAnswerCall<ul style="list-style-type: none">See voiceAnswerCall for more information
----------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 10 Secs

This API is used to answer an incoming voice call when the incoming voice call is the only call present at that time. If there are other calls while an incoming call (waiting call) is received, API "SLQSVoiceSendFlash" can be used case of 3GPP2(CDMA) and API "SLQSVoiceManageCalls" in the case of 3GPP(UMTS). If the result indicates success, the device has started the requested operation and it does not mean that the call has been answered. "SLQSVoiceSetAllCallStatusCallback" can be subscribed to check the call Information/State.

9.44.4.8 ULONG SLQSVoiceBindSubscription (voiceBindSubscriptionInfo * pVoiceBindSubscriptionInfo)

This API binds a subscription type to a specific voice client ID.

Parameters

<i>pVoiceBindSubscriptionInfo</i> [IN]	<ul style="list-style-type: none">See voiceBindSubscriptionInfo for more information.
--	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

Some versions of the modem support the Dual SIM feature. With this feature the modem can register with two different cellular networks simultaneously. Each network registration is associated with a different subscription, e.g., phone number, such that the modem appears to the network to be two different users. By default, the Voice client is bound to the primary subscription. This command allows the Voice client to change this binding. After receiving a successful response to this command, all future commands sent by the client will affect the newly bound subscription only.

9.44.4.9 ULONG SLQSVoiceBurstDTMF (voiceBurstDTMFInfo * pBurstDTMFInfo)

Sends a burst Dual-Tone Multi frequency (DTMF) (applicable only for 3GPP2)

Parameters

<i>pBurstDTMF</i> ↔ <i>Info</i> [IN/OUT]	<ul style="list-style-type: none"> • Structure containing parameters of burst DTMF. • See voiceBurstDTMFInfo for more information
---	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA

Timeout: 30 Secs

Applicable only for 3GPP2. This API sends a burst DTMF. If API result indicates success, this means the device has started the requested operation. It does not mean that the burst DTMF request has been sent to the network. A burst DTMF request is sent to the current active/alerting call when CallId is set to 0xFF. This API is applicable only in 3GPP2.

9.44.4.10 ULONG SLQSVoiceDialCall (voiceCallRequestParams * pCallRequestParams, voiceCallResponseParams * pCallResponseParams)

Originates a voice call (MO call).

Parameters

<i>pCallRequest</i> ↔ <i>Params</i> [IN]	<ul style="list-style-type: none"> • Pointer to structure of voiceCallRequestParams <ul style="list-style-type: none"> – See voiceCallRequestParams for more information
<i>pCall</i> ↔ <i>Response</i> ↔ <i>Params</i> [OUT]	<ul style="list-style-type: none"> • Pointer to structure of voiceCallResponseParams <ul style="list-style-type: none"> – See voiceCallResponseParams for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This API originates a voice call (MO). If the function returns success with a call_id, the device has started the requested operation. It does not mean that the call has been connected. SLQSVoiceSetAllCallStatus↔ Callback() callback can be subscribed to learn if the call was successful.

9.44.4.11 ULONG SLQSVoiceEndCall (BYTE * *pCallId*)

This message ends a voice call

Parameters

<i>pCallId</i>	[IN/OUT] <ul style="list-style-type: none">Unique call identifier for the call that must be ended
----------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

If the function returns success, the device has started the requested operation. It does not mean that the call has been ended. The application should always process the SLQSVoiceSetAllCallStatusCallback() callback to learn if the call was ended.

9.44.4.12 ULONG SLQSVoiceGetAllCallInfo (voiceGetAllCallInfo * *pGetAllCallInfo*)

This API queries the information associated with all the calls originating or terminating from a particular device.

Parameters

<i>pGetAllCallInfo</i>	[OUT] <ul style="list-style-type: none">See voiceGetAllCallInfo for more information.
------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This command is used by the control point to get information of all the calls to and fro from the device in progress. The information keeps on updating constantly, as the state of a call changes example, from incoming to conversation to terminated.

This API requires a firmware with at least voice 2.0 support.

9.44.4.13 ULONG SLQSVoiceGetCallBarring (voiceGetCallBarringReq * pVoiceGetCallBarringReq, voiceGetCallBarringResp * pVoiceGetCallBarringResp)

Queries the status of Call Barring Supplementary Service (applicable only for 3GPP).

Parameters

<i>pVoiceGetCallBarringReq</i> [IN]	<ul style="list-style-type: none"> Pointer to structure of voiceGetCallBarringReq <ul style="list-style-type: none"> See voiceGetCallBarringReq for more information
<i>pVoiceGetCallBarringResp</i> [OUT]	<ul style="list-style-type: none"> Pointer to structure of voiceGetCallBarringResp <ul style="list-style-type: none"> See voiceGetCallBarringResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 30 Secs

This API queries the status of the call barring supplementary service, i.e., to find whether the call barring supplementary service is active and, if active, for which service classes it is active. The request is a blocking request, i.e., the response is sent only after confirmation is received from the network. The maximum time it takes for the response to be sent is approximately 30 sec. This API is applicable only in 3GPP devices.

9.44.4.14 ULONG SLQSVoiceGetCallForwardingStatus (voiceGetCallFWReq * pVoiceGetCallFWReq, voiceGetCallFWResp * pVoiceGetCallFWResp)

Queries the status of Call Forwarding Supplementary Service.

Parameters

<i>pVoiceGetCallFWReq</i> [IN]	<ul style="list-style-type: none"> Pointer to structure of voiceGetCallFWReq <ul style="list-style-type: none"> See voiceGetCallFWReq for more information
<i>pVoiceGetCallFWResp</i> [OUT]	<ul style="list-style-type: none"> Pointer to structure of voiceGetCallFWResp <ul style="list-style-type: none"> See voiceGetCallFWResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 30 Secs

This API queries the status of the call forwarding supplementary service, i.e., to find whether the call forwarding supplementary service is active and, if active, for which service classes and call forwarding number it is active. The request is a blocking request, i.e., the response is sent only after confirmation is received from the network. The maximum time it takes for the response to be sent is approximately 30 sec. This API is applicable only in 3GPP devices.

9.44.4.15 `ULONG SLQSVoiceGetCallInfo (voiceCallInfoReq * pGetCallInfoReq, voiceCallInfoResp * pGetCallInfoResp)`

This API queries the information associated with a call and gives information about a particular call whose call Id is sent in as request.

Parameters

<i>pGetCallInfoReq</i> <i>Req[IN]</i>	<ul style="list-style-type: none"> See voiceCallInfoReq for more information.
<i>pGetCallInfoResp</i> <i>Resp[OUT]</i>	<ul style="list-style-type: none"> See voiceCallInfoResp for more information.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

If no voice call is in progress or an invalid call_id is sent in the request, an error is returned as the response. This API requires a firmware with atleast voice 2.0 support.

9.44.4.16 `ULONG SLQSVoiceGetCallWaiting (voiceGetCallWaitInfo * pVoiceGetCallWaitInfo)`

Queries the status of Call Waiting Supplementary Service (applicable only for 3GPP).

Parameters

<i>pVoiceGetCallWaitInfo</i> [IN/OUT]	<ul style="list-style-type: none"> • Pointer to structure of voiceGetCallWaitInfo <ul style="list-style-type: none"> – See voiceGetCallWaitInfo for more information
---------------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 30 Secs

This API queries the status of the call waiting supplementary service, i.e., to find whether the call waiting supplementary service is active. The request is a blocking request, i.e., the response is sent only after confirmation is received from the network. The maximum time it takes for the response to be sent is approximately 30 sec. This API is applicable only in 3GPP devices.

9.44.4.17 ULONG SLQSVoiceGetCLIP (voiceGetCLIPResp * pVoiceGetCLIPResp)

Queries the status of the Calling Line Identification Presentation (CLIP) supplementary service (applicable only for 3GPP).

Parameters

<i>pVoiceGetCLIPResp</i> [OUT]	<ul style="list-style-type: none"> • Pointer to structure of voiceGetCLIPResp <ul style="list-style-type: none"> – See voiceGetCLIPResp for more information
--------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 30 Secs

This API queries the status of the CLIP supplementary service. The active_status field is only applicable when provision_status is PROVISIONED, i.e., there is not any case where provision_status is NOT_PROVISIONED and active_status is ACTIVE. The request is a blocking request, i.e., the response is sent only after confirmation is received from the network. The maximum time it takes for the response to be sent is approximately 30 sec. This API is applicable only in 3GPP devices.

9.44.4.18 ULONG SLQSVoiceGetCLIR (voiceGetCLIRResp * pVoiceGetCLIRResp)

Queries the status of the Calling Line Identification Restriction (CLIR) supplementary service (applicable only for 3GPP).

Parameters

<i>pVoiceGetCLIRResp[OUT]</i>	<ul style="list-style-type: none"> • Pointer to structure of voiceGetCLIRResp <ul style="list-style-type: none"> – See voiceGetCLIRResp for more information
-------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 30 Secs

This API queries the status of the CLIR supplementary service. The active_status field is only applicable when provision_status is PROVISIONED, i.e., there is not any case where provision_status is NOT_PROVISIONED and active_status is ACTIVE. The request is a blocking request, i.e., the response is sent only after confirmation is received from the network. The maximum time it takes for the response to be sent is approximately 30 sec. This API is applicable only in 3GPP devices.

9.44.4.19 ULONG SLQSVoiceGetCNAP (voiceGetCNAPResp * pVoiceGetCNAPResp)

Queries the status of the Calling Name Presentation(CNAP) supplementary service (applicable only for 3GPP).

Parameters

<i>pVoiceGetCNAPResp[OUT]</i>	<ul style="list-style-type: none"> • Pointer to structure of voiceGetCNAPResp <ul style="list-style-type: none"> – See voiceGetCNAPResp for more information
-------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
 Timeout: 30 Secs

This API queries the status of the CNAP supplementary service. A response indicates whether CNAP is active/inactive and provisioned/not provisioned in the network. The active_status field is only applicable when provision_status is PROVISIONED, i.e., there is not any case where provision_status is NOT_PROVISIONED and active_status is ACTIVE. This API is applicable only in 3GPP devices.

9.44.4.20 ULONG SLQSVoiceGetCOLP (voiceGetCOLPResp * pVoiceGetCOLPResp)

Queries the status of the Connected Line Identification Presentation (COLP) supplementary service (applicable only for 3GPP).

Parameters

<i>pVoiceGetCOLPResp[OUT]</i>	<ul style="list-style-type: none"> • Pointer to structure of voiceGetCOLPResp <ul style="list-style-type: none"> – See voiceGetCOLPResp for more information
-------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
 Timeout: 30 Secs

This API queries the status of the COLP supplementary service. A response indicates whether COLP is active/inactive and provisioned/not provisioned in the network. The active_status field is only applicable when provision_status is PROVISIONED, i.e., there is not any case where provision_status is NOT_PROVISIONED and active_status is ACTIVE. This API is applicable only in 3GPP devices.

9.44.4.21 ULONG SLQSVoiceGetCOLR (voiceGetCOLRResp * pVoiceGetCOLRResp)

Queries the status of the Connected Line Identification Restriction (COLR) supplementary service (applicable only for 3GPP).

Parameters

<i>pVoiceGetCOLRResp[OUT]</i>	<ul style="list-style-type: none"> • Pointer to structure of voiceGetCOLRResp <ul style="list-style-type: none"> – See voiceGetCOLRResp for more information
-------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
Timeout: 30 Secs

This API queries the status of the COLR supplementary service. A response indicates whether COLR is active/inactive and provisioned/not provisioned in the network. The active_status field is only applicable when provision_status is PROVISIONED, i.e., there is not any case where provision_status is NOT_PROVISIONED and active_status is ACTIVE. This API is applicable only in 3GPP devices.

9.44.4.22 ULONG SLQSVoiceGetConfig (voiceGetConfigReq * pVoiceGetConfigReq, voiceGetConfigResp * pVoiceGetConfigResp)

This API retrieves various configuration parameters that control the modem behavior related to circuit switched services.

Parameters

<i>pVoiceGetConfigReq</i>	<ul style="list-style-type: none"> Structure containing Get Config request parameters. <ul style="list-style-type: none"> See voiceGetConfigReq for more information.
<i>pVoiceGetConfigResp</i>	<ul style="list-style-type: none"> Structure containing Get Config response parameters. <ul style="list-style-type: none"> See voiceGetConfigResp for more information.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs
Any invalid value in a request message causes the service point to reject the message without retrieving any configuration information.

9.44.4.23 ULONG SLQSVoiceIndicationRegister (voiceIndicationRegisterInfo * pVoiceIndicationRegisterInfo)

Sets the registration state for different QMI_VOICE indications for the requesting control point

Parameters

<i>pVoiceIndicationRegisterInfo</i> [IN]	<ul style="list-style-type: none"> Structure containing Indication Register Information. <ul style="list-style-type: none"> See voiceIndicationRegisterInfo for more information.
--	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 10 Secs

This API is used by a device to register/deregister for different QMI_VOICE indications. The device's registration state variables that control registration for indications will be modified to reflect the settings indicated in the request message. At least one optional parameter must be present in the request.

9.44.4.24 ULONG SLQSVoiceManageCalls (voiceManageCallsReq * pVoiceManageCallsReq, voiceManageCallsResp * pVoiceManageCallsResp)

Manages the calls by using the supplementary service applicable during the call. In cases of successful API completion if the state of any call is changed, it is indicated using CallBack SLQSVoiceSetAllCallStatusCallBack. If there are other calls while an incoming voice call (waiting call) is received, this API is used to answer the call. This API is applicable only in "3GPP devices".

Parameters

<i>pVoiceManageCallsReq</i> [IN]	<ul style="list-style-type: none"> Request structure of to manage calls.
<i>pVoiceManageCallsResp</i> [OUT]	<ul style="list-style-type: none"> Response Structure to manage Calls

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 10 Secs

Applicable only for "3GPP"

9.44.4.25 ULONG SLQSVoiceOrigUSSDNoWait (voiceOrigUSSDNoWaitInfo * pVoiceOrigUSSDNoWaitInfo)

This API initiates a USSD operation such that the response for this request is returned immediately and the data is returned via an indication.

Parameters

<i>pVoiceOrigUSSDNoWaitInfo</i> [IN]	<ul style="list-style-type: none">See voiceOrigUSSDNoWaitInfo for more information.
--------------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 30 Secs

This API starts a new USSD operation. The response to the request is sent immediately. The response result is sent to the client via the SLQSVoiceOrigUSSDNoWaitCallback. This command is applicable only in 3GPP devices.

9.44.4.26 ULONG SLQSVoiceSendFlash (voiceFlashInfo * pFlashInfo)

This API sends a simple flash message. Applicable only for 3GPP2 devices.

Parameters

<i>pFlashInfo</i> [IN/OUT]	<ul style="list-style-type: none">See voiceFlashInfo for more information.
----------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA

Timeout: 10 Secs

If success, it only means the device has started the requested operation and not that the Flash has been sent. If the optional parameter Flash Type is not set, the default flash type is assumed to be a simple flash. If the parameter Flash Type is set to 1 the call ID corresponding to it is either an incoming or waiting call's call ID. If the parameter Flash Type is set to 2 the call ID corresponding to it is a held call's call ID. A Flash request is sent to the appropriate call when call_id is set to 0xFF.

9.44.4.27 ULONG SLQSVoiceSetCallBarringPassword (voiceSetCallBarringPwdInfo * pVoiceSetCallBarringPwdInfo, voiceSetCallBarringPwdResp * pSetCallBarringPwdResp)

Sets a Call Barring Password (applicable only for 3GPP).

Parameters

<i>pVoiceSetCallBarringPwdInfo</i> [IN]	<ul style="list-style-type: none"> • Pointer to structure of voiceSetCallBarringPwdInfo <ul style="list-style-type: none"> – See voiceSetCallBarringPwdInfo for more information
<i>pSetCallBarringPwdResp</i> [OUT]	<ul style="list-style-type: none"> • Pointer to structure of voiceSetCallBarringPwdResp <ul style="list-style-type: none"> – See voiceSetCallBarringPwdResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS

Timeout: 30 Secs

This API changes the call barring supplementary service password. The request is a blocking request, i.e., the response is sent only after confirmation is received from the network. The maximum time it takes for the response to be sent is approximately 30 sec. This API is applicable only in 3GPP devices.

9.44.4.28 ULONG SLQSVoiceSetConfig (voiceSetConfigReq * pVoiceSetConfigReq, voiceSetConfigResp * pVoiceSetConfigResp)

This message sets various configuration parameters that control the modem behavior related to circuit-switched services.

Parameters

<i>pVoiceSetConfigReq</i> [IN]	<ul style="list-style-type: none"> Pointer to structure of voiceSetConfigReq <ul style="list-style-type: none"> See voiceSetConfigReq for more information
<i>pVoiceSetConfigResp</i> [OUT]	<ul style="list-style-type: none"> Pointer to structure of voiceSetConfigResp <ul style="list-style-type: none"> See voiceSetConfigResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

Any invalid value in a request message causes the device to reject the message without updating any configuration information. In the case of a successful update of all requested information, a QMI_ERR_NONE error is returned. In the case where a subset of information failed to be written, a QMI_ERR_INTERNAL error is returned with corresponding optional information requested in the request message.

9.44.4.29 ULONG SLQSVoiceSetPreferredPrivacy (voiceSetPrefPrivacy * pSetPrefPrivacy)

This API sets the voice privacy preference. Applicable only for 3GPP2 devices.

Parameters

<i>pSetPrefPrivacy</i> [IN]	<ul style="list-style-type: none"> See voiceSetPrefPrivacy for more information.
-----------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
 Device Supported: SL9090
 Timeout: 10 Secs

9.44.4.30 **ULONG** SLQSVoiceSetSUPSService (**voiceSetSUPSServiceReq** * *pVoiceSetSUPSServiceReq*,
voiceSetSUPSServiceResp * *pVoiceSetSUPSServiceResp*)

This API manages call-independent supplementary services, e.g., activation of call forwarding (to forward incoming calls to a third party), activation of call barring (to request the network to block some of the call attempts), and activation of call waiting (to be notified of an incoming call even when the user is engaged in an active or held call).

Parameters

<i>pVoiceSetSUPSServiceReq</i>	[IN] <ul style="list-style-type: none"> Pointer to structure of voiceSetSUPSServiceReq <ul style="list-style-type: none"> See voiceSetSUPSServiceReq for more information
<i>pVoiceSetSUPSServiceResp</i>	[OUT] <ul style="list-style-type: none"> Pointer to structure of voiceSetSUPSServiceResp <ul style="list-style-type: none"> See voiceSetSUPSServiceResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
 Timeout: 30 Secs
 Some of the call-independent services are provided by the network operator as part of the service agreement. If they are not provided by default, the user has to explicitly request them. This API provides the facility to the control point for sending the explicit request to the network for enabling/disabling

9.44.4.31 **ULONG** SLQSVoiceStartContDTMF (**voiceContDTMFInfo** * *pContDTMFInfo*)

Starts a continuous DTMF.

Parameters

<i>pContDTMFInfo</i> Info[IN/OUT]	<ul style="list-style-type: none"> Structure containing Continuous DTMF Information. <ul style="list-style-type: none"> See voiceContDTMFInfo for more Information.
--------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This API starts a continuous DTMF. If the API results indicates success, it means that the device has started the requested operation. It does not mean that the start continuous DTMF request has been sent to the network. A start continuous DTMF request is sent to the current active/alerting call when CallId is set to 0xFF.

9.44.4.32 ULONG SLQSVoiceStopContDTMF (voiceStopContDTMFInfo * pVoiceStopContDTMFInfo)

Stops a continuous DTMF.

Parameters

<p><i>pVoiceStopContDTMFInfo</i></p>	<ul style="list-style-type: none"> • Structure containing Continuous Stop DTMF Information. <ul style="list-style-type: none"> – See voiceStopContDTMFInfo for more information. • Start continuous DTMF request is sent to the current active/alerting call when CallId is set to 0xFF. • This is IN/OUT params, value passed by user will packed in request and before unpacking response this will be assigned with an invalid callID value "0". It change to a valid value if received as part of response otherwise Invalid value will be present.
--------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 30 Secs

This API starts a continuous DTMF. If the API results indicates success, it means that the device has started the requested operation. It does not mean that the start continuous DTMF request has been sent to the network. A stop continuous DTMF request is sent to the current active/alerting call when CallId is set to 0xFF.

9.45 qaGobiApiWds.h File Reference

Wireless Data Service API function prototypes.

Data Structures

- struct [dataBearerTechnology](#)
- struct [dataBearers](#)
- struct [QmiWdsIpAddressInfo](#)
- struct [WdsIpAddressInfoReq](#)
- struct [UMTSQoS](#)
- struct [ProfileIdentifier](#)
- struct [GPRSQoS](#)
- struct [PCSCFIPv4ServerAddressList](#)
- struct [PCSCFFQDNAddress](#)
- struct [PCSCFFQDNAddressList](#)
- struct [Domain](#)
- struct [DomainNameList](#)
- struct [IPv6AddressInfo](#)
- struct [IPv6GWAddressInfo](#)
- struct [qmiWdsRunTimeSettings](#)
- struct [WdsRunTimeSettings](#)
- struct [ssdatasession_params](#)
- struct [SLQSDDeleteProfileParams](#)
- struct [Profile3GPP](#)
- struct [Profile3GPP2](#)
- union [WdsProfileParam](#)
- struct [CreateProfileIn](#)
- struct [UMTSMinQoS](#)
- struct [GPRSRequestedQoS](#)
- struct [TFTIDParams](#)
- struct [UMTSReqQoSSigInd](#)
- struct [QosClassID](#)
- struct [CreateProfileOut](#)
- struct [slqsautoconnect](#)
- struct [ModifyProfileIn](#)
- struct [ModifyProfileOut](#)
- struct [_slqs3GPPConfigItem](#)
- struct [_GetProfileSettingIn](#)
- struct [_GetProfileSettingOut](#)
- struct [TrStatInd](#)
- struct [wdsSetEventReportReq](#)
- struct [swiPDPRuntimeSettingsReq](#)
- struct [swiPDPRuntimeSettingsResp](#)
- struct [WdsConnectionRateElmnts](#)
- struct [WdsConnectionRate](#)
- struct [WdsByteTotalsElmnts](#)
- struct [WdsByteTotals](#)
- struct [GetInstIDResp](#)
- struct [GetSessionIDResp](#)
- struct [TransferStatInd](#)
- struct [getDUNCallInfoReq](#)
- struct [ConnectionStatus](#)
- struct [ChannelRate](#)
- struct [getDUNCallInfoResp](#)
- struct [WdsPktStatisticsElmnts](#)
- struct [WdsPktStatisticsResp](#)
- struct [WdsPktStatisticsReq](#)
- struct [CurrNetworkInfo](#)

- struct [CurrDataSysStat](#)
- struct [swiRMTrasnferStaticsReq](#)
- struct [DataULongTlv](#)
- struct [DataULongLongTlv](#)
- struct [QmiCbkWdsStatisticsIndState](#)
- struct [DataBearerTech](#)
- struct [DataBearerTechExt](#)
- struct [WDSSWICurrentChannelRates](#)
- struct [WDSSetLoopbackData](#)
- struct [WDSGetLoopbackData](#)
- struct [WdsDHCPv4ProfileId](#)
- struct [WdsDHCPv4HWConfig](#)
- struct [WdsDHCPv4Option](#)
- struct [WdsDHCPv4OptionList](#)
- struct [WdsDHCPv4Config](#)
- struct [WdsClientLeaseChange](#)

Macros

- `#define IPV6_ADDRESS_ARRAY_SIZE 8`

Typedefs

- typedef struct [dataBearerTechnology](#) [QmiWSDDataBearerTechnology](#)
- typedef struct [dataBearers](#) [QmiWSDDataBearers](#)
- typedef union [WdsProfileParam](#) [QmiProfileInfo](#)
- typedef struct [_slqs3GPPConfigItem](#) [slqs3GPPConfigItem](#)
- typedef struct [_GetProfileSettingIn](#) [GetProfileSettingIn](#)
- typedef struct [_GetProfileSettingOut](#) [GetProfileSettingOut](#)

Enumerations

- enum [qmiDataBearerMasks](#) {
[QMI_WDS_CURRENT_CALL_DB_MASK](#) = 0x01,
[QMI_WDS_LAST_CALL_DB_MASK](#) = 0x02 }

Functions

- [ULONG SetMobileIP](#) ([ULONG](#) mode)
- [ULONG GetMobileIP](#) ([ULONG](#) *pMode)
- [ULONG SetMobileIPParameters](#) ([CHAR](#) *pSPC, [ULONG](#) *pMode, [BYTE](#) *pRetryLimit, [BYTE](#) *pRetryInterval, [BYTE](#) *pReRegPeriod, [BYTE](#) *pReRegTraffic, [BYTE](#) *pHAAAuthenticator, [BYTE](#) *pHA2002bis)
- [ULONG SetAutoconnect](#) ([ULONG](#) setting)
- [ULONG GetAutoconnect](#) ([ULONG](#) *pSetting)
- [ULONG SetDefaultProfile](#) ([ULONG](#) profileType, [ULONG](#) *pPDPTType, [ULONG](#) *pIPAddress, [ULONG](#) *pPrimaryDNS, [ULONG](#) *pSecondaryDNS, [ULONG](#) *pAuthentication, [CHAR](#) *pName, [CHAR](#) *pAPNName, [CHAR](#) *pUsername, [CHAR](#) *pPassword)
- [ULONG SetDefaultProfileLTE](#) ([ULONG](#) profileType, [ULONG](#) *pPDPTType, [ULONG](#) *pIPv4Address, [ULONG](#) *pPrimaryDNSv4, [ULONG](#) *pSecondaryDNSv4, [USHORT](#) *pIPv6Address, [USHORT](#) *pPrimaryDNSv6, [USHORT](#) *pSecondaryDNSv6, [ULONG](#) *pAuthentication, [CHAR](#) *pName, [CHAR](#) *pAPNName, [CHAR](#) *pUsername, [CHAR](#) *pPassword)

- [ULONG SetDefaultProfileLTEV2](#) (ULONG profileType, ULONG *pPDPTType, ULONG *pIPAddressv4, ULONG *pPrimaryDNSv4, ULONG *pSecondaryDNSv4, USHORT *pIPAddressv6, USHORT *pPrimaryDNSv6, USHORT *pSecondaryDNSv6, ULONG *pAuthentication, CHAR *pName, CHAR *pAPNName, CHAR *pUsername, CHAR *pPassword)
- [ULONG GetDefaultProfile](#) (ULONG profileType, ULONG *pPDPTType, ULONG *pIPAddress, ULONG *pPrimaryDNS, ULONG *pSecondaryDNS, ULONG *pAuthentication, BYTE nameSize, CHAR *pName, BYTE apnSize, CHAR *pAPNName, BYTE userSize, CHAR *pUsername)
- [ULONG GetDefaultProfileLTE](#) (ULONG profileType, ULONG *pPDPTType, ULONG *pIPAddressv4, ULONG *pPrimaryDNSv4, ULONG *pSecondaryDNSv4, USHORT *pIPAddressv6, USHORT *pPrimaryDNSv6, USHORT *pSecondaryDNSv6, ULONG *pAuthentication, BYTE nameSize, CHAR *pName, BYTE apnSize, CHAR *pAPNName, BYTE userSize, CHAR *pUsername)
- [ULONG GetSessionState](#) (ULONG *pState, BYTE instance)
- [ULONG GetPacketStatus](#) (ULONG *pTXPacketSuccesses, ULONG *pRXPacketSuccesses, ULONG *pTXPacketErrors, ULONG *pRXPacketErrors, ULONG *pTXPacketOverflows, ULONG *pRXPacketOverflows, BYTE instance)
- [ULONG GetByteTotals](#) (ULONGLONG *pTXTotalBytes, ULONGLONG *pRXTotalBytes, BYTE instance)
- [ULONG GetDormancyState](#) (ULONG *pDormancyState, BYTE instance)
- [ULONG GetDataBearerTechnology](#) (ULONG *pDataBearer, BYTE instance)
- [ULONG SLQSGetDataBearerTechnology](#) (QmiWDSDataBearers *pDataBearers, BYTE instance)
- [ULONG GetSessionDuration](#) (ULONGLONG *pDuration, BYTE instance)
- [ULONG GetIPAddressLTE](#) (WdsIpAddressInfoReq *)
- [ULONG GetConnectionRate](#) (ULONG *pCurrentChannelTXRate, ULONG *pCurrentChannelRXRate, ULONG *pMaxChannelTXRate, ULONG *pMaxChannelRXRate, BYTE instance)
- [ULONG GetMobileIPProfile](#) (BYTE index, BYTE *pEnabled, ULONG *pAddress, ULONG *pPrimaryHA, ULONG *pSecondaryHA, BYTE *pRevTunneling, BYTE naiSize, CHAR *pNAI, ULONG *pHASPI, ULONG *pAAASPI, ULONG *pHASState, ULONG *pAAASState)
- [ULONG GetLastMobileIPError](#) (ULONG *pError)
- [ULONG iSLQSMISetIPFamilyPreference](#) (BYTE IPFamilyPreference, BYTE instance)
- [BOOL WDS_IsGobiDevice](#) ()
- [ULONG SetActiveMobileIPProfile](#) (CHAR *pSPC, BYTE index)
- [ULONG SetMobileIPProfile](#) (CHAR *pSPC, BYTE index, BYTE *pEnabled, ULONG *pAddress, ULONG *pPrimaryHA, ULONG *pSecondaryHA, BYTE *pRevTunneling, CHAR *pNAI, ULONG *pHASPI, ULONG *pAAASPI, CHAR *pMNHA, CHAR *pMNAHA)
- [ULONG SLQSGetRuntimeSettings](#) (struct WdsRunTimeSettings *pRunTimeSettings)
- [ULONG SLQSSetProfile](#) (ULONG profileType, BYTE profileId, ULONG *pPDPTType, ULONG *pIPAddress, ULONG *pPrimaryDNS, ULONG *pSecondaryDNS, ULONG *pAuthentication, CHAR *pName, CHAR *pAPNName, CHAR *pUsername, CHAR *pPassword)
- [ULONG SLQSGetProfile](#) (ULONG profileType, BYTE profileId, ULONG *pPDPTType, ULONG *pIPAddress, ULONG *pPrimaryDNS, ULONG *pSecondaryDNS, ULONG *pAuthentication, BYTE nameSize, CHAR *pName, BYTE apnSize, CHAR *pAPNName, BYTE userSize, CHAR *pUsername, WORD *pExtendedErrorCode)
- [ULONG SLQSStartStopDataSession](#) (struct ssdatasession_params *pin)
- [ULONG SLQSDeleteProfile](#) (struct SLQSDeleteProfileParams *pProfileToDelete, WORD *pExtendedErrorCode)
- [ULONG SLQSCreateProfile](#) (struct CreateProfileIn *pReq, struct CreateProfileOut *pResp)
- [ULONG SLQSAutoConnect](#) (struct slqsautoconnect *pacreq)
- [ULONG SLQSModifyProfile](#) (struct ModifyProfileIn *pReq, struct ModifyProfileOut *pResp)
- [ULONG SLQSSet3GPPConfigItem](#) (slqs3GPPConfigItem *pSLQS3GPPConfigItem)
- [ULONG SLQSGet3GPPConfigItem](#) (slqs3GPPConfigItem *pSLQS3GPPConfigItem)
- [ULONG SLQSGetProfileSettings](#) (GetProfileSettingIn *pReq, GetProfileSettingOut *pResp)
- [ULONG SLQSWdsSetEventReport](#) (wdsSetEventReportReq *pSetEventReportReq)
- [ULONG SLQSWdsSwiPDPRuntimeSettings](#) (swiPDPRuntimeSettingsReq *pPDPRuntimeSettingsReq, swiPDPRuntimeSettingsResp *pPDPRuntimeSettingsResp)
- [ULONG iGetConnectionRate](#) (ULONG *pv4sessionId, ULONG *pv6sessionId, struct WdsConnectionRateElmnts *pConnectionRateElmnt)
- [ULONG SLQSGetConnectionRate](#) (struct WdsConnectionRate *pConnectionRate)

- [ULONG iGetByteTotals](#) ([ULONG](#) *pv4sessionId, [ULONG](#) *pv6sessionId, struct [WdsByteTotalsElmnts](#) *pByteTotalsElmnt)
- [ULONG SLQSGetByteTotals](#) (struct [WdsByteTotals](#) *pByteTotals)
- [ULONG SLQSWdsGoDormant](#) (void)
- [ULONG SLQSWdsGoActive](#) (void)
- [ULONG SLQSGetSessionState](#) ([ULONG](#) *pStateV4, [ULONG](#) *pStateV6, [BYTE](#) instance)
- [ULONG SLQSGetDUNCallInfo](#) ([getDUNCallInfoReq](#) *pGetDUNCallInfoReq, [getDUNCallInfoResp](#) *pGetDUNCallInfoResp)
- [ULONG GetPacketStatistics](#) (struct [WdsPktStatisticsReq](#) *pStatMask, struct [WdsPktStatisticsElmnts](#) *pPktStatisticsElmnt, [BYTE](#) instance)
- [ULONG iGetPacketStatistics](#) ([ULONG](#) *pV4sessionId, [ULONG](#) *pV6sessionId, struct [WdsPktStatisticsReq](#) *pStatMask, struct [WdsPktStatisticsElmnts](#) *pPktStatisticsElmnt)
- [ULONG SLQSGetPacketStatistics](#) (struct [WdsPktStatisticsReq](#) *pStatMask, struct [WdsPktStatisticsResp](#) *pPktStatistics)
- [ULONG SLQSGetCurrDataSystemStat](#) ([CurrDataSysStat](#) *pCurrDataSysStat)
- [ULONG RMSetTransferStatistics](#) ([swiRMTrasferStaticsReq](#) *pSwiRMTrasferStaticsReq)
- [ULONG SLQSResetPacketStatics](#) ()
- [ULONG SLQSGetDataBearerTechnologyExt](#) ([DataBearerTechExt](#) *pDataBearerTech, [BYTE](#) instance)
- [ULONG SLQSGetCurrentChannelRate](#) ([WDSSWICurrentChannelRates](#) *pRates, [BYTE](#) instance)
- [ULONG SLQSSetLoopback](#) ([WDSSetLoopbackData](#) *pReq)
- [ULONG SLQSSetLoopback](#) ([WDSGetLoopbackData](#) *data)
- [ULONG GetDefaultProfileNum](#) ([BYTE](#) profile_type, [BYTE](#) profile_family, [BYTE](#) *pProfile_no)
- [ULONG SetDefaultProfileNum](#) ([BYTE](#) profile_type, [BYTE](#) profile_family, [BYTE](#) profile_index)
- [ULONG SLQSSetDHCPv4ClientConfig](#) ([WdsDHCPv4Config](#) *pReq)
- [ULONG SLQSSetDHCPv4ClientConfig](#) ([WdsDHCPv4Config](#) *pReqResp)

9.45.1 Detailed Description

Wireless Data Service API function prototypes.

9.45.2 Macro Definition Documentation

9.45.2.1 `#define IPV6_ADDRESS_ARRAY_SIZE 8`

9.45.3 Typedef Documentation

9.45.3.1 `typedef struct _GetProfileSettingIn GetProfileSettingIn`

This structure contains the input parameters for `SLQSGetProfileSettings`

Parameters

<i>ProfileType</i>	<ul style="list-style-type: none"> Identifies the technology type of the profile <ul style="list-style-type: none"> 0x00 - 3GPP 0x01 - 3GPP2
<i>ProfileID</i>	<ul style="list-style-type: none"> index identifying the profile

9.45.3.2 typedef struct _GetProfileSettingOut GetProfileSettingOut

This structure contains the profile settings retrieved by the API SLQSGetProfileSettings

Parameters

<i>curProfile</i>	<ul style="list-style-type: none"> Structure containing details of the profile See QmiProfileInfo for more details
<i>pExtErrCode</i>	<ul style="list-style-type: none"> pointer to a 2 byte extended error code Error code will only be present if error code eQCWWAN_ERR_QMI_EXTENDED_INTER↔NAL is returned by device. See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description.

9.45.3.3 typedef union WdsProfileParam QmiProfileInfo

This union [WdsProfileParam](#) consist of [Profile3GPP](#) and [Profile3GPP2](#) out of which one will be used to create profile.

9.45.3.4 typedef struct dataBearers QmiWSDDataBearers

Structure to hold the data bearer technology values

Parameters

<i>dataBearer↔ Mask[OUT]</i>	<ul style="list-style-type: none"> This bit mask indicates if data bearer information for the current and/or last call has been received from the device. If a bit is set, then the information is available in the corresponding structure i.e. the one provided by the caller. Refer to qmiDataBearerMasks for bit-mask positions.
<i>pCurData↔ Bearer↔ Technology[O↔ UT]</i>	<ul style="list-style-type: none"> current data bearer technology value. <ul style="list-style-type: none"> – NULL if the parameter is not required
<i>pLastCallData↔ Bearer↔ Technology[O↔ UT]</i>	<ul style="list-style-type: none"> last call data bearer technology value. <ul style="list-style-type: none"> – NULL if the parameter is not required

9.45.3.5 typedef struct dataBearerTechnology QmiWSDDataBearerTechnology

Structure to hold the current data bearer technology values

Parameters

<i>pCurrent</i> ↔ <i>Network[OUT]</i>	<ul style="list-style-type: none"> current selected network <ul style="list-style-type: none"> 0 - UNKNOWN 1 - 3GPP2 2 - 3GPP
<i>pRatMask[OUT]</i>	<ul style="list-style-type: none"> Radio Access Technology (RAT) mask to indicate the type of technology (RAT mask value of zero indicates that this field is ignored) <ul style="list-style-type: none"> 0x8000 - NULL Bearer 0x0000 - DO_NOT_CARE CDMA RAT mask 0x01 - CDMA_1X 0x02 - EVDO_REV0 0x04 - EVDO_REVA UMTS RAT mask 0x01 - WCDMA 0x02 - GPRS 0x04 - HSDPA 0x08 - HSUPA 0x10 - EDGE 0x20 - LTE 0x40 - HSDPA+ 0x80 - DC_HSDPA+
<i>pSoMask[OUT]</i>	<ul style="list-style-type: none"> Service Option (SO) mask to indicate the SO or type of application (SO mask value of zero indicates that this field is ignored) <ul style="list-style-type: none"> 0x00 - DO_NOT_CARE CDMA 1X SO mask 0x01 - CDMA_1X_IS95 0x02 - CDMA_1X_IS2000 0x04 - CDMA_1X_IS2000_REL_A CDMA EV-DO Rev A SO mask 0x01 - EVDO_REVA_DPA 0x02 - EVDO_REVA_MFPA 0x04 - EVDO_REVA_EMPA 0x08 - EVDO_REVA_EMPA_EHRPD

9.45.3.6 typedef struct _slqs3GPPConfigItem slqs3GPPConfigItem

This structure contains the 3gpp Configuration Item information.

Parameters

<i>pLTEAttach</i> ↔ <i>Profile</i>	<ul style="list-style-type: none"> LTE Attach Profile <ul style="list-style-type: none"> points to a single WORD Value indicating the attached LTE Profile Optional parameter with possible values 1-16 (EM/MC73xx or earlier) function SLQSGet3GPPConfigItem() returns a default value 255 if no LTE Attach Profile is configured This setting is deprecated on MC/EM74xx
---------------------------------------	--

<i>pProfileList</i>	<ul style="list-style-type: none"> • Profile List <ul style="list-style-type: none"> – an array of 4 profile configurations – Each element points to a single WORD value indicating profile – Optional parameter with possible values <ul style="list-style-type: none"> * 1 - 16 (MC/EM73xx and before) * 1 - 24 (MC/EM74xx and onwards) – function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present
<i>pDefaultPDN↔ Enabled</i>	<ul style="list-style-type: none"> • Always Connect Default PDN <ul style="list-style-type: none"> – A single BYTE value indicating the status of Always connect default PDN <ul style="list-style-type: none"> * 0 - disabled * 1 - enabled – Optional parameter – function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present
<i>p3gppRelease</i>	<ul style="list-style-type: none"> • 3gpp release <ul style="list-style-type: none"> – A single BYTE value indicating the 3gpp release <ul style="list-style-type: none"> * 0 - Release 99 * 1 - Release 5 * 2 - Release 6 * 3 - Release 7 * 4 - Release 8 – In 9x30 and onwards <ul style="list-style-type: none"> * 5 - Release 9 * 6 - Release 10 * 7 - Release 11 – Optional parameter – function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present
<i>pLTEAttach↔ ProfileList</i>	<ul style="list-style-type: none"> • pointer to WORD array indicating LTE Attach Profile List <ul style="list-style-type: none"> – Optional parameter – possible values: 1-24 – This setting is only supported for MC/EM74xx onwards – The new equivalent option for "pLTEAttachProfile" on 74xx modems is "pLTEAttachProfile↔List". Please provide attach profiles in order of decreasing priority in this list.
<i>LTEAttach↔ ProfileListLen</i>	<ul style="list-style-type: none"> • Number of element in pLTEAttachProfileList <ul style="list-style-type: none"> – valid range: 1-24 <ul style="list-style-type: none"> * This setting is only supported for MC/EM74xx onwards

9.45.4 Enumeration Type Documentation

9.45.4.1 enum qmiDataBearerMasks

Bit mask values to indicate the presence of data bearer information for the current and last data calls

Enumerator

QMI_WDS_CURRENT_CALL_DB_MASK

QMI_WDS_LAST_CALL_DB_MASK

9.45.5 Function Documentation

9.45.5.1 ULONG GetAutoconnect (ULONG * pSetting)

Returns the current auto connect data session setting.

Parameters

<i>pSetting[OUT]</i>	<ul style="list-style-type: none"> • NDIS auto connect setting <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled
----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.45.5.2 ULONG GetByteTotals (ULONGLONG * pTXTotalBytes, ULONGLONG * pRXTotalBytes, BYTE instance)

Returns the Rx/Tx byte counts since the start of the last packet data session for IPV4 session only.

Parameters

<i>pTXTotalBytes[OUT]</i>	<ul style="list-style-type: none"> • Bytes transmitted without error
<i>pRXTotalBytes[OUT]</i>	<ul style="list-style-type: none"> • Bytes received without error
<i>instance</i>	<ul style="list-style-type: none"> • PDP instance
Generated by Doxygen	

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds, Rx/Tx byte counts for IPV4 only

9.45.5.3 `ULONG GetConnectionRate (ULONG * pCurrentChannelTXRate, ULONG * pCurrentChannelRXRate, ULONG * pMaxChannelTXRate, ULONG * pMaxChannelRXRate, BYTE instance)`

Returns connection rate information for the packet data connection. This API is not applicable when multiple data session is up. For multiple PDN, please use API [SLQSGetConnectionRate\(\)](#)

Parameters

<i>pCurrentChannelTXRate[OUT]</i>	<ul style="list-style-type: none">• Current channel Tx rate (in bps)
<i>pCurrentChannelRXRate[OUT]</i>	<ul style="list-style-type: none">• Current channel Rx rate (in bps)
<i>pMaxChannelTXRate[OUT]</i>	<ul style="list-style-type: none">• Maximum Tx rate (bps) that may be assigned to device by serving system.
<i>pMaxChannelRXRate[OUT]</i>	<ul style="list-style-type: none">• Maximum Rx rate (bps) that may be assigned to device by serving system.
<i>instance</i>	<ul style="list-style-type: none">• PDP instance

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.45.5.4 ULONG GetDataBearerTechnology (ULONG * pDataBearer, BYTE instance)

Retrieves the current data bearer technology (only valid when connected). This API is deprecated on MC73xx/↔ EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API [SLQSGetDataBearerTechnologyExt\(\)](#) for new firmware versions and new modules.

Parameters

<i>pDataBearer</i> [↔] <i>OUT</i>	<ul style="list-style-type: none"> • Data bearer technology <ul style="list-style-type: none"> – 0x01 - CDMA2000 1x – 0x02 - CDMA 1xEV-DO Rev 0 – 0x03 - GSM – 0x04 - UMTS – 0x05 - CDMA2000 HRPD (1xEV-DO Rev A) – 0x06 - EDGE – 0x07 - HSDPA AND WCDMA – 0x08 - WCDMA AND HSUPA – 0x09 - HSDPA AND HSUPA – 0x0A - LTE – 0x0B - CDMA2000 EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA – 0x10 - HSDPA+ and 64QAM – 0x11 - HSDPA+, 64QAM and HSUPA – 0x12 - TDSCDMA – 0x13 - TDSCDMA and HSDPA – 0xFF - Unknown
<i>instance</i>	<ul style="list-style-type: none"> • PDP instance

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

The QMI command of this API is deprecated. Use [SLQSGetDataBearerTechnologyExt\(\)](#) for new modules (chipset 9x15, 9x30). Version Introduced: Major-1, Minor-12 Version Deprecated: Major-1, Minor-40
Timeout: 2 seconds

9.45.5.5 **ULONG** GetDefaultProfile (**ULONG** *profileType*, **ULONG** * *pPDPTType*, **ULONG** * *pIPAddress*, **ULONG** * *pPrimaryDNS*, **ULONG** * *pSecondaryDNS*, **ULONG** * *pAuthentication*, **BYTE** *nameSize*, **CHAR** * *pName*, **BYTE** *apnSize*, **CHAR** * *pAPNName*, **BYTE** *userSize*, **CHAR** * *pUsername*)

Reads the default profile settings from the device. The default profile is used to establish an auto connect data session.

Parameters

<i>profileType</i>	<ul style="list-style-type: none"> Type of profile <ul style="list-style-type: none"> 0 - UMTS
<i>pPDPTType</i> [OUT]	<ul style="list-style-type: none"> Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile <ul style="list-style-type: none"> 0 - PDP-IP (IPv4)
<i>pIPAddress</i> [OUT]	<ul style="list-style-type: none"> Preferred IPv4 address to be assigned to device
<i>pPrimaryDNS</i> [OUT]	<ul style="list-style-type: none"> Primary DNS ipv4 address preference
<i>pSecondaryDNS</i> [OUT]	<ul style="list-style-type: none"> Secondary DNS ipv4 address preference
<i>pAuthentication</i> [OUT]	<ul style="list-style-type: none"> Bitmap that indicates authentication algorithm preference <ul style="list-style-type: none"> 0x00000001 - PAP preference <ul style="list-style-type: none"> 0 - Never performed 1 - May be performed 0x00000002 - CHAP preference <ul style="list-style-type: none"> 0 - Never performed 1 - May be performed All other bits are reserved and must be set to 0 If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g. the device may have a policy to select the most secure authentication mechanism.
<i>nameSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that profile name array can contain.
<i>pName</i> [OUT]	<ul style="list-style-type: none"> Profile name
<i>apnSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that APN name array can contain
<i>pAPNName</i> [OUT]	<ul style="list-style-type: none"> Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network. If value is NULL or omitted, then subscription default value will be requested.

<i>userSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that username array can contain.
<i>pUsername</i> [O↔ UT]	<ul style="list-style-type: none"> Username used during network authentication

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.45.5.6 **ULONG** GetDefaultProfileLTE (**ULONG** *profileType*, **ULONG** * *pPDPTType*, **ULONG** * *pIPAddressv4*, **ULONG** * *pPrimaryDNSv4*, **ULONG** * *pSecondaryDNSv4*, **USHORT** * *pIPAddressv6*, **USHORT** * *pPrimaryDNSv6*, **USHORT** * *pSecondaryDNSv6*, **ULONG** * *pAuthentication*, **BYTE** *nameSize*, **CHAR** * *pName*, **BYTE** *apnSize*, **CHAR** * *pAPNName*, **BYTE** *userSize*, **CHAR** * *pUsername*)

Reads the default profile settings from the device. The default profile is used to establish an auto connect data session.

Parameters

<i>profileType</i>	<ul style="list-style-type: none"> Type of profile <ul style="list-style-type: none"> 0 - UMTS
<i>pPDPTType</i> [OUT]	<ul style="list-style-type: none"> Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile <ul style="list-style-type: none"> 0 - PDP-IP (IPv4)
<i>pIPAddressv4</i> [↔ OUT]	<ul style="list-style-type: none"> Preferred IPv4 addr to be assigned to device
<i>pPrimaryDN</i> ↔ <i>Sv4</i> [OUT]	<ul style="list-style-type: none"> Primary DNS ipv4 address preference
<i>pSecondaryD</i> ↔ <i>NSv4</i> [OUT]	<ul style="list-style-type: none"> Secondary DNS ipv4 address preference

<i>pIPAddressv6</i> [↔ OUT]	<ul style="list-style-type: none"> Preferred IPv6 addr to be assigned to device Space for storing 8 element array for the IPv6 addresses is allocated by the application. The IP Address will be retrieved in the big endian format. For example User buffer contents: [<U0>..<<U7>] IPv6 address: 1234:2A01:.....:5678 U0 corresponds to 1234 U1 corresponds to 2A01 ----- ----- U7 corresponds to 5678
<i>pPrimaryDN</i> [↔ Sv6[OUT]	<ul style="list-style-type: none"> Primary DNS ipv6 address preference
<i>pSecondaryDN</i> [↔ NSv6[OUT]	<ul style="list-style-type: none"> Secondary DNS ipv6 address preference
<i>p</i> ↔ <i>Authentication</i> [↔ OUT]	<ul style="list-style-type: none"> Bitmap that indicates authentication algorithm preference <ul style="list-style-type: none"> 0x00000001 - PAP preference <ul style="list-style-type: none"> * 0 - Never performed * 1 - May be performed 0x00000002 - CHAP preference <ul style="list-style-type: none"> * 0 - Never performed * 1 - May be performed All other bits are reserved and must be set to 0 If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g. the device may have a policy to select the most secure authentication mechanism.
<i>nameSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that Profile name array can contain
<i>pName</i> [OUT]	<ul style="list-style-type: none"> Profile name
<i>apnSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that APN name array can contain
<i>pAPNName</i> [IN]	<ul style="list-style-type: none"> Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network. If value is NULL or omitted, then subscription default value will be requested.
<i>userSize</i>	<ul style="list-style-type: none"> Maximum number of characters including NULL terminator) that username array can contain.

<i>pUsername</i> [O↔ UT]	<ul style="list-style-type: none"> Username used during network authentication
-----------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: LTE
Timeout: 2 seconds

9.45.5.7 **ULONG** GetDefaultProfileNum (**BYTE** *profile_type*, **BYTE** *profile_family*, **BYTE** * *pProfile_no*)

This API to Get default profile number

Parameters

<i>profile_type</i>	[IN] <ul style="list-style-type: none"> 0 - 3GPP 1 - 3GPP2
<i>profile_family</i>	[IN] <ul style="list-style-type: none"> 0 - Embedded 1 - Tethered
<i>pProfile_no</i>	[OUT]

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 2 seconds\n

9.45.5.8 **ULONG** GetDormancyState (**ULONG** * *pDormancyState*, **BYTE** *instance*)

Returns the dormancy state of the current packet data session when connected.

Parameters

<i>pDormancy↔ State[OUT]</i>	<ul style="list-style-type: none">• Dormancy state of current packet data session<ul style="list-style-type: none">– 1 - Traffic channel dormant– 2 - Traffic channel active
<i>instance</i>	<ul style="list-style-type: none">• PDP instance

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.45.5.9 ULONG GetIPAddressLTE (WdsIpAddressInfoReq *)

Returns the current packet data session IP address(es)

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: LTE
Timeout: 2 seconds.

9.45.5.10 ULONG GetLastMobileIPError (ULONG * pError)

Returns the last mobile IP error.

Parameters

<i>pError[OUT]</i>	<ul style="list-style-type: none">• Status of last MIP call (or attempt)<ul style="list-style-type: none">– Zero - Success– NonZero - Error codeSee qaGobiApiTableCallEndReasons.h for Mobile IP Error codes
--------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA

Device Supported: MC83x5

Timeout: 2 seconds

9.45.5.11 ULONG GetMobileIP (ULONG * pMode)

Returns the current mobile IP setting.

Parameters

<i>mode[OUT]</i>	<ul style="list-style-type: none">• Mobile IP setting<ul style="list-style-type: none">– 0 - Mobile IP off (simple IP only)– 1 - Mobile IP preferred– 2 - Mobile IP only
------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA

Timeout: 2 seconds

9.45.5.12 **ULONG** GetMobileIPProfile (**BYTE** *index*, **BYTE** * *pEnabled*, **ULONG** * *pAddress*, **ULONG** * *pPrimaryHA*, **ULONG** * *pSecondaryHA*, **BYTE** * *pRevTunneling*, **BYTE** *naiSize*, **CHAR** * *pNAI*, **ULONG** * *pHASPI*, **ULONG** * *pAAASPI*, **ULONG** * *pHASState*, **ULONG** * *pAAASState*)

Returns the specified mobile IP profile settings.

Parameters

<i>index</i>	<ul style="list-style-type: none"> Mobile IP profile ID
<i>pEnabled</i> [OUT]	<ul style="list-style-type: none"> Profile enabled: <ul style="list-style-type: none"> 0 - Disabled 1 - Enabled 0xFF - Unknown
<i>pAddress</i> [OUT]	<ul style="list-style-type: none"> Home IPv4 address: <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
<i>pPrimaryHA</i> [OUT]	<ul style="list-style-type: none"> Primary home agent IPv4 address <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
<i>pSecondaryHA</i> [OUT]	<ul style="list-style-type: none"> Secondary home agent IPv4 address <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
<i>pRevTunneling</i> [OUT]	<ul style="list-style-type: none"> Reverse tunneling enabled <ul style="list-style-type: none"> 0 - Disabled 1 - Enabled 0xFF - Unknown
<i>naiSize</i>	<ul style="list-style-type: none"> The maximum number of characters (including NULL terminator) that the NAI array can contain.
<i>pNAI</i> [OUT]	<ul style="list-style-type: none"> Network access identifier string
<i>pHASPI</i> [OUT]	<ul style="list-style-type: none"> Home agent security parameter index
<i>pAAASPI</i> [OUT]	<ul style="list-style-type: none"> AAA server security parameter index <ul style="list-style-type: none"> 0xFFFFFFFF - Unknown
<i>pHASState</i> [OUT]	<ul style="list-style-type: none"> Home agent key state <ul style="list-style-type: none"> 0 - Unset 1 - Set, default value 2 - Set, modified from default 3 - 0xFFFFFFFF - Unknown

<i>pAAASState</i> [O↔ UT]	<ul style="list-style-type: none"> • AAA key state <ul style="list-style-type: none"> – 0 - Unset – 1 - Set, default value – 2 - Set, modified from default – 3 - 0xFFFFFFFF - Unknown
------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 2 seconds

9.45.5.13 ULONG GetPacketStatistics (struct WdsPktStatisticsReq * *pStatMask*, struct WdsPktStatisticsElmnts * *pPktStatisticsElmnt*, BYTE *instance*)

Returns the current packet transfer counter values from the device. Since the start of the last packet data session.

Parameters

<i>pStatMask</i> [IN]	<ul style="list-style-type: none"> • See WdsPktStatisticsReq for more information
<i>pPktStatistics</i> ↔ <i>Elmnt</i> [OUT]	<ul style="list-style-type: none"> • See WdsPktStatisticsElmnts for more information
<i>instance</i>	<ul style="list-style-type: none"> • PDP instance

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.45.5.14 `ULONG GetPacketStatus (ULONG * pTXPacketSuccesses, ULONG * pRXPacketSuccesses, ULONG * pTXPacketErrors, ULONG * pRXPacketErrors, ULONG * pTXPacketOverflows, ULONG * pRXPacketOverflows, BYTE instance)`

Returns the packet data transfer statistics since the start of the current packet data.

Parameters

<i>pTXPacketSuccesses[OUT]</i>	<ul style="list-style-type: none"> No. of packets transmitted without error
<i>pRXPacketSuccesses[OUT]</i>	<ul style="list-style-type: none"> No. of packets received without error
<i>pTXPacketErrors[OUT]</i>	<ul style="list-style-type: none"> No. of outgoing packets with framing errors
<i>pRXPacketErrors[OUT]</i>	<ul style="list-style-type: none"> No. of incoming packets with framing errors
<i>pTXPacketOverflows[OUT]</i>	<ul style="list-style-type: none"> Number of packets dropped because Tx buffer overflowed
<i>pRXPacketOverflows[OUT]</i>	<ul style="list-style-type: none"> Number of packets dropped because Rx buffer overflowed
<i>instance</i>	<ul style="list-style-type: none"> PDP instance

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.45.5.15 `ULONG GetSessionDuration (ULONGLONG * pDuration, BYTE instance)`

Returns the duration of the current packet data session.

Parameters

<i>pDuration[OUT]</i>	<ul style="list-style-type: none"> Duration of the current packet session in milliseconds
<i>instance</i>	<ul style="list-style-type: none"> PDP instance

Generated by Doxygen

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS/CDMA
 Device Supported: MC83x5, MC7700/50
 Timeout: 2 seconds

9.45.5.16 **ULONG** GetSessionState (**ULONG** * *pState*, **BYTE** *instance*)

Returns the state of the current packet data session.

Parameters

<i>pState[OUT]</i>	<ul style="list-style-type: none"> • Current link status <ul style="list-style-type: none"> – 1 - DISCONNECTED – 2 - CONNECTED – 3 - SUSPENDED (not supported) – 4 - AUTHENTICATING
<i>instance</i>	<ul style="list-style-type: none"> • PDP instance

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.45.5.17 **ULONG** iGetByteTotals (**ULONG** * *pv4sessionId*, **ULONG** * *pv6sessionId*, struct WdsByteTotalsElmnts * *pByteTotalsElmnt*)

9.45.5.18 **ULONG** iGetConnectionRate (**ULONG** * *pv4sessionId*, **ULONG** * *pv6sessionId*, struct WdsConnectionRateElmnts * *pConnectionRateElmnt*)

9.45.5.19 **ULONG** iGetPacketStatistics (**ULONG** * *pV4sessionId*, **ULONG** * *pV6sessionId*, struct WdsPktStatisticsReq * *pStatMask*, struct WdsPktStatisticsElmnts * *pPktStatisticsElmnt*)

9.45.5.20 **ULONG** iSLQSMISetIPFamilyPreference (**BYTE** *IPFamilyPreference*, **BYTE** *instance*)

9.45.5.21 **ULONG** RMSetTransferStatistics (**swiRMTrasferStaticsReq** * *pSwiRMTrasferStaticsReq*)

This API request the device to fetch current data system transfer Statistics.

Parameters

<i>pSwiRMTrasferStaticsReq</i> [IN]	<ul style="list-style-type: none"> See swiRMTrasferStaticsReq for more information
-------------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 2 seconds\n

9.45.5.22 **ULONG** SetActiveMobileIPProfile (**CHAR** * *pSPC*, **BYTE** *index*)

Sets active mobile IP profile.

Parameters

<i>pSPC</i> [IN]	<ul style="list-style-type: none"> NULL-terminated string representing six digit service programming code
<i>index</i> [IN]	<ul style="list-style-type: none"> Index of the profile to be set as the active profile

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Timeout: 2 seconds

9.45.5.23 ULONG SetAutoconnect (ULONG *setting*)

Sets the auto connect data session setting.

Parameters

<i>setting</i> [IN]	<ul style="list-style-type: none"> • NDIS autoconnect setting <ul style="list-style-type: none"> – 0 - Disabled – 1 - Enabled
---------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

When enabling, timeout is 5 minutes,
When disabling, timeout is 5 seconds

9.45.5.24 ULONG SetDefaultProfile (ULONG *profileType*, ULONG * *pPDPTType*, ULONG * *pIPAddress*, ULONG * *pPrimaryDNS*, ULONG * *pSecondaryDNS*, ULONG * *pAuthentication*, CHAR * *pName*, CHAR * *pAPNName*, CHAR * *pUsername*, CHAR * *pPassword*)

Writes the default profile settings to the device. The default profile is used to establish an autoconnect data session.

Parameters

<i>profileType</i>	<ul style="list-style-type: none"> • Type of profile <ul style="list-style-type: none"> – 0 - UMTS
<i>pPDPTType</i> [IN]	<ul style="list-style-type: none"> • Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile (optional) <ul style="list-style-type: none"> – 0 - PDP-IP (IPv4)
<i>pIPAddress</i> [IN]	<ul style="list-style-type: none"> • Preferred IPv4 addr to be assigned to device (optional)
<i>pPrimaryDNS</i> [↔ IN]	<ul style="list-style-type: none"> • Primary DNS ipv4 address preference (optional)
<i>pSecondaryDNS</i> [↔ IN]	<ul style="list-style-type: none"> • Secondary DNS ipv4 address preference (optional)

<i>pAuthentication</i> [IN]	<ul style="list-style-type: none"> • Bitmap that indicates authentication algorithm preference (optional) <ul style="list-style-type: none"> – 0x00000001 - PAP preference <ul style="list-style-type: none"> * 0 - Never performed * 1 - May be performed – 0x00000002 - CHAP preference <ul style="list-style-type: none"> * 0 - Never performed * 1 - May be performed – All other bits are reserved and must be set to 0 – If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g. the device may have a policy to select the most secure authentication mechanism.
<i>pName</i> [IN]	<ul style="list-style-type: none"> • profile Name (optional)
<i>pAPNName</i> [IN]	<ul style="list-style-type: none"> • Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network (optional) • If value is NULL or omitted, then subscription default value will be requested.
<i>pUsername</i> [IN]	<ul style="list-style-type: none"> • Username used during network authentication (optional)
<i>pPassword</i> [IN]	<ul style="list-style-type: none"> • Password used during network authentication (optional)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout is 2 seconds.

9.45.5.25 `ULONG SetDefaultProfileLTE (ULONG profileType, ULONG * pPDPTType, ULONG * pIPAddressv4, ULONG * pPrimaryDNSv4, ULONG * pSecondaryDNSv4, USHORT * pIPAddressv6, USHORT * PrimaryDNSv6, USHORT * pSecondaryDNSv6, ULONG * pAuthentication, CHAR * pName, CHAR * pAPNName, CHAR * pUsername, CHAR * pPassword)`

Writes the default profile settings to the device. The default profile is used to establish an auto connect data session.

Parameters

<i>profileType</i>	<ul style="list-style-type: none"> Type of profile <ul style="list-style-type: none"> 0 - UMTS
<i>pPDPTType</i> [IN]	<ul style="list-style-type: none"> Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile (optional) <ul style="list-style-type: none"> 0 - PDP-IP (IPv4)
<i>pIPvAddressv4</i> [↔ IN]	<ul style="list-style-type: none"> Preferred IPv4 address to be assigned to device (optional)
<i>pPrimaryDN</i> [↔ Sv4][IN]	<ul style="list-style-type: none"> Primary DNS ipv4 address preference (optional)
<i>pSecondaryDN</i> [↔ NSv4][IN]	<ul style="list-style-type: none"> Secondary DNS ipv4 address preference (optional)
<i>pIPvAddressv6</i> [↔ IN]	<ul style="list-style-type: none"> Preferred IPv6 address to be assigned to device (optional)
<i>pPrimaryDN</i> [↔ Sv6][IN]	<ul style="list-style-type: none"> Primary DNS ipv6 address preference (optional)
<i>pSecondaryDN</i> [↔ NSv6][IN]	<ul style="list-style-type: none"> Secondary DNS ipv6 address preference (optional)
<i>p</i> [↔ <i>Authentication</i>][↔ IN]	<ul style="list-style-type: none"> Bitmap that indicates authentication algorithm preference (optional) <ul style="list-style-type: none"> 0x00000001 - PAP preference <ul style="list-style-type: none"> 0 - Never performed 1 - May be performed 0x00000002 - CHAP preference <ul style="list-style-type: none"> 0 - Never performed 1 - May be performed All other bits are reserved and must be set to 0 If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g.the device may have a policy to select the most secure authentication mechanism.
<i>pName</i> [IN]	<ul style="list-style-type: none"> profile Name (optional)
<i>pAPNName</i> [IN]	<ul style="list-style-type: none"> Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network (optional) If value is NULL or omitted, then subscription default value will be requested
<i>pUsername</i> [IN]	<ul style="list-style-type: none"> Username used during network authentication (optional)

<i>pPassword</i> [IN]	<ul style="list-style-type: none"> • Password used during network authentication (optional)
-----------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Deprecated, please use SetDefaultProfileLTEV2 instead
 Technology Supported: LTE
 Timeout: 2 seconds

9.45.5.26 **ULONG** SetDefaultProfileLTEV2 (**ULONG** *profileType*, **ULONG** * *pPDPTType*, **ULONG** * *pIPAddressv4*, **ULONG** * *pPrimaryDNSv4*, **ULONG** * *pSecondaryDNSv4*, **USHORT** * *pIPAddressv6*, **USHORT** * *pPrimaryDNSv6*, **USHORT** * *pSecondaryDNSv6*, **ULONG** * *pAuthentication*, **CHAR** * *pName*, **CHAR** * *pAPNName*, **CHAR** * *pUsername*, **CHAR** * *pPassword*)

Writes the default profile settings to the device. The default profile is used to establish an auto connect data session.

Parameters

<i>profileType</i>	<ul style="list-style-type: none"> • Type of profile <ul style="list-style-type: none"> – 0 - UMTS
<i>pPDPTType</i> [IN]	<ul style="list-style-type: none"> • Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile (optional) <ul style="list-style-type: none"> – 0 - PDP-IP (IPv4)
<i>pIPAddressv4</i> [↔ IN]	<ul style="list-style-type: none"> • Preferred IPv4 address to be assigned to device (optional)
<i>pPrimaryDN</i> ↔ <i>Sv4</i> [IN]	<ul style="list-style-type: none"> • Primary DNS ipv4 address preference (optional)
<i>pSecondaryD</i> ↔ <i>NSv4</i> [IN]	<ul style="list-style-type: none"> • Secondary DNS ipv4 address preference (optional)
<i>pIPAddressv6</i> [↔ IN]	<ul style="list-style-type: none"> • Preferred IPv6 addr to be assigned to device (optional)
<i>pPrimaryDN</i> ↔ <i>Sv6</i> [IN]	<ul style="list-style-type: none"> • Primary DNS ipv6 address preference (optional)

<i>pSecondaryDNSv6</i> [IN]	<ul style="list-style-type: none"> • Secondary DNS ipv6 address preference (optional)
<i>pAuthentication</i> [IN]	<ul style="list-style-type: none"> • Bitmap that indicates authentication algorithm preference (optional) <ul style="list-style-type: none"> – 0x00000001 - PAP preference <ul style="list-style-type: none"> * 0 - Never performed * 1 - May be performed – 0x00000002 - CHAP preference <ul style="list-style-type: none"> * 0 - Never performed * 1 - May be performed – All other bits are reserved and must be set to 0 – If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g.the device may have a policy to select the most secure authentication mechanism.
<i>pName</i> [IN]	<ul style="list-style-type: none"> • profile Name (optional)
<i>pAPNName</i> [IN]	<ul style="list-style-type: none"> • Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network (optional) • If value is NULL or omitted, then subscription default value will be requested
<i>pUsername</i> [IN]	<ul style="list-style-type: none"> • Username used during network authentication (optional)
<i>pPassword</i> [IN]	<ul style="list-style-type: none"> • Password used during network authentication (optional)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: LTE

Timeout: 2 seconds

Replaces deprecated Function SetDefaultProfileLTE

9.45.5.27 ULONG SetDefaultProfileNum (BYTE profile_type, BYTE profile_family, BYTE profile_index)

This API to Set default profile number

Parameters

<i>profile_type</i>	[IN] <ul style="list-style-type: none">• 0 - 3GPP• 1 - 3GPP2
<i>profile_family</i>	[IN] <ul style="list-style-type: none">• 0 - Embedded• 1 - Tethered
<i>profile_index</i>	[IN]

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 2 seconds\n

9.45.5.28 ULONG SetMobileIP (ULONG mode)

Sets the current mobile IP setting.

Parameters

<i>mode</i> [IN]	<ul style="list-style-type: none">• Mobile IP setting<ul style="list-style-type: none">– 0 - Mobile IP off (simple IP only)– 1 - Mobile IP preferred– 2 - Mobile IP only
------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA

Timeout: 2 seconds

9.45.5.29 **ULONG** SetMobileIPParameters (**CHAR** * *pSPC*, **ULONG** * *pMode*, **BYTE** * *pRetryLimit*, **BYTE** * *pRetryInterval*, **BYTE** * *pReRegPeriod*, **BYTE** * *pReRegTraffic*, **BYTE** * *pHAAuthenticator*, **BYTE** * *pHA2002bis*)

Sets the specified mobile IP parameters.

Parameters

<i>pSPC</i> [IN]	<ul style="list-style-type: none"> • NULL-terminated string representing six digit service programming code.
<i>pMode</i> [IN]	<ul style="list-style-type: none"> • Mode to be set (optional) <ul style="list-style-type: none"> – 0 - Mobile IP off (simple IP only) – 1 - Mobile IP preferred – 2 - Mobile IP only
<i>pRetryLimit</i> [IN]	<ul style="list-style-type: none"> • Registration retry attempt limit (optional)
<i>pRetryInterval</i> [↔ IN]	<ul style="list-style-type: none"> • Registration retry attempt interval used to determine the time between registration attempts (optional)
<i>pReReg</i> ↔ <i>Period</i> [IN]	<ul style="list-style-type: none"> • Period (in minutes) to attempt re-registration before current registration expires (optional)
<i>pReReg</i> ↔ <i>Traffic</i> [IN]	<ul style="list-style-type: none"> • Re-registration only if traffic since last attempt (optional) <ul style="list-style-type: none"> – Zero - Disabled – NonZero - Enabled
<i>pHA</i> ↔ <i>Authenticator</i> [IN]	<ul style="list-style-type: none"> • MH-HA authenticator calculator (optional) <ul style="list-style-type: none"> – Zero - Disabled – NonZero - Enabled
<i>pHA2002bis</i> [IN]	<ul style="list-style-type: none"> • MH-HA RFC 2002bis authentication instead of RFC2002 (optional) <ul style="list-style-type: none"> – Zero - Disabled – NonZero - Enabled

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
Device Supported: None
Timeout: 2 seconds

9.45.5.30 **ULONG** SetMobileIPProfile (**CHAR** * *pSPC*, **BYTE** *index*, **BYTE** * *pEnabled*, **ULONG** * *pAddress*, **ULONG** * *pPrimaryHA*, **ULONG** * *pSecondaryHA*, **BYTE** * *pRevTunneling*, **CHAR** * *pNAI*, **ULONG** * *pHASPI*, **ULONG** * *pAAASPI*, **CHAR** * *pMNHA*, **CHAR** * *pMNAAA*)

Sets the mobile IP parameters.

Parameters

<i>pSPC</i> [IN]	<ul style="list-style-type: none"> Six digit service programming code string
<i>index</i> [IN]	<ul style="list-style-type: none"> Index of the profile to modify
<i>pEnabled</i> [IN]	<ul style="list-style-type: none"> (Optional) Enable profile? 0 - Disabled Nonzero - Enabled
<i>pAddress</i> [IN]	<ul style="list-style-type: none"> (Optional) Home IPv4 address
<i>pPrimaryHA</i> [IN]	<ul style="list-style-type: none"> (Optional) Primary home agent IPv4 address
<i>pSecondaryHA</i> [IN]	<ul style="list-style-type: none"> (Optional) Secondary home agent IPv4 address
<i>pRevTunneling</i> [IN]	<ul style="list-style-type: none"> (Optional) Enable reverse tunneling? 0 - Disabled Nonzero - Enabled
<i>pNAI</i> [IN]	<ul style="list-style-type: none"> (Optional) Network access identifier string
<i>pHASPI</i> [IN]	<ul style="list-style-type: none"> (Optional) Home agent security parameter index
<i>pAAASPI</i> [IN]	<ul style="list-style-type: none"> (Optional) AAA server security parameter index
<i>pMNHA</i> [IN]	<ul style="list-style-type: none"> (Optional) MN-HA key string
<i>pMNAAA</i> [IN]	<ul style="list-style-type: none"> (Optional) MN-AAA key string

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA
 Timeout: 2 seconds

9.45.5.31 ULONG SLQSAutoConnect (struct slqsaconnect * *pacreq*)

Returns auto connect settings

Parameters

<i>slqsaconnect</i> [<i>IN</i>]	<ul style="list-style-type: none"> SLQS auto connect settings
-----------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: CDMA/UMTS
 Device Supported: MC83x5, MC7700
 Timeout: 2 seconds

9.45.5.32 ULONG SLQSCreateProfile (struct CreateProfileIn * *pReq*, struct CreateProfileOut * *pResp*)

Create a new profile with the specified parameters. Note that some firmware versions do not support the optional Profile ID parameter. In this case an error will be returned and the caller can subsequently create a profile by specifying a NULL pointer for the Profile ID parameter. The Profile ID pertaining to the newly created profile is returned in the response structure (pResp).

Parameters

<i>pReq</i> [<i>IN</i>]	<ul style="list-style-type: none"> SLQS Create profile Information
<i>pResp</i> [<i>OUT</i>]	<ul style="list-style-type: none"> SLQS profile identifier information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS
 Device Supported: MC83x5, MC7700
 Timeout: 2 seconds

9.45.5.33 ULONG SLQSDeleteProfile (struct SLQSDeleteProfileParams * pProfileToDelete, WORD * pExtendedErrorCode)

Deletes a configured profile stored on the device. The deletion of a profile does not affect profile index assignments.

Parameters

<i>pProfileToDelete</i> [IN]	<ul style="list-style-type: none"> Information about the profile to be deleted. See SLQSDeleteProfileParams for more details.
<i>pExtendedErrorCode</i> [OUT]	<ul style="list-style-type: none"> The extended error code received from DS Profile subsystem of type eWDS_ERR_PROFILE_ REG_ xxx. Error code will only be present if error code eQCWWAN_ERR_QMI_EXTENDED_INTERNAL is returned by device. See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values.

Note

Timeout: 2 seconds

9.45.5.34 ULONG SLQSGet3GPPConfigItem (slqs3GPPConfigItem * pSLQS3GPPConfigItem)

Reads the 3gpp configuration item.

Parameters

<i>pSLQS3GPPConfigItem</i> [OUT]	<ul style="list-style-type: none"> See slqs3GPPConfigItem for more information
Generated by Doxygen 1.7.7	

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS/LTE
Timeout: 2 seconds

9.45.5.35 ULONG SLQSGetByteTotals (struct WdsByteTotals * pByteTotals)

This API request the device to fetch ByteTotals for IPV4 and IPV6.

Parameters

<i>pByteTotals</i> [↔ IN/OUT]	<ul style="list-style-type: none">See WdsByteTotals for more information
----------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 2 seconds\n

9.45.5.36 ULONG SLQSGetConnectionRate (struct WdsConnectionRate * pConnectionRate)

This API request the device to fetch ConnectionRate. It can be used for both mono and multiple PDN use case.

Parameters

<i>pConnection</i> ↔ Rate[IN/OUT]	<ul style="list-style-type: none">See WdsConnectionRate for more information
--------------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS/CDMA
Device Supported: MC77XX
Timeout: 2 seconds

9.45.5.37 ULONG SLQSGetCurrDataSystemStat (CurrDataSysStat * pCurrDataSysStat)

This API request the device to fetch current data system status.

Parameters

<i>pCurrDataSysStat</i> [IN/OUT]	<ul style="list-style-type: none">See CurrDataSysStat for more information
----------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 5 seconds\n

9.45.5.38 ULONG SLQSGetCurrentChannelRate (WDSSWICurrentChannelRates * pRates, BYTE instance)

This API Queries current bitrate of a packet data connection.

Parameters

<i>pRates</i>	[IN] <ul style="list-style-type: none">See WDSSWICurrentChannelRates for more information
---------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

This feature depends on custom feature setting IPCHANNELRATEEN which can be set via SLQSSetCustomFeatures

Timeout: 2 seconds

9.45.5.39 ULONG SLQSGetDataBearerTechnology (QmiWDSDataBearers * pDataBearers, BYTE instance)

Retrieves the data bearer technology values for current and/or last data calls. The device must be in a data call for this function to execute successfully.

Parameters

<i>qmiWDSDataBearers[OUT]</i>	<ul style="list-style-type: none"> Indicates the current and the last call data bearer technology. Should not be NULL, on input.
<i>instance</i>	<ul style="list-style-type: none"> PDP instance

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS/CDMA
 Device Supported: MC7750,GOBI,MC7700
 Timeout: 2 seconds

9.45.5.40 ULONG SLQSGetDataBearerTechnologyExt (DataBearerTechExt * pDataBearerTech, BYTE instance)

This API Get Data Bearer Technology. This is a new API to replace API [GetDataBearerTechnology\(\)](#). see the description of [GetDataBearerTechnology\(\)](#) to get more information

Parameters

<i>pDataBearerTech</i>	[IN] <ul style="list-style-type: none"> See DataBearerTechExt for more information
<i>instance</i>	[IN] <ul style="list-style-type: none"> PDP instance

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 2 seconds\n

9.45.5.41 **ULONG** SLQSGetDUNCallInfo (**getDUNCallInfoReq** * *pGetDUNCallInfoReq*, **getDUNCallInfoResp** * *pGetDUNCallInfoResp*)

This API queries the current modem connection status.

Parameters

<i>pGetDUNCallInfoReq</i> [IN]	<ul style="list-style-type: none"> See getDUNCallInfoReq for more information
<i>pGetDUNCallInfoResp</i> [OUT]	<ul style="list-style-type: none"> See getDUNCallInfoResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 seconds

9.45.5.42 **ULONG** SLQSGetPacketStatistics (**struct WdsPktStatisticsReq** * *pStatMask*, **struct WdsPktStatisticsResp** * *pPktStatistics*)

This API request the device to fetch current packet transfer counter values from the device

Parameters

<i>pStatMask</i> [IN]	<ul style="list-style-type: none"> See WdsPktStatisticsReq for more information
<i>pPktStatistics</i> [OUT]	<ul style="list-style-type: none"> See WdsPktStatisticsResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 5 seconds\n

9.45.5.43 **ULONG** SLQSGetProfile (**ULONG** *profileType*, **BYTE** *profileId*, **ULONG** * *pPDPTType*, **ULONG** * *pIPAddress*, **ULONG** * *pPrimaryDNS*, **ULONG** * *pSecondaryDNS*, **ULONG** * *pAuthentication*, **BYTE** *nameSize*, **CHAR** * *pName*, **BYTE** *apnSize*, **CHAR** * *pAPNName*, **BYTE** *userSize*, **CHAR** * *pUsername*, **WORD** * *pExtendedErrorCode*)

Reads the profile settings from the device for the specified profile id.

Parameters

<i>profileType</i>	<ul style="list-style-type: none"> Type of profile <ul style="list-style-type: none"> 0 - UMTS
<i>profileId</i>	<ul style="list-style-type: none"> Index of the configured profile for which settings are read <ul style="list-style-type: none"> Value between 1 - 16 (EM/MC73xx or earlier) Value between 1 - 24 (EM/MC74xx onwards)
<i>pPDPTType</i> [OUT]	<ul style="list-style-type: none"> Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile <ul style="list-style-type: none"> 0 - PDP-IP (IPv4)
<i>pIPAddress</i> [OUT]	<ul style="list-style-type: none"> Preferred IPv4 address to be assigned to device
<i>pPrimaryDNS</i> [OUT]	<ul style="list-style-type: none"> Primary DNS ipv4 address preference
<i>pSecondaryDNS</i> [OUT]	<ul style="list-style-type: none"> Secondary DNS ipv4 address preference
<i>pAuthentication</i> [OUT]	<ul style="list-style-type: none"> Bitmap that indicates authentication algorithm preference <ul style="list-style-type: none"> 0x00000001 - PAP preference <ul style="list-style-type: none"> 0 - Never performed 1 - May be performed 0x00000002 - CHAP preference <ul style="list-style-type: none"> 0 - Never performed 1 - May be performed All other bits are reserved and must be set to 0 If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g. the device may have a policy to select the most secure authentication mechanism.

<i>nameSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that profile name array can contain.
<i>pName[OUT]</i>	<ul style="list-style-type: none"> Profile name
<i>apnSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that APN name array can contain
<i>pAPNName[OUT]</i>	<ul style="list-style-type: none"> Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network. If value is NULL or omitted, then subscription default value will be requested.
<i>userSize</i>	<ul style="list-style-type: none"> Maximum number of characters (including NULL terminator) that username array can contain.
<i>pUsername[OUT]</i>	<ul style="list-style-type: none"> Username used during network authentication
<i>pExtendedError</i>	<ul style="list-style-type: none"> The extended error code received from DS Profile subsystem of type <code>eWDS_ERR_PROFILE_REG_XXX</code>. Error code will only be present if error code <code>eQCWWAN_ERR_QMI_EXTENDED_INTERNAL</code> is returned by device. See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description.

Returns

`eQCWWAN_ERR_NONE` on success, `eQCWWAN_XXX` error value otherwise

See also

See [qmerrno.h](#) for `eQCWWAN_XXX` error values

Note

Timeout: 2 seconds

9.45.5.44 `ULONG SLQSGetProfileSettings (GetProfileSettingIn * pReq, GetProfileSettingOut * pResp)`

Retrieves a profile(3GPP/3GPP2) with the specified parameters.

Parameters

<i>pReq[IN]</i>	<ul style="list-style-type: none"> details of the profile to be fetched See GetProfileSettingIn for more information
<i>pResp[OUT]</i>	
Generated by Doxygen	<ul style="list-style-type: none"> The profile settings and/or extended error code returned by the device based on input parameters. See GetProfileSettingOut for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.45.5.45 ULONG SLQSGetRuntimeSettings (struct WdsRunTimeSettings * pRuntimeSettings)

Returns the packet data session settings currently in use.

Parameters

<i>pRuntimeSettings[OUT]</i>	<ul style="list-style-type: none">• SLQS Runtime Settings Information
------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.45.5.46 ULONG SLQSSetSessionState (ULONG * pStateV4, ULONG * pStateV6, BYTE instance)

Returns the state of the current packet data session.

Parameters

<i>pStateV4[OUT]</i>	<ul style="list-style-type: none">• Current link status for IPV4 Session<ul style="list-style-type: none">– 1 - DISCONNECTED– 2 - CONNECTED– 3 - SUSPENDED (not supported)– 4 - AUTHENTICATING
----------------------	---

<i>pStateV6[OUT]</i>	<ul style="list-style-type: none">• Current link status for IPV6 Session<ul style="list-style-type: none">– 1 - DISCONNECTED– 2 - CONNECTED– 3 - SUSPENDED (not supported)– 4 - AUTHENTICATING
<i>instance</i>	<ul style="list-style-type: none">• PDP instance

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.45.5.47 ULONG SLQSMModifyProfile (struct ModifyProfileIn * pReq, struct ModifyProfileOut * pResp)

Modify a profile(3GPP/3GPP2) with the specified parameters.

Parameters

<i>pReq[IN]</i>	<ul style="list-style-type: none">• Contains parameters which can be modified
<i>pResp[OUT]</i>	<ul style="list-style-type: none">• Contains parameters which indicates modification success or failure

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.45.5.48 **ULONG** SLQSResetPacketStatics ()

This API request the device to reset packet data transfer statistics.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 2 seconds\n

9.45.5.49 **ULONG** SLQSSet3GPPConfigItem (slqs3GPPConfigItem * pSLQS3GPPConfigItem)

Sets the 3gpp configuration item.

Parameters

<i>pSLQS3GPPConfigItem</i> [IN]	<ul style="list-style-type: none"> See slqs3GPPConfigItem for more information
---------------------------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS/LTE

Timeout: 2 seconds

9.45.5.50 **ULONG** SLQSSetProfile (**ULONG** profileType, **BYTE** profileId, **ULONG** * pPDPTType, **ULONG** * pIPAddress, **ULONG** * pPrimaryDNS, **ULONG** * pSecondaryDNS, **ULONG** * pAuthentication, **CHAR** * pName, **CHAR** * pAPNName, **CHAR** * pUsername, **CHAR** * pPassword)

Writes the profile settings for the specified profile Id.

Parameters

<i>profileType</i>	<ul style="list-style-type: none"> Type of profile <ul style="list-style-type: none"> 0 - UMTS
<i>profileId</i>	<ul style="list-style-type: none"> Profile number to be modified <ul style="list-style-type: none"> Value between 1 - 16 (EM/MC73xx or earlier) Value between 1 - 24 (EM/MC74xx onwards)
<i>pPDPTType[IN]</i>	<ul style="list-style-type: none"> Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile (optional) <ul style="list-style-type: none"> 0 - PDP-IP (IPv4)
<i>pIPAddress[IN]</i>	<ul style="list-style-type: none"> Preferred IPv4 address to be assigned to device (optional)
<i>pPrimaryDNS[IN]</i>	<ul style="list-style-type: none"> Primary DNS Ipv4 address preference (optional)
<i>pSecondaryDNS[IN]</i>	<ul style="list-style-type: none"> Secondary DNS Ipv4 address preference (optional)
<i>pAuthentication[IN]</i>	<ul style="list-style-type: none"> Bitmap that indicates authentication algorithm preference (optional) <ul style="list-style-type: none"> 0x00000001 - PAP preference <ul style="list-style-type: none"> 0 - Never performed 1 - May be performed 0x00000002 - CHAP preference <ul style="list-style-type: none"> 0 - Never performed 1 - May be performed All other bits are reserved and must be set to 0 If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g. the device may have a policy to select the most secure authentication mechanism.
<i>pName[IN]</i>	<ul style="list-style-type: none"> profile Name (optional)
<i>pAPNName[IN]</i>	<ul style="list-style-type: none"> Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network (optional) If value is NULL or omitted, then subscription default value will be requested.
<i>pUsername[IN]</i>	<ul style="list-style-type: none"> Username used during network authentication (optional)
<i>pPassword[IN]</i>	<ul style="list-style-type: none"> Password used during network authentication (optional)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout is 2 seconds.

9.45.5.51 ULONG SLQSSGetDHCPv4ClientConfig (WdsDHCPv4Config * pReqResp)

This API gets the DHCP Client V4 Configuration.

Parameters

<i>pReq</i>	[IN] <ul style="list-style-type: none">• See WdsDHCPv4Config for more information
-------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 2 seconds\n

9.45.5.52 ULONG SLQSSGetLoopback (WDSGetLoopbackData * data)

This API to Get the value of loopback mode and multiplier.

Parameters

<i>pReq</i>	[IN] <ul style="list-style-type: none">• See WDSGetLoopbackData for more information
-------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 2 seconds\n

9.45.5.53 ULONG SLQSSetDHCPv4ClientConfig (WdsDHCPv4Config * *pReq*)

This API sets the DHCP Client V4 Configuration.

Parameters

<i>pReq</i>	[IN] <ul style="list-style-type: none"> See WdsDHCPv4Config for more information
-------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 2 seconds\n

9.45.5.54 ULONG SLQSSetLoopback (WDSSetLoopbackData * *pReq*)

This API to Enable/disable Data Loopback Mode and set the value of loopback multiplier.

Parameters

<i>pReq</i>	[IN] <ul style="list-style-type: none"> See WDSSetLoopbackData for more information
-------------	---

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Timeout: 2 seconds\n

9.45.5.55 ULONG SLQSStartStopDataSession (struct ssdatasession_params * *pin*)

Starts or stops a 3GPP/3GPP2 data session on a preconfigured profile. To set the IP family for the data session, execute SLQSSetIPFamilyPreference prior to calling this API.

Parameters

<i>pin</i> [IN]	<ul style="list-style-type: none"> ssdatasession_params structure See ssdatasession_params for more details
Generated by Doxygen	

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 5 Minutes

Use [SLQSSetProfile](#) to configure 3GPP profiles

9.45.5.56 ULONG SLQSWdsGoActive (void)

Forces the device to immediately reestablish the traffic channel on the serving radio interface

Parameters

<i>None</i>	
-------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

This channel can go dormant any time after it has been reactivated. There is no assurance that the channel remains active for any guaranteed period. Timeout: 5 seconds

9.45.5.57 ULONG SLQSWdsGoDormant (void)

Forces the device to immediately drop the traffic channel on the serving radio interface

Parameters

<i>None</i>	
-------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

This channel can be reactivated as soon as data is sent over the network interface. There is no assurance that the channel remains dormant for any guaranteed period. Timeout: 5 seconds

9.45.5.58 ULONG SLQSWdsSetEventReport (wdsSetEventReportReq * pSetEventReportReq)

This API sets the wireless data connection state reporting conditions for the requesting control point.

Parameters

<i>pSetEventReportReq</i> [IN]	<ul style="list-style-type: none"> See wdsSetEventReportReq for more information.
--------------------------------	--

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

The control point event reporting state variables are modified to reflect the settings indicated in the request message. The service maintains a set of state variables for each control point. Relevant wireless data connection state changes are communicated to the registered WDS control point via the SLQSWdsSetEventReportCallBack. The AT command equivalents to this command are AT+CMER, AT+CIND, and AT+CIEV

9.45.5.59 ULONG SLQSWdsSwiPDPRuntimeSettings (swiPDPRuntimeSettingsReq * pPDPRuntimeSettingsReq, swiPDPRuntimeSettingsResp * pPDPRuntimeSettingsResp)

This API requests the device to return the active PDP context associated with a context id.

Parameters

<i>pPDPRuntimeSettingsReq</i> [IN]	<ul style="list-style-type: none"> See swiPDPRuntimeSettingsReq for more information
<i>pPDPRuntimeSettingsResp</i> [OUT]	<ul style="list-style-type: none"> See swiPDPRuntimeSettingsResp for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Technology Supported: UMTS/CDMA

Device Supported: MC77XX

Timeout: 2 seconds

The AT command equivalent to this command is AT+CGCONTRDP

9.45.5.60 **BOOL** WDS_IsGobiDevice ()

9.46 qaNasGetRFBandInfo.h File Reference

Data Structures

- struct [QmiNasGetRFBandInfoResp](#)

Enumerations

- enum [eQMI_NAS_GET_RF_INFO_RESP](#) { [eTLV_RF_BAND_INFO](#) = 0x01 }

Functions

- enum [eQCWWANError](#) [PkQmiNasGetRFBandInfo](#) ([WORD](#) *pMlength, [BYTE](#) *pBuffer)
- enum [eQCWWANError](#) [UpkQmiNasGetRFBandInfo](#) ([BYTE](#) *pMdmResp, struct [QmiNasGetRFBandInfoResp](#) *pApiResp)

9.46.1 Enumeration Type Documentation

9.46.1.1 enum [eQMI_NAS_GET_RF_INFO_RESP](#)

Enumerator

eTLV_RF_BAND_INFO

9.46.2 Function Documentation

9.46.2.1 enum eQCWWANError PkQmiNasGetRFBandInfo (WORD * pMlength, BYTE * pBuffer)

9.46.2.2 enum eQCWWANError UpkQmiNasGetRFBandInfo (BYTE * pMdmResp, struct QmiNasGetRFBandInfoResp * pApiResp)

9.47 qaNasPerformNetworkScan.h File Reference

Data Structures

- struct [QmiNas3GppNetworkInfo](#)
- struct [QmiNasPerformNetworkScanResp](#)

Macros

- #define [QMI_NAS_NETSTATUS_MASK](#) 0x03
- #define [QMI_NAS_MAX_INSTANCES](#) 20
- #define [INDEX_ZERO](#) 0
- #define [ROAMING_INDEX](#) 2
- #define [FORBIDDEN_INDEX](#) 4
- #define [PREFERRED_INDEX](#) 6
- #define [MAX_DESCRIPTION_LENGTH](#) 255

Enumerations

- enum [eQMI_NAS_PERFORM_NETWORK_SCAN_RESP](#) { [eTLV_3GPP_NETWORK_INFO](#) = 0x10 }

Functions

- enum [eQCWWANError](#) PkQmiNasPerformNetworkScan (WORD *pMlength, BYTE *pParamField)
- enum [eQCWWANError](#) UpkQmiNasPerformNetworkScan (BYTE *pMdmResp, struct [QmiNasPerformNetworkScanResp](#) *pAipResp)

9.47.1 Macro Definition Documentation

9.47.1.1 #define FORBIDDEN_INDEX 4

9.47.1.2 #define INDEX_ZERO 0

9.47.1.3 #define MAX_DESCRIPTION_LENGTH 255

9.47.1.4 #define PREFERRED_INDEX 6

9.47.1.5 #define QMI_NAS_MAX_INSTANCES 20

9.47.1.6 #define QMI_NAS_NETSTATUS_MASK 0x03

9.47.1.7 #define ROAMING_INDEX 2

9.47.2 Enumeration Type Documentation

9.47.2.1 enum eQMI_NAS_PERFORM_NETWORK_SCAN_RESP

Enumerator

[eTLV_3GPP_NETWORK_INFO](#)

9.47.3 Function Documentation

9.47.3.1 enum eQCWWANError PkQmiNasPerformNetworkScan (WORD * *pMlength*, BYTE * *pParamField*)

9.47.3.2 enum eQCWWANError UpkQmiNasPerformNetworkScan (BYTE * *pMdmResp*, struct QmiNasPerformNetworkScanResp * *pAipResp*)

9.48 qmerrno.h File Reference

Enumerations

- enum eQCWWANError {
 - eQCWWAN_ERR_ENUM_BEGIN = -1,
 - eQCWWAN_ERR_NONE,
 - eQCWWAN_ERR_GENERAL,
 - eQCWWAN_ERR_INTERNAL,
 - eQCWWAN_ERR_MEMORY,
 - eQCWWAN_ERR_INVALID_ARG,
 - eQCWWAN_ERR_BUFFER_SZ,
 - eQCWWAN_ERR_NO_DEVICE,
 - eQCWWAN_ERR_INVALID_DEVID,
 - eQCWWAN_ERR_NO_CONNECTION,
 - eQCWWAN_ERR_QMI_IFACE,
 - eQCWWAN_ERR_QMI_CONNECT,
 - eQCWWAN_ERR_QMI_REQ_SCH,
 - eQCWWAN_ERR_QMI_REQ,
 - eQCWWAN_ERR_QMI_RSP,
 - eQCWWAN_ERR_QMI_REQ_TO,
 - eQCWWAN_ERR_QMI_RSP_TO,
 - eQCWWAN_ERR_MALFORMED_QMI_RSP,
 - eQCWWAN_ERR_INVALID_QMI_RSP,
 - eQCWWAN_ERR_INVALID_FILE,
 - eQCWWAN_ERR_FILE_OPEN,
 - eQCWWAN_ERR_FILE_COPY,
 - eQCWWAN_ERR_OFFLINE = 27,
 - eQCWWAN_ERR_RESET,
 - eQCWWAN_ERR_NO_SIGNAL,
 - eQCWWAN_ERR_MULTIPLE_DEVICES,
 - eQCWWAN_ERR_DRIVER,
 - eQCWWAN_ERR_NO_CANCELABLE_OP,
 - eQCWWAN_ERR_CANCEL_OP,
 - eQCWWAN_ERR_API_MUTEX_TIMEOUT,
 - eQCWWAN_ERR_PDU_GENERATION,
 - eQCWWAN_ERR_INVALID_XID,
 - eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED,
 - eQCWWAN_ERR_ENUM_END,
 - eQCWWAN_ERR_QMI_OFFSET = 1000,
 - eQCWWAN_ERR_QMI_MALFORMED_MSG = 1001,
 - eQCWWAN_ERR_QMI_NO_MEMORY,
 - eQCWWAN_ERR_QMI_INTERNAL,
 - eQCWWAN_ERR_QMI_ABORTED,
 - eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED,
 - eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION,
 - eQCWWAN_ERR_QMI_INVALID_CLIENT_ID,
 - eQCWWAN_ERR_QMI_NO_THRESHOLDS,
 - eQCWWAN_ERR_QMI_INVALID_HANDLE,
 - eQCWWAN_ERR_QMI_INVALID_PROFILE,
 - eQCWWAN_ERR_QMI_INVALID_PINID,
 - eQCWWAN_ERR_QMI_INCORRECT_PIN,
 - eQCWWAN_ERR_QMI_NO_NETWORK_FOUND,
 - eQCWWAN_ERR_QMI_CALL_FAILED,
 - eQCWWAN_ERR_QMI_OUT_OF_CALL,
 - eQCWWAN_ERR_QMI_NOT_PROVISIONED,
 - eQCWWAN_ERR_QMI_MISSING_ARG,
 - eQCWWAN_ERR_QMI_ARG_TOO_LONG = 1019,
 - eQCWWAN_ERR_QMI_INVALID_TX_ID = 1022,
 - eQCWWAN_ERR_QMI_DEVICE_IN_USE,
 - eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED,
 - eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED,
 - eQCWWAN_ERR_QMI_NO_EFFECT,
 - eQCWWAN_ERR_QMI_NO_FREE_PROFILE,
 - eQCWWAN_ERR_QMI_INVALID_PDP_TYPE,
 - eQCWWAN_ERR_QMI_INVALID_TECH_PREF,
 - eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE,

```

eQCWWAN_ERR_QMI_WIDTH = 0xFFFF }
• enum qm_wds_ds_profile_extended_err_codes {
eWDS_ERR_PROFILE_REG_RESULT_FAIL = 1,
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_HNDL,
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_OP,
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_PROFILE_TYPE,
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_PROFILE_NUM,
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_IDENT,
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID,
eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED,
eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID,
eWDS_ERR_PROFILE_REG_RESULT_LIST_END,
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_SUBS_ID,
eWDS_ERR_PROFILE_REG_INVALID_PROFILE_FAMILY,
eWDS_ERR_PROFILE_REG_3GPP_INVALID_PROFILE_FAMILY = 1001,
eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR,
eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED,
eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET,
eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET,
eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES,
eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE = 1101,
eWDS_ERR_PROFILE_REG_END }

```

9.48.1 Enumeration Type Documentation

9.48.1.1 enum eQCWWANError

QMI Error Code Enumeration

Enumerator

```

eQCWWAN_ERR_ENUM_BEGIN
eQCWWAN_ERR_NONE 00 - Success
eQCWWAN_ERR_GENERAL 01 - General error
eQCWWAN_ERR_INTERNAL 02 - Internal error
eQCWWAN_ERR_MEMORY 03 - Memory error
eQCWWAN_ERR_INVALID_ARG 04 - Invalid argument
eQCWWAN_ERR_BUFFER_SZ 05 - Buffer too small
eQCWWAN_ERR_NO_DEVICE 06 - Unable to detect WWAN device
eQCWWAN_ERR_INVALID_DEVID 07 - Invalid WWAN device ID
eQCWWAN_ERR_NO_CONNECTION 08 - No connection to WWAN device
eQCWWAN_ERR_QMI_IFACE 09 - Unable to obtain QMI interface
eQCWWAN_ERR_QMI_CONNECT 10 - Unable to connect to QMI interface
eQCWWAN_ERR_QMI_REQ_SCH 11 - Unable to schedule QMI request
eQCWWAN_ERR_QMI_REQ 12 - Error sending QMI request
eQCWWAN_ERR_QMI_RSP 13 - Error receiving QMI response
eQCWWAN_ERR_QMI_REQ_TO 14 - Timeout while sending QMI request
eQCWWAN_ERR_QMI_RSP_TO 15 - Timeout while receiving QMI response
eQCWWAN_ERR_MALFORMED_QMI_RSP 16 - Malformed QMI response received
eQCWWAN_ERR_INVALID_QMI_RSP 17 - Invalid QMI response received

```

eQCWWAN_ERR_INVALID_FILE 18 - Invalid file path

eQCWWAN_ERR_FILE_OPEN 19 - Unable to open file

eQCWWAN_ERR_FILE_COPY 20 - Unable to copy file

eQCWWAN_ERR_OFFLINE 27 - Unable to set WWAN device offline

eQCWWAN_ERR_RESET 28 - Unable to reset WWAN device

eQCWWAN_ERR_NO_SIGNAL 29 - No available signal

eQCWWAN_ERR_MULTIPLE_DEVICES 30 - Multiple WWAN devices detected

eQCWWAN_ERR_DRIVER 31 - Error interfacing to driver

eQCWWAN_ERR_NO_CANCELABLE_OP 32 - No cancelable operation is pending

eQCWWAN_ERR_CANCEL_OP 33- Error canceling outstanding operation

eQCWWAN_ERR_API_MUTEX_TIMEOUT 34- api mutex lock timeout

eQCWWAN_ERR_PDU_GENERATION 35- PDU generation error

eQCWWAN_ERR_INVALID_XID 36- Invalid transaction id

eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED 37- Unsupported multiple SMS

eQCWWAN_ERR_ENUM_END End of SLQS SDK specific error codes

eQCWWAN_ERR_QMI_OFFSET 1000 - This is not an error code but the offset from which mapped QMI error codes start from

eQCWWAN_ERR_QMI_MALFORMED_MSG 1001 - Malformed or Corrupted QMI msg

eQCWWAN_ERR_QMI_NO_MEMORY 1002 - Device could not allocate memory for QMI Resp

eQCWWAN_ERR_QMI_INTERNAL 1003 - Unexpected error occurred during processing

eQCWWAN_ERR_QMI_ABORTED 1004 - Processing aborted

eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED 1005 - QMI client IDs have been exhausted

eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION 1006 - Unable to abort QMI transaction

eQCWWAN_ERR_QMI_INVALID_CLIENT_ID 1007 - Invalid QMI client ID

eQCWWAN_ERR_QMI_NO_THRESHOLDS 1008 - No thresholds were provided

eQCWWAN_ERR_QMI_INVALID_HANDLE 1009 - Invalid Handle provided in the QMI request

eQCWWAN_ERR_QMI_INVALID_PROFILE 1010 - Profile specified is invalid

eQCWWAN_ERR_QMI_INVALID_PINID 1011 - Invalid PIN ID specified

eQCWWAN_ERR_QMI_INCORRECT_PIN 1012 - Incorrect PIN ID specified

eQCWWAN_ERR_QMI_NO_NETWORK_FOUND 1013 - No network found

eQCWWAN_ERR_QMI_CALL_FAILED 1014 - Call failed

eQCWWAN_ERR_QMI_OUT_OF_CALL 1015 - Device is not in a call

eQCWWAN_ERR_QMI_NOT_PROVISIONED 1016 - Requested information element not provisioned on device

eQCWWAN_ERR_QMI_MISSING_ARG 1017 - Mandatory QMI TLV not provided

eQCWWAN_ERR_QMI_ARG_TOO_LONG 1019 - Arg passed in QMI TLV larger than available storage in device

eQCWWAN_ERR_QMI_INVALID_TX_ID 1022 - Invalid TX ID specified

eQCWWAN_ERR_QMI_DEVICE_IN_USE 1023 - Device currently in a call

eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED 1024 - The selected operation is not supported by the network

eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED 1025 - The selected operation is not supported by the device

eQCWWAN_ERR_QMI_NO_EFFECT 1026 - Requested operation would have no effect

eQCWWAN_ERR_QMI_NO_FREE_PROFILE 1027 - No space for a profile is available

eQCWWAN_ERR_QMI_INVALID_PDP_TYPE 1028 - Invalid PDP type specified

eQCWWAN_ERR_QMI_INVALID_TECH_PREF 1029 - Invalid technology preference specified

eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE 1030 - Invalid profile type specified

eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE 1031 - Invalid service type specified

eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION 1032 - Invalid register action specified

eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION 1033 - Invalid PS attach/detach action specified

eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED 1034 - Authentication of supplied information element failed

eQCWWAN_ERR_QMI_PIN_BLOCKED 1035 - PIN is blocked; an unblock operation needs to be issued

eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED 1036 - PIN is permanently blocked; the UIM is unusable

eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED 1037 - UIM initialization has not completed

eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE 1038 - Max QOS requests are used

eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER 1039 - The Flow filter is incorrect

eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE 1040 - Network unaware of the QOS requested

eQCWWAN_ERR_QMI_INVALID_ID 1041 - Invalid QOS ID

eQCWWAN_ERR_QMI_INVALID_QOS_ID 1041 - Invalid QOS ID

eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSUPPORTED 1042 - The request number is not supported

eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND 1043 - Unable to find the interface

eQCWWAN_ERR_QMI_FLOW_SUSPENDED 1044 - Flow suspended

eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT 1045 - Data format is invalid

eQCWWAN_ERR_QMI_GENERAL 1046 - General error

eQCWWAN_ERR_QMI_UNKNOWN 1047 - Unknown error

eQCWWAN_ERR_QMI_INVALID_ARG 1048 - A specified argument is invalid

eQCWWAN_ERR_QMI_INVALID_INDEX 1049 - A specified index is invalid

eQCWWAN_ERR_QMI_NO_ENTRY 1050 - No information element exists at specified memory designation

eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL 1051 - The memory storage specified in the request is full

eQCWWAN_ERR_QMI_DEVICE_NOT_READY 1052 - Device not in a ready state

eQCWWAN_ERR_QMI_NETWORK_NOT_READY 1053 - Network not in a ready state

eQCWWAN_ERR_QMI_CAUSE_CODE 1054 - Error provided in SMS cause code

eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT 1055 - The message could not be sent

eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE 1056 - The message could not be delivered

eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID 1057 - The message ID specified for the message is invalid

eQCWWAN_ERR_QMI_ENCODING 1058 - The message is not encoded properly

eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK 1059 - Maximum number of authentication failures has been reached

eQCWWAN_ERR_QMI_INVALID_TRANSITION 1060 - Operating mode transition from the current mode is invalid

eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE 1061 - The interace is not muticast

eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE 1062 - Maximum requests in use

eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE 1063 - Invalid muticast handle

eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF 1064 - Invalid IP family preference

eQCWWAN_ERR_QMI_SESSION_INACTIVE 1065 - No tracking session has been started

eQCWWAN_ERR_QMI_SESSION_INVALID 1066 - Current session does not allow this operation

eQCWWAN_ERR_QMI_SESSION_OWNERSHIP 1067 - Current tracking session not started by this QMI control point

eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCES 1068 - Device GPS service resources insufficient for request

eQCWWAN_ERR_QMI_DISABLED 1069 - Device GPS service disabled

eQCWWAN_ERR_QMI_INVALID_OPERATION 1070 - Invalid operation specified

eQCWWAN_ERR_QMI_INVALID_QMI_CMD 1071 - Invalid/unknown QMI command specified

eQCWWAN_ERR_QMI_TPDU_TYPE 1072 - Message contains TPDU type that cannot be read as raw message

eQCWWAN_ERR_QMI_SMSC_ADDR 1073 - The SMSC address specified is invalid

eQCWWAN_ERR_QMI_INFO_UNAVAILABLE 1074 - Information element is unavailable at this point

eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG 1075 - Segment size too large

eQCWWAN_ERR_QMI_SEGMENT_ORDER 1076 - Segment order is incorrect

eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED 1077 - Bundling not supported

eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE 1078 - The operation failed partially

eQCWWAN_ERR_QMI_POLICY_MISMATCH 1079 - Policy mismatch

eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND 1080 - SIM file not found

eQCWWAN_ERR_QMI_EXTENDED_INTERNAL 1081 - Extended internal error

eQCWWAN_ERR_QMI_ACCESS_DENIED 1082 - Access to a required entity is not available

eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED 1083 - Selected operating mode is invalid with current hardware setting

eQCWWAN_ERR_QMI_ACK_NOT_SENT 1084 - ACK not sent

eQCWWAN_ERR_QMI_INJECT_TIMEOUT 1084 - Inject a timeout for the request

eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE 1090 - Incompatible state

eQCWWAN_ERR_QMI_FDN_RESTRICT 1091 - FDN Restrict

eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE 1092 - SUPS failure cause

eQCWWAN_ERR_QMI_NO_RADIO 1093 - No Radio

eQCWWAN_ERR_QMI_NOT_SUPPORTED 1094 - Not Supported

eQCWWAN_ERR_QMI_NO_SUBSCRIPTION 1095 - No Subscription

eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED 1096 - Card call control failed

eQCWWAN_ERR_QMI_NETWORK_ABORTED 1097 - Network Aborted

eQCWWAN_ERR_QMI_MSG_BLOCKED 1098 - Open Error

eQCWWAN_ERR_QMI_MAX Error - End of QMI specific defines

eQCWWAN_ERR_SWICM_START Vendor defines - **Connection Manager error codes**

eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED 0xE001 - The API is yet to be implemented

eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED 0xE002 - The service is not supported

eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED 0xE003 - The client is not supported

eQCWWAN_ERR_SWICM_TIMEOUT 0xE004 - API Timeout

eQCWWAN_ERR_SWICM_SOCKET_IN_USE 0xE005 - The communication socket is in use

eQCWWAN_ERR_SWICM_AM_VERS_ERROR 0xE006 - SLQS API and SDK version mismatch

eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS 0xE007 - Failed to kill SDK process

eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS 0xE008 - Call in progress

eQCWWAN_ERR_SWICM_V4DWN_V6DWN 0xE009 - IPV4 and IPV6 is down

eQCWWAN_ERR_SWICM_V4DWN_V6UP 0xE00A - IPV4 is down and IPV6 is up

eQCWWAN_ERR_SWICM_V4UP_V6DWN 0xE00B - IPV4 is up and IPV6 is down

eQCWWAN_ERR_SWICM_V4UP_V6UP 0xE00C - IPV4 and IPV6 is up

eQCWWAN_ERR_SWICM_INVALID_SESSION_ID 0xE00D - Invalid V4 Session ID
eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID 0xE00E - Invalid V4 Session ID
eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID 0xE00F - Invalid V6 Session ID
eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS 0xE010 - No available Session Manager slots for additional data sessions
eQCWWAN_ERR_SWICM_END 0xE011 - End of connection manager specific codes
eQCWWAN_ERR_SWISMS_START Vendor defines - SMS Error codes
eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG 0xE101 - SMS message length is long
eQCWWAN_ERR_SWISMS_MSG_CORRUPTED 0xE102 - The SMS message is corrupted (encoding wrong)
eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED 0xE103 - The SMS number is corrupted (incorrect number)
eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND 0xE104 - The SMS bearer data is not available

eQCWWAN_ERR_SWISM_END
eQCWWAN_ERR_SWIIM_START Vendor defines - Image Management error codes
eQCWWAN_ERR_SWIIM_INVALID_PATH 0xE801 - Invalid directory path
eQCWWAN_ERR_SWIIM_OPENING_DIR 0xE802 - Unable to open the directory
eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND 0xE803 - No Firmware image present in the path
eQCWWAN_ERR_SWIIM_OPENING_FILE 0xE804 - Unable to open the file
eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE 0xE805 - Firmware image is corrupted
eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED 0xE806 - No Firmware image download needed
eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL 0xE807 - Firmware update failed
eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH 0xE808 - Update success but pri/fw preference mismatch
eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS 0xE809 - Update successful
eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE 0xE80A - Enter Download Mode
eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE 0xE80B - File transfer to modem complete
eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT 0xE80C - Wait for modem to reboot
eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE 0xE80D - Invalid Crash State for Firmware Download
eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE 0xE80E - Same as current active image
eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX 0xE80F - invalid slot index
eQCWWAN_ERR_SWIIM_END
eQCWWAN_ERR_SWIDCS_START Vendor defines - Device Connectivity error codes
eQCWWAN_ERR_SWIDCS_IOCTL_ERR 0xE901 - IO Control error
eQCWWAN_ERR_SWIDCS_FILEIO_ERR 0xE902 - file open/read/write error
eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND 0xE903 - The device is not found
eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED 0xE904 - Application is disconnected from SDK
eQCWWAN_ERR_SWIDCS_END
eQCWWAN_ERR_QMI_CAT_START QMI errors related to CAT
eQCWWAN_ERR_QMI_EVENT_REG_FAILED 62441 - CAT event registration failed
eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP 62442 - Invalid terminal response
eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD 62443 - Invalid envelope command
eQCWWAN_ERR_QMI_CARD_BUSY_RSP 62444 - Card busy response for envelope command
eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE 62445 - Envelope command failure
eQCWWAN_ERR_QMI_CAT_END
eQCWWAN_ERR_NULL_TLV
eQCWWAN_ERR_QMI_WIDTH 0xFFFF - Not an error, represent the end of QMI errors

9.48.1.2 enum qm_wds_ds_profile_extended_err_codes

WDS DS profile extended error codes

Enumerator

- eWDS_ERR_PROFILE_REG_RESULT_FAIL** 1 - General Failure
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_HANDLE** 2 - The request contains an invalid profile handle
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_OP** 3 - An invalid operation was requested.
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_PROFILE_TYPE** 4 - The request contains an invalid technology type
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_PROFILE_NUM** 5 - The request contains an invalid profile number
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_IDENT** 6 - The request contains an invalid profile identifier
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID** 7 - The request contains an invalid argument other than profile number and profile identifier received.
- eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED** 8 - Profile registry has not been initialized yet
- eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID** 9 - The request contains a parameter with invalid length.
- eWDS_ERR_PROFILE_REG_RESULT_LIST_END** 10 - End of the profile list was reached while searching for the requested profile.
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_SUBS_ID** 11 - The request contains an invalid subscription identifier.
- eWDS_ERR_PROFILE_REG_INVALID_PROFILE_FAMILY** 12 - The request contains an invalid profile family.
- eWDS_ERR_PROFILE_REG_3GPP_INVALID_PROFILE_FAMILY** 1001 - The request contains an invalid 3GPP profile family.
- eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR** 1002 - An error was encountered while accessing the 3GPP profiles.
- eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED** 1003 - The given 3GPP profile doesn't have a valid context.
- eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET** 1004 - The given 3GPP profile is marked invalid.
- eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET** 1005 - The given 3GPP profile is marked read-only.
- eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES** 1006 - Creation of a new 3GPP profile failed because the limit of 16 profiles has already been reached.
- eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE** 1101 - An invalid profile identifier was received as part of the 3GPP2 profile modification request.
- eWDS_ERR_PROFILE_REG_END**

9.49 qos.h File Reference

Data Structures

- struct [unpack_qos_SLQSQosGetNetworkStatus_t](#)
- struct [pack_qos_SLQSQosSmiReadApnExtraParams_t](#)

- struct [unpack_qos_SLQSQosSwiReadApnExtraParams_t](#)
- struct [pack_qos_SLQSQosSwiReadDataStats_t](#)
- struct [unpack_qosFlowStat_t](#)
- struct [unpack_qos_SLQSQosSwiReadDataStats_t](#)
- struct [unpack_qos_SLQSSetQosNWStatusCallback_ind_t](#)
- struct [unpack_qos_SLQSSetQosStatusCallback_ind_t](#)
- struct [unpack_qos_SLQSSetQosPriEventCallback_ind_t](#)
- struct [pack_qos_SLQSSetQosEventCallback_t](#)
- struct [unpack_qos_qosFlowInfoState_t](#)
- struct [unpack_qos_dataRate_t](#)
- struct [unpack_qos_tokenBucket_t](#)
- struct [unpack_qos_pktErrRate_t](#)
- struct [unpack_qos_swiQosFlow_t](#)
- struct [unpack_qos_IPv4Addr_t](#)
- struct [unpack_qos_Tos_t](#)
- struct [unpack_qos_IPv6Addr_t](#)
- struct [unpack_qos_IPv6TrafCls_t](#)
- struct [unpack_qos_Port_t](#)
- struct [unpack_qos_swiQosFilter_t](#)
- struct [unpack_qos_qosFlowInfo_t](#)
- struct [unpack_qos_SLQSSetQosEventCallback_ind_t](#)

Macros

- `#define LIBPACK_MAX_QOS_FLOW_PER_APN_STATS 10`
- `#define LIBPACK_MAX_QOS_FILTERS 25`
- `#define LIBPACK_MAX_QOS_FLOWS 8`

Functions

- int [pack_qos_SLQSQosGetNetworkStatus](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_qos_SLQSQosGetNetworkStatus](#) (uint8_t *pResp, uint16_t respLen, [unpack_qos_SLQSQosGetNetworkStatus_t](#) *pOutput)
- int [pack_qos_SLQSQosSwiReadApnExtraParams](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_qos_SLQSQosSwiReadApnExtraParams_t](#) reqParam)
- int [unpack_qos_SLQSQosSwiReadApnExtraParams](#) (uint8_t *pResp, uint16_t respLen, [unpack_qos_SLQSQosSwiReadApnExtraParams_t](#) *pOutput)
- int [pack_qos_SLQSQosSwiReadDataStats](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_qos_SLQSQosSwiReadDataStats_t](#) reqParam)
- int [unpack_qos_SLQSQosSwiReadDataStats](#) (uint8_t *pResp, uint16_t respLen, [unpack_qos_SLQSQosSwiReadDataStats_t](#) *pOutput)
- int [unpack_qos_SLQSSetQosNWStatusCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_qos_SLQSSetQosNWStatusCallback_ind_t](#) *pOutput)
- int [unpack_qos_SLQSSetQosStatusCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_qos_SLQSSetQosStatusCallback_ind_t](#) *pOutput)
- int [unpack_qos_SLQSSetQosPriEventCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_qos_SLQSSetQosPriEventCallback_ind_t](#) *pOutput)
- int [pack_qos_SLQSSetQosEventCallback](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_qos_SLQSSetQosEventCallback_t](#) reqParam)
- int [unpack_qos_SLQSSetQosEventCallback](#) (uint8_t *pResp, uint16_t respLen)
- int [unpack_qos_SLQSSetQosEventCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_qos_SLQSSetQosEventCallback_ind_t](#) *pOutput)

9.49.1 Macro Definition Documentation

9.49.1.1 `#define LIBPACK_MAX_QOS_FILTERS 25`

9.49.1.2 `#define LIBPACK_MAX_QOS_FLOW_PER_APN_STATS 10`

9.49.1.3 `#define LIBPACK_MAX_QOS_FLOWS 8`

9.49.2 Function Documentation

9.49.2.1 `int pack_qos_SLQSQosGetNetworkStatus (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Function to pack command to retrieve QoS status of the network. This maps to SLQSQosGetNetworkStatus

Parameters

<i>pCtx</i> [OUT]	<ul style="list-style-type: none"> See pack_qmi_t for more information
<i>pReqBuf</i> [IN/OUT]	<ul style="list-style-type: none"> Buffer for packed QMI command to be provided by the host application Minimum expected size is 2048 bytes
<i>pLen</i> [IN/OUT]	<ul style="list-style-type: none"> On input, size of pReqBuf On output, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
- Technology Supported: CDMA
- PDN Specific: No

9.49.2.2 `int pack_qos_SLQSQosSviReadApnExtraParams (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_qos_SLQSQosSviReadApnExtraParams_t reqParam)`

Function to pack QMI command to query extra APN parameters This maps to SLQSQosSviReadApnExtraParams

Parameters

<i>pCtx</i> [OUT]	<ul style="list-style-type: none"> See pack_qmi_t for more information
<i>pReqBuf</i> [IN/OUT]	<ul style="list-style-type: none"> Buffer for packed QMI command to be provided by the host application Minimum expected size is 2048 bytes
<i>pLen</i> [IN/OUT]	<ul style="list-style-type: none"> On input, size of <i>pReqBuf</i> On output, number of bytes actually packed
<i>reqParam</i> [IN]	<ul style="list-style-type: none"> See pack_qos_SLQSQosSwiReadApnExtraParams_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
 - PDN Specific: Yes

9.49.2.3 `int pack_qos_SLQSQosSwiReadDataStats (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_qos_SLQSQosSwiReadDataStats_t reqParam)`

Function to pack QMI command to query APN data statistics This maps to SLQSQosSwiReadDataStats

Parameters

<i>pCtx</i> [OUT]	<ul style="list-style-type: none"> See pack_qmi_t for more information
<i>pReqBuf</i> [IN/OUT]	<ul style="list-style-type: none"> Buffer for packed QMI command to be provided by the host application Minimum expected size is 2048 bytes
<i>pLen</i> [IN/OUT]	<ul style="list-style-type: none"> On input, size of <i>pReqBuf</i> On output, number of bytes actually packed
<i>reqParam</i> [IN]	<ul style="list-style-type: none"> See pack_qos_SLQSQosSwiReadDataStats_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
 - PDN Specific: Yes

9.49.2.4 `int pack_qos_SLQSSetQosEventCallback (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_qos_SLQSSetQosEventCallback_t reqParam)`

Function to pack QMI command to enable QoS event indications This maps to SLQSSetQosEventCallback

Parameters

<i>pCtx</i> [OUT]	<ul style="list-style-type: none"> • See pack_qmi_t for more information
<i>pReqBuf</i> [IN/OUT]	<ul style="list-style-type: none"> • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
<i>pLen</i> [IN/OUT]	<ul style="list-style-type: none"> • On input, size of pReqBuf • On output, number of bytes actually packed
<i>reqParam</i> [IN]	<ul style="list-style-type: none"> • See pack_qos_SLQSSetQosEventCallback_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
 - PDN Specific: Yes

9.49.2.5 `int unpack_qos_SLQSQosGetNetworkStatus (uint8_t * pResp, uint16_t respLen,
unpack_qos_SLQSQosGetNetworkStatus_t * pOutput)`

Function to unpack the response to get NW QoS status command This maps to SLQSQosGetNetworkStatus

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none"> Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none"> Length of pResp from modem
<i>pOutput</i> [OUT]	<ul style="list-style-type: none"> See unpack_qos_SLQSQosGetNetworkStatus_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.49.2.6 `int unpack_qos_SLQSQosSwiReadApnExtraParams (uint8_t * pResp, uint16_t respLen,
unpack_qos_SLQSQosSwiReadApnExtraParams_t * pOutput)`

Function to unpack the response to get NW QoS status command This maps to SLQSQosSwiReadApnExtraParams

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none"> Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none"> Length of pResp from modem
<i>pOutput</i> [OUT]	<ul style="list-style-type: none"> See unpack_qos_SLQSQosSwiReadApnExtraParams_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.49.2.7 `int unpack_qos_SLQSQosSwiReadDataStats (uint8_t * pResp, uint16_t respLen,
unpack_qos_SLQSQosSwiReadDataStats_t * pOutput)`

Function to unpack APN data statistics response This maps to SLQSQosSwiReadDataStats

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none">• Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none">• Length of pResp from modem
<i>pOutput</i> [OUT]	<ul style="list-style-type: none">• See unpack_qos_SLQSQosSwiReadDataStats_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.49.2.8 `int unpack_qos_SLQSSetQosEventCallback (uint8_t * pResp, uint16_t respLen)`

Function to unpack enable QoS event indications command's response This maps to SLQSSetQosEventCallback

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none">• Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none">• Length of pResp from modem

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.49.2.9 `int unpack_qos_SLQSSetQosEventCallback_ind (uint8_t * pResp, uint16_t respLen,
unpack_qos_SLQSSetQosEventCallback_ind_t * pOutput)`

Function to unpack QoS event indications This maps to SLQSSetQosEventCallback

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none"> • Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none"> • Length of pResp from modem
<i>pOutput</i> [OUT]	<ul style="list-style-type: none"> • See unpack_qos_SLQSSetQosEventCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_NETWORK_STATUS_IND indication to identify this event from QOS service read function

9.49.2.10 int unpack_qos_SLQSSetQosNWStatusCallback_ind (uint8_t * *pResp*, uint16_t *respLen*,
unpack_qos_SLQSSetQosNWStatusCallback_ind_t * *pOutput*)

Function to unpack QoS NW status indication. This maps to SLQSSetQosNWStatusCallback

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none"> • Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none"> • Length of pResp from modem
<i>pOutput</i> [OUT]	<ul style="list-style-type: none"> • See unpack_qos_SLQSSetQosNWStatusCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

- Technology Supported: CDMA
- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_NETWORK_STATUS_IND indication to identify this event from QOS service read function

9.49.2.11 `int unpack_qos_SLQSSetQosPriEventCallback_ind (uint8_t * pResp, uint16_t respLen, unpack_qos_SLQSSetQosPriEventCallback_ind_t * pOutput)`

Function to unpack QoS primary flow events. This maps to SLQSSetQosPriEventCallback

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none"> • Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none"> • Length of pResp from modem
<i>pOutput</i> [OUT]	<ul style="list-style-type: none"> • See unpack_qos_SLQSSetQosPriEventCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_PRIMARY_QOS_EVENT_IND indication to identify this event from QOS service read function
- This is only generated when the primary flow is modified by the host

9.49.2.12 `int unpack_qos_SLQSSetQosStatusCallback_ind (uint8_t * pResp, uint16_t respLen, unpack_qos_SLQSSetQosStatusCallback_ind_t * pOutput)`

Function to unpack QoS status indications. This maps to SLQSSetQosStatusCallback

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none"> • Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none"> • Length of pResp from modem
<i>pOutput</i> [OUT]	<ul style="list-style-type: none"> • See unpack_qos_SLQSSetQosStatusCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_FLOW_STATUS_IND indication to identify this event from QOS service read function

9.50 sms.h File Reference

Data Structures

- struct [pack_sms_SLQSGetSMS_t](#)
- struct [unpack_sms_SLQSGetSMS_t](#)
- struct [pack_sms_SLQSGetSMSList_t](#)
- struct [qmiSmsMessageList](#)
- struct [unpack_sms_SLQSGetSMSList_t](#)
- struct [pack_sms_SLQSMModifySMSStatus_t](#)
- struct [unpack_sms_SLQSMModifySMSStatus_t](#)
- struct [pack_sms_SLQSDDeleteSMS_t](#)
- struct [unpack_sms_SLQSDDeleteSMS_t](#)
- struct [pack_sms_SendSMS_t](#)
- struct [unpack_sms_SendSMS_t](#)
- struct [pack_sms_SetNewSMSCallback_t](#)
- struct [unpack_sms_SetNewSMSCallback_t](#)
- struct [sMSMTMessage](#)
- struct [newMTMessageTlv](#)
- struct [sMSTransferRouteMTMessage](#)
- struct [transferRouteMessageTlv](#)
- struct [sMSMessageMode](#)
- struct [messageModeTlv](#)
- struct [sMSEtwsMessage](#)
- struct [sMSEtwsMessageTlv](#)

- struct [sMSEtwSPlmn](#)
- struct [eTWSPLMNInfoTlv](#)
- struct [sMSCAddress](#)
- struct [sMSCAddressTlv](#)
- struct [sMSONIMS](#)
- struct [sMSONIMSTlv](#)
- struct [unpack_sms_SetNewSMSCallback_ind_t](#)
- struct [unpack_sms_SLQSWmsMemoryFullCallBack_ind_t](#)

Macros

- #define [MAX_SMS_MESSAGE_SIZE](#) 2048
- #define [MAX_SMS_LIST_SIZE](#) 255
- #define [MAX_MS_TRANSFER_ROUTE_MSG](#) 256
- #define [MAX_MSE_TWS_MSG](#) 1254
- #define [MAX_MSC_ADDRESS_SIZE](#) 256
- #define [MAX_CDMA_ENC_MO_TXT_MSG_SIZE](#) 255

Typedefs

- typedef struct [sMSMTMessage](#) [sMSMTMessageInfo](#)
- typedef struct [sMSTransferRouteMTMessage](#) [sMSTransferRouteMTMessageInfo](#)
- typedef struct [sSMSMessageMode](#) [sSMSMessageModelInfo](#)
- typedef struct [sMSEtwSMessage](#) [sMSEtwSMessageInfo](#)
- typedef struct [sMSEtwSPlmn](#) [sMSEtwSPlmnInfo](#)
- typedef struct [sMSCAddress](#) [sMSCAddressInfo](#)
- typedef struct [sMSONIMS](#) [sMSONIMSInfo](#)

Enumerations

- enum [eqmiCbKsetStatus](#) {
[LIBPACK_QMI_CBK_PARAM_RESET](#) = 0,
[LIBPACK_QMI_CBK_PARAM_SET](#) = 1,
[LIBPACK_QMI_CBK_PARAM_NOCHANGE](#) }

Functions

- int [pack_sms_SLQSGetSMS](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_sms_SLQSGetSMS_t](#) *reqParam)
- int [unpack_sms_SLQSGetSMS](#) (uint8_t *pResp, uint16_t respLen, [unpack_sms_SLQSGetSMS_t](#) *pOutput)
- int [pack_sms_SLQSGetSMSList](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_sms_SLQSGetSMSList_t](#) *reqParam)
- int [unpack_sms_SLQSGetSMSList](#) (uint8_t *pResp, uint16_t respLen, [unpack_sms_SLQSGetSMSList_t](#) *pOutput)
- int [pack_sms_SLQSModifySMSStatus](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_sms_SLQSModifySMSStatus_t](#) *reqParam)
- int [unpack_sms_SLQSModifySMSStatus](#) (uint8_t *pResp, uint16_t respLen, [unpack_sms_SLQSModifySMSStatus_t](#) *pOutput)
- int [pack_sms_SLQSDeleteSMS](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_sms_SLQSDeleteSMS_t](#) *reqParam)

- int [unpack_sms_SLQSDDeleteSMS](#) (uint8_t *pResp, uint16_t respLen, [unpack_sms_SLQSDDeleteSMS_t](#) *p↵
Output)
- int [pack_sms_SendSMS](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_sms_SendSMS_t](#) ↵
*reqParam)
- int [unpack_sms_SendSMS](#) (uint8_t *pResp, uint16_t respLen, [unpack_sms_SendSMS_t](#) *pOutput)
- int [pack_sms_SetNewSMSCallback](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_sms_Set](#)↵
[NewSMSCallback_t](#) reqParam)
- int [unpack_sms_SetNewSMSCallback](#) (uint8_t *pResp, uint16_t respLen, [unpack_sms_SetNewSMS](#)↵
[Callback_t](#) *Output)
- int [unpack_sms_SetNewSMSCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_sms_SetNewSMS](#)↵
[Callback_ind_t](#) *pOutput)
- int [unpack_sms_SLQSWmsMemoryFullCallBack_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_sms_SL](#)↵
[QSWmsMemoryFullCallBack_ind_t](#) *pOutput)

9.50.1 Macro Definition Documentation

9.50.1.1 `#define MAX_CDMA_ENC_MO_TXT_MSG_SIZE 255`

9.50.1.2 `#define MAX_MS_TRANSFER_ROUTE_MSG 256`

9.50.1.3 `#define MAX_MSC_ADDRESS_SIZE 256`

9.50.1.4 `#define MAX_MSE_TWS_MSG 1254`

9.50.1.5 `#define MAX_SMS_LIST_SIZE 255`

9.50.1.6 `#define MAX_SMS_MESSAGE_SIZE 2048`

9.50.2 Typedef Documentation

9.50.2.1 typedef struct `sMSCAddress` `sMSCAddressInfo`

Parameters

<i>length</i>	<ul style="list-style-type: none"> • Number of sets of following element
<i>data</i>	<ul style="list-style-type: none"> • SMSC address

9.50.2.2 typedef struct `sMSEtwsMessage` `sMSEtwsMessageInfo`

Parameters

<i>notificationType</i>	<ul style="list-style-type: none"> • Message mode 0x00 - Primary 0x01 - Secondary GSM 0x02 - Secondary UMTS
-------------------------	--

<i>length</i>	<ul style="list-style-type: none"> Number of sets of following elements
<i>data</i>	<ul style="list-style-type: none"> Raw message data

9.50.2.3 typedef struct **sMSEtwsPlmn** **sMSEtwsPlmnInfo**

Parameters

<i>mobileCountry↔ Code</i>	<ul style="list-style-type: none"> 16 bit representation of MCC value range : 0 -999
<i>mobile↔ NetworkCode</i>	<ul style="list-style-type: none"> 16 bit representation of MNC value range : 0 -999

9.50.2.4 typedef struct **sMSMessageMode** **sMSMessageModeInfo**

Parameters

<i>messageMode</i>	Message Mode
--------------------	--------------

9.50.2.5 typedef struct **sMSMTMessage** **sMSMTMessageInfo**

Parameters

<i>storageType</i>	memory storage 0x00-UIIM 0x01-NV
<i>messageIndex</i>	MT Message index

9.50.2.6 typedef struct **sMSOnIMS** **sMSOnIMSInfo**

Parameters

<i>smsOnIMS</i>	SMS on IMS
-----------------	------------

9.50.2.7 typedef struct **sMSTransferRouteMTMessage** **sMSTransferRouteMTMessageInfo**

Parameters

<i>ackIndicator</i>	<ul style="list-style-type: none"> Parameter to indicate if ACK must be sent by the control point 0x00 - Send ACK 0x01 - Do not send ACK
---------------------	---

<i>transactionID</i>	<ul style="list-style-type: none"> Transaction ID of the message
<i>format</i>	<ul style="list-style-type: none"> Message format 0x00 - CDMA 0x02 - 0x05 - Reserved 0x06 - GW_PP 0x07 - GW_BC
<i>length</i>	<ul style="list-style-type: none"> Length of the raw message. This length should not exceed the maximum WMS payload length of 256 bytes
<i>data</i>	<ul style="list-style-type: none"> Raw message data

9.50.3 Enumeration Type Documentation

9.50.3.1 enum eqmiCbkJSetStatus

Enumerator

LIBPACK_QMI_CBK_PARAM_RESET
LIBPACK_QMI_CBK_PARAM_SET
LIBPACK_QMI_CBK_PARAM_NOCHANGE

9.50.4 Function Documentation

9.50.4.1 `int pack_sms_SendSMS (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_sms_SendSMS_t * reqParam)`

send sms list pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
	<i>reqParam</i>	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.50.4.2 `int pack_sms_SetNewSMSCallback (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_sms_SetNewSMSCallback_t reqParam)`

set new sms callback pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
	<i>reqParam</i>	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.3 `int pack_sms_SLQSDeleteSMS (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_sms_SLQSDeleteSMS_t * reqParam)`

delete sms pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
	<i>reqParam</i>	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.4 `int pack_sms_SLQSGetSMS (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_sms_SLQSGetSMS_t * reqParam)`

get sms pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
	<i>reqParam</i>	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.5 int pack_sms_SLQSGetSMSList (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen,
pack_sms_SLQSGetSMSList_t * reqParam)

get sms list pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
	<i>reqParam</i>	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.6 int pack_sms_SLQSMModifySMSStatus (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen,
pack_sms_SLQSMModifySMSStatus_t * reqParam)

modify sms status pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
	<i>reqParam</i>	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.7 `int unpack_sms_SendSMS (uint8_t * pResp, uint16_t respLen, unpack_sms_SendSMS_t * pOutput)`

send sms unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.8 `int unpack_sms_SetNewSMSCallback (uint8_t * pResp, uint16_t respLen, unpack_sms_SetNewSMSCallback_t * Output)`

set new sms callback unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.9 `int unpack_sms_SetNewSMSCallback_ind (uint8_t * pResp, uint16_t respLen, unpack_sms_SetNewSMSCallback_ind_t * pOutput)`

set new sms callback indication unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.10 `int unpack_sms_SLQSDeleteSMS (uint8_t * pResp, uint16_t respLen, unpack_sms_SLQSDeleteSMS_t * pOutput)`

delete sms unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.11 `int unpack_sms_SLQSGetSMS (uint8_t * pResp, uint16_t respLen, unpack_sms_SLQSGetSMS_t * pOutput)`

get sms unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.12 `int unpack_sms_SLQSGetSMSList (uint8_t * pResp, uint16_t respLen, unpack_sms_SLQSGetSMSList_t * pOutput)`

get sms list unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.13 int unpack_sms_SLQSMModifySMSStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSMModifySMSStatus_t * *pOutput*)

modify sms status unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.50.4.14 int unpack_sms_SLQSWmsMemoryFullCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSWmsMemoryFullCallBack_ind_t * *pOutput*)

sms full callback indication unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.51 SwiDataTypes.h File Reference

SWI data types.

