Continuous Delivery and GitOps on OpenShift

Wanja Pernath

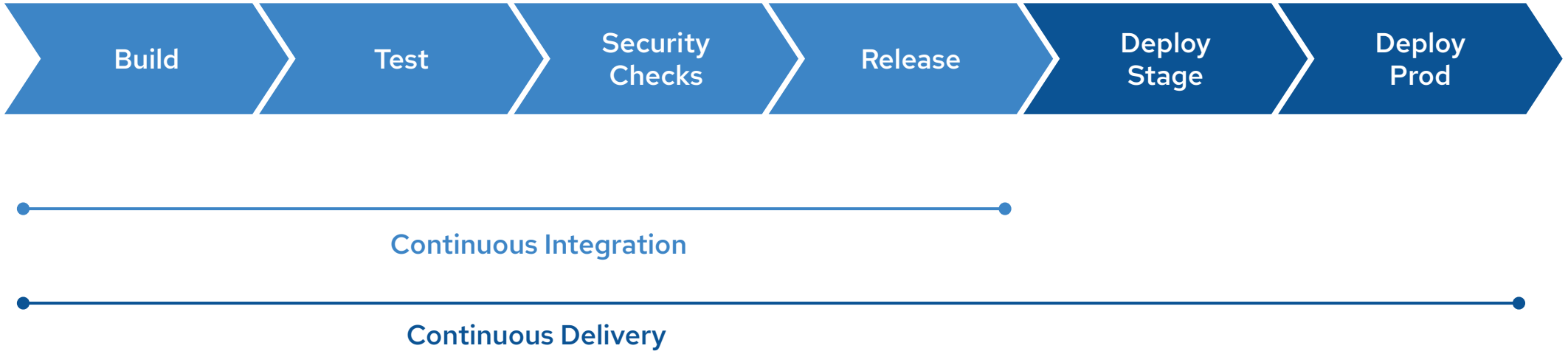
Red Hat

DevOps is the key to meet the insatiable demand for delivering quality applications rapidly

Red Hat

Continuous Integration(CI) & Continuous Delivery (CD)

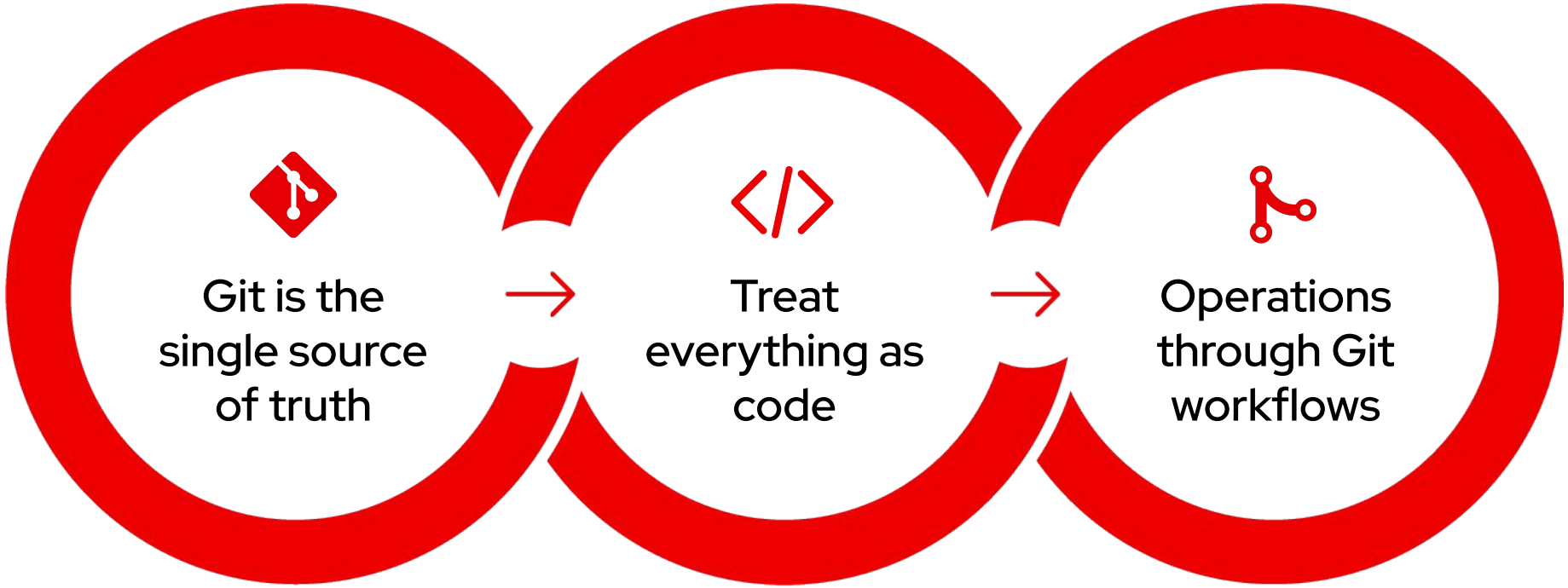
A key DevOps principle for automation, consistency and reliability



Red Hat

What is GitOps?

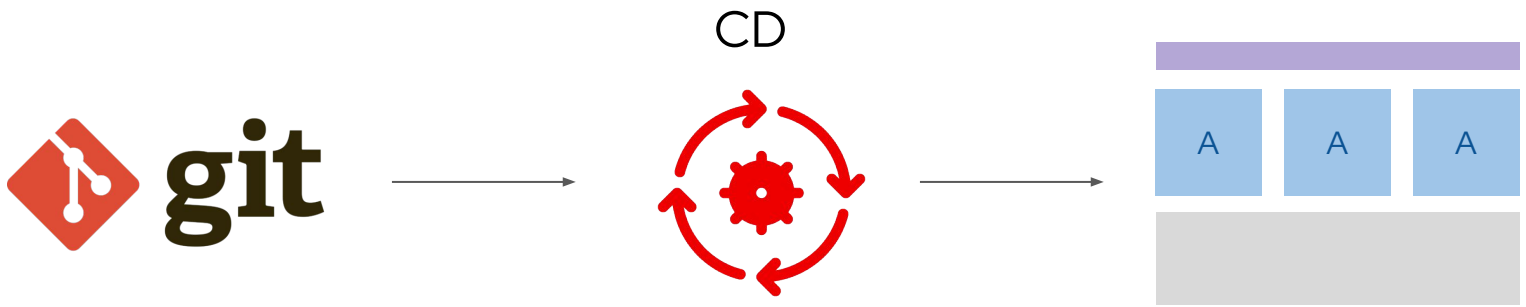
An developer-centric approach to Continuous Delivery and infrastructure operation



Red Hat

GitOps Workflow

a declarative approach to application delivery



What you want
(desired state)

What you have
(current state)

Red Hat

Why GitOps?

