



# OpenShift Container Platform 4

Emre Özkan  
Solution Architect

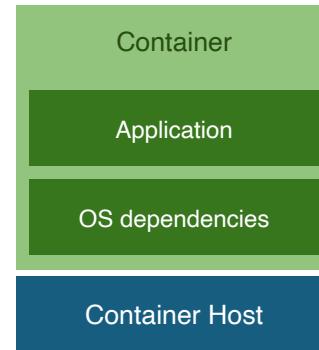
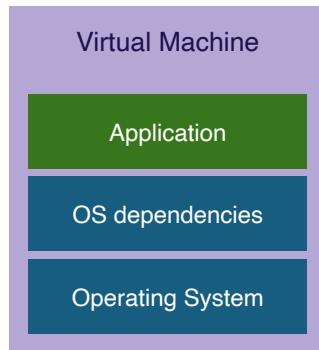
# Agenda

- OpenShift and Kubernetes Core Concepts
- OpenShift 4 Architecture
- OpenShift Virtualization
- OpenShift Security
- OpenShift Developer Services



# OpenShift and Kubernetes core concepts

# VIRTUAL MACHINES AND CONTAINERS



- + VM Isolation
- Complete OS
- Static Compute
- Static Memory
- High Resource Usage

- + Container Isolation
- + Shared Kernel
- + Burstable Compute
- + Burstable Memory
- + Low Resource Usage

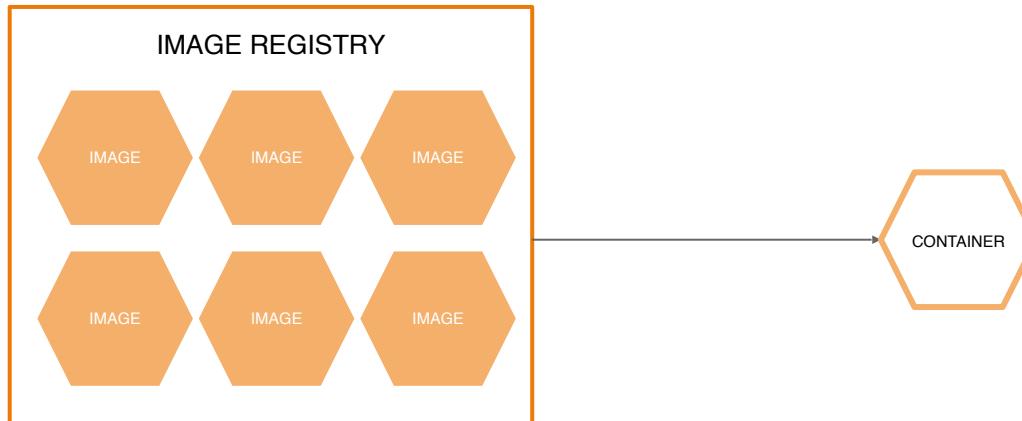
# a container is the smallest compute unit



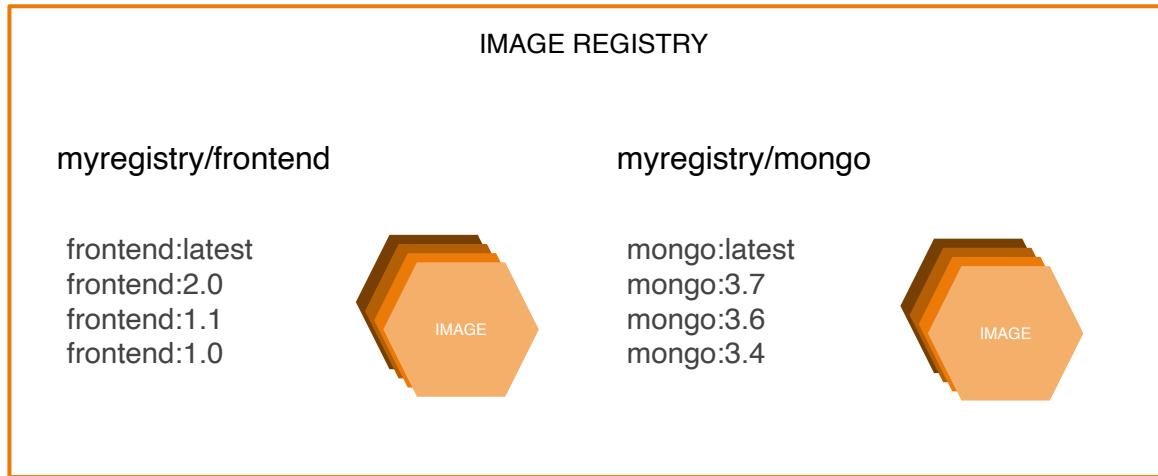
# containers are created from container images



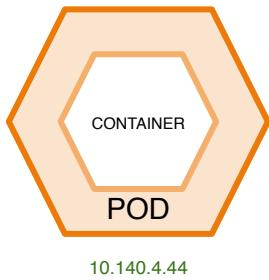
# container images are stored in an image registry



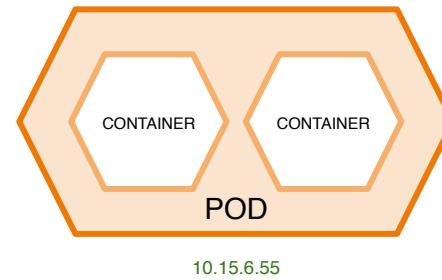
# an image repository contains all versions of an image in the image registry



containers are wrapped in pods which are units of deployment and management

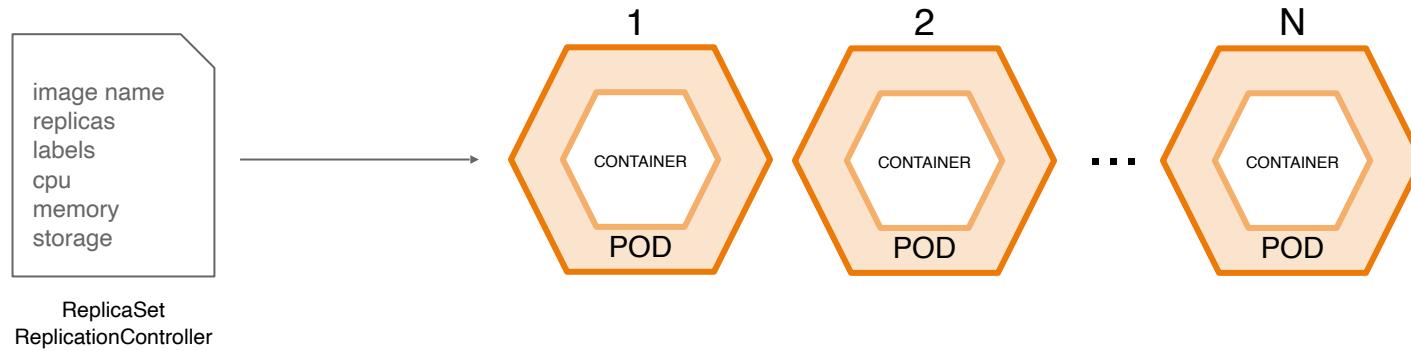


10.140.4.44

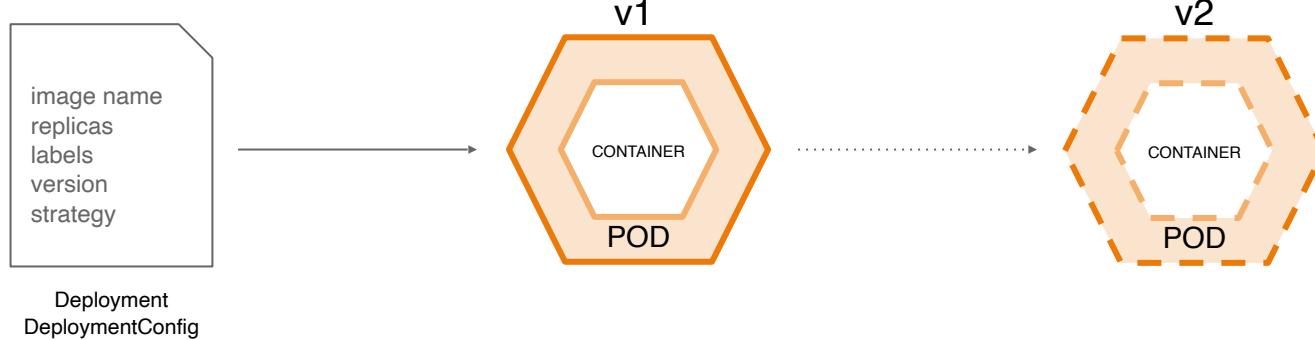


10.15.6.55

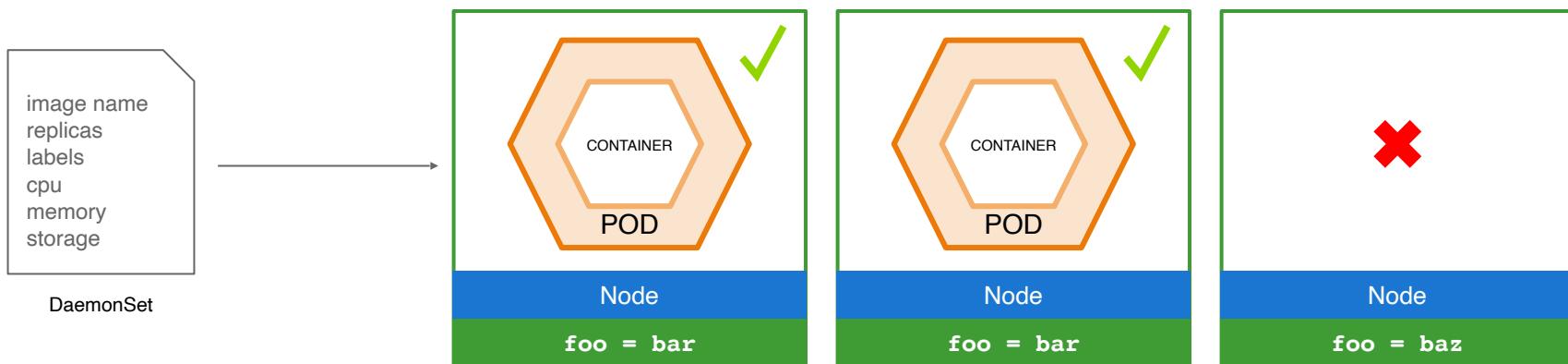
# ReplicationControllers & ReplicaSets ensure a specified number of pods are running at any given time



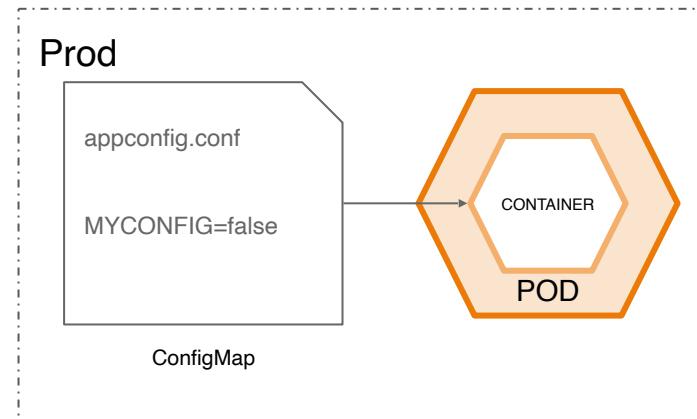
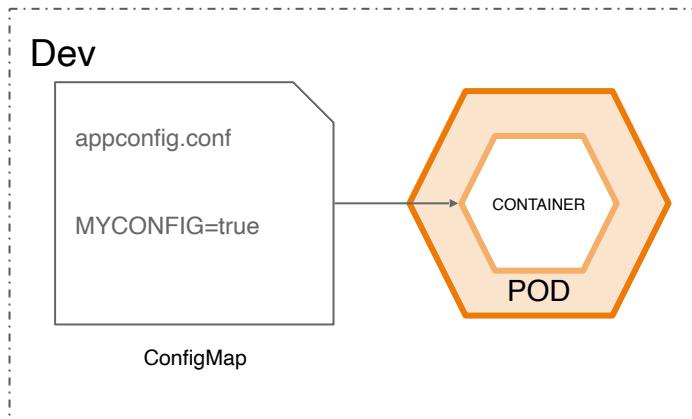
# Deployments and DeploymentConfigurations define how to roll out new versions of Pods



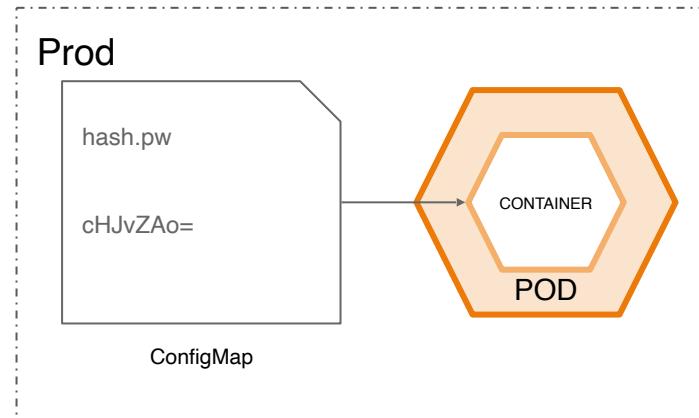
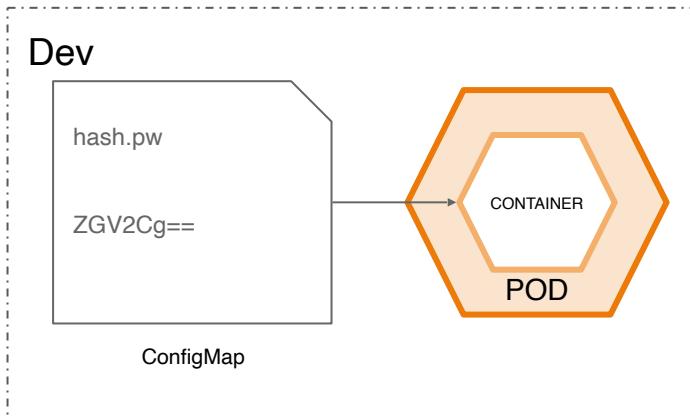
# a daemonset ensures that all (or some) nodes run a copy of a pod



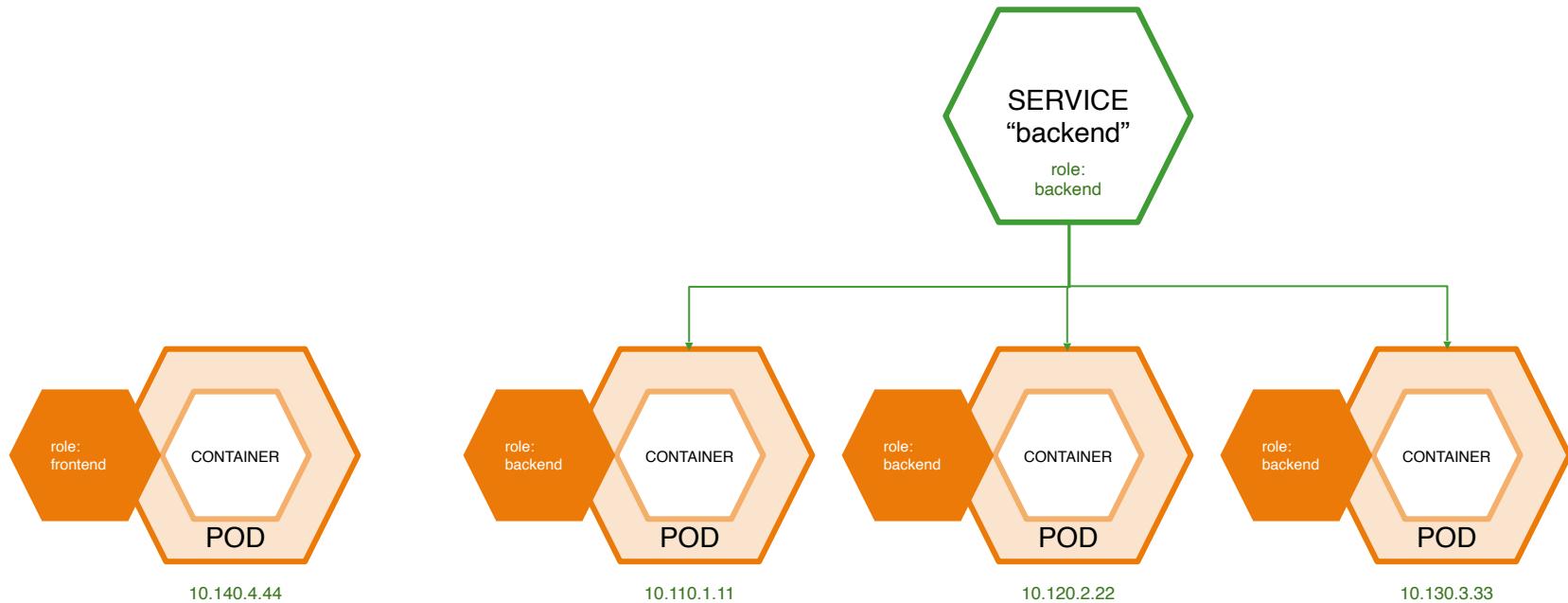
# configmaps allow you to decouple configuration artifacts from image content



