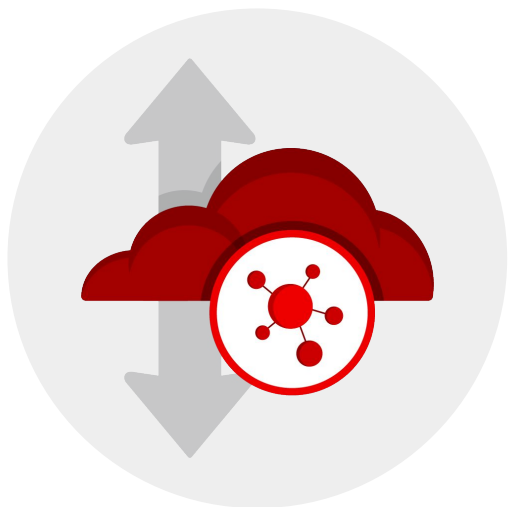


OPENSIFT OVERVIEW & ROADMAP

Kirsten Newcomer
Sr. Principal Product Manager
February 2020

Customer Challenges



Enjoy simplicity of public cloud in an on-premise environment

Create a consistent experience across public and on-premise

Plan for growth in container adoption while still running virtual machines

Expectations For A Hybrid Cloud Platform

BROAD ECOSYSTEM

**BROADEST APPLICATION
SUPPORT**

**DEVELOPER EXPERIENCE &
ON-DEMAND**

AUTOMATED OPERATIONS

**STANDARDS, PORTABILITY
& INTEROPERABILITY**

SECURITY & COMPLIANCE



Edge



Datacenter



Hybrid Multi-cloud

Red Hat OpenShift - the Hybrid Cloud platform

 Red Hat
Ansible Automation

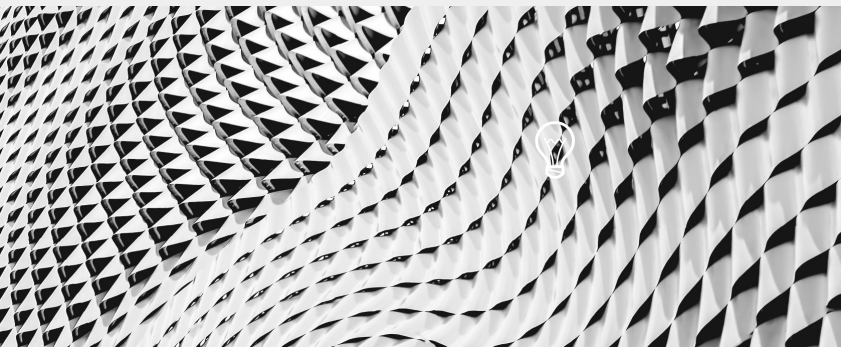
 Red Hat
CloudForms

 Red Hat
Satellite

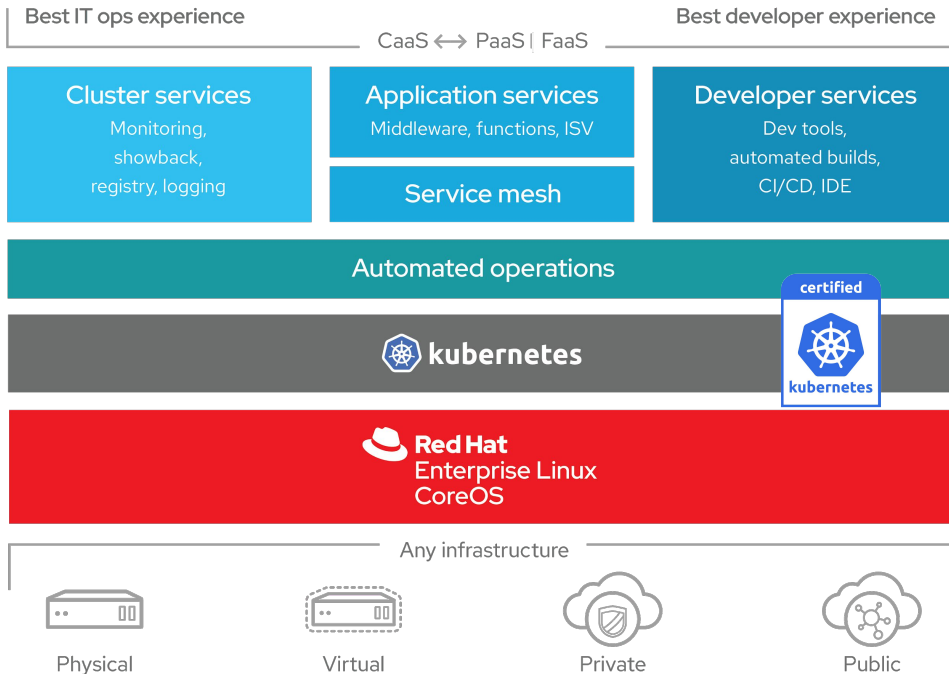
 Red Hat
Insights



AUTOMATED OPERATIONS WITH OPENSIFT 4



OpenShift 4 - A smarter Kubernetes platform



Automated, full-stack installation from the container host to application services

Seamless Kubernetes deployment to any cloud or on-premises environment

Autoscaling of cloud resources

One-click updates for platform, services, and applications

Automated Container Operations

FULLY AUTOMATED DAY-1 AND DAY-2 OPERATIONS

INSTALL	DEPLOY	HARDEN	OPERATE
AUTOMATED OPERATIONS			
Infra provisioning	Full-stack deployment	Secure defaults	Multicluster aware
Embedded OS	On-premises and cloud	Network isolation	Monitoring and alerts
	Unified experience	Audit and logs	Full-stack patch & upgrade
		Signing and policies	Zero-downtime upgrades
			Vulnerability scanning

Comprehensive Container Security



CONTROL

Application Security

Container Content

CI/CD Pipeline

Container Registry

Deployment Policies



DEFEND

Infrastructure

Container Platform

Container Host Multi-tenancy

Network Isolation

Storage

Audit & Logging

API Management



EXTEND

Security Ecosystem

Red Hat Contributions To Kubernetes



RBAC Authorization | Stateful Sets | Init Containers |
Rolling Update Status | Pod Security Policy Limits |
Memory based Pod Eviction | Quota Controlled
Services | 1,000+ Nodes | Dynamic PV Provisioning |
Multiple Schedulers | SECCOMP | Audit | Job
Scheduler | Access Review API | Whitelisting Sysctls |
Secure Cluster Policy | Evict Pods Disk IO | Storage
Classes | Azure Data Disk | etcdv3 | RBAC API | Auth to
kubelet API | Pod-level cGroups QoS | Kublet Eviction
Model | RBAC | Storage Class |
CustomResourceDefinitions | API Aggregation |
Encrypted secrets in etcd | Limit Node Access | HPA
Status Conditions | Network Policy | CRI Validation
Test Suite | Local Persistent Storage | Audit Logging |

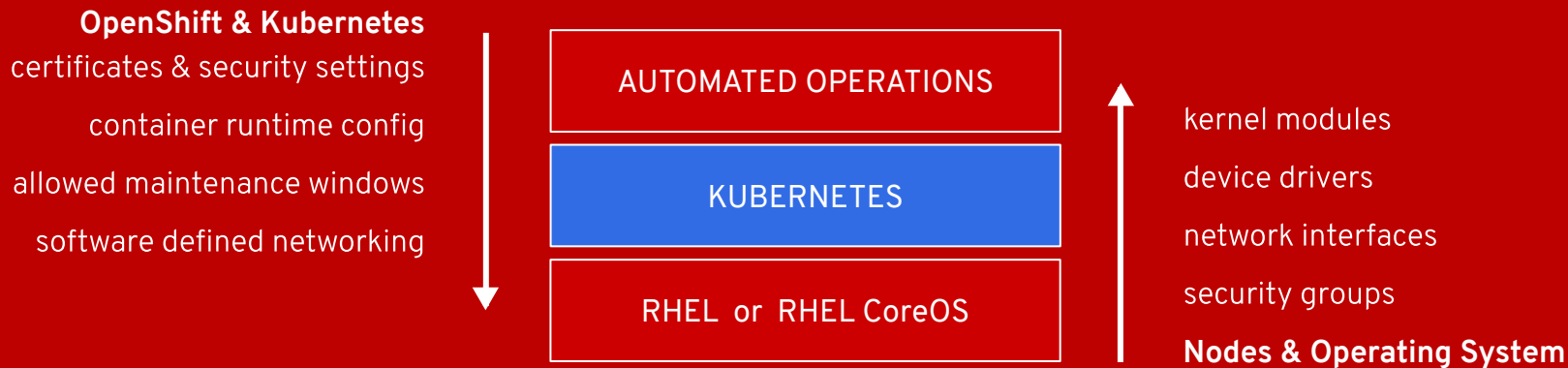


OPENSIFT



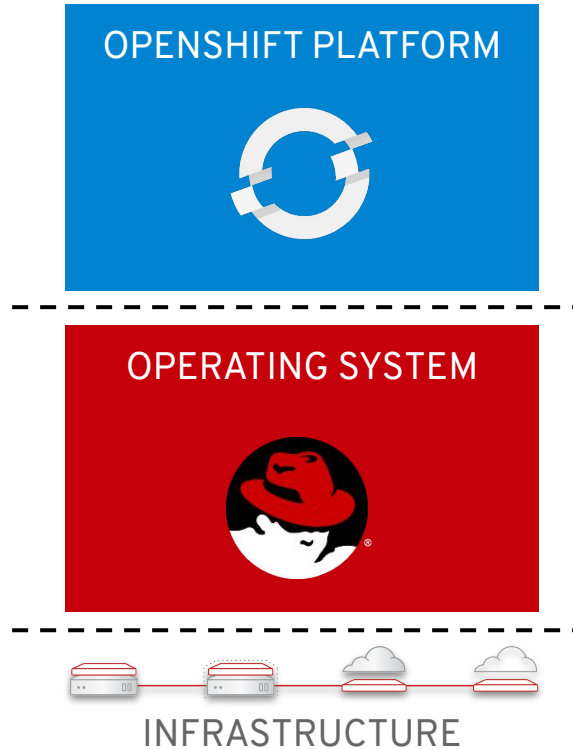
The New Platform Boundary

OpenShift 4 is aware of the entire infrastructure and brings the Operating System under management

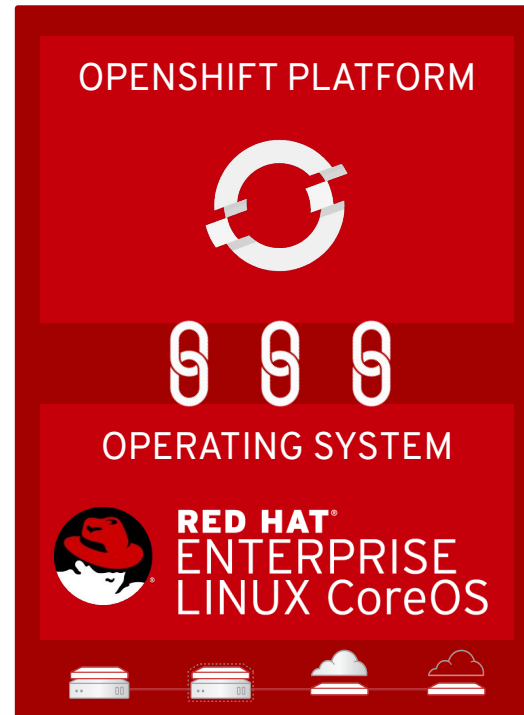


Full Stack Automated Install

OPENSIFT 3



OPENSIFT 4



Immutable Operating System

OPENSIFT 4

Red Hat Enterprise Linux CoreOS is versioned with OpenShift

CoreOS is tested and shipped in conjunction with the platform. Red Hat runs thousands of tests against these configurations.

Red Hat Enterprise Linux CoreOS is managed by the cluster

The Operating system is operated as part of the cluster, with the config for components managed by Machine Config Operator:

- CRI-O config
- Kubelet config
- Authorized registries
- SSH config



RED HAT ENTERPRISE LINUX

RED HAT[®] ENTERPRISE LINUX[®]

General Purpose OS

RED HAT[®] ENTERPRISE LINUX CoreOS

Immutable container host

BENEFITS

- 10+ year enterprise life cycle
- Industry standard security
- High performance on any infrastructure
- Customizable and compatible with wide ecosystem of partner solutions

- Self-managing, over-the-air updates
- Immutable and tightly integrated with OpenShift
- Host isolation is enforced via Containers
- Optimized performance on popular infrastructure

WHEN TO USE

When customization and integration with additional solutions is required

When cloud-native, hands-free operations are a top priority

Installation Experiences

OPENSIFT CONTAINER PLATFORM

Full Stack Automation

Simplified opinionated “Best Practices” for cluster provisioning

Fully automated installation and updates including host container OS.