Standard Workflow

Familiar tools and Git workflows from application development teams

Enhanced Security

Review changes beforehand, detect configuration drifts, and take action

Visibility and Audit

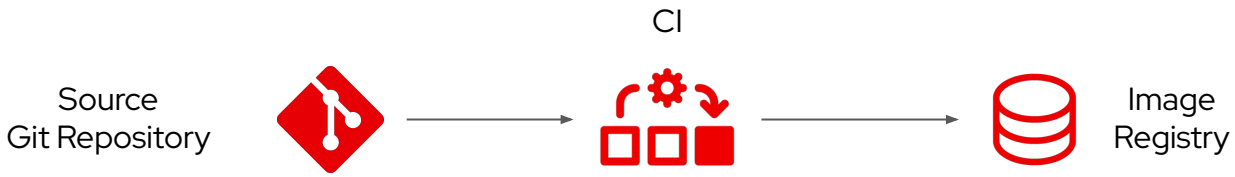
Capturing and tracing any change to clusters through Git history

Multi-cluster consistency

Reliably and consistently configure multiple Kubernetes clusters and deployment

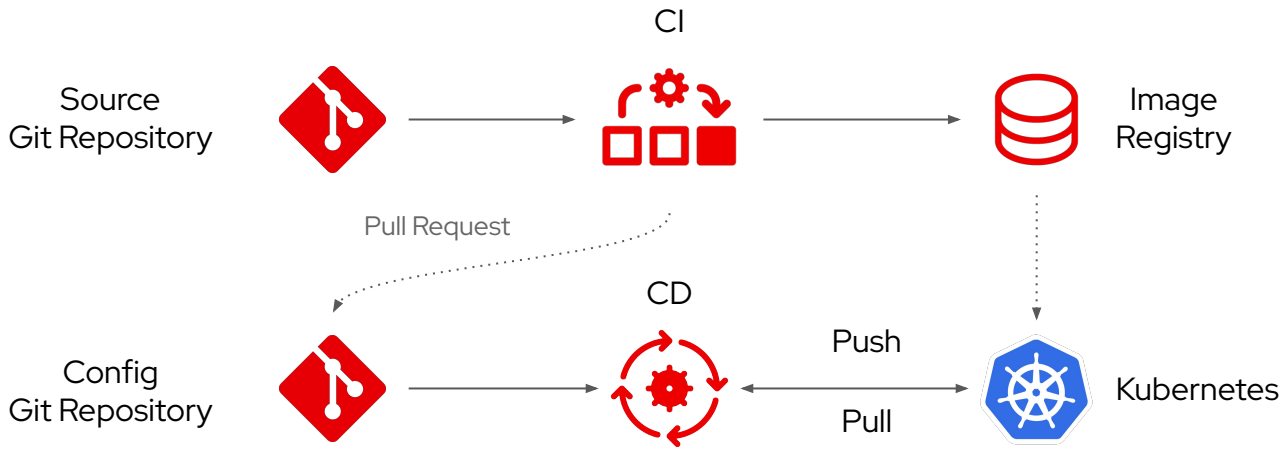
Red Hat

The GitOps Application Delivery Model



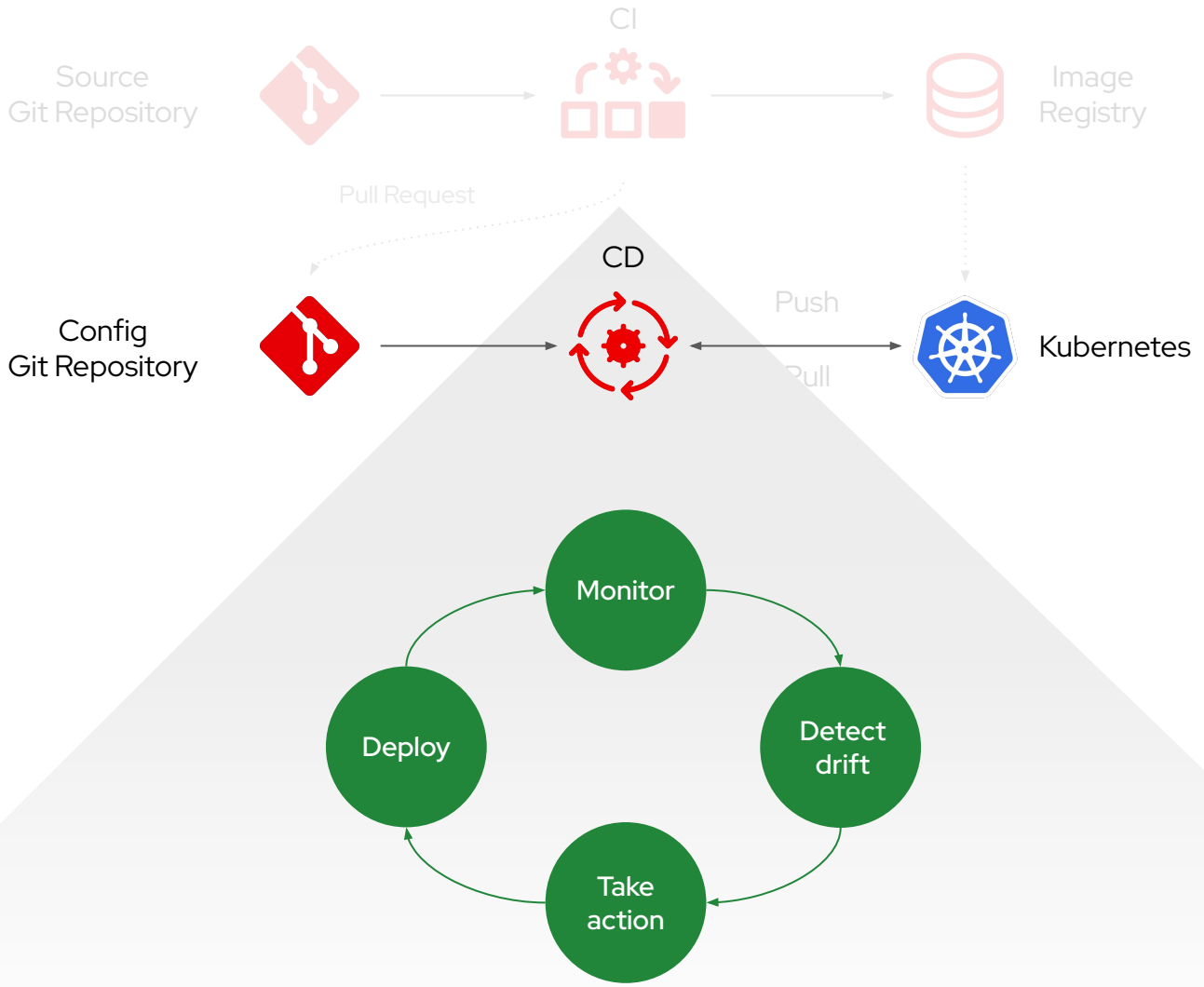
Red Hat

The GitOps Application Delivery Model



Red Hat

The GitOps Application Delivery Model

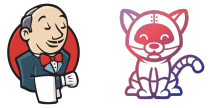


Red Hat

A Comprehensive DevOps Platform for Hybrid Cloud

Automate building container images using Kubernetes tools

Kubernetes-native on-demand delivery pipelines



Declarative GitOps for multi-cluster continuous delivery

OpenShift Builds | **OpenShift Pipelines** | **OpenShift GitOps**

OpenShift

GitHub GitLab Azure DevOps Jenkins

CD.FOUNDATION
CLOUD NATIVE COMPUTING FOUNDATION
Red Hat

Continuous Integration & Continuous Delivery

OpenShift Build

Automate building container images using Kubernetes tools