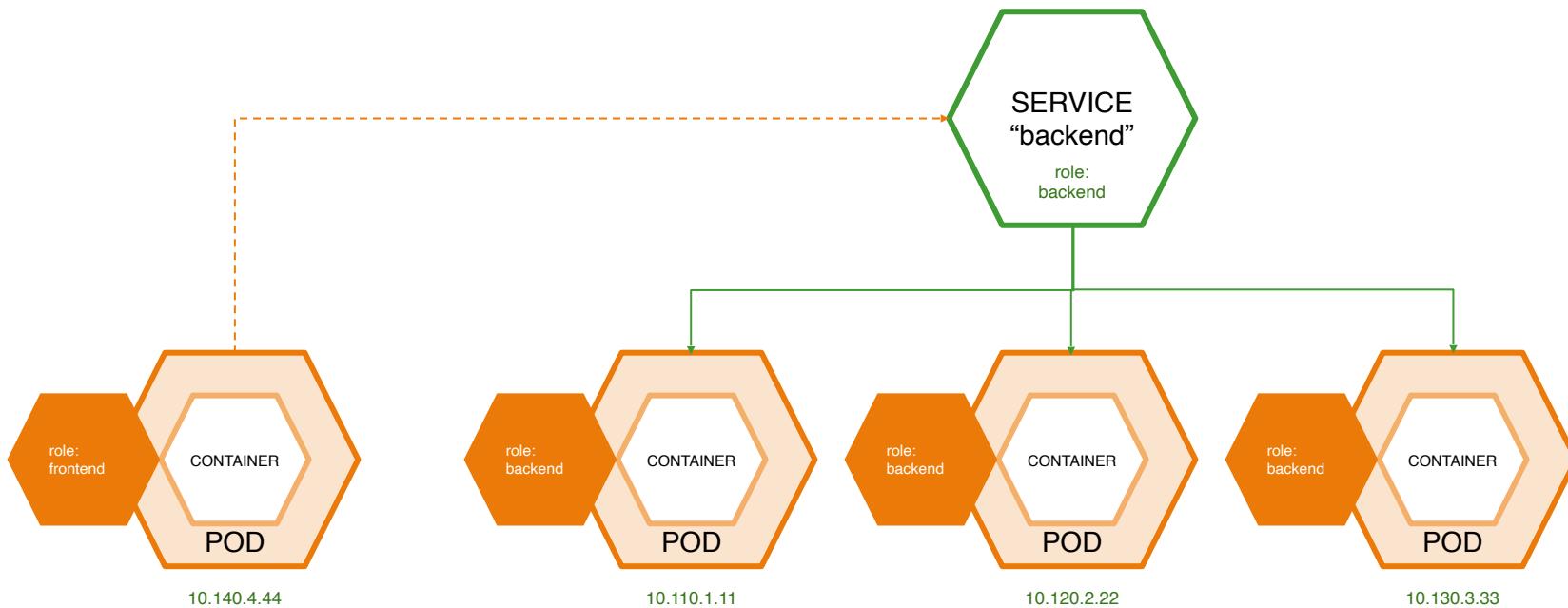
secrets provide a mechanism to hold sensitive information such as passwords



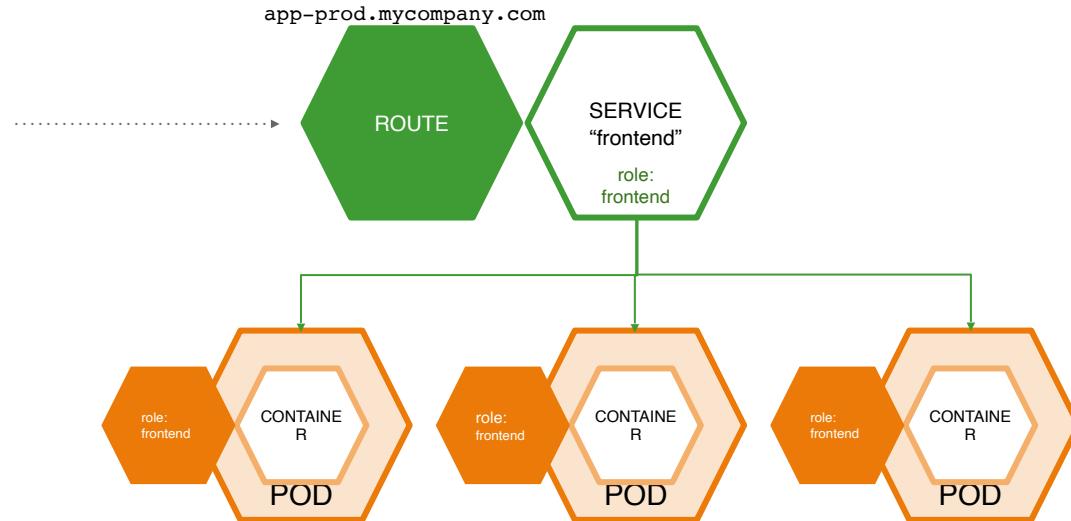
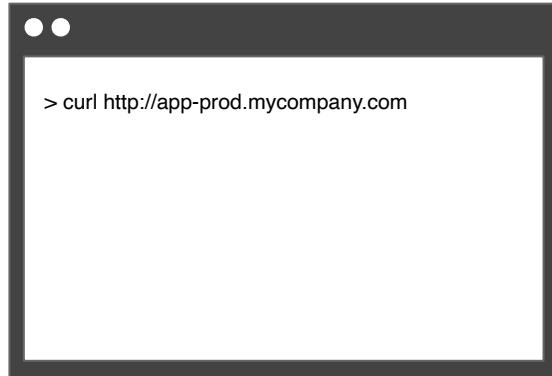
# services provide internal load-balancing and service discovery across pods



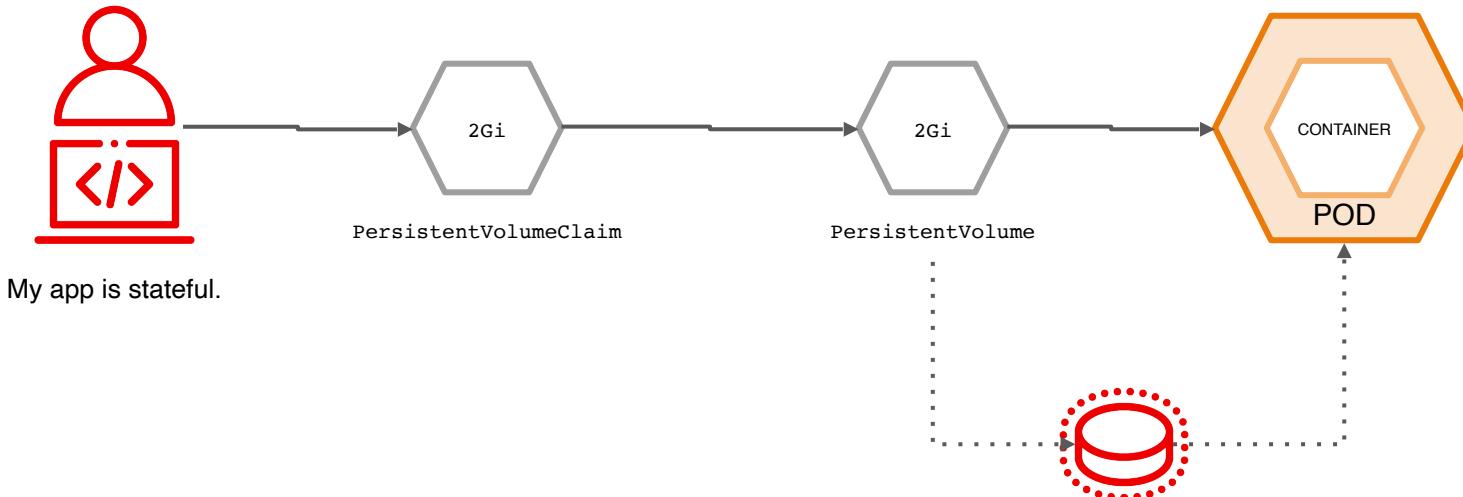
# apps can talk to each other via services



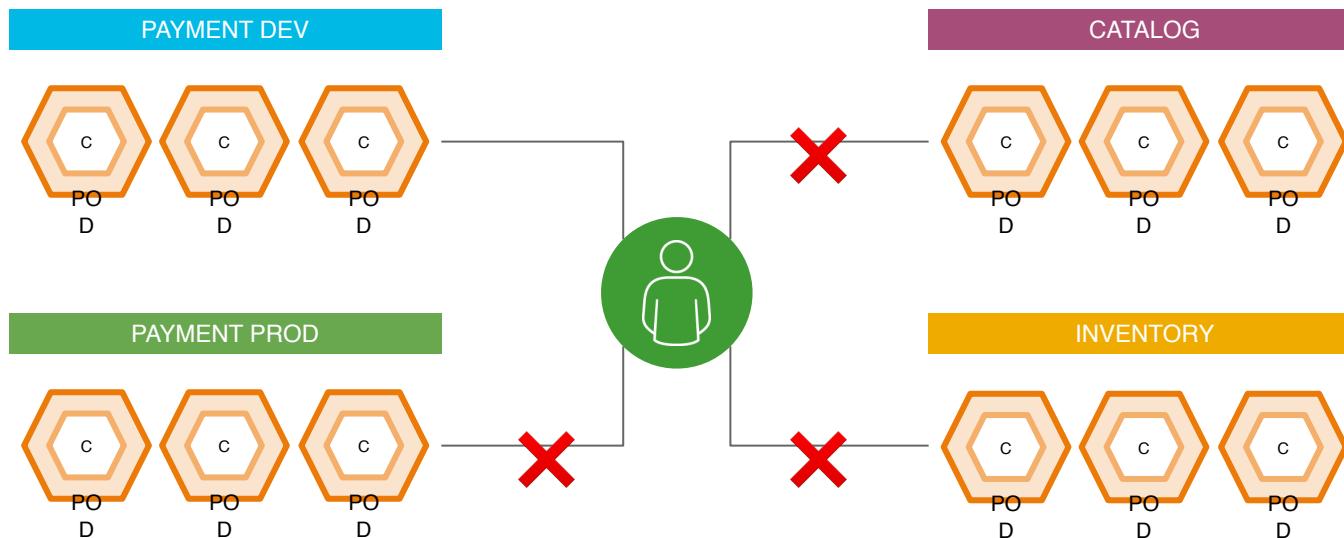
# routes make services accessible to clients outside the environment via real-world urls



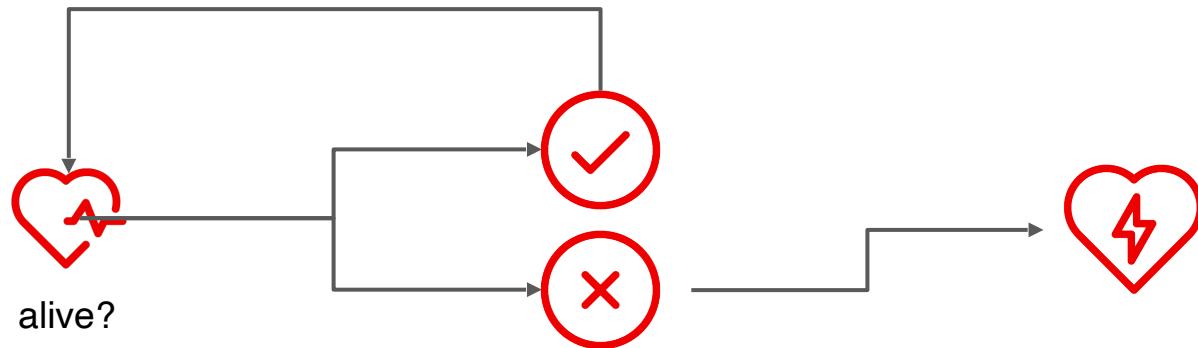
# Persistent Volume and Claims



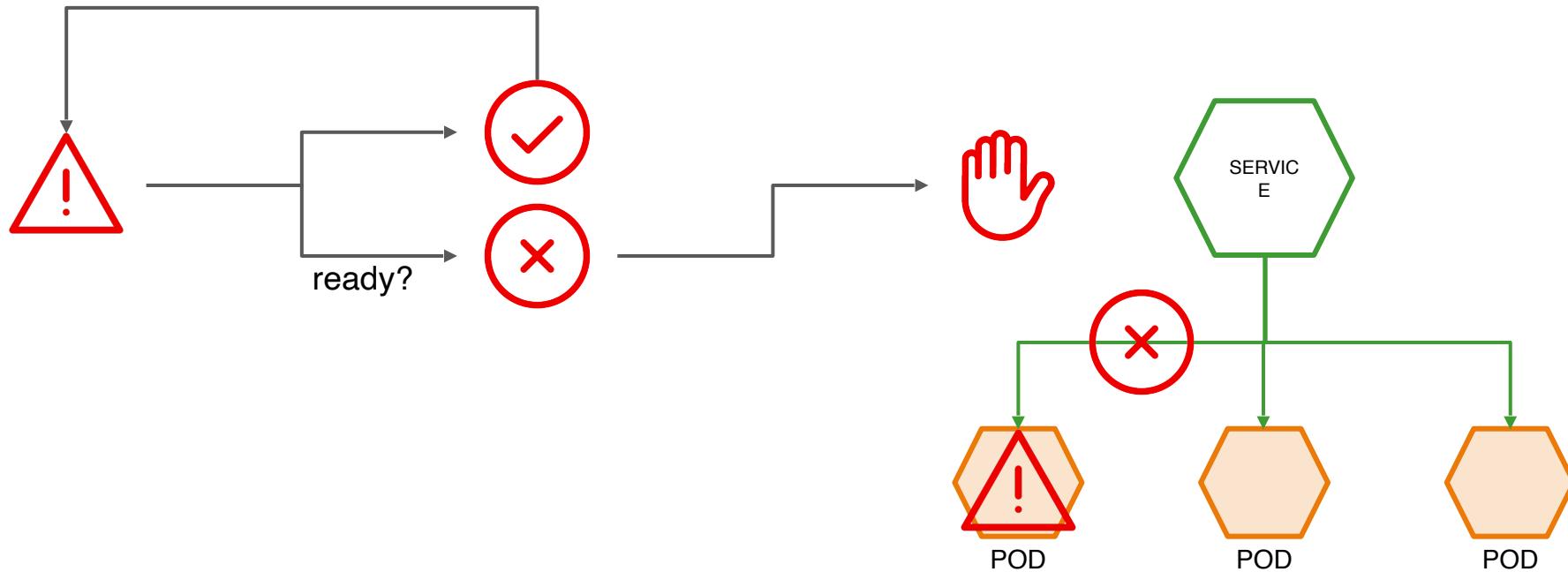
projects isolate apps across environments,  
teams, groups and departments



