Tekton

Kubernetes Native CI/CD Building Blocks

Alan Ranciato, Cloud Architect @ Google

DEVOPS WORLD by CloudBees

Introduction

Alan Ranciato
Cloud Architect, Customer Engineering @ Google

ranciato@google.com
https://www.linkedin.com/in/alanranciato/

Agenda

- What is Tekton?
- Basic Concepts
- Building It
- Getting Started

What is Tekton?

TEKTON

Tekton Project

- OSS CI/CD primitives for Kubernetes
- CRDs for Task and Pipeline
  - Supersedes original Knative Build primitive
- Broad community support

Continuous Delivery Foundation

Spinnaker
NETFLIX
JFrog
IBM
circleci
cloudbees
salesforce

source: cd.foundation/members

Tekton – making CI/CD vendor-agnostic

- There's been little standardization in the field of CI – and lots of lock-in
- Enter **Tekton**
  - Kubernetes-based open source API specification of CI/CD
  - Governed by the Continuous Delivery Foundation (CDF)
  - Grew out of Knative's Build component
- Managed CI/CD without vendor lock-in
- Contributors from Google, Cloudbees, Pivotal, Red Hat, IBM, etc.

Products Built on Tekton

…. a partial list

- Jenkins X
- Redhat Openshift pipelines
- IBM Kabanero
- Puppet Relay
- Racher Rio
- Triggermesh FaaS
- Scaleway FaaS

Tekton ❤ Kubernetes

- Kubernetes is the key to run-anywhere
- K8s provides
  - Container orchestration
  - Persistent Storage
  - CLI / API / client libraries
  - Ingress + Load Balancers for webhook triggering
  - … constant improvements
  - … huge community providing new features
- Tekton adds
  - Custom Resource Definitions (CRDs) for Tekton API objects
  - Controllers to watch and update those resources
  - CLI and API

> “Tekton Pipelines lets us power Jenkins X's execution and management of pipelines natively within Kubernetes. Without Tekton, we would have been forced to write such a system ourselves, thereby costing us a lot of time and effort. Further, Tekton is benefiting from a strong development community support for solving cloud-native CI/CD challenges.”

Andrew Bayer, Software Engineer, CloudBees. Creator and maintainer of Declarative Pipelines in Jenkins