OpenShift Pipelines

Kubernetes-native on-demand delivery pipelines

OpenShift GitOps

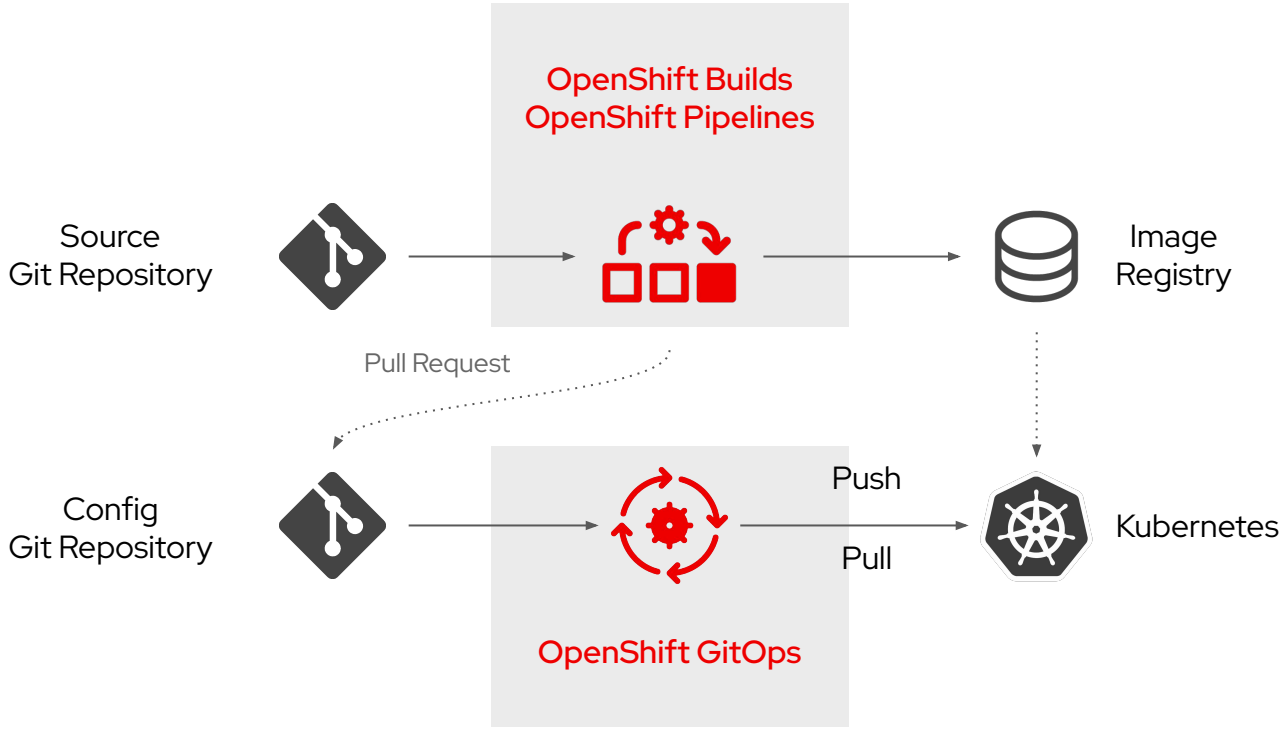
Declarative GitOps for multi-cluster continuous delivery

Ecosystem Integrations

GitHub GitLab Azure DevOps Travis CI

Red Hat

The GitOps Application Delivery Model on OpenShift



Red Hat

OpenShift Builds

Automate building container images
using Kubernetes tools

Red Hat

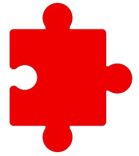
OpenShift Builds

Kubernatives-native image build

A Kubernative-native way to building container images on OpenShift which is portable across Kubernetes distros

Supports multiple build strategies

Choose the build strategy that fits best your applications and skills: source-to-image, Dockerfile, and Cloud-Native Buildpacks



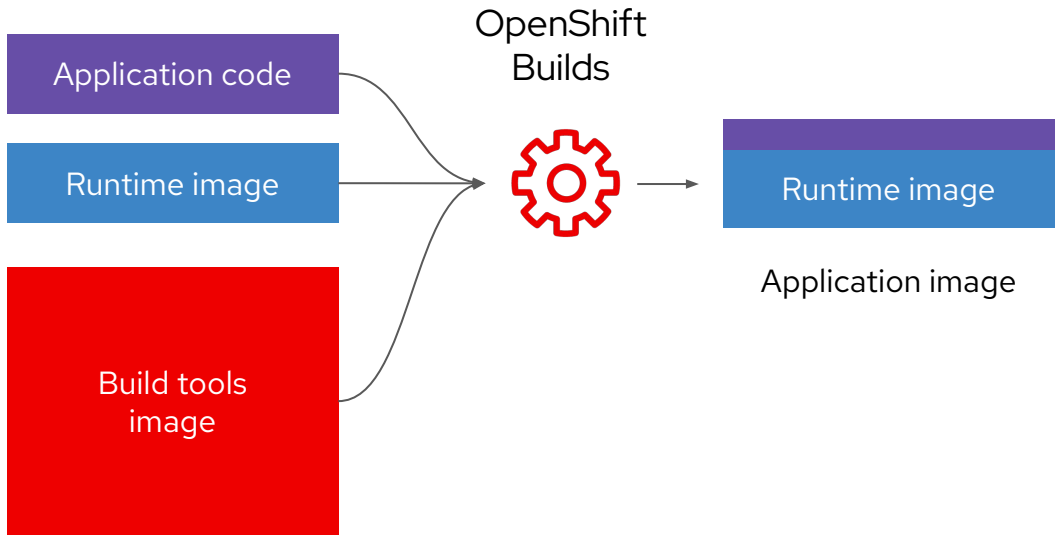
Extend with additional build strategies

Extend to use community Kubernetes builds strategies or your own custom builds

Red Hat

OpenShift Builds

- Build images on OpenShift and Kubernetes
- Use Kubernetes builds tools
  - Source-to-Image
  - Buildpacks
  - Buildah
  - Kaniko
  - ...more
- Create lean application images
- Extend with your own build tools
- Based on Shipwright open-source project



