

Continuous Delivery Pipeline - Patterns and Anti-patterns

Juni Mukherjee (@JuniTweets)



Jenkins World

2016

#JenkinsWorld

Domain-driven Design (DDD) for Continuous Delivery



Jenkins World
2016

Thank you Eric Evans for DDD!

Connect the implementation to an evolving domain model

Entity : Defined by ID, not attributes

Value Object : Defined by attributes, no ID

Aggregate : Collection of objects

Domain : Sphere of activity

Model : System of abstractions

Domain Event : Domain experts care for

Ubiquitous Language : Shared language

Domain Services : Shared functionality

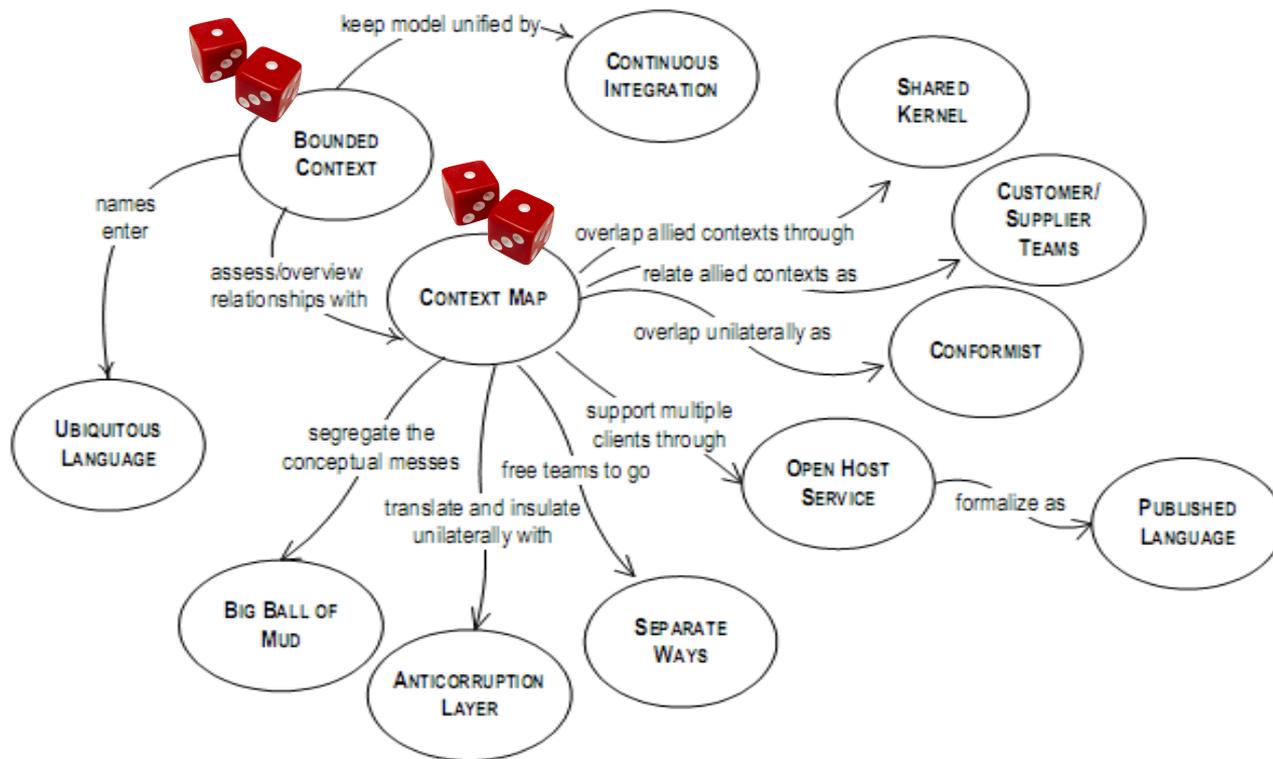
#JenkinsWorld

Domain Model Integrity - Bounded Context, Context Map



Jenkins World
2016

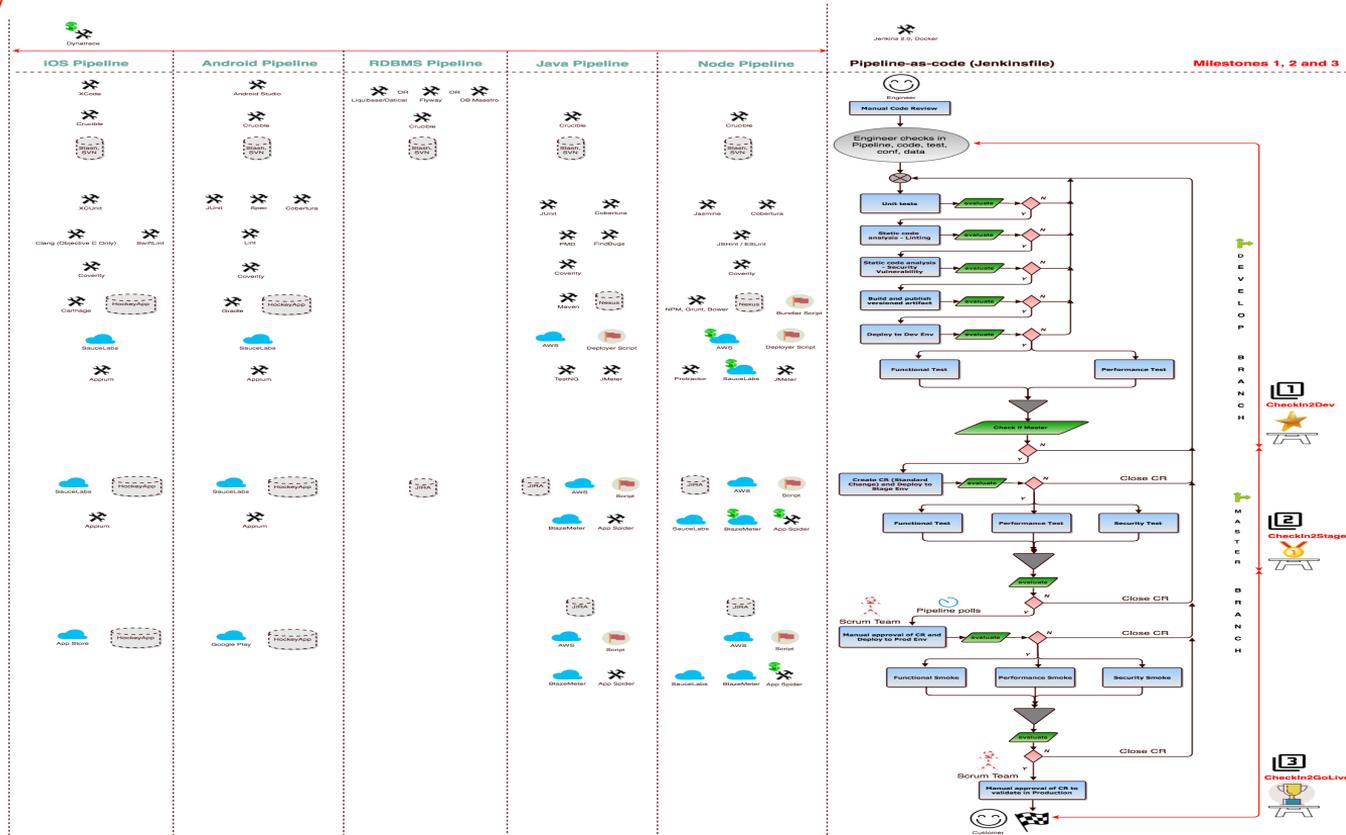
Maintaining Model Integrity



Fragmentation, Blurred Edges, Bounded Context, Context Map



Jenkins World
2016



Node,
Java,
iOS,
Android,
Database,

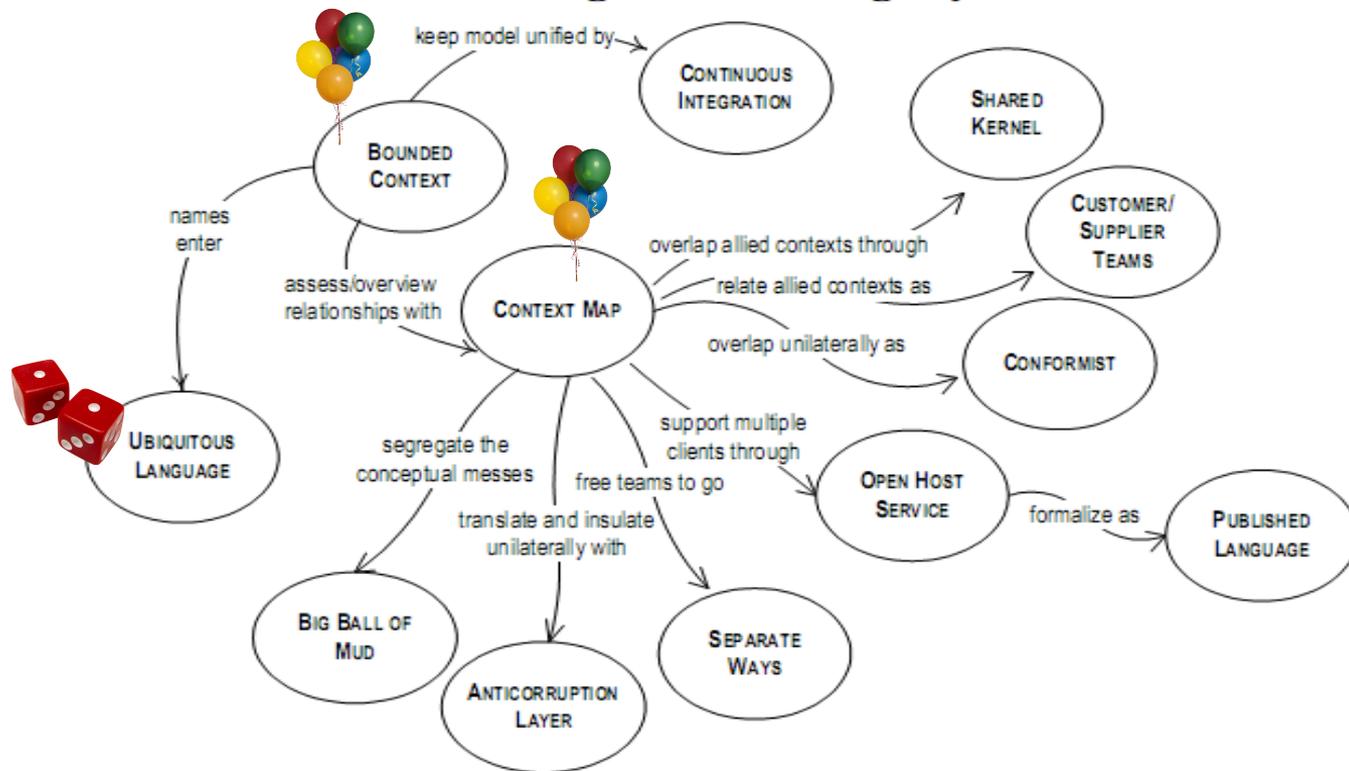
#JenkinsWorld

Domain Model Integrity - Ubiquitous Language



Jenkins World
2016

Maintaining Model Integrity



#JenkinsWorld

Ubiquitous Language



Jenkins World
2016

Component