Pre-existing Infrastructure

Customer managed resources & infrastructure provisioning

Plug into existing DNS and security boundaries



HOSTED OPENSIFT

Azure Red Hat OpenShift

Deploy directly from the Azure console. Jointly managed by Red Hat and Microsoft Azure engineers.

OpenShift Dedicated

Get a powerful cluster, fully Managed by Red Hat engineers and support.

Day 2 Configuration

Global Configuration

You complete most of the cluster configuration and customization after you deploy your OpenShift Container Platform cluster.

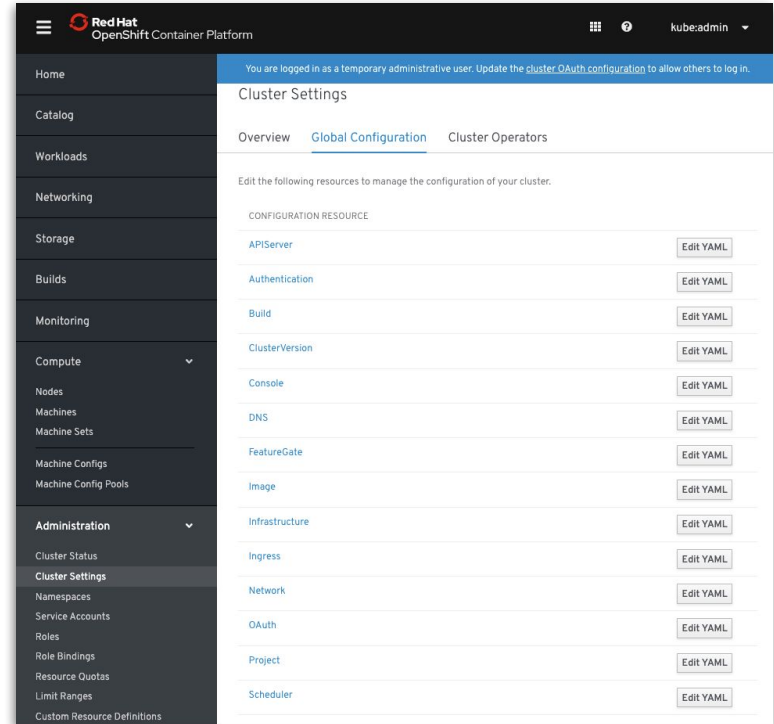
Change via Cluster Settings screen

Once you have discovered your desired settings (prev. slide), changes can be made via Console or CLI.

Operators apply these updates

One or more Operators are responsible for propagating these settings through the infrastructure

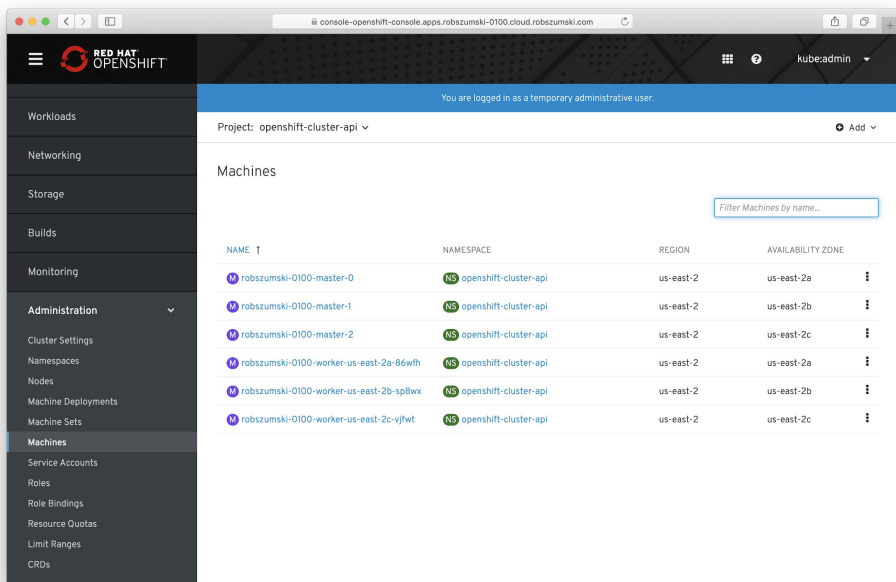
- Networking (SDN)
- Identity Provider
- Logging, Metrics



Generally Available

Kubernetes Machine Api Operator

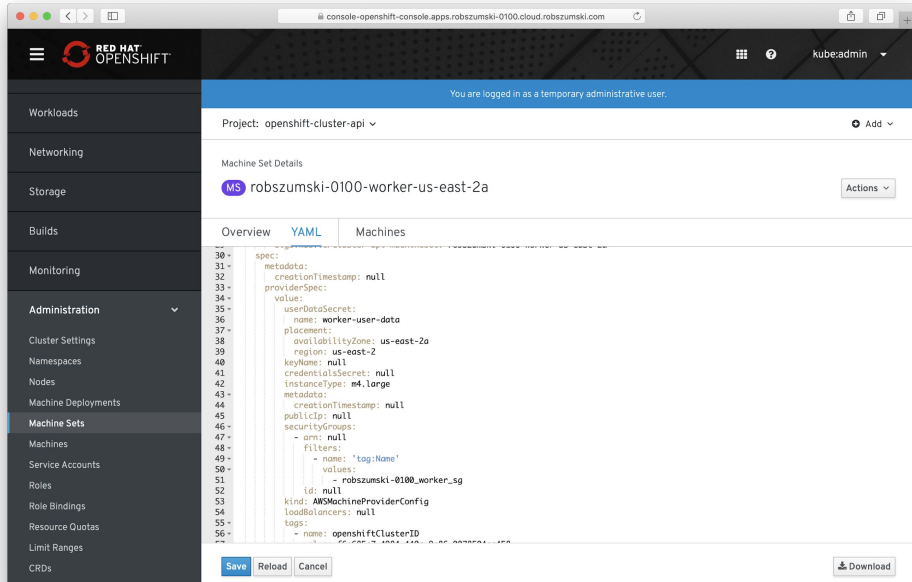
Using Kubernetes To Provision And Scale Clusters



Project: openshift-cluster-api

Machines

NAME	NAMESPACE	REGION	AVAILABILITY_ZONE
robszumski-0100-master-0	openshift-cluster-api	us-east-2	us-east-2a
robszumski-0100-master-1	openshift-cluster-api	us-east-2	us-east-2b
robszumski-0100-master-2	openshift-cluster-api	us-east-2	us-east-2c
robszumski-0100-worker-us-east-2a-86wfh	openshift-cluster-api	us-east-2	us-east-2a
robszumski-0100-worker-us-east-2b-sp8wx	openshift-cluster-api	us-east-2	us-east-2b
robszumski-0100-worker-us-east-2c-vjfw	openshift-cluster-api	us-east-2	us-east-2c



Project: openshift-cluster-api

Machine Set Details

robszumski-0100-worker-us-east-2a

Overview | **YAML** | Machines

```
spec:
  metadata:
    creationTimestamp: null
  providerSpec:
    value:
      userDataSecret:
        name: worker-user-data
      placement:
        availabilityZone: us-east-2a
        region: us-east-2
      keyName: null
      credentialsSecret: null
      instanceType: m4.large
      metadata:
        creationTimestamp: null
      publicIp: null
      securityGroups:
        - name: null
      filters:
        - name: 'tag:Name'
          values:
            - robszumski-0100_worker_sg
      id: null
    kind: AWSMachineProviderConfig
  loadBalancer: null
  tags:
    - name: openshiftClusterID
```

Cluster Logging

Cluster Logging is lifecycle managed via Operator Lifecycle Management

- Install the Elasticsearch and Cluster Logging Operators from OperatorHub
- Create an instance of Cluster Logging. fluentd, Elasticsearch and Kibana (with Operators) are created
- Changing the out-of-box configuration:
 - CPU, memory requests and limits, PVC sizes etc can be changed by editing the Cluster Logging Operator YAML
- Direct Elasticsearch and Kibana Deployments to dedicated Nodes (recommended for production usage)

Create Operator Subscription

Keep your service up to date by selecting a channel and approval strategy. The strategy determines either manual or automatic updates.

Installation Mode *

All namespaces on the cluster
This mode is not available for this operator

A specific namespace on the cluster
Operator will be available in a single namespace only.

A specific namespace on the cluster
Operator will be available in a single namespace only.

A specific namespace on the cluster
Operator will be available in a single namespace only.

PR openshift-logging

Update Channel *

preview

Approval Strategy *

Automatic

Manual

Subscribe Cancel

```
# configure via CRD
apiVersion: "logging.openshift.io/v1"
kind: "ClusterLogging"
metadata:
  name: "instance"
  namespace: "openshift-logging"
spec:
  managementState: "Managed"
  logStore:
    type: "elasticsearch"
    elasticsearch:
      nodeCount: 3
      resources:
        limits:
          cpu: 800m
          memory: 1Gi
          requests:
            cpu: 800m
            memory: 1Gi
      storage:
        storageClassName: gp2
        size: 100G
        redundancypolicy: "SingleRedundancy"
    visualization:
      type: "kibana"
      kibana:
        replicas: 1
```

Generally Available

Cluster Monitoring

Cluster monitoring is installed by default

- Exposes resource metrics for Horizontal Pod Autoscaling (HPA) by default
 - HPA based on custom metric is tech preview
- No manual etcd monitoring configuration anymore
- New screens for managing Alerts & Silences
- More metrics available for troubleshooting purposes (e.g. HAproxy)
- Configuration via ConfigMaps and Secrets

Alerts [Alertmanager UI](#)

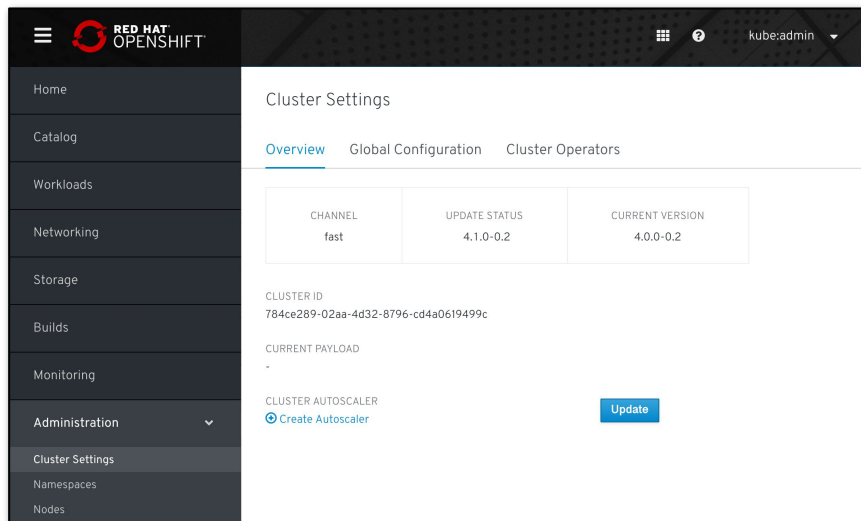
Alerts help notify you when certain conditions in your environment are met. [Learn more about how alerts are](#)

12 Firing 0 Silenced 0 Pending 77 Not Firing [Select All Filters](#)

NAME ↑	STATE
AL CPUThrottlingHigh 39% throttling of CPU in namespace metering-demo for container tiller in pod metering-operator-5c9c754b85-19ds2.	Firing Since Apr 29, 11:52
AL CPUThrottlingHigh 28% throttling of CPU in namespace metering-demo for container reporting-operator in pod reporting-operator-6c666b88db-qvbb5.	Firing Since May 2, 6:47 a
AL CPUThrottlingHigh 81% throttling of CPU in namespace metering-demo for container metering-operator in pod metering-operator-5c9c754b85-19ds2.	Firing Since Apr 29, 11:52
AL KubeDeploymentReplicasMismatch Deployment openshift-operators/mongodb-enterprise-operator has not matched the expected number of replicas for longer than an hour.	Firing Since May 2, 1:34 p
AL KubePodCrashLooping Pod openshift-operators/mongodb-enterprise-operator-7b6954d84d-g69b4 (mongodb-enterprise-operator) is crashing 0.02 times / 1.5 minutes	Firing Since Apr 29, 2:52

Over-the-air Updates

- OpenShift retrieves list of available updates
- Admin selects the target version
- OpenShift is updated over the air
- Auto-update support



Smarter Software Updates

No downtime for well behaving apps

Applications with multiple replicas, using liveness probes, health checks and taints/tolerations
Node Pools with more than one worker and slack resources