Red Hat

OpenShift Builds

BuildStrategy

How to build images e.g. S2I, Buildpacks, etc

Build

What to build

BuildRun

Build execution details

Red Hat

OpenShift Builds

Cloud-Native Buildpacks

```
kind: Build
metadata:
  name: myapp-buildpack
spec:
  source:
    url: https://github.com/myorg/myapp
  strategy:
    name: buildpacks-v3
  builder:
    image: paketobuildpacks/builder:full
  output:
    image: quay.io/myorg/myapp:v1
```

Source-to-Image (S2I)

```
kind: Build
metadata:
 name: myapp-s2i
spec:
 source:
   url: https://github.com/myorg/myapp
 strategy:
   name: source-to-image
 builder:
   image: registry.redhat.io/openjdk/openjdk-11-rhel8
 output:
   image: quay.io/myorg/myapp:v1
 runtime:
   image: docker.io/openjdk:11-jre-slim
```


Red Hat

OpenShift Builds v2 Roadmap

OpenShift Builds v2 Dev Preview
- Build from application binary
- Kaniko build strategy

Q4 CY20

1H CY21

OpenShift Builds v2 Tech Preview
- Builds in air-gapped clusters
- Proxy support
- Shipwright CLI

OpenShift Builds v2 GA
- Supported S2I, Buildah and Buildpacks strategies
- Trigger builds from Git
- Unprivileged builds
- Volume support
- Integrated in Dev Console

2H CY21

Red Hat

OpenShift Pipelines

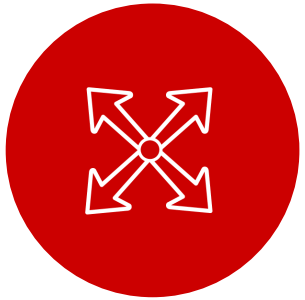
Kubernetes-native on-demand delivery pipelines

Red Hat

What is Cloud-Native CI/CD?

Containers

Built for container apps and runs on Kubernetes



Serverless

Runs serverless with no CI/CD engine to manage and maintain

DevOps

Designed with microservices and distributed teams in mind

Red Hat

Why Cloud-Native CI/CD?

Traditional CI/CD	Cloud-Native CI/CD
Designed for Virtual Machines	Designed for Containers and Kubernetes
Require IT Ops for CI engine maintenance	Pipeline as a service with no Ops overhead
Plugins shared across CI engine	Pipelines fully isolated from each other
Plugin dependencies with undefined update cycles	Everything lifecycled as container images
No interoperability with Kubernetes resources	Native Kubernetes resources
Admin manages persistence	Platform manages persistence
Config baked into CI engine container	Configured via Kubernetes ConfigMaps

Red Hat

An open-source project for providing a set of shared and standard components for building Kubernetes-style CI/CD systems

Governed by the Continuous Delivery Foundation

Contributions from Google, Red Hat, Cloudbees, IBM, Pivotal and many more

OpenShift Pipelines

Built for Kubernetes

Cloud-native pipelines taking advantage of Kubernetes execution and , operational model and concepts

Scale on-demand

Pipelines run and scale on-demand in isolated containers, with repeatable and predictable outcomes

Secure pipeline execution

Kubernetes RBAC and security model ensures security consistently across pipelines and workloads

Flexible and powerful

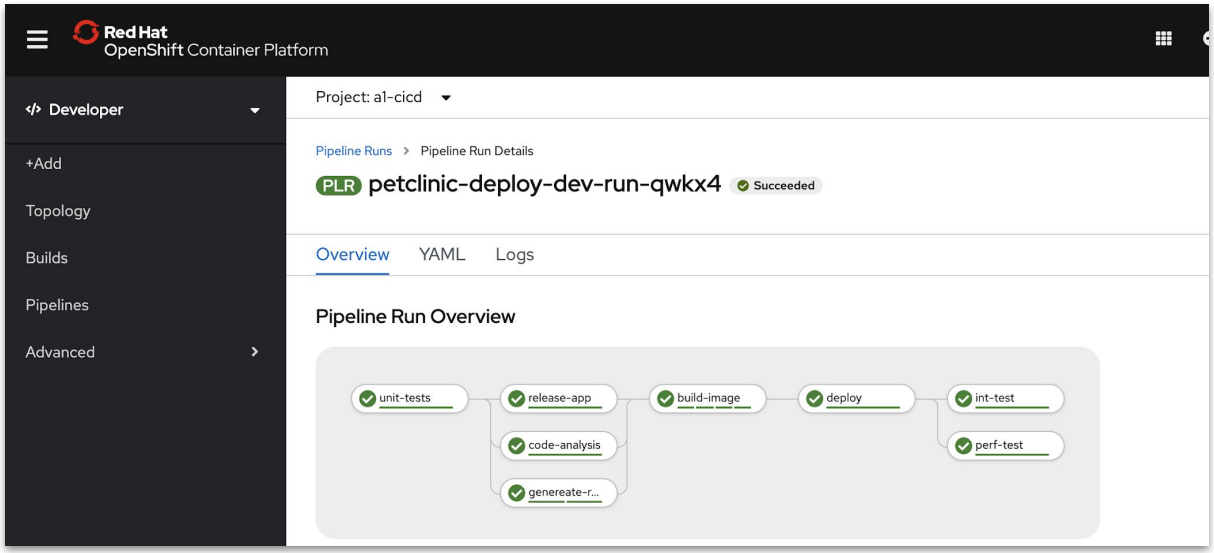
Granular control over pipeline execution details on Kubernetes, to support your exact requirements