## Liveness Probes



## Readiness Probes





# OpenShift 4 Architecture

# your choice of infrastructure

COMPUTE

NETWORK

STORAGE

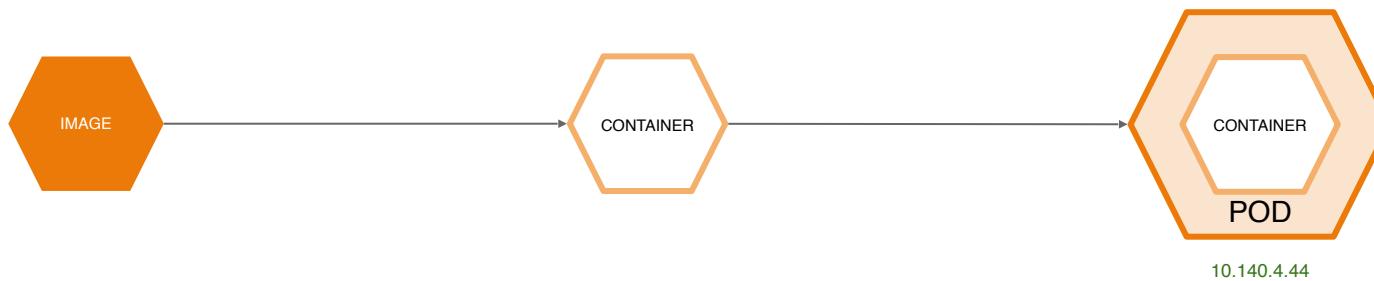
# workers run workloads



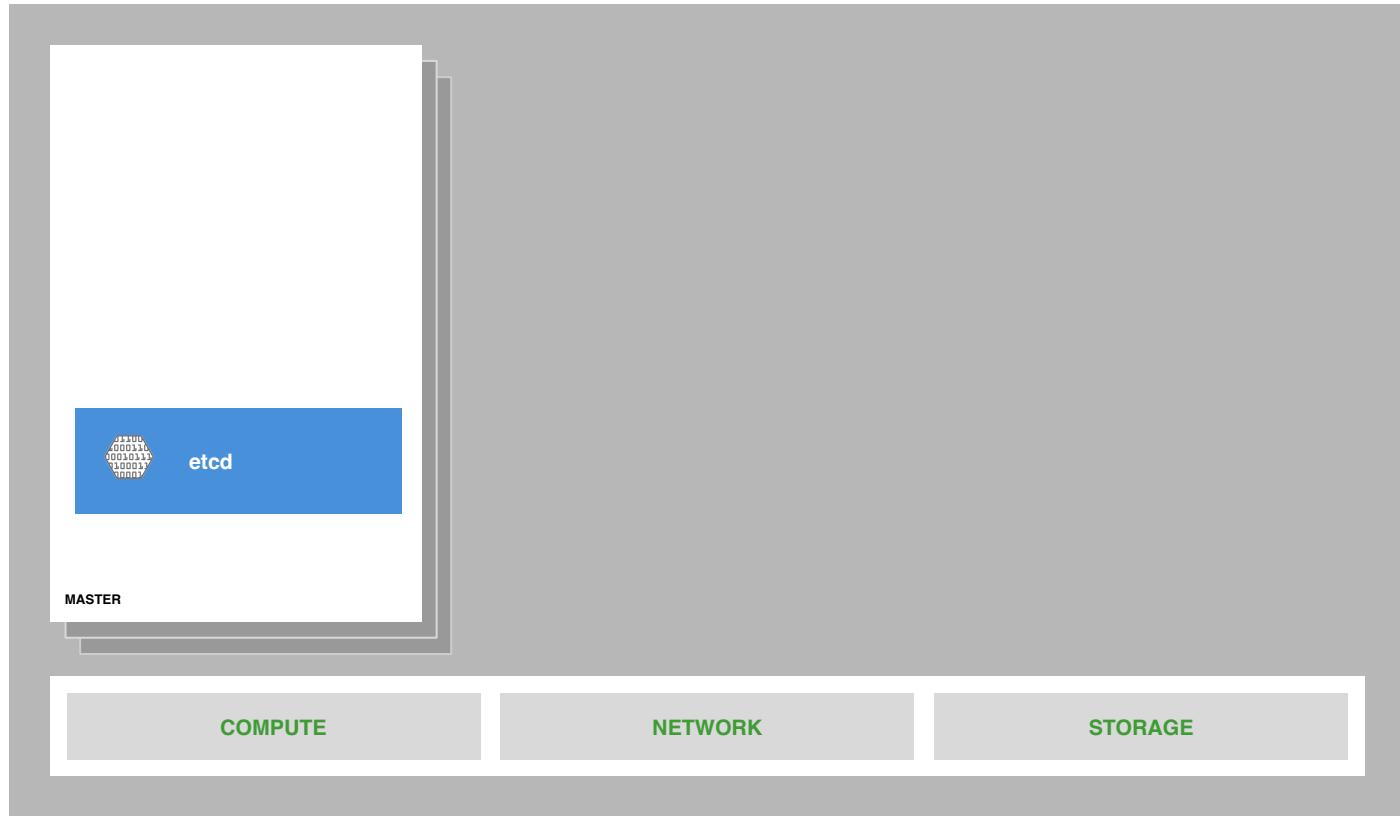
# masters are the control plane



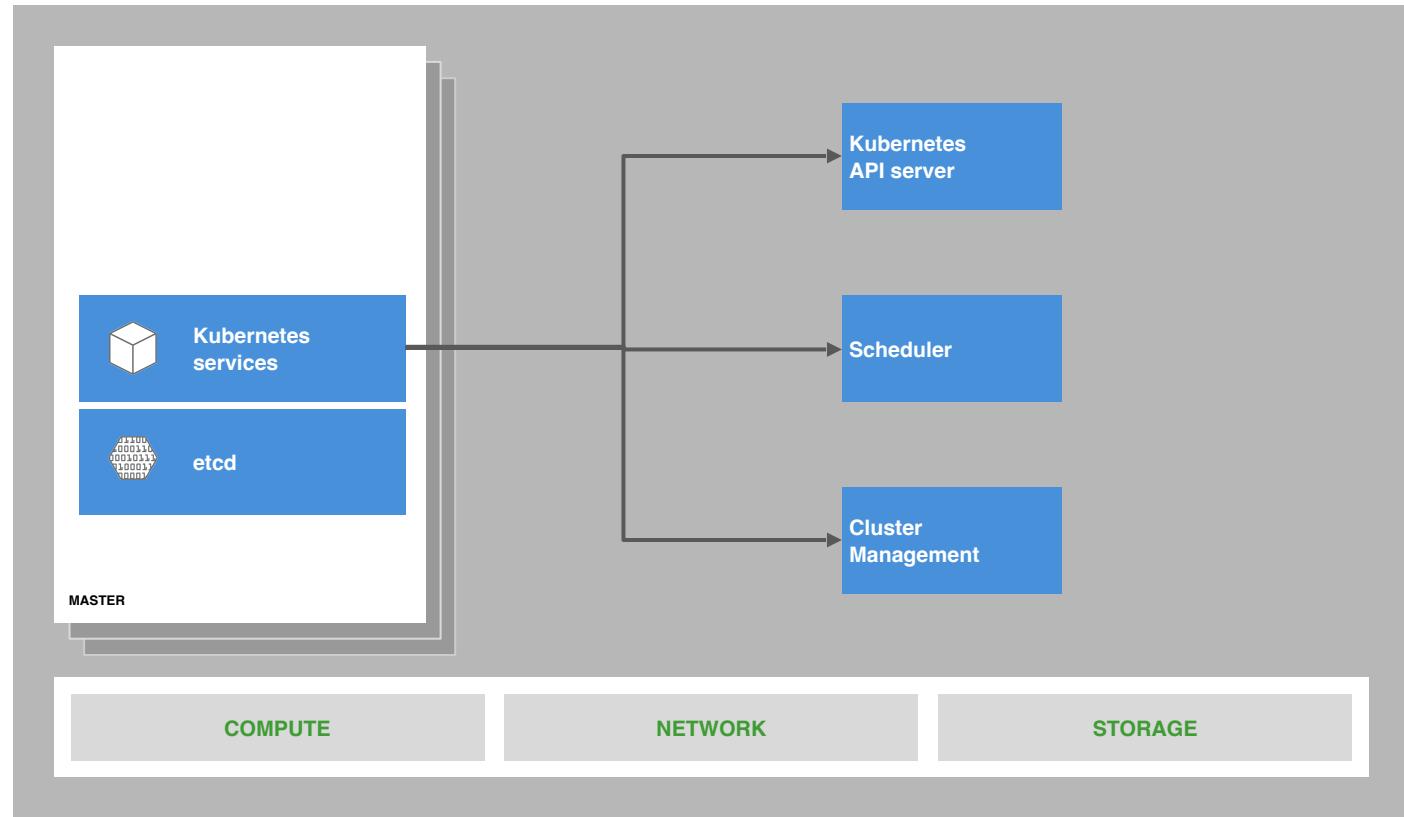
# everything runs in pods



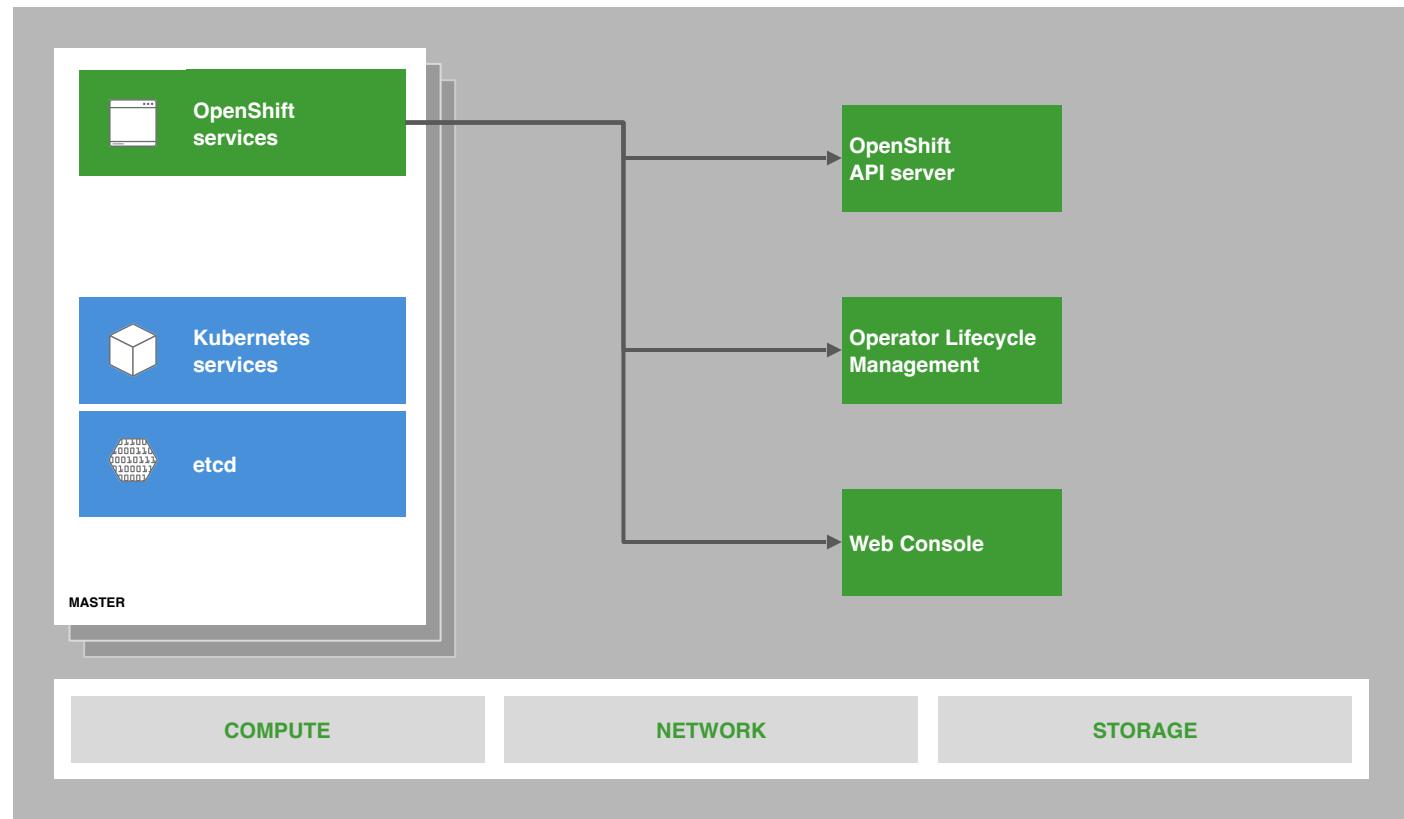
# state of everything



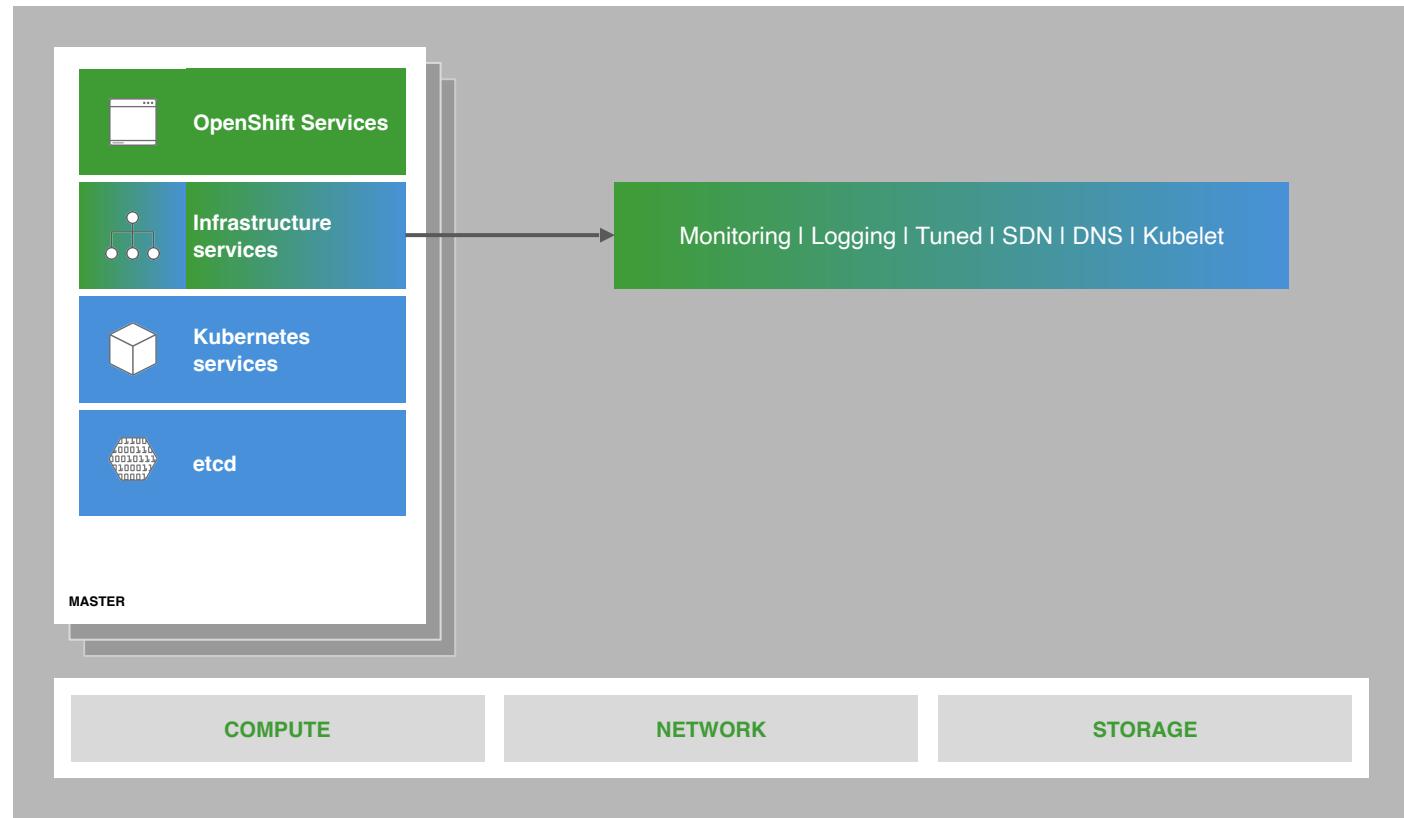
# core kubernetes components



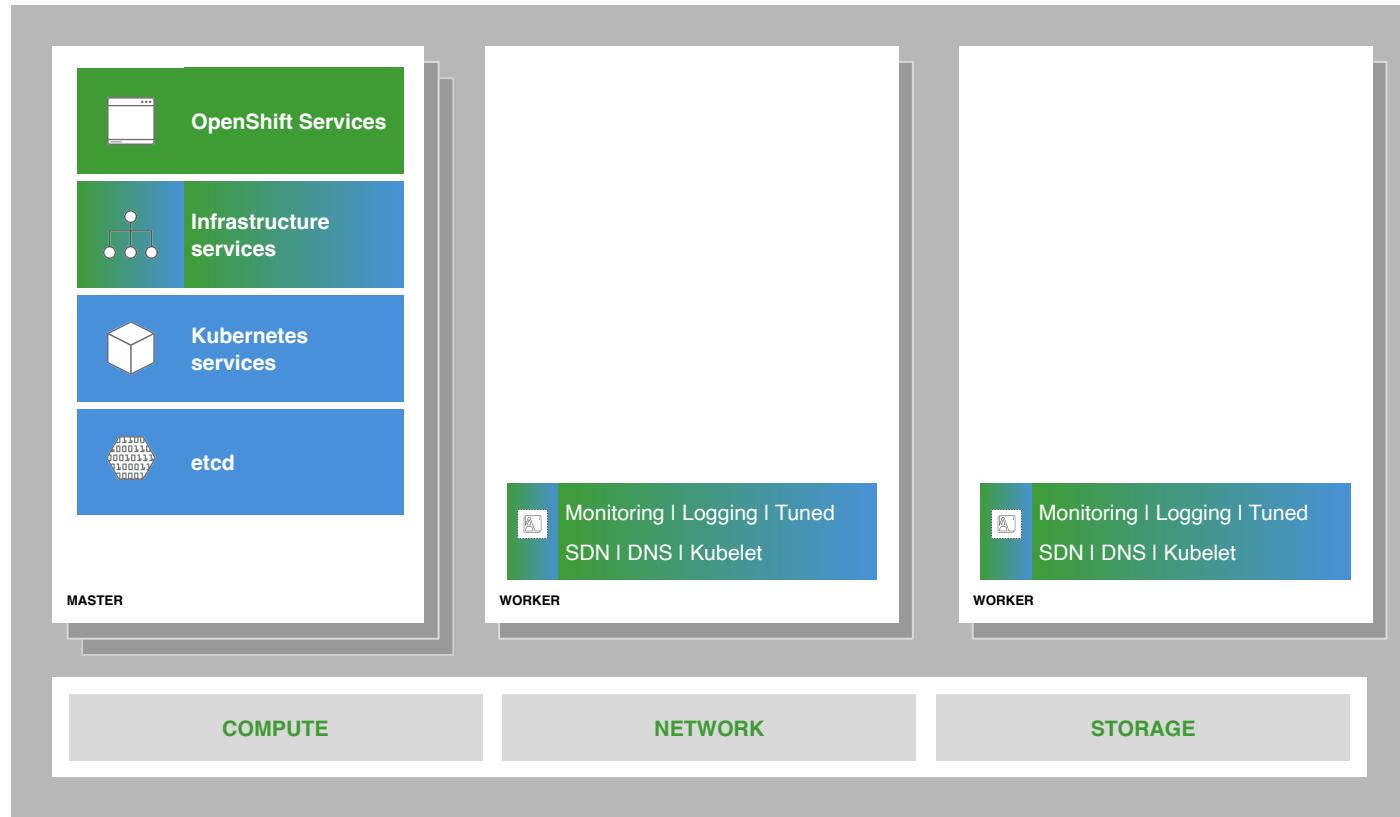
# core OpenShift components



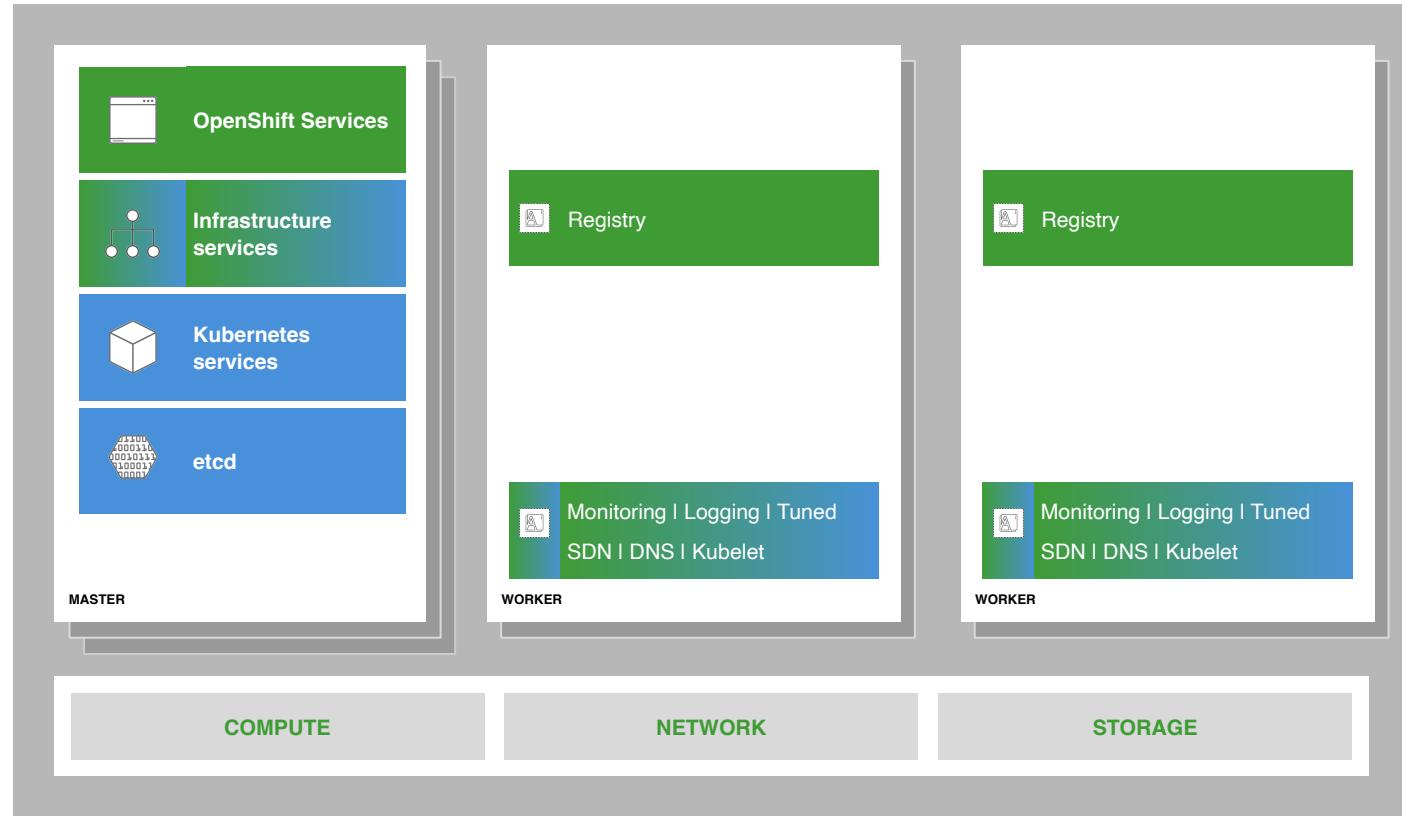
# internal and support infrastructure services