The smallest distributable and testable unit

Unit, SCA – Linter, SCA – Security Vulnerabilities

Modules, Libraries

Subsystem

The smallest deployable and runnable unit

Functional (Security, I18N, L10N, ...), Performance

JAR | IPA | APK | Emulators
| Simulators ...

System

Loosely coupled subsystems

Integration (Interfaces, Network), Performance

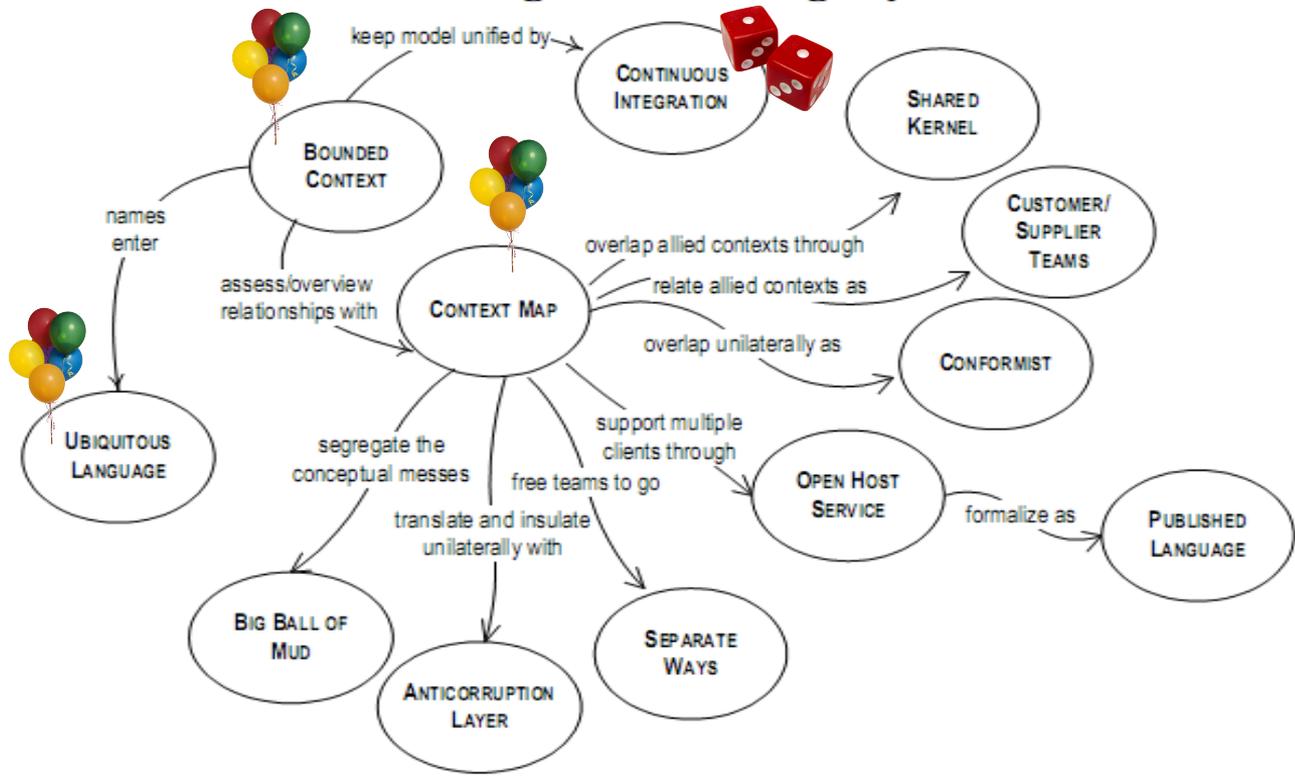
JAR & IPA & APK & Real Devices & Accessories ...

Domain Model Integrity - Continuous Integration



Jenkins World
2016

Maintaining Model Integrity



#JenkinsWorld

Continuous Integration, Continuous Delivery, Value Stream Map



Jenkins World
2016

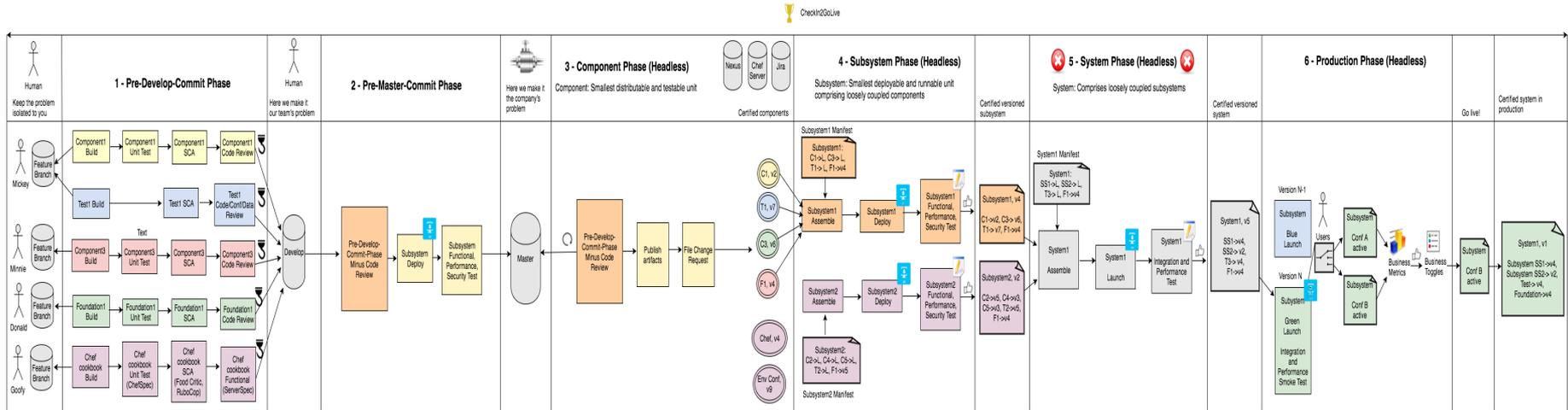
Product : System Under Test (SUT)

Configuration : Application, Environment

Test : Unit, Static, Funct, Int, Perf

Infrastructure : Infrastructure-As-Code

Pipeline-As-Code : Dockerized Jenkins 2.0, Jenkinsfile (Process-As-Code), Shared Kernel (Utils.groovy, Libs)



#JenkinsWorld

MVP - Continuous Delivery Pipeline, Slice over Dice



Jenkins World
2016

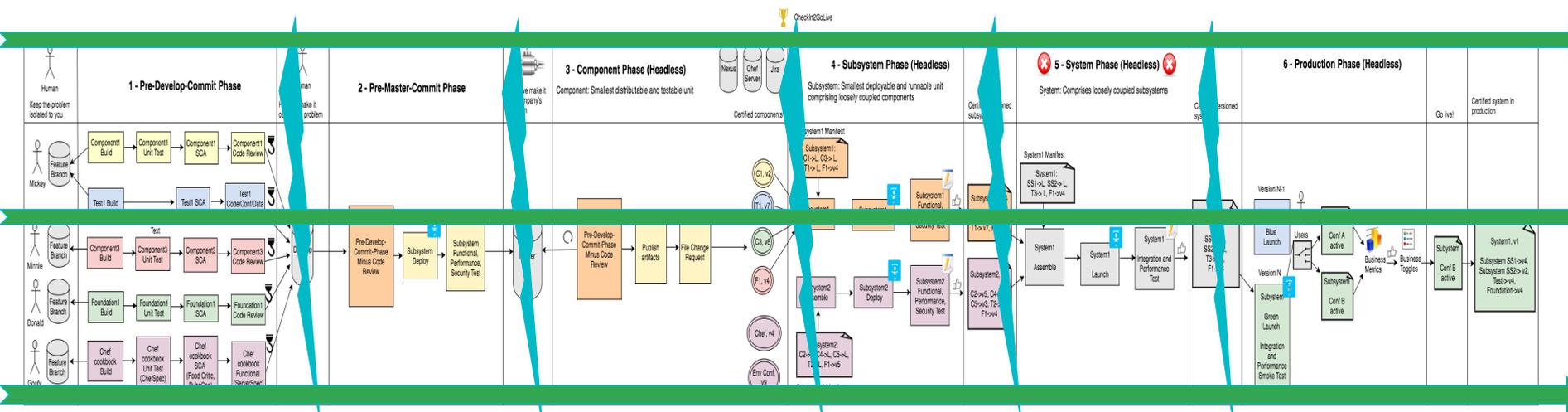
Product Slice : System Under Test (SUT)