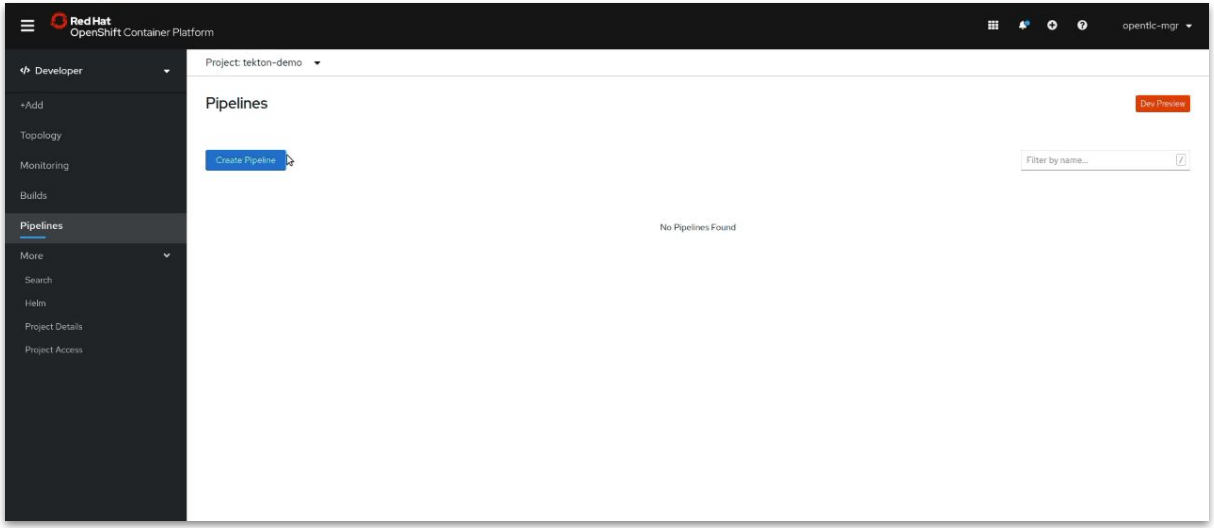
Powered by 

Red Hat

OpenShift Pipelines

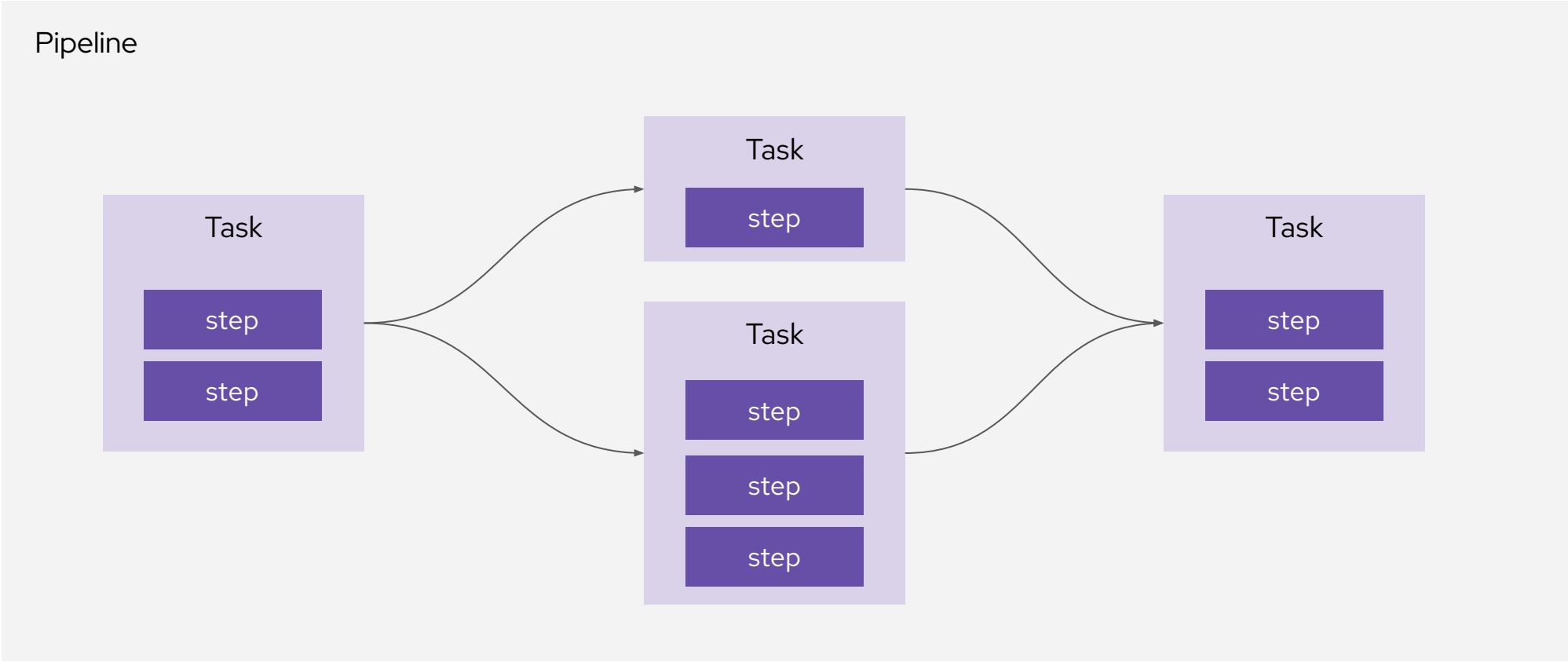
- Based on Tekton Pipelines
- Kubernetes-native declarative CI/CD
- Pipelines run on-demand in isolated containers
- No central server to maintain! No plugin conflicts!
- Task library and integration with Tekton Hub
- Secure pipelines aligned with Kubernetes RBAC
- Visual and IDE-based pipeline authoring
- Pipeline templates when importing apps
- Automated install and upgrades via OperatorHub
- CLI, Web, VS Code and IntelliJ plugins





Red Hat

Tekton Concepts



Red Hat

Tekton Concepts: `step`

- Run command or script in a container
- Kubernetes container spec
  - Env vars
  - Volumes
  - Config maps
  - Secrets

```
- name: build
  image: maven:3.6.0-jdk-8-slim
  command: [“mvn”]
  args: [“install”]
```

```
- name: parse-yaml
  image: python3
  script:|-
    #!/usr/bin/env python3
    ...
```


Red Hat

Tekton Concepts: Task

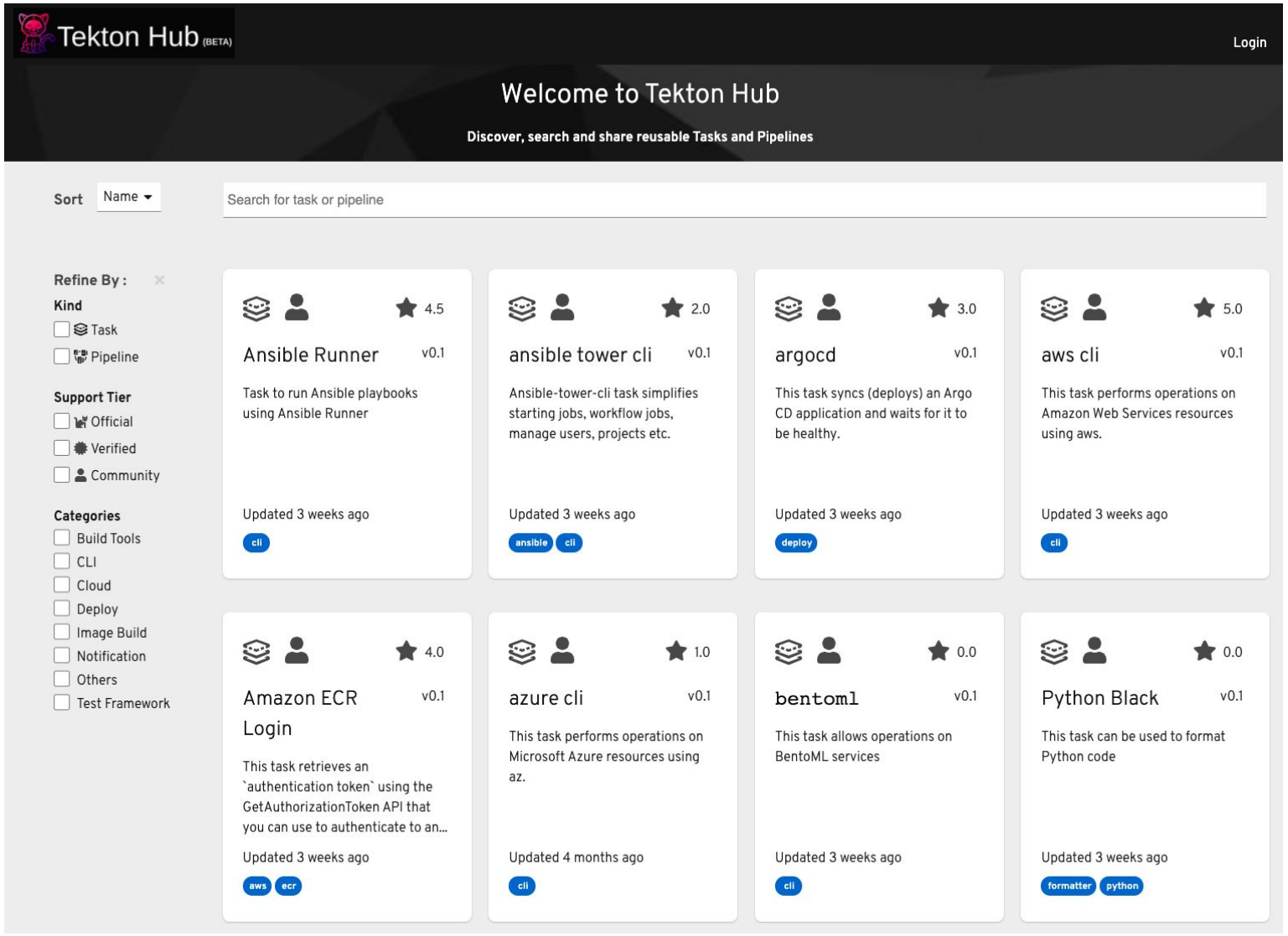
- Performs a specific task
- List of steps
- Steps run sequentially
- Reusable

```yaml
kind: Task
metadata:
 name: buildah
spec:
 params:
 - name: IMAGE
 steps:
 - name: build
   image: quay.io/buildah/stable:latest
   command: ["buildah"]
   args: ["bud", ".", "-t", "$(params.IMAGE)"]
 - name: push
   image: quay.io/buildah/stable:latest
   script: |
     buildah push $(params.IMAGE) docker://$(params.IMAGE)
```