Maintenance window for entire cluster

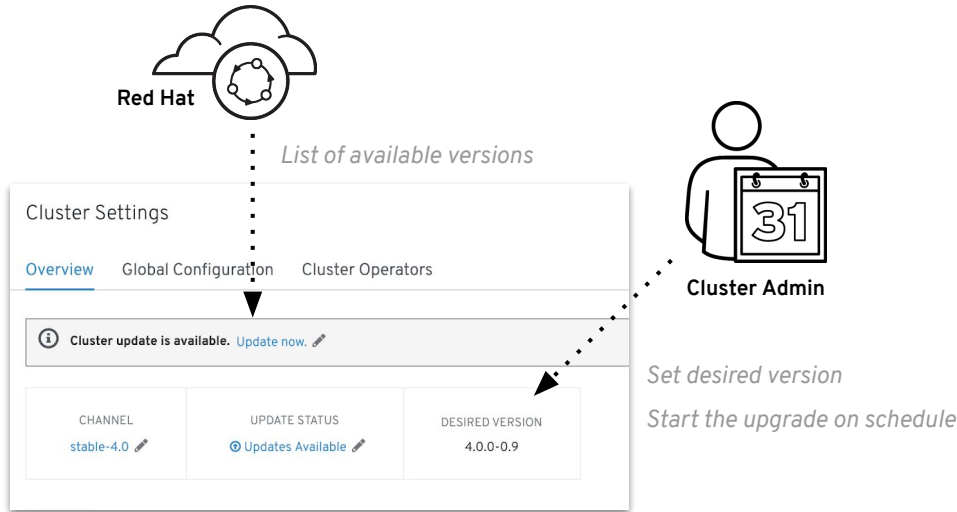
No need for separate windows for each component

Upgrade runs completely on the cluster

No more long running processes on a workstation

Constant health checking from each Operator

Operators are constantly looking for incompatibilities and issues that might arise



Openshift Upgrades

Orchestrating node upgrades

Machine Config Controller

Coordinates upgrade of machines

Machine Config Daemon

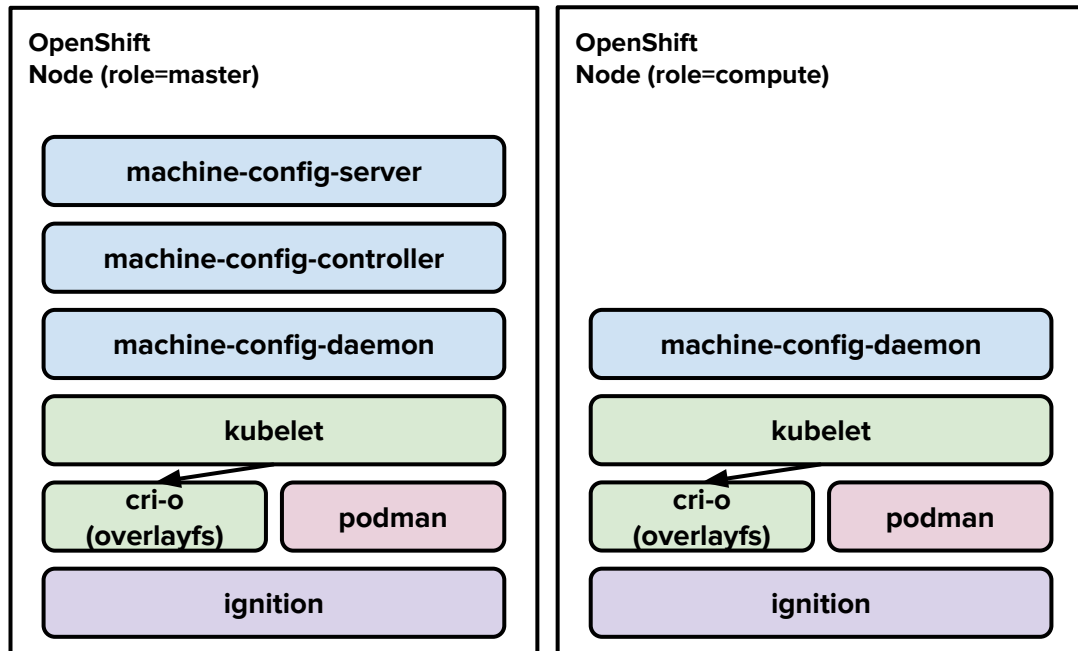
Applies desired machine configuration

Subset of ignition config

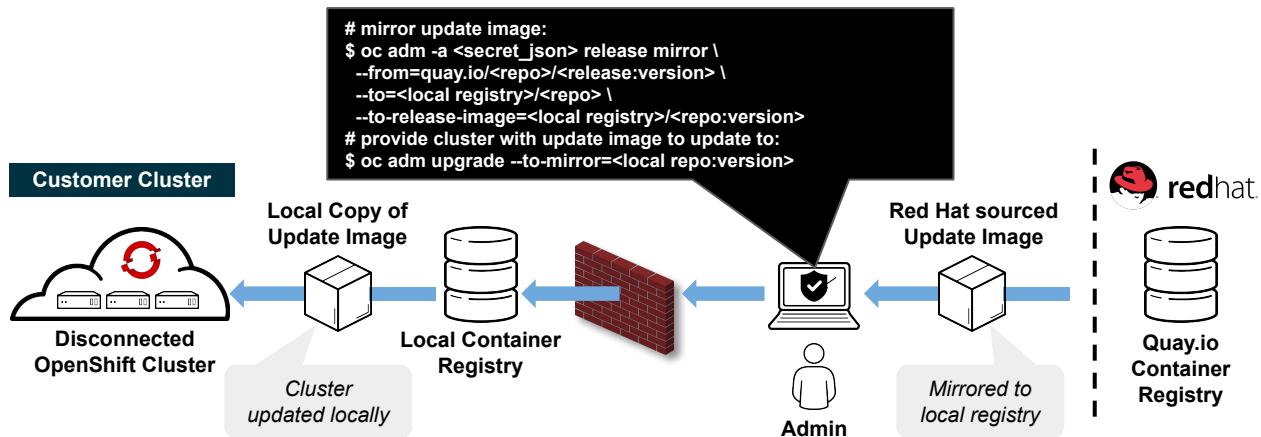
Controls kubelet configuration

Machine Config Server

Provide ignition config to new machines



Disconnected "Air-gapped" Install & Upgrade



Overview

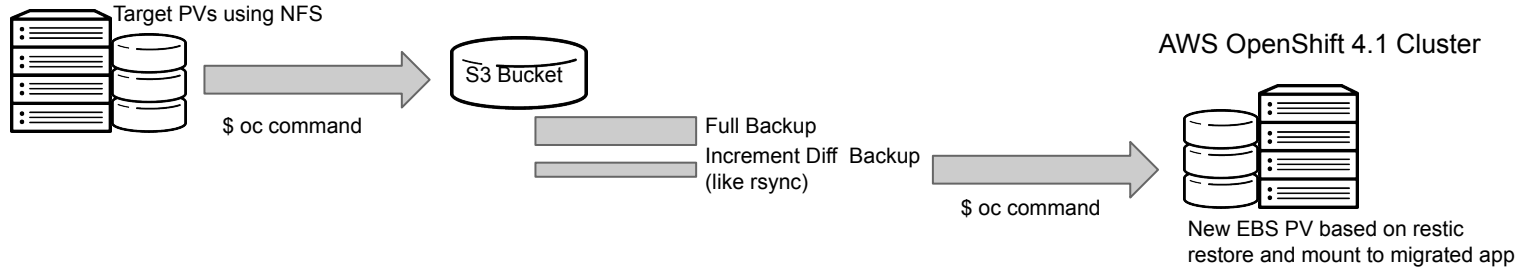
- 4.2 introduces support for installing and updating OpenShift clusters in disconnected environments
- Requires local Docker 2.2 spec compliant container registry to host OpenShift content
- Designed to work with the user provisioned infrastructure deployment method
 - *Note: Will not work with Installer provisioned infrastructure deployments*

Installation Procedure

- Mirror OpenShift content to local container registry in the disconnected environment
- Generate install-config.yaml: `./openshift-install create install-config --dir <dir>`
 - Edit and add pull secret (PullSecret), CA certificate (AdditionalTrustBundle), and image content sources (ImageContentSources) to install-config.yaml
- Set the `OPENSHIFT_INSTALL_RELEASE_IMAGE_OVERRIDE` environment variable during the creation of the ignition configs
- Generate the ignition configuration: `./openshift-install create ignition-configs --dir <dir>`
- Use the resulting ignition files to bootstrap the cluster deployment

Cluster Migration OpenShift 3 To 4

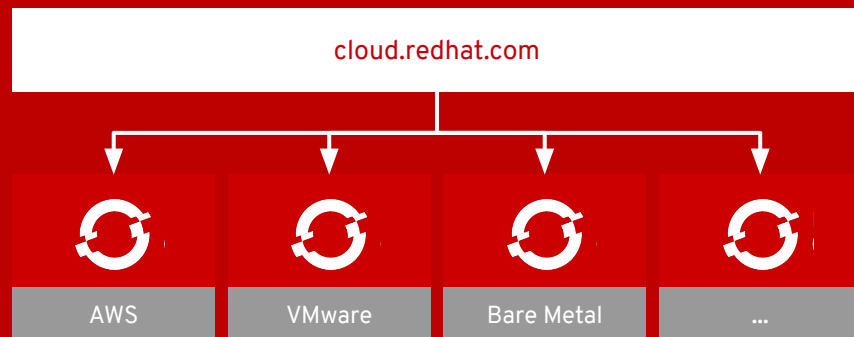
vSphere OpenShift 3.10 Cluster



- Deploy a replication of your applications from one OpenShift cluster to a different OpenShift cluster
- Enable cluster specific configuration from OpenShift 3 to work on a OpenShift 4 cluster
- Documentation on how to handle common network, storage, and machine/node re-use scenarios between OpenShift 3 and OpenShift 4 clusters

Cloud-like Simplicity, Everywhere

Full-stack automated operations across any on-premises,
cloud, or hybrid infrastructure



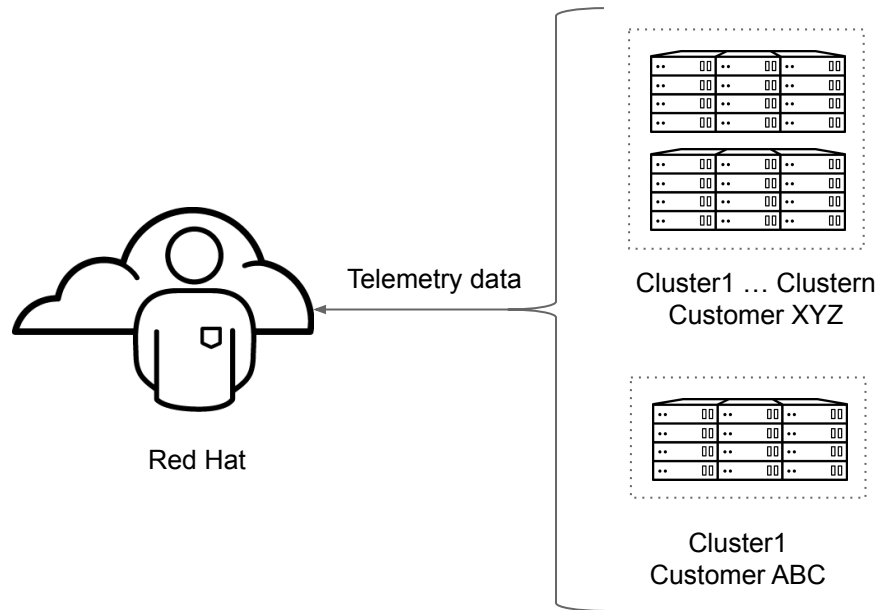
Telemetry

Collects anonymized data from any OpenShift 4 cluster deployment

- Red Hat gains quality assurance with anonymous data reporting faults encountered during upgrade
- Show utilization of all your clusters at cloud.redhat.com
- Perform subscription management at cloud.redhat.com

Opt-out is only available for self-managed OpenShift clusters but we strongly discourage that as you will lose all of the features described above.

[Complete list of collected metrics](#)



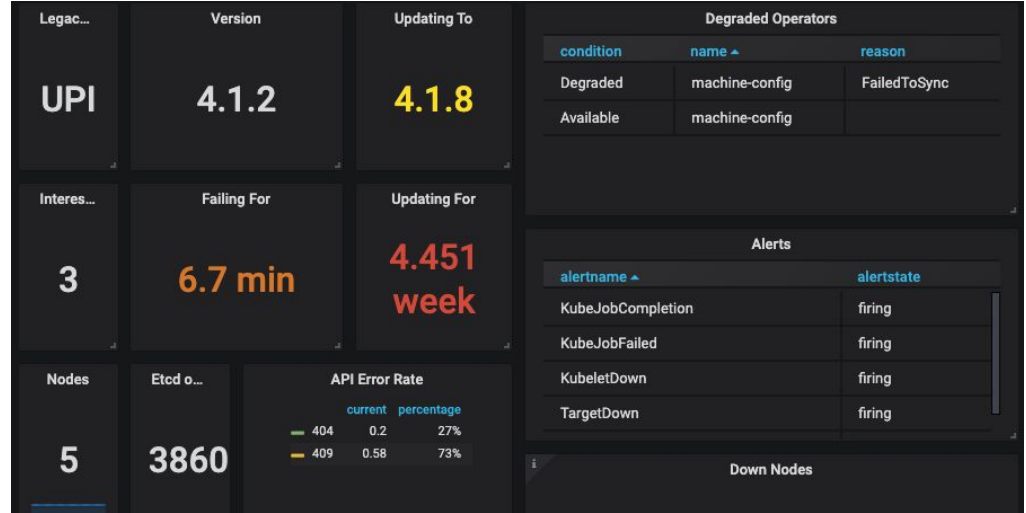
Connected Customer

Proactive support for customer issues

- Active upgrades
- Overall cluster health
- Firing alerts
- Node health

Driving a high quality product

- Monitor and improve upon the health of the customer base
- Prioritize engineering roadmap for platforms and prove they are improving over time
- Active monitoring of fast and stable channels



OpenShift Cluster Manager on cloud.redhat.com

Automatic registration of OpenShift clusters

View cluster versions and capacity in one place, no matter what infrastructure you are running on. Integrated with RHSM.