Configuration Slice : Application, Environment

Test Slice : Unit, Static, Funct, Int, Perf

Infrastructure Slice : Infrastructure-As-Code

Pipeline-As-Code : Dockerized Jenkins 2.0, Jenkinsfile (Process-As-Code), Shared Kernel (Utils.groovy, Libs)

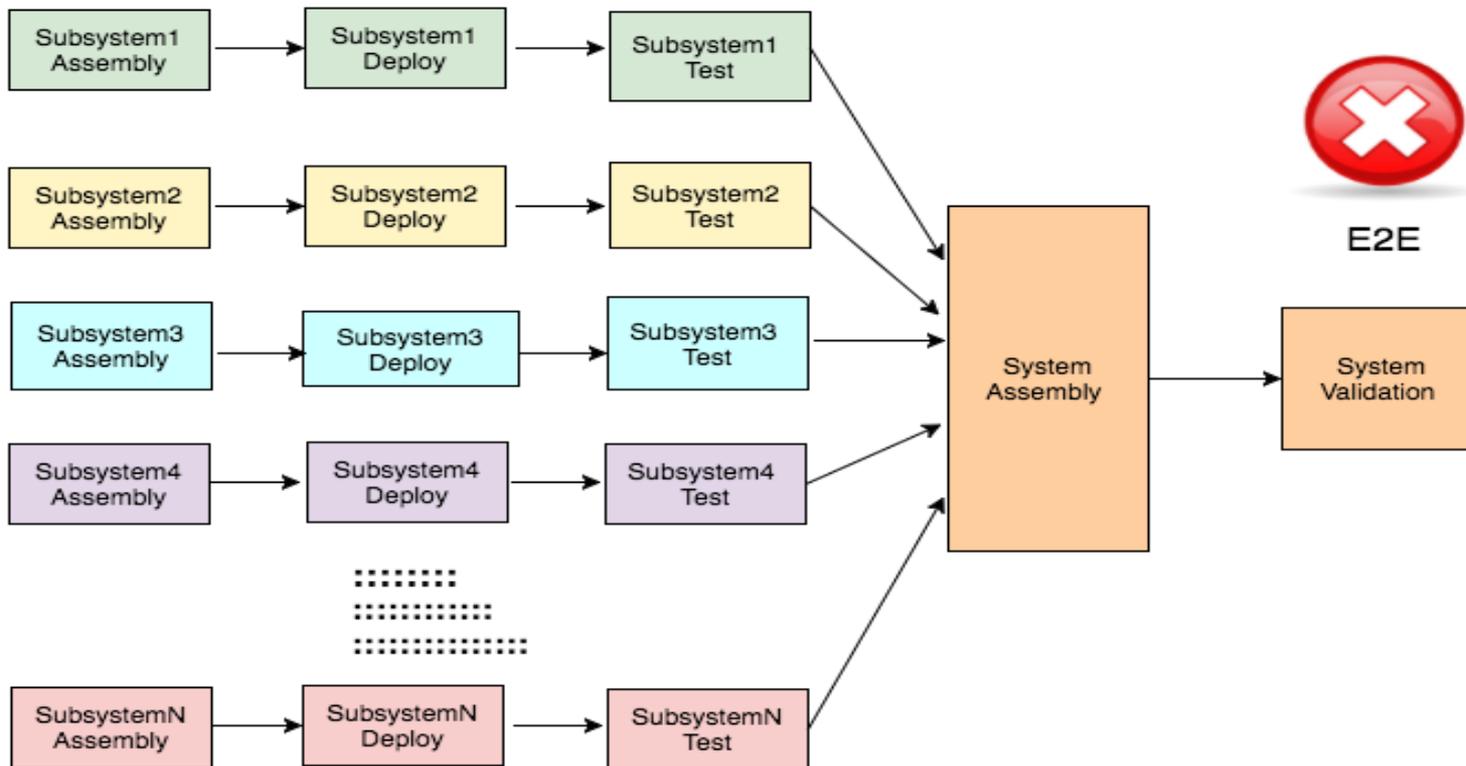


#JenkinsWorld

Pipeline - Graph - Tightly Coupled Product Architecture



Jenkins World
2016



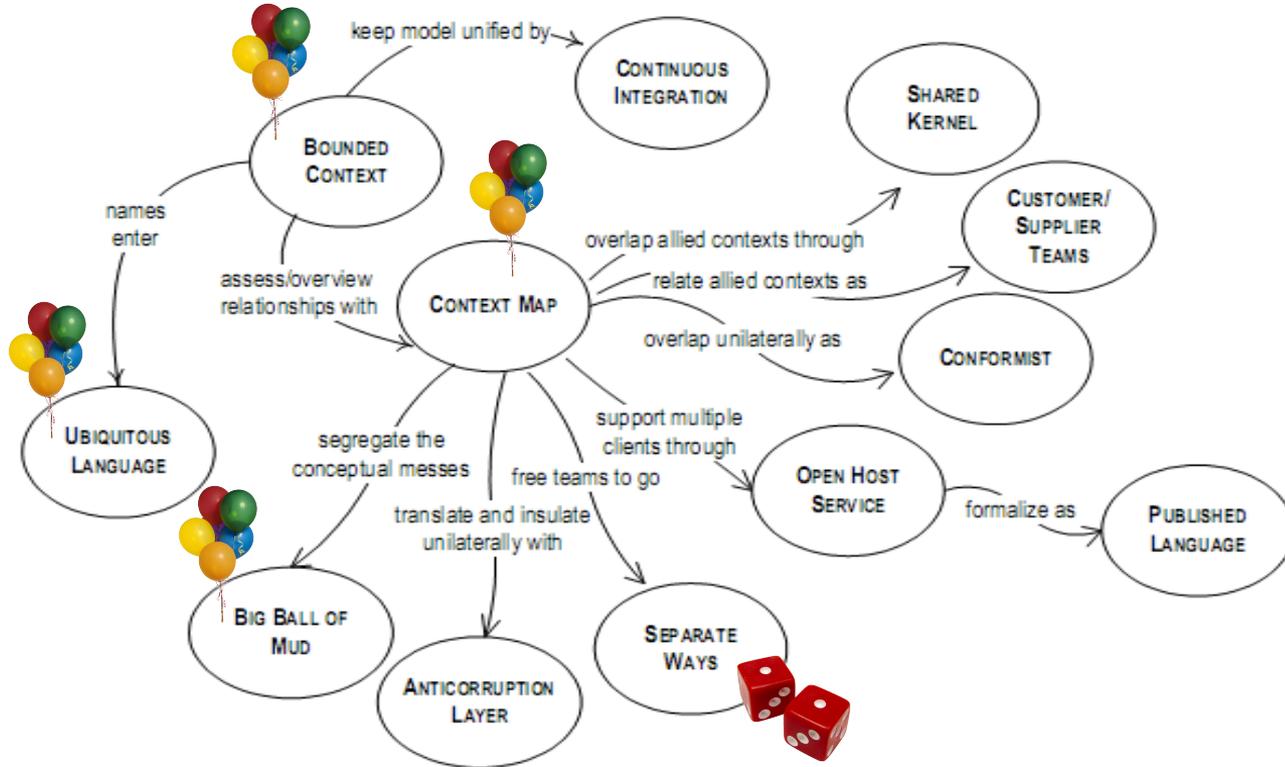
#JenkinsWorld

Domain Model Integrity - Separate Ways



Jenkins World
2016

Maintaining Model Integrity

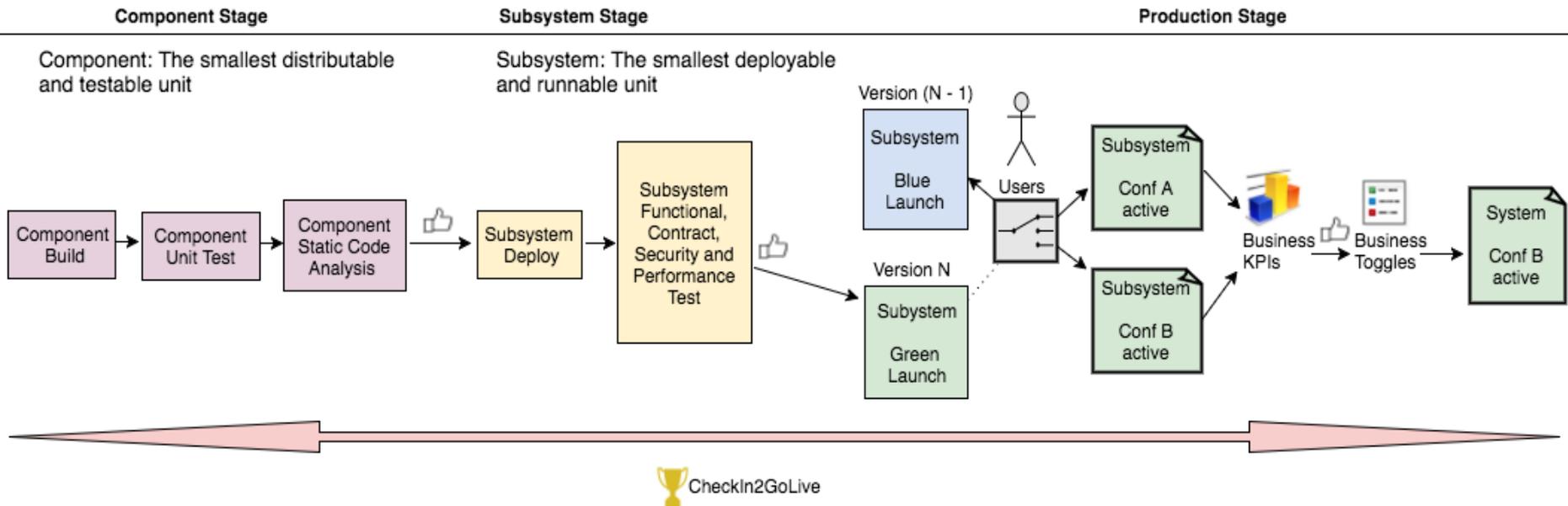


#JenkinsWorld

Pipeline - Linear - Loosely Coupled Product Architecture

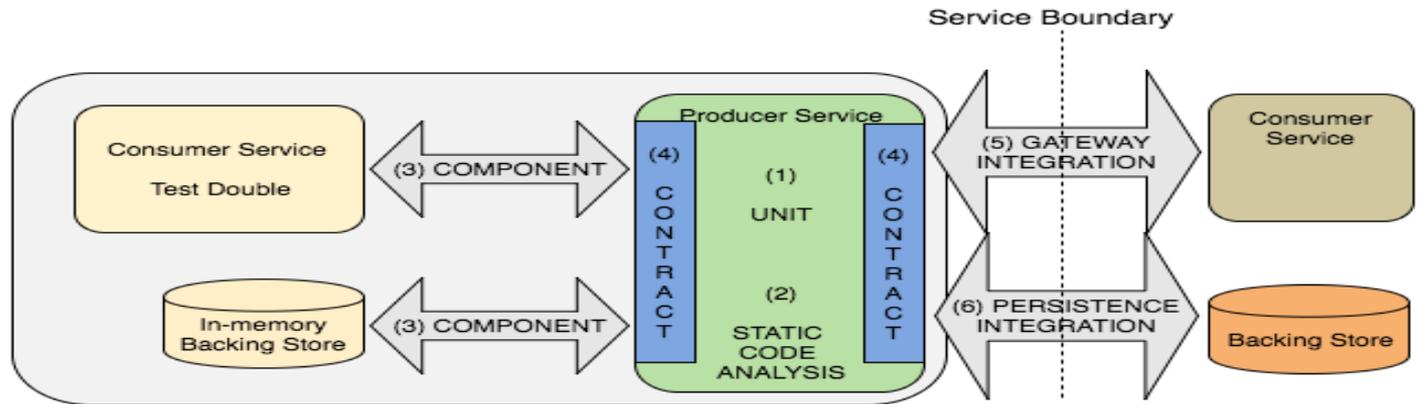


Jenkins World
2016



#JenkinsWorld

Test Architecture - Services



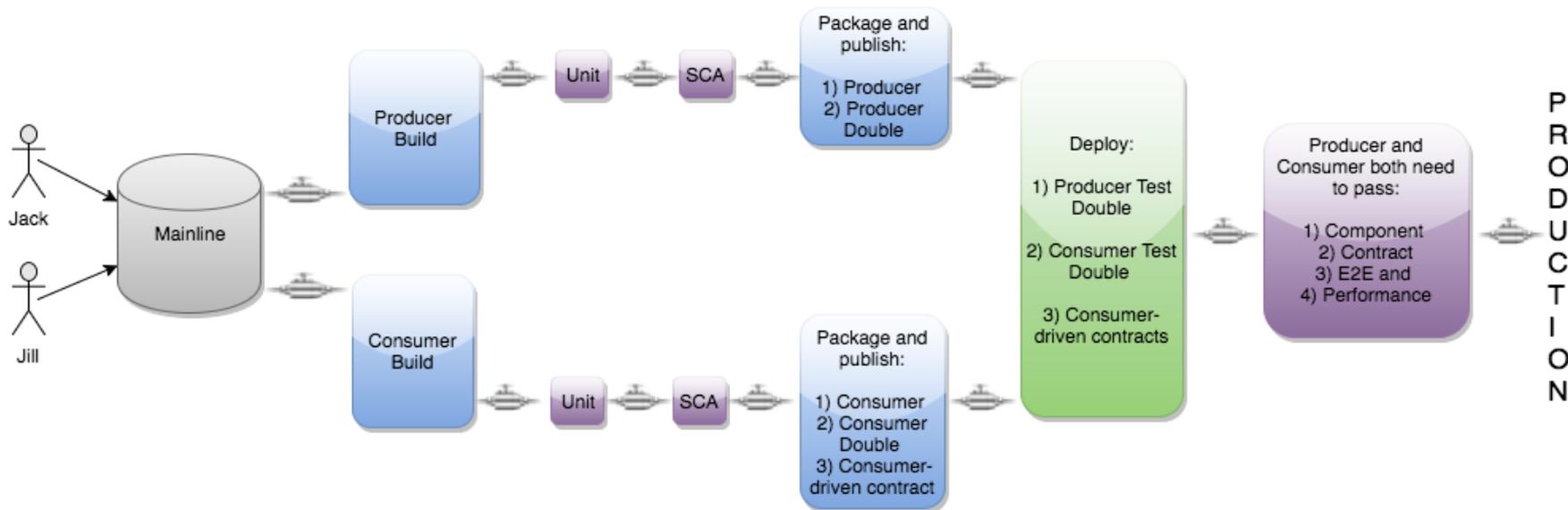
- | | | |