Red Hat

Tekton Hub

Search, discover and install Tekton Tasks

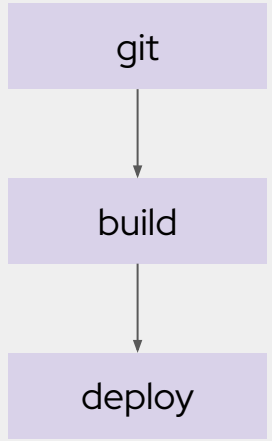


Red Hat

Tekton Concepts: `Pipeline`

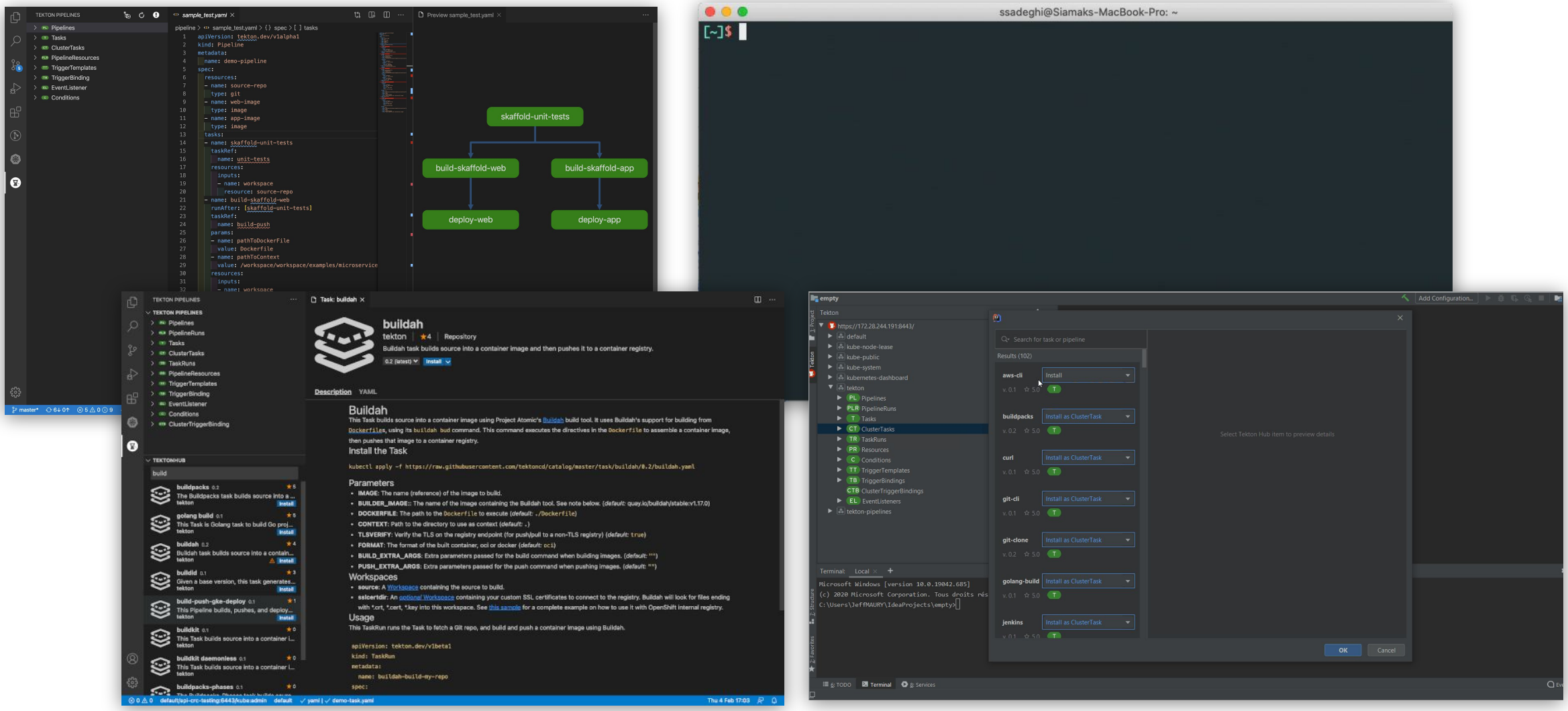
- A graph of Tasks: concurrent & sequential
- Tasks run on different nodes
- Task execution logic
  - Conditional
  - Retries
- Share data between tasks

```yaml
kind: Pipeline
metadata:
 name: deploy-dev
spec:
 params:
   - name: IMAGE_TAG
 tasks:
   - name: git
     taskRef:
       name: git-clone
     params: [...]
   - name: build
     taskRef:
       name: maven
     params: [...]
     runAfter: ["git"]
   - name: deploy
     taskRef:
       name: knative-deploy
     params: [...]
     runAfter: ["build"]
```