Macros

- #define [SWI_API](#)
- #define [QMI_NO_LTE_FW_SUPPORT](#) 0
- #define [QMI_TLV_PLACEHOLDER](#) 0x8F
- #define [UNUSEDPARAM\(x\)](#) (void)x

Typedefs

- typedef unsigned long [ULONG](#)
- typedef unsigned long long [ULONGLONG](#)
- typedef signed char [INT8](#)
- typedef unsigned char [BYTE](#)
- typedef char [CHAR](#)
- typedef unsigned short [WORD](#)
- typedef unsigned short [USHORT](#)
- typedef const char * [LPCSTR](#)
- typedef int [BOOL](#)
- typedef signed short [SHORT](#)
- typedef signed int [INT32](#)
- typedef float [FLOAT](#)

9.51.1 Detailed Description

SWI data types.

9.51.2 Macro Definition Documentation

9.51.2.1 #define [QMI_NO_LTE_FW_SUPPORT](#) 0

9.51.2.2 #define [QMI_TLV_PLACEHOLDER](#) 0x8F

9.51.2.3 #define [SWI_API](#)

9.51.2.4 #define [UNUSEDPARAM\(x\)](#) (void)x

Macro used to avoid "unused variable" compiler warnings generated due to the inclusion of the "-Wextra" flag in our make files.

9.51.3 Typedef Documentation

9.51.3.1 typedef int **BOOL**

9.51.3.2 typedef unsigned char **BYTE**

9.51.3.3 typedef char **CHAR**

9.51.3.4 typedef float **FLOAT**

9.51.3.5 typedef signed int **INT32**

9.51.3.6 typedef signed char **INT8**

9.51.3.7 typedef const char* **LPCSTR**

9.51.3.8 typedef signed short **SHORT**

9.51.3.9 typedef unsigned long **ULONG**

9.51.3.10 typedef unsigned long long **ULONGLONG**

9.51.3.11 typedef unsigned short **USHORT**

9.51.3.12 typedef unsigned short **WORD**

9.52 swiloc.h File Reference

Data Structures

- struct [unpack_swiloc_SwiLocGetAutoStart_t](#)
- struct [pack_swiloc_SwiLocSetAutoStart_t](#)

Functions

- int [pack_swiloc_SwiLocGetAutoStart](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_swiloc_SwiLocGetAutoStart](#) (uint8_t *pResp, uint16_t respLen, [unpack_swiloc_SwiLocGetAutoStart_t](#) *pOutput)
- int [pack_swiloc_SwiLocSetAutoStart](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_swiloc_SwiLocSetAutoStart_t](#) *reqArg)
- int [unpack_swiloc_SwiLocSetAutoStart](#) (uint8_t *pResp, uint16_t respLen)

9.52.1 Function Documentation

9.52.1.1 int [pack_swiloc_SwiLocGetAutoStart](#) ([pack_qmi_t](#) * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get Auto Start pack

Parameters

in	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.52.1.2 int pack_swiloc_SwiLocSetAutoStart (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_swiloc_SwiLocSetAutoStart_t * *reqArg*)

Set Auto Start pack

Parameters

in	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.52.1.3 int unpack_swiloc_SwiLocGetAutoStart (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiloc_SwiLocGetAutoStart_t * *pOutput*)

Get Auto Start unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.52.1.4 int unpack_swiloc_SwiLocSetAutoStart (uint8_t * *pResp*, uint16_t *respLen*)

Set Auto Start unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.53 swioma.h File Reference

Data Structures

- struct [pack_swioma_SLQSOMADMStartSession_t](#)
- struct [unpack_swioma_SLQSOMADMStartSession_t](#)
- struct [pack_swioma_SLQSOMADMCancelSession_t](#)
- struct [unpack_swioma_SLQSOMADMGetSettings_t](#)
- struct [pack_swioma_SLQSOMADMSetSettings_t](#)
- struct [pack_swioma_SLQSOMADMSelectSelection_t](#)
- struct [pack_swioma_SLQSOMADMGetSessionInfo_t](#)
- struct [unpack_swioma_SLQSOMADMGetSessionInfo_t](#)
- struct [unpack_omaDmFotaTlv_t](#)
- struct [unpack_omaDmConfigTlv_t](#)
- struct [unpack_omaDmNotificationsTlv_t](#)
- struct [unpack_swioma_SLQSOMADMAAlertCallback_ind_t](#)

Macros

- #define [LIBPACK_MAX_SWIOMA_STR_LEN](#) 255

Functions

- int [pack_swisma_SLQSOMADMStartSession](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_swisma_SLQSOMADMStartSession_t](#) reqParam)
- int [unpack_swisma_SLQSOMADMStartSession](#) (uint8_t *pResp, uint16_t respLen, [unpack_swisma_SLQSOMADMStartSession_t](#) *pOutput)
- int [pack_swisma_SLQSOMADMCancelSession](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_swisma_SLQSOMADMCancelSession_t](#) reqParam)
- int [unpack_swisma_SLQSOMADMCancelSession](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_swisma_SLQSOMADMGetSettings](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_swisma_SLQSOMADMGetSettings](#) (uint8_t *pResp, uint16_t respLen, [unpack_swisma_SLQSOMADMGetSettings_t](#) *pOutput)
- int [pack_swisma_SLQSOMADMSetSettings](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_swisma_SLQSOMADMSetSettings_t](#) reqParam)
- int [unpack_swisma_SLQSOMADMSetSettings](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_swisma_SLQSOMADMSendSelection](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_swisma_SLQSOMADMSendSelection_t](#) reqParam)
- int [unpack_swisma_SLQSOMADMSendSelection](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_swisma_SLQSOMADMGetSessionInfo](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_swisma_SLQSOMADMGetSessionInfo_t](#) reqParam)
- int [unpack_swisma_SLQSOMADMGetSessionInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_swisma_SLQSOMADMGetSessionInfo_t](#) *pOutput)
- int [pack_swisma_SLQSOMADMAAlertCallback](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_swisma_SLQSOMADMAAlertCallback](#) (uint8_t *pResp, uint16_t respLen)
- int [unpack_swisma_SLQSOMADMAAlertCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_swisma_SLQSOMADMAAlertCallback_ind_t](#) *pOutput)

9.53.1 Macro Definition Documentation

9.53.1.1 `#define LIBPACK_MAX_SWIOMA_STR_LEN 255`

9.53.2 Function Documentation

9.53.2.1 int [pack_swisma_SLQSOMADMAAlertCallback](#) ([pack_qmi_t](#) * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack QMI command to enable the SWIOMADM network-initiated alert callback function. This maps to SetSLQSOMADMAAlertCallback

Parameters

<i>pCtx</i> [OUT]	<ul style="list-style-type: none"> • See pack_qmi_t for more information
<i>pReqBuf</i> [IN/OUT]	<ul style="list-style-type: none"> • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
<i>pLen</i> [IN/OUT]	<ul style="list-style-type: none"> • On input, size of pReqBuf • On output, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.53.2.2 `int pack_swioma_SLQSOMADMCancelSession (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_swioma_SLQSOMADMCancelSession_t reqParam)`

Function to pack cancel OMA-DM session command This maps to SLQSOMADMCancelSession

Parameters

<i>pCtx</i> [OUT]	<ul style="list-style-type: none"> See pack_qmi_t for more information
<i>pReqBuf</i> [IN/OUT]	<ul style="list-style-type: none"> Buffer for packed QMI command to be provided by the host application Minimum expected size is 2048 bytes
<i>pLen</i> [IN/OUT]	<ul style="list-style-type: none"> On input, size of pReqBuf On output, number of bytes actually packed
<i>reqParam</i> [IN]	<ul style="list-style-type: none"> See pack_swioma_SLQSOMADMCancelSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.53.2.3 `int pack_swima_SLQSOMADMGetSessionInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_swima_SLQSOMADMGetSessionInfo_t reqParam)`

Function to pack QMI command to return information related to the current (or previous if no session is active) OMA-DM session. This maps to SLQSOMADMGetSessionInfo

Parameters

<i>pCtx</i> [OUT]	<ul style="list-style-type: none"> See pack_qmi_t for more information
<i>pReqBuf</i> [IN/OUT]	<ul style="list-style-type: none"> Buffer for packed QMI command to be provided by the host application Minimum expected size is 2048 bytes
<i>pLen</i> [IN/OUT]	<ul style="list-style-type: none"> On input, size of pReqBuf On output, number of bytes actually packed
<i>reqParam</i> [IN]	<ul style="list-style-type: none"> See pack_swima_SLQSOMADMGetSessionInfo_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.53.2.4 `int pack_swima_SLQSOMADMGetSettings (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Function to pack command to retrieve the OMA-DM settings from the device. This maps to SLQSOMADMGetSettings2

Parameters

<i>pCtx</i> [OUT]	<ul style="list-style-type: none"> See pack_qmi_t for more information
<i>pReqBuf</i> [IN/OUT]	<ul style="list-style-type: none"> Buffer for packed QMI command to be provided by the host application Minimum expected size is 2048 bytes
<i>pLen</i> [IN/OUT]	<ul style="list-style-type: none"> On input, size of pReqBuf On output, number of bytes actually packed
	Generated by Doxygen

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.53.2.5 int pack_swioma_SLQSOMADMSendSelection (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_swioma_SLQSOMADMSendSelection_t reqParam)

Function to pack OMA-DM send selection command This maps to SLQSOMADMSendSelection2

Parameters

<i>pCtx</i> [OUT]	<ul style="list-style-type: none"> See pack_qmi_t for more information
<i>pReqBuf</i> [IN/OUT]	<ul style="list-style-type: none"> Buffer for packed QMI command to be provided by the host application Minimum expected size is 2048 bytes
<i>pLen</i> [IN/OUT]	<ul style="list-style-type: none"> On input, size of pReqBuf On output, number of bytes actually packed
<i>reqParam</i> [IN]	<ul style="list-style-type: none"> See pack_swioma_SLQSOMADMSendSelection_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.53.2.6 int pack_swioma_SLQSOMADMSetSettings (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_swioma_SLQSOMADMSetSettings_t reqParam)

Function to pack OMA-DM set settings command This maps to SLQSOMADMSetSettings3

Parameters

<i>pCtx</i> [OUT]	<ul style="list-style-type: none"> See pack_qmi_t for more information
<i>pReqBuf</i> [IN/OUT]	<ul style="list-style-type: none"> Buffer for packed QMI command to be provided by the host application Minimum expected size is 2048 bytes
<i>pLen</i> [IN/OUT]	<ul style="list-style-type: none"> On input, size of pReqBuf On output, number of bytes actually packed
<i>reqParam</i> [IN]	<ul style="list-style-type: none"> See pack_swima_SLQSOMADMSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.53.2.7 `int pack_swima_SLQSOMADMStartSession (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_swima_SLQSOMADMStartSession_t reqParam)`

Function to pack Start OMA-DM session command This maps to SLQSOMADMStartSession2

Parameters

<i>pCtx</i> [OUT]	<ul style="list-style-type: none"> See pack_qmi_t for more information
<i>pReqBuf</i> [IN/OUT]	<ul style="list-style-type: none"> Buffer for packed QMI command to be provided by the host application Minimum expected size is 2048 bytes
<i>pLen</i> [IN/OUT]	<ul style="list-style-type: none"> On input, size of pReqBuf On output, number of bytes actually packed
<i>reqParam</i> [IN]	<ul style="list-style-type: none"> See pack_swima_SLQSOMADMStartSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.53.2.8 int unpack_swioma_SLQSOMADMAAlertCallback (uint8_t * pResp, uint16_t respLen)

Function to unpack response of QMI command to enable the SWIOMADM network-initiated alert callback function. This maps to SetSLQSOMADMAAlertCallback

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none"> Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none"> Length of pResp from modem

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

- Please use eQMI_SWIOMA_EVENT_IND indication to identify this event from SWIOMA service read function

9.53.2.9 int unpack_swioma_SLQSOMADMAAlertCallback_ind (uint8_t * pResp, uint16_t respLen, unpack_swioma_SLQSOMADMAAlertCallback_ind_t * pOutput)

Function to unpack SWIOMADM alert indications This maps to SetSLQSOMADMAAlertCallback

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none"> Response from modem
<i>respLen</i> [IN] Generated by Doxygen	<ul style="list-style-type: none"> Length of pResp from modem
<i>pOutput</i> [OUT]	<ul style="list-style-type: none"> See unpack_swioma_SLQSOMADMAAlertCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.53.2.10 `int unpack_swima_SLQSOMADMCancelSession (uint8_t * pResp, uint16_t respLen)`

Function to pack cancel OMA-DM session command This maps to SLQSOMADMCancelSession

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none">• Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none">• Length of pResp from modem

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.53.2.11 `int unpack_swima_SLQSOMADMGetSessionInfo (uint8_t * pResp, uint16_t respLen,
unpack_swima_SLQSOMADMGetSessionInfo_t * pOutput)`

Function to unpack information related to the current (or previous if no session is active) OMA-DM session. This maps to SLQSOMADMGetSessionInfo

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none">• Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none">• Length of pResp from modem
<i>pOutput</i> [OUT]	<ul style="list-style-type: none">• See unpack_swima_SLQSOMADMGetSessionInfo_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.53.2.12 `int unpack_swioma_SLQSOMADMGetSettings (uint8_t * pResp, uint16_t respLen,
unpack_swioma_SLQSOMADMGetSettings_t * pOutput)`

Function to unpack OMA-DM get settings response from modem This maps to SLQSOMADMGetSettings2

Parameters

<i>pResp[IN]</i>	<ul style="list-style-type: none">• Response from modem
<i>respLen[IN]</i>	<ul style="list-style-type: none">• Length of pResp from modem
<i>pOutput[OUT]</i>	<ul style="list-style-type: none">• See unpack_swioma_SLQSOMADMGetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.53.2.13 `int unpack_swioma_SLQSOMADMSendSelection (uint8_t * pResp, uint16_t respLen)`

Function to unpack OMA-DM send selection command This maps to SLQSOMADMSendSelection2

Parameters

<i>pResp[IN]</i>	<ul style="list-style-type: none">• Response from modem
<i>respLen[IN]</i>	<ul style="list-style-type: none">• Length of pResp from modem

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.53.2.14 `int unpack_swima_SLQSOMADMSetSettings (uint8_t * pResp, uint16_t respLen)`

Function to unpack OMA-DM set settings command This maps to SLQSOMADMSetSettings3

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none">• Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none">• Length of pResp from modem

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.53.2.15 `int unpack_swima_SLQSOMADMStartSession (uint8_t * pResp, uint16_t respLen,
unpack_swima_SLQSOMADMStartSession_t * pOutput)`

Function to unpack Start OMA-DM session response from modem This maps to SLQSOMADMStartSession2

Parameters

<i>pResp</i> [IN]	<ul style="list-style-type: none">• Response from modem
<i>respLen</i> [IN]	<ul style="list-style-type: none">• Length of pResp from modem
<i>pOutput</i> [OUT]	<ul style="list-style-type: none">• See unpack_swima_SLQSOMADMStartSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.54 SWIWWANCMAPI.h File Reference

9.55 uim.h File Reference

Data Structures

- struct [uim_appStatus](#)
- struct [uim_slotInfo](#)
- struct [uim_cardStatus](#)
- struct [uim_hotSwapStatus](#)
- struct [unpack_uim_GetCardStatus_t](#)
- struct [uim_encryptedPIN1](#)
- struct [uim_remainingRetries](#)
- struct [uim_sessionInformation](#)
- struct [uim_verifyUIMPIN](#)
- struct [uim_unblockUIMPIN](#)
- struct [uim_cardResult](#)
- struct [uim_setPINProtection](#)
- struct [uim_changeUIMPIN](#)
- struct [uim_fileInfo](#)
- struct [uim_UIMSessionInformation](#)
- struct [uim_readTransparentInfo](#)
- struct [uim_readResult](#)
- struct [pack_uim_VerifyPin_t](#)
- struct [unpack_uim_VerifyPin_t](#)
- struct [pack_uim_UnblockPin_t](#)
- struct [unpack_uim_UnblockPin_t](#)
- struct [pack_uim_SetPinProtection_t](#)
- struct [unpack_uim_SetPinProtection_t](#)
- struct [pack_uim_ChangePin_t](#)
- struct [unpack_uim_ChangePin_t](#)
- struct [pack_uim_ReadTransparent_t](#)
- struct [unpack_uim_ReadTransparent_t](#)
- struct [pack_uim_SLQSUIEventRegister_t](#)
- struct [unpack_uim_SLQSUIEventRegister_t](#)
- struct [appStats](#)
- struct [slotInf](#)
- struct [unpack_uim_SLQSUISetStatusChangeCallBack_ind_t](#)
- struct [slot_t](#)
- struct [slots_t](#)
- struct [unpack_uim_SLQSUIGetSlotsStatus_t](#)
- struct [pack_uim_SLQSUISwitchSlot_t](#)
- struct [unpack_uim_SetUimSlotStatusChangeCallback_ind_t](#)
- struct [pack_uim_SLQSUIPowerUp_t](#)
- struct [pack_uim_SLQSUIPowerDown_t](#)

Macros

- [#define UIM_UINT8_MAX_STRING_SZ 255](#)
- [#define UIM_MAX_DESCRIPTION_LENGTH 255](#)
- [#define UIM_MAX_NO_OF_SLOTS 5](#)
- [#define UIM_MAX_NO_OF_APPLICATIONS 10](#)
- [#define MAX_NO_OF_SLOTS 5](#)
- [#define MAX_NO_OF_APPLICATIONS 10](#)
- [#define MAX_DESCRIPTION_LENGTH 255](#)
- [#define MAX_SLOTS_STATUS 255](#)
- [#define MAX_ICCID_LENGTH 255](#)

Functions

- int [pack_uim_GetCardStatus](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int [unpack_uim_GetCardStatus](#) (uint8_t *pResp, uint16_t respLen, [unpack_uim_GetCardStatus_t](#) *pOutput)
- int [pack_uim_VerifyPin](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_uim_VerifyPin_t](#) *reqArg)
- int [unpack_uim_VerifyPin](#) (uint8_t *pResp, uint16_t respLen, [unpack_uim_VerifyPin_t](#) *pOutput)
- int [pack_uim_UnblockPin](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_uim_UnblockPin_t](#) *reqArg)
- int [unpack_uim_UnblockPin](#) (uint8_t *pResp, uint16_t respLen, [unpack_uim_UnblockPin_t](#) *pOutput)
- int [pack_uim_SetPinProtection](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_uim_SetPinProtection_t](#) *reqArg)
- int [unpack_uim_SetPinProtection](#) (uint8_t *pResp, uint16_t respLen, [unpack_uim_SetPinProtection_t](#) *pOutput)
- int [pack_uim_ChangePin](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_uim_ChangePin_t](#) *reqArg)
- int [unpack_uim_ChangePin](#) (uint8_t *pResp, uint16_t respLen, [unpack_uim_ChangePin_t](#) *pOutput)
- int [pack_uim_ReadTransparent](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_uim_ReadTransparent_t](#) *reqArg)
- int [unpack_uim_ReadTransparent](#) (uint8_t *pResp, uint16_t respLen, [unpack_uim_ReadTransparent_t](#) *pOutput)
- int [pack_uim_SLQSUIMEventRegister](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_uim_SLQSUIMEventRegister_t](#) *reqArg)
- int [unpack_uim_SLQSUIMEventRegister](#) (uint8_t *pResp, uint16_t respLen, [unpack_uim_SLQSUIMEventRegister_t](#) *pOutput)
- int [unpack_uim_SLQSUIMSetStatusChangeCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_uim_SLQSUIMSetStatusChangeCallback_ind_t](#) *pOutput)
- int [pack_uim_SLQSUIMGetSlotsStatus](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_uim_SLQSUIMGetSlotsStatus](#) (uint8_t *pResp, uint16_t respLen, [unpack_uim_SLQSUIMGetSlotsStatus_t](#) *pOutput)
- int [pack_uim_SLQSUIMSwitchSlot](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_uim_SLQSUIMSwitchSlot_t](#) *reqArg)
- int [unpack_uim_SLQSUIMSwitchSlot](#) (uint8_t *pResp, uint16_t respLen)
- int [unpack_uim_SetUimSlotStatusChangeCallback_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_uim_SetUimSlotStatusChangeCallback_ind_t](#) *pOutput)
- int [pack_uim_SLQSUIMPowerUp](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_uim_SLQSUIMPowerUp_t](#) *reqArg)
- int [unpack_uim_SLQSUIMPowerUp](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_uim_SLQSUIMPowerDown](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_uim_SLQSUIMPowerDown_t](#) *reqArg)
- int [unpack_uim_SLQSUIMPowerDown](#) (uint8_t *pResp, uint16_t respLen)

9.55.1 Macro Definition Documentation

9.55.1.1 `#define MAX_DESCRIPTION_LENGTH 255`

9.55.1.2 `#define MAX_ICCID_LENGTH 255`

9.55.1.3 `#define MAX_NO_OF_APPLICATIONS 10`

9.55.1.4 `#define MAX_NO_OF_SLOTS 5`

9.55.1.5 `#define MAX_SLOTS_STATUS 255`

9.55.1.6 `#define UIM_MAX_DESCRIPTION_LENGTH 255`

9.55.1.7 `#define UIM_MAX_NO_OF_APPLICATIONS 10`

9.55.1.8 `#define UIM_MAX_NO_OF_SLOTS 5`

9.55.1.9 `#define UIM_UINT8_MAX_STRING_SZ 255`

9.55.2 Function Documentation

9.55.2.1 `int pack_uim_ChangePin (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_uim_ChangePin_t * reqArg)`

Change Pin pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

`eQCWWAN_ERR_NONE` on success, `eQCWWAN_XXX` error value otherwise

See also

See [qmerrno.h](#) for `eQCWWAN_XXX` error values

9.55.2.2 `int pack_uim_GetCardStatus (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)`

Get Card Status pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

`eQCWWAN_ERR_NONE` on success, `eQCWWAN_XXX` error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.3 int pack_uim_ReadTransparent (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_uim_ReadTransparent_t * *reqArg*)

SLQS ReadTransparent pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.4 int pack_uim_SetPinProtection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_uim_SetPinProtection_t * *reqArg*)

Set Pin Protection pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.5 int pack_uim_SLQSUIMEventRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_uim_SLQSUIMEventRegister_t * *reqArg*)

UIM Status Change callback enable pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.6 int pack_uim_SLQSUIGetSlotsStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.7 int pack_uim_SLQSUIPowerDown (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_uim_SLQSUIPowerDown_t * *reqArg*)

Powers down the card pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.8 `int pack_uim_SLQSUIMPowerUp (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_uim_SLQSUIMPowerUp_t * reqArg)`

Powers up the card pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.9 `int pack_uim_SLQSUIMSwitchSlot (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_uim_SLQSUIMSwitchSlot_t * reqArg)`

switch slot pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.10 `int pack_uim_UnblockPin (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_uim_UnblockPin_t * reqArg)`

Unblock Pin pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.11 `int pack_uim_VerifyPin (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_uim_VerifyPin_t * reqArg)`

Verify Pin Status pack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response
in	<i>reqArg</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.12 `int unpack_uim_ChangePin (uint8_t * pResp, uint16_t respLen, unpack_uim_ChangePin_t * pOutput)`

Change Pin unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.13 `int unpack_uim_GetCardStatus (uint8_t * pResp, uint16_t respLen, unpack_uim_GetCardStatus_t * pOutput)`

Get Card Status unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.14 `int unpack_uim_ReadTransparent (uint8_t * pResp, uint16_t respLen, unpack_uim_ReadTransparent_t * pOutput)`

SLQS ReadTransparent unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.15 `int unpack_uim_SetPinProtection (uint8_t * pResp, uint16_t respLen, unpack_uim_SetPinProtection_t * pOutput)`

Set Pin Protection unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.16 int unpack_uim_SetUimSlotStatusChangeCallback_ind (uint8_t * *pResp*, uint16_t *respLen*,
unpack_uim_SetUimSlotStatusChangeCallback_ind_t * *pOutput*)

UIM Slot Status Change indication unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

use pack_uim_SLQSUIEventRegister to subscribe

9.55.2.17 int unpack_uim_SLQSUIEventRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIEvent↔
Register_t * *pOutput*)

UIM Status Change callback enable unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.18 int unpack_uim_SLQSUIMGetSlotsStatus (uint8_t * *pResp*, uint16_t *respLen*,
unpack_uim_SLQSUIMGetSlotsStatus_t * *pOutput*)

get slot status unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.19 int unpack_uim_SLQSUIMPowerDown (uint8_t * *pResp*, uint16_t *respLen*)

Powers down the card unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.20 int unpack_uim_SLQSUIMPowerUp (uint8_t * *pResp*, uint16_t *respLen*)

Powers up the card unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.21 int unpack_uim_SLQSUIMSetStatusChangeCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*,
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t * *pOutput*)

UIM Status Change indication unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

use pack_uim_SLQSUIMEventRegister to subscribe

9.55.2.22 int unpack_uim_SLQSUIMSwitchSlot (uint8_t * *pResp*, uint16_t *respLen*)

switch slot unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.23 `int unpack_uim_UnblockPin (uint8_t * pResp, uint16_t respLen, unpack_uim_UnblockPin_t * pOutput)`

Unblock Pin unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.55.2.24 `int unpack_uim_VerifyPin (uint8_t * pResp, uint16_t respLen, unpack_uim_VerifyPin_t * pOutput)`

Verify Pin unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56 wds.h File Reference

Data Structures

- struct [LibPackQosClassID](#)
- struct [LibPackTFTIDParams](#)
- struct [LibPackGPRSRequestedQoS](#)
- struct [LibPackUMTSQoS](#)
- struct [LibPackUMTSReqQoSSigInd](#)
- struct [pack_wds_SLQSSStartDataSession_t](#)
- struct [unpack_wds_SLQSSStartDataSession_t](#)
- struct [unpack_wds_SLQSSetPacketSrvStatusCallback_t](#)

- struct [pack_wds_SLQSStopDataSession_t](#)
- struct [wds_ProfileIdentifier](#)
- struct [wds_GPRSQoS](#)
- struct [wds_PCSCFIPv4ServerAddressList](#)
- struct [wds_PCSCFFQDNAddress](#)
- struct [wds_PCSCFFQDNAddressList](#)
- struct [wds_Domain](#)
- struct [wds_DomainNameList](#)
- struct [wds_IPV6AddressInfo](#)
- struct [wds_IPV6GWAddressInfo](#)
- struct [unpack_wds_SLQSGetRuntimeSettings_t](#)
- struct [wds_currNetworkInfo](#)
- struct [unpack_wds_SLQSSetWdsEventCallback_ind_t](#)
- struct [pack_wds_SLQSSetWdsEventCallback_t](#)
- struct [pack_wds_SLQSGetRuntimeSettings_t](#)
- struct [wds_UMTSMInQoS](#)
- struct [LibPackprofile_3GPP](#)
- struct [LibPackprofile_3GPP2](#)
- union [wds_profileInfo](#)
- struct [pack_wds_SLQSCreateProfile_t](#)
- struct [PackCreateProfileOut](#)
- struct [unpack_wds_SLQSCreateProfile_t](#)
- struct [pack_wds_SLQSModifyProfile_t](#)
- struct [unpack_wds_SLQSModifyProfile_t](#)
- struct [pack_wds_SLQSGetProfileSettings_t](#)
- struct [LibpackProfile3GPP](#)
- struct [LibpackProfile3GPP2](#)
- union [unpackWdsProfileParam](#)
- struct [UnPackGetProfileSettingOut](#)
- struct [unpack_wds_SLQSGetProfileSettings_t](#)
- struct [unpack_wds_GetSessionState_t](#)
- struct [pack_wds_GetDefaultProfile_t](#)
- struct [unpack_wds_GetDefaultProfile_t](#)
- struct [unpack_wds_GetConnectionRate_t](#)
- struct [pack_wds_GetPacketStatus_t](#)
- struct [unpack_wds_GetPacketStatus_t](#)
- struct [unpack_wds_GetSessionDuration_t](#)
- struct [pack_wds_GetSessionDuration_t](#)
- struct [unpack_wds_GetDormancyState_t](#)
- struct [pack_wds_GetDormancyState_t](#)
- struct [pack_wds_SLQSDeleteProfile_t](#)
- struct [unpack_wds_SLQSDeleteProfile_t](#)
- struct [pack_wds_SetDefaultProfile_t](#)
- struct [unpack_wds_SLQSGet3GPPConfigItem_t](#)
- struct [pack_wds_SLQSSet3GPPConfigItem_t](#)
- struct [unpack_wds_GetMobileIP_t](#)
- struct [pack_wds_GetMobileIP_t](#)
- struct [pack_wds_GetMobileIPProfile_t](#)
- struct [unpack_wds_GetMobileIPProfile_t](#)
- struct [currNetworkInfo](#)
- struct [unpack_wds_SLQSGetCurrDataSystemStat_t](#)
- struct [pack_wds_SLQSGetCurrDataSystemStat_t](#)
- struct [unpack_wds_GetLastMobileIPError_t](#)
- struct [pack_wds_GetLastMobileIPError_t](#)
- struct [rmTrasferStaticsReq](#)

- struct [pack_wds_RMSetTransferStatistics_t](#)
- struct [unpack_wds_RMSetTransferStatistics_t](#)
- struct [pack_wds_SetMobileIPProfile_t](#)
- struct [unpack_wds_SetMobileIPProfile_t](#)
- struct [pack_wds_SLQSWdsSwiPDPRuntimeSettings_t](#)
- struct [ipv6AddressInfo](#)
- struct [unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t](#)
- struct [transferStatInd](#)
- struct [pack_wds_SLQSGetDUNCallInfo_t](#)
- struct [connectionStatus](#)
- struct [dunchannelRate](#)
- struct [unpack_wds_SLQSGetDUNCallInfo_t](#)
- struct [qmiWSDDataBearerTechnology](#)
- struct [unpack_wds_SLQSGetDataBearerTechnology_t](#)
- struct [pack_wds_SLQSGetDataBearerTechnology_t](#)
- struct [pack_wds_SLQSSetIPFamilyPreference_t](#)
- struct [unpack_wds_SLQSSetIPFamilyPreference_t](#)
- struct [pack_wds_SetDefaultProfileNum_t](#)
- struct [pack_wds_GetDefaultProfileNum_t](#)
- struct [unpack_wds_GetDefaultProfileNum_t](#)
- struct [wdsDhcpv4ProfileId](#)
- struct [wdsDhcpv4HwConfig](#)
- struct [wdsDhcpv4Option](#)
- struct [wdsDhcpv4OptionList](#)
- struct [pack_wds_SLQSSetDHCPv4ClientConfig_t](#)
- struct [unpack_wds_SLQSSetDHCPv4ClientConfig_t](#)
- struct [pack_wds_GetPacketStatistics_t](#)
- struct [unpack_wds_GetPacketStatistics_t](#)
- struct [unpack_wds_GetByteTotals_t](#)
- struct [unpack_wds_SLQSGetCurrentChannelRate_t](#)
- struct [unpack_wds_SLQSSetLoopback_t](#)
- struct [pack_wds_SLQSSetLoopback_t](#)
- struct [wds_DataULongTlv](#)
- struct [wds_DataULongLongTlv](#)
- struct [unpack_RMTransferStatistics_ind_t](#)
- struct [pack_wds_DHCPv4ClientLeaseChange_t](#)
- struct [wds_DHCPProfileIdTlv](#)
- struct [wds_DHCPLeaseStateTlv](#)
- struct [wds_IPv4AdTlv](#)
- struct [wds_DHCPOpt](#)
- struct [wds_DHCPLeaseOptTlv](#)
- struct [unpack_wds_DHCPv4ClientLease_ind_t](#)
- struct [pack_wds_SetMobileIP_t](#)
- struct [pack_wds_SetMobileIPParameters_t](#)
- struct [pack_wds_SetAutoconnect_t](#)
- struct [unpack_wds_GetAutoconnect_t](#)
- struct [wds_TrStatInd](#)
- struct [pack_wds_SLQSWdsSetEventReport_t](#)
- struct [wds_DHCPv4ProfileId](#)
- struct [wds_DHCPv4HWConfig](#)
- struct [wds_DHCPv4Option](#)
- struct [wds_DHCPv4OptionList](#)
- struct [pack_wds_SLQSSetDHCPv4ClientConfig_t](#)
- struct [unpack_wds_GetDataBearerTechnology_t](#)

Macros

- `#define IPV6_ADDRESS_ARRAY_SIZE 8`
- `#define MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE 24`
- `#define PACK_WDS_IPV4 4`
- `#define PACK_WDS_IPV6 6`
- `#define BYT_STAT_STAT_MASK 0X000000C0`
- `#define WDS_DHCP_MAX_NUM_OPTIONS 30`
- `#define WDS_DHCP_OPTION_DATA_BUF_SIZE 2048 /* current max size of raw message in SDK process is 2048 */`

Typedefs

- typedef union `unpackWdsProfileParam UnpackQmiProfileInfo`

Functions

- int `pack_wds_SLQSSStartDataSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSStartDataSession_t *reqArg)`
- int `unpack_wds_SLQSSStartDataSession (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSStartDataSession_t *pOutput)`
- int `unpack_wds_SLQSSetPacketSrvStatusCallback (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetPacketSrvStatusCallback_t *pOutput)`
- int `pack_wds_SLQSSStopDataSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSStopDataSession_t *reqArg)`
- int `unpack_wds_SLQSSStopDataSession (uint8_t *pResp, uint16_t respLen)`
- int `unpack_wds_SLQSGetRuntimeSettings (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetRuntimeSettings_t *pOutput)`
- int `unpack_wds_SLQSSetWdsEventCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetWdsEventCallback_ind_t *pOutput)`
- int `unpack_wds_SLQSSetWdsEventCallback (uint8_t *pResp, uint16_t respLen)`
- int `pack_wds_SLQSSetWdsEventCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSetWdsEventCallback_t *reqArg)`
- int `pack_wds_SLQSGetRuntimeSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetRuntimeSettings_t *reqArg)`
- int `pack_wds_SLQSCreateProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSCreateProfile_t *reqArg)`
- int `unpack_wds_SLQSCreateProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSCreateProfile_t *pOutput)`
- int `pack_wds_SLQSModifyProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSModifyProfile_t *reqArg)`
- int `unpack_wds_SLQSModifyProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSModifyProfile_t *pOutput)`
- int `pack_wds_SLQSGetProfileSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetProfileSettings_t *reqArg)`
- int `unpack_wds_SLQSGetProfileSettings (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetProfileSettings_t *pOutput)`
- int `pack_wds_GetSessionState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)`
- int `unpack_wds_GetSessionState (uint8_t *pResp, uint16_t respLen, unpack_wds_GetSessionState_t *pOutput)`
- int `pack_wds_GetDefaultProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetDefaultProfile_t *reqParam)`
- int `unpack_wds_GetDefaultProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_GetDefaultProfile_t *pOutput)`

- int [pack_wds_GetConnectionRate](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_wds_GetConnectionRate](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetConnectionRate_t](#) *pOutput)
- int [pack_wds_GetPacketStatus](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_GetPacketStatus_t](#) *reqParam)
- int [unpack_wds_GetPacketStatus](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetPacketStatus_t](#) *pOutput)
- int [pack_wds_GetSessionDuration](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_GetSessionDuration_t](#) *reqParam)
- int [unpack_wds_GetSessionDuration](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetSessionDuration_t](#) *pOutput)
- int [pack_wds_GetDormancyState](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_GetDormancyState_t](#) *reqParam)
- int [unpack_wds_GetDormancyState](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetDormancyState_t](#) *pOutput)
- int [pack_wds_SLQSDeleteProfile](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SLQSDeleteProfile_t](#) *reqParam)
- int [unpack_wds_SLQSDeleteProfile](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_SLQSDeleteProfile_t](#) *pOutput)
- int [pack_wds_SetDefaultProfile](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SetDefaultProfile_t](#) *reqParam)
- int [unpack_wds_SetDefaultProfile](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_wds_SLQSGet3GPPConfigItem](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_wds_SLQSGet3GPPConfigItem](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_SLQSGet3GPPConfigItem_t](#) *pOutput)
- int [pack_wds_SLQSSet3GPPConfigItem](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SLQSSet3GPPConfigItem_t](#) *reqParam)
- int [unpack_wds_SLQSSet3GPPConfigItem](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_wds_GetMobileIP](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_GetMobileIP_t](#) *pReqParam)
- int [unpack_wds_GetMobileIP](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetMobileIP_t](#) *pOutput)
- int [pack_wds_GetMobileIPProfile](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_GetMobileIPProfile_t](#) *reqParam)
- int [unpack_wds_GetMobileIPProfile](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetMobileIPProfile_t](#) *pOutput)
- int [pack_wds_SLQSGetCurrDataSystemStat](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SLQSGetCurrDataSystemStat_t](#) *pReqParam)
- int [unpack_wds_SLQSGetCurrDataSystemStat](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_SLQSGetCurrDataSystemStat_t](#) *pOutput)
- int [pack_wds_GetLastMobileIPError](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_GetLastMobileIPError_t](#) *pReqParam)
- int [unpack_wds_GetLastMobileIPError](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetLastMobileIPError_t](#) *pOutput)
- int [pack_wds_RMSetTransferStatistics](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_RMSetTransferStatistics_t](#) *reqParam)
- int [unpack_wds_RMSetTransferStatistics](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_RMSetTransferStatistics_t](#) *pOutput)
- int [pack_wds_SetMobileIPProfile](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SetMobileIPProfile_t](#) *reqParam)
- int [unpack_wds_SetMobileIPProfile](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_SetMobileIPProfile_t](#) *pOutput)
- int [pack_wds_SLQSWdsSwiPDPRuntimeSettings](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SLQSWdsSwiPDPRuntimeSettings_t](#) *reqParam)
- int [unpack_wds_SLQSWdsSwiPDPRuntimeSettings](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t](#) *pOutput)

- int [pack_wds_SLQSGetDUNCallInfo](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SLQSGetDUNCallInfo_t](#) *reqParam)
- int [unpack_wds_SLQSGetDUNCallInfo](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_SLQSGetDUNCallInfo_t](#) *pOutput)
- int [pack_wds_SLQSGetDataBearerTechnology](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SLQSGetDataBearerTechnology_t](#) *pReqParam)
- int [unpack_wds_SLQSGetDataBearerTechnology](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_SLQSGetDataBearerTechnology_t](#) *pOutput)
- int [pack_wds_SLQSSetIPFamilyPreference](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SLQSSetIPFamilyPreference_t](#) *pReqParam)
- int [unpack_wds_SLQSSetIPFamilyPreference](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_SLQSSetIPFamilyPreference_t](#) *pOutput)
- int [pack_wds_SetDefaultProfileNum](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SetDefaultProfileNum_t](#) *pReqParam)
- int [unpack_wds_SetDefaultProfileNum](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_wds_GetDefaultProfileNum](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_GetDefaultProfileNum_t](#) *pReqParam)
- int [unpack_wds_GetDefaultProfileNum](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetDefaultProfileNum_t](#) *pOutput)
- int [pack_wds_SLQSSetDHCPv4ClientConfig](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SLQSSetDHCPv4ClientConfig_t](#) *pReq)
- int [unpack_wds_SLQSSetDHCPv4ClientConfig](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_SLQSSetDHCPv4ClientConfig_t](#) *pOutput)
- int [pack_wds_GetPacketStatistics](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_GetPacketStatistics_t](#) *pReq)
- int [unpack_wds_GetPacketStatistics](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetPacketStatistics_t](#) *pOutput)
- int [pack_wds_GetByteTotals](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_wds_GetByteTotals](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetByteTotals_t](#) *pOutput)
- int [pack_wds_SLQSGetCurrentChannelRate](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_wds_SLQSGetCurrentChannelRate](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_SLQSGetCurrentChannelRate_t](#) *pOutput)
- int [pack_wds_SLQSSetLoopback](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_wds_SLQSSetLoopback](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_SLQSSetLoopback_t](#) *pOutput)
- int [pack_wds_SLQSSetLoopback](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SLQSSetLoopback_t](#) *reqArg)
- int [unpack_wds_SLQSSetLoopback](#) (uint8_t *pResp, uint16_t respLen)
- int [unpack_wds_RMTransferStatistics_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_RMTransferStatistics_ind_t](#) *pOutput)
- int [pack_wds_DHCPv4ClientLeaseChange](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_DHCPv4ClientLeaseChange_t](#) *reqArg)
- int [unpack_wds_DHCPv4ClientLeaseChange](#) (uint8_t *pResp, uint16_t respLen)
- int [unpack_wds_DHCPv4ClientLease_ind](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_DHCPv4ClientLease_ind_t](#) *pOutput)
- int [pack_wds_SetMobileIP](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SetMobileIP_t](#) *reqArg)
- int [unpack_wds_SetMobileIP](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_wds_SetMobileIPParameters](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SetMobileIPParameters_t](#) *reqArg)
- int [unpack_wds_SetMobileIPParameters](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_wds_SetAutoconnect](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_SetAutoconnect_t](#) *reqArg)
- int [unpack_wds_SetAutoconnect](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_wds_GetAutoconnect](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)

- int [unpack_wds_GetAutoconnect](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetAutoconnect_t](#) *p↔
Output)
- int [pack_wds_SLQSWdsSetEventReport](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_wds_↔
_SLQSWdsSetEventReport_t](#) *reqArg)
- int [unpack_wds_SLQSWdsSetEventReport](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_wds_SLQSWdsGoDormant](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_wds_SLQSWdsGoDormant](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_wds_SLQSWdsGoActive](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_wds_SLQSWdsGoActive](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_wds_SLQSRResetPacketStatics](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_wds_SLQSRResetPacketStatics](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_wds_SLQSSSetDHCPv4ClientConfig](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen, [pack_↔
_wds_SLQSSSetDHCPv4ClientConfig_t](#) *reqArg)
- int [unpack_wds_SLQSSSetDHCPv4ClientConfig](#) (uint8_t *pResp, uint16_t respLen)
- int [pack_wds_GetDataBearerTechnology](#) ([pack_qmi_t](#) *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int [unpack_wds_GetDataBearerTechnology](#) (uint8_t *pResp, uint16_t respLen, [unpack_wds_GetData_↔
BearerTechnology_t](#) *pOutput)

9.56.1 Macro Definition Documentation

9.56.1.1 `#define BYT_STAT_STAT_MASK 0X000000C0`

9.56.1.2 `#define IPV6_ADDRESS_ARRAY_SIZE 8`

9.56.1.3 `#define MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE 24`

9.56.1.4 `#define PACK_WDS_IPV4 4`

9.56.1.5 `#define PACK_WDS_IPV6 6`

9.56.1.6 `#define WDS_DHCP_MAX_NUM_OPTIONS 30`

9.56.1.7 `#define WDS_DHCP_OPTION_DATA_BUF_SIZE 2048 /* current max size of raw message in SDK process is 2048 */`

9.56.2 Typedef Documentation

9.56.2.1 `typedef union unpackWdsProfileParam UnpackQmiProfileInfo`

9.56.3 Function Documentation

9.56.3.1 `int pack_wds_DHCPv4ClientLeaseChange (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen,
pack_wds_DHCPv4ClientLeaseChange_t * reqArg)`

DHCPv4 lease state changes pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.2 `int pack_wds_GetAutoconnect (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Gets auto connect data session setting pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.3 `int pack_wds_GetByteTotals (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

get Rx/Tx byte counts since the start of the last packet data session pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.4 `int pack_wds_GetConnectionRate (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

get connection rate pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: No

9.56.3.5 `int pack_wds_GetDataBearerTechnology (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

get current data bearer technology pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.6 `int pack_wds_GetDefaultProfile (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_GetDefaultProfile_t * reqParam)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.7 `int pack_wds_GetDefaultProfileNum (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_GetDefaultProfileNum_t * pReqParam)`

get default profile number pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.8 `int pack_wds_GetDormancyState (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_GetDormancyState_t * reqParam)`

get dormancy state pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: No

9.56.3.9 int pack_wds_GetLastMobileIPError (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_GetLastMobileIPError_t * *pReqParam*)

get current data system pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: No

9.56.3.10 int pack_wds_GetMobileIP (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_GetMobileIP_t * *pReqParam*)

get mobile ip mode pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: No

9.56.3.11 int pack_wds_GetMobileIPProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_GetMobileIPProfile_t * *reqParam*)

get mobile ip profile pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.12 int pack_wds_GetPacketStatistics (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_GetPacketStatistics_t * *pReq*)

gets current packet transfer counter values pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>pReq</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.13 int pack_wds_GetPacketStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_GetPacketStatus_t * *reqParam*)

get packet status pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.14 int pack_wds_GetSessionDuration (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_GetSessionDuration_t * *reqParam*)

get session duration pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: No

9.56.3.15 int pack_wds_GetSessionState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get session state pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.16 int pack_wds_RMSetTransferStatistics (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_RMSetTransferStatistics_t * *reqParam*)

rm set transfer statistics pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>reqParam</i>	request parameter

Note

PDN Specific: No

```
9.56.3.17 int pack_wds_SetAutoconnect ( pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen,
    pack_wds_SetAutoconnect_t * reqArg )
```

Auto connect data session parameters pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

```
9.56.3.18 int pack_wds_SetDefaultProfile ( pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen,
    pack_wds_SetDefaultProfile_t * reqParam )
```

set default profile pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.19 `int pack_wds_SetDefaultProfileNum (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SetDefaultProfileNum_t * pReqParam)`

set default profile number pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.20 `int pack_wds_SetMobileIP (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SetMobileIP_t * reqArg)`

Sets the current mobile IP setting pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.21 `int pack_wds_SetMobileIPParameters (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SetMobileIPParameters_t * reqArg)`

Sets the specified mobile IP parameters pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.22 `int pack_wds_SetMobileIPProfile (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SetMobileIPProfile_t * reqParam)`

set mobile ip profile pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.23 `int pack_wds_SLQSCreateProfile (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSCreateProfile_t * reqArg)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.24 int pack_wds_SLQSDeleteProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_SLQSDeleteProfile_t * *reqParam*)

delete stored profile pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.25 int pack_wds_SLQSGet3GPPConfigItem (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get 3Gpp config items pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.26 `int pack_wds_SLQSGetCurrDataSystemStat (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSGetCurrDataSystemStat_t * pReqParam)`

get current data system pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.27 `int pack_wds_SLQSGetCurrentChannelRate (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

get current Tx/Rx channel bitrate of the current packet data pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.28 `int pack_wds_SLQSGetDataBearerTechnology (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSGetDataBearerTechnology_t * pReqParam)`

get data bearer technology pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.29 `int pack_wds_SLQSGetDUNCallInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSGetDUNCallInfo_t * reqParam)`

get dun call info pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.30 `int pack_wds_SLQSGetProfileSettings (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSGetProfileSettings_t * reqArg)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.31 `int pack_wds_SLQSGetRuntimeSettings (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSGetRuntimeSettings_t * reqArg)`

get runtime settings pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.32 `int pack_wds_SLQSModifyProfile (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSModifyProfile_t * reqArg)`

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.33 int pack_wds_SLQSResetPacketStatics (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Reset packet data transfer statistics pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.34 int pack_wds_SLQSSet3GPPConfigItem (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_SLQSSet3GPPConfigItem_t * *reqParam*)

set 3Gpp config items pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.35 int pack_wds_SLQSSetIPFamilyPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_SLQSSetIPFamilyPreference_t * *pReqParam*)

Set IP Family Preference pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
out	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.36 int pack_wds_SLQSSetWdsEventCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_SLQSSetWdsEventCallback_t * *reqArg*)

set event callback pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: No

9.56.3.37 int pack_wds_SLQSSGetDHCPv4ClientConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*,
pack_wds_SLQSSGetDHCPv4ClientConfig_t * *pReq*)

get DHCPv4 Client Config pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>pReq</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.38 int pack_wds_SLQSSGetLoopback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get the value of loopback mode and multiplier pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReq</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

```
9.56.3.39 int pack_wds_SLQSSSetDHCPv4ClientConfig ( pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen,
pack_wds_SLQSSSetDHCPv4ClientConfig_t * reqArg )
```

Gets the DHCP Client V4 Configuration pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

see [qmerrno.h](#) for eQCWWAN_xxx error values

```
9.56.3.40 int pack_wds_SLQSSSetLoopback ( pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen,
pack_wds_SLQSSSetLoopback_t * reqArg )
```

Enable/disable Data Loopback Mode and set the value of loopback multiplier pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.41 `int pack_wds_SLQSStartDataSession (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSStartDataSession_t * reqArg)`

Start data session

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.42 `int pack_wds_SLQSStopDataSession (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSStopDataSession_t * reqArg)`

stop data session pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length
in	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.43 `int pack_wds_SLQSWdsGoActive (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Gets the device into Active state pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.44 `int pack_wds_SLQSWdsGoDormant (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)`

Gets the device into dormant state pack

Parameters

in, out	<i>pCtx</i>	qmi request context
out	<i>pReqBuf</i>	qmi request buffer
out	<i>pLen</i>	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.45 `int pack_wds_SLQSWdsSetEventReport (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSWdsSetEventReport_t * reqArg)`

Sets the event report parameters pack

Parameters

<i>in, out</i>	<i>pCtx</i>	qmi request context
<i>out</i>	<i>pReqBuf</i>	qmi request buffer
<i>out</i>	<i>pLen</i>	qmi request length
<i>in</i>	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.46 `int pack_wds_SLQSWdsSwiPDPRuntimeSettings (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSWdsSwiPDPRuntimeSettings_t * reqParam)`

swi pdp runtime settings pack

Parameters

<i>in, out</i>	<i>pCtx</i>	qmi request context
<i>out</i>	<i>pReq</i>	qmi request buffer
<i>out</i>	<i>pLen</i>	qmi request length
<i>out</i>	<i>reqParam</i>	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.56.3.47 `int unpack_wds_DHCPv4ClientLease_ind (uint8_t * pResp, uint16_t respLen, unpack_wds_DHCPv4ClientLease_ind_t * pOutput)`

DHCP lease state has changed indication unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.48 int unpack_wds_DHCPv4ClientLeaseChange (uint8_t * *pResp*, uint16_t *respLen*)

DHCPv4 lease state changes unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.49 int unpack_wds_GetAutoconnect (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetAutoconnect_t * *pOutput*)

Gets auto connect data session setting unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.50 `int unpack_wds_GetByteTotals (uint8_t * pResp, uint16_t respLen, unpack_wds_GetByteTotals_t * pOutput)`

get Rx/Tx byte counts since the start of the last packet data session unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.51 `int unpack_wds_GetConnectionRate (uint8_t * pResp, uint16_t respLen, unpack_wds_GetConnectionRate_t * pOutput)`

get connection rate unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.52 `int unpack_wds_GetDataBearerTechnology (uint8_t * pResp, uint16_t respLen, unpack_wds_GetDataBearerTechnology_t * pOutput)`

get current data bearer technology unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.53 `int unpack_wds_GetDefaultProfile (uint8_t * pResp, uint16_t respLen, unpack_wds_GetDefaultProfile_t * pOutput)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.54 `int unpack_wds_GetDefaultProfileNum (uint8_t * pResp, uint16_t respLen, unpack_wds_GetDefaultProfile↔
Num_t * pOutput)`

get default profile number unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.55 `int unpack_wds_GetDormancyState (uint8_t * pResp, uint16_t respLen, unpack_wds_GetDormancyState_t * pOutput)`

get dormancy state unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.56 int unpack_wds_GetLastMobileIPError (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetLastMobileIP↵
Error_t * *pOutput*)

get current data system unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.57 int unpack_wds_GetMobileIP (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetMobileIP_t * *pOutput*)

get mobile ip mode unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.56.3.58 int unpack_wds_GetMobileIPProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetMobileIPProfile_t * *pOutput*)

get mobile ip profile unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.56.3.59 int unpack_wds_GetPacketStatistics (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetPacketStatistics_t * *pOutput*)

gets current packet transfer counter values unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_XXX error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_XXX error values

9.56.3.60 int unpack_wds_GetPacketStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetPacketStatus_t * *pOutput*)

get packet status unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.61 int unpack_wds_GetSessionDuration (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetSessionDuration_t * *pOutput*)

get session duration unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.62 int unpack_wds_GetSessionState (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetSessionState_t * *pOutput*)

get session state unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.63 `int unpack_wds_RMSetTransferStatistics (uint8_t * pResp, uint16_t respLen,
unpack_wds_RMSetTransferStatistics_t * pOutput)`

rm set transfer statistics unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.64 `int unpack_wds_RMTransferStatistics_ind (uint8_t * pResp, uint16_t respLen,
unpack_RMTransferStatistics_ind_t * pOutput)`

RM transfer statistics indication unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.65 `int unpack_wds_SetAutoconnect (uint8_t * pResp, uint16_t respLen)`

Auto connect data session parameters unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.66 int unpack_wds_SetDefaultProfile (uint8_t * *pResp*, uint16_t *respLen*)

set default profile unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.67 int unpack_wds_SetDefaultProfileNum (uint8_t * *pResp*, uint16_t *respLen*)

set default profile number unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.68 `int unpack_wds_SetMobileIP (uint8_t * pResp, uint16_t respLen)`

Sets the current mobile IP setting unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.69 `int unpack_wds_SetMobileIPParameters (uint8_t * pResp, uint16_t respLen)`

Sets the specified mobile IP parameters unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.70 `int unpack_wds_SetMobileIPProfile (uint8_t * pResp, uint16_t respLen, unpack_wds_SetMobileIPProfile_t * pOutput)`

set mobile ip profile unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.71 `int unpack_wds_SLQSCreateProfile (uint8_t * pResp, uint16_t respLen, unpack_wds_SLQSCreateProfile_t * pOutput)`

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response structure to fill
in	<i>pProfileId</i>	profile id pointer passed in req

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.72 `int unpack_wds_SLQSDeleteProfile (uint8_t * pResp, uint16_t respLen, unpack_wds_SLQSDeleteProfile_t * pOutput)`

delete stored profile unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.73 `int unpack_wds_SLQSGet3GPPConfigItem (uint8_t * pResp, uint16_t respLen, unpack_wds_SLQSGet3GPPConfigItem_t * pOutput)`

get 3GPP config items unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.74 int unpack_wds_SLQSGetCurrDataSystemStat (uint8_t * *pResp*, uint16_t *respLen*,
unpack_wds_SLQSGetCurrDataSystemStat_t * *pOutput*)

get current data system unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.75 int unpack_wds_SLQSGetCurrentChannelRate (uint8_t * *pResp*, uint16_t *respLen*,
unpack_wds_SLQSGetCurrentChannelRate_t * *pOutput*)

get current Tx/Rx channel bitrate of the current packet data unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.76 int unpack_wds_SLQSGetDataBearerTechnology (uint8_t * *pResp*, uint16_t *respLen*,
unpack_wds_SLQSGetDataBearerTechnology_t * *pOutput*)

get data bearer technology unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.77 int unpack_wds_SLQSGetDUNCallInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetDUNCall↵
Info_t * *pOutput*)

get dun call info unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.78 int unpack_wds_SLQSGetProfileSettings (uint8_t * *pResp*, uint16_t *respLen*,
unpack_wds_SLQSGetProfileSettings_t * *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
Generated by Doxygen	<i>pOutput</i>	response structure to fill

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.79 int unpack_wds_SLQSGetRuntimeSettings (uint8_t * *pResp*, uint16_t *respLen*,
unpack_wds_SLQSGetRuntimeSettings_t * *pOutput*)

get runtime settings unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.80 int unpack_wds_SLQSModifyProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSModifyProfile_t
* *pOutput*)

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.81 int unpack_wds_SLQSResetPacketStatics (uint8_t * *pResp*, uint16_t *respLen*)

Reset packet data transfer statistics unpack.

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.82 int unpack_wds_SLQSSet3GPPConfigItem (uint8_t * *pResp*, uint16_t *respLen*)

set 3GPP config items unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

9.56.3.83 int unpack_wds_SLQSSetIPFamilyPreference (uint8_t * *pResp*, uint16_t *respLen*,
unpack_wds_SLQSSetIPFamilyPreference_t * *pOutput*)

Set IP Family Preference unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.84 int unpack_wds_SLQSSetPacketSrvStatusCallback (uint8_t * *pResp*, uint16_t *respLen*,
unpack_wds_SLQSSetPacketSrvStatusCallback_t * *pOutput*)

set packet srv status callback unpack

Parameters

in	<i>pResp</i>	qmi response
in	<i>respLen</i>	length
out	<i>pOutput</i>	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.85 `int unpack_wds_SLQSSetWdsEventCallback (uint8_t * pResp, uint16_t respLen)`

set event callback unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.86 `int unpack_wds_SLQSSetWdsEventCallback_ind (uint8_t * pResp, uint16_t respLen,
unpack_wds_SLQSSetWdsEventCallback_ind_t * pOutput)`

set event callback unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.87 `int unpack_wds_SLQSSGetDHCPv4ClientConfig (uint8_t * pResp, uint16_t respLen,
unpack_wds_SLQSSGetDHCPv4ClientConfig_t * pOutput)`

get DHCPv4 Client Config unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.88 `int unpack_wds_SLQSSGetLoopback (uint8_t * pResp, uint16_t respLen, unpack_wds_SLQSSGet↵
Loopback_t * pOutput)`

get the value of loopback mode and multiplier unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.89 `int unpack_wds_SLQSSSetDHCPv4ClientConfig (uint8_t * pResp, uint16_t respLen)`

Gets the DHCP Client V4 Configuration unpack.

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.90 int unpack_wds_SLQSSetLoopback (uint8_t * *pResp*, uint16_t *respLen*)

Enable/disable Data Loopback Mode and set the value of loopback multiplier unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.91 int unpack_wds_SLQSStartDataSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSStartData↵
Session_t * *pOutput*)

start data session unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.92 int unpack_wds_SLQSStopDataSession (uint8_t * *pResp*, uint16_t *respLen*)

stop data session unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.93 int unpack_wds_SLQSWdsGoActive (uint8_t * *pResp*, uint16_t *respLen*)

Gets the device into Active state unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.94 int unpack_wds_SLQSWdsGoDormant (uint8_t * *pResp*, uint16_t *respLen*)

Gets the device into dormant state unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.95 int unpack_wds_SLQSWdsSetEventReport (uint8_t * *pResp*, uint16_t *respLen*)

Sets the event report parameters unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

9.56.3.96 int unpack_wds_SLQSWdsSwiPDPRuntimeSettings (uint8_t * *pResp*, uint16_t *respLen*,
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t * *pOutput*)

get current data system unpack

Parameters

in	<i>pResp</i>	qmi response from modem
in	<i>respLen</i>	qmi response length
out	<i>pOutput</i>	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See also

See [qmerrno.h](#) for eQCWWAN_xxx error values

Index

- [_3gppRelease](#)
 - [unpack_wds_SLQSGet3GPPConfigItem_t, 1073](#)
- [_GetProfileSettingIn, 58](#)
 - [ProfileID, 58](#)
 - [ProfileType, 58](#)
- [_GetProfileSettingOut, 58](#)
 - [curProfile, 59](#)
 - [pExtErrCode, 59](#)
- [_MitigationDevInfo, 61](#)
 - [deviceId, 62](#)
 - [deviceIdLen, 62](#)
- [_NAMS_RADIO_IF_TECHNOLOGY_](#)
 - [qaGobiApiNas.h, 1511](#)
- [_SLQSOMADMSessionInfo, 71](#)
 - [pDate, 73](#)
 - [pDateLength, 73](#)
 - [pPkgDescLength, 73](#)
 - [pPkgDescription, 73](#)
 - [pPkgName, 73](#)
 - [pPkgNameLength, 73](#)
 - [pRetryCount, 74](#)
 - [pSessionState, 74](#)
 - [pSessionType, 74](#)
 - [pSeverity, 74](#)
 - [pSource, 74](#)
 - [pSourceLength, 74](#)
 - [pStatus, 74](#)
 - [pTime, 74](#)
 - [pTimeLength, 74](#)
 - [pUpdateCompleteStatus, 74](#)
- [_SLQSOMADMSettings, 74](#)
 - [pAutosdm, 76](#)
 - [pFOTAUpdate, 76](#)
 - [pFOTAdownload, 76](#)
 - [pFwAutoCheck, 76](#)
 - [pOMADMEEnabled, 76](#)
- [_SLQSOMADMSettingsReqParams, 76](#)
 - [FOTAUpdate, 77](#)
 - [FOTAdownload, 77](#)
 - [pAutosdm, 77](#)
- [_SLQSOMADMSettingsReqParams3, 77](#)
 - [FOTAUpdate, 78](#)
 - [FOTAdownload, 78](#)
 - [pAutosdm, 78](#)
 - [pFwAutoCheck, 78](#)
- [_SLQSSwiGetHostDevInfoParams, 78](#)
 - [bManSize, 79](#)
 - [bModelSize, 79](#)
 - [bPlasmaIDSize, 79](#)
 - [bSWVerSize, 79](#)
 - [pManString, 79](#)
 - [pModelString, 79](#)
 - [pPlasmaIDString, 79](#)
 - [pSWVerString, 79](#)
- [_SLQSSwiGetOSInfoParams, 79](#)
 - [bNameSize, 80](#)
 - [bVersionSize, 80](#)
 - [pNameString, 80](#)
 - [pVersionString, 80](#)
- [_SLQSSwiGetSerialNoExtParams, 80](#)
 - [meidLength, 81](#)
 - [pMeidString, 81](#)
- [_SLQSSwiSetHostDevInfoParams, 81](#)
 - [bManSize, 82](#)
 - [bModelSize, 82](#)
 - [bPlasmaIDSize, 82](#)
 - [bSWVerSize, 82](#)
 - [pManString, 82](#)
 - [pModelString, 82](#)
 - [pPlasmaIDString, 82](#)
 - [pSWVerString, 82](#)
- [_SLQSSwiSetOSInfoParams, 82](#)
 - [bNameSize, 83](#)
 - [bVersionSize, 83](#)
 - [pNameString, 83](#)
 - [pVersionString, 83](#)
- [_SlqsNas3GppNetworkRAT_, 69](#)
 - [MCC, 70](#)
 - [MNC, 70](#)
 - [RAT, 70](#)
- [__GOBI_API_CODING_SCHEME_H__](#)
 - [qaGobiApiTableCodingScheme.h, 1642](#)
- [_getIndicationRegResp, 57](#)
 - [pRegCallStatInfoEvt, 58](#)
 - [pRegTransLayerInfoEvt, 58](#)
 - [pRegTransNWRegInfoEvt, 58](#)
- [_getResetInfoNotification, 59](#)
 - [source, 60](#)
 - [type, 60](#)
- [_getTransLayerInfoResp, 60](#)
 - [pRegInd, 61](#)
 - [pTransLayerInfo, 61](#)
- [_getTransNWRegInfoResp, 61](#)
 - [pRegStatus, 61](#)
- [_modemTempNotification, 62](#)
 - [ModemTempState, 63](#)
 - [ModemTemperature, 62](#)
- [_packetSrvStatus, 63](#)

- bearerID, 64
- connStatus, 64
- ipFamily, 64
- pQmiInterfaceInfo, 64
- reconfigReqd, 64
- sessionEndReason, 64
- techName, 64
- verboseSessnEndReason, 64
- verboseSessnEndReasonType, 64
- _qaQmi3GPP2BroadcastCfgInfo, 64
 - activated_ind, 65
 - CDMABroadcastConfig, 65
 - num_instances, 65
- _qaQmi3GPPBroadcastCfgInfo, 65
 - activated_ind, 66
 - broadcastConfig, 66
 - num_instances, 66
- _setIndicationRegReq, 66
 - pRegCallStatInfoEvt, 67
 - pRegTransLayerInfoEvt, 67
 - pRegTransNWRegInfoEvt, 67
- _slqs3GPPConfigItem, 67
 - LTEAttachProfileListLen, 69
 - p3gppRelease, 69
 - pDefaultPDNEEnabled, 69
 - pLTEAttachProfile, 69
 - pLTEAttachProfileList, 69
 - pProfileList, 69
- _slqsNetworkScanInfo, 70
 - pNetworkInfo, 71
 - pNetworkInfoInstances, 71
 - pPCSDigitInfo, 71
 - pPCSDigitInstances, 71
 - pRATInfo, 71
 - pRATInstances, 71
 - pScanResult, 71
- _sysSelectPrefInfo, 83
 - pBandPref, 87
 - pEmerMode, 87
 - pGWAcqOrderPref, 87
 - pLTEBandPref, 87
 - pModePref, 87
 - pNetSelPref, 87
 - pPRLPref, 87
 - pRoamPref, 87
 - pSrvDomainPref, 88
- _sysSelectPrefParams, 88
 - pAcqOrderPref, 93
 - pBandPref, 93
 - pCSGID, 93
 - pChgDuration, 93
 - pEmerMode, 93
 - pGWAcqOrderPref, 93
 - pLTEBandPref, 93
 - pMNCIncPCSDigStat, 93
 - pModePref, 93
 - pNetSelPref, 93
 - pPRLPref, 93
 - pRAT, 93
 - pRoamPref, 94
 - pSrvDomainPref, 94
 - pSrvRegRestriction, 94
 - pTdsdmaBandPref, 94
 - _transLayerInfoNotification, 94
 - pTransLayerInfo, 95
 - regInd, 95
 - _transLayerinfo, 94
 - TransCap, 94
 - TransType, 94
 - _transNWRegInfoNotification, 95
 - NWRegStat, 96
- AAASPI
 - unpack_wds_GetMobileIPProfile_t, 1066
- AAASState
 - unpack_wds_GetMobileIPProfile_t, 1066
- ABSOLUTE_VALIDITY
 - qaGobiApiSms.h, 1577
- ACT_CODE_MAX_SIZE
 - dms.h, 1206
- ALS
 - getAllCallInformation, 239
- AMSSString
 - unpack_dms_GetFirmwareRevision_t, 932
 - unpack_dms_GetFirmwareRevisions_t, 932
- APNName
 - unpack_wds_SLQSGetRuntimeSettings_t, 1078
- absoluteValidity
 - cdmaMsgDecodingParams, 150
- accelAcceptReady
 - qaGobiApiCbK.h, 1328
- accelAcceptReady_s, 96
 - batchPerSec, 96
 - injectEnable, 96
 - samplesPerBatch, 96
- accelTemp
 - pack_loc_SLQSLOCInjectSensorData_t, 578
- accelTempAcceptReady
 - qaGobiApiCbK.h, 1328
- accelTempAcceptReady_s, 97
 - batchPerSec, 97
 - injectEnable, 97
 - samplesPerBatch, 97
- acceleroData
 - pack_loc_SLQSLOCInjectSensorData_t, 578
- acceleroTimeSrc
 - pack_loc_SLQSLOCInjectSensorData_t, 578
- AccessMac
 - protocolSubtypeElement, 676
- accolc
 - pack_nas_SetACCOLC_t, 583
- ackIndicator
 - SMSTransferRouteMTMessage, 822
 - sMSTransferRouteMTMessage, 821
- acqOrdeLen
 - acqOrderPref, 98
 - nas_acqOrderPref, 405

- acqOrderPref, [97](#)
 - acqOrdeLen, [98](#)
 - pAcqOrder, [98](#)
- acroamsetting
 - pack_wds_SetAutoconnect_t, [622](#)
 - slqsautoconnect, [788](#)
- acsetting
 - pack_wds_SetAutoconnect_t, [622](#)
 - slqsautoconnect, [788](#)
- actCode
 - pack_dms_ActivateAutomatic_t, [552](#)
- ActPilotPNElement, [98](#)
 - ActSetPilotPNStrength, [99](#)
 - ActSetPilotPN, [98](#)
- ActSetCnt
 - NetworkStat1x, [536](#)
- ActSetPilotPNStrength
 - ActPilotPNElement, [99](#)
- ActSetPilotPN
 - ActPilotPNElement, [98](#)
- action
 - slqsautoconnect, [788](#)
 - ssdatasession_params, [826](#)
- ActivateAutomatic
 - qaGobiApiDms.h, [1424](#)
- activated_ind
 - _qaQmi3GPP2BroadcastCfgInfo, [65](#)
 - _qaQmi3GPPBroadcastCfgInfo, [66](#)
- activationState
 - pack_dms_SetActivationStatusCallback_t, [553](#)
- activationStatus
 - dms_ActivationStatusTlv, [209](#)
- ActivationStatusTlv
 - unpack_dms_SetEventReport_ind_t, [941](#)
- activeBandClass
 - nas_RFInfoTlv, [465](#)
 - RFBandInfoElements, [727](#)
- activeChannel
 - nas_RFInfoTlv, [465](#)
 - RFBandInfoElements, [727](#)
- activeInd
 - messageWaitingInfoContent, [399](#)
- ActiveStatus
 - CLIPResp, [164](#)
 - CLIRResp, [164](#)
 - CNAPResp, [166](#)
 - COLPResp, [167](#)
 - COLRResp, [168](#)
- ActiveTechPref
 - unpack_nas_GetNetworkPreference_t, [984](#)
- AddCDMASysInfo, [99](#)
 - geoSysIdx, [99](#)
 - regPrd, [99](#)
- AddSysInfo, [99](#)
 - cellBroadcastCap, [100](#)
 - geoSysIdx, [100](#)
- addr
 - IPv4Addr, [317](#)
 - IPv6Addr, [318](#)
 - unpack_qos_IPv4Addr_t, [1019](#)
 - unpack_qos_IPv6Addr_t, [1020](#)
- address
 - unpack_wds_GetMobileIPProfile_t, [1066](#)
- aid
 - UIMRefreshEvent, [904](#)
 - UIMSessionInformation, [907](#)
 - uim_UIMSessionInformation, [888](#)
 - uim_sessionInformation, [885](#)
- aidLength
 - appStats, [107](#)
 - appStatus, [110](#)
 - UIMRefreshEvent, [904](#)
 - UIMSessionInformation, [907](#)
 - uim_UIMSessionInformation, [888](#)
 - uim_appStatus, [877](#)
 - uim_sessionInformation, [885](#)
- aidVal
 - appStats, [107](#)
 - appStatus, [110](#)
 - uim_appStatus, [877](#)
- aidingIndicatorMask
 - loc_sensorDataUsage, [361](#)
 - sensorDataUsage_s, [747](#)
 - t_sensor, [852](#)
- airTimer, [100](#)
 - airTimerValue, [100](#)
 - namID, [101](#)
- airTimerValue
 - airTimer, [100](#)
- alertPitch
 - signalInfo, [782](#)
- alertingPattern
 - arrAlertingPattern, [112](#)
- AlertingType
 - arrAlertingType, [113](#)
- alertmsg
 - omaDmConfigTlv, [542](#)
 - omaDmConfigTlvExt, [545](#)
 - unpack_omaDmConfigTlv_t, [1015](#)
- alertmsglength
 - omaDmConfigTlv, [542](#)
 - omaDmConfigTlvExt, [545](#)
 - unpack_omaDmConfigTlv_t, [1015](#)
- allCallsAlphaIDInfo, [101](#)
 - AlphaIDInfo, [101](#)
 - callID, [101](#)
- allCallsAlphaIDInfoArr
 - arrAlphaID, [113](#)
- allCallsDiagInfo, [101](#)
 - callID, [102](#)
 - DiagInfo, [102](#)
- AllCallsUUSInfo
 - arrUUSInfo, [120](#)
- allCallsUUSInfo, [102](#)
 - callID, [102](#)
 - uusInfo, [102](#)

- alphaDcs
 - alphaIDInfo, [103](#)
- AlphaIDInfo
 - allCallsAlphaIDInfo, [101](#)
- alphaIDInfo, [102](#)
 - alphaDcs, [103](#)
 - alphaLen, [103](#)
 - alphaText, [103](#)
- alphaIDLen
 - SMSAsyncRawSend_s, [806](#)
- AlphaIDLength
 - CatAlphaIdentifierTlv, [141](#)
- AlphaID
 - CatAlphaIdentifierTlv, [141](#)
- alphaLen
 - alphaIDInfo, [103](#)
- alphaText
 - alphaIDInfo, [103](#)
- alphabet
 - wcdmaMsgEncodingParams, [1153](#)
- altSrcInfo_t, [104](#)
 - coverage, [105](#)
 - linkage, [105](#)
 - source, [105](#)
- Altitude
 - GPSSStateInfo, [276](#)
- altitudeAssumed
 - unpack_loc_GnssSvInfo_Ind_t, [971](#)
- altitudeSrcInfo, [103](#)
 - coverage, [104](#)
 - linkage, [104](#)
 - pack_loc_SLQSLOCInjectPosition_t, [575](#)
 - source, [104](#)
- altitudeWrtEllipsoid
 - pack_loc_SLQSLOCInjectPosition_t, [575](#)
- altitudeWrtMeanSeaLevel
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- ambr_dl
 - sApnExtraParams, [740](#)
 - unpack_qos_SLQSQosSviReadApnExtra↔
Params_t, [1026](#)
- ambr_dl_ext
 - sApnExtraParams, [740](#)
 - unpack_qos_SLQSQosSviReadApnExtra↔
Params_t, [1026](#)
- ambr_dl_ext2
 - sApnExtraParams, [740](#)
 - unpack_qos_SLQSQosSviReadApnExtra↔
Params_t, [1026](#)
- ambr_ul
 - sApnExtraParams, [740](#)
 - unpack_qos_SLQSQosSviReadApnExtra↔
Params_t, [1026](#)
- ambr_ul_ext
 - sApnExtraParams, [740](#)
 - unpack_qos_SLQSQosSviReadApnExtra↔
Params_t, [1026](#)
- ambr_ul_ext2
 - sApnExtraParams, [740](#)
 - unpack_qos_SLQSQosSviReadApnExtra↔
Params_t, [1026](#)
- amssSize
 - unpack_dms_GetFirmwareRevision_t, [932](#)
 - unpack_dms_GetFirmwareRevisions_t, [932](#)
- AnswerUSSD
 - qaGobiApiVoice.h, [1676](#)
- apdoxypages.c, [1195](#)
- apnId
 - pack_qos_SLQSQosSviReadApnExtraParams_t, [601](#)
 - pack_qos_SLQSQosSviReadDataStats_t, [601](#)
 - sApnExtraParams, [740](#)
 - sQosStat, [824](#)
 - unpack_qos_SLQSQosSviReadApnExtra↔
Params_t, [1026](#)
 - unpack_qos_SLQSQosSviReadDataStats_t, [1027](#)
- apnName
 - unpack_wds_SLQSWdsSviPDPRuntimeSettings↔
_t, [1085](#)
- apnname
 - unpack_wds_GetDefaultProfile_t, [1063](#)
- apnsize
 - unpack_wds_GetDefaultProfile_t, [1063](#)
- appNameLength
 - loc_LocApplicationInfo, [357](#)
 - LocApplicationInfo, [364](#)
- appProviderLength
 - loc_LocApplicationInfo, [357](#)
 - LocApplicationInfo, [364](#)
- appState
 - appStats, [107](#)
 - appStatus, [110](#)
 - uim_appStatus, [877](#)
- appStats, [105](#)
 - aidLength, [107](#)
 - aidVal, [107](#)
 - appState, [107](#)
 - appType, [108](#)
 - persoFeature, [108](#)
 - persoRetries, [108](#)
 - persoState, [108](#)
 - persoUnblockRetries, [108](#)
 - pin1Retries, [108](#)
 - pin1State, [108](#)
 - pin2Retries, [108](#)
 - pin2State, [108](#)
 - puk1Retries, [108](#)
 - puk2Retries, [108](#)
 - univPin, [108](#)
- AppStatus
 - slotInf, [786](#)
 - slotInfo, [787](#)
 - uim_slotInfo, [887](#)
- appStatus, [108](#)
 - aidLength, [110](#)
 - aidVal, [110](#)

- appState, 110
- appType, 111
- persoFeature, 111
- persoRetries, 111
- persoState, 111
- persoUnblockRetries, 111
- pin1Retries, 111
- pin1State, 111
- pin2Retries, 111
- pin2State, 111
- puk1Retries, 111
- puk2Retries, 111
- univPin, 111
- appType
 - appStats, 108
 - appStatus, 111
 - uim_appStatus, 877
- appVersionLength
 - loc_LocApplicationInfo, 357
 - LocApplicationInfo, 364
- appVersionValid
 - loc_LocApplicationInfo, 357
 - LocApplicationInfo, 364
- Application
 - unpack_nas_GetCDMANetworkParameters_t, 982
- appversion_str
 - slqsfwinfo_s, 790
 - unpack_dms_GetFirmwareInfo_t, 931
- arfcn
 - GERANInfo, 237
 - gsmCellInfo, 279
 - nas_GERANInfo, 425
 - nas_gsmCellInfo, 427
- arrAlertingPattern, 111
 - alertingPattern, 112
 - callID, 112
 - numInstances, 112
- arrAlertingType, 112
 - AlertingType, 113
 - callID, 113
 - numInstances, 113
- arrAlphaID, 113
 - allCallsAlphaIDInfoArr, 113
 - numInstances, 113
- arrCallEndReason, 114
 - callEndReason, 115
 - callID, 115
 - numInstances, 115
- arrCallInfo, 115
 - getAllCallInfo, 115
 - numInstances, 115
- arrCallInformation
 - voiceSetAllCallStatusCbInfo, 1132
- arrCalledPartyNum, 114
 - CalledPartyNum, 114
 - numInstances, 114
- arrConnectPartyNum, 116
 - ConnectedPartyNum, 116
 - numInstances, 116
- arrDiagInfo, 116
 - DiagInfo, 117
 - numInstances, 117
- arrRedirPartyNum, 117
 - numInstances, 117
 - RedirPartyNum, 117
- arrRemotePartyName, 117
 - GetAllCallRmtPtyName, 118
 - numInstances, 118
- arrRemotePartyNum, 118
 - numInstances, 118
 - RmtPtyNum, 118
- arrSvcOption, 118
 - callID, 119
 - numInstances, 119
 - srvOption, 119
- arrUUSInfo, 119
 - AllCallsUUSInfo, 120
 - numInstances, 120
- arrfileInfo
 - registerRefresh, 723
 - UIMRefreshEvent, 904
- AtCmdPort
 - DcsUsbPortNames, 201
- Audio Service (AUDIO), 49
- auth
 - unpack_wds_GetDefaultProfile_t, 1063
- authData
 - UIMAuthenticateReq, 891
- AuthProt
 - protocolSubtypeElement, 676
- authenticateResult, 120
 - content, 120
 - contentLen, 120
- Authentication
 - unpack_wds_SLQSGetRuntimeSettings_t, 1078
- authentication
 - pack_wds_SetDefaultProfile_t, 623
- authenticationData, 120
 - context, 121
 - data, 121
 - dataLen, 121
- Autosdm
 - unpack_swima_SLQSOMADMGetSettings_t, 1052
- avgPeriod
 - LTESigRptCfg, 391
 - LTESigRptConfig, 391
 - nas_LTESigRptConfig, 448
- azimuth
 - loc_satelliteInfo, 360
 - satelliteInfo, 742
- bAltitudeAssumed
 - gnssSvInfoNotification, 271
- bEnable
 - pack_dms_UIMSetPINProtection_t, 562

- pack_nas_SLQSSetSignalStrengthsCallback_t, 595
- bForceDownload
 - pack_fms_SetImagesPreference_t, 566
- bICCIDLength
 - slot_t, 784
 - UIMSlotStatus, 910
- bICCID
 - slot_t, 784
 - UIMSlotStatus, 910
- bLogicalSlot
 - pack_uim_SLQSUIMSwitchSlot_t, 617
 - slot_t, 784
 - UIMSlotStatus, 910
 - UIMSwitchSlotReq, 911
- bManSize
 - _SLQSSwiGetHostDevInfoParams, 79
 - _SLQSSwiSetHostDevInfoParams, 82
- bModelSize
 - _SLQSSwiGetHostDevInfoParams, 79
 - _SLQSSwiSetHostDevInfoParams, 82
- bNameSize
 - _SLQSSwiGetOSInfoParams, 80
 - _SLQSSwiSetOSInfoParams, 83
- bNumberOfPhySlots
 - UIMSlotStatusChangeInfo, 910
 - unpack_uim_SetUimSlotStatusChangeCallback↔_ind_t, 1057
- BOOL
 - SwiDataTypes.h, 1769
- BPTlv
 - NASQmiCbkNasSystemSelPrefInd, 521
- bPlasmaIDSize
 - _SLQSSwiGetHostDevInfoParams, 79
 - _SLQSSwiSetHostDevInfoParams, 82
- bResetStatistics
 - rmTrasferStaticsReq, 728
 - swiRMTrasferStaticsReq, 849
- bSWVerSize
 - _SLQSSwiGetHostDevInfoParams, 79
 - _SLQSSwiSetHostDevInfoParams, 82
- BUILD_ID_LEN
 - qaGobiApiFms.h, 1459
- BUILD_ID_MAX_LEN
 - qaGobiApiFms.h, 1459
- bVersionSize
 - _SLQSSwiGetOSInfoParams, 80
 - _SLQSSwiSetOSInfoParams, 83
- BYT_STAT_STAT_MASK
 - wds.h, 1798
- BYTE
 - SwiDataTypes.h, 1769
- band
 - LTEInfo, 381
 - nas_LTEInfo, 440
- band1900
 - gsmCellInfo, 279
 - nas_gsmCellInfo, 427
- band_pref
 - NASBandPreferenceTlv, 494
- BandCapability
 - unpack_dms_GetBandCapability_t, 926
- bandCapability
 - BandCapabilityResp, 122
 - unpack_dms_SLQSGetBandCapability_t, 946
- BandCapabilityResp, 121
 - bandCapability, 122
 - pLteBandCapability, 122
 - pTdsBandCapability, 123
- bandwidth
 - LTEInfo, 381
 - nas_LTEInfo, 440
- baseId
 - CDMAInfo, 148
 - CDMASysInfo, 157
 - nas_CDMAInfo, 410
 - nas_CDMASysInfo, 413
- baseLat
 - CDMAInfo, 148
 - CDMASysInfo, 157
 - nas_CDMAInfo, 410
 - nas_CDMASysInfo, 413
- baseLong
 - CDMAInfo, 148
 - CDMASysInfo, 158
 - nas_CDMAInfo, 410
 - nas_CDMASysInfo, 413
- BasestationID
 - qaQmiServingSystemParam, 682
 - unpack_nas_SLQSGetServingSystem_t, 995
- BasestationLatitude
 - qaQmiServingSystemParam, 682
 - unpack_nas_SLQSGetServingSystem_t, 995
- BasestationLongitude
 - qaQmiServingSystemParam, 682
 - unpack_nas_SLQSGetServingSystem_t, 995
- batchPerSec
 - accelAcceptReady_s, 96
 - accelTempAcceptReady_s, 97
 - gyroAcceptReady_s, 284
 - gyroTempAcceptReady_s, 285
- BdsSVInfo, 123
 - len, 124
 - pSV, 124
- BdsSV, 123
 - id, 123
 - mask, 123
- BearerID
 - unpack_qos_QosFlowInfo_t, 1023
- bearerID
 - _packetSrvStatus, 64
 - unpack_wds_SLQSSetPacketSrvStatusCallback↔_t, 1080
- bearerId
 - sQosFlowStat, 822
 - unpack_QosFlowStat_t, 1040

- unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔
_t, 1085
- bootSize
 - unpack_dms_GetFirmwareRevisions_t, 932
- BootString
 - unpack_dms_GetFirmwareRevisions_t, 932
- bootversion_str
 - slqsfwinfo_s, 790
 - unpack_dms_GetFirmwareInfo_t, 931
- Broadcast
 - unpack_nas_GetCDMANetworkParameters_t, 982
- BroadcastConfig, 124
 - fromServiceId, 124
 - selected, 124
 - toServiceId, 124
- broadcastConfig
 - _qaQmi3GPPBroadcastCfgInfo, 66
- bsInfoValid
 - CDMASysInfo, 158
 - nas_CDMASysInfo, 413
- bsPRev
 - CDMASysInfo, 158
 - nas_CDMASysInfo, 413
- bsPRevValid
 - CDMASysInfo, 158
 - nas_CDMASysInfo, 414
- bsic
 - GERANInfo, 237
 - nas_GERANInfo, 425
- bsicId
 - gsmCellInfo, 279
 - nas_gsmCellInfo, 427
- bucketSz
 - tokenBucket, 867
 - unpack_qos_tokenBucket_t, 1039
- buildIDLen
 - CurrImageInfo, 184
 - image_info_t, 296
- buildIDLength
 - FMSImageIdElement, 232
 - ImageIdElement, 297
- buildID
 - CurrImageInfo, 184
 - FMSImageIdElement, 232
 - image_info_t, 296
 - ImageIdElement, 297
- buildId
 - FMSImageElement, 231
 - ImageElement, 296
- buildIdLength
 - FMSImageElement, 232
 - ImageElement, 296
- BurstDTMFInfo
 - voiceBurstDTMFInfo, 1096
- burstDTMFInfo, 125
 - digitCnt, 125
 - pCallID, 125
 - pDigitBuff, 125
- ByteLoopbackMode
 - unpack_wds_SLQSSGetLoopback_t, 1083
 - WDSGetLoopbackData, 1185
- ByteLoopbackMultiplier
 - unpack_wds_SLQSSGetLoopback_t, 1083
 - WDSGetLoopbackData, 1185
- ByteTotalsElmntsV4
 - WdsByteTotals, 1175
- ByteTotalsElmntsV6
 - WdsByteTotals, 1175
- CATEventDataType, 142
 - eventMask, 142
 - pErrorMask, 142
- CATSendEnvelopeCommand
 - qaGobiApiCat.h, 1317
- CATSendTerminalResponse
 - qaGobiApiCat.h, 1317
- CBK_DISABLE_EVENT
 - qaGobiApiCbK.h, 1326
- CBK_ENABLE_EVENT
 - qaGobiApiCbK.h, 1326
- CBK_NOCHANGE
 - qaGobiApiCbK.h, 1326
- CCETlv
 - QmiCbKCatEventStatusReportInd, 683
- CDMA_P_Rev
 - qaQmiServingSystemParam, 682
 - unpack_nas_SLQSSGetServingSystem_t, 995
- CDMABroadcastConfig, 145
 - _qaQmi3GPP2BroadcastCfgInfo, 65
 - language, 146
 - selected, 146
 - serviceCategory, 146
- CDMAChannel, 146
 - priChA, 146
 - priChB, 146
 - secChA, 146
 - secChB, 146
- CDMAECIOThresh, 147
 - CDMAECIOThreshListLen, 147
 - pCDMAECIOThreshList, 147
- CDMAECIOThreshListLen
 - CDMAECIOThresh, 147
 - nas_CDMAECIOThresh, 409
- CDMAInfo, 147
 - baseId, 148
 - baseLat, 148
 - baseLong, 148
 - nid, 148
 - refpn, 148
 - sid, 148
- CDMARSSIThresh, 153
 - CDMARSSIThreshListLen, 153
 - pCDMARSSIThreshList, 153
- CDMARSSIThreshListLen
 - CDMARSSIThresh, 153
 - nas_CDMARSSIThresh, 410
- CDMASSInfo, 154

- ecio, [154](#)
- rsi, [154](#)
- unpack_nas_SLQSNasGetSigInfo_t, [1005](#)
- CDMASysInfo, [155](#)
 - baseId, [157](#)
 - baseLat, [157](#)
 - baseLong, [158](#)
 - bsInfoValid, [158](#)
 - bsPRev, [158](#)
 - bsPRevValid, [158](#)
 - ccsSupported, [158](#)
 - ccsSupportedValid, [158](#)
 - cdmaSysIdValid, [158](#)
 - isSysPrIMatch, [158](#)
 - isSysPrIMatchValid, [158](#)
 - MCC, [158](#)
 - MNC, [158](#)
 - networkID, [158](#)
 - networkIdValid, [158](#)
 - pRevInUse, [158](#)
 - pRevInUseValid, [158](#)
 - packetZone, [158](#)
 - packetZoneValid, [158](#)
 - sysInfoCDMA, [158](#)
 - systemID, [158](#)
- CDMASysInfoExt, [158](#)
 - imsi_11_12, [159](#)
 - MCC, [159](#)
- CDMASystemInfoExt
 - qaQmiServingSystemParam, [682](#)
 - unpack_nas_SLQSGetServingSystem_t, [995](#)
- CHAR
 - SwiDataTypes.h, [1769](#)
- CK_MAX_SIZE
 - dms.h, [1206](#)
- CLIPResp, [163](#)
 - ActiveStatus, [164](#)
 - ProvisionStatus, [164](#)
- CLIRResp, [164](#)
 - ActiveStatus, [164](#)
 - ProvisionStatus, [164](#)
- CNAPResp, [165](#)
 - ActiveStatus, [166](#)
 - ProvisionStatus, [166](#)
- COLPResp, [166](#)
 - ActiveStatus, [167](#)
 - ProvisionStatus, [167](#)
- COLRResp, [167](#)
 - ActiveStatus, [168](#)
 - ProvisionStatus, [168](#)
- CONFIG_LEN
 - qaGobiApiSms.h, [1577](#)
- CQIValueCW0
 - LteCQIParm, [377](#)
 - unpack_nas_SLQSSwiGetLteCQI_t, [1010](#)
- CQIValueCW1
 - LteCQIParm, [377](#)
 - unpack_nas_SLQSSwiGetLteCQI_t, [1010](#)
- CSDomain
 - unpack_nas_GetServingNetwork_t, [985](#)
- CSGID, [177](#)
 - id, [178](#)
 - mcc, [178](#)
 - mnc, [178](#)
 - mncPcsDigits, [178](#)
 - rat, [178](#)
- CUGIndex
 - CUGInfo, [179](#)
- CUGInfo, [178](#)
 - CUGIndex, [179](#)
 - SuppOA, [179](#)
 - SuppPrefCUG, [179](#)
- CallBackK registration (CBK), [37](#)
- CallBarStatus
 - qaQmiServingSystemParam, [682](#)
 - unpack_nas_SLQSGetServingSystem_t, [995](#)
- callBarStatus, [126](#)
 - csBarStatus, [127](#)
 - psBarStatus, [127](#)
- CallBarringSysInfo, [125](#)
 - csBarStatus, [126](#)
 - psBarStatus, [126](#)
- callDuration
 - unpack_wds_GetSessionDuration_t, [1070](#)
- CallEndReason
 - DUNCallInfoInd, [218](#)
- callEndReason
 - arrCallEndReason, [115](#)
 - unpack_wds_SLQSGetDUNCallInfo_t, [1076](#)
- CallFWExtInfo
 - getCallFWExtInfo, [247](#)
- callFWExtInfo, [131](#)
 - noReplyTimer, [133](#)
 - numLen, [133](#)
 - numPlan, [133](#)
 - numType, [133](#)
 - number, [133](#)
 - PI, [133](#)
 - SI, [133](#)
 - SvcClass, [133](#)
 - SvcStatus, [133](#)
- CallFWInfo
 - getCallFWInfo, [247](#)
- callFWInfo, [133](#)
 - noReplyTimer, [134](#)
 - numLen, [134](#)
 - number, [134](#)
 - SvcClass, [134](#)
 - SvcStatus, [134](#)
- callFwdTypeAndPlan, [130](#)
 - numberPlan, [131](#)
 - numberType, [131](#)
- callID
 - allCallsAlphaIDInfo, [101](#)
 - allCallsDiagInfo, [102](#)
 - allCallsUUSInfo, [102](#)

- arrAlertingPattern, [112](#)
- arrAlertingType, [113](#)
- arrCallEndReason, [115](#)
- arrSvcOption, [119](#)
- callInfo, [136](#)
- DTMFInfo, [216](#)
- getAllCallRmtPtyName, [240](#)
- getAllCallRmtPtyNum, [240](#)
- peerNumberInfo, [649](#)
- voiceCallInfoReq, [1097](#)
- voiceInfoRec, [1126](#)
- voiceOTASPStatusInfo, [1130](#)
- voicePrivacyInfo, [1130](#)
- voiceSUPSNotification, [1146](#)
- voiceStopContDTMFInfo, [1142](#)
- callInfo, [134](#)
 - callID, [136](#)
 - callState, [136](#)
 - callType, [136](#)
 - direction, [136](#)
 - mode, [136](#)
- callNumber
 - voiceCallRequestParams, [1101](#)
- callState
 - callInfo, [136](#)
- callType
 - callInfo, [136](#)
- calledPartyInfo, [127](#)
 - numLen, [129](#)
 - numPlan, [129](#)
 - numType, [129](#)
 - number, [128](#)
 - PI, [129](#)
 - SI, [129](#)
- CalledPartyNum
 - arrCalledPartyNum, [114](#)
- calledPartySubAdd, [129](#)
 - extBit, [130](#)
 - oddEvenInd, [130](#)
 - subAddr, [130](#)
 - subAddrLen, [130](#)
 - subAddrType, [130](#)
- callerIDInfo, [130](#)
 - callerIDLen, [130](#)
 - callerID, [130](#)
 - PI, [130](#)
- callerIDLen
 - callerIDInfo, [130](#)
 - connectNumInfo, [172](#)
- callerID
 - callerIDInfo, [130](#)
 - connectNumInfo, [172](#)
- callerName
 - remotePartyName, [724](#)
- Callinfo
 - getAllCallInformation, [239](#)
- callingPartyInfo, [136](#)
 - numLen, [138](#)
 - numPlan, [138](#)
 - numType, [138](#)
 - number, [137](#)
 - PI, [138](#)
 - SI, [138](#)
- CancelUSSD
 - qaGobiApiVoice.h, [1677](#)
- Card Application Toolkit (CAT), [40](#)
- cardResult, [138](#)
 - sw1, [138](#)
 - sw2, [138](#)
- cardState
 - slotInf, [786](#)
 - slotInfo, [787](#)
 - uim_slotInfo, [887](#)
- cardStatus, [138](#)
 - index1xPri, [139](#)
 - index1xSec, [139](#)
 - indexGwPri, [139](#)
 - indexGwSec, [139](#)
 - numSlot, [139](#)
 - SlotInfo, [139](#)
- Carrier
 - fwinfow_s, [236](#)
- carrier
 - CurrentImgList, [182](#)
 - unpack_dms_SLQSSwiGetFirmwareCurr_t, [949](#)
- carrier_str
 - slqsfwinfow_s, [790](#)
 - unpack_dms_GetFirmwareInfo_t, [931](#)
- CarrierImage_t, [140](#)
 - m_FwBuildId, [140](#)
 - m_FwImgId, [140](#)
 - m_PriBuildId, [141](#)
 - m_PriImgId, [141](#)
 - m_nCarrierId, [141](#)
 - m_nFolderId, [141](#)
 - m_nStorage, [141](#)
- CatAlPhalIdentifierTlv, [141](#)
 - AlphaIDLength, [141](#)
 - AlphaID, [141](#)
 - ReferenceID, [141](#)
- CatAlphaIdtfr
 - currentCatEvent, [181](#)
- CatCommonEventTlv, [141](#)
 - CatEvent, [142](#)
 - EventID, [142](#)
 - EventLength, [142](#)
 - TlvPresent, [142](#)
- CatEndProactiveSessionTlv, [142](#)
 - EndProactiveSession, [142](#)
- CatEndPS
 - currentCatEvent, [181](#)
- CatEvIDData
 - currentCatEvent, [181](#)
- CatEvent
 - CatCommonEventTlv, [142](#)
- CatEventIDDataTlv, [143](#)

- Data, 143
- DataLength, 143
- ReferenceID, 143
- CatEventListTlv, 143
- SetupEventList, 144
- CatEventLst
 - currentCatEvent, 181
- CatRefresh
 - currentCatEvent, 181
- CatRefreshTlv, 144
 - RefreshMode, 144
 - RefreshStage, 144
- causeCode
 - SMSAsyncRawSend_s, 806
- ccSUPSType, 144
 - reason, 145
 - svcType, 145
- ccsSupported
 - CDMASysInfo, 158
 - nas_CDMASysInfo, 414
- ccsSupportedValid
 - CDMASysInfo, 158
 - nas_CDMASysInfo, 414
- cdmaMsgDecodingParams, 148
 - absoluteValidity, 150
 - mcTimeStamp, 150
 - messageLength, 151
 - pAlertPriority, 151
 - pCallbkAddr, 151
 - pCallbkAddrLength, 151
 - pDisplayMode, 151
 - pLanguage, 151
 - pMessage, 151
 - pMessageID, 151
 - pPriority, 151
 - pPrivacy, 151
 - pReadAcknowledgementReq, 151
 - pRelativeValidity, 151
 - pSenderAddr, 151
 - pSenderAddrLength, 151
 - pTextMsg, 151
 - pTextMsgLength, 151
 - pUserAcknowledgementReq, 151
- cdmaMsgEncodingParams, 151
 - messageld, 152
 - pCallbackAddr, 152
 - pDestAddr, 153
 - pEncodingAlphabet, 153
 - pMessage, 153
 - pMessageSize, 153
 - pPriority, 153
 - pRelValidity, 153
 - pTextMsg, 153
 - textMsgLength, 153
- cdmaSSInfo, 154
 - ecio, 154
 - rsi, 154
- cdmaSysIdValid
 - CDMASysInfo, 158
 - nas_CDMASysInfo, 414
- cell_resel_priority
 - infoInterFreq, 316
 - nas_infoInterFreq, 437
- cellBroadcastCap
 - AddSysInfo, 100
 - nas_AddSysInfo, 406
- CellDb, 159
 - mask, 159
- CellID
 - qaQmiServingSystemParam, 682
 - unpack_nas_SLQSGetservingSystem_t, 995
- cellID
 - GERANInfo, 237
 - nas_GERANInfo, 425
 - nas_UMTSInfo, 483
 - UMTSInfo, 915
- cellId
 - GSMSysInfo, 283
 - LTESysInfo, 397
 - nas_GSMSysInfo, 431
 - nas_LTESysInfo, 451
 - nas_WCDMASysInfo, 493
 - WCDMASysInfo, 1157
- cellIdValid
 - GSMSysInfo, 283
 - gsmCellInfo, 279
 - LTESysInfo, 397
 - nas_GSMSysInfo, 431
 - nas_LTESysInfo, 451
 - nas_WCDMASysInfo, 493
 - nas_gsmCellInfo, 427
 - WCDMASysInfo, 1157
- cellInterFreqParams
 - infoInterFreq, 316
 - nas_infoInterFreq, 437
- cellsTDD
 - nas_umtsLTENbrCell, 486
 - umtsLTENbrCell, 917
- CellParams
 - LTEInfoIntrafreq, 384
 - nas_LTEInfoIntrafreq, 443
- cellParams, 159
 - pci, 160
 - rsrp, 160
 - rsrq, 160
 - rsi, 160
 - srxlev, 160
- cellReselPriority
 - LTEInfoIntrafreq, 384
 - lteGsmCellInfo, 379
 - lteWcdmaCellInfo, 398
 - nas_LTEInfoIntrafreq, 443
 - nas_lteGsmCellInfo, 438
 - nas_lteWcdmaCellInfo, 453
- cells_len
 - infoInterFreq, 316

- lteGsmCellInfo, 379
- nas_infoInterFreq, 437
- nas_lteGsmCellInfo, 438
- cellsLen
 - lteInfoIntraFreq, 384
 - lteWcdmaCellInfo, 398
 - nas_lteInfoIntraFreq, 443
 - nas_lteWcdmaCellInfo, 453
- chaddr
 - wds_DHCPv4HWConfig, 1163
 - WdsDHCPv4HWConfig, 1180
 - wdsDhcpv4HwConfig, 1180
- chaddrLen
 - wds_DHCPv4HWConfig, 1163
 - WdsDHCPv4HWConfig, 1180
 - wdsDhcpv4HwConfig, 1180
- changePIN
 - pack_uim_ChangePin_t, 612
 - UIMChangePinReq, 892
- changeUIMPIN, 161
 - oldPINLen, 161
 - oldPINVal, 161
 - pinID, 161
 - pinLen, 161
 - pinVal, 161
- ChannelRate, 162
 - CurrChanRxRate, 162
 - CurrChanTxRate, 162
 - DUNCallInfoInd, 218
 - MaxChanRxRate, 162
 - MaxChanTxRate, 162
- channelRate, 162
 - CurrChanRxRate, 163
 - CurrChanTxRate, 163
 - unpack_wds_SLQSGetDUNCallInfo_t, 1076
- Chipset
 - DeviceConfigDetail, 206
- ckLen
 - depersonalizationInformation, 203
- ckVal
 - depersonalizationInformation, 203
- clear
 - pack_dms_SLQSSwiGetCrashInfo_t, 557
- ClkInfo, 164
 - mask, 165
- codingScheme
 - PLMNNetworkNameData, 659
 - remotePartyName, 724
- CommInfo, 168
 - imsRegState, 169
 - modemMode, 169
 - psState, 169
 - systemMode, 169
 - temperature, 169
- common.h, 1195
 - DEFAULT_LOC_TIMEOUT_IN_SEC, 1197
 - eCAT, 1198
 - eCTL, 1198
 - eDMS, 1198
 - eIND, 1198
 - eLOG_DEBUG, 1197
 - eLOG_FATAL, 1197
 - eLOG_INFO, 1197
 - eLOG_LEVEL, 1197
 - eLOG_WARN, 1197
 - eLOC, 1198
 - eNAS, 1198
 - eQMI_SVC, 1197
 - eQOS, 1198
 - eREQ, 1198
 - eRSP, 1198
 - eSMS, 1198
 - eSWILOC, 1198
 - eSWIOMA, 1198
 - eTIMEOUT_10_S, 1198
 - eTIMEOUT_20_S, 1198
 - eTIMEOUT_2_S, 1198
 - eTIMEOUT_300_S, 1198
 - eTIMEOUT_30_S, 1198
 - eTIMEOUT_5_S, 1198
 - eTIMEOUT_60_S, 1198
 - eTIMEOUT_8_S, 1198
 - eTIMEOUT_DEFAULT, 1198
 - eTMD, 1198
 - eTimeout, 1198
 - eUIM, 1198
 - eWDS, 1198
 - fill_pack_ctx, 1199
 - fill_sdu_hdr, 1199
 - get_version, 1199
 - glog, 1200
 - gloglvl, 1200
 - helper_get_resp_ctx, 1199
 - helper_get_xid, 1199
 - helper_set_log_func, 1199
 - helper_set_log_lvl, 1199
 - libpack_GetVersion, 1199
 - libpack_log, 1199
 - logger, 1197
 - MINREQBKLEN, 1197
 - MSGID_AND_LEN, 1197
 - MSGID_DONT_CARE, 1197
 - msgtype, 1198
 - SDK_VALIDATE_INPUT_PACK_PARAM_AND←_FILL_XID, 1197
 - SDK_VALIDATE_INPUT_PACK_PARAM, 1197
 - SDU_HDR_LEN, 1197
 - UNUSEDPARAM, 1197
 - unpack_result_code_only, 1200
- commonInfo
 - swiModemStatusResp, 836
 - unpack_nas_SLQSNasSwiModemStatus_t, 1008
- ConcSvcInfo
 - unpack_nas_SLQSGetServingSystem_t, 995
- concSvcInfo
 - qaQmiServingSystemParam, 682

- confidence
 - pack_loc_SLQSLOCSetCradleMountConfig_t, 580
- conn_status
 - unpack_wds_SLQSSetPacketSrvStatusCallback↔_t, 1080
- ConnRateElmntsV4
 - WdsConnectionRate, 1177
- ConnRateElmntsV6
 - WdsConnectionRate, 1177
- connStatus
 - _packetSrvStatus, 64
- connectNumInfo, 171
 - callerIDLen, 172
 - callerID, 172
 - numPlan, 172
 - numPresInd, 172
 - numType, 172
 - screeningInd, 172
- ConnectedPartyNum
 - arrConnectPartyNum, 116
- ConnectionStatus, 169
 - MDMCallDuration, 170
 - MDMConnStatus, 170
- connectionStatus, 170
 - MDMCallDuration, 170
 - MDMConnStatus, 170
 - unpack_wds_GetSessionState_t, 1070
 - unpack_wds_SLQSGetDUNCallInfo_t, 1076
- connetionState
 - imsaPdpStatusInfo, 301
- content
 - authenticateResult, 120
 - readResult, 719
 - uim_readResult, 883
- contentLen
 - authenticateResult, 120
 - readResult, 720
 - uim_readResult, 883
- context
 - authenticationData, 121
- contextId
 - pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 640
 - swiPDPRuntimeSettingsReq, 837
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, 1085
- contextType
 - pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 640
 - swiPDPRuntimeSettingsReq, 837
- ControlMac
 - protocolSubtypeElement, 676
- Count1
 - RankIndicatorInd, 719
- Count2
 - RankIndicatorInd, 719
- countryInitials
 - PLMNNetworkNameData, 659
- coverage
 - altSrcInfo_t, 105
 - altitudeSrcInfo, 104
- cpich_ecno
 - nas_wcdmaCellInfo, 488
 - wcdmaCellInfo, 1147
- cpich_rscp
 - nas_wcdmaCellInfo, 488
 - wcdmaCellInfo, 1147
- cradleMountConfigStatus
 - QmiCbkLocCradleMountInd, 690
- crashAction
 - pack_dms_SetCrashAction_t, 554
- crashData
 - CrashInfo, 173
 - crashInformation, 175
- crashId
 - CrashInfo, 173
 - crashInformation, 175
- CrashInfo, 172
 - crashData, 173
 - crashId, 173
 - crashStrLen, 173
 - gcDumpStrLen, 173
 - numCrashes, 173
 - pCrashString, 173
 - pGCDumpString, 173
- crashInfo
 - crashInfoParams, 174
- crashInfoParam
 - unpack_dms_SLQSSwiGetCrashInfo_t, 947
- CrashInfoParams, 174
 - pCrashInfo, 174
 - pDevCrashStatus, 174
- crashInfoParams, 173
 - crashInfo, 174
 - crashStatus, 174
- crashInformation, 174
 - crashData, 175
 - crashId, 175
 - crashString, 175
 - crashStrlen, 175
 - gcdumpString, 175
 - gcdumpStrlen, 175
 - numCrashes, 176
- crashStatus
 - crashInfoParams, 174
- crashStrLen
 - CrashInfo, 173
- crashString
 - crashInformation, 175
- crashStrlen
 - crashInformation, 175
- CreateProfileIn, 176
 - curProfile, 176
 - pProfileID, 176
 - pProfileType, 176
- CreateProfileOut, 176