|---|-------------------------|---|
| 1 | Unit | Certifies the smallest piece of testable software |
| 2 | Static Code Analysis | Highlights possible vulnerabilities within source code without executing it |
| 3 | Component | Uses test doubles to isolate the code under test |
| 4 | Contract Integration | Verifies that contracts written by consumers are honored by producers |
| 5 | Gateway Integration | Verifies integration between neighboring services and components |
| 6 | Persistence Integration | Verifies integration between services and backing stores |
| 7 | E2E | Verifies user personas and journeys via Synthetic Transactions |

Customer Supplier - Pipeline Design Pattern



Consumer-driven contracts

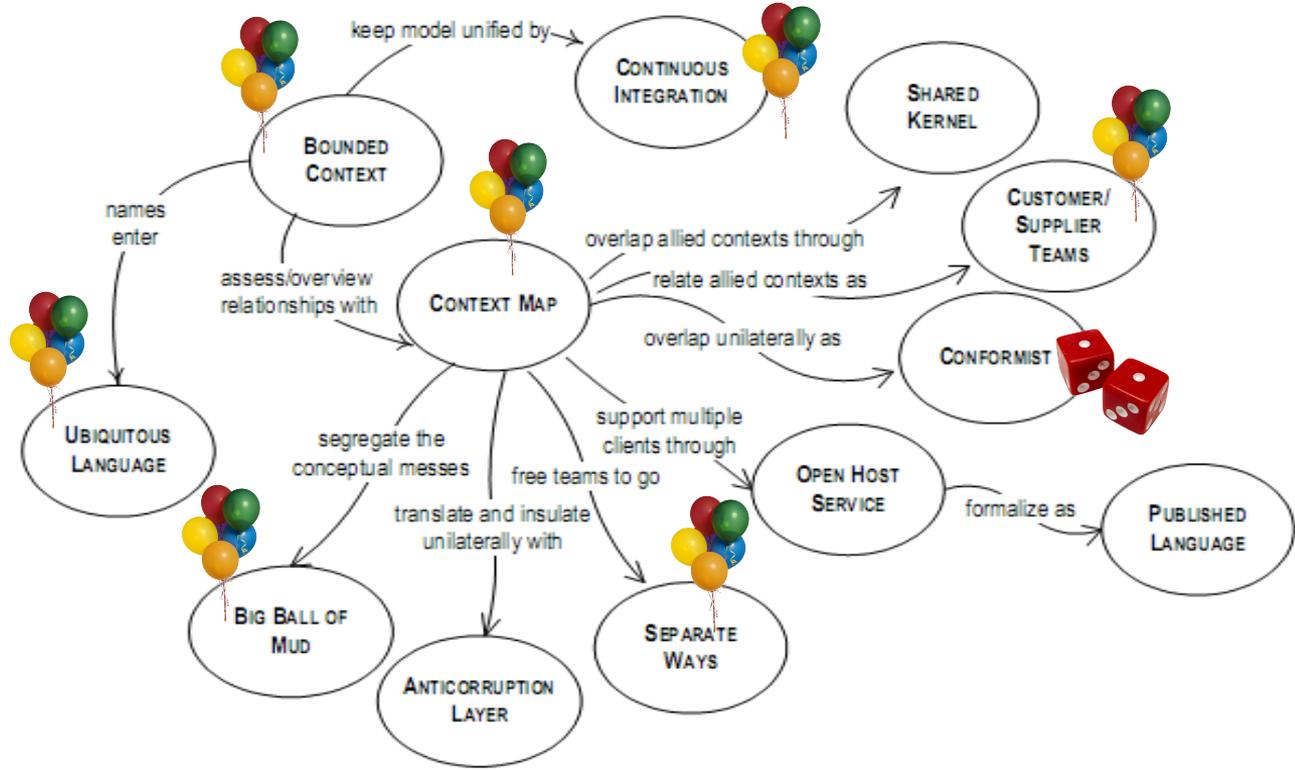
Intranet Model - Producer is aware of all Consumers