Red Hat

Tekton CLI, Visual Studio Code, and IntelliJ



Red Hat

OpenShift Pipelines Roadmap

OpenShift Pipelines 1.3 (TP)

- Proxy support
- Unprivileged pipelines
- TLS support in trigger EventListener
- Tekton CLI - search and install from Tekton Hub
- VS Code - Tekton Hub integration
- IntelliJ - Tekton Hub integration
- Console - pipeline metrics and events

Q1 2021

Q2 2021

OpenShift Pipelines 1.4 - GA

- Pipeline logs aggregation in cluster logging
- Enable TLS for EventListeners pods
- Automatic proxy env vars on TaskRuns
- ClusterTriggerBindings for BitBucket and GitLab

OpenShift Pipelines 1.5

- Pipeline as code
- Auto-pruning pipelineruns and taskruns
- OpenShift sandboxed containers (kata) support
- Jenkins to Tekton migration guide

Q3 2021

2H 2021

OpenShift Pipelines 1.6+

- Managed OpenShift GitOps on OSD
- Unprivileged builds in pipelines
- Approval gates in pipelines
- Auto workspace configs
- Default security tasks in pipeline templates
- In-cluster Tekton Hub and catalog
- Admin metrics dashboards
- Pipeline artifact archiving

Red Hat

OpenShift GitOps

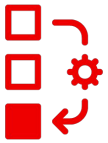
Declarative GitOps for multi-cluster continuous delivery

Red Hat

OpenShift GitOps

Multi-cluster config management

Declaratively manage cluster and application configurations across multi-cluster OpenShift and Kubernetes infrastructure with Argo CD



Automated Argo CD install and upgrade

Automated install, configurations and upgrade of Argo CD through OperatorHub

Opinionated GitOps bootstrapping

Bootstrap end-to-end GitOps workflows for application delivery using Argo CD and Tekton with GitOps Application Manager CLI

Deployments and environments insights

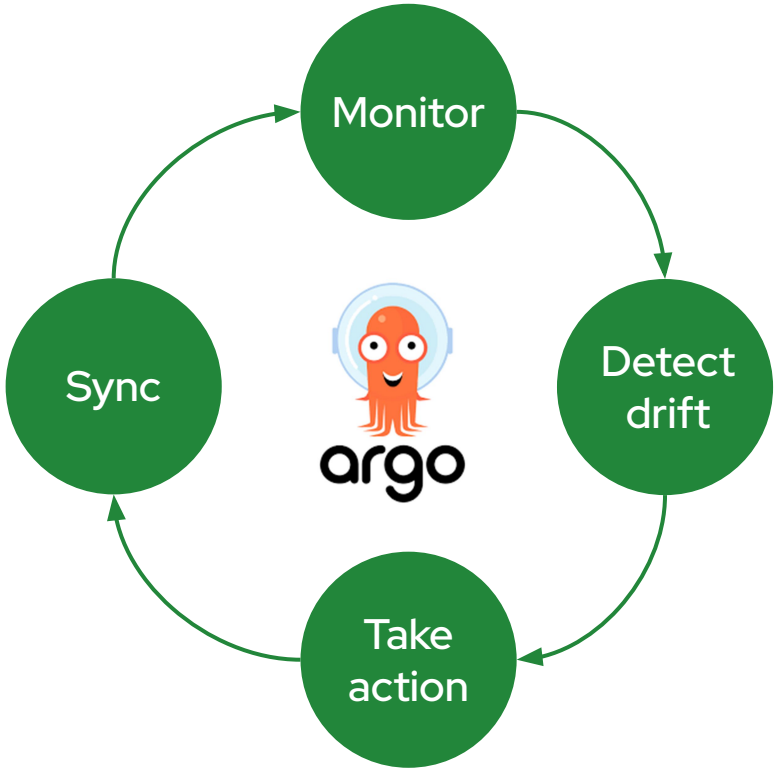
Visibility into application deployments across environments and the history of deployments in the OpenShift Console

Powered by argo

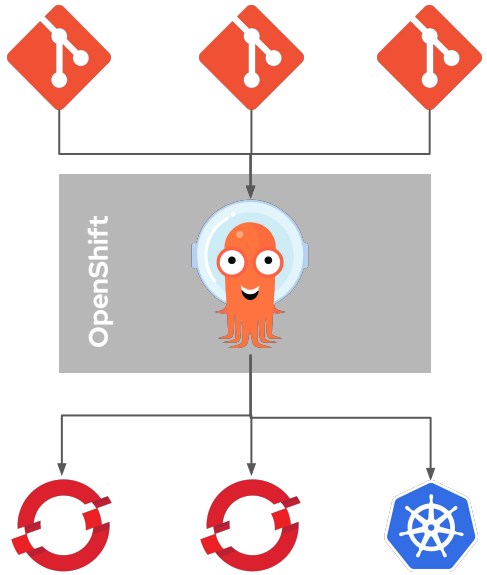
Red Hat

Argo CD

- Cluster and application configuration versioned in Git
- Automatically syncs configuration from Git to clusters
- Drift detection, visualization and correction
- Granular control over sync order for complex rollouts
- Rollback and rollforward to any Git commit
- Manifest templating support (Helm, Kustomize, etc)
- Visual insight into sync status and history

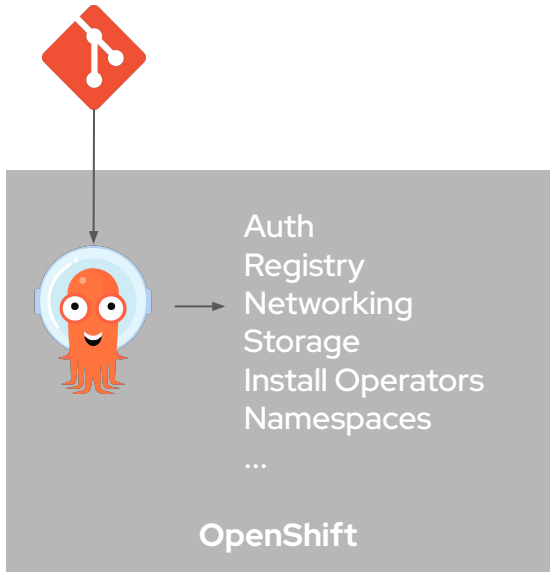


Flexible Deployment Strategies



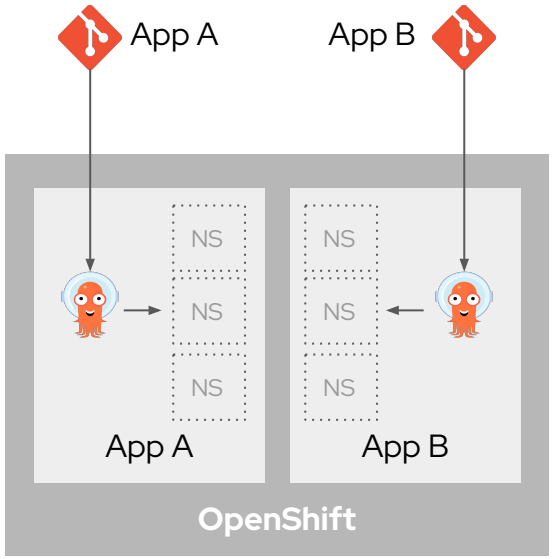
Central Hub (Push)