JENKINS X

The Basics

Tekton Pipelines

02

Tekton Pipelines

Tasks & Pipelines

- Parameterized, reusable templates
- Stateless

TaskRuns & PipelineRuns

- Instantiated, running executions of the templates
- Stateful

Triggers

- EventListeners
- TriggerBindings
- TriggerTemplates

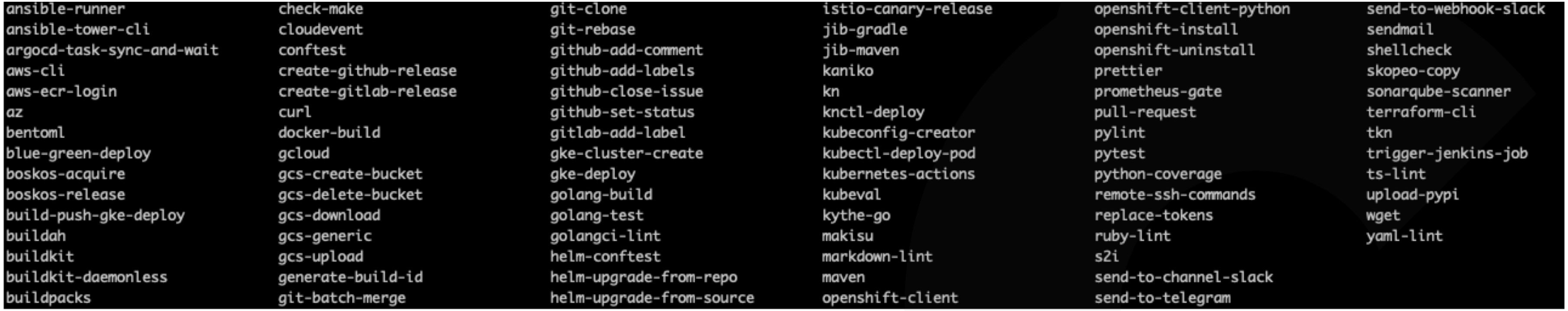
Tasks

A `Task` is a collection of `Steps` that you define and arrange in a specific order of execution as part of your continuous integration flow. A `Task` executes as a Pod on your Kubernetes cluster. A `Task` is available within a specific namespace, while a `ClusterTask` is available across the entire cluster.

```
apiVersion: tekton.dev/v1beta1
kind: Task
metadata:
 name: golang-build
....
spec:
 description: >-
   This Task is Golang task to build Go projects.
 params:
 - name: packages
   description: "packages to build (default: ./cmd/...)"
   default: "./cmd/..."
....
 steps:
 - name: build
   image: golang:$(params.version)
   workingDir: $(workspaces.source.path)
   script: |
     go build $(params.flags) $(params.packages)
```

```
apiVersion: tekton.dev/v1beta1
kind: Task
metadata:
 name: gcloud
....
 params:
 - name: gcloud-image
   description: gcloud CLI container image to run this task
   default: google/cloud-sdk:slim
....
 steps:
 - name: gcloud
   image: "$(params.gcloud-image)"
   command: ["/usr/bin/gcloud"]
   args: ["$(params.ARGS)"]
```

Task Catalog (Examples)



```
ansible-runner                  check-make                      git-clone                       istio-canary-release            openshift-client-python         send-to-webhook-slack
ansible-tower-cli               cloudevent                      git-rebase                      jib-gradle                      openshift-install               sendmail
argocd-task-sync-and-wait       conftest                        github-add-comment              jib-maven                       openshift-uninstall             shellcheck
aws-cli                         create-github-release           github-add-labels               kaniko                          prettier                        skopeo-copy
aws-ecr-login                   create-gitlab-release           github-close-issue              kn                              prometheus-gate                 sonarqube-scanner
az                              curl                            github-set-status               knctl-deploy                    pull-request                    terraform-cli
bentoml                         docker-build                    gitlab-add-label                kubeconfig-creator              pylint                          tkn
blue-green-deploy               gcloud                          gke-cluster-create              kubectl-deploy-pod              pytest                          trigger-jenkins-job
boskos-acquire                  gcs-create-bucket               gke-deploy                      kubernetes-actions              python-coverage                 ts-lint
boskos-release                  gcs-delete-bucket               golang-build                    kubeval                         remote-ssh-commands             upload-pypi
build-push-gke-deploy           gcs-download                    golang-test                     kythe-go                        replace-tokens                  wget
buildah                         gcs-generic                     golangci-lint                   makisu                          ruby-lint                       yaml-lint
buildkit                        gcs-upload                      helm-conftest                   markdown-lint                   s2i
buildkit-daemonless             generate-build-id               helm-upgrade-from-repo          maven                           send-to-channel-slack
buildpacks                      git-batch-merge                 helm-upgrade-from-source        openshift-client                send-to-telegram
```