Domain Model Integrity - Conformist



Maintaining Model Integrity

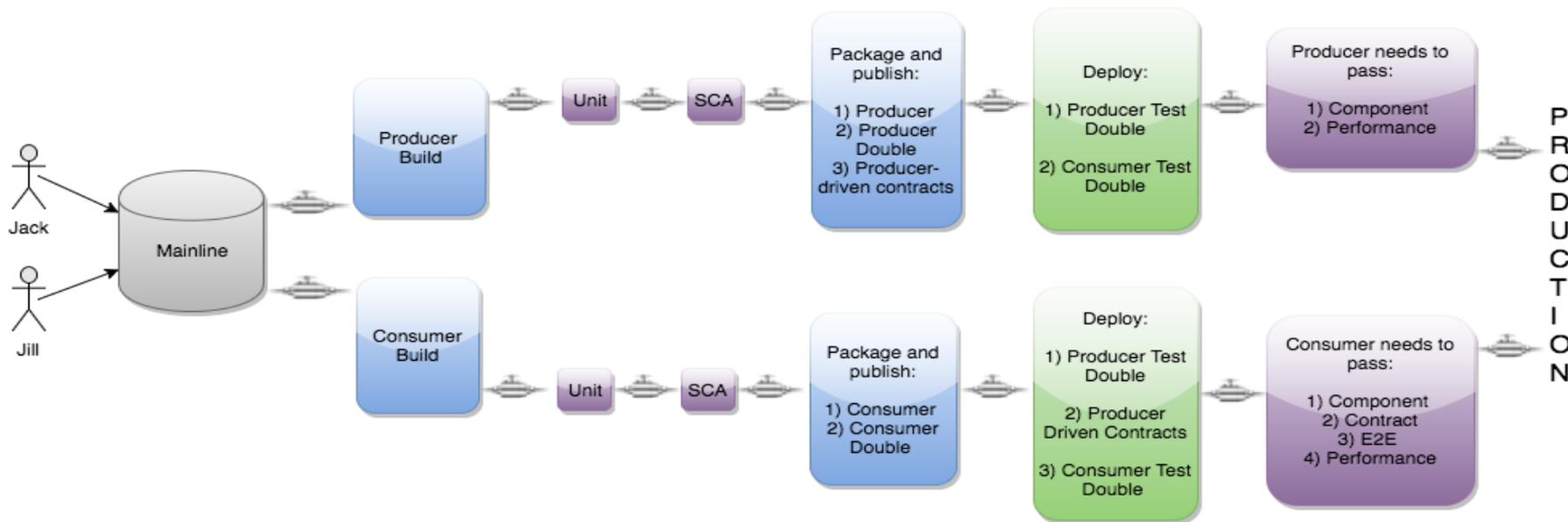


Conformist - Pipeline Design Pattern



Producer-driven contracts

Internet Model - Producer is not aware of all Consumers

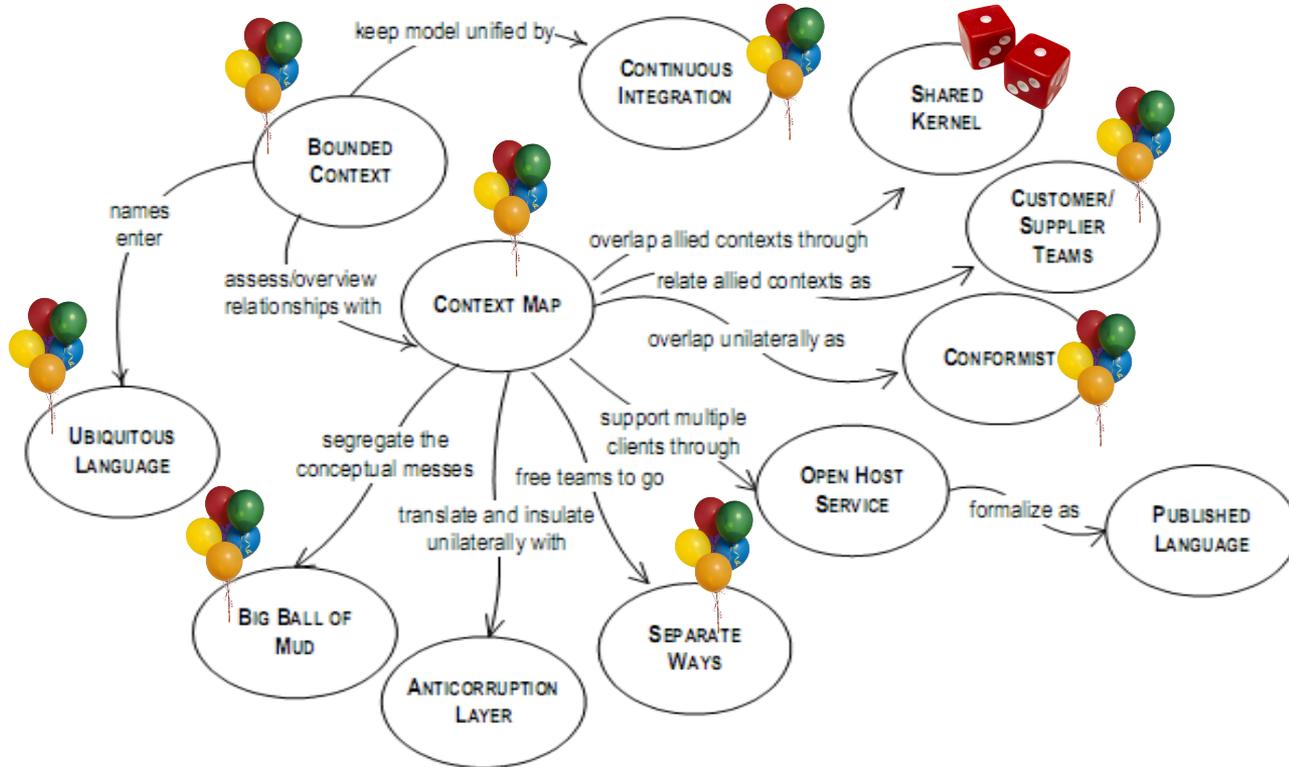


Domain Model Integrity - Shared Kernel



Jenkins World
2016

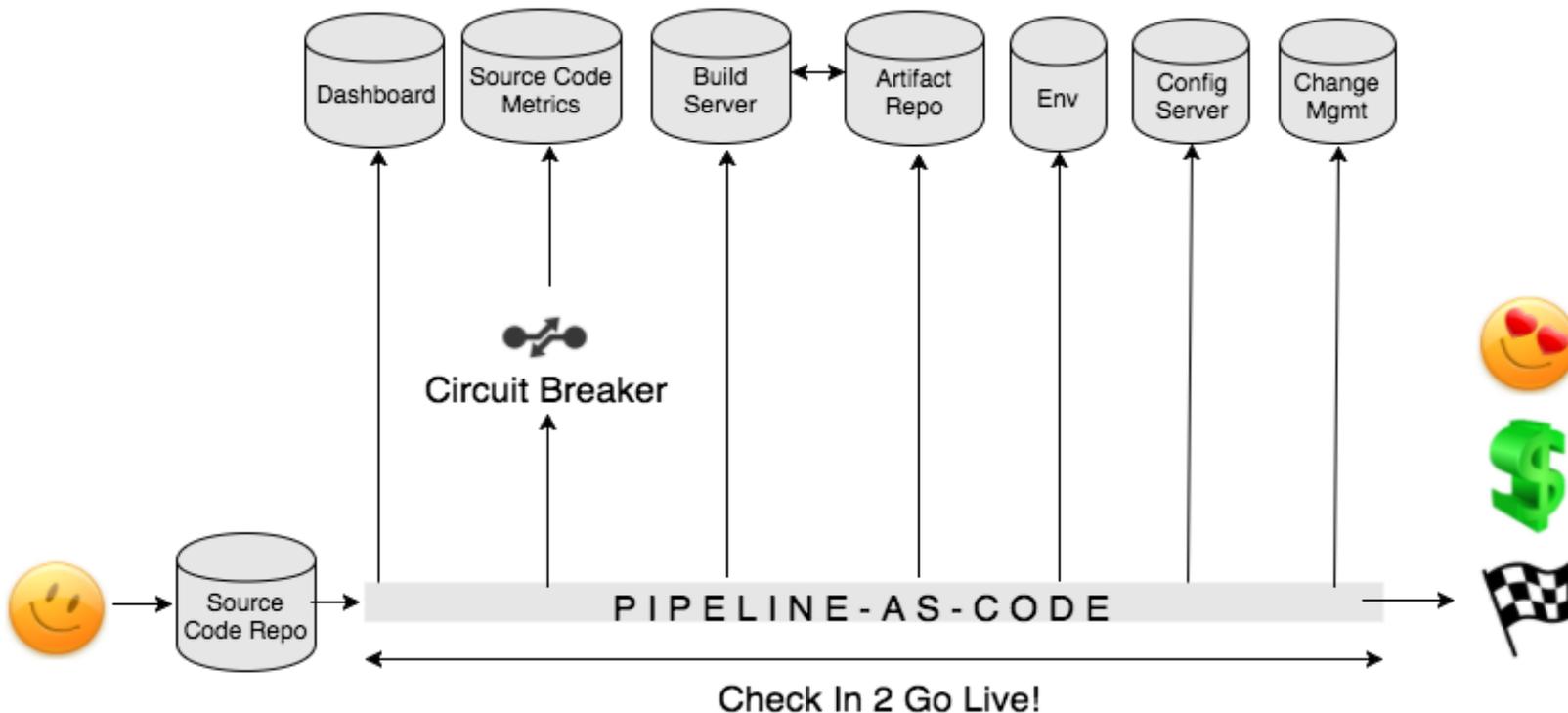
Maintaining Model Integrity