- pExtErrorCode, 177
 - pProfileIndex, 177
 - pProfileType, 177
- csAttachState
 - NASServingSystemInfo, 523
 - nas_servSystem, 471
 - servSystem, 752
 - ServingSystemInfo, 750
- csBarStatus
 - callBarStatus, 127
 - CallBarringSysInfo, 126
 - nas_CallBarringSysInfo, 407
 - nas_callBarStatus, 408
- cur_carr_name
 - slqsfwinfo_s, 790
 - unpack_dms_GetFirmwareInfo_t, 931
- cur_carr_rev
 - slqsfwinfo_s, 790
 - unpack_dms_GetFirmwareInfo_t, 931
- curAMRConfig, 179
 - gsmAmrStat, 180
 - wcdmaAmrStat, 180
- curDataBearerTechnology
 - unpack_wds_SLQSGetDataBearerTechnology_t, 1075
- curProfile
 - _GetProfileSettingOut, 59
 - CreateProfileIn, 176
 - ModifyProfileIn, 403
 - pack_wds_SLQSMModifyProfile_t, 632
 - UnPackGetProfileSettingOut, 1086
- CurrChanRxRate
 - ChannelRate, 162
 - channelRate, 163
 - dunchannelRate, 218
- CurrChanTxRate
 - ChannelRate, 162
 - channelRate, 163
 - dunchannelRate, 218
- currDBTechAvail
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, 1082
- CurrDataSysStat, 180
 - pCurrNetworkInfo, 180
 - pNetworkInfoLen, 180
 - pPrefNetwork, 180
- CurrImageInfo, 183
 - buildIDLen, 184
 - buildID, 184
 - imageType, 184
 - uniqueID, 184
- currNWInfo
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, 1082
- CurrNetworkInfo, 184
 - NetworkType, 186
 - RATMask, 186
 - SOMask, 186
- currNetworkInfo, 186
 - NetworkType, 187
 - RATMask, 187
 - SOMask, 187
 - unpack_wds_SLQSGetCurrDataSystemStat_t, 1073
- current_channel_rx_rate
 - unpack_wds_SLQSGetCurrentChannelRate_t, 1074
 - WDSSWICurrentChannelRates, 1193
- current_channel_tx_rate
 - unpack_wds_SLQSGetCurrentChannelRate_t, 1074
 - WDSSWICurrentChannelRates, 1193
- currentCatEvent, 180
 - CatAlphaIdtfr, 181
 - CatEndPS, 181
 - CatEvIDData, 181
 - CatEventLst, 181
 - CatRefresh, 181
- currentChannelRXRate
 - unpack_wds_GetConnectionRate_t, 1062
- currentChannelTXRate
 - unpack_wds_GetConnectionRate_t, 1062
- currentDataBearer
 - pack_wds_SLQSSetWdsEventCallback_t, 635
- CurrentImglst, 181
 - carrier, 182
 - fwvers, 182
 - numEntries, 182
 - pCurrImglstInfo, 182
 - pkgver, 182
 - priver, 182
- currentMitigationLvl
 - QmiCbkTmdMitiLvlRptInd, 704
- currentNetwork
 - dataBearerTechnology, 197
 - qmiWDSDataBearerTechnology, 709
- CurrentPLMN
 - qaQmiServingSystemParam, 682
 - unpack_nas_SLQSGetServingSystem_t, 995
- currentPLMN, 182
 - MCC, 183
 - MNC, 183
 - netDescr, 183
 - netDescrLength, 183
- cust_attr
 - custSettingInfo, 192
 - DMScustSettingInfo, 211
- cust_id
 - custSettingInfo, 192
 - DMScustSettingInfo, 211
 - DMSgetCustomInput, 213
 - getCustomInput, 248
 - pack_dms_GetCustFeaturesV2_t, 552
 - pack_dms_SetCustFeaturesV2_t, 555
 - setCustomSettingV2, 760
- cust_value

- custSettingInfo, [192](#)
 - DMScustSettingInfo, [211](#)
 - pack_dms_SetCustFeaturesV2_t, [555](#)
 - setCustomSettingV2, [760](#)
- custFeaturesInfo, [187](#)
 - GpsEnable, [189](#)
 - pDHCPRelayEnabled, [189](#)
 - pDisableIMSI, [189](#)
 - pGPSLPM, [189](#)
 - pGPSSel, [189](#)
 - pIPFamSupport, [189](#)
 - plsVoiceEnabled, [189](#)
 - pRMAutoConnect, [189](#)
 - pSMSSupport, [189](#)
 - qaGobiApiDms.h, [1417](#)
- custFeaturesSetting, [189](#)
 - pDHCPRelayEnabled, [191](#)
 - pGPSEnable, [191](#)
 - pGPSLPM, [191](#)
 - pGPSSel, [191](#)
 - plsVoiceEnabled, [191](#)
 - qaGobiApiDms.h, [1419](#)
- custSetting
 - custSettingList, [192](#)
 - DMScustSettingList, [212](#)
- custSettingInfo, [191](#)
 - cust_attr, [192](#)
 - cust_id, [192](#)
 - cust_value, [192](#)
 - id_length, [192](#)
 - value_length, [192](#)
- custSettingList, [192](#)
 - custSetting, [192](#)
 - list_type, [192](#)
 - num_instances, [192](#)
- CustomSCP
 - unpack_nas_GetCDMANetworkParameters_t, [982](#)
- CwtMute
 - GetM2MAVMuteResp, [264](#)
 - GetM2MAudioProfileResp, [262](#)
- dBTechAvail
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- dBTechnology
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- DEFAULT_LOC_TIMEOUT_IN_SEC
 - common.h, [1197](#)
- DEFAULTBYTEVALUE
 - qaGobiApiPds.h, [1549](#)
- DEFAULTLONGVALUE
 - qaGobiApiPds.h, [1549](#)
- DEFAULTWORDVALUE
 - qaGobiApiPds.h, [1549](#)
- DEREGISTER_EVENT
 - qaGobiApiCbk.h, [1326](#)
- DEREGISTER_SRV
 - qaGobiApiCbk.h, [1326](#)
- DEVICE_OFFLINE
 - qaGobiApiFms.h, [1459](#)
- DEVICE_RESET
 - qaGobiApiFms.h, [1459](#)
- DEVICE_SHUTDOWN
 - qaGobiApiFms.h, [1459](#)
- DEVICE_STATE_BOOT
 - qaGobiApiCbk.h, [1364](#)
- DEVICE_STATE_DISCONNECTED
 - qaGobiApiCbk.h, [1364](#)
- DEVICE_STATE_READY
 - qaGobiApiCbk.h, [1364](#)
- DHCP_MAX_NUM_OPTIONS
 - qaGobiApiCbk.h, [1326](#)
- DHCP_OPTION_DATA_BUF_SIZE
 - qaGobiApiCbk.h, [1326](#)
- DHCPOption, [206](#)
 - optCode, [206](#)
 - optValLen, [206](#)
 - pOptVal, [206](#)
- DHCPOptionList, [206](#)
 - numOpt, [207](#)
 - pOptions, [207](#)
- DHCPRelayEnabled
 - pack_dms_SetCustFeature_t, [554](#)
 - unpack_dms_GetCustFeature_t, [927](#)
- DHCPv4LeaseOptTlv
 - unpack_wds_DHCPv4ClientLease_ind_t, [1060](#)
- DHCPv4LeaseStateTlv
 - unpack_wds_DHCPv4ClientLease_ind_t, [1060](#)
- DMS_IMGDETAILS_LEN
 - dms.h, [1206](#)
- DMS_MAX_CUST_ID_LEN
 - dms.h, [1206](#)
- DMS_MAX_CUST_VALUE_LEN
 - dms.h, [1206](#)
- DMS_MAX_FWUPDATE_LOG_STR_SZ
 - dms.h, [1206](#)
- DMS_MAX_FWUPDATE_REF_STR_SZ
 - dms.h, [1207](#)
- DMS_PM_FACTORY
 - dms.h, [1207](#)
- DMS_PM_LOW
 - dms.h, [1207](#)
- DMS_PM_OFFLINE
 - dms.h, [1207](#)
- DMS_PM_ONLINE
 - dms.h, [1207](#)
- DMS_PM_PERSISTENT_LOW
 - dms.h, [1207](#)
- DMS_PM_RESET
 - dms.h, [1207](#)
- DMS_PM_SHUT_DOWN
 - dms.h, [1207](#)
- DMS_SET_REPORT_DISABLE
 - dms.h, [1207](#)
- DMS_SET_REPORT_ENABLE
 - dms.h, [1207](#)

- DMS_SLQSFWINFO_APPVERSION_SZ
 - dms.h, [1207](#)
- DMS_SLQSFWINFO_BOOTVERSION_SZ
 - dms.h, [1207](#)
- DMS_SLQSFWINFO_CARRIER_SZ
 - dms.h, [1207](#)
- DMS_SLQSFWINFO_CUR_CARR_NAME
 - dms.h, [1207](#)
- DMS_SLQSFWINFO_CUR_CARR_REV
 - dms.h, [1207](#)
- DMS_SLQSFWINFO_MODELID_SZ
 - dms.h, [1207](#)
- DMS_SLQSFWINFO_PACKAGEID_SZ
 - dms.h, [1207](#)
- DMS_SLQSFWINFO_PRIVERSION_SZ
 - dms.h, [1207](#)
- DMS_SLQSFWINFO_SKU_SZ
 - dms.h, [1207](#)
- DMS_SWI_SET_IND_DISABLE
 - dms.h, [1207](#)
- DMS_SWI_SET_IND_ENABLE
 - dms.h, [1207](#)
- DMS_UINT8_MAX_STRING_SZ
 - dms.h, [1207](#)
- DMScstSettingInfo, [210](#)
 - cust_attr, [211](#)
 - cust_id, [211](#)
 - cust_value, [211](#)
 - id_length, [211](#)
 - value_length, [211](#)
- DMScstSettingList, [211](#)
 - custSetting, [212](#)
 - list_type, [212](#)
 - num_instances, [212](#)
- DMSgetCustomFeatureV2, [212](#)
 - pCustSettingInfo, [212](#)
 - pCustSettingList, [212](#)
 - pGetCustomInput, [212](#)
- DMSgetCustomInput, [212](#)
 - cust_id, [213](#)
 - list_type, [213](#)
- DRCover
 - DRCParams, [216](#)
- DRCParams, [215](#)
 - DRCover, [216](#)
 - DRCValue, [216](#)
- DRCValue
 - DRCParams, [216](#)
- DTMFEvent
 - DTMFInfo, [217](#)
- DTMFInfo, [216](#)
 - callID, [216](#)
 - DTMFEvent, [217](#)
 - digitBuff, [216](#)
 - digitCnt, [216](#)
- DTMFInformation
 - voiceDTMFEventInfo, [1104](#)
- DTMFInterdigitInterval
 - DTMFLengths, [217](#)
- DTMFLengths, [217](#)
 - DTMFInterdigitInterval, [217](#)
 - DTMFPulseWidth, [217](#)
- DTMFPulseWidth
 - DTMFLengths, [217](#)
- DTMFdigit
 - voiceContDTMFInfo, [1103](#)
- DTMInd
 - qaQmiServingSystemParam, [682](#)
 - unpack_nas_SLQSGetServingSystem_t, [995](#)
- DUNCallInfoInd, [217](#)
 - CallEndReason, [218](#)
 - ChannelRate, [218](#)
 - DataBearerTech, [218](#)
 - DormancyStatus, [218](#)
 - MdmConnStatus, [218](#)
 - RXOKBytesCount, [218](#)
 - TXOKBytesCount, [218](#)
- Data
 - CatEventIDDDataTlv, [143](#)
- data
 - authenticationData, [121](#)
 - SMSCAddress, [808](#)
 - sMSCAddress, [807](#)
 - SMSEtwsMessage, [809](#)
 - sMSEtwsMessage, [809](#)
 - SMSTransferRouteMTMessage, [822](#)
 - sMSTransferRouteMTMessage, [821](#)
 - SwiOTAMsg_s, [837](#)
- data_buf
 - NASOTAMessageTlv, [512](#)
- data_len
 - NASOTAMessageTlv, [512](#)
 - SwiOTAMsg_s, [837](#)
- dataBearer
 - pack_wds_SLQSSetWdsEventCallback_t, [635](#)
- dataBearerMask
 - dataBearers, [193](#)
 - unpack_wds_SLQSGetDataBearerTechnology_t, [1075](#)
- DataBearerTech, [193](#)
 - DUNCallInfoInd, [218](#)
 - ratValue, [195](#)
 - soMask, [195](#)
 - techType, [195](#)
- dataBearerTech
 - unpack_wds_SLQSGetDUNCallInfo_t, [1076](#)
- DataBearerTechExt, [195](#)
 - pBearerTech, [196](#)
 - pLastBearerTech, [196](#)
- dataBearerTechnology, [196](#)
 - currentNetwork, [197](#)
 - ratMask, [197](#)
 - soMask, [197](#)
- dataBearers, [192](#)
 - dataBearerMask, [193](#)
 - pCurDataBearerTechnology, [193](#)

- pLastCallDataBearerTechnology, [193](#)
- dataCapabilities
 - dataSrvCapabilities, [199](#)
 - nas_dataSrvCapabilities, [420](#)
- dataCapabilitiesLen
 - dataSrvCapabilities, [199](#)
 - nas_dataSrvCapabilities, [420](#)
- DataCaps
 - unpack_nas_GetServingNetwork_t, [985](#)
 - unpack_nas_GetServingNetworkCapabilities_↔
t, [986](#)
- dataCaps
 - unpack_nas_SetDataCapabilitiesCallback_ind_t,
[988](#)
- DataCapsLen
 - unpack_nas_GetServingNetwork_t, [985](#)
 - unpack_nas_GetServingNetworkCapabilities_↔
t, [986](#)
- dataCapsSize
 - unpack_nas_SetDataCapabilitiesCallback_ind_t,
[988](#)
- dataLen
 - authenticationData, [121](#)
- DataLength
 - CatEventIDDDataTlv, [143](#)
- DataRate
 - unpack_qos_swiQosFlow_t, [1037](#)
- dataRate, [197](#)
 - dataRateMax, [198](#)
 - guaranteedRate, [198](#)
- dataRateMax
 - dataRate, [198](#)
 - unpack_qos_dataRate_t, [1019](#)
- dataServiceCaCapability
 - unpack_dms_GetDeviceCapabilities_t, [929](#)
- DataServiceCapability
 - unpack_dms_GetDeviceCap_t, [928](#)
- DataSrvCapabilities
 - qaQmiServingSystemParam, [682](#)
 - unpack_nas_SLQSGetServingSystem_t, [995](#)
- dataSrvCapabilities, [198](#)
 - dataCapabilities, [199](#)
 - dataCapabilitiesLen, [199](#)
- DataStatusDetail, [199](#)
 - IPAddress, [200](#)
 - LastErrCode, [200](#)
- dataSysStatAvail
 - unpack_wds_SLQSSetWdsEventCallback_ind_t,
[1082](#)
- dataSystemStatus
 - pack_wds_SLQSSetWdsEventCallback_t, [635](#)
- DataULongLongTlv, [201](#)
 - TlvPresent, [201](#)
 - ulldata, [201](#)
- DataULongTlv, [201](#)
 - TlvPresent, [201](#)
 - ulldata, [201](#)
- Date
 - unpack_swioma_SLQSOMADMGetSessionInfo_↔
t, [1050](#)
 - wcdmaLongMsgDecodingParams, [1151](#)
 - wcdmaMsgDecodingParams, [1152](#)
- DateLength
 - unpack_swioma_SLQSOMADMGetSessionInfo_↔
t, [1050](#)
- day
 - nas_UniversalTime, [487](#)
 - nas_timeInfo, [482](#)
 - timeInfo, [862](#)
 - UniversalTime, [925](#)
- dayLtSavingAdj
 - nas_timeInfo, [482](#)
 - timeInfo, [862](#)
- dayOfWeek
 - nas_UniversalTime, [487](#)
 - nas_timeInfo, [482](#)
 - timeInfo, [862](#)
 - UniversalTime, [925](#)
- daylightSavings
 - nas_qaQmi3Gpp2TimeZone, [462](#)
 - qaQmi3Gpp2TimeZone, [678](#)
- DcsUsbPortNames, [201](#)
 - AtCmdPort, [201](#)
 - DmPort, [201](#)
 - NmeaPort, [201](#)
- defaultPDNEnabled
 - unpack_wds_SLQSGet3GPPConfigItem_t, [1073](#)
- DefaultRoamInd
 - unpack_nas_SLQSGetServingSystem_t, [995](#)
- defaultRoamInd
 - qaQmiServingSystemParam, [682](#)
- delAssistDataStatus, [201](#)
 - status, [202](#)
- delayClass
 - GPRSQoS, [272](#)
 - GPRSRequestedQoS, [273](#)
 - LibPackGPRSRequestedQoS, [321](#)
 - wds_GPRSQoS, [1166](#)
- DeleteStoredImage
 - qaGobiApiFms.h, [1462](#)
- deliveryErrSDU
 - LibPackUMTSQoS, [349](#)
 - UMTSMinQoS, [919](#)
 - UMTSQoS, [922](#)
 - wds_UMTSMinQoS, [1173](#)
- depersonalizationInformation, [202](#)
 - ckLen, [203](#)
 - ckVal, [203](#)
 - feature, [203](#)
 - operation, [203](#)
- depersonalisationInfo
 - UIMDepersonalizationReq, [893](#)
- Description
 - SlqsNas3GppNetworkInfo, [792](#)
- description
 - omaDmFotaTlv, [547](#)

- omaDmFotaTlvExt, [549](#)
- unpack_omaDmFotaTlv_t, [1017](#)
- descriptionlength
 - omaDmFotaTlv, [547](#)
 - omaDmFotaTlvExt, [549](#)
 - unpack_omaDmFotaTlv_t, [1017](#)
- Description
 - nas_QmiNas3GppNetworkInfo, [462](#)
- destPortRangeEnd
 - LibPackTFTIDParams, [347](#)
 - TFTIDParams, [860](#)
- destPortRangeStart
 - LibPackTFTIDParams, [347](#)
 - TFTIDParams, [860](#)
- detailSvcInfo, [203](#)
 - hdrHybrid, [204](#)
 - hdrSrvStatus, [205](#)
 - isSysForbidden, [205](#)
 - srvCapability, [205](#)
 - srvStatus, [205](#)
- DetailedSvcInfo
 - qaQmiServingSystemParam, [682](#)
 - unpack_nas_SLQSGetServingSystem_t, [995](#)
- dev
 - qmifwinfo_s, [706](#)
- DevCrashState
 - unpack_dms_GetCrashAction_t, [927](#)
- Device
 - SetM2MAudioAVCFGRReq, [766](#)
- Device Connectivity Service (DCS), [33](#)
- Device Management Service (DMS), [35](#)
- device_state_enum
 - qaGobiApiCbk.h, [1364](#)
- DeviceConfigDetail, [205](#)
 - Chipset, [206](#)
 - HWVersion, [206](#)
 - QLIC, [206](#)
 - Technology, [206](#)
- deviceId
 - _MitigationDevInfo, [62](#)
- deviceIdLen
 - _MitigationDevInfo, [62](#)
- DiagInfo
 - allCallsDiagInfo, [102](#)
 - arrDiagInfo, [117](#)
- diagInfo, [207](#)
 - diagInfoLen, [208](#)
 - diagnosticInfo, [208](#)
- diagInfoLen
 - diagInfo, [208](#)
- diagnosticInfo
 - diagInfo, [208](#)
- digitBuff
 - DTMFInfo, [216](#)
- digitCnt
 - burstDTMFInfo, [125](#)
 - DTMFInfo, [216](#)
- dirNum, [208](#)
- dirNum, [208](#)
- dirNumLen, [208](#)
- dirNumLen
 - dirNum, [208](#)
- direction
 - callInfo, [136](#)
- DisableIMSI
 - pack_dms_SetCustFeature_t, [554](#)
 - unpack_dms_GetCustFeature_t, [927](#)
- dispType
 - extDispRecInfo, [225](#)
- displayCondition
 - serviceProviderName, [749](#)
- dl_bw_value
 - NASPhyCaAggPcellInfo, [513](#)
 - NASPhyCaAggScellIDBw, [514](#)
 - NASPhyCaAggScellInfo, [516](#)
 - nas_PhyCaAggPcellInfo, [456](#)
 - nas_PhyCaAggScellIDBw, [457](#)
 - nas_PhyCaAggScellInfo, [461](#)
 - PhyCaAggPcellInfo, [650](#)
 - PhyCaAggScellIDBw, [651](#)
 - PhyCaAggScellInfo, [655](#)
- DmPort
 - DcsUsbPortNames, [201](#)
- dms.h, [1200](#)
 - ACT_CODE_MAX_SIZE, [1206](#)
 - CK_MAX_SIZE, [1206](#)
 - DMS_IMGDETAILS_LEN, [1206](#)
 - DMS_MAX_CUST_ID_LEN, [1206](#)
 - DMS_MAX_CUST_VALUE_LEN, [1206](#)
 - DMS_MAX_FWUPDATE_LOG_STR_SZ, [1206](#)
 - DMS_MAX_FWUPDATE_REF_STR_SZ, [1207](#)
 - DMS_PM_FACTORY, [1207](#)
 - DMS_PM_LOW, [1207](#)
 - DMS_PM_OFFLINE, [1207](#)
 - DMS_PM_ONLINE, [1207](#)
 - DMS_PM_PERSISTENT_LOW, [1207](#)
 - DMS_PM_RESET, [1207](#)
 - DMS_PM_SHUT_DOWN, [1207](#)
 - DMS_SET_REPORT_DISABLE, [1207](#)
 - DMS_SET_REPORT_ENABLE, [1207](#)
 - DMS_SLQSFWINFO_APPVERSION_SZ, [1207](#)
 - DMS_SLQSFWINFO_BOOTVERSION_SZ, [1207](#)
 - DMS_SLQSFWINFO_CARRIER_SZ, [1207](#)
 - DMS_SLQSFWINFO_CUR_CARR_NAME, [1207](#)
 - DMS_SLQSFWINFO_CUR_CARR_REV, [1207](#)
 - DMS_SLQSFWINFO_MODELID_SZ, [1207](#)
 - DMS_SLQSFWINFO_PACKAGEID_SZ, [1207](#)
 - DMS_SLQSFWINFO_PRIVERSION_SZ, [1207](#)
 - DMS_SLQSFWINFO_SKU_SZ, [1207](#)
 - DMS_SWI_SET_IND_DISABLE, [1207](#)
 - DMS_SWI_SET_IND_ENABLE, [1207](#)
 - DMS_UINT8_MAX_STRING_SZ, [1207](#)
 - ERI_DATA_MAX_SIZE, [1207](#)
 - MAX_BUILD_ID_LEN, [1208](#)
 - MEID_MAX_SIZE, [1208](#)
 - pack_dms_ActivateAutomatic, [1208](#)

- pack_dms_GetActivationState, 1208
- pack_dms_GetBandCapability, 1209
- pack_dms_GetCrashAction, 1209
- pack_dms_GetCustFeature, 1209
- pack_dms_GetCustFeaturesV2, 1210
- pack_dms_GetDeviceCap, 1210
- pack_dms_GetDeviceCapabilities, 1210
- pack_dms_GetDeviceHardwareRev, 1211
- pack_dms_GetDeviceMfr, 1211
- pack_dms_GetDeviceSerialNumbers, 1212
- pack_dms_GetFSN, 1213
- pack_dms_GetFirmwareInfo, 1212
- pack_dms_GetFirmwareRevision, 1212
- pack_dms_GetFirmwareRevisions, 1213
- pack_dms_GetHardwareRevision, 1214
- pack_dms_GetIMSI, 1214
- pack_dms_GetManufacturer, 1214
- pack_dms_GetModelID, 1215
- pack_dms_GetNetworkTime, 1215
- pack_dms_GetOfflineReason, 1216
- pack_dms_GetPRLVersion, 1216
- pack_dms_GetPower, 1216
- pack_dms_GetSerialNumbers, 1217
- pack_dms_GetUSBComp, 1217
- pack_dms_GetVoiceNumber, 1218
- pack_dms_ResetToFactoryDefaults, 1218
- pack_dms_SLQSDmsSwiGetResetInfo, 1222
- pack_dms_SLQSDmsSwiIndicationRegister, 1222
- pack_dms_SLQSGetBandCapability, 1223
- pack_dms_SLQSGetERIFile, 1223
- pack_dms_SLQSSwiClearDyingGaspStatistics, 1224
- pack_dms_SLQSSwiGetCrashInfo, 1224
- pack_dms_SLQSSwiGetDyingGaspCfg, 1224
- pack_dms_SLQSSwiGetDyingGaspStatistics, 1225
- pack_dms_SLQSSwiGetFirmwareCurr, 1225
- pack_dms_SLQSSwiGetFwUpdateStatus, 1226
- pack_dms_SLQSSwiGetHostDevInfo, 1226
- pack_dms_SLQSSwiGetOSInfo, 1226
- pack_dms_SLQSSwiGetSerialNoExt, 1227
- pack_dms_SLQSSwiSetDyingGaspCfg, 1227
- pack_dms_SLQSSwiSetHostDevInfo, 1228
- pack_dms_SLQSSwiSetOSInfo, 1228
- pack_dms_SLQSUIMGetState, 1228
- pack_dms_SetActivationStatusCallback, 1218
- pack_dms_SetCrashAction, 1219
- pack_dms_SetCustFeature, 1219
- pack_dms_SetCustFeaturesV2, 1220
- pack_dms_SetEventReport, 1220
- pack_dms_SetFirmwarePreference, 1221
- pack_dms_SetPower, 1221
- pack_dms_SetUSBComp, 1222
- pack_dms_UIMChangePIN, 1229
- pack_dms_UIMGetControlKeyStatus, 1229
- pack_dms_UIMGetICCID, 1230
- pack_dms_UIMGetPINStatus, 1230
- pack_dms_UIMSetControlKeyProtection, 1230
- pack_dms_UIMSetPINProtection, 1231
- pack_dms_UIMUnblockControlKey, 1231
- pack_dms_UIMUnblockPIN, 1232
- pack_dms_UIMVerifyPIN, 1232
- pack_dms_ValidateSPC, 1232
- SLQS_MAX_DYING_GASP_CFG_SMS_CONT↔
ENT_LENGTH, 1208
- SLQS_MAX_DYING_GASP_CFG_SMS_NUMB↔
ER_LENGTH, 1208
- SPC_SIZE, 1208
- UNIQUE_ID_LEN, 1208
- unpack_dms_ActivateAutomatic, 1233
- unpack_dms_GetActivationState, 1233
- unpack_dms_GetBandCapability, 1234
- unpack_dms_GetCrashAction, 1234
- unpack_dms_GetCustFeature, 1234
- unpack_dms_GetCustFeaturesV2, 1235
- unpack_dms_GetDeviceCap, 1235
- unpack_dms_GetDeviceCapabilities, 1235
- unpack_dms_GetDeviceHardwareRev, 1236
- unpack_dms_GetDeviceMfr, 1236
- unpack_dms_GetDeviceSerialNumbers, 1237
- unpack_dms_GetFSN, 1238
- unpack_dms_GetFirmwareInfo, 1237
- unpack_dms_GetFirmwareRevision, 1237
- unpack_dms_GetFirmwareRevisions, 1238
- unpack_dms_GetHardwareRevision, 1239
- unpack_dms_GetIMSI, 1239
- unpack_dms_GetManufacturer, 1239
- unpack_dms_GetModelID, 1240
- unpack_dms_GetNetworkTime, 1240
- unpack_dms_GetOfflineReason, 1241
- unpack_dms_GetPRLVersion, 1241
- unpack_dms_GetPower, 1241
- unpack_dms_GetSerialNumbers, 1242
- unpack_dms_GetUSBComp, 1242
- unpack_dms_GetVoiceNumber, 1243
- unpack_dms_ResetToFactoryDefaults, 1243
- unpack_dms_SLQSDmsSwiGetResetInfo, 1247
- unpack_dms_SLQSDmsSwiGetResetInfo_Ind, 1247
- unpack_dms_SLQSDmsSwiIndicationRegister, 1248
- unpack_dms_SLQSGetBandCapability, 1248
- unpack_dms_SLQSGetERIFile, 1249
- unpack_dms_SLQSSwiClearDyingGaspStatistics, 1249
- unpack_dms_SLQSSwiGetCrashInfo, 1250
- unpack_dms_SLQSSwiGetDyingGaspCfg, 1250
- unpack_dms_SLQSSwiGetDyingGaspStatistics, 1250
- unpack_dms_SLQSSwiGetFirmwareCurr, 1251
- unpack_dms_SLQSSwiGetFwUpdateStatus, 1251
- unpack_dms_SLQSSwiGetHostDevInfo, 1252
- unpack_dms_SLQSSwiGetOSInfo, 1252
- unpack_dms_SLQSSwiGetSerialNoExt, 1252
- unpack_dms_SLQSSwiSetDyingGaspCfg, 1253
- unpack_dms_SLQSSwiSetHostDevInfo, 1253

- unpack_dms_SLQSSwiSetOSInfo, [1254](#)
- unpack_dms_SLQSUIMGetState, [1254](#)
- unpack_dms_SetActivationStatusCallback, [1243](#)
- unpack_dms_SetCrashAction, [1244](#)
- unpack_dms_SetCustFeature, [1244](#)
- unpack_dms_SetCustFeaturesV2, [1245](#)
- unpack_dms_SetEventReport, [1245](#)
- unpack_dms_SetEventReport_ind, [1245](#)
- unpack_dms_SetFirmwarePreference, [1246](#)
- unpack_dms_SetPower, [1246](#)
- unpack_dms_SetUSBComp, [1247](#)
- unpack_dms_UIMChangePIN, [1254](#)
- unpack_dms_UIMGetControlKeyStatus, [1255](#)
- unpack_dms_UIMGetICCID, [1255](#)
- unpack_dms_UIMGetPINStatus, [1256](#)
- unpack_dms_UIMSetControlKeyProtection, [1256](#)
- unpack_dms_UIMSetPINProtection, [1256](#)
- unpack_dms_UIMUnblockControlKey, [1257](#)
- unpack_dms_UIMUnblockPIN, [1257](#)
- unpack_dms_UIMVerifyPIN, [1258](#)
- unpack_dms_ValidateSPC, [1258](#)
- dms_ActivationStatusTlv, [208](#)
 - activationStatus, [209](#)
 - TlvPresent, [209](#)
- dms_OperatingModeTlv, [209](#)
 - operatingMode, [210](#)
 - TlvPresent, [210](#)
- dmsCurrentPRLInfo, [210](#)
 - pPRLPreference, [210](#)
 - pPRLVersion, [210](#)
 - qaGobiApiDms.h, [1420](#)
- dmsIndicationRegisterReq, [213](#)
 - pSwiGetResetInd, [213](#)
- dmsSwiGetResetInfo, [213](#)
 - source, [214](#)
 - type, [214](#)
- Domain, [214](#)
 - domainLen, [214](#)
 - domainName, [214](#)
- domain
 - DomainNameList, [215](#)
 - wds_DomainNameList, [1166](#)
- domainLen
 - Domain, [214](#)
 - wds_Domain, [1165](#)
- DomainList
 - unpack_wds_SLQSGetRuntimeSettings_t, [1078](#)
- domainName
 - Domain, [214](#)
 - wds_Domain, [1165](#)
- DomainNameList, [215](#)
 - domain, [215](#)
 - numInstances, [215](#)
- dormancyStatAvail
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- dormancyState
 - unpack_wds_GetDormancyState_t, [1065](#)
- DormancyStatus
 - DUNCallInfoInd, [218](#)
- dormancyStatus
 - pack_wds_SLQSSetWdsEventCallback_t, [635](#)
 - unpack_wds_SLQSGetDUNCallInfo_t, [1076](#)
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- downLink
 - NSSAudioCtrl, [541](#)
- DownloadToSlot
 - qaGobiApiFms.h, [1462](#)
- dscp
 - QosMap, [719](#)
- dtmSupp
 - GSMSysInfo, [283](#)
 - nas_GSMSysInfo, [431](#)
- dtmSuppValid
 - GSMSysInfo, [283](#)
 - nas_GSMSysInfo, [431](#)
- dunchannelRate, [218](#)
 - CurrChanRxRate, [218](#)
 - CurrChanTxRate, [218](#)
 - MaxChanRxRate, [219](#)
 - MaxChanTxRate, [219](#)
- Duration
 - pack_nas_SetNetworkPreference_t, [584](#)
 - unpack_nas_GetNetworkPreference_t, [984](#)
- eCAT
 - common.h, [1198](#)
- ECIOThresListLen
 - ECIOThresh, [220](#)
- ECIOThresh, [219](#)
 - ECIOThresListLen, [220](#)
 - pECIOThresList, [220](#)
- ECTCallState
 - ECTNum, [221](#)
- ECTNum, [220](#)
 - ECTCallState, [221](#)
 - number, [221](#)
 - presentationInd, [221](#)
- eCTL
 - common.h, [1198](#)
- eDMS
 - common.h, [1198](#)
- eDevState
 - qaGobiApiCbk.h, [1329](#)
- eDevice
 - sGetDeviceSeriesResult, [780](#)
- eGOBI_DEV_SERIES_9X15
 - qaGobiApiFms.h, [1460](#)
- eGOBI_DEV_SERIES_9X30
 - qaGobiApiFms.h, [1460](#)
- eGOBI_DEV_SERIES_G3K
 - qaGobiApiFms.h, [1460](#)
- eGOBI_DEV_SERIES_NON_GOBI
 - qaGobiApiFms.h, [1460](#)
- eGOBI_DEV_SERIES_SIERRA_GOBI
 - qaGobiApiFms.h, [1460](#)

eGOBI_DEV_SERIES_UNKNOWN
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_3
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_AERIS
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_ALLTEL
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_AMX_TELCEL
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_ATT
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_BELL
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_BHARTI
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_BRASIL_VIVO
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_CHINA_MOBILE
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_CHINA_TELECOM
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_CHINA_UNICOM
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_EMOBILE
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_FACTORY
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_GENERIC_CDMA
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_GENERIC
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_IUSACELL
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_KDDI
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_KT_FREETEL
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_LEAP
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_METROPCS
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_NETCOM
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_NORF
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_NTT_DOCOMO
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_O2
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_OMH
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_ORANGE
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_RELIANCE1
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_RELIANCE2
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_ROGERS
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_SFR
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_SINGTEL_OPTUS
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_SK_TELCOM1
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_SK_TELCOM2
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_SOFTBANK
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_SPRINT
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_SWISSCOM
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_TATA
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_TELCOM_ITALIA
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_TELCOM_NZ
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_TELEFONICA
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_TELNOR
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_TELIASONERA
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_TELSTRA1
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_TELSTRA2
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_TELUS
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_TMOBILE
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_CAR_US
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_VERIZON
 qaGobiApiFms.h, [1460](#)
eGOBI_IMG_CAR_VODAFONE
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_GPS_ASSISTED
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_GPS_NO_XTRA
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_GPS_NONE
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_GPS_STAND_ALONE
 qaGobiApiFms.h, [1461](#)
eGOBI_IMG_REG_ASIA
 qaGobiApiFms.h, [1462](#)
eGOBI_IMG_REG_AUS
 qaGobiApiFms.h, [1462](#)
eGOBI_IMG_REG_EU
 qaGobiApiFms.h, [1462](#)
eGOBI_IMG_REG_GLOBAL
 qaGobiApiFms.h, [1462](#)

- eGOBI_IMG_REG_LA
 - qaGobiApiFms.h, [1462](#)
- eGOBI_IMG_REG_NA
 - qaGobiApiFms.h, [1462](#)
- eGOBI_IMG_TECH_CDMA
 - qaGobiApiFms.h, [1462](#)
- eGOBI_IMG_TECH_UMTS
 - qaGobiApiFms.h, [1462](#)
- eGetDeviceSeries
 - qaGobiApiFms.h, [1463](#)
- eGobi_DEV_SERIES_MC83
 - qaGobiApiFms.h, [1460](#)
- eGobiDeviceSeries
 - qaGobiApiFms.h, [1460](#)
- eGobiImageCarrier
 - qaGobiApiFms.h, [1460](#)
- eGobiImageGPS
 - qaGobiApiFms.h, [1461](#)
- eGobiImageRegion
 - qaGobiApiFms.h, [1461](#)
- eGobiImageTech
 - qaGobiApiFms.h, [1462](#)
- eIND
 - common.h, [1198](#)
- eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_100
 - nas.h, [1285](#)
- eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_15
 - nas.h, [1285](#)
- eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_25
 - nas.h, [1285](#)
- eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_50
 - nas.h, [1285](#)
- eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_6
 - nas.h, [1285](#)
- eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_75
 - nas.h, [1285](#)
- eLIBPACK_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_↔
 - FIGURED_ACTIVATED
 - nas.h, [1285](#)
- eLIBPACK_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_↔
 - FIGURED_DEACTIVATED
 - nas.h, [1285](#)
- eLIBPACK_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_↔
 - ONFIGURED
 - nas.h, [1285](#)
- eLOG_DEBUG
 - common.h, [1197](#)
- eLOG_FATAL
 - common.h, [1197](#)
- eLOG_INFO
 - common.h, [1197](#)
- eLOG_LEVEL
 - common.h, [1197](#)
- eLOG_WARN
 - common.h, [1197](#)
- eLOC
 - common.h, [1198](#)
- EMTlv
 - NASQmiCbkNasSystemSelPrefInd, [521](#)
- eNAS_LTE_CPHY_CA_BW_NRB_100
 - qaGobiApiNas.h, [1512](#)
- eNAS_LTE_CPHY_CA_BW_NRB_15
 - qaGobiApiNas.h, [1512](#)
- eNAS_LTE_CPHY_CA_BW_NRB_25
 - qaGobiApiNas.h, [1512](#)
- eNAS_LTE_CPHY_CA_BW_NRB_50
 - qaGobiApiNas.h, [1512](#)
- eNAS_LTE_CPHY_CA_BW_NRB_6
 - qaGobiApiNas.h, [1512](#)
- eNAS_LTE_CPHY_CA_BW_NRB_75
 - qaGobiApiNas.h, [1512](#)
- eNAS_LTE_CPHY_CA_BW_NRB_LITE_100
 - nas.h, [1285](#)
- eNAS_LTE_CPHY_CA_BW_NRB_LITE_15
 - nas.h, [1285](#)
- eNAS_LTE_CPHY_CA_BW_NRB_LITE_25
 - nas.h, [1285](#)
- eNAS_LTE_CPHY_CA_BW_NRB_LITE_50
 - nas.h, [1285](#)
- eNAS_LTE_CPHY_CA_BW_NRB_LITE_6
 - nas.h, [1285](#)
- eNAS_LTE_CPHY_CA_BW_NRB_LITE_75
 - nas.h, [1285](#)
- eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_↔
 - ACTIVATED_LITE
 - nas.h, [1285](#)
- eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_↔
 - ACTIVATED
 - qaGobiApiNas.h, [1512](#)
- eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_↔
 - DEACTIVATED_LITE
 - nas.h, [1285](#)
- eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_↔
 - DEACTIVATED
 - qaGobiApiNas.h, [1512](#)
- eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_↔
 - ED_LITE
 - nas.h, [1285](#)
- eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED
 - qaGobiApiNas.h, [1512](#)
- eNAS_RADIO_IF_GSM
 - qaGobiApiNas.h, [1511](#)
- eNAS_RADIO_IF_LTE
 - qaGobiApiNas.h, [1511](#)
- eNAS_RADIO_IF_TDSCDMA
 - qaGobiApiNas.h, [1511](#)
- eNAS_RADIO_IF_UMTS
 - qaGobiApiNas.h, [1511](#)
- eNAS
 - common.h, [1198](#)
- eQA_QMI_SVC_NAS
 - qaGobiApiCbk.h, [1364](#)
- eQA_QMI_SVC_NA
 - qaGobiApiCbk.h, [1364](#)
- eQA_QMI_SVC_WDS
 - qaGobiApiCbk.h, [1364](#)

eQCWWAN_ERR_API_MUTEX_TIMEOUT
 qmerrno.h, 1745
 eQCWWAN_ERR_BUFFER_SZ
 qmerrno.h, 1744
 eQCWWAN_ERR_CANCEL_OP
 qmerrno.h, 1745
 eQCWWAN_ERR_DRIVER
 qmerrno.h, 1745
 eQCWWAN_ERR_ENUM_BEGIN
 qmerrno.h, 1744
 eQCWWAN_ERR_ENUM_END
 qmerrno.h, 1745
 eQCWWAN_ERR_FILE_COPY
 qmerrno.h, 1745
 eQCWWAN_ERR_FILE_OPEN
 qmerrno.h, 1745
 eQCWWAN_ERR_GENERAL
 qmerrno.h, 1744
 eQCWWAN_ERR_INTERNAL
 qmerrno.h, 1744
 eQCWWAN_ERR_INVALID_ARG
 qmerrno.h, 1744
 eQCWWAN_ERR_INVALID_DEVID
 qmerrno.h, 1744
 eQCWWAN_ERR_INVALID_FILE
 qmerrno.h, 1744
 eQCWWAN_ERR_INVALID_QMI_RSP
 qmerrno.h, 1744
 eQCWWAN_ERR_INVALID_XID
 qmerrno.h, 1745
 eQCWWAN_ERR_MALFORMED_QMI_RSP
 qmerrno.h, 1744
 eQCWWAN_ERR_MEMORY
 qmerrno.h, 1744
 eQCWWAN_ERR_MULTIPLE_DEVICES
 qmerrno.h, 1745
 eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED
 qmerrno.h, 1745
 eQCWWAN_ERR_NO_CANCELABLE_OP
 qmerrno.h, 1745
 eQCWWAN_ERR_NO_CONNECTION
 qmerrno.h, 1744
 eQCWWAN_ERR_NO_DEVICE
 qmerrno.h, 1744
 eQCWWAN_ERR_NO_SIGNAL
 qmerrno.h, 1745
 eQCWWAN_ERR_NONE
 qmerrno.h, 1744
 eQCWWAN_ERR_NULL_TLV
 qmerrno.h, 1748
 eQCWWAN_ERR_OFFLINE
 qmerrno.h, 1745
 eQCWWAN_ERR_PDU_GENERATION
 qmerrno.h, 1745
 eQCWWAN_ERR_QMI_ABORTED
 qmerrno.h, 1745
 eQCWWAN_ERR_QMI_ACCESS_DENIED
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_ACK_NOT_SENT
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_ARG_TOO_LONG
 qmerrno.h, 1745
 eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_CALL_FAILED
 qmerrno.h, 1745
 eQCWWAN_ERR_QMI_CARD_BUSY_RSP
 qmerrno.h, 1748
 eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_CAT_END
 qmerrno.h, 1748
 eQCWWAN_ERR_QMI_CAT_START
 qmerrno.h, 1748
 eQCWWAN_ERR_QMI_CAUSE_CODE
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED
 qmerrno.h, 1745
 eQCWWAN_ERR_QMI_CONNECT
 qmerrno.h, 1744
 eQCWWAN_ERR_QMI_DEVICE_IN_USE
 qmerrno.h, 1745
 eQCWWAN_ERR_QMI_DEVICE_NOT_READY
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_DISABLED
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_ENCODING
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE
 qmerrno.h, 1748
 eQCWWAN_ERR_QMI_EVENT_REG_FAILED
 qmerrno.h, 1748
 eQCWWAN_ERR_QMI_EXTENDED_INTERNAL
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_FDN_RESTRICT
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_FLOW_SUSPENDED
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_GENERAL
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_IFACE
 qmerrno.h, 1744
 eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER
 qmerrno.h, 1746

- eQCWWAN_ERR_QMI_INCORRECT_PIN
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_INFO_UNAVAILABLE
qmerrno.h, [1747](#)
- eQCWWAN_ERR_QMI_INJECT_TIMEOUT
qmerrno.h, [1747](#)
- eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCE
qmerrno.h, [1747](#)
- eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INTERNAL
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_INVALID_ARG
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_CLIENT_ID
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD
qmerrno.h, [1748](#)
- eQCWWAN_ERR_QMI_INVALID_HANDLE
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_INVALID_INDEX
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_ID
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_OPERATION
qmerrno.h, [1747](#)
- eQCWWAN_ERR_QMI_INVALID_PDP_TYPE
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_INVALID_PINID
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_PROFILE
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_QMI_CMD
qmerrno.h, [1747](#)
- eQCWWAN_ERR_QMI_INVALID_QOS_ID
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_TECH_PREF
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP
qmerrno.h, [1748](#)
- eQCWWAN_ERR_QMI_INVALID_TRANSITION
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_INVALID_TX_ID
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_MALFORMED_MSG
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_MAX
qmerrno.h, [1747](#)
- eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_MISSING_ARG
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_MSG_BLOCKED
qmerrno.h, [1747](#)
- eQCWWAN_ERR_QMI_NETWORK_ABORTED
qmerrno.h, [1747](#)
- eQCWWAN_ERR_QMI_NETWORK_NOT_READY
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_NO_EFFECT
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_NO_ENTRY
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_NO_FREE_PROFILE
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_NO_MEMORY
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_NO_NETWORK_FOUND
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_NO_RADIO
qmerrno.h, [1747](#)
- eQCWWAN_ERR_QMI_NO_SUBSCRIPTION
qmerrno.h, [1747](#)
- eQCWWAN_ERR_QMI_NO_THRESHOLDS
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE
qmerrno.h, [1746](#)
- eQCWWAN_ERR_QMI_NOT_PROVISIONED
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_NOT_SUPPORTED
qmerrno.h, [1747](#)
- eQCWWAN_ERR_QMI_OFFSET
qmerrno.h, [1745](#)
- eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED
qmerrno.h, [1745](#)

eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPO↔
 RTED
 qmerrno.h, 1745
 eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_OUT_OF_CALL
 qmerrno.h, 1745
 eQCWWAN_ERR_QMI_PIN_BLOCKED
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_POLICY_MISMATCH
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_REQ_SCH
 qmerrno.h, 1744
 eQCWWAN_ERR_QMI_REQ_TO
 qmerrno.h, 1744
 eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSU↔
 PPORTED
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_REQ
 qmerrno.h, 1744
 eQCWWAN_ERR_QMI_RSP_TO
 qmerrno.h, 1744
 eQCWWAN_ERR_QMI_RSP
 qmerrno.h, 1744
 eQCWWAN_ERR_QMI_SEGMENT_ORDER
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_SESSION_INACTIVE
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_SESSION_INVALID
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_SESSION_OWNERSHIP
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_SMSC_ADDR
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_TPDU_TYPE
 qmerrno.h, 1747
 eQCWWAN_ERR_QMI_UNABORTABLE_TRANSA↔
 CTION
 qmerrno.h, 1745
 eQCWWAN_ERR_QMI_UNKNOWN
 qmerrno.h, 1746
 eQCWWAN_ERR_QMI_WIDTH
 qmerrno.h, 1748
 eQCWWAN_ERR_RESET
 qmerrno.h, 1745
 eQCWWAN_ERR_SWICM_AM_VERS_ERROR
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_END
 qmerrno.h, 1748
 eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK↔
 _PROCESS
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_INVALID_SESSION_ID
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_INVALID_V4_SESSION↔
 _ID
 qmerrno.h, 1748
 eQCWWAN_ERR_SWICM_INVALID_V6_SESSION↔
 _ID
 qmerrno.h, 1748
 eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPP↔
 ORTED
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPP↔
 ORTED
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_S↔
 SESSIONS
 qmerrno.h, 1748
 eQCWWAN_ERR_SWICM_SOCKET_IN_USE
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_START
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_TIMEOUT
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_V4DWN_V6DWN
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_V4DWN_V6UP
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_V4UP_V6DWN
 qmerrno.h, 1747
 eQCWWAN_ERR_SWICM_V4UP_V6UP
 qmerrno.h, 1747
 eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED
 qmerrno.h, 1748
 eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOU↔
 ND
 qmerrno.h, 1748
 eQCWWAN_ERR_SWIDCS_END
 qmerrno.h, 1748
 eQCWWAN_ERR_SWIDCS_FILEIO_ERR
 qmerrno.h, 1748
 eQCWWAN_ERR_SWIDCS_IOCTL_ERR
 qmerrno.h, 1748
 eQCWWAN_ERR_SWIDCS_START
 qmerrno.h, 1748
 eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE
 qmerrno.h, 1748
 eQCWWAN_ERR_SWIIM_END
 qmerrno.h, 1748
 eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND
 qmerrno.h, 1748

- eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADING
NLOADED
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOADING
D_MODE
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX
EX
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH
MATCH
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE
NT_ACTIVE_IMAGE
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_INVALID_PATH
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_OPENING_DIR
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_OPENING_FILE
qmerrno.h, 1748
- eQCWWAN_ERR_SWIIM_START
qmerrno.h, 1748
- eQCWWAN_ERR_SWISM_END
qmerrno.h, 1748
- eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND
FOUND
qmerrno.h, 1748
- eQCWWAN_ERR_SWISMS_MSG_CORRUPTED
qmerrno.h, 1748
- eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG
qmerrno.h, 1748
- eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED
PTED
qmerrno.h, 1748
- eQCWWAN_ERR_SWISMS_START
qmerrno.h, 1748
- eQCWWANError
qmerrno.h, 1744
- eQMI_CAT_EVENT_REPORT_IND_TLV_LENGTH
qaCbkCatEventReportInd.h, 1311
- eQMI_CAT_EVENT_REPORT_IND_TLV
qaCbkCatEventReportInd.h, 1311
- eQMI_LOC_SESS_STATUS_FAILURE
loc.h, 1267
- eQMI_LOC_SESS_STATUS_IN_PROGRESS
loc.h, 1267
- eQMI_LOC_SESS_STATUS_SUCCESS
loc.h, 1267
- eQMI_LOC_SESS_STATUS_TIMEOUT
loc.h, 1267
- eQMI_NAS_GET_RF_INFO_RESP
qaNasGetRFBandInfo.h, 1740
- eQMI_NAS_PERFORM_NETWORK_SCAN_RESP
qaNasPerformNetworkScan.h, 1741
- eQMI_SVC
common.h, 1197
- eQMI_SWIOMA_DM_EVENT_REPORT_IND
qaCbkSwiOmaDmEventReportInd.h, 1312
- eQMISARRFState
qaGobiApiSar.h, 1573
- eQOS
common.h, 1198
- eQaQMIService
qaGobiApiCbk.h, 1364
- eREQ
common.h, 1198
- ERI_DATA_MAX_SIZE
dms.h, 1207
- ERIFileparams, 222
pFile, 222
pFileSize, 222
qaGobiApiDms.h, 1420
- eRSP
common.h, 1198
- eSMSEventType
qaGobiApiCbk.h, 1329
- eSMS
common.h, 1198
- ESNString
unpack_dms_GetDeviceSerialNumbers_t, 930
- eSWILOC
common.h, 1198
- eSWIOMA
common.h, 1198
- eSYS_SRV_DOMAIN_CAMPED
qaGobiApiNas.h, 1511
- eSYS_SRV_DOMAIN_CS_ONLY
qaGobiApiNas.h, 1511
- eSYS_SRV_DOMAIN_CS_PS
qaGobiApiNas.h, 1511
- eSYS_SRV_DOMAIN_NO_SRV
qaGobiApiNas.h, 1511
- eSYS_SRV_DOMAIN_PS_ONLY
qaGobiApiNas.h, 1511
- eSYS_SRV_DOMAIN_UNKNOWN
qaGobiApiNas.h, 1511
- eSYS_SRV_DOMAIN
qaGobiApiNas.h, 1511
- eSetServiceAutomaticTrackingDisable
qaGobiApiPds.h, 1549
- eSetServiceAutomaticTrackingEnable
qaGobiApiPds.h, 1549
- eTIMEOUT_10_S
common.h, 1198
- eTIMEOUT_20_S

- common.h, 1198
- eTIMEOUT_2_S
 - common.h, 1198
- eTIMEOUT_300_S
 - common.h, 1198
- eTIMEOUT_30_S
 - common.h, 1198
- eTIMEOUT_5_S
 - common.h, 1198
- eTIMEOUT_60_S
 - common.h, 1198
- eTIMEOUT_8_S
 - common.h, 1198
- eTIMEOUT_DEFAULT
 - common.h, 1198
- eTLV_3GPP_NETWORK_INFO
 - qaNasPerformNetworkScan.h, 1741
- eTLV_CBK_ALPHA_IDENTIFIER
 - qaCbkCatEventReportInd.h, 1311
- eTLV_CBK_DISPLAY_TEXT
 - qaCbkCatEventReportInd.h, 1311
- eTLV_CBK_END_PROACTIVE_SESSION
 - qaCbkCatEventReportInd.h, 1311
- eTLV_CBK_GET_IN_KEY
 - qaCbkCatEventReportInd.h, 1311
- eTLV_CBK_GET_INPUT
 - qaCbkCatEventReportInd.h, 1311
- eTLV_CBK_LANGUAGE_NOTIFICATION
 - qaCbkCatEventReportInd.h, 1311
- eTLV_CBK_REFRESH
 - qaCbkCatEventReportInd.h, 1311
- eTLV_CBK_SELECT_ITEM
 - qaCbkCatEventReportInd.h, 1311
- eTLV_CBK_SETUP_EVENT_LIST
 - qaCbkCatEventReportInd.h, 1311
- eTLV_CBK_SETUP_IDLE_MODE_TEXT
 - qaCbkCatEventReportInd.h, 1311
- eTLV_CBK_SETUP_MENU
 - qaCbkCatEventReportInd.h, 1311
- eTLV_END_PROACTIVE_SESSION_LENGTH
 - qaCbkCatEventReportInd.h, 1311
- eTLV_IND_OMA_DM_CONFIG
 - qaCbkSwiOmaDmEventReportInd.h, 1312
- eTLV_IND_OMA_DM_FOTA
 - qaCbkSwiOmaDmEventReportInd.h, 1312
- eTLV_IND_OMA_DM_NOT
 - qaCbkSwiOmaDmEventReportInd.h, 1312
- eTLV_REFRESH_LENGTH
 - qaCbkCatEventReportInd.h, 1311
- eTLV_RF_BAND_INFO
 - qaNasGetRFBandInfo.h, 1740
- eTLV_SETUP_EVENT_LIST_LENGTH
 - qaCbkCatEventReportInd.h, 1311
- eTMD
 - common.h, 1198
- ETWSPLMNInfo
 - eTWSPLMNInfoTlv, 224
- eTWSPLMNInfoTlv, 224
- ETWSPLMNInfo, 224
 - TlvPresent, 224
- ETWSPLMNTlv
 - unpack_sms_SetNewSMSCallback_ind_t, 1043
- ETWSTlv
 - unpack_sms_SetNewSMSCallback_ind_t, 1043
- eTimeout
 - common.h, 1198
- eUIM
 - common.h, 1198
- EVENT_MASK_CARD
 - qaGobiApiCbk.h, 1326
- EVENT_MASK_DEREGISTER_ALL
 - qaGobiApiCbk.h, 1326
- EVENT_MASK_PHY_SLOT_STATUS
 - qaGobiApiCbk.h, 1326
- eValid
 - LibPackTFTIDParams, 347
 - TFTIDParams, 860
- eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_3GPP_INVALID_PROFILE_FAMILY
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_END
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_INVALID_PROFILE_FAMILY
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_HNDL
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_IDENT
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_OP
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_PROFILE_NUM
 - qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_PROFILE_TYPE
 - qmerrno.h, 1749

- qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID↔
_SUBS_ID
qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID↔
qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID↔
qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED↔
qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_RESULT_FAIL
qmerrno.h, 1749
- eWDS_ERR_PROFILE_REG_RESULT_LIST_END
qmerrno.h, 1749
- eWDS
common.h, 1198
- EarMute
 - GetAudioProfileResp, 244
 - GetM2MAVMuteResp, 264
 - GetM2MAudioProfileResp, 262
 - SetAudioProfileReq, 757
 - SetM2MAVMuteReq, 769
- earfcn
 - infoInterFreq, 316
 - LTEInfoIntrafreq, 384
 - ltePCI, 387
 - nas_LTEInfoIntrafreq, 443
 - nas_infoInterFreq, 437
 - nas_umtsLTENbrCell, 486
 - umtsLTENbrCell, 917
- earfcn0
 - lteEARFCN, 378
- earfcn1
 - lteEARFCN, 378
- ecio
 - CDMASSInfo, 154
 - cdmaSSInfo, 154
 - ecioListElement, 219
 - HDRSSInfo, 291
 - hdrSSInfo, 292
 - nas_UMTSInfo, 484
 - nas_ecioListElement, 422
 - rxInfo, 735
 - TDSCDMASigInfoExt, 855
 - tdscdmaSigInfoExt, 855
 - UMTSInfo, 915
- ecioDelta
 - nas_SLQSSignalStrengthsIndReq, 473
 - SLQSSignalStrengthsIndReq, 801
- ecioInfo
 - nas_SLQSSignalStrengthsInformation, 474
 - SLQSSignalStrengthsInformation, 803
- ecioList
 - slqsSignalStrengthInfo, 799
 - unpack_nas_SLQSGetSignalStrength_t, 997
- ecioListElement, 219
- ecio, 219
- radiolf, 219
- ecioListLen
 - slqsSignalStrengthInfo, 799
 - unpack_nas_SLQSGetSignalStrength_t, 997
- ecioThresholdList
 - nas_SLQSSignalStrengthsIndReq, 473
 - SLQSSignalStrengthsIndReq, 801
- ecioThresholdListLen
 - nas_SLQSSignalStrengthsIndReq, 473
 - SLQSSignalStrengthsIndReq, 801
- egprsSupp
 - GSMSysInfo, 283
 - nas_GSMSysInfo, 431
- egprsSuppValid
 - GSMSysInfo, 283
 - nas_GSMSysInfo, 431
- elevation
 - loc_satelliteInfo, 360
 - satelliteInfo, 742
- EmerMode
 - NASEmergencyModeTlv, 496
- emmConnState
 - LTEInfo, 381
 - nas_LTEInfo, 440
- emmState
 - LTEInfo, 381
 - nas_LTEInfo, 440
- emmSubState
 - LTEInfo, 381
 - nas_LTEInfo, 440
- Enable
 - SetM2MAudioLPBKReq, 767
- enable
 - pack_qos_SLQSSetQosEventCallback_t, 602
- enabled
 - unpack_wds_GetMobileIPProfile_t, 1066
- EncryptProt
 - protocolSubtypeElement, 676
- EncryptedPIN1
 - pack_uim_ChangePin_t, 612
 - pack_uim_SetPinProtection_t, 614
 - pack_uim_UnblockPin_t, 617
- encryptedPIN1, 221
 - pin1Len, 221
 - pin1Val, 222
- EndProactiveSession
 - CatEndProactiveSessionTlv, 142
- EngineState
 - GPSSStateInfo, 276
- engineState
 - QmiCbkLocEngineStateInd, 690
 - unpack_loc_EngineState_Ind_t, 970
- entries
 - t_Sv, 852
- eqmiCbkSetStatus
 - sms.h, 1762
- eriData

- eriDataparams, 222
- eriDataLen
 - eriDataparams, 222
- eriDataparams, 222
 - eriData, 222
 - eriDataLen, 222
- eriFile
 - unpack_dms_SLQSGetERIFile_t, 946
- error
 - unpack_wds_GetLastMobileIPError_t, 1065
- errorClass
 - SMSAsyncRawSend_s, 806
- errorRate
 - errorRateListElement, 223
 - nas_errorRateListElement, 423
- errorRateInfo
 - nas_SLQSSignalStrengthsInformation, 474
 - SLQSSignalStrengthsInformation, 803
- errorRateList
 - slqsSignalStrengthInfo, 799
 - unpack_nas_SLQSGetSignalStrength_t, 997
- errorRateListElement, 223
 - errorRate, 223
 - radioIf, 223
- errorRateListLen
 - slqsSignalStrengthInfo, 799
 - unpack_nas_SLQSGetSignalStrength_t, 997
- errorState
 - slotInf, 786
 - slotInfo, 787
 - uim_slotInfo, 887
- esn
 - unpack_dms_GetSerialNumbers_t, 937
- esnSize
 - serialNumbersInfo, 748
 - unpack_dms_GetDeviceSerialNumbers_t, 930
- EspSpi
 - unpack_qos_swiQosFilter_t, 1032
- EtwsMessageInfo
 - sMSEtwsMessageTlv, 810
- event
 - unpack_qos_SLQSSetQosPriEventCallback_ind↵_t, 1029
 - unpack_qos_SLQSSetQosStatusCallback_ind_t, 1030
- event_Index
 - QmiCbkCatEventStatusReportInd, 683
- EventID
 - CatCommonEventTlv, 142
- EventLength
 - CatCommonEventTlv, 142
- eventMask
 - CATEventDataType, 142
 - pack_uim_SLQSUIMEventRegister_t, 615
 - UIMEventRegisterReqResp, 894
 - unpack_uim_SLQSUIMEventRegister_t, 1057
- eventRegister
 - LOCEventRegisterReqResp, 368
- pack_loc_EventRegister_t, 570
- eventType
 - unpack_swioma_SLQSOMADMAAlertCallback↵_ind_t, 1048
- evrcCapability
 - prefVoiceSO, 663
- executingImage
 - FMSImageIDEntries, 233
 - ImageIDEntries, 298
- exponent
 - pktErrRate, 657
 - unpack_qos_pktErrRate_t, 1021
- extBit
 - calledPartySubAdd, 130
- extDisplInfo
 - extDispRecInfo, 225
- extDisplInfoLen
 - extDispRecInfo, 225
- extDispRecInfo, 224
 - dispType, 225
 - extDisplInfo, 225
 - extDisplInfoLen, 225
- ExtErrorCode
 - PackCreateProfileOut, 641
- extPowerState
 - LOCExtPowerStateReqResp, 368
 - pack_loc_SetExtPowerState_t, 570
- extendedErrorCode
 - unpack_wds_SLQSDelateProfile_t, 1071
- FIRMWARE_UPDATE_FAIL
 - qaGobiApiFms.h, 1459
- FIRMWARE_UPDATE_SUCCESS
 - qaGobiApiFms.h, 1459
- FIRMWARE_UPGRADE_SUCCESS
 - qaGobiApiFms.h, 1459
- FIRST_INSTANCE
 - qaGobiApiCbk.h, 1326
- FLOAT
 - SwiDataTypes.h, 1769
- FMS_FW_PRI_BUILD_MATCH_LEN
 - fms.h, 1259
- FMS_GOBI_LISTENTRIES_MAX
 - fms.h, 1259
- FMS_GOBI_MBN_BUILD_ID_STR_LEN
 - fms.h, 1259
- FMS_GOBI_MBN_IMG_ID_STR_LEN
 - fms.h, 1259
- FMS_IMAGE_ID_BUILD_ID_LEN
 - fms.h, 1260
- FMS_IMAGE_ID_IMG_ID_LEN
 - fms.h, 1260
- FMS_IMAGE_ID_MAX_ENTRIES
 - fms.h, 1260
- FMS_IMAGE_ID_PRI_IMGTYPE
 - fms.h, 1260
- FMS_MAX_IMAGE_ID_ELEMENT
 - fms.h, 1260
- FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE

- fms.h, [1260](#)
- FMSImageElement, [231](#)
 - buildId, [231](#)
 - buildIdLength, [232](#)
 - imageId, [232](#)
 - imageType, [232](#)
- FMSImageIDEntries, [233](#)
 - executingImage, [233](#)
 - imageIDElement, [233](#)
 - imageIDSize, [233](#)
 - imageType, [234](#)
 - maxImages, [234](#)
- FMSImageIDElement, [232](#)
 - buildIDLength, [232](#)
 - buildID, [232](#)
 - failureCount, [232](#)
 - imageID, [233](#)
 - storageIndex, [233](#)
- FMSImageList, [234](#)
 - imageIDEntries, [234](#)
 - listSize, [234](#)
- FMSPrefImageList, [234](#)
 - listEntries, [235](#)
 - listSize, [235](#)
- FORBIDDEN_INDEX
 - qaNasPerformNetworkScan.h, [1741](#)
- FOTAUpdate
 - _SLQSOMADMSettingsReqParams, [77](#)
 - _SLQSOMADMSettingsReqParams3, [78](#)
 - pack_swioama_SLQSOMADMSetSettings_t, [611](#)
 - unpack_swioama_SLQSOMADMGetSettings_↔
t, [1052](#)
- FOTAdownload
 - _SLQSOMADMSettingsReqParams, [77](#)
 - _SLQSOMADMSettingsReqParams3, [78](#)
 - pack_swioama_SLQSOMADMSetSettings_t, [610](#)
 - unpack_swioama_SLQSOMADMGetSettings_↔
t, [1052](#)
- FSNumber
 - FactorySequenceNumber, [225](#)
- facility
 - pack_dms_UIMGetControlKeyStatus_t, [560](#)
 - pack_dms_UIMSetControlKeyProtection_t, [562](#)
 - pack_dms_UIMUnblockControlKey_t, [563](#)
- facilityCk
 - pack_dms_UIMSetControlKeyProtection_t, [562](#)
 - pack_dms_UIMUnblockControlKey_t, [563](#)
- facilityState
 - pack_dms_UIMSetControlKeyProtection_t, [562](#)
 - unpack_dms_UIMGetControlKeyStatus_t, [956](#)
- FactorySequenceNumber, [225](#)
 - FSNumber, [225](#)
- failureCount
 - FMSImageIDElement, [232](#)
 - ImageIDElement, [297](#)
- failureReason
 - ssdatasession_params, [826](#)
- failureReasonv4
 - ssdatasession_params, [826](#)
- failureReasonv6
 - ssdatasession_params, [826](#)
- family
 - pack_wds_GetDefaultProfileNum_t, [620](#)
 - pack_wds_SetDefaultProfileNum_t, [624](#)
- feature
 - depersonalizationInformation, [203](#)
 - personalizationStatus, [650](#)
- fileAttributes, [225](#)
 - fileID, [228](#)
 - fileSize, [228](#)
 - fileType, [228](#)
 - rawLen, [228](#)
 - rawValue, [228](#)
 - recordCount, [228](#)
 - recordSize, [228](#)
 - secActivate, [228](#)
 - secActivateMask, [228](#)
 - secDeactivate, [228](#)
 - secDeactivateMask, [228](#)
 - secIncrease, [228](#)
 - secIncreaseMask, [228](#)
 - secRead, [228](#)
 - secReadMask, [228](#)
 - secWrite, [228](#)
 - secWriteMask, [229](#)
- fileID
 - fileAttributes, [228](#)
 - fileInfo, [229](#)
 - uim_fileInfo, [881](#)
- fileIndex
 - pack_uim_ReadTransparent_t, [613](#)
 - UIMGetFileAttributesReq, [897](#)
 - UIMReadTransparentReq, [901](#)
- fileInfo, [229](#)
 - fileID, [229](#)
 - path, [229](#)
 - pathLen, [229](#)
- fileSize
 - fileAttributes, [228](#)
- fileType
 - fileAttributes, [228](#)
- fill_pack_ctx
 - common.h, [1199](#)
- fill_sdu_hdr
 - common.h, [1199](#)
- filterId
 - LibPackTFTIDParams, [347](#)
 - TFTIDParams, [860](#)
- Firmware Management Service (FMS), [42](#)
- FirmwareID
 - fwinfo_s, [236](#)
- FirmwareUpdatStat, [229](#)
 - plmgType, [231](#)
 - pLogString, [231](#)
 - pLogStringLen, [231](#)
 - pRefData, [231](#)

- pRefString, 231
 - pRefStringLength, 231
 - ResCode, 231
- fix_rate
 - pack_swiloc_SwiLocSetAutoStart_t, 608
 - SwiLocGetAutoStartResp, 833
 - SwiLocSetAutoStartReq, 835
 - unpack_swiloc_SwiLocGetAutoStart_t, 1046
- fix_rate_reported
 - SwiLocGetAutoStartResp, 833
 - unpack_swiloc_SwiLocGetAutoStart_t, 1046
- fix_type
 - pack_swiloc_SwiLocSetAutoStart_t, 608
 - SwiLocGetAutoStartResp, 833
 - SwiLocSetAutoStartReq, 835
 - unpack_swiloc_SwiLocGetAutoStart_t, 1046
- fix_type_reported
 - SwiLocGetAutoStartResp, 833
 - unpack_swiloc_SwiLocGetAutoStart_t, 1047
- flags
 - sensorData, 744
 - sensorData_t, 746
- flowLabel
 - LibPackTFTIDParams, 347
 - TFTIDParams, 860
- fms.h, 1258
 - FMS_FW_PRI_BUILD_MATCH_LEN, 1259
 - FMS_GOB_I_LISTENTRIES_MAX, 1259
 - FMS_GOB_I_MBN_BUILD_ID_STR_LEN, 1259
 - FMS_GOB_I_MBN_IMG_ID_STR_LEN, 1259
 - FMS_IMAGE_ID_BUILD_ID_LEN, 1260
 - FMS_IMAGE_ID_IMG_ID_LEN, 1260
 - FMS_IMAGE_ID_MAX_ENTRIES, 1260
 - FMS_IMAGE_ID_PRI_IMGTYP, 1260
 - FMS_MAX_IMAGE_ID_ELEMENT, 1260
 - FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE, 1260
 - GetValidFwPriCombinations, 1260
 - pack_fms_GetImagesPreference, 1260
 - pack_fms_GetStoredImages, 1261
 - pack_fms_SetImagesPreference, 1261
 - unpack_fms_GetImagesPreference, 1261
 - unpack_fms_GetStoredImages, 1261
 - unpack_fms_SetImagesPreference, 1262
- Forbidden
 - nas_QmiNas3GppNetworkInfo, 462
 - SlqsNas3GppNetworkInfo, 792
- ForceRev0
 - unpack_nas_GetCDMANetworkParameters_t, 982
- ForceXTRADownload
 - qaGobiApiPds.h, 1550
- format
 - SMSTransferRouteMTMessage, 822
 - sMSTransferRouteMTMessage, 821
- ForwardMac
 - protocolSubtypeElement, 676
- fqdnAddr
 - PCSCFFQDNAddress, 643
- wds_PCSCFFQDNAddress, 1168
- fqdnLen
 - PCSCFFQDNAddress, 643
 - wds_PCSCFFQDNAddress, 1169
- freeSlots
 - smsMaxStorageSizeResp, 814
- freq
 - NASPhyCaAggPcellInfo, 513
 - NASPhyCaAggScellIndType, 515
 - NASPhyCaAggScellInfo, 516
 - nas_PhyCaAggPcellInfo, 456
 - nas_PhyCaAggScellIndType, 458
 - nas_PhyCaAggScellInfo, 461
 - PhyCaAggPcellInfo, 651
 - PhyCaAggScellIndType, 652
 - PhyCaAggScellInfo, 655
- freqsLen
 - LTEInfoInterfreq, 382
 - LTEInfoNeighboringGSM, 385
 - LTEInfoNeighboringWCDMA, 386
 - nas_LTEInfoInterfreq, 441
 - nas_LTEInfoNeighboringGSM, 444
 - nas_LTEInfoNeighboringWCDMA, 445
- fromServiceId
 - BroadcastConfig, 124
- fumoResultCode
 - omaDmFotaTlvExt, 549
- function
 - pack_swiloc_SwiLocSetAutoStart_t, 608
 - SwiLocGetAutoStartResp, 833
 - SwiLocSetAutoStartReq, 835
 - unpack_swiloc_SwiLocGetAutoStart_t, 1047
- function_reported
 - SwiLocGetAutoStartResp, 833
 - unpack_swiloc_SwiLocGetAutoStart_t, 1047
- FwAutoCheck
 - unpack_swiooma_SLQSOMADMGetSettings_t, 1052
- FwAvailability
 - unpack_swiooma_SLQSOMADMStartSession_t, 1053
- fwloadsize
 - omaDmFotaTlv, 547
 - unpack_omaDmFotaTlv_t, 1017
- fwinfo_s, 235
 - Carrier, 236
 - FirmwareID, 236
 - GPSCapability, 236
 - Region, 236
 - Technology, 236
- fwloadComplete
 - omaDmFotaTlv, 547
 - unpack_omaDmFotaTlv_t, 1017
- fwvers
 - CurrentImgList, 182
 - unpack_dms_SLQSSwiGetFirmwareCurr_t, 949
- g
 - qmifwinfo_s, 706

- G3K_FIRMWARE_DOWNLOAD
 - qaGobiApiFms.h, [1459](#)
- gDIBitRate
 - LibPackQosClassID, [345](#)
 - QosClassID, [715](#)
- GERANInfo, [236](#)
 - arfcn, [237](#)
 - bsic, [237](#)
 - cellID, [237](#)
 - insNmrCellInfo, [237](#)
 - lac, [237](#)
 - nmrInst, [237](#)
 - plmn, [237](#)
 - rxLev, [237](#)
 - timingAdvance, [237](#)
- GOBI_LISTENTRIES_MAX
 - qaGobiApiFms.h, [1459](#)
- GOBI_MBN_BUILD_ID_STR_LEN
 - qaGobiApiFms.h, [1459](#)
- GOBI_MBN_IMG_ID_STR_LEN
 - qaGobiApiFms.h, [1459](#)
- GOBI_SET_IMG_PREF_RSPLN
 - qaGobiApiFms.h, [1459](#)
- GPRSGrantedQoS
 - unpack_wds_SLQSGetRuntimeSettings_t, [1078](#)
- GPRSQoS, [271](#)
 - delayClass, [272](#)
 - meanThroughputClass, [272](#)
 - peakThroughputClass, [272](#)
 - precedenceClass, [272](#)
 - reliabilityClass, [272](#)
- GPRSRequestedQoS, [272](#)
 - delayClass, [273](#)
 - meanThroughputClass, [273](#)
 - peakThroughputClass, [273](#)
 - precedenceClass, [273](#)
 - reliabilityClass, [273](#)
- GPSCapability
 - finfo_s, [236](#)
- GPSPLM
 - pack_dms_SetCustFeature_t, [554](#)
 - unpack_dms_GetCustFeature_t, [927](#)
- GPSSel
 - pack_dms_SetCustFeature_t, [554](#)
 - unpack_dms_GetCustFeature_t, [927](#)
- GPSStateInfo, [273](#)
 - Altitude, [276](#)
 - EngineState, [276](#)
 - glo_almanac_sv_msk, [276](#)
 - glo_ephemeris_sv_msk, [276](#)
 - glo_health_sv_msk, [276](#)
 - glo_visible_sv_msk, [276](#)
 - gps_almanac_sv_msk, [276](#)
 - gps_ephemeris_sv_msk, [276](#)
 - gps_health_sv_msk, [276](#)
 - gps_visible_sv_msk, [276](#)
 - HorizontalUncertainty, [277](#)
 - lono_valid, [277](#)
 - Latitude, [277](#)
 - Longitude, [277](#)
 - sbas_almanac_sv_msk, [277](#)
 - sbas_ephemeris_sv_msk, [277](#)
 - sbas_health_sv_msk, [277](#)
 - sbas_visible_sv_msk, [277](#)
 - Time_uncert_ms, [277](#)
 - TimeStmp_gps_week, [277](#)
 - TimeStmp_tow_ms, [277](#)
 - ValidMask, [277](#)
 - VerticalUncertainty, [277](#)
 - xtra_start_gps_minutes, [277](#)
 - xtra_start_gps_week, [277](#)
 - xtra_valid_duration_hours, [277](#)
- GSMRSSIThresh, [279](#)
 - GSMRSSIThreshListLen, [280](#)
 - pGSMRSSIThreshList, [280](#)
- GSMRSSIThreshListLen
 - GSMRSSIThresh, [280](#)
 - nas_GSMRSSIThresh, [427](#)
- GSMSSInfo
 - unpack_nas_SLQSNasGetSigInfo_t, [1005](#)
- GSMSrvStatusInfo, [280](#)
 - isPrefDataPath, [280](#)
 - srvStatus, [281](#)
 - trueSrvStatus, [281](#)
- GSMSysInfo, [281](#)
 - cellId, [283](#)
 - cellIdValid, [283](#)
 - dtmSupp, [283](#)
 - dtmSuppValid, [283](#)
 - egprsSupp, [283](#)
 - egprsSuppValid, [283](#)
 - lac, [283](#)
 - lacValid, [283](#)
 - MCC, [283](#)
 - MNC, [283](#)
 - networkIdValid, [283](#)
 - regRejectInfoValid, [283](#)
 - rejCause, [283](#)
 - rejectSrvDomain, [284](#)
 - sysInfoGSM, [284](#)
- gUIBitRate
 - LibPackQosClassID, [345](#)
 - QosClassID, [715](#)
- GWAOPTlv
 - NASQmiCbkNasSystemSelPrefInd, [521](#)
- GWAcqOrderPref
 - NASGWAcqOrderPrefTlv, [504](#)
- GWAddressV4
 - unpack_wds_SLQSGetRuntimeSettings_t, [1078](#)
- gcDumpStrLen
 - CrashInfo, [173](#)
- gcdumpString
 - crashInformation, [175](#)
- gcdumpStrlen
 - crashInformation, [175](#)
- Generator

- GetAudioProfileReq, 244
- GetAudioVolTLBConfigReq, 245
- GetM2MAudioProfileResp, 262
- GetM2MAudioVolumeReq, 262
- SetAudioProfileReq, 757
- SetAudioVolTLBConfigReq, 758
- SetM2MAudioVolumeReq, 769
- geoSysIdx
 - AddCDMASysInfo, 99
 - AddSysInfo, 100
 - nas_AddCDMASysInfo, 406
 - nas_AddSysInfo, 406
- geranArfcn
 - geranInstInfo, 238
 - nas_geranInstInfo, 426
- geranBsicBcc
 - geranInstInfo, 238
 - nas_geranInstInfo, 426
- geranBsicNcc
 - geranInstInfo, 238
 - nas_geranInstInfo, 426
- geranInst
 - nas_UMTSInfo, 484
 - UMTSInfo, 915
- GeranInstInfo
 - nas_UMTSInfo, 484
 - UMTSInfo, 915
- geranInstInfo, 238
 - geranArfcn, 238
 - geranBsicBcc, 238
 - geranBsicNcc, 238
 - geranRssi, 238
- geranRssi
 - geranInstInfo, 238
 - nas_geranInstInfo, 426
- get_version
 - common.h, 1199
- GetACCOLC
 - qaGobiApiNas.h, 1512
- GetANAAAAAuthenticationStatus
 - qaGobiApiNas.h, 1512
- GetActivationState
 - qaGobiApiDms.h, 1424
- getAllCallInfo
 - arrCallInfo, 115
- getAllCallInformation, 238
 - ALS, 239
 - CallInfo, 239
 - isEmpty, 239
- GetAllCallRmtPtyName
 - arrRemotePartyName, 118
- getAllCallRmtPtyName, 239
 - callID, 240
 - RemotePartyName, 240
- getAllCallRmtPtyNum, 240
 - callID, 240
 - RemotePartyNum, 240
- GetAudioPathConfigReq, 240
 - Item, 241
 - Profile, 241
- GetAudioPathConfigResp, 241
 - pCodecSTGain, 242
 - pDTMFTXGain, 243
 - pECMode, 243
 - pMICGainSelect, 243
 - pNSEnable, 243
 - pRXAGCList, 243
 - pRXAVCAGCSwitch, 243
 - pRXAVCList, 243
 - pRXPCMIIRFtr, 243
 - pTXAGCList, 243
 - pTXAVCSwitch, 243
 - pTXGain, 243
 - pTXPCMIIRFtr, 243
- GetAudioProfileReq, 243
 - Generator, 244
- GetAudioProfileResp, 244
 - EarMute, 244
 - MicMute, 245
 - Profile, 245
 - Volume, 245
- GetAudioVolTLBConfigReq, 245
 - Generator, 245
 - Item, 245
 - Profile, 245
 - Volume, 246
- GetAudioVolTLBConfigResp, 246
 - ResCode, 246
- GetAutoconnect
 - qaGobiApiWds.h, 1702
- GetByteTotals
 - qaGobiApiWds.h, 1702
- GetCDMANetworkParameters
 - qaGobiApiNas.h, 1513
- getCallFWExtInfo, 246
 - CallFWExtInfo, 247
 - numInstances, 247
- getCallFWInfo, 247
 - CallFWInfo, 247
 - numInstances, 247
- GetConnectionRate
 - qaGobiApiWds.h, 1703
- GetCustomFeatureV2
 - unpack_dms_GetCustFeaturesV2_t, 928
- getCustomFeatureV2, 247
 - pCustSettingInfo, 248
 - pCustSettingList, 248
 - pGetCustomInput, 248
- getCustomInput, 248
 - cust_id, 248
 - list_type, 248
- getDUNCallInfoReq, 248
 - Mask, 249
 - pReportChannelRate, 249
 - pReportConnStatus, 249
 - pReportDataBearerTech, 249

- pReportDormStatus, [250](#)
 - pTransferStatInd, [250](#)
- getDUNCallInfoResp, [250](#)
 - pCallEndReason, [252](#)
 - pChannelRate, [252](#)
 - pConnectionStatus, [252](#)
 - pDataBearerTech, [252](#)
 - pDormancyStatus, [252](#)
 - pLastCallDataBearerTech, [252](#)
 - pLastCallRXOKBytesCnt, [252](#)
 - pLastCallTXOKBytesCnt, [252](#)
 - pMdmCallDurationActive, [252](#)
 - pRXOKBytesCount, [252](#)
 - pTXOKBytesCount, [252](#)
- GetDataBearerTechnology
 - qaGobiApiWds.h, [1703](#)
- GetDefaultProfile
 - qaGobiApiWds.h, [1704](#)
- GetDefaultProfileLTE
 - qaGobiApiWds.h, [1706](#)
- GetDefaultProfileNum
 - qaGobiApiWds.h, [1708](#)
- GetDeviceCapabilities
 - qaGobiApiDms.h, [1425](#)
- GetDormancyState
 - qaGobiApiWds.h, [1708](#)
- getDyingGaspCfg, [253](#)
 - pDestSMSContent, [253](#)
 - pDestSMSNum, [253](#)
- getDyingGaspStatistics, [253](#)
 - pSMSAttemptedFlag, [253](#)
 - pTimeStamp, [253](#)
- GetErrRateResp, [254](#)
 - pCDMAFrameErrRate, [254](#)
 - pGSMBER, [254](#)
 - pHDRPackErrRate, [254](#)
 - pWCDMABER, [254](#)
- GetFirmwareRevision
 - qaGobiApiDms.h, [1426](#)
- GetFirmwareRevisions
 - qaGobiApiDms.h, [1427](#)
- GetHRPDStatsResp, [255](#)
 - pDRCPParams, [255](#)
 - pPilotSetData, [255](#)
 - pUATI, [255](#)
- GetHardwareRevision
 - qaGobiApiDms.h, [1427](#)
- GetHomeNetwork
 - qaGobiApiNas.h, [1515](#)
- GetHomeNetwork3GPP2
 - qaGobiApiNas.h, [1516](#)
- GetIMSSMSConfigParams, [255](#)
 - pPhoneCtxURLen, [256](#)
 - pPhoneCtxURI, [256](#)
 - pSMSFormat, [256](#)
 - pSMSOverIPNWInd, [256](#)
 - pSettingResp, [256](#)
- GetIMSUserConfigParams, [256](#)
 - pIMSDomain, [257](#)
 - pIMSDomainLen, [257](#)
 - pSettingResp, [257](#)
- GetIMSVoIPConfigResp, [257](#)
 - pAmrMode, [259](#)
 - pAmrOctetAligned, [259](#)
 - pAmrWBMode, [259](#)
 - pAmrWBOctetAligned, [259](#)
 - pAmrWbEnable, [259](#)
 - pMinSessionExpiryTimer, [259](#)
 - pRTPRTCPInactTimer, [260](#)
 - pRingBackTimer, [260](#)
 - pRingingTimer, [260](#)
 - pScrAmrEnable, [260](#)
 - pScrAmrWbEnable, [260](#)
 - pSessionExpiryTimer, [260](#)
 - pSettingResp, [260](#)
- GetIMSI
 - qaGobiApiDms.h, [1428](#)
- GetIPAddressLTE
 - qaGobiApiWds.h, [1709](#)
- GetImageStore
 - qaGobiApiFms.h, [1464](#)
- GetImagesPreference
 - qaGobiApiFms.h, [1463](#)
- getIndicationRegResp
 - qaGobiApiSms.h, [1577](#)
- GetInstIDResp, [260](#)
 - pIPFamily, [260](#)
 - pInstanceID, [260](#)
- GetLastMobileIPError
 - qaGobiApiWds.h, [1709](#)
- GetM2MAVMuteReq, [263](#)
 - Profile, [263](#)
- GetM2MAVMuteResp, [263](#)
 - CwtMute, [264](#)
 - EarMute, [264](#)
 - MicMute, [264](#)
- GetM2MAudioProfileReq, [260](#)
 - pGenerator, [261](#)
- GetM2MAudioProfileResp, [261](#)
 - CwtMute, [262](#)
 - EarMute, [262](#)
 - Generator, [262](#)
 - MicMute, [262](#)
 - Profile, [262](#)
 - Volume, [262](#)
- GetM2MAudioVolumeReq, [262](#)
 - Generator, [262](#)
 - Profile, [262](#)
- GetM2MAudioVolumeResp, [262](#)
 - Level, [263](#)
- GetM2MSpkrGainReq, [264](#)
 - Profile, [265](#)
- GetM2MSpkrGainResp, [265](#)
 - Value, [265](#)
- GetManufacturer
 - qaGobiApiDms.h, [1428](#)

- GetMobileIPProfile
 - qaGobiApiWds.h, [1710](#)
- GetMobileIP
 - qaGobiApiWds.h, [1710](#)
- GetModelID
 - qaGobiApiDms.h, [1429](#)
- getMsgWaitingInfo, [265](#)
 - msgWaitInfo, [266](#)
 - numInstances, [266](#)
- GetNetworkPreference
 - qaGobiApiNas.h, [1517](#)
- GetNetworkTime
 - qaGobiApiDms.h, [1429](#)
- GetNetworkTimeResp, [266](#)
 - p3GPP2TimeInfo, [266](#)
 - p3GPPTIMEInfo, [266](#)
- GetOfflineReason
 - qaGobiApiDms.h, [1430](#)
- GetPDSDDefaults
 - qaGobiApiPds.h, [1550](#)
- GetPDSSState
 - qaGobiApiPds.h, [1551](#)
- GetPRLVersion
 - qaGobiApiDms.h, [1431](#)
- GetPacketStatistics
 - qaGobiApiWds.h, [1712](#)
- GetPacketStatus
 - qaGobiApiWds.h, [1712](#)
- GetPortAutomaticTracking
 - qaGobiApiPds.h, [1551](#)
- GetPower
 - qaGobiApiDms.h, [1431](#)
- GetProfileSettingIn
 - qaGobiApiWds.h, [1698](#)
- GetProfileSettingOut
 - qaGobiApiWds.h, [1699](#)
- GetRFInfo
 - qaGobiApiNas.h, [1518](#)
- GetRegMgrConfigParams, [266](#)
 - pIMSTestMode, [267](#)
 - pPCSCFPort, [267](#)
 - pPriCSCFPortName, [267](#)
 - pPriCSCFPortNameLen, [267](#)
 - pSettingResp, [267](#)
- GetSIPConfigResp, [268](#)
 - pSIPLocalPort, [269](#)
 - pSettingResp, [269](#)
 - pSigCompEnabled, [269](#)
 - pSubscribeTimer, [269](#)
 - pTimerSIPReg, [269](#)
 - pTimerT1, [269](#)
 - pTimerT2, [269](#)
 - pTimerTf, [269](#)
- GetSMSCAddress
 - qaGobiApiSms.h, [1581](#)
- GetSMSWake
 - qaGobiApiRms.h, [1571](#)
- GetSerialNumbers
 - qaGobiApiDms.h, [1432](#)
- GetServiceAutomaticTracking
 - qaGobiApiPds.h, [1552](#)
- GetServingNetwork
 - qaGobiApiNas.h, [1519](#)
- GetServingNetworkCapabilities
 - qaGobiApiNas.h, [1520](#)
- GetSessionDuration
 - qaGobiApiWds.h, [1713](#)
- GetSessionIDResp, [267](#)
 - pSessionIDv4, [268](#)
 - pSessionIDv6, [268](#)
- GetSessionState
 - qaGobiApiWds.h, [1714](#)
- GetSignalStrengths
 - qaGobiApiNas.h, [1521](#)
- GetStoredImages
 - qaGobiApiFms.h, [1464](#)
- getTransLayerInfoResp
 - qaGobiApiSms.h, [1578](#)
- getTransNWRegInfoResp
 - qaGobiApiSms.h, [1578](#)
- GetValidFwPriCombinations
 - fms.h, [1260](#)
- GetVoiceNumber
 - qaGobiApiDms.h, [1433](#)
- GetXTRAAutomaticDownload
 - qaGobiApiPds.h, [1552](#)
- GetXTRANetwork
 - qaGobiApiPds.h, [1553](#)
- GetXTRAValidity
 - qaGobiApiPds.h, [1553](#)
- glo_almanac_sv_msk
 - GPSSStateInfo, [276](#)
- glo_ephemeris_sv_msk
 - GPSSStateInfo, [276](#)
- glo_health_sv_msk
 - GPSSStateInfo, [276](#)
- glo_visible_sv_msk
 - GPSSStateInfo, [276](#)
- globalCellId
 - LTEInfoIntrafreq, [384](#)
 - nas_LTEInfoIntrafreq, [443](#)
- glog
 - common.h, [1200](#)
- gloglvl
 - common.h, [1200](#)
- GnssData, [269](#)
 - mask, [270](#)
- gnssSvId
 - loc_satelliteInfo, [360](#)
 - satelliteInfo, [742](#)
- gnssSvInfoNotification, [270](#)
 - bAltitudeAssumed, [271](#)
 - pSatelliteInfo, [271](#)
- gnssSvUsedList
 - loc_svUsedforFix, [363](#)
 - svUsedforFix_s, [830](#)

- gnssSvUsedList_len
 - loc_svUsedforFix, [363](#)
 - svUsedforFix_s, [830](#)
- Gpp2TimeZone
 - qaQmiServingSystemParam, [682](#)
 - unpack_nas_SLQSGetServingSystem_t, [995](#)
- GppNetworkDSTAdjustment
 - qaQmiServingSystemParam, [682](#)
 - unpack_nas_SLQSGetServingSystem_t, [995](#)
- GppTimeZone
 - qaQmiServingSystemParam, [682](#)
 - unpack_nas_SLQSGetServingSystem_t, [995](#)
- gps_almanac_sv_msk
 - GPSSStateInfo, [276](#)
- gps_ephemeris_sv_msk
 - GPSSStateInfo, [276](#)
- gps_health_sv_msk
 - GPSSStateInfo, [276](#)
- gps_visible_sv_msk
 - GPSSStateInfo, [276](#)
- GpsEnable
 - custFeaturesInfo, [189](#)
 - pack_dms_SetCustFeature_t, [554](#)
 - unpack_dms_GetCustFeature_t, [927](#)
- gpsTime
 - qaGobiApiCbK.h, [1329](#)
- gpsTime_s, [277](#)
 - gpsTimeOfWeekMs, [278](#)
 - gpsWeek, [278](#)
- gpsTimeOfWeekMs
 - gpsTime_s, [278](#)
 - loc_gpsTime, [356](#)
 - t_gpsTime, [852](#)
- gpsWeek
 - gpsTime_s, [278](#)
 - loc_gpsTime, [356](#)
 - t_gpsTime, [852](#)
- grntDownlinkBitrate
 - LibPackUMTSQoS, [349](#)
 - UMTSMinQoS, [919](#)
 - UMTSQoS, [922](#)
 - wds_UMTSMinQoS, [1173](#)
- grntUplinkBitrate
 - LibPackUMTSQoS, [349](#)
 - UMTSMinQoS, [919](#)
 - UMTSQoS, [922](#)
 - wds_UMTSMinQoS, [1174](#)
- gsmAmrStat
 - curAMRConfig, [180](#)
- GsmCellInfo
 - lteGsmCellInfo, [379](#)
 - nas_lteGsmCellInfo, [438](#)
- gsmCellInfo, [278](#)
 - arfcn, [279](#)
 - band1900, [279](#)
 - bsicId, [279](#)
 - cellIdValid, [279](#)
 - rsi, [279](#)
 - srxlev, [279](#)
- gsmUmtsDI
 - NasSwIndReg, [526](#)
 - pack_nas_SLQSNasSwIndicationRegister_t, [594](#)
- gsmUmtsUI
 - NasSwIndReg, [526](#)
 - pack_nas_SLQSNasSwIndicationRegister_t, [594](#)
- guaranteedRate
 - dataRate, [198](#)
 - unpack_qos_dataRate_t, [1019](#)
- gwAddressV6
 - IPV6GWAddressInfo, [320](#)
 - wds_IPV6GWAddressInfo, [1168](#)
- gwV6PrefixLen
 - IPV6GWAddressInfo, [320](#)
 - wds_IPV6GWAddressInfo, [1168](#)
- gyroAcceptReady
 - qaGobiApiCbK.h, [1330](#)
- gyroAcceptReady_s, [284](#)
 - batchPerSec, [284](#)
 - injectEnable, [284](#)
 - samplesPerBatch, [284](#)
- gyroData
 - pack_loc_SLQSLOCInjectSensorData_t, [578](#)
- gyroTemp
 - pack_loc_SLQSLOCInjectSensorData_t, [579](#)
- gyroTempAcceptReady
 - qaGobiApiCbK.h, [1330](#)
- gyroTempAcceptReady_s, [284](#)
 - batchPerSec, [285](#)
 - injectEnable, [285](#)
 - samplesPerBatch, [285](#)
- gyroTimeSrc
 - pack_loc_SLQSLOCInjectSensorData_t, [579](#)
- HASPI
 - unpack_wds_GetMobileIPProfile_t, [1066](#)
- HASState
 - unpack_wds_GetMobileIPProfile_t, [1066](#)
- HDOP
 - loc_precisionDilution, [358](#)
 - precisionDilution_s, [660](#)
- HDRECIOTresh, [285](#)
 - HDRECIOTreshListLen, [286](#)
 - pHDRECIOTreshList, [286](#)
- HDRECIOTreshListLen
 - HDRECIOTresh, [286](#)
 - nas_HDRECIOTresh, [432](#)
- HDRIOTresh, [286](#)
 - HDRIOTreshListLen, [286](#)
 - pHDRIOTreshList, [286](#)
- HDRIOTreshListLen
 - HDRIOTresh, [286](#)
 - nas_HDRIOTresh, [432](#)
- HDRPersonalityInd, [286](#)
 - pCurrentPersonality, [287](#)
 - pPersonalityListLength, [287](#)
 - pProtocolSubtypeElement, [287](#)
- HDRPersonalityResp, [287](#)

- pCurrentPersonality, [287](#)
 - pPersonalityListLength, [287](#)
 - pProtocolSubtypeElement, [287](#)
- HDRProtSubtypResp, [288](#)
 - pAppSubType, [288](#)
 - pCurrentPrsnlty, [288](#)
 - pPersonalityListLength, [288](#)
 - pProtoSubTypElmnt, [288](#)
- HDRRSSIThresh, [288](#)
 - HDRRSSIThreshListLen, [289](#)
 - pHDRRSSIThreshList, [289](#)
- HDRRSSIThreshListLen
 - HDRRSSIThresh, [289](#)
 - nas_HDRRSSIThresh, [433](#)
- HDRSINRThresListLen
 - HDRSINRThresh, [290](#)
- HDRSINRThresh, [289](#)
 - HDRSINRThresListLen, [290](#)
 - pHDRSINRThresList, [290](#)
- HDRSINRThreshListLen
 - HDRSINRThreshold, [290](#)
 - nas_HDRSINRThreshold, [434](#)
- HDRSINRThreshold, [290](#)
 - HDRSINRThreshListLen, [290](#)
 - pHDRSINRThreshList, [290](#)
- HDRSSInfo, [290](#)
 - ecio, [291](#)
 - io, [291](#)
 - rsi, [291](#)
 - sinr, [291](#)
 - unpack_nas_SLQSNasGetSigInfo_t, [1005](#)
- HDRSysInfo, [292](#)
 - hdrActiveProt, [294](#)
 - hdrActiveProtValid, [294](#)
 - hdrPersonality, [294](#)
 - hdrPersonalityValid, [294](#)
 - is856SysId, [294](#)
 - is856SysIdValid, [294](#)
 - isSysPrIMatch, [294](#)
 - isSysPrIMatchValid, [294](#)
 - sysInfoHDR, [294](#)
- HWVersion
 - DeviceConfigDetail, [206](#)
- HardwareControlledMode
 - unpack_dms_GetPower_t, [936](#)
- has_accelTemp
 - pack_loc_SLQSLOCInjectSensorData_t, [579](#)
- has_acceleroTimeSrc
 - pack_loc_SLQSLOCInjectSensorData_t, [579](#)
- has_accleroData
 - pack_loc_SLQSLOCInjectSensorData_t, [579](#)
- has_altitudeSrcInfo
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_altitudeWrtEllipsoid
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_altitudeWrtMeanSeaLevel
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_confidence
 - pack_loc_SLQSLOCSetCradleMountConfig_t, [580](#)
- has_gyroData
 - pack_loc_SLQSLOCInjectSensorData_t, [579](#)
- has_gyroTemp
 - pack_loc_SLQSLOCInjectSensorData_t, [579](#)
- has_gyroTimeSrc
 - pack_loc_SLQSLOCInjectSensorData_t, [579](#)
- has_horConfidence
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_horReliability
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_horUncCircular
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_latitude
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_longitude
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_opaqueId
 - pack_loc_SLQSLOCInjectSensorData_t, [579](#)
- has_positionSrc
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_rawHorConfidence
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_rawHorUncCircular
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_timestampAge
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_timestampUtc
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_vertConfidence
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_vertRelicability
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- has_vertUnc
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- hdrActiveProt
 - HDRSysInfo, [294](#)
 - nas_HDRSysInfo, [436](#)
- hdrActiveProtValid
 - HDRSysInfo, [294](#)
 - nas_HDRSysInfo, [436](#)
- hdrHybrid
 - detailSvcInfo, [204](#)
 - nas_detailSvcInfo, [421](#)
- HdrPersonality
 - unpack_nas_SLQSGetServingSystem_t, [995](#)
- hdrPersonality
 - HDRSysInfo, [294](#)
 - NASServingSystemInfo, [523](#)
 - nas_HDRSysInfo, [436](#)
 - qaQmiServingSystemParam, [683](#)
 - ServingSystemInfo, [751](#)
- hdrPersonalityValid
 - HDRSysInfo, [294](#)
 - nas_HDRSysInfo, [436](#)
- hdrSSInfo, [292](#)
 - ecio, [292](#)
 - io, [292](#)

- rssI, 292
 - sinr, 292
- hdrSrvStatus
 - detailSvcInfo, 205
 - nas_detailSvcInfo, 421
- healthStatus
 - loc_satelliteInfo, 360
 - satelliteInfo, 742
- helper_get_resp_ctx
 - common.h, 1199
- helper_get_xid
 - common.h, 1199
- helper_set_log_func
 - common.h, 1199
- helper_set_log_lvl
 - common.h, 1199
- homeOrigVoiceSO
 - prefVoiceSO, 663
- homePageVoiceSO
 - prefVoiceSO, 663
- homeSIDNID, 294
 - numInstances, 295
 - SidNid, 295
- horConfidence
 - pack_loc_SLQSLOCInjectPosition_t, 576
- horReliability
 - pack_loc_SLQSLOCInjectPosition_t, 576
- horUncCircular
 - pack_loc_SLQSLOCInjectPosition_t, 576
- HorizontalUncertainty
 - GPSSStateInfo, 277
- hotSwap
 - hotSwapStatus, 295
 - uim_hotSwapStatus, 882
- hotSwapLength
 - hotSwapStatus, 295
 - uim_hotSwapStatus, 882
- hotSwapStatus, 295
 - hotSwap, 295
 - hotSwapLength, 295
- hour
 - nas_UniversalTime, 487
 - nas_timeInfo, 482
 - timeInfo, 862
 - UniversalTime, 925
- hsCallStatus
 - nas_WCDMASysInfo, 493
 - WCDMASysInfo, 1157
- hsCallStatusValid
 - nas_WCDMASysInfo, 493
 - WCDMASysInfo, 1157
- hsInd
 - nas_WCDMASysInfo, 493
 - WCDMASysInfo, 1157
- hsIndValid
 - nas_WCDMASysInfo, 493
 - WCDMASysInfo, 1157
- hwType
 - wds_DHCPv4HWConfig, 1163
 - WdsDHCPv4HWConfig, 1180
 - wdsDhcpv4HwConfig, 1180
- hwVer
 - unpack_dms_GetHardwareRevision_t, 933
- iFaceTab
 - PCMparams, 642
- iFaceTabLen
 - PCMparams, 642
- iGetByteTotals
 - qaGobiApiWds.h, 1714
- iGetConnectionRate
 - qaGobiApiWds.h, 1714
- iGetPacketStatistics
 - qaGobiApiWds.h, 1714
- iLTEbandValue
 - NASPhyCaAggPcellInfo, 513
 - NASPhyCaAggScellInfo, 516
 - nas_PhyCaAggPcellInfo, 456
 - nas_PhyCaAggScellInfo, 461
 - PhyCaAggPcellInfo, 651
 - PhyCaAggScellInfo, 655
- IMCNflag
 - unpack_wds_SLQSGetRuntimeSettings_t, 1078
- IMEIString
 - unpack_dms_GetDeviceSerialNumbers_t, 930
- IMG_ID_LEN
 - qaGobiApiFms.h, 1459
- IMGDETAILS_LEN
 - qaGobiApiFms.h, 1459
- IMS Service (IMS), 51
- IMSALndRegisterInfo, 299
 - pPdpStatusConfig, 300
 - pRatHandoverStatusConfig, 300
 - pRegStatusConfig, 300
 - pServiceStatusConfig, 300
- IMSARegistrationStatus, 301
 - plmsRegErrCode, 302
 - plmsRegStatus, 302
 - pNewlmsRegStatus, 302
- IMASServiceStatus, 303
 - pSmsServiceRat, 305
 - pSmsServiceStatus, 305
 - pUtServiceRat, 305
 - pUtServiceStatus, 305
 - pVoipServiceRat, 305
 - pVoipServiceStatus, 305
 - pVsServiceRat, 305
 - pVsServiceStatus, 305
 - pVtServiceRat, 305
 - pVtServiceStatus, 305
- IMASupportedFieldsResp, 305
 - plndFieldsList, 306
 - pReqFieldsList, 306
 - pRespFieldsList, 306
- IMASupportedMsgInfo, 306
 - pSupportedMsgList, 306
- IMSI_M_S1_LENGTH

- qaGobiApiNas.h, [1499](#)
- IMSI_M_S2_LENGTH
 - qaGobiApiNas.h, [1499](#)
- IMSInfo
 - sMSOnIMSTlv, [818](#)
- IMSTlv
 - unpack_sms_SetNewSMSCallback_ind_t, [1043](#)
- INDEX_ZERO
 - qaNasPerformNetworkScan.h, [1741](#)
- INT32
 - SwiDataTypes.h, [1769](#)
- INT8
 - SwiDataTypes.h, [1769](#)
- INVALID_INSTACNE
 - qaGobiApiCbk.h, [1326](#)
- IOThresListLen
 - IOThresh, [317](#)
- IOThresh, [316](#)
 - IOThresListLen, [317](#)
 - plIOThresList, [317](#)
- IPAddress
 - DataStatusDetail, [200](#)
- IPAddressV6
 - IPV6AddressInfo, [319](#)
 - ipv6AddressInfo, [319](#)
 - wds_IPV6AddressInfo, [1167](#)
- IPFamSupport
 - pack_dms_SetCustFeature_t, [555](#)
 - unpack_dms_GetCustFeature_t, [927](#)
- IPFamilyPreference
 - pack_wds_SLQSSetIPFamilyPreference_t, [634](#)
 - unpack_wds_SLQSGetRuntimeSettings_t, [1078](#)
- IPSECSPI
 - LibPackTFTIDParams, [347](#)
 - TFTIDParams, [860](#)
- IPV4
 - qaGobiApiCbk.h, [1326](#)
- IPV4V6
 - qaGobiApiCbk.h, [1327](#)
- IPV6
 - qaGobiApiCbk.h, [1327](#)
- IPV6_ADDRESS_ARRAY_SIZE
 - qaGobiApiWds.h, [1698](#)
 - wds.h, [1798](#)
- IPV6AddrInfo
 - unpack_wds_SLQSGetRuntimeSettings_t, [1078](#)
- IPV6AddressInfo, [318](#)
 - IPAddressV6, [319](#)
 - IPV6PrefixLen, [319](#)
- IPV6GWAddrInfo
 - unpack_wds_SLQSGetRuntimeSettings_t, [1078](#)
- IPV6GWAddressInfo, [319](#)
 - gwAddressV6, [320](#)
 - gwV6PrefixLen, [320](#)
- IPV6PrefixLen
 - IPV6AddressInfo, [319](#)
 - ipv6AddressInfo, [319](#)
 - wds_IPV6AddressInfo, [1167](#)
- IPv4
 - unpack_wds_SLQSGetRuntimeSettings_t, [1078](#)
- IPv4Addr, [317](#)
 - addr, [317](#)
 - subnetMask, [318](#)
 - wds_IPv4AdTlv, [1167](#)
- IPv4AddrTlv
 - unpack_wds_DHCPv4ClientLease_ind_t, [1060](#)
- IPv4DstAddr
 - unpack_qos_swiQosFilter_t, [1033](#)
- IPv4SrcAddr
 - unpack_qos_swiQosFilter_t, [1033](#)
- IPv4Tos
 - unpack_qos_swiQosFilter_t, [1033](#)
- IPv6Addr, [318](#)
 - addr, [318](#)
 - prefixLen, [318](#)
- IPv6DstAddr
 - unpack_qos_swiQosFilter_t, [1033](#)
- IPv6Label
 - unpack_qos_swiQosFilter_t, [1033](#)
- IPv6SrcAddr
 - unpack_qos_swiQosFilter_t, [1033](#)
- IPv6TrafCls, [320](#)
 - mask, [320](#)
 - unpack_qos_swiQosFilter_t, [1033](#)
 - val, [320](#)
- iSLQSMISetIPFamilyPreference
 - qaGobiApiWds.h, [1715](#)
- iSLQSSetDUNCallInfoCallback
 - qaGobiApiCbk.h, [1365](#)
- iSLQSSetSignalStrengthsCallback
 - qaGobiApiCbk.h, [1365](#)
- iSLQSSetWdsFirstInstEventCallback
 - qaGobiApiCbk.h, [1365](#)
- iSLQSSetWdsSecondInstEventCallback
 - qaGobiApiCbk.h, [1365](#)
- iSLQSSetWdsThirdInstEventCallback
 - qaGobiApiCbk.h, [1365](#)
- iSLQSSetWdsXferStatsFirstInstCallback
 - qaGobiApiCbk.h, [1365](#)
- iSLQSSetWdsXferStatsSecondInstCallback
 - qaGobiApiCbk.h, [1365](#)
- iSetCATEventCallback
 - qaGobiApiCbk.h, [1365](#)
- iSetSignalStrengthCallback
 - qaGobiApiCbk.h, [1365](#)
- Id
 - unpack_qos_swiQosFilter_t, [1032](#)
- id
 - BdsSV, [123](#)
 - CSGID, [178](#)
 - loc_BdsSV, [351](#)
 - loc_SV, [362](#)
 - nas_CSGID, [418](#)
 - pack_dms_UIMChangePIN_t, [560](#)
 - pack_dms_UIMSetPINProtection_t, [562](#)
 - pack_dms_UIMUnblockPIN_t, [564](#)

- pack_dms_UIMVerifyPIN_t, 564
 - QosFlowInfoState, 718
 - SV, 829
 - swiQosModifyReq, 847
 - unpack_qos_QosFlowInfoState_t, 1024
 - unpack_qos_SLQSSetQosStatusCallback_ind_t, 1030
- id_length
 - custSettingInfo, 192
 - DMScustSettingInfo, 211
- IdleState
 - protocolSubtypeElement, 676
- image_info_t, 295
 - buildIDLen, 296
 - buildID, 296
 - imageType, 296
 - uniqueID, 296
- ImageElement, 296
 - buildId, 296
 - buildIdLength, 296
 - imageId, 296
 - imageType, 296
- imageIDElement
 - FMSImageIDEntries, 233
 - ImageIDEntries, 298
- ImageIDEntries, 298
 - executingImage, 298
 - imageIDElement, 298
 - imageIDSize, 298
 - imageType, 298
 - maxImages, 298
- imageIDEntries
 - FMSImageList, 234
 - ImageList, 299
- imageIDSize
 - FMSImageIDEntries, 233
 - ImageIDEntries, 298
- imageID
 - FMSImageIdElement, 233
 - ImageIdElement, 297
- imageId
 - FMSImageElement, 232
 - ImageElement, 296
- ImageIdElement, 297
 - buildIDLength, 297
 - buildID, 297
 - failureCount, 297
 - imageID, 297
 - storageIndex, 297
- ImageList, 298
 - imageIDEntries, 299
 - listSize, 299
- imageList
 - unpack_fms_GetStoredImages_t, 961
- ImageListSize
 - unpack_fms_GetImagesPreference_t, 960
- imageListSize
 - pack_fms_SetImagesPreference_t, 566
- imageType
 - CurrImageInfo, 184
 - FMSImageElement, 232
 - FMSImageIDEntries, 234
 - image_info_t, 296
 - ImageElement, 296
 - ImageIDEntries, 298
- ImageTypes
 - unpack_fms_SetImagesPreference_t, 962
- ImageTypesSize
 - unpack_fms_SetImagesPreference_t, 962
- imagelistSize
 - unpack_fms_GetStoredImages_t, 961
- imei_no
 - unpack_dms_GetSerialNumbers_t, 937
- imeiSize
 - serialNumbersInfo, 748
 - unpack_dms_GetDeviceSerialNumbers_t, 930
- imeiSvnSize
 - serialNumbersInfo, 748
 - unpack_dms_GetDeviceSerialNumbers_t, 930
- ImeiSvnString
 - unpack_dms_GetDeviceSerialNumbers_t, 930
- imeisv_svn
 - unpack_dms_GetSerialNumbers_t, 938
- imgType
 - unpack_dms_SLQSSwiGetFwUpdateStatus_t, 951
- imsCfgIndRegisterInfo, 307
 - pRegMgrConfigEvents, 309
 - pSIPConfigEvents, 309
 - pSMSConfigEvents, 309
 - pUserConfigEvents, 309
 - pVoIPConfigEvents, 309
- imsRegMgrConfigInfo, 309
 - pCSCFPortName, 309
 - pIMSTestMode, 309
 - pPriCSCFPort, 309
- imsRegState
 - CommInfo, 169
 - nas_CommInfo, 417
- imsSIPConfigInfo, 310
 - pSIPLocalPort, 310
 - pSigCompEnabled, 310
 - pSubscribeTimer, 311
 - pTimerSIPReg, 311
 - pTimerT1, 311
 - pTimerT2, 311
 - pTimerTf, 311
- imsSMSConfigInfo, 311
 - pPhoneCtxtURI, 311
 - pSMSFormat, 311
 - pSMSOverIPNwInd, 311
- imsUserConfigInfo, 312
 - pIMSDomain, 312
- imsVoIPConfigInfo, 312
 - pAmrMode, 314
 - pAmrOctetAligned, 314
 - pAmrWBMode, 314

- pAmrWBOctetAligned, [314](#)
 - pAmrWbEnable, [314](#)
 - pMinSessionExpiryTimer, [314](#)
 - pRTPRTCPInactTimer, [314](#)
 - pRingBackTimer, [314](#)
 - pRingingTimer, [314](#)
 - pScrAmrEnable, [314](#)
 - pScrAmrWbEnable, [314](#)
 - pSessionExpiryTimer, [315](#)
- imsaPdpStatusInfo, [300](#)
 - connetionState, [301](#)
 - pFailErrorCode, [301](#)
- imsaRatStatusInfo, [301](#)
 - pErrorCodeStr, [301](#)
 - pRATStatus, [301](#)
 - pSrcRAT, [301](#)
 - pTgtRAT, [301](#)
- imsaRegStatusInfo, [302](#)
 - plmsRegStatus, [303](#)
 - pRegStatusErrorCode, [303](#)
 - pbIMSRegistered, [303](#)
- imsaSvcStatusInfo, [306](#)
 - pSMSSvcRAT, [307](#)
 - pSMSSvcStatus, [307](#)
 - pUTSvcRAT, [307](#)
 - pUTSvcStatus, [307](#)
 - pVOIPSvcRAT, [307](#)
 - pVOIPSvcStatus, [307](#)
 - pVTSvcRAT, [307](#)
 - pVTSvcStatus, [307](#)
- imsi
 - unpack_dms_GetIMSI_t, [933](#)
- imsi_11_12
 - CDMASysInfoExt, [159](#)
 - nas_CDMASysInfoExt, [415](#)
- imsiM1112
 - minBasedIMSI, [400](#)
- imsiMS1
 - minBasedIMSI, [400](#)
- imsiMS2
 - minBasedIMSI, [400](#)
- imsiT1112
 - trueIMSI, [871](#)
- imsiTS1
 - trueIMSI, [871](#)
- imsiTS2
 - trueIMSI, [871](#)
- imsiTaddrNum
 - trueIMSI, [871](#)
- InUse
 - nas_QmiNas3GppNetworkInfo, [462](#)
 - SlqsNas3GppNetworkInfo, [792](#)
- includes_pcs_digit
 - nas_QmisNasPcsDigit, [464](#)
 - SlqsNasPcsDigit, [792](#)
- IndFieldsList, [315](#)
 - indicationFields, [315](#)
 - indicationFieldsLen, [315](#)
- index
 - pack_wds_GetMobileIPProfile_t, [621](#)
 - pack_wds_SetDefaultProfileNum_t, [624](#)
 - pack_wds_SetMobileIPProfile_t, [627](#)
 - swiQosFilter, [842](#)
 - swiQosFlow, [845](#)
 - swiQosReq, [848](#)
 - unpack_qos_swiQosFilter_t, [1033](#)
 - unpack_qos_swiQosFlow_t, [1037](#)
 - unpack_wds_GetDefaultProfileNum_t, [1064](#)
- index1xPri
 - cardStatus, [139](#)
 - uim_cardStatus, [879](#)
- index1xSec
 - cardStatus, [139](#)
 - uim_cardStatus, [879](#)
- indexGwPri
 - cardStatus, [139](#)
 - uim_cardStatus, [879](#)
- indexGwSec
 - cardStatus, [139](#)
 - uim_cardStatus, [879](#)
- indicationFields
 - IndFieldsList, [315](#)
- indicationFieldsLen
 - IndFieldsList, [315](#)
- Info
 - unpack_nas_SLQSNasSwiOTAMessageCallback↔
_ind_t, [1008](#)
 - unpack_nas_SLQSSetSysSelectionPrefCall↔
Back_ind_t, [1010](#)
- infoInterFreq, [315](#)
 - cell_resel_priority, [316](#)
 - cellInterFreqParams, [316](#)
 - cells_len, [316](#)
 - earfcn, [316](#)
 - threshXHigh, [316](#)
 - threshXLow, [316](#)
- InfoInterfreq
 - LTEInfoInterfreq, [382](#)
 - nas_LTEInfoInterfreq, [441](#)
- InitiateDomainAttach
 - qaGobiApiNas.h, [1522](#)
- InitiateNetworkRegistration
 - qaGobiApiNas.h, [1522](#)
- injectEnable
 - accelAcceptReady_s, [96](#)
 - accelTempAcceptReady_s, [97](#)
 - gyroAcceptReady_s, [284](#)
 - gyroTempAcceptReady_s, [285](#)
- injectSensorDataStatus
 - QmiCbkLocInjectSensorDataInd, [693](#)
- injectTimeSyncStatus
 - QmiCbkLocInjectTimeInd, [694](#)
- insNmrCellInfo
 - GERANInfo, [237](#)
 - nas_GERANInfo, [425](#)
- instanceId

- ssdatasession_params, 826
- instancesSize
 - unpack_nas_GetRFInfo_t, 984
- interval
 - pack_wds_SLQSSetWdsEventCallback_t, 635
 - TransferStatsDataType, 869
- Io
 - slqsSignalStrengthInfo, 799
 - unpack_nas_SLQSGetSignalStrength_t, 997
- io
 - HDRSSInfo, 291
 - hdrSSInfo, 292
 - nas_SLQSSignalStrengthsInformation, 474
 - SLQSSignalStrengthsInformation, 803
- ioDelta
 - nas_SLQSSignalStrengthsIndReq, 473
 - SLQSSignalStrengthsIndReq, 801
- iono_valid
 - GPSSStateInfo, 277
- ip
 - WdsIpAddressInfoReq, 1185
- ipAddress
 - pack_wds_SetDefaultProfile_t, 623
- ipFamily
 - _packetSrvStatus, 64
 - unpack_wds_SLQSSetPacketSrvStatusCallback↔_t, 1080
- ipVersion
 - LibPackTFTIDParams, 347
 - TFTIDParams, 860
- ipaddr
 - unpack_wds_GetDefaultProfile_t, 1063
- ipaddrv6
 - unpack_wds_GetDefaultProfile_t, 1064
- ipfamily
 - ssdatasession_params, 826
- ipv4Address
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, 1085
- ipv4GWAddress
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, 1085
- ipv6Address
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, 1085
- ipv6AddressInfo, 319
 - IPAddressV6, 319
 - IPv6PrefixLen, 319
- ipv6GWAddress
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, 1085
- is856SysId
 - HDRSysInfo, 294
 - nas_HDRSysInfo, 436
- is856SysIdValid
 - HDRSysInfo, 294
 - nas_HDRSysInfo, 436
- is_DataRate_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_EspSpi_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_IPv4DstAddr_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_IPv4SrcAddr_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_IPv4Tos_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_IPv6DstAddr_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_IPv6Label_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_IPv6SrcAddr_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_IPv6TrafCls_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_Id_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_Jitter_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_Latency_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_LteBandCapability_Available
 - unpack_dms_SLQSGetBandCapability_t, 946
- is_LteQci_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_MaxAllowedPktSz_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_MinPolicedPktSz_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_NxtHdrProto_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_PktErrRate_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_Precedence_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_ProfileId3GPP2_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_RxQFlowGranted_Available
 - unpack_qos_QosFlowInfo_t, 1023
- is_TCPDstPort_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_TCPSrcPort_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_TdsBandCapability_Available
 - unpack_dms_SLQSGetBandCapability_t, 946
- is-TokenBucket_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_TrafficClass_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_TranDstPort_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_TranSrcPort_Available
 - unpack_qos_swiQosFilter_t, 1033
- is_TxQFlowGranted_Available
 - unpack_qos_QosFlowInfo_t, 1023
- is_UDPDstPort_Available

- unpack_qos_swiQosFilter_t, 1034
- is_UDPSrcPort_Available
 - unpack_qos_swiQosFilter_t, 1034
- is_val_3GPP2Pri_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_val_3GPPImCn_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_val_3GPPResResidualBER_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_val_3GPPSigInd_Available
 - unpack_qos_swiQosFlow_t, 1037
- is_val_3GPPTraHdlPri_Available
 - unpack_qos_swiQosFlow_t, 1037
- isEmpty
 - getAllCallInformation, 239
- isInTraffic
 - txInfo, 873
- isModByCC
 - SUPSInfo, 828
- isNewFlow
 - QosFlowInfoState, 718
 - unpack_qos_QosFlowInfoState_t, 1024
- isPrefDataPath
 - GSMSrvStatusInfo, 280
 - nas_GSMSrvStatusInfo, 428
 - nas_SrvStatusInfo, 475
 - SrvStatusInfo, 825
- isRadioTuned
 - nas_RxSigInfo, 468
 - rxInfo, 735
 - RxSigInfo, 738
- isSysForbidden
 - detailSvcInfo, 205
 - nas_detailSvcInfo, 421
 - nas_sysInfoCommon, 477
 - sysInfoCommon, 851
- isSysForbiddenValid
 - nas_sysInfoCommon, 477
 - sysInfoCommon, 851
- isSysPrIMatch
 - CDMASysInfo, 158
 - HDRSysInfo, 294
 - nas_CDMASysInfo, 414
 - nas_HDRSysInfo, 436
- isSysPrIMatchValid
 - CDMASysInfo, 158
 - HDRSysInfo, 294
 - nas_CDMASysInfo, 414
 - nas_HDRSysInfo, 436
- IsVoiceEnabled
 - pack_dms_SetCustFeature_t, 555
 - unpack_dms_GetCustFeature_t, 927
- Item
 - GetAudioPathConfigReq, 241
 - GetAudioVolTLBConfigReq, 245
 - SetAudioVolTLBConfigReq, 758
- Jitter
 - unpack_qos_swiQosFlow_t, 1037
- KeyExchange
 - protocolSubtypeElement, 676
- LBPTiv
 - NASQmiCbKnasSystemSelPrefInd, 521
- LEN
 - qaGobiApiDcs.h, 1406
- LIBPACK_MAX_QOS_FILTERS
 - qos.h, 1751
- LIBPACK_MAX_QOS_FLOW_PER_APN_STATS
 - qos.h, 1751
- LIBPACK_MAX_QOS_FLOWS
 - qos.h, 1751
- LIBPACK_MAX_SWIOMA_STR_LEN
 - swioma.h, 1772
- LIBPACK_NAS_LTE_CPHY_CA_BW_NRB
 - nas.h, 1285
- LIBPACK_NAS_LTE_CPHY_SCELL_STATE
 - nas.h, 1285
- LIBPACK_QMI_CBK_PARAM_NOCHANGE
 - sms.h, 1762
- LIBPACK_QMI_CBK_PARAM_RESET
 - sms.h, 1762
- LIBPACK_QMI_CBK_PARAM_SET
 - sms.h, 1762
- LOC_EVENT_MASK_ENG_STATE
 - qaGobiApiCbK.h, 1327
- LOC_EVENT_MASK_GNSS_SV_INFO
 - qaGobiApiCbK.h, 1327
- LOC_EVENT_MASK_INJECT_TIME
 - qaGobiApiCbK.h, 1327
- LOC_EVENT_MASK_SENSOR_STREAM
 - qaGobiApiCbK.h, 1327
- LOC_EVENT_MASK_TIME_SYNC
 - qaGobiApiCbK.h, 1327
- LOC_EVENT_POSITION_REPORT
 - qaGobiApiCbK.h, 1327
- LOC_UINT8_MAX_STRING_SZ
 - loc.h, 1264
- LOCEVENTMASKBATCHFULLNOTIFICATION
 - loc.h, 1264
- LOCEVENTMASKENGINESTATE
 - loc.h, 1264
- LOCEVENTMASKFIXSESSIONSTATE
 - loc.h, 1265
- LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION
 - loc.h, 1265
- LOCEVENTMASKGEOFENCEBREACHNOTIFICATION
 - loc.h, 1265
- LOCEVENTMASKGNSSMEASUREMENTREPORT
 - loc.h, 1265
- LOCEVENTMASKGNSSSVINFO
 - loc.h, 1265
- LOCEVENTMASKINJECTPOSITIONREQ
 - loc.h, 1265