OpenShift Dedicated cluster management

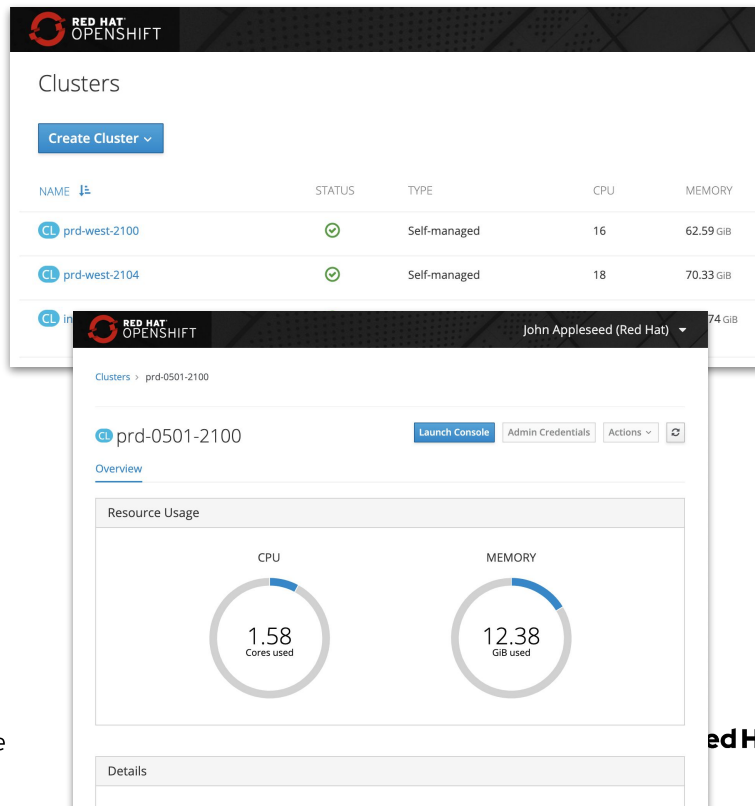
Self-service cluster deployment, scaling, and management for OpenShift Dedicated coming soon.

Azure Red Hat OpenShift

Information about these clusters will be coming at a later date.

Hosted in the United States

Other geographies may come later. You can [opt-out](#) too.



OpenShift Subscription Management

Moves from node management to cluster management

Entitle clusters and not nodes. Nodes too dynamic. We do not block on usage. Requires telemeter opt-in.

Dynamically adds and removes nodes

OpenShift Cluster Manager (OCM) will dynamically add and remove nodes from your subscription allocations to the cluster in 24 hour intervals. This will move to instantaneous across the next several releases.

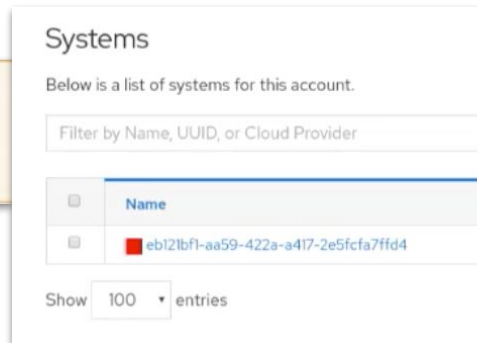


Connected to the same backend as Subscription Portal and Satellite

Allocation numbers you see at cloud.redhat.com for OCP can be also seen on the subscription portal at access.redhat.com

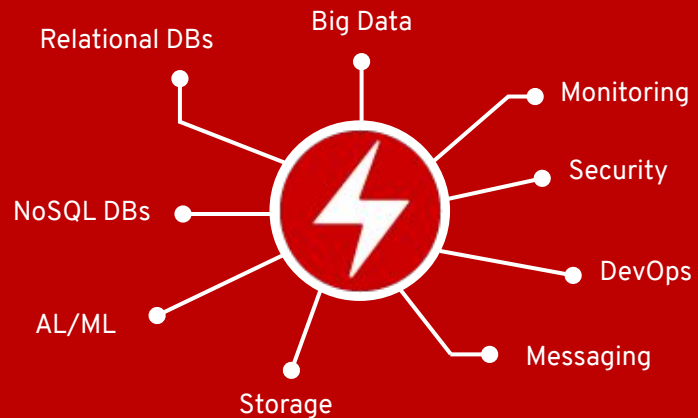
Removes OCP Infrastructure from the count

OCM will figure out which pods are your OCP infra pods and subtract out their usage from your core count so you are not charged.




A broad ecosystem of workloads

Operator-backed services allow for a SaaS experience on your own infrastructure




Operators – Simplify Complexity


Openshift Gains New Capabilities Every Day



**OPENSIFT
PLATFORM**



**Red Hat
Enterprise Linux
CoreOS**



DEVOPS  dynatrace

APM  APPDYNAMICS **INSTANA**  New Relic.  **Sysdig**

DATA SERVICES  GIGASPACEs  hazelcast  PlanetScale

DATABASE  Couchbase  mongoDB.  nuODB  PingCAP

SECURITY  aqua **anchore** **BLACKDUCK**
BY SYNOPSYS  TREMOLO
SECURITY **tufin**

STORAGE  ROBIN  STORAGEOS

AND MANY MORE TO COME...

The Value Of Kubernetes Operators

No need for operator

Requires custom Operator built with SDK



Installation

Automated application provisioning and configuration management

Upgrades

Patch and minor version upgrades supported

Lifecycle

App lifecycle, storage lifecycle (backup, failure recovery)

Deep Insights

Metrics, alerts, log processing and workload analysis

Auto-pilot

Horizontal/vertical scaling, auto config tuning, abnormal detection, scheduling tuning...



OperatorHub In OpenShift 4

For Cluster Admins:

OperatorHub

Discover Operators from the Kubernetes community and optional add-ons and shared services to your developers, providing a self-service experience.

All Items

AI/Machine Learning

Big Data

BigData

Database

Integration & Delivery

Logging & Tracing

Monitoring

All Items

34 items

AMQ Streams
provided by Red Hat, Inc.

Aqua Security
provided by Aqua Security, Inc.

AMQ Streams
1.1.0 provided by Red Hat, Inc.

Install

Red Hat AMQ Streams based on the Apache microservices and latency.

OPERATOR VERSION
1.1.0

PROVIDER TYPE
The core capability

- Discovery/install/upgrade of Operators
- Community, Red Hat products, Certified ISVs
- Granular access via specific Projects

For Developers:

Developer Catalog

Add shared apps, services, or source-to-image builders to your project show up here automatically.

All Items

Languages

Middleware

Other

Kafka

TYPE

- Service Class (0)
- Source-to-Image (0)
- Installed Operators (6)

Kafka
provided by Red Hat, Inc.
Represents a Kafka cluster

Kafka Connect
provided by Red Hat, Inc.
Represents a Kafka Connect cluster

amqstreams.v1.1.0 > Kafka Details

my-cluster

Overview YAML Resources

0 Route 4 Service 2 StatefulSet 1 Deployment 1 ReplicaSet

NAME	TYPE	STATUS
my-cluster-entity-operator	Deployment	Created
my-cluster-entity-operator-5778f899c-lddlz	Pod	Running
my-cluster-kafka	StatefulSet	Created
my-cluster-kafka-0	Pod	Running

- Developers can't see admin screens
- Operator capabilities are exposed in Catalog
- Self-service management

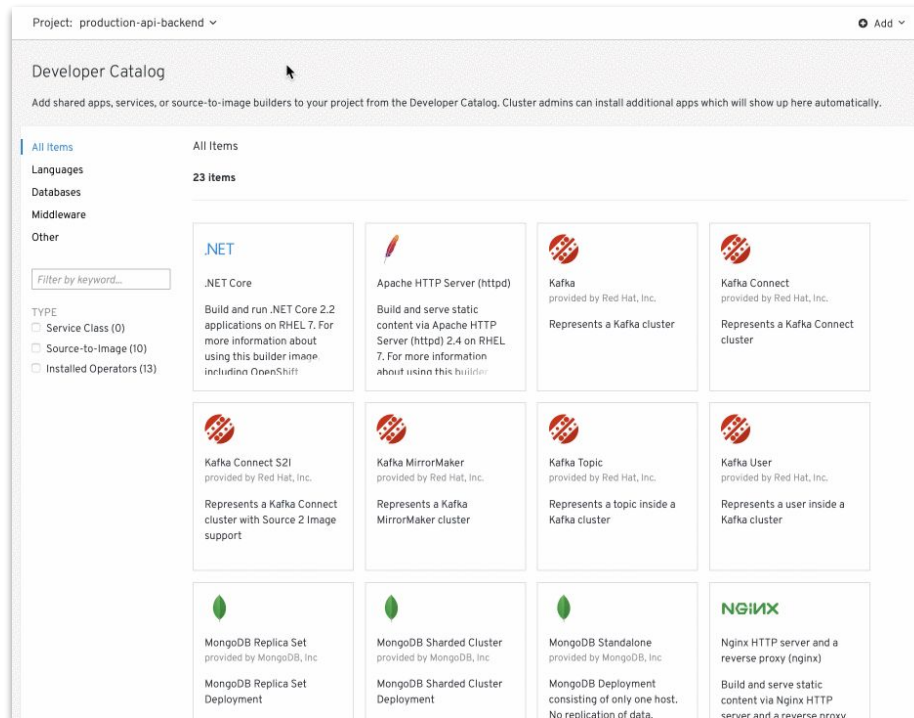
Services Ready For Your Developers

New Developer Catalog aggregates apps

- Blended view of Operators, Templates and Broker backed services
- Operators can expose multiple CRDs. Example:
 - MongoDBReplicaSet
 - MongoDBSharded Cluster
 - MongoDBStandalone
- Developers can't see any of the admin screens

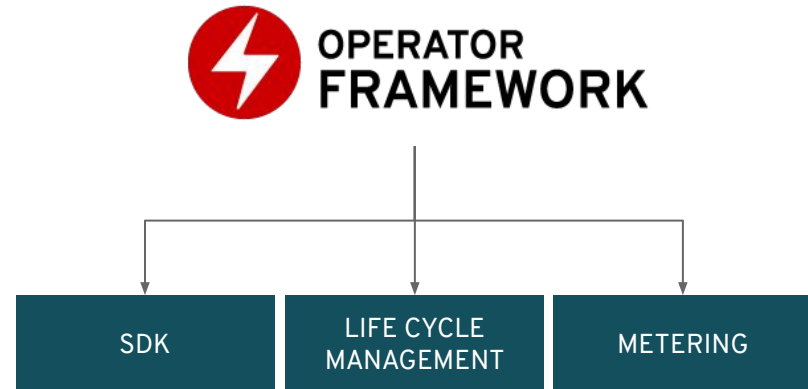
Self-service is key for productivity

- Developers with access can change settings and test out new services at any time