https://github.com/tektoncd/catalog/tree/master/task

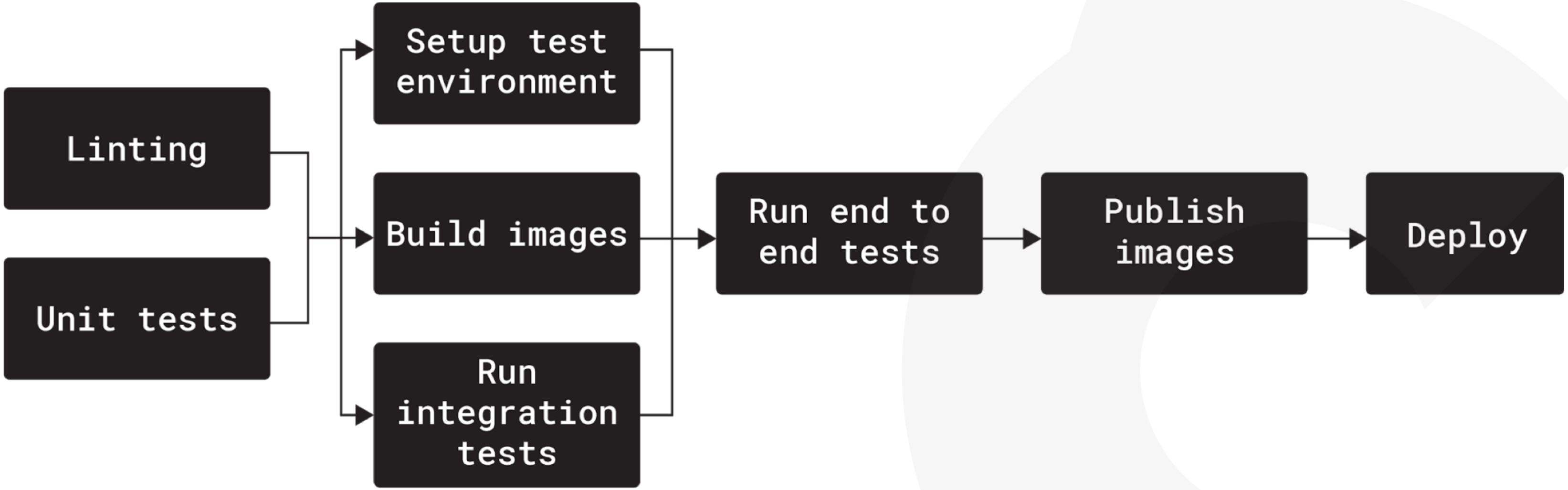
Pipelines / PipelineRuns

A `Pipeline` is a collection of `Tasks` that you define and arrange in a specific order of execution as part of your continuous integration flow. Each `Task` in a `Pipeline` executes as a `Pod` on your Kubernetes cluster. You can configure various execution conditions to fit your business needs.

```
apiVersion: tekton.dev/v1beta1
kind: Pipeline
metadata:
 name: sum-and-multiply-pipeline
spec:
 params:
   - name: a
     type: string
     default: "1"
 tasks:
   - name: sum-inputs
     taskRef:
       name: sum
     params:
....
         value: "$(params.b)"
   - name: multiply-inputs
     taskRef:
       name: multiply
....
```

```
---
apiVersion: tekton.dev/v1beta1
kind: PipelineRun
metadata:
 generateName: sum-and-multiply-pipeline-run-
spec:
 pipelineRef:
   name: sum-and-multiply-pipeline
 params:
   - name: a
     value: "2"
   - name: b
     value: "10"
```

Pipelines



Linting

Unit tests

Setup test environment

Build images

Run integration tests

Run end to end tests

Publish images

Deploy

Tekton Triggers

Triggers enables users to map fields from an event payload into resource templates. Put another way, this allows events to both model and instantiate themselves as Kubernetes resources. In the case of `tektoncd/pipeline`, this makes it easy to encapsulate configuration into `PipelineRuns` and `PipelineResources`.

Trigger Templates

A resource that can template other resources. Fully parameterized and able to be substituted `anywhere` within the template.

Trigger Bindings

`TriggerBindings` bind against events/triggers. `TriggerBindings` enable you to capture fields from an event and store them as parameters. The separation of `TriggerBindings` from `TriggerTemplates` was deliberate to encourage reuse between them.

Event Listeners

Kubernetes CRDs that allows users a declarative way to process incoming HTTP based events with JSON payloads. Users can declare `TriggerBindings` to extract fields from events, and apply them to `TriggerTemplates` in order to create Tekton resources.

Building It

03

Google Cloud

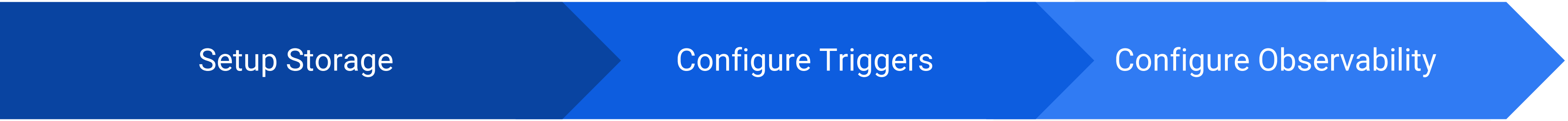
Standing it up

```
#create cluster
gcloud container clusters create tekton --zone=us-central1-a --cluster-version=1.16.13-gke.1


#grant permissions
kubectl create clusterrolebinding cluster-admin-binding \
--clusterrole=cluster-admin \
--user=$(gcloud config get-value core/account)


#install pipelines
kubectl apply --filename https://storage.googleapis.com/tekton-releases/pipeline/latest/release.yaml

#install triggers
kubectl apply --filename https://storage.googleapis.com/tekton-releases/triggers/latest/release.yaml

#install dashboard
kubectl apply --filename https://github.com/tektoncd/dashboard/releases/latest/download/tekton-dashboard-release.yaml

#watch for installation to complete
kubectl get pods --namespace tekton-pipelines --watch
```