- LOCEVENTMASKINJECTPREDICTEDORBITSREQ
 - loc.h, [1265](#)
- LOCEVENTMASKINJECTTIMERREQ
 - loc.h, [1265](#)
- LOCEVENTMASKINJECTWIFIAPDATAREQ
 - loc.h, [1266](#)
- LOCEVENTMASKINVALIDVALUE
 - loc.h, [1266](#)
- LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT
 - loc.h, [1266](#)
- LOCEVENTMASKLOCATIONSERVERCONNECTIO↔NREQ
 - loc.h, [1266](#)
- LOCEVENTMASKMOTIONDATACONTROL
 - loc.h, [1266](#)
- LOCEVENTMASKNIGEOFENCENOTIFICATION
 - loc.h, [1266](#)
- LOCEVENTMASKNINOTIFYVERIFYREQ
 - loc.h, [1266](#)
- LOCEVENTMASKNMEA
 - loc.h, [1266](#)
- LOCEVENTMASKPEDOMETERCONTROL
 - loc.h, [1266](#)
- LOCEVENTMASKPOSITIONREPORT
 - loc.h, [1267](#)
- LOCEVENTMASKSENSORSTREAMINGREADYST↔ATUS
 - loc.h, [1267](#)
- LOCEVENTMASKSETSPISTREAMINGREPORT
 - loc.h, [1267](#)
- LOCEVENTMASKTIMESYNCREQ
 - loc.h, [1267](#)
- LOCEVENTMASKVEHICLEDATAAREADYSTATUS
 - loc.h, [1267](#)
- LOCEVENTMASKWIFIREQ
 - loc.h, [1267](#)
- LOCEventRegisterReqResp, [366](#)
 - eventRegister, [368](#)
- LOCExtPowerStateReqResp, [368](#)
 - extPowerState, [368](#)
- LOCStartReq, [374](#)
 - pApplicationInfo, [376](#)
 - pConfigAltitudeAssumed, [376](#)
 - pHorizontalAccuracyLvl, [376](#)
 - pIntermediateReportState, [376](#)
 - pMinIntervalTime, [376](#)
 - pRecurrenceType, [376](#)
 - SessionId, [376](#)
- LOCStopReq, [376](#)
 - sessionId, [376](#)
- LPCSTR
 - SwiDataTypes.h, [1769](#)
- LTEAttachProfile
 - unpack_wds_SLQSGet3GPPConfigItem_t, [1073](#)
- LTEAttachProfileList
 - unpack_wds_SLQSGet3GPPConfigItem_t, [1073](#)
- LTEAttachProfileListLen
 - _slqs3GPPConfigItem, [69](#)
- pack_wds_SLQSSet3GPPConfigItem_t, [633](#)
- unpack_wds_SLQSGet3GPPConfigItem_t, [1073](#)
- LTETBandPref
 - NASLTETBandPreferenceTlv, [509](#)
- LTETCphyCAInfo
 - unpack_nas_SlqsGetLTETCphyCAInfo_t, [992](#)
- LTEInfo, [379](#)
 - band, [381](#)
 - bandwidth, [381](#)
 - emmConnState, [381](#)
 - emmState, [381](#)
 - emmSubState, [381](#)
 - RXChan, [381](#)
 - TXChan, [381](#)
- LTEInfoInterfreq, [382](#)
 - freqsLen, [382](#)
 - InfoInterfreq, [382](#)
 - uelIdle, [382](#)
- LTEInfoIntrafreq, [382](#)
 - CellParams, [384](#)
 - cellReselPriority, [384](#)
 - cellsLen, [384](#)
 - earfcn, [384](#)
 - globalCellId, [384](#)
 - plmn, [384](#)
 - sIntraSearch, [384](#)
 - sNonIntraSearch, [384](#)
 - servingCellId, [384](#)
 - tac, [384](#)
 - threshServingLow, [384](#)
 - uelIdle, [384](#)
- LTEInfoNeighboringGSM, [384](#)
 - freqsLen, [385](#)
 - LteGsmCellInfo, [385](#)
 - uelIdle, [385](#)
- LTEInfoNeighboringWCDMA, [385](#)
 - freqsLen, [386](#)
 - LTEWCDMACellInfo, [386](#)
 - uelIdle, [386](#)
- LTERSRPThresh, [388](#)
 - LTERSRPThreshListLen, [388](#)
 - pLTERSRPThreshList, [388](#)
- LTERSRPThreshListLen
 - LTERSRPThresh, [388](#)
 - nas_LTERSRPThresh, [446](#)
- LTERSQRThresh, [388](#)
 - LTERSQRThreshListLen, [389](#)
 - pLTERSQRThreshList, [389](#)
- LTERSQRThreshListLen
 - LTERSQRThresh, [389](#)
 - nas_LTERSQRThresh, [447](#)
- LTERSSIThresh, [389](#)
 - LTERSSIThreshListLen, [389](#)
 - pLTERSSIThreshList, [389](#)
- LTERSSIThreshListLen
 - LTERSSIThresh, [389](#)
 - nas_LTERSSIThresh, [447](#)
- LTESNRThresListLen

- LTESNRThresh, [393](#)
- LTESNRThresh, [392](#)
 - LTESNRThresListLen, [393](#)
 - pLTESNRThresList, [393](#)
- LTESNRThreshListLen
 - LTESNRThreshold, [393](#)
 - nas_LTESNRThreshold, [449](#)
- LTESNRThreshold, [393](#)
 - LTESNRThreshListLen, [393](#)
 - pLTESNRThreshList, [393](#)
- LTESInfo, [393](#)
 - rsrp, [394](#)
 - rsrq, [394](#)
 - rsqi, [394](#)
 - snr, [394](#)
 - unpack_nas_SLQSNasGetSigInfo_t, [1005](#)
- LTESigRptCfg, [390](#)
 - avgPeriod, [391](#)
 - rptRate, [391](#)
- LTESigRptConfig, [391](#)
 - avgPeriod, [391](#)
 - rptRate, [391](#)
- LTESysInfo, [395](#)
 - cellId, [397](#)
 - cellIdValid, [397](#)
 - lac, [397](#)
 - lacValid, [397](#)
 - MCC, [397](#)
 - MNC, [397](#)
 - networkIdValid, [397](#)
 - regRejectInfoValid, [397](#)
 - rejCause, [397](#)
 - rejectSrvDomain, [397](#)
 - sysInfoLTE, [397](#)
 - tac, [397](#)
 - tacValid, [397](#)
- LTEWCDMACellInfo
 - LTEInfoNeighboringWCDMA, [386](#)
 - nas_LTEInfoNeighboringWCDMA, [445](#)
- Lac
 - qaQmiServingSystemParam, [683](#)
 - unpack_nas_SLQSGetServingSystem_t, [995](#)
- lac
 - GERANInfo, [237](#)
 - GSMSysInfo, [283](#)
 - LTESysInfo, [397](#)
 - nas_GERANInfo, [425](#)
 - nas_GSMSysInfo, [431](#)
 - nas_LTESysInfo, [451](#)
 - nas_UMTSInfo, [484](#)
 - nas_WCDMASysInfo, [493](#)
 - UMTSInfo, [915](#)
 - WCDMASysInfo, [1157](#)
- lac1
 - OperatorPLMNData, [551](#)
- lac2
 - OperatorPLMNData, [551](#)
- lacValid
 - GSMSysInfo, [283](#)
 - LTESysInfo, [397](#)
 - nas_GSMSysInfo, [431](#)
 - nas_LTESysInfo, [451](#)
 - nas_WCDMASysInfo, [493](#)
 - WCDMASysInfo, [1157](#)
- language
 - CDMABroadcastConfig, [146](#)
- lastCallDataBearerTech
 - unpack_wds_SLQSGetDUNCallInfo_t, [1076](#)
- lastCallDataBearerTechnology
 - unpack_wds_SLQSGetDataBearerTechnology_t, [1075](#)
- lastCallRXOKBytesCnt
 - unpack_wds_SLQSGetDUNCallInfo_t, [1076](#)
- lastCallTXOKBytesCnt
 - unpack_wds_SLQSGetDUNCallInfo_t, [1076](#)
- LastErrCode
 - DataStatusDetail, [200](#)
- Latency
 - unpack_qos_swiQosFlow_t, [1038](#)
- Latitude
 - GPSSStateInfo, [277](#)
- latitude
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- leapSeconds
 - nas_qaQmi3Gpp2TimeZone, [462](#)
 - qaQmi3Gpp2TimeZone, [678](#)
- leaseState
 - wds_DHCPLeaseStateTlv, [1161](#)
- len
 - BdsSVInfo, [124](#)
 - loc_BdsSVInfo, [352](#)
 - loc_SVInfo, [363](#)
 - SVInfo, [830](#)
 - t_Sv, [852](#)
 - unpack_nas_GetSignalStrengths_t, [987](#)
- length
 - readTransparentInfo, [720](#)
 - SMSCAddress, [808](#)
 - sMSCAddress, [807](#)
 - SMSEtwSMessage, [809](#)
 - sMSEtwSMessage, [809](#)
 - SMSTransferRouteMTMessage, [822](#)
 - sMSTransferRouteMTMessage, [821](#)
 - uim_readTransparentInfo, [883](#)
- Level
 - GetM2MAudioVolumeResp, [263](#)
 - SetM2MAudioVolumeReq, [769](#)
- LibPackGPRSRequestedQoS, [320](#)
 - delayClass, [321](#)
 - meanThroughputClass, [321](#)
 - peakThroughputClass, [321](#)
 - precedenceClass, [321](#)
 - reliabilityClass, [321](#)
- LibPackQoSClassID, [344](#)
 - gDIBitRate, [345](#)
 - gUIBitRate, [345](#)

- maxDlBitRate, [345](#)
- maxUlBitRate, [345](#)
- QCI, [345](#)
- LibPackTFTIDParams, [345](#)
 - destPortRangeEnd, [347](#)
 - destPortRangeStart, [347](#)
 - eValid, [347](#)
 - filterId, [347](#)
 - flowLabel, [347](#)
 - IPSECSPi, [347](#)
 - ipVersion, [347](#)
 - nextHeader, [347](#)
 - pSourceIP, [347](#)
 - sourceIPMask, [347](#)
 - srcPortRangeEnd, [347](#)
 - srcPortRangeStart, [347](#)
 - tosMask, [347](#)
- LibPackUMTSQoS, [347](#)
 - deliveryErrSDU, [349](#)
 - grntDownlinkBitrate, [349](#)
 - grntUplinkBitrate, [349](#)
 - maxDownlinkBitrate, [349](#)
 - maxSDUSize, [349](#)
 - maxUplinkBitrate, [349](#)
 - qosDeliveryOrder, [349](#)
 - resBerRatio, [349](#)
 - sduErrorRatio, [349](#)
 - trafficClass, [349](#)
 - trafficPriority, [349](#)
 - transferDelay, [349](#)
- LibPackUMTSReqQoSsigInd, [349](#)
 - SigInd, [350](#)
 - UMTSReqQoS, [350](#)
- LibPackprofile_3GPP2, [339](#)
 - pAPNClass3GPP2, [343](#)
 - pAPNEnabled3GPP2, [343](#)
 - pAllowLinger, [343](#)
 - pApnString, [343](#)
 - pApnStringSize, [343](#)
 - pAppPriority, [343](#)
 - pAppType, [343](#)
 - pAuthPassword, [343](#)
 - pAuthPassword_tSize, [343](#)
 - pAuthProtocol, [343](#)
 - pAuthRetryCount, [343](#)
 - pAuthTimeout, [343](#)
 - pDataMode, [343](#)
 - pDataRate, [343](#)
 - plpcpAckTimeout, [343](#)
 - plpcpCreqRetryCount, [343](#)
 - plsPcscfAddressNedded, [344](#)
 - pLcpAckTimeout, [344](#)
 - pLcpCreqRetryCount, [344](#)
 - pNegoDnsSrvrPref, [344](#)
 - pPDNInactivTimeout3GPP2, [344](#)
 - pPdnType, [344](#)
 - pPppSessCloseTimer1x, [344](#)
 - pPppSessCloseTimerDO, [344](#)
 - pPriV6DnsAddress, [344](#)
 - pPrimaryV4DnsAddress, [344](#)
 - pRATType, [344](#)
 - pSecV6DnsAddress, [344](#)
 - pSecondaryV4DnsAddress, [344](#)
 - pUserId, [344](#)
 - pUserIdSize, [344](#)
- LibPackprofile_3GPP, [333](#)
 - pAPNClass, [337](#)
 - pAPNDisabledFlag, [337](#)
 - pAPNName, [337](#)
 - pAPNnameSize, [337](#)
 - pAddrAllocPref, [337](#)
 - pAuthenticationPref, [337](#)
 - pGPRSMMinimumQoS, [337](#)
 - pGPRSRequestedQos, [338](#)
 - pIPv4AddrPref, [338](#)
 - pIPv6AddPref, [338](#)
 - plmCnFlag, [338](#)
 - pPDNInactivTimeout, [338](#)
 - pPDPtype, [338](#)
 - pPassword, [338](#)
 - pPasswordSize, [338](#)
 - pPcscfAddrUsingDhcp, [338](#)
 - pPcscfAddrUsingPCO, [338](#)
 - pPdpAccessConFlag, [338](#)
 - pPdpContext, [338](#)
 - pPdpDataCompType, [338](#)
 - pPdpHdrCompType, [338](#)
 - pPriDNSIPv4AddPref, [338](#)
 - pPriDNSIPv6addpref, [338](#)
 - pPrimaryID, [338](#)
 - pProfilename, [338](#)
 - pProfilenameSize, [338](#)
 - pQosClassID, [338](#)
 - pSecDNSIPv4AddPref, [338](#)
 - pSecDNSIPv6addpref, [338](#)
 - pSecondaryFlag, [338](#)
 - pTFTID1Params, [339](#)
 - pTFTID2Params, [339](#)
 - pUMTSMInQoS, [339](#)
 - pUMTSMInQosSigInd, [339](#)
 - pUMTSReqQoSsigInd, [339](#)
 - pUMTSReqQoS, [339](#)
 - pUsername, [339](#)
 - pUsernameSize, [339](#)
- libpack_GetVersion
 - common.h, [1199](#)
- libpack_log
 - common.h, [1199](#)
- LibpackProfile3GPP2, [327](#)
 - pAPNClass3GPP2, [331](#)
 - pAPNEnabled3GPP2, [332](#)
 - pAllowLinger, [331](#)
 - pApnString, [332](#)
 - pApnStringSize, [332](#)
 - pAppPriority, [332](#)
 - pAppType, [332](#)

- pAuthPassword, [332](#)
- pAuthPasswordSize, [332](#)
- pAuthProtocol, [332](#)
- pAuthRetryCount, [332](#)
- pAuthTimeout, [332](#)
- pDataMode, [332](#)
- pDataRate, [332](#)
- plpcpAckTimeout, [332](#)
- plpcpCreqRetryCount, [332](#)
- plsPcscfAddressNedded, [332](#)
- pLcpAckTimeout, [332](#)
- pLcpCreqRetryCount, [332](#)
- pNegoDnsSrvrPref, [332](#)
- pPDNInactivTimeout3GPP2, [332](#)
- pPdnType, [332](#)
- pPppSessCloseTimer1x, [332](#)
- pPppSessCloseTimerDO, [332](#)
- pPriV6DnsAddress, [333](#)
- pPrimaryV4DnsAddress, [332](#)
- pRATType, [333](#)
- pSecV6DnsAddress, [333](#)
- pSecondaryV4DnsAddress, [333](#)
- pUserId, [333](#)
- pUserIdSize, [333](#)
- LibpackProfile3GPP, [321](#)
 - pAPNClass, [326](#)
 - pAPNDisabledFlag, [326](#)
 - pAPNName, [326](#)
 - pAPNnameSize, [326](#)
 - pAddrAllocPref, [326](#)
 - pAuthenticationPref, [326](#)
 - pGPRSMinimumQoS, [326](#)
 - pGPRSRequestedQoS, [326](#)
 - pIPv4AddrPref, [326](#)
 - pIPv6AddPref, [326](#)
 - plmCnFlag, [326](#)
 - pPDNInactivTimeout, [326](#)
 - pPDPTYPE, [327](#)
 - pPassword, [326](#)
 - pPasswordSize, [326](#)
 - pPcscfAddrUsingDhcp, [326](#)
 - pPcscfAddrUsingPCO, [326](#)
 - pPdpAccessConFlag, [326](#)
 - pPdpContext, [326](#)
 - pPdpDataCompType, [326](#)
 - pPdpHdrCompType, [327](#)
 - pPriDNSIPv4AddPref, [327](#)
 - pPriDNSIPv6addpref, [327](#)
 - pPrimaryID, [327](#)
 - pProfileName, [327](#)
 - pProfileNameSize, [327](#)
 - pQoSClassID, [327](#)
 - pSecDNSIPv4AddPref, [327](#)
 - pSecDNSIPv6addpref, [327](#)
 - pSecondaryFlag, [327](#)
 - pTFTID1Params, [327](#)
 - pTFTID2Params, [327](#)
 - pUMTSMInQoS, [327](#)
 - pUMTSMInQoS, [327](#)
 - pUMTSReqQoS, [327](#)
 - pUMTSReqQoS, [327](#)
 - pUsername, [327](#)
 - pUsernameSize, [327](#)
- lineCtrlInfo, [350](#)
 - polarityIncluded, [351](#)
 - pwrDenialTime, [351](#)
 - revPolarity, [351](#)
 - toggleMode, [351](#)
- lineValue
 - voiceALSSelectLineInfo, [1094](#)
- linkage
 - altSrcInfo_t, [105](#)
 - altitudeSrcInfo, [104](#)
- list_type
 - custSettingList, [192](#)
 - DMScustSettingList, [212](#)
 - DMSgetCustomInput, [213](#)
 - getCustomInput, [248](#)
 - pack_dms_GetCustFeaturesV2_t, [552](#)
- listEntries
 - FMSPrefImageList, [235](#)
 - PrefImageList, [661](#)
- listSize
 - FMSImageList, [234](#)
 - FMSPrefImageList, [235](#)
 - ImageList, [299](#)
 - PrefImageList, [661](#)
- loc.h, [1262](#)
 - eQMI_LOC_SESS_STATUS_FAILURE, [1267](#)
 - eQMI_LOC_SESS_STATUS_IN_PROGRESS, [1267](#)
 - eQMI_LOC_SESS_STATUS_SUCCESS, [1267](#)
 - eQMI_LOC_SESS_STATUS_TIMEOUT, [1267](#)
 - LOC_UINT8_MAX_STRING_SZ, [1264](#)
 - LOCEVENTMASKBATCHFULLNOTIFICATION, [1264](#)
 - LOCEVENTMASKENGINESTATE, [1264](#)
 - LOCEVENTMASKFIXSESSIONSTATE, [1265](#)
 - LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION, [1265](#)
 - LOCEVENTMASKGEOFENCEBREACHNOTIFICATION, [1265](#)
 - LOCEVENTMASKGEOFENCEGENALERT, [1265](#)
 - LOCEVENTMASKGNSSMEASUREMENTREPORT, [1265](#)
 - LOCEVENTMASKGNSSSVINFO, [1265](#)
 - LOCEVENTMASKINJECTPOSITIONREQ, [1265](#)
 - LOCEVENTMASKINJECTPREDICTEDORBITSREQ, [1265](#)
 - LOCEVENTMASKINJECTTIMERREQ, [1265](#)
 - LOCEVENTMASKINJECTWIFIAPDATAREQ, [1266](#)
 - LOCEVENTMASKINVALIDVALUE, [1266](#)
 - LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT, [1266](#)

- LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ, [1266](#)
- LOCEVENTMASKMOTIONDATACONTROL, [1266](#)
- LOCEVENTMASKNIGEOFENCENOTIFICATION, [1266](#)
- LOCEVENTMASKNINOTIFYVERIFYREQ, [1266](#)
- LOCEVENTMASKNMEA, [1266](#)
- LOCEVENTMASKPEDOMETERCONTROL, [1266](#)
- LOCEVENTMASKPOSITIONREPORT, [1267](#)
- LOCEVENTMASKSENSORSTREAMINGREADYSTATUS, [1267](#)
- LOCEVENTMASKSETSPSTREAMINGREPORT, [1267](#)
- LOCEVENTMASKTIMESYNCREQ, [1267](#)
- LOCEVENTMASKVEHICLEDATAREADYSTATUS, [1267](#)
- LOCEVENTMASKWIFIREQ, [1267](#)
- MAX_SENSOR_DATA_LEN, [1267](#)
- MAX_TEMP_DATA_LEN, [1267](#)
- pack_loc_DeleteAssistData, [1268](#)
- pack_loc_EventRegister, [1268](#)
- pack_loc_SLQSLOCGetBestAvailPos, [1269](#)
- pack_loc_SLQSLOCInjectPosition, [1269](#)
- pack_loc_SLQSLOCInjectSensorData, [1270](#)
- pack_loc_SLQSLOCInjectUTCTime, [1270](#)
- pack_loc_SLQSLOCSetCradleMountConfig, [1271](#)
- pack_loc_SetExtPowerState, [1268](#)
- pack_loc_SetOperationMode, [1269](#)
- pack_loc_Start, [1271](#)
- pack_loc_Stop, [1271](#)
- unpack_loc_BestAvailPos_Ind, [1272](#)
- unpack_loc_DeleteAssistData, [1272](#)
- unpack_loc_DeleteAssistData_Ind, [1272](#)
- unpack_loc_EngineState_Ind, [1273](#)
- unpack_loc_EventRegister, [1273](#)
- unpack_loc_GnssSvInfo_Ind, [1274](#)
- unpack_loc_PositionRpt_Ind, [1274](#)
- unpack_loc_SLQSLOCGetBestAvailPos, [1276](#)
- unpack_loc_SLQSLOCInjectPosition, [1276](#)
- unpack_loc_SLQSLOCInjectSensorData, [1277](#)
- unpack_loc_SLQSLOCInjectUTCTime, [1277](#)
- unpack_loc_SLQSLOCSetCradleMountConfig, [1277](#)
- unpack_loc_SetExtPowerConfig_Ind, [1274](#)
- unpack_loc_SetExtPowerState, [1275](#)
- unpack_loc_SetOperationMode, [1275](#)
- unpack_loc_SetOperationMode_Ind, [1276](#)
- unpack_loc_Start, [1278](#)
- unpack_loc_Stop, [1278](#)
- loc_BdsSVInfo, [351](#)
 - len, [352](#)
 - pSV, [352](#)
- loc_BdsSV, [351](#)
 - id, [351](#)
 - mask, [351](#)
- loc_CellDb, [352](#)
 - mask, [353](#)
- loc_ClkInfo, [353](#)
 - mask, [354](#)
- loc_GnssData, [354](#)
 - mask, [355](#)
- loc_LocApplicationInfo, [356](#)
 - appNameLength, [357](#)
 - appProviderLength, [357](#)
 - appVersionLength, [357](#)
 - appVersionValid, [357](#)
 - pAppName, [357](#)
 - pAppProvider, [357](#)
 - pAppVersion, [357](#)
- loc_SVInfo, [362](#)
 - len, [363](#)
 - pSV, [363](#)
- loc_SV, [361](#)
 - id, [362](#)
 - mask, [362](#)
 - system, [362](#)
- loc_gpsTime, [355](#)
 - gpsTimeOfWeekMs, [356](#)
 - gpsWeek, [356](#)
- loc_precisionDilution, [357](#)
 - HDOP, [358](#)
 - PDOP, [358](#)
 - VDOP, [358](#)
- loc_satelliteInfo, [358](#)
 - azimuth, [360](#)
 - elevation, [360](#)
 - gnssSvId, [360](#)
 - healthStatus, [360](#)
 - snr, [360](#)
 - svInfoMask, [360](#)
 - svListLen, [360](#)
 - svStatus, [360](#)
 - system, [360](#)
 - validMask, [360](#)
- loc_sensorDataUsage, [360](#)
 - aidingIndicatorMask, [361](#)
 - usageMask, [361](#)
- loc_svUsedforFix, [363](#)
 - gnssSvUsedList, [363](#)
 - gnssSvUsedList_len, [363](#)
- LocApplicationInfo, [363](#)
 - appNameLength, [364](#)
 - appProviderLength, [364](#)
 - appVersionLength, [364](#)
 - appVersionValid, [364](#)
 - pAppName, [364](#)
 - pAppProvider, [364](#)
 - pAppVersion, [364](#)
- LocDelAssDataReq, [365](#)
 - pBdsSVInfo, [365](#)
 - pCellDb, [365](#)
 - pClkInfo, [365](#)
 - pGnssData, [365](#)
 - pSVInfo, [365](#)
- LocInjectPositionReq, [368](#)
 - pAltitudeSrcInfo, [372](#)

- pAltitudeWrtEllipsoid, [372](#)
- pAltitudeWrtMeanSeaLevel, [372](#)
- pHorConfidence, [372](#)
- pHorReliability, [372](#)
- pHorUncCircular, [372](#)
- pLatitude, [372](#)
- pLongitude, [372](#)
- pPositionSrc, [372](#)
- pRawHorConfidence, [372](#)
- pRawHorUncCircular, [372](#)
- pTimestampAge, [372](#)
- pTimestampUtc, [372](#)
- pVertConfidence, [372](#)
- pVertReliability, [372](#)
- pVertUnc, [372](#)
- LocInjectSensorDataReq, [372](#)
 - pAcceleroData, [373](#)
 - pAcceleroTempData, [373](#)
 - pAcceleroTimeSrc, [373](#)
 - pGyroData, [373](#)
 - pGyroTempData, [374](#)
 - pGyroTimeSrc, [374](#)
 - pOpaqueIdentifier, [374](#)
- LocSetCradleMountReq, [374](#)
 - pConfidence, [374](#)
 - state, [374](#)
- localTimeOffset
 - nas_qaQmi3Gpp2TimeZone, [462](#)
 - qaQmi3Gpp2TimeZone, [678](#)
- Location Service(LOC), [53](#)
- logString
 - unpack_dms_SLQSSwiGetFwUpdateStatus_t, [951](#)
- logger
 - common.h, [1197](#)
- longName
 - nasPLMNNNameResp, [519](#)
 - PLMNNNetworkNameData, [659](#)
 - unpack_nas_SLQSGetPLMNNName_t, [993](#)
- longNameCi
 - nasPLMNNNameResp, [519](#)
 - unpack_nas_SLQSGetPLMNNName_t, [993](#)
- longNameEn
 - nasPLMNNNameResp, [519](#)
 - unpack_nas_SLQSGetPLMNNName_t, [993](#)
- longNameLen
 - nasPLMNNNameResp, [519](#)
 - PLMNNNetworkNameData, [659](#)
 - unpack_nas_SLQSGetPLMNNName_t, [993](#)
- longNameSB
 - nasPLMNNNameResp, [519](#)
 - unpack_nas_SLQSGetPLMNNName_t, [993](#)
- longNameSpareBits
 - PLMNNNetworkNameData, [659](#)
- Longitude
 - GPSSStateInfo, [277](#)
- longitude
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- loopbackMode
 - pack_wds_SLQSSSetLoopback_t, [637](#)
- loopbackMultiplier
 - pack_wds_SLQSSSetLoopback_t, [637](#)
- LteBandCapability
 - unpack_dms_SLQSGetBandCapability_t, [946](#)
- LteCQIParm, [377](#)
 - CQIValueCW0, [377](#)
 - CQIValueCW1, [377](#)
 - ValidityCW0, [377](#)
 - ValidityCW1, [377](#)
- IteEARFCN, [377](#)
 - earfcn0, [378](#)
 - earfcn1, [378](#)
 - status, [378](#)
- IteEmmDI
 - NasSwiIndReg, [526](#)
 - pack_nas_SLQSNasSwiIndicationRegister_t, [594](#)
- IteEmmUI
 - NasSwiIndReg, [526](#)
 - pack_nas_SLQSNasSwiIndicationRegister_t, [594](#)
- IteEsmDI
 - NasSwiIndReg, [526](#)
 - pack_nas_SLQSNasSwiIndicationRegister_t, [594](#)
- IteEsmUI
 - NasSwiIndReg, [526](#)
 - pack_nas_SLQSNasSwiIndicationRegister_t, [594](#)
- LteGsmCellInfo
 - LTEInfoNeighboringGSM, [385](#)
 - nas_LTEInfoNeighboringGSM, [444](#)
- IteGsmCellInfo, [378](#)
 - cellReselPriority, [379](#)
 - cells_len, [379](#)
 - GsmCellInfo, [379](#)
 - nccPermitted, [379](#)
 - threshGsmHigh, [379](#)
 - threshGsmLow, [379](#)
- LteNasReleaseInfo
 - qaGobiApiCbk.h, [1330](#)
- LteNasReleaseInfo_s, [386](#)
 - nas_major, [386](#)
 - nas_minor, [386](#)
 - nas_release, [386](#)
- ItePCI, [386](#)
 - earfcn, [387](#)
 - pci, [387](#)
 - status, [387](#)
- LteQci
 - unpack_qos_swiQosFlow_t, [1038](#)
- IteRsrpDelta
 - nas_SLQSSignalStrengthsIndReq, [473](#)
 - SLQSSignalStrengthsIndReq, [801](#)
- IteRsrpinfo
 - nas_SLQSSignalStrengthsInformation, [474](#)
 - SLQSSignalStrengthsInformation, [803](#)
- IteRsrpinformation, [387](#)
 - rsrplevel, [387](#)
- IteSSInfo, [394](#)
 - rsrp, [395](#)

- rsrq, 395
- rssi, 395
- snr, 395
- LteSccRxInfoResp, 389
 - pSccRxInfo, 390
- LteSnrDelta
 - nas_SLQSSignalStrengthsIndReq, 473
 - SLQSSignalStrengthsIndReq, 801
- LteSnrinfo
 - nas_SLQSSignalStrengthsInformation, 474
 - SLQSSignalStrengthsInformation, 803
- LteSnrinformation, 392
 - snrlevel, 392
- LteWcdmaCellInfo, 397
 - cellReselPriority, 398
 - cellsLen, 398
 - threshXhigh, 398
 - threshXlow, 398
 - uarfcn, 398
 - WCDMACellInfo, 398
- ltersrp
 - slqsSignalStrengthInfo, 799
 - unpack_nas_SLQSGetSignalStrength_t, 997
- ltesnr
 - slqsSignalStrengthInfo, 799
 - unpack_nas_SLQSGetSignalStrength_t, 997
- m_FwBuildId
 - CarrierImage_t, 140
 - SWI_STRUCT_CarrierImage, 831
- m_FwImageld
 - CarrierImage_t, 140
 - SWI_STRUCT_CarrierImage, 831
- m_PriBuildId
 - CarrierImage_t, 141
 - SWI_STRUCT_CarrierImage, 832
- m_PrImageld
 - CarrierImage_t, 141
 - SWI_STRUCT_CarrierImage, 832
- m_nCarrierId
 - CarrierImage_t, 141
 - SWI_STRUCT_CarrierImage, 832
- m_nFolderId
 - CarrierImage_t, 141
 - SWI_STRUCT_CarrierImage, 832
- m_nStorage
 - CarrierImage_t, 141
 - SWI_STRUCT_CarrierImage, 832
- MACIndex
 - NetworkStatEVDO, 538
- MAX_ACTIVE_PERS_FEATURES
 - qaGobiApiUim.h, 1661
- MAX_BUILD_ID_LEN
 - dms.h, 1208
- MAX_CALL_NO_LEN
 - qaGobiApiVoice.h, 1676
- MAX_CDMA_ENC_MO_TXT_MSG_SIZE
 - sms.h, 1760
- MAX_CONTENT_LENGTH
 - qaGobiApiUim.h, 1661
- MAX_CUST_ID_LEN
 - qaGobiApiDms.h, 1417
- MAX_CUST_VALUE_LEN
 - qaGobiApiDms.h, 1417
- MAX_DATA_SRV_CAPABILITIES
 - qaGobiApiNas.h, 1499
- MAX_DESCRIPTION_LENGTH
 - qaGobiApiNas.h, 1499
 - qaGobiApiUim.h, 1661
 - qaGobiApiVoice.h, 1676
 - qaNasPerformNetworkScan.h, 1741
 - uim.h, 1782
- MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH
 - qaGobiApiDms.h, 1417
- MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH
 - qaGobiApiDms.h, 1417
- MAX_FSN_LENGTH
 - qaGobiApiDms.h, 1417
- MAX_ICCID_LENGTH
 - qaGobiApiUim.h, 1661
 - uim.h, 1782
- MAX_IMAGE_IDE_ELEMENTS
 - qaGobiApiFms.h, 1459
- MAX_LEN_IFACE_TABLE
 - qaGobiApiSwiAudio.h, 1601
- MAX_MITIGATION_DEV_ID_LEN
 - qaGobiApiCbk.h, 1327
 - qaGobiApiTmd.h, 1657
- MAX_MITIGATION_DEV_LIST_LEN
 - qaGobiApiTmd.h, 1657
- MAX_MS_TRANSFER_ROUTE_MSG
 - sms.h, 1760
- MAX_MSC_ADDRESS_SIZE
 - sms.h, 1760
- MAX_MSE_TWS_MSG
 - sms.h, 1760
- MAX_NO_OF_APPLICATIONS
 - qaGobiApiCbk.h, 1327
 - qaGobiApiUim.h, 1661
 - uim.h, 1782
- MAX_NO_OF_CALLS
 - qaGobiApiCbk.h, 1327
 - qaGobiApiVoice.h, 1676
- MAX_NO_OF_FILES
 - qaGobiApiCbk.h, 1327
- MAX_NO_OF_SLOTS
 - qaGobiApiCbk.h, 1327
 - qaGobiApiUim.h, 1661
 - uim.h, 1782
- MAX_NO_OF_UUSINFO
 - qaGobiApiCbk.h, 1327
- MAX_PATH_LENGTH
 - qaGobiApiCbk.h, 1327
 - qaGobiApiUim.h, 1661
- MAX_PILOT_SETS
 - qaGobiApiNas.h, 1499
- MAX_PUK_LENGTH

- qaGobiApiUim.h, 1661
- MAX_QOS_FILTER_TLV
 - qaGobiApiQos.h, 1565
- MAX_QOS_SPEC_PER_APN
 - qaGobiApiQos.h, 1565
- MAX_RADIO_INTERFACE_LIST
 - qaGobiApiCbk.h, 1327
- MAX_SENSOR_DATA_LEN
 - loc.h, 1267
 - qaGobiApiLoc.h, 1487
- MAX_SERV_SYSTEM_RADIO_INTERFACES
 - qaGobiApiNas.h, 1499
- MAX_SLOTS_STATUS
 - qaGobiApiUim.h, 1661
 - uim.h, 1782
- MAX_SMS_LIST_SIZE
 - sms.h, 1760
- MAX_SMS_MESSAGE_SIZE
 - sms.h, 1760
- MAX_SMS_ROUTES
 - qaGobiApiSms.h, 1577
- MAX_TEMP_DATA_LEN
 - loc.h, 1267
 - qaGobiApiLoc.h, 1487
- MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE↔
 - _LIST_SIZE
 - wds.h, 1798
- MAXUSSDLENGTH
 - qaGobiApiCbk.h, 1327
 - qaGobiApiVoice.h, 1676
- MCC
 - _SlqsNas3GppNetworkRAT_, 70
 - CDMASysInfo, 158
 - CDMASysInfoExt, 159
 - currentPLMN, 183
 - GSMSysInfo, 283
 - LTESysInfo, 397
 - nas_CDMASysInfo, 414
 - nas_CDMASysInfoExt, 415
 - nas_GSMSysInfo, 431
 - nas_LTESysInfo, 451
 - nas_QmiNas3GppNetworkInfo, 462
 - nas_QmiNas3GppNetworkRAT, 463
 - nas_QmisNasPcsDigit, 464
 - nas_WCDMASysInfo, 493
 - nas_currentPLMN, 419
 - SlqsNas3GppNetworkInfo, 792
 - SlqsNasPcsDigit, 793
 - unpack_nas_GetServingNetwork_t, 985
 - WCDMASysInfo, 1157
- MDMCallDuration
 - ConnectionStatus, 170
 - connectionStatus, 170
- MDMConnStatus
 - ConnectionStatus, 170
 - connectionStatus, 170
- MEID_MAX_SIZE
 - dms.h, 1208
- MEIDString
 - unpack_dms_GetDeviceSerialNumbers_t, 930
- MINREQBKLEN
 - common.h, 1197
- MIN
 - unpack_dms_GetVoiceNumber_t, 938
- MMTlv
 - unpack_sms_SetNewSMSCallback_ind_t, 1043
- MNRInfo, 401
 - mcc, 402
 - mnc, 402
 - rat, 402
- MNC
 - _SlqsNas3GppNetworkRAT_, 70
 - CDMASysInfo, 158
 - currentPLMN, 183
 - GSMSysInfo, 283
 - LTESysInfo, 397
 - nas_CDMASysInfo, 414
 - nas_GSMSysInfo, 431
 - nas_LTESysInfo, 451
 - nas_QmiNas3GppNetworkInfo, 462
 - nas_QmiNas3GppNetworkRAT, 463
 - nas_QmisNasPcsDigit, 464
 - nas_WCDMASysInfo, 494
 - nas_currentPLMN, 419
 - SlqsNas3GppNetworkInfo, 792
 - SlqsNasPcsDigit, 793
 - unpack_nas_GetServingNetwork_t, 985
 - WCDMASysInfo, 1157
- MPTlv
 - NASQmiCbkNasSystemSelPrefInd, 521
- MSGID_AND_LEN
 - common.h, 1197
- MSGID_DONT_CARE
 - common.h, 1197
- MTMessageInfo
 - newMTMessageTlv, 538
- manString
 - pack_dms_SLQSSwiSetHostDevInfo_t, 558
 - unpack_dms_SLQSSwiGetHostDevInfo_t, 952
- manufacturer
 - unpack_dms_GetManufacturer_t, 934
- Mask
 - getDUNCallInfoReq, 249
 - pack_wds_SLQSGetDUNCallInfo_t, 629
- mask
 - BdsSV, 123
 - CellDb, 159
 - ClkInfo, 165
 - GnssData, 270
 - IPv6TrafCls, 320
 - loc_BdsSV, 351
 - loc_CellDb, 353
 - loc_ClkInfo, 354
 - loc_GnssData, 355
 - loc_SV, 362
 - SV, 829

- Tos, [867](#)
- unpack_qos_IPv6TrafCls_t, [1020](#)
- unpack_qos_Tos_t, [1039](#)
- max_channel_rx_rate
 - unpack_wds_SLQSGetCurrentChannelRate_t, [1074](#)
 - WDSSWICurrentChannelRates, [1193](#)
- max_channel_tx_rate
 - unpack_wds_SLQSGetCurrentChannelRate_t, [1074](#)
 - WDSSWICurrentChannelRates, [1193](#)
- max_dist
 - pack_swiloc_SwiLocSetAutoStart_t, [608](#)
 - SwiLocGetAutoStartResp, [833](#)
 - SwiLocSetAutoStartReq, [835](#)
 - unpack_swiloc_SwiLocGetAutoStart_t, [1047](#)
- max_dist_reported
 - SwiLocGetAutoStartResp, [833](#)
 - unpack_swiloc_SwiLocGetAutoStart_t, [1047](#)
- max_time
 - pack_swiloc_SwiLocSetAutoStart_t, [608](#)
 - SwiLocGetAutoStartResp, [833](#)
 - SwiLocSetAutoStartReq, [835](#)
 - unpack_swiloc_SwiLocGetAutoStart_t, [1047](#)
- max_time_reported
 - SwiLocGetAutoStartResp, [833](#)
 - unpack_swiloc_SwiLocGetAutoStart_t, [1047](#)
- MaxAllowedPktSz
 - unpack_qos_swiQosFlow_t, [1038](#)
- MaxChanRxRate
 - ChannelRate, [162](#)
 - dunchannelRate, [219](#)
- MaxChanTxRate
 - ChannelRate, [162](#)
 - dunchannelRate, [219](#)
- maxChannelRXRate
 - unpack_wds_GetConnectionRate_t, [1062](#)
- maxChannelTXRate
 - unpack_wds_GetConnectionRate_t, [1062](#)
- maxDIBitRate
 - LibPackQosClassID, [345](#)
 - QosClassID, [715](#)
- maxDownlinkBitrate
 - LibPackUMTSQoS, [349](#)
 - UMTSMinQoS, [919](#)
 - UMTSQoS, [922](#)
 - wds_UMTSMinQoS, [1174](#)
- maxImages
 - FMSImageIDEntries, [234](#)
 - ImageIDEntries, [298](#)
- maxMitigationLevel
 - mitigationDevList, [401](#)
- MaxRXChannelRate
 - unpack_dms_GetDeviceCap_t, [928](#)
- maxRxChannelRate
 - unpack_dms_GetDeviceCapabilities_t, [929](#)
- maxSDUSize
 - LibPackUMTSQoS, [349](#)
 - UMTSMinQoS, [919](#)
 - UMTSQoS, [922](#)
 - wds_UMTSMinQoS, [1174](#)
- maxStorageSize
 - smsMaxStorageSizeResp, [814](#)
- MaxTXChannelRate
 - unpack_dms_GetDeviceCap_t, [928](#)
- maxTxChannelRate
 - unpack_dms_GetDeviceCapabilities_t, [929](#)
- maxUIBitRate
 - LibPackQosClassID, [345](#)
 - QosClassID, [715](#)
- maxUplinkBitrate
 - LibPackUMTSQoS, [349](#)
 - UMTSMinQoS, [919](#)
 - UMTSQoS, [922](#)
 - wds_UMTSMinQoS, [1174](#)
- mcTimeStamp
 - cdmaMsgDecodingParams, [150](#)
- mcc
 - CSGID, [178](#)
 - MNRInfo, [402](#)
 - nas_CSGID, [418](#)
 - nas_MNRInfo, [453](#)
 - nas_netSelectionPref, [454](#)
 - nasPLMNNameReq, [517](#)
 - netSelectionPref, [532](#)
 - OperatorPLMNData, [551](#)
 - pack_nas_SLQSGetPLMNName_t, [584](#)
 - unpack_nas_GetHomeNetwork_t, [983](#)
- mccM
 - minBasedIMSI, [400](#)
- mccT
 - trueIMSI, [871](#)
- mdmCallDurationActive
 - unpack_wds_SLQSGetDUNCallInfo_t, [1076](#)
- MdmConnStatus
 - DUNCallInfoInd, [218](#)
- meanThroughputClass
 - GPRSQoS, [272](#)
 - GPRSRequestedQoS, [273](#)
 - LibPackGPRSRequestedQoS, [321](#)
 - wds_GPRSQoS, [1166](#)
- meid
 - unpack_dms_GetSerialNumbers_t, [938](#)
- meidLength
 - _SLQSSwiGetSerialNoExtParams, [81](#)
- meidSize
 - serialNumbersInfo, [748](#)
 - unpack_dms_GetDeviceSerialNumbers_t, [930](#)
- meidString
 - unpack_dms_SLQSSwiGetSerialNoExt_t, [953](#)
- message
 - unpack_sms_SLQSGetSMS_t, [1044](#)
- message_type
 - NASOTAMessageTlv, [512](#)
- messageClass
 - smsRouteEntry, [819](#)

- messageFailureCode
 - slqssendsmsparams_s, [796](#)
 - unpack_sms_SendSMS_t, [1042](#)
- messageFormat
 - pack_sms_SendSMS_t, [603](#)
 - slqssendasyncsmsparams_s, [794](#)
 - slqssendsmsparams_s, [796](#)
 - unpack_sms_SLQSGetSMS_t, [1044](#)
- messageID
 - SMSAsyncRawSend_s, [806](#)
 - slqssendsmsparams_s, [796](#)
 - unpack_sms_SendSMS_t, [1042](#)
- messageId
 - cdmaMsgEncodingParams, [152](#)
- messageIndex
 - pack_sms_SLQSGetSMS_t, [605](#)
 - pack_sms_SLQSMModifySMSStatus_t, [606](#)
 - qmiSmsMessageList, [709](#)
 - SMSMTMessage, [817](#)
 - sSMSMTMessage, [816](#)
- messageLength
 - cdmaMsgDecodingParams, [151](#)
- messageList
 - unpack_sms_SLQSGetSMSList_t, [1044](#)
- messageListSize
 - unpack_sms_SLQSGetSMSList_t, [1044](#)
- messageMode
 - SMSMemoryInfo, [814](#)
 - SMSMessageMode, [815](#)
 - sSMSMessageMode, [815](#)
 - unpack_sms_SLQSWmsMemoryFullCallBack_↔
ind_t, [1045](#)
- MessageModelInfo
 - messageModeTlv, [399](#)
- messageModeTlv, [398](#)
 - MessageModelInfo, [399](#)
 - TlvPresent, [399](#)
- messageSize
 - pack_sms_SendSMS_t, [603](#)
 - slqssendasyncsmsparams_s, [794](#)
 - slqssendsmsparams_s, [796](#)
 - unpack_sms_SLQSGetSMS_t, [1044](#)
 - wcdmaMsgEncodingParams, [1153](#)
- messageTag
 - pack_sms_SLQSMModifySMSStatus_t, [606](#)
 - qmiSmsMessageList, [709](#)
 - unpack_sms_SLQSGetSMS_t, [1044](#)
- messageType
 - smsRouteEntry, [819](#)
- messageWaitingInfoContent, [399](#)
 - activeInd, [399](#)
 - msgCount, [400](#)
 - msgType, [400](#)
- MicMute
 - GetAudioProfileResp, [245](#)
 - GetM2MAVMMuteResp, [264](#)
 - GetM2MAudioProfileResp, [262](#)
 - SetAudioProfileReq, [757](#)
 - SetM2MAVMMuteReq, [769](#)
- minBasedIMSI, [400](#)
 - imsiM1112, [400](#)
 - imsiMS1, [400](#)
 - imsiMS2, [400](#)
 - mccM, [400](#)
- MinPolicedPktSz
 - unpack_qos_swiQosFlow_t, [1038](#)
- minSize
 - unpack_dms_GetVoiceNumber_t, [938](#)
- minute
 - nas_UniversalTime, [487](#)
 - nas_timeInfo, [482](#)
 - timeInfo, [862](#)
 - UniversalTime, [925](#)
- mipMode
 - unpack_wds_GetMobileIP_t, [1065](#)
- mipStatus
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- mipstatAvail
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- mitigationDevIDLen
 - TmdDeRegNotMitigationLvlReq, [863](#)
 - TmdGetMitigationLvlReq, [864](#)
 - TmdMitigationLvlIndReq, [866](#)
 - TmdRegNotMitigationLvlReq, [866](#)
- mitigationDevID
 - TmdDeRegNotMitigationLvlReq, [863](#)
 - TmdGetMitigationLvlReq, [864](#)
 - TmdMitigationLvlIndReq, [866](#)
 - TmdRegNotMitigationLvlReq, [866](#)
- mitigationDevId
 - mitigationDevList, [401](#)
- mitigationDevIdLen
 - mitigationDevList, [401](#)
- MitigationDevInfo
 - QmiCbkTmdMitiLvlRptInd, [704](#)
- mitigationDevList, [400](#)
 - maxMitigationLevel, [401](#)
 - mitigationDevId, [401](#)
 - mitigationDevIdLen, [401](#)
- mnc
 - CSGID, [178](#)
 - MNRInfo, [402](#)
 - nas_CSGID, [418](#)
 - nas_MNRInfo, [453](#)
 - nas_netSelectionPref, [454](#)
 - nasPLMNNNameReq, [517](#)
 - netSelectionPref, [532](#)
 - OperatorPLMNData, [551](#)
 - pack_nas_SLQSGetPLMNNName_t, [584](#)
 - unpack_nas_GetHomeNetwork_t, [983](#)
- mncPcsDigits
 - CSGID, [178](#)
 - nas_CSGID, [418](#)
- mobileCountryCode

- SMSEtwsPlmn, [811](#)
- sMSEtwsPlmn, [811](#)
- mobileIP
 - pack_wds_SLQSSetWdsEventCallback_t, [635](#)
- mobileNetworkCode
 - SMSEtwsPlmn, [811](#)
 - sMSEtwsPlmn, [811](#)
- mode
 - callInfo, [136](#)
 - pack_dms_SetEventReport_t, [556](#)
 - pack_dms_SetPower_t, [556](#)
 - pack_loc_SetOperationMode_t, [571](#)
 - pack_wds_SetMobileIP_t, [624](#)
 - UIMRefreshEvent, [904](#)
- ModePref
 - NASModePreferenceTlv, [509](#)
- modelString
 - pack_dms_SLQSSwiSetHostDevInfo_t, [558](#)
 - unpack_dms_SLQSSwiGetHostDevInfo_t, [952](#)
- modelid
 - unpack_dms_GetModelID_t, [934](#)
- modelid_str
 - slqsfwinfno_s, [790](#)
 - unpack_dms_GetFirmwareInfo_t, [931](#)
- modemMode
 - CommInfo, [169](#)
 - nas_CommInfo, [417](#)
- modemTempNotification
 - qaGobiApiCbK.h, [1331](#)
- ModemTempState
 - _modemTempNotification, [63](#)
- ModemTemperature
 - _modemTempNotification, [62](#)
- modemindex
 - pack_fms_SetImagesPreference_t, [566](#)
- ModifyProfileIn, [402](#)
 - curProfile, [403](#)
 - pProfileID, [403](#)
 - pProfileType, [403](#)
- ModifyProfileOut, [403](#)
 - pExtErrorCode, [403](#)
- month
 - nas_UniversalTime, [487](#)
 - nas_timeInfo, [482](#)
 - timeInfo, [862](#)
 - UniversalTime, [925](#)
- msgCount
 - messageWaitingInfoContent, [400](#)
- msgDelFailureCause
 - SMSAsyncRawSend_s, [806](#)
- msgDelFailureType
 - SMSAsyncRawSend_s, [806](#)
- msgProtocol
 - smsMsgprotocolResp, [816](#)
- msgType
 - messageWaitingInfoContent, [400](#)
- msgWaitInfo
 - getMsgWaitingInfo, [266](#)
 - msgWaitingInfo, [404](#)
 - msgWaitInfo, [404](#)
 - numInstances, [404](#)
- msgid
 - pack_qmi_t, [600](#)
 - unpack_qmi_t, [1018](#)
- msgtype
 - common.h, [1198](#)
- Mtu
 - unpack_wds_SLQSSetRuntimeSettings_t, [1078](#)
- MultDisc
 - protocolSubtypeElement, [676](#)
- multiplier
 - pktErrRate, [657](#)
 - unpack_qos_pktErrRate_t, [1021](#)
- NAM_NAME_LENGTH
 - qaGobiApiNas.h, [1499](#)
- NAS_LTE_CPHY_CA_BW_NRB_LITE
 - nas.h, [1285](#)
- NAS_LTE_CPHY_CA_BW_NRB
 - qaGobiApiNas.h, [1511](#)
- NAS_LTE_CPHY_SCELL_STATE_LITE
 - nas.h, [1285](#)
- NAS_LTE_CPHY_SCELL_STATE
 - qaGobiApiNas.h, [1512](#)
- NAS_MAX_DESCRIPTION_LENGTH
 - nas.h, [1284](#)
- NAS_MAX_NUM_NETWORKS
 - nas.h, [1284](#)
- NAS_MAX_SCC_RX_INFO_INSTANCES
 - nas.h, [1284](#)
 - qaGobiApiNas.h, [1499](#)
- NAS_OTA_MESSAGE_MAX_BUF_SIZE
 - nas.h, [1284](#)
- NAS_PLMN_LENGTH
 - nas.h, [1284](#)
- NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INT↔
 - ERFACE_LIST
 - nas.h, [1284](#)
- NAS_SIG_INFO_MAX_TDSCDMA_THRESHOLDS↔
 - LIST_SIZE
 - qaGobiApiNas.h, [1499](#)
- NAS_SIG_INFO_MIN_dB_FLOAT_VALUE
 - qaGobiApiNas.h, [1499](#)
- NAS_SIG_INFO_MIN_dBm_FLOAT_VALUE
 - qaGobiApiNas.h, [1499](#)
- NAS_SRV
 - qaGobiApiCbK.h, [1327](#)
- NASBandPreferenceTlv, [494](#)
 - band_pref, [494](#)
 - TlvPresent, [494](#)
- NASEmergencyModeTlv, [495](#)
 - EmerMode, [496](#)
 - TlvPresent, [496](#)
- NASGWAcqOrderPrefTlv, [504](#)
 - GWAcqOrderPref, [504](#)
 - TlvPresent, [504](#)

- NASLTEBandPreferenceTlv, 508
 - LTEBandPref, 509
 - TlvPresent, 509
- NASLteNasReleaseInfoTlv, 509
 - nas_major, 509
 - nas_minor, 509
 - nas_release, 509
 - TlvPresent, 509
- NASModePreferenceTlv, 509
 - ModePref, 509
 - TlvPresent, 509
- NASNetSelPreferenceTlv, 509
 - NetSelPref, 510
 - TlvPresent, 510
- NASOTAMessageTlv, 512
 - data_buf, 512
 - data_len, 512
 - message_type, 512
 - TlvPresent, 512
- NASPRLPreferenceTlv, 520
 - PRLPref, 520
 - TlvPresent, 520
- NASPhyCaAggPcellInfo, 512
 - dl_bw_value, 513
 - freq, 513
 - iLTEbandValue, 513
 - pci, 513
 - TlvPresent, 513
- NASPhyCaAggScellIDBw, 513
 - dl_bw_value, 514
 - TlvPresent, 514
- NASPhyCaAggScellIndType, 514
 - freq, 515
 - pci, 515
 - scell_state, 515
 - TlvPresent, 515
- NASPhyCaAggScellIndex, 514
 - scell_idx, 514
 - TlvPresent, 514
- NASPhyCaAggScellInfo, 515
 - dl_bw_value, 516
 - freq, 516
 - iLTEbandValue, 516
 - pci, 516
 - scell_state, 516
 - TlvPresent, 516
- NASQmiCbkNasSwiOTAMessageInd, 520
 - nasRelInfoTlv, 520
 - otaMsgTlv, 520
 - timeTlv, 520
- NASQmiCbkNasSystemSelPrefInd, 520
 - BPTlv, 521
 - EMTlv, 521
 - GWAOPTlv, 521
 - LBPTlv, 521
 - MPTlv, 521
 - NSPTlv, 521
 - PRLPTlv, 521
 - RPTlv, 521
 - SDPTlv, 521
- NASRoamPreferenceTlv, 521
 - RoamPref, 521
 - TlvPresent, 521
- NASServDomainPrefTlv, 521
 - SrvDomainPref, 522
 - TlvPresent, 522
- NASServingSystemInfo, 522
 - csAttachState, 523
 - hdrPersonality, 523
 - psAttachState, 523
 - radioInterfaceList, 523
 - radioInterfaceNo, 523
 - registrationState, 523
 - selectedNetwork, 523
- NASTimeInfoTlv, 530
 - time, 531
 - TlvPresent, 531
- NAI
 - unpack_wds_GetMobileIPProfile_t, 1066
- NSPTlv
 - NASQmiCbkNasSystemSelPrefInd, 521
- NSSAudioCtrl, 541
 - downLink, 541
 - upLink, 541
- NUM_OF_SET
 - qaGobiApiCbk.h, 1327
 - qaGobiApiSms.h, 1577
- NWProfile, 541
 - pProfSz, 542
 - pProfValues, 542
 - tech, 542
- NWQoSStatus
 - unpack_qos_SLQSQosGetNetworkStatus_t, 1024
- NWRegStat
 - _transNWRegInfoNotification, 96
- naiSize
 - unpack_wds_GetMobileIPProfile_t, 1066
- namID
 - airTimer, 101
 - nasGet3GPP2SubscriptionInfoReq, 496
 - prefVoiceSO, 663
 - roamTimer, 729
- namName, 404
 - namName, 404
 - namNameLen, 404
- namNameLen
 - namName, 404
- Name
 - unpack_nas_GetServingNetwork_t, 986
- name
 - unpack_nas_GetHomeNetwork_t, 983
 - unpack_wds_GetDefaultProfile_t, 1064
- nameLen
 - remotePartyName, 724
- namePI
 - remotePartyName, 724

- nameSize
 - unpack_nas_GetServingNetwork_t, [986](#)
- nameString
 - pack_dms_SLQSSwiSetOSInfo_t, [559](#)
 - unpack_dms_SLQSSwiGetOSInfo_t, [952](#)
- namelength
 - omaDmFotaTlv, [547](#)
 - omaDmFotaTlvExt, [549](#)
 - unpack_omaDmFotaTlv_t, [1017](#)
- namesize
 - unpack_wds_GetDefaultProfile_t, [1064](#)
- nas.h, [1279](#)
 - eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_100, [1285](#)
 - eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_15, [1285](#)
 - eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_25, [1285](#)
 - eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_50, [1285](#)
 - eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_6, [1285](#)
 - eLIBPACK_NAS_LTE_CPHY_CA_BW_NRB_75, [1285](#)
 - eLIBPACK_NAS_LTE_CPHY_SCELL_STATE↔CONFIGURED_ACTIVATED, [1285](#)
 - eLIBPACK_NAS_LTE_CPHY_SCELL_STATE↔CONFIGURED_DEACTIVATED, [1285](#)
 - eLIBPACK_NAS_LTE_CPHY_SCELL_STATE↔DECONFIGURED, [1285](#)
 - eNAS_LTE_CPHY_CA_BW_NRB_LITE_100, [1285](#)
 - eNAS_LTE_CPHY_CA_BW_NRB_LITE_15, [1285](#)
 - eNAS_LTE_CPHY_CA_BW_NRB_LITE_25, [1285](#)
 - eNAS_LTE_CPHY_CA_BW_NRB_LITE_50, [1285](#)
 - eNAS_LTE_CPHY_CA_BW_NRB_LITE_6, [1285](#)
 - eNAS_LTE_CPHY_CA_BW_NRB_LITE_75, [1285](#)
 - eNAS_LTE_CPHY_SCELL_STATE_CONFIGU↔RED_ACTIVATED_LITE, [1285](#)
 - eNAS_LTE_CPHY_SCELL_STATE_CONFIGU↔RED_DEACTIVATED_LITE, [1285](#)
 - eNAS_LTE_CPHY_SCELL_STATE_DECONFI↔GURED_LITE, [1285](#)
 - LIBPACK_NAS_LTE_CPHY_CA_BW_NRB, [1285](#)
 - LIBPACK_NAS_LTE_CPHY_SCELL_STATE, [1285](#)
 - NAS_LTE_CPHY_CA_BW_NRB_LITE, [1285](#)
 - NAS_LTE_CPHY_SCELL_STATE_LITE, [1285](#)
 - NAS_MAX_DESCRIPTION_LENGTH, [1284](#)
 - NAS_MAX_NUM_NETWORKS, [1284](#)
 - NAS_MAX_SCC_RX_INFO_INSTANCES, [1284](#)
 - NAS_OTA_MESSAGE_MAX_BUF_SIZE, [1284](#)
 - NAS_PLMN_LENGTH, [1284](#)
 - NAS_SERVING_SYSTEM_INFO_MAX_RADIO↔INTERFACE_LIST, [1284](#)
 - pack_nas_GetACCOLC, [1285](#)
 - pack_nas_GetANAAAAAuthenticationStatus, [1286](#)
 - pack_nas_GetCDMANetworkParameters, [1286](#)
 - pack_nas_GetHomeNetwork, [1286](#)
 - pack_nas_GetNetworkPreference, [1287](#)
 - pack_nas_GetRFInfo, [1287](#)
 - pack_nas_GetServingNetwork, [1287](#)
 - pack_nas_GetServingNetworkCapabilities, [1288](#)
 - pack_nas_GetSignalStrengths, [1288](#)
 - pack_nas_PerformNetworkScan, [1288](#)
 - pack_nas_SLQSGetNetworkTime, [1290](#)
 - pack_nas_SLQSGetPLMNName, [1290](#)
 - pack_nas_SLQSGetServingSystem, [1291](#)
 - pack_nas_SLQSGetSignalStrength, [1291](#)
 - pack_nas_SLQSGetSysInfo, [1291](#)
 - pack_nas_SLQSGetSysSelectionPref, [1292](#)
 - pack_nas_SLQSInitiateNetworkRegistration, [1292](#)
 - pack_nas_SLQSNasConfigSigInfo2, [1292](#)
 - pack_nas_SLQSNasGetCellLocationInfo, [1293](#)
 - pack_nas_SLQSNasGetSigInfo, [1293](#)
 - pack_nas_SLQSNasIndicationRegisterExt, [1293](#)
 - pack_nas_SLQSNasSwtIndicationRegister, [1294](#)
 - pack_nas_SLQSNasSwtModemStatus, [1294](#)
 - pack_nas_SLQSSetBandPreference, [1294](#)
 - pack_nas_SLQSSetSignalStrengthsCallback, [1295](#)
 - pack_nas_SLQSSetSysSelectionPref, [1295](#)
 - pack_nas_SLQSSwiGetLteCQI, [1295](#)
 - pack_nas_SLQSSwiGetLteSccRxInfo, [1296](#)
 - pack_nas_SetACCOLC, [1289](#)
 - pack_nas_SetLURejectCallback, [1289](#)
 - pack_nas_SetNetworkPreference, [1289](#)
 - pack_nas_SetRFInfoCallback, [1290](#)
 - pack_nas_SlqsGetLTECphyCAInfo, [1290](#)
 - unpack_nas_GetACCOLC, [1296](#)
 - unpack_nas_GetANAAAAAuthenticationStatus, [1296](#)
 - unpack_nas_GetCDMANetworkParameters, [1297](#)
 - unpack_nas_GetHomeNetwork, [1297](#)
 - unpack_nas_GetNetworkPreference, [1297](#)
 - unpack_nas_GetRFInfo, [1297](#)
 - unpack_nas_GetServingNetwork, [1298](#)
 - unpack_nas_GetServingNetworkCapabilities, [1298](#)
 - unpack_nas_GetSignalStrengths, [1298](#)
 - unpack_nas_PerformNetworkScan, [1299](#)
 - unpack_nas_SLQSGetNetworkTime, [1302](#)
 - unpack_nas_SLQSGetPLMNName, [1302](#)
 - unpack_nas_SLQSGetServingSystem, [1302](#)
 - unpack_nas_SLQSGetSignalStrength, [1303](#)
 - unpack_nas_SLQSGetSysInfo, [1303](#)
 - unpack_nas_SLQSGetSysSelectionPref, [1303](#)
 - unpack_nas_SLQSInitiateNetworkRegistration, [1304](#)
 - unpack_nas_SLQSNasConfigSigInfo2, [1304](#)
 - unpack_nas_SLQSNasGetCellLocationInfo, [1304](#)
 - unpack_nas_SLQSNasGetSigInfo, [1305](#)
 - unpack_nas_SLQSNasIndicationRegisterExt, [1305](#)
 - unpack_nas_SLQSNasNetworkTimeCallBack_ind, [1305](#)

- unpack_nas_SLQSNasSigInfoCallback_ind, 1306
- unpack_nas_SLQSNasSwiIndicationRegister, 1306
- unpack_nas_SLQSNasSwiModemStatus, 1306
- unpack_nas_SLQSNasSwiOTAMessageCallback_ind, 1307
- unpack_nas_SLQSNasSysInfoCallback_ind, 1307
- unpack_nas_SLQSNasTimerCallback_ind, 1307
- unpack_nas_SLQSSetBandPreference, 1308
- unpack_nas_SLQSSetSignalStrengthsCallback, 1308
- unpack_nas_SLQSSetSysSelectionPref, 1308
- unpack_nas_SLQSSetSysSelectionPrefCall←Back_ind, 1309
- unpack_nas_SLQSSwiGetLteCQI, 1309
- unpack_nas_SLQSSwiGetLteSccRxInfo, 1309
- unpack_nas_SetACCOLC, 1299
- unpack_nas_SetDataCapabilitiesCallback_ind, 1300
- unpack_nas_SetEventReportInd, 1300
- unpack_nas_SetLURejectCallback, 1300
- unpack_nas_SetNasLTECphyCalIndCallback_ind, 1300
- unpack_nas_SetNetworkPreference, 1301
- unpack_nas_SetRFInfoCallback, 1301
- unpack_nas_SetRoamingIndicatorCallback_ind, 1301
- unpack_nas_SetServingSystemCallback_ind, 1301
- unpack_nas_SlqsGetLTECphyCAInfo, 1301
- nas_AddCDMASysInfo, 405
 - geoSysIdx, 406
 - regPrd, 406
- nas_AddSysInfo, 406
 - cellBroadcastCap, 406
 - geoSysIdx, 406
- nas_CDMAECIOThresh, 408
 - CDMAECIOThreshListLen, 409
 - pCDMAECIOThreshList, 409
- nas_CDMAInfo, 409
 - baseId, 410
 - baseLat, 410
 - baseLong, 410
 - nid, 410
 - refpn, 410
 - sid, 410
- nas_CDMARSSIThresh, 410
 - CDMARSSIThreshListLen, 410
 - pCDMARSSIThreshList, 410
- nas_CDMASysInfo, 410
 - baseId, 413
 - baseLat, 413
 - baseLong, 413
 - bsInfoValid, 413
 - bsPRev, 413
 - bsPRevValid, 414
 - ccsSupported, 414
 - ccsSupportedValid, 414
 - cdmaSysIdValid, 414
 - isSysPriMatch, 414
 - isSysPriMatchValid, 414
 - MCC, 414
 - MNC, 414
 - networkID, 414
 - networkIdValid, 414
 - pRevInUse, 414
 - pRevInUseValid, 414
 - packetZone, 414
 - packetZoneValid, 414
 - sysInfoCDMA, 414
 - systemID, 414
- nas_CDMASysInfoExt, 414
 - imsi_11_12, 415
 - MCC, 415
- nas_CSGID, 417
 - id, 418
 - mcc, 418
 - mnc, 418
 - mncPcsDigits, 418
 - rat, 418
- nas_CallBarringSysInfo, 406
 - csBarStatus, 407
 - psBarStatus, 407
- nas_CommInfo, 416
 - imsRegState, 417
 - modemMode, 417
 - psState, 417
 - systemMode, 417
 - temperature, 417
- nas_GERANInfo, 423
 - arfcn, 425
 - bsic, 425
 - cellID, 425
 - insNmrCellInfo, 425
 - lac, 425
 - nmrInst, 425
 - plmn, 425
 - rxLev, 425
 - timingAdvance, 425
- nas_GSMRSSIThresh, 427
 - GSMRSSIThreshListLen, 427
 - pGSMRSSIThreshList, 427
- nas_GSMSrvStatusInfo, 427
 - isPrefDataPath, 428
 - srvStatus, 428
 - trueSrvStatus, 428
- nas_GSMSysInfo, 428
 - cellId, 431
 - cellIdValid, 431
 - dtmSupp, 431
 - dtmSuppValid, 431
 - egprsSupp, 431
 - egprsSuppValid, 431
 - lac, 431
 - lacValid, 431
 - MCC, 431

- MNC, [431](#)
- networkIdValid, [431](#)
- regRejectInfoValid, [431](#)
- rejCause, [431](#)
- rejectSrvDomain, [431](#)
- sysInfoGSM, [431](#)
- nas_HDRECIOTthresh, [431](#)
 - HDRECIOTthreshListLen, [432](#)
 - pHDRECIOTthreshList, [432](#)
- nas_HDRIOTthresh, [432](#)
 - HDRIOTthreshListLen, [432](#)
 - pHDRIOTthreshList, [432](#)
- nas_HDRRSSITthresh, [432](#)
 - HDRRSSITthreshListLen, [433](#)
 - pHRRSSITthreshList, [433](#)
- nas_HDRSINRThreshold, [433](#)
 - HDRSINRthreshListLen, [434](#)
 - pHRRSINRthreshList, [434](#)
- nas_HDRSysInfo, [434](#)
 - hdrActiveProt, [436](#)
 - hdrActiveProtValid, [436](#)
 - hdrPersonality, [436](#)
 - hdrPersonalityValid, [436](#)
 - is856SysId, [436](#)
 - is856SysIdValid, [436](#)
 - isSysPrIMatch, [436](#)
 - isSysPrIMatchValid, [436](#)
 - sysInfoHDR, [436](#)
- nas_LTEInfo, [439](#)
 - band, [440](#)
 - bandwidth, [440](#)
 - emmConnState, [440](#)
 - emmState, [440](#)
 - emmSubState, [440](#)
 - RXChan, [440](#)
 - TXChan, [440](#)
- nas_LTEInfoInterfreq, [441](#)
 - freqsLen, [441](#)
 - InfoInterfreq, [441](#)
 - ueInIdle, [441](#)
- nas_LTEInfoIntrafreq, [441](#)
 - CellParams, [443](#)
 - cellReselPriority, [443](#)
 - cellsLen, [443](#)
 - earfcn, [443](#)
 - globalCellId, [443](#)
 - plmn, [443](#)
 - sIntraSearch, [443](#)
 - sNonIntraSearch, [443](#)
 - servingCellId, [443](#)
 - tac, [443](#)
 - threshServingLow, [443](#)
 - ueInIdle, [443](#)
- nas_LTEInfoNeighboringGSM, [444](#)
 - freqsLen, [444](#)
 - LteGsmCellInfo, [444](#)
 - ueInIdle, [444](#)
- nas_LTEInfoNeighboringWCDMA, [444](#)
 - freqsLen, [445](#)
 - LTEWCDMACellInfo, [445](#)
 - ueInIdle, [445](#)
- nas_LTERSRPThresh, [446](#)
 - LTERSRPThreshListLen, [446](#)
 - pLTERSRPThreshList, [446](#)
- nas_LTERSRQThresh, [446](#)
 - LTERSRQThreshListLen, [447](#)
 - pLTERSRQThreshList, [447](#)
- nas_LTERSSITthresh, [447](#)
 - LTERSSITthreshListLen, [447](#)
 - pLTERSSITthreshList, [447](#)
- nas_LTESNRThreshold, [449](#)
 - LTESNRthreshListLen, [449](#)
 - pLTESNRthreshList, [449](#)
- nas_LTESigRptConfig, [447](#)
 - avgPeriod, [448](#)
 - rptRate, [448](#)
- nas_LTESysInfo, [449](#)
 - cellId, [451](#)
 - cellIdValid, [451](#)
 - lac, [451](#)
 - lacValid, [451](#)
 - MCC, [451](#)
 - MNC, [451](#)
 - networkIdValid, [451](#)
 - regRejectInfoValid, [451](#)
 - rejCause, [451](#)
 - rejectSrvDomain, [452](#)
 - sysInfoLTE, [452](#)
 - tac, [452](#)
 - tacValid, [452](#)
- nas_MNRInfo, [453](#)
 - mcc, [453](#)
 - mnc, [453](#)
 - rat, [453](#)
- nas_PhyCaAggPcellInfo, [456](#)
 - dl_bw_value, [456](#)
 - freq, [456](#)
 - iLTEbandValue, [456](#)
 - pci, [456](#)
 - TlvPresent, [457](#)
- nas_PhyCaAggScellDIBw, [457](#)
 - dl_bw_value, [457](#)
 - TlvPresent, [457](#)
- nas_PhyCaAggScellIndType, [458](#)
 - freq, [458](#)
 - pci, [458](#)
 - scell_state, [458](#)
 - TlvPresent, [458](#)
- nas_PhyCaAggScellIndex, [457](#)
 - scell_idx, [458](#)
 - TlvPresent, [458](#)
- nas_PhyCaAggScellInfo, [459](#)
 - dl_bw_value, [461](#)
 - freq, [461](#)
 - iLTEbandValue, [461](#)
 - pci, [461](#)

- scell_state, [461](#)
 - TlvPresent, [461](#)
- nas_QmiNas3GppNetworkInfo, [462](#)
 - Description, [462](#)
 - Forbidden, [462](#)
 - InUse, [462](#)
 - MCC, [462](#)
 - MNC, [462](#)
 - Preferred, [462](#)
 - Roaming, [462](#)
- nas_QmiNas3GppNetworkRAT, [463](#)
 - MCC, [463](#)
 - MNC, [463](#)
 - RAT, [463](#)
- nas_QmisNasPcsDigit, [463](#)
 - includes_pcs_digit, [464](#)
 - MCC, [464](#)
 - MNC, [464](#)
- nas_RFInfoTlv, [465](#)
 - activeBandClass, [465](#)
 - activeChannel, [465](#)
 - radioInterface, [465](#)
 - radioInterfaceSize, [465](#)
 - TlvPresent, [465](#)
- nas_RejectReasonTlv, [464](#)
 - rejectCause, [464](#)
 - serviceDomain, [464](#)
 - TlvPresent, [464](#)
- nas_RxSigInfo, [467](#)
 - isRadioTuned, [468](#)
 - rsrp, [468](#)
 - rxChainIndex, [468](#)
 - rxPower, [468](#)
- nas_SLQSSignalStrengthsIndReq, [472](#)
 - ecioDelta, [473](#)
 - ecioThresholdList, [473](#)
 - ecioThresholdListLen, [473](#)
 - ioDelta, [473](#)
 - lteRsrpDelta, [473](#)
 - lteSnrDelta, [473](#)
 - rsrqDelta, [473](#)
 - rxSignalStrengthDelta, [473](#)
 - sinrDelta, [473](#)
 - sinrThresholdList, [473](#)
 - sinrThresholdListLen, [473](#)
- nas_SLQSSignalStrengthsInformation, [473](#)
 - ecioInfo, [474](#)
 - errorRateInfo, [474](#)
 - io, [474](#)
 - lteRsrpinfo, [474](#)
 - lteSnrinfo, [474](#)
 - rsrqInfo, [474](#)
 - rxSignalStrengthInfo, [474](#)
 - sinr, [474](#)
- nas_SLQSSignalStrengthsTlv, [474](#)
 - sSLQSSignalStrengthsInfo, [474](#)
 - TlvPresent, [474](#)
- nas_SccRxInfo, [469](#)
 - numInstances, [469](#)
 - rsrq, [469](#)
 - sigInfo, [469](#)
 - snr, [469](#)
 - TlvPresent, [470](#)
- nas_SignalStrengthTlv, [471](#)
 - radioInterface, [472](#)
 - signalStrength, [472](#)
 - TlvPresent, [472](#)
- nas_SrvStatusInfo, [474](#)
 - isPrefDataPath, [475](#)
 - srvStatus, [475](#)
- nas_TDSCDMAECIOThresh, [478](#)
 - pTDSCDMAECIOThreshList, [478](#)
 - TDSCDMAECIOThreshListLen, [478](#)
- nas_TDSCDMARSCPTThresh, [478](#)
 - pTDSCDMARSCPTThreshList, [479](#)
 - TDSCDMARSCPTThreshListLen, [479](#)
- nas_TDSCDMARSSIThresh, [479](#)
 - pTDSCDMARSSIThreshList, [479](#)
 - TDSCDMARSSIThreshListLen, [479](#)
- nas_TDSCDMASINRThresh, [479](#)
 - pTDSCDMASINRThreshList, [480](#)
 - TDSCDMASINRThreshListLen, [480](#)
- nas_UMTSInfo, [482](#)
 - cellID, [483](#)
 - ecio, [484](#)
 - geranInst, [484](#)
 - GeranInstInfo, [484](#)
 - lac, [484](#)
 - plmn, [484](#)
 - psc, [484](#)
 - rscp, [484](#)
 - UMTSInstInfo, [484](#)
 - uarfcn, [484](#)
 - umtsInst, [484](#)
- nas_UMTSinstInfo, [484](#)
 - umtsEcio, [485](#)
 - umtsPsc, [485](#)
 - umtsRscp, [485](#)
 - umtsUarfcn, [485](#)
- nas_UniversalTime, [486](#)
 - day, [487](#)
 - dayOfWeek, [487](#)
 - hour, [487](#)
 - minute, [487](#)
 - month, [487](#)
 - second, [487](#)
 - year, [487](#)
- nas_WCDMAECIOThresh, [488](#)
 - pWCDMAECIOThreshList, [489](#)
 - WCDMAECIOThreshListLen, [489](#)
- nas_WCDMAInfoLTENeighborCell, [489](#)
 - UMTSLTENbrCell, [490](#)
 - umtsLTENbrCellLen, [490](#)
 - wcdmaRRState, [490](#)
- nas_WCDMARSSIThresh, [490](#)
 - pWCDMARSSIThreshList, [490](#)

- WCDMARSSIThreshListLen, 490
- nas_WCDMASysInfo, 490
 - cellId, 493
 - cellIdValid, 493
 - hsCallStatus, 493
 - hsCallStatusValid, 493
 - hsInd, 493
 - hsIndValid, 493
 - lac, 493
 - lacValid, 493
 - MCC, 493
 - MNC, 494
 - networkIdValid, 494
 - psc, 494
 - pscValid, 494
 - regRejectInfoValid, 494
 - rejCause, 494
 - rejectSrvDomain, 494
 - sysInfoWCDMA, 494
- nas_acqOrderPref, 404
 - acqOrdeLen, 405
 - pAcqOrder, 405
- nas_callBarStatus, 407
 - csBarStatus, 408
 - psBarStatus, 408
- nas_cellParams, 415
 - pci, 416
 - rsrp, 416
 - rsrq, 416
 - rsi, 416
 - srxlev, 416
- nas_currentPLMN, 418
 - MCC, 419
 - MNC, 419
 - netDescr, 419
 - netDescrLength, 419
- nas_dataSrvCapabilities, 419
 - dataCapabilities, 420
 - dataCapabilitiesLen, 420
- nas_detailSvcInfo, 420
 - hdrHybrid, 421
 - hdrSrvStatus, 421
 - isSysForbidden, 421
 - srvCapability, 421
 - srvStatus, 421
- nas_ecioListElement, 422
 - ecio, 422
 - radioIf, 422
- nas_errorRateListElement, 422
 - errorRate, 423
 - radioIf, 423
- nas_geranInstInfo, 425
 - geranArfcn, 426
 - geranBsicBcc, 426
 - geranBsicNcc, 426
 - geranRssi, 426
- nas_gsmCellInfo, 426
 - arfcn, 427
 - band1900, 427
 - bsicId, 427
 - cellIdValid, 427
 - rsi, 427
 - srxlev, 427
- nas_infoInterFreq, 436
 - cell_resel_priority, 437
 - cellInterFreqParams, 437
 - cells_len, 437
 - earfcn, 437
 - threshXHigh, 437
 - threshXLow, 437
- nas_lteGsmCellInfo, 437
 - cellReselPriority, 438
 - cells_len, 438
 - GsmCellInfo, 438
 - nccPermitted, 438
 - threshGsmHigh, 438
 - threshGsmLow, 438
- nas_lteRsrpInformation, 445
 - rsrpLevel, 445
- nas_lteSnrInformation, 448
 - snrLevel, 449
- nas_lteWcdmaCellInfo, 452
 - cellReselPriority, 453
 - cellsLen, 453
 - threshXhigh, 453
 - threshXlow, 453
 - uarfcn, 453
 - WCDMACellInfo, 453
- nas_major
 - LteNasReleaseInfo_s, 386
 - NASLteNasReleaseInfoTlv, 509
- nas_minor
 - LteNasReleaseInfo_s, 386
 - NASLteNasReleaseInfoTlv, 509
- nas_netSelectionPref, 454
 - mcc, 454
 - mnc, 454
 - netReg, 454
- nas_nmrCellInfo, 454
 - nmrArfcn, 455
 - nmrBsic, 455
 - nmrCellID, 455
 - nmrLac, 455
 - nmrPlmn, 455
 - nmrRxLev, 456
- nas_qaQmi3Gpp2TimeZone, 461
 - daylightSavings, 462
 - leapSeconds, 462
 - localTimeOffset, 462
- nas_release
 - LteNasReleaseInfo_s, 386
 - NASLteNasReleaseInfoTlv, 509
- nas_roamIndList, 465
 - numInstances, 466
 - radioInterface, 466
 - roamIndicator, 466

- nas_rsrqInformation, 466
 - radiolf, 467
 - rsrq, 467
- nas_rxSignalStrengthListElement, 468
 - radiolf, 468
 - rxSignalStrength, 468
- nas_servSystem, 470
 - csAttachState, 471
 - numRadioInterfaces, 471
 - psAttachState, 471
 - radiolInterface, 471
 - regState, 471
 - selNetwork, 471
- nas_sysInfoCommon, 475
 - isSysForbidden, 477
 - isSysForbiddenValid, 477
 - roamStatus, 477
 - roamStatusValid, 477
 - srvCapability, 477
 - srvCapabilityValid, 477
 - srvDomain, 477
 - srvDomainValid, 478
- nas_timeInfo, 480
 - day, 482
 - dayLtSavingAdj, 482
 - dayOfWeek, 482
 - hour, 482
 - minute, 482
 - month, 482
 - radiolInterface, 482
 - second, 482
 - timeZone, 482
 - TlvPresent, 482
 - year, 482
- nas_umtsLTENbrCell, 485
 - cellsTDD, 486
 - earfcn, 486
 - pci, 486
 - rsrp, 486
 - rsrq, 486
 - srxlev, 486
- nas_wcdmaCellInfo, 487
 - cpich_ecno, 488
 - cpich_rscp, 488
 - psc, 488
 - srxlev, 488
- nasCellLocationInfoResp, 494
 - pCDMAInfo, 495
 - pGERANInfo, 495
 - pLTEInfoInterfreq, 495
 - pLTEInfoIntrafreq, 495
 - pLTEInfoNeighboringGSM, 495
 - pLTEInfoNeighboringWCDMA, 495
 - pUMTSCellID, 495
 - pUMTSInfo, 495
 - pWCDMAInfoLTENeighborCell, 495
- nasGet3GPP2SubscriptionInfoReq, 496
 - namID, 496
- nasGet3GPP2SubscriptionInfoResp, 496
 - pCDMAChannel, 497
 - pDirNum, 497
 - pHomeSIDNID, 497
 - pMinBasedIMSI, 497
 - pNAMNameInfo, 497
 - pTrueIMSI, 497
- nasGetHDRColorCodeResp, 497
 - pColorCode, 498
- nasGetLTECphyCa, 498
 - sPhyCaAggPcellInfo, 498
 - sPhyCaAggScellIDBw, 498
 - sPhyCaAggScellIndType, 498
 - sPhyCaAggScellIndex, 498
 - sPhyCaAggScellInfo, 498
- NasGetLTECphyCaInfo, 498
 - PhyCaAggPcellInfo, 498
 - PhyCaAggScellIDBw, 498
 - PhyCaAggScellIndType, 498
 - PhyCaAggScellIndex, 498
 - PhyCaAggScellInfo, 498
- nasGetLTECphyCaResp, 498
 - pPhyCaAggPcellInfo, 499
 - pPhyCaAggScellIDBw, 499
 - pPhyCaAggScellIndType, 499
 - pPhyCaAggScellIndex, 499
 - pPhyCaAggScellInfo, 499
- nasGetSigInfoResp, 499
 - pCDMASSInfo, 500
 - pGSMSSInfo, 500
 - pHDRSSInfo, 500
 - pLTESSInfo, 500
 - pTDSCDMASigInfoExt, 500
 - pTDSCDMASigInfoRscp, 500
 - pWCDMASSInfo, 500
- nasGetSysInfoResp, 500
 - pAddCDMASysInfo, 502
 - pAddGSMSysInfo, 502
 - pAddHDRSysInfo, 502
 - pAddLTESysInfo, 502
 - pAddWCDMASysInfo, 502
 - pCDMASrvStatusInfo, 502
 - pCDMASysInfo, 502
 - pGSMCallBarringSysInfo, 502
 - pGSMCipherDomainSysInfo, 502
 - pGSMSrvStatusInfo, 502
 - pGSMSysInfo, 502
 - pHDRSrvStatusInfo, 503
 - pHDRSysInfo, 503
 - pLTESrvStatusInfo, 503
 - pLTESysInfo, 503
 - pLTEVoiceSupportSysInfo, 503
 - pWCDMACallBarringSysInfo, 503
 - pWCDMACipherDomainSysInfo, 503
 - pWCDMASrvStatusInfo, 503
 - pWCDMASysInfo, 503
- nasGetTxRxInfoReq, 503
 - radio_if, 503

- nasGetTxRxInfoResp, 504
 - pRXChain0Info, 504
 - pRXChain1Info, 504
 - pTXInfo, 504
- nasIndicationRegisterReq, 504
 - pDDTMInd, 507
 - pDualStandByPrefInd, 507
 - pErrorRateInd, 507
 - pHDRNewUATIAssInd, 507
 - pHDRSessionCloseInd, 507
 - pLTECphyCa, 507
 - pManagedRoamingInd, 507
 - pNetworkTimeInd, 507
 - pServingSystemInd, 507
 - pSignalStrengthInd, 507
 - pSubscriptionInfoInd, 507
 - pSysInfoInd, 507
 - pSystemSelectionInd, 507
- nasInitNetworkReg, 507
 - pChangeDuration, 508
 - pMNRInfo, 508
 - pMncPcsDigitStatus, 508
 - regAction, 508
- nasNetworkTime, 510
 - pDayltSavAdj, 510
 - pRadioInterface, 510
 - pTimeZone, 511
 - universalTime, 511
- nasOperatorNameResp, 511
 - pNITZInformation, 511
 - pOperatorNameString, 511
 - pOperatorPLMNList, 511
 - pPLMNNetworkName, 512
 - pSrvProviderName, 512
- nasPLMNNameReq, 516
 - mcc, 517
 - mnc, 517
 - pMncPcsStatus, 517
- nasPLMNNameResp, 517
 - longName, 519
 - longNameCI, 519
 - longNameEn, 519
 - longNameLen, 519
 - longNameSB, 519
 - shortName, 519
 - shortNameCI, 519
 - shortNameEn, 519
 - shortNameLen, 519
 - shortNameSB, 520
 - spn, 520
 - spnEncoding, 520
 - spnLength, 520
- nasRelInfoTlv
 - NASQmiCbkNasSwiOTAMessageInd, 520
- nasSigInfo, 523
 - pCDMASigInfo, 524
 - pGSMSigInfo, 524
 - pHDRSigInfo, 524
 - pLTESigInfo, 524
 - pRscp, 524
 - pTDSCDMASigInfoExt, 524
 - pWCDMASigInfo, 524
- nasSwiGetChannelLockResp, 524
 - pLteEARFCN, 525
 - pLtePCI, 525
 - pWcdmaUARFCN, 525
- NasSwiIndReg, 525
 - gsmUmtsDI, 526
 - gsmUmtsUI, 526
 - lteEmmDI, 526
 - lteEmmUI, 526
 - lteEsmDI, 526
 - lteEsmUI, 526
 - pRankIndicatorInd, 526
 - pTimer, 526
- nasSwiSetChannelLockReq, 527
 - pLteEARFCN, 527
 - pLtePCI, 527
 - pWcdmaUARFCN, 527
- nasSysInfo, 527
 - pAddCDMASysInfo, 529
 - pAddGSMSysInfo, 529
 - pAddHDRSysInfo, 530
 - pAddLTESysInfo, 530
 - pAddWCDMASysInfo, 530
 - pCDMASrvStatusInfo, 530
 - pCDMASysInfo, 530
 - pGSMCallBarringSysInfo, 530
 - pGSMCipherDomainSysInfo, 530
 - pGSMSrvStatusInfo, 530
 - pGSMSysInfo, 530
 - pHDRSrvStatusInfo, 530
 - pHDRSysInfo, 530
 - pLTESrvStatusInfo, 530
 - pLTESysInfo, 530
 - pLTEVoiceSupportSysInfo, 530
 - pSysInfoNoChange, 530
 - pWCDMACallBarringSysInfo, 530
 - pWCDMACipherDomainSysInfo, 530
 - pWCDMASrvStatusInfo, 530
 - pWCDMASysInfo, 530
- nasTimers, 531
 - t3396_apn, 531
 - t3396_plmn_id, 531
 - t3396_val, 531
- nccPermitted
 - lteGsmCellInfo, 379
 - nas_lteGsmCellInfo, 438
- NeighborSetCnt
 - NetworkStat1x, 536
- netDescr
 - currentPLMN, 183
 - nas_currentPLMN, 419
- netDescrLength
 - currentPLMN, 183
 - nas_currentPLMN, 419

- netInfoLen
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, 1082
- netReg
 - nas_netSelectionPref, 454
 - netSelectionPref, 532
- NetSelPref
 - NASNetSelPreferenceTlv, 510
- netSelectionPref, 531
 - mcc, 532
 - mnc, 532
 - netReg, 532
- NetStats, 532
 - rx_bytes, 533
 - rx_errors, 533
 - rx_overflows, 533
 - rx_packets, 533
 - tx_bytes, 533
 - tx_errors, 533
 - tx_overflows, 533
 - tx_packets, 533
- Network Access Service (NAS), 36
- NetworkDebugResp, 533
 - pDataStatusDetail, 534
 - pDeviceConfigDetail, 534
 - pNetworkStat1x, 534
 - pNetworkStatEVDO, 534
 - pObjectVer, 534
- NetworkID
 - qaQmiServingSystemParam, 683
 - unpack_nas_SLQSGetServingSystem_t, 995
- networkID
 - CDMASysInfo, 158
 - nas_CDMASysInfo, 414
- networkIdValid
 - CDMASysInfo, 158
 - GSMSysInfo, 283
 - LTESysInfo, 397
 - nas_CDMASysInfo, 414
 - nas_GSMSysInfo, 431
 - nas_LTESysInfo, 451
 - nas_WCDMASysInfo, 494
 - WCDMASysInfo, 1157
- networkInfoLen
 - unpack_wds_SLQSGetCurrDataSystemStat_t, 1073
- NetworkStat1x, 534
 - ActSetCnt, 536
 - NeighborSetCnt, 536
 - pActPilotPNElements, 536
 - pNeighborSetPilotPN, 536
 - RX_EC_IO, 536
 - RX_PWR, 536
 - SO, 536
 - State, 536
 - TX_PWR, 536
- NetworkStatEVDO, 536
 - MACIndex, 538
 - PER, 538
 - pSectorID, 538
 - PilotEnergy, 538
 - RX_PWR, 538
 - SNR, 538
 - SectorIDLen, 538
 - State, 538
- NetworkType
 - CurrNetworkInfo, 186
 - currNetworkInfo, 187
 - wds_currNetworkInfo, 1160
- NewMMTlv
 - unpack_sms_SetNewSMSCallback_ind_t, 1043
- newMTMessageTlv, 538
 - MTMessageInfo, 538
 - TlvPresent, 538
- newPINLen
 - uim_unblockUIMPIN, 889
 - unblockUIMPIN, 924
- newPINVal
 - uim_unblockUIMPIN, 889
 - unblockUIMPIN, 924
- newPasswd
 - voiceSetCallBarringPwdInfo, 1134
- newPasswdAgain
 - voiceSetCallBarringPwdInfo, 1134
- newPin
 - pack_dms_UIMUnblockPIN_t, 564
- newPwd
 - newPwdData, 539
- newPwdAgain
 - newPwdData, 539
- newPwdData, 538
 - newPwd, 539
 - newPwdAgain, 539
- newValue
 - pack_dms_UIMChangePIN_t, 560
- nextHeader
 - LibPackTFTIDParams, 347
 - TFTIDParams, 860
- nid
 - CDMAInfo, 148
 - nas_CDMAInfo, 410
 - sidNid, 780
 - unpack_nas_GetHomeNetwork_t, 983
- NmeaPort
 - DcsUsbPortNames, 201
- nmrArfcn
 - nas_nmrCellInfo, 455
 - nmrCellInfo, 540
- nmrBsic
 - nas_nmrCellInfo, 455
 - nmrCellInfo, 540
- nmrCellID
 - nas_nmrCellInfo, 455
 - nmrCellInfo, 540
- nmrCellInfo, 539
 - nmrArfcn, 540

- nmrBsic, [540](#)
- nmrCellID, [540](#)
- nmrLac, [540](#)
- nmrPlmn, [540](#)
- nmrRxLev, [540](#)
- nmrInst
 - GERANInfo, [237](#)
 - nas_GERANInfo, [425](#)
- nmrLac
 - nas_nmrCellInfo, [455](#)
 - nmrCellInfo, [540](#)
- nmrPlmn
 - nas_nmrCellInfo, [455](#)
 - nmrCellInfo, [540](#)
- nmrRxLev
 - nas_nmrCellInfo, [456](#)
 - nmrCellInfo, [540](#)
- noReplyTimer
 - callFWExtInfo, [133](#)
 - callFWInfo, [134](#)
- Non-service specific APIs (SWI), [47](#)
- notifType
 - voiceSUPSNotification, [1147](#)
- notification
 - omaDmNotificationsTlv, [549](#)
 - unpack_omaDmNotificationsTlv_t, [1018](#)
- notificationType
 - SMSEtwsMessage, [810](#)
 - sMSEtwsMessage, [809](#)
- notused
 - unpack_dms_SetCrashAction_t, [940](#)
- num_instances
 - _qaQmi3GPP2BroadcastCfgInfo, [65](#)
 - _qaQmi3GPPBroadcastCfgInfo, [66](#)
 - custSettingList, [192](#)
 - DMScustSettingList, [212](#)
- numApp
 - slotInf, [786](#)
 - slotInfo, [787](#)
 - uim_slotInfo, [887](#)
- numCrashes
 - CrashInfo, [173](#)
 - crashInformation, [176](#)
- numEntries
 - CurrentImgList, [182](#)
 - unpack_dms_SLQSSwiGetFirmwareCurr_t, [949](#)
- numFeatures
 - personalizationStatus, [650](#)
- numFiles
 - registerRefresh, [723](#)
- NumFlows
 - unpack_qos_SLQSSetQosEventCallback_ind_t, [1028](#)
- numInstance
 - operatorPLMNLList, [551](#)
 - PLMNNetworkName, [657](#)
- numInstances
 - arrAlertingPattern, [112](#)
 - arrAlertingType, [113](#)
 - arrAlphaID, [113](#)
 - arrCallEndReason, [115](#)
 - arrCallInfo, [115](#)
 - arrCalledPartyNum, [114](#)
 - arrConnectPartyNum, [116](#)
 - arrDiagInfo, [117](#)
 - arrRedirPartyNum, [117](#)
 - arrRemotePartyName, [118](#)
 - arrRemotePartyNum, [118](#)
 - arrSvcOption, [119](#)
 - arrUUSInfo, [120](#)
 - DomainNameList, [215](#)
 - getCallFWExtInfo, [247](#)
 - getCallFWInfo, [247](#)
 - getMsgWaitingInfo, [266](#)
 - homeSIDNID, [295](#)
 - msgWaitingInfo, [404](#)
 - nas_SccRxInfo, [469](#)
 - nas_roamIndList, [466](#)
 - PCSCFFQDNAddressList, [643](#)
 - PCSCFIPv4ServerAddressList, [644](#)
 - roamIndList, [728](#)
 - SccRxInfo, [743](#)
 - wds_DomainNameList, [1166](#)
 - wds_PCSCFFQDNAddressList, [1169](#)
 - wds_PCSCFIPv4ServerAddressList, [1170](#)
- numLen
 - callFWExtInfo, [133](#)
 - callFWInfo, [134](#)
 - calledPartyInfo, [129](#)
 - callingPartyInfo, [138](#)
 - peerNumberInfo, [649](#)
 - redirNumInfo, [722](#)
 - remotePartyNum, [725](#)
- numOfFiles
 - UIMRefreshEvent, [904](#)
- numOfRoutes
 - smsSetRoutesReq, [820](#)
- numOpt
 - DHCPOptionList, [207](#)
 - wds_DHCPLeaseOptTlv, [1161](#)
 - wds_DHCPv4OptionList, [1164](#)
 - WdsDHCPv4OptionList, [1182](#)
 - wdsDhcpv4OptionList, [1183](#)
- numPI
 - peerNumberInfo, [649](#)
- NumPilots
 - PilotSetData, [656](#)
- numPlan
 - callFWExtInfo, [133](#)
 - calledPartyInfo, [129](#)
 - callingPartyInfo, [138](#)
 - connectNumInfo, [172](#)
 - peerNumberInfo, [649](#)
 - redirNumInfo, [722](#)
- numPresInd
 - connectNumInfo, [172](#)

- numQosFlow
 - sQosStat, [824](#)
 - unpack_qos_SLQSQosSwiReadDataStats_t, [1027](#)
- numRadioInterfaces
 - nas_servSystem, [471](#)
 - servSystem, [752](#)
- NumRxFilters
 - unpack_qos_QosFlowInfo_t, [1023](#)
- numSI
 - peerNumberInfo, [649](#)
- numSlot
 - cardStatus, [139](#)
 - uim_cardStatus, [879](#)
- NumSupUSBComps
 - unpack_dms_GetUSBComp_t, [938](#)
- NumTxFilters
 - unpack_qos_QosFlowInfo_t, [1023](#)
- numType
 - callFWExtInfo, [133](#)
 - calledPartyInfo, [129](#)
 - callingPartyInfo, [138](#)
 - connectNumInfo, [172](#)
 - peerNumberInfo, [649](#)
 - redirNumInfo, [722](#)
- number
 - callFWExtInfo, [133](#)
 - callFWInfo, [134](#)
 - calledPartyInfo, [128](#)
 - callingPartyInfo, [137](#)
 - ECTNum, [221](#)
 - peerNumberInfo, [649](#)
 - redirNumInfo, [722](#)
- numberPlan
 - callFwdTypeAndPlan, [131](#)
- numberType
 - callFwdTypeAndPlan, [131](#)
- NxtHdrProto
 - unpack_qos_swiQosFilter_t, [1034](#)
- OKtoRefresh
 - UIMRefreshOKReq, [906](#)
- OMADMCancelSession
 - qaGobiApiOmadm.h, [1545](#)
- OMADMEnabled
 - unpack_swioma_SLQSOMADMGetSettings_↵
t, [1052](#)
- OMADMGetPendingNIA
 - qaGobiApiOmadm.h, [1546](#)
- OMADMGetSessionInfo
 - qaGobiApiOmadm.h, [1546](#)
- OMADMStartSession
 - qaGobiApiOmadm.h, [1547](#)
- OTASPStatus
 - voiceOTASPStatusInfo, [1130](#)
- oddEvenInd
 - calledPartySubAdd, [130](#)
- OfflineReason
 - unpack_dms_GetPower_t, [937](#)
- offset
 - readTransparentInfo, [720](#)
 - uim_readTransparentInfo, [883](#)
- oldPINLen
 - changeUIMPIN, [161](#)
 - uim_changeUIMPIN, [880](#)
- oldPINVal
 - changeUIMPIN, [161](#)
 - uim_changeUIMPIN, [880](#)
- oldPasswd
 - voiceSetCallBarringPwdInfo, [1134](#)
- oldValue
 - pack_dms_UIMChangePIN_t, [560](#)
- omaDmConfig
 - sessionInfo, [753](#)
 - sessionInfoExt, [753](#)
- omaDmConfigTlv, [542](#)
 - alertmsg, [542](#)
 - alertmsglength, [542](#)
 - state, [543](#)
 - userInputReq, [543](#)
 - userInputTimeout, [543](#)
- omaDmConfigTlvExt, [543](#)
 - alertmsg, [545](#)
 - alertmsglength, [545](#)
 - state, [545](#)
 - userInputReq, [545](#)
 - userInputTimeout, [545](#)
- omaDmFota
 - sessionInfo, [753](#)
 - sessionInfoExt, [753](#)
- omaDmFotaTlv, [545](#)
 - description, [547](#)
 - descriptionlength, [547](#)
 - fwdloadsize, [547](#)
 - fwloadComplete, [547](#)
 - namelength, [547](#)
 - package_name, [547](#)
 - sessionType, [547](#)
 - severity, [547](#)
 - state, [547](#)
 - updateCompleteStatus, [547](#)
 - userInputReq, [547](#)
 - userInputTimeout, [547](#)
 - version, [547](#)
 - versionlength, [547](#)
- omaDmFotaTlvExt, [547](#)
 - description, [549](#)
 - descriptionlength, [549](#)
 - fumoResultCode, [549](#)
 - namelength, [549](#)
 - package_name, [549](#)
 - packageSize, [549](#)
 - receivedBytes, [549](#)
 - reserved, [549](#)
 - state, [549](#)
 - userInputTimeout, [549](#)
 - version, [549](#)
 - versionlength, [549](#)

- omaDmNotifications
 - sessionInfo, [753](#)
- omaDmNotificationsTlv, [549](#)
 - notification, [549](#)
 - sessionStatus, [549](#)
- opaqueId
 - pack_loc_SLQSLOCInjectSensorData_t, [579](#)
- Open Mobile Alliance Service (OMA), [43](#)
- operatingMode
 - dms_OperatingModeTlv, [210](#)
- OperatingModeTlv
 - unpack_dms_SetEventReport_ind_t, [941](#)
- operation
 - depersonalizationInformation, [203](#)
- OperationMode
 - unpack_dms_GetPower_t, [937](#)
- operatorNameString, [549](#)
 - PLMNName, [550](#)
- OperatorPLMNData, [550](#)
 - lac1, [551](#)
 - lac2, [551](#)
 - mcc, [551](#)
 - mnc, [551](#)
 - PLMNRecID, [551](#)
- operatorPLMNList, [551](#)
 - numInstance, [551](#)
 - PLMNData, [551](#)
- optCode
 - DHCPOption, [206](#)
 - wds_DHCPOpt, [1162](#)
 - wds_DHCPv4Option, [1163](#)
 - WdsDHCPv4Option, [1182](#)
 - wdsDhcpv4Option, [1181](#)
- optList
 - wds_DHCPLeaseOptTlv, [1161](#)
- optListData
 - wds_DHCPLeaseOptTlv, [1161](#)
- optVal
 - wds_DHCPv4Option, [1163](#)
 - WdsDHCPv4Option, [1182](#)
 - wdsDhcpv4Option, [1181](#)
- optValLen
 - DHCPOption, [206](#)
 - wds_DHCPOpt, [1162](#)
 - wds_DHCPv4Option, [1163](#)
 - WdsDHCPv4Option, [1182](#)
 - wdsDhcpv4Option, [1181](#)
- OriginateUSSD
 - qaGobiApiVoice.h, [1677](#)
- otaMsgTlv
 - NASQmiCbkNasSwtOTAMessageInd, [520](#)
- p1Status
 - unpack_dms_UIMGetPINStatus_t, [958](#)
- p1UnblockRetriesLeft
 - unpack_dms_UIMGetPINStatus_t, [958](#)
- p1VerifyRetriesLeft
 - unpack_dms_UIMGetPINStatus_t, [958](#)
- p2Status
 - unpack_dms_UIMGetPINStatus_t, [958](#)
- p2UnblockRetriesLeft
 - unpack_dms_UIMGetPINStatus_t, [958](#)
- p2VerifyRetriesLeft
 - unpack_dms_UIMGetPINStatus_t, [958](#)
- p3GPP2Pri
 - swiQosFlow, [845](#)
- p3GPP2TimeInfo
 - GetNetworkTimeResp, [266](#)
 - unpack_nas_SLQSGetNetworkTime_t, [992](#)
- p3GPPIImCn
 - swiQosFlow, [845](#)
- p3GPPResResidualBER
 - swiQosFlow, [845](#)
- p3GPPSigInd
 - swiQosFlow, [845](#)
- p3GPPTimeInfo
 - GetNetworkTimeResp, [266](#)
 - unpack_nas_SLQSGetNetworkTime_t, [992](#)
- p3GPPTraHdlPri
 - swiQosFlow, [845](#)
- p3GppNetworkInfoInstances
 - unpack_nas_PerformNetworkScan_t, [987](#)
- p3GppNetworkInstanceSize
 - unpack_nas_PerformNetworkScan_t, [987](#)
- p3gppRelease
 - _slqs3GPPConfigItem, [69](#)
 - pack_wds_SLQSSet3GPPConfigItem_t, [633](#)
- pAAASPI
 - pack_wds_SetMobileIPProfile_t, [627](#)
- PACK_WDS_IPV4
 - wds.h, [1798](#)
- PACK_WDS_IPV6
 - wds.h, [1798](#)
- pAMRStatus
 - voiceGetConfigReq, [1121](#)
- pAPNClass
 - LibPackprofile_3GPP, [337](#)
 - LibpackProfile3GPP, [326](#)
 - Profile3GPP, [667](#)
- pAPNClass3GPP2
 - LibPackprofile_3GPP2, [343](#)
 - LibpackProfile3GPP2, [331](#)
 - Profile3GPP2, [673](#)
- pAPNDisabledFlag
 - LibPackprofile_3GPP, [337](#)
 - LibpackProfile3GPP, [326](#)
 - Profile3GPP, [667](#)
- pAPNEnabled3GPP2
 - LibPackprofile_3GPP2, [343](#)
 - LibpackProfile3GPP2, [332](#)
 - Profile3GPP2, [673](#)
- pAPNName
 - LibPackprofile_3GPP, [337](#)
 - LibpackProfile3GPP, [326](#)
 - Profile3GPP, [667](#)
 - qmiWdsRunTimeSettings, [713](#)
 - swiPDPRuntimeSettingsResp, [839](#)

- pAPNnameSize
 - LibPackprofile_3GPP, 337
 - LibpackProfile3GPP, 326
 - Profile3GPP, 667
- PASSWORD_LENGTH
 - qaGobiApiVoice.h, 1676
- pAVRXAVCHadroom
 - RXAVCList, 734
- pAVRXAVCSens
 - RXAVCList, 734
- pAccelAcceptReady
 - QmiCbkLocSensorStreamingInd, 701
- pAccelSamplesAccepted
 - QmiCbkLocInjectSensorDataInd, 693
- pAccelTempAcceptReady
 - QmiCbkLocSensorStreamingInd, 701
- pAccelTempSamplesAccepted
 - QmiCbkLocInjectSensorDataInd, 693
- pAcceleroData
 - LocInjectSensorDataReq, 373
- pAcceleroTempData
 - LocInjectSensorDataReq, 373
- pAcceleroTimeSrc
 - LocInjectSensorDataReq, 373
- pAcqOrder
 - acqOrderPref, 98
 - nas_acqOrderPref, 405
- pAcqOrderPref
 - _sysSelectPrefParams, 93
 - pack_nas_SLQSSetSysSelectionPref_t, 599
- pActPilotPNElements
 - NetworkStat1x, 536
- pAddCDMASysInfo
 - nasGetSysInfoResp, 502
 - nasSysInfo, 529
 - unpack_nas_SLQSGetSysInfo_t, 999
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1013
- pAddGSMsInfo
 - nasGetSysInfoResp, 502
 - nasSysInfo, 529
 - unpack_nas_SLQSGetSysInfo_t, 999
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1013
- pAddHDRsInfo
 - nasGetSysInfoResp, 502
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 999
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1013
- pAddLTESysInfo
 - nasGetSysInfoResp, 502
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 999
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1013
- pAddWCDMASysInfo
 - nasGetSysInfoResp, 502
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 999
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1013
- pAddrAllocPref
 - LibPackprofile_3GPP, 337
 - LibpackProfile3GPP, 326
 - Profile3GPP, 667
- pAddress
 - pack_wds_SetMobileIPProfile_t, 627
- pAirTimer
 - voiceGetConfigReq, 1121
- pAirTimerCnt
 - voiceGetConfigResp, 1123
- pAirTimerConfig
 - voiceSetConfigReq, 1136
- pAirTimerStatus
 - voiceSetConfigResp, 1138
- pAlertPriority
 - cdmaMsgDecodingParams, 151
- pAlertType
 - voiceCallInfoResp, 1099
- pAlertingPattern
 - voiceCallInfoResp, 1099
- pAllowLinger
 - LibPackprofile_3GPP2, 343
 - LibpackProfile3GPP2, 331
 - Profile3GPP2, 673
- pAlphaIDInfo
 - USSResp, 1092
 - voiceCallInfoResp, 1099
 - voiceCallResponseParams, 1102
 - voiceGetCLIPResp, 1114
 - voiceGetCLIRResp, 1116
 - voiceGetCNAPResp, 1117
 - voiceGetCOLPResp, 1118
 - voiceGetCOLRResp, 1120
 - voiceGetCallBarringResp, 1109
 - voiceGetCallFWResp, 1111
 - voiceGetCallWaitInfo, 1113
 - voiceSUPSInfo, 1144
 - voiceSetCallBarringPwdResp, 1135
 - voiceSetSUPSServiceResp, 1141
- pAlphaID
 - SMSAsyncRawSend_s, 806
- pAlphaIdentifier
 - USSDNoWaitIndicationInfo, 1090
- pAltitudeAssumed
 - QmiCbkLocPositionReportInd, 699
 - unpack_loc_PositionRpt_Ind_t, 976
- pAltitudeSrcInfo
 - LocInjectPositionReq, 372
- pAltitudeWrtEllipsoid
 - LocInjectPositionReq, 372
 - PDSPositionData, 646
 - QmiCbkLocBestAvailPosInd, 688
 - QmiCbkLocPositionReportInd, 699
 - unpack_loc_BestAvailPos_Ind_t, 966
 - unpack_loc_PositionRpt_Ind_t, 976
- pAltitudeWrtMeanSeaLevel
 - LocInjectPositionReq, 372
 - QmiCbkLocBestAvailPosInd, 688
 - QmiCbkLocPositionReportInd, 699

- unpack_loc_BestAvailPos_Ind_t, 966
- unpack_loc_PositionRpt_Ind_t, 976
- pAltitudeWrtSealevel
 - PDSPositionData, 646
- pAmrMode
 - GetIMSVoIPConfigResp, 259
 - imsVoIPConfigInfo, 314
 - SetIMSVoIPConfigReq, 764
- pAmrOctetAligned
 - GetIMSVoIPConfigResp, 259
 - imsVoIPConfigInfo, 314
 - SetIMSVoIPConfigReq, 765
- pAmrWBMode
 - GetIMSVoIPConfigResp, 259
 - imsVoIPConfigInfo, 314
 - SetIMSVoIPConfigReq, 765
- pAmrWBOctetAligned
 - GetIMSVoIPConfigResp, 259
 - imsVoIPConfigInfo, 314
 - SetIMSVoIPConfigReq, 765
- pAmrWbEnable
 - GetIMSVoIPConfigResp, 259
 - imsVoIPConfigInfo, 314
 - SetIMSVoIPConfigReq, 765
- pApnString
 - LibPackprofile_3GPP2, 343
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 673
- pApnStringSize
 - LibPackprofile_3GPP2, 343
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 673
- pApnname
 - pack_wds_SetDefaultProfile_t, 623
- pAppName
 - loc_LocApplicationInfo, 357
 - LocApplicationInfo, 364
- pAppPriority
 - LibPackprofile_3GPP2, 343
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 673
- pAppProvider
 - loc_LocApplicationInfo, 357
 - LocApplicationInfo, 364
- pAppSubType
 - HDRProtSubtypResp, 288
- pAppType
 - LibPackprofile_3GPP2, 343
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 673
- pAppVersion
 - loc_LocApplicationInfo, 357
 - LocApplicationInfo, 364
- pApplicationInfo
 - LOCStartReq, 376
 - pack_loc_Start_t, 581
- pArrAlertingPattern
 - voiceGetAllCallInfo, 1107
- voiceSetAllCallStatusCbkInfo, 1132
- pArrAlertingType
 - voiceGetAllCallInfo, 1107
 - voiceSetAllCallStatusCbkInfo, 1132
- pArrAlphaID
 - voiceGetAllCallInfo, 1107
 - voiceSetAllCallStatusCbkInfo, 1132
- pArrCallEndReason
 - voiceGetAllCallInfo, 1107
 - voiceSetAllCallStatusCbkInfo, 1132
- pArrCallInfo
 - voiceGetAllCallInfo, 1107
- pArrCalledPartyNum
 - voiceGetAllCallInfo, 1107
 - voiceSetAllCallStatusCbkInfo, 1132
- pArrConnectPartyNum
 - voiceGetAllCallInfo, 1107
 - voiceSetAllCallStatusCbkInfo, 1132
- pArrDiagInfo
 - voiceGetAllCallInfo, 1107
 - voiceSetAllCallStatusCbkInfo, 1132
- pArrRedirPartyNum
 - voiceGetAllCallInfo, 1107
 - voiceSetAllCallStatusCbkInfo, 1132
- pArrRemotePartyName
 - voiceGetAllCallInfo, 1107
 - voiceSetAllCallStatusCbkInfo, 1132
- pArrRemotePartyNum
 - voiceGetAllCallInfo, 1107
 - voiceSetAllCallStatusCbkInfo, 1133
- pArrSvcOption
 - voiceGetAllCallInfo, 1107
 - voiceSetAllCallStatusCbkInfo, 1133
- pArrUUSInfo
 - voiceGetAllCallInfo, 1107
- pAuth
 - pack_wds_SLQSSStartDataSession_t, 638
- pAuthPassword
 - LibPackprofile_3GPP2, 343
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 673
- pAuthPassword_tSize
 - LibPackprofile_3GPP2, 343
- pAuthPasswordSize
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 673
- pAuthProtocol
 - LibPackprofile_3GPP2, 343
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 673
- pAuthRetryCount
 - LibPackprofile_3GPP2, 343
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 673
- pAuthTimeout
 - LibPackprofile_3GPP2, 343
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 673

- pAuthenticateResult
 - UIMAuthenticateResp, [891](#)
- pAuthentication
 - qmiWdsRunTimeSettings, [713](#)
 - ssdatasession_params, [826](#)
- pAuthenticationPref
 - LibPackprofile_3GPP, [337](#)
 - LibpackProfile3GPP, [326](#)
 - Profile3GPP, [667](#)
- pAutoAnsStatus
 - voiceSetConfigResp, [1138](#)
- pAutoAnswer
 - voiceGetConfigReq, [1121](#)
 - voiceSetConfigReq, [1136](#)
- pAutoAnswerStat
 - voiceGetConfigResp, [1123](#)
- pAutoSelection
 - UIMGetConfigurationResp, [896](#)
- pAutosdm
 - _SLQSOMADMSettings, [76](#)
 - _SLQSOMADMSettingsReqParams, [77](#)
 - _SLQSOMADMSettingsReqParams3, [78](#)
 - pack_swima_SLQSOMADMSetSettings_t, [611](#)
- pBandPref
 - _sysSelectPrefInfo, [87](#)
 - _sysSelectPrefParams, [93](#)
 - pack_nas_SLQSSetSysSelectionPref_t, [599](#)
 - unpack_nas_SLQSGetSysSelectionPref_t, [1003](#)
- pBdsSVInfo
 - LocDelAssDataReq, [365](#)
 - pack_loc_Delete_Assist_Data_t, [567](#)
- pBearerID
 - QosFlowInfo, [717](#)
- pBearerId
 - swiPDPRuntimeSettingsResp, [839](#)
- pBearerTech
 - DataBearerTechExt, [196](#)
- pBurstDTMFLengths
 - voiceBurstDTMFInfo, [1096](#)
- pCCResType
 - voiceGetCLIPResp, [1114](#)
 - voiceGetCLIRResp, [1116](#)
 - voiceGetCNAPResp, [1117](#)
 - voiceGetCOLPResp, [1118](#)
 - voiceGetCOLRResp, [1120](#)
 - voiceGetCallBarringResp, [1109](#)
 - voiceGetCallFWResp, [1111](#)
 - voiceGetCallWaitInfo, [1113](#)
 - voiceSetCallBarringPwdResp, [1135](#)
- pCCResultType
 - voiceCallResponseParams, [1102](#)
 - voiceSetSUPSServiceResp, [1141](#)
- pCCSUPSType
 - voiceCallResponseParams, [1102](#)
 - voiceGetCLIPResp, [1114](#)
 - voiceGetCLIRResp, [1116](#)
 - voiceGetCNAPResp, [1117](#)
 - voiceGetCOLPResp, [1118](#)
 - voiceGetCOLRResp, [1120](#)
 - voiceGetCallBarringResp, [1109](#)
 - voiceGetCallFWResp, [1111](#)
 - voiceGetCallWaitInfo, [1113](#)
 - voiceSetCallBarringPwdResp, [1135](#)
 - voiceSetSUPSServiceResp, [1141](#)
- pCCSuppsType
 - USSResp, [1092](#)
- pCDMAChannel
 - nasGet3GPP2SubscriptionInfoResp, [497](#)
- pCDMAECIODelta
 - pack_nas_SLQSNasConfigSigInfo2_t, [589](#)
 - setSignalStrengthInfo, [776](#)
- pCDMAECIOThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, [589](#)
 - setSignalStrengthInfo, [776](#)
- pCDMAECIOThreshList
 - CDMAECIOThresh, [147](#)
 - nas_CDMAECIOThresh, [409](#)
- pCDMAFrameErrRate
 - GetErrRateResp, [254](#)
- pCDMAInfo
 - nasCellLocationInfoResp, [495](#)
 - unpack_nas_SLQSNasGetCellLocationInfo_↵
t, [1004](#)
- pCDMARSSIDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, [589](#)
 - setSignalStrengthInfo, [776](#)
- pCDMARSSIThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, [589](#)
 - setSignalStrengthInfo, [776](#)
- pCDMARSSIThreshList
 - CDMARSSIThresh, [153](#)
 - nas_CDMARSSIThresh, [410](#)
- pCDMASSInfo
 - nasGetSigInfoResp, [500](#)
- pCDMASigInfo
 - nasSigInfo, [524](#)
 - unpack_nas_SLQSNasSigInfoCallback_ind_↵
t, [1007](#)
- pCDMASrvStatusInfo
 - nasGetSysInfoResp, [502](#)
 - nasSysInfo, [530](#)
 - unpack_nas_SLQSGetSysInfo_t, [999](#)
 - unpack_nas_SLQSSysInfoCallback_ind_t, [1013](#)
- pCDMASysInfo
 - nasGetSysInfoResp, [502](#)
 - nasSysInfo, [530](#)
 - unpack_nas_SLQSGetSysInfo_t, [999](#)
 - unpack_nas_SLQSSysInfoCallback_ind_t, [1013](#)
- pCLIPResp
 - voiceGetCLIPResp, [1114](#)
- pCLIPstatus
 - voiceSUPSInfo, [1144](#)
- pCLIRCause
 - voicInfoRec, [1126](#)
- pCLIRResp
 - voiceGetCLIRResp, [1116](#)