Shared Kernel, Network Latency, Circuit-breaker Pattern



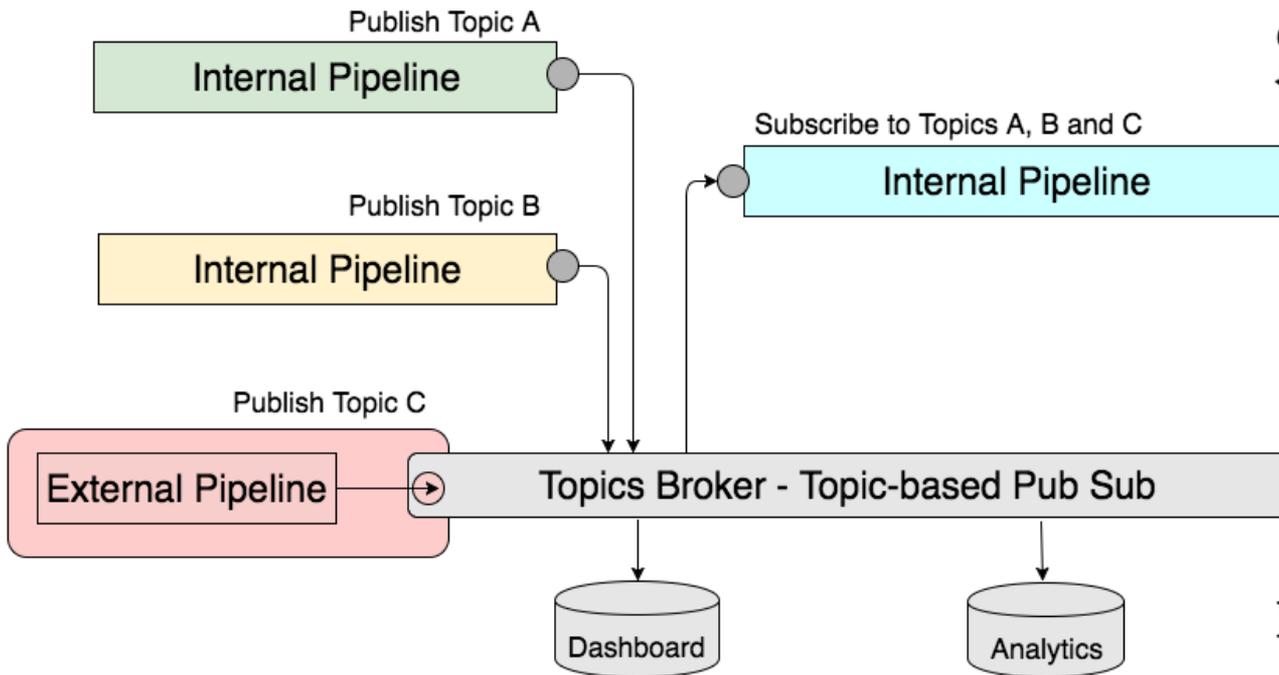
Jenkins World
2016



Inter-system communication - Publish Subscribe Pattern



Jenkins World
2016



Topic Data Model

```
class topic
{
    // auto-sequence
    private int topic_id;

    // auditability
    private int pipeline_id;
    private String env_id;

    // temporal info
    private Date time;

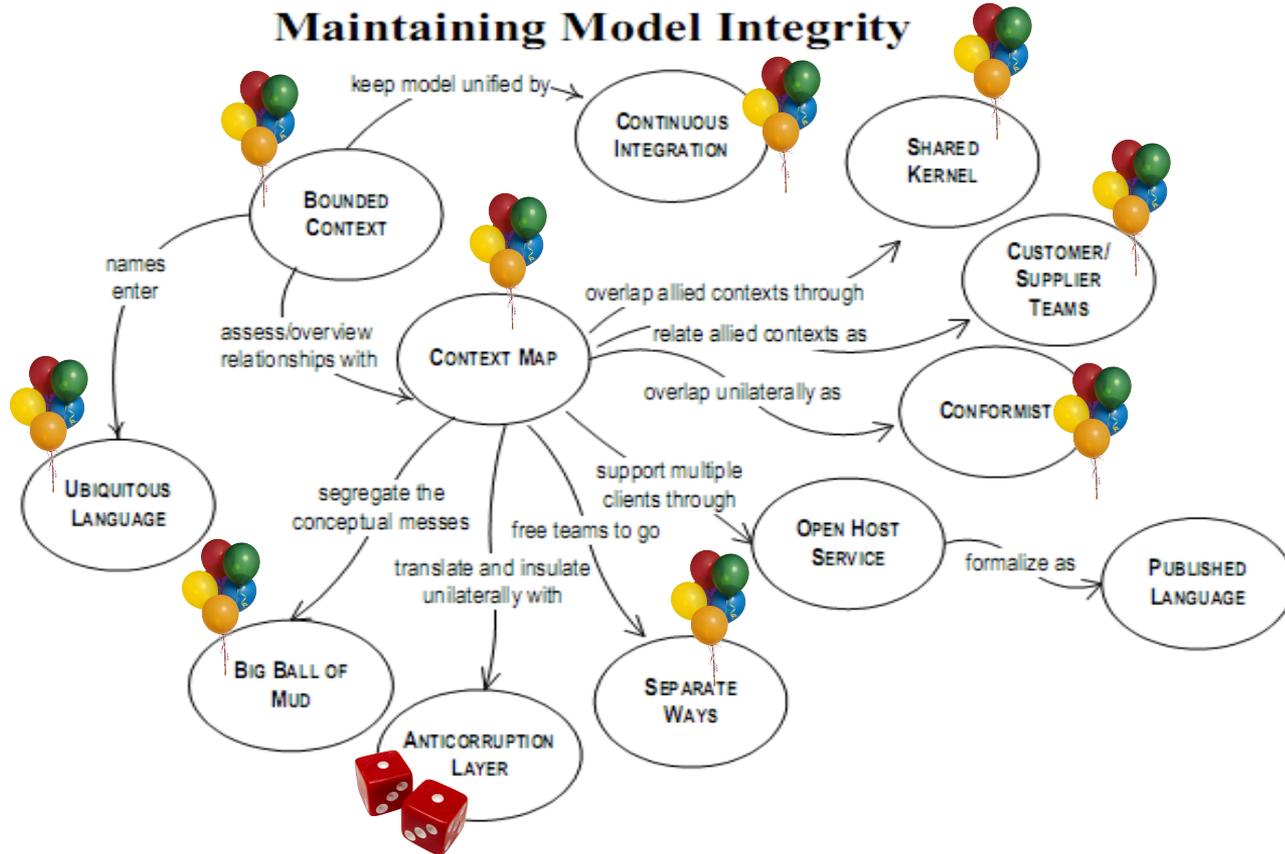
    .....
}
```