Let’s Get Going!

04

Other Tasks

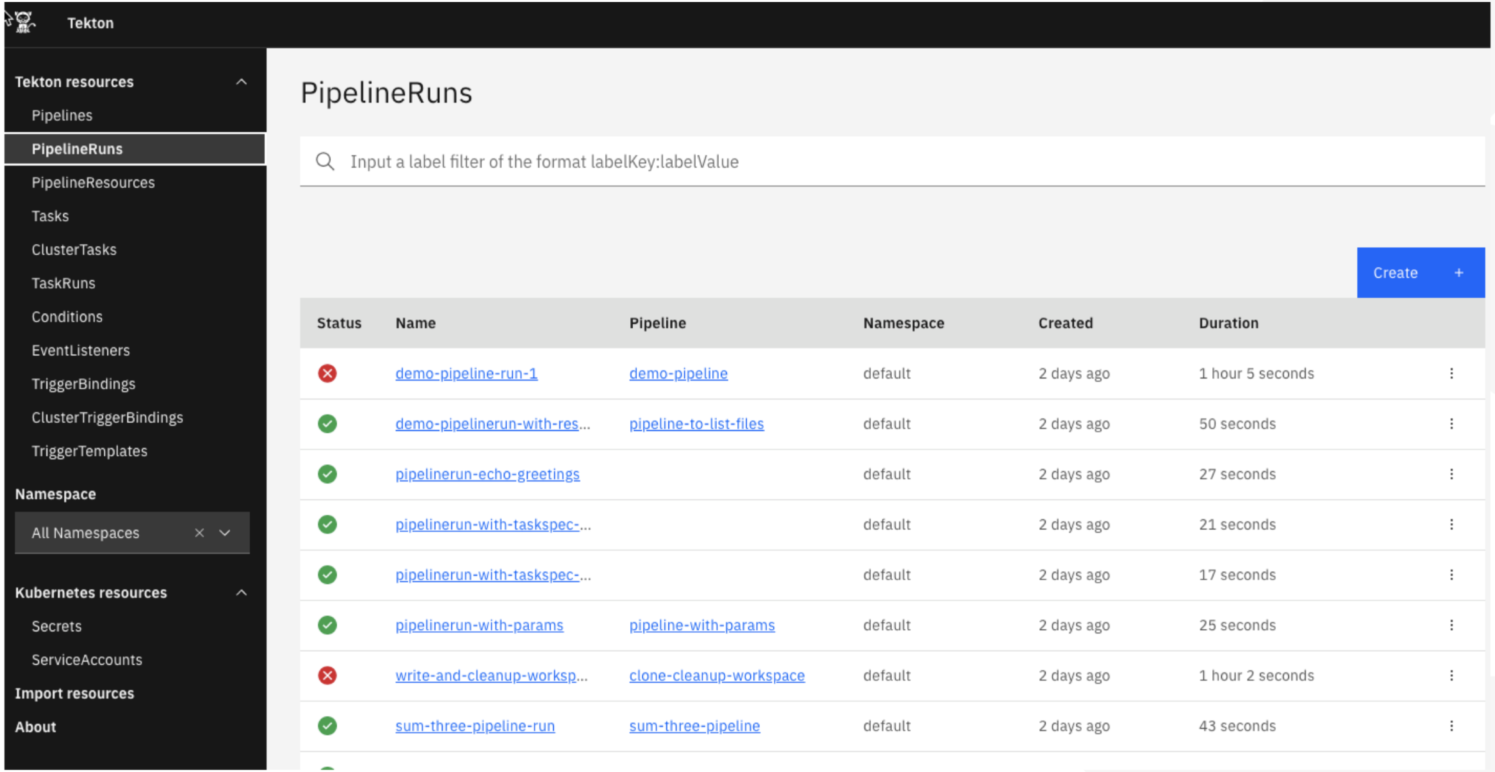


Setup Storage

Configure Triggers

Configure Observability

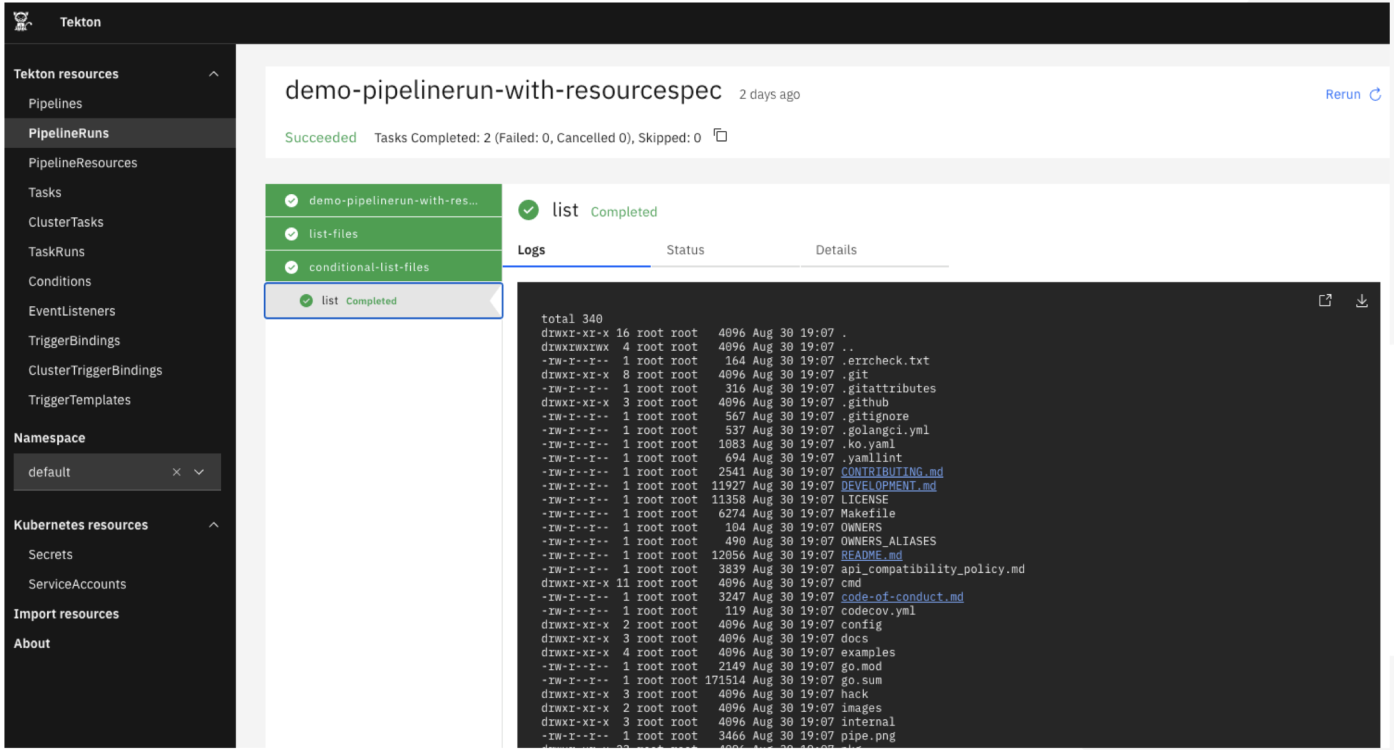
Pipeline Runs



PipelineRuns

Status	Name	Pipeline	Namespace	Created	Duration
✗	demo-pipeline-run-1	demo-pipeline	default	2 days ago	1 hour 5 seconds
✓	demo-pipelinerun-with-res...	pipeline-to-list-files	default	2 days ago	50 seconds
✓	pipelinerun-echo-greetings		default	2 days ago	27 seconds
✓	pipelinerun-with-taskspec-...		default	2 days ago	21 seconds
✓	pipelinerun-with-taskspec-...		default	2 days ago	17 seconds
✓	pipelinerun-with-params	pipeline-with-params	default	2 days ago	25 seconds
✗	write-and-cleanup-worksp...	clone-cleanup-workspace	default	2 days ago	1 hour 2 seconds
✓	sum-three-pipeline-run	sum-three-pipeline	default	2 days ago	43 seconds