- pCLIRType
 - voiceCallRequestParams, 1101
- pCLIRstatus
 - voiceSUPSInfo, 1144
- PCMparams, 642
 - iFaceTab, 642
 - iFaceTabLen, 642
- pCNAPResp
 - voiceGetCNAPResp, 1117
- pCNAPstatus
 - voiceSUPSInfo, 1144
- pCOLPResp
 - voiceGetCOLPResp, 1118
- pCOLPstatus
 - voiceSUPSInfo, 1144
- pCOLRResp
 - voiceGetCOLRResp, 1120
- pCOLRstatus
 - voiceSUPSInfo, 1144
- PCSCFAddrPCO
 - unpack_wds_SLQSGetRuntimeSettings_t, 1078
- PCSCFFQDNAddrList
 - unpack_wds_SLQSGetRuntimeSettings_t, 1078
- PCSCFFQDNAddress, 642
 - fqdnAddr, 643
 - fqdnLen, 643
- PCSCFFQDNAddressList, 643
 - numInstances, 643
 - pcsfFQDNAddress, 644
- PCSCFIPv4ServerAddressList, 644
 - numInstances, 644
 - pcsfIPv4Addr, 644
- pCSCFPortName
 - imsRegMgrConfigInfo, 309
 - SetRegMgrConfigReq, 772
- pCSCFPortNameLen
 - SetRegMgrConfigReq, 772
- pCSGID
 - _sysSelectPrefParams, 93
 - pack_nas_SLQSSetSysSelectionPref_t, 599
- pCUGIndex
 - voiceSUPSNotification, 1147
- pCUGInfo
 - voiceCallRequestParams, 1101
- pCallBarPasswd
 - voiceSUPSInfo, 1144
- pCallBarringPasswd
 - voiceSetSUPSServiceReq, 1140
- pCallEndReason
 - getDUNCallInfoResp, 252
- pCallFWNum
 - voiceSUPSInfo, 1144
- pCallFWTimerVal
 - voiceSUPSInfo, 1144
- pCallForwardingNumber
 - voiceSetSUPSServiceReq, 1140
- pCallFwdInfo
 - voiceSUPSInfo, 1144
- pCallFwdTypeAndPlan
 - voiceSetSUPSServiceReq, 1140
- pCallID
 - burstDTMFInfo, 125
 - voiceCallResponseParams, 1102
 - voiceContDTMFInfo, 1103
 - voiceFlashInfo, 1104
 - voiceGetCLIPResp, 1114
 - voiceGetCLIRResp, 1116
 - voiceGetCNAPResp, 1117
 - voiceGetCOLPResp, 1118
 - voiceGetCOLRResp, 1120
 - voiceGetCallBarringResp, 1109
 - voiceGetCallFWResp, 1111
 - voiceGetCallWaitInfo, 1113
 - voiceManageCallsReq, 1128
 - voiceSUPSInfo, 1144
 - voiceSetCallBarringPwdResp, 1135
 - voiceSetSUPSServiceResp, 1141
- pCallId
 - USSResp, 1092
 - voiceAnswerCall, 1095
- pCallInfo
 - voiceCallInfoResp, 1099
- pCallPartySubAdd
 - voiceCallRequestParams, 1101
- pCallType
 - voiceCallRequestParams, 1101
- pCallWaitInd
 - voiceInfoRec, 1126
- pCallbackAddr
 - cdmaMsgEncodingParams, 152
- pCallbkAddr
 - cdmaMsgDecodingParams, 151
- pCallbkAddrLength
 - cdmaMsgDecodingParams, 151
- pCalledPartyInfo
 - voiceInfoRec, 1126
- pCallerIDInfo
 - voiceInfoRec, 1126
- pCallerNameInfo
 - voiceInfoRec, 1126
- pCallingPartyInfo
 - voiceInfoRec, 1126
- pCardResult
 - UIMAuthenticateResp, 891
 - UIMGetFileAttributesResp, 897
 - UIMReadTransparentResp, 902
 - unpack_uim_ReadTransparent_t, 1055
- pCardStatus
 - UIMGetCardStatusResp, 895
 - unpack_uim_GetCardStatus_t, 1054
 - unpack_uim_SLQSUIMSetStatusChangeCall↔
Back_ind_t, 1058
- pCcResultType
 - USSResp, 1092
- pCellDb
 - LocDelAssDataReq, 365

- pack_loc_Delete_Assist_Data_t, 567
- pChangeDuration
 - nasInitNetworkReq, 508
- pack_nas_SLQSIInitiateNetworkRegistration_t, 585
- pChannelRate
 - getDUNCallInfoResp, 252
- pChgDuration
 - _sysSelectPrefParams, 93
 - pack_nas_SLQSSetSysSelectionPref_t, 599
- pClkInfo
 - LocDelAssDataReq, 365
 - pack_loc_Delete_Assist_Data_t, 567
- pCodecSTGain
 - GetAudioPathConfigResp, 242
 - SetAudioPathConfigReq, 756
- pColorCode
 - nasGetHDRColorCodeResp, 498
- pConfidence
 - LocSetCradleMountReq, 374
- pConfigAltitudeAssumed
 - LOCStartReq, 376
 - pack_loc_Start_t, 581
- pConfigurationMask
 - UIMGetConfigurationReq, 895
- pConnectNumInfo
 - voiceCallInfoResp, 1099
 - voiceInfoRec, 1127
- pConnectionStatus
 - getDUNCallInfoResp, 252
- pContextId
 - swiPDPRuntimeSettingsResp, 839
- pCrashInfo
 - CrashInfoParams, 174
- pCrashString
 - CrashInfo, 173
- pCreateProfileOut
 - unpack_wds_SLQSCreateProfile_t, 1071
- pCurAMRConfig
 - voiceGetConfigResp, 1123
- pCurDataBearerTechnology
 - dataBearers, 193
- pCurPrefVoiceSO
 - voiceGetConfigResp, 1123
- pCurProfile
 - pack_wds_SLQSCreateProfile_t, 628
- pCurVoiceDomainPref
 - voiceGetConfigResp, 1123
- pCurVoicePrivacyPref
 - voiceGetConfigResp, 1123
- pCurrChannelRateInd
 - pack_wds_SLQSWdsSetEventReport_t, 640
 - wdsSetEventReportReq, 1191
- pCurrDataBearerTechInd
 - pack_wds_SLQSWdsSetEventReport_t, 640
 - wdsSetEventReportReq, 1191
- pCurrImgInfo
 - CurrentImgList, 182
 - unpack_dms_SLQSSwiGetFirmwareCurr_t, 949
- pCurrNetworkInfo
 - CurrDataSysStat, 180
- pCurrPrefDataSysInd
 - pack_wds_SLQSWdsSetEventReport_t, 640
 - wdsSetEventReportReq, 1191
- pCurrTTYMode
 - voiceGetConfigResp, 1123
- pCurrentChannelRXRate
 - WdsConnectionRateElmnts, 1177
- pCurrentChannelTXRate
 - WdsConnectionRateElmnts, 1177
- pCurrentPersonality
 - HDRPersonalityInd, 287
 - HDRPersonalityResp, 287
- pCurrentPrsnlty
 - HDRProtSubtypResp, 288
- pCurrentmitigationLvl
 - TmdGetMitigationLvlResp, 865
- pCustSettingInfo
 - DMSgetCustomFeatureV2, 212
 - getCustomFeatureV2, 248
- pCustSettingList
 - DMSgetCustomFeatureV2, 212
 - getCustomFeatureV2, 248
- pCwtMute
 - SetM2MAVMuteReq, 769
 - SetM2MAudioProfileReq, 768
- pDDTMInd
 - nasIndicationRegisterReq, 507
 - pack_nas_SLQSNasIndicationRegisterExt_t, 593
- pDHCPRelayEnabled
 - custFeaturesInfo, 189
 - custFeaturesSetting, 191
- PDOP
 - loc_precisionDilution, 358
 - precisionDilution_s, 660
- PDPTType
 - unpack_wds_SLQSGetRuntimeSettings_t, 1078
- pDRCPParams
 - GetHRPDStatsResp, 255
- PDS_SRV
 - qaGobiApiCbk.h, 1327
- PDSInjectTimeReference
 - qaGobiApiPds.h, 1554
- PDSPosMethodStateReq, 646
 - pWifiState, 647
 - pXtraDataState, 647
 - pXtraTimeState, 647
- PDSPositionData, 644
 - pAltitudeWrtEllipsoid, 646
 - pAltitudeWrtSealevel, 646
 - pHorizontalConfidence, 646
 - pHorizontalUncCircular, 646
 - pLatitude, 646
 - pLongitude, 646
 - pPositionSource, 646
 - pTimeStamp, 646
 - pTimeType, 646

- pVerticalConfidence, [646](#)
 - pVerticalUnc, [646](#)
- pDTMFTXGain
 - GetAudioPathConfigResp, [243](#)
 - SetAudioPathConfigReq, [756](#)
- pDataBearer
 - QosEventInfo, [716](#)
 - slqsWdsEventInfo, [804](#)
 - unpack_wds_GetDataBearerTechnology_t, [1063](#)
- pDataBearerTech
 - getDUNCallInfoResp, [252](#)
- pDataBearerTechInd
 - pack_wds_SLQSWdsSetEventReport_t, [640](#)
 - wdsSetEventReportReq, [1191](#)
- pDataCallStatusChangeInd
 - pack_wds_SLQSWdsSetEventReport_t, [640](#)
 - wdsSetEventReportReq, [1191](#)
- pDataMode
 - LibPackprofile_3GPP2, [343](#)
 - LibpackProfile3GPP2, [332](#)
 - Profile3GPP2, [673](#)
- pDataRate
 - LibPackprofile_3GPP2, [343](#)
 - LibpackProfile3GPP2, [332](#)
 - Profile3GPP2, [673](#)
 - swiQosFlow, [845](#)
- pDataSrc
 - voiceSUPSInfo, [1144](#)
- pDataStatusDetail
 - NetworkDebugResp, [534](#)
- pDataSystemStatusChangeInd
 - pack_wds_SLQSWdsSetEventReport_t, [640](#)
 - wdsSetEventReportReq, [1191](#)
- pDate
 - _SLQSOMADMSessionInfo, [73](#)
- pDateLength
 - _SLQSOMADMSessionInfo, [73](#)
- pDayItSavAdj
 - nasNetworkTime, [510](#)
 - unpack_nas_SLQSNasNetworkTimeCallBack_↔
ind_t, [1006](#)
- pDefaultPDNEnabled
 - _slqs3GPPConfigItem, [69](#)
 - pack_wds_SLQSSet3GPPConfigItem_t, [634](#)
- pDeferTime
 - pack_swioa_SLQSOMADMSendSelection_↔
t, [610](#)
- pDescription
 - QmiNas3GppNetworkInfo, [707](#)
- pDestAddr
 - cdmaMsgEncodingParams, [153](#)
 - wcdmaMsgEncodingParams, [1153](#)
- pDestSMSContent
 - getDyingGaspCfg, [253](#)
 - pack_dms_SLQSSwiSetDyingGaspCfg_t, [558](#)
 - packgetDyingGaspCfg, [641](#)
 - setDyingGaspCfg, [760](#)
- pDestSMSNum
 - getDyingGaspCfg, [253](#)
 - pack_dms_SLQSSwiSetDyingGaspCfg_t, [558](#)
 - packgetDyingGaspCfg, [641](#)
 - setDyingGaspCfg, [760](#)
- pDetachAction
 - PSDetachReq, [677](#)
- pDevCrashStatus
 - CrashInfoParams, [174](#)
- pDeviceConfigDetail
 - NetworkDebugResp, [534](#)
- pDiagInfo
 - voiceCallInfoResp, [1099](#)
- pDigitBuff
 - burstDTMFInfo, [125](#)
- pDirNum
 - nasGet3GPP2SubscriptionInfoResp, [497](#)
- pDisableIMSI
 - custFeaturesInfo, [189](#)
- pDisplInfo
 - voiceInfoRec, [1127](#)
- pDisplayMode
 - cdmaMsgDecodingParams, [151](#)
- pDomainList
 - qmiWdsRunTimeSettings, [713](#)
- pDormancyStatus
 - getDUNCallInfoResp, [252](#)
 - slqsWdsEventInfo, [804](#)
- pDormancyStatusInd
 - pack_wds_SLQSWdsSetEventReport_t, [640](#)
 - wdsSetEventReportReq, [1191](#)
- pDualStandByPrefInd
 - nasIndicationRegisterReq, [507](#)
 - pack_nas_SLQSNasIndicationRegisterExt_t, [593](#)
- pECIOThresList
 - ECIOThresh, [220](#)
- pECIOThresh
 - sigInfo, [782](#)
- pECMode
 - GetAudioPathConfigResp, [243](#)
 - SetAudioPathConfigReq, [756](#)
- pECTNum
 - voiceSUPSNotification, [1147](#)
- pESNString
 - serialNumbersInfo, [748](#)
- pEVDOPageMonPerChangeInd
 - pack_wds_SLQSWdsSetEventReport_t, [640](#)
 - wdsSetEventReportReq, [1192](#)
- pEarMute
 - SetM2MAudioProfileReq, [768](#)
- pEmerMode
 - _sysSelectPrefInfo, [87](#)
 - _sysSelectPrefParams, [93](#)
 - pack_nas_SLQSSetSysSelectionPref_t, [599](#)
 - unpack_nas_SLQSGetSysSelectionPref_t, [1003](#)
- pEmergencyCategory
 - voiceCallRequestParams, [1101](#)
- pEnableNotification
 - pack_wds_DHCPv4ClientLeaseChange_t, [619](#)

- WdsClientLeaseChange, 1176
- pEnabled
 - pack_wds_SetMobileIPProfile_t, 627
- pEncodingAlphabet
 - cdmaMsgEncodingParams, 153
- pEncryptData
 - pack_uim_ReadTransparent_t, 613
 - UIMReadTransparentReq, 901
- pEncryptedData
 - UIMReadTransparentResp, 902
 - unpack_uim_ReadTransparent_t, 1055
- pEncryptedPIN1
 - pack_uim_VerifyPin_t, 619
 - UIMPinResp, 899
 - UIMVerifyPinReq, 913
 - unpack_uim_ChangePin_t, 1053
 - unpack_uim_SetPinProtection_t, 1056
 - unpack_uim_UnblockPin_t, 1059
 - unpack_uim_VerifyPin_t, 1060
- PER
 - NetworkStatEVDO, 538
- pError
 - USSDNoWaitIndicationInfo, 1090
- pErrorCodeStr
 - imsaRatStatusInfo, 301
- pErrorMask
 - CATEventDataType, 142
- pErrorRateInd
 - nasIndicationRegisterReq, 507
 - pack_nas_SLQSNasIndicationRegisterExt_t, 593
- pEspSpi
 - swiQosFilter, 842
- pEtwsMessageInfo
 - SMSEventInfo_s, 812
- pEtwsPlmnInfo
 - SMSEventInfo_s, 812
- pExtDispInfo
 - voiceInfoRec, 1127
- pExtDispRecInfo
 - voiceInfoRec, 1127
- pExtErrCode
 - _GetProfileSettingOut, 59
 - UnPackGetProfileSettingOut, 1086
- pExtErrorCode
 - CreateProfileOut, 177
 - ModifyProfileOut, 403
 - unpack_wds_SLQSMModifyProfile_t, 1079
- pFOTAUpdate
 - _SLQSOMADMSSettings, 76
- pFOTAdownload
 - _SLQSOMADMSSettings, 76
- pFailCause
 - voiceGetCLIPResp, 1114
 - voiceGetCLIRResp, 1116
 - voiceGetCNAPResp, 1117
 - voiceGetCOLPResp, 1118
 - voiceGetCOLRResp, 1120
 - voiceGetCallBarringResp, 1109
 - voiceGetCallFWResp, 1112
 - voiceGetCallWaitInfo, 1113
 - voiceManageCallsResp, 1128
 - voiceSUPSInfo, 1144
 - voiceSetCallBarringPwdResp, 1135
 - voiceSetSUPSServiceResp, 1141
- pFailErrorCode
 - imsaPdpStatusInfo, 301
- pFailureCause
 - USSDNoWaitIndicationInfo, 1090
- pFailureReason
 - unpack_wds_SLQSStartDataSession_t, 1084
- pFile
 - ERIFileparams, 222
- pFileAttributes
 - UIMGetFileAttributesResp, 897
- pFileSize
 - ERIFileparams, 222
- pFixId
 - QmiCbkLocPositionReportInd, 699
 - unpack_loc_PositionRpt_Ind_t, 976
- pFlag
 - RXPCMIIRFiltr, 737
 - TXPCMIIRFiltr, 874
- pFlashPayLd
 - voiceFlashInfo, 1104
- pFlashType
 - voiceFlashInfo, 1104
- pFollowOnDC
 - slqssendasyncsmsparams_s, 794
- pForbidden
 - QmiNas3GppNetworkInfo, 707
- pForceOnDC
 - slqssendasyncsmsparams_s, 794
- pFwAutoCheck
 - _SLQSOMADMSSettings, 76
 - _SLQSOMADMSSettingsReqParams3, 78
 - pack_swima_SLQSOMADMSSetSettings_t, 611
- pGCDumpString
 - CrashInfo, 173
- pGERANInfo
 - nasCellLocationInfoResp, 495
 - unpack_nas_SLQSNasGetCellLocationInfo_←
t, 1004
- pGPRSGrantedQoS
 - qmiWdsRunTimeSettings, 713
- pGPRSMInimumQoS
 - LibPackprofile_3GPP, 337
 - LibpackProfile3GPP, 326
 - Profile3GPP, 667
- pGPRSRequestedQos
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 326
 - Profile3GPP, 668
- pGPSEnable
 - custFeaturesSetting, 191
- pGPSLPM
 - custFeaturesInfo, 189

- custFeaturesSetting, 191
- pGPSSel
 - custFeaturesInfo, 189
 - custFeaturesSetting, 191
- pGSMBER
 - GetErrRateResp, 254
- pGSMCallBarringSysInfo
 - nasGetSysInfoResp, 502
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 999
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1013
- pGSMCipherDomainSysInfo
 - nasGetSysInfoResp, 502
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 999
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1013
- pGSMRSSIDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, 589
 - setSignalStrengthInfo, 776
- pGSMRSSIThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, 589
 - setSignalStrengthInfo, 776
- pGSMRSSIThreshList
 - GSMRSSIThresh, 280
 - nas_GSMRSSIThresh, 427
- pGSMSSInfo
 - nasGetSigInfoResp, 500
- pGSMSSigInfo
 - nasSigInfo, 524
 - unpack_nas_SLQSNasSigInfoCallback_ind_t, 1007
- pGSMSrvStatusInfo
 - nasGetSysInfoResp, 502
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 999
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1013
- pGSMSysInfo
 - nasGetSysInfoResp, 502
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 999
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1014
- pGWAcqOrderPref
 - _sysSelectPrefInfo, 87
 - _sysSelectPrefParams, 93
 - pack_nas_SLQSSetSysSelectionPref_t, 599
 - unpack_nas_SLQSGetSysSelectionPref_t, 1003
- pGWAddressV4
 - qmiWdsRunTimeSettings, 713
- pGenerator
 - GetM2MAudioProfileReq, 261
 - SetM2MAudioProfileReq, 768
- pGetCallFWExtInfo
 - voiceGetCallFWResp, 1112
- pGetCallFWInfo
 - voiceGetCallFWResp, 1112
- pGetCustomInput
 - DMSgetCustomFeatureV2, 212
 - getCustomFeatureV2, 248
- pGetDyingGaspCfg
 - unpack_dms_SLQSSwiGetDyingGaspCfg_t, 948
- pGetDyingGaspStatistics
 - unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 948
- pGnssData
 - LocDelAssDataReq, 365
 - pack_loc_Delete_Assist_Data_t, 567
- pGpsTime
 - QmiCbkLocBestAvailPosInd, 688
 - QmiCbkLocPositionReportInd, 700
 - unpack_loc_BestAvailPos_Ind_t, 967
 - unpack_loc_PositionRpt_Ind_t, 976
- pGyroAcceptReady
 - QmiCbkLocSensorStreamingInd, 701
- pGyroData
 - LocInjectSensorDataReq, 373
- pGyroSamplesAccepted
 - QmiCbkLocInjectSensorDataInd, 693
- pGyroTempAcceptReady
 - QmiCbkLocSensorStreamingInd, 702
- pGyroTempData
 - LocInjectSensorDataReq, 374
- pGyroTempSamplesAccepted
 - QmiCbkLocInjectSensorDataInd, 693
- pGyroTimeSrc
 - LocInjectSensorDataReq, 374
- pHA2002bis
 - pack_wds_SetMobileIPParameters_t, 625
- pHAAAuthenticator
 - pack_wds_SetMobileIPParameters_t, 626
- pHASPI
 - pack_wds_SetMobileIPProfile_t, 627
- pHDRECIODelta
 - pack_nas_SLQSNasConfigSigInfo2_t, 589
 - setSignalStrengthInfo, 776
- pHDRECIOThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, 589
 - setSignalStrengthInfo, 776
- pHDRECIOThreshList
 - HDRECIOThresh, 286
 - nas_HDRECIOThresh, 432
- pHDRIODelta
 - pack_nas_SLQSNasConfigSigInfo2_t, 589
 - setSignalStrengthInfo, 777
- pHDRIOThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, 589
 - setSignalStrengthInfo, 777
- pHDRIOThreshList
 - HDRIOThresh, 286
 - nas_HDRIOThresh, 432
- pHDRNewUATIAssInd
 - nasIndicationRegisterReq, 507
 - pack_nas_SLQSNasIndicationRegisterExt_t, 593
- pHDRPackErrRate
 - GetErrRateResp, 254
- pHRRSSIDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, 589

- setSignalStrengthInfo, [777](#)
- pHRRSSIThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, [589](#)
 - setSignalStrengthInfo, [777](#)
- pHRRSSIThreshList
 - HDRSSIThresh, [289](#)
 - nas_HDRSSIThresh, [433](#)
- pHRSINRDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, [589](#)
 - setSignalStrengthInfo, [777](#)
- pHRSINRThreshList
 - HDRSINRThresh, [290](#)
- pHRSINRThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, [589](#)
 - setSignalStrengthInfo, [777](#)
 - sigInfo, [782](#)
- pHRSINRThreshList
 - HDRSINRThreshold, [290](#)
 - nas_HDRSINRThreshold, [434](#)
- pHRRSSInfo
 - nasGetSigInfoResp, [500](#)
- pHDRSessionCloseInd
 - nasIndicationRegisterReq, [507](#)
 - pack_nas_SLQSNasIndicationRegisterExt_t, [593](#)
- pHDRSigInfo
 - nasSigInfo, [524](#)
 - unpack_nas_SLQSNasSigInfoCallback_ind_t, [1007](#)
- pHDRSrvStatusInfo
 - nasGetSysInfoResp, [503](#)
 - nasSysInfo, [530](#)
 - unpack_nas_SLQSSysInfo_t, [999](#)
 - unpack_nas_SLQSSysInfoCallback_ind_t, [1014](#)
- pHDRSysInfo
 - nasGetSysInfoResp, [503](#)
 - nasSysInfo, [530](#)
 - unpack_nas_SLQSSysInfo_t, [1000](#)
 - unpack_nas_SLQSSysInfoCallback_ind_t, [1014](#)
- pHaltSubscription
 - UIMGetConfigurationResp, [896](#)
- pHeading
 - QmiCbkLocBestAvailPosInd, [688](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [976](#)
- pHeadingUnc
 - QmiCbkLocBestAvailPosInd, [688](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [976](#)
- pHomeSIDNID
 - nasGet3GPP2SubscriptionInfoResp, [497](#)
- pHorCirConf
 - QmiCbkLocBestAvailPosInd, [688](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
- pHorConfidence
 - LocInjectPositionReq, [372](#)
 - QmiCbkLocPositionReportInd, [700](#)
- unpack_loc_PositionRpt_Ind_t, [976](#)
- pHorEllpConf
 - QmiCbkLocBestAvailPosInd, [688](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
- pHorReliability
 - LocInjectPositionReq, [372](#)
 - QmiCbkLocBestAvailPosInd, [688](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [976](#)
- pHorUncCircular
 - LocInjectPositionReq, [372](#)
 - QmiCbkLocBestAvailPosInd, [688](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [976](#)
- pHorUncEllipseOrientAzimuth
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [976](#)
- pHorUncEllipseSemiMajor
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [976](#)
- pHorUncEllipseSemiMinor
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [976](#)
- pHorizontalAccuracyLvl
 - LOCStartReq, [376](#)
 - pack_loc_Start_t, [582](#)
- pHorizontalConfidence
 - PDSPositionData, [646](#)
- pHorizontalUncCircular
 - PDSPositionData, [646](#)
- pHotSwapStatus
 - UIMGetCardStatusResp, [895](#)
 - unpack_uim_GetCardStatus_t, [1054](#)
- pHwConfig
 - pack_wds_SLQSSetDHCPv4ClientConfig_t, [636](#)
 - unpack_wds_SLQSSetDHCPv4ClientConfig_t, [1083](#)
 - WdsDHCPv4Config, [1179](#)
- PIFACEId
 - SetM2MAudioAVCFGRReq, [766](#)
- pIMCNflag
 - qmiWdsRunTimeSettings, [713](#)
- pIMEIString
 - serialNumbersInfo, [748](#)
- pIMSDomain
 - GetIMSUserConfigParams, [257](#)
 - imsUserConfigInfo, [312](#)
 - SetIMSUserConfigReq, [762](#)
- pIMSDomainLen
 - GetIMSUserConfigParams, [257](#)

- SetIMSUserConfigReq, 762
- plMSTestMode
 - GetRegMgrConfigParams, 267
 - imsRegMgrConfigInfo, 309
 - SetRegMgrConfigReq, 772
- plOThresList
 - IOTresh, 317
- plOThresh
 - sigInfo, 782
- plPAddressV4
 - QmiWdsIpAddressInfo, 710
 - qmiWdsRunTimeSettings, 713
- plPAddressV6
 - QmiWdsIpAddressInfo, 710
- plPFamSupport
 - custFeaturesInfo, 189
- plPFamily
 - GetInstIDResp, 260
- plPFamilyPreference
 - qmiWdsRunTimeSettings, 713
- plPV6AddrInfo
 - qmiWdsRunTimeSettings, 713
- plPV6GWAddrInfo
 - qmiWdsRunTimeSettings, 713
- plPv4Addr
 - WdsDHCPv4ClientLeaseInd, 1178
- plPv4AddrPref
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 326
 - Profile3GPP, 668
- plPv4Address
 - swiPDPRuntimeSettingsResp, 839
- plPv4DstAddr
 - swiQosFilter, 842
- plPv4GWAddress
 - swiPDPRuntimeSettingsResp, 839
- plPv4SrcAddr
 - swiQosFilter, 842
- plPv6AddPref
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 326
 - Profile3GPP, 668
- plPv6Address
 - swiPDPRuntimeSettingsResp, 839
- plPv6DstAddr
 - swiQosFilter, 842
- plPv6GWAddress
 - swiPDPRuntimeSettingsResp, 839
- plPv6Label
 - swiQosFilter, 842
- plPv6SrcAddr
 - swiQosFilter, 842
- plPv6TrafCls
 - swiQosFilter, 842
- plPv6prefixlen
 - QmiWdsIpAddressInfo, 710
- pld
 - swiQosFilter, 842
- plds
 - swiQosIds, 847
- plgnoreHotSwapSwitch
 - pack_uim_SLQSUIMPowerUp_t, 616
 - UIMPowerUpReq, 900
- plmCnFlag
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 326
 - Profile3GPP, 668
- plmageList
 - pack_fms_SetImagesPreference_t, 566
 - unpack_fms_GetImagesPreference_t, 961
- plmeiSvnString
 - serialNumbersInfo, 748
- plmgType
 - FirmwareUpdatStat, 231
- plmsRegErrCode
 - IMSARegistrationStatus, 302
- plmsRegStatus
 - IMSARegistrationStatus, 302
 - imsaRegStatusInfo, 303
- plnUse
 - QmiNas3GppNetworkInfo, 707
- plndFieldsList
 - IMSASupportedFieldsResp, 306
- plndicationToken
 - pack_uim_ChangePin_t, 612
 - pack_uim_ReadTransparent_t, 613
 - pack_uim_SetPinProtection_t, 614
 - pack_uim_UnblockPin_t, 618
 - pack_uim_VerifyPin_t, 619
 - UIMAuthenticateReq, 891
 - UIMAuthenticateResp, 891
 - UIMChangePinReq, 892
 - UIMGetFileAttributesReq, 897
 - UIMGetFileAttributesResp, 897
 - UIMPINResp, 899
 - UIMReadTransparentReq, 901
 - UIMReadTransparentResp, 902
 - UIMSetPinProtectionReq, 908
 - UIMUnblockPinReq, 912
 - UIMVerifyPinReq, 913
 - unpack_uim_ChangePin_t, 1053
 - unpack_uim_ReadTransparent_t, 1055
 - unpack_uim_SetPinProtection_t, 1056
 - unpack_uim_UnblockPin_t, 1059
 - unpack_uim_VerifyPin_t, 1060
- plnstanceID
 - GetInstIDResp, 260
- plnstanceSize
 - QmiNasPerformNetworkScanResp, 708
- plnstances
 - QmiNasPerformNetworkScanResp, 708
- plnstancesSize
 - QmiNasGetRFBandInfoResp, 708
- plntermediateReportState
 - LOCStartReq, 376
 - pack_loc_Start_t, 582

- plpcpAckTimeout
 - LibPackprofile_3GPP2, [343](#)
 - LibpackProfile3GPP2, [332](#)
 - Profile3GPP2, [673](#)
- plpcpCreqRetryCount
 - LibPackprofile_3GPP2, [343](#)
 - LibpackProfile3GPP2, [332](#)
 - Profile3GPP2, [673](#)
- plsPcscfAddressNedded
 - LibPackprofile_3GPP2, [344](#)
 - LibpackProfile3GPP2, [332](#)
 - Profile3GPP2, [674](#)
- plsUDHPresent
 - wcdmaLongMsgDecodingParams, [1151](#)
- plsVoiceEnabled
 - custFeaturesInfo, [189](#)
 - custFeaturesSetting, [191](#)
- pJitter
 - swiQosFlow, [845](#)
- pKeyReferenceID
 - pack_uim_ChangePin_t, [612](#)
 - pack_uim_SetPinProtection_t, [614](#)
 - pack_uim_UnblockPin_t, [618](#)
 - pack_uim_VerifyPin_t, [619](#)
 - UIMChangePinReq, [892](#)
 - UIMSetPinProtectionReq, [908](#)
 - UIMUnblockPinReq, [912](#)
 - UIMVerifyPinReq, [913](#)
- PLMN_LENGTH
 - qaGobiApiNas.h, [1499](#)
- PLMNData
 - operatorPLMNList, [551](#)
- PLMNName
 - operatorNameString, [550](#)
- PLMNNetName
 - PLMNNetworkName, [657](#)
- PLMNNetworkName, [657](#)
 - numInstance, [657](#)
 - PLMNNetName, [657](#)
- PLMNNetworkNameData, [657](#)
 - codingScheme, [659](#)
 - countryInitials, [659](#)
 - longName, [659](#)
 - longNameLen, [659](#)
 - longNameSpareBits, [659](#)
 - shortName, [659](#)
 - shortNameLen, [659](#)
 - shortNameSpareBits, [659](#)
- PLMNRecID
 - OperatorPLMNData, [551](#)
- pLTEAttachProfile
 - _slqs3GPPConfigItem, [69](#)
 - pack_wds_SLQSSet3GPPConfigItem_t, [634](#)
- pLTEAttachProfileList
 - _slqs3GPPConfigItem, [69](#)
 - pack_wds_SLQSSet3GPPConfigItem_t, [634](#)
- pLTEBandPref
 - _sysSelectPrefInfo, [87](#)
 - _sysSelectPrefParams, [93](#)
 - pack_nas_SLQSSetSysSelectionPref_t, [600](#)
 - unpack_nas_SLQSGetSysSelectionPref_t, [1003](#)
- pLTCEphyCa
 - nasIndicationRegisterReq, [507](#)
 - pack_nas_SLQSNasIndicationRegisterExt_t, [593](#)
- pLTEInfo
 - swiModemStatusResp, [836](#)
 - unpack_nas_SLQSNasSwiModemStatus_t, [1008](#)
- pLTEInfoInterfreq
 - nasCellLocationInfoResp, [495](#)
 - unpack_nas_SLQSNasGetCellLocationInfo_↵
t, [1004](#)
- pLTEInfoIntrafreq
 - nasCellLocationInfoResp, [495](#)
 - unpack_nas_SLQSNasGetCellLocationInfo_↵
t, [1005](#)
- pLTEInfoNeighboringGSM
 - nasCellLocationInfoResp, [495](#)
 - unpack_nas_SLQSNasGetCellLocationInfo_↵
t, [1005](#)
- pLTEInfoNeighboringWCDMA
 - nasCellLocationInfoResp, [495](#)
 - unpack_nas_SLQSNasGetCellLocationInfo_↵
t, [1005](#)
- pLTERSRPDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pLTERSRPThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pLTERSRPThreshList
 - LTERSRPThresh, [388](#)
 - nas_LTERSRPThresh, [446](#)
- pLTERSRQDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pLTERSRQThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pLTERSRQThreshList
 - LTERSRQThresh, [389](#)
 - nas_LTERSRQThresh, [447](#)
- pLTERSSIDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pLTERSSIThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pLTERSSIThreshList
 - LTERSSIThresh, [389](#)
 - nas_LTERSSIThresh, [447](#)
- pLTESNRDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pLTESNRThresList
 - LTESNRThresh, [393](#)
- pLTESNRThresh

- pack_nas_SLQSNasConfigSigInfo2_t, 590
- setSignalStrengthInfo, 777
- sigInfo, 782
- pLTESNRThreshList
 - LTESNRThreshold, 393
 - nas_LTESNRThreshold, 449
- pLTESSInfo
 - nasGetSigInfoResp, 500
- pLTESigInfo
 - nasSigInfo, 524
 - unpack_nas_SLQSNasSigInfoCallback_ind_↔
t, 1007
- pLTESigRptCfg
 - sigInfo, 782
- pLTESigRptConfig
 - pack_nas_SLQSNasConfigSigInfo2_t, 590
 - setSignalStrengthInfo, 777
- pLTESrvStatusInfo
 - nasGetSysInfoResp, 503
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 1000
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1014
- pLTESysInfo
 - nasGetSysInfoResp, 503
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 1000
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1014
- pLTEVoiceSupportSysInfo
 - nasGetSysInfoResp, 503
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 1000
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1014
- pLanguage
 - cdmaMsgDecodingParams, 151
- pLastBearerTech
 - DataBearerTechExt, 196
- pLastCallDataBearerTech
 - getDUNCallInfoResp, 252
- pLastCallDataBearerTechnology
 - dataBearers, 193
- pLastCallRXOKBytesCnt
 - getDUNCallInfoResp, 252
- pLastCallTXOKBytesCnt
 - getDUNCallInfoResp, 252
- pLatency
 - swiQosFlow, 845
- pLatitude
 - LocInjectPositionReq, 372
 - PDSPositionData, 646
 - QmiCbkLocBestAvailPosInd, 689
 - QmiCbkLocPositionReportInd, 700
 - unpack_loc_BestAvailPos_Ind_t, 967
 - unpack_loc_PositionRpt_Ind_t, 976
- pLcpAckTimeout
 - LibPackprofile_3GPP2, 344
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 674
- pLcpCreqRetryCount
 - LibPackprofile_3GPP2, 344
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 674
- pLeapSeconds
 - QmiCbkLocPositionReportInd, 700
 - unpack_loc_PositionRpt_Ind_t, 976
- pLeaseState
 - WdsDHCPv4ClientLeaseInd, 1178
- pLineCtrlInfo
 - voiceInfoRec, 1127
- pLinktimer
 - pack_sms_SendSMS_t, 603
 - slqssendasyncsmsparams_s, 794
 - slqssendsmsparams_s, 796
- pLogString
 - FirmwareUpdatStat, 231
- pLogStringLength
 - FirmwareUpdatStat, 231
- pLongitude
 - LocInjectPositionReq, 372
 - PDSPositionData, 646
 - QmiCbkLocBestAvailPosInd, 689
 - QmiCbkLocPositionReportInd, 700
 - unpack_loc_BestAvailPos_Ind_t, 967
 - unpack_loc_PositionRpt_Ind_t, 976
- pLoopbackMode
 - WDSSetLoopbackData, 1192
- pLoopbackMultiplier
 - WDSSetLoopbackData, 1192
- pLteBandCapability
 - BandCapabilityResp, 122
- pLteEARFCN
 - nasSwiGetChannelLockResp, 525
 - nasSwiSetChannelLockReq, 527
- pLteNasRelInfo
 - SwiOTAMsg_s, 837
- pLtePCI
 - nasSwiGetChannelLockResp, 525
 - nasSwiSetChannelLockReq, 527
- pLteQci
 - swiQosFlow, 845
- pMCC
 - QmiNas3GppNetworkInfo, 707
- pMEIDString
 - serialNumbersInfo, 748
- pMICGainSelect
 - GetAudioPathConfigResp, 243
- pMIPStatusInd
 - pack_wds_SLQSWdsSetEventReport_t, 640
 - wdsSetEventReportReq, 1192
- pMNAAA
 - pack_wds_SetMobileIPProfile_t, 627
- pMNCIncPCSDigStat
 - _sysSelectPrefParams, 93
 - pack_nas_SLQSSetSysSelectionPref_t, 600
- pMNHA
 - pack_wds_SetMobileIPProfile_t, 627
- pMNRInfo

- nasInitNetworkReg, 508
- pack_nas_SLQSIInitiateNetworkRegistration_t, 585
- pMNC
 - QmiNas3GppNetworkInfo, 707
- pMTMessageInfo
 - SMSEventInfo_s, 812
- pMagneticDeviation
 - QmiCbkLocBestAvailPosInd, 689
 - QmiCbkLocPositionReportInd, 700
 - unpack_loc_BestAvailPos_Ind_t, 967
 - unpack_loc_PositionRpt_Ind_t, 976
- pManString
 - _SLQSSwiGetHostDevInfoParams, 79
 - _SLQSSwiSetHostDevInfoParams, 82
- pManagedRoamingInd
 - nasIndicationRegisterReq, 507
 - pack_nas_SLQSNasIndicationRegisterExt_t, 593
- pMaxAllowedPktSz
 - swiQosFlow, 845
- pMaxChannelRXRate
 - WdsConnectionRateElmnts, 1177
- pMaxChannelTXRate
 - WdsConnectionRateElmnts, 1177
- pMdmCallDurationActive
 - getDUNCallInfoResp, 252
- pMeidString
 - _SLQSSwiGetSerialNoExtParams, 81
- pMessage
 - cdmaMsgDecodingParams, 151
 - cdmaMsgEncodingParams, 153
 - pack_sms_SendSMS_t, 603
 - slqssendasyncsmsparams_s, 794
 - slqssendsmsparams_s, 796
 - wcdmaLongMsgDecodingParams, 1151
 - wcdmaMsgDecodingParams, 1152
- pMessageID
 - cdmaMsgDecodingParams, 151
- pMessageIndex
 - pack_sms_SLQSDDeleteSMS_t, 604
- pMessageMode
 - pack_sms_SLQSDDeleteSMS_t, 604
 - pack_sms_SLQSGetSMS_t, 605
 - pack_sms_SLQSGetSMSList_t, 605
 - pack_sms_SLQSMModifySMSStatus_t, 606
 - smsMaxStorageSizeReq, 813
- pMessageModelInfo
 - SMSEventInfo_s, 812
- pMessageSize
 - cdmaMsgEncodingParams, 153
- pMessageTag
 - pack_sms_SLQSDDeleteSMS_t, 604
- pMicMute
 - SetM2MAudioProfileReq, 768
- pMinBasedIMSI
 - nasGet3GPP2SubscriptionInfoResp, 497
- pMinIntervalTime
 - LOCStartReq, 376
 - pack_loc_Start_t, 582
- pMinPolicedPktSz
 - swiQosFlow, 845
- pMinSessionExpiryTimer
 - GetIMSVoIPConfigResp, 259
 - imsVoIPConfigInfo, 314
 - SetIMSVoIPConfigReq, 765
- pMitigationDevList
 - TmdGetMitigationDevListResp, 864
- pMitigationDevListLen
 - TmdGetMitigationDevListResp, 864
- pMncPcsDigitStatus
 - nasInitNetworkReg, 508
 - pack_nas_SLQSIInitiateNetworkRegistration_t, 585
- pMncPcsStatus
 - nasPLMNNameReq, 517
 - pack_nas_SLQSGetPLMNName_t, 584
- pMode
 - pack_wds_SetMobileIPParameters_t, 626
- pModePref
 - _sysSelectPrefInfo, 87
 - _sysSelectPrefParams, 93
 - pack_nas_SLQSSetSysSelectionPref_t, 600
 - unpack_nas_SLQSGetSysSelectionPref_t, 1003
- pModelString
 - _SLQSSwiGetHostDevInfoParams, 79
 - _SLQSSwiSetHostDevInfoParams, 82
- pMtu
 - qmiWdsRunTimeSettings, 713
- pNAMNameInfo
 - nasGet3GPP2SubscriptionInfoResp, 497
- pNAI
 - pack_wds_SetMobileIPProfile_t, 627
- pNITZInformation
 - nasOperatorNameResp, 511
- pNSEnable
 - GetAudioPathConfigResp, 243
 - SetAudioPathConfigReq, 756
- pNSSAudioCtrl
 - voiceInfoRec, 1127
- pNSSRelease
 - voiceInfoRec, 1127
- pNamID
 - voiceGetConfigReq, 1121
- pName
 - pack_wds_SetDefaultProfile_t, 623
- pNameString
 - _SLQSSwiGetOSInfoParams, 80
 - _SLQSSwiSetOSInfoParams, 83
- pNegoDnsSrvrPref
 - LibPackprofile_3GPP2, 344
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 674
- pNeighborSetPilotPN
 - NetworkStat1x, 536
- pNetSelPref
 - _sysSelectPrefInfo, 87
 - _sysSelectPrefParams, 93
 - pack_nas_SLQSSetSysSelectionPref_t, 600

- unpack_nas_SLQSGetSysSelectionPref_t, 1003
- pNetworkInfo
 - _slqsNetworkScanInfo, 71
- pNetworkInfoInstances
 - _slqsNetworkScanInfo, 71
- pNetworkInfoLen
 - CurrDataSysStat, 180
- pNetworkStat1x
 - NetworkDebugResp, 534
- pNetworkStatEVDO
 - NetworkDebugResp, 534
- pNetworkTimeInd
 - nasIndicationRegisterReq, 507
 - pack_nas_SLQSNasIndicationRegisterExt_t, 593
- pNewImsRegStatus
 - IMSARegistrationStatus, 302
- pNewPwdData
 - voiceSUPSInfo, 1145
- pNumSupUSBComps
 - USBCompParams, 1089
- pNumberOfPhySlot
 - UIMGetSlotsStatusResp, 898
 - unpack_uim_SLQSUIMGetSlotsStatus_t, 1058
- pNxtHdrProto
 - swiQosFilter, 842
- pOMADMEEnabled
 - _SLQSOMADMSettings, 76
- PORTNAM_LEN
 - qaGobiApiDcs.h, 1406
- pOTASPStatus
 - voiceCallInfoResp, 1099
 - voiceGetAllCallInfo, 1107
- pObjectVer
 - NetworkDebugResp, 534
- pOffLength
 - voiceDTMFEventInfo, 1104
- pOnLength
 - voiceDTMFEventInfo, 1104
- pOpaqueIdentifier
 - LocInjectSensorDataReq, 374
 - QmiCbkLocInjectSensorDataInd, 693
- pOperatorNameString
 - nasOperatorNameResp, 511
- pOperatorPLMNList
 - nasOperatorNameResp, 511
- pOptList
 - wds_DHCPv4OptionList, 1164
 - WdsDHCPv4ClientLeaseInd, 1178
 - WdsDHCPv4OptionList, 1182
 - wdsDhcpv4OptionList, 1183
- pOptVal
 - DHCPOption, 206
 - wds_DHCPOpt, 1162
- pOptions
 - DHCPOptionList, 207
- pPCMPParams
 - SetM2MAudioAVCFGReq, 766
- pPCSCFAddrPCO
 - qmiWdsRunTimeSettings, 713
- pPCSCFFQDNAddrList
 - qmiWdsRunTimeSettings, 713
- pPCSCFPort
 - GetRegMgrConfigParams, 267
- pPCSDigitInfo
 - _slqsNetworkScanInfo, 71
- pPCSDigitInstances
 - _slqsNetworkScanInfo, 71
- pPCSInstance
 - unpack_nas_PerformNetworkScan_t, 987
- pPCSInstanceSize
 - unpack_nas_PerformNetworkScan_t, 987
- pPDNInactivTimeout
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 326
 - Profile3GPP, 668
- pPDNInactivTimeout3GPP2
 - LibPackprofile_3GPP2, 344
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 674
- pPDPTYPE
 - qmiWdsRunTimeSettings, 713
- pPDPTYPE
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 327
 - Profile3GPP, 668
- pPDUMessage
 - wcdmaMsgEncodingParams, 1154
- pPLMNNetworkName
 - nasOperatorNameResp, 512
- pPRLPref
 - _sysSelectPrefInfo, 87
 - _sysSelectPrefParams, 93
 - pack_nas_SLQSSetSysSelectionPref_t, 600
 - unpack_nas_SLQSGetSysSelectionPref_t, 1003
- pPRLPreference
 - dmsCurrentPRLInfo, 210
- pPRLVersion
 - dmsCurrentPRLInfo, 210
- pPacketsCountRX
 - QosEventInfo, 716
 - slqsWdsEventInfo, 804
- pPacketsCountTX
 - QosEventInfo, 716
 - slqsWdsEventInfo, 804
- pPartNum
 - wcdmaLongMsgDecodingParams, 1151
- pPass
 - pack_wds_SLQSStartDataSession_t, 638
- pPassword
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 326
 - pack_wds_SetDefaultProfile_t, 623
 - Profile3GPP, 668
 - ssdatasession_params, 826
- pPasswordSize
 - LibPackprofile_3GPP, 338

- LibpackProfile3GPP, 326
- Profile3GPP, 668
- pPcscfAddrUsingDhcp
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 326
 - Profile3GPP, 668
- pPcscfAddrUsingPCO
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 326
 - Profile3GPP, 668
- pPdnType
 - LibPackprofile_3GPP2, 344
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 674
- pPdpAccessConFlag
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 326
 - Profile3GPP, 668
- pPdpContext
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 326
 - Profile3GPP, 668
- pPdpDataCompType
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 326
 - Profile3GPP, 668
- pPdpHdrCompType
 - LibPackprofile_3GPP, 338
 - LibpackProfile3GPP, 327
 - Profile3GPP, 668
- pPdpStatusConfig
 - IMSAIndRegisterInfo, 300
- pPersonalityListLength
 - HDRPersonalityInd, 287
 - HDRPersonalityResp, 287
 - HDRProtSubtypResp, 288
- pPersonalizationStatus
 - UIMGetConfigurationResp, 896
- pPhoneCtxtURLen
 - GetIMSSMSConfigParams, 256
 - SetIMSSMSConfigReq, 761
- pPhoneCtxtURI
 - GetIMSSMSConfigParams, 256
 - imsSMSConfigInfo, 311
 - SetIMSSMSConfigReq, 761
- pPhyCaAggPcellInfo
 - nasGetLTECphyCaResp, 499
- pPhyCaAggScellIDIBw
 - nasGetLTECphyCaResp, 499
- pPhyCaAggScellIndType
 - nasGetLTECphyCaResp, 499
- pPhyCaAggScellIndex
 - nasGetLTECphyCaResp, 499
- pPhyCaAggScellInfo
 - nasGetLTECphyCaResp, 499
- pPilotSetData
 - GetHRPDStatsResp, 255
- pPilotSetInfo
 - PilotSetData, 656
- pPkgDescLength
 - _SLQSOMADMSessionInfo, 73
- pPkgDescription
 - _SLQSOMADMSessionInfo, 73
- pPkgName
 - _SLQSOMADMSessionInfo, 73
- pPkgNameLength
 - _SLQSOMADMSessionInfo, 73
- pPktErrRate
 - swiQosFlow, 846
- pPlasmaIDString
 - _SLQSSwiGetHostDevInfoParams, 79
 - _SLQSSwiSetHostDevInfoParams, 82
- pPositionSource
 - PDSPositionData, 646
- pPositionSrc
 - LocInjectPositionReq, 372
- pPppSessCloseTimer1x
 - LibPackprofile_3GPP2, 344
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 674
- pPppSessCloseTimerDO
 - LibPackprofile_3GPP2, 344
 - LibpackProfile3GPP2, 332
 - Profile3GPP2, 674
- pPrDNSIPv4Address
 - swiPDPRuntimeSettingsResp, 840
- pPrDNSIPv6Address
 - swiPDPRuntimeSettingsResp, 840
- pPrPCSCFIPv4Address
 - swiPDPRuntimeSettingsResp, 840
- pPrPCSCFIPv6Address
 - swiPDPRuntimeSettingsResp, 840
- pPrecedence
 - swiQosFilter, 842
- pPrecisionDilution
 - QmiCbkLocBestAvailPosInd, 689
 - QmiCbkLocPositionReportInd, 700
 - unpack_loc_BestAvailPos_Ind_t, 967
 - unpack_loc_PositionRpt_Ind_t, 976
- pPrefNetwork
 - CurrDataSysStat, 180
- pPrefVoiceDomain
 - voiceSetConfigReq, 1136
- pPrefVoicePrivacy
 - voiceGetConfigReq, 1121
- pPrefVoiceSOStatus
 - voiceSetConfigResp, 1138
- pPrefVoiceSO
 - voiceGetConfigReq, 1121
 - voiceSetConfigReq, 1136
- pPreferred
 - QmiNas3GppNetworkInfo, 707
- pPriCSCFPort
 - imsRegMgrConfigInfo, 309
 - SetRegMgrConfigReq, 772
- pPriCSCFPortName

- GetRegMgrConfigParams, [267](#)
- pPriCSCFPortNameLen
 - GetRegMgrConfigParams, [267](#)
- pPriDNSIPv4AddPref
 - LibPackprofile_3GPP, [338](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [668](#)
- pPriDNSIPv6addpref
 - LibPackprofile_3GPP, [338](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [668](#)
- pPriV6DnsAddress
 - LibPackprofile_3GPP2, [344](#)
 - LibpackProfile3GPP2, [333](#)
 - Profile3GPP2, [674](#)
- pPrimaryDNSV4
 - qmiWdsRunTimeSettings, [713](#)
- pPrimaryDNSV6
 - qmiWdsRunTimeSettings, [714](#)
- pPrimaryHA
 - pack_wds_SetMobileIPProfile_t, [627](#)
- pPrimaryID
 - LibPackprofile_3GPP, [338](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [668](#)
- pPrimaryV4DnsAddress
 - LibPackprofile_3GPP2, [344](#)
 - LibpackProfile3GPP2, [332](#)
 - Profile3GPP2, [674](#)
- pPriority
 - cdmaMsgDecodingParams, [151](#)
 - cdmaMsgEncodingParams, [153](#)
- pPrivacy
 - cdmaMsgDecodingParams, [151](#)
- pProfSz
 - NWProfile, [542](#)
- pProfValues
 - NWProfile, [542](#)
- pProfileID
 - CreateProfileIn, [176](#)
 - ModifyProfileIn, [403](#)
 - qmiWdsRunTimeSettings, [714](#)
 - unpack_wds_SLQSCreateProfile_t, [1071](#)
- pProfileId
 - pack_wds_SLQSCreateProfile_t, [628](#)
 - pack_wds_SLQSModifyProfile_t, [632](#)
 - pack_wds_SLQSSetDHCPv4ClientConfig_t, [635](#)
 - pack_wds_SLQSSetDHCPv4ClientConfig_t, [636](#)
 - WdsDHCPv4ClientLeaseInd, [1178](#)
 - WdsDHCPv4Config, [1179](#)
- pProfileId3GPP2
 - ssdatasession_params, [826](#)
 - swiQosFlow, [846](#)
- pProfileId3GPP
 - ssdatasession_params, [826](#)
- pProfileIndex
 - CreateProfileOut, [177](#)
- pProfileList
 - _slqs3GPPConfigItem, [69](#)
 - pack_wds_SLQSSet3GPPConfigItem_t, [634](#)
- pProfileName
 - qmiWdsRunTimeSettings, [714](#)
- pProfileSettings
 - unpack_wds_SLQSGetProfileSettings_t, [1077](#)
- pProfileType
 - CreateProfileIn, [176](#)
 - CreateProfileOut, [177](#)
 - ModifyProfileIn, [403](#)
 - pack_wds_SLQSCreateProfile_t, [628](#)
 - pack_wds_SLQSModifyProfile_t, [632](#)
- pProfilename
 - LibPackprofile_3GPP, [338](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [668](#)
- pProfilenameSize
 - LibPackprofile_3GPP, [338](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [668](#)
- pProtoSubTypElmnt
 - HDRProtSubtypResp, [288](#)
- pProtocolSubtypeElement
 - HDRPersonalityInd, [287](#)
 - HDRPersonalityResp, [287](#)
- pQFlowState
 - QosFlowInfo, [717](#)
- pQmiInterfaceInfo
 - _packetSrvStatus, [64](#)
 - slqsSessionStateInfo, [797](#)
 - slqsWdsEventInfo, [805](#)
- pQosClassID
 - LibPackprofile_3GPP, [338](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [668](#)
- pRATInstance
 - unpack_nas_PerformNetworkScan_t, [987](#)
- pRATInfo
 - _slqsNetworkScanInfo, [71](#)
- pRATInstanceSize
 - unpack_nas_PerformNetworkScan_t, [987](#)
- pRATInstances
 - _slqsNetworkScanInfo, [71](#)
- pRATStatus
 - imsaRatStatusInfo, [301](#)
- pRATType
 - LibPackprofile_3GPP2, [344](#)
 - LibpackProfile3GPP2, [333](#)
 - Profile3GPP2, [674](#)
- pRAT
 - _sysSelectPrefParams, [93](#)
 - pack_nas_SLQSSetSysSelectionPref_t, [600](#)
- PREFERRED_INDEX
 - qaNasPerformNetworkScan.h, [1741](#)
- pRFBandInfoElements
 - QmiNasGetRFBandInfoResp, [708](#)
- PRI_UPDATE_FAIL
 - qaGobiApiFms.h, [1459](#)

- PRISring
 - unpack_dms_GetFirmwareRevision_t, 932
 - unpack_dms_GetFirmwareRevisions_t, 932
- PRLInd
 - qaQmiServingSystemParam, 683
 - unpack_nas_SLQSGetServingSystem_t, 996
- PRLPTlv
 - NASQmiCbkNasSystemSelPrefInd, 521
- PRLPref
 - NASPRPreferenceTlv, 520
- pRMAutoConnect
 - custFeaturesInfo, 189
- pRSRPThresList
 - RSRPThresh, 730
- pRSRPThresh
 - sigInfo, 782
- pRSRQThresList
 - RSRQThresh, 731
- pRSRQThresh
 - sigInfo, 782
- pRSSIThresList
 - RSSIThresh, 732
- pRSSIThresh
 - sigInfo, 782
- pRTPRTCPInactTimer
 - GetIMSVoIPConfigResp, 260
 - imsVoIPConfigInfo, 314
 - SetIMSVoIPConfigReq, 765
- pRXAGCList
 - GetAudioPathConfigResp, 243
 - SetAudioPathConfigReq, 756
- pRXAIG
 - RXAGCList, 733
- pRXAVCAGCSwitch
 - GetAudioPathConfigResp, 243
 - SetAudioPathConfigReq, 756
- pRXAVCList
 - GetAudioPathConfigResp, 243
 - SetAudioPathConfigReq, 756
- pRXChain0Info
 - nasGetTxRxInfoResp, 504
- pRXChain1Info
 - nasGetTxRxInfoResp, 504
- pRXComprSlope
 - RXAGCList, 733
- pRXComprThres
 - RXAGCList, 733
- pRXDroppedCount
 - unpack_wds_GetPacketStatistics_t, 1068
 - WdsPktStatisticsElmnts, 1187
- pRXExpSlope
 - RXAGCList, 733
- pRXExpThres
 - RXAGCList, 733
- pRXOKBytesCount
 - getDUNCallInfoResp, 252
- pRXOKBytesLastCall
 - unpack_wds_GetPacketStatistics_t, 1068
- WdsPktStatisticsElmnts, 1187
- pRXOKBytesCount
 - unpack_wds_GetPacketStatistics_t, 1068
 - WdsPktStatisticsElmnts, 1187
- pRXPCMIIRFtr
 - GetAudioPathConfigResp, 243
 - SetAudioPathConfigReq, 756
- pRXPacketErrors
 - unpack_wds_GetPacketStatistics_t, 1068
 - WdsPktStatisticsElmnts, 1187
- pRXPacketOverflows
 - unpack_wds_GetPacketStatistics_t, 1068
 - WdsPktStatisticsElmnts, 1187
- pRXPacketSuccesses
 - unpack_wds_GetPacketStatistics_t, 1068
 - WdsPktStatisticsElmnts, 1187
- pRXStaticGain
 - RXAGCList, 733
- pRXTotalBytes
 - unpack_wds_GetByteTotals_t, 1061
 - WdsByteTotalsElmnts, 1175
- pRadioInterface
 - nasNetworkTime, 510
 - unpack_nas_SLQSNasNetworkTimeCallBack_↔
ind_t, 1006
- pRankIndicatorInd
 - NasSwIndReg, 526
 - pack_nas_SLQSNasSwIndicationRegister_t, 594
- pRatHandoverStatusConfig
 - IMSAIndRegisterInfo, 300
- pRawHorConfidence
 - LocInjectPositionReq, 372
- pRawHorUncCircular
 - LocInjectPositionReq, 372
- pReRegPeriod
 - pack_wds_SetMobileIPParameters_t, 626
- pReRegTraffic
 - pack_wds_SetMobileIPParameters_t, 626
- pReadAcknowledgementReq
 - cdmaMsgDecodingParams, 151
- pReadResult
 - UIMReadTransparentResp, 902
 - unpack_uim_ReadTransparent_t, 1055
- pReason
 - voiceSUPSInfo, 1145
- pReasonMask
 - unpack_dms_GetOfflineReason_t, 936
- pRecurrenceType
 - LOCStartReq, 376
 - pack_loc_Start_t, 582
- pRedirNumInfo
 - voiceInfoRec, 1127
- pRefData
 - FirmwareUpdatStat, 231
- pRefString
 - FirmwareUpdatStat, 231
- pRefStringLen
 - FirmwareUpdatStat, 231

- pReferenceNum
 - wcdmaLongMsgDecodingParams, [1151](#)
- pRefreshEvent
 - UIMRefreshGetLastEventResp, [905](#)
- pRegCallStatInfoEvt
 - _getIndicationRegResp, [58](#)
 - _setIndicationRegReq, [67](#)
- pRegDTMFEvents
 - voiceIndicationRegisterInfo, [1124](#)
- pRegInd
 - _getTransLayerInfoResp, [61](#)
- pRegMgrConfigEvents
 - imsCfgIndRegisterInfo, [309](#)
- pRegStatus
 - _getTransNWRegInfoResp, [61](#)
- pRegStatusConfig
 - IMSASupportedFieldsResp, [306](#)
- pRegStatusErrorCode
 - imsaRegStatusInfo, [303](#)
- pRegTransLayerInfoEvt
 - _getIndicationRegResp, [58](#)
 - _setIndicationRegReq, [67](#)
- pRegTransNWRegInfoEvt
 - _getIndicationRegResp, [58](#)
 - _setIndicationRegReq, [67](#)
- pRegVoicePrivacyEvents
 - voiceIndicationRegisterInfo, [1124](#)
- pRejectReason
 - pack_swima_SLQSOMADMSSendSelection_↔
t, [610](#)
- pRelValidity
 - cdmaMsgEncodingParams, [153](#)
- pRelativeValidity
 - cdmaMsgDecodingParams, [151](#)
- pRemainingRetries
 - UIMDepersonalizationResp, [893](#)
 - UIMPinResp, [899](#)
 - unpack_uim_ChangePin_t, [1054](#)
 - unpack_uim_SetPinProtection_t, [1056](#)
 - unpack_uim_UnblockPin_t, [1059](#)
 - unpack_uim_VerifyPin_t, [1060](#)
- pRemotePartyName
 - voiceCallInfoResp, [1099](#)
- pRemotePartyNum
 - voiceCallInfoResp, [1099](#)
- pReportChannelRate
 - getDUNCallInfoReq, [249](#)
 - pack_wds_SLQSGetDUNCallInfo_t, [629](#)
- pReportConnStatus
 - getDUNCallInfoReq, [249](#)
 - pack_wds_SLQSGetDUNCallInfo_t, [629](#)
- pReportDataBearerTech
 - getDUNCallInfoReq, [249](#)
 - pack_wds_SLQSGetDUNCallInfo_t, [629](#)
- pReportDormStatus
 - getDUNCallInfoReq, [250](#)
 - pack_wds_SLQSGetDUNCallInfo_t, [629](#)
- pReqFieldsList
 - IMSASupportedFieldsResp, [306](#)
- pReqMitigationLvl
 - TmdGetMitigationLvlResp, [865](#)
- pReqSettings
 - pack_wds_SLQSGetRuntimeSettings_t, [631](#)
- pRequestOptionList
 - pack_wds_SLQSSetDHCPv4ClientConfig_t, [636](#)
 - unpack_wds_SLQSSetDHCPv4ClientConfig_t,
[1083](#)
 - WdsDHCPv4Config, [1179](#)
- pRequestedTag
 - pack_sms_SLQSGetSMSList_t, [605](#)
- pRespData
 - USSDRespFNetwork, [1090](#)
- pRespFieldsList
 - IMSASupportedFieldsResp, [306](#)
- pRetryCount
 - _SLQSOMADMSessionInfo, [74](#)
- pRetryInterval
 - pack_wds_SetMobileIPParameters_t, [626](#)
- pRetryLimit
 - pack_wds_SetMobileIPParameters_t, [626](#)
- pRetryMessage
 - slqssendasyncsmsparams_s, [794](#)
- pRetryMessageId
 - slqssendasyncsmsparams_s, [795](#)
- pRevInUse
 - CDMASysInfo, [158](#)
 - nas_CDMASysInfo, [414](#)
- pRevInUseValid
 - CDMASysInfo, [158](#)
 - nas_CDMASysInfo, [414](#)
- pRevTunneling
 - pack_wds_SetMobileIPProfile_t, [627](#)
- pRingBackTimer
 - GetIMSVoIPConfigResp, [260](#)
 - imsVoIPConfigInfo, [314](#)
 - SetIMSVoIPConfigReq, [765](#)
- pRingingTimer
 - GetIMSVoIPConfigResp, [260](#)
 - imsVoIPConfigInfo, [314](#)
 - SetIMSVoIPConfigReq, [765](#)
- pRoamPref
 - _sysSelectPrefInfo, [87](#)
 - _sysSelectPrefParams, [94](#)
 - pack_nas_SLQSSetSysSelectionPref_t, [600](#)
 - unpack_nas_SLQSGetSysSelectionPref_t, [1003](#)
- pRoamTimer
 - voiceGetConfigReq, [1121](#)
- pRoamTimerCnt
 - voiceGetConfigResp, [1123](#)
- pRoamTimerConfig
 - voiceSetConfigReq, [1136](#)
- pRoamTimerStatus
 - voiceSetConfigResp, [1138](#)
- pRoaming
 - QmiNas3GppNetworkInfo, [707](#)
- pRscp

- nasSigInfo, [524](#)
- unpack_nas_SLQSNasSigInfoCallback_ind_↔
t, [1007](#)
- pRxFilter
 - swiQosModifyReq, [847](#)
 - swiQosReq, [848](#)
- pRxFlow
 - swiQosGranted, [846](#)
 - swiQosModifyReq, [847](#)
 - swiQosReq, [848](#)
- pRxQFilter
 - QosFlowInfo, [717](#)
- pRxQFlowGranted
 - QosFlowInfo, [717](#)
- PSDetachReq, [677](#)
 - pDetachAction, [677](#)
- PSDomain
 - unpack_nas_GetServingNetwork_t, [986](#)
- pSIPConfigEvents
 - imsCfgIndRegisterInfo, [309](#)
- pSIPLocalPort
 - GetSIPConfigResp, [269](#)
 - imsSIPConfigInfo, [310](#)
 - SetSIPConfigReq, [779](#)
- pSMSAttemptedFlag
 - getDyingGaspStatistics, [253](#)
 - packgetDyingGaspStatistics, [642](#)
- pSMSCAddressInfo
 - SMSEventInfo_s, [812](#)
- pSMSConfigEvents
 - imsCfgIndRegisterInfo, [309](#)
- pSMSFormat
 - GetIMSSMSConfigParams, [256](#)
 - imsSMSConfigInfo, [311](#)
 - SetIMSSMSConfigReq, [761](#)
- pSMSONIMSInfo
 - SMSEventInfo_s, [812](#)
- pSMSOverIPNwInd
 - GetIMSSMSConfigParams, [256](#)
 - imsSMSConfigInfo, [311](#)
 - SetIMSSMSConfigReq, [761](#)
- pSMSSupport
 - custFeaturesInfo, [189](#)
- pSMSSvcRAT
 - imsaSvcStatusInfo, [307](#)
- pSMSSvcStatus
 - imsaSvcStatusInfo, [307](#)
- pSPC
 - pack_wds_SetMobileIPParameters_t, [626](#)
- pSVInfo
 - LocDelAssDataReq, [365](#)
 - pack_loc_Delete_Assist_Data_t, [567](#)
- pSWVerString
 - _SLQSSwiGetHostDevInfoParams, [79](#)
 - _SLQSSwiSetHostDevInfoParams, [82](#)
- pSatelliteInfo
 - gnssSvInfoNotification, [271](#)
 - unpack_loc_GnssSvInfo_Ind_t, [971](#)
- pScAddr
 - wcdmaLongMsgDecodingParams, [1151](#)
 - wcdmaMsgDecodingParams, [1152](#)
- pScAddrLength
 - wcdmaLongMsgDecodingParams, [1151](#)
 - wcdmaMsgDecodingParams, [1152](#)
- pScanResult
 - _slqsNetworkScanInfo, [71](#)
 - unpack_nas_PerformNetworkScan_t, [988](#)
- pSccRxInfo
 - LteSccRxInfoResp, [390](#)
 - unpack_nas_SLQSSwiGetLteSccRxInfo_t, [1011](#)
- pScrAmrEnable
 - GetIMSVoIPConfigResp, [260](#)
 - imsVoIPConfigInfo, [314](#)
 - SetIMSVoIPConfigReq, [765](#)
- pScrAmrWbEnable
 - GetIMSVoIPConfigResp, [260](#)
 - imsVoIPConfigInfo, [314](#)
 - SetIMSVoIPConfigReq, [765](#)
- pSeDNSIPv4Address
 - swiPDPRuntimeSettingsResp, [840](#)
- pSeDNSIPv6Address
 - swiPDPRuntimeSettingsResp, [840](#)
- pSePCSCFIPv4Address
 - swiPDPRuntimeSettingsResp, [840](#)
- pSePCSCFIPv6Address
 - swiPDPRuntimeSettingsResp, [840](#)
- pSecDNSIPv4AddPref
 - LibPackprofile_3GPP, [338](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [668](#)
- pSecDNSIPv6addpref
 - LibPackprofile_3GPP, [338](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [668](#)
- pSecV6DnsAddress
 - LibPackprofile_3GPP2, [344](#)
 - LibpackProfile3GPP2, [333](#)
 - Profile3GPP2, [674](#)
- pSecondaryDNSV4
 - qmiWdsRunTimeSettings, [714](#)
- pSecondaryDNSV6
 - qmiWdsRunTimeSettings, [714](#)
- pSecondaryFlag
 - LibPackprofile_3GPP, [338](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [668](#)
- pSecondaryHA
 - pack_wds_SetMobileIPProfile_t, [627](#)
- pSecondaryV4DnsAddress
 - LibPackprofile_3GPP2, [344](#)
 - LibpackProfile3GPP2, [333](#)
 - Profile3GPP2, [674](#)
- pSectorID
 - NetworkStatEVDO, [538](#)
- pSenderAddr
 - cdmaMsgDecodingParams, [151](#)

- wcdmaLongMsgDecodingParams, [1151](#)
- wcdmaMsgDecodingParams, [1152](#)
- pSenderAddrLength
 - cdmaMsgDecodingParams, [151](#)
 - wcdmaLongMsgDecodingParams, [1151](#)
 - wcdmaMsgDecodingParams, [1152](#)
- pSensorDataUsage
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [976](#)
- pServerAddrList
 - qmiWdsRunTimeSettings, [714](#)
- pServiceClass
 - voiceSetSUPSServiceReq, [1140](#)
- pServiceOption
 - slqssendasyncsmsparams_s, [795](#)
- pServiceStatusConfig
 - IMSASndRegisterInfo, [300](#)
- pServingSystemInd
 - nasIndicationRegisterReq, [507](#)
 - pack_nas_SLQSNasIndicationRegisterExt_t, [593](#)
- pSessionExpiryTimer
 - GetIMSVoIPConfigResp, [260](#)
 - imsVoIPConfigInfo, [315](#)
 - SetIMSVoIPConfigReq, [765](#)
- pSessionIDv4
 - GetSessionIDResp, [268](#)
- pSessionIDv6
 - GetSessionIDResp, [268](#)
- pSessionState
 - _SLQSOMADMSessionInfo, [74](#)
- pSessionType
 - _SLQSOMADMSessionInfo, [74](#)
- pSettingResp
 - GetIMSSMSConfigParams, [256](#)
 - GetIMSUserConfigParams, [257](#)
 - GetIMSVoIPConfigResp, [260](#)
 - GetRegMgrConfigParams, [267](#)
 - GetSIPConfigResp, [269](#)
 - SetIMSSMSConfigResp, [762](#)
 - SetIMSUserConfigResp, [763](#)
 - SetIMSVoIPConfigResp, [765](#)
 - SetRegMgrConfigResp, [772](#)
 - SetSIPConfigResp, [779](#)
- pSeverity
 - _SLQSOMADMSessionInfo, [74](#)
- pSigCompEnabled
 - GetSIPConfigResp, [269](#)
 - imsSIPConfigInfo, [310](#)
 - SetSIPConfigReq, [779](#)
- pSigIndReq
 - pack_nas_SLQSSetSignalStrengthsCallback_t, [595](#)
- pSignalInfo
 - voicelInfoRec, [1127](#)
- pSignalStrengthInd
 - nasIndicationRegisterReq, [507](#)
- pack_nas_SLQSNasIndicationRegisterExt_t, [593](#)
- pSmsOnlms
 - slqssendasyncsmsparams_s, [795](#)
 - slqssendsmsparams_s, [796](#)
- pSmsServiceRat
 - IMSAServiceStatus, [305](#)
- pSmsServiceStatus
 - IMSAServiceStatus, [305](#)
- pSource
 - _SLQSOMADMSessionInfo, [74](#)
- pSourceIP
 - LibPackTFTIDParams, [347](#)
 - TFTIDParams, [861](#)
- pSourceLength
 - _SLQSOMADMSessionInfo, [74](#)
- pSpeedHorizontal
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [976](#)
- pSpeedUnc
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [977](#)
- pSpeedVertical
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [977](#)
- pSpeedVerticalUnc
 - QmiCbkLocBestAvailPosInd, [689](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
- pSrcRAT
 - imsaRatStatusInfo, [301](#)
- pSrvDomainPref
 - _sysSelectPrefInfo, [88](#)
 - _sysSelectPrefParams, [94](#)
 - pack_nas_SLQSSetSysSelectionPref_t, [600](#)
 - unpack_nas_SLQSGetSysSelectionPref_t, [1003](#)
- pSrvOpt
 - voiceCallInfoResp, [1099](#)
- pSrvRegRestriction
 - _sysSelectPrefParams, [94](#)
 - pack_nas_SLQSSetSysSelectionPref_t, [600](#)
- pSrvcProviderName
 - nasOperatorNameResp, [512](#)
- pStage0Val
 - RXPCMIIRFiltr, [737](#)
 - TXPCMIIRFiltr, [874](#)
- pStage1Val
 - RXPCMIIRFiltr, [737](#)
 - TXPCMIIRFiltr, [874](#)
- pStage2Val
 - RXPCMIIRFiltr, [737](#)
 - TXPCMIIRFiltr, [874](#)
- pStage3Val
 - RXPCMIIRFiltr, [737](#)

- TXPCMIIRFiltr, [874](#)
- pStage4Val
 - RXPCMIIRFiltr, [737](#)
 - TXPCMIIRFiltr, [874](#)
- pStageCnt
 - RXPCMIIRFiltr, [737](#)
 - TXPCMIIRFiltr, [874](#)
- pStatMask
 - pack_wds_GetPacketStatistics_t, [621](#)
 - WdsPktStatisticsReq, [1188](#)
- pStatus
 - _SLQSOMADMSessionInfo, [74](#)
- pSubnetMaskV4
 - qmiWdsRunTimeSettings, [714](#)
- pSubscribeTimer
 - GetSIPConfigResp, [269](#)
 - imsSIPConfigInfo, [311](#)
 - SetSIPConfigReq, [779](#)
- pSubscriptionInfoInd
 - nasIndicationRegisterReq, [507](#)
 - pack_nas_SLQSNasIndicationRegisterExt_t, [593](#)
- pSupUSBComps
 - USBCompParams, [1089](#)
- pSupportedMsgList
 - IMASupportedMsgInfo, [306](#)
- pSuppsNotifEvents
 - voiceIndicationRegisterInfo, [1124](#)
- pSV
 - BdsSVInfo, [124](#)
 - loc_BdsSVInfo, [352](#)
 - loc_SVInfo, [363](#)
 - SVInfo, [830](#)
- pSvUsedforFix
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [977](#)
- pSvcClass
 - voiceGetCallBarringReq, [1108](#)
 - voiceGetCallBarringResp, [1109](#)
 - voiceGetCallFWReq, [1110](#)
 - voiceGetCallWaitInfo, [1113](#)
 - voiceSUPSInfo, [1145](#)
- pSvcType
 - voiceCallRequestParams, [1101](#)
- pSwiGetResetInd
 - dmsIndicationRegisterReq, [213](#)
- pSysInfoInd
 - nasIndicationRegisterReq, [507](#)
 - pack_nas_SLQSNasIndicationRegisterExt_t, [593](#)
- pSysInfoNoChange
 - nasSysInfo, [530](#)
 - unpack_nas_SLQSSysInfoCallback_ind_t, [1014](#)
- pSystemSelectionInd
 - nasIndicationRegisterReq, [507](#)
 - pack_nas_SLQSNasIndicationRegisterExt_t, [593](#)
- pTCPDStPort
 - swiQosFilter, [842](#)
- pTCPSrcPort
 - swiQosFilter, [842](#)
- pTDSCDMAECIODelta
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pTDSCDMAECIOTresh
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pTDSCDMAECIOTreshList
 - nas_TDSCDMAECIOTresh, [478](#)
 - TDSCDMAECIOTresh, [853](#)
- pTDSCDMARSCPDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pTDSCDMARSCPTresh
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pTDSCDMARSCPTreshList
 - nas_TDSCDMARSCPTresh, [479](#)
 - TDSCDMARSCPTresh, [853](#)
- pTDSCDMARSSIDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pTDSCDMARSSITresh
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pTDSCDMARSSITreshList
 - nas_TDSCDMARSSITresh, [479](#)
 - TDSCDMARSSITresh, [854](#)
- pTDSCDMASINRCONFTresh
 - sigInfo, [782](#)
- pTDSCDMASINRCONFTreshList
 - TDSCDMASINRCONFTresh, [856](#)
- pTDSCDMASINRDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pTDSCDMASINRThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, [590](#)
 - setSignalStrengthInfo, [777](#)
- pTDSCDMASINRThreshList
 - nas_TDSCDMASINRThresh, [480](#)
 - TDSCDMASINRThresh, [857](#)
- pTDSCDMASigInfoExt
 - nasGetSigInfoResp, [500](#)
 - nasSigInfo, [524](#)
 - unpack_nas_SLQSNasSigInfoCallback_ind_←
t, [1007](#)
- pTDSCDMASigInfoRscp
 - nasGetSigInfoResp, [500](#)
- pTFTID1Params
 - LibPackprofile_3GPP, [339](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [669](#)
- pTFTID2Params
 - LibPackprofile_3GPP, [339](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [669](#)
- pTTYConfigStatus

- voiceSetConfigResp, [1138](#)
- pTTYMode
 - voiceGetConfigReq, [1121](#)
 - voiceSetConfigReq, [1136](#)
- pTXAGCList
 - GetAudioPathConfigResp, [243](#)
 - SetAudioPathConfigReq, [756](#)
- pTXAIG
 - TXAGCList, [872](#)
- pTXAVCSwitch
 - GetAudioPathConfigResp, [243](#)
 - SetAudioPathConfigReq, [756](#)
- pTXComprSlope
 - TXAGCList, [872](#)
- pTXComprThres
 - TXAGCList, [872](#)
- pTXDroppedCount
 - unpack_wds_GetPacketStatistics_t, [1068](#)
 - WdsPktStatisticsElmnts, [1187](#)
- pTXExpSlope
 - TXAGCList, [872](#)
- pTXExpThres
 - TXAGCList, [872](#)
- pTXGain
 - GetAudioPathConfigResp, [243](#)
 - SetAudioPathConfigReq, [756](#)
- pTXInfo
 - nasGetTxRxInfoResp, [504](#)
- pTXOKBytesCount
 - getDUNCallInfoResp, [252](#)
- pTXOKBytesLastCall
 - unpack_wds_GetPacketStatistics_t, [1068](#)
 - WdsPktStatisticsElmnts, [1187](#)
- pTXOkBytesCount
 - unpack_wds_GetPacketStatistics_t, [1068](#)
 - WdsPktStatisticsElmnts, [1187](#)
- pTXPCMIIRFtr
 - GetAudioPathConfigResp, [243](#)
 - SetAudioPathConfigReq, [756](#)
- pTXPacketErrors
 - unpack_wds_GetPacketStatistics_t, [1068](#)
 - WdsPktStatisticsElmnts, [1187](#)
- pTXPacketOverflows
 - unpack_wds_GetPacketStatistics_t, [1068](#)
 - WdsPktStatisticsElmnts, [1187](#)
- pTXPacketSuccesses
 - unpack_wds_GetPacketStatistics_t, [1068](#)
 - WdsPktStatisticsElmnts, [1187](#)
- pTXStaticGain
 - TXAGCList, [872](#)
- pTXTotalBytes
 - unpack_wds_GetByteTotals_t, [1061](#)
 - WdsByteTotalsElmnts, [1175](#)
- pTdsBandCapability
 - BandCapabilityResp, [123](#)
- pTdsdmaBandPref
 - _sysSelectPrefParams, [94](#)
 - pack_nas_SLQSSetSysSelectionPref_t, [600](#)
- pTech
 - pack_wds_SLQSStartDataSession_t, [638](#)
- pTechnology
 - qmiWdsRunTimeSettings, [714](#)
 - ssdatasession_params, [827](#)
- pTechnologyMask
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [977](#)
- pTextMsg
 - cdmaMsgDecodingParams, [151](#)
 - cdmaMsgEncodingParams, [153](#)
 - wcdmaLongMsgDecodingParams, [1151](#)
 - wcdmaMsgDecodingParams, [1152](#)
 - wcdmaMsgEncodingParams, [1154](#)
- pTextMsgLength
 - cdmaMsgDecodingParams, [151](#)
 - wcdmaLongMsgDecodingParams, [1151](#)
 - wcdmaMsgDecodingParams, [1153](#)
- pTgtRAT
 - imsaRatStatusInfo, [301](#)
- pTime
 - _SLQSOMADMSessionInfo, [74](#)
 - SwiOTAMsg_s, [837](#)
- pTimeLength
 - _SLQSOMADMSessionInfo, [74](#)
- pTimeSrc
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [977](#)
- pTimeStamp
 - getDyingGaspStatistics, [253](#)
 - PDSPositionData, [646](#)
 - packgetDyingGaspStatistics, [642](#)
- pTimeType
 - PDSPositionData, [646](#)
- pTimeUnc
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [968](#)
 - unpack_loc_PositionRpt_Ind_t, [977](#)
- pTimeZone
 - nasNetworkTime, [511](#)
 - unpack_nas_SLQSNasNetworkTimeCallBack_↔
ind_t, [1006](#)
- pTimer
 - NasSwiIndReg, [526](#)
 - pack_nas_SLQSNasSwiIndicationRegister_t, [594](#)
- pTimerSIPReg
 - GetSIPConfigResp, [269](#)
 - imsSIPConfigInfo, [311](#)
 - SetSIPConfigReq, [779](#)
- pTimerT1
 - GetSIPConfigResp, [269](#)
 - imsSIPConfigInfo, [311](#)
 - SetSIPConfigReq, [779](#)

- pTimerT2
 - GetSIPConfigResp, [269](#)
 - imsSIPConfigInfo, [311](#)
 - SetSIPConfigReq, [779](#)
- pTimerTf
 - GetSIPConfigResp, [269](#)
 - imsSIPConfigInfo, [311](#)
 - SetSIPConfigReq, [779](#)
- pTimerVal
 - voiceSetSUPSServiceReq, [1140](#)
- pTimestampAge
 - LocInjectPositionReq, [372](#)
- pTimestampUtc
 - LocInjectPositionReq, [372](#)
 - QmiCbkLocBestAvailPosInd, [689](#)
 - QmiCbkLocPositionReportInd, [700](#)
 - unpack_loc_BestAvailPos_Ind_t, [967](#)
 - unpack_loc_PositionRpt_Ind_t, [977](#)
- pTokenBucket
 - swiQosFlow, [846](#)
- pTos
 - swiQosFilter, [842](#)
- pTotalBytesRX
 - QosEventInfo, [716](#)
 - slqsWdsEventInfo, [805](#)
- pTotalBytesTX
 - QosEventInfo, [716](#)
 - slqsWdsEventInfo, [805](#)
- pTotalNum
 - wcdmaLongMsgDecodingParams, [1151](#)
- pTrafficClass
 - swiQosFlow, [846](#)
- pTranDstPort
 - swiQosFilter, [842](#)
- pTranSrcPort
 - swiQosFilter, [842](#)
- pTransLayerInfo
 - _getTransLayerInfoResp, [61](#)
 - _transLayerInfoNotification, [95](#)
- pTransferRouteMTMessageInfo
 - SMSEventInfo_s, [813](#)
- pTransferStatInd
 - getDUNCallInfoReq, [250](#)
 - pack_wds_SLQSGetDUNCallInfo_t, [629](#)
 - pack_wds_SLQSWdsSetEventReport_t, [640](#)
 - wdsSetEventReportReq, [1192](#)
- pTransferStatusReport
 - smsSetRoutesReq, [820](#)
- pTrueIMSI
 - nasGet3GPP2SubscriptionInfoResp, [497](#)
- pTxFilter
 - swiQosModifyReq, [847](#)
 - swiQosReq, [848](#)
- pTxFlow
 - swiQosGranted, [846](#)
 - swiQosModifyReq, [847](#)
 - swiQosReq, [848](#)
- pTxQFilter
 - QosFlowInfo, [717](#)
- pTxQFlowGranted
 - QosFlowInfo, [717](#)
- pTypeCode
 - USSDRespFNetwork, [1090](#)
- pUATI
 - GetHRPDStatsResp, [255](#)
- pUDPDstPort
 - swiQosFilter, [842](#)
- pUDPSrcPort
 - swiQosFilter, [842](#)
- pUMTSCellID
 - nasCellLocationInfoResp, [495](#)
 - unpack_nas_SLQSNasGetCellLocationInfo_↔
t, [1005](#)
- pUMTSGrantedQoS
 - qmiWdsRunTimeSettings, [714](#)
- pUMTSInfo
 - nasCellLocationInfoResp, [495](#)
 - unpack_nas_SLQSNasGetCellLocationInfo_↔
t, [1005](#)
- pUMTSMinQoS
 - LibPackprofile_3GPP, [339](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [669](#)
- pUMTSMinQoSSigInd
 - LibPackprofile_3GPP, [339](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [669](#)
- pUMTSReqQoSSigInd
 - LibPackprofile_3GPP, [339](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [669](#)
- pUMTSReqQoS
 - LibPackprofile_3GPP, [339](#)
 - LibpackProfile3GPP, [327](#)
 - Profile3GPP, [669](#)
- pUSBComp
 - USBCompConfig, [1087](#)
 - USBCompParams, [1089](#)
- pUSSDData
 - USSDNoWaitIndicationInfo, [1090](#)
- pUSSDInfo
 - USSResp, [1092](#)
- pUSSInfo
 - voiceSUPSInfo, [1145](#)
- pUTSvcRAT
 - imsaSvcStatusInfo, [307](#)
- pUTSvcStatus
 - imsaSvcStatusInfo, [307](#)
- pUUSInfo
 - voiceCallRequestParams, [1101](#)
- pUUSInfo
 - voiceCallInfoResp, [1099](#)
- pUimSlotsStatus
 - UIMGetSlotsStatusResp, [898](#)
 - unpack_uim_SLQSUIMGetSlotsStatus_t, [1058](#)
- pUpdateCompleteStatus

- [_SLQSOMADMSessionInfo, 74](#)
- [pUser](#)
 - [pack_wds_SLQSStartDataSession_t, 638](#)
- [pUserAcknowledgementReq](#)
 - [cdmaMsgDecodingParams, 151](#)
- [pUserConfigEvents](#)
 - [imsCfgIndRegisterInfo, 309](#)
- [pUserData](#)
 - [slqssendasyncsmsparams_s, 795](#)
- [pUserId](#)
 - [LibPackprofile_3GPP2, 344](#)
 - [LibpackProfile3GPP2, 333](#)
 - [Profile3GPP2, 674](#)
- [pUserIdSize](#)
 - [LibPackprofile_3GPP2, 344](#)
 - [LibpackProfile3GPP2, 333](#)
 - [Profile3GPP2, 674](#)
- [pUsername](#)
 - [LibPackprofile_3GPP, 339](#)
 - [LibpackProfile3GPP, 327](#)
 - [pack_wds_SetDefaultProfile_t, 623](#)
 - [Profile3GPP, 669](#)
 - [qmiWdsRunTimeSettings, 714](#)
 - [ssdatasession_params, 827](#)
- [pUsernameSize](#)
 - [LibPackprofile_3GPP, 339](#)
 - [LibpackProfile3GPP, 327](#)
 - [Profile3GPP, 669](#)
- [pUtServiceRat](#)
 - [IMSAServiceStatus, 305](#)
- [pUtServiceStatus](#)
 - [IMSAServiceStatus, 305](#)
- [pV4sessionId](#)
 - [WdsByteTotals, 1175](#)
 - [WdsConnectionRate, 1177](#)
 - [WdsPktStatisticsResp, 1189](#)
- [pV6sessionId](#)
 - [WdsByteTotals, 1175](#)
 - [WdsConnectionRate, 1177](#)
 - [WdsPktStatisticsResp, 1189](#)
- [pVOIPSvcRAT](#)
 - [imsaSvcStatusInfo, 307](#)
- [pVOIPSvcStatus](#)
 - [imsaSvcStatusInfo, 307](#)
- [pVTSvcRAT](#)
 - [imsaSvcStatusInfo, 307](#)
- [pVTSvcStatus](#)
 - [imsaSvcStatusInfo, 307](#)
- [pVerboseFailReasonType](#)
 - [unpack_wds_SLQSStartDataSession_t, 1084](#)
- [pVerboseFailureReason](#)
 - [unpack_wds_SLQSStartDataSession_t, 1084](#)
- [pVersionString](#)
 - [_SLQSSwiGetOSInfoParams, 80](#)
 - [_SLQSSwiSetOSInfoParams, 83](#)
- [pVertConfidence](#)
 - [LocInjectPositionReq, 372](#)
 - [QmiCbkLocBestAvailPosInd, 689](#)
- [QmiCbkLocPositionReportInd, 701](#)
- [unpack_loc_BestAvailPos_Ind_t, 968](#)
- [unpack_loc_PositionRpt_Ind_t, 977](#)
- [pVertReliability](#)
 - [LocInjectPositionReq, 372](#)
 - [QmiCbkLocBestAvailPosInd, 689](#)
 - [QmiCbkLocPositionReportInd, 701](#)
 - [unpack_loc_BestAvailPos_Ind_t, 968](#)
 - [unpack_loc_PositionRpt_Ind_t, 977](#)
- [pVertUnc](#)
 - [LocInjectPositionReq, 372](#)
 - [QmiCbkLocBestAvailPosInd, 689](#)
 - [QmiCbkLocPositionReportInd, 701](#)
 - [unpack_loc_BestAvailPos_Ind_t, 968](#)
 - [unpack_loc_PositionRpt_Ind_t, 977](#)
- [pVerticalConfidence](#)
 - [PDSPositionData, 646](#)
- [pVerticalUnc](#)
 - [PDSPositionData, 646](#)
- [pVoIPConfigEvents](#)
 - [imsCfgIndRegisterInfo, 309](#)
- [pVoiceDomainPref](#)
 - [voiceGetConfigReq, 1122](#)
- [pVoiceDomainPrefStatus](#)
 - [voiceSetConfigResp, 1138](#)
- [pVoicePrivacy](#)
 - [voiceCallInfoResp, 1099](#)
 - [voiceGetAllCallInfo, 1107](#)
- [pVoipServiceRat](#)
 - [IMSAServiceStatus, 305](#)
- [pVoipServiceStatus](#)
 - [IMSAServiceStatus, 305](#)
- [pVolume](#)
 - [SetM2MAudioProfileReq, 768](#)
- [pVsServiceRat](#)
 - [IMSAServiceStatus, 305](#)
- [pVsServiceStatus](#)
 - [IMSAServiceStatus, 305](#)
- [pVtServiceRat](#)
 - [IMSAServiceStatus, 305](#)
- [pVtServiceStatus](#)
 - [IMSAServiceStatus, 305](#)
- [pWCMDABER](#)
 - [GetErrRateResp, 254](#)
- [pWCDMACallBarringSysInfo](#)
 - [nasGetSysInfoResp, 503](#)
 - [nasSysInfo, 530](#)
 - [unpack_nas_SLQSGetSysInfo_t, 1000](#)
 - [unpack_nas_SLQSSysInfoCallback_ind_t, 1014](#)
- [pWCDMACipherDomainSysInfo](#)
 - [nasGetSysInfoResp, 503](#)
 - [nasSysInfo, 530](#)
 - [unpack_nas_SLQSGetSysInfo_t, 1000](#)
 - [unpack_nas_SLQSSysInfoCallback_ind_t, 1014](#)
- [pWCDMAECIODelta](#)
 - [pack_nas_SLQSNasConfigSigInfo2_t, 590](#)
 - [setSignalStrengthInfo, 778](#)
- [pWCDMAECIOThresh](#)

- pack_nas_SLQSNasConfigSigInfo2_t, 590
- setSignalStrengthInfo, 778
- pWCDMAECIOThreshList
 - nas_WCDMAECIOThresh, 489
 - WCDMAECIOThresh, 1148
- pWCDMAInfoLTENeighborCell
 - nasCellLocationInfoResp, 495
 - unpack_nas_SLQSNasGetCellLocationInfo_↔
t, 1005
- pWCDMARSSIDelta
 - pack_nas_SLQSNasConfigSigInfo2_t, 590
 - setSignalStrengthInfo, 778
- pWCDMARSSIThresh
 - pack_nas_SLQSNasConfigSigInfo2_t, 590
 - setSignalStrengthInfo, 778
- pWCDMARSSIThreshList
 - nas_WCDMARSSIThresh, 490
 - WCDMARSSIThresh, 1154
- pWCDMASSInfo
 - nasGetSigInfoResp, 500
- pWCDMASigInfo
 - nasSigInfo, 524
 - unpack_nas_SLQSNasSigInfoCallback_ind_↔
t, 1007
- pWCDMASrvStatusInfo
 - nasGetSysInfoResp, 503
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 1000
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1014
- pWCDMASysInfo
 - nasGetSysInfoResp, 503
 - nasSysInfo, 530
 - unpack_nas_SLQSGetSysInfo_t, 1000
 - unpack_nas_SLQSSysInfoCallback_ind_t, 1014
- pWcdmaUARFCN
 - nasSwiGetChannelLockResp, 525
 - nasSwiSetChannelLockReq, 527
- pWifiState
 - PDSPosMethodStateReq, 647
- pXid
 - QmiCbkLocBestAvailPosInd, 689
 - unpack_loc_BestAvailPos_Ind_t, 968
- pXtraDataState
 - PDSPosMethodStateReq, 647
- pXtraTimeState
 - PDSPosMethodStateReq, 647
- pack_dms_ActivateAutomatic
 - dms.h, 1208
- pack_dms_ActivateAutomatic_t, 551
 - actCode, 552
- pack_dms_GetActivationState
 - dms.h, 1208
- pack_dms_GetBandCapability
 - dms.h, 1209
- pack_dms_GetCrashAction
 - dms.h, 1209
- pack_dms_GetCustFeature
 - dms.h, 1209
- pack_dms_GetCustFeaturesV2
 - dms.h, 1210
- pack_dms_GetCustFeaturesV2_t, 552
 - cust_id, 552
 - list_type, 552
 - Tlvresult, 552
- pack_dms_GetDeviceCap
 - dms.h, 1210
- pack_dms_GetDeviceCapabilities
 - dms.h, 1210
- pack_dms_GetDeviceHardwareRev
 - dms.h, 1211
- pack_dms_GetDeviceMfr
 - dms.h, 1211
- pack_dms_GetDeviceSerialNumbers
 - dms.h, 1212
- pack_dms_GetFSN
 - dms.h, 1213
- pack_dms_GetFirmwareInfo
 - dms.h, 1212
- pack_dms_GetFirmwareRevision
 - dms.h, 1212
- pack_dms_GetFirmwareRevisions
 - dms.h, 1213
- pack_dms_GetHardwareRevision
 - dms.h, 1214
- pack_dms_GetIMSI
 - dms.h, 1214
- pack_dms_GetManufacturer
 - dms.h, 1214
- pack_dms_GetModelID
 - dms.h, 1215
- pack_dms_GetNetworkTime
 - dms.h, 1215
- pack_dms_GetOfflineReason
 - dms.h, 1216
- pack_dms_GetPRLVersion
 - dms.h, 1216
- pack_dms_GetPower
 - dms.h, 1216
- pack_dms_GetSerialNumbers
 - dms.h, 1217
- pack_dms_GetUSBComp
 - dms.h, 1217
- pack_dms_GetVoiceNumber
 - dms.h, 1218
- pack_dms_ResetToFactoryDefaults
 - dms.h, 1218
- pack_dms_ResetToFactoryDefaults_t, 553
 - spc, 553
- pack_dms_SLQSDmsSwiGetResetInfo
 - dms.h, 1222
- pack_dms_SLQSDmsSwiIndicationRegister
 - dms.h, 1222
- pack_dms_SLQSDmsSwiIndicationRegister_t, 556
 - resetInfoInd, 557
- pack_dms_SLQSGetBandCapability
 - dms.h, 1223

- pack_dms_SLQSGetERIFile
 - dms.h, [1223](#)
- pack_dms_SLQSSwiClearDyingGaspStatistics
 - dms.h, [1224](#)
- pack_dms_SLQSSwiGetCrashInfo
 - dms.h, [1224](#)
- pack_dms_SLQSSwiGetCrashInfo_t, [557](#)
 - clear, [557](#)
- pack_dms_SLQSSwiGetDyingGaspCfg
 - dms.h, [1224](#)
- pack_dms_SLQSSwiGetDyingGaspStatistics
 - dms.h, [1225](#)
- pack_dms_SLQSSwiGetFirmwareCurr
 - dms.h, [1225](#)
- pack_dms_SLQSSwiGetFwUpdateStatus
 - dms.h, [1226](#)
- pack_dms_SLQSSwiGetHostDevInfo
 - dms.h, [1226](#)
- pack_dms_SLQSSwiGetOSInfo
 - dms.h, [1226](#)
- pack_dms_SLQSSwiGetSerialNoExt
 - dms.h, [1227](#)
- pack_dms_SLQSSwiSetDyingGaspCfg
 - dms.h, [1227](#)
- pack_dms_SLQSSwiSetDyingGaspCfg_t, [557](#)
 - pDestSMSContent, [558](#)
 - pDestSMSNum, [558](#)
- pack_dms_SLQSSwiSetHostDevInfo
 - dms.h, [1228](#)
- pack_dms_SLQSSwiSetHostDevInfo_t, [558](#)
 - manString, [558](#)
 - modelString, [558](#)
 - plasmaIDString, [559](#)
 - swVerString, [559](#)
- pack_dms_SLQSSwiSetOSInfo
 - dms.h, [1228](#)
- pack_dms_SLQSSwiSetOSInfo_t, [559](#)
 - nameString, [559](#)
 - versionString, [559](#)
- pack_dms_SLQSUIMGetState
 - dms.h, [1228](#)
- pack_dms_SetActivationStatusCallback
 - dms.h, [1218](#)
- pack_dms_SetActivationStatusCallback_t, [553](#)
 - activationState, [553](#)
- pack_dms_SetCrashAction
 - dms.h, [1219](#)
- pack_dms_SetCrashAction_t, [553](#)
 - crashAction, [554](#)
- pack_dms_SetCustFeature
 - dms.h, [1219](#)
- pack_dms_SetCustFeature_t, [554](#)
 - DHCPRelayEnabled, [554](#)
 - DisableIMSI, [554](#)
 - GPSPMP, [554](#)
 - GPSSel, [554](#)
 - GpsEnable, [554](#)
 - IPFamSupport, [555](#)
 - IsVoiceEnabled, [555](#)
 - RMAutoConnect, [555](#)
 - SMSSupport, [555](#)
- pack_dms_SetCustFeaturesV2
 - dms.h, [1220](#)
- pack_dms_SetCustFeaturesV2_t, [555](#)
 - cust_id, [555](#)
 - cust_value, [555](#)
 - Tlvresult, [555](#)
 - value_length, [555](#)
- pack_dms_SetEventReport
 - dms.h, [1220](#)
- pack_dms_SetEventReport_t, [556](#)
 - mode, [556](#)
- pack_dms_SetFirmwarePreference
 - dms.h, [1221](#)
- pack_dms_SetPower
 - dms.h, [1221](#)
- pack_dms_SetPower_t, [556](#)
 - mode, [556](#)
 - Tlvresult, [556](#)
- pack_dms_SetUSBComp
 - dms.h, [1222](#)
- pack_dms_SetUSBComp_t, [556](#)
 - Tlvresult, [556](#)
 - USBComp, [556](#)
- pack_dms_UIMChangePIN_t, [559](#)
 - id, [560](#)
 - newValue, [560](#)
 - oldValue, [560](#)
- pack_dms_UIMChangePIN
 - dms.h, [1229](#)
- pack_dms_UIMGetControlKeyStatus
 - dms.h, [1229](#)
- pack_dms_UIMGetControlKeyStatus_t, [560](#)
 - facility, [560](#)
- pack_dms_UIMGetICCID_t, [561](#)
 - Tlvresult, [561](#)
- pack_dms_UIMGetICCID
 - dms.h, [1230](#)
- pack_dms_UIMGetPINStatus
 - dms.h, [1230](#)
- pack_dms_UIMSetControlKeyProtection
 - dms.h, [1230](#)
- pack_dms_UIMSetControlKeyProtection_t, [561](#)
 - facility, [562](#)
 - facilityCk, [562](#)
 - facilityState, [562](#)
- pack_dms_UIMSetPINProtection
 - dms.h, [1231](#)
- pack_dms_UIMSetPINProtection_t, [562](#)
 - bEnable, [562](#)
 - id, [562](#)
 - value, [562](#)
- pack_dms_UIMUnblockControlKey
 - dms.h, [1231](#)
- pack_dms_UIMUnblockControlKey_t, [563](#)
 - facility, [563](#)

- facilityCk, 563
- pack_dms_UIMUnblockPIN_t, 563
 - id, 564
 - newPin, 564
 - pukValue, 564
- pack_dms_UIMUnblockPIN
 - dms.h, 1232
- pack_dms_UIMVerifyPIN_t, 564
 - id, 564
 - value, 564
- pack_dms_UIMVerifyPIN
 - dms.h, 1232
- pack_dms_ValidateSPC
 - dms.h, 1232
- pack_fms_GetImagesPreference
 - fms.h, 1260
- pack_fms_GetImagesPreference_t, 565
 - Tlvresult, 565
- pack_fms_GetStoredImages
 - fms.h, 1261
- pack_fms_GetStoredImages_t, 565
 - Tlvresult, 565
- pack_fms_SetImagesPreference
 - fms.h, 1261
- pack_fms_SetImagesPreference_t, 565
 - bForceDownload, 566
 - imageListSize, 566
 - modemindex, 566
 - pImageList, 566
 - Tlvresult, 566
- pack_loc_Delete_Assist_Data_t, 566
 - pBdsSVInfo, 567
 - pCellDb, 567
 - pClkInfo, 567
 - pGnssData, 567
 - pSVInfo, 567
 - Tlvresult, 567
- pack_loc_DeleteAssistData
 - loc.h, 1268
- pack_loc_EventRegister
 - loc.h, 1268
- pack_loc_EventRegister_t, 567
 - eventRegister, 570
 - Tlvresult, 570
- pack_loc_SLQSLOCGetBestAvailPos
 - loc.h, 1269
- pack_loc_SLQSLOCGetBestAvailPos_t, 571
 - Tlvresult, 572
 - xid, 572
- pack_loc_SLQSLOCInjectPosition
 - loc.h, 1269
- pack_loc_SLQSLOCInjectPosition_t, 572
 - altitudeSrcInfo, 575
 - altitudeWrtEllipsoid, 575
 - altitudeWrtMeanSeaLevel, 576
 - has_altitudeSrcInfo, 576
 - has_altitudeWrtEllipsoid, 576
 - has_altitudeWrtMeanSeaLevel, 576
 - has_horConfidence, 576
 - has_horReliability, 576
 - has_horUncCircular, 576
 - has_latitude, 576
 - has_longitude, 576
 - has_positionSrc, 576
 - has_rawHorConfidence, 576
 - has_rawHorUncCircular, 576
 - has_timestampAge, 576
 - has_timestampUtc, 576
 - has_vertConfidence, 576
 - has_vertReliability, 576
 - has_vertUnc, 576
 - horConfidence, 576
 - horReliability, 576
 - horUncCircular, 576
 - latitude, 576
 - longitude, 576
 - positionSrc, 576
 - rawHorConfidence, 577
 - rawHorUncCircular, 577
 - timestampAge, 577
 - timestampUtc, 577
 - vertConfidence, 577
 - vertReliability, 577
 - vertUnc, 577
- pack_loc_SLQSLOCInjectSensorData
 - loc.h, 1270
- pack_loc_SLQSLOCInjectSensorData_t, 577
 - accelTemp, 578
 - acceleroData, 578
 - acceleroTimeSrc, 578
 - gyroData, 578
 - gyroTemp, 579
 - gyroTimeSrc, 579
 - has_accelTemp, 579
 - has_acceleroTimeSrc, 579
 - has_accleroData, 579
 - has_gyroData, 579
 - has_gyroTemp, 579
 - has_gyroTimeSrc, 579
 - has_opaqueId, 579
 - opaqueId, 579
- pack_loc_SLQSLOCInjectUTCTime
 - loc.h, 1270
- pack_loc_SLQSLOCInjectUTCTime_t, 579
 - timeMsec, 579
 - timeUncMsec, 579
- pack_loc_SLQSLOCSetCradleMountConfig
 - loc.h, 1271
- pack_loc_SLQSLOCSetCradleMountConfig_t, 580
 - confidence, 580
 - has_confidence, 580
 - state, 580
- pack_loc_SetExtPowerState
 - loc.h, 1268
- pack_loc_SetExtPowerState_t, 570
 - extPowerState, 570

- Tlvresult, [570](#)
- pack_loc_SetOperationMode
 - loc.h, [1269](#)
- pack_loc_SetOperationMode_t, [571](#)
 - mode, [571](#)
 - Tlvresult, [571](#)
- pack_loc_Start
 - loc.h, [1271](#)
- pack_loc_Start_t, [580](#)
 - pApplicationInfo, [581](#)
 - pConfigAltitudeAssumed, [581](#)
 - pHorizontalAccuracyLvl, [582](#)
 - pIntermediateReportState, [582](#)
 - pMinIntervalTime, [582](#)
 - pRecurrenceType, [582](#)
 - SessionId, [582](#)
 - Tlvresult, [582](#)
- pack_loc_Stop
 - loc.h, [1271](#)
- pack_loc_Stop_t, [582](#)
 - SessionId, [582](#)
 - Tlvresult, [582](#)
- pack_nas_GetACCOLC
 - nas.h, [1285](#)
- pack_nas_GetANAAAAuthenticationStatus
 - nas.h, [1286](#)
- pack_nas_GetCDMANetworkParameters
 - nas.h, [1286](#)
- pack_nas_GetHomeNetwork
 - nas.h, [1286](#)
- pack_nas_GetNetworkPreference
 - nas.h, [1287](#)
- pack_nas_GetRFInfo
 - nas.h, [1287](#)
- pack_nas_GetServingNetwork
 - nas.h, [1287](#)
- pack_nas_GetServingNetworkCapabilities
 - nas.h, [1288](#)
- pack_nas_GetSignalStrengths
 - nas.h, [1288](#)
- pack_nas_PerformNetworkScan
 - nas.h, [1288](#)
- pack_nas_SLQSGetNetworkTime
 - nas.h, [1290](#)
- pack_nas_SLQSGetPLMNName
 - nas.h, [1290](#)
- pack_nas_SLQSGetPLMNName_t, [584](#)
 - mcc, [584](#)
 - mnc, [584](#)
 - pMncPcsStatus, [584](#)
- pack_nas_SLQSGetServingSystem
 - nas.h, [1291](#)
- pack_nas_SLQSGetSignalStrength
 - nas.h, [1291](#)
- pack_nas_SLQSGetSysInfo
 - nas.h, [1291](#)
- pack_nas_SLQSGetSysSelectionPref
 - nas.h, [1292](#)
- pack_nas_SLQSInitiateNetworkRegistration
 - nas.h, [1292](#)
- pack_nas_SLQSInitiateNetworkRegistration_t, [585](#)
 - pChangeDuration, [585](#)
 - pMNRInfo, [585](#)
 - pMncPcsDigitStatus, [585](#)
 - regAction, [585](#)
- pack_nas_SLQSNasConfigSigInfo2
 - nas.h, [1292](#)
- pack_nas_SLQSNasConfigSigInfo2_t, [586](#)
 - pCDMAECIODelta, [589](#)
 - pCDMAECIOThresh, [589](#)
 - pCDMARSSIDelta, [589](#)
 - pCDMARSSIThresh, [589](#)
 - pGSMRSSIDelta, [589](#)
 - pGSMRSSIThresh, [589](#)
 - pHDRECIODelta, [589](#)
 - pHDRECIOThresh, [589](#)
 - pHDRIODelta, [589](#)
 - pHDRIOThresh, [589](#)
 - pHDDRSSIDelta, [589](#)
 - pHDDRSSIThresh, [589](#)
 - pHDRSINRDelta, [589](#)
 - pHDRSINRThresh, [589](#)
 - pLTERSRPDelta, [590](#)
 - pLTERSRPThresh, [590](#)
 - pLTERSRQDelta, [590](#)
 - pLTERSRQThresh, [590](#)
 - pLTERSSIDelta, [590](#)
 - pLTERSSIThresh, [590](#)
 - pLTESNRDelta, [590](#)
 - pLTESNRThresh, [590](#)
 - pLTESigRptConfig, [590](#)
 - pTDSCDMAECIODelta, [590](#)
 - pTDSCDMAECIOThresh, [590](#)
 - pTDSCDMARSCPDelta, [590](#)
 - pTDSCDMARSCPThresh, [590](#)
 - pTDSCDMARSSIDelta, [590](#)
 - pTDSCDMARSSIThresh, [590](#)
 - pTDSCDMASINRDelta, [590](#)
 - pTDSCDMASINRThresh, [590](#)
 - pWCDMAECIODelta, [590](#)
 - pWCDMAECIOThresh, [590](#)
 - pWCDMARSSIDelta, [590](#)
 - pWCDMARSSIThresh, [590](#)
- pack_nas_SLQSNasGetCellLocationInfo
 - nas.h, [1293](#)
- pack_nas_SLQSNasGetSigInfo
 - nas.h, [1293](#)
- pack_nas_SLQSNasIndicationRegisterExt
 - nas.h, [1293](#)
- pack_nas_SLQSNasIndicationRegisterExt_t, [590](#)
 - pDDTMInd, [593](#)
 - pDualStandByPrefInd, [593](#)
 - pErrorRateInd, [593](#)
 - pHDRNewUATIAssInd, [593](#)
 - pHDRSessionCloseInd, [593](#)
 - pLTECphyCa, [593](#)

- pManagedRoamingInd, 593
- pNetworkTimeInd, 593
- pServingSystemInd, 593
- pSignalStrengthInd, 593
- pSubscriptionInfoInd, 593
- pSysInfoInd, 593
- pSystemSelectionInd, 593
- pack_nas_SLQSNasSwiIndicationRegister
 - nas.h, 1294
- pack_nas_SLQSNasSwiIndicationRegister_t, 593
 - gsmUmtsDI, 594
 - gsmUmtsUI, 594
 - lteEmmDI, 594
 - lteEmmUI, 594
 - lteEsmDI, 594
 - lteEsmUI, 594
 - pRankIndicatorInd, 594
 - pTimer, 594
- pack_nas_SLQSNasSwiModemStatus
 - nas.h, 1294
- pack_nas_SLQSSetBandPreference
 - nas.h, 1294
- pack_nas_SLQSSetSignalStrengthsCallback
 - nas.h, 1295
- pack_nas_SLQSSetSignalStrengthsCallback_t, 595
 - bEnable, 595
 - pSigIndReq, 595
- pack_nas_SLQSSetSysSelectionPref
 - nas.h, 1295
- pack_nas_SLQSSetSysSelectionPref_t, 595
 - pAcqOrderPref, 599
 - pBandPref, 599
 - pCSGID, 599
 - pChgDuration, 599
 - pEmerMode, 599
 - pGWAqOrderPref, 599
 - pLTEBandPref, 600
 - pMNCIncPCSDigStat, 600
 - pModePref, 600
 - pNetSelPref, 600
 - pPRLPref, 600
 - pRAT, 600
 - pRoamPref, 600
 - pSrvDomainPref, 600
 - pSrvRegRestriction, 600
 - pTdsdmaBandPref, 600
- pack_nas_SLQSSwiGetLteCQI
 - nas.h, 1295
- pack_nas_SLQSSwiGetLteSccRxInfo
 - nas.h, 1296
- pack_nas_SetACCOLC_t, 582
 - accolc, 583
 - spc, 583
- pack_nas_SetACCOLC
 - nas.h, 1289
- pack_nas_SetLURejectCallback
 - nas.h, 1289
- pack_nas_SetNetworkPreference
 - nas.h, 1289
- pack_nas_SetNetworkPreference_t, 583
 - Duration, 584
 - TechnologyPref, 584
 - Tlvresult, 584
- pack_nas_SetRFInfoCallback
 - nas.h, 1290
- pack_nas_SlqsGetLTECphyCAInfo
 - nas.h, 1290
- pack_qmi_t, 600
 - msgid, 600
 - svc, 600
 - timeout, 601
 - xid, 601
- pack_qos_SLQSQosGetNetworkStatus
 - qos.h, 1751
- pack_qos_SLQSQosSwiReadApnExtraParams
 - qos.h, 1751
- pack_qos_SLQSQosSwiReadApnExtraParams_t, 601
 - apnId, 601
- pack_qos_SLQSQosSwiReadDataStats
 - qos.h, 1752
- pack_qos_SLQSQosSwiReadDataStats_t, 601
 - apnId, 601
- pack_qos_SLQSSetQosEventCallback
 - qos.h, 1753
- pack_qos_SLQSSetQosEventCallback_t, 602
 - enable, 602
- pack_sms_SLQSDeleteSMS_t, 603
 - pMessageIndex, 604
 - pMessageMode, 604
 - pMessageTag, 604
 - storageType, 604
- pack_sms_SLQSDeleteSMS
 - sms.h, 1763
- pack_sms_SLQSGetSMS_t, 604
 - messageIndex, 605
 - pMessageMode, 605
 - storageType, 605
- pack_sms_SLQSGetSMSList
 - sms.h, 1764
- pack_sms_SLQSGetSMSList_t, 605
 - pMessageMode, 605
 - pRequestedTag, 605
 - storageType, 605
- pack_sms_SLQSGetSMS
 - sms.h, 1763
- pack_sms_SLQSModifySMSStatus
 - sms.h, 1764
- pack_sms_SLQSModifySMSStatus_t, 605
 - messageIndex, 606
 - messageTag, 606
 - pMessageMode, 606
 - storageType, 606
- pack_sms_SendSMS_t, 602
 - messageFormat, 603
 - messageSize, 603
 - pLinktimer, 603

- pMessage, [603](#)
- pack_sms_SendSMS
 - sms.h, [1762](#)
- pack_sms_SetNewSMSCallback
 - sms.h, [1762](#)
- pack_sms_SetNewSMSCallback_t, [603](#)
 - status, [603](#)
- pack_swiloc_SwiLocGetAutoStart
 - swiloc.h, [1769](#)
- pack_swiloc_SwiLocSetAutoStart
 - swiloc.h, [1770](#)
- pack_swiloc_SwiLocSetAutoStart_t, [606](#)
 - fix_rate, [608](#)
 - fix_type, [608](#)
 - function, [608](#)
 - max_dist, [608](#)
 - max_time, [608](#)
 - set_fix_rate, [608](#)
 - set_fix_type, [608](#)
 - set_function, [608](#)
 - set_max_dist, [608](#)
 - set_max_time, [608](#)
- pack_swioima_SLQSOMADMAAlertCallback
 - swioima.h, [1772](#)
- pack_swioima_SLQSOMADMCancelSession
 - swioima.h, [1773](#)
- pack_swioima_SLQSOMADMCancelSession_t, [608](#)
 - sessionType, [608](#)
- pack_swioima_SLQSOMADMGetSessionInfo
 - swioima.h, [1773](#)
- pack_swioima_SLQSOMADMGetSessionInfo_t, [609](#)
 - SessionType, [609](#)
- pack_swioima_SLQSOMADMGetSettings
 - swioima.h, [1774](#)
- pack_swioima_SLQSOMADMSendSelection
 - swioima.h, [1775](#)
- pack_swioima_SLQSOMADMSendSelection_t, [609](#)
 - pDeferTime, [610](#)
 - pRejectReason, [610](#)
 - selection, [610](#)
- pack_swioima_SLQSOMADMSetSettings
 - swioima.h, [1775](#)
- pack_swioima_SLQSOMADMSetSettings_t, [610](#)
 - FOTAUpdate, [611](#)
 - FOTAdownload, [610](#)
 - pAutosdm, [611](#)
 - pFwAutoCheck, [611](#)
- pack_swioima_SLQSOMADMStartSession
 - swioima.h, [1776](#)
- pack_swioima_SLQSOMADMStartSession_t, [611](#)
 - sessionType, [611](#)
- pack_uim_ChangePin
 - uim.h, [1783](#)
- pack_uim_ChangePin_t, [611](#)
 - changePIN, [612](#)
 - EncryptedPIN1, [612](#)
 - pIndicationToken, [612](#)
 - pKeyReferenceID, [612](#)
 - sessionInfo, [612](#)
 - Tlvresult, [612](#)
- pack_uim_GetCardStatus
 - uim.h, [1783](#)
- pack_uim_ReadTransparent
 - uim.h, [1784](#)
- pack_uim_ReadTransparent_t, [612](#)
 - fileIndex, [613](#)
 - pEncryptData, [613](#)
 - pIndicationToken, [613](#)
 - readTransparent, [613](#)
 - sessionInfo, [613](#)
 - Tlvresult, [613](#)
- pack_uim_SLQSUIEventRegister
 - uim.h, [1784](#)
- pack_uim_SLQSUIEventRegister_t, [614](#)
 - eventMask, [615](#)
- pack_uim_SLQSUIMGetSlotsStatus
 - uim.h, [1785](#)
- pack_uim_SLQSUIPowerDown
 - uim.h, [1785](#)
- pack_uim_SLQSUIPowerDown_t, [615](#)
 - slot, [615](#)
- pack_uim_SLQSUIPowerUp
 - uim.h, [1785](#)
- pack_uim_SLQSUIPowerUp_t, [615](#)
 - plgnoreHotSwapSwitch, [616](#)
 - slot, [616](#)
- pack_uim_SLQSUIMSwitchSlot
 - uim.h, [1786](#)
- pack_uim_SLQSUIMSwitchSlot_t, [616](#)
 - bLogicalSlot, [617](#)
 - ulPhysicalSlot, [617](#)
- pack_uim_SetPinProtection
 - uim.h, [1784](#)
- pack_uim_SetPinProtection_t, [613](#)
 - EncryptedPIN1, [614](#)
 - pIndicationToken, [614](#)
 - pKeyReferenceID, [614](#)
 - pinProtection, [614](#)
 - sessionInfo, [614](#)
 - Tlvresult, [614](#)
- pack_uim_UnblockPin
 - uim.h, [1786](#)
- pack_uim_UnblockPin_t, [617](#)
 - EncryptedPIN1, [617](#)
 - pIndicationToken, [618](#)
 - pKeyReferenceID, [618](#)
 - pinProtection, [618](#)
 - sessionInfo, [618](#)
 - Tlvresult, [618](#)
- pack_uim_VerifyPin
 - uim.h, [1787](#)
- pack_uim_VerifyPin_t, [618](#)
 - pEncryptedPIN1, [619](#)
 - pIndicationToken, [619](#)
 - pKeyReferenceID, [619](#)
 - sessionInfo, [619](#)

- Tlvresult, [619](#)
- verifyPIN, [619](#)
- pack_wds_DHCPv4ClientLeaseChange
 - wds.h, [1798](#)
- pack_wds_DHCPv4ClientLeaseChange_t, [619](#)
 - pEnableNotification, [619](#)
- pack_wds_GetAutoconnect
 - wds.h, [1799](#)
- pack_wds_GetByteTotals
 - wds.h, [1799](#)
- pack_wds_GetConnectionRate
 - wds.h, [1800](#)
- pack_wds_GetDataBearerTechnology
 - wds.h, [1800](#)
- pack_wds_GetDefaultProfile
 - wds.h, [1801](#)
- pack_wds_GetDefaultProfile_t, [619](#)
 - profiletype, [620](#)
- pack_wds_GetDefaultProfileNum
 - wds.h, [1801](#)
- pack_wds_GetDefaultProfileNum_t, [620](#)
 - family, [620](#)
 - type, [620](#)
- pack_wds_GetDormancyState
 - wds.h, [1801](#)
- pack_wds_GetDormancyState_t, [620](#)
- pack_wds_GetLastMobileIPError
 - wds.h, [1802](#)
- pack_wds_GetLastMobileIPError_t, [620](#)
- pack_wds_GetMobileIP_t, [620](#)
- pack_wds_GetMobileIPProfile
 - wds.h, [1803](#)
- pack_wds_GetMobileIPProfile_t, [620](#)
 - index, [621](#)
- pack_wds_GetMobileIP
 - wds.h, [1802](#)
- pack_wds_GetPacketStatistics
 - wds.h, [1803](#)
- pack_wds_GetPacketStatistics_t, [621](#)
 - pStatMask, [621](#)
- pack_wds_GetPacketStatus
 - wds.h, [1804](#)
- pack_wds_GetPacketStatus_t, [621](#)
 - statmask, [621](#)
- pack_wds_GetSessionDuration
 - wds.h, [1804](#)
- pack_wds_GetSessionDuration_t, [622](#)
- pack_wds_GetSessionState
 - wds.h, [1805](#)
- pack_wds_RMSetTransferStatistics
 - wds.h, [1805](#)
- pack_wds_RMSetTransferStatistics_t, [622](#)
 - RmTrasnferStaticsReq, [622](#)
- pack_wds_SLQSCreateProfile
 - wds.h, [1808](#)
- pack_wds_SLQSCreateProfile_t, [627](#)
 - pCurProfile, [628](#)
 - pProfileId, [628](#)
 - pProfileType, [628](#)
- pack_wds_SLQSDeleteProfile
 - wds.h, [1809](#)
- pack_wds_SLQSDeleteProfile_t, [628](#)
 - profileIndex, [628](#)
 - profileType, [628](#)
- pack_wds_SLQSGet3GPPConfigItem
 - wds.h, [1809](#)
- pack_wds_SLQSGetCurrDataSystemStat
 - wds.h, [1810](#)
- pack_wds_SLQSGetCurrDataSystemStat_t, [628](#)
- pack_wds_SLQSGetCurrentChannelRate
 - wds.h, [1810](#)
- pack_wds_SLQSGetDUNCallInfo
 - wds.h, [1811](#)
- pack_wds_SLQSGetDUNCallInfo_t, [628](#)
 - Mask, [629](#)
 - pReportChannelRate, [629](#)
 - pReportConnStatus, [629](#)
 - pReportDataBearerTech, [629](#)
 - pReportDormStatus, [629](#)
 - pTransferStatInd, [629](#)
- pack_wds_SLQSGetDataBearerTechnology
 - wds.h, [1811](#)
- pack_wds_SLQSGetDataBearerTechnology_t, [628](#)
- pack_wds_SLQSGetProfileSettings
 - wds.h, [1811](#)
- pack_wds_SLQSGetProfileSettings_t, [629](#)
 - ProfileId, [630](#)
 - ProfileType, [630](#)
- pack_wds_SLQSGetRuntimeSettings
 - wds.h, [1812](#)
- pack_wds_SLQSGetRuntimeSettings_t, [630](#)
 - pReqSettings, [631](#)
- pack_wds_SLQSModifyProfile
 - wds.h, [1812](#)
- pack_wds_SLQSModifyProfile_t, [631](#)
 - curProfile, [632](#)
 - pProfileId, [632](#)
 - pProfileType, [632](#)
- pack_wds_SLQSResetPacketStatics
 - wds.h, [1813](#)
- pack_wds_SLQSSetDHCPv4ClientConfig
 - wds.h, [1815](#)
- pack_wds_SLQSSetDHCPv4ClientConfig_t, [635](#)
 - pProfileId, [635](#)
- pack_wds_SLQSSetLoopback
 - wds.h, [1815](#)
- pack_wds_SLQSSetDHCPv4ClientConfig
 - wds.h, [1816](#)
- pack_wds_SLQSSetDHCPv4ClientConfig_t, [635](#)
 - pHwConfig, [636](#)
 - pProfileId, [636](#)
 - pRequestOptionList, [636](#)
- pack_wds_SLQSSetLoopback
 - wds.h, [1816](#)
- pack_wds_SLQSSetLoopback_t, [636](#)
 - loopbackMode, [637](#)