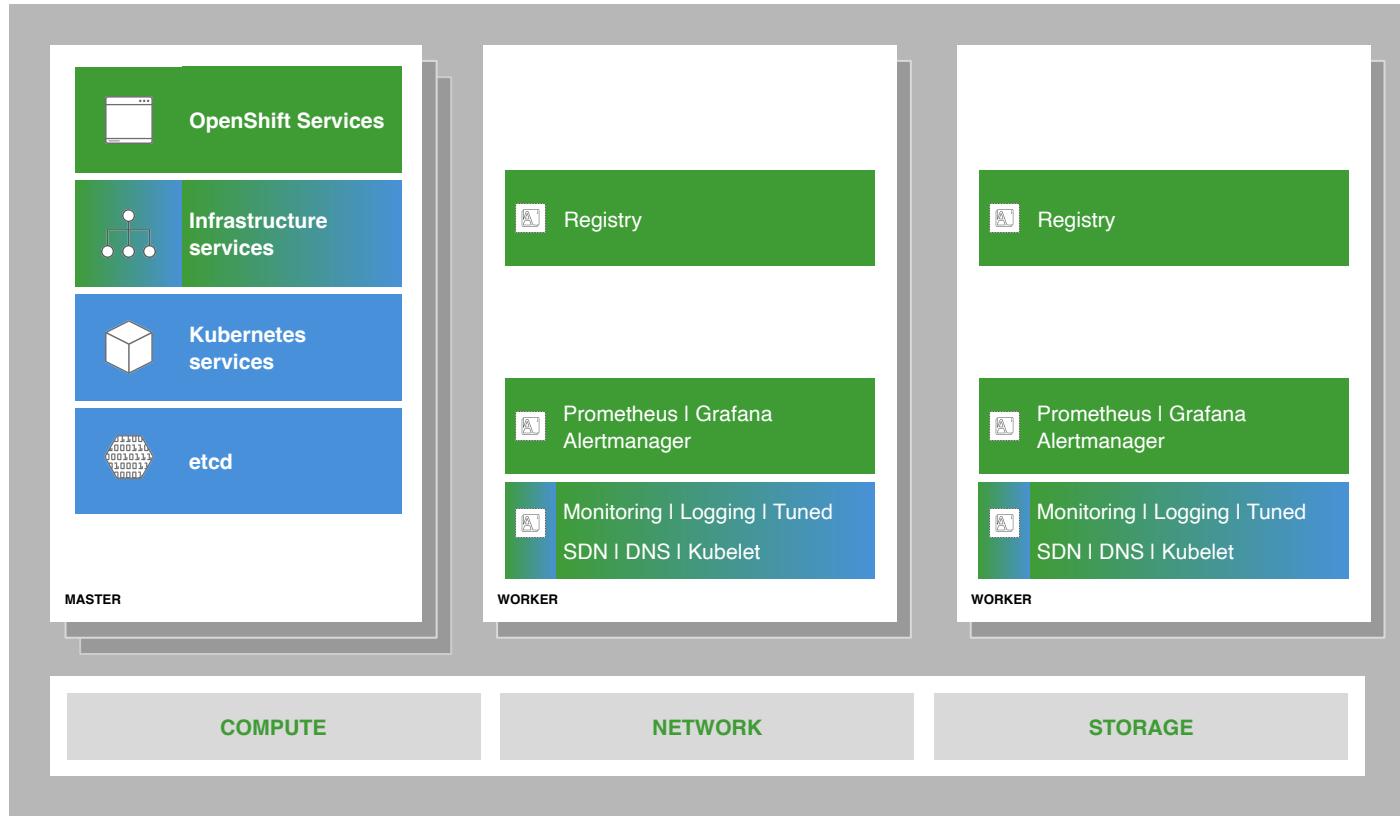
# run on all hosts



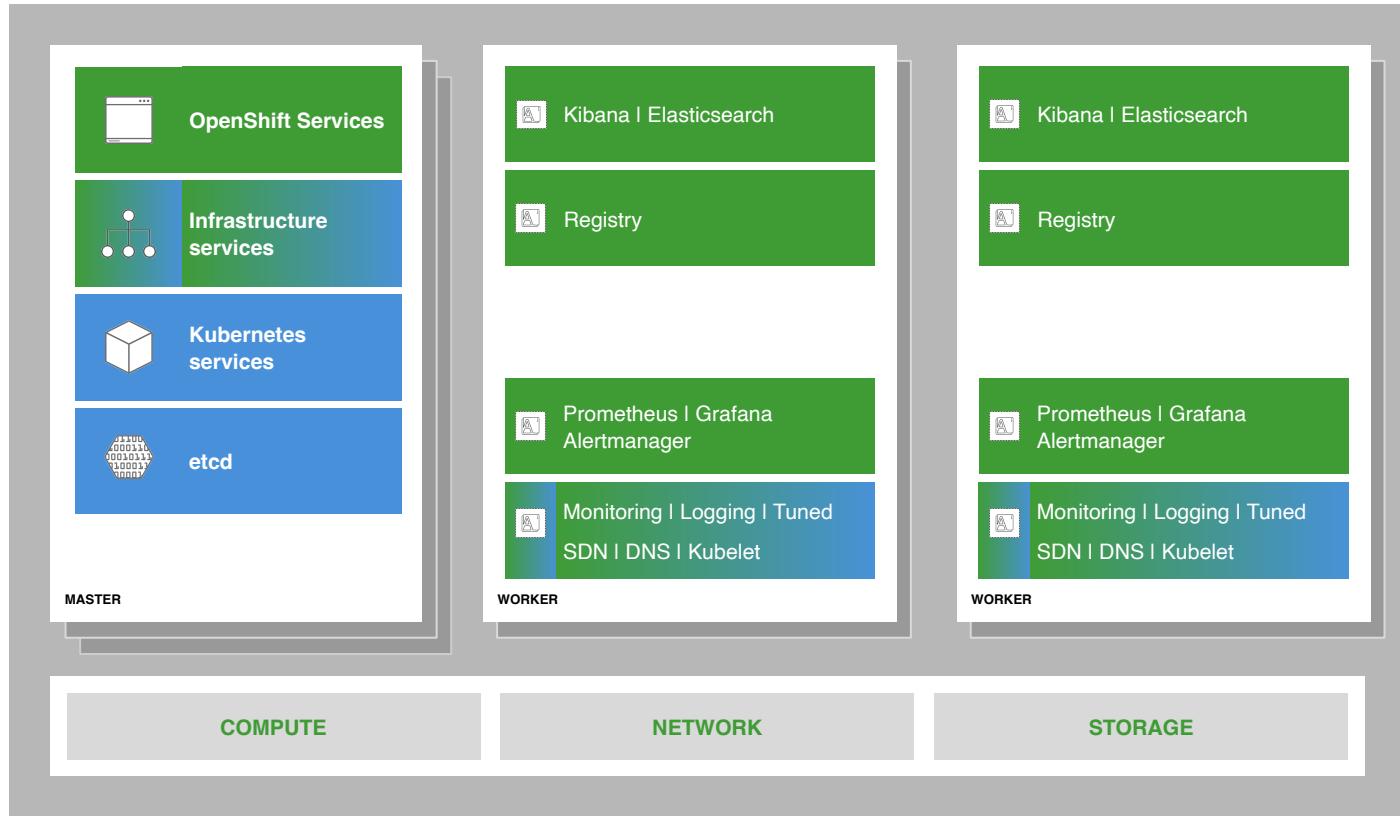
# integrated image registry



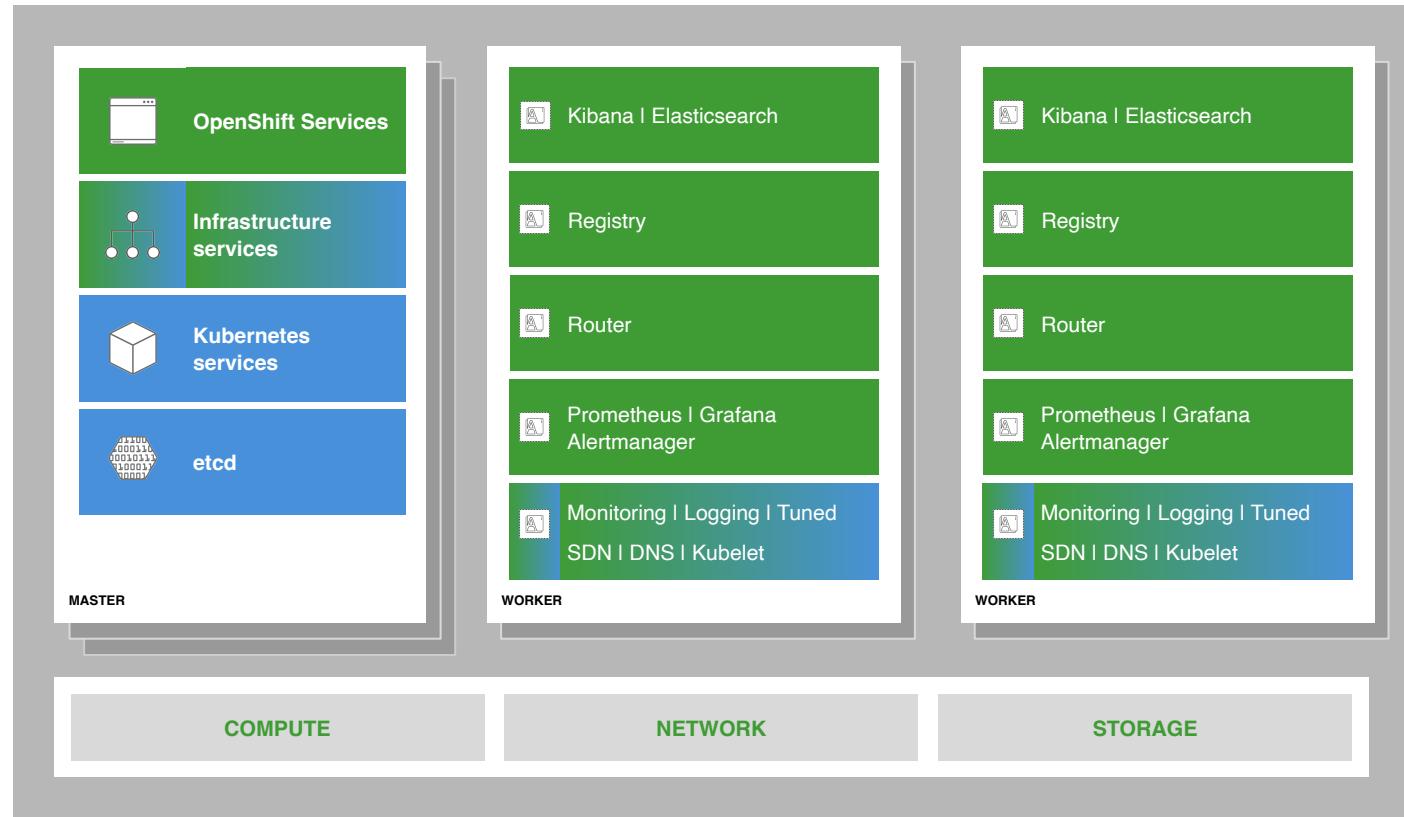
# cluster monitoring



# log aggregation



# integrated routing

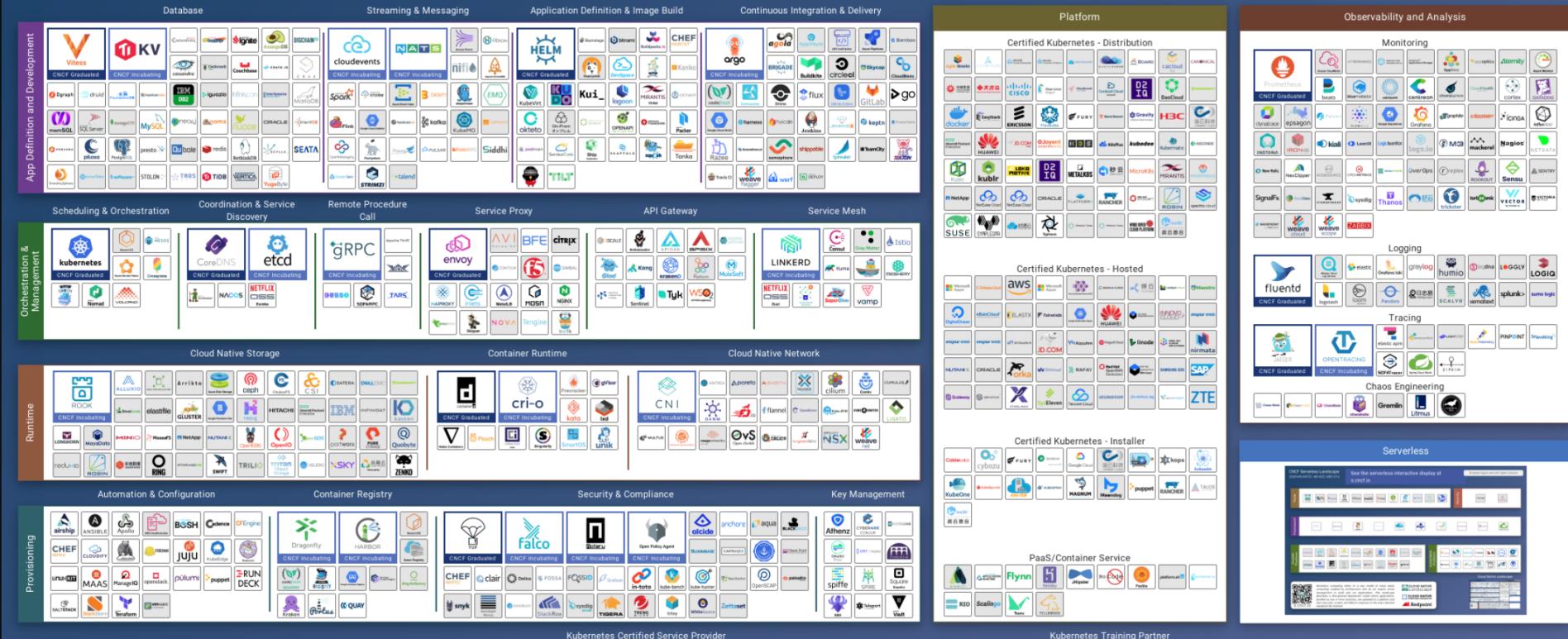


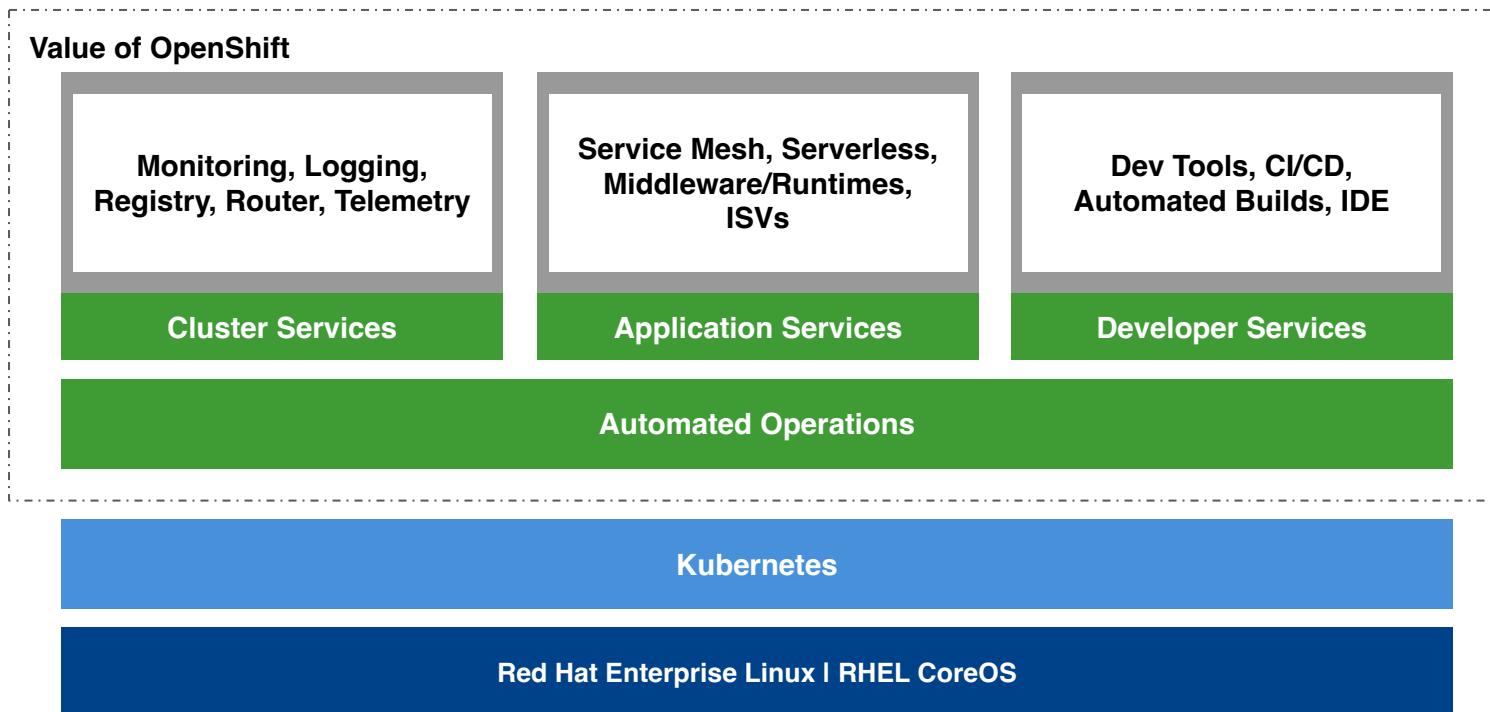
# The Cloud Native App Dev Challenge

CNCF Cloud Native Landscape  
2020-06-04T21:40:43Z 3d5131c

Overwhelmed? Please see the CNCF Trail Map. That and the interactive landscape are at [l.cncf.io](https://l.cncf.io)

Greyed logos are not open source





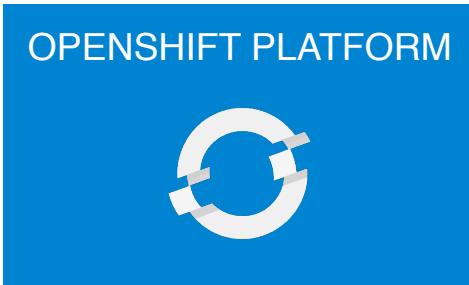
Best IT Ops Experience

CaaS  $\longleftrightarrow$  PaaS  $\longleftrightarrow$  FaaS

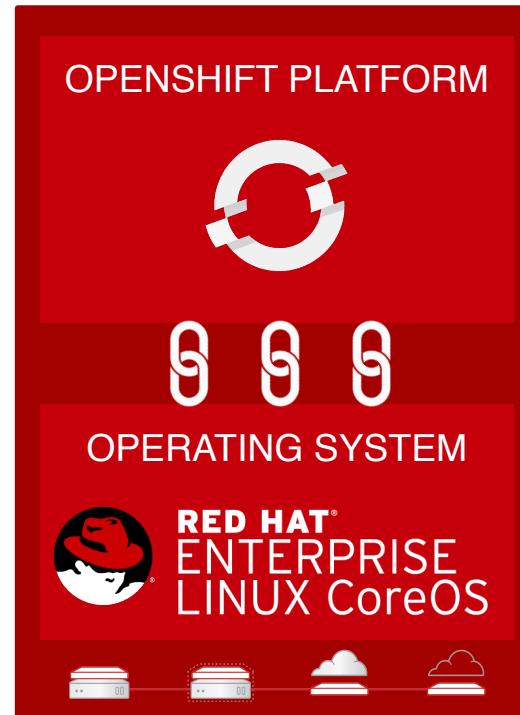