#JenkinsWorld

Domain Model Integrity - Anti-corruption Layer



Jenkins World
2016

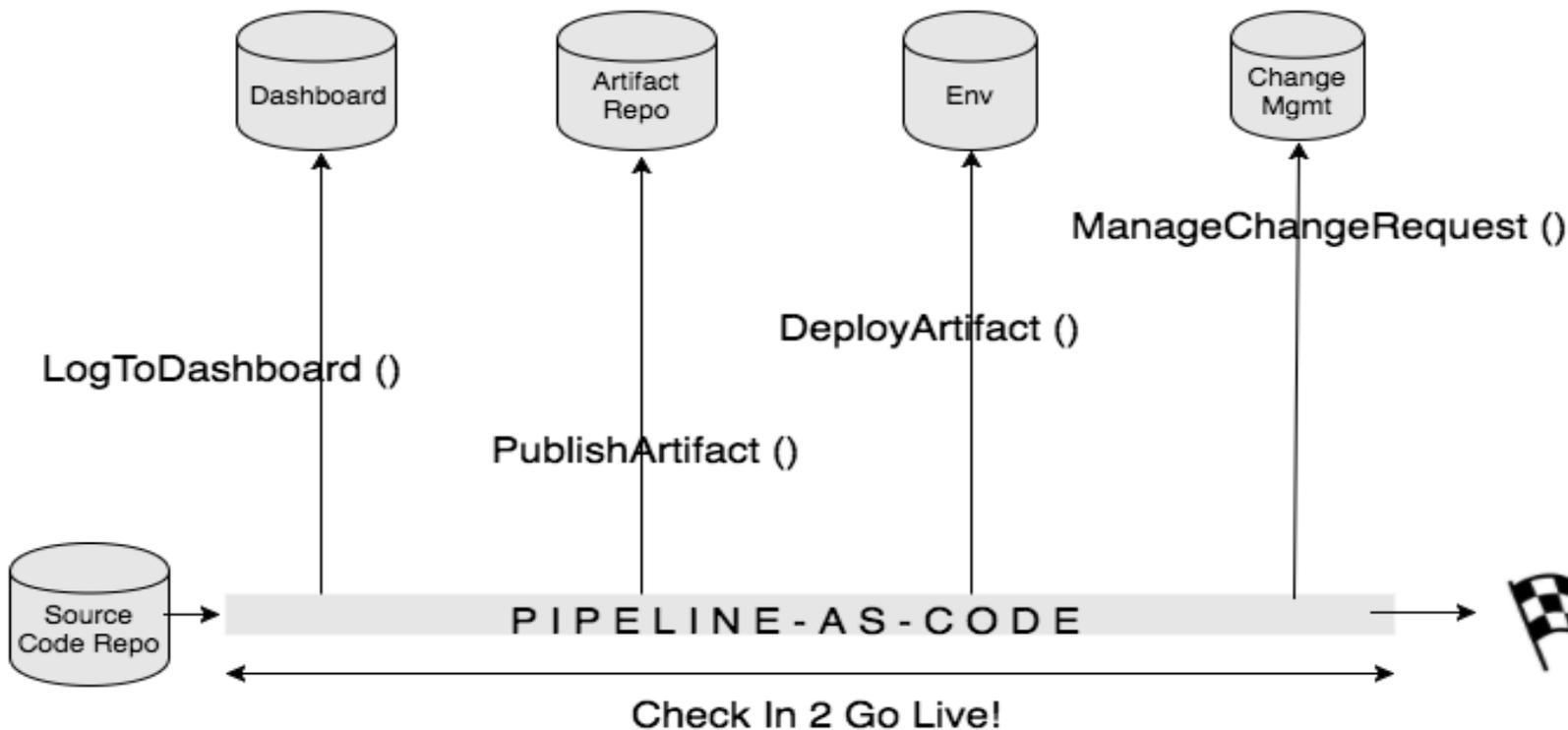


#JenkinsWorld

Anti-corruption Layer - Domain Services, Sharp Interfaces



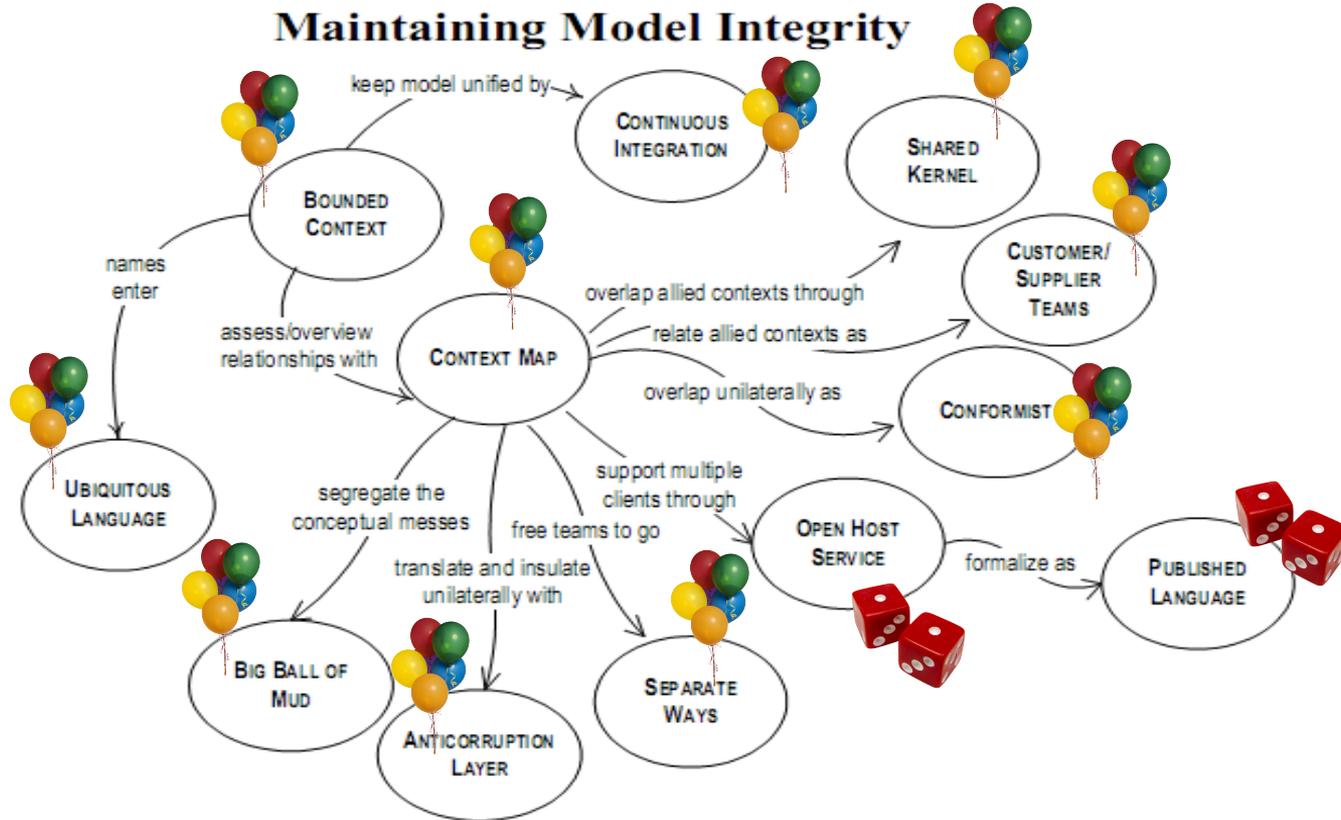
Jenkins World
2016



Domain Model Integrity - Published Language



Jenkins World
2016



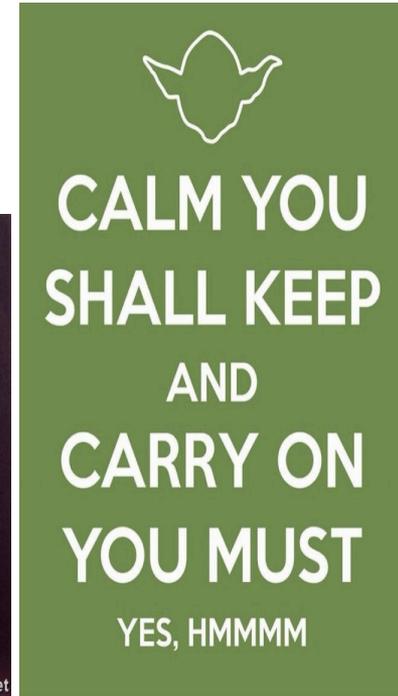
Evolution / Revolution - May the Butler be with you!



Jenkins World
2016

*Unmanageable freestyle jobs you've become,
Pipeline you must be,
Declarative, versioned and ephemeral become you must,
Or the dark side won't set you free.*

*Pipeline analytics report you must,
In the groove you must get,
Make happy our engineers you must,
Don't make them fret.*



FUNSubstance.com For your entertainment.

#JenkinsWorld

Published Language - DSL, It's groovy!, 1-800-Plumber



Jenkins World
2016

```
node ('label') {}
```

```
stage 'deploy'
```

```
    timeout (time:2, unit:'DAYS')
```

```
    {input message: 'Proceed?', submitter: 'Operations'}
```

```
stage 'functional, performance, security test'
```

```
    parallel ( parallel_task_stage_sauce : {
```

```
    }, parallel_task_stage_blazemeter : {
```

```
    }, parallel_task_stage_appspider : { })
```

```
try {}
```

```
catch (InterruptedException x) { slackSend color: '#008000', message: "" }
```

#JenkinsWorld

Shared Kernel - Domain Services, @Library, @Grab



Jenkins World
2016

```
bash-3.2$ cat services/SumoUtils.groovy

package services

class SumoUtils
{
    def sumoLogger(String endpoint, String msg)
    {
        // log messages
    }
}
```

```
bash-3.2$ cat services/
SauceUtils.groovy

package services

class SauceUtils
{
    def sauceConnect()
    {
        // secure tunnel
    }
}
```

Jenkinsfile - Class-y versus Script-y



Jenkins World
2016

```
import services.SumoUtils
import services.SauceUtils
import groovy.transform.BaseScript
@BaseScript Credentials credentials

// Logging to SumoLogic Dashboard
def sumoUtils = new services.SumoUtils ()
sumoUtils.sumoLogger ("dummy_endpoint", "The Butler is now serving Pipelines!")

// Establish connection to SauceLabs
def sauceUtils = new services.SauceUtils ()
sauceUtils.sauceConnect ()

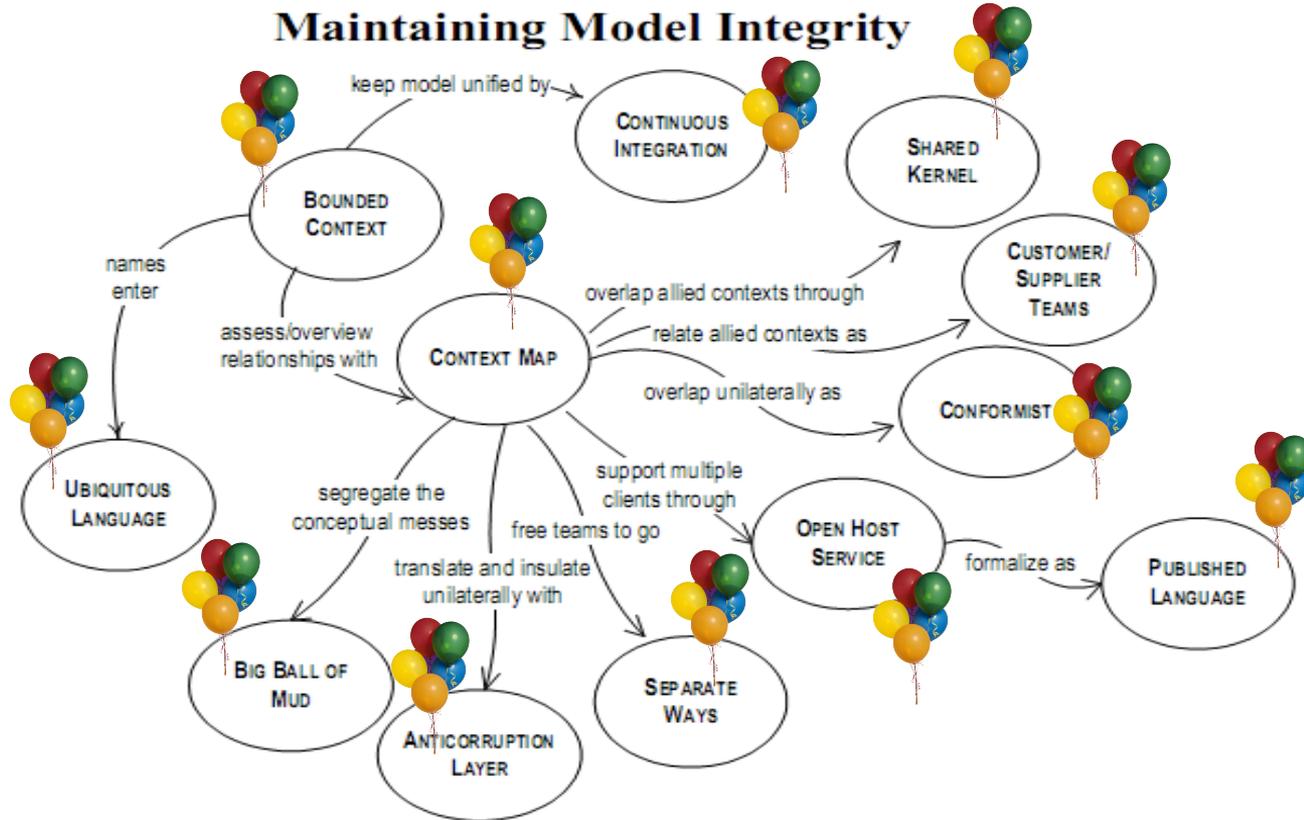
// Read credentials without being able to edit them
println "The value of CRED_ID_JENKINS_USER is " + CRED_ID_JENKINS_USER
```

#JenkinsWorld

Domain Model Integrity - Flying colors!