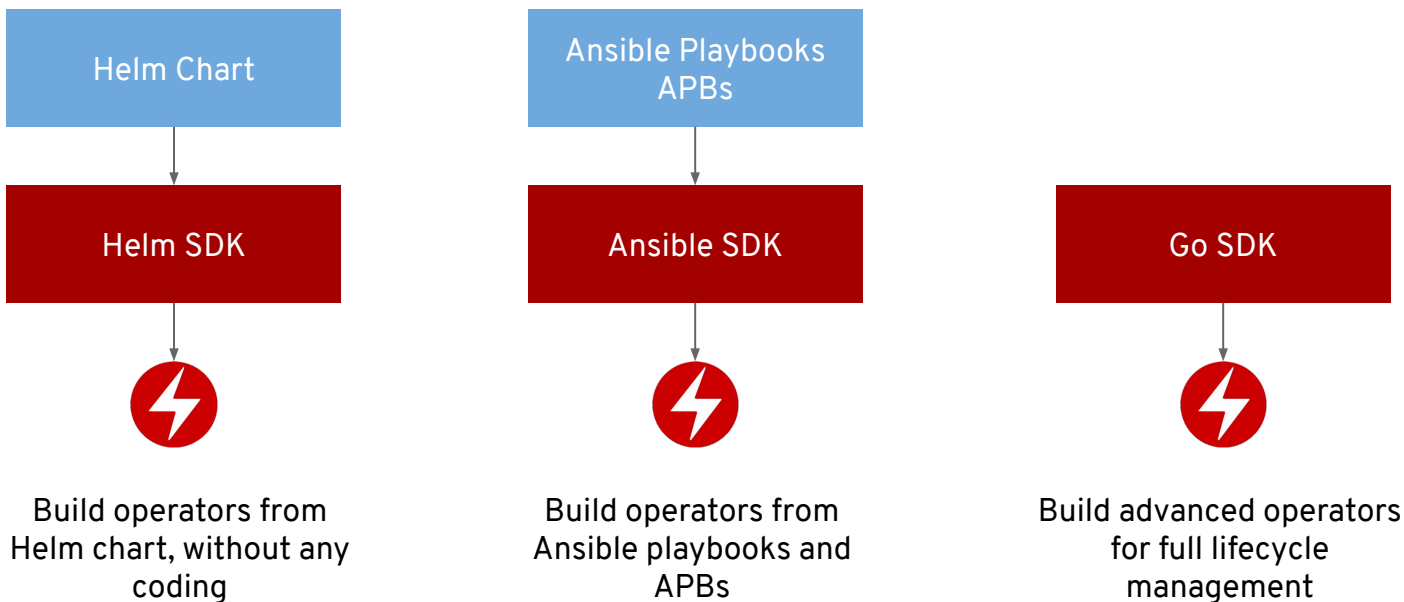


Operator Framework

Operators codify operational knowledge and workflows to automate life cycle management of containerized applications with Kubernetes



Build Operators For Your Apps



Red Hat Operator SDK as of 4.3

Operator Testing (scorecard v2)

- Operator tests now categorized as required/optional
- Configurable test selection and pass/fail behavior
- Ship Custom tests

Framework Integration

- Single command to deploy OLM:
`operator-sdk alpha [install|uninstall|status] olm`

Ansible-based Operator

- Support for Prometheus Metrics
- Uses UBI base-image
- Molecule-based e2e testing

Helm-based Operator

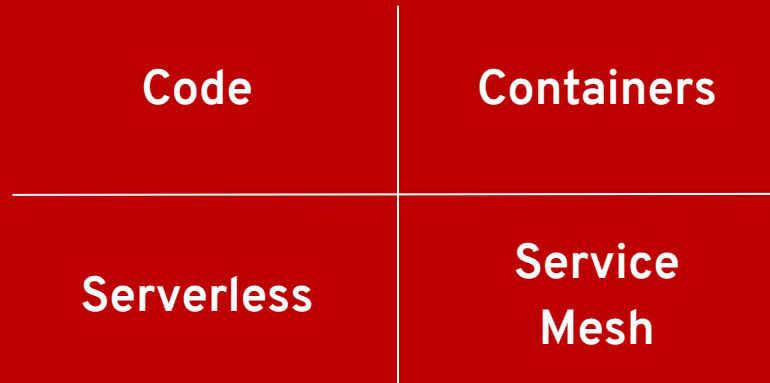
- Helm v3 support (starting SDK 0.14.0)
- SDK automatically generates RBAC for your chart

Golang-based Operator

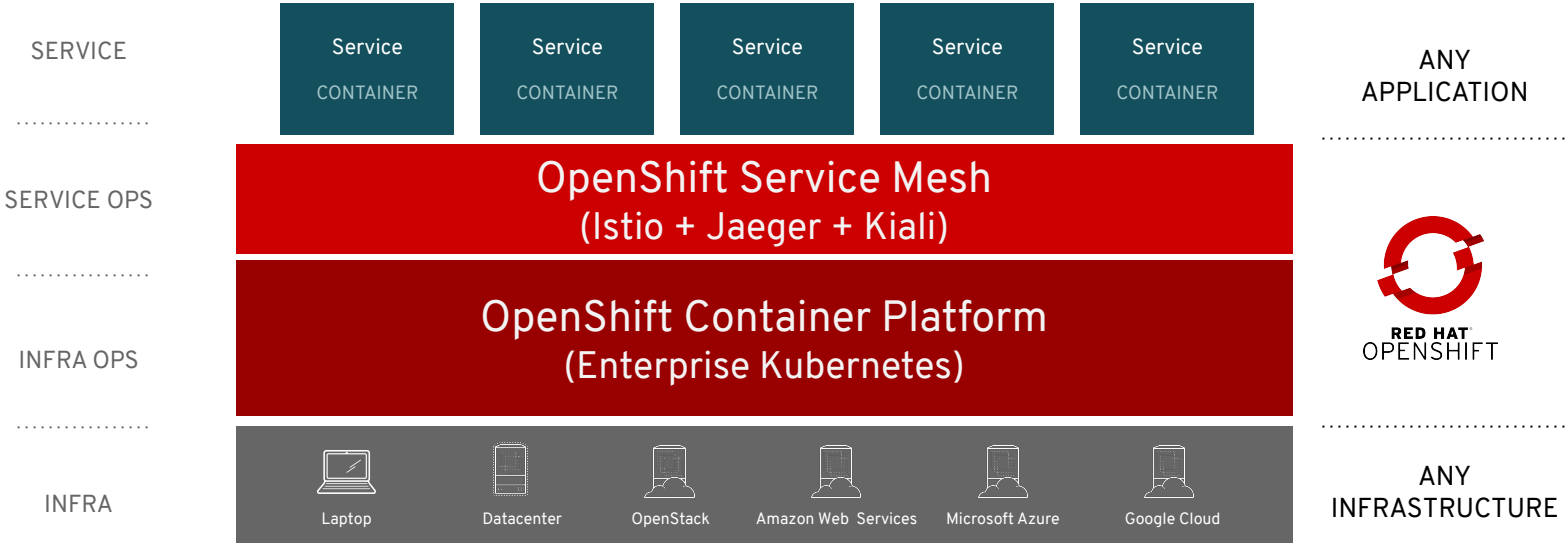
- Generate OpenAPI spec
- Supporting Kubernetes 1.14
- Go module support
- Controller-runtime 0.2.0
- Support for Prometheus Metrics
- Upstreamed Operator SDK features into k8s

Next wave of developer tools

OpenShift has all of the latest tools to make
your devs more productive



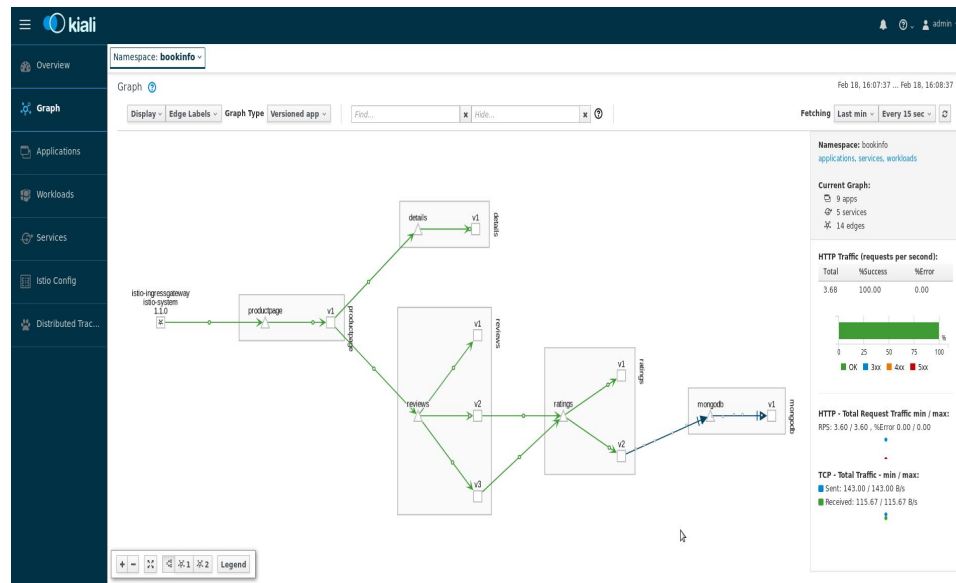
Red Hat Service Mesh



Red Hat Service Mesh

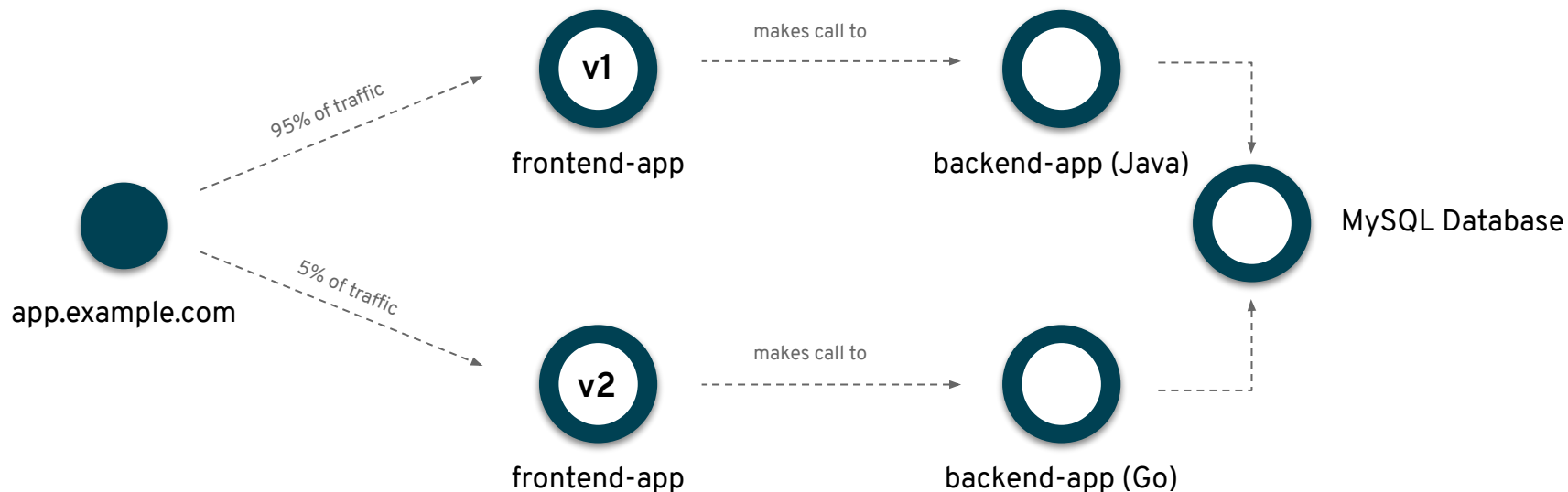
Key Features

- A dedicated network for service to service communications
- Observability and distributed tracing
- Policy-driven security
- Routing rules & chaos engineering
- Powerful visualization & monitoring
- Will be available via OperatorHub












Control Traffic Flow

Control flow of traffic between application components



Guided Configuration of Traffic Policies


Create Weighted Routing ✕

WORKLOAD		TRAFFIC WEIGHT
 reviews-v1		- 5 + % 
 reviews-v2		- 80 + % 
 reviews-v3		- 15 + % 

▼ [Hide Advanced Options](#)

VirtualService Hosts

The destination hosts to which traffic is being sent. Enter one or multiple hosts separated by comma.