- loopbackMultiplier, [637](#)
- pack_wds_SLQSSet3GPPConfigItem
 - wds.h, [1813](#)
- pack_wds_SLQSSet3GPPConfigItem_t, [632](#)
 - LTEAttachProfileListLen, [633](#)
 - p3gppRelease, [633](#)
 - pDefaultPDNEnabled, [634](#)
 - pLTEAttachProfile, [634](#)
 - pLTEAttachProfileList, [634](#)
 - pProfileList, [634](#)
- pack_wds_SLQSSetIPFamilyPreference
 - wds.h, [1814](#)
- pack_wds_SLQSSetIPFamilyPreference_t, [634](#)
 - IPFamilyPreference, [634](#)
- pack_wds_SLQSSetWdsEventCallback
 - wds.h, [1814](#)
- pack_wds_SLQSSetWdsEventCallback_t, [634](#)
 - currentDataBearer, [635](#)
 - dataBearer, [635](#)
 - dataSystemStatus, [635](#)
 - dormancyStatus, [635](#)
 - interval, [635](#)
 - mobileIP, [635](#)
 - transferStats, [635](#)
- pack_wds_SLQSStartDataSession
 - wds.h, [1816](#)
- pack_wds_SLQSStartDataSession_t, [637](#)
 - pAuth, [638](#)
 - pPass, [638](#)
 - pTech, [638](#)
 - pUser, [638](#)
 - pprofileid3gpp, [638](#)
 - pprofileid3gpp2, [638](#)
- pack_wds_SLQSStopDataSession
 - wds.h, [1817](#)
- pack_wds_SLQSStopDataSession_t, [638](#)
 - psid, [638](#)
- pack_wds_SLQSWdsGoActive
 - wds.h, [1817](#)
- pack_wds_SLQSWdsGoDormant
 - wds.h, [1818](#)
- pack_wds_SLQSWdsSetEventReport
 - wds.h, [1818](#)
- pack_wds_SLQSWdsSetEventReport_t, [638](#)
 - pCurrChannelRateInd, [640](#)
 - pCurrDataBearerTechInd, [640](#)
 - pCurrPrefDataSysInd, [640](#)
 - pDataBearerTechInd, [640](#)
 - pDataCallStatusChangeInd, [640](#)
 - pDataSystemStatusChangeInd, [640](#)
 - pDormancyStatusInd, [640](#)
 - pEVDOPageMonPerChangeInd, [640](#)
 - pMIPStatusInd, [640](#)
 - pTransferStatInd, [640](#)
- pack_wds_SLQSWdsSwiPDPRuntimeSettings
 - wds.h, [1819](#)
- pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, [640](#)
 - contextId, [640](#)
 - contextType, [640](#)
- pack_wds_SetAutoconnect
 - wds.h, [1806](#)
- pack_wds_SetAutoconnect_t, [622](#)
 - acroamsetting, [622](#)
 - acsetting, [622](#)
- pack_wds_SetDefaultProfile
 - wds.h, [1806](#)
- pack_wds_SetDefaultProfile_t, [623](#)
 - authentication, [623](#)
 - ipAddress, [623](#)
 - pApnname, [623](#)
 - pName, [623](#)
 - pPassword, [623](#)
 - pUsername, [623](#)
 - pdpType, [623](#)
 - primaryDNS, [623](#)
 - profileType, [623](#)
 - secondaryDNS, [624](#)
- pack_wds_SetDefaultProfileNum
 - wds.h, [1806](#)
- pack_wds_SetDefaultProfileNum_t, [624](#)
 - family, [624](#)
 - index, [624](#)
 - type, [624](#)
- pack_wds_SetMobileIP_t, [624](#)
 - mode, [624](#)
- pack_wds_SetMobileIPParameters
 - wds.h, [1807](#)
- pack_wds_SetMobileIPParameters_t, [624](#)
 - pHA2002bis, [625](#)
 - pHAAAuthenticator, [626](#)
 - pMode, [626](#)
 - pReRegPeriod, [626](#)
 - pReRegTraffic, [626](#)
 - pRetryInterval, [626](#)
 - pRetryLimit, [626](#)
 - pSPC, [626](#)
- pack_wds_SetMobileIPProfile
 - wds.h, [1808](#)
- pack_wds_SetMobileIPProfile_t, [626](#)
 - index, [627](#)
 - pAAASPI, [627](#)
 - pAddress, [627](#)
 - pEnabled, [627](#)
 - pHASPI, [627](#)
 - pMNAHA, [627](#)
 - pMNHA, [627](#)
 - pNAI, [627](#)
 - pPrimaryHA, [627](#)
 - pRevTunneling, [627](#)
 - pSecondaryHA, [627](#)
 - spc, [627](#)
- pack_wds_SetMobileIP
 - wds.h, [1807](#)
- PackCreateProfileOut, [641](#)
 - ExtErrorCode, [641](#)
 - ProfileIndex, [641](#)

- ProfileType, 641
- package_name
 - omaDmFotaTlv, 547
 - omaDmFotaTlvExt, 549
 - unpack_omaDmFotaTlv_t, 1017
- packageSize
 - omaDmFotaTlvExt, 549
- packageid_str
 - slqsfwinfo_s, 790
 - unpack_dms_GetFirmwareInfo_t, 931
- packetSrvStatus
 - qaGobiApiCbk.h, 1331
- packetZone
 - CDMASysInfo, 158
 - nas_CDMASysInfo, 414
- packetZoneValid
 - CDMASysInfo, 158
 - nas_CDMASysInfo, 414
- packgetDyingGaspCfg, 641
 - pDestSMSContent, 641
 - pDestSMSNum, 641
- packgetDyingGaspStatistics, 641
 - pSMSAttemptedFlag, 642
 - pTimeStamp, 642
- path
 - fileInfo, 229
 - uim_fileInfo, 881
- pathLen
 - fileInfo, 229
 - uim_fileInfo, 882
- pbIMSRegistered
 - imsaRegStatusInfo, 303
- pbPlatform
 - unpack_dms_GetOfflineReason_t, 936
- pci
 - cellParams, 160
 - ltePCI, 387
 - NASPhyCaAggPcellInfo, 513
 - NASPhyCaAggScellIndType, 515
 - NASPhyCaAggScellInfo, 516
 - nas_PhyCaAggPcellInfo, 456
 - nas_PhyCaAggScellIndType, 458
 - nas_PhyCaAggScellInfo, 461
 - nas_cellParams, 416
 - nas_umtsLTENbrCell, 486
 - PhyCaAggPcellInfo, 651
 - PhyCaAggScellIndType, 653
 - PhyCaAggScellInfo, 655
 - umtsLTENbrCell, 917
- pcsFQDNAddress
 - PCSCFFQDNAddressList, 644
 - wds_PCSCFFQDNAddressList, 1169
- pdpType
 - pack_wds_SetDefaultProfile_t, 623
- pdptype
 - unpack_wds_GetDefaultProfile_t, 1064
- peakRate
 - tokenBucket, 867
- unpack_qos_tokenBucket_t, 1039
- peakThroughputClass
 - GPRSQoS, 272
 - GPRSRequestedQoS, 273
 - LibPackGPRSRequestedQoS, 321
 - wds_GPRSQoS, 1166
- peerNumberInfo, 647
 - callID, 649
 - numLen, 649
 - numPI, 649
 - numPlan, 649
 - numSI, 649
 - numType, 649
 - number, 649
- PerformNetworkScan
 - qaGobiApiNas.h, 1523
- PersistentTechPref
 - unpack_nas_GetNetworkPreference_t, 984
- persoFeature
 - appStats, 108
 - appStatus, 111
 - uim_appStatus, 877
- persoRetries
 - appStats, 108
 - appStatus, 111
 - uim_appStatus, 877
- persoState
 - appStats, 108
 - appStatus, 111
 - uim_appStatus, 877
- persoUnblockRetries
 - appStats, 108
 - appStatus, 111
 - uim_appStatus, 877
- personalizationStatus, 649
 - feature, 650
 - numFeatures, 650
 - unblockLeft, 650
 - verifyLeft, 650
- pfailureCause
 - USSResp, 1092
- phase
 - rxInfo, 735
- PhyCaAggPcellInfo, 650
 - dl_bw_value, 650
 - freq, 651
 - iLTEbandValue, 651
 - NasGetLTECphyCaInfo, 498
 - pci, 651
 - TlvPresent, 651
- PhyCaAggScellDIBw, 651
 - dl_bw_value, 651
 - NasGetLTECphyCaInfo, 498
 - TlvPresent, 651
- PhyCaAggScellIndType, 652
 - freq, 652
 - NasGetLTECphyCaInfo, 498
 - pci, 653

- scell_state, [653](#)
 - TlvPresent, [653](#)
- PhyCaAggScellIndex, [651](#)
 - NasGetLTECphyCaInfo, [498](#)
 - scell_idx, [652](#)
 - TlvPresent, [652](#)
- PhyCaAggScellInfo, [653](#)
 - dl_bw_value, [655](#)
 - freq, [655](#)
 - iLTEbandValue, [655](#)
 - NasGetLTECphyCaInfo, [498](#)
 - pci, [655](#)
 - scell_state, [655](#)
 - TlvPresent, [655](#)
- PhysicalLayer
 - protocolSubtypeElement, [676](#)
- PI
 - callFWExtInfo, [133](#)
 - calledPartyInfo, [129](#)
 - callerIDInfo, [130](#)
 - callingPartyInfo, [138](#)
 - redirNumInfo, [722](#)
- PilotEnergy
 - NetworkStatEVDO, [538](#)
- PilotPN
 - PilotSetParams, [656](#)
- PilotSetData, [655](#)
 - NumPilots, [656](#)
 - pPilotSetInfo, [656](#)
- PilotSetParams, [656](#)
 - PilotPN, [656](#)
 - PilotStrength, [656](#)
 - PilotType, [656](#)
- PilotStrength
 - PilotSetParams, [656](#)
- PilotType
 - PilotSetParams, [656](#)
- pin1Len
 - encryptedPIN1, [221](#)
 - uim_encryptedPIN1, [881](#)
- pin1Retries
 - appStats, [108](#)
 - appStatus, [111](#)
 - uim_appStatus, [877](#)
- pin1State
 - appStats, [108](#)
 - appStatus, [111](#)
 - uim_appStatus, [877](#)
- pin1Val
 - encryptedPIN1, [222](#)
 - uim_encryptedPIN1, [881](#)
- pin2Retries
 - appStats, [108](#)
 - appStatus, [111](#)
 - uim_appStatus, [877](#)
- pin2State
 - appStats, [108](#)
 - appStatus, [111](#)
- uim_appStatus, [877](#)
- pinID
 - changeUIMPIN, [161](#)
 - setPINProtection, [771](#)
 - uim_changeUIMPIN, [880](#)
 - uim_setPINProtection, [886](#)
 - uim_unblockUIMPIN, [889](#)
 - uim_verifyUIMPIN, [890](#)
 - unblockUIMPIN, [924](#)
 - verifyUIMPIN, [1094](#)
- pinLen
 - changeUIMPIN, [161](#)
 - uim_changeUIMPIN, [880](#)
 - uim_verifyUIMPIN, [890](#)
 - verifyUIMPIN, [1094](#)
- pinLength
 - setPINProtection, [771](#)
 - uim_setPINProtection, [886](#)
- pinOperation
 - setPINProtection, [771](#)
 - uim_setPINProtection, [886](#)
- pinProtection
 - pack_uim_SetPinProtection_t, [614](#)
 - pack_uim_UnblockPin_t, [618](#)
 - UIMSetPinProtectionReq, [908](#)
- pinVal
 - changeUIMPIN, [161](#)
 - uim_changeUIMPIN, [880](#)
 - uim_verifyUIMPIN, [890](#)
 - verifyUIMPIN, [1094](#)
- pinValue
 - setPINProtection, [771](#)
 - uim_setPINProtection, [886](#)
- PkQmiNasGetRFBandInfo
 - qaNasGetRFBandInfo.h, [1741](#)
- PkQmiNasPerformNetworkScan
 - qaNasPerformNetworkScan.h, [1742](#)
- PkgDescLength
 - unpack_swioma_SLQSOMADMGetSessionInfo↔_t, [1050](#)
- PkgDescription
 - unpack_swioma_SLQSOMADMGetSessionInfo↔_t, [1050](#)
- PkgName
 - unpack_swioma_SLQSOMADMGetSessionInfo↔_t, [1050](#)
- PkgNameLength
 - unpack_swioma_SLQSOMADMGetSessionInfo↔_t, [1050](#)
- pkgver
 - CurrentImgList, [182](#)
 - unpack_dms_SLQSSwiGetFirmwareCurr_t, [949](#)
- PktErrRate
 - unpack_qos_swiQosFlow_t, [1038](#)
- pktErrRate, [656](#)
 - exponent, [657](#)
 - multiplier, [657](#)
- PktStatElmntsV4

- WdsPktStatisticsResp, [1188](#)
- PktStatElmntsV6
 - WdsPktStatisticsResp, [1188](#)
- plasmaIDString
 - pack_dms_SLQSSwiSetHostDevInfo_t, [559](#)
 - unpack_dms_SLQSSwiGetHostDevInfo_t, [952](#)
- plmn
 - GERANInfo, [237](#)
 - LTEInfoIntrafreq, [384](#)
 - nas_GERANInfo, [425](#)
 - nas_LTEInfoIntrafreq, [443](#)
 - nas_UMTSInfo, [484](#)
 - UMTSInfo, [915](#)
- polarityIncluded
 - lineCtrlInfo, [351](#)
- Port, [659](#)
 - port, [660](#)
 - range, [660](#)
- port
 - Port, [660](#)
 - unpack_qos_Port_t, [1021](#)
- Position Determination Service (PDS), [39](#)
- positionSrc
 - pack_loc_SLQSLOCInjectPosition_t, [576](#)
- pprofileid3gpp
 - pack_wds_SLQSSStartDataSession_t, [638](#)
- pprofileid3gpp2
 - pack_wds_SLQSSStartDataSession_t, [638](#)
- prDNSIPv4Address
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, [1085](#)
- prDNSIPv6Address
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, [1085](#)
- prPCSCFIPv4Address
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, [1085](#)
- prPCSCFIPv6Address
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, [1085](#)
- Precedence
 - unpack_qos_swiQoSFilter_t, [1034](#)
- precedenceClass
 - GPRSQoS, [272](#)
 - GPRSRequestedQoS, [273](#)
 - LibPackGPRSRequestedQoS, [321](#)
 - wds_GPRSQoS, [1166](#)
- precisionDilution
 - qaGobiApiCbk.h, [1332](#)
- precisionDilution_s, [660](#)
 - HDOP, [660](#)
 - PDOP, [660](#)
 - VDOP, [660](#)
- PrefImageList, [660](#)
 - listEntries, [661](#)
 - listSize, [661](#)
- prefNetwork
 - unpack_wds_SLQSGetCurrDataSystemStat↔_t, [1073](#)
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- prefVoiceSO, [661](#)
 - evrcCapability, [663](#)
 - homeOrigVoiceSO, [663](#)
 - homePageVoiceSO, [663](#)
 - namID, [663](#)
 - roamOrigVoiceSO, [663](#)
- Preferred
 - nas_QmiNas3GppNetworkInfo, [462](#)
 - SlqsNas3GppNetworkInfo, [792](#)
- prefixLen
 - IPv6Addr, [318](#)
 - unpack_qos_IPv6Addr_t, [1020](#)
- presentationInd
 - ECTNum, [221](#)
 - remotePartyNum, [725](#)
- priChA
 - CDMAChannel, [146](#)
- priChB
 - CDMAChannel, [146](#)
- priSize
 - unpack_dms_GetFirmwareRevisions_t, [932](#)
- pridns
 - unpack_wds_GetDefaultProfile_t, [1064](#)
- pridnsv6
 - unpack_wds_GetDefaultProfile_t, [1064](#)
- PrimaryDNSV4
 - unpack_wds_SLQSGetRuntimeSettings_t, [1079](#)
- PrimaryDNSV6
 - unpack_wds_SLQSGetRuntimeSettings_t, [1079](#)
- primaryDNS
 - pack_wds_SetDefaultProfile_t, [623](#)
- primaryHA
 - unpack_wds_GetMobileIPPProfile_t, [1066](#)
- privacyPref
 - voiceSetPrefPrivacy, [1138](#)
- priver
 - CurrentImgList, [182](#)
 - unpack_dms_SLQSSwiGetFirmwareCurr_t, [949](#)
- priversion_str
 - slqsfwinfno_s, [790](#)
 - unpack_dms_GetFirmwareInfo_t, [931](#)
- Profile
 - GetAudioPathConfigReq, [241](#)
 - GetAudioProfileResp, [245](#)
 - GetAudioVoITLBCConfigReq, [245](#)
 - GetM2MAVMuteReq, [263](#)
 - GetM2MAudioProfileResp, [262](#)
 - GetM2MAudioVolumeReq, [262](#)
 - GetM2MSpkrGainReq, [265](#)
 - SetAudioPathConfigReq, [756](#)
 - SetAudioProfileReq, [757](#)
 - SetAudioVoITLBCConfigReq, [758](#)
 - SetM2MAVMuteReq, [769](#)
 - SetM2MAudioAVCFGReq, [766](#)

- SetM2MAudioProfileReq, [768](#)
- SetM2MAudioVolumeReq, [769](#)
- SetM2MSpkrGainReq, [770](#)
- Profile3GPP2, [669](#)
 - pAPNClass3GPP2, [673](#)
 - pAPNEnabled3GPP2, [673](#)
 - pAllowLinger, [673](#)
 - pApnString, [673](#)
 - pApnStringSize, [673](#)
 - pAppPriority, [673](#)
 - pAppType, [673](#)
 - pAuthPassword, [673](#)
 - pAuthPasswordSize, [673](#)
 - pAuthProtocol, [673](#)
 - pAuthRetryCount, [673](#)
 - pAuthTimeout, [673](#)
 - pDataMode, [673](#)
 - pDataRate, [673](#)
 - plpcpAckTimeout, [673](#)
 - plpcpCreqRetryCount, [673](#)
 - plsPscfAddressNedded, [674](#)
 - pLcpAckTimeout, [674](#)
 - pLcpCreqRetryCount, [674](#)
 - pNegoDnsSrvrPref, [674](#)
 - pPDNInactivTimeout3GPP2, [674](#)
 - pPdnType, [674](#)
 - pPppSessCloseTimer1x, [674](#)
 - pPppSessCloseTimerDO, [674](#)
 - pPriV6DnsAddress, [674](#)
 - pPrimaryV4DnsAddress, [674](#)
 - pRATType, [674](#)
 - pSecV6DnsAddress, [674](#)
 - pSecondaryV4DnsAddress, [674](#)
 - pUserId, [674](#)
 - pUserIdSize, [674](#)
- Profile3GPP, [663](#)
 - pAPNClass, [667](#)
 - pAPNDisabledFlag, [667](#)
 - pAPNName, [667](#)
 - pAPNnameSize, [667](#)
 - pAddrAllocPref, [667](#)
 - pAuthenticationPref, [667](#)
 - pGPRSMinimumQoS, [667](#)
 - pGPRSRequestedQoS, [668](#)
 - pIPv4AddrPref, [668](#)
 - pIPv6AddPref, [668](#)
 - plmCnFlag, [668](#)
 - pPDNInactivTimeout, [668](#)
 - pPDPTtype, [668](#)
 - pPassword, [668](#)
 - pPasswordSize, [668](#)
 - pPscfAddrUsingDhcp, [668](#)
 - pPscfAddrUsingPCO, [668](#)
 - pPdpAccessConFlag, [668](#)
 - pPdpContext, [668](#)
 - pPdpDataCompType, [668](#)
 - pPdpHdrCompType, [668](#)
 - pPriDNSIPv4AddPref, [668](#)
 - pPriDNSIPv6addpref, [668](#)
 - pPrimaryID, [668](#)
 - pProfilename, [668](#)
 - pProfilenameSize, [668](#)
 - pQoSClassID, [668](#)
 - pSecDNSIPv4AddPref, [668](#)
 - pSecDNSIPv6addpref, [668](#)
 - pSecondaryFlag, [668](#)
 - pTFTID1Params, [669](#)
 - pTFTID2Params, [669](#)
 - pUMTSMInQoS, [669](#)
 - pUMTSMInQoSSigInd, [669](#)
 - pUMTSReqQoSsigInd, [669](#)
 - pUMTSReqQoS, [669](#)
 - pUsername, [669](#)
 - pUsernameSize, [669](#)
- ProfileID
 - _GetProfileSettingIn, [58](#)
 - unpack_wds_SLQSGetRuntimeSettings_t, [1079](#)
- ProfileId
 - pack_wds_SLQSGetProfileSettings_t, [630](#)
- profileId
 - wds_DHCPPProfileIdTlv, [1162](#)
 - wds_DHCPv4ProfileId, [1164](#)
 - WdsDHCPv4ProfileId, [1183](#)
 - wdsDhcpv4ProfileId, [1184](#)
- ProfileId3GPP2
 - unpack_qos_swiQoSFlow_t, [1038](#)
- ProfileIdTlv
 - unpack_wds_DHCPv4ClientLease_ind_t, [1060](#)
- ProfileIdentifier, [674](#)
 - profileIndex, [675](#)
 - profileType, [675](#)
- ProfileIndex
 - PackCreateProfileOut, [641](#)
- profileIndex
 - pack_wds_SLQSDeleteProfile_t, [628](#)
 - ProfileIdentifier, [675](#)
 - SLQSDeleteProfileParams, [789](#)
 - wds_ProfileIdentifier, [1170](#)
- profileList
 - unpack_wds_SLQSGet3GPPConfigItem_t, [1073](#)
- ProfileName
 - unpack_wds_SLQSGetRuntimeSettings_t, [1079](#)
- ProfileType
 - _GetProfileSettingIn, [58](#)
 - pack_wds_SLQSGetProfileSettings_t, [630](#)
 - PackCreateProfileOut, [641](#)
 - unpack_wds_SLQSGetProfileSettings_t, [1077](#)
- profileType
 - pack_wds_SLQSDeleteProfile_t, [628](#)
 - pack_wds_SetDefaultProfile_t, [623](#)
 - ProfileIdentifier, [675](#)
 - SLQSDeleteProfileParams, [789](#)
 - wds_DHCPPProfileIdTlv, [1162](#)
 - wds_DHCPv4ProfileId, [1164](#)
 - wds_ProfileIdentifier, [1170](#)
 - WdsDHCPv4ProfileId, [1184](#)

- wdsDhcpv4ProfileId, 1184
- profiletype
 - pack_wds_GetDefaultProfile_t, 620
- Protocol
 - unpack_nas_GetCDMANetworkParameters_t, 982
- protocolSubtypeElement, 675
 - AccessMac, 676
 - AuthProt, 676
 - ControlMac, 676
 - EncryptProt, 676
 - ForwardMac, 676
 - IdleState, 676
 - KeyExchange, 676
 - MultDisc, 676
 - PhysicalLayer, 676
 - ReverseMac, 676
 - SecProt, 677
 - VirtStream, 677
- ProvisionStatus
 - CLIPResp, 164
 - CLIRResp, 164
 - CNAPResp, 166
 - COLPResp, 167
 - COLRResp, 168
- psAttachState
 - NASServingSystemInfo, 523
 - nas_servSystem, 471
 - servSystem, 752
 - ServingSystemInfo, 751
- psBarStatus
 - callBarStatus, 127
 - CallBarringSysInfo, 126
 - nas_CallBarringSysInfo, 407
 - nas_callBarStatus, 408
- psState
 - CommInfo, 169
 - nas_CommInfo, 417
- psc
 - nas_UMTSInfo, 484
 - nas_WCDMASysInfo, 494
 - nas_wcdmaCellInfo, 488
 - UMTSInfo, 915
 - WCDMASysInfo, 1158
 - wcdmaCellInfo, 1147
- pscValid
 - nas_WCDMASysInfo, 494
 - WCDMASysInfo, 1158
- pcscfIPv4Addr
 - PCSCFIPv4ServerAddressList, 644
 - wds_PCSCFIPv4ServerAddressList, 1170
- psetting
 - unpack_wds_GetAutoconnect_t, 1061
- psid
 - pack_wds_SLQSSStopDataSession_t, 638
 - unpack_wds_SLQSSStartDataSession_t, 1084
- puk1Retries
 - appStats, 108
 - appStatus, 111
- uim_appStatus, 877
- puk2Retries
 - appStats, 108
 - appStatus, 111
 - uim_appStatus, 877
- pukLen
 - uim_unblockUIMPIN, 889
 - unblockUIMPIN, 924
- pukVal
 - uim_unblockUIMPIN, 889
 - unblockUIMPIN, 924
- pukValue
 - pack_dms_UIMUnblockPIN_t, 564
- pv4sessionId
 - WdsIpAddressInfoReq, 1185
- pv6sessionId
 - WdsIpAddressInfoReq, 1185
- pwrDenialTime
 - lineCtrlInfo, 351
- QCWWAN2kConnect
 - qaGobiApiDcs.h, 1407
- QCWWAN2kEnumerateDevices
 - qaGobiApiDcs.h, 1407
- QCWWAN2kGetConnectedDeviceID
 - qaGobiApiDcs.h, 1408
- QCWWANConnect
 - qaGobiApiDcs.h, 1408
- QCWWANDisconnect
 - qaGobiApiDcs.h, 1409
- QCWWANEnumerateDevices
 - qaGobiApiDcs.h, 1409
- QCI
 - LibPackQosClassID, 345
 - QosClassID, 715
- QFlowState
 - unpack_qos_QosFlowInfo_t, 1023
- QLIC
 - DeviceConfigDetail, 206
- QMI_CAN_COMMON_EVENT_TLV_NUMBER
 - qaCbkCatEventReportInd.h, 1311
- QMI_ETWS_MAX_PAYLOAD_LENGTH
 - qaGobiApiCbk.h, 1327
- QMI_MAX_CAT_EVENT_DATA_LENGTH
 - qaCbkCatEventReportInd.h, 1311
- QMI_MAX_VOICE_NUMBER_LENGTH
 - qaGobiApiCbk.h, 1327
- QMI_NAS_MAX_INSTANCES
 - qaNasPerformNetworkScan.h, 1741
- QMI_NAS_NETSTATUS_MASK
 - qaNasPerformNetworkScan.h, 1741
- QMI_NO_LTE_FW_SUPPORT
 - SwiDataTypes.h, 1768
- QMI_SAR_RF_STATE_1
 - qaGobiApiSar.h, 1573
- QMI_SAR_RF_STATE_2
 - qaGobiApiSar.h, 1574
- QMI_SAR_RF_STATE_3
 - qaGobiApiSar.h, 1574

- QMI_SAR_RF_STATE_4
 - qaGobiApiSar.h, [1574](#)
- QMI_SAR_RF_STATE_5
 - qaGobiApiSar.h, [1574](#)
- QMI_SAR_RF_STATE_6
 - qaGobiApiSar.h, [1574](#)
- QMI_SAR_RF_STATE_7
 - qaGobiApiSar.h, [1574](#)
- QMI_SAR_RF_STATE_8
 - qaGobiApiSar.h, [1574](#)
- QMI_SAR_RF_STATE_DEFAULT
 - qaGobiApiSar.h, [1573](#)
- QMI_SWIOMA_DM_CONFIG
 - qaCbkSwiOmaDmEventReportInd.h, [1312](#)
- QMI_SWIOMA_DM_FOTA
 - qaCbkSwiOmaDmEventReportInd.h, [1312](#)
- QMI_SWIOMA_DM_NOT
 - qaCbkSwiOmaDmEventReportInd.h, [1312](#)
- QMI_TLV_PLACEHOLDER
 - SwiDataTypes.h, [1768](#)
- QMI_WDS_CURRENT_CALL_DB_MASK
 - qaGobiApiWds.h, [1702](#)
- QMI_WDS_LAST_CALL_DB_MASK
 - qaGobiApiWds.h, [1702](#)
- QMI_WMS_MAX_PAYLOAD_LENGTH
 - qaGobiApiCbk.h, [1327](#)
- qaCbkCatEventReportInd.h, [1310](#)
 - eQMI_CAT_EVENT_REPORT_IND_TLV_LENGTH, [1311](#)
 - eQMI_CAT_EVENT_REPORT_IND_TLV, [1311](#)
 - eTLV_CBK_ALPHA_IDENTIFIER, [1311](#)
 - eTLV_CBK_DISPLAY_TEXT, [1311](#)
 - eTLV_CBK_END_PROACTIVE_SESSION, [1311](#)
 - eTLV_CBK_GET_IN_KEY, [1311](#)
 - eTLV_CBK_GET_INPUT, [1311](#)
 - eTLV_CBK_LANGUAGE_NOTIFICATION, [1311](#)
 - eTLV_CBK_REFRESH, [1311](#)
 - eTLV_CBK_SELECT_ITEM, [1311](#)
 - eTLV_CBK_SETUP_EVENT_LIST, [1311](#)
 - eTLV_CBK_SETUP_IDLE_MODE_TEXT, [1311](#)
 - eTLV_CBK_SETUP_MENU, [1311](#)
 - eTLV_END_PROACTIVE_SESSION_LENGTH, [1311](#)
 - eTLV_REFRESH_LENGTH, [1311](#)
 - eTLV_SETUP_EVENT_LIST_LENGTH, [1311](#)
 - QMI_CAN_COMMON_EVENT_TLV_NUMBER, [1311](#)
 - QMI_MAX_CAT_EVENT_DATA_LENGTH, [1311](#)
 - UpkQmiCbkCatEventReportInd, [1311](#)
- qaCbkSwiOmaDmEventReportInd.h, [1311](#)
 - eQMI_SWIOMA_DM_EVENT_REPORT_IND, [1312](#)
 - eTLV_IND_OMA_DM_CONFIG, [1312](#)
 - eTLV_IND_OMA_DM_FOTA, [1312](#)
 - eTLV_IND_OMA_DM_NOT, [1312](#)
 - QMI_SWIOMA_DM_CONFIG, [1312](#)
 - QMI_SWIOMA_DM_FOTA, [1312](#)
 - QMI_SWIOMA_DM_NOT, [1312](#)
 - UpkQmiCbkSwiOmaDmEventReportInd, [1312](#)
 - UpkQmiCbkSwiOmaDmEventReportIndExt, [1312](#)
- qaGobiApiAudio.h, [1312](#)
 - SLQSGetAudioPathConfig, [1313](#)
 - SLQSGetAudioProfile, [1314](#)
 - SLQSGetAudioVolTLBConfig, [1314](#)
 - SLQSSetAudioPathConfig, [1315](#)
 - SLQSSetAudioProfile, [1315](#)
 - SLQSSetAudioVolTLBConfig, [1316](#)
- qaGobiApiCat.h, [1317](#)
 - CATSendEnvelopeCommand, [1317](#)
 - CATSendTerminalResponse, [1317](#)
- qaGobiApiCbk.h, [1318](#)
 - accelAcceptReady, [1328](#)
 - accelTempAcceptReady, [1328](#)
 - CBK_DISABLE_EVENT, [1326](#)
 - CBK_ENABLE_EVENT, [1326](#)
 - CBK_NOCHANGE, [1326](#)
 - DEREGISTER_EVENT, [1326](#)
 - DEREGISTER_SRV, [1326](#)
 - DEVICE_STATE_BOOT, [1364](#)
 - DEVICE_STATE_DISCONNECTED, [1364](#)
 - DEVICE_STATE_READY, [1364](#)
 - DHCP_MAX_NUM_OPTIONS, [1326](#)
 - DHCP_OPTION_DATA_BUF_SIZE, [1326](#)
 - device_state_enum, [1364](#)
 - eDevState, [1329](#)
 - eQA_QMI_SVC_NAS, [1364](#)
 - eQA_QMI_SVC_NA, [1364](#)
 - eQA_QMI_SVC_WDS, [1364](#)
 - eQaQMIService, [1364](#)
 - eSMSEventType, [1329](#)
 - EVENT_MASK_CARD, [1326](#)
 - EVENT_MASK_DEREGISTER_ALL, [1326](#)
 - EVENT_MASK_PHY_SLOT_STATUS, [1326](#)
 - FIRST_INSTANCE, [1326](#)
 - gpsTime, [1329](#)
 - gyroAcceptReady, [1330](#)
 - gyroTempAcceptReady, [1330](#)
 - INVALID_INSTACNE, [1326](#)
 - IPV4, [1326](#)
 - IPV4V6, [1327](#)
 - IPV6, [1327](#)
 - iSLQSSetDUNCallInfoCallback, [1365](#)
 - iSLQSSetSignalStrengthsCallback, [1365](#)
 - iSLQSSetWdsFirstInstEventCallback, [1365](#)
 - iSLQSSetWdsSecondInstEventCallback, [1365](#)
 - iSLQSSetWdsThirdInstEventCallback, [1365](#)
 - iSLQSSetWdsXferStatsFirstInstCallback, [1365](#)
 - iSLQSSetWdsXferStatsSecondInstCallback, [1365](#)
 - iSetCATEventCallback, [1365](#)
 - iSetSignalStrengthCallback, [1365](#)
 - LOC_EVENT_MASK_ENG_STATE, [1327](#)
 - LOC_EVENT_MASK_GNSS_SV_INFO, [1327](#)
 - LOC_EVENT_MASK_INJECT_TIME, [1327](#)
 - LOC_EVENT_MASK_SENSOR_STREAM, [1327](#)
 - LOC_EVENT_MASK_TIME_SYNC, [1327](#)
 - LOC_EVENT_POSITION_REPORT, [1327](#)

- LteNasReleaseInfo, [1330](#)
- MAX_MITIGATION_DEV_ID_LEN, [1327](#)
- MAX_NO_OF_APPLICATIONS, [1327](#)
- MAX_NO_OF_CALLS, [1327](#)
- MAX_NO_OF_FILES, [1327](#)
- MAX_NO_OF_SLOTS, [1327](#)
- MAX_NO_OF_UUSINFO, [1327](#)
- MAX_PATH_LENGTH, [1327](#)
- MAX_RADIO_INTERFACE_LIST, [1327](#)
- MAXUSSDLENGTH, [1327](#)
- modemTempNotification, [1331](#)
- NAS_SRV, [1327](#)
- NUM_OF_SET, [1327](#)
- PDS_SRV, [1327](#)
- packetSrvStatus, [1331](#)
- precisionDilution, [1332](#)
- QMI_ETWS_MAX_PAYLOAD_LENGTH, [1327](#)
- QMI_MAX_VOICE_NUMBER_LENGTH, [1327](#)
- QMI_WMS_MAX_PAYLOAD_LENGTH, [1327](#)
- REGISTER_EVENT, [1328](#)
- REGISTER_SRV, [1328](#)
- ResetInfoNotification, [1333](#)
- SECOND_INSTANCE, [1328](#)
- SIGSTRENGTH_THRESHOLD_ARR_SZ, [1328](#)
- SLQSNasNetworkTimeCallBack, [1381](#)
- SLQSNasSigInfo2CallBack, [1381](#)
- SLQSNasSigInfoCallBack, [1382](#)
- SLQSNasSwiOTAMessageCallback, [1383](#)
- SLQSNasSysInfoCallBack, [1383](#)
- SLQSNasTimerCallback, [1384](#)
- SLQSSetBandPreferenceCbk, [1384](#)
- SLQSSetDHCPv4ClientLeaseStatusCallback, [1385](#)
- SLQSSetDUNCallInfoCallback, [1385](#)
- SLQSSetDataSystemStatusCallback, [1384](#)
- SLQSSetIMSAPdpStatusCallback, [1386](#)
- SLQSSetIMSARatStatusCallback, [1386](#)
- SLQSSetIMSARegStatusCallback, [1387](#)
- SLQSSetIMSASvcStatusCallback, [1387](#)
- SLQSSetIMSSMSCConfigCallback, [1388](#)
- SLQSSetIMSUserConfigCallback, [1388](#)
- SLQSSetIMSVoIPConfigCallback, [1389](#)
- SLQSSetLocInjectPositionCallback, [1389](#)
- SLQSSetLocInjectUTCTimeCallback, [1390](#)
- SLQSSetModemTempCallback, [1390](#)
- SLQSSetPacketSrvStatusCallback, [1390](#)
- SLQSSetQosEventCallback, [1391](#)
- SLQSSetQosNWStatusCallback, [1391](#)
- SLQSSetQosPriEventCallback, [1392](#)
- SLQSSetQosStatusCallback, [1392](#)
- SLQSSetRegMgrConfigCallback, [1393](#)
- SLQSSetSDKTerminatedCallback, [1393](#)
- SLQSSetSIPConfigCallback, [1395](#)
- SLQSSetSMSEventCallback, [1396](#)
- SLQSSetServingSystemCallback, [1394](#)
- SLQSSetSessionStateCallback, [1394](#)
- SLQSSetSignalStrengthsCallback, [1395](#)
- SLQSSetSwiGetResetInfoCallback, [1396](#)
- SLQSSetSwiHDRPersCallback, [1396](#)
- SLQSSetSysSelectionPrefCallBack, [1397](#)
- SLQSSetTransLayerInfoCallback, [1397](#)
- SLQSSetTransNWRegInfoCallback, [1398](#)
- SLQSSetWdsEventCallback, [1398](#)
- SLQSSetWdsTransferStatisticCallback, [1399](#)
- SLQSTmdMitigationLvlRptCallback, [1400](#)
- SLQSUIMSetRefreshCallBack, [1400](#)
- SLQSUIMSetStatusChangeCallBack, [1400](#)
- SLQSVoiceInfoRecCallback, [1401](#)
- SLQSVoiceSetAllCallStatusCallBack, [1401](#)
- SLQSVoiceSetDTMFEventCallBack, [1402](#)
- SLQSVoiceSetOTASPStatusCallBack, [1402](#)
- SLQSVoiceSetPrivacyChangeCallBack, [1403](#)
- SLQSVoiceSetSUPSCallBack, [1403](#)
- SLQSVoiceSetSUPSNotificationCallback, [1404](#)
- SLQSWmsAsyncRawSendCallBack, [1404](#)
- SLQSWmsMemoryFullCallBack, [1405](#)
- SLQSWmsMessageWaitingCallBack, [1405](#)
- SMS_EVENT_ETWS_PLMN, [1365](#)
- SMS_EVENT_ETWS, [1365](#)
- SMS_EVENT_MESSAGE_MODE, [1365](#)
- SMS_EVENT_MT_MESSAGE, [1365](#)
- SMS_EVENT_SMS_ON_IMS, [1365](#)
- SMS_EVENT_SMSC_ADDRESS, [1365](#)
- SMS_EVENT_TRANSFER_ROUTE_MT_MESSAGE, [1365](#)
- SMSAsyncRawSend, [1334](#)
- SMSCAddressInfo, [1335](#)
- SMSEtwsMessageInfo, [1335](#)
- SMSEtwsPlmnInfo, [1336](#)
- SMSEventInfo, [1336](#)
- SMSEventType, [1364](#)
- SMSMTMessageInfo, [1337](#)
- SMSMessageModelInfo, [1337](#)
- SMSONIMSInfo, [1337](#)
- SMSTransferRouteMTMessageInfo, [1338](#)
- sensorDataUsage, [1333](#)
- sessionInformation, [1334](#)
- sessionInformationExt, [1334](#)
- SetActivationStatusCallback, [1365](#)
- SetCATEventCallback, [1366](#)
- SetDataCapabilitiesCallback, [1367](#)
- SetDeviceStateChangeCbk, [1367](#)
- SetFwDldCompletionCbk, [1368](#)
- SetGPSCallback, [1368](#)
- SetLURRejectCallback, [1372](#)
- SetLocBestAvailPosCallback, [1368](#)
- SetLocCradleMountCallback, [1369](#)
- SetLocDeleteAssistDataCallback, [1369](#)
- SetLocEngineStateCallback, [1369](#)
- SetLocEventPositionCallback, [1370](#)
- SetLocEventTimeSyncCallback, [1370](#)
- SetLocGnssSvInfoCallback, [1370](#)
- SetLocInjectSensorDataCallback, [1371](#)
- SetLocInjectTimeCallback, [1371](#)
- SetLocOpModeCallback, [1371](#)
- SetLocSensorStreamingCallback, [1372](#)

- SetLocSetExtPowerConfigCallback, 1372
- SetMobileIPStatusCallback, 1373
- SetNMEACallback, 1375
- SetNasLTECphyCalIndCallback, 1373
- SetNetChangeCbk, 1374
- SetNewSMSCallback, 1374
- SetOMADMStateCallback, 1375
- SetPDSSStateCallback, 1376
- SetPowerCallback, 1376
- SetRFInfoCallback, 1376
- SetRMTTransferStatisticsCallback, 1377
- SetRankIndicatorCallback, 1376
- SetRoamingIndicatorCallback, 1377
- SetSLQSOMADMAAlertCallback, 1378
- SetSLQSOMADMAAlertCallbackExt, 1379
- SetSignalStrengthCallback, 1378
- SetUSSDNoWaitIndicationCallback, 1380
- SetUSSDNotificationCallback, 1380
- SetUSSDReleaseCallback, 1381
- SetUimSlotStatusChangeCallback, 1379
- svUsedforFix, 1338
- SwiOTAMsg, 1339
- tFNASwiLTECphyCallInfo, 1340
- tFNASwiOTAMsg, 1340
- tFNAActivationStatus, 1339
- tFNAAllCallStatus, 1339
- tFNAsyncRawSend, 1340
- tFNBandPreference, 1340
- tFNBstAvailPos, 1341
- tFNCATEvent, 1342
- tFNCbkUimSlotStatusChangeInd, 1342
- tFNDHCPv4ClientLeaseStatus, 1344
- tFNDTMFEvent, 1344
- tFNDUNCallInfo, 1344
- tFNDDataCapabilities, 1342
- tFNDDataSysStatus, 1343
- tFNDelAssistData, 1343
- tFNDeviceStateChange, 1343
- tFNEventPosition, 1344
- tFNFwdDIdCompletion, 1344
- tFNGnssSvInfo, 1345
- tFNHDRPersonality, 1345
- tFNImRegMgrConfig, 1346
- tFNImSIPConfig, 1346
- tFNImSMSConfig, 1347
- tFNImUserConfig, 1347
- tFNImVoIPConfig, 1347
- tFNImsaPdpStatus, 1345
- tFNImsaRatStatus, 1346
- tFNImsaRegStatus, 1346
- tFNImsaSvcStatus, 1346
- tFNInfoRec, 1347
- tFNInjectPosition, 1348
- tFNInjectSensorData, 1348
- tFNInjectTimeStatus, 1348
- tFNInjectUTCTime, 1348
- tFNLURReject, 1348
- tFNMemoryFull, 1349
- tFNMessageWaiting, 1350
- tFNMTiLvlRpt, 1350
- tFNMobileIPStatus, 1350
- tFNModemTempInfo, 1350
- tFNNasTimer, 1350
- tFNNet, 1350
- tFNNetworkTime, 1351
- tFNNewGPS, 1351
- tFNNewNMEA, 1352
- tFNNewRMTTransferStatistics, 1352
- tFNNewSMS, 1353
- tFNOMADMState, 1353
- tFNOTASPStatus, 1354
- tFNOpMode, 1354
- tFNPDSState, 1354
- tFNPacketSrvState, 1354
- tFNPower, 1355
- tFNPrivacyChange, 1355
- tFNQosNWStatus, 1355
- tFNQosPriEvent, 1355
- tFNQosStatus, 1356
- tFNRInfo, 1357
- tFNRankIndicator, 1357
- tFNResetInfo, 1357
- tFNRoamingIndicator, 1358
- tFNSDKTerminated, 1358
- tFNSLQSOMADMAAlert, 1359
- tFNSLQSQOSEvent, 1360
- tFNSLQSSessionState, 1360
- tFNSLQSSignalStrengths, 1360
- tFNSLQSWDSEvent, 1360
- tFNSMSEvents, 1361
- tFNSUPSInfo, 1361
- tFNSUPSNotification, 1361
- tFNSensorStreaming, 1358
- tFNServingSystem, 1358
- tFNSetCradleMount, 1359
- tFNSetEngineState, 1359
- tFNSetEventTimeSync, 1359
- tFNSetExtPowerConfig, 1359
- tFNSigInfo, 1359
- tFNSignalStrength, 1359
- tFNSysInfo, 1361
- tFNSysSelectionPref, 1361
- tFNUIMRefresh, 1362
- tFNUIMStatusChangeInfo, 1362
- tFNUSSDNoWaitIndication, 1363
- tFNUSSDNotification, 1363
- tFNUSSDRelease, 1363
- tFNtransLayerInfo, 1362
- tFNtransNWRegInfo, 1362
- THIRD_INSTANCE, 1328
- transLayerNotification, 1363
- transNWRegInfoNotification, 1364
- USSD_DCS_8BIT, 1328
- USSD_DCS_ASCII, 1328
- USSD_DCS_UCS2, 1328
- VOICE_SRV, 1328

- WDS_SRV, 1328
- qaGobiApiDcs.h, 1406
 - LEN, 1406
 - PORTNAM_LEN, 1406
 - QCWWAN2kConnect, 1407
 - QCWWAN2kEnumerateDevices, 1407
 - QCWWAN2kGetConnectedDeviceID, 1408
 - QCWWANConnect, 1408
 - QCWWANDisconnect, 1409
 - QCWWANEnumerateDevices, 1409
 - SLQSGetDeviceMode, 1410
 - SLQSGetNetStatistic, 1411
 - SLQSGetUsbPortNames, 1411
 - SLQSKillSDKProcess, 1412
 - SLQSSetLoggingMask, 1412
 - SLQSSStart, 1413
 - SLQSSStart_AVAgent, 1413
 - SLQSSStartSrv, 1414
 - SetSDKImagePath, 1410
- qaGobiApiDms.h, 1414
 - ActivateAutomatic, 1424
 - custFeaturesInfo, 1417
 - custFeaturesSetting, 1419
 - dmsCurrentPRLInfo, 1420
 - ERIFileparams, 1420
 - GetActivationState, 1424
 - GetDeviceCapabilities, 1425
 - GetFirmwareRevision, 1426
 - GetFirmwareRevisions, 1427
 - GetHardwareRevision, 1427
 - GetIMSI, 1428
 - GetManufacturer, 1428
 - GetModelID, 1429
 - GetNetworkTime, 1429
 - GetOfflineReason, 1430
 - GetPRLVersion, 1431
 - GetPower, 1431
 - GetSerialNumbers, 1432
 - GetVoiceNumber, 1433
 - MAX_CUST_ID_LEN, 1417
 - MAX_CUST_VALUE_LEN, 1417
 - MAX_DYING_GASP_CFG_SMS_CONTENT_LEN, 1417
 - MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH, 1417
 - MAX_FSN_LENGTH, 1417
 - ResetToFactoryDefaults, 1433
 - SLQSDmsSwiGetResetInfo, 1434
 - SLQSDmsSwiIndicationRegister, 1434
 - SLQSGetBandCapabilities, 1435
 - SLQSGetBandCapability, 1435
 - SLQSGetCurrentPRLInfo, 1437
 - SLQSGetCustFeatures, 1437
 - SLQSGetCustFeaturesV2, 1438
 - SLQSGetERIFile, 1438
 - SLQSGetSerialNumbers, 1438
 - SLQSSetCustFeatures, 1439
 - SLQSSetCustFeaturesV2, 1439
 - SLQSSwiClearDyingGaspStatistics, 1440
 - SLQSSwiGetCrashAction, 1440
 - SLQSSwiGetCrashInfo, 1440
 - SLQSSwiGetDyingGaspCfg, 1441
 - SLQSSwiGetDyingGaspStatistics, 1441
 - SLQSSwiGetFSN, 1441
 - SLQSSwiGetFwUpdateStatus, 1442
 - SLQSSwiGetHostDevInfo, 1442
 - SLQSSwiGetHostDevInfoParams, 1421
 - SLQSSwiGetOSInfo, 1443
 - SLQSSwiGetOSInfoParams, 1422
 - SLQSSwiGetSerialNoExt, 1443
 - SLQSSwiGetSerialNoExtParams, 1422
 - SLQSSwiGetUSBComp, 1444
 - SLQSSwiSetCrashAction, 1444
 - SLQSSwiSetDyingGaspCfg, 1445
 - SLQSSwiSetHostDevInfo, 1445
 - SLQSSwiSetHostDevInfoParams, 1423
 - SLQSSwiSetOSInfo, 1446
 - SLQSSwiSetOSInfoParams, 1423
 - SLQSSwiSetUSBComp, 1446
 - SLQSUIMGetState, 1447
 - serialNumbersInfo, 1421
 - SetPower, 1434
 - UIMChangePIN, 1447
 - UIMGetControlKeyStatus, 1448
 - UIMGetICCID, 1449
 - UIMGetPINStatus, 1450
 - UIMSetControlKeyProtection, 1451
 - UIMSetPINProtection, 1451
 - UIMUnblockControlKey, 1452
 - UIMUnblockPIN, 1453
 - UIMVerifyPIN, 1454
 - ValidateSPC, 1455
- qaGobiApiFms.h, 1455
 - BUILD_ID_LEN, 1459
 - BUILD_ID_MAX_LEN, 1459
 - DEVICE_OFFLINE, 1459
 - DEVICE_RESET, 1459
 - DEVICE_SHUTDOWN, 1459
 - DeleteStoredImage, 1462
 - DownloadToSlot, 1462
 - eGOBI_DEV_SERIES_9X15, 1460
 - eGOBI_DEV_SERIES_9X30, 1460
 - eGOBI_DEV_SERIES_G3K, 1460
 - eGOBI_DEV_SERIES_NON_GOBI, 1460
 - eGOBI_DEV_SERIES_SIERRA_GOBI, 1460
 - eGOBI_DEV_SERIES_UNKNOWN, 1460
 - eGOBI_IMG_CAR_3, 1461
 - eGOBI_IMG_CAR_AERIS, 1461
 - eGOBI_IMG_CAR_ALLTEL, 1460
 - eGOBI_IMG_CAR_AMX_TELCEL, 1461
 - eGOBI_IMG_CAR_ATT, 1461
 - eGOBI_IMG_CAR_BELL, 1460
 - eGOBI_IMG_CAR_BHARTI, 1461
 - eGOBI_IMG_CAR_BRASIL_VIVO, 1461
 - eGOBI_IMG_CAR_CHINA_MOBILE, 1461
 - eGOBI_IMG_CAR_CHINA_TELECOM, 1461

- eGOBI_IMG_CAR_CHINA_UNICOM, 1460
- eGOBI_IMG_CAR_EMOBILE, 1461
- eGOBI_IMG_CAR_FACTORY, 1460
- eGOBI_IMG_CAR_GENERIC_CDMA, 1461
- eGOBI_IMG_CAR_GENERIC, 1460
- eGOBI_IMG_CAR_IUSACELL, 1461
- eGOBI_IMG_CAR_KDDI, 1461
- eGOBI_IMG_CAR_KT_FREETEL, 1461
- eGOBI_IMG_CAR_LEAP, 1461
- eGOBI_IMG_CAR_METROPCS, 1461
- eGOBI_IMG_CAR_NETCOM, 1461
- eGOBI_IMG_CAR_NORF, 1460
- eGOBI_IMG_CAR_NTT_DOCOMO, 1461
- eGOBI_IMG_CAR_O2, 1461
- eGOBI_IMG_CAR_OMH, 1461
- eGOBI_IMG_CAR_ORANGE, 1461
- eGOBI_IMG_CAR_RELIANCE1, 1461
- eGOBI_IMG_CAR_RELIANCE2, 1461
- eGOBI_IMG_CAR_ROGERS, 1461
- eGOBI_IMG_CAR_SFR, 1461
- eGOBI_IMG_CAR_SINGTEL_OPTUS, 1461
- eGOBI_IMG_CAR_SK_TELCOM1, 1460
- eGOBI_IMG_CAR_SK_TELCOM2, 1461
- eGOBI_IMG_CAR_SOFTBANK, 1461
- eGOBI_IMG_CAR_SPRINT, 1460
- eGOBI_IMG_CAR_SWISSCOM, 1461
- eGOBI_IMG_CAR_TATA, 1461
- eGOBI_IMG_CAR_TELCOM_ITALIA, 1461
- eGOBI_IMG_CAR_TELCOM_NZ, 1460
- eGOBI_IMG_CAR_TELEFONICA, 1461
- eGOBI_IMG_CAR_TELENOR, 1461
- eGOBI_IMG_CAR_TELIASONERA, 1461
- eGOBI_IMG_CAR_TELSTRA1, 1460
- eGOBI_IMG_CAR_TELSTRA2, 1461
- eGOBI_IMG_CAR_TELUS, 1460
- eGOBI_IMG_CAR_TMOBILE, 1461
- eGOBI_IMG_CAR_US, 1460
- eGOBI_IMG_CAR_VERIZON, 1460
- eGOBI_IMG_CAR_VODAFONE, 1461
- eGOBI_IMG_GPS_ASSISTED, 1461
- eGOBI_IMG_GPS_NO_XTRA, 1461
- eGOBI_IMG_GPS_NONE, 1461
- eGOBI_IMG_GPS_STAND_ALONE, 1461
- eGOBI_IMG_REG_ASIA, 1462
- eGOBI_IMG_REG_AUS, 1462
- eGOBI_IMG_REG_EU, 1462
- eGOBI_IMG_REG_GLOBAL, 1462
- eGOBI_IMG_REG_LA, 1462
- eGOBI_IMG_REG_NA, 1462
- eGOBI_IMG_TECH_CDMA, 1462
- eGOBI_IMG_TECH_UMTS, 1462
- eGetDeviceSeries, 1463
- eGobi_DEV_SERIES_MC83, 1460
- eGobiDeviceSeries, 1460
- eGobiImageCarrier, 1460
- eGobiImageGPS, 1461
- eGobiImageRegion, 1461
- eGobiImageTech, 1462
- FIRMWARE_UPDATE_FAIL, 1459
- FIRMWARE_UPDATE_SUCCESS, 1459
- FIRMWARE_UPGRADE_SUCCESS, 1459
- G3K_FIRMWARE_DOWNLOAD, 1459
- GOBI_LISTENTRIES_MAX, 1459
- GOBI_MBN_BUILD_ID_STR_LEN, 1459
- GOBI_MBN_IMG_ID_STR_LEN, 1459
- GOBI_SET_IMG_PREF_RSPLEN, 1459
- GetImageStore, 1464
- GetImagesPreference, 1463
- GetStoredImages, 1464
- IMG_ID_LEN, 1459
- IMGDETAILS_LEN, 1459
- MAX_IMAGE_IDE_ELEMENTS, 1459
- PRI_UPDATE_FAIL, 1459
- SLQSDownloadFirmwareToSlot, 1466
- SLQSFWINFO_APPVERSION_SZ, 1459
- SLQSFWINFO_BOOTVERSION_SZ, 1459
- SLQSFWINFO_CARRIER_SZ, 1459
- SLQSFWINFO_CUR_CARR_NAME, 1459
- SLQSFWINFO_CUR_CARR_REV, 1459
- SLQSFWINFO_MODELID_SZ, 1459
- SLQSFWINFO_PACKAGEID_SZ, 1460
- SLQSFWINFO_PRIVERSION_SZ, 1460
- SLQSFWINFO_SKU_SZ, 1460
- SLQSGetBootVersionNumber, 1467
- SLQSGetFirmwareInfo, 1467
- SLQSGetImageInfo, 1468
- SLQSGetImageInfo_9x15, 1468
- SLQSGetImageInfoMC77xx, 1469
- SLQSGetImageInfoMC83xx, 1470
- SLQSGetValidFwPriCombinations, 1470
- SLQSIIsSpkgFormatRequired, 1471
- SLQSSetCrashStateCheckIgnore, 1471
- SLQSSetSIMBasedImageSwitching, 1472
- SLQSSetSpkgFormatRequired, 1472
- SLQSSwiGetAllCarrierImages, 1472
- SLQSSwiGetFirmwareCurr, 1473
- SLQSUpgradeFirmware9x15, 1473
- SPKG_FIRMWARE_DOWNLOAD, 1460
- SetImagesPreference, 1465
- UNIQUE_ID_LEN, 1460
- upgrade_mc77xx_fw, 1475
- UpgradeFirmware2k, 1475
- qaGobiApilms.h, 1476
 - SLQSGetIMSSMSConfig, 1477
 - SLQSGetIMSUserConfig, 1477
 - SLQSGetIMSVoIPConfig, 1478
 - SLQSGetRegMgrConfig, 1478
 - SLQSGetSIPConfig, 1479
 - SLQSImsConfigIndicationRegister, 1479
 - SLQSSetIMSSMSConfig, 1480
 - SLQSSetIMSUserConfig, 1480
 - SLQSSetIMSVoIPConfig, 1481
 - SLQSSetRegMgrConfig, 1482
 - SLQSSetSIPConfig, 1482
- qaGobiApilmsa.h, 1483
 - SLQSGetIMSARegStatus, 1483

- SLQSGetIMSAServiceStatus, 1484
- SLQSGetIMSASupportedFields, 1484
- SLQSGetIMSASupportedMsg, 1485
- SLQSRegisterMSAIndication, 1486
- qaGobiApiLoc.h, 1486
- MAX_SENSOR_DATA_LEN, 1487
- MAX_TEMP_DATA_LEN, 1487
- SLQSLOCDeAssData, 1487
- SLQSLOCEventRegister, 1488
- SLQSLOCGetBestAvailPos, 1488
- SLQSLOCInjectPosition, 1489
- SLQSLOCInjectSensorData, 1489
- SLQSLOCInjectUTCTime, 1490
- SLQSLOCSetCradleMountConfig, 1490
- SLQSLOCSetExtPowerState, 1491
- SLQSLOCSetOpMode, 1491
- SLQSLOCStart, 1492
- SLQSLOCStop, 1492
- SwiLocGetAutoStart, 1493
- SwiLocSetAutoStart, 1493
- qaGobiApiNas.h, 1493
- _NAMS_RADIO_IF_TECHNOLOGY_, 1511
- eNAS_LTE_CPHY_CA_BW_NRB_100, 1512
- eNAS_LTE_CPHY_CA_BW_NRB_15, 1512
- eNAS_LTE_CPHY_CA_BW_NRB_25, 1512
- eNAS_LTE_CPHY_CA_BW_NRB_50, 1512
- eNAS_LTE_CPHY_CA_BW_NRB_6, 1512
- eNAS_LTE_CPHY_CA_BW_NRB_75, 1512
- eNAS_LTE_CPHY_SCELL_STATE_CONFIGU↵
RED_ACTIVATED, 1512
- eNAS_LTE_CPHY_SCELL_STATE_CONFIGU↵
RED_DEACTIVATED, 1512
- eNAS_LTE_CPHY_SCELL_STATE_DECONFI↵
GURED, 1512
- eNAS_RADIO_IF_GSM, 1511
- eNAS_RADIO_IF_LTE, 1511
- eNAS_RADIO_IF_TDSCDMA, 1511
- eNAS_RADIO_IF_UMTS, 1511
- eSYS_SRV_DOMAIN_CAMPED, 1511
- eSYS_SRV_DOMAIN_CS_ONLY, 1511
- eSYS_SRV_DOMAIN_CS_PS, 1511
- eSYS_SRV_DOMAIN_NO_SRV, 1511
- eSYS_SRV_DOMAIN_PS_ONLY, 1511
- eSYS_SRV_DOMAIN_UNKNOWN, 1511
- eSYS_SRV_DOMAIN, 1511
- GetACCOLC, 1512
- GetANAAAAuthenticationStatus, 1512
- GetCDMANetworkParameters, 1513
- GetHomeNetwork, 1515
- GetHomeNetwork3GPP2, 1516
- GetNetworkPreference, 1517
- GetRFInfo, 1518
- GetServingNetwork, 1519
- GetServingNetworkCapabilities, 1520
- GetSignalStrengths, 1521
- IMSI_M_S1_LENGTH, 1499
- IMSI_M_S2_LENGTH, 1499
- InitiateDomainAttach, 1522
- InitiateNetworkRegistration, 1522
- MAX_DATA_SRV_CAPABILITIES, 1499
- MAX_DESCRIPTION_LENGTH, 1499
- MAX_PILOT_SETS, 1499
- MAX_SERV_SYSTEM_RADIO_INTERFACES, 1499
- NAM_NAME_LENGTH, 1499
- NAS_LTE_CPHY_CA_BW_NRB, 1511
- NAS_LTE_CPHY_SCELL_STATE, 1512
- NAS_MAX_SCC_RX_INFO_INSTANCES, 1499
- NAS_SIG_INFO_MAX_TDSCDMA_THRESHOL↵
DS_LIST_SIZE, 1499
- NAS_SIG_INFO_MIN_dB_FLOAT_VALUE, 1499
- NAS_SIG_INFO_MIN_dBm_FLOAT_VALUE, 1499
- PLMN_LENGTH, 1499
- PerformNetworkScan, 1523
- SLQS_SS_INFO_LIST_MAX_ELEMENTS, 1499
- SLQS_SYSTEM_ID_SIZE, 1499
- SLQSConfigSigInfo, 1527
- SLQSGetErrorRate, 1527
- SLQSGetNetworkTime, 1527
- SLQSGetOperatorNameData, 1528
- SLQSGetPLMNName, 1528
- SLQSGetServingSystem, 1529
- SLQSGetSignalStrength, 1530
- SLQSGetSysSelectionPref, 1530
- SLQSInitiateNetworkRegistration, 1531
- SLQSNASGetLTECPHYCaInfo, 1533
- SLQSNASSwiGetChannelLock, 1537
- SLQSNASSwiSetChannelLock, 1538
- SLQSNasConfigSigInfo2, 1531
- SLQSNasGet3GPP2Subscription, 1532
- SLQSNasGetCellLocationInfo, 1532
- SLQSNasGetHDRColorCode, 1533
- SLQSNasGetSigInfo, 1533
- SLQSNasGetSysInfo, 1534
- SLQSNasGetTxRxInfo, 1534
- SLQSNasIndicationRegister, 1535
- SLQSNasIndicationRegisterExt, 1536
- SLQSNasIndicationRegisterLTECphyCa, 1536
- SLQSNasSwiIndicationRegister, 1537
- SLQSNasSwiModemStatus, 1538
- SLQSPerformNetworkScan, 1539
- SLQSSetBandPreference, 1539
- SLQSSetSysSelectionPref, 1541
- SLQSSwiGetHDRPersonality, 1541
- SLQSSwiGetHDRProtSubtype, 1542
- SLQSSwiGetHRPDStats, 1542
- SLQSSwiGetLteCQI, 1543
- SLQSSwiGetLteSccRxInfo, 1543
- SLQSSwiNetworkDebug, 1544
- SLQSSwiPSDetach, 1544
- SetACCOLC, 1524
- SetCDMANetworkParameters, 1524
- SetNetworkPreference, 1526
- SlqsNas3GppNetworkRAT, 1499
- slqsNetworkScanInfo, 1500

- sysSelectPrefInfo, [1500](#)
- sysSelectPrefParams, [1505](#)
- UATISIZE, [1499](#)
- qaGobiApiOadm.h, [1545](#)
 - OMADMCancelSession, [1545](#)
 - OMADMGetPendingNIA, [1546](#)
 - OMADMGetSessionInfo, [1546](#)
 - OMADMStartSession, [1547](#)
- qaGobiApiPds.h, [1548](#)
 - DEFAULTBYTEVALUE, [1549](#)
 - DEFAULTLONGVALUE, [1549](#)
 - DEFAULTWORDVALUE, [1549](#)
 - eSetServiceAutomaticTrackingDisable, [1549](#)
 - eSetServiceAutomaticTrackingEnable, [1549](#)
 - ForceXTRADownload, [1550](#)
 - GetPDSDefaults, [1550](#)
 - GetPDSSState, [1551](#)
 - GetPortAutomaticTracking, [1551](#)
 - GetServiceAutomaticTracking, [1552](#)
 - GetXTRAAutomaticDownload, [1552](#)
 - GetXTRANetwork, [1553](#)
 - GetXTRAValidity, [1553](#)
 - PDSInjectTimeReference, [1554](#)
 - ResetPDSData, [1555](#)
 - SLQSGetAGPSConfig, [1559](#)
 - SLQSGetGPSSStateInfo, [1559](#)
 - SLQSPDSDeterminePosition, [1560](#)
 - SLQSPDSInjectAbsoluteTimeReference, [1560](#)
 - SLQSPDSInjectPositionData, [1561](#)
 - SLQSSetAGPSConfig, [1562](#)
 - SLQSSetPositionMethodState, [1562](#)
 - SetPDSDefaults, [1555](#)
 - SetPDSSState, [1556](#)
 - SetPortAutomaticTracking, [1557](#)
 - SetServiceAutomaticTracking, [1557](#)
 - SetXTRAAutomaticDownload, [1558](#)
 - SetXTRANetwork, [1558](#)
 - StartPDSTrackingSessionExt, [1563](#)
 - StopPDSTrackingSession, [1564](#)
- qaGobiApiQos.h, [1564](#)
 - MAX_QOS_FILTER_TLV, [1565](#)
 - MAX_QOS_SPEC_PER_APN, [1565](#)
 - SLQSQosGetFlowStatus, [1565](#)
 - SLQSQosGetGranted, [1566](#)
 - SLQSQosGetNWProf, [1567](#)
 - SLQSQosGetNetworkStatus, [1566](#)
 - SLQSQosModify, [1567](#)
 - SLQSQosRel, [1567](#)
 - SLQSQosReq, [1568](#)
 - SLQSQosReset, [1569](#)
 - SLQSQosResume, [1569](#)
 - SLQSQosSuspend, [1570](#)
 - SLQSQosSwiReadApnExtraParams, [1570](#)
 - SLQSQosSwiReadDataStats, [1571](#)
- qaGobiApiRms.h, [1571](#)
 - GetSMSWake, [1571](#)
 - SetSMSWake, [1572](#)
- qaGobiApiSar.h, [1573](#)
 - eQMISARRFState, [1573](#)
 - QMI_SAR_RF_STATE_1, [1573](#)
 - QMI_SAR_RF_STATE_2, [1574](#)
 - QMI_SAR_RF_STATE_3, [1574](#)
 - QMI_SAR_RF_STATE_4, [1574](#)
 - QMI_SAR_RF_STATE_5, [1574](#)
 - QMI_SAR_RF_STATE_6, [1574](#)
 - QMI_SAR_RF_STATE_7, [1574](#)
 - QMI_SAR_RF_STATE_8, [1574](#)
 - QMI_SAR_RF_STATE_DEFAULT, [1573](#)
 - SLQSGetRfSarState, [1574](#)
 - SLQSSetRfSarState, [1574](#)
- qaGobiApiSms.h, [1575](#)
 - ABSOLUTE_VALIDITY, [1577](#)
 - CONFIG_LEN, [1577](#)
 - getIndicationRegResp, [1577](#)
 - GetSMSCAddress, [1581](#)
 - getTransLayerInfoResp, [1578](#)
 - getTransNWRegInfoResp, [1578](#)
 - MAX_SMS_ROUTES, [1577](#)
 - NUM_OF_SET, [1577](#)
 - qaQmi3GPP2BroadcastCfgInfo, [1579](#)
 - qaQmi3GPPBroadcastCfgInfo, [1579](#)
 - SLQSCDMADecodeMTTextMsg, [1584](#)
 - SLQSCDMAEncodeMOTextMsg, [1584](#)
 - SLQSDeleteSMS, [1585](#)
 - SLQSGetIndicationRegister, [1586](#)
 - SLQSGetMessageWaiting, [1586](#)
 - SLQSGetSMSList, [1588](#)
 - SLQSGetSMS, [1587](#)
 - SLQSGetSmsBroadcastConfig, [1588](#)
 - SLQSGetTransLayerInfo, [1589](#)
 - SLQSGetTransNWRegInfo, [1590](#)
 - SLQSModifySMSStatus, [1590](#)
 - SLQSSendAsyncSMS, [1591](#)
 - SLQSSendLongSMS, [1591](#)
 - SLQSSendSMS, [1592](#)
 - SLQSSetIndicationRegister, [1593](#)
 - SLQSSetSmsBroadcastActivation, [1593](#)
 - SLQSSetSmsBroadcastConfig, [1594](#)
 - SLQSSetSmsStorage, [1595](#)
 - SLQSSmsGetMaxStorageSize, [1595](#)
 - SLQSSmsGetMessageProtocol, [1596](#)
 - SLQSSmsSetRoutes, [1596](#)
 - SLQSSwiGetSMSStorage, [1597](#)
 - SLQSWCDMADecodeLongTextMsg, [1597](#)
 - SLQSWCDMADecodeMTTextMsg, [1598](#)
 - SLQSWCDMAEncodeMOTextMsg, [1598](#)
 - SaveSMS, [1581](#)
 - SendSMS, [1582](#)
 - setIndicationRegReq, [1580](#)
 - SetSMSCAddress, [1583](#)
 - TIME_DATE_BUF, [1577](#)
 - TIME_STAMP_BUF, [1577](#)
 - transLayerInfo, [1580](#)
- qaGobiApiSwi.h, [1599](#)
 - SLQSGetPidof, [1599](#)
 - SLQSGetSdkVersion, [1600](#)

- SLQSSendRawQMI, 1600
- qaGobiApiSwiAudio.h, 1600
 - MAX_LEN_IFACE_TABLE, 1601
 - SLQSGetM2MAVMute, 1602
 - SLQSGetM2MAudioProfile, 1601
 - SLQSGetM2MAudioVolume, 1602
 - SLQSGetM2MSpkrGain, 1603
 - SLQSSetM2MAVMute, 1605
 - SLQSSetM2MAudioAVCFG, 1603
 - SLQSSetM2MAudioLPBK, 1604
 - SLQSSetM2MAudioNVDef, 1604
 - SLQSSetM2MAudioProfile, 1604
 - SLQSSetM2MAudioVolume, 1605
 - SLQSSetM2MSpkrGain, 1606
- qaGobiApiSwiOmadms.h, 1606
 - SLQSOMADMCancelSession, 1612
 - SLQSOMADMGetSessionInfo, 1612
 - SLQSOMADMGetSettings, 1613
 - SLQSOMADMGetSettings2, 1613
 - SLQSOMADMSendSelection, 1614
 - SLQSOMADMSendSelection2, 1614
 - SLQSOMADMSessionInfo, 1607
 - SLQSOMADMSetSettings, 1615
 - SLQSOMADMSetSettings2, 1616
 - SLQSOMADMSetSettings3, 1616
 - SLQSOMADMSettings, 1609
 - SLQSOMADMSettingsReqParams, 1610
 - SLQSOMADMSettingsReqParams3, 1611
 - SLQSOMADMStartSession, 1616
 - SLQSOMADMStartSession2, 1617
- qaGobiApiTableBandClasses.h, 1618
- qaGobiApiTableCallControlReturnReasons.h, 1621
- qaGobiApiTableCallEndReasons.h, 1621
- qaGobiApiTableCarrierCodes.h, 1637
- qaGobiApiTableCodingScheme.h, 1639
 - __GOBI_API_CODING_SCHEME_H__, 1642
- qaGobiApiTableGpsCapabilityCodes.h, 1642
- qaGobiApiTablePowerModes.h, 1643
- qaGobiApiTableRadioInterfaces.h, 1643
- qaGobiApiTableRegionCodes.h, 1644
- qaGobiApiTableServiceOptions.h, 1644
- qaGobiApiTableSupServiceInfoClasses.h, 1647
- qaGobiApiTableSwiAudio.h, 1647
- qaGobiApiTableSwiOMADMUpdateCompleteStatus.h, 1648
- qaGobiApiTableVoiceCallEndReasons.h, 1649
- qaGobiApiTmd.h, 1656
 - MAX_MITIGATION_DEV_ID_LEN, 1657
 - MAX_MITIGATION_DEV_LIST_LEN, 1657
 - SLQSTmdDeRegNotMitigationLvl, 1657
 - SLQSTmdGetMitigationDevList, 1657
 - SLQSTmdGetMitigationLvl, 1657
 - SLQSTmdRegNotMitigationLvl, 1658
- qaGobiApiUim.h, 1658
 - MAX_ACTIVE_PERS_FEATURES, 1661
 - MAX_CONTENT_LENGTH, 1661
 - MAX_DESCRIPTION_LENGTH, 1661
 - MAX_ICCID_LENGTH, 1661
 - MAX_NO_OF_APPLICATIONS, 1661
 - MAX_NO_OF_SLOTS, 1661
 - MAX_PATH_LENGTH, 1661
 - MAX_PUK_LENGTH, 1661
 - MAX_SLOTS_STATUS, 1661
 - SLQSUIMAuthenticate, 1661
 - SLQSUIMChangePin, 1662
 - SLQSUIMDepersonalization, 1662
 - SLQSUIMEventRegister, 1663
 - SLQSUIMGetCardStatus, 1663
 - SLQSUIMGetConfiguration, 1664
 - SLQSUIMGetFileAttributes, 1664
 - SLQSUIMGetSlotsStatus, 1665
 - SLQSUIMPowerDown, 1665
 - SLQSUIMPowerUp, 1666
 - SLQSUIMReadTransparent, 1666
 - SLQSUIMRefreshComplete, 1667
 - SLQSUIMRefreshGetLastEvent, 1668
 - SLQSUIMRefreshOK, 1668
 - SLQSUIMRefreshRegister, 1669
 - SLQSUIMReset, 1669
 - SLQSUIMSetPinProtection, 1670
 - SLQSUIMSwitchSlot, 1670
 - SLQSUIMUnblockPin, 1671
 - SLQSUIMVerifyPin, 1672
- qaGobiApiVoice.h, 1672
 - AnswerUSSD, 1676
 - CancelUSSD, 1677
 - MAX_CALL_NO_LEN, 1676
 - MAX_DESCRIPTION_LENGTH, 1676
 - MAX_NO_OF_CALLS, 1676
 - MAXUSSDLENGTH, 1676
 - OriginateUSSD, 1677
 - PASSWORD_LENGTH, 1676
 - SLQSOriginateUSSD, 1678
 - SLQSVoiceALSSelectLine, 1678
 - SLQSVoiceALSSetLineSwitching, 1679
 - SLQSVoiceAnswerCall, 1679
 - SLQSVoiceBindSubscription, 1680
 - SLQSVoiceBurstDTMF, 1680
 - SLQSVoiceDialCall, 1681
 - SLQSVoiceEndCall, 1681
 - SLQSVoiceGetAllCallInfo, 1682
 - SLQSVoiceGetCLIP, 1685
 - SLQSVoiceGetCLIR, 1685
 - SLQSVoiceGetCNAP, 1686
 - SLQSVoiceGetCOLP, 1687
 - SLQSVoiceGetCOLR, 1687
 - SLQSVoiceGetCallBarring, 1682
 - SLQSVoiceGetCallForwardingStatus, 1683
 - SLQSVoiceGetCallInfo, 1684
 - SLQSVoiceGetCallWaiting, 1684
 - SLQSVoiceGetConfig, 1688
 - SLQSVoiceIndicationRegister, 1688
 - SLQSVoiceManageCalls, 1689
 - SLQSVoiceOrigUSSDNoWait, 1690
 - SLQSVoiceSendFlash, 1690
 - SLQSVoiceSetCallBarringPassword, 1691

- SLQSVoiceSetConfig, [1691](#)
- SLQSVoiceSetPreferredPrivacy, [1692](#)
- SLQSVoiceSetSUPSService, [1693](#)
- SLQSVoiceStartContDTMF, [1693](#)
- SLQSVoiceStopContDTMF, [1694](#)
- serviceClassInformation, [1676](#)
- VOICE_SUPS_SRV_CLASS_DATACIRCUITSY↵
YNC, [1676](#)
- VOICE_SUPS_SRV_CLASS_DATACIRCUITSY↵
NC, [1676](#)
- VOICE_SUPS_SRV_CLASS_DATA, [1676](#)
- VOICE_SUPS_SRV_CLASS_FAX, [1676](#)
- VOICE_SUPS_SRV_CLASS_NONE, [1676](#)
- VOICE_SUPS_SRV_CLASS_PACKETACCESS,
[1676](#)
- VOICE_SUPS_SRV_CLASS_PADACCESS, [1676](#)
- VOICE_SUPS_SRV_CLASS_SMS, [1676](#)
- VOICE_SUPS_SRV_CLASS_VOICE, [1676](#)
- qaGobiApiWds.h, [1694](#)
 - GetAutoconnect, [1702](#)
 - GetByteTotals, [1702](#)
 - GetConnectionRate, [1703](#)
 - GetDataBearerTechnology, [1703](#)
 - GetDefaultProfile, [1704](#)
 - GetDefaultProfileLTE, [1706](#)
 - GetDefaultProfileNum, [1708](#)
 - GetDormancyState, [1708](#)
 - GetIPAddressLTE, [1709](#)
 - GetLastMobileIPError, [1709](#)
 - GetMobileIPProfile, [1710](#)
 - GetMobileIP, [1710](#)
 - GetPacketStatistics, [1712](#)
 - GetPacketStatus, [1712](#)
 - GetProfileSettingIn, [1698](#)
 - GetProfileSettingOut, [1699](#)
 - GetSessionDuration, [1713](#)
 - GetSessionState, [1714](#)
 - iGetByteTotals, [1714](#)
 - iGetConnectionRate, [1714](#)
 - iGetPacketStatistics, [1714](#)
 - IPV6_ADDRESS_ARRAY_SIZE, [1698](#)
 - iSLQSMISetIPFamilyPreference, [1715](#)
 - QMI_WDS_CURRENT_CALL_DB_MASK, [1702](#)
 - QMI_WDS_LAST_CALL_DB_MASK, [1702](#)
 - qmiDataBearerMasks, [1701](#)
 - QmiProfileInfo, [1699](#)
 - QmiWSDDataBearerTechnology, [1699](#)
 - QmiWSDDataBearers, [1699](#)
 - RMSetTransferStatistics, [1715](#)
 - SLQSAutoConnect, [1724](#)
 - SLQSCreateProfile, [1724](#)
 - SLQSDeleteProfile, [1725](#)
 - SLQSGet3GPPConfigItem, [1725](#)
 - SLQSGetByteTotals, [1726](#)
 - SLQSGetConnectionRate, [1726](#)
 - SLQSGetCurrDataSystemStat, [1727](#)
 - SLQSGetCurrentChannelRate, [1727](#)
 - SLQSGetDUNCallInfo, [1729](#)
 - SLQSGetDataBearerTechnology, [1728](#)
 - SLQSGetDataBearerTechnologyExt, [1728](#)
 - SLQSGetPacketStatistics, [1729](#)
 - SLQSGetProfile, [1730](#)
 - SLQSGetProfileSettings, [1731](#)
 - SLQSGetRuntimeSettings, [1732](#)
 - SLQSGetSessionState, [1732](#)
 - SLQSModifyProfile, [1733](#)
 - SLQSResetPacketStatics, [1733](#)
 - SLQSSGetDHCPv4ClientConfig, [1736](#)
 - SLQSSGetLoopback, [1736](#)
 - SLQSSSetDHCPv4ClientConfig, [1736](#)
 - SLQSSSetLoopback, [1737](#)
 - SLQSSet3GPPConfigItem, [1734](#)
 - SLQSSetProfile, [1734](#)
 - SLQSStartStopDataSession, [1737](#)
 - SLQSWdsGoActive, [1738](#)
 - SLQSWdsGoDormant, [1738](#)
 - SLQSWdsSetEventReport, [1739](#)
 - SLQSWdsSwiPDPRuntimeSettings, [1739](#)
 - SetActiveMobileIPProfile, [1715](#)
 - SetAutoconnect, [1715](#)
 - SetDefaultProfile, [1716](#)
 - SetDefaultProfileLTEV2, [1719](#)
 - SetDefaultProfileLTE, [1717](#)
 - SetDefaultProfileNum, [1720](#)
 - SetMobileIPParameters, [1721](#)
 - SetMobileIPProfile, [1722](#)
 - SetMobileIP, [1721](#)
 - slqs3GPPConfigItem, [1700](#)
 - WDS_IsGobiDevice, [1740](#)
- qaNasGetRFBandInfo.h, [1740](#)
 - eQMI_NAS_GET_RF_INFO_RESP, [1740](#)
 - eTLV_RF_BAND_INFO, [1740](#)
 - PkQmiNasGetRFBandInfo, [1741](#)
 - UpkQmiNasGetRFBandInfo, [1741](#)
- qaNasPerformNetworkScan.h, [1741](#)
 - eQMI_NAS_PERFORM_NETWORK_SCAN_R↵
ESP, [1741](#)
 - eTLV_3GPP_NETWORK_INFO, [1741](#)
 - FORBIDDEN_INDEX, [1741](#)
 - INDEX_ZERO, [1741](#)
 - MAX_DESCRIPTION_LENGTH, [1741](#)
 - PREFERRED_INDEX, [1741](#)
 - PkQmiNasPerformNetworkScan, [1742](#)
 - QMI_NAS_MAX_INSTANCES, [1741](#)
 - QMI_NAS_NETSTATUS_MASK, [1741](#)
 - ROAMING_INDEX, [1741](#)
 - UpkQmiNasPerformNetworkScan, [1742](#)
- qaQmi3GPP2BroadcastCfgInfo
 - qaGobiApiSms.h, [1579](#)
- qaQmi3GPPBroadcastCfgInfo
 - qaGobiApiSms.h, [1579](#)
- qaQmi3Gpp2TimeZone, [677](#)
 - daylightSavings, [678](#)
 - leapSeconds, [678](#)
 - localTimeOffset, [678](#)
- qaQmiInterfaceInfo, [678](#)

- qaQmiinstanceid, [679](#)
- qaQmisvctype, [679](#)
- v4sessionId, [679](#)
- v6sessionId, [679](#)
- qaQmiServingSystemParam, [679](#)
 - BasestationID, [682](#)
 - BasestationLatitude, [682](#)
 - BasestationLongitude, [682](#)
 - CDMA_P_Rev, [682](#)
 - CDMASystemInfoExt, [682](#)
 - CallBarStatus, [682](#)
 - CellID, [682](#)
 - concSvcInfo, [682](#)
 - CurrentPLMN, [682](#)
 - DTMInd, [682](#)
 - DataSrvCapabilities, [682](#)
 - defaultRoamInd, [682](#)
 - DetailedSvcInfo, [682](#)
 - Gpp2TimeZone, [682](#)
 - GppNetworkDSTAdjustment, [682](#)
 - GppTimeZone, [682](#)
 - hdrPersonality, [683](#)
 - Lac, [683](#)
 - NetworkID, [683](#)
 - PRLInd, [683](#)
 - roamIndicatorVal, [683](#)
 - RoamingIndicatorList, [683](#)
 - ServingSystem, [683](#)
 - SystemID, [683](#)
 - trackAreaCode, [683](#)
- qaQmiinstanceid
 - qaQmiInterfaceInfo, [679](#)
- qaQmisvctype
 - qaQmiInterfaceInfo, [679](#)
- qm_wds_ds_profile_extended_err_codes
 - qmerrno.h, [1748](#)
- qmerrno.h, [1742](#)
 - eQCWWAN_ERR_API_MUTEX_TIMEOUT, [1745](#)
 - eQCWWAN_ERR_BUFFER_SZ, [1744](#)
 - eQCWWAN_ERR_CANCEL_OP, [1745](#)
 - eQCWWAN_ERR_DRIVER, [1745](#)
 - eQCWWAN_ERR_ENUM_BEGIN, [1744](#)
 - eQCWWAN_ERR_ENUM_END, [1745](#)
 - eQCWWAN_ERR_FILE_COPY, [1745](#)
 - eQCWWAN_ERR_FILE_OPEN, [1745](#)
 - eQCWWAN_ERR_GENERAL, [1744](#)
 - eQCWWAN_ERR_INTERNAL, [1744](#)
 - eQCWWAN_ERR_INVALID_ARG, [1744](#)
 - eQCWWAN_ERR_INVALID_DEVID, [1744](#)
 - eQCWWAN_ERR_INVALID_FILE, [1744](#)
 - eQCWWAN_ERR_INVALID_QMI_RSP, [1744](#)
 - eQCWWAN_ERR_INVALID_XID, [1745](#)
 - eQCWWAN_ERR_MALFORMED_QMI_RSP, [1744](#)
 - eQCWWAN_ERR_MEMORY, [1744](#)
 - eQCWWAN_ERR_MULTIPLE_DEVICES, [1745](#)
 - eQCWWAN_ERR_MULTIPLE_SMS_UNSUPP↵ORTED, [1745](#)
 - eQCWWAN_ERR_NO_CANCELABLE_OP, [1745](#)
 - eQCWWAN_ERR_NO_CONNECTION, [1744](#)
 - eQCWWAN_ERR_NO_DEVICE, [1744](#)
 - eQCWWAN_ERR_NO_SIGNAL, [1745](#)
 - eQCWWAN_ERR_NONE, [1744](#)
 - eQCWWAN_ERR_NULL_TLV, [1748](#)
 - eQCWWAN_ERR_OFFLINE, [1745](#)
 - eQCWWAN_ERR_PDU_GENERATION, [1745](#)
 - eQCWWAN_ERR_QMI_ABORTED, [1745](#)
 - eQCWWAN_ERR_QMI_ACCESS_DENIED, [1747](#)
 - eQCWWAN_ERR_QMI_ACK_NOT_SENT, [1747](#)
 - eQCWWAN_ERR_QMI_ARG_TOO_LONG, [1745](#)
 - eQCWWAN_ERR_QMI_AUTHENTICATION_F↵AILED, [1746](#)
 - eQCWWAN_ERR_QMI_AUTHENTICATION_L↵OCK, [1746](#)
 - eQCWWAN_ERR_QMI_BUNDLING_NOT_SUP↵ORTED, [1747](#)
 - eQCWWAN_ERR_QMI_CALL_FAILED, [1745](#)
 - eQCWWAN_ERR_QMI_CARD_BUSY_RSP, [1748](#)
 - eQCWWAN_ERR_QMI_CARD_CALL_CONTR↵OL_FAILED, [1747](#)
 - eQCWWAN_ERR_QMI_CAT_END, [1748](#)
 - eQCWWAN_ERR_QMI_CAT_START, [1748](#)
 - eQCWWAN_ERR_QMI_CAUSE_CODE, [1746](#)
 - eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUS↵TED, [1745](#)
 - eQCWWAN_ERR_QMI_CONNECT, [1744](#)
 - eQCWWAN_ERR_QMI_DEVICE_IN_USE, [1745](#)
 - eQCWWAN_ERR_QMI_DEVICE_NOT_READY, [1746](#)
 - eQCWWAN_ERR_QMI_DEVICE_STORAGE_F↵ULL, [1746](#)
 - eQCWWAN_ERR_QMI_DISABLED, [1747](#)
 - eQCWWAN_ERR_QMI_ENCODING, [1746](#)
 - eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAI↵LURE, [1748](#)
 - eQCWWAN_ERR_QMI_EVENT_REG_FAILED, [1748](#)
 - eQCWWAN_ERR_QMI_EXTENDED_INTERNAL, [1747](#)
 - eQCWWAN_ERR_QMI_FDN_RESTRICT, [1747](#)
 - eQCWWAN_ERR_QMI_FLOW_SUSPENDED, [1746](#)
 - eQCWWAN_ERR_QMI_GENERAL, [1746](#)
 - eQCWWAN_ERR_QMI_HARDWARE_RESTRI↵CTED, [1747](#)
 - eQCWWAN_ERR_QMI_IFACE, [1744](#)
 - eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE, [1747](#)
 - eQCWWAN_ERR_QMI_INCORRECT_FLOW↵FILTER, [1746](#)
 - eQCWWAN_ERR_QMI_INCORRECT_PIN, [1745](#)
 - eQCWWAN_ERR_QMI_INFO_UNAVAILABLE, [1747](#)
 - eQCWWAN_ERR_QMI_INJECT_TIMEOUT, [1747](#)
 - eQCWWAN_ERR_QMI_INSUFFICIENT_RESO↵URCES, [1747](#)

- eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND, 1746
- eQCWWAN_ERR_QMI_INTERNAL, 1745
- eQCWWAN_ERR_QMI_INVALID_ARG, 1746
- eQCWWAN_ERR_QMI_INVALID_CLIENT_ID, 1745
- eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT, 1746
- eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD, 1748
- eQCWWAN_ERR_QMI_INVALID_HANDLE, 1745
- eQCWWAN_ERR_QMI_INVALID_INDEX, 1746
- eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF, 1746
- eQCWWAN_ERR_QMI_INVALID_ID, 1746
- eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE, 1746
- eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID, 1746
- eQCWWAN_ERR_QMI_INVALID_OPERATION, 1747
- eQCWWAN_ERR_QMI_INVALID_PDP_TYPE, 1745
- eQCWWAN_ERR_QMI_INVALID_PINID, 1745
- eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE, 1746
- eQCWWAN_ERR_QMI_INVALID_PROFILE, 1745
- eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION, 1746
- eQCWWAN_ERR_QMI_INVALID_QMI_CMD, 1747
- eQCWWAN_ERR_QMI_INVALID_QOS_ID, 1746
- eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION, 1746
- eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE, 1746
- eQCWWAN_ERR_QMI_INVALID_TECH_PREF, 1745
- eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP, 1748
- eQCWWAN_ERR_QMI_INVALID_TRANSITION, 1746
- eQCWWAN_ERR_QMI_INVALID_TX_ID, 1745
- eQCWWAN_ERR_QMI_MALFORMED_MSG, 1745
- eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE, 1746
- eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE, 1746
- eQCWWAN_ERR_QMI_MAX, 1747
- eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE, 1746
- eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT, 1746
- eQCWWAN_ERR_QMI_MISSING_ARG, 1745
- eQCWWAN_ERR_QMI_MSG_BLOCKED, 1747
- eQCWWAN_ERR_QMI_NETWORK_ABORTED, 1747
- eQCWWAN_ERR_QMI_NETWORK_NOT_READY, 1746
- eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE, 1746
- eQCWWAN_ERR_QMI_NO_EFFECT, 1745
- eQCWWAN_ERR_QMI_NO_ENTRY, 1746
- eQCWWAN_ERR_QMI_NO_FREE_PROFILE, 1745
- eQCWWAN_ERR_QMI_NO_MEMORY, 1745
- eQCWWAN_ERR_QMI_NO_NETWORK_FOULND, 1745
- eQCWWAN_ERR_QMI_NO_RADIO, 1747
- eQCWWAN_ERR_QMI_NO_SUBSCRIPTION, 1747
- eQCWWAN_ERR_QMI_NO_THRESHOLDS, 1745
- eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE, 1746
- eQCWWAN_ERR_QMI_NOT_PROVISIONED, 1745
- eQCWWAN_ERR_QMI_NOT_SUPPORTED, 1747
- eQCWWAN_ERR_QMI_OFFSET, 1745
- eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED, 1745
- eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED, 1745
- eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE, 1747
- eQCWWAN_ERR_QMI_OUT_OF_CALL, 1745
- eQCWWAN_ERR_QMI_PIN_BLOCKED, 1746
- eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED, 1746
- eQCWWAN_ERR_QMI_POLICY_MISMATCH, 1747
- eQCWWAN_ERR_QMI_REQ_SCH, 1744
- eQCWWAN_ERR_QMI_REQ_TO, 1744
- eQCWWAN_ERR_QMI_REQUESTED_NUM_UNNSUPPORTED, 1746
- eQCWWAN_ERR_QMI_REQ, 1744
- eQCWWAN_ERR_QMI_RSP_TO, 1744
- eQCWWAN_ERR_QMI_RSP, 1744
- eQCWWAN_ERR_QMI_SEGMENT_ORDER, 1747
- eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG, 1747
- eQCWWAN_ERR_QMI_SESSION_INACTIVE, 1746
- eQCWWAN_ERR_QMI_SESSION_INVALID, 1746
- eQCWWAN_ERR_QMI_SESSION_OWNERSHIP, 1746
- eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND, 1747
- eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED, 1746
- eQCWWAN_ERR_QMI_SMSC_ADDR, 1747
- eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE, 1747

- SE, [1747](#)
- eQCWWAN_ERR_QMI_TPDU_TYPE, [1747](#)
- eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION, [1745](#)
- eQCWWAN_ERR_QMI_UNKNOWN, [1746](#)
- eQCWWAN_ERR_QMI_WIDTH, [1748](#)
- eQCWWAN_ERR_RESET, [1745](#)
- eQCWWAN_ERR_SWICM_AM_VERS_ERROR, [1747](#)
- eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS, [1747](#)
- eQCWWAN_ERR_SWICM_END, [1748](#)
- eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS, [1747](#)
- eQCWWAN_ERR_SWICM_INVALID_SESSION_ID, [1747](#)
- eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID, [1748](#)
- eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID, [1748](#)
- eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED, [1747](#)
- eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED, [1747](#)
- eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED, [1747](#)
- eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS, [1748](#)
- eQCWWAN_ERR_SWICM_SOCKET_IN_USE, [1747](#)
- eQCWWAN_ERR_SWICM_START, [1747](#)
- eQCWWAN_ERR_SWICM_TIMEOUT, [1747](#)
- eQCWWAN_ERR_SWICM_V4DWN_V6DWN, [1747](#)
- eQCWWAN_ERR_SWICM_V4DWN_V6UP, [1747](#)
- eQCWWAN_ERR_SWICM_V4UP_V6DWN, [1747](#)
- eQCWWAN_ERR_SWICM_V4UP_V6UP, [1747](#)
- eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED, [1748](#)
- eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND, [1748](#)
- eQCWWAN_ERR_SWIDCS_END, [1748](#)
- eQCWWAN_ERR_SWIDCS_FILEIO_ERR, [1748](#)
- eQCWWAN_ERR_SWIDCS_IOCTL_ERR, [1748](#)
- eQCWWAN_ERR_SWIDCS_START, [1748](#)
- eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE, [1748](#)
- eQCWWAN_ERR_SWIIM_END, [1748](#)
- eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND, [1748](#)
- eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED, [1748](#)
- eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE, [1748](#)
- eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETION, [1748](#)
- eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX, [1748](#)
- eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH, [1748](#)
- eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE, [1748](#)
- eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL, [1748](#)
- eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS, [1748](#)
- eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT, [1748](#)
- eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE, [1748](#)
- eQCWWAN_ERR_SWIIM_INVALID_PATH, [1748](#)
- eQCWWAN_ERR_SWIIM_OPENING_DIR, [1748](#)
- eQCWWAN_ERR_SWIIM_OPENING_FILE, [1748](#)
- eQCWWAN_ERR_SWIIM_START, [1748](#)
- eQCWWAN_ERR_SWISM_END, [1748](#)
- eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND, [1748](#)
- eQCWWAN_ERR_SWISMS_MSG_CORRUPTED, [1748](#)
- eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG, [1748](#)
- eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED, [1748](#)
- eQCWWAN_ERR_SWISMS_START, [1748](#)
- eQCWWANError, [1744](#)
- eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE, [1749](#)
- eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR, [1749](#)
- eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED, [1749](#)
- eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES, [1749](#)
- eWDS_ERR_PROFILE_REG_3GPP_INVALID_PROFILE_FAMILY, [1749](#)
- eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET, [1749](#)
- eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET, [1749](#)
- eWDS_ERR_PROFILE_REG_END, [1749](#)
- eWDS_ERR_PROFILE_REG_INVALID_PROFILE_FAMILY, [1749](#)
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_HNDL, [1749](#)
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_IDENT, [1749](#)
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_OP, [1749](#)
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_PROFILE_NUM, [1749](#)
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_PROFILE_TYPE, [1749](#)
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID_SUBS_ID, [1749](#)
- eWDS_ERR_PROFILE_REG_RESULT_ERR_INVALID, [1749](#)

- eWDS_ERR_PROFILE_REG_RESULT_ERR_L↔
EN_INVALID, 1749
- eWDS_ERR_PROFILE_REG_RESULT_ERR_L↔
IB_NOT_INITED, 1749
- eWDS_ERR_PROFILE_REG_RESULT_FAIL,
1749
- eWDS_ERR_PROFILE_REG_RESULT_LIST_E↔
ND, 1749
- qm_wds_ds_profile_extended_err_codes, 1748
- QmiCbkCatEventStatusReportInd, 683
 - CCETlv, 683
 - event_Index, 683
- QmiCbkLocBestAvailPosInd, 683
 - pAltitudeWrtEllipsoid, 688
 - pAltitudeWrtMeanSeaLevel, 688
 - pGpsTime, 688
 - pHeading, 688
 - pHeadingUnc, 688
 - pHorCirConf, 688
 - pHorEllpConf, 688
 - pHorReliability, 688
 - pHorUncCircular, 688
 - pHorUncEllipseOrientAzimuth, 689
 - pHorUncEllipseSemiMajor, 689
 - pHorUncEllipseSemiMinor, 689
 - pLatitude, 689
 - pLongitude, 689
 - pMagneticDeviation, 689
 - pPrecisionDilution, 689
 - pSensorDataUsage, 689
 - pSpeedHorizontal, 689
 - pSpeedUnc, 689
 - pSpeedVertical, 689
 - pSpeedVerticalUnc, 689
 - pSvUsedforFix, 689
 - pTechnologyMask, 689
 - pTimeSrc, 689
 - pTimeUnc, 689
 - pTimestampUtc, 689
 - pVertConfidence, 689
 - pVertReliability, 689
 - pVertUnc, 689
 - pXid, 689
 - status, 689
- QmiCbkLocCradleMountInd, 689
 - cradleMountConfigStatus, 690
- QmiCbkLocEngineStateInd, 690
 - engineState, 690
- QmiCbkLocEventTimeSyncInd, 691
 - timeSyncRefCounter, 691
- QmiCbkLocInjectPositionInd, 691
 - status, 692
- QmiCbkLocInjectSensorDataInd, 692
 - injectSensorDataStatus, 693
 - pAccelSamplesAccepted, 693
 - pAccelTempSamplesAccepted, 693
 - pGyroSamplesAccepted, 693
 - pGyroTempSamplesAccepted, 693
- pOpaqueIdentifier, 693
- QmiCbkLocInjectTimeInd, 693
 - injectTimeSyncStatus, 694
- QmiCbkLocInjectUTCTimeInd, 694
 - status, 695
- QmiCbkLocPositionReportInd, 695
 - pAltitudeAssumed, 699
 - pAltitudeWrtEllipsoid, 699
 - pAltitudeWrtMeanSeaLevel, 699
 - pFixId, 699
 - pGpsTime, 700
 - pHeading, 700
 - pHeadingUnc, 700
 - pHorConfidence, 700
 - pHorReliability, 700
 - pHorUncCircular, 700
 - pHorUncEllipseOrientAzimuth, 700
 - pHorUncEllipseSemiMajor, 700
 - pHorUncEllipseSemiMinor, 700
 - pLatitude, 700
 - pLeapSeconds, 700
 - pLongitude, 700
 - pMagneticDeviation, 700
 - pPrecisionDilution, 700
 - pSensorDataUsage, 700
 - pSpeedHorizontal, 700
 - pSpeedUnc, 700
 - pSpeedVertical, 700
 - pSvUsedforFix, 700
 - pTechnologyMask, 700
 - pTimeSrc, 700
 - pTimeUnc, 700
 - pTimestampUtc, 700
 - pVertConfidence, 701
 - pVertReliability, 701
 - pVertUnc, 701
 - sessionId, 701
 - sessionStatus, 701
- QmiCbkLocSensorStreamingInd, 701
 - pAccelAcceptReady, 701
 - pAccelTempAcceptReady, 701
 - pGyroAcceptReady, 701
 - pGyroTempAcceptReady, 702
- QmiCbkLocSetExtPowerConfigInd, 702
 - status, 702
- QmiCbkNasLTECphyCaInfo, 702
 - sPhyCaAggPcellInfo, 703
 - sPhyCaAggScellIDBw, 703
 - sPhyCaAggScellIndType, 703
 - sPhyCaAggScellIndex, 703
 - sPhyCaAggScellInfo, 703
- QmiCbkSwiOmaDmEventStatusReportInd, 703
 - SITlv, 703
- QmiCbkSwiOmaDmEventStatusReportIndExt, 703
 - SITlv, 704
- QmiCbkTmdMitiLvlRptInd, 704
 - currentMitigationLvl, 704
 - MitigationDevInfo, 704

- QmiCbkWdsStatisticsIndState, 704
 - RxDropConutTlv, 705
 - RxOkByteCountTlv, 705
 - RxOkConutTlv, 705
 - TxDropConutTlv, 705
 - TxOkByteCountTlv, 705
 - TxOkConutTlv, 705
- qmiDataBearerMasks
 - qaGobiApiWds.h, 1701
- QmiNas3GppNetworkInfo, 706
 - pDescription, 707
 - pForbidden, 707
 - pInUse, 707
 - pMCC, 707
 - pMNC, 707
 - pPreferred, 707
 - pRoaming, 707
- QmiNasGetRFBandInfoResp, 708
 - pInstancesSize, 708
 - pRFBandInfoElements, 708
 - results, 708
- QmiNasPerformNetworkScanResp, 708
 - pInstanceSize, 708
 - pInstances, 708
 - results, 708
- QmiProfileInfo
 - qaGobiApiWds.h, 1699
- qmiSmsMessageList, 708
 - messageIndex, 709
 - messageTag, 709
- QmiWSDDataBearerTechnology
 - qaGobiApiWds.h, 1699
- qmiWSDDataBearerTechnology, 709
 - currentNetwork, 709
 - ratMask, 709
 - soMask, 709
- QmiWSDDataBearers
 - qaGobiApiWds.h, 1699
- QmiWdsIpAddressInfo, 709
 - pIPAddressV4, 710
 - pIPAddressV6, 710
 - pIPv6prefixlen, 710
- qmiWdsRunTimeSettings, 710
 - pAPNName, 713
 - pAuthentication, 713
 - pDomainList, 713
 - pGPRSGrantedQoS, 713
 - pGWAddressV4, 713
 - pIMCNflag, 713
 - pIPAddressV4, 713
 - pIPFamilyPreference, 713
 - pIPv6AddrInfo, 713
 - pIPv6GWAddrInfo, 713
 - pMtu, 713
 - pPCSCFAddrPCO, 713
 - pPCSCFFQDNAddrList, 713
 - pPDPTtype, 713
 - pPrimaryDNSV4, 713
 - pPrimaryDNSV6, 714
 - pProfileID, 714
 - pProfileName, 714
 - pSecondaryDNSV4, 714
 - pSecondaryDNSV6, 714
 - pServerAddrList, 714
 - pSubnetMaskV4, 714
 - pTechnology, 714
 - pUMTSGrantedQoS, 714
 - pUsername, 714
- qmifwinfo_s, 705
 - dev, 706
 - g, 706
 - s, 706
- qos.h, 1749
 - LIBPACK_MAX_QOS_FILTERS, 1751
 - LIBPACK_MAX_QOS_FLOW_PER_APN_STATS, 1751
 - LIBPACK_MAX_QOS_FLOWS, 1751
 - pack_qos_SLQSQosGetNetworkStatus, 1751
 - pack_qos_SLQSQosSwiReadApnExtraParams, 1751
 - pack_qos_SLQSQosSwiReadDataStats, 1752
 - pack_qos_SLQSSetQosEventCallback, 1753
 - unpack_qos_SLQSQosGetNetworkStatus, 1753
 - unpack_qos_SLQSQosSwiReadApnExtraParams, 1754
 - unpack_qos_SLQSQosSwiReadDataStats, 1754
 - unpack_qos_SLQSSetQosEventCallback, 1755
 - unpack_qos_SLQSSetQosEventCallback_ind, 1755
 - unpack_qos_SLQSSetQosNWStatusCallback_ind, 1756
 - unpack_qos_SLQSSetQosPriEventCallback_ind, 1757
 - unpack_qos_SLQSSetQosStatusCallback_ind, 1757
- qos_id
 - QosMap, 719
- QosClassID, 714
 - gDIBitRate, 715
 - gUIBitRate, 715
 - maxDIBitRate, 715
 - maxUIBitRate, 715
 - QCI, 715
- qosDeliveryOrder
 - LibPackUMTSQoS, 349
 - UMTSMinQoS, 919
 - UMTSQoS, 922
 - wds_UMTSMinQoS, 1174
- QosEventInfo, 715
 - pDataBearer, 716
 - pPacketsCountRX, 716
 - pPacketsCountTX, 716
 - pTotalBytesRX, 716
 - pTotalBytesTX, 716
- qosFlow
 - sQosStat, 824

- unpack_qos_SLQSQosSwiReadDataStats_t, 1027
- QosFlowInfo, 716
 - pBearerID, 717
 - pQFlowState, 717
 - pRxQFilter, 717
 - pRxQFlowGranted, 717
 - pTxQFilter, 717
 - pTxQFlowGranted, 717
 - unpack_qos_SLQSSetQosEventCallback_ind_t, 1028
- QosFlowInfoState, 717
 - id, 718
 - isNewFlow, 718
 - state, 718
- QosMap, 718
 - dscp, 719
 - qos_id, 719
 - state, 719
- Quality of Service (QOS), 50
- RATMask
 - CurrNetworkInfo, 186
 - currNetworkInfo, 187
 - wds_currNetworkInfo, 1160
- RAN
 - unpack_nas_GetServingNetwork_t, 986
- RAT
 - _SlqsNas3GppNetworkRAT_, 70
 - nas_QmiNas3GppNetworkRAT, 463
- REGISTER_EVENT
 - qaGobiApiCbK.h, 1328
- REGISTER_SRV
 - qaGobiApiCbK.h, 1328
- RFBandInfoElements, 726
 - activeBandClass, 727
 - activeChannel, 727
 - radiolInterface, 727
 - unpack_nas_GetRFInfo_t, 984
- RFTlv
 - unpack_nas_SetEventReportInd_t, 988
- RMAutoConnect
 - pack_dms_SetCustFeature_t, 555
 - unpack_dms_GetCustFeature_t, 927
- RMSetTransferStatistics
 - qaGobiApiWds.h, 1715
- ROAMING_INDEX
 - qaNasPerformNetworkScan.h, 1741
- RPCause
 - SMSAsyncRawSend_s, 806
- RPTlv
 - NASQmiCbK NasSystemSelPrefInd, 521
- RRTlv
 - unpack_nas_SetEventReportInd_t, 988
- RSRPThresListLen
 - RSRPThresh, 730
- RSRPThresh, 730
 - pRSRPThresList, 730
 - RSRPThresListLen, 730
- RSRQThresListLen
 - RSRQThresh, 731
- RSRQThresh, 731
 - pRSRQThresList, 731
 - RSRQThresListLen, 731
- RSSIThresListLen
 - RSSIThresh, 732
- RSSIThresh, 732
 - pRSSIThresList, 732
 - RSSIThresListLen, 732
- RX_EC_IO
 - NetworkStat1x, 536
- RX_PWR
 - NetworkStat1x, 536
 - NetworkStatEVDO, 538
- RXAGCList, 732
 - pRXAIG, 733
 - pRXComprSlope, 733
 - pRXComprThres, 733
 - pRXExpSlope, 733
 - pRXExpThres, 733
 - pRXStaticGain, 733
- RXAVCList, 733
 - pAVRXAVCHheadroom, 734
 - pAVRXAVCSens, 734
- RXChan
 - LTEInfo, 381
 - nas_LTEInfo, 440
- rXDroppedCount
 - unpack_wds_GetPacketStatus_t, 1069
- RXOKBytesCount
 - DUNCallInfoInd, 218
- rXOKBytesLastCall
 - unpack_wds_GetPacketStatus_t, 1069
- rXOkBytesCount
 - unpack_wds_GetPacketStatus_t, 1069
- RXPCMIIRFltr, 735
 - pFlag, 737
 - pStage0Val, 737
 - pStage1Val, 737
 - pStage2Val, 737
 - pStage3Val, 737
 - pStage4Val, 737
 - pStageCnt, 737
- rXPacketErrors
 - unpack_wds_GetPacketStatus_t, 1069
- rXPacketOverflows
 - unpack_wds_GetPacketStatus_t, 1069
- rXPacketSuccesses
 - unpack_wds_GetPacketStatus_t, 1069
- radio
 - unpack_nas_GetSignalStrengths_t, 987
- radio_if
 - nasGetTxRxInfoReq, 503
- radiolf
 - ecioListElement, 219
 - errorRateListElement, 223
 - nas_ecioListElement, 422
 - nas_errorRateListElement, 423

- nas_rsrqInformation, [467](#)
- nas_rxSignalStrengthListElement, [468](#)
- rsrqInformation, [731](#)
- rxSignalStrengthListElement, [739](#)
- Radiolfaces
 - unpack_dms_GetDeviceCap_t, [928](#)
 - unpack_dms_GetDeviceCapabilities_t, [929](#)
 - unpack_nas_GetServingNetwork_t, [986](#)
- RadiolfacesSize
 - unpack_dms_GetDeviceCap_t, [928](#)
 - unpack_nas_GetServingNetwork_t, [986](#)
- radiolfacesSize
 - unpack_dms_GetDeviceCapabilities_t, [929](#)
- radiolInterface
 - nas_RFInfoTlv, [465](#)
 - nas_SignalStrengthTlv, [472](#)
 - nas_roamIndList, [466](#)
 - nas_servSystem, [471](#)
 - nas_timeInfo, [482](#)
 - RFBandInfoElements, [727](#)
 - roamIndList, [728](#)
 - servSystem, [752](#)
 - timeInfo, [862](#)
- radiolInterfaceList
 - NASServingSystemInfo, [523](#)
 - ServingSystemInfo, [751](#)
- radiolInterfaceNo
 - NASServingSystemInfo, [523](#)
 - ServingSystemInfo, [751](#)
- radiolInterfaceSize
 - nas_RFInfoTlv, [465](#)
- range
 - Port, [660](#)
 - unpack_qos_Port_t, [1021](#)
- RankIndicatorInd, [719](#)
 - Count1, [719](#)
 - Count2, [719](#)
- rat
 - CSGID, [178](#)
 - MNRInfo, [402](#)
 - nas_CSGID, [418](#)
 - nas_MNRInfo, [453](#)
- ratMask
 - dataBearerTechnology, [197](#)
 - qmiWSDDataBearerTechnology, [709](#)
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- ratValue
 - DataBearerTech, [195](#)
- rawHorConfidence
 - pack_loc_SLQSLOCInjectPosition_t, [577](#)
- rawHorUncCircular
 - pack_loc_SLQSLOCInjectPosition_t, [577](#)
- rawLen
 - fileAttributes, [228](#)
- rawValue
 - fileAttributes, [228](#)
- rcv4
 - ssdatasession_params, [827](#)
- rcv6
 - ssdatasession_params, [827](#)
- readResult, [719](#)
 - content, [719](#)
 - contentLen, [720](#)
- readTransparent
 - pack_uim_ReadTransparent_t, [613](#)
 - UIMReadTransparentReq, [901](#)
- readTransparentInfo, [720](#)
 - length, [720](#)
 - offset, [720](#)
- Reason
 - voiceGetCallFWReq, [1110](#)
 - voiceSetCallBarringPwdInfo, [1134](#)
- reason
 - ccSUPSType, [145](#)
 - redirNumInfo, [722](#)
 - unpack_qos_SLQSSetQosStatusCallback_ind_t, [1030](#)
 - voiceGetCallBarringReq, [1108](#)
 - voiceSetSUPSServiceReq, [1140](#)
- receiptAction
 - smsRouteEntry, [819](#)
- receivedBytes
 - omaDmFotaTlvExt, [549](#)
- reconfigReqd
 - _packetSrvStatus, [64](#)
 - unpack_wds_SLQSSetPacketSrvStatusCallback↔_t, [1081](#)
- reconfiguration_required
 - slqsSessionStateInfo, [797](#)
- recordCount
 - fileAttributes, [228](#)
- recordSize
 - fileAttributes, [228](#)
- redirNumInfo, [720](#)
 - numLen, [722](#)
 - numPlan, [722](#)
 - numType, [722](#)
 - number, [722](#)
 - PI, [722](#)
 - reason, [722](#)
 - SI, [722](#)
- RedirPartyNum
 - arrRedirPartyNum, [117](#)
- refData
 - unpack_dms_SLQSSwiGetFwUpdateStatus_t, [951](#)
- refString
 - unpack_dms_SLQSSwiGetFwUpdateStatus_t, [951](#)
- ReferenceID
 - CatAIPhIdIdentifierTlv, [141](#)
 - CatEventIDDDataTlv, [143](#)
- refpn
 - CDMAInfo, [148](#)
 - nas_CDMAInfo, [410](#)
- refreshComplete
 - UIMRefreshCompleteReq, [903](#)

- RefreshMode
 - CatRefreshTlv, [144](#)
- RefreshStage
 - CatRefreshTlv, [144](#)
- regAction
 - nasInitNetworkReg, [508](#)
 - pack_nas_SLQSIInitiateNetworkRegistration_t, [585](#)
- RegForeignNID
 - unpack_nas_GetCDMANetworkParameters_t, [982](#)
- RegForeignSID
 - unpack_nas_GetCDMANetworkParameters_t, [982](#)
- RegHomeSID
 - unpack_nas_GetCDMANetworkParameters_t, [982](#)
- regInd
 - _transLayerInfoNotification, [95](#)
- regPrd
 - AddCDMASysInfo, [99](#)
 - nas_AddCDMASysInfo, [406](#)
- regRefresh
 - UIMRefreshRegisterReq, [906](#)
- regRejectInfoValid
 - GSMSysInfo, [283](#)
 - LTESysInfo, [397](#)
 - nas_GSMSysInfo, [431](#)
 - nas_LTESysInfo, [451](#)
 - nas_WCDMASysInfo, [494](#)
 - WCDMASysInfo, [1158](#)
- regState
 - nas_servSystem, [471](#)
 - servSystem, [752](#)
- Region
 - fwinfo_s, [236](#)
- registerFlag
 - registerRefresh, [723](#)
- registerRefresh, [722](#)
 - arrfileInfo, [723](#)
 - numFiles, [723](#)
 - registerFlag, [723](#)
 - voteForInit, [723](#)
- RegistrationState
 - unpack_nas_GetServingNetwork_t, [986](#)
- registrationState
 - NASServingSystemInfo, [523](#)
 - ServingSystemInfo, [751](#)
- rejCause
 - GSMSysInfo, [283](#)
 - LTESysInfo, [397](#)
 - nas_GSMSysInfo, [431](#)
 - nas_LTESysInfo, [451](#)
 - nas_WCDMASysInfo, [494](#)
 - WCDMASysInfo, [1158](#)
- rejectCause
 - nas_RejectReasonTlv, [464](#)
- rejectSrvDomain
 - GSMSysInfo, [284](#)
 - LTESysInfo, [397](#)
 - nas_GSMSysInfo, [431](#)
 - nas_LTESysInfo, [452](#)
 - nas_WCDMASysInfo, [494](#)
 - WCDMASysInfo, [1158](#)
- reliabilityClass
 - GPRSQoS, [272](#)
 - GPRSRequestedQoS, [273](#)
 - LibPackGPRSRequestedQoS, [321](#)
 - wds_GPRSQoS, [1166](#)
- remPartyNumber
 - remotePartyNum, [725](#)
- remainingRetries, [723](#)
 - unblockLeft, [723](#)
 - verifyLeft, [723](#)
- Remote Management Service (RMS), [41](#)
- RemotePartyName
 - getAllCallRmtPtyName, [240](#)
- remotePartyName, [723](#)
 - callerName, [724](#)
 - codingScheme, [724](#)
 - nameLen, [724](#)
 - namePI, [724](#)
- RemotePartyNum
 - getAllCallRmtPtyNum, [240](#)
- remotePartyNum, [724](#)
 - numLen, [725](#)
 - presentationInd, [725](#)
 - remPartyNumber, [725](#)
- ReqFieldsList, [725](#)
 - requestFields, [726](#)
 - requestFieldsLen, [726](#)
- requestFields
 - ReqFieldsList, [726](#)
- requestFieldsLen
 - ReqFieldsList, [726](#)
- resBerRatio
 - LibPackUMTSQoS, [349](#)
 - UMTSMinQoS, [920](#)
 - UMTSQoS, [922](#)
 - wds_UMTSMinQoS, [1174](#)
- ResCode
 - FirmwareUpdatStat, [231](#)
 - GetAudioVoITLBConfigResp, [246](#)
 - SetAudioVoITLBConfigResp, [759](#)
 - unpack_dms_SLQSSwiGetFwUpdateStatus_t, [951](#)
- reserved
 - omaDmFotaTlvExt, [549](#)
- resetInfoInd
 - pack_dms_SLQSDmsSwiIndicationRegister_t, [557](#)
- ResetInfoNotification
 - qaGobiApiCbK.h, [1333](#)
- ResetPDSDData
 - qaGobiApiPds.h, [1555](#)
- ResetToFactoryDefaults
 - qaGobiApiDms.h, [1433](#)
- RespFieldsList, [726](#)
 - responseFields, [726](#)
 - responseFieldsLen, [726](#)
- responseFields
 - RespFieldsList, [726](#)

- responseFieldsLen
 - RespFieldsList, [726](#)
- results
 - QmiNasGetRFBandInfoResp, [708](#)
 - QmiNasPerformNetworkScanResp, [708](#)
- RetryCount
 - unpack_swioma_SLQSOMADMGetSessionInfo←_t, [1050](#)
- revPolarity
 - lineCtrlInfo, [351](#)
- revTunneling
 - unpack_wds_GetMobileIPProfile_t, [1066](#)
- ReverseMac
 - protocolSubtypeElement, [676](#)
- RmTrasnferStaticsReq
 - pack_wds_RMSetTransferStatistics_t, [622](#)
- rmTrasnferStaticsReq, [727](#)
 - bResetStatistics, [728](#)
 - ulMask, [728](#)
- RmtPtyNum
 - arrRemotePartyNum, [118](#)
- roamIndList, [728](#)
 - numInstances, [728](#)
 - radiolInterface, [728](#)
 - roamIndicator, [729](#)
- roamIndicator
 - nas_roamIndList, [466](#)
 - roamIndList, [729](#)
- RoamIndicatorVal
 - unpack_nas_SLQSGetServingSystem_t, [996](#)
- roamIndicatorVal
 - qaQmiServingSystemParam, [683](#)
- roamOrigVoiceSO
 - prefVoiceSO, [663](#)
- RoamPref
 - NASRoamPreferenceTlv, [521](#)
- roamStatus
 - nas_sysInfoCommon, [477](#)
 - sysInfoCommon, [851](#)
- roamStatusValid
 - nas_sysInfoCommon, [477](#)
 - sysInfoCommon, [851](#)
- roamTimer, [729](#)
 - namID, [729](#)
 - roamTimerValue, [729](#)
- roamTimerValue
 - roamTimer, [729](#)
- Roaming
 - nas_QmiNas3GppNetworkInfo, [462](#)
 - SlqsNas3GppNetworkInfo, [792](#)
 - unpack_nas_GetCDMANetworkParameters_t, [982](#)
 - unpack_nas_GetServingNetwork_t, [986](#)
- roaming
 - unpack_nas_SetRoamingIndicatorCallback_ind←_t, [991](#)
- roaming_ind
 - RoamingInfo, [729](#)
- RoamingIndicatorList
 - qaQmiServingSystemParam, [683](#)
 - unpack_nas_SLQSGetServingSystem_t, [996](#)
- RoamingInfo, [729](#)
 - roaming_ind, [729](#)
 - TlvPresent, [729](#)
- routeList
 - smsSetRoutesReq, [820](#)
- routeStorage
 - smsRouteEntry, [819](#)
- rptRate
 - LTESigRptCfg, [391](#)
 - LTESigRptConfig, [391](#)
 - nas_LTESigRptConfig, [448](#)
- rscp
 - nas_UMTSInfo, [484](#)
 - rxInfo, [735](#)
 - TDSCDMASigInfoExt, [855](#)
 - tdscdmaSigInfoExt, [855](#)
 - UMTSInfo, [915](#)
- rsrp
 - cellParams, [160](#)
 - LTESSInfo, [394](#)
 - lteSSInfo, [395](#)
 - nas_RxSigInfo, [468](#)
 - nas_cellParams, [416](#)
 - nas_umtsLTENbrCell, [486](#)
 - rxInfo, [735](#)
 - RxSigInfo, [738](#)
 - umtsLTENbrCell, [917](#)
- rsrplevel
 - lteRsrpinformation, [387](#)
 - nas_lteRsrpinformation, [445](#)
- rsrq
 - cellParams, [160](#)
 - LTESSInfo, [394](#)
 - lteSSInfo, [395](#)
 - nas_SccRxInfo, [469](#)
 - nas_cellParams, [416](#)
 - nas_rsrqInformation, [467](#)
 - nas_umtsLTENbrCell, [486](#)
 - rsrqInformation, [731](#)
 - SccRxInfo, [743](#)
 - umtsLTENbrCell, [917](#)
- rsrqDelta
 - nas_SLQSSignalStrengthsIndReq, [473](#)
 - SLQSSignalStrengthsIndReq, [801](#)
- rsrqInfo
 - nas_SLQSSignalStrengthsInformation, [474](#)
 - SLQSSignalStrengthsInformation, [803](#)
 - slqsSignalStrengthInfo, [799](#)
 - unpack_nas_SLQSGetSignalStrength_t, [997](#)
- rsrqInformation, [730](#)
 - radiolf, [731](#)
 - rsrq, [731](#)
- rsri
 - CDMASSInfo, [154](#)
 - cdmaSSInfo, [154](#)
 - cellParams, [160](#)