Best Developer Experience

# FULL STACK AUTOMATED INSTALL

OPENShift 3

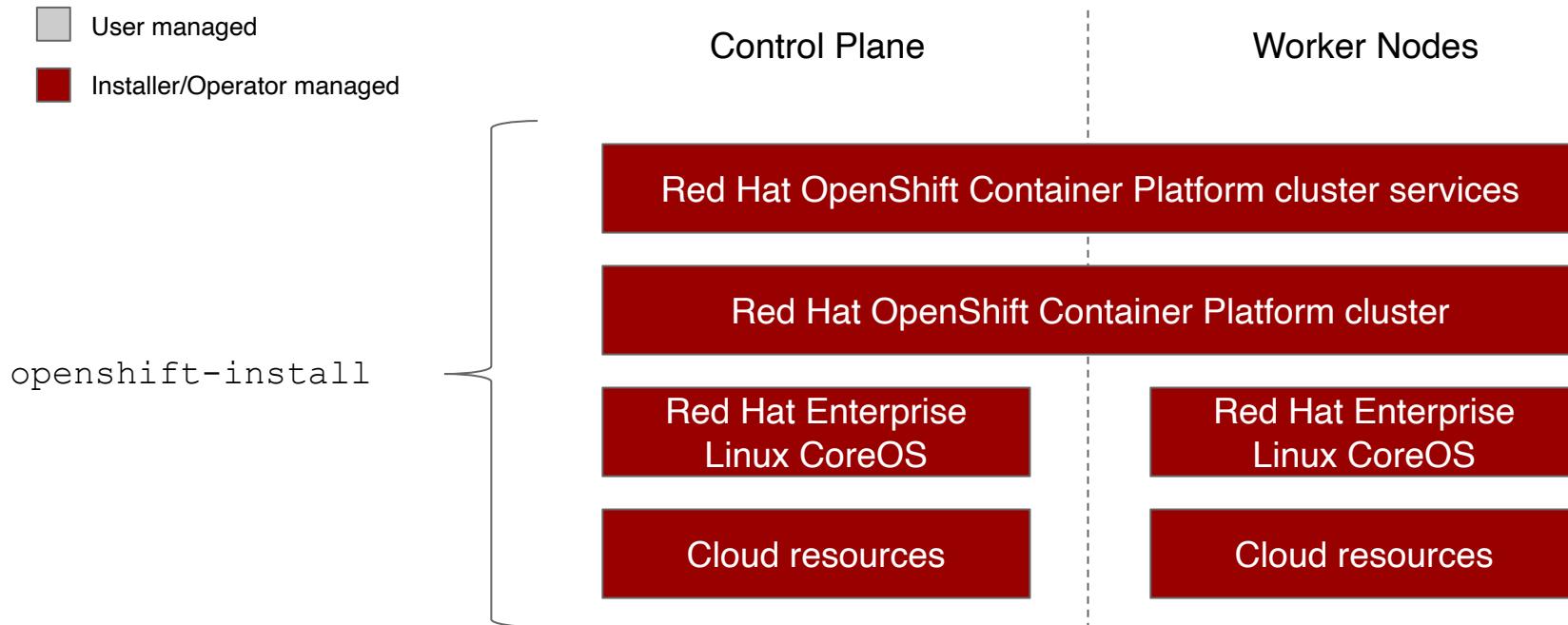


OPENSIFT 4



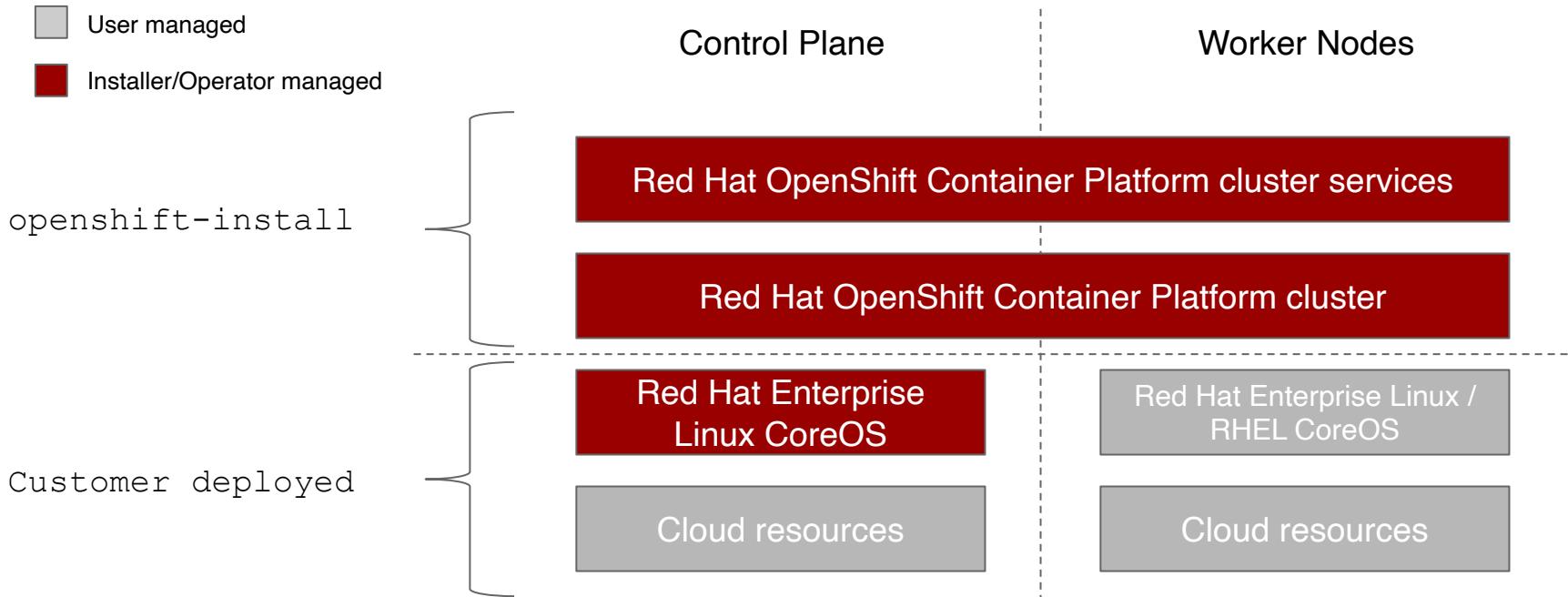
# INSTALLER PROVISIONED INFRASTRUCTURE (IPI)

Day 1: OpenShift install - Day 2: Operators

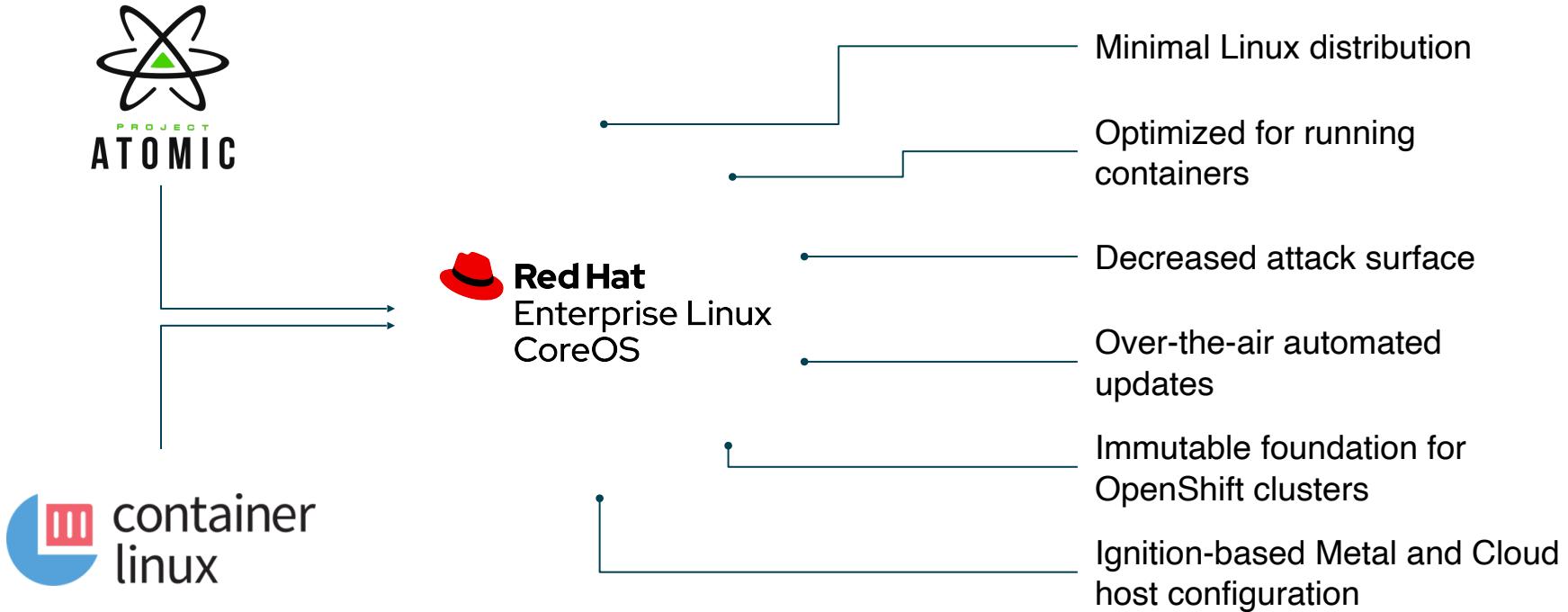


# USER PROVISIONED INFRASTRUCTURE (UPI)

Day 1: OpenShift install - Day 2: Operators + Customer Managed Nodes & Infra



# Red Hat Enterprise Linux CoreOS



# Immutable Operating System

**Red Hat Enterprise Linux CoreOS is versioned with OpenShift**

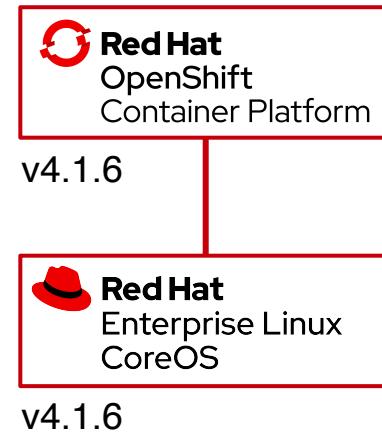
CoreOS is tested and shipped in conjunction with the platform.  
Red Hat runs thousands of tests against these configurations.