 TLS

Add LoadBalancer

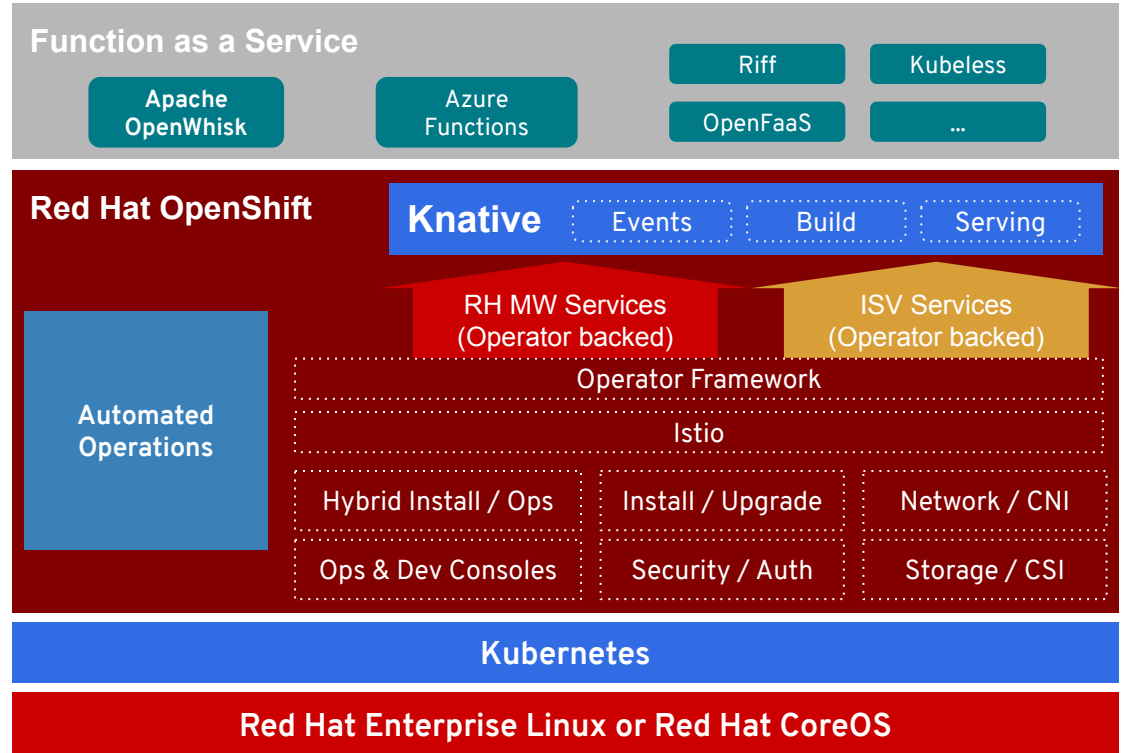
Add Gateway

Red Hat OpenShift And Serverless

Developer experience
APIs, CLI, service binding

Building blocks for serverless
Source-centric and container-based

The leading enterprise Kubernetes platform
Automated Operations
Build an run anywhere (Hybrid Cloud)



OpenShift Serverless + Azure Functions

Functions as a Service



Building blocks for
Containers &
Microservices

OpenShift serverless

Serving

Eventing

Kubernetes

OpenShift

OpenShift Serverless in 4.3

Key features and updates

- **Serverless Operator v1.3.0**
- **Knative v0.10**
- **OLM dependency resolution for Service Mesh**
- **Dropped support for Kubernetes 1.14 (OCP 4.1)**

Learn more

<https://openshift.com/learn/topics/serverless>

[Knative Tutorial](#)

The screenshot displays the OpenShift Serverless Operator interface. At the top, it shows 'Installed Operators > Operator Details' for the 'OpenShift Serverless Operator' (version 1.3.0) provided by Red Hat, Inc. Below this, there are tabs for 'Overview', 'YAML', 'Events', and 'Knative Serving'. The main view shows a deployment overview for 'spring-petclinic-bchpw-deployment' with a large circular progress indicator indicating '4 scaling to 10'. A detailed table on the right provides configuration parameters for the deployment.

Parameter	Value
Name	spring-petclinic-bchpw-deployment
Update Strategy	RollingUpdate
Namespace	markito-rhte
Max Unavailable	25% of 10 pods
Labels	app=spring-petclinic-bchpw, app.kubernetes.io/...=spring-petclinic, app.kubernetes.io/...=spring-petclinic, serving.knative.dev/configuration=...
Max Surge	25% greater than 10 pods
Progress Deadline	2m0s
Min Ready Seconds	Not Configured

OpenShift Serverless in 4.3

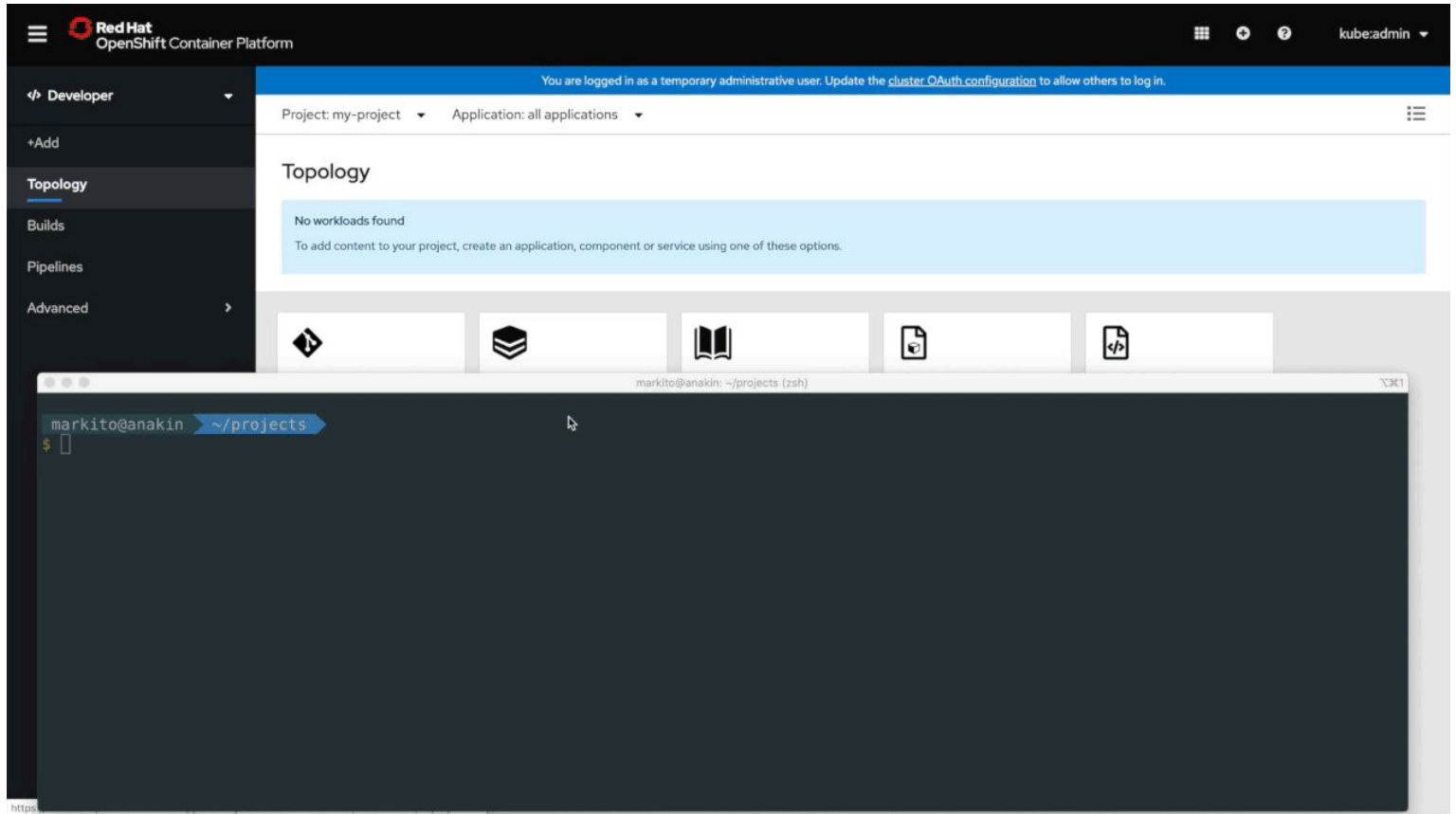
Traffic Split for Revisions

The screenshot displays the OpenShift Serverless console interface for a service named 'store-app'. On the left, a visual diagram shows two revision icons, each with a 'K' icon and a circular arrow. Below the icons are labels for revisions: 'store-app-lcvcb...' and 'store-app-bbggc...'. A blue box highlights these two revisions, with dashed arrows pointing to them from a central point, each labeled '50%', indicating a 50/50 traffic split. Below the diagram are search and zoom controls. On the right, the console details for 'store-app' are shown, including tabs for 'Overview' and 'Resources'. The 'Revisions' section is active, showing a table with two revisions: 'store-app-bbggc-1' and 'store-app-lcvcb-2', both with a 50% traffic distribution. A 'Set Traffic Distribution' button is highlighted in blue. Below the revisions, the 'Routes' section shows a route for 'store-app'.