- gsmCellInfo, [279](#)
- HDRSSInfo, [291](#)
- hdrSSInfo, [292](#)
- LTESSInfo, [394](#)
- lteSSInfo, [395](#)
- nas_cellParams, [416](#)
- nas_gsmCellInfo, [427](#)
- TDSCDMASigInfoExt, [855](#)
- tdscdmaSigInfoExt, [855](#)
- unpack_nas_GetSignalStrengths_t, [987](#)
- rts
 - WdsRunTimeSettings, [1190](#)
- rx_bytes
 - NetStats, [533](#)
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- rx_errors
 - NetStats, [533](#)
- rx_overflows
 - NetStats, [533](#)
- rx_packets
 - NetStats, [533](#)
- rx_pkts
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- rxChainIndex
 - nas_RxSigInfo, [468](#)
 - RxSigInfo, [738](#)
- RxDropConutTlv
 - QmiCbkWdsStatisticsIndState, [705](#)
 - unpack_RMTransferStatistics_ind_t, [1041](#)
- rxInfo, [734](#)
 - ecio, [735](#)
 - isRadioTuned, [735](#)
 - phase, [735](#)
 - rscp, [735](#)
 - rsrp, [735](#)
 - rxPower, [735](#)
- rxLev
 - GERANInfo, [237](#)
 - nas_GERANInfo, [425](#)
- rxOKBytesCount
 - unpack_wds_SLQSGetDUNCallInfo_t, [1076](#)
- RxOkByteCountTlv
 - QmiCbkWdsStatisticsIndState, [705](#)
 - unpack_RMTransferStatistics_ind_t, [1041](#)
- RxOkConutTlv
 - QmiCbkWdsStatisticsIndState, [705](#)
 - unpack_RMTransferStatistics_ind_t, [1041](#)
- rxPower
 - nas_RxSigInfo, [468](#)
 - rxInfo, [735](#)
 - RxSigInfo, [738](#)
- RxQFilter
 - unpack_qos_QosFlowInfo_t, [1023](#)
- RxQFlowGranted
 - unpack_qos_QosFlowInfo_t, [1023](#)
- RxSigInfo, [737](#)
- isRadioTuned, [738](#)
- rsrp, [738](#)
- rxChainIndex, [738](#)
- rxPower, [738](#)
- rxSignalStrength
 - nas_rxSignalStrengthListElement, [468](#)
 - rxSignalStrengthListElement, [739](#)
- rxSignalStrengthDelta
 - nas_SLQSSignalStrengthsIndReq, [473](#)
 - SLQSSignalStrengthsIndReq, [801](#)
- rxSignalStrengthInfo
 - nas_SLQSSignalStrengthsInformation, [474](#)
 - SLQSSignalStrengthsInformation, [803](#)
- rxSignalStrengthList
 - slqsSignalStrengthInfo, [799](#)
 - unpack_nas_SLQSGetSignalStrength_t, [997](#)
- rxSignalStrengthListElement, [738](#)
- radiolf, [739](#)
- rxSignalStrength, [739](#)
- rxSignalStrengthListLen
 - slqsSignalStrengthInfo, [799](#)
 - unpack_nas_SLQSGetSignalStrength_t, [997](#)
- s
 - qmifwinfo_s, [706](#)
- sApnExtraParams, [739](#)
 - ambr_dl, [740](#)
 - ambr_dl_ext, [740](#)
 - ambr_dl_ext2, [740](#)
 - ambr_ul, [740](#)
 - ambr_ul_ext, [740](#)
 - ambr_ul_ext2, [740](#)
 - apnId, [740](#)
- SCI
 - unpack_nas_GetCDMANetworkParameters_t, [982](#)
- SCM
 - unpack_nas_GetCDMANetworkParameters_t, [982](#)
- SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL↔_XID
 - common.h, [1197](#)
- SDK_VALIDATE_INPUT_PACK_PARAM
 - common.h, [1197](#)
- SDPTlv
 - NASQmiCbkNasSystemSelPrefInd, [521](#)
- SDU_HDR_LEN
 - common.h, [1197](#)
- SECOND_INSTANCE
 - qaGobiApiCbk.h, [1328](#)
- sGetDeviceSeriesResult, [779](#)
 - eDevice, [780](#)
 - uResult, [780](#)
- SHORT
 - SwiDataTypes.h, [1769](#)
- SIGSTRENGTH_THRESHOLD_ARR_SZ
 - qaGobiApiCbk.h, [1328](#)
- SITlv
 - QmiCbkSwiOmaDmEventStatusReportInd, [703](#)
 - QmiCbkSwiOmaDmEventStatusReportIndExt, [704](#)
- sIntraSearch

- LTEInfoIntraFreq, [384](#)
- nas_LTEInfoIntraFreq, [443](#)
- SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT↔
_LENGTH
dms.h, [1208](#)
- SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER↔
LENGTH
dms.h, [1208](#)
- SLQS_SS_INFO_LIST_MAX_ELEMENTS
qaGobiApiNas.h, [1499](#)
- SLQS_SYSTEM_ID_SIZE
qaGobiApiNas.h, [1499](#)
- SLQSAutoConnect
qaGobiApiWds.h, [1724](#)
- SLQSCDMADecodeMTTextMsg
qaGobiApiSms.h, [1584](#)
- SLQSCDMAEncodeMOTextMsg
qaGobiApiSms.h, [1584](#)
- SLQSConfigSigInfo
qaGobiApiNas.h, [1527](#)
- SLQSCreateProfile
qaGobiApiWds.h, [1724](#)
- SLQSDeleteProfile
qaGobiApiWds.h, [1725](#)
- SLQSDeleteProfileParams, [788](#)
profileIndex, [789](#)
profileType, [789](#)
- SLQSDeleteSMS
qaGobiApiSms.h, [1585](#)
- SLQSDmsSwiGetResetInfo
qaGobiApiDms.h, [1434](#)
- SLQSDmsSwiIndicationRegister
qaGobiApiDms.h, [1434](#)
- SLQSDownloadFirmwareToSlot
qaGobiApiFms.h, [1466](#)
- SLQSFWINFO_APPVERSION_SZ
qaGobiApiFms.h, [1459](#)
- SLQSFWINFO_BOOTVERSION_SZ
qaGobiApiFms.h, [1459](#)
- SLQSFWINFO_CARRIER_SZ
qaGobiApiFms.h, [1459](#)
- SLQSFWINFO_CUR_CARR_NAME
qaGobiApiFms.h, [1459](#)
- SLQSFWINFO_CUR_CARR_REV
qaGobiApiFms.h, [1459](#)
- SLQSFWINFO_MODELID_SZ
qaGobiApiFms.h, [1459](#)
- SLQSFWINFO_PACKAGEID_SZ
qaGobiApiFms.h, [1460](#)
- SLQSFWINFO_PRIVERSION_SZ
qaGobiApiFms.h, [1460](#)
- SLQSFWINFO_SKU_SZ
qaGobiApiFms.h, [1460](#)
- SLQSGet3GPPConfigItem
qaGobiApiWds.h, [1725](#)
- SLQSGetAGPSConfig
qaGobiApiPds.h, [1559](#)
- SLQSGetAudioPathConfig
qaGobiApiAudio.h, [1313](#)
- SLQSGetAudioProfile
qaGobiApiAudio.h, [1314](#)
- SLQSGetAudioVolTLBConfig
qaGobiApiAudio.h, [1314](#)
- SLQSGetBandCapabilities
qaGobiApiDms.h, [1435](#)
- SLQSGetBandCapability
qaGobiApiDms.h, [1435](#)
- SLQSGetBootVersionNumber
qaGobiApiFms.h, [1467](#)
- SLQSGetByteTotals
qaGobiApiWds.h, [1726](#)
- SLQSGetConnectionRate
qaGobiApiWds.h, [1726](#)
- SLQSGetCurrDataSystemStat
qaGobiApiWds.h, [1727](#)
- SLQSGetCurrentChannelRate
qaGobiApiWds.h, [1727](#)
- SLQSGetCurrentPRLInfo
qaGobiApiDms.h, [1437](#)
- SLQSGetCustFeatures
qaGobiApiDms.h, [1437](#)
- SLQSGetCustFeaturesV2
qaGobiApiDms.h, [1438](#)
- SLQSGetDUNCallInfo
qaGobiApiWds.h, [1729](#)
- SLQSGetDataBearerTechnology
qaGobiApiWds.h, [1728](#)
- SLQSGetDataBearerTechnologyExt
qaGobiApiWds.h, [1728](#)
- SLQSGetDeviceMode
qaGobiApiDcs.h, [1410](#)
- SLQSGetERIFile
qaGobiApiDms.h, [1438](#)
- SLQSGetErrorRate
qaGobiApiNas.h, [1527](#)
- SLQSGetFirmwareInfo
qaGobiApiFms.h, [1467](#)
- SLQSGetGPSStateInfo
qaGobiApiPds.h, [1559](#)
- SLQSGetIMSARegStatus
qaGobiApiImsa.h, [1483](#)
- SLQSGetIMSAServiceStatus
qaGobiApiImsa.h, [1484](#)
- SLQSGetIMSASupportedFields
qaGobiApiImsa.h, [1484](#)
- SLQSGetIMSASupportedMsg
qaGobiApiImsa.h, [1485](#)
- SLQSGetIMSSMSConfig
qaGobiApiIms.h, [1477](#)
- SLQSGetIMSUserConfig
qaGobiApiIms.h, [1477](#)
- SLQSGetIMSVoIPConfig
qaGobiApiIms.h, [1478](#)
- SLQSGetImageInfo
qaGobiApiFms.h, [1468](#)
- SLQSGetImageInfo_9x15

- qaGobiApiFms.h, [1468](#)
- SLQSGetImageInfoMC77xx
 - qaGobiApiFms.h, [1469](#)
- SLQSGetImageInfoMC83xx
 - qaGobiApiFms.h, [1470](#)
- SLQSGetIndicationRegister
 - qaGobiApiSms.h, [1586](#)
- SLQSGetM2MAVMute
 - qaGobiApiSwiAudio.h, [1602](#)
- SLQSGetM2MAudioProfile
 - qaGobiApiSwiAudio.h, [1601](#)
- SLQSGetM2MAudioVolume
 - qaGobiApiSwiAudio.h, [1602](#)
- SLQSGetM2MSpkrGain
 - qaGobiApiSwiAudio.h, [1603](#)
- SLQSGetMessageWaiting
 - qaGobiApiSms.h, [1586](#)
- SLQSGetNetStatistic
 - qaGobiApiDcs.h, [1411](#)
- SLQSGetNetworkTime
 - qaGobiApiNas.h, [1527](#)
- SLQSGetOperatorNameData
 - qaGobiApiNas.h, [1528](#)
- SLQSGetPLMNName
 - qaGobiApiNas.h, [1528](#)
- SLQSGetPacketStatistics
 - qaGobiApiWds.h, [1729](#)
- SLQSGetPidof
 - qaGobiApiSwi.h, [1599](#)
- SLQSGetProfile
 - qaGobiApiWds.h, [1730](#)
- SLQSGetProfileSettings
 - qaGobiApiWds.h, [1731](#)
- SLQSGetRegMgrConfig
 - qaGobiApiIms.h, [1478](#)
- SLQSGetRfSarState
 - qaGobiApiSar.h, [1574](#)
- SLQSGetRuntimeSettings
 - qaGobiApiWds.h, [1732](#)
- SLQSGetSIPConfig
 - qaGobiApiIms.h, [1479](#)
- SLQSGetSMSList
 - qaGobiApiSms.h, [1588](#)
- SLQSGetSMS
 - qaGobiApiSms.h, [1587](#)
- SLQSGetSdkVersion
 - qaGobiApiSwi.h, [1600](#)
- SLQSGetSerialNumbers
 - qaGobiApiDms.h, [1438](#)
- SLQSGetServingSystem
 - qaGobiApiNas.h, [1529](#)
- SLQSGetSessionState
 - qaGobiApiWds.h, [1732](#)
- SLQSGetSignalStrength
 - qaGobiApiNas.h, [1530](#)
- SLQSGetSmsBroadcastConfig
 - qaGobiApiSms.h, [1588](#)
- SLQSGetSysSelectionPref
 - qaGobiApiNas.h, [1530](#)
- SLQSGetTransLayerInfo
 - qaGobiApiSms.h, [1589](#)
- SLQSGetTransNWRegInfo
 - qaGobiApiSms.h, [1590](#)
- SLQSGetUsbPortNames
 - qaGobiApiDcs.h, [1411](#)
- SLQSGetValidFwPriCombinations
 - qaGobiApiFms.h, [1470](#)
- SLQSImsConfigIndicationRegister
 - qaGobiApiIms.h, [1479](#)
- SLQSIInitiateNetworkRegistration
 - qaGobiApiNas.h, [1531](#)
- SLQSIIsSpkgFormatRequired
 - qaGobiApiFms.h, [1471](#)
- SLQSKillSDKProcess
 - qaGobiApiDcs.h, [1412](#)
- SLQSLOCDelAssData
 - qaGobiApiLoc.h, [1487](#)
- SLQSLOCEventRegister
 - qaGobiApiLoc.h, [1488](#)
- SLQSLOCGetBestAvailPos
 - qaGobiApiLoc.h, [1488](#)
- SLQSLOCInjectPosition
 - qaGobiApiLoc.h, [1489](#)
- SLQSLOCInjectSensorData
 - qaGobiApiLoc.h, [1489](#)
- SLQSLOCInjectUTCTime
 - qaGobiApiLoc.h, [1490](#)
- SLQSLOCSetCradleMountConfig
 - qaGobiApiLoc.h, [1490](#)
- SLQSLOCSetExtPowerState
 - qaGobiApiLoc.h, [1491](#)
- SLQSLOCSetOpMode
 - qaGobiApiLoc.h, [1491](#)
- SLQSLOCStart
 - qaGobiApiLoc.h, [1492](#)
- SLQSLOCStop
 - qaGobiApiLoc.h, [1492](#)
- SLQSModifyProfile
 - qaGobiApiWds.h, [1733](#)
- SLQSModifySMSStatus
 - qaGobiApiSms.h, [1590](#)
- SLQSNASGetLTECPHYCaInfo
 - qaGobiApiNas.h, [1533](#)
- SLQSNASSwiGetChannelLock
 - qaGobiApiNas.h, [1537](#)
- SLQSNASSwiSetChannelLock
 - qaGobiApiNas.h, [1538](#)
- SLQSNasConfigSigInfo2
 - qaGobiApiNas.h, [1531](#)
- SLQSNasGet3GPP2Subscription
 - qaGobiApiNas.h, [1532](#)
- SLQSNasGetCellLocationInfo
 - qaGobiApiNas.h, [1532](#)
- SLQSNasGetHDRColorCode
 - qaGobiApiNas.h, [1533](#)
- SLQSNasGetSigInfo

- qaGobiApiNas.h, [1533](#)
- SLQSNasGetSysInfo
 - qaGobiApiNas.h, [1534](#)
- SLQSNasGetTxRxInfo
 - qaGobiApiNas.h, [1534](#)
- SLQSNasIndicationRegister
 - qaGobiApiNas.h, [1535](#)
- SLQSNasIndicationRegisterExt
 - qaGobiApiNas.h, [1536](#)
- SLQSNasIndicationRegisterLTECphyCa
 - qaGobiApiNas.h, [1536](#)
- SLQSNasNetworkTimeCallBack
 - qaGobiApiCbK.h, [1381](#)
- SLQSNasSigInfo2CallBack
 - qaGobiApiCbK.h, [1381](#)
- SLQSNasSigInfoCallBack
 - qaGobiApiCbK.h, [1382](#)
- SLQSNasSwiIndicationRegister
 - qaGobiApiNas.h, [1537](#)
- SLQSNasSwiModemStatus
 - qaGobiApiNas.h, [1538](#)
- SLQSNasSwiOTAMessageCallback
 - qaGobiApiCbK.h, [1383](#)
- SLQSNasSysInfoCallBack
 - qaGobiApiCbK.h, [1383](#)
- SLQSNasTimerCallback
 - qaGobiApiCbK.h, [1384](#)
- SLQSOMADMCancelSession
 - qaGobiApiSwiOmadms.h, [1612](#)
- SLQSOMADMGetSessionInfo
 - qaGobiApiSwiOmadms.h, [1612](#)
- SLQSOMADMGetSettings
 - qaGobiApiSwiOmadms.h, [1613](#)
- SLQSOMADMGetSettings2
 - qaGobiApiSwiOmadms.h, [1613](#)
- SLQSOMADMSendSelection
 - qaGobiApiSwiOmadms.h, [1614](#)
- SLQSOMADMSendSelection2
 - qaGobiApiSwiOmadms.h, [1614](#)
- SLQSOMADMSessionInfo
 - qaGobiApiSwiOmadms.h, [1607](#)
- SLQSOMADMSetSettings
 - qaGobiApiSwiOmadms.h, [1615](#)
- SLQSOMADMSetSettings2
 - qaGobiApiSwiOmadms.h, [1616](#)
- SLQSOMADMSetSettings3
 - qaGobiApiSwiOmadms.h, [1616](#)
- SLQSOMADMSettings
 - qaGobiApiSwiOmadms.h, [1609](#)
- SLQSOMADMSettingsReqParams
 - qaGobiApiSwiOmadms.h, [1610](#)
- SLQSOMADMSettingsReqParams3
 - qaGobiApiSwiOmadms.h, [1611](#)
- SLQSOMADMStartSession
 - qaGobiApiSwiOmadms.h, [1616](#)
- SLQSOMADMStartSession2
 - qaGobiApiSwiOmadms.h, [1617](#)
- SLQSOriginateUSSD
 - qaGobiApiVoice.h, [1678](#)
- SLQSPDSDeterminePosition
 - qaGobiApiPds.h, [1560](#)
- SLQSPDSInjectAbsoluteTimeReference
 - qaGobiApiPds.h, [1560](#)
- SLQSPDSInjectPositionData
 - qaGobiApiPds.h, [1561](#)
- SLQSPPerformNetworkScan
 - qaGobiApiNas.h, [1539](#)
- SLQSQosGetFlowStatus
 - qaGobiApiQos.h, [1565](#)
- SLQSQosGetGranted
 - qaGobiApiQos.h, [1566](#)
- SLQSQosGetNWProf
 - qaGobiApiQos.h, [1567](#)
- SLQSQosGetNetworkStatus
 - qaGobiApiQos.h, [1566](#)
- SLQSQosModify
 - qaGobiApiQos.h, [1567](#)
- SLQSQosRel
 - qaGobiApiQos.h, [1567](#)
- SLQSQosReq
 - qaGobiApiQos.h, [1568](#)
- SLQSQosReset
 - qaGobiApiQos.h, [1569](#)
- SLQSQosResume
 - qaGobiApiQos.h, [1569](#)
- SLQSQosSuspend
 - qaGobiApiQos.h, [1570](#)
- SLQSQosSwiReadApnExtraParams
 - qaGobiApiQos.h, [1570](#)
- SLQSQosSwiReadDataStats
 - qaGobiApiQos.h, [1571](#)
- SLQSRegisterIMSAIndication
 - qaGobiApiImsa.h, [1486](#)
- SLQSResetPacketStatics
 - qaGobiApiWds.h, [1733](#)
- SLQSSGetDHCPv4ClientConfig
 - qaGobiApiWds.h, [1736](#)
- SLQSSGetLoopback
 - qaGobiApiWds.h, [1736](#)
- SLQSSSTlv
 - unpack_nas_SetEventReportInd_t, [989](#)
- SLQSSSetDHCPv4ClientConfig
 - qaGobiApiWds.h, [1736](#)
- SLQSSSetLoopback
 - qaGobiApiWds.h, [1737](#)
- SLQSSendAsyncSMS
 - qaGobiApiSms.h, [1591](#)
- SLQSSendLongSMS
 - qaGobiApiSms.h, [1591](#)
- SLQSSendRawQMI
 - qaGobiApiSwi.h, [1600](#)
- SLQSSendSMS
 - qaGobiApiSms.h, [1592](#)
- SLQSSet3GPPConfigItem
 - qaGobiApiWds.h, [1734](#)
- SLQSSetAGPSCConfig

- qaGobiApiPds.h, [1562](#)
- SLQSSetAudioPathConfig
 - qaGobiApiAudio.h, [1315](#)
- SLQSSetAudioProfile
 - qaGobiApiAudio.h, [1315](#)
- SLQSSetAudioVolTLBConfig
 - qaGobiApiAudio.h, [1316](#)
- SLQSSetBandPreference
 - qaGobiApiNas.h, [1539](#)
- SLQSSetBandPreferenceCbK
 - qaGobiApiCbK.h, [1384](#)
- SLQSSetCrashStateCheckIgnore
 - qaGobiApiFms.h, [1471](#)
- SLQSSetCustFeatures
 - qaGobiApiDms.h, [1439](#)
- SLQSSetCustFeaturesV2
 - qaGobiApiDms.h, [1439](#)
- SLQSSetDHCPv4ClientLeaseStatusCallback
 - qaGobiApiCbK.h, [1385](#)
- SLQSSetDUNCallInfoCallback
 - qaGobiApiCbK.h, [1385](#)
- SLQSSetDataSystemStatusCallback
 - qaGobiApiCbK.h, [1384](#)
- SLQSSetIMSAPdpStatusCallback
 - qaGobiApiCbK.h, [1386](#)
- SLQSSetIMSARegStatusCallback
 - qaGobiApiCbK.h, [1386](#)
- SLQSSetIMSARatStatusCallback
 - qaGobiApiCbK.h, [1387](#)
- SLQSSetIMSASvcStatusCallback
 - qaGobiApiCbK.h, [1387](#)
- SLQSSetIMSSMSConfig
 - qaGobiApiIms.h, [1480](#)
- SLQSSetIMSSMSConfigCallback
 - qaGobiApiCbK.h, [1388](#)
- SLQSSetIMSUserConfig
 - qaGobiApiIms.h, [1480](#)
- SLQSSetIMSUserConfigCallback
 - qaGobiApiCbK.h, [1388](#)
- SLQSSetIMSVolPConfig
 - qaGobiApiIms.h, [1481](#)
- SLQSSetIMSVolPConfigCallback
 - qaGobiApiCbK.h, [1389](#)
- SLQSSetIndicationRegister
 - qaGobiApiSms.h, [1593](#)
- SLQSSetLocInjectPositionCallback
 - qaGobiApiCbK.h, [1389](#)
- SLQSSetLocInjectUTCTimeCallback
 - qaGobiApiCbK.h, [1390](#)
- SLQSSetLoggingMask
 - qaGobiApiDcs.h, [1412](#)
- SLQSSetM2MAVMMute
 - qaGobiApiSwiAudio.h, [1605](#)
- SLQSSetM2MAudioAVCFG
 - qaGobiApiSwiAudio.h, [1603](#)
- SLQSSetM2MAudioLPBK
 - qaGobiApiSwiAudio.h, [1604](#)
- SLQSSetM2MAudioNVDef
 - qaGobiApiSwiAudio.h, [1604](#)
- SLQSSetM2MAudioProfile
 - qaGobiApiSwiAudio.h, [1604](#)
- SLQSSetM2MAudioVolume
 - qaGobiApiSwiAudio.h, [1605](#)
- SLQSSetM2MSpkrGain
 - qaGobiApiSwiAudio.h, [1606](#)
- SLQSSetModemTempCallback
 - qaGobiApiCbK.h, [1390](#)
- SLQSSetPacketSrvStatusCallback
 - qaGobiApiCbK.h, [1390](#)
- SLQSSetPositionMethodState
 - qaGobiApiPds.h, [1562](#)
- SLQSSetProfile
 - qaGobiApiWds.h, [1734](#)
- SLQSSetQosEventCallback
 - qaGobiApiCbK.h, [1391](#)
- SLQSSetQosNWStatusCallback
 - qaGobiApiCbK.h, [1391](#)
- SLQSSetQosPriEventCallback
 - qaGobiApiCbK.h, [1392](#)
- SLQSSetQosStatusCallback
 - qaGobiApiCbK.h, [1392](#)
- SLQSSetRegMgrConfig
 - qaGobiApiIms.h, [1482](#)
- SLQSSetRegMgrConfigCallback
 - qaGobiApiCbK.h, [1393](#)
- SLQSSetRfSarState
 - qaGobiApiSar.h, [1574](#)
- SLQSSetSDKTerminatedCallback
 - qaGobiApiCbK.h, [1393](#)
- SLQSSetSIMBasedImageSwitching
 - qaGobiApiFms.h, [1472](#)
- SLQSSetSIPConfig
 - qaGobiApiIms.h, [1482](#)
- SLQSSetSIPConfigCallback
 - qaGobiApiCbK.h, [1395](#)
- SLQSSetSMSEventCallback
 - qaGobiApiCbK.h, [1396](#)
- SLQSSetServingSystemCallback
 - qaGobiApiCbK.h, [1394](#)
- SLQSSetSessionStateCallback
 - qaGobiApiCbK.h, [1394](#)
- SLQSSetSignalStrengthsCallback
 - qaGobiApiCbK.h, [1395](#)
- SLQSSetSmsBroadcastActivation
 - qaGobiApiSms.h, [1593](#)
- SLQSSetSmsBroadcastConfig
 - qaGobiApiSms.h, [1594](#)
- SLQSSetSmsStorage
 - qaGobiApiSms.h, [1595](#)
- SLQSSetSpkgFormatRequired
 - qaGobiApiFms.h, [1472](#)
- SLQSSetSwiGetResetInfoCallback
 - qaGobiApiCbK.h, [1396](#)
- SLQSSetSwiHDRPersCallback
 - qaGobiApiCbK.h, [1396](#)
- SLQSSetSysSelectionPref

- qaGobiApiNas.h, [1541](#)
- SLQSSetSysSelectionPrefCallBack
 - qaGobiApiCbK.h, [1397](#)
- SLQSSetTransLayerInfoCallback
 - qaGobiApiCbK.h, [1397](#)
- SLQSSetTransNWRegInfoCallback
 - qaGobiApiCbK.h, [1398](#)
- SLQSSetWdsEventCallback
 - qaGobiApiCbK.h, [1398](#)
- SLQSSetWdsTransferStatisticCallback
 - qaGobiApiCbK.h, [1399](#)
- SLQSSignalStrengthsIndReq, [800](#)
 - ecioDelta, [801](#)
 - ecioThresholdList, [801](#)
 - ecioThresholdListLen, [801](#)
 - ioDelta, [801](#)
 - lteRsrpDelta, [801](#)
 - lteSnrDelta, [801](#)
 - rsrqDelta, [801](#)
 - rxSignalStrengthDelta, [801](#)
 - sinrDelta, [801](#)
 - sinrThresholdList, [802](#)
 - sinrThresholdListLen, [802](#)
- SLQSSignalStrengthsInformation, [802](#)
 - ecioInfo, [803](#)
 - errorRateInfo, [803](#)
 - io, [803](#)
 - lteRsrpinfo, [803](#)
 - lteSnrinfo, [803](#)
 - rsrqInfo, [803](#)
 - rxSignalStrengthInfo, [803](#)
 - sinr, [803](#)
- SLQSSmsGetMaxStorageSize
 - qaGobiApiSms.h, [1595](#)
- SLQSSmsGetMessageProtocol
 - qaGobiApiSms.h, [1596](#)
- SLQSSmsSetRoutes
 - qaGobiApiSms.h, [1596](#)
- SLQSSstart
 - qaGobiApiDcs.h, [1413](#)
- SLQSSstart_AVAgent
 - qaGobiApiDcs.h, [1413](#)
- SLQSSstartSrv
 - qaGobiApiDcs.h, [1414](#)
- SLQSSstartStopDataSession
 - qaGobiApiWds.h, [1737](#)
- SLQSSwiClearDyingGaspStatistics
 - qaGobiApiDms.h, [1440](#)
- SLQSSwiGetAllCarrierImages
 - qaGobiApiFms.h, [1472](#)
- SLQSSwiGetCrashAction
 - qaGobiApiDms.h, [1440](#)
- SLQSSwiGetCrashInfo
 - qaGobiApiDms.h, [1440](#)
- SLQSSwiGetDyingGaspCfg
 - qaGobiApiDms.h, [1441](#)
- SLQSSwiGetDyingGaspStatistics
 - qaGobiApiDms.h, [1441](#)
- SLQSSwiGetFSN
 - qaGobiApiDms.h, [1441](#)
- SLQSSwiGetFirmwareCurr
 - qaGobiApiFms.h, [1473](#)
- SLQSSwiGetFwUpdateStatus
 - qaGobiApiDms.h, [1442](#)
- SLQSSwiGetHDRPersonality
 - qaGobiApiNas.h, [1541](#)
- SLQSSwiGetHDRProtSubtype
 - qaGobiApiNas.h, [1542](#)
- SLQSSwiGetHRPDStats
 - qaGobiApiNas.h, [1542](#)
- SLQSSwiGetHostDevInfo
 - qaGobiApiDms.h, [1442](#)
- SLQSSwiGetHostDevInfoParams
 - qaGobiApiDms.h, [1421](#)
- SLQSSwiGetLteCQI
 - qaGobiApiNas.h, [1543](#)
- SLQSSwiGetLteSccRxInfo
 - qaGobiApiNas.h, [1543](#)
- SLQSSwiGetOSInfo
 - qaGobiApiDms.h, [1443](#)
- SLQSSwiGetOSInfoParams
 - qaGobiApiDms.h, [1422](#)
- SLQSSwiGetSMSStorage
 - qaGobiApiSms.h, [1597](#)
- SLQSSwiGetSerialNoExt
 - qaGobiApiDms.h, [1443](#)
- SLQSSwiGetSerialNoExtParams
 - qaGobiApiDms.h, [1422](#)
- SLQSSwiGetUSBComp
 - qaGobiApiDms.h, [1444](#)
- SLQSSwiNetworkDebug
 - qaGobiApiNas.h, [1544](#)
- SLQSSwiPSDetach
 - qaGobiApiNas.h, [1544](#)
- SLQSSwiSetCrashAction
 - qaGobiApiDms.h, [1444](#)
- SLQSSwiSetDyingGaspCfg
 - qaGobiApiDms.h, [1445](#)
- SLQSSwiSetHostDevInfo
 - qaGobiApiDms.h, [1445](#)
- SLQSSwiSetHostDevInfoParams
 - qaGobiApiDms.h, [1423](#)
- SLQSSwiSetOSInfo
 - qaGobiApiDms.h, [1446](#)
- SLQSSwiSetOSInfoParams
 - qaGobiApiDms.h, [1423](#)
- SLQSSwiSetUSBComp
 - qaGobiApiDms.h, [1446](#)
- SLQSTmdDeRegNotMitigationLvl
 - qaGobiApiTmd.h, [1657](#)
- SLQSTmdGetMitigationDevList
 - qaGobiApiTmd.h, [1657](#)
- SLQSTmdGetMitigationLvl
 - qaGobiApiTmd.h, [1657](#)
- SLQSTmdMitigationLvlRptCallback
 - qaGobiApiCbK.h, [1400](#)

- SLQSTmdRegNotMitigationLvl
 - qaGobiApiTmd.h, [1658](#)
- SLQSUIMAuthenticate
 - qaGobiApiUim.h, [1661](#)
- SLQSUIMChangePin
 - qaGobiApiUim.h, [1662](#)
- SLQSUIMDepersonalization
 - qaGobiApiUim.h, [1662](#)
- SLQSUIMEventRegister
 - qaGobiApiUim.h, [1663](#)
- SLQSUIMGetCardStatus
 - qaGobiApiUim.h, [1663](#)
- SLQSUIMGetConfiguration
 - qaGobiApiUim.h, [1664](#)
- SLQSUIMGetFileAttributes
 - qaGobiApiUim.h, [1664](#)
- SLQSUIMGetSlotsStatus
 - qaGobiApiUim.h, [1665](#)
- SLQSUIMGetState
 - qaGobiApiDms.h, [1447](#)
- SLQSUIMPowerDown
 - qaGobiApiUim.h, [1665](#)
- SLQSUIMPowerUp
 - qaGobiApiUim.h, [1666](#)
- SLQSUIMReadTransparent
 - qaGobiApiUim.h, [1666](#)
- SLQSUIMRefreshComplete
 - qaGobiApiUim.h, [1667](#)
- SLQSUIMRefreshGetLastEvent
 - qaGobiApiUim.h, [1668](#)
- SLQSUIMRefreshOK
 - qaGobiApiUim.h, [1668](#)
- SLQSUIMRefreshRegister
 - qaGobiApiUim.h, [1669](#)
- SLQSUIMReset
 - qaGobiApiUim.h, [1669](#)
- SLQSUIMSetPinProtection
 - qaGobiApiUim.h, [1670](#)
- SLQSUIMSetRefreshCallBack
 - qaGobiApiCbk.h, [1400](#)
- SLQSUIMSetStatusChangeCallBack
 - qaGobiApiCbk.h, [1400](#)
- SLQSUIMSwitchSlot
 - qaGobiApiUim.h, [1670](#)
- SLQSUIMUnblockPin
 - qaGobiApiUim.h, [1671](#)
- SLQSUIMVerifyPin
 - qaGobiApiUim.h, [1672](#)
- SLQSUUpgradeFirmware9x15
 - qaGobiApiFms.h, [1473](#)
- SLQSVoiceALSSelectLine
 - qaGobiApiVoice.h, [1678](#)
- SLQSVoiceALSSetLineSwitching
 - qaGobiApiVoice.h, [1679](#)
- SLQSVoiceAnswerCall
 - qaGobiApiVoice.h, [1679](#)
- SLQSVoiceBindSubscription
 - qaGobiApiVoice.h, [1680](#)
- SLQSVoiceBurstDTMF
 - qaGobiApiVoice.h, [1680](#)
- SLQSVoiceDialCall
 - qaGobiApiVoice.h, [1681](#)
- SLQSVoiceEndCall
 - qaGobiApiVoice.h, [1681](#)
- SLQSVoiceGetAllCallInfo
 - qaGobiApiVoice.h, [1682](#)
- SLQSVoiceGetCLIP
 - qaGobiApiVoice.h, [1685](#)
- SLQSVoiceGetCLIR
 - qaGobiApiVoice.h, [1685](#)
- SLQSVoiceGetCNAP
 - qaGobiApiVoice.h, [1686](#)
- SLQSVoiceGetCOLP
 - qaGobiApiVoice.h, [1687](#)
- SLQSVoiceGetCOLR
 - qaGobiApiVoice.h, [1687](#)
- SLQSVoiceGetCallBarring
 - qaGobiApiVoice.h, [1682](#)
- SLQSVoiceGetCallForwardingStatus
 - qaGobiApiVoice.h, [1683](#)
- SLQSVoiceGetCallInfo
 - qaGobiApiVoice.h, [1684](#)
- SLQSVoiceGetCallWaiting
 - qaGobiApiVoice.h, [1684](#)
- SLQSVoiceGetConfig
 - qaGobiApiVoice.h, [1688](#)
- SLQSVoiceIndicationRegister
 - qaGobiApiVoice.h, [1688](#)
- SLQSVoiceInfoRecCallback
 - qaGobiApiCbk.h, [1401](#)
- SLQSVoiceManageCalls
 - qaGobiApiVoice.h, [1689](#)
- SLQSVoiceOrigUSSDNoWait
 - qaGobiApiVoice.h, [1690](#)
- SLQSVoiceSendFlash
 - qaGobiApiVoice.h, [1690](#)
- SLQSVoiceSetAllCallStatusCallBack
 - qaGobiApiCbk.h, [1401](#)
- SLQSVoiceSetCallBarringPassword
 - qaGobiApiVoice.h, [1691](#)
- SLQSVoiceSetConfig
 - qaGobiApiVoice.h, [1691](#)
- SLQSVoiceSetDTMFEventCallBack
 - qaGobiApiCbk.h, [1402](#)
- SLQSVoiceSetOTASPStatusCallBack
 - qaGobiApiCbk.h, [1402](#)
- SLQSVoiceSetPreferredPrivacy
 - qaGobiApiVoice.h, [1692](#)
- SLQSVoiceSetPrivacyChangeCallBack
 - qaGobiApiCbk.h, [1403](#)
- SLQSVoiceSetSUPSCallBack
 - qaGobiApiCbk.h, [1403](#)
- SLQSVoiceSetSUPSNotificationCallback
 - qaGobiApiCbk.h, [1404](#)
- SLQSVoiceSetSUPSService
 - qaGobiApiVoice.h, [1693](#)

- SLQSVoiceStartContDTMF
 - qaGobiApiVoice.h, [1693](#)
- SLQSVoiceStopContDTMF
 - qaGobiApiVoice.h, [1694](#)
- SLQSWCDMADecodeLongTextMsg
 - qaGobiApiSms.h, [1597](#)
- SLQSWCDMADecodeMTTextMsg
 - qaGobiApiSms.h, [1598](#)
- SLQSWCDMAEncodeMOTextMsg
 - qaGobiApiSms.h, [1598](#)
- SLQSWdsGoActive
 - qaGobiApiWds.h, [1738](#)
- SLQSWdsGoDormant
 - qaGobiApiWds.h, [1738](#)
- SLQSWdsSetEventReport
 - qaGobiApiWds.h, [1739](#)
- SLQSWdsSwiPDPRuntimeSettings
 - qaGobiApiWds.h, [1739](#)
- SLQSWmsAsyncRawSendCallBack
 - qaGobiApiCbK.h, [1404](#)
- SLQSWmsMemoryFullCallBack
 - qaGobiApiCbK.h, [1405](#)
- SLQSWmsMessageWaitingCallBack
 - qaGobiApiCbK.h, [1405](#)
- SMS_EVENT_ETWS_PLMN
 - qaGobiApiCbK.h, [1365](#)
- SMS_EVENT_ETWS
 - qaGobiApiCbK.h, [1365](#)
- SMS_EVENT_MESSAGE_MODE
 - qaGobiApiCbK.h, [1365](#)
- SMS_EVENT_MT_MESSAGE
 - qaGobiApiCbK.h, [1365](#)
- SMS_EVENT_SMS_ON_IMS
 - qaGobiApiCbK.h, [1365](#)
- SMS_EVENT_SMSC_ADDRESS
 - qaGobiApiCbK.h, [1365](#)
- SMS_EVENT_TRANSFER_ROUTE_MT_MESSAGE
 - qaGobiApiCbK.h, [1365](#)
- SMSAsyncRawSend
 - qaGobiApiCbK.h, [1334](#)
- SMSAsyncRawSend_s, [805](#)
 - alphaIDLen, [806](#)
 - causeCode, [806](#)
 - errorClass, [806](#)
 - messageID, [806](#)
 - msgDelFailureCause, [806](#)
 - msgDelFailureType, [806](#)
 - pAlphaID, [806](#)
 - RPCause, [806](#)
 - sendStatus, [806](#)
 - TPCause, [806](#)
 - userData, [806](#)
- SMSCAddress, [807](#)
 - data, [808](#)
 - length, [808](#)
- sMSCAddress, [807](#)
 - data, [807](#)
 - length, [807](#)
- SMSCAddressInfo
 - qaGobiApiCbK.h, [1335](#)
- sMSCAddressInfo
 - sms.h, [1760](#)
- sMSCAddressTlv, [808](#)
 - SMSCInfo, [808](#)
 - TlvPresent, [808](#)
- SMSCInfo
 - sMSCAddressTlv, [808](#)
- SMSCtlv
 - unpack_sms_SetNewSMSCallback_ind_t, [1043](#)
- SMSEtwsMessage, [809](#)
 - data, [809](#)
 - length, [809](#)
 - notificationType, [810](#)
- sMSEtwsMessage, [808](#)
 - data, [809](#)
 - length, [809](#)
 - notificationType, [809](#)
- SMSEtwsMessageInfo
 - qaGobiApiCbK.h, [1335](#)
- sMSEtwsMessageInfo
 - sms.h, [1760](#)
- sMSEtwsMessageTlv, [810](#)
 - EtwsMessageInfo, [810](#)
 - TlvPresent, [810](#)
- SMSEtwsPlmn, [811](#)
 - mobileCountryCode, [811](#)
 - mobileNetworkCode, [811](#)
- sMSEtwsPlmn, [810](#)
 - mobileCountryCode, [811](#)
 - mobileNetworkCode, [811](#)
- SMSEtwsPlmnInfo
 - qaGobiApiCbK.h, [1336](#)
- sMSEtwsPlmnInfo
 - sms.h, [1761](#)
- SMSEventInfo
 - qaGobiApiCbK.h, [1336](#)
- SMSEventInfo_s, [811](#)
 - pEtwsMessageInfo, [812](#)
 - pEtwsPlmnInfo, [812](#)
 - pMTMessageInfo, [812](#)
 - pMessageModelInfo, [812](#)
 - pSMSCAddressInfo, [812](#)
 - pSMSOnIMSInfo, [812](#)
 - pTransferRouteMTMessageInfo, [813](#)
 - smsEventType, [813](#)
- SMSEventType
 - qaGobiApiCbK.h, [1364](#)
- SMSMTMessage, [816](#)
 - messageIndex, [817](#)
 - storageType, [817](#)
- sSMSMTMessage, [816](#)
 - messageIndex, [816](#)
 - storageType, [816](#)
- SMSMTMessageInfo
 - qaGobiApiCbK.h, [1337](#)
- sSMSMTMessageInfo

- sms.h, [1761](#)
- SMSMemoryInfo, [814](#)
 - messageMode, [814](#)
 - storageType, [814](#)
- SMSMessageMode, [815](#)
 - messageMode, [815](#)
- sSMSMessageMode, [815](#)
 - messageMode, [815](#)
- SMSMessageModelInfo
 - qaGobiApiCbk.h, [1337](#)
- sSMSMessageModelInfo
 - sms.h, [1761](#)
- SMSOnIMSInfo
 - qaGobiApiCbk.h, [1337](#)
- sMSOnIMSInfo
 - sms.h, [1761](#)
- sMSOnIMSTlv, [818](#)
 - IMSInfo, [818](#)
 - TlvPresent, [818](#)
- SMSOnIMS, [817](#)
 - smsOnIMS, [817](#)
- sMSOnIMS, [817](#)
 - smsOnIMS, [818](#)
- SMSSupport
 - pack_dms_SetCustFeature_t, [555](#)
 - unpack_dms_GetCustFeature_t, [927](#)
- SMSTransferRouteMTMessage, [821](#)
 - ackIndicator, [822](#)
 - data, [822](#)
 - format, [822](#)
 - length, [822](#)
 - transactionID, [822](#)
- sMSTransferRouteMTMessage, [820](#)
 - ackIndicator, [821](#)
 - data, [821](#)
 - format, [821](#)
 - length, [821](#)
 - transactionID, [821](#)
- SMSTransferRouteMTMessageInfo
 - qaGobiApiCbk.h, [1338](#)
- sMSTransferRouteMTMessageInfo
 - sms.h, [1761](#)
- sNonIntraSearch
 - LTEInfoIntrafreq, [384](#)
 - nas_LTEInfoIntrafreq, [443](#)
- SNR
 - NetworkStatEVDO, [538](#)
- SOMask
 - CurrNetworkInfo, [186](#)
 - currNetworkInfo, [187](#)
 - wds_currNetworkInfo, [1160](#)
- SPC_SIZE
 - dms.h, [1208](#)
- SPKG_FIRMWARE_DOWNLOAD
 - qaGobiApiFms.h, [1460](#)
- sPhyCaAggPcellInfo
 - nasGetLTECphyCa, [498](#)
 - QmiCbkNasLTECphyCaInfo, [703](#)
- unpack_nas_SetNasLTECphyCaIndCallback_↔
 - ind_t, [989](#)
- sPhyCaAggScellIDBw
 - nasGetLTECphyCa, [498](#)
 - QmiCbkNasLTECphyCaInfo, [703](#)
 - unpack_nas_SetNasLTECphyCaIndCallback_↔
 - ind_t, [989](#)
- sPhyCaAggScellIndType
 - nasGetLTECphyCa, [498](#)
 - QmiCbkNasLTECphyCaInfo, [703](#)
 - unpack_nas_SetNasLTECphyCaIndCallback_↔
 - ind_t, [989](#)
- sPhyCaAggScellIndex
 - nasGetLTECphyCa, [498](#)
 - QmiCbkNasLTECphyCaInfo, [703](#)
 - unpack_nas_SetNasLTECphyCaIndCallback_↔
 - ind_t, [989](#)
- sPhyCaAggScellInfo
 - nasGetLTECphyCa, [498](#)
 - QmiCbkNasLTECphyCaInfo, [703](#)
 - unpack_nas_SetNasLTECphyCaIndCallback_↔
 - ind_t, [989](#)
- sQosFlowStat, [822](#)
 - bearerId, [822](#)
 - tx_bytes, [822](#)
 - tx_bytes_drp, [823](#)
 - tx_pkt, [823](#)
 - tx_pkt_drp, [823](#)
- sQosStat, [823](#)
 - apnId, [824](#)
 - numQosFlow, [824](#)
 - qosFlow, [824](#)
 - total_rx_bytes, [824](#)
 - total_rx_pkt, [824](#)
 - total_tx_bytes, [824](#)
 - total_tx_bytes_drp, [824](#)
 - total_tx_pkt, [824](#)
 - total_tx_pkt_drp, [824](#)
- SSInfo
 - unpack_nas_SetServingSystemCallback_ind_t, [991](#)
- sSLQSSignalStrengthsInfo
 - nas_SLQSSignalStrengthsTlv, [474](#)
- SSTlv
 - unpack_nas_SetEventReportInd_t, [989](#)
- SUPInfo, [828](#)
 - isModByCC, [828](#)
 - svcType, [828](#)
- SUPInformation
 - voiceSUPInfo, [1145](#)
- SUPSType
 - voiceManageCallsReq, [1128](#)
- SVInfo, [829](#)
 - len, [830](#)
 - pSV, [830](#)
- SWI Audio Service(SWIAUDIO), [52](#)
- SWI Open Mobile Alliance Service (SWIOMA), [45](#)
- SWI_API

- SwiDataTypes.h, [1768](#)
- SWI_STRUCT_CarrierImage, [831](#)
 - m_FwBuildId, [831](#)
 - m_FwImagId, [831](#)
 - m_PriBuildId, [832](#)
 - m_PriImagId, [832](#)
 - m_nCarrierId, [832](#)
 - m_nFolderId, [832](#)
 - m_nStorage, [832](#)
- SWIWWANCMAPI.h, [1781](#)
- samplesPerBatch
 - accelAcceptReady_s, [96](#)
 - accelTempAcceptReady_s, [97](#)
 - gyroAcceptReady_s, [284](#)
 - gyroTempAcceptReady_s, [285](#)
- satelliteInfo, [740](#)
 - azimuth, [742](#)
 - elevation, [742](#)
 - gnssSvId, [742](#)
 - healthStatus, [742](#)
 - snr, [742](#)
 - svInfoMask, [742](#)
 - svListLen, [742](#)
 - svStatus, [742](#)
 - system, [742](#)
 - validMask, [742](#)
- SaveSMS
 - qaGobiApiSms.h, [1581](#)
- sbas_almanac_sv_msk
 - GPSSStateInfo, [277](#)
- sbas_ephemeris_sv_msk
 - GPSSStateInfo, [277](#)
- sbas_health_sv_msk
 - GPSSStateInfo, [277](#)
- sbas_visible_sv_msk
 - GPSSStateInfo, [277](#)
- SccRxInfo, [742](#)
 - numInstances, [743](#)
 - rsrq, [743](#)
 - sigInfo, [743](#)
 - snr, [743](#)
 - TlvPresent, [743](#)
- scell_idx
 - NASPhyCaAggScellIndex, [514](#)
 - nas_PhyCaAggScellIndex, [458](#)
 - PhyCaAggScellIndex, [652](#)
- scell_state
 - NASPhyCaAggScellIndType, [515](#)
 - NASPhyCaAggScellInfo, [516](#)
 - nas_PhyCaAggScellIndType, [458](#)
 - nas_PhyCaAggScellInfo, [461](#)
 - PhyCaAggScellIndType, [653](#)
 - PhyCaAggScellInfo, [655](#)
- screeningInd
 - connectNumInfo, [172](#)
- sduErrorRatio
 - LibPackUMTSQoS, [349](#)
 - UMTSMinQoS, [920](#)
 - UMTSQoS, [922](#)
 - wds_UMTSMinQoS, [1174](#)
- seDNSIPv4Address
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, [1085](#)
- seDNSIPv6Address
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, [1086](#)
- sePCSCFIPv4Address
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, [1086](#)
- sePCSCFIPv6Address
 - unpack_wds_SLQSWdsSwiPDPRuntimeSettings↔_t, [1086](#)
- secActivate
 - fileAttributes, [228](#)
- secActivateMask
 - fileAttributes, [228](#)
- secChA
 - CDMAChannel, [146](#)
- secChB
 - CDMAChannel, [146](#)
- secDeactivate
 - fileAttributes, [228](#)
- secDeactivateMask
 - fileAttributes, [228](#)
- secIncrease
 - fileAttributes, [228](#)
- secIncreaseMask
 - fileAttributes, [228](#)
- SecProt
 - protocolSubtypeElement, [677](#)
- secRead
 - fileAttributes, [228](#)
- secReadMask
 - fileAttributes, [228](#)
- secWrite
 - fileAttributes, [228](#)
- secWriteMask
 - fileAttributes, [229](#)
- secdns
 - unpack_wds_GetDefaultProfile_t, [1064](#)
- secdnsv6
 - unpack_wds_GetDefaultProfile_t, [1064](#)
- second
 - nas_UniversalTime, [487](#)
 - nas_timeInfo, [482](#)
 - timeInfo, [863](#)
 - UniversalTime, [925](#)
- SecondaryDNSV4
 - unpack_wds_SLQSGetRuntimeSettings_t, [1079](#)
- SecondaryDNSV6
 - unpack_wds_SLQSGetRuntimeSettings_t, [1079](#)
- secondaryDNS
 - pack_wds_SetDefaultProfile_t, [624](#)
- secondaryHA
 - unpack_wds_GetMobileIPProfile_t, [1067](#)
- SectorIDLen

- NetworkStatEVDO, 538
- selNetwork
 - nas_servSystem, 471
 - servSystem, 752
- selected
 - BroadcastConfig, 124
 - CDMABroadcastConfig, 146
- selectedNetwork
 - NASServingSystemInfo, 523
 - ServingSystemInfo, 751
- selection
 - pack_swiooma_SLQSOMADMSendSelection_↔
t, 610
- SendSMS
 - qaGobiApiSms.h, 1582
- sendStatus
 - SMSAsyncRawSend_s, 806
- sensorData, 743
 - flags, 744
 - sensorDataLen, 744
 - timeOfFirstSample, 744
 - timeOffset, 744
 - xAxis, 744
 - yAxis, 745
 - zAxis, 745
- sensorData_t, 745
 - flags, 746
 - sensorDataLen, 746
 - timeOfFirstSample, 746
 - timeOffset, 746
 - xAxis, 746
 - yAxis, 746
 - zAxis, 746
- sensorDataLen
 - sensorData, 744
 - sensorData_t, 746
- sensorDataUsage
 - qaGobiApiCbk.h, 1333
- sensorDataUsage_s, 746
 - aidingIndicatorMask, 747
 - usageMask, 747
- serialNumbersInfo, 747
 - esnSize, 748
 - imeiSize, 748
 - imeiSvnSize, 748
 - meidSize, 748
 - pESNString, 748
 - pIMEIString, 748
 - pImeiSvnString, 748
 - pMEIDString, 748
 - qaGobiApiDms.h, 1421
- servSystem, 751
 - csAttachState, 752
 - numRadiolInterfaces, 752
 - psAttachState, 752
 - radiolInterface, 752
 - regState, 752
 - selNetwork, 752
- ServerAddrList
 - unpack_wds_SLQSGetRuntimeSettings_t, 1079
- serviceCategory
 - CDMABroadcastConfig, 146
- serviceClassInformation
 - qaGobiApiVoice.h, 1676
- serviceDomain
 - nas_RejectReasonTlv, 464
- serviceProviderName, 749
 - displayCondition, 749
 - spn, 749
 - spnLength, 749
- servingCellId
 - LTEInfoIntrafreq, 384
 - nas_LTEInfoIntrafreq, 443
- ServingSystem
 - qaQmiServingSystemParam, 683
 - unpack_nas_SLQSGetServingSystem_t, 996
- ServingSystemInfo, 749
 - csAttachState, 750
 - hdrPersonality, 751
 - psAttachState, 751
 - radiolInterfaceList, 751
 - radiolInterfaceNo, 751
 - registrationState, 751
 - selectedNetwork, 751
- sessionEndReason
 - _packetSrvStatus, 64
 - slqsSessionStateInfo, 797
 - unpack_wds_SLQSSetPacketSrvStatusCallback↔
_t, 1081
- SessionId
 - LOCStartReq, 376
 - pack_loc_Start_t, 582
 - pack_loc_Stop_t, 582
- sessionId
 - LOCStopReq, 376
 - QmiCbkLocPositionReportInd, 701
 - ssdatasession_params, 827
 - unpack_loc_PositionRpt_Ind_t, 977
- sessionInfo, 752
 - omaDmConfig, 753
 - omaDmFota, 753
 - omaDmNotifications, 753
 - pack_uim_ChangePin_t, 612
 - pack_uim_ReadTransparent_t, 613
 - pack_uim_SetPinProtection_t, 614
 - pack_uim_UnblockPin_t, 618
 - pack_uim_VerifyPin_t, 619
 - sessionInfoTlv, 754
 - sessionInfoTlvExt, 754
 - UIMAuthenticateReq, 891
 - UIMChangePinReq, 892
 - UIMGetFileAttributesReq, 897
 - UIMReadTransparentReq, 901
 - UIMRefreshCompleteReq, 903
 - UIMRefreshGetLastEventReq, 905
 - UIMRefreshOKReq, 906

- UIMRefreshRegisterReq, 906
- UIMSetPinProtectionReq, 908
- UIMUnblockPinReq, 912
- UIMVerifyPinReq, 913
- SessionInfoConfig
 - unpack_swima_SLQSOMADMAAlertCallback_↔
ind_t, 1048
- sessionInfoExt, 753
 - omaDmConfig, 753
 - omaDmFota, 753
- SessionInfoFota
 - unpack_swima_SLQSOMADMAAlertCallback_↔
ind_t, 1048
- SessionInfoNotification
 - unpack_swima_SLQSOMADMAAlertCallback_↔
ind_t, 1048
- sessionInfoTlv, 753
 - sessionInfo, 754
 - sessionType, 754
 - TlvPresent, 754
- sessionInfoTlvExt, 754
 - sessionInfo, 754
 - sessionType, 754
 - TlvPresent, 754
- sessionInformation
 - qaGobiApiCbK.h, 1334
- sessionInformationExt
 - qaGobiApiCbK.h, 1334
- SessionState
 - unpack_swima_SLQSOMADMGetSessionInfo_↔
_t, 1050
- sessionStatus
 - omaDmNotificationsTlv, 549
 - QmiCbKLocPositionReportInd, 701
 - unpack_loc_PositionRpt_Ind_t, 977
 - unpack_omaDmNotificationsTlv_t, 1018
- SessionType
 - pack_swima_SLQSOMADMGetSessionInfo_t,
609
 - unpack_swima_SLQSOMADMGetSessionInfo_↔
_t, 1050
- sessionType
 - omaDmFotaTlv, 547
 - pack_swima_SLQSOMADMCancelSession_↔
t, 608
 - pack_swima_SLQSOMADMStartSession_t, 611
 - sessionInfoTlv, 754
 - sessionInfoTlvExt, 754
 - UIMRefreshEvent, 904
 - UIMSessionInformation, 907
 - uim_UIMSessionInformation, 888
 - uim_sessionInformation, 885
 - unpack_omaDmFotaTlv_t, 1017
- set_fix_rate
 - pack_swiloc_SwiLocSetAutoStart_t, 608
 - SwiLocSetAutoStartReq, 835
- set_fix_type
 - pack_swiloc_SwiLocSetAutoStart_t, 608
- SwiLocSetAutoStartReq, 835
- set_function
 - pack_swiloc_SwiLocSetAutoStart_t, 608
 - SwiLocSetAutoStartReq, 835
- set_max_dist
 - pack_swiloc_SwiLocSetAutoStart_t, 608
 - SwiLocSetAutoStartReq, 835
- set_max_time
 - pack_swiloc_SwiLocSetAutoStart_t, 608
 - SwiLocSetAutoStartReq, 835
- SetACCOLC
 - qaGobiApiNas.h, 1524
- SetActivationStatusCallback
 - qaGobiApiCbK.h, 1365
- SetActiveMobileIPProfile
 - qaGobiApiWds.h, 1715
- SetAudioPathConfigReq, 754
 - pCodecSTGain, 756
 - pDTMFTXGain, 756
 - pECMode, 756
 - pNSEnable, 756
 - pRXAGCList, 756
 - pRXAVCAGCSwitch, 756
 - pRXAVCList, 756
 - pRXPCMIIRFtr, 756
 - pTXAGCList, 756
 - pTXAVCSwitch, 756
 - pTXGain, 756
 - pTXPCMIIRFtr, 756
 - Profile, 756
- SetAudioProfileReq, 756
 - EarMute, 757
 - Generator, 757
 - MicMute, 757
 - Profile, 757
 - Volume, 757
- SetAudioVolTLBConfigReq, 758
 - Generator, 758
 - Item, 758
 - Profile, 758
 - VolValue, 759
 - Volume, 758
- SetAudioVolTLBConfigResp, 759
 - ResCode, 759
- SetAutoconnect
 - qaGobiApiWds.h, 1715
- SetCATEventCallback
 - qaGobiApiCbK.h, 1366
- SetCDMANetworkParameters
 - qaGobiApiNas.h, 1524
- setCustomSettingV2, 759
 - cust_id, 760
 - cust_value, 760
 - value_length, 760
- SetDataCapabilitiesCallback
 - qaGobiApiCbK.h, 1367
- SetDefaultProfile
 - qaGobiApiWds.h, 1716

- SetDefaultProfileLTEV2
 - qaGobiApiWds.h, [1719](#)
- SetDefaultProfileLTE
 - qaGobiApiWds.h, [1717](#)
- SetDefaultProfileNum
 - qaGobiApiWds.h, [1720](#)
- SetDeviceStateChangeCb
 - qaGobiApiCb.h, [1367](#)
- setDyingGaspCfg, [760](#)
 - pDestSMSContent, [760](#)
 - pDestSMSNum, [760](#)
- SetFwDldCompletionCb
 - qaGobiApiCb.h, [1368](#)
- SetGPSCallback
 - qaGobiApiCb.h, [1368](#)
- SetIMSSMSConfigReq, [760](#)
 - pPhoneCtxtURLen, [761](#)
 - pPhoneCtxtURI, [761](#)
 - pSMSFormat, [761](#)
 - pSMSOverIPNwInd, [761](#)
- SetIMSSMSConfigResp, [761](#)
 - pSettingResp, [762](#)
- SetIMSUserConfigReq, [762](#)
 - pIMSDomain, [762](#)
 - pIMSDomainLen, [762](#)
- SetIMSUserConfigResp, [762](#)
 - pSettingResp, [763](#)
- SetIMSVoIPConfigReq, [763](#)
 - pAmrMode, [764](#)
 - pAmrOctetAligned, [765](#)
 - pAmrWBMode, [765](#)
 - pAmrWBOctetAligned, [765](#)
 - pAmrWbEnable, [765](#)
 - pMinSessionExpiryTimer, [765](#)
 - pRTPRTCPInactTimer, [765](#)
 - pRingBackTimer, [765](#)
 - pRingingTimer, [765](#)
 - pScrAmrEnable, [765](#)
 - pScrAmrWbEnable, [765](#)
 - pSessionExpiryTimer, [765](#)
- SetIMSVoIPConfigResp, [765](#)
 - pSettingResp, [765](#)
- SetImagesPreference
 - qaGobiApiFms.h, [1465](#)
- setIndicationRegReq
 - qaGobiApiSms.h, [1580](#)
- SetLURectCallback
 - qaGobiApiCb.h, [1372](#)
- SetLocBestAvailPosCallback
 - qaGobiApiCb.h, [1368](#)
- SetLocCradleMountCallback
 - qaGobiApiCb.h, [1369](#)
- SetLocDeleteAssistDataCallback
 - qaGobiApiCb.h, [1369](#)
- SetLocEngineStateCallback
 - qaGobiApiCb.h, [1369](#)
- SetLocEventPositionCallback
 - qaGobiApiCb.h, [1370](#)
- SetLocEventTimeSyncCallback
 - qaGobiApiCb.h, [1370](#)
- SetLocGnssSvInfoCallback
 - qaGobiApiCb.h, [1370](#)
- SetLocInjectSensorDataCallback
 - qaGobiApiCb.h, [1371](#)
- SetLocInjectTimeCallback
 - qaGobiApiCb.h, [1371](#)
- SetLocOpModeCallback
 - qaGobiApiCb.h, [1371](#)
- SetLocSensorStreamingCallback
 - qaGobiApiCb.h, [1372](#)
- SetLocSetExtPowerConfigCallback
 - qaGobiApiCb.h, [1372](#)
- SetM2MAVMuteReq, [769](#)
 - EarMute, [769](#)
 - MicMute, [769](#)
 - pCwtMute, [769](#)
 - Profile, [769](#)
- SetM2MAudioAVCFGReq, [766](#)
 - Device, [766](#)
 - PIFACEId, [766](#)
 - pPCMPParams, [766](#)
 - Profile, [766](#)
- SetM2MAudioLPBKReq, [766](#)
 - Enable, [767](#)
- SetM2MAudioProfileReq, [767](#)
 - pCwtMute, [768](#)
 - pEarMute, [768](#)
 - pGenerator, [768](#)
 - pMicMute, [768](#)
 - pVolume, [768](#)
 - Profile, [768](#)
- SetM2MAudioVolumeReq, [768](#)
 - Generator, [769](#)
 - Level, [769](#)
 - Profile, [769](#)
- SetM2MSprkGainReq, [770](#)
 - Profile, [770](#)
 - Value, [770](#)
- SetMobileIPParameters
 - qaGobiApiWds.h, [1721](#)
- SetMobileIPProfile
 - qaGobiApiWds.h, [1722](#)
- SetMobileIPStatusCallback
 - qaGobiApiCb.h, [1373](#)
- SetMobileIP
 - qaGobiApiWds.h, [1721](#)
- SetNMEACallback
 - qaGobiApiCb.h, [1375](#)
- SetNasLTECphyCalIndCallback
 - qaGobiApiCb.h, [1373](#)
- SetNetChangeCb
 - qaGobiApiCb.h, [1374](#)
- SetNetworkPreference
 - qaGobiApiNas.h, [1526](#)
- SetNewSMSCallback
 - qaGobiApiCb.h, [1374](#)

- SetOMADMStateCallback
 - qaGobiApiCbk.h, [1375](#)
- SetPDSDefaults
 - qaGobiApiPds.h, [1555](#)
- SetPDSState
 - qaGobiApiPds.h, [1556](#)
- SetPDSStateCallback
 - qaGobiApiCbk.h, [1376](#)
- setPINProtection, [770](#)
 - pinID, [771](#)
 - pinLength, [771](#)
 - pinOperation, [771](#)
 - pinValue, [771](#)
- SetPortAutomaticTracking
 - qaGobiApiPds.h, [1557](#)
- SetPower
 - qaGobiApiDms.h, [1434](#)
- SetPowerCallback
 - qaGobiApiCbk.h, [1376](#)
- SetRFInfoCallback
 - qaGobiApiCbk.h, [1376](#)
- SetRMTransferStatisticsCallback
 - qaGobiApiCbk.h, [1377](#)
- SetRankIndicatorCallback
 - qaGobiApiCbk.h, [1376](#)
- SetRegMgrConfigReq, [771](#)
 - pCSCFPortName, [772](#)
 - pCSCFPortNameLen, [772](#)
 - pIMSTestMode, [772](#)
 - pPriCSCFPort, [772](#)
- SetRegMgrConfigResp, [772](#)
 - pSettingResp, [772](#)
- SetRoamingIndicatorCallback
 - qaGobiApiCbk.h, [1377](#)
- SetSDKImagePath
 - qaGobiApiDcs.h, [1410](#)
- SetSIPConfigReq, [778](#)
 - pSIPLocalPort, [779](#)
 - pSigCompEnabled, [779](#)
 - pSubscribeTimer, [779](#)
 - pTimerSIPReg, [779](#)
 - pTimerT1, [779](#)
 - pTimerT2, [779](#)
 - pTimerTf, [779](#)
- SetSIPConfigResp, [779](#)
 - pSettingResp, [779](#)
- SetSLQSOMADMAAlertCallback
 - qaGobiApiCbk.h, [1378](#)
- SetSLQSOMADMAAlertCallbackExt
 - qaGobiApiCbk.h, [1379](#)
- SetSMSCAddress
 - qaGobiApiSms.h, [1583](#)
- SetSMSWake
 - qaGobiApiRms.h, [1572](#)
- SetServiceAutomaticTracking
 - qaGobiApiPds.h, [1557](#)
- SetSignalStrengthCallback
 - qaGobiApiCbk.h, [1378](#)
- setSignalStrengthInfo, [773](#)
 - pCDMAECIODelta, [776](#)
 - pCDMAECIOThresh, [776](#)
 - pCDMARSSIDelta, [776](#)
 - pCDMARSSIOThresh, [776](#)
 - pGSMRSSIDelta, [776](#)
 - pGSMRSSIOThresh, [776](#)
 - pHDRECIODelta, [776](#)
 - pHDRECIOTThresh, [776](#)
 - pHDRIODelta, [777](#)
 - pHDRIOTThresh, [777](#)
 - pHDDRSSIDelta, [777](#)
 - pHDDRSSIOThresh, [777](#)
 - pHDSINRDelta, [777](#)
 - pHDSINRThresh, [777](#)
 - pLTERSRPDelta, [777](#)
 - pLTERSRPThresh, [777](#)
 - pLTERSRQDelta, [777](#)
 - pLTERSRQThresh, [777](#)
 - pLTERSSIDelta, [777](#)
 - pLTERSSIOThresh, [777](#)
 - pLTESNRDelta, [777](#)
 - pLTESNRThresh, [777](#)
 - pLTESigRptConfig, [777](#)
 - pTDSCDMAECIODelta, [777](#)
 - pTDSCDMAECIOThresh, [777](#)
 - pTDSCDMARSCPDelta, [777](#)
 - pTDSCDMARSCPThresh, [777](#)
 - pTDSCDMARSSIDelta, [777](#)
 - pTDSCDMARSSIOThresh, [777](#)
 - pTDSCDMASINRDelta, [777](#)
 - pTDSCDMASINRThresh, [777](#)
 - pWCDMAECIODelta, [778](#)
 - pWCDMAECIOThresh, [778](#)
 - pWCDMARSSIDelta, [778](#)
 - pWCDMARSSIOThresh, [778](#)
- SetUSSDNoWaitIndicationCallback
 - qaGobiApiCbk.h, [1380](#)
- SetUSSDNotificationCallback
 - qaGobiApiCbk.h, [1380](#)
- SetUSSDReleaseCallback
 - qaGobiApiCbk.h, [1381](#)
- SetUimSlotStatusChangeCallback
 - qaGobiApiCbk.h, [1379](#)
- SetXTRAAutomaticDownload
 - qaGobiApiPds.h, [1558](#)
- SetXTRANetwork
 - qaGobiApiPds.h, [1558](#)
- SetupEventList
 - CatEventListTlv, [144](#)
- Severity
 - unpack_swima_SLQSOMADMGetSessionInfo↔
_t, [1050](#)
- severity
 - omaDmFotaTlv, [547](#)
 - unpack_omaDmFotaTlv_t, [1017](#)
- Short Message Service (SMS), [38](#)
- shortName

- nasPLMNNNameResp, [519](#)
- PLMNNetworkNameData, [659](#)
- unpack_nas_SLQSGetPLMNNName_t, [993](#)
- shortNameCI
 - nasPLMNNNameResp, [519](#)
 - unpack_nas_SLQSGetPLMNNName_t, [993](#)
- shortNameEn
 - nasPLMNNNameResp, [519](#)
 - unpack_nas_SLQSGetPLMNNName_t, [993](#)
- shortNameLen
 - nasPLMNNNameResp, [519](#)
 - PLMNNetworkNameData, [659](#)
 - unpack_nas_SLQSGetPLMNNName_t, [993](#)
- shortNameSB
 - nasPLMNNNameResp, [520](#)
 - unpack_nas_SLQSGetPLMNNName_t, [993](#)
- shortNameSpareBits
 - PLMNNetworkNameData, [659](#)
- SI
 - callFWExtInfo, [133](#)
 - calledPartyInfo, [129](#)
 - callingPartyInfo, [138](#)
 - redirNumInfo, [722](#)
- sid
 - CDMAInfo, [148](#)
 - nas_CDMAInfo, [410](#)
 - sidNid, [780](#)
 - unpack_nas_GetHomeNetwork_t, [983](#)
- SidNid
 - homeSIDNID, [295](#)
- sidNid, [780](#)
 - nid, [780](#)
 - sid, [780](#)
- SigInd
 - LibPackUMTSReqQoSSigInd, [350](#)
 - UMTSReqQoSSigInd, [923](#)
- sigInfo, [780](#)
 - nas_SccRxInfo, [469](#)
 - pECIOThresh, [782](#)
 - pHRSINRThresh, [782](#)
 - pLOThresh, [782](#)
 - pLTESNRThresh, [782](#)
 - pLTESigRptCfg, [782](#)
 - pRSRPThresh, [782](#)
 - pRSRQThresh, [782](#)
 - pRSSIThresh, [782](#)
 - pTDSCDMASINRCONFTThresh, [782](#)
 - SccRxInfo, [743](#)
- signal
 - signalInfo, [783](#)
- signalInfo, [782](#)
 - alertPitch, [782](#)
 - signal, [783](#)
 - signalType, [783](#)
- signalStrength
 - nas_SignalStrengthTlv, [472](#)
- SignalStrengthDataType, [783](#)
 - thresholds, [783](#)
 - thresholdsSize, [783](#)
- signalStrengthReqMask
 - slqsSignalStrengthInfo, [799](#)
 - unpack_nas_SLQSGetSignalStrength_t, [997](#)
- signalType
 - signalInfo, [783](#)
- SimCapability
 - unpack_dms_GetDeviceCap_t, [928](#)
- simCapability
 - unpack_dms_GetDeviceCapabilities_t, [929](#)
- sinr
 - HDRSSInfo, [291](#)
 - hdrSSInfo, [292](#)
 - nas_SLQSSignalStrengthsInformation, [474](#)
 - SLQSSignalStrengthsInformation, [803](#)
 - slqsSignalStrengthInfo, [799](#)
 - TDSCDMASigInfoExt, [855](#)
 - tdscdmaSigInfoExt, [856](#)
 - unpack_nas_SLQSGetSignalStrength_t, [997](#)
- sinrDelta
 - nas_SLQSSignalStrengthsIndReq, [473](#)
 - SLQSSignalStrengthsIndReq, [801](#)
- sinrThresholdList
 - nas_SLQSSignalStrengthsIndReq, [473](#)
 - SLQSSignalStrengthsIndReq, [802](#)
- sinrThresholdListLen
 - nas_SLQSSignalStrengthsIndReq, [473](#)
 - SLQSSignalStrengthsIndReq, [802](#)
- sku_str
 - slqsfwinfno_s, [790](#)
 - unpack_dms_GetFirmwareInfo_t, [931](#)
- slot
 - pack_uim_SLQSUIMPowerDown_t, [615](#)
 - pack_uim_SLQSUIMPowerUp_t, [616](#)
 - UIMPowerDownReq, [899](#)
 - UIMPowerUpReq, [900](#)
- slot_t, [783](#)
 - bCCCIDLength, [784](#)
 - bCCCID, [784](#)
 - bLogicalSlot, [784](#)
 - uPhyCardStatus, [784](#)
 - uPhySlotStatus, [784](#)
- slotInf, [784](#)
 - AppStatus, [786](#)
 - cardState, [786](#)
 - errorState, [786](#)
 - numApp, [786](#)
 - upinRetries, [786](#)
 - upinState, [786](#)
 - upukRetries, [786](#)
- SlotInfo
 - cardStatus, [139](#)
 - uim_cardStatus, [879](#)
- slotInfo, [786](#)
 - AppStatus, [787](#)
 - cardState, [787](#)
 - errorState, [787](#)
 - numApp, [787](#)

- upinRetries, 787
- upinState, 787
- upukRetries, 787
- slots_t, 787
 - uimSlotStatus, 788
- slotsstatusChange
 - UIMSlotStatusChangeInfo, 910
 - unpack_uim_SetUimSlotStatusChangeCallback↔_ind_t, 1057
- slqs3GPPConfigItem
 - qaGobiApiWds.h, 1700
- SlqsNas3GppNetworkInfo, 791
 - Description, 792
 - Forbidden, 792
 - InUse, 792
 - MCC, 792
 - MNC, 792
 - Preferred, 792
 - Roaming, 792
- SlqsNas3GppNetworkRAT
 - qaGobiApiNas.h, 1499
- SlqsNasPcsDigit, 792
 - includes_pcs_digit, 792
 - MCC, 792
 - MNC, 793
- slqsNetworkScanInfo
 - qaGobiApiNas.h, 1500
- SlqsProfile3GPP2
 - unpackWdsProfileParam, 1086
 - wds_profileInfo, 1171
 - WdsProfileParam, 1189
- SlqsProfile3GPP
 - unpackWdsProfileParam, 1086
 - wds_profileInfo, 1171
 - WdsProfileParam, 1189
- slqsSessionStateInfo, 796
 - pQmiInterfaceInfo, 797
 - reconfiguration_required, 797
 - sessionEndReason, 797
 - state, 797
- slqsSignalStrengthInfo, 797
 - ecioList, 799
 - ecioListLen, 799
 - errorRateList, 799
 - errorRateListLen, 799
 - lo, 799
 - ltersrp, 799
 - ltesnr, 799
 - rsrqInfo, 799
 - rxSignalStrengthList, 799
 - rxSignalStrengthListLen, 799
 - signalStrengthReqMask, 799
 - sinr, 799
- slqsWdsEventInfo, 803
 - pDataBearer, 804
 - pDormancyStatus, 804
 - pPacketsCountRX, 804
 - pPacketsCountTX, 804
 - pQmiInterfaceInfo, 805
 - pTotalBytesRX, 805
 - pTotalBytesTX, 805
- slqsautoconnect, 788
 - acroamsetting, 788
 - acsetting, 788
 - action, 788
- slqsfwinfo_s, 789
 - appversion_str, 790
 - bootversion_str, 790
 - carrier_str, 790
 - cur_carr_name, 790
 - cur_carr_rev, 790
 - modelid_str, 790
 - packageid_str, 790
 - priversion_str, 790
 - sku_str, 790
- slqssendasyncsmsparams_s, 793
 - messageFormat, 794
 - messageSize, 794
 - pFollowOnDC, 794
 - pForceOnDC, 794
 - pLinktimer, 794
 - pMessage, 794
 - pRetryMessage, 794
 - pRetryMessageId, 795
 - pServiceOption, 795
 - pSmsOnIms, 795
 - pUserData, 795
- slqssendsmsparams_s, 795
 - messageFailureCode, 796
 - messageFormat, 796
 - messageID, 796
 - messageSize, 796
 - pLinktimer, 796
 - pMessage, 796
 - pSmsOnIms, 796
- sms.h, 1758
 - eqmiCbKSetStatus, 1762
 - LIBPACK_QMI_CBK_PARAM_NOCHANGE, 1762
 - LIBPACK_QMI_CBK_PARAM_RESET, 1762
 - LIBPACK_QMI_CBK_PARAM_SET, 1762
 - MAX_CDMA_ENC_MO_TXT_MSG_SIZE, 1760
 - MAX_MS_TRANSFER_ROUTE_MSG, 1760
 - MAX_MSC_ADDRESS_SIZE, 1760
 - MAX_MSE_TWS_MSG, 1760
 - MAX_SMS_LIST_SIZE, 1760
 - MAX_SMS_MESSAGE_SIZE, 1760
 - pack_sms_SLQSDeleteSMS, 1763
 - pack_sms_SLQSGetSMSList, 1764
 - pack_sms_SLQSGetSMS, 1763
 - pack_sms_SLQSModifySMSStatus, 1764
 - pack_sms_SendSMS, 1762
 - pack_sms_SetNewSMSCallback, 1762
 - sMSCAddressInfo, 1760
 - sMSEtwSMSMessageInfo, 1760
 - sMSEtwSPIMnInfo, 1761
 - sMSMTMessageInfo, 1761

- sMSMessageModelInfo, 1761
- sMSOnIMSInfo, 1761
- sMSTransferRouteMTMessageInfo, 1761
- unpack_sms_SLQSDDeleteSMS, 1766
- unpack_sms_SLQSGetSMSList, 1766
- unpack_sms_SLQSGetSMS, 1766
- unpack_sms_SLQSModifySMSStatus, 1767
- unpack_sms_SLQSWmsMemoryFullCallback_ind, 1767
- unpack_sms_SendSMS, 1764
- unpack_sms_SetNewSMSCallback, 1765
- unpack_sms_SetNewSMSCallback_ind, 1765
- smsEventType
 - SMSEventInfo_s, 813
- smsMaxStorageSizeReq, 813
 - pMessageMode, 813
 - storageType, 813
- smsMaxStorageSizeResp, 813
 - freeSlots, 814
 - maxStorageSize, 814
- smsMsgprotocolResp, 815
 - msgProtocol, 816
- smsOnIMS
 - SMSOnIMS, 817
 - sMSOnIMS, 818
- smsRouteEntry, 818
 - messageClass, 819
 - messageType, 819
 - receiptAction, 819
 - routeStorage, 819
- smsSetRoutesReq, 819
 - numOfRoutes, 820
 - pTransferStatusReport, 820
 - routeList, 820
- snr
 - LTESSInfo, 394
 - loc_satelliteInfo, 360
 - lteSSInfo, 395
 - nas_SccRxInfo, 469
 - satelliteInfo, 742
 - SccRxInfo, 743
- snrlevel
 - lteSnrinformation, 392
 - nas_lteSnrinformation, 449
- SO
 - NetworkStat1x, 536
- soMask
 - DataBearerTech, 195
 - dataBearerTechnology, 197
 - qmiWSDDataBearerTechnology, 709
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, 1082
- Source
 - unpack_swioma_SLQSOMADMGetSessionInfo←_t, 1050
- source
 - _getResetInfoNotification, 60
 - altSrcInfo_t, 105
 - altitudeSrcInfo, 104
 - dmsSwiGetResetInfo, 214
 - unpack_dms_GetNetworkTime_t, 935
 - unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 942
 - unpack_dms_SLQSDmsSwiGetResetInfo_t, 943
- sourceIPMask
 - LibPackTFTIDParams, 347
 - TFTIDParams, 861
- SourceLength
 - unpack_swioma_SLQSOMADMGetSessionInfo←_t, 1050
- spc
 - pack_dms_ResetToFactoryDefaults_t, 553
 - pack_nas_SetACCOLC_t, 583
 - pack_wds_SetMobileIPProfile_t, 627
- Specific Absorption Rate (SAR), 44
- spn
 - nasPLMNNNameResp, 520
 - serviceProviderName, 749
 - unpack_nas_SLQSGetPLMNNName_t, 993
- spnEncoding
 - nasPLMNNNameResp, 520
 - unpack_nas_SLQSGetPLMNNName_t, 993
- spnLength
 - nasPLMNNNameResp, 520
 - serviceProviderName, 749
 - unpack_nas_SLQSGetPLMNNName_t, 993
- srcPortRangeEnd
 - LibPackTFTIDParams, 347
 - TFTIDParams, 861
- srcPortRangeStart
 - LibPackTFTIDParams, 347
 - TFTIDParams, 861
- srvCapability
 - detailSvcInfo, 205
 - nas_detailSvcInfo, 421
 - nas_sysInfoCommon, 477
 - sysInfoCommon, 851
- srvCapabilityValid
 - nas_sysInfoCommon, 477
 - sysInfoCommon, 851
- srvDomain
 - nas_sysInfoCommon, 477
 - sysInfoCommon, 851
- SrvDomainPref
 - NASServDomainPrefTlv, 522
- srvDomainValid
 - nas_sysInfoCommon, 478
 - sysInfoCommon, 852
- srvOption
 - arrSvcOption, 119
- srvStatus
 - detailSvcInfo, 205
 - GSMSrvStatusInfo, 281
 - nas_GSMSrvStatusInfo, 428
 - nas_SrvStatusInfo, 475
 - nas_detailSvcInfo, 421

- SrvStatusInfo, 825
- SrvStatusInfo, 824
 - isPrefDataPath, 825
 - srvStatus, 825
- srxlev
 - cellParams, 160
 - gsmCellInfo, 279
 - nas_cellParams, 416
 - nas_gsmCellInfo, 427
 - nas_umtsLTENbrCell, 486
 - nas_wcdmaCellInfo, 488
 - umtsLTENbrCell, 917
 - wcdmaCellInfo, 1148
- ssdatasession_params, 825
 - action, 826
 - failureReason, 826
 - failureReasonv4, 826
 - failureReasonv6, 826
 - instanceId, 826
 - ipfamily, 826
 - pAuthentication, 826
 - pPassword, 826
 - pProfileId3GPP2, 826
 - pProfileId3GPP, 826
 - pTechnology, 827
 - pUsername, 827
 - rcv4, 827
 - rcv6, 827
 - sessionId, 827
 - v4sessionId, 827
 - v6sessionId, 827
 - verbFailReason, 827
 - verbFailReasonType, 827
- stage
 - UIMRefreshEvent, 904
- StartPDSTrackingSessionExt
 - qaGobiApiPds.h, 1563
- State
 - NetworkStat1x, 536
 - NetworkStatEVDO, 538
- state
 - LocSetCradleMountReq, 374
 - omaDmConfigTlv, 543
 - omaDmConfigTlvExt, 545
 - omaDmFotaTlv, 547
 - omaDmFotaTlvExt, 549
 - pack_loc_SLQSLOCSetCradleMountConfig_t, 580
 - QosFlowInfoState, 718
 - QosMap, 719
 - slqsSessionStateInfo, 797
 - unpack_dms_GetActivationState_t, 926
 - unpack_dms_SLQSUIMGetState_t, 955
 - unpack_omaDmConfigTlv_t, 1015
 - unpack_omaDmFotaTlv_t, 1017
 - unpack_qos_QosFlowInfoState_t, 1024
- statmask
 - pack_wds_GetPacketStatus_t, 621
- StatsMask
 - TransferStatInd, 869
 - transferStatInd, 869
- statsMask
 - TrStatInd, 870
 - wds_TrStatInd, 1171
- StatsPeriod
 - TransferStatInd, 869
 - transferStatInd, 869
- statsPeriod
 - TrStatInd, 870
 - wds_TrStatInd, 1171
- Status
 - unpack_swima_SLQSOMADMGetSessionInfo↔_t, 1051
- status
 - delAssistDataStatus, 202
 - lteEARFCN, 378
 - ltePCI, 387
 - pack_sms_SetNewSMSCallback_t, 603
 - QmiCbkLocBestAvailPosInd, 689
 - QmiCbkLocInjectPositionInd, 692
 - QmiCbkLocInjectUTCTimeInd, 695
 - QmiCbkLocSetExtPowerConfigInd, 702
 - unpack_loc_BestAvailPos_Ind_t, 968
 - unpack_loc_DeleteAssistData_Ind_t, 969
 - unpack_loc_SetExtPowerConfig_Ind_t, 978
 - unpack_loc_SetOperationMode_Ind_t, 979
 - unpack_qos_SLQSSetQosNWStatusCallback↔_ind_t, 1028
 - unpack_qos_SLQSSetQosStatusCallback_ind_t, 1030
 - wcdmaUARFCN, 1158
- statusChange
 - UIMStatusChangeInfo, 911
- StopPDSTrackingSession
 - qaGobiApiPds.h, 1564
- storageIndex
 - FMSImageIdElement, 233
 - ImageIdElement, 297
- storageType
 - pack_sms_SLQSDeleteSMS_t, 604
 - pack_sms_SLQSGetSMS_t, 605
 - pack_sms_SLQSGetSMSList_t, 605
 - pack_sms_SLQSModifySMSStatus_t, 606
 - SMSMTMessage, 817
 - sSMSMTMessage, 816
 - SMSMemoryInfo, 814
 - smsMaxStorageSizeReq, 813
 - unpack_sms_SLQSWmsMemoryFullCallBack↔_ind_t, 1045
- String
 - unpack_dms_GetDeviceHardwareRev_t, 929
 - unpack_dms_GetDeviceMfr_t, 930
 - unpack_dms_GetFSN_t, 933
 - unpack_dms_UIMGetICCID_t, 956
- stringSize
 - unpack_dms_GetDeviceHardwareRev_t, 929
 - unpack_dms_GetDeviceMfr_t, 930

- unpack_dms_UIMGetICCID_t, 956
- subAddr
 - calledPartySubAdd, 130
- subAddrLen
 - calledPartySubAdd, 130
- subAddrType
 - calledPartySubAdd, 130
- subnetMask
 - IPv4Addr, 318
 - unpack_qos_IPv4Addr_t, 1019
- SubnetMaskV4
 - unpack_wds_SLQSSGetRuntimeSettings_t, 1079
- subsType
 - voiceBindSubscriptionInfo, 1095
- SupUSBComps
 - unpack_dms_GetUSBComp_t, 938
- SuppOA
 - CUGInfo, 179
- SuppPrefCUG
 - CUGInfo, 179
- supportedMsgLen
 - SupportedMsgList, 827
- SupportedMsgList, 827
 - supportedMsgLen, 827
 - supportedMsgs, 827
- supportedMsgs
 - SupportedMsgList, 827
- SV, 828
 - id, 829
 - mask, 829
 - system, 829
- svInfoMask
 - loc_satelliteInfo, 360
 - satelliteInfo, 742
- svListLen
 - loc_satelliteInfo, 360
 - satelliteInfo, 742
- svStatus
 - loc_satelliteInfo, 360
 - satelliteInfo, 742
- svUsedforFix
 - qaGobiApiCbK.h, 1338
- svUsedforFix_s, 830
 - gnssSvUsedList, 830
 - gnssSvUsedList_len, 830
- svc
 - pack_qmi_t, 600
- SvcClass
 - callFWExtInfo, 133
 - callFWInfo, 134
- SvcStatus
 - callFWExtInfo, 133
 - callFWInfo, 134
- svcType
 - ccSUPSType, 145
 - SUPSType, 828
- sw1
 - cardResult, 138
- uim_cardResult, 878
- sw2
 - cardResult, 138
 - uim_cardResult, 878
- swVerString
 - pack_dms_SLQSSwiSetHostDevInfo_t, 559
 - unpack_dms_SLQSSwiGetHostDevInfo_t, 952
- SwiDataTypes.h, 1768
 - BOOL, 1769
 - BYTE, 1769
 - CHAR, 1769
 - FLOAT, 1769
 - INT32, 1769
 - INT8, 1769
 - LPCSTR, 1769
 - QMI_NO_LTE_FW_SUPPORT, 1768
 - QMI_TLV_PLACEHOLDER, 1768
 - SHORT, 1769
 - SWI_API, 1768
 - ULONGLONG, 1769
 - ULONG, 1769
 - UNUSEDPARAM, 1768
 - USHORT, 1769
 - WORD, 1769
- SwiLocGetAutoStart
 - qaGobiApiLoc.h, 1493
- SwiLocGetAutoStartResp, 832
 - fix_rate, 833
 - fix_rate_reported, 833
 - fix_type, 833
 - fix_type_reported, 833
 - function, 833
 - function_reported, 833
 - max_dist, 833
 - max_dist_reported, 833
 - max_time, 833
 - max_time_reported, 833
- SwiLocSetAutoStart
 - qaGobiApiLoc.h, 1493
- SwiLocSetAutoStartReq, 834
 - fix_rate, 835
 - fix_type, 835
 - function, 835
 - max_dist, 835
 - max_time, 835
 - set_fix_rate, 835
 - set_fix_type, 835
 - set_function, 835
 - set_max_dist, 835
 - set_max_time, 835
- swiModemStatusResp, 835
 - commonInfo, 836
 - pLTEInfo, 836
- SwiOTAMsg
 - qaGobiApiCbK.h, 1339
- SwiOTAMsg_s, 836
 - data, 837
 - data_len, 837

- pLteNasRelInfo, [837](#)
 - pTime, [837](#)
 - type, [837](#)
- swiPDPRuntimeSettingsReq, [837](#)
 - contextId, [837](#)
 - contextType, [837](#)
- swiPDPRuntimeSettingsResp, [837](#)
 - pAPNName, [839](#)
 - pBearerId, [839](#)
 - pContextId, [839](#)
 - pIPv4Address, [839](#)
 - pIPv4GWAddress, [839](#)
 - pIPv6Address, [839](#)
 - pIPv6GWAddress, [839](#)
 - pPrDNSIPv4Address, [840](#)
 - pPrDNSIPv6Address, [840](#)
 - pPrPCSCFIPv4Address, [840](#)
 - pPrPCSCFIPv6Address, [840](#)
 - pSeDNSIPv4Address, [840](#)
 - pSeDNSIPv6Address, [840](#)
 - pSePCSCFIPv4Address, [840](#)
 - pSePCSCFIPv6Address, [840](#)
- swiQosFilter, [840](#)
 - index, [842](#)
 - pEspSpi, [842](#)
 - pIPv4DstAddr, [842](#)
 - pIPv4SrcAddr, [842](#)
 - pIPv6DstAddr, [842](#)
 - pIPv6Label, [842](#)
 - pIPv6SrcAddr, [842](#)
 - pIPv6TrafCls, [842](#)
 - pld, [842](#)
 - pNxtHdrProto, [842](#)
 - pPrecedence, [842](#)
 - pTCPDstPort, [842](#)
 - pTCPSrcPort, [842](#)
 - pTos, [842](#)
 - pTranDstPort, [842](#)
 - pTranSrcPort, [842](#)
 - pUDPDstPort, [842](#)
 - pUDPSrcPort, [842](#)
 - version, [842](#)
- swiQosFlow, [842](#)
 - index, [845](#)
 - p3GPP2Pri, [845](#)
 - p3GPPImCn, [845](#)
 - p3GPPResResidualBER, [845](#)
 - p3GPPSigInd, [845](#)
 - p3GPPTraHdlPri, [845](#)
 - pDataRate, [845](#)
 - pJitter, [845](#)
 - pLatency, [845](#)
 - pLteQci, [845](#)
 - pMaxAllowedPktSz, [845](#)
 - pMinPolicedPktSz, [845](#)
 - pPktErrRate, [846](#)
 - pProfileId3GPP2, [846](#)
 - pTokenBucket, [846](#)
 - pTrafficClass, [846](#)
- swiQosGranted, [846](#)
 - pRxFlow, [846](#)
 - pTxFlow, [846](#)
- swiQosIds, [846](#)
 - plds, [847](#)
 - sz, [847](#)
- swiQosModifyReq, [847](#)
 - id, [847](#)
 - pRxFilter, [847](#)
 - pRxFlow, [847](#)
 - pTxFilter, [847](#)
 - pTxFlow, [847](#)
- swiQosReq, [848](#)
 - index, [848](#)
 - pRxFilter, [848](#)
 - pRxFlow, [848](#)
 - pTxFilter, [848](#)
 - pTxFlow, [848](#)
- swiRMTrasferStaticsReq, [848](#)
 - bResetStatistics, [849](#)
 - ulMask, [849](#)
- swiloc.h, [1769](#)
 - pack_swiloc_SwiLocGetAutoStart, [1769](#)
 - pack_swiloc_SwiLocSetAutoStart, [1770](#)
 - unpack_swiloc_SwiLocGetAutoStart, [1770](#)
 - unpack_swiloc_SwiLocSetAutoStart, [1771](#)
- swioma.h, [1771](#)
 - LIBPACK_MAX_SWIOMA_STR_LEN, [1772](#)
 - pack_swioma_SLQSOMADMAAlertCallback, [1772](#)
 - pack_swioma_SLQSOMADMCancelSession, [1773](#)
 - pack_swioma_SLQSOMADMGetSessionInfo, [1773](#)
 - pack_swioma_SLQSOMADMGetSettings, [1774](#)
 - pack_swioma_SLQSOMADMSelectSendSelection, [1775](#)
 - pack_swioma_SLQSOMADMSetSettings, [1775](#)
 - pack_swioma_SLQSOMADMStartSession, [1776](#)
 - unpack_swioma_SLQSOMADMAAlertCallback, [1777](#)
 - unpack_swioma_SLQSOMADMAAlertCallback_ind, [1777](#)
 - unpack_swioma_SLQSOMADMCancelSession, [1778](#)
 - unpack_swioma_SLQSOMADMGetSessionInfo, [1778](#)
 - unpack_swioma_SLQSOMADMGetSettings, [1779](#)
 - unpack_swioma_SLQSOMADMSelectSendSelection, [1779](#)
 - unpack_swioma_SLQSOMADMSetSettings, [1780](#)
 - unpack_swioma_SLQSOMADMStartSession, [1780](#)
- switchOption
 - voiceALSSetLineSwitchInfo, [1095](#)
- sysInfoCDMA
 - CDMASysInfo, [158](#)
 - nas_CDMASysInfo, [414](#)
- sysInfoCommon, [849](#)
 - isSysForbidden, [851](#)

- isSysForbiddenValid, [851](#)
- roamStatus, [851](#)
- roamStatusValid, [851](#)
- srvCapability, [851](#)
- srvCapabilityValid, [851](#)
- srvDomain, [851](#)
- srvDomainValid, [852](#)
- sysInfoGSM
 - GSMSysInfo, [284](#)
 - nas_GSMSysInfo, [431](#)
- sysInfoHDR
 - HDRSysInfo, [294](#)
 - nas_HDRSysInfo, [436](#)
- sysInfoLTE
 - LTESysInfo, [397](#)
 - nas_LTESysInfo, [452](#)
- sysInfoWCDMA
 - nas_WCDMASysInfo, [494](#)
 - WCDMASysInfo, [1158](#)
- sysSelectPrefInfo
 - qaGobiApiNas.h, [1500](#)
- sysSelectPrefParams
 - qaGobiApiNas.h, [1505](#)
- system
 - loc_SV, [362](#)
 - loc_satelliteInfo, [360](#)
 - satelliteInfo, [742](#)
 - SV, [829](#)
- SystemID
 - qaQmiServingSystemParam, [683](#)
 - unpack_nas_SLQSGetServingSystem_t, [996](#)
- systemID
 - CDMASysInfo, [158](#)
 - nas_CDMASysInfo, [414](#)
- systemMode
 - CommInfo, [169](#)
 - nas_CommInfo, [417](#)
- sz
 - swiQosIds, [847](#)
- t3396_apn
 - nasTimers, [531](#)
 - unpack_nas_SLQSNasTimerCallback_ind_t, [1009](#)
- t3396_plmn_id
 - nasTimers, [531](#)
 - unpack_nas_SLQSNasTimerCallback_ind_t, [1009](#)
- t3396_val
 - nasTimers, [531](#)
 - unpack_nas_SLQSNasTimerCallback_ind_t, [1009](#)
- t_Sv, [852](#)
 - entries, [852](#)
 - len, [852](#)
- t_gpsTime, [852](#)
 - gpsTimeOfWeekMs, [852](#)
 - gpsWeek, [852](#)
- t_sensor, [852](#)
 - aidingIndicatorMask, [852](#)
 - usageMask, [852](#)
- TCPDstPort
 - unpack_qos_swiQosFilter_t, [1034](#)
- TCPSrcPort
 - unpack_qos_swiQosFilter_t, [1034](#)
- TDSCDMAECIOThresh, [852](#)
 - pTDSCDMAECIOThreshList, [853](#)
 - TDSCDMAECIOThreshListLen, [853](#)
- TDSCDMAECIOThreshListLen
 - nas_TDSCDMAECIOThresh, [478](#)
 - TDSCDMAECIOThresh, [853](#)
- TDSCDMARSCPTHresh, [853](#)
 - pTDSCDMARSCPTHreshList, [853](#)
 - TDSCDMARSCPTHreshListLen, [853](#)
- TDSCDMARSCPTHreshListLen
 - nas_TDSCDMARSCPTHresh, [479](#)
 - TDSCDMARSCPTHresh, [853](#)
- TDSCDMARSSIThresh, [854](#)
 - pTDSCDMARSSIThreshList, [854](#)
 - TDSCDMARSSIThreshListLen, [854](#)
- TDSCDMARSSIThreshListLen
 - nas_TDSCDMARSSIThresh, [479](#)
 - TDSCDMARSSIThresh, [854](#)
- TDSCDMASINRCONFTthresh, [856](#)
 - pTDSCDMASINRCONFTthreshList, [856](#)
 - TDSCDMASINRCONFTthreshListLen, [856](#)
- TDSCDMASINRCONFTthreshListLen
 - TDSCDMASINRCONFTthresh, [856](#)
- TDSCDMASINRThresh, [856](#)
 - pTDSCDMASINRThreshList, [857](#)
 - TDSCDMASINRThreshListLen, [857](#)
- TDSCDMASINRThreshListLen
 - nas_TDSCDMASINRThresh, [480](#)
 - TDSCDMASINRThresh, [857](#)
- TDSCDMASigInfoExt, [854](#)
 - ecio, [855](#)
 - rscp, [855](#)
 - rsi, [855](#)
 - sinr, [855](#)
- tFNASwiLTECphyCallInfo
 - qaGobiApiCbk.h, [1340](#)
- tFNASwiOTAMsg
 - qaGobiApiCbk.h, [1340](#)
- tFNAActivationStatus
 - qaGobiApiCbk.h, [1339](#)
- tFNAllCallStatus
 - qaGobiApiCbk.h, [1339](#)
- tFNAsyncRawSend
 - qaGobiApiCbk.h, [1340](#)
- tFNBandPreference
 - qaGobiApiCbk.h, [1340](#)
- tFNBestAvailPos
 - qaGobiApiCbk.h, [1341](#)
- tFNCATEvent
 - qaGobiApiCbk.h, [1342](#)
- tFNCbkUimSlotStatusChangeInd
 - qaGobiApiCbk.h, [1342](#)
- tFNDHCPv4ClientLeaseStatus
 - qaGobiApiCbk.h, [1344](#)
- tFNDTMFEvent

- qaGobiApiCbK.h, [1344](#)
- tFNDUNCallInfo
 - qaGobiApiCbK.h, [1344](#)
- tFNDataCapabilities
 - qaGobiApiCbK.h, [1342](#)
- tFNDataSysStatus
 - qaGobiApiCbK.h, [1343](#)
- tFNDeIAssistData
 - qaGobiApiCbK.h, [1343](#)
- tFNDeviceStateChange
 - qaGobiApiCbK.h, [1343](#)
- tFNEventPosition
 - qaGobiApiCbK.h, [1344](#)
- tFNFWdIdCompletion
 - qaGobiApiCbK.h, [1344](#)
- tFNGnssSvInfo
 - qaGobiApiCbK.h, [1345](#)
- tFNHDRPersonaity
 - qaGobiApiCbK.h, [1345](#)
- tFNlmsRegMgrConfig
 - qaGobiApiCbK.h, [1346](#)
- tFNlmsSIPConfig
 - qaGobiApiCbK.h, [1346](#)
- tFNlmsSMSCConfig
 - qaGobiApiCbK.h, [1347](#)
- tFNlmsUserConfig
 - qaGobiApiCbK.h, [1347](#)
- tFNlmsVoIPConfig
 - qaGobiApiCbK.h, [1347](#)
- tFNlmsaPdpStatus
 - qaGobiApiCbK.h, [1345](#)
- tFNlmsaRatStatus
 - qaGobiApiCbK.h, [1346](#)
- tFNlmsaRegStatus
 - qaGobiApiCbK.h, [1346](#)
- tFNlmsaSvcStatus
 - qaGobiApiCbK.h, [1346](#)
- tFNInfoRec
 - qaGobiApiCbK.h, [1347](#)
- tFNInjectPosition
 - qaGobiApiCbK.h, [1348](#)
- tFNInjectSensorData
 - qaGobiApiCbK.h, [1348](#)
- tFNInjectTimeStatus
 - qaGobiApiCbK.h, [1348](#)
- tFNInjectUTCTime
 - qaGobiApiCbK.h, [1348](#)
- tFNLURject
 - qaGobiApiCbK.h, [1348](#)
- tFNMemoryFull
 - qaGobiApiCbK.h, [1349](#)
- tFNMessageWaiting
 - qaGobiApiCbK.h, [1350](#)
- tFNMitiLvIRpt
 - qaGobiApiCbK.h, [1350](#)
- tFNMobileIPStatus
 - qaGobiApiCbK.h, [1350](#)
- tFNModemTempInfo
 - qaGobiApiCbK.h, [1350](#)
- tFNNasTimer
 - qaGobiApiCbK.h, [1350](#)
- tFNNet
 - qaGobiApiCbK.h, [1350](#)
- tFNNetworkTime
 - qaGobiApiCbK.h, [1351](#)
- tFNNewGPS
 - qaGobiApiCbK.h, [1351](#)
- tFNNewNMEA
 - qaGobiApiCbK.h, [1352](#)
- tFNNewRMTransferStatistics
 - qaGobiApiCbK.h, [1352](#)
- tFNNewSMS
 - qaGobiApiCbK.h, [1353](#)
- tFNOMADMState
 - qaGobiApiCbK.h, [1353](#)
- tFNOTASPStatus
 - qaGobiApiCbK.h, [1354](#)
- tFNOpMode
 - qaGobiApiCbK.h, [1354](#)
- tFNPDSState
 - qaGobiApiCbK.h, [1354](#)
- tFNPacketSrvState
 - qaGobiApiCbK.h, [1354](#)
- tFNPower
 - qaGobiApiCbK.h, [1355](#)
- tFNPrivacyChange
 - qaGobiApiCbK.h, [1355](#)
- tFNQosNWStatus
 - qaGobiApiCbK.h, [1355](#)
- tFNQosPriEvent
 - qaGobiApiCbK.h, [1355](#)
- tFNQosStatus
 - qaGobiApiCbK.h, [1356](#)
- tFNRFInfo
 - qaGobiApiCbK.h, [1357](#)
- tFNRankIndicator
 - qaGobiApiCbK.h, [1357](#)
- tFNResetInfo
 - qaGobiApiCbK.h, [1357](#)
- tFNRoamingIndicator
 - qaGobiApiCbK.h, [1358](#)
- tFNSDKTerminated
 - qaGobiApiCbK.h, [1358](#)
- tFNSLQSOMADMAAlert
 - qaGobiApiCbK.h, [1359](#)
- tFNSLQSQOSEvent
 - qaGobiApiCbK.h, [1360](#)
- tFNSLQSSessionState
 - qaGobiApiCbK.h, [1360](#)
- tFNSLQSSignalStrengths
 - qaGobiApiCbK.h, [1360](#)
- tFNSLQSWDSEvent
 - qaGobiApiCbK.h, [1360](#)
- tFNSMSEvents
 - qaGobiApiCbK.h, [1361](#)
- tFNSUPSInfo

- qaGobiApiCbK.h, [1361](#)
- tFNSUPSNotification
 - qaGobiApiCbK.h, [1361](#)
- tFNSensorStreaming
 - qaGobiApiCbK.h, [1358](#)
- tFNServingSystem
 - qaGobiApiCbK.h, [1358](#)
- tFNSetCradleMount
 - qaGobiApiCbK.h, [1359](#)
- tFNSetEngineState
 - qaGobiApiCbK.h, [1359](#)
- tFNSetEventTimeSync
 - qaGobiApiCbK.h, [1359](#)
- tFNSetExtPowerConfig
 - qaGobiApiCbK.h, [1359](#)
- tFNSigInfo
 - qaGobiApiCbK.h, [1359](#)
- tFNSignalStrength
 - qaGobiApiCbK.h, [1359](#)
- tFNSysInfo
 - qaGobiApiCbK.h, [1361](#)
- tFNSysSelectionPref
 - qaGobiApiCbK.h, [1361](#)
- tFNUIMRefresh
 - qaGobiApiCbK.h, [1362](#)
- tFNUIMStatusChangeInfo
 - qaGobiApiCbK.h, [1362](#)
- tFNUSSDNoWaitIndication
 - qaGobiApiCbK.h, [1363](#)
- tFNUSSDNotification
 - qaGobiApiCbK.h, [1363](#)
- tFNUSSDRelease
 - qaGobiApiCbK.h, [1363](#)
- tFNtransLayerInfo
 - qaGobiApiCbK.h, [1362](#)
- tFNtransNWRegInfo
 - qaGobiApiCbK.h, [1362](#)
- TFTIDParams, [859](#)
 - destPortRangeEnd, [860](#)
 - destPortRangeStart, [860](#)
 - eValid, [860](#)
 - filterId, [860](#)
 - flowLabel, [860](#)
 - IPSECSPi, [860](#)
 - ipVersion, [860](#)
 - nextHeader, [860](#)
 - pSourceIP, [861](#)
 - sourceIPMask, [861](#)
 - srcPortRangeEnd, [861](#)
 - srcPortRangeStart, [861](#)
 - tosMask, [861](#)
- THIRD_INSTANCE
 - qaGobiApiCbK.h, [1328](#)
- TIME_DATE_BUF
 - qaGobiApiSms.h, [1577](#)
- TIME_STAMP_BUF
 - qaGobiApiSms.h, [1577](#)
- TPCause
 - SMSAsyncRawSend_s, [806](#)
- TRMessageTlv
 - unpack_sms_SetNewSMSCallback_ind_t, [1043](#)
- TX_PWR
 - NetworkStat1x, [536](#)
- TXAGCList, [871](#)
 - pTXAIG, [872](#)
 - pTXComprSlope, [872](#)
 - pTXComprThres, [872](#)
 - pTXExpSlope, [872](#)
 - pTXExpThres, [872](#)
 - pTXStaticGain, [872](#)
- TXChan
 - LTEInfo, [381](#)
 - nas_LTEInfo, [440](#)
- tXDroppedCount
 - unpack_wds_GetPacketStatus_t, [1069](#)
- TXOKBytesCount
 - DUNCallInfoInd, [218](#)
- tXOKBytesLastCall
 - unpack_wds_GetPacketStatus_t, [1070](#)
- tXOkBytesCount
 - unpack_wds_GetPacketStatus_t, [1069](#)
- TXPCMIIRFiltr, [873](#)
 - pFlag, [874](#)
 - pStage0Val, [874](#)
 - pStage1Val, [874](#)
 - pStage2Val, [874](#)
 - pStage3Val, [874](#)
 - pStage4Val, [874](#)
 - pStageCnt, [874](#)
- tXPacketErrors
 - unpack_wds_GetPacketStatus_t, [1070](#)
- tXPacketOverflows
 - unpack_wds_GetPacketStatus_t, [1070](#)
- tXPacketSuccesses
 - unpack_wds_GetPacketStatus_t, [1070](#)
- Tables, [55](#)
- tac
 - LTEInfoIntrafreq, [384](#)
 - LTESysInfo, [397](#)
 - nas_LTEInfoIntrafreq, [443](#)
 - nas_LTESysInfo, [452](#)
- tacValid
 - LTESysInfo, [397](#)
 - nas_LTESysInfo, [452](#)
- TdsBandCapability
 - unpack_dms_SLQSGetBandCapability_t, [946](#)
- tdscdmaSigInfoExt, [855](#)
 - ecio, [855](#)
 - rscp, [855](#)
 - rssI, [855](#)
 - sinr, [856](#)
- tech
 - NWProfile, [542](#)
- techName
 - _packetSrvStatus, [64](#)

- unpack_wds_SLQSSetPacketSrvStatusCallback↔
_t, 1081
- techType
 - DataBearerTech, 195
- Technology
 - DeviceConfigDetail, 206
 - fwinfo_s, 236
 - unpack_wds_SLQSGetRuntimeSettings_t, 1079
- TechnologyPref
 - pack_nas_SetNetworkPreference_t, 584
- tempData_t, 857
 - temperature, 858
 - temperatureDataLen, 858
 - timeOfFirstSample, 858
 - timeOffset, 858
 - timeSource, 858
- temperature
 - CommInfo, 169
 - nas_CommInfo, 417
 - tempData_t, 858
 - tempratureData, 859
- temperatureDataLen
 - tempData_t, 858
 - tempratureData, 859
- temperatureData, 858
 - temperature, 859
 - temperatureDataLen, 859
 - timeOfFirstSample, 859
 - timeOffset, 859
 - timeSource, 859
- textMsgLength
 - cdmaMsgEncodingParams, 153
- Thermal Mitigation Device(TMD), 54
- threshGsmHigh
 - lteGsmCellInfo, 379
 - nas_lteGsmCellInfo, 438
- threshGsmLow
 - lteGsmCellInfo, 379
 - nas_lteGsmCellInfo, 438
- threshServingLow
 - LTEInfoIntraFreq, 384
 - nas_LTEInfoIntraFreq, 443
- threshXHigh
 - infoInterFreq, 316
 - nas_infoInterFreq, 437
- threshXLow
 - infoInterFreq, 316
 - nas_infoInterFreq, 437
- threshXhigh
 - lteWcdmaCellInfo, 398
 - nas_lteWcdmaCellInfo, 453
- threshXlow
 - lteWcdmaCellInfo, 398
 - nas_lteWcdmaCellInfo, 453
- thresholds
 - SignalStrengthDataType, 783
- thresholdsSize
 - SignalStrengthDataType, 783
- Time
 - unpack_swioma_SLQSOMADMGetSessionInfo↔
_t, 1051
 - wcdmaLongMsgDecodingParams, 1151
 - wcdmaMsgDecodingParams, 1153
- time
 - NASTimeInfoTlv, 531
- Time_uncert_ms
 - GPSSStateInfo, 277
- timeInfo, 861
 - day, 862
 - dayLtSavingAdj, 862
 - dayOfWeek, 862
 - hour, 862
 - minute, 862
 - month, 862
 - radiolInterface, 862
 - second, 863
 - timeZone, 863
 - TlvPresent, 863
 - year, 863
- TimeLength
 - unpack_swioma_SLQSOMADMGetSessionInfo↔
_t, 1051
- timeMsec
 - pack_loc_SLQSLOCInjectUTCTime_t, 579
- timeOfFirstSample
 - sensorData, 744
 - sensorData_t, 746
 - tempData_t, 858
 - tempratureData, 859
- timeOffset
 - sensorData, 744
 - sensorData_t, 746
 - tempData_t, 858
 - tempratureData, 859
- timeSource
 - tempData_t, 858
 - tempratureData, 859
- TimeStmp_gps_week
 - GPSSStateInfo, 277
- TimeStmp_tow_ms
 - GPSSStateInfo, 277
- timeSyncRefCounter
 - QmiCbkLocEventTimeSyncInd, 691
- timeTlv
 - NASQmiCbkNasSwtOTAMessageInd, 520
- timeUncMsec
 - pack_loc_SLQSLOCInjectUTCTime_t, 579
- timeZone
 - nas_timeInfo, 482
 - timeInfo, 863
- timeout
 - pack_qmi_t, 601
- timestamp
 - unpack_dms_GetNetworkTime_t, 935
- timestampAge
 - pack_loc_SLQSLOCInjectPosition_t, 577

- timestampUtc
 - pack_loc_SLQSLOCInjectPosition_t, 577
- timingAdvance
 - GERANInfo, 237
 - nas_GERANInfo, 425
- TlvPresent
 - CatCommonEventTlv, 142
 - DataULongLongTlv, 201
 - DataULongTlv, 201
 - dms_ActivationStatusTlv, 209
 - dms_OperatingModeTlv, 210
 - eTWSPLMNInfoTlv, 224
 - messageModeTlv, 399
 - NASBandPreferenceTlv, 494
 - NASEmergencyModeTlv, 496
 - NASGWAcqOrderPrefTlv, 504
 - NASLTEBandPreferenceTlv, 509
 - NASLteNasReleaseInfoTlv, 509
 - NASModePreferenceTlv, 509
 - NASNetSelPreferenceTlv, 510
 - NASOTAMessageTlv, 512
 - NASPRLPreferenceTlv, 520
 - NASPhyCaAggPcellInfo, 513
 - NASPhyCaAggScellIDIBw, 514
 - NASPhyCaAggScellIndType, 515
 - NASPhyCaAggScellIndex, 514
 - NASPhyCaAggScellInfo, 516
 - NASRoamPreferenceTlv, 521
 - NASServDomainPrefTlv, 522
 - NASTimeInfoTlv, 531
 - nas_PhyCaAggPcellInfo, 457
 - nas_PhyCaAggScellIDIBw, 457
 - nas_PhyCaAggScellIndType, 458
 - nas_PhyCaAggScellIndex, 458
 - nas_PhyCaAggScellInfo, 461
 - nas_RFInfoTlv, 465
 - nas_RejectReasonTlv, 464
 - nas_SLQSSignalStrengthsTlv, 474
 - nas_SccRxInfo, 470
 - nas_SignalStrengthTlv, 472
 - nas_timeInfo, 482
 - newMTMessageTlv, 538
 - PhyCaAggPcellInfo, 651
 - PhyCaAggScellIDIBw, 651
 - PhyCaAggScellIndType, 653
 - PhyCaAggScellIndex, 652
 - PhyCaAggScellInfo, 655
 - RoamingInfo, 729
 - sMSCAddressTlv, 808
 - sMSEtwsMessageTlv, 810
 - sMSOnIMSTlv, 818
 - SccRxInfo, 743
 - sessionInfoTlv, 754
 - sessionInfoTlvExt, 754
 - timeInfo, 863
 - transferRouteMessageTlv, 868
 - wds_DHCPLeaseOptTlv, 1161
 - wds_DHCPLeaseStateTlv, 1161
 - wds_DHCPProfileIdTlv, 1162
 - wds_DataULongLongTlv, 1160
 - wds_DataULongTlv, 1161
 - wds_IPv4AdTlv, 1167
- Tlvresult
 - pack_dms_GetCustFeaturesV2_t, 552
 - pack_dms_SetCustFeaturesV2_t, 555
 - pack_dms_SetPower_t, 556
 - pack_dms_SetUSBComp_t, 556
 - pack_dms_UIMGetICCID_t, 561
 - pack_fms_GetImagesPreference_t, 565
 - pack_fms_GetStoredImages_t, 565
 - pack_fms_SetImagesPreference_t, 566
 - pack_loc_Delete_Assist_Data_t, 567
 - pack_loc_EventRegister_t, 570
 - pack_loc_SLQSLOCGetBestAvailPos_t, 572
 - pack_loc_SetExtPowerState_t, 570
 - pack_loc_SetOperationMode_t, 571
 - pack_loc_Start_t, 582
 - pack_loc_Stop_t, 582
 - pack_nas_SetNetworkPreference_t, 584
 - pack_uim_ChangePin_t, 612
 - pack_uim_ReadTransparent_t, 613
 - pack_uim_SetPinProtection_t, 614
 - pack_uim_UnblockPin_t, 618
 - pack_uim_VerifyPin_t, 619
 - unpack_dms_GetBandCapability_t, 926
 - unpack_dms_GetCrashAction_t, 927
 - unpack_dms_GetCustFeature_t, 927
 - unpack_dms_GetCustFeaturesV2_t, 928
 - unpack_dms_GetDeviceCap_t, 928
 - unpack_dms_GetDeviceHardwareRev_t, 929
 - unpack_dms_GetDeviceMfr_t, 930
 - unpack_dms_GetDeviceSerialNumbers_t, 930
 - unpack_dms_GetFSN_t, 933
 - unpack_dms_GetFirmwareInfo_t, 931
 - unpack_dms_GetFirmwareRevision_t, 932
 - unpack_dms_GetFirmwareRevisions_t, 932
 - unpack_dms_GetIMSI_t, 933
 - unpack_dms_GetManufacturer_t, 934
 - unpack_dms_GetModelID_t, 934
 - unpack_dms_GetNetworkTime_t, 935
 - unpack_dms_GetOfflineReason_t, 936
 - unpack_dms_GetPRLVersion_t, 937
 - unpack_dms_GetPower_t, 937
 - unpack_dms_GetUSBComp_t, 938
 - unpack_dms_GetVoiceNumber_t, 938
 - unpack_dms_ResetToFactoryDefaults_t, 939
 - unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 942
 - unpack_dms_SLQSDmsSwiGetResetInfo_t, 943
 - unpack_dms_SLQSDmsSwiIndicationRegister_t, 943
 - unpack_dms_SLQSGetERIFile_t, 946
 - unpack_dms_SLQSSwiClearDyingGaspStatistics↔
_t, 947
 - unpack_dms_SLQSSwiGetCrashInfo_t, 947
 - unpack_dms_SLQSSwiGetDyingGaspCfg_t, 948

- unpack_dms_SLQSSwiGetDyingGaspStatistics↔
_t, 948
- unpack_dms_SLQSSwiGetFwUpdateStatus_t, 951
- unpack_dms_SLQSSwiGetHostDevInfo_t, 952
- unpack_dms_SLQSSwiGetOSInfo_t, 952
- unpack_dms_SLQSSwiGetSerialNoExt_t, 953
- unpack_dms_SLQSSwiSetDyingGaspCfg_t, 953
- unpack_dms_SLQSSwiSetHostDevInfo_t, 954
- unpack_dms_SLQSSwiSetOSInfo_t, 954
- unpack_dms_SLQSUIGetState_t, 955
- unpack_dms_SetActivationStatusCallback_t, 939
- unpack_dms_SetCustFeature_t, 940
- unpack_dms_SetCustFeaturesV2_t, 940
- unpack_dms_SetEventReport_ind_t, 941
- unpack_dms_SetEventReport_t, 941
- unpack_dms_SetFirmwarePreference_t, 941
- unpack_dms_SetPower_t, 942
- unpack_dms_SetUSBComp_t, 942
- unpack_dms_UIMGetControlKeyStatus_t, 956
- unpack_dms_UIMGetICCID_t, 956
- unpack_dms_UIMGetPINStatus_t, 958
- unpack_dms_UIMSetControlKeyProtection_t, 959
- unpack_dms_UIMSetPINProtection_t, 959
- unpack_dms_UIMUnblockControlKey_t, 960
- unpack_fms_GetImagesPreference_t, 961
- unpack_fms_GetStoredImages_t, 961
- unpack_fms_SetImagesPreference_t, 962
- unpack_loc_BestAvailPos_Ind_t, 968
- unpack_loc_Delete_Assist_Data_t, 968
- unpack_loc_DeleteAssistData_Ind_t, 969
- unpack_loc_EngineState_Ind_t, 970
- unpack_loc_EventRegister_t, 970
- unpack_loc_GnssSvInfo_Ind_t, 971
- unpack_loc_PositionRpt_Ind_t, 977
- unpack_loc_SLQSLOCGetBestAvailPos_t, 980
- unpack_loc_SetExtPowerConfig_Ind_t, 978
- unpack_loc_SetExtPowerState_t, 979
- unpack_loc_SetOperationMode_Ind_t, 979
- unpack_loc_SetOperationMode_t, 980
- unpack_loc_Start_t, 981
- unpack_loc_Stop_t, 981
- unpack_nas_GetNetworkPreference_t, 984
- unpack_nas_SLQSNasSwtOTAMessageCallback↔
_ind_t, 1008
- unpack_nas_SLQSSetSysSelectionPrefCall↔
Back_ind_t, 1010
- unpack_nas_SetNetworkPreference_t, 990
- unpack_nas_SetServingSystemCallback_ind_t,
991
- unpack_nas_SlqsGetLTECphyCAInfo_t, 992
- unpack_uim_ChangePin_t, 1054
- unpack_uim_GetCardStatus_t, 1054
- unpack_uim_ReadTransparent_t, 1055
- unpack_uim_SetPinProtection_t, 1056
- unpack_uim_UnblockPin_t, 1059
- unpack_uim_VerifyPin_t, 1060
- unpack_wds_SLQSCreateProfile_t, 1071
- unpack_wds_SLQSGetProfileSettings_t, 1077
- unpack_wds_SLQSSetIPFamilyPreference↔
t, 1080
- TmdDeRegNotMitigationLvlReq, 863
 - mitigationDevIDLen, 863
 - mitigationDevID, 863
- TmdGetMitigationDevListResp, 863
 - pMitigationDevList, 864
 - pMitigationDevListLen, 864
- TmdGetMitigationLvlReq, 864
 - mitigationDevIDLen, 864
 - mitigationDevID, 864
- TmdGetMitigationLvlResp, 865
 - pCurrentmitigationLvl, 865
 - pReqMitigationLvl, 865
- TmdMitigationLvlIndReq, 865
 - mitigationDevIDLen, 866
 - mitigationDevID, 866
- TmdRegNotMitigationLvlReq, 866
 - mitigationDevIDLen, 866
 - mitigationDevID, 866
- toServiceId
 - BroadcastConfig, 124
- toggleMode
 - lineCtrlInfo, 351
- TokenBucket
 - unpack_qos_swtQosFlow_t, 1038
- tokenBucket, 866
 - bucketSz, 867
 - peakRate, 867
 - tokenRate, 867
- tokenRate
 - tokenBucket, 867
 - unpack_qos_tokenBucket_t, 1039
- Tos, 867
 - mask, 867
 - val, 867
- tosMask
 - LibPackTFTIDParams, 347
 - TFTIDParams, 861
- total_rx_bytes
 - sQosStat, 824
 - unpack_qos_SLQSQosSwtReadDataStats_t, 1027
- total_rx_pkt
 - sQosStat, 824
 - unpack_qos_SLQSQosSwtReadDataStats_t, 1027
- total_tx_bytes
 - sQosStat, 824
 - unpack_qos_SLQSQosSwtReadDataStats_t, 1027
- total_tx_bytes_drp
 - sQosStat, 824
 - unpack_qos_SLQSQosSwtReadDataStats_t, 1027
- total_tx_pkt
 - sQosStat, 824
 - unpack_qos_SLQSQosSwtReadDataStats_t, 1027
- total_tx_pkt_drp
 - sQosStat, 824
 - unpack_qos_SLQSQosSwtReadDataStats_t, 1027
- TrStatInd, 869