**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

**RHEL CoreOS admins are responsible for:**  
Nothing. 😊 🙌





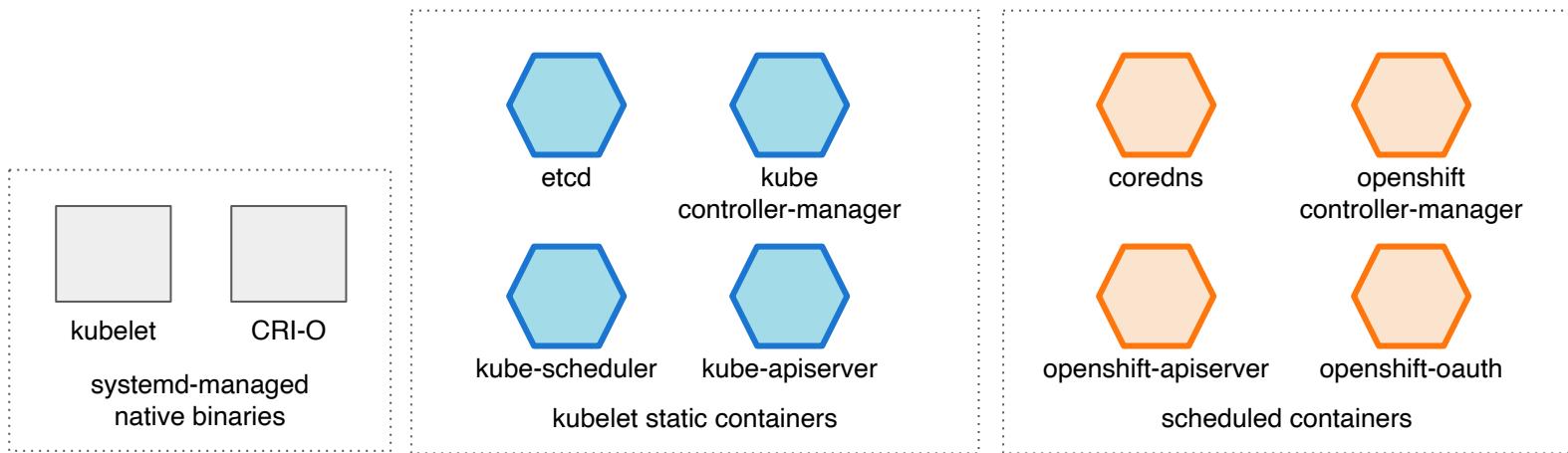
A lightweight, OCI-compliant container runtime

Minimal and Secure  
Architecture

Optimized for  
Kubernetes

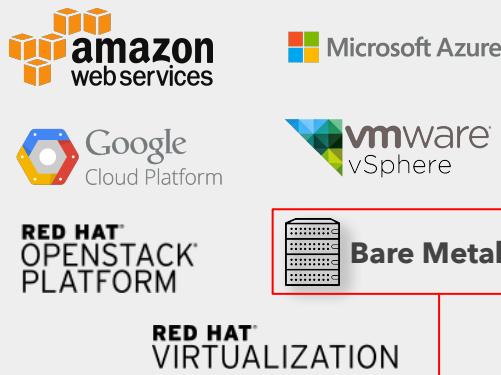
Runs any OCI-  
compliant image  
(including docker)

# CoreOS “pod” architecture



## Supported Providers

### Full Stack Automation (IPI)



New addition in OCP 4.6

### Pre-existing Infrastructure (UPI)



Now supports deploying to VMware vSphere 7.0

# OpenShift offers the broadest set of hybrid cloud services



## OpenShift Dedicated

*Managed By Red Hat  
or*

## Amazon Red Hat OpenShift

*Managed by Red Hat & AWS*

*or  
Customer  
Managed*



## Azure Red Hat OpenShift

*Jointly Managed &  
Supported*

*or*

**Customer  
Managed**



Google Cloud Platform

## Red Hat OpenShift Dedicated

*Managed By Red  
Hat*

*or*

**Customer  
Managed**



IBM Cloud

## Red Hat OpenShift on IBM Cloud

*Jointly Engineered*

*or*

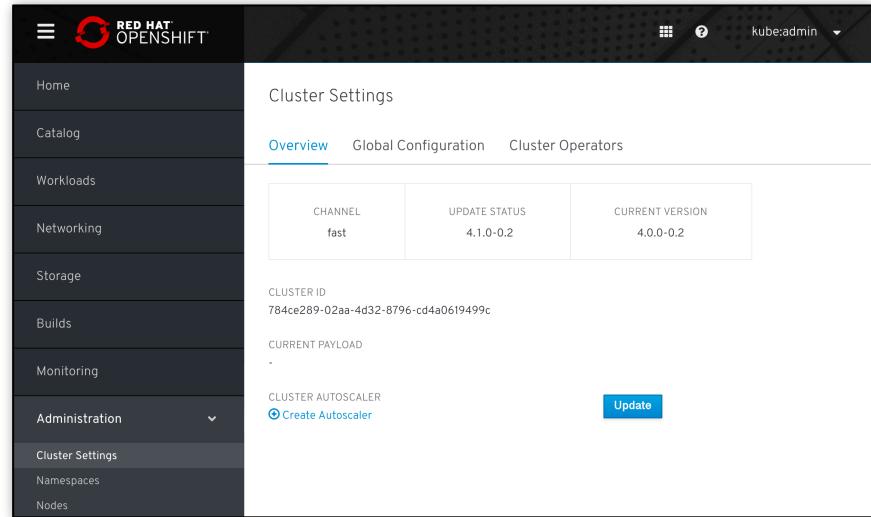
**Customer  
Managed (UPI)**

On-premises

**Customer  
Managed**

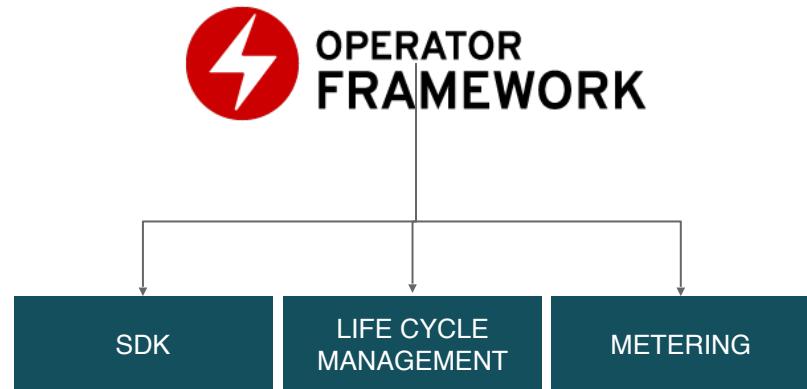
# OVER-THE-AIR UPDATES

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



# OPERATOR FRAMEWORK

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



# OPERATOR FRAMEWORK



## Basic Install

Automated application provisioning and configuration management

## Seamless Upgrades

Patch and minor version upgrades supported

## Full Lifecycle

App lifecycle, storage lifecycle (backup, failure recovery)

## Deep Insights

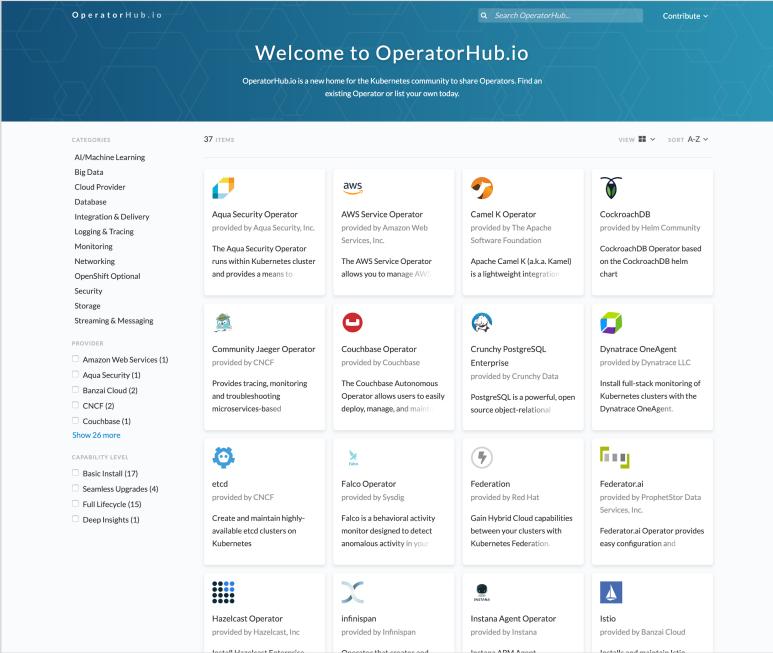
Metrics, alerts, log processing and workload analysis

## Auto Pilot

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



# OperatorHub.io Ecosystem



The public registry for finding  
Kubernetes Operator backed  
services

# OperatorHub in OpenShift

The embedded registry for  
Community and Certified  
Operators from Red Hat and  
Partners, tested and verified on  
OpenShift 4

The screenshot shows the OperatorHub interface within the OpenShift web console. At the top, there's a navigation bar with 'Project: default' and a dropdown. Below it, the title 'OperatorHub' is displayed, followed by a brief description: 'Discover Operators from the Kubernetes community and Red Hat partners, curated by Red Hat. Operators can be installed on your clusters to provide optional add-ons and shared services to your developers. Once installed appear in the [Developer Catalog](#), providing a self-service experience.' On the left, there's a sidebar with various filter categories: All Items, AI/Machine Learning, Application Monitoring, Big Data, Database, Developer Tools, Integration & Delivery, Logging & Tracing, Monitoring, Networking, OpenShift Optional, Security, Security Policy Management, Storage, Streaming & Messaging, and Other. There are also filters for INSTALL STATE (Installed 3, Not Installed 40), PROVIDER TYPE (Red Hat 2, Certified 16, Community 25), and PROVIDER (Red Hat 13, AppDynamics 1, Aqua Security 1). A 'Filter by keyword...' input field is also present. The main area contains a grid of 15 operator cards, each with a thumbnail, name, provider, and a brief description. The operators listed are: AMQ Streams (Red Hat, Community), AppDynamics ClusterAgent (AppDynamics LLC, Community), Aqua Security Operator (Aqua Security, Inc., Community), Automation Broker Operator (provided by Red Hat, Inc., Community), Camel-K Operator (provided by The Apache Software Foundation, Community), CockroachDB (provided by Helln Community, Community), Community Jaeger Operator (provided by CNCF, Community), Couchbase Operator (provided by Couchbase, Community), Crunchy PostgreSQL Enterprise (provided by Crunchy Data, Community), Descheduler (provided by Red Hat, Community), Elasticsearch Operator (provided by Red Hat, Inc., Community), Federation (provided by Red Hat, Community), FederationAI Operator (provided by ProphetStor Data Services, Inc., Community), Hazelcast Operator (provided by Hazelcast, Inc., Community), and The Aqua Security Operator (runs within a OpenShift cluster and provides a means to deploy and manage Aqua Security services).

# OperatorHub in OpenShift

## For Cluster Admins:

The screenshot shows the OperatorHub interface. On the left, there's a sidebar with categories like All Items, AI/Machine Learning, Big Data, Database, Integration & Delivery, Logging & Tracing, and Monitoring. The main area displays the AMQ Streams operator details. It includes a logo, the name "AMQ Streams", the version "1.1.0 provided by Red Hat, Inc.", and a large blue "Install" button. Below this, it shows "OPERATOR VERSION 1.1.0" and "PROVIDER TYPE The core capability". There are also sections for Languages, Middleware, and Other. At the bottom, there are cards for "AMQ Streams" and "Aqua Security". A tooltip for the "AMQ Streams" card says "Red Hat AMQ Streams is a distributed event processing system based on the Apache Kafka™ open source project".

## For Developers:

The screenshot shows the Developer Catalog in OpenShift. It has tabs for Overview, YAML, and Resources, with Resources selected. Under Resources, there are tabs for Route, Service, StatefulSet, Deployment, and ReplicaSet. The main area shows a table of resources under the heading "Developer Catalog". It lists "my-cluster" with 6 items: "my-cluster-entity-operator" (Deployment, Created), "my-cluster-entity-operator-5778f899cc-iddz" (Pod, Running), "my-cluster-kafka" (StatefulSet, Created), and "my-cluster-hafka-0" (Pod, Running). Below this, there are cards for "Kafka" and "Kafka Connect", each representing a cluster.

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

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



# Machine Config Operator

A Kube-native way to configure hosts

OS configuration is stored and applied across the cluster via the Machine Config Operator.

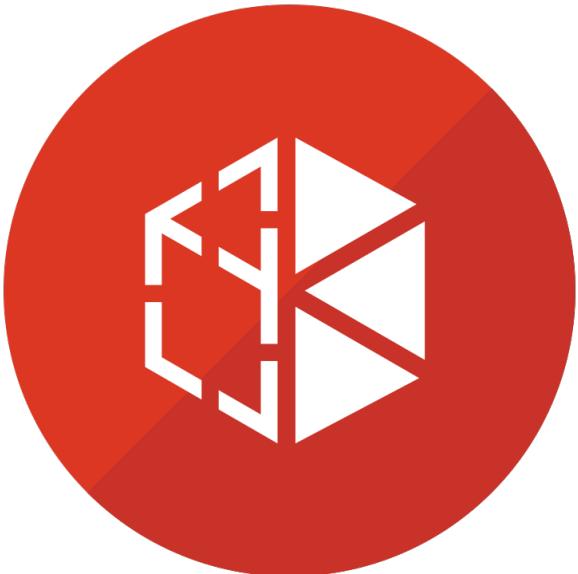
- Subset of ignition modules applicable post provisioning
  - SSH keys
  - Files
  - systemd units
  - kernel arguments
- Standard k8s YAML/JSON manifests
- Desired state of nodes is checked/fixed regularly
- Can be paused to suspend operations

```
# test.yaml
apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
  labels:
    machineconfiguration.openshift.io/role: worker
  name: test-file
spec:
  config:
    storage:
      files:
        - contents:
            source: data:,hello%20world%0A
            verification: {}
      filesystem: root
      mode: 420
      path: /etc/test
```



# OpenShift Virtualization

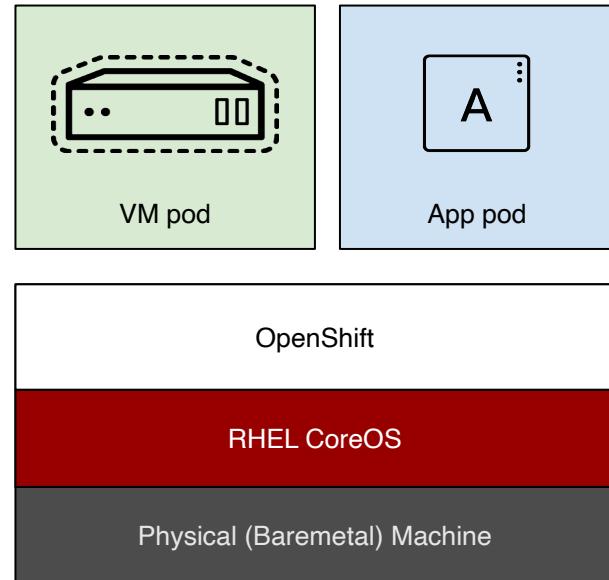
## OpenShift Virtualization



A virtualization API and runtime for OpenShift, built on KubeVirt, to run and manage virtual machines using a Kubernetes-native way

# Virtual machines in a container world

- Provides a way to transition application components which cannot be directly containerized into a Kubernetes system
  - Integrates directly into existing k8s clusters
  - Follows Kubernetes paradigms:
    - Container Networking Interface (CNI)
    - Container Storage Interface (CSI)
    - Custom Resource Definitions (CRD, CR)
- Schedule, connect, and consume VM resources as container-native
- VMs and containers living side-by-side



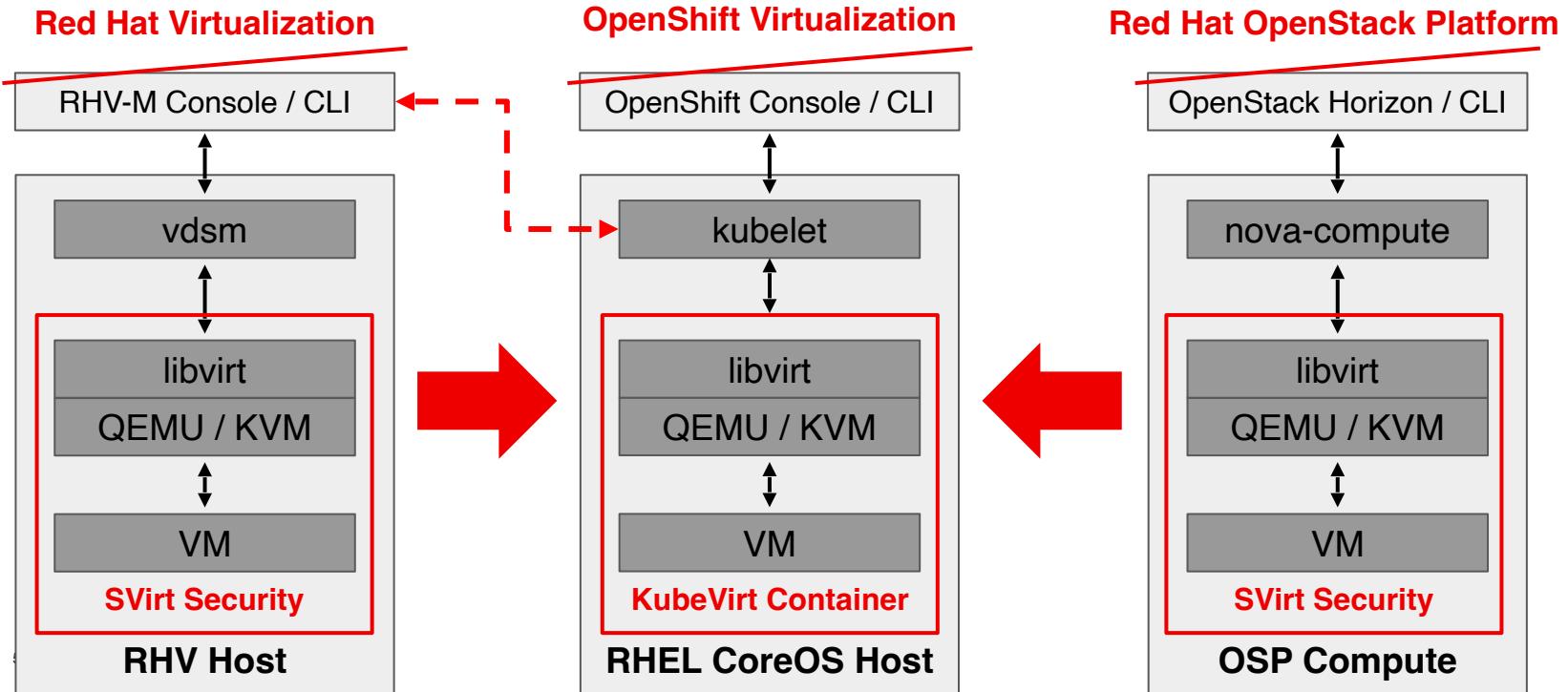
# Virtualization native to Kubernetes

- New CustomResourceDefinitions (CRDs) for native VM integration, for example:
  - VirtualMachine
  - VirtualMachineInstance
  - VirtualMachineInstanceMigration
  - DataVolume
- Operators are a Kubernetes-native way to introduce new capabilities

```
apiVersion: kubevirt.io/v1alpha3
kind: VirtualMachine
metadata:
  labels:
    app: demo
    flavor.template.kubevirt.io/small: "true"
    name: rhel
spec:
  dataVolumeTemplates:
  - apiVersion: cdi.kubevirt.io/v1alpha1
    kind: DataVolume
    metadata:
      creationTimestamp: null
      name: rhel-rootdisk
    spec:
      pvc:
        accessModes:
        - ReadWriteMany
        resources:
          requests:
            storage: 20Gi
        storageClassName: managed-nfs-storage
        volumeMode: Filesystem
```