Revision	Traffic Distribution
store-app-bbggc-1	50%
store-app-lcvcb-2	50%

OpenShift Serverless in 4.3

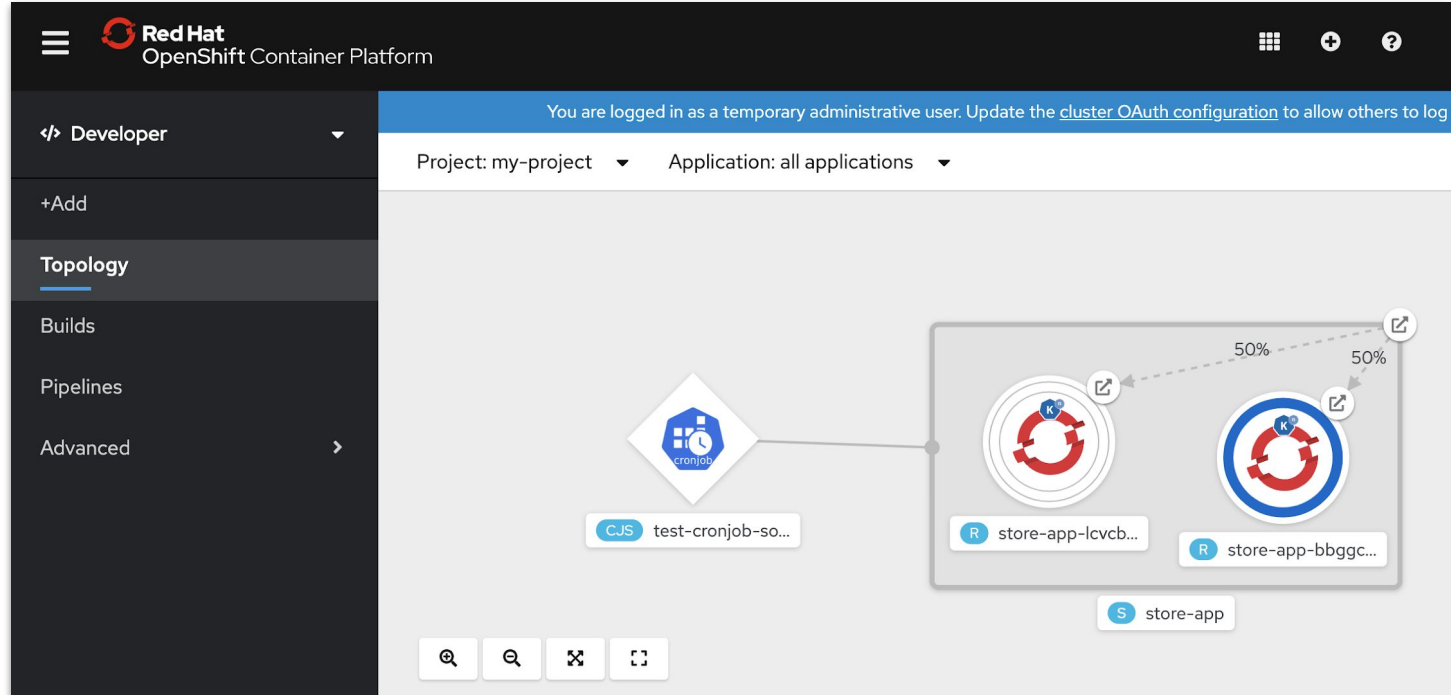
CLI & UI
Integration



OpenShift Serverless in 4.3

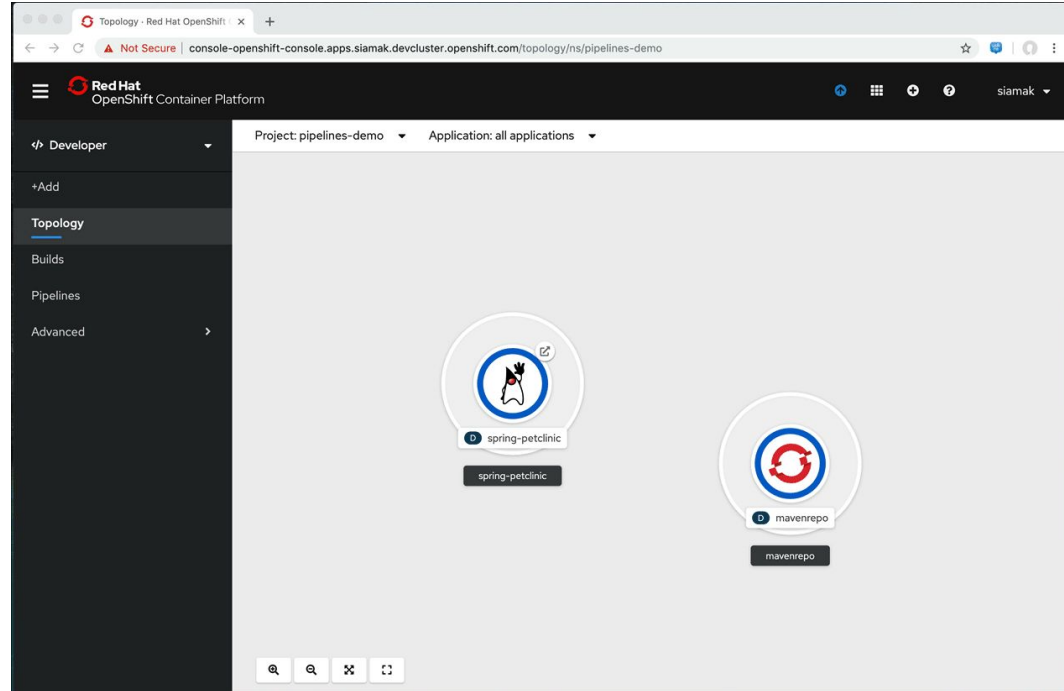
Event Sources

- KafkaSource
- CamelSource
- CronJobSource
- ContainerSource
- ApiServerSource



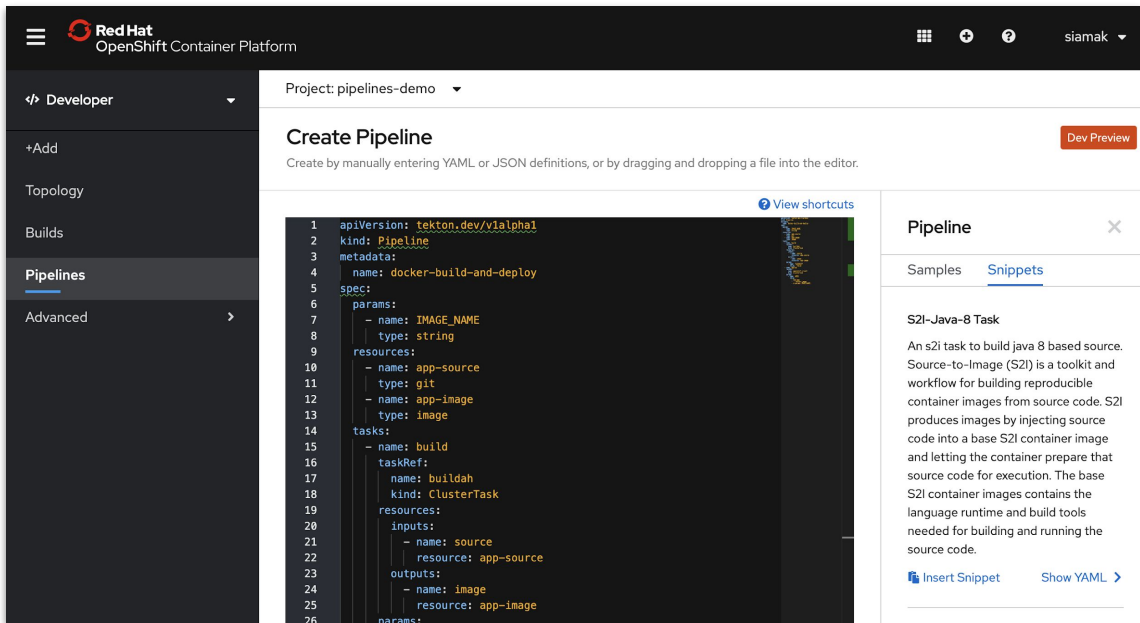
Cloud-native CI/CD with OpenShift Pipelines

- Based on Tekton Pipelines
- Runs serverless (no CI engine!)
- Containers as building blocks
- Build images with Kubernetes tools (s2i, buildah, kaniko, jib, buildpack, etc)
- Pipelines portable to any Kubernetes
- Available in OperatorHub
- Tekton CLI



OpenShift Pipelines in OCP 4.3

- Git triggers (webhook)
- Automated RBAC setup
- Default curated tasks
- Pipeline metrics in Prometheus
- Pipeline samples and Task ref snippets in YAML editor



The screenshot shows the OpenShift Pipelines console interface. The top navigation bar includes the Red Hat logo, 'OpenShift Container Platform', and a user profile 'siamak'. The left sidebar shows navigation options: Developer, +Add, Topology, Builds, Pipelines (selected), and Advanced. The main content area is titled 'Project: pipelines-demo' and 'Create Pipeline', with a 'Dev Preview' badge. Below the title, there's a description: 'Create by manually entering YAML or JSON definitions, or by dragging and dropping a file into the editor.' A 'View shortcuts' link is visible. The central part of the screen is a dark-themed code editor displaying a YAML pipeline definition:

```
1 apiVersion: tekton.dev/v1alpha1
2 kind: Pipeline
3 metadata:
4   name: docker-build-and-deploy
5 spec:
6   params:
7     - name: IMAGE_NAME
8       type: string
9   resources:
10    - name: app-source
11      type: git
12    - name: app-image
13      type: image
14   tasks:
15     - name: build
16       taskRef:
17         name: buildah
18         kind: ClusterTask
19       resources:
20         inputs:
21           - name: source
22             resource: app-source
23         outputs:
24           - name: image
25             resource: app-image
26   params:
```

On the right side, there's a 'Pipeline' panel with tabs for 'Samples' and 'Snippets'. Under 'Snippets', there's a section for 'S2I-Java-8 Task' with a description: 'An s2i task to build java 8 based source. Source-to-Image (S2I) is a toolkit and workflow for building reproducible container images from source code. S2I produces images by injecting source code into a base S2I container image and letting the container prepare that source code for execution. The base S2I container images contains the language runtime and build tools needed for building and running the source code.' There are also links for 'Insert Snippet' and 'Show YAML'.

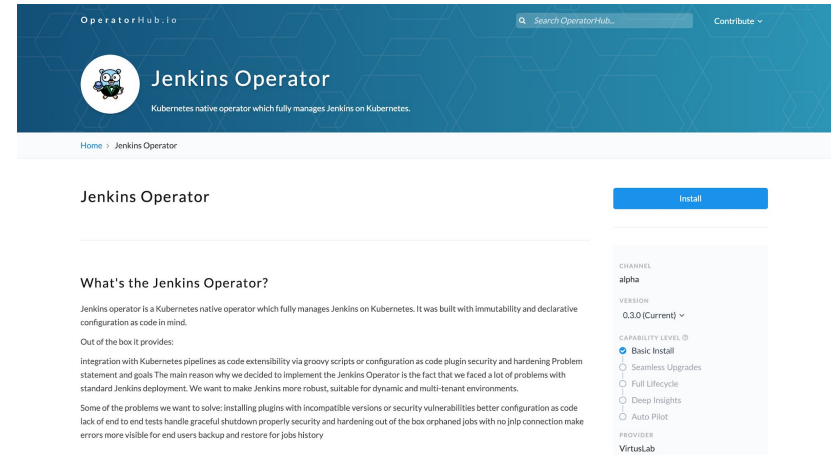
OpenShift Pipelines in OCP 4.3

- Default pipeline on app import (+Add) in Dev Console
- Pipeline objects in Admin Console
- New Tekton CLI commands
 - Start pipelines
 - Start tasks
 - Create resources

The screenshot displays the OpenShift Developer Console interface. On the left, the navigation menu is open, showing the 'Pipelines' section under the 'Developer' role. The main content area shows the configuration for a pipeline named 'Red Hat OpenJDK 11'. The pipeline is currently in a 'Dev Preview' state. The configuration includes a dropdown for 'Application' set to 'Triggers', a 'Name' field, and a checkbox for 'Add pipeline' which is checked. Below the configuration, there is a visual representation of the pipeline with two stages: 'build' and 'deploy'.

Jenkins

- Jenkins server on JDK 8 & 11
- Jenkins agents
 - JDK 11
 - Node.js 10
- Official Jenkins Operator
 - github.com/jenkinsci/kubernetes-operator
 - Available in OperatorHub.io
 - Developer Preview on OCP 4.3
 - Collaboration upstream



The screenshot shows the Jenkins Operator page on OperatorHub.io. The header includes the OperatorHub.io logo, a search bar, and a 'Contribute' button. The main content area features the Jenkins Operator logo and the text 'Kubernetes native operator which fully manages Jenkins on Kubernetes.' Below this, there is a navigation breadcrumb 'Home > Jenkins Operator' and a prominent blue 'Install' button. The page content is divided into two columns. The left column, titled 'Jenkins Operator', contains a section 'What's the Jenkins Operator?' with a detailed description of the operator's purpose and goals, followed by a list of capabilities and a list of problems it aims to solve. The right column contains a sidebar with a 'CHANNEL' dropdown set to 'alpha', a 'VERSION' dropdown set to '0.3.0 (Current)', and a 'CAPABILITY LEVEL' dropdown with 'Basic Install' selected. Other options include 'Seamless Upgrades', 'Full Lifecycle', 'Deep Insights', and 'Auto Pilot'. The provider is listed as 'VirtusLab'.

CodeReady Containers: OpenShift on your Laptop

New in 4.3:

- Automatic certificate rotation for internal node<->master communication
- 4.3 embedded GA version targeted for February 4th
- Ongoing updates with 4.2 z-stream updates
- Deprecated: removed VirtualBox support
- crc version outputs embedded OCP version number
- Many stability fixes around host networking

Provides a pre-built development environment based on **Red Hat Enterprise Linux** and **OpenShift** for quick container-based application development. Use with OpenShift on-premises or cloud.

```
$ crc setup
Prepare your machine for running OpenShift

$ crc start
Start with the Hyperkit 4.3 bundle

$ crc status
Get the status of the cluster
```


OpenShift Console

The future is now.

**Extending the
Console**

**Administration
made easy**

**Improve
Observability**

**Developer
Focused**

Enhanced Visibility with the New Project Dashboard

Project-scope Dashboard gives Developer Clear Insights

Drill down in context from the new project dashboard widgets:

- Project Details
- Project Status/Health
- Project External Links (Launcher)
- Project Inventory
- Project Utilization
- Project Resource Quota
- Project Activity (Top consumers)

The screenshot shows the Red Hat OpenShift Container Platform interface. The top navigation bar includes the Red Hat logo, 'OpenShift Container Platform', and the user 'kube:admin'. The left sidebar contains a navigation menu with categories like Administrator, Home, Projects, Search, Operators, Workloads, Networking, Storage, Builds, Monitoring, Compute, User Management, and Administration. The main content area is titled 'Projects > Project Details' and shows the project 'tony' with a 'PR' icon and 'Active' status. Below this, there are tabs for 'Dashboard', 'Overview', 'YAML', 'Workloads', and 'Role Bindings'. The 'Dashboard' tab is active, displaying several widgets: 'Details' (Name: tony, Requester: kube:admin, Labels: No labels), 'Inventory' (4 Deployments, 4 Pods, 0 PVCs, 1 Service, 0 Routes, 4 Config Maps, 21 Secrets), 'Status' (Active), 'Utilization' (CPU: 8.39m, Memory: 96.31 MiB, Pod count: 4), and 'Activity' (Recent Events). A 'Launcher' widget is also visible, showing 'Service Mesh'.

Expose Third Party App Console for Operator-backed Services

“Cluster-wide” ConsoleLink CRD

- Easily integrate/onboard **cluster-wide** third-party user interfaces to develop, administer, and configure Operator-backed services.

“Project-scoped” ConsoleLink CRD

- Customize the access to integrated **project-scoped** third-party user interfaces for your users.
- With the project-scoped external link launch mechanism, **link in context** to your interface.

This is an example notification message with an optional link. [Optional link text](#)

Red Hat
OpenShift Container Platform

Project: tony

Installed Operators

Installed Operators are represented by Cluster Service Versions within this namespace. For more information, see the [Operator Lifecycle Manager documentation](#). Or create an Operator and Cluster Service Version using the [Operator SDK](#).

Name	Namespace	Deployment	Status	Provided APIs
AMQ Streams	NS tony	amq-streams-	InstallSucceeded	Kafka

Red Hat
OpenShift Container Platform

Projects > Project Details

tony Active

Dashboard Overview YAML Workloads Role Bindings

Details View all

Name
tony

Requester
kube:admin

Labels
No labels

Status

Active

Launcher

Service Mesh

Activity View events

Ongoing

There are no ongoing activities.

Add YAML Samples for a specific resource

Educate your Users with an Easy Way to Understand Kubernetes Resources

- You can now add cluster-wide samples to any Kube Resource with **Console YAML Samples CRD**.
- Each team that manages kube resources owns their samples and should make it part of their Operator.
- Any Operators can add YAML samples including Third-Party ISVs

The screenshot shows the Red Hat OpenShift Container Platform console. The left sidebar is open to the 'Workloads' section. The main area displays the 'Create Job' page, which includes a YAML editor with the following content:

```

1  apiVersion: batch/v1
2  kind: Job
3  metadata:
4  name: example
5  namespace: brie
6  spec:
7  selector: {}
8  template:
9  metadata:
10  name: pi
11  spec:
12  containers:
13  - name: pi
14  image: perl
15  command:
16  - perl
17  - -Mbignum=bp1
18  - -wle
19  - print bp1(2000)
20  restartPolicy: Never
  
```

On the right side of the 'Create Job' page, there is a 'Job' tab with a 'Samples' sub-tab. Below the 'Samples' tab, there is a list of samples, including '1. Example Job' and 'An example Job YAML sample'. A 'Download YAML' button is visible next to the second sample.