- statsMask, [870](#)
- statsPeriod, [870](#)
- TrackAreaCode
 - unpack_nas_SLQSGetservingSystem_t, [996](#)
- trackAreaCode
 - qaQmiServingSystemParam, [683](#)
- TrafficClass
 - unpack_qos_swiQosFlow_t, [1038](#)
- trafficClass
 - LibPackUMTSQoS, [349](#)
 - UMTSMinQoS, [920](#)
 - UMTSQoS, [922](#)
 - wds_UMTSMinQoS, [1174](#)
- trafficPriority
 - LibPackUMTSQoS, [349](#)
 - UMTSMinQoS, [920](#)
 - UMTSQoS, [922](#)
 - wds_UMTSMinQoS, [1174](#)
- TranDstPort
 - unpack_qos_swiQosFilter_t, [1034](#)
- TranSrcPort
 - unpack_qos_swiQosFilter_t, [1034](#)
- TransCap
 - _transLayerinfo, [94](#)
- transLayerInfo
 - qaGobiApiSms.h, [1580](#)
- transLayerNotification
 - qaGobiApiCbk.h, [1363](#)
- transNWRegInfoNotification
 - qaGobiApiCbk.h, [1364](#)
- TransType
 - _transLayerinfo, [94](#)
- transactionID
 - SMSTransferRouteMTMessage, [822](#)
 - sMSTransferRouteMTMessage, [821](#)
- transferDelay
 - LibPackUMTSQoS, [349](#)
 - UMTSMinQoS, [920](#)
 - UMTSQoS, [922](#)
 - wds_UMTSMinQoS, [1174](#)
- TransferRouteMTMessageInfo
 - transferRouteMessageTlv, [868](#)
- transferRouteMessageTlv, [867](#)
 - TlvPresent, [868](#)
 - TransferRouteMTMessageInfo, [868](#)
- TransferStatInd, [868](#)
 - StatsMask, [869](#)
 - StatsPeriod, [869](#)
- transferStatInd, [869](#)
 - StatsMask, [869](#)
 - StatsPeriod, [869](#)
- transferStats
 - pack_wds_SLQSSetWdsEventCallback_t, [635](#)
- TransferStatsDataType, [869](#)
 - interval, [869](#)
- trueIMSI, [870](#)
 - imsiT112, [871](#)
 - imsiTS1, [871](#)
 - imsiTS2, [871](#)
 - imsiTaddrNum, [871](#)
 - mccT, [871](#)
- trueSrvStatus
 - GSMSrvStatusInfo, [281](#)
 - nas_GSMSrvStatusInfo, [428](#)
- tx_bytes
 - NetStats, [533](#)
 - sQosFlowStat, [822](#)
 - unpack_QosFlowStat_t, [1040](#)
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- tx_bytes_drp
 - sQosFlowStat, [823](#)
 - unpack_QosFlowStat_t, [1040](#)
- tx_errors
 - NetStats, [533](#)
- tx_overflows
 - NetStats, [533](#)
- tx_packets
 - NetStats, [533](#)
- tx_pkt
 - sQosFlowStat, [823](#)
 - unpack_QosFlowStat_t, [1040](#)
- tx_pkt_drp
 - sQosFlowStat, [823](#)
 - unpack_QosFlowStat_t, [1040](#)
- tx_pkts
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- TxDropConutTlv
 - QmiCbkWdsStatisticsIndState, [705](#)
 - unpack_RMTransferStatistics_ind_t, [1041](#)
- txInfo, [872](#)
 - isInTraffic, [873](#)
 - txPower, [873](#)
- txOKBytesCount
 - unpack_wds_SLQSGetDUNCallInfo_t, [1076](#)
- TxOkByteCountTlv
 - QmiCbkWdsStatisticsIndState, [705](#)
 - unpack_RMTransferStatistics_ind_t, [1041](#)
- TxOkConutTlv
 - QmiCbkWdsStatisticsIndState, [705](#)
 - unpack_RMTransferStatistics_ind_t, [1041](#)
- txPower
 - txInfo, [873](#)
- TxQFilter
 - unpack_qos_QosFlowInfo_t, [1023](#)
- TxQFlowGranted
 - unpack_qos_QosFlowInfo_t, [1023](#)
- type
 - _getResetInfoNotification, [60](#)
 - dmsSwiGetResetInfo, [214](#)
 - pack_wds_GetDefaultProfileNum_t, [620](#)
 - pack_wds_SetDefaultProfileNum_t, [624](#)
 - SwiOTAMsg_s, [837](#)
 - unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, [942](#)

- unpack_dms_SLQSDmsSwiGetResetInfo_t, 943
- unpack_qmi_t, 1018
- u16PRLVersion
 - unpack_dms_GetPRLVersion_t, 937
- u8PRLPreference
 - unpack_dms_GetPRLVersion_t, 937
- UATISIZE
 - qaGobiApiNas.h, 1499
- UDPDstPort
 - unpack_qos_swiQosFilter_t, 1034
- UDPSrcPort
 - unpack_qos_swiQosFilter_t, 1034
- UIM_MAX_DESCRIPTION_LENGTH
 - uim.h, 1783
- UIM_MAX_NO_OF_APPLICATIONS
 - uim.h, 1783
- UIM_MAX_NO_OF_SLOTS
 - uim.h, 1783
- UIM_UINT8_MAX_STRING_SZ
 - uim.h, 1783
- UIMAuthenticateReq, 890
 - authData, 891
 - pIndicationToken, 891
 - sessionInfo, 891
- UIMAuthenticateResp, 891
 - pAuthenticateResult, 891
 - pCardResult, 891
 - pIndicationToken, 891
- UIMChangePIN
 - qaGobiApiDms.h, 1447
- UIMChangePinReq, 892
 - changePIN, 892
 - pIndicationToken, 892
 - pKeyReferenceID, 892
 - sessionInfo, 892
- UIMDepersonalizationReq, 893
 - depersonalisationInfo, 893
- UIMDepersonalizationResp, 893
 - pRemainingRetries, 893
- UIMEventRegisterReqResp, 894
 - eventMask, 894
- UIMGetCardStatusResp, 894
 - pCardStatus, 895
 - pHotSwapStatus, 895
- UIMGetConfigurationReq, 895
 - pConfigurationMask, 895
- UIMGetConfigurationResp, 895
 - pAutoSelection, 896
 - pHaltSubscription, 896
 - pPersonalizationStatus, 896
- UIMGetControlKeyStatus
 - qaGobiApiDms.h, 1448
- UIMGetFileAttributesReq, 896
 - fileIndex, 897
 - pIndicationToken, 897
 - sessionInfo, 897
- UIMGetFileAttributesResp, 897
 - pCardResult, 897
- pFileAttributes, 897
- pIndicationToken, 897
- UIMGetICCID
 - qaGobiApiDms.h, 1449
- UIMGetPINStatus
 - qaGobiApiDms.h, 1450
- UIMGetSlotsStatusResp, 898
 - pNumberOfPhySlot, 898
 - pUimSlotsStatus, 898
- UIMPinResp, 898
 - pEncryptedPIN1, 899
 - pIndicationToken, 899
 - pRemainingRetries, 899
- UIMPowerDownReq, 899
 - slot, 899
- UIMPowerUpReq, 900
 - pIgnoreHotSwapSwitch, 900
 - slot, 900
- UIMReadTransparentReq, 900
 - fileIndex, 901
 - pEncryptData, 901
 - pIndicationToken, 901
 - readTransparent, 901
 - sessionInfo, 901
- UIMReadTransparentResp, 901
 - pCardResult, 902
 - pEncryptedData, 902
 - pIndicationToken, 902
 - pReadResult, 902
- UIMRefreshCompleteReq, 902
 - refreshComplete, 903
 - sessionInfo, 903
- UIMRefreshEvent, 903
 - aid, 904
 - aidLength, 904
 - arrfileInfo, 904
 - mode, 904
 - numOfFiles, 904
 - sessionType, 904
 - stage, 904
- UIMRefreshGetLastEventReq, 904
 - sessionInfo, 905
- UIMRefreshGetLastEventResp, 905
 - pRefreshEvent, 905
- UIMRefreshOKReq, 905
 - OKtoRefresh, 906
 - sessionInfo, 906
- UIMRefreshRegisterReq, 906
 - regRefresh, 906
 - sessionInfo, 906
- UIMSessionInformation, 906
 - aid, 907
 - aidLength, 907
 - sessionType, 907
- UIMSetControlKeyProtection
 - qaGobiApiDms.h, 1451
- UIMSetPINProtection
 - qaGobiApiDms.h, 1451

- UIMSetPinProtectionReq, 907
 - pIndicationToken, 908
 - pKeyReferenceID, 908
 - pinProtection, 908
 - sessionInfo, 908
- UIMSlotStatus, 909
 - bCCIDLength, 910
 - bCCID, 910
 - bLogicalSlot, 910
 - uPhyCardStatus, 910
 - uPhySlotStatus, 910
- UIMSlotStatusChangeInfo, 910
 - bNumberOfPhySlots, 910
 - slotsstatusChange, 910
- UIMSlotsStatus, 908
 - uimSlotStatus, 909
- UIMStatusChangeInfo, 910
 - statusChange, 911
- UIMSwitchSlotReq, 911
 - bLogicalSlot, 911
 - uPhysicalSlot, 911
- UIMUnblockControlKey
 - qaGobiApiDms.h, 1452
- UIMUnblockPIN
 - qaGobiApiDms.h, 1453
- UIMUnblockPinReq, 912
 - pIndicationToken, 912
 - pKeyReferenceID, 912
 - sessionInfo, 912
 - unblockPIN, 912
- UIMVerifyPIN
 - qaGobiApiDms.h, 1454
- UIMVerifyPinReq, 913
 - pEncryptedPIN1, 913
 - pIndicationToken, 913
 - pKeyReferenceID, 913
 - sessionInfo, 913
 - verifyPIN, 913
- ULONGLONG
 - SwiDataTypes.h, 1769
- ULONG
 - SwiDataTypes.h, 1769
- UMTSGrantedQoS
 - unpack_wds_SLQSGetRuntimeSettings_t, 1079
- UMTSInfo, 914
 - cellID, 915
 - ecio, 915
 - geranInst, 915
 - GeranInstInfo, 915
 - lac, 915
 - plmn, 915
 - psc, 915
 - rsc, 915
 - UMTSInstInfo, 915
 - uarfcn, 915
 - umtsInst, 915
- UMTSInstInfo
 - nas_UMTSInfo, 484
- UMTSInfo, 915
- UMTSLTENbrCell
 - nas_WCDMAInfoLTENeighborCell, 490
 - WCDMAInfoLTENeighborCell, 1149
- UMTSMinQoS, 917
 - deliveryErrSDU, 919
 - grntDownlinkBitrate, 919
 - grntUplinkBitrate, 919
 - maxDownlinkBitrate, 919
 - maxSDUSize, 919
 - maxUplinkBitrate, 919
 - qosDeliveryOrder, 919
 - resBerRatio, 920
 - sduErrorRatio, 920
 - trafficClass, 920
 - trafficPriority, 920
 - transferDelay, 920
- UMTSQoS, 920
 - deliveryErrSDU, 922
 - grntDownlinkBitrate, 922
 - grntUplinkBitrate, 922
 - maxDownlinkBitrate, 922
 - maxSDUSize, 922
 - maxUplinkBitrate, 922
 - qosDeliveryOrder, 922
 - resBerRatio, 922
 - sduErrorRatio, 922
 - trafficClass, 922
 - trafficPriority, 922
 - transferDelay, 922
- UMTSReqQoSSigInd, 922
 - SigInd, 923
 - UMTSReqQoS, 923
- UMTSReqQoS
 - LibPackUMTSReqQoSSigInd, 350
 - UMTSReqQoSSigInd, 923
- UMTSinstInfo, 915
 - umtsEcio, 916
 - umtsPsc, 916
 - umtsRsc, 916
 - umtsUarfcn, 916
- UNIQUE_ID_LEN
 - dms.h, 1208
 - qaGobiApiFms.h, 1460
- UNUSEDPARAM
 - common.h, 1197
 - SwiDataTypes.h, 1768
- uPhyCardStatus
 - slot_t, 784
 - UIMSlotStatus, 910
- uPhySlotStatus
 - slot_t, 784
 - UIMSlotStatus, 910
- uResult
 - sGetDeviceSeriesResult, 780
- USBComp
 - pack_dms_SetUSBComp_t, 556
 - unpack_dms_GetUSBComp_t, 938

- USBCompConfig, 1086
 - pUSBComp, 1087
- USBCompParams, 1087
 - pNumSupUSBComps, 1089
 - pSupUSBComps, 1089
 - pUSBComp, 1089
- USHORT
 - SwiDataTypes.h, 1769
- USSD_DCS_8BIT
 - qaGobiApiCbk.h, 1328
- USSD_DCS_ASCII
 - qaGobiApiCbk.h, 1328
- USSD_DCS_UCS2
 - qaGobiApiCbk.h, 1328
- USSDNoWaitIndicationInfo, 1089
 - pAlphaIdentifier, 1090
 - pError, 1090
 - pFailureCause, 1090
 - pUSSDData, 1090
- USSDRespFNetwork, 1090
 - pRespData, 1090
 - pTypeCode, 1090
- USSInfo, 1091
 - ussDCS, 1091
 - ussData, 1091
 - ussLen, 1091
- USSInformation
 - voiceOrigUSSDNoWaitInfo, 1129
- USSResp, 1091
 - pAlphaIDInfo, 1092
 - pCCSuppsType, 1092
 - pCallId, 1092
 - pCcResultType, 1092
 - pUSSDInfo, 1092
 - pfailureCause, 1092
- UUSData
 - UUSInfo, 1093
- UUSDatalen
 - UUSInfo, 1093
- UUSDcs
 - UUSInfo, 1093
- UUSInfo, 1092
 - UUSData, 1093
 - UUSDatalen, 1093
 - UUSDcs, 1093
 - UUSType, 1093
- UUSType
 - UUSInfo, 1093
- uarfcn
 - lteWcdmaCellInfo, 398
 - nas_UMTSInfo, 484
 - nas_lteWcdmaCellInfo, 453
 - UMTSInfo, 915
 - wcdmaUARFCN, 1158
- ueInIdle
 - LTEInfoInterfreq, 382
 - LTEInfoIntrafreq, 384
 - LTEInfoNeighboringGSM, 385
 - LTEInfoNeighboringWCDMA, 386
 - nas_LTEInfoInterfreq, 441
 - nas_LTEInfoIntrafreq, 443
 - nas_LTEInfoNeighboringGSM, 444
 - nas_LTEInfoNeighboringWCDMA, 445
- uim.h, 1781
 - MAX_DESCRIPTION_LENGTH, 1782
 - MAX_ICCID_LENGTH, 1782
 - MAX_NO_OF_APPLICATIONS, 1782
 - MAX_NO_OF_SLOTS, 1782
 - MAX_SLOTS_STATUS, 1782
 - pack_uim_ChangePin, 1783
 - pack_uim_GetCardStatus, 1783
 - pack_uim_ReadTransparent, 1784
 - pack_uim_SLQSUIMEventRegister, 1784
 - pack_uim_SLQSUIMGetSlotsStatus, 1785
 - pack_uim_SLQSUIMPowerDown, 1785
 - pack_uim_SLQSUIMPowerUp, 1785
 - pack_uim_SLQSUIMSwitchSlot, 1786
 - pack_uim_SetPinProtection, 1784
 - pack_uim_UnblockPin, 1786
 - pack_uim_VerifyPin, 1787
 - UIM_MAX_DESCRIPTION_LENGTH, 1783
 - UIM_MAX_NO_OF_APPLICATIONS, 1783
 - UIM_MAX_NO_OF_SLOTS, 1783
 - UIM_UINT8_MAX_STRING_SZ, 1783
 - unpack_uim_ChangePin, 1787
 - unpack_uim_GetCardStatus, 1787
 - unpack_uim_ReadTransparent, 1788
 - unpack_uim_SLQSUIMEventRegister, 1789
 - unpack_uim_SLQSUIMGetSlotsStatus, 1789
 - unpack_uim_SLQSUIMPowerDown, 1790
 - unpack_uim_SLQSUIMPowerUp, 1790
 - unpack_uim_SLQSUIMSetStatusChangeCall↔
 - Back_ind, 1791
 - unpack_uim_SLQSUIMSwitchSlot, 1791
 - unpack_uim_SetPinProtection, 1788
 - unpack_uim_SetUimSlotStatusChangeCallback↔
 - _ind, 1789
 - unpack_uim_UnblockPin, 1791
 - unpack_uim_VerifyPin, 1792
- uim_UIMSessionInformation, 887
 - aid, 888
 - aidLength, 888
 - sessionType, 888
- uim_appStatus, 874
 - aidLength, 877
 - aidVal, 877
 - appState, 877
 - appType, 877
 - persoFeature, 877
 - persoRetries, 877
 - persoState, 877
 - persoUnblockRetries, 877
 - pin1Retries, 877
 - pin1State, 877
 - pin2Retries, 877
 - pin2State, 877

- puk1Retries, [877](#)
 - puk2Retries, [877](#)
 - univPin, [877](#)
- uim_cardResult, [877](#)
 - sw1, [878](#)
 - sw2, [878](#)
- uim_cardStatus, [878](#)
 - index1xPri, [879](#)
 - index1xSec, [879](#)
 - indexGwPri, [879](#)
 - indexGwSec, [879](#)
 - numSlot, [879](#)
 - SlotInfo, [879](#)
- uim_changeUIMPIN, [879](#)
 - oldPINLen, [880](#)
 - oldPINVal, [880](#)
 - pinID, [880](#)
 - pinLen, [880](#)
 - pinVal, [880](#)
- uim_encryptedPIN1, [880](#)
 - pin1Len, [881](#)
 - pin1Val, [881](#)
- uim_fileInfo, [881](#)
 - fileID, [881](#)
 - path, [881](#)
 - pathLen, [882](#)
- uim_hotSwapStatus, [882](#)
 - hotSwap, [882](#)
 - hotSwapLength, [882](#)
- uim_readResult, [882](#)
 - content, [883](#)
 - contentLen, [883](#)
- uim_readTransparentInfo, [883](#)
 - length, [883](#)
 - offset, [883](#)
- uim_remainingRetries, [883](#)
 - unblockLeft, [884](#)
 - verifyLeft, [884](#)
- uim_sessionInformation, [884](#)
 - aid, [885](#)
 - aidLength, [885](#)
 - sessionType, [885](#)
- uim_setPINProtection, [885](#)
 - pinID, [886](#)
 - pinLength, [886](#)
 - pinOperation, [886](#)
 - pinValue, [886](#)
- uim_slotInfo, [886](#)
 - AppStatus, [887](#)
 - cardState, [887](#)
 - errorState, [887](#)
 - numApp, [887](#)
 - upinRetries, [887](#)
 - upinState, [887](#)
 - upukRetries, [887](#)
- uim_unblockUIMPIN, [888](#)
 - newPINLen, [889](#)
 - newPINVal, [889](#)
 - pinID, [889](#)
 - pukLen, [889](#)
 - pukVal, [889](#)
- uim_verifyUIMPIN, [889](#)
 - pinID, [890](#)
 - pinLen, [890](#)
 - pinVal, [890](#)
- uimSlotStatus
 - slots_t, [788](#)
 - UIMSlotsStatus, [909](#)
- ulData
 - DataULongTlv, [201](#)
 - wds_DataULongTlv, [1161](#)
- ulMask
 - rmTrasferStaticsReq, [728](#)
 - swiRMTrasferStaticsReq, [849](#)
- ulPhysicalSlot
 - pack_uim_SLQSUIMSwitchSlot_t, [617](#)
 - UIMSwitchSlotReq, [911](#)
- ulldata
 - DataULongLongTlv, [201](#)
 - wds_DataULongLongTlv, [1160](#)
- umtsEcio
 - nas_UMTSInstInfo, [485](#)
 - UMTSInstInfo, [916](#)
- umtsInst
 - nas_UMTSInfo, [484](#)
 - UMTSInfo, [915](#)
- umtsLTENbrCell, [916](#)
 - cellsTDD, [917](#)
 - earfcn, [917](#)
 - pci, [917](#)
 - rsrp, [917](#)
 - rsrq, [917](#)
 - srxlev, [917](#)
- umtsLTENbrCellLen
 - nas_WCDMAInfoLTENeighborCell, [490](#)
 - WCDMAInfoLTENeighborCell, [1149](#)
- umtsPsc
 - nas_UMTSInstInfo, [485](#)
 - UMTSInstInfo, [916](#)
- umtsRscp
 - nas_UMTSInstInfo, [485](#)
 - UMTSInstInfo, [916](#)
- umtsUarfcn
 - nas_UMTSInstInfo, [485](#)
 - UMTSInstInfo, [916](#)
- UnPackGetProfileSettingOut, [1086](#)
 - curProfile, [1086](#)
 - pExtErrCode, [1086](#)
- unblockLeft
 - personalizationStatus, [650](#)
 - remainingRetries, [723](#)
 - uim_remainingRetries, [884](#)
- unblockPIN
 - UIMUnblockPinReq, [912](#)
- unblockRetriesLeft
 - unpack_dms_UIMGetControlKeyStatus_t, [956](#)

- unpack_dms_UIMSetPINProtection_t, 959
 - unpack_dms_UIMUnlockControlKey_t, 960
- unlockUIMPIN, 923
 - newPINLen, 924
 - newPINVal, 924
 - pinID, 924
 - pukLen, 924
 - pukVal, 924
- uniqueID
 - CurrImageInfo, 184
 - image_info_t, 296
- univPin
 - appStats, 108
 - appStatus, 111
 - uim_appStatus, 877
- UniversalTime, 924
 - day, 925
 - dayOfWeek, 925
 - hour, 925
 - minute, 925
 - month, 925
 - second, 925
 - year, 925
- universalTime
 - nasNetworkTime, 511
 - unpack_nas_SLQSNasNetworkTimeCallBack_↔
ind_t, 1006
- unpack_QosFlowStat_t, 1039
 - bearerId, 1040
 - tx_bytes, 1040
 - tx_bytes_drp, 1040
 - tx_pkt, 1040
 - tx_pkt_drp, 1040
- unpack_RMTransferStatistics_ind_t, 1040
 - RxDropConutTlv, 1041
 - RxOkByteCountTlv, 1041
 - RxOkConutTlv, 1041
 - TxDropConutTlv, 1041
 - TxOkByteCountTlv, 1041
 - TxOkConutTlv, 1041
- unpack_dms_ActivateAutomatic
 - dms.h, 1233
- unpack_dms_GetActivationState
 - dms.h, 1233
- unpack_dms_GetActivationState_t, 925
 - state, 926
- unpack_dms_GetBandCapability
 - dms.h, 1234
- unpack_dms_GetBandCapability_t, 926
 - BandCapability, 926
 - Tlvresult, 926
- unpack_dms_GetCrashAction
 - dms.h, 1234
- unpack_dms_GetCrashAction_t, 926
 - DevCrashState, 927
 - Tlvresult, 927
- unpack_dms_GetCustFeature
 - dms.h, 1234
- unpack_dms_GetCustFeature_t, 927
 - DHCPRelayEnabled, 927
 - DisableIMSI, 927
 - GPSLPM, 927
 - GPSSel, 927
 - GpsEnable, 927
 - IPFamSupport, 927
 - IsVoiceEnabled, 927
 - RMAutoConnect, 927
 - SMSSupport, 927
 - Tlvresult, 927
- unpack_dms_GetCustFeaturesV2
 - dms.h, 1235
- unpack_dms_GetCustFeaturesV2_t, 927
 - GetCustomFeatureV2, 928
 - Tlvresult, 928
- unpack_dms_GetDeviceCap
 - dms.h, 1235
- unpack_dms_GetDeviceCap_t, 928
 - DataServiceCapability, 928
 - MaxRXChannelRate, 928
 - MaxTXChannelRate, 928
 - Radiolfaces, 928
 - RadiolfacesSize, 928
 - SimCapability, 928
 - Tlvresult, 928
- unpack_dms_GetDeviceCapabilities
 - dms.h, 1235
- unpack_dms_GetDeviceCapabilities_t, 928
 - dataServiceCaCapability, 929
 - maxRxChannelRate, 929
 - maxTxChannelRate, 929
 - Radiolfaces, 929
 - radiolfacesSize, 929
 - simCapability, 929
- unpack_dms_GetDeviceHardwareRev
 - dms.h, 1236
- unpack_dms_GetDeviceHardwareRev_t, 929
 - String, 929
 - stringSize, 929
 - Tlvresult, 929
- unpack_dms_GetDeviceMfr
 - dms.h, 1236
- unpack_dms_GetDeviceMfr_t, 929
 - String, 930
 - stringSize, 930
 - Tlvresult, 930
- unpack_dms_GetDeviceSerialNumbers
 - dms.h, 1237
- unpack_dms_GetDeviceSerialNumbers_t, 930
 - ESNString, 930
 - esnSize, 930
 - IMEIString, 930
 - imeiSize, 930
 - imeiSvnSize, 930
 - ImeiSvnString, 930
 - MEIDString, 930
 - meidSize, 930

- Tlvresult, [930](#)
- unpack_dms_GetFSN_t, [933](#)
 - String, [933](#)
 - Tlvresult, [933](#)
- unpack_dms_GetFSN
 - dms.h, [1238](#)
- unpack_dms_GetFirmwareInfo
 - dms.h, [1237](#)
- unpack_dms_GetFirmwareInfo_t, [930](#)
 - appversion_str, [931](#)
 - bootversion_str, [931](#)
 - carrier_str, [931](#)
 - cur_carr_name, [931](#)
 - cur_carr_rev, [931](#)
 - modelid_str, [931](#)
 - packageid_str, [931](#)
 - priversion_str, [931](#)
 - sku_str, [931](#)
 - Tlvresult, [931](#)
- unpack_dms_GetFirmwareRevision
 - dms.h, [1237](#)
- unpack_dms_GetFirmwareRevision_t, [931](#)
 - AMSSString, [932](#)
 - amssSize, [932](#)
 - PRIStr, [932](#)
 - Tlvresult, [932](#)
- unpack_dms_GetFirmwareRevisions
 - dms.h, [1238](#)
- unpack_dms_GetFirmwareRevisions_t, [932](#)
 - AMSSString, [932](#)
 - amssSize, [932](#)
 - bootSize, [932](#)
 - BootString, [932](#)
 - PRIStr, [932](#)
 - priSize, [932](#)
 - Tlvresult, [932](#)
- unpack_dms_GetHardwareRevision
 - dms.h, [1239](#)
- unpack_dms_GetHardwareRevision_t, [933](#)
 - hwVer, [933](#)
- unpack_dms_GetIMSI_t, [933](#)
 - imsi, [933](#)
 - Tlvresult, [933](#)
- unpack_dms_GetIMSI
 - dms.h, [1239](#)
- unpack_dms_GetManufacturer
 - dms.h, [1239](#)
- unpack_dms_GetManufacturer_t, [933](#)
 - manufacturer, [934](#)
 - Tlvresult, [934](#)
- unpack_dms_GetModelID_t, [934](#)
 - modelid, [934](#)
 - Tlvresult, [934](#)
- unpack_dms_GetModelID
 - dms.h, [1240](#)
- unpack_dms_GetNetworkTime
 - dms.h, [1240](#)
- unpack_dms_GetNetworkTime_t, [934](#)
 - source, [935](#)
 - timestamp, [935](#)
 - Tlvresult, [935](#)
- unpack_dms_GetOfflineReason
 - dms.h, [1241](#)
- unpack_dms_GetOfflineReason_t, [935](#)
 - pReasonMask, [936](#)
 - pbPlatform, [936](#)
 - Tlvresult, [936](#)
- unpack_dms_GetPRLVersion
 - dms.h, [1241](#)
- unpack_dms_GetPRLVersion_t, [937](#)
 - Tlvresult, [937](#)
 - u16PRLVersion, [937](#)
 - u8PRLPreference, [937](#)
- unpack_dms_GetPower
 - dms.h, [1241](#)
- unpack_dms_GetPower_t, [936](#)
 - HardwareControlledMode, [936](#)
 - OfflineReason, [937](#)
 - OperationMode, [937](#)
 - Tlvresult, [937](#)
- unpack_dms_GetSerialNumbers
 - dms.h, [1242](#)
- unpack_dms_GetSerialNumbers_t, [937](#)
 - esn, [937](#)
 - imei_no, [937](#)
 - imeisv_svn, [938](#)
 - meid, [938](#)
- unpack_dms_GetUSBComp
 - dms.h, [1242](#)
- unpack_dms_GetUSBComp_t, [938](#)
 - NumSupUSBComps, [938](#)
 - SupUSBComps, [938](#)
 - Tlvresult, [938](#)
 - USBComp, [938](#)
- unpack_dms_GetVoiceNumber
 - dms.h, [1243](#)
- unpack_dms_GetVoiceNumber_t, [938](#)
 - MIN, [938](#)
 - minSize, [938](#)
 - Tlvresult, [938](#)
 - VoiceNumber, [938](#)
 - voiceNumberSize, [938](#)
- unpack_dms_ResetToFactoryDefaults
 - dms.h, [1243](#)
- unpack_dms_ResetToFactoryDefaults_t, [938](#)
 - Tlvresult, [939](#)
- unpack_dms_SLQSDmsSwiGetResetInfo
 - dms.h, [1247](#)
- unpack_dms_SLQSDmsSwiGetResetInfo_Ind
 - dms.h, [1247](#)
- unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, [942](#)
 - source, [942](#)
 - Tlvresult, [942](#)
 - type, [942](#)
- unpack_dms_SLQSDmsSwiGetResetInfo_t, [942](#)
 - source, [943](#)

- Tlvresult, 943
- type, 943
- unpack_dms_SLQSDmsSwiIndicationRegister
 - dms.h, 1248
- unpack_dms_SLQSDmsSwiIndicationRegister_t, 943
 - Tlvresult, 943
- unpack_dms_SLQSGetBandCapability
 - dms.h, 1248
- unpack_dms_SLQSGetBandCapability_t, 943
 - bandCapability, 946
 - is_LteBandCapability_Available, 946
 - is_TdsBandCapability_Available, 946
 - LteBandCapability, 946
 - TdsBandCapability, 946
- unpack_dms_SLQSGetERIFile
 - dms.h, 1249
- unpack_dms_SLQSGetERIFile_t, 946
 - eriFile, 946
 - Tlvresult, 946
- unpack_dms_SLQSSwiClearDyingGaspStatistics
 - dms.h, 1249
- unpack_dms_SLQSSwiClearDyingGaspStatistics_↔
 - t, 947
 - Tlvresult, 947
- unpack_dms_SLQSSwiGetCrashInfo
 - dms.h, 1250
- unpack_dms_SLQSSwiGetCrashInfo_t, 947
 - crashInfoParam, 947
 - Tlvresult, 947
- unpack_dms_SLQSSwiGetDyingGaspCfg
 - dms.h, 1250
- unpack_dms_SLQSSwiGetDyingGaspCfg_t, 948
 - pGetDyingGaspCfg, 948
 - Tlvresult, 948
- unpack_dms_SLQSSwiGetDyingGaspStatistics
 - dms.h, 1250
- unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 948
 - pGetDyingGaspStatistics, 948
 - Tlvresult, 948
- unpack_dms_SLQSSwiGetFirmwareCurr
 - dms.h, 1251
- unpack_dms_SLQSSwiGetFirmwareCurr_t, 949
 - carrier, 949
 - fwvers, 949
 - numEntries, 949
 - pCurrImgInfo, 949
 - pkgver, 949
 - priver, 949
- unpack_dms_SLQSSwiGetFwUpdateStatus
 - dms.h, 1251
- unpack_dms_SLQSSwiGetFwUpdateStatus_t, 950
 - imgType, 951
 - logString, 951
 - refData, 951
 - refString, 951
 - ResCode, 951
 - Tlvresult, 951
- unpack_dms_SLQSSwiGetHostDevInfo
 - dms.h, 1252
- unpack_dms_SLQSSwiGetHostDevInfo_t, 951
 - manString, 952
 - modelString, 952
 - plasmaIDString, 952
 - swVerString, 952
 - Tlvresult, 952
- unpack_dms_SLQSSwiGetOSInfo
 - dms.h, 1252
- unpack_dms_SLQSSwiGetOSInfo_t, 952
 - nameString, 952
 - Tlvresult, 952
 - versionString, 952
- unpack_dms_SLQSSwiGetSerialNoExt
 - dms.h, 1252
- unpack_dms_SLQSSwiGetSerialNoExt_t, 952
 - meidString, 953
 - Tlvresult, 953
- unpack_dms_SLQSSwiSetDyingGaspCfg
 - dms.h, 1253
- unpack_dms_SLQSSwiSetDyingGaspCfg_t, 953
 - Tlvresult, 953
- unpack_dms_SLQSSwiSetHostDevInfo
 - dms.h, 1253
- unpack_dms_SLQSSwiSetHostDevInfo_t, 953
 - Tlvresult, 954
- unpack_dms_SLQSSwiSetOSInfo
 - dms.h, 1254
- unpack_dms_SLQSSwiSetOSInfo_t, 954
 - Tlvresult, 954
- unpack_dms_SLQSUIMGetState
 - dms.h, 1254
- unpack_dms_SLQSUIMGetState_t, 954
 - state, 955
 - Tlvresult, 955
- unpack_dms_SetActivationStatusCallback
 - dms.h, 1243
- unpack_dms_SetActivationStatusCallback_t, 939
 - Tlvresult, 939
- unpack_dms_SetCrashAction
 - dms.h, 1244
- unpack_dms_SetCrashAction_t, 939
 - notused, 940
- unpack_dms_SetCustFeature
 - dms.h, 1244
- unpack_dms_SetCustFeature_t, 940
 - Tlvresult, 940
- unpack_dms_SetCustFeaturesV2
 - dms.h, 1245
- unpack_dms_SetCustFeaturesV2_t, 940
 - Tlvresult, 940
- unpack_dms_SetEventReport
 - dms.h, 1245
- unpack_dms_SetEventReport_ind
 - dms.h, 1245
- unpack_dms_SetEventReport_ind_t, 940
 - ActivationStatusTlv, 941
 - OperatingModeTlv, 941

- Tlvresult, [941](#)
- unpack_dms_SetEventReport_t, [941](#)
 - Tlvresult, [941](#)
- unpack_dms_SetFirmwarePreference
 - dms.h, [1246](#)
- unpack_dms_SetFirmwarePreference_t, [941](#)
 - Tlvresult, [941](#)
- unpack_dms_SetPower
 - dms.h, [1246](#)
- unpack_dms_SetPower_t, [941](#)
 - Tlvresult, [942](#)
- unpack_dms_SetUSBComp
 - dms.h, [1247](#)
- unpack_dms_SetUSBComp_t, [942](#)
 - Tlvresult, [942](#)
- unpack_dms_UIMChangePIN
 - dms.h, [1254](#)
- unpack_dms_UIMGetControlKeyStatus
 - dms.h, [1255](#)
- unpack_dms_UIMGetControlKeyStatus_t, [955](#)
 - facilityState, [956](#)
 - Tlvresult, [956](#)
 - unblockRetriesLeft, [956](#)
 - verifyRetriesLeft, [956](#)
- unpack_dms_UIMGetICCID_t, [956](#)
 - String, [956](#)
 - stringSize, [956](#)
 - Tlvresult, [956](#)
- unpack_dms_UIMGetICCID
 - dms.h, [1255](#)
- unpack_dms_UIMGetPINStatus
 - dms.h, [1256](#)
- unpack_dms_UIMGetPINStatus_t, [956](#)
 - p1Status, [958](#)
 - p1UnblockRetriesLeft, [958](#)
 - p1VerifyRetriesLeft, [958](#)
 - p2Status, [958](#)
 - p2UnblockRetriesLeft, [958](#)
 - p2VerifyRetriesLeft, [958](#)
 - Tlvresult, [958](#)
- unpack_dms_UIMSetControlKeyProtection
 - dms.h, [1256](#)
- unpack_dms_UIMSetControlKeyProtection_t, [958](#)
 - Tlvresult, [959](#)
 - verifyRetriesLeft, [959](#)
- unpack_dms_UIMSetPINProtection
 - dms.h, [1256](#)
- unpack_dms_UIMSetPINProtection_t, [959](#)
 - Tlvresult, [959](#)
 - unblockRetriesLeft, [959](#)
 - verifyRetriesLeft, [959](#)
- unpack_dms_UIMUnblockControlKey
 - dms.h, [1257](#)
- unpack_dms_UIMUnblockControlKey_t, [959](#)
 - Tlvresult, [960](#)
 - unblockRetriesLeft, [960](#)
- unpack_dms_UIMUnblockPIN
 - dms.h, [1257](#)
- unpack_dms_UIMVerifyPIN
 - dms.h, [1258](#)
- unpack_dms_ValidateSPC
 - dms.h, [1258](#)
- unpack_fms_GetImagesPreference
 - fms.h, [1261](#)
- unpack_fms_GetImagesPreference_t, [960](#)
 - ImageListSize, [960](#)
 - pImageList, [961](#)
 - Tlvresult, [961](#)
- unpack_fms_GetStoredImages
 - fms.h, [1261](#)
- unpack_fms_GetStoredImages_t, [961](#)
 - imageList, [961](#)
 - imagelistSize, [961](#)
 - Tlvresult, [961](#)
- unpack_fms_SetImagesPreference
 - fms.h, [1262](#)
- unpack_fms_SetImagesPreference_t, [961](#)
 - ImageTypes, [962](#)
 - ImageTypesSize, [962](#)
 - Tlvresult, [962](#)
- unpack_loc_BestAvailPos_Ind
 - loc.h, [1272](#)
- unpack_loc_BestAvailPos_Ind_t, [962](#)
 - pAltitudeWrtEllipsoid, [966](#)
 - pAltitudeWrtMeanSeaLevel, [966](#)
 - pGpsTime, [967](#)
 - pHeading, [967](#)
 - pHeadingUnc, [967](#)
 - pHorCirConf, [967](#)
 - pHorEllpConf, [967](#)
 - pHorReliability, [967](#)
 - pHorUncCircular, [967](#)
 - pHorUncEllipseOrientAzimuth, [967](#)
 - pHorUncEllipseSemiMajor, [967](#)
 - pHorUncEllipseSemiMinor, [967](#)
 - pLatitude, [967](#)
 - pLongitude, [967](#)
 - pMagneticDeviation, [967](#)
 - pPrecisionDilution, [967](#)
 - pSensorDataUsage, [967](#)
 - pSpeedHorizontal, [967](#)
 - pSpeedUnc, [967](#)
 - pSpeedVertical, [967](#)
 - pSpeedVerticalUnc, [967](#)
 - pSvUsedforFix, [967](#)
 - pTechnologyMask, [967](#)
 - pTimeSrc, [967](#)
 - pTimeUnc, [968](#)
 - pTimestampUtc, [967](#)
 - pVertConfidence, [968](#)
 - pVertReliability, [968](#)
 - pVertUnc, [968](#)
 - pXid, [968](#)
 - status, [968](#)
 - Tlvresult, [968](#)
- unpack_loc_Delete_Assist_Data_t, [968](#)

- Tlvresult, 968
- unpack_loc_DeleteAssistData
 - loc.h, 1272
- unpack_loc_DeleteAssistData_Ind
 - loc.h, 1272
- unpack_loc_DeleteAssistData_Ind_t, 968
 - status, 969
 - Tlvresult, 969
- unpack_loc_EngineState_Ind
 - loc.h, 1273
- unpack_loc_EngineState_Ind_t, 969
 - engineState, 970
 - Tlvresult, 970
- unpack_loc_EventRegister
 - loc.h, 1273
- unpack_loc_EventRegister_t, 970
 - Tlvresult, 970
- unpack_loc_GnssSvInfo_Ind
 - loc.h, 1274
- unpack_loc_GnssSvInfo_Ind_t, 970
 - altitudeAssumed, 971
 - pSatelliteInfo, 971
 - Tlvresult, 971
- unpack_loc_PositionRpt_Ind
 - loc.h, 1274
- unpack_loc_PositionRpt_Ind_t, 971
 - pAltitudeAssumed, 976
 - pAltitudeWrtEllipsoid, 976
 - pAltitudeWrtMeanSeaLevel, 976
 - pFixId, 976
 - pGpsTime, 976
 - pHeading, 976
 - pHeadingUnc, 976
 - pHorConfidence, 976
 - pHorReliability, 976
 - pHorUncCircular, 976
 - pHorUncEllipseOrientAzimuth, 976
 - pHorUncEllipseSemiMajor, 976
 - pHorUncEllipseSemiMinor, 976
 - pLatitude, 976
 - pLeapSeconds, 976
 - pLongitude, 976
 - pMagneticDeviation, 976
 - pPrecisionDilution, 976
 - pSensorDataUsage, 976
 - pSpeedHorizontal, 976
 - pSpeedUnc, 977
 - pSpeedVertical, 977
 - pSvUsedforFix, 977
 - pTechnologyMask, 977
 - pTimeSrc, 977
 - pTimeUnc, 977
 - pTimestampUtc, 977
 - pVertConfidence, 977
 - pVertReliability, 977
 - pVertUnc, 977
 - sessionId, 977
 - sessionStatus, 977
- Tlvresult, 977
- unpack_loc_SLQSLOCGetBestAvailPos
 - loc.h, 1276
- unpack_loc_SLQSLOCGetBestAvailPos_t, 980
 - Tlvresult, 980
- unpack_loc_SLQSLOCInjectPosition
 - loc.h, 1276
- unpack_loc_SLQSLOCInjectSensorData
 - loc.h, 1277
- unpack_loc_SLQSLOCInjectUTCTime
 - loc.h, 1277
- unpack_loc_SLQSLOCSetCradleMountConfig
 - loc.h, 1277
- unpack_loc_SetExtPowerConfig_Ind
 - loc.h, 1274
- unpack_loc_SetExtPowerConfig_Ind_t, 977
 - status, 978
 - Tlvresult, 978
- unpack_loc_SetExtPowerState
 - loc.h, 1275
- unpack_loc_SetExtPowerState_t, 978
 - Tlvresult, 979
- unpack_loc_SetOperationMode
 - loc.h, 1275
- unpack_loc_SetOperationMode_Ind
 - loc.h, 1276
- unpack_loc_SetOperationMode_Ind_t, 979
 - status, 979
 - Tlvresult, 979
- unpack_loc_SetOperationMode_t, 979
 - Tlvresult, 980
- unpack_loc_Start
 - loc.h, 1278
- unpack_loc_Start_t, 980
 - Tlvresult, 981
- unpack_loc_Stop
 - loc.h, 1278
- unpack_loc_Stop_t, 981
 - Tlvresult, 981
- unpack_nas_GetACCOLC
 - nas.h, 1296
- unpack_nas_GetANAAAAAuthenticationStatus
 - nas.h, 1296
- unpack_nas_GetCDMANetworkParameters
 - nas.h, 1297
- unpack_nas_GetCDMANetworkParameters_t, 981
 - Application, 982
 - Broadcast, 982
 - CustomSCP, 982
 - ForceRev0, 982
 - Protocol, 982
 - RegForeignNID, 982
 - RegForeignSID, 982
 - RegHomeSID, 982
 - Roaming, 982
 - SCI, 982
 - SCM, 982
- unpack_nas_GetHomeNetwork

- nas.h, [1297](#)
- unpack_nas_GetHomeNetwork_t, [982](#)
 - mcc, [983](#)
 - mnc, [983](#)
 - name, [983](#)
 - nid, [983](#)
 - sid, [983](#)
- unpack_nas_GetNetworkPreference
 - nas.h, [1297](#)
- unpack_nas_GetNetworkPreference_t, [983](#)
 - ActiveTechPref, [984](#)
 - Duration, [984](#)
 - PersistentTechPref, [984](#)
 - Tlvresult, [984](#)
- unpack_nas_GetRFInfo
 - nas.h, [1297](#)
- unpack_nas_GetRFInfo_t, [984](#)
 - instancesSize, [984](#)
 - RFBandInfoElements, [984](#)
- unpack_nas_GetServingNetwork
 - nas.h, [1298](#)
- unpack_nas_GetServingNetwork_t, [985](#)
 - CSDomain, [985](#)
 - DataCaps, [985](#)
 - DataCapsLen, [985](#)
 - MCC, [985](#)
 - MNC, [985](#)
 - Name, [986](#)
 - nameSize, [986](#)
 - PSDomain, [986](#)
 - RAN, [986](#)
 - Radiolfaces, [986](#)
 - RadiolfacesSize, [986](#)
 - RegistrationState, [986](#)
 - Roaming, [986](#)
- unpack_nas_GetServingNetworkCapabilities
 - nas.h, [1298](#)
- unpack_nas_GetServingNetworkCapabilities_t, [986](#)
 - DataCaps, [986](#)
 - DataCapsLen, [986](#)
- unpack_nas_GetSignalStrengths
 - nas.h, [1298](#)
- unpack_nas_GetSignalStrengths_t, [986](#)
 - len, [987](#)
 - radio, [987](#)
 - rsi, [987](#)
- unpack_nas_PerformNetworkScan
 - nas.h, [1299](#)
- unpack_nas_PerformNetworkScan_t, [987](#)
 - p3GppNetworkInfoInstances, [987](#)
 - p3GppNetworkInstanceSize, [987](#)
 - pCSIInstance, [987](#)
 - pCSIInstanceSize, [987](#)
 - pRATInstance, [987](#)
 - pRATInstanceSize, [987](#)
 - pScanResult, [988](#)
- unpack_nas_SLQSGetNetworkTime
 - nas.h, [1302](#)
- unpack_nas_SLQSGetNetworkTime_t, [992](#)
 - p3GPP2TimeInfo, [992](#)
 - p3GPPTIMEInfo, [992](#)
- unpack_nas_SLQSGetPLMNName
 - nas.h, [1302](#)
- unpack_nas_SLQSGetPLMNName_t, [993](#)
 - longName, [993](#)
 - longNameCI, [993](#)
 - longNameEn, [993](#)
 - longNameLen, [993](#)
 - longNameSB, [993](#)
 - shortName, [993](#)
 - shortNameCI, [993](#)
 - shortNameEn, [993](#)
 - shortNameLen, [993](#)
 - shortNameSB, [993](#)
 - spn, [993](#)
 - spnEncoding, [993](#)
 - spnLength, [993](#)
- unpack_nas_SLQSGetServingSystem
 - nas.h, [1302](#)
- unpack_nas_SLQSGetServingSystem_t, [993](#)
 - BasestationID, [995](#)
 - BasestationLatitude, [995](#)
 - BasestationLongitude, [995](#)
 - CDMA_P_Rev, [995](#)
 - CDMASystemInfoExt, [995](#)
 - CallBarStatus, [995](#)
 - CellID, [995](#)
 - ConcSvcInfo, [995](#)
 - CurrentPLMN, [995](#)
 - DTMInd, [995](#)
 - DataSrvCapabilities, [995](#)
 - DefaultRoamInd, [995](#)
 - DetailedSvcInfo, [995](#)
 - Gpp2TimeZone, [995](#)
 - GppNetworkDSTAdjustment, [995](#)
 - GppTimeZone, [995](#)
 - HdrPersonality, [995](#)
 - Lac, [995](#)
 - NetworkID, [995](#)
 - PRLInd, [996](#)
 - RoamIndicatorVal, [996](#)
 - RoamingIndicatorList, [996](#)
 - ServingSystem, [996](#)
 - SystemID, [996](#)
 - TrackAreaCode, [996](#)
- unpack_nas_SLQSGetSignalStrength
 - nas.h, [1303](#)
- unpack_nas_SLQSGetSignalStrength_t, [996](#)
 - ecioList, [997](#)
 - ecioListLen, [997](#)
 - errorRateList, [997](#)
 - errorRateListLen, [997](#)
 - lo, [997](#)
 - ltsr, [997](#)
 - ltsnr, [997](#)
 - rsrqInfo, [997](#)

- rxSignalStrengthList, 997
- rxSignalStrengthListLen, 997
- signalStrengthReqMask, 997
- sinr, 997
- unpack_nas_SLQSGetSysInfo
 - nas.h, 1303
- unpack_nas_SLQSGetSysInfo_t, 997
 - pAddCDMASysInfo, 999
 - pAddGSMSysInfo, 999
 - pAddHDRSysInfo, 999
 - pAddLTESysInfo, 999
 - pAddWCDMASysInfo, 999
 - pCDMASrvStatusInfo, 999
 - pCDMASysInfo, 999
 - pGSMCallBarringSysInfo, 999
 - pGSMCipherDomainSysInfo, 999
 - pGSMSrvStatusInfo, 999
 - pGSMSysInfo, 999
 - pHDRSrvStatusInfo, 999
 - pHDRSysInfo, 1000
 - pLTESrvStatusInfo, 1000
 - pLTESysInfo, 1000
 - pLTEVoiceSupportSysInfo, 1000
 - pWCDMACallBarringSysInfo, 1000
 - pWCDMACipherDomainSysInfo, 1000
 - pWCDMASrvStatusInfo, 1000
 - pWCDMASysInfo, 1000
- unpack_nas_SLQSGetSysSelectionPref
 - nas.h, 1303
- unpack_nas_SLQSGetSysSelectionPref_t, 1000
 - pBandPref, 1003
 - pEmerMode, 1003
 - pGWAcqOrderPref, 1003
 - pLTEBandPref, 1003
 - pModePref, 1003
 - pNetSelPref, 1003
 - pPRLPref, 1003
 - pRoamPref, 1003
 - pSrvDomainPref, 1003
- unpack_nas_SLQSInitiateNetworkRegistration
 - nas.h, 1304
- unpack_nas_SLQSNasConfigSigInfo2
 - nas.h, 1304
- unpack_nas_SLQSNasGetCellLocationInfo
 - nas.h, 1304
- unpack_nas_SLQSNasGetCellLocationInfo_t, 1003
 - pCDMAInfo, 1004
 - pGERANInfo, 1004
 - pLTEInfoInterfreq, 1004
 - pLTEInfoIntrafreq, 1005
 - pLTEInfoNeighboringGSM, 1005
 - pLTEInfoNeighboringWCDMA, 1005
 - pUMTSCellID, 1005
 - pUMTSInfo, 1005
 - pWCDMAInfoLTENeighborCell, 1005
- unpack_nas_SLQSNasGetSigInfo
 - nas.h, 1305
- unpack_nas_SLQSNasGetSigInfo_t, 1005
 - CDMASSInfo, 1005
 - GSMSSInfo, 1005
 - HDRSSInfo, 1005
 - LTESInfo, 1005
 - WCDMASSInfo, 1005
- unpack_nas_SLQSNasIndicationRegisterExt
 - nas.h, 1305
- unpack_nas_SLQSNasNetworkTimeCallBack_ind
 - nas.h, 1305
- unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 1006
 - pDayltSavAdj, 1006
 - pRadioInterface, 1006
 - pTimeZone, 1006
 - universalTime, 1006
- unpack_nas_SLQSNasSigInfoCallback_ind
 - nas.h, 1306
- unpack_nas_SLQSNasSigInfoCallback_ind_t, 1007
 - pCDMASigInfo, 1007
 - pGSMSigInfo, 1007
 - pHDRSigInfo, 1007
 - pLTESigInfo, 1007
 - pRscp, 1007
 - pTDSCDMASigInfoExt, 1007
 - pWCDMASigInfo, 1007
- unpack_nas_SLQSNasSwiIndicationRegister
 - nas.h, 1306
- unpack_nas_SLQSNasSwiModemStatus
 - nas.h, 1306
- unpack_nas_SLQSNasSwiModemStatus_t, 1007
 - commonInfo, 1008
 - pLTEInfo, 1008
- unpack_nas_SLQSNasSwiOTAMessageCallback_ind
 - nas.h, 1307
- unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 1008
 - Info, 1008
 - Tlvresult, 1008
- unpack_nas_SLQSNasSysInfoCallback_ind
 - nas.h, 1307
- unpack_nas_SLQSNasTimerCallback_ind
 - nas.h, 1307
- unpack_nas_SLQSNasTimerCallback_ind_t, 1008
 - t3396_apn, 1009
 - t3396_plmn_id, 1009
 - t3396_val, 1009
- unpack_nas_SLQSSetBandPreference
 - nas.h, 1308
- unpack_nas_SLQSSetSignalStrengthsCallback
 - nas.h, 1308
- unpack_nas_SLQSSetSysSelectionPref
 - nas.h, 1308
- unpack_nas_SLQSSetSysSelectionPrefCallBack_ind
 - nas.h, 1309
- unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 1009
 - Info, 1010
 - Tlvresult, 1010

- unpack_nas_SLQSSwiGetLteCQI_t, 1010
 - CQIValueCW0, 1010
 - CQIValueCW1, 1010
 - ValidityCW0, 1010
 - ValidityCW1, 1010
- unpack_nas_SLQSSwiGetLteCQI
 - nas.h, 1309
- unpack_nas_SLQSSwiGetLteScsRxInfo
 - nas.h, 1309
- unpack_nas_SLQSSwiGetLteScsRxInfo_t, 1011
 - pScsRxInfo, 1011
- unpack_nas_SLQSSysInfoCallback_ind_t, 1011
 - pAddCDMASysInfo, 1013
 - pAddGSMSysInfo, 1013
 - pAddHDRSysInfo, 1013
 - pAddLTESysInfo, 1013
 - pAddWCDMASysInfo, 1013
 - pCDMASrvStatusInfo, 1013
 - pCDMASysInfo, 1013
 - pGSMCallBarringSysInfo, 1013
 - pGSMCipherDomainSysInfo, 1013
 - pGSMSrvStatusInfo, 1013
 - pGSMSysInfo, 1014
 - pHDRSrvStatusInfo, 1014
 - pHDRSysInfo, 1014
 - pLTESrvStatusInfo, 1014
 - pLTESysInfo, 1014
 - pLTEVoiceSupportSysInfo, 1014
 - pSysInfoNoChange, 1014
 - pWCDMACallBarringSysInfo, 1014
 - pWCDMACipherDomainSysInfo, 1014
 - pWCDMASrvStatusInfo, 1014
 - pWCDMASysInfo, 1014
- unpack_nas_SetACCOLC
 - nas.h, 1299
- unpack_nas_SetDataCapabilitiesCallback_ind
 - nas.h, 1300
- unpack_nas_SetDataCapabilitiesCallback_ind_t, 988
 - dataCaps, 988
 - dataCapsSize, 988
- unpack_nas_SetEventReportInd
 - nas.h, 1300
- unpack_nas_SetEventReportInd_t, 988
 - RFTlv, 988
 - RRTlv, 988
 - SLQSSSTlv, 989
 - SSTlv, 989
- unpack_nas_SetLURejectCallback
 - nas.h, 1300
- unpack_nas_SetNasLTECphyCaIndCallback_ind
 - nas.h, 1300
- unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 989
 - sPhyCaAggPcellInfo, 989
 - sPhyCaAggScellIDBw, 989
 - sPhyCaAggScellIndType, 989
 - sPhyCaAggScellIndex, 989
 - sPhyCaAggScellInfo, 989
- unpack_nas_SetNetworkPreference
 - nas.h, 1301
- unpack_nas_SetNetworkPreference_t, 990
 - Tlvresult, 990
- unpack_nas_SetRFInfoCallback
 - nas.h, 1301
- unpack_nas_SetRoamingIndicatorCallback_ind
 - nas.h, 1301
- unpack_nas_SetRoamingIndicatorCallback_ind_t, 990
 - roaming, 991
- unpack_nas_SetServingSystemCallback_ind
 - nas.h, 1301
- unpack_nas_SetServingSystemCallback_ind_t, 991
 - SSInfo, 991
 - Tlvresult, 991
- unpack_nas_SlqsGetLTECphyCAInfo
 - nas.h, 1301
- unpack_nas_SlqsGetLTECphyCAInfo_t, 991
 - LTECphyCAInfo, 992
 - Tlvresult, 992
- unpack_omaDmConfigTlv_t, 1014
 - alertmsg, 1015
 - alertmsglength, 1015
 - state, 1015
 - userInputReq, 1015
 - userInputTimeout, 1015
- unpack_omaDmFotaTlv_t, 1015
 - description, 1017
 - descriptionlength, 1017
 - fwloadsize, 1017
 - fwloadComplete, 1017
 - namelength, 1017
 - package_name, 1017
 - sessionType, 1017
 - severity, 1017
 - state, 1017
 - updateCompleteStatus, 1017
 - userInputReq, 1017
 - userInputTimeout, 1017
 - version, 1017
 - versionlength, 1017
- unpack_omaDmNotificationsTlv_t, 1017
 - notification, 1018
 - sessionStatus, 1018
- unpack_qmi_t, 1018
 - msgid, 1018
 - type, 1018
 - xid, 1018
- unpack_qos_IPv4Addr_t, 1019
 - addr, 1019
 - subnetMask, 1019
- unpack_qos_IPv6Addr_t, 1019
 - addr, 1020
 - prefixLen, 1020
- unpack_qos_IPv6TrafCls_t, 1020
 - mask, 1020
 - val, 1020
- unpack_qos_Port_t, 1021
 - port, 1021

- range, [1021](#)
- unpack_qos_QosFlowInfo_t, [1022](#)
 - BearerID, [1023](#)
 - is_RxQFlowGranted_Available, [1023](#)
 - is_TxQFlowGranted_Available, [1023](#)
 - NumRxFilters, [1023](#)
 - NumTxFilters, [1023](#)
 - QFlowState, [1023](#)
 - RxQFilter, [1023](#)
 - RxQFlowGranted, [1023](#)
 - TxQFilter, [1023](#)
 - TxQFlowGranted, [1023](#)
- unpack_qos_QosFlowInfoState_t, [1023](#)
 - id, [1024](#)
 - isNewFlow, [1024](#)
 - state, [1024](#)
- unpack_qos_SLQSQosGetNetworkStatus
 - qos.h, [1753](#)
- unpack_qos_SLQSQosGetNetworkStatus_t, [1024](#)
 - NWQoSStatus, [1024](#)
- unpack_qos_SLQSQosSwiReadApnExtraParams
 - qos.h, [1754](#)
- unpack_qos_SLQSQosSwiReadApnExtraParams_↔
 - t, [1025](#)
 - ambr_dl, [1026](#)
 - ambr_dl_ext, [1026](#)
 - ambr_dl_ext2, [1026](#)
 - ambr_ul, [1026](#)
 - ambr_ul_ext, [1026](#)
 - ambr_ul_ext2, [1026](#)
 - apnId, [1026](#)
- unpack_qos_SLQSQosSwiReadDataStats
 - qos.h, [1754](#)
- unpack_qos_SLQSQosSwiReadDataStats_t, [1026](#)
 - apnId, [1027](#)
 - numQosFlow, [1027](#)
 - qosFlow, [1027](#)
 - total_rx_bytes, [1027](#)
 - total_rx_pkt, [1027](#)
 - total_tx_bytes, [1027](#)
 - total_tx_bytes_drp, [1027](#)
 - total_tx_pkt, [1027](#)
 - total_tx_pkt_drp, [1027](#)
- unpack_qos_SLQSSetQosEventCallback
 - qos.h, [1755](#)
- unpack_qos_SLQSSetQosEventCallback_ind
 - qos.h, [1755](#)
- unpack_qos_SLQSSetQosEventCallback_ind_t, [1027](#)
 - NumFlows, [1028](#)
 - QosFlowInfo, [1028](#)
- unpack_qos_SLQSSetQosNWStatusCallback_ind
 - qos.h, [1756](#)
- unpack_qos_SLQSSetQosNWStatusCallback_ind_t,
 - [1028](#)
 - status, [1028](#)
- unpack_qos_SLQSSetQosPriEventCallback_ind
 - qos.h, [1757](#)
- unpack_qos_SLQSSetQosPriEventCallback_ind_↔
 - t, [1028](#)
 - event, [1029](#)
- unpack_qos_SLQSSetQosStatusCallback_ind
 - qos.h, [1757](#)
- unpack_qos_SLQSSetQosStatusCallback_ind_t, [1029](#)
 - event, [1030](#)
 - id, [1030](#)
 - reason, [1030](#)
 - status, [1030](#)
- unpack_qos_Tos_t, [1039](#)
 - mask, [1039](#)
 - val, [1039](#)
- unpack_qos_dataRate_t, [1018](#)
 - dataRateMax, [1019](#)
 - guaranteedRate, [1019](#)
- unpack_qos_pktErrRate_t, [1021](#)
 - exponent, [1021](#)
 - multiplier, [1021](#)
- unpack_qos_swiQosFilter_t, [1030](#)
 - EspSpi, [1032](#)
 - IPv4DstAddr, [1033](#)
 - IPv4SrcAddr, [1033](#)
 - IPv4Tos, [1033](#)
 - IPv6DstAddr, [1033](#)
 - IPv6Label, [1033](#)
 - IPv6SrcAddr, [1033](#)
 - IPv6TrafCls, [1033](#)
 - Id, [1032](#)
 - index, [1033](#)
 - is_EspSpi_Available, [1033](#)
 - is_IPv4DstAddr_Available, [1033](#)
 - is_IPv4SrcAddr_Available, [1033](#)
 - is_IPv4Tos_Available, [1033](#)
 - is_IPv6DstAddr_Available, [1033](#)
 - is_IPv6Label_Available, [1033](#)
 - is_IPv6SrcAddr_Available, [1033](#)
 - is_IPv6TrafCls_Available, [1033](#)
 - is_Id_Available, [1033](#)
 - is_NxtHdrProto_Available, [1033](#)
 - is_Precedence_Available, [1033](#)
 - is_TCPDstPort_Available, [1033](#)
 - is_TCPSrcPort_Available, [1033](#)
 - is_TranDstPort_Available, [1033](#)
 - is_TranSrcPort_Available, [1033](#)
 - is_UDPDstPort_Available, [1034](#)
 - is_UDPSrcPort_Available, [1034](#)
 - NxtHdrProto, [1034](#)
 - Precedence, [1034](#)
 - TCPDstPort, [1034](#)
 - TCPSrcPort, [1034](#)
 - TranDstPort, [1034](#)
 - TranSrcPort, [1034](#)
 - UDPDstPort, [1034](#)
 - UDPSrcPort, [1034](#)
 - version, [1034](#)
- unpack_qos_swiQosFlow_t, [1034](#)
 - DataRate, [1037](#)

- index, [1037](#)
- is_DataRate_Available, [1037](#)
- is_Jitter_Available, [1037](#)
- is_Latency_Available, [1037](#)
- is_LteQci_Available, [1037](#)
- is_MaxAllowedPktSz_Available, [1037](#)
- is_MinPolicedPktSz_Available, [1037](#)
- is_PktErrRate_Available, [1037](#)
- is_ProfileId3GPP2_Available, [1037](#)
- is_TokenBucket_Available, [1037](#)
- is_TrafficClass_Available, [1037](#)
- is_val_3GPP2Pri_Available, [1037](#)
- is_val_3GPPImCn_Available, [1037](#)
- is_val_3GPPResResidualBER_Available, [1037](#)
- is_val_3GPPSigInd_Available, [1037](#)
- is_val_3GPPTraHdlPri_Available, [1037](#)
- Jitter, [1037](#)
- Latency, [1038](#)
- LteQci, [1038](#)
- MaxAllowedPktSz, [1038](#)
- MinPolicedPktSz, [1038](#)
- PktErrRate, [1038](#)
- ProfileId3GPP2, [1038](#)
- TokenBucket, [1038](#)
- TrafficClass, [1038](#)
- val_3GPP2Pri, [1038](#)
- val_3GPPImCn, [1038](#)
- val_3GPPResResidualBER, [1038](#)
- val_3GPPSigInd, [1038](#)
- val_3GPPTraHdlPri, [1038](#)
- unpack_qos_tokenBucket_t, [1038](#)
 - bucketSz, [1039](#)
 - peakRate, [1039](#)
 - tokenRate, [1039](#)
- unpack_result_code_only
 - common.h, [1200](#)
- unpack_sms_SLQSDDeleteSMS_t, [1043](#)
- unpack_sms_SLQSDDeleteSMS
 - sms.h, [1766](#)
- unpack_sms_SLQSGetSMS_t, [1043](#)
 - message, [1044](#)
 - messageFormat, [1044](#)
 - messageSize, [1044](#)
 - messageTag, [1044](#)
- unpack_sms_SLQSGetSMSList
 - sms.h, [1766](#)
- unpack_sms_SLQSGetSMSList_t, [1044](#)
 - messageList, [1044](#)
 - messageListSize, [1044](#)
- unpack_sms_SLQSGetSMS
 - sms.h, [1766](#)
- unpack_sms_SLQSModifySMSStatus
 - sms.h, [1767](#)
- unpack_sms_SLQSModifySMSStatus_t, [1044](#)
- unpack_sms_SLQSWmsMemoryFullCallBack_ind
 - sms.h, [1767](#)
- unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, [1045](#)
- messageMode, [1045](#)
- storageType, [1045](#)
- unpack_sms_SendSMS_t, [1041](#)
 - messageFailureCode, [1042](#)
 - messageID, [1042](#)
- unpack_sms_SendSMS
 - sms.h, [1764](#)
- unpack_sms_SetNewSMSCallback
 - sms.h, [1765](#)
- unpack_sms_SetNewSMSCallback_ind
 - sms.h, [1765](#)
- unpack_sms_SetNewSMSCallback_ind_t, [1042](#)
 - ETWSPLMNTIv, [1043](#)
 - ETWSTIv, [1043](#)
 - IMSTIv, [1043](#)
 - MMTIv, [1043](#)
 - NewMMTIv, [1043](#)
 - SMSCTIv, [1043](#)
 - TRMessageTIv, [1043](#)
- unpack_sms_SetNewSMSCallback_t, [1043](#)
- unpack_swilloc_SwiLocGetAutoStart
 - swilloc.h, [1770](#)
- unpack_swilloc_SwiLocGetAutoStart_t, [1045](#)
 - fix_rate, [1046](#)
 - fix_rate_reported, [1046](#)
 - fix_type, [1046](#)
 - fix_type_reported, [1047](#)
 - function, [1047](#)
 - function_reported, [1047](#)
 - max_dist, [1047](#)
 - max_dist_reported, [1047](#)
 - max_time, [1047](#)
 - max_time_reported, [1047](#)
- unpack_swilloc_SwiLocSetAutoStart
 - swilloc.h, [1771](#)
- unpack_swima_SLQSOMADMAAlertCallback
 - swioma.h, [1777](#)
- unpack_swima_SLQSOMADMAAlertCallback_ind
 - swioma.h, [1777](#)
- unpack_swima_SLQSOMADMAAlertCallback_ind_↵
 - t, [1047](#)
 - eventType, [1048](#)
 - SessionInfoConfig, [1048](#)
 - SessionInfoFota, [1048](#)
 - SessionInfoNotification, [1048](#)
- unpack_swima_SLQSOMADMCancelSession
 - swioma.h, [1778](#)
- unpack_swima_SLQSOMADMGetSessionInfo
 - swioma.h, [1778](#)
- unpack_swima_SLQSOMADMGetSessionInfo_t, [1048](#)
 - Date, [1050](#)
 - DateLength, [1050](#)
 - PkgDescLength, [1050](#)
 - PkgDescription, [1050](#)
 - PkgName, [1050](#)
 - PkgNameLength, [1050](#)
 - RetryCount, [1050](#)
 - SessionState, [1050](#)

- SessionType, [1050](#)
- Severity, [1050](#)
- Source, [1050](#)
- SourceLength, [1050](#)
- Status, [1051](#)
- Time, [1051](#)
- TimeLength, [1051](#)
- UpdateCompleteStatus, [1051](#)
- unpack_swisma_SLQSOMADMGetSettings
 - swisma.h, [1779](#)
- unpack_swisma_SLQSOMADMGetSettings_t, [1051](#)
 - Autosdm, [1052](#)
 - FOTAUpdate, [1052](#)
 - FOTAdownload, [1052](#)
 - FwAutoCheck, [1052](#)
 - OMADMEEnabled, [1052](#)
- unpack_swisma_SLQSOMADMSendSelection
 - swisma.h, [1779](#)
- unpack_swisma_SLQSOMADMSetSettings
 - swisma.h, [1780](#)
- unpack_swisma_SLQSOMADMStartSession
 - swisma.h, [1780](#)
- unpack_swisma_SLQSOMADMStartSession_t, [1052](#)
 - FwAvailability, [1053](#)
- unpack_uim_ChangePin
 - uim.h, [1787](#)
- unpack_uim_ChangePin_t, [1053](#)
 - pEncryptedPIN1, [1053](#)
 - pIndicationToken, [1053](#)
 - pRemainingRetries, [1054](#)
 - Tlvresult, [1054](#)
- unpack_uim_GetCardStatus
 - uim.h, [1787](#)
- unpack_uim_GetCardStatus_t, [1054](#)
 - pCardStatus, [1054](#)
 - pHotSwapStatus, [1054](#)
 - Tlvresult, [1054](#)
- unpack_uim_ReadTransparent
 - uim.h, [1788](#)
- unpack_uim_ReadTransparent_t, [1054](#)
 - pCardResult, [1055](#)
 - pEncryptedData, [1055](#)
 - pIndicationToken, [1055](#)
 - pReadResult, [1055](#)
 - Tlvresult, [1055](#)
- unpack_uim_SLQSUIEventRegister
 - uim.h, [1789](#)
- unpack_uim_SLQSUIEventRegister_t, [1057](#)
 - eventMask, [1057](#)
- unpack_uim_SLQSUIGetSlotsStatus
 - uim.h, [1789](#)
- unpack_uim_SLQSUIGetSlotsStatus_t, [1057](#)
 - pNumberOfPhySlot, [1058](#)
 - pUimSlotsStatus, [1058](#)
- unpack_uim_SLQSUIPowerDown
 - uim.h, [1790](#)
- unpack_uim_SLQSUIPowerUp
 - uim.h, [1790](#)
- unpack_uim_SLQSUISetStatusChangeCallBack_ind
 - uim.h, [1791](#)
- unpack_uim_SLQSUISetStatusChangeCallBack_↔
 - ind_t, [1058](#)
 - pCardStatus, [1058](#)
- unpack_uim_SLQSUISwitchSlot
 - uim.h, [1791](#)
- unpack_uim_SetPinProtection
 - uim.h, [1788](#)
- unpack_uim_SetPinProtection_t, [1055](#)
 - pEncryptedPIN1, [1056](#)
 - pIndicationToken, [1056](#)
 - pRemainingRetries, [1056](#)
 - Tlvresult, [1056](#)
- unpack_uim_SetUimSlotStatusChangeCallback_ind
 - uim.h, [1789](#)
- unpack_uim_SetUimSlotStatusChangeCallback_ind_t, [1056](#)
 - bNumberOfPhySlots, [1057](#)
 - slotsstatusChange, [1057](#)
- unpack_uim_UnblockPin
 - uim.h, [1791](#)
- unpack_uim_UnblockPin_t, [1058](#)
 - pEncryptedPIN1, [1059](#)
 - pIndicationToken, [1059](#)
 - pRemainingRetries, [1059](#)
 - Tlvresult, [1059](#)
- unpack_uim_VerifyPin
 - uim.h, [1792](#)
- unpack_uim_VerifyPin_t, [1059](#)
 - pEncryptedPIN1, [1060](#)
 - pIndicationToken, [1060](#)
 - pRemainingRetries, [1060](#)
 - Tlvresult, [1060](#)
- unpack_wds_DHCPv4ClientLease_ind
 - wds.h, [1819](#)
- unpack_wds_DHCPv4ClientLease_ind_t, [1060](#)
 - DHCPv4LeaseOptTlv, [1060](#)
 - DHCPv4LeaseStateTlv, [1060](#)
 - IPv4AddrTlv, [1060](#)
 - ProfileIdTlv, [1060](#)
- unpack_wds_DHCPv4ClientLeaseChange
 - wds.h, [1820](#)
- unpack_wds_GetAutoconnect
 - wds.h, [1820](#)
- unpack_wds_GetAutoconnect_t, [1060](#)
 - psetting, [1061](#)
- unpack_wds_GetByteTotals
 - wds.h, [1820](#)
- unpack_wds_GetByteTotals_t, [1061](#)
 - pRXTotalBytes, [1061](#)
 - pTXTotalBytes, [1061](#)
- unpack_wds_GetConnectionRate
 - wds.h, [1821](#)
- unpack_wds_GetConnectionRate_t, [1061](#)
 - currentChannelRXRate, [1062](#)
 - currentChannelTXRate, [1062](#)
 - maxChannelRXRate, [1062](#)

- maxChannelTXRate, 1062
- unpack_wds_GetDataBearerTechnology
 - wds.h, 1821
- unpack_wds_GetDataBearerTechnology_t, 1062
 - pDataBearer, 1063
- unpack_wds_GetDefaultProfile
 - wds.h, 1822
- unpack_wds_GetDefaultProfile_t, 1063
 - apnname, 1063
 - apnsize, 1063
 - auth, 1063
 - ipaddr, 1063
 - ipaddrv6, 1064
 - name, 1064
 - namesize, 1064
 - pdptype, 1064
 - pridns, 1064
 - pridnsv6, 1064
 - secdns, 1064
 - secdnsv6, 1064
 - username, 1064
 - usersize, 1064
- unpack_wds_GetDefaultProfileNum
 - wds.h, 1822
- unpack_wds_GetDefaultProfileNum_t, 1064
 - index, 1064
- unpack_wds_GetDormancyState
 - wds.h, 1822
- unpack_wds_GetDormancyState_t, 1064
 - dormancyState, 1065
- unpack_wds_GetLastMobileIPError
 - wds.h, 1823
- unpack_wds_GetLastMobileIPError_t, 1065
 - error, 1065
- unpack_wds_GetMobileIP_t, 1065
 - mipMode, 1065
- unpack_wds_GetMobileIPProfile
 - wds.h, 1824
- unpack_wds_GetMobileIPProfile_t, 1065
 - AAASPI, 1066
 - AAASState, 1066
 - address, 1066
 - enabled, 1066
 - HASPI, 1066
 - HASState, 1066
 - NAI, 1066
 - naiSize, 1066
 - primaryHA, 1066
 - revTunneling, 1066
 - secondaryHA, 1067
- unpack_wds_GetMobileIP
 - wds.h, 1823
- unpack_wds_GetPacketStatistics
 - wds.h, 1824
- unpack_wds_GetPacketStatistics_t, 1067
 - pRXDroppedCount, 1068
 - pRXOKBytesLastCall, 1068
 - pRXOkBytesCount, 1068
 - pRXPacketErrors, 1068
 - pRXPacketOverflows, 1068
 - pRXPacketSuccesses, 1068
 - pTXDroppedCount, 1068
 - pTXOKBytesLastCall, 1068
 - pTXOkBytesCount, 1068
 - pTXPacketErrors, 1068
 - pTXPacketOverflows, 1068
 - pTXPacketSuccesses, 1068
- unpack_wds_GetPacketStatus
 - wds.h, 1824
- unpack_wds_GetPacketStatus_t, 1068
 - rXDroppedCount, 1069
 - rXOKBytesLastCall, 1069
 - rXOkBytesCount, 1069
 - rXPacketErrors, 1069
 - rXPacketOverflows, 1069
 - rXPacketSuccesses, 1069
 - tXDroppedCount, 1069
 - tXOKBytesLastCall, 1070
 - tXOkBytesCount, 1069
 - tXPacketErrors, 1070
 - tXPacketOverflows, 1070
 - tXPacketSuccesses, 1070
- unpack_wds_GetSessionDuration
 - wds.h, 1825
- unpack_wds_GetSessionDuration_t, 1070
 - callDuration, 1070
- unpack_wds_GetSessionState
 - wds.h, 1825
- unpack_wds_GetSessionState_t, 1070
 - connectionStatus, 1070
- unpack_wds_RMSetTransferStatistics
 - wds.h, 1826
- unpack_wds_RMSetTransferStatistics_t, 1071
- unpack_wds_RMTransferStatistics_ind
 - wds.h, 1826
- unpack_wds_SLQSCreateProfile
 - wds.h, 1829
- unpack_wds_SLQSCreateProfile_t, 1071
 - pCreateProfileOut, 1071
 - pProfileID, 1071
 - Tlvresult, 1071
- unpack_wds_SLQSDeleteProfile
 - wds.h, 1829
- unpack_wds_SLQSDeleteProfile_t, 1071
 - extendedErrorCode, 1071
- unpack_wds_SLQSGet3GPPConfigItem
 - wds.h, 1829
- unpack_wds_SLQSGet3GPPConfigItem_t, 1072
 - _3gppRelease, 1073
 - defaultPDNEEnabled, 1073
 - LTEAttachProfile, 1073
 - LTEAttachProfileList, 1073
 - LTEAttachProfileListLen, 1073
 - profileList, 1073
- unpack_wds_SLQSGetCurrDataSystemStat
 - wds.h, 1830

- unpack_wds_SLQSGetCurrDataSystemStat_t, 1073
 - currNetworkInfo, 1073
 - networkInfoLen, 1073
 - prefNetwork, 1073
- unpack_wds_SLQSGetCurrentChannelRate
 - wds.h, 1830
- unpack_wds_SLQSGetCurrentChannelRate_t, 1074
 - current_channel_rx_rate, 1074
 - current_channel_tx_rate, 1074
 - max_channel_rx_rate, 1074
 - max_channel_tx_rate, 1074
- unpack_wds_SLQSGetDUNCallInfo
 - wds.h, 1831
- unpack_wds_SLQSGetDUNCallInfo_t, 1075
 - callEndReason, 1076
 - channelRate, 1076
 - connectionStatus, 1076
 - dataBearerTech, 1076
 - dormancyStatus, 1076
 - lastCallDataBearerTech, 1076
 - lastCallRXOKBytesCnt, 1076
 - lastCallTXOKBytesCnt, 1076
 - mdmCallDurationActive, 1076
 - rxOKBytesCount, 1076
 - txOKBytesCount, 1076
- unpack_wds_SLQSGetDataBearerTechnology
 - wds.h, 1831
- unpack_wds_SLQSGetDataBearerTechnology_t, 1075
 - curDataBearerTechnology, 1075
 - dataBearerMask, 1075
 - lastCallDataBearerTechnology, 1075
- unpack_wds_SLQSGetProfileSettings
 - wds.h, 1831
- unpack_wds_SLQSGetProfileSettings_t, 1076
 - pProfileSettings, 1077
 - ProfileType, 1077
 - Tlvresult, 1077
- unpack_wds_SLQSGetRuntimeSettings
 - wds.h, 1832
- unpack_wds_SLQSGetRuntimeSettings_t, 1077
 - APNName, 1078
 - Authentication, 1078
 - DomainList, 1078
 - GPRSGrantedQoS, 1078
 - GWAddressV4, 1078
 - IMCNflag, 1078
 - IPFamilyPreference, 1078
 - IPv6AddrInfo, 1078
 - IPv6GWAddrInfo, 1078
 - IPv4, 1078
 - Mtu, 1078
 - PCSCFAddrPCO, 1078
 - PCSCFFQDNAddrList, 1078
 - PDPTType, 1078
 - PrimaryDNSV4, 1079
 - PrimaryDNSV6, 1079
 - ProfileID, 1079
 - ProfileName, 1079
 - SecondaryDNSV4, 1079
 - SecondaryDNSV6, 1079
 - ServerAddrList, 1079
 - SubnetMaskV4, 1079
 - Technology, 1079
 - UMTSGrantedQoS, 1079
 - Username, 1079
- unpack_wds_SLQSMModifyProfile
 - wds.h, 1832
- unpack_wds_SLQSMModifyProfile_t, 1079
 - pExtErrorCode, 1079
- unpack_wds_SLQSResetPacketStatics
 - wds.h, 1832
- unpack_wds_SLQSSGetDHCPv4ClientConfig
 - wds.h, 1834
- unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 1082
 - pHwConfig, 1083
 - pRequestOptionList, 1083
- unpack_wds_SLQSSGetLoopback
 - wds.h, 1835
- unpack_wds_SLQSSGetLoopback_t, 1083
 - ByteLoopbackMode, 1083
 - ByteLoopbackMultiplier, 1083
- unpack_wds_SLQSSSetDHCPv4ClientConfig
 - wds.h, 1835
- unpack_wds_SLQSSSetLoopback
 - wds.h, 1836
- unpack_wds_SLQSSet3GPPConfigItem
 - wds.h, 1833
- unpack_wds_SLQSSetIPFamilyPreference
 - wds.h, 1833
- unpack_wds_SLQSSetIPFamilyPreference_t, 1079
 - Tlvresult, 1080
- unpack_wds_SLQSSetPacketSrvStatusCallback
 - wds.h, 1833
- unpack_wds_SLQSSetPacketSrvStatusCallback_↵
 - t, 1080
 - bearerID, 1080
 - conn_status, 1080
 - ipFamily, 1080
 - reconfigReqd, 1081
 - sessionEndReason, 1081
 - techName, 1081
 - verboseSessnEndReason, 1081
 - verboseSessnEndReasonType, 1081
- unpack_wds_SLQSSetWdsEventCallback
 - wds.h, 1834
- unpack_wds_SLQSSetWdsEventCallback_ind
 - wds.h, 1834
- unpack_wds_SLQSSetWdsEventCallback_ind_t, 1081
 - currDBTechAvail, 1082
 - currNWInfo, 1082
 - dBTechAvail, 1082
 - dBTechnology, 1082
 - dataSysStatAvail, 1082
 - dormancyStatAvail, 1082
 - dormancyStatus, 1082
 - mipStatus, 1082

- mipstatAvail, [1082](#)
- netInfoLen, [1082](#)
- prefNetwork, [1082](#)
- ratMask, [1082](#)
- rx_bytes, [1082](#)
- rx_pkts, [1082](#)
- soMask, [1082](#)
- tx_bytes, [1082](#)
- tx_pkts, [1082](#)
- xferStatAvail, [1082](#)
- unpack_wds_SLQSSStartDataSession
 - wds.h, [1836](#)
- unpack_wds_SLQSSStartDataSession_t, [1083](#)
 - pFailureReason, [1084](#)
 - pVerboseFailReasonType, [1084](#)
 - pVerboseFailureReason, [1084](#)
 - psid, [1084](#)
- unpack_wds_SLQSSStopDataSession
 - wds.h, [1836](#)
- unpack_wds_SLQSWdsGoActive
 - wds.h, [1837](#)
- unpack_wds_SLQSWdsGoDormant
 - wds.h, [1837](#)
- unpack_wds_SLQSWdsSetEventReport
 - wds.h, [1837](#)
- unpack_wds_SLQSWdsSwiPDPRuntimeSettings
 - wds.h, [1838](#)
- unpack_wds_SLQSWdsSwiPDPRuntimeSettings_↔
 - t, [1084](#)
 - apnName, [1085](#)
 - bearerId, [1085](#)
 - contextId, [1085](#)
 - ipv4Address, [1085](#)
 - ipv4GWAddress, [1085](#)
 - ipv6Address, [1085](#)
 - ipv6GWAddress, [1085](#)
 - prDNSIPv4Address, [1085](#)
 - prDNSIPv6Address, [1085](#)
 - prPCSCFIPv4Address, [1085](#)
 - prPCSCFIPv6Address, [1085](#)
 - seDNSIPv4Address, [1085](#)
 - seDNSIPv6Address, [1086](#)
 - sePCSCFIPv4Address, [1086](#)
 - sePCSCFIPv6Address, [1086](#)
- unpack_wds_SetAutoconnect
 - wds.h, [1826](#)
- unpack_wds_SetDefaultProfile
 - wds.h, [1827](#)
- unpack_wds_SetDefaultProfileNum
 - wds.h, [1827](#)
- unpack_wds_SetMobileIPParameters
 - wds.h, [1828](#)
- unpack_wds_SetMobileIPProfile
 - wds.h, [1828](#)
- unpack_wds_SetMobileIPProfile_t, [1071](#)
- unpack_wds_SetMobileIP
 - wds.h, [1827](#)
- UnpackQmiProfileInfo
 - wds.h, [1798](#)
- unpackWdsProfileParam, [1086](#)
 - SlqsProfile3GPP2, [1086](#)
 - SlqsProfile3GPP, [1086](#)
- upLink
 - NSSAudioCtrl, [541](#)
- UpdateCompleteStatus
 - unpack_swioma_SLQSOMADMGetSessionInfo_↔
 - _t, [1051](#)
- updateCompleteStatus
 - omaDmFotaTlv, [547](#)
 - unpack_omaDmFotaTlv_t, [1017](#)
- upgrade_mc77xx_fw
 - qaGobiApiFms.h, [1475](#)
- UpgradeFirmware2k
 - qaGobiApiFms.h, [1475](#)
- upinRetries
 - slotInf, [786](#)
 - slotInfo, [787](#)
 - uim_slotInfo, [887](#)
- upinState
 - slotInf, [786](#)
 - slotInfo, [787](#)
 - uim_slotInfo, [887](#)
- UpkQmiCbkCatEventReportInd
 - qaCbkCatEventReportInd.h, [1311](#)
- UpkQmiCbkSwiOmaDmEventReportInd
 - qaCbkSwiOmaDmEventReportInd.h, [1312](#)
- UpkQmiCbkSwiOmaDmEventReportIndExt
 - qaCbkSwiOmaDmEventReportInd.h, [1312](#)
- UpkQmiNasGetRFBandInfo
 - qaNasGetRFBandInfo.h, [1741](#)
- UpkQmiNasPerformNetworkScan
 - qaNasPerformNetworkScan.h, [1742](#)
- upukRetries
 - slotInf, [786](#)
 - slotInfo, [787](#)
 - uim_slotInfo, [887](#)
- usageMask
 - loc_sensorDataUsage, [361](#)
 - sensorDataUsage_s, [747](#)
 - t_sensor, [852](#)
- User Identity Module Service (UIM), [48](#)
- userData
 - SMSAsyncRawSend_s, [806](#)
- userInputReq
 - omaDmConfigTlv, [543](#)
 - omaDmConfigTlvExt, [545](#)
 - omaDmFotaTlv, [547](#)
 - unpack_omaDmConfigTlv_t, [1015](#)
 - unpack_omaDmFotaTlv_t, [1017](#)
- userInputTimeout
 - omaDmConfigTlv, [543](#)
 - omaDmConfigTlvExt, [545](#)
 - omaDmFotaTlv, [547](#)
 - omaDmFotaTlvExt, [549](#)
 - unpack_omaDmConfigTlv_t, [1015](#)
 - unpack_omaDmFotaTlv_t, [1017](#)

- Username
 - unpack_wds_SLQSGetRuntimeSettings_t, 1079
- username
 - unpack_wds_GetDefaultProfile_t, 1064
- usersize
 - unpack_wds_GetDefaultProfile_t, 1064
- ussDCS
 - USSInfo, 1091
- ussData
 - USSInfo, 1091
- ussLen
 - USSInfo, 1091
- uusInfo
 - allCallsUUSInfo, 102
- v4sessionId
 - qaQmiInterfaceInfo, 679
 - ssdatasession_params, 827
 - WdsRunTimeSettings, 1190
- v6sessionId
 - qaQmiInterfaceInfo, 679
 - ssdatasession_params, 827
 - WdsRunTimeSettings, 1190
- VDOP
 - loc_precisionDilution, 358
 - precisionDilution_s, 660
- VOICE_SRV
 - qaGobiApiCbK.h, 1328
- VOICE_SUPS_SRV_CLASS_DATA_CIRCUIT_ASYNC
 - qaGobiApiVoice.h, 1676
- VOICE_SUPS_SRV_CLASS_DATA_CIRCUITSYNC
 - qaGobiApiVoice.h, 1676
- VOICE_SUPS_SRV_CLASS_DATA
 - qaGobiApiVoice.h, 1676
- VOICE_SUPS_SRV_CLASS_FAX
 - qaGobiApiVoice.h, 1676
- VOICE_SUPS_SRV_CLASS_NONE
 - qaGobiApiVoice.h, 1676
- VOICE_SUPS_SRV_CLASS_PACKETACCESS
 - qaGobiApiVoice.h, 1676
- VOICE_SUPS_SRV_CLASS_PADACCESS
 - qaGobiApiVoice.h, 1676
- VOICE_SUPS_SRV_CLASS_SMS
 - qaGobiApiVoice.h, 1676
- VOICE_SUPS_SRV_CLASS_VOICE
 - qaGobiApiVoice.h, 1676
- val
 - IPv6TrafCls, 320
 - Tos, 867
 - unpack_qos_IPv6TrafCls_t, 1020
 - unpack_qos_Tos_t, 1039
- val_3GPP2Pri
 - unpack_qos_swiQosFlow_t, 1038
- val_3GPPImCn
 - unpack_qos_swiQosFlow_t, 1038
- val_3GPPResResidualBER
 - unpack_qos_swiQosFlow_t, 1038
- val_3GPPSigInd
 - unpack_qos_swiQosFlow_t, 1038
- val_3GPPTraHdIPri
 - unpack_qos_swiQosFlow_t, 1038
- ValidMask
 - GPSSStateInfo, 277
- validMask
 - loc_satelliteInfo, 360
 - satelliteInfo, 742
- ValidateSPC
 - qaGobiApiDms.h, 1455
- ValidityCW0
 - LteCQIParm, 377
 - unpack_nas_SLQSSwiGetLteCQI_t, 1010
- ValidityCW1
 - LteCQIParm, 377
 - unpack_nas_SLQSSwiGetLteCQI_t, 1010
- Value
 - GetM2MSpkrGainResp, 265
 - SetM2MSpkrGainReq, 770
- value
 - pack_dms_UIMSetPINProtection_t, 562
 - pack_dms_UIMVerifyPIN_t, 564
- value_length
 - custSettingInfo, 192
 - DMScustSettingInfo, 211
 - pack_dms_SetCustFeaturesV2_t, 555
 - setCustomSettingV2, 760
- verbFailReason
 - ssdatasession_params, 827
- verbFailReasonType
 - ssdatasession_params, 827
- verboseSessnEndReason
 - _packetSrvStatus, 64
 - unpack_wds_SLQSSetPacketSrvStatusCallback↔_t, 1081
- verboseSessnEndReasonType
 - _packetSrvStatus, 64
 - unpack_wds_SLQSSetPacketSrvStatusCallback↔_t, 1081
- verifyLeft
 - personalizationStatus, 650
 - remainingRetries, 723
 - uim_remainingRetries, 884
- verifyPIN
 - pack_uim_VerifyPin_t, 619
 - UIMVerifyPinReq, 913
- verifyRetriesLeft
 - unpack_dms_UIMGetControlKeyStatus_t, 956
 - unpack_dms_UIMSetControlKeyProtection_t, 959
 - unpack_dms_UIMSetPINProtection_t, 959
- verifyUIMPIN, 1093
 - pinID, 1094
 - pinLen, 1094
 - pinVal, 1094
- version
 - omaDmFotaTlv, 547
 - omaDmFotaTlvExt, 549
 - swiQosFilter, 842
 - unpack_omaDmFotaTlv_t, 1017

- unpack_qos_swiQosFilter_t, 1034
- versionString
 - pack_dms_SLQSSwiSetOSInfo_t, 559
 - unpack_dms_SLQSSwiGetOSInfo_t, 952
- versionlength
 - omaDmFotaTlv, 547
 - omaDmFotaTlvExt, 549
 - unpack_omaDmFotaTlv_t, 1017
- vertConfidence
 - pack_loc_SLQSLOCInjectPosition_t, 577
- vertReliability
 - pack_loc_SLQSLOCInjectPosition_t, 577
- vertUnc
 - pack_loc_SLQSLOCInjectPosition_t, 577
- VerticalUncertainty
 - GPSSStateInfo, 277
- VirtStream
 - protocolSubtypeElement, 677
- Voice Service (VOICE), 46
- voiceALSSelectLineInfo, 1094
 - lineValue, 1094
- voiceALSSetLineSwitchInfo, 1094
 - switchOption, 1095
- voiceAnswerCall, 1095
 - pCallId, 1095
- voiceBindSubscriptionInfo, 1095
 - subsType, 1095
- voiceBurstDTMFInfo, 1096
 - BurstDTMFInfo, 1096
 - pBurstDTMFLengths, 1096
- voiceCallInfoReq, 1096
 - callID, 1097
- voiceCallInfoResp, 1097
 - pAlertType, 1099
 - pAlertingPattern, 1099
 - pAlphaIDInfo, 1099
 - pCallInfo, 1099
 - pConnectNumInfo, 1099
 - pDiagInfo, 1099
 - pOTASPStatus, 1099
 - pRemotePartyName, 1099
 - pRemotePartyNum, 1099
 - pSrvOpt, 1099
 - pUUSInfo, 1099
 - pVoicePrivacy, 1099
- voiceCallRequestParams, 1099
 - callNumber, 1101
 - pCLIRType, 1101
 - pCUGInfo, 1101
 - pCallPartySubAdd, 1101
 - pCallType, 1101
 - pEmergencyCategory, 1101
 - pSvcType, 1101
 - pUUSInfo, 1101
- voiceCallResponseParams, 1101
 - pAlphaIDInfo, 1102
 - pCCRResultType, 1102
 - pCCSUPSType, 1102
 - pCallID, 1102
- voiceContDTMFInfo, 1102
 - DTMFDigit, 1103
 - pCallID, 1103
- voiceDTMFEventInfo, 1103
 - DTMFInformation, 1104
 - pOffLength, 1104
 - pOnLength, 1104
- voiceFlashInfo, 1104
 - pCallID, 1104
 - pFlashPayLd, 1104
 - pFlashType, 1104
- voiceGetAllCallInfo, 1105
 - pArrAlertingPattern, 1107
 - pArrAlertingType, 1107
 - pArrAlphaID, 1107
 - pArrCallEndReason, 1107
 - pArrCallInfo, 1107
 - pArrCalledPartyNum, 1107
 - pArrConnectPartyNum, 1107
 - pArrDiagInfo, 1107
 - pArrRedirPartyNum, 1107
 - pArrRemotePartyName, 1107
 - pArrRemotePartyNum, 1107
 - pArrSvcOption, 1107
 - pArrUUSInfo, 1107
 - pOTASPStatus, 1107
 - pVoicePrivacy, 1107
- voiceGetCLIPResp, 1113
 - pAlphaIDInfo, 1114
 - pCCRResType, 1114
 - pCCSUPSType, 1114
 - pCLIPResp, 1114
 - pCallID, 1114
 - pFailCause, 1114
- voiceGetCLIRResp, 1115
 - pAlphaIDInfo, 1116
 - pCCRResType, 1116
 - pCCSUPSType, 1116
 - pCLIRResp, 1116
 - pCallID, 1116
 - pFailCause, 1116
- voiceGetCNAPResp, 1116
 - pAlphaIDInfo, 1117
 - pCCRResType, 1117
 - pCCSUPSType, 1117
 - pCNAPResp, 1117
 - pCallID, 1117
 - pFailCause, 1117
- voiceGetCOLPResp, 1117
 - pAlphaIDInfo, 1118
 - pCCRResType, 1118
 - pCCSUPSType, 1118
 - pCOLPResp, 1118
 - pCallID, 1118
 - pFailCause, 1118
- voiceGetCOLRResp, 1119
 - pAlphaIDInfo, 1120

- pCCResType, 1120
 - pCCSUPSType, 1120
 - pCOLRResp, 1120
 - pCallID, 1120
 - pFailCause, 1120
- voiceGetCallBarringReq, 1107
 - pSvcClass, 1108
 - reason, 1108
- voiceGetCallBarringResp, 1108
 - pAlphaIDInfo, 1109
 - pCCResType, 1109
 - pCCSUPSType, 1109
 - pCallID, 1109
 - pFailCause, 1109
 - pSvcClass, 1109
- voiceGetCallFWReq, 1110
 - pSvcClass, 1110
 - Reason, 1110
- voiceGetCallFWResp, 1110
 - pAlphaIDInfo, 1111
 - pCCResType, 1111
 - pCCSUPSType, 1112
 - pCallID, 1111
 - pFailCause, 1112
 - pGetCallFWExtInfo, 1112
 - pGetCallFWInfo, 1112
- voiceGetCallWaitInfo, 1112
 - pAlphaIDInfo, 1113
 - pCCResType, 1113
 - pCCSUPSType, 1113
 - pCallID, 1113
 - pFailCause, 1113
 - pSvcClass, 1113
- voiceGetConfigReq, 1120
 - pAMRStatus, 1121
 - pAirTimer, 1121
 - pAutoAnswer, 1121
 - pNameID, 1121
 - pPrefVoicePrivacy, 1121
 - pPrefVoiceSO, 1121
 - pRoamTimer, 1121
 - pTTYMode, 1121
 - pVoiceDomainPref, 1122
- voiceGetConfigResp, 1122
 - pAirTimerCnt, 1123
 - pAutoAnswerStat, 1123
 - pCurAMRConfig, 1123
 - pCurPrefVoiceSO, 1123
 - pCurVoiceDomainPref, 1123
 - pCurVoicePrivacyPref, 1123
 - pCurrTTYMode, 1123
 - pRoamTimerCnt, 1123
- voiceIndicationRegisterInfo, 1124
 - pRegDTMFEvents, 1124
 - pRegVoicePrivacyEvents, 1124
 - pSuppsNotifEvents, 1124
- voiceInfoRec, 1125
 - callID, 1126
 - pCLIRCause, 1126
 - pCallWaitInd, 1126
 - pCalledPartyInfo, 1126
 - pCallerIDInfo, 1126
 - pCallerNameInfo, 1126
 - pCallingPartyInfo, 1126
 - pConnectNumInfo, 1127
 - pDispInfo, 1127
 - pExtDispInfo, 1127
 - pExtDispRecInfo, 1127
 - pLineCtrlInfo, 1127
 - pNSSAudioCtrl, 1127
 - pNSSRelease, 1127
 - pRedirNumInfo, 1127
 - pSignalInfo, 1127
- voiceManageCallsReq, 1127
 - pCallID, 1128
 - SUPSType, 1128
- voiceManageCallsResp, 1128
 - pFailCause, 1128
- VoiceNumber
 - unpack_dms_GetVoiceNumber_t, 938
- voiceNumberSize
 - unpack_dms_GetVoiceNumber_t, 938
- voiceOTASPStatusInfo, 1129
 - callID, 1130
 - OTASPStatus, 1130
- voiceOrigUSSDNoWaitInfo, 1128
 - USSInformation, 1129
- voicePrivacy
 - voicePrivacyInfo, 1130
- voicePrivacyInfo, 1130
 - callID, 1130
 - voicePrivacy, 1130
- voiceSUPSInfo, 1142
 - pAlphaIDInfo, 1144
 - pCLIPstatus, 1144
 - pCLIRstatus, 1144
 - pCNAPstatus, 1144
 - pCOLPstatus, 1144
 - pCOLRstatus, 1144
 - pCallBarPasswd, 1144
 - pCallFWNum, 1144
 - pCallFWTimerVal, 1144
 - pCallFwdInfo, 1144
 - pCallID, 1144
 - pDataSrc, 1144
 - pFailCause, 1144
 - pNewPwdData, 1145
 - pReason, 1145
 - pSvcClass, 1145
 - pUSSInfo, 1145
 - SUPSInformation, 1145
- voiceSUPSNotification, 1145
 - callID, 1146
 - notifType, 1147
 - pCUGIndex, 1147
 - pECTNum, 1147

- voiceSetAllCallStatusCbKInfo, 1130
 - arrCallInformation, 1132
 - pArrAlertingPattern, 1132
 - pArrAlertingType, 1132
 - pArrAlphaID, 1132
 - pArrCallEndReason, 1132
 - pArrCalledPartyNum, 1132
 - pArrConnectPartyNum, 1132
 - pArrDiagInfo, 1132
 - pArrRedirPartyNum, 1132
 - pArrRemotePartyName, 1132
 - pArrRemotePartyNum, 1133
 - pArrSvcOption, 1133
- voiceSetCallBarringPwdInfo, 1133
 - newPasswd, 1134
 - newPasswdAgain, 1134
 - oldPasswd, 1134
 - Reason, 1134
- voiceSetCallBarringPwdResp, 1134
 - pAlphaIDInfo, 1135
 - pCCResType, 1135
 - pCCSUPSType, 1135
 - pCallID, 1135
 - pFailCause, 1135
- voiceSetConfigReq, 1135
 - pAirTimerConfig, 1136
 - pAutoAnswer, 1136
 - pPrefVoiceDomain, 1136
 - pPrefVoiceSO, 1136
 - pRoamTimerConfig, 1136
 - pTTYMode, 1136
- voiceSetConfigResp, 1136
 - pAirTimerStatus, 1138
 - pAutoAnsStatus, 1138
 - pPrefVoiceSOSStatus, 1138
 - pRoamTimerStatus, 1138
 - pTTYConfigStatus, 1138
 - pVoiceDomainPrefStatus, 1138
- voiceSetPrefPrivacy, 1138
 - privacyPref, 1138
- voiceSetSUPSServiceReq, 1138
 - pCallBarringPasswd, 1140
 - pCallForwardingNumber, 1140
 - pCallFwdTypeAndPlan, 1140
 - pServiceClass, 1140
 - pTimerVal, 1140
 - reason, 1140
 - voiceSvc, 1140
- voiceSetSUPSServiceResp, 1140
 - pAlphaIDInfo, 1141
 - pCCResultType, 1141
 - pCCSUPSType, 1141
 - pCallID, 1141
 - pFailCause, 1141
- voiceStopContDTMFInfo, 1141
 - callID, 1142
- voiceSvc
 - voiceSetSUPSServiceReq, 1140
- VolValue
 - SetAudioVolTLBConfigReq, 759
- Volume
 - GetAudioProfileResp, 245
 - GetAudioVolTLBConfigReq, 246
 - GetM2MAudioProfileResp, 262
 - SetAudioProfileReq, 757
 - SetAudioVolTLBConfigReq, 758
- voteForInit
 - registerRefresh, 723
- WCDMACellInfo
 - lteWcdmaCellInfo, 398
 - nas_lteWcdmaCellInfo, 453
- WCDMAECIOThresh, 1148
 - pWCDMAECIOThreshList, 1148
 - WCDMAECIOThreshListLen, 1148
- WCDMAECIOThreshListLen
 - nas_WCDMAECIOThresh, 489
 - WCDMAECIOThresh, 1148
- WCDMAInfoLTENeighborCell, 1148
 - UMTSLTENbrCell, 1149
 - umtsLTENbrCellLen, 1149
 - wcdmaRRCState, 1149
- WCDMARSSIThresh, 1154
 - pWCDMARSSIThreshList, 1154
 - WCDMARSSIThreshListLen, 1154
- WCDMARSSIThreshListLen
 - nas_WCDMARSSIThresh, 490
 - WCDMARSSIThresh, 1154
- WCDMASSInfo
 - unpack_nas_SLQSNasGetSigInfo_t, 1005
- WCDMASysInfo, 1154
 - cellId, 1157
 - cellIdValid, 1157
 - hsCallStatus, 1157
 - hsCallStatusValid, 1157
 - hsInd, 1157
 - hsIndValid, 1157
 - lac, 1157
 - lacValid, 1157
 - MCC, 1157
 - MNC, 1157
 - networkIdValid, 1157
 - psc, 1158
 - pscValid, 1158
 - regRejectInfoValid, 1158
 - rejCause, 1158
 - rejectSrvDomain, 1158
 - sysInfoWCDMA, 1158
- WDS_DHCP_MAX_NUM_OPTIONS
 - wds.h, 1798
- WDS_DHCP_OPTION_DATA_BUF_SIZE
 - wds.h, 1798
- WDS_IsGobiDevice
 - qaGobiApiWds.h, 1740
- WDS_SRV
 - qaGobiApiCbK.h, 1328
- WDSGetLoopbackData, 1184

- ByteLoopbackMode, 1185
- ByteLoopbackMultiplier, 1185
- WDSSWICurrentChannelRates, 1192
 - current_channel_rx_rate, 1193
 - current_channel_tx_rate, 1193
 - max_channel_rx_rate, 1193
 - max_channel_tx_rate, 1193
- WDSSetLoopbackData, 1192
 - pLoopbackMode, 1192
 - pLoopbackMultiplier, 1192
- WORD
 - SwiDataTypes.h, 1769
- wcdmaAmrStat
 - curAMRConfig, 180
- wcdmaCellInfo, 1147
 - cpich_ecno, 1147
 - cpich_rscp, 1147
 - psc, 1147
 - srxlev, 1148
- wcdmaLongMsgDecodingParams, 1149
 - Date, 1151
 - plsUDHPresent, 1151
 - pMessage, 1151
 - pPartNum, 1151
 - pReferenceNum, 1151
 - pScAddr, 1151
 - pScAddrLength, 1151
 - pSenderAddr, 1151
 - pSenderAddrLength, 1151
 - pTextMsg, 1151
 - pTextMsgLength, 1151
 - pTotalNum, 1151
 - Time, 1151
- wcdmaMsgDecodingParams, 1151
 - Date, 1152
 - pMessage, 1152
 - pScAddr, 1152
 - pScAddrLength, 1152
 - pSenderAddr, 1152
 - pSenderAddrLength, 1152
 - pTextMsg, 1152
 - pTextMsgLength, 1153
 - Time, 1153
- wcdmaMsgEncodingParams, 1153
 - alphabet, 1153
 - messageSize, 1153
 - pDestAddr, 1153
 - pPDUMessage, 1154
 - pTextMsg, 1154
- wcdmaRRCTest
 - nas_WCDMAInfoLTENeighborCell, 490
 - WCDMAInfoLTENeighborCell, 1149
- wcdmaUARFCN, 1158
 - status, 1158
 - uarfcn, 1158
- wds.h, 1792
 - BYT_STAT_STAT_MASK, 1798
 - IPV6_ADDRESS_ARRAY_SIZE, 1798
 - MAX_WDS_3GPP_CONF_LTE_ATTACH_PRO←
FILE_LIST_SIZE, 1798
 - PACK_WDS_IPV4, 1798
 - PACK_WDS_IPV6, 1798
 - pack_wds_DHCPv4ClientLeaseChange, 1798
 - pack_wds_GetAutoconnect, 1799
 - pack_wds_GetByteTotals, 1799
 - pack_wds_GetConnectionRate, 1800
 - pack_wds_GetDataBearerTechnology, 1800
 - pack_wds_GetDefaultProfile, 1801
 - pack_wds_GetDefaultProfileNum, 1801
 - pack_wds_GetDormancyState, 1801
 - pack_wds_GetLastMobileIPError, 1802
 - pack_wds_GetMobileIPProfile, 1803
 - pack_wds_GetMobileIP, 1802
 - pack_wds_GetPacketStatistics, 1803
 - pack_wds_GetPacketStatus, 1804
 - pack_wds_GetSessionDuration, 1804
 - pack_wds_GetSessionState, 1805
 - pack_wds_RMSetTransferStatistics, 1805
 - pack_wds_SLQSCreateProfile, 1808
 - pack_wds_SLQSDeleteProfile, 1809
 - pack_wds_SLQSGet3GPPConfigItem, 1809
 - pack_wds_SLQSGetCurrDataSystemStat, 1810
 - pack_wds_SLQSGetCurrentChannelRate, 1810
 - pack_wds_SLQSGetDUNCallInfo, 1811
 - pack_wds_SLQSGetDataBearerTechnology, 1811
 - pack_wds_SLQSGetProfileSettings, 1811
 - pack_wds_SLQSGetRuntimeSettings, 1812
 - pack_wds_SLQSModifyProfile, 1812
 - pack_wds_SLQSResetPacketStatics, 1813
 - pack_wds_SLQSSetDHCPv4ClientConfig, 1815
 - pack_wds_SLQSSetLoopback, 1815
 - pack_wds_SLQSSetDHCPv4ClientConfig, 1816
 - pack_wds_SLQSSetLoopback, 1816
 - pack_wds_SLQSSet3GPPConfigItem, 1813
 - pack_wds_SLQSSetIPFamilyPreference, 1814
 - pack_wds_SLQSSetWdsEventCallback, 1814
 - pack_wds_SLQSStartDataSession, 1816
 - pack_wds_SLQSStopDataSession, 1817
 - pack_wds_SLQSWdsGoActive, 1817
 - pack_wds_SLQSWdsGoDormant, 1818
 - pack_wds_SLQSWdsSetEventReport, 1818
 - pack_wds_SLQSWdsSwiPDPRuntimeSettings,
1819
 - pack_wds_SetAutoconnect, 1806
 - pack_wds_SetDefaultProfile, 1806
 - pack_wds_SetDefaultProfileNum, 1806
 - pack_wds_SetMobileIPParameters, 1807
 - pack_wds_SetMobileIPProfile, 1808
 - pack_wds_SetMobileIP, 1807
 - unpack_wds_DHCPv4ClientLease_ind, 1819
 - unpack_wds_DHCPv4ClientLeaseChange, 1820
 - unpack_wds_GetAutoconnect, 1820
 - unpack_wds_GetByteTotals, 1820
 - unpack_wds_GetConnectionRate, 1821
 - unpack_wds_GetDataBearerTechnology, 1821
 - unpack_wds_GetDefaultProfile, 1822

- unpack_wds_GetDefaultProfileNum, [1822](#)
- unpack_wds_GetDormancyState, [1822](#)
- unpack_wds_GetLastMobileIPError, [1823](#)
- unpack_wds_GetMobileIPProfile, [1824](#)
- unpack_wds_GetMobileIP, [1823](#)
- unpack_wds_GetPacketStatistics, [1824](#)
- unpack_wds_GetPacketStatus, [1824](#)
- unpack_wds_GetSessionDuration, [1825](#)
- unpack_wds_GetSessionState, [1825](#)
- unpack_wds_RMSetTransferStatistics, [1826](#)
- unpack_wds_RMTransferStatistics_ind, [1826](#)
- unpack_wds_SLQSCreateProfile, [1829](#)
- unpack_wds_SLQSDeleteProfile, [1829](#)
- unpack_wds_SLQSGet3GPPConfigItem, [1829](#)
- unpack_wds_SLQSGetCurrDataSystemStat, [1830](#)
- unpack_wds_SLQSGetCurrentChannelRate, [1830](#)
- unpack_wds_SLQSGetDUNCallInfo, [1831](#)
- unpack_wds_SLQSGetDataBearerTechnology, [1831](#)
- unpack_wds_SLQSGetProfileSettings, [1831](#)
- unpack_wds_SLQSGetRuntimeSettings, [1832](#)
- unpack_wds_SLQSModifyProfile, [1832](#)
- unpack_wds_SLQSResetPacketStatics, [1832](#)
- unpack_wds_SLQSSetDHCPv4ClientConfig, [1834](#)
- unpack_wds_SLQSSetLoopback, [1835](#)
- unpack_wds_SLQSSetDHCPv4ClientConfig, [1835](#)
- unpack_wds_SLQSSetLoopback, [1836](#)
- unpack_wds_SLQSSet3GPPConfigItem, [1833](#)
- unpack_wds_SLQSSetIPFamilyPreference, [1833](#)
- unpack_wds_SLQSSetPacketSrvStatusCallback, [1833](#)
- unpack_wds_SLQSSetWdsEventCallback, [1834](#)
- unpack_wds_SLQSSetWdsEventCallback_ind, [1834](#)
- unpack_wds_SLQSStartDataSession, [1836](#)
- unpack_wds_SLQSStopDataSession, [1836](#)
- unpack_wds_SLQSWdsGoActive, [1837](#)
- unpack_wds_SLQSWdsGoDormant, [1837](#)
- unpack_wds_SLQSWdsSetEventReport, [1837](#)
- unpack_wds_SLQSWdsSwiPDPRuntimeSettings, [1838](#)
- unpack_wds_SetAutoconnect, [1826](#)
- unpack_wds_SetDefaultProfile, [1827](#)
- unpack_wds_SetDefaultProfileNum, [1827](#)
- unpack_wds_SetMobileIPParameters, [1828](#)
- unpack_wds_SetMobileIPProfile, [1828](#)
- unpack_wds_SetMobileIP, [1827](#)
- UnpackQmiProfileInfo, [1798](#)
- WDS_DHCP_MAX_NUM_OPTIONS, [1798](#)
- WDS_DHCP_OPTION_DATA_BUF_SIZE, [1798](#)
- wds_DHCPLeaseOptTlv, [1161](#)
 - numOpt, [1161](#)
 - optList, [1161](#)
 - optListData, [1161](#)
 - TlvPresent, [1161](#)
- wds_DHCPLeaseStateTlv, [1161](#)
 - leaseState, [1161](#)
 - TlvPresent, [1161](#)
- wds_DHCPOpt, [1161](#)
 - optCode, [1162](#)
 - optValLen, [1162](#)
 - pOptVal, [1162](#)
- wds_DHCPProfileIdTlv, [1162](#)
 - profileId, [1162](#)
 - profileType, [1162](#)
 - TlvPresent, [1162](#)
- wds_DHCPv4HWConfig, [1162](#)
 - chaddr, [1163](#)
 - chaddrLen, [1163](#)
 - hwType, [1163](#)
- wds_DHCPv4Option, [1163](#)
 - optCode, [1163](#)
 - optVal, [1163](#)
 - optValLen, [1163](#)
- wds_DHCPv4OptionList, [1163](#)
 - numOpt, [1164](#)
 - pOptList, [1164](#)
- wds_DHCPv4ProfileId, [1164](#)
 - profileId, [1164](#)
 - profileType, [1164](#)
- wds_DataULongLongTlv, [1160](#)
 - TlvPresent, [1160](#)
 - ulldata, [1160](#)
- wds_DataULongTlv, [1160](#)
 - TlvPresent, [1161](#)
 - ulldata, [1161](#)
- wds_Domain, [1165](#)
 - domainLen, [1165](#)
 - domainName, [1165](#)
- wds_DomainNameList, [1165](#)
 - domain, [1166](#)
 - numInstances, [1166](#)
- wds_GPRSQoS, [1166](#)
 - delayClass, [1166](#)
 - meanThroughputClass, [1166](#)
 - peakThroughputClass, [1166](#)
 - precedenceClass, [1166](#)
 - reliabilityClass, [1166](#)
- wds_IPV6AddressInfo, [1167](#)
 - IPAddressV6, [1167](#)
 - IPv6PrefixLen, [1167](#)
- wds_IPV6GWAddressInfo, [1167](#)
 - gwAddressV6, [1168](#)
 - gwV6PrefixLen, [1168](#)
- wds_IPv4AdTlv, [1167](#)
 - IPv4Addr, [1167](#)
 - TlvPresent, [1167](#)
- wds_PCSCFFQDNAddress, [1168](#)
 - fqdnAddr, [1168](#)
 - fqdnLen, [1169](#)
- wds_PCSCFFQDNAddressList, [1169](#)
 - numInstances, [1169](#)
 - pcsfQDNAddress, [1169](#)
- wds_PCSCFIPv4ServerAddressList, [1169](#)

- numInstances, 1170
- pScsflIPv4Addr, 1170
- wds_ProfileIdentifier, 1170
 - profileIndex, 1170
 - profileType, 1170
- wds_TrStatInd, 1171
 - statsMask, 1171
 - statsPeriod, 1171
- wds_UMTSMInQoS, 1171
 - deliveryErrSDU, 1173
 - grntDownlinkBitrate, 1173
 - grntUplinkBitrate, 1174
 - maxDownlinkBitrate, 1174
 - maxSDUSize, 1174
 - maxUplinkBitrate, 1174
 - qosDeliveryOrder, 1174
 - resBerRatio, 1174
 - sduErrorRatio, 1174
 - trafficClass, 1174
 - trafficPriority, 1174
 - transferDelay, 1174
- wds_currNetworkInfo, 1158
 - NetworkType, 1160
 - RATMask, 1160
 - SOMask, 1160
- wds_profileInfo, 1170
 - SlqsProfile3GPP2, 1171
 - SlqsProfile3GPP, 1171
- WdsByteTotals, 1174
 - ByteTotalsElmntsV4, 1175
 - ByteTotalsElmntsV6, 1175
 - pV4sessionId, 1175
 - pV6sessionId, 1175
- WdsByteTotalsElmnts, 1175
 - pRXTotalBytes, 1175
 - pTXTotalBytes, 1175
- WdsClientLeaseChange, 1175
 - pEnableNotification, 1176
- WdsConnectionRate, 1176
 - ConnRateElmntsV4, 1177
 - ConnRateElmntsV6, 1177
 - pV4sessionId, 1177
 - pV6sessionId, 1177
- WdsConnectionRateElmnts, 1177
 - pCurrentChannelRXRate, 1177
 - pCurrentChannelTXRate, 1177
 - pMaxChannelRXRate, 1177
 - pMaxChannelTXRate, 1177
- WdsDHCPv4ClientLeaseInd, 1178
 - piIPv4Addr, 1178
 - pLeaseState, 1178
 - pOptList, 1178
 - pProfileId, 1178
- WdsDHCPv4Config, 1179
 - pHwConfig, 1179
 - pProfileId, 1179
 - pRequestOptionList, 1179
- WdsDHCPv4HwConfig, 1179
 - chaddr, 1180
 - chaddrLen, 1180
 - hwType, 1180
- WdsDHCPv4Option, 1181
 - optCode, 1182
 - optVal, 1182
 - optValLen, 1182
- WdsDHCPv4OptionList, 1182
 - numOpt, 1182
 - pOptList, 1182
- WdsDHCPv4ProfileId, 1183
 - profileId, 1183
 - profileType, 1184
- wdsDhcpv4HwConfig, 1180
 - chaddr, 1180
 - chaddrLen, 1180
 - hwType, 1180
- wdsDhcpv4Option, 1181
 - optCode, 1181
 - optVal, 1181
 - optValLen, 1181
- wdsDhcpv4OptionList, 1183
 - numOpt, 1183
 - pOptList, 1183
- wdsDhcpv4ProfileId, 1184
 - profileId, 1184
 - profileType, 1184
- WdsIpAddressInfoReq, 1185
 - ip, 1185
 - pV4sessionId, 1185
 - pV6sessionId, 1185
- WdsPktStatisticsElmnts, 1185
 - pRXDroppedCount, 1187
 - pRXOKBytesLastCall, 1187
 - pRXOkBytesCount, 1187
 - pRXPacketErrors, 1187
 - pRXPacketOverflows, 1187
 - pRXPacketSuccesses, 1187
 - pTXDroppedCount, 1187
 - pTXOKBytesLastCall, 1187
 - pTXOkBytesCount, 1187
 - pTXPacketErrors, 1187
 - pTXPacketOverflows, 1187
 - pTXPacketSuccesses, 1187
- WdsPktStatisticsReq, 1187
 - pStatMask, 1188
- WdsPktStatisticsResp, 1188
 - pV4sessionId, 1189
 - pV6sessionId, 1189
 - PktStatElmntsV4, 1188
 - PktStatElmntsV6, 1188
- WdsProfileParam, 1189
 - SlqsProfile3GPP2, 1189
 - SlqsProfile3GPP, 1189
- WdsRunTimeSettings, 1189
 - rts, 1190
 - v4sessionId, 1190
 - v6sessionId, 1190

- wdsSetEventReportReq, [1190](#)
 - pCurrChannelRateInd, [1191](#)
 - pCurrDataBearerTechInd, [1191](#)
 - pCurrPrefDataSysInd, [1191](#)
 - pDataBearerTechInd, [1191](#)
 - pDataCallStatusChangeInd, [1191](#)
 - pDataSystemStatusChangeInd, [1191](#)
 - pDormancyStatusInd, [1191](#)
 - pEVDOPageMonPerChangeInd, [1192](#)
 - pMIPStatusInd, [1192](#)
 - pTransferStatInd, [1192](#)
- Wireless Data Service (WDS), [34](#)
- xAxis
 - sensorData, [744](#)
 - sensorData_t, [746](#)
- xferStatAvail
 - unpack_wds_SLQSSetWdsEventCallback_ind_t, [1082](#)
- xid
 - pack_loc_SLQSLOCGetBestAvailPos_t, [572](#)
 - pack_qmi_t, [601](#)
 - unpack_qmi_t, [1018](#)
- xtra_start_gps_minutes
 - GPSSStateInfo, [277](#)
- xtra_start_gps_week
 - GPSSStateInfo, [277](#)
- xtra_valid_duration_hours
 - GPSSStateInfo, [277](#)
- yAxis
 - sensorData, [745](#)
 - sensorData_t, [746](#)
- year
 - nas_UniversalTime, [487](#)
 - nas_timeInfo, [482](#)
 - timeInfo, [863](#)
 - UniversalTime, [925](#)
- zAxis
 - sensorData, [745](#)
 - sensorData_t, [746](#)