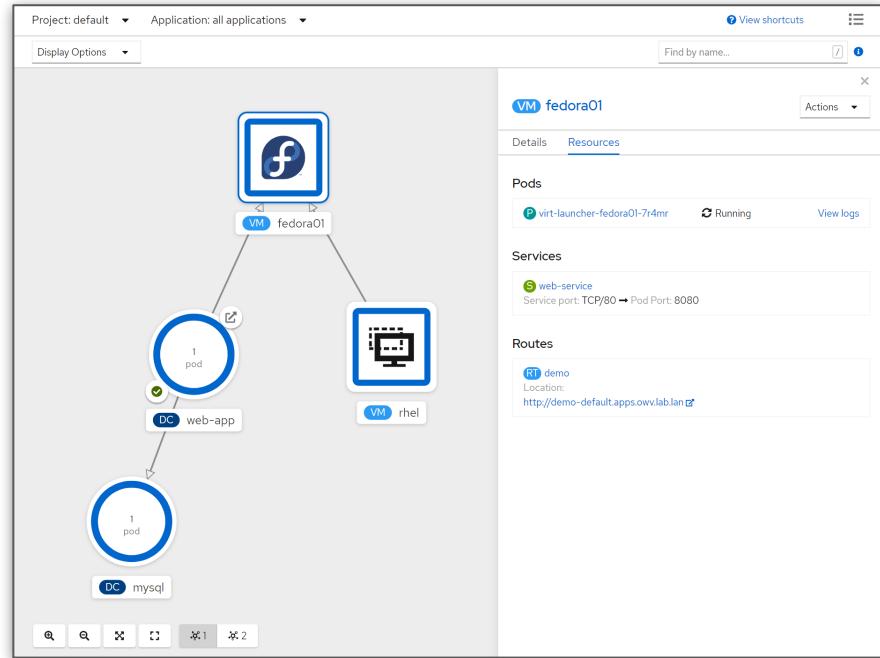
# Containerizing KVM

Same trusted, mature KVM-based stack, wrapped in modern management and automation



# Using VMs and containers together

- Virtual Machines connected to pod networks are accessible using standard Kubernetes methods:
  - Service
  - Route
  - Ingress
- Network policies apply to VM pods the same as application pods
- VM-to-pod, and vice-versa, communication happens over SDN or ingress depending on network connectivity



# Virtual Machine Management

- Create, modify, and destroy virtual machines, and their resources, using the OpenShift web interface or CLI
- Use the `virtctl` command to simplify virtual machine interaction from the CLI

The screenshot shows the Red Hat OpenShift Container Platform web interface. The left sidebar is titled "Workloads" and includes options like "Pods", "Virtualization", "Deployments", "Deployment Configs", "Stateful Sets", "Secrets", "Config Maps", "Cron Jobs", "Jobs", "Daemon Sets", "Replica Sets", and "Replication Controllers". The main content area is titled "Virtualization" and "Virtual Machines". It displays a list of running virtual machines with columns for Name, Namespace, Status, Created, Node, and IP Address. The list includes:

Name	Namespace	Status	Created	Node	IP Address
VM fedora01	NS default	Running	Jul 9, 5:00 pm	N worker-0.owv.lab.lan	10.131.0.74
VM rhel	NS default	Running	Jul 8, 4:18 pm	N worker-0.owv.lab.lan	192.168.14.163/24, fe80::87cc:48e1%e2: 9d23/64
VM rhel01	NS default	Off	Jul 9, 4:58 pm		
VM windows2019	NS default	Running	Jul 9, 5:01 pm	N worker-1.owv.lab.lan	10.128.2.52

# Modernize existing virtual machine workloads

Supports VMware & Red Hat Virtualization (now), Red Hat OpenStack Platform (later)

The screenshot shows the Red Hat OpenShift Container Platform web interface. On the left, there's a sidebar with navigation links: Home, Dashboards, Projects, Search, Explore, Events, Operators, Workloads (which is currently selected), Pods, Virtual Machines, Virtual Machine Templates, and Deployments. The main content area has a header 'Import Virtual Machine' and a sub-header 'You are logged in as a temporary administrative user. Update the password if necessary.' Below this, it says 'Project: default'. The workflow steps are listed on the left: 1 General (selected), 2 Networking, 3 Storage, 4 Advanced, Cloud-init, 5 Review, and 6 Result. On the right, configuration fields include 'Provider' set to 'VMware', 'vCenter instance' set to 'administrator-10.8.58.136-g77hr', 'VM or Template to Import' set to 'v2v-rhel7-mini', and 'Operating System' with a dropdown menu showing '--- Select Operating System ---' and a note 'Select matching for: Red Hat Enterprise Linux 7 (64-bit)'.

1. Run Red Hat Migration Analytics from [cloud.redhat.com](http://cloud.redhat.com) to select best candidates for import to OpenShift
2. For single VMs, run the Import Virtual Machine workflow from OpenShift
3. For multiple VMs, leverage the Red Hat Migration tool for batch import (longer term)
4. Import as a single VM or create a template for multiple ones



# OpenShift Security



## CONTROL

Application  
Security



## DEFEND

Infrastructure

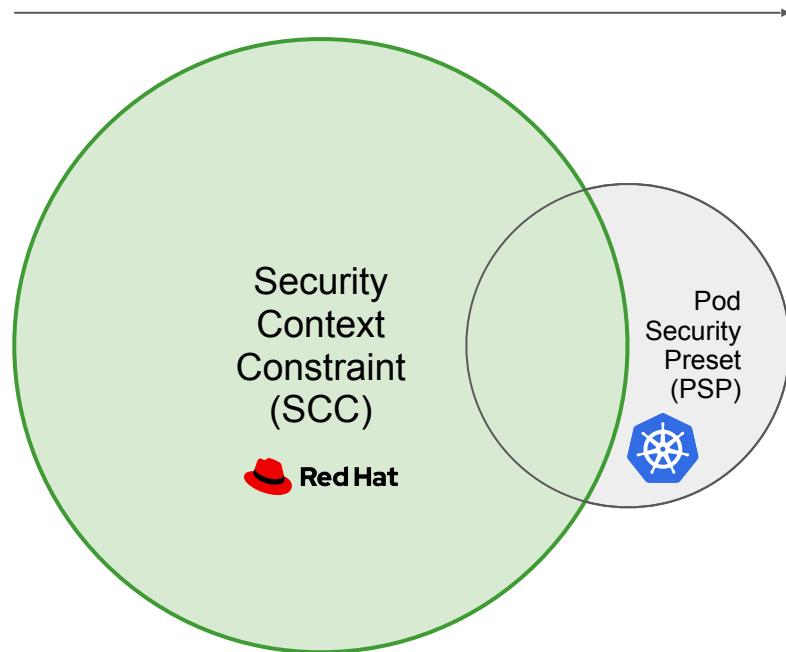


## EXTEND

Container Content	CI/CD Pipeline
Container Registry	Deployment Policies
Container Platform	Container Host Multi-tenancy
Network Isolation	Storage
Audit & Logging	API Management
Security Ecosystem	

# Extended Depth of Protection

Feature Transfer (upstream)



Feature Development (joint)

# Certificates and Certificate Management

- OpenShift provides its own internal CA
- Certificates are used to provide secure connections to
  - master (APIs) and nodes
  - Ingress controller and registry
  - etcd
- Certificate rotation is automated
- Optionally configure external endpoints to use custom certificates



# Container Security Operator (CSO) and Quay

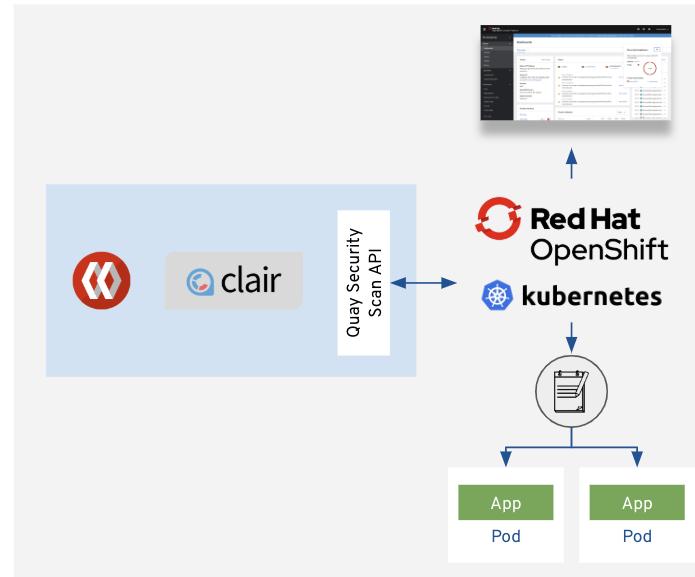
## What is it?

- Container Security Operator (CSO) runs on OpenShift and watches pod objects

## Why do you need it?

- Easily visualize image security vulnerability data from Quay / Clair

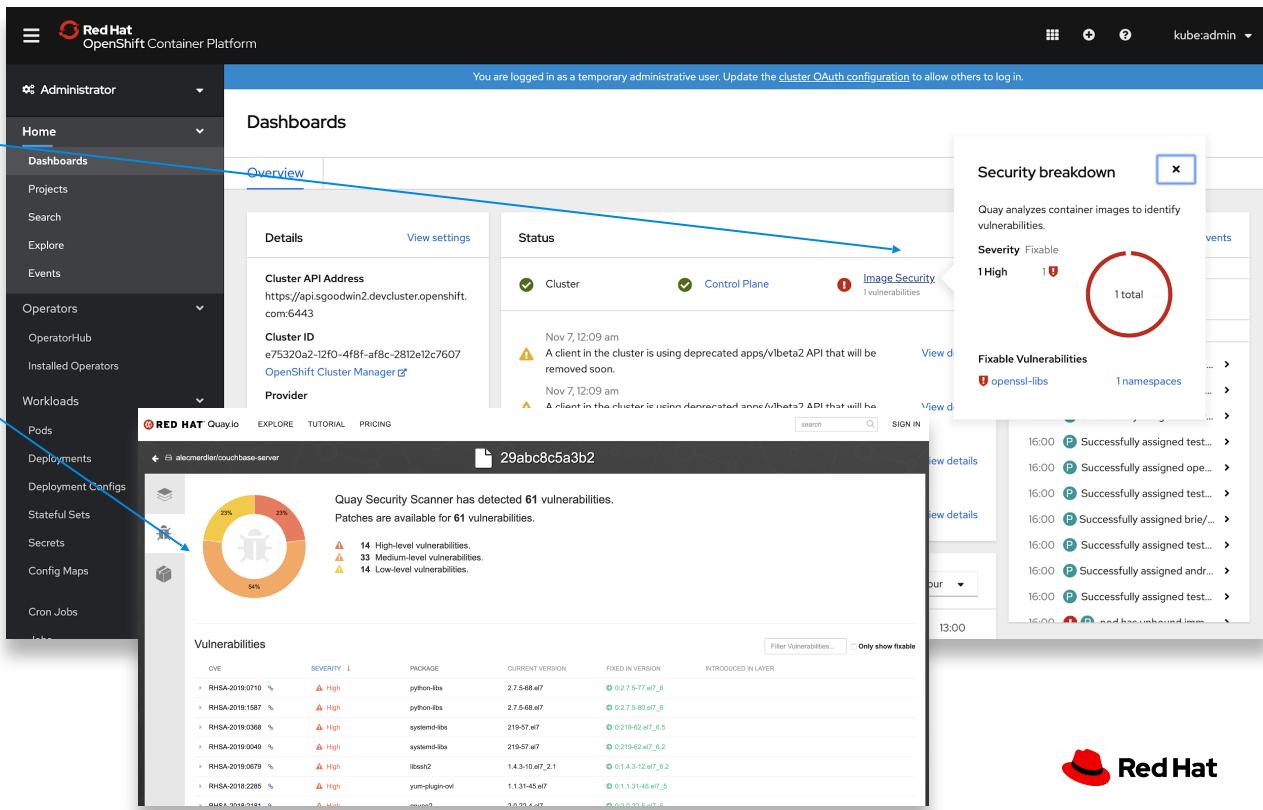
## How does it work?



# View Image Security Vulnerabilities in OpenShift

**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*

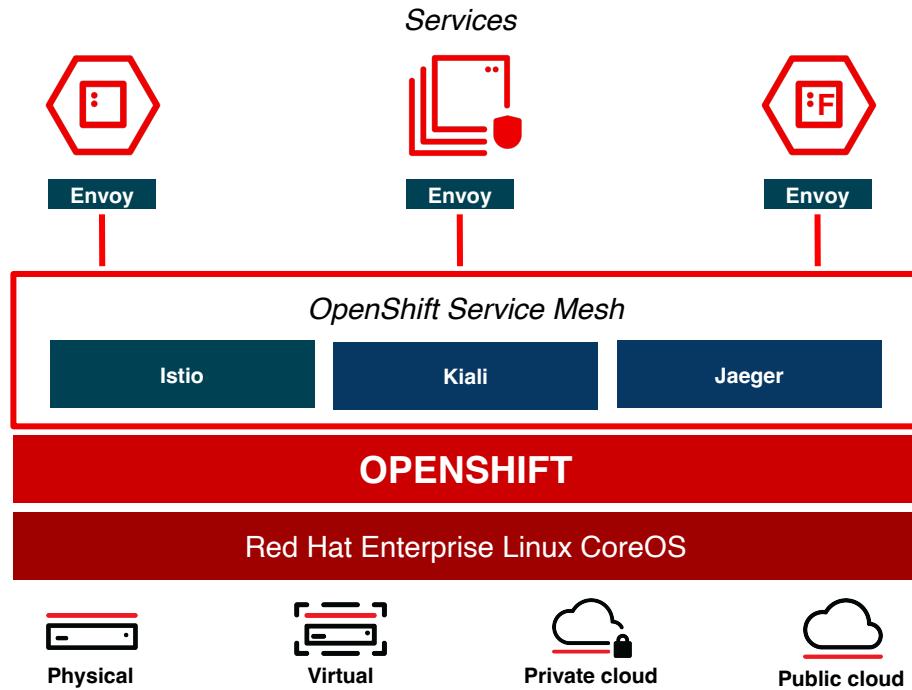




# OpenShift Developer Services

# Openshift Service Mesh

- Connect services securely with zero-trust network policies.
- Automatically secure your services with managed authentication, authorization and encryption.
- Control traffic to safely manage deployments, A/B testing, chaos engineering and more.
- See what's happening with out of the box distributed tracing, metrics and logging.
- Manage OpenShift Service Mesh with the **Kiali** web console.