Below the 'Create Job' page, there is a screenshot of the 'consoleyamlsamples.console.openshift.io' CRD page. The page shows the 'Overview' tab with a table of instances:

Name	Namespace	Created
example	None	2 minutes ago

View Security Vulnerabilities with the Quay Operator

See all your Container Vulnerabilities right from the Console Dashboard

- Link out to **Red Hat Quay** for more in depth information
- The Quay Operator supports both **On-premise and External** Quay Registries
- Currently uses **Clair for Security Scan**; Planning to expand to other Vendors(TwistLock, Aqua, e.g.)
- *Only works for images managed by Quay*

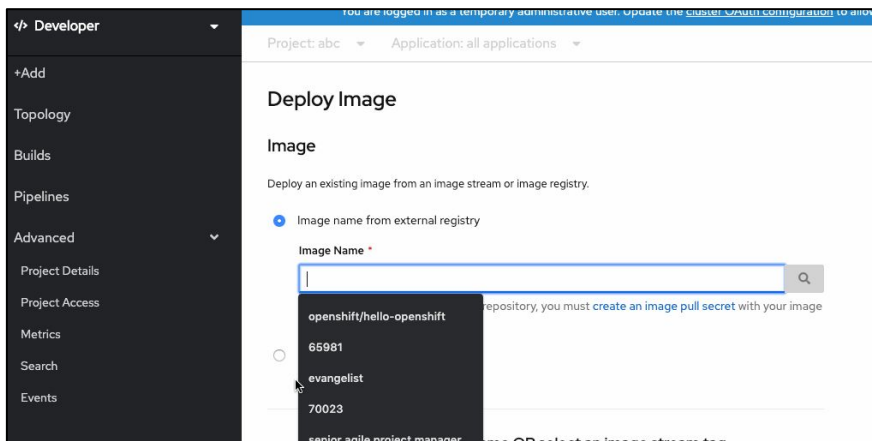
The screenshot shows the Red Hat OpenShift Container Platform console. The left sidebar contains navigation options like Administrator, Home, Dashboards, Projects, Search, Explore, Events, OperatorsHub, Installed Operators, Workloads, Pods, Deployments, Stateful Sets, Secrets, Config Maps, and Cron Jobs. The main content area displays a 'Dashboards' overview with a 'Security breakdown' pop-up window. The pop-up shows 'Quay analyzes container images to identify vulnerabilities.' and 'Severity Fixable' with a '1 total' indicator. Below the pop-up, a 'Quay Security Scanner' report indicates '61 vulnerabilities' detected, with 'Patches are available for 61 vulnerabilities.' The report includes a pie chart showing the distribution of severity levels: 14 High-level vulnerabilities (34%), 33 Medium-level vulnerabilities (54%), and 14 Low-level vulnerabilities (22%). A table lists the vulnerabilities with columns for CVE, Severity, Package, Current Version, Fixed in Version, and Introduced in Layer.

CVE	SEVERITY	PACKAGE	CURRENT VERSION	FIXED IN VERSION	INTRODUCED IN LAYER
RHSA-2019-0710	High	python-lib	2.7.5-48.el7	0.2.7.5-37.el7_6	
RHSA-2019-1587	High	python-lib	2.7.5-48.el7	0.2.7.5-40.el7_6	
RHSA-2019-0368	High	systemd-lib	219-57.el7	0.219-45.el7_6.5	
RHSA-2019-0049	High	systemd-lib	219-57.el7	0.219-45.el7_6.2	
RHSA-2019-0679	High	libssh2	1.4.3-10.el7_2.1	0.1.4.3-12.el7_6.2	
RHSA-2018-2285	High	yum-plugin-ovf	1.1.31-45.el7	0.1.1.31-46.el7_5	

Deploy Applications streamlining flows

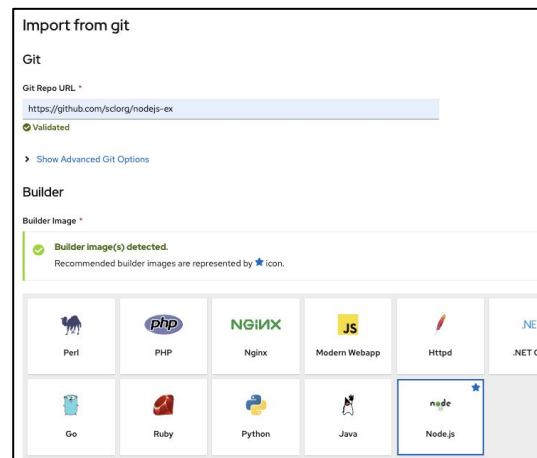
Deploy Image from Internal Registry

- Allow for rapidly deploying with alternate paths
- No need to repush/pull images



Auto-detect builder image

- Recommends builder images based on detected language by git provider



Deploy Applications alternate deployment targets

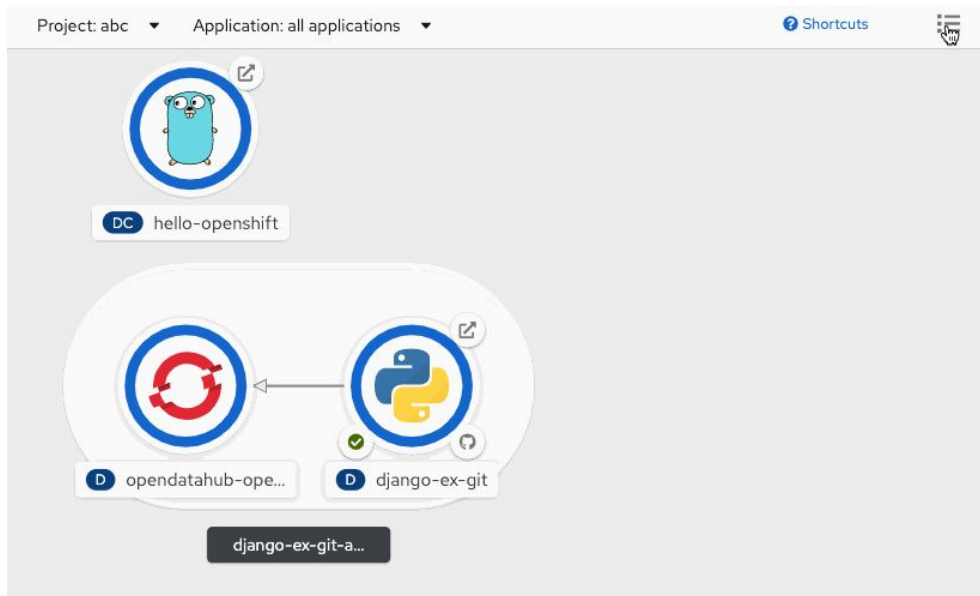
- Default to Kubernetes Deployments
- Alternately can use OpenShift's DeploymentConfigs or Knative Service (tech preview) objects
- Advanced options changes accordingly

The screenshot shows the Red Hat OpenShift Container Platform Developer console. The top navigation bar includes the Red Hat logo and the text "OpenShift Container Platform". A notification banner at the top right states: "You are logged in as a temporary administrative user. Update the [cluster OAuth configuration](#) to allow others to". Below the navigation bar, the left sidebar is expanded to the "Developer" section, showing options like "+Add", "Topology", "Builds", "Pipelines", "Advanced" (with a dropdown arrow), "Project Details", "Project Access", "Metrics", "Search", and "Events". The main content area displays "Project: abc" and "Application: all applications". A message box indicates: "There are no pipeline templates available for this runtime." Below this, the "Resources" section is visible, with the instruction "Select the resource type to generate". Three resource types are listed: "Deployment" (selected), "Deployment Config", and "Knative Service" (marked as "Tech Preview").

Application Topology

streamlined flows

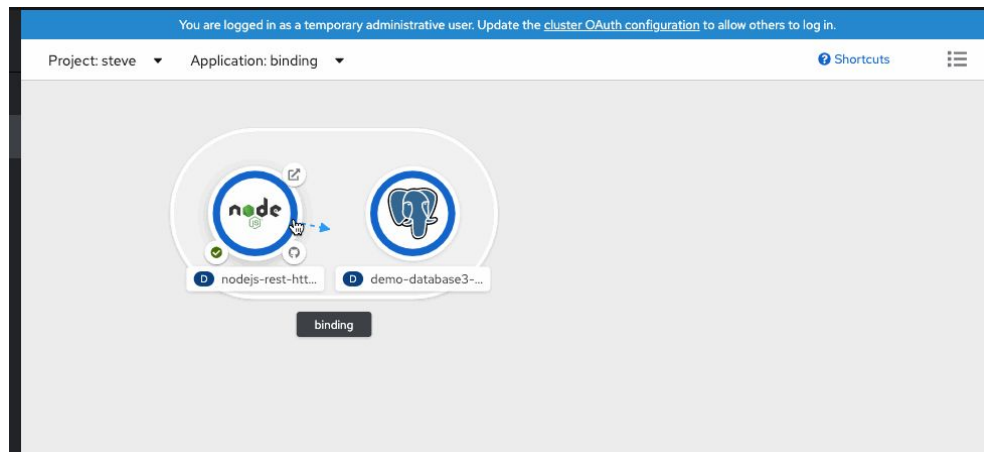
- Toggle between List and Topology views
- Easily group applications
- Connect/bind applications easily
- Contextual actions
- Quickly delete applications



Service Binding

easily connecting apps

- Leverages new ServiceBindingRequest and Operator to handle binding requests
- Easily create in Topology by dropping connector to valid drop target
- Injects config into source pod template as environment variables as a secret
- Pods are redeployed to pick up binding credentials

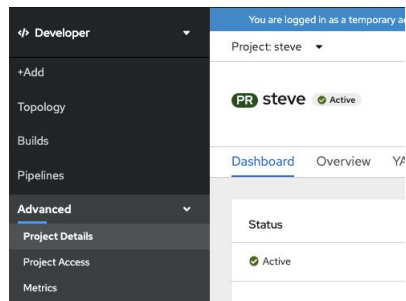


Learn more about service binding:
<https://github.com/redhat-developer/service-binding-operator>

Project Details & Access

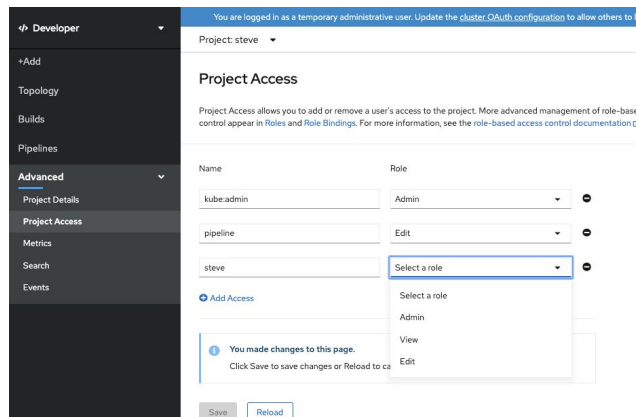
Project Details

- Quick access to current project details
- View dashboard for status and resource utilization
- Actions for edit or delete



Project Access

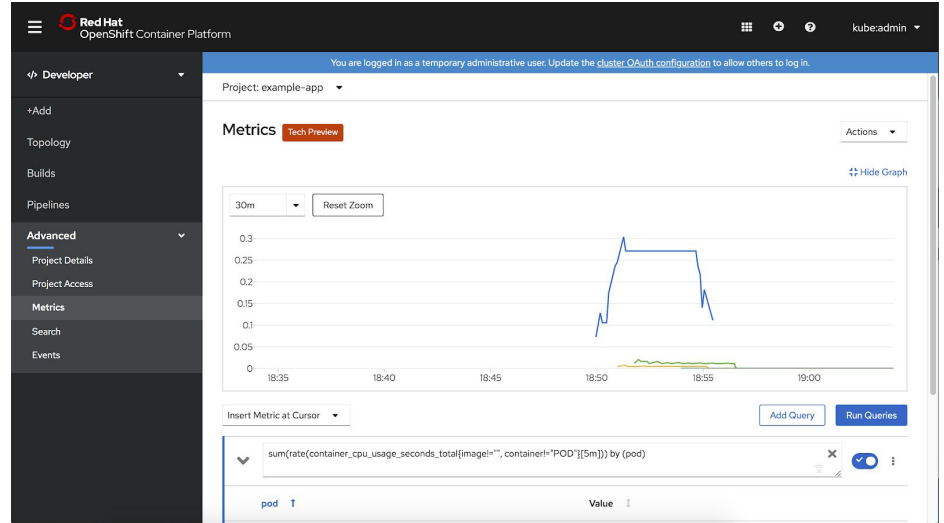
- Simplify sharing projects
- Reduces to a simple set of Roles that developer frequently use



Application Metrics

Quick access to key application metrics

- Use of Prometheus Query Language
- Easily build up queries and plot to visualize application and component trends



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](https://www.linkedin.com/company/red-hat)



[youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)



[facebook.com/redhatinc](https://www.facebook.com/redhatinc)



twitter.com/RedHat