Pipeline Run Output



demo-pipelinerun-with-resourcespec 2 days ago

Succeeded Tasks Completed: 2 (Failed: 0, Cancelled 0), Skipped: 0

list Completed

```
total 340
drwxr-xr-x 16 root root   4096 Aug 30 19:07 .
drwxrwxrwx  4 root root   4096 Aug 30 19:07 ..
-rw-r--r--  1 root root    164 Aug 30 19:07 .errcheck.txt
drwxr-xr-x  8 root root   4096 Aug 30 19:07 .git
-rw-r--r--  1 root root    316 Aug 30 19:07 .gitattributes
drwxr-xr-x  3 root root   4096 Aug 30 19:07 .github
-rw-r--r--  1 root root    567 Aug 30 19:07 .gitignore
-rw-r--r--  1 root root    537 Aug 30 19:07 .golangci.yml
-rw-r--r--  1 root root   1083 Aug 30 19:07 .ko.yaml
-rw-r--r--  1 root root    694 Aug 30 19:07 .yamllint
-rw-r--r--  1 root root   2541 Aug 30 19:07 CONTRIBUTING.md
-rw-r--r--  1 root root  11927 Aug 30 19:07 DEVELOPMENT.md
-rw-r--r--  1 root root  11358 Aug 30 19:07 LICENSE
-rw-r--r--  1 root root   6274 Aug 30 19:07 Makefile
-rw-r--r--  1 root root    104 Aug 30 19:07 OWNERS
-rw-r--r--  1 root root    490 Aug 30 19:07 OWNERS_ALIASES
-rw-r--r--  1 root root  12056 Aug 30 19:07 README.md
-rw-r--r--  1 root root   3839 Aug 30 19:07 api_compatibility_policy.md
drwxr-xr-x 11 root root   4096 Aug 30 19:07 cmd
-rw-r--r--  1 root root   3247 Aug 30 19:07 code-of-conduct.md
-rw-r--r--  1 root root    119 Aug 30 19:07 codecov.yml
drwxr-xr-x  2 root root   4096 Aug 30 19:07 config
drwxr-xr-x  3 root root   4096 Aug 30 19:07 docs
drwxr-xr-x  4 root root   4096 Aug 30 19:07 examples
-rw-r--r--  1 root root   2149 Aug 30 19:07 go.mod
-rw-r--r--  1 root root 171514 Aug 30 19:07 go.sum
drwxr-xr-x  3 root root   4096 Aug 30 19:07 hack
drwxr-xr-x  2 root root   4096 Aug 30 19:07 images
drwxr-xr-x  3 root root   4096 Aug 30 19:07 internal
-rw-r--r--  1 root root   3466 Aug 30 19:07 pipe.png
```