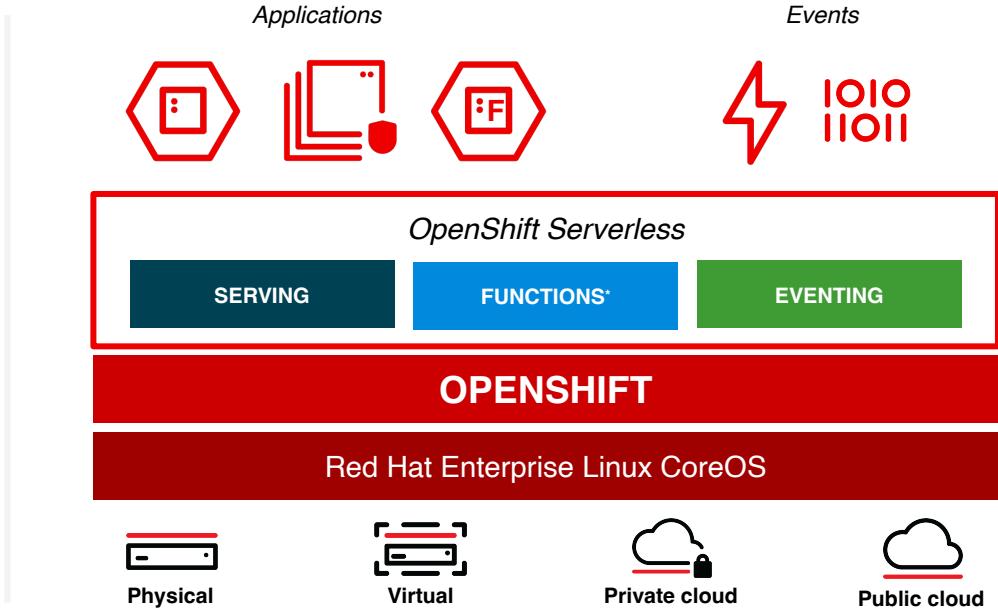




# OpenShift Serverless

Event-driven serverless containers and functions

- ▶ Deploy and run **serverless containers**
- ▶ Use any programming language or runtime
- ▶ Modernize existing applications to run serverless
- ▶ Powered by a rich ecosystem of event sources
- ▶ Manage serverless apps natively in Kubernetes
- ▶ Based on open source project **Knative**
- ▶ Run anywhere OpenShift runs



\* Functions are currently a work in progress initiative

# Cloud-native CI/CD with OpenShift Pipelines

Tech Preview



- Pipeline templates for serverless when importing application (+Add)
- Pipeline templates use workspaces instead of PipelineResources
- Default workspace per PipelineRun or globally
- Expanded Task library
  - Helm tasks
  - Skopeo tasks
  - Trigger Jenkins jobs from Tekton
- Support for disconnected clusters
- Pipeline metrics in cluster monitoring
- Pipeline Quick Start tours in Dev Console
- Enhancements in Tekton CLI: workspaces, results

The screenshot shows the Red Hat OpenShift Container Platform Dev Console interface. At the top, there's a navigation bar with 'Project: sm1-stage' and 'Application: all applications'. Below it, a sidebar has 'Topology' selected. The main area is titled 'Topology' and says 'No resources found'. A message below it reads: 'To add content to your project, create an application, component or service using one of these options.' On the right, a 'Deploying an application with a pipeline' quick start tour is displayed, showing steps 1 and 2: 'Importing an application and associate it with a pipeline' and 'Exploring your application'. At the bottom, a blue banner says 'You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.' A 'Pipeline' section in the sidebar has a 'Select task' dropdown.



# CodeReady Workspaces

- Web-based Eclipse Che IDE
- Developer workspaces in pods
- Bundled development stacks
- Available in OperatorHub

The screenshot shows the CodeReady Workspaces interface. At the top, there's a browser bar with the URL <https://che.openshift.io/dashboard/#/ide/bmicklea/wksp-8k2>. Below it is a dark-themed IDE interface.

- Workspace View:** Shows a project structure for "web-java-spring (master)" with folders like src, main, java, target, and pom.xml, along with files README.md and pom.xml.
- Code Editor:** Displays a Java file named GreetingController.java. A tooltip is visible over the code, showing the completion of a method call: "String result = org.eclipse.che.examples.GreetingController.handleRequest(HttpServletRequest)".
- Terminal View:** Shows a terminal window titled "Terminal" with the command "top" running. The output includes system statistics: "top - 21:28:41 up 133 days, 7:17, 0 users, load average: 13.88, 6.37, 3.86", task counts, CPU usage, memory usage, and a PID list table.
- Machines View:** Shows a list of machines under "dev-machine". One machine, "Terminal", is selected and its terminal window is shown above.

# Podman and Buildah



podman

A docker-compatible  
CLI for containers

- Remote management API via Varlink
- Image/container tagging
- Advanced namespace isolation



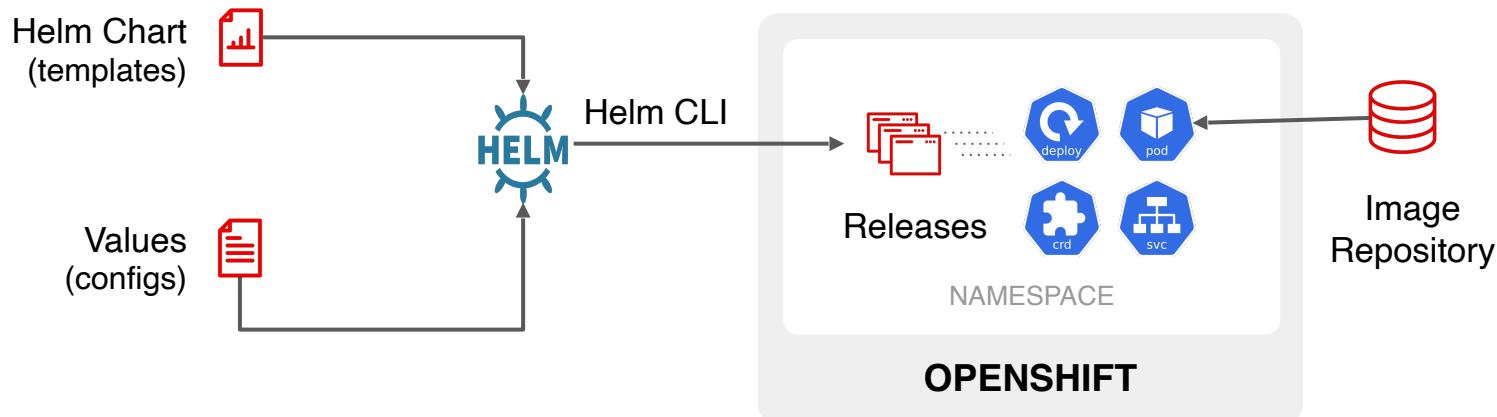
buildah

Secure & flexible OCI container builds

- Integrated into OCP build pods
- Performance improvements for knative enablement
- Image signing improvements

# Helm 3 on OpenShift

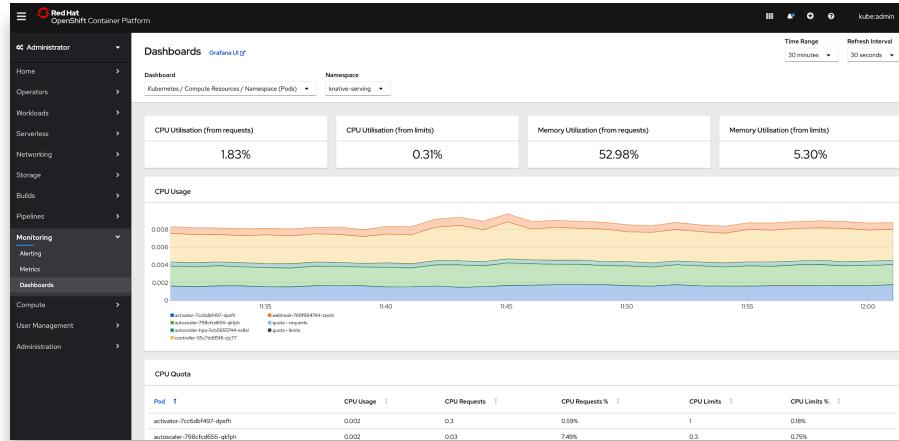
Helm is a package manager for Kubernetes applications and helps to define, install and update apps



# Metrics Dashboard

At-a-glance views for your mission critical OpenShift components right from the OpenShift Console.

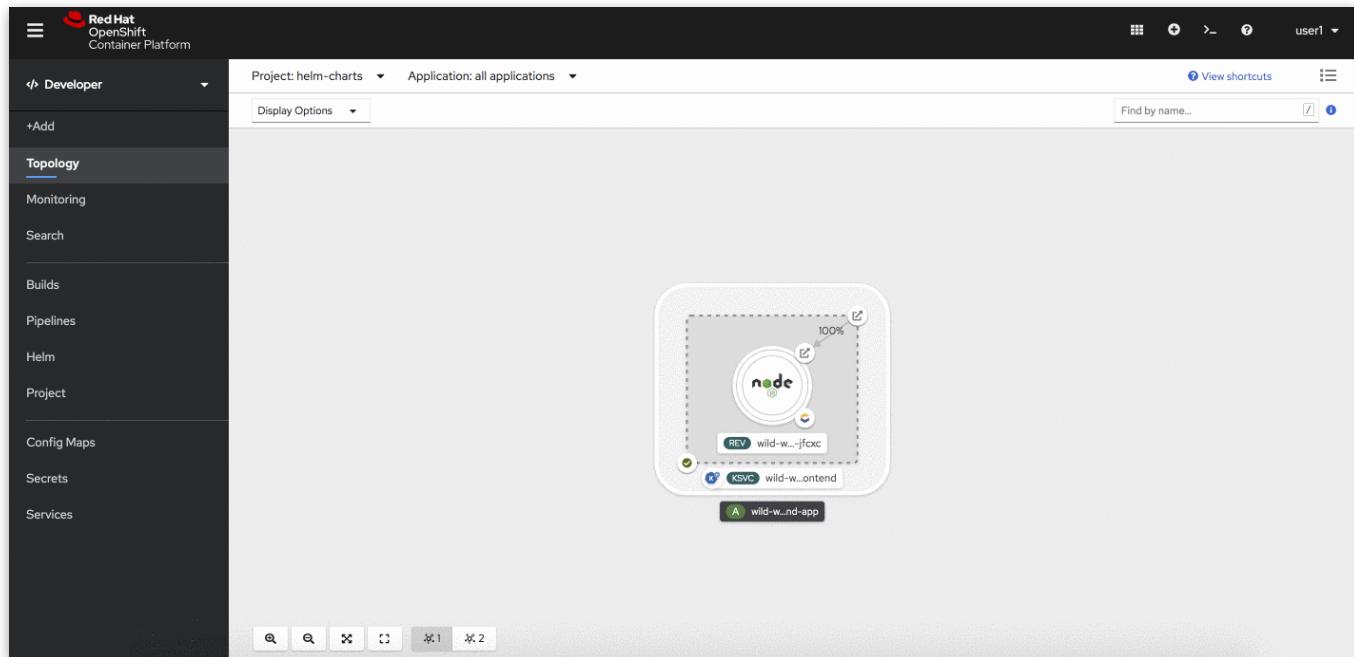
- Offers a single, centralized place for all OpenShift components to provide insights into critical KPIs back to you.
- Switch between dashboards, representing different parts of your cluster.
- Filter by different criterias defined by a dashboards and different time ranges.



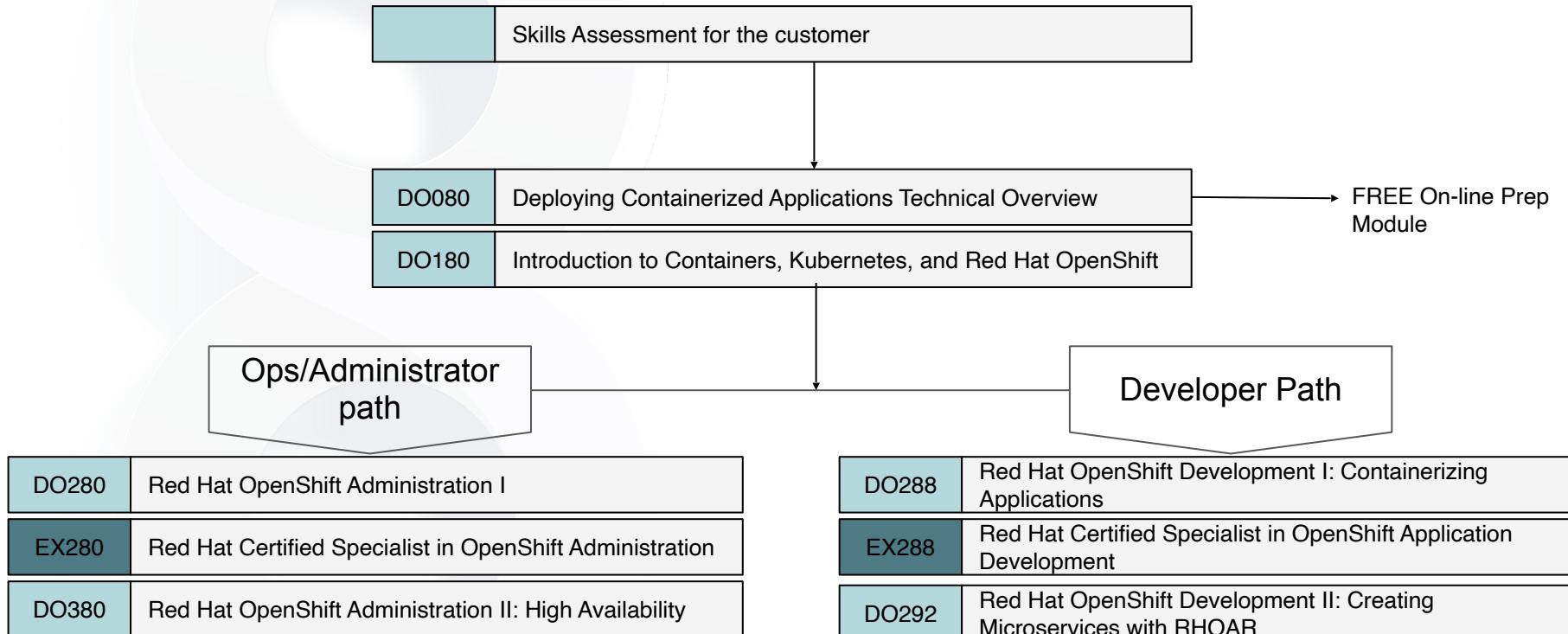
# Easy access to Web Terminal

**Use CLIs direct from web console, and fully authenticated when you need them.**

- oc, odo, kubectl
- Terminal can be minimized and opened in new tab
- History is not retained between sessions
- Not supported for cluster admins
- Operator enabled
- Available in first z stream



# Unlock the potential - get skilled



# RED HAT LEARNING SUBSCRIPTION

## BASIC, STANDARD & DEVELOPER



	Online Red Hat courses	Hands-on labs	Instructor videos	Video courses	Expert seminars	Instructor office hours	Learning paths	Certification exams
BASIC	X	X	X	X				
DEVELOPER	X	X	X	X				X
STANDARD	X	X	X	X	X	X	X	X

### Simple, flexible, on-demand training

- Annual subscription offering, no prerequisite required
- On-demand access, 24 hours per day, 7 days per week, worldwide
- Includes Red Hat online and video Training courses and hands-on labs
- Interactive learning interface allows users and managers to track progress

# LEARN.OPENShift.COM

## Foundations of OpenShift

[START COURSE](#)

## Building Applications On OpenShift

[START COURSE](#)

## Subsystems, Components, and Internals

[START COURSE](#)

## OpenShift Playgrounds

[START COURSE](#)

## Service Mesh workshop with Istio

[START COURSE](#)

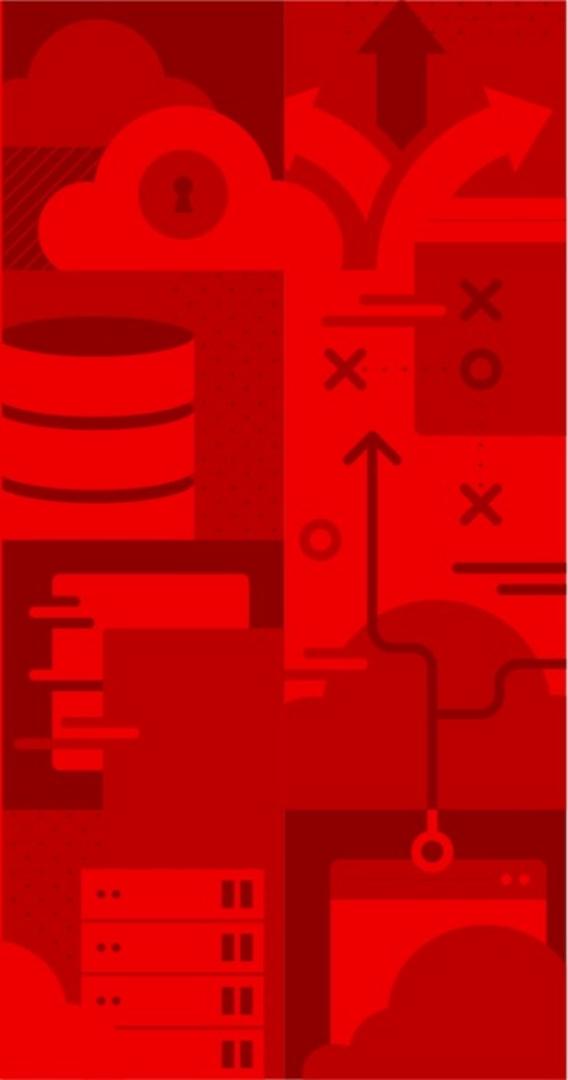
## Serverless scenarios with OpenShift Cloud Functions

[START COURSE](#)

Interactive Learning Scenarios provide you with a pre-configured OpenShift instance, accessible from your browser without any downloads or configuration.



# Demo



# 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)



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



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



[twitter.com/RedHat](https://twitter.com/RedHat)