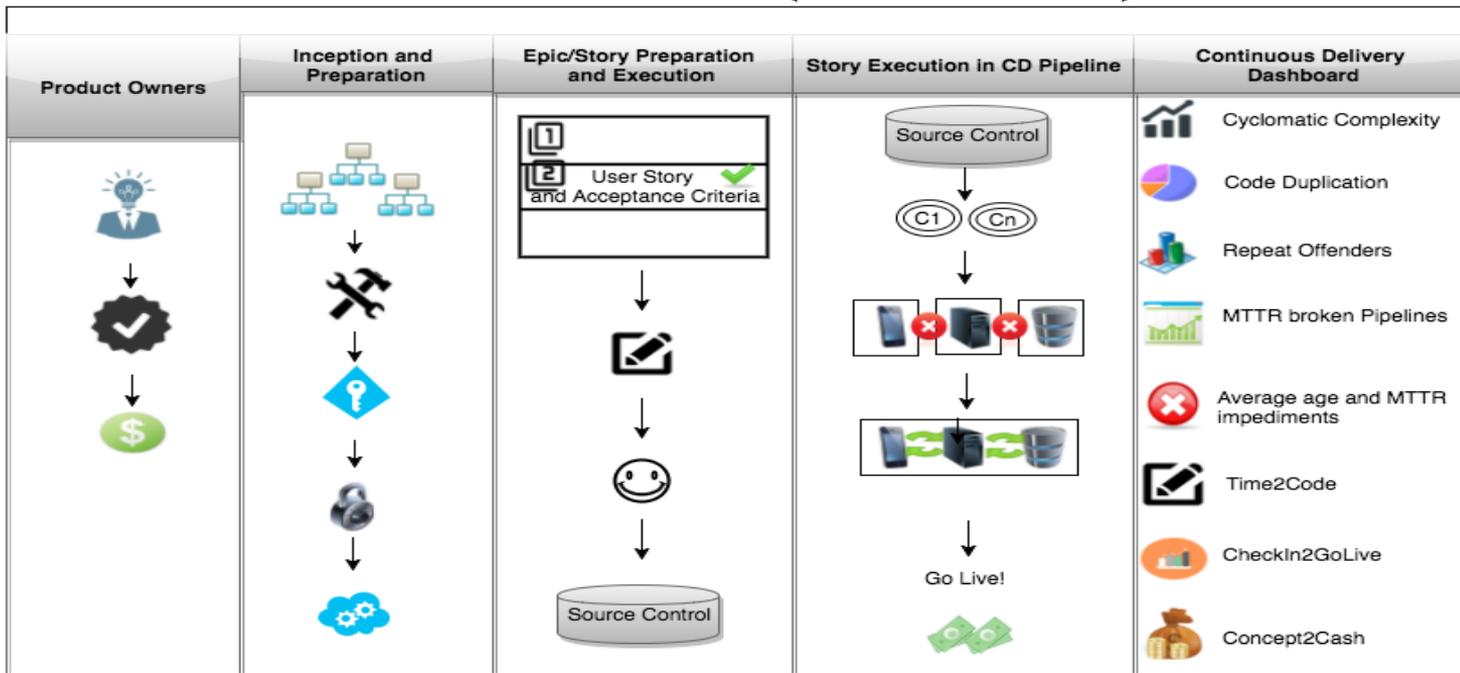
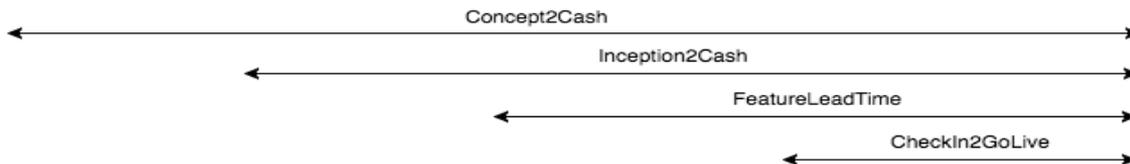
Jenkins World
2016



Concept2Cash = Fn (CheckIn2GoLive), VSM



Jenkins World
2016



KPIs - What can be measured, can be fixed!



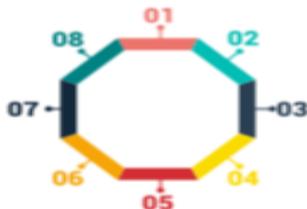
Jenkins World
2016



CheckIn2Dev
CheckIn2Stage
CheckIn2GoLive!



Pipeline deploys on Dev
Pipeline deploys on Stage
Pipeline deploys in Production



Pipeline failures due to Unit Tests
Pipeline failures due to Functional Tests
Pipeline failures due to Performance Tests
Pipeline failures due to Security Vulnerabilities

.....
.....



Code Quality = Fn (Cyclomatic Complexity,
Code Duplication,
Unit Test Coverage)

Avoid death
by a thousand
test types.

And by a
thousand
environments.

KPIs - Analytics, RoI, Show Me The Money!



Jenkins World
2016



Business Value Per Sprint

Not # of releases
Not # of tests executed



Pipeline failures in Dev
Pipeline failures in Stage
Pipeline failures in Production
of Escaped Defects



Stability Index=
Fn (CheckIn2GoLive,
of Escaped Defects,
Customer Delight,
...)

“Let’s stop the line
and build a
Pipeline!” said no
Product Owner
ever.

#JenkinsWorld

Firmware, Embedded Systems - RoC, Supply Chain, VSM, Cost



Jenkins World
2016

Component

The smallest distributable and testable unit

Modules, Libraries

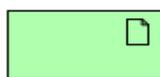


~50X

Subsystem

The smallest deployable and runnable unit

JAR | IPA | APK | 3D |
Emulators | Simulators ...



X

System

Loosely coupled subsystems

JAR & IPA & APK &
Device Farm



Internet of Things - Continuous Delivery of Eggs (Seriously?)



Jenkins World
2016

Component

The smallest distributable and testable unit

Modules, Libraries



Subsystem

The smallest deployable and runnable unit

Egg & Egg Tray & Simulator



System

Loosely coupled subsystems

Egg & Egg Tray & Drone



#JenkinsWorld

These are a few of my favorite things!



Jenkins World
2016



#JenkinsWorld

IoT - Cars, OTA, Industrial Revolution, Self-heal, Immunity ++



Jenkins World
2016

Component

The smallest distributable and testable unit

Hardware, software



Subsystem

The smallest deployable and runnable unit

Emulator/Simulator

System

Loosely coupled subsystems

Parts, Car



Pipeline is not a pipe dream!



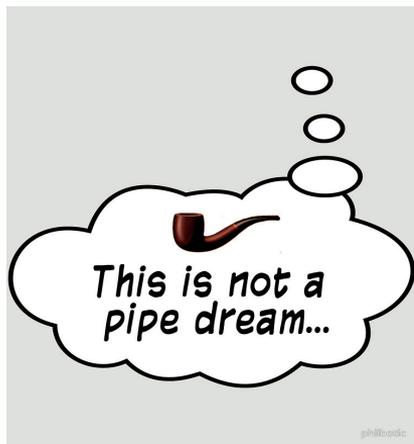
Jenkins World
2016

Software is eating the world. Hotel, Taxi, Retail,

Corollary: Software **has eaten** the world, **and its inhabitants**. Enter Users 2.0.

Continuous Integration? -> Continuous Delivery? -> Continuous Deployment?

The last mile is the longest!



Continuous Improvement is the key!

#JenkinsWorld

Closing - Conway's Law, Maya Angelou's Human Family



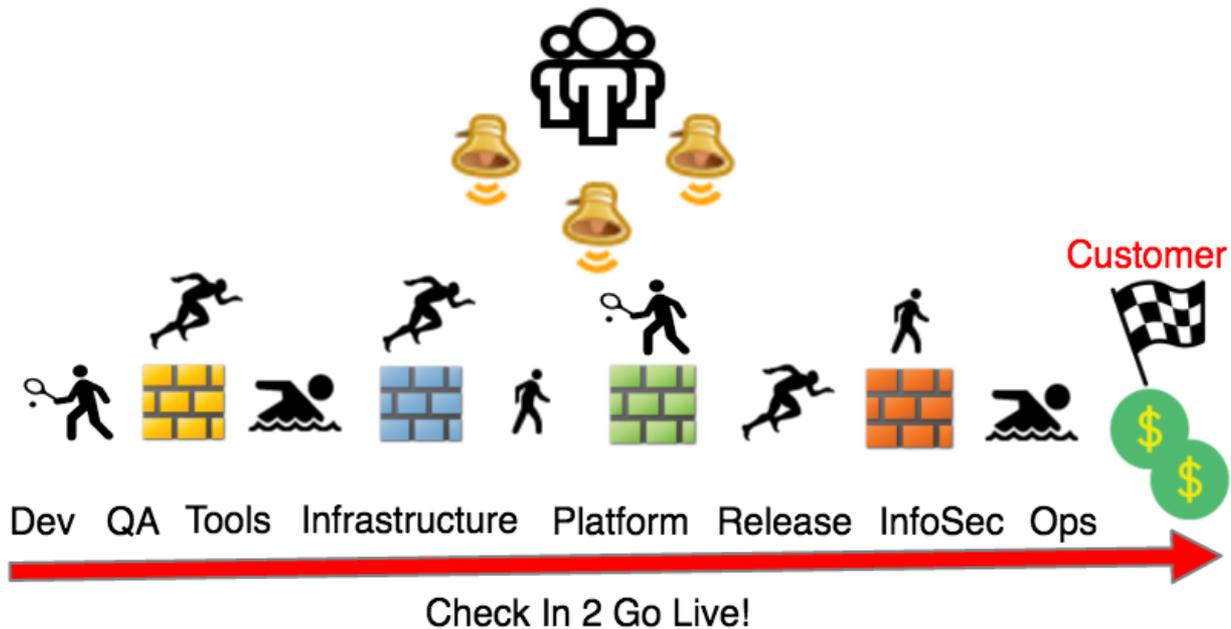
Jenkins World
2016

“Organizations which design systems ... are constrained to produce designs which are copies of the communication structures of these organizations”

Melvin Conway

“I note the obvious differences between each sort and type, We are more alike, my friends, than we are unlike.”

Maya Angelou





Jenkins World

2016

#JenkinsWorld