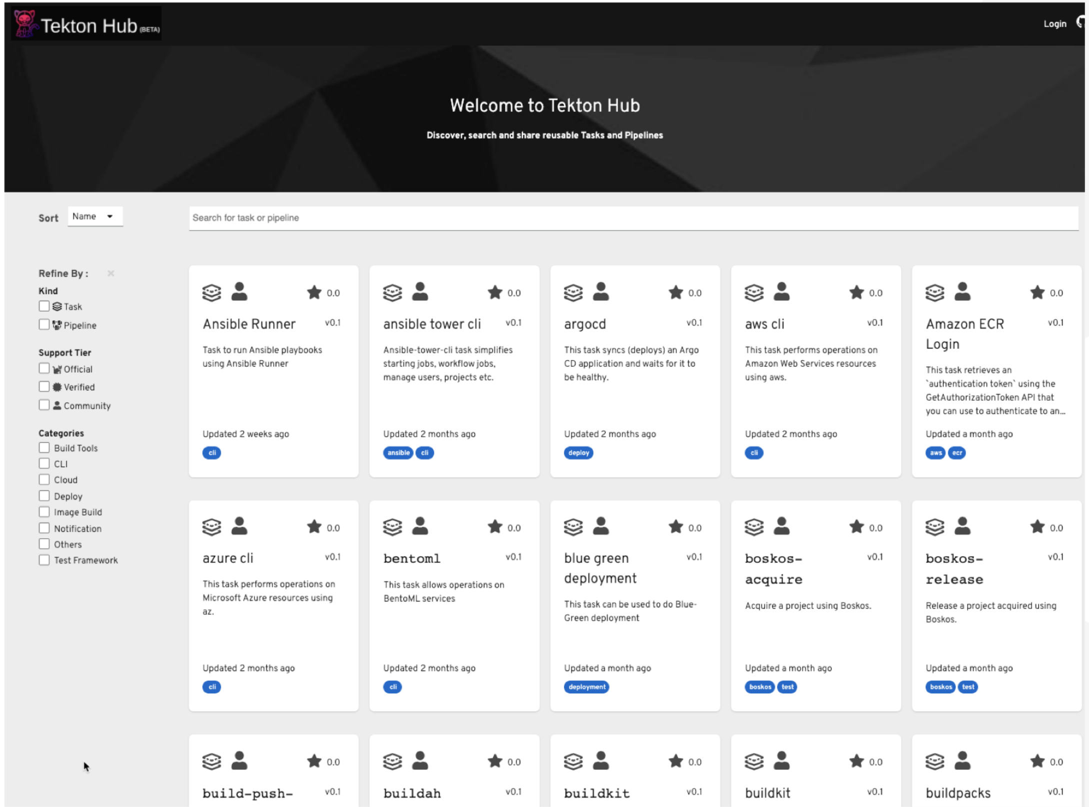
05

Additional Resources

Tekton Resources

- ➔ https://tekton.dev/docs/
- ➔ https://github.com/tektoncd
- ➔ https://github.com/tektoncd/community
- ➔ https://hub-preview.tekton.dev/

Hub (Preview)



Tekton Hub (BETA) Login

Welcome to Tekton Hub

Discover, search and share reusable Tasks and Pipelines

Sort: Name ▾

Search for task or pipeline

Refine By : ×

Kind
- □ Task
- □ Pipeline

Support Tier
- □ Official
- □ Verified
- □ Community

Categories
- □ Build Tools
- □ CLI
- □ Cloud
- □ Deploy
- □ Image Build
- □ Notification
- □ Others
- □ Test Framework

★ 0.0 Ansible Runner v0.1
Task to run Ansible playbooks using Ansible Runner
Updated 2 weeks ago
cli

★ 0.0 ansible tower cli v0.1
Ansible-tower-cli task simplifies starting jobs, workflow jobs, manage users, projects etc.
Updated 2 months ago
ansible cli

★ 0.0 argocd v0.1
This task syncs (deploys) an Argo CD application and waits for it to be healthy.
Updated 2 months ago
deploy

★ 0.0 aws cli v0.1
This task performs operations on Amazon Web Services resources using aws.
Updated 2 months ago
cli

★ 0.0 Amazon ECR Login v0.1
This task retrieves an `authentication token` using the GetAuthorizationToken API that you can use to authenticate to an...
Updated a month ago
aws ecr

★ 0.0 azure cli v0.1
This task performs operations on Microsoft Azure resources using az.
Updated 2 months ago
cli

★ 0.0 bentoml v0.1
This task allows operations on BentoML services
Updated 2 months ago
cli

★ 0.0 blue green deployment v0.1
This task can be used to do Blue-Green deployment
Updated a month ago
deployment

★ 0.0 boskos-acquire v0.1
Acquire a project using Boskos.
Updated a month ago
boskos test

★ 0.0 boskos-release v0.1
Release a project acquired using Boskos.
Updated a month ago
boskos test

★ 0.0 build-push- v0.1

★ 0.0 buildah v0.1

★ 0.0 buildkit v0.1

★ 0.0 buildkit v0.1

★ 0.0 buildpacks v0.1

Thank you