A central Argo CD pushes Git repository content to remote OpenShift and Kubernetes clusters



Cluster Scoped (Pull)

A cluster-scope Argo CD pulls cluster service configurations into into the OpenShift cluster



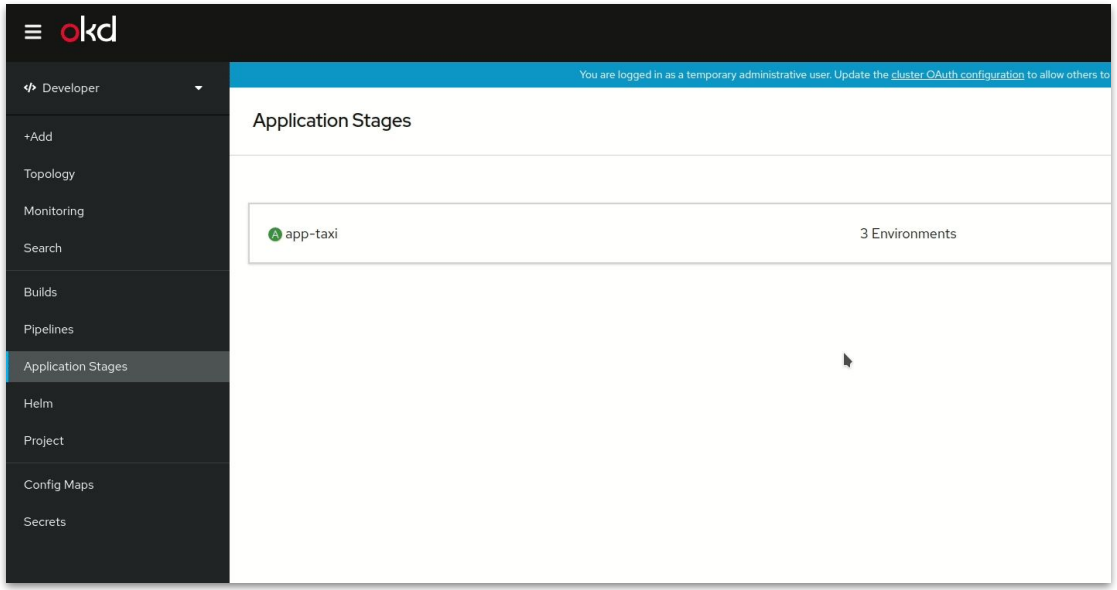
Application Scoped (Pull)

An application scoped Argo CD pulls application deployment and configurations into app namespaces

Red Hat

GitOps Application Manager CLI

- Bootstraps Git repos for GitOps
- Configures deployment environments
- Configures webhooks for Tekton Pipelines for CI
- Configures Argo CD for deployment to environments
- Kustomize for environment-specific configs
- Integration with secret managers

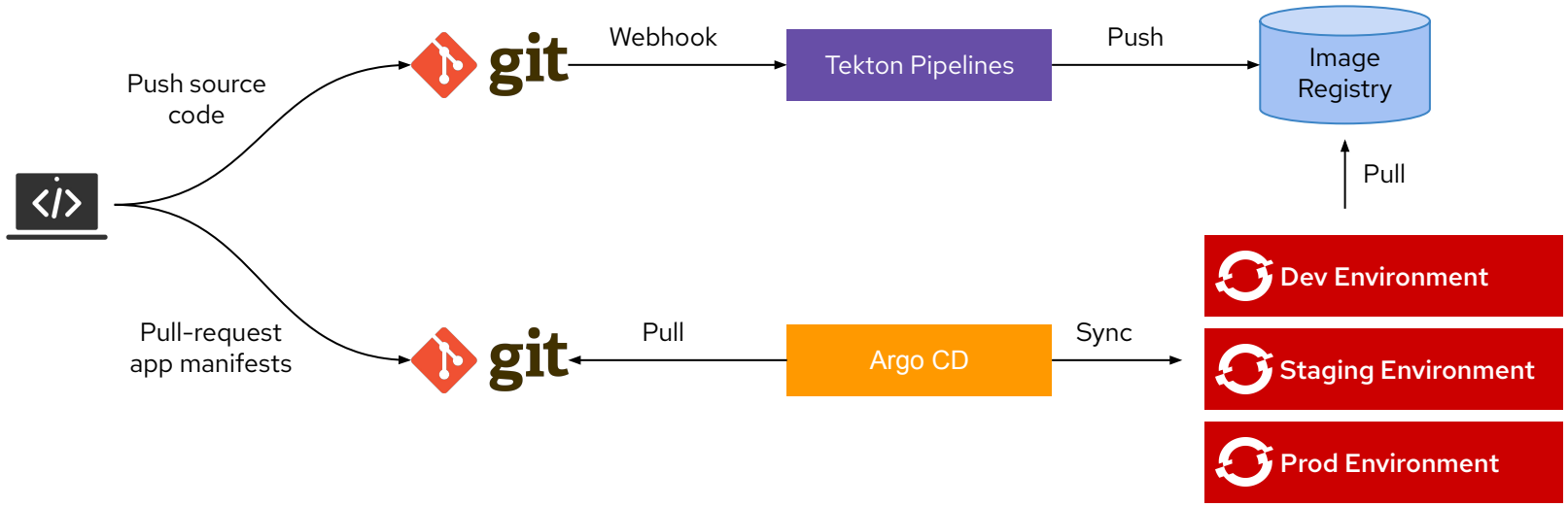



```
$ kam bootstrap
$ kam environment add stage
```


Red Hat

GitOps Application Manager CLI

```
$ kam bootstrap
```



Red Hat

OpenShift GitOps Roadmap

Q1 2021

OpenShift GitOps 1.0 - Tech Preview

- Pre-configured Argo CD for cluster configs
- Argo DD in air-gapped clusters
- Argo CD proxy config support
- Argo CD deployment guide
- Argo CD metrics in cluster Prometheus
- Out of sync alerts in cluster AlertManager

Q2 2021

OpenShift GitOps 1.1 - GA

- Argo CD logs aggregated in cluster logging stack
- Guidance on auth integrations with OpenShift
- Dynamic generation of Applications (ApplicationSets)

Q3 2021

OpenShift GitOps 1.2

- Cluster Argo CD uses OpenShift auth
- kam - simplified Git repo layout
- Add custom Argo CD plugins and tools
- Secret management guidance
- Dev Console – application environments views

Q4 2021

OpenShift GitOps 1.3+

- Namespaced Argo CD uses OpenShift auth
- Managed OpenShift GitOps on OSD
- ApplicationSets integration with RHACM
- Health status for OpenShift resources
- Multi-tenant Argo CD control plane
- Helm Charts in kam

Red Hat

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat