



# OpenShift e Container Storage

KUBERNETES ENTERPRISE PER LE GRANDI IDEE

**PIERLUIGI SFORZA**

Solution Architect

psforza@redhat.com



#RedHatOSD



... so you want to do  
containers and Kubernetes?



#RedHatOSD



# RED HAT HAS BEEN A **KUBERNETES LEADER** SINCE DAY 1



CoreOS

K8S 1.0



K8S 1.0

We were very lucky to be joined early on by the very capable OpenShift team ... without their perspective and contributions, I don't think we would be standing here today



*Brendan Burns, co-creator of Kubernetes*



K8S 1.6



MESOSPHERE

K8S 1.8



K8S 1.9



2015

2016

2017

2018



#RedHatOSD



# OPENSIFT IS KUBERNETES FOR THE ENTERPRISE

Kubernetes  
Release



**1-3 months  
hardening**

OpenShift  
Release



Security fixes  
100s of defect and performance fixes  
200+ validated integrations  
Middleware integrations  
(container images, storage, networking, cloud services, etc)  
9 year enterprise lifecycle management  
Certified Kubernetes



#RedHatOSD





# Kubernetes Workloads

MANAGE YOUR APPLICATIONS

**PIERLUIGI SFORZA**

Solution Architect  
psforza@redhat.com

**ALESSANDRO ARRICHIELLO**

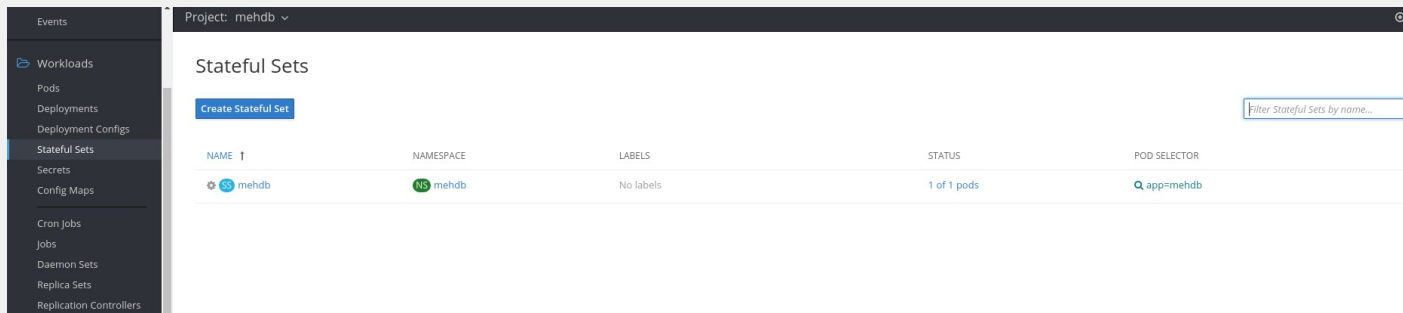
Solution Architect  
ale@redhat.com



#RedHatOSD

# CONTROLLERS MATTERS!

Different types of applications (stateful, stateless, batch, agent, ...) require different orchestrator behaviors



The screenshot shows the OpenShift console interface. On the left is a navigation sidebar with options like Workloads, Pods, Deployments, Stateful Sets, etc. The main area displays 'Stateful Sets' for the 'mehdb' namespace. A table lists one Stateful Set named 'mehdb' with 'No labels', a status of '1 of 1 pods', and a pod selector 'app=mehdb'. A 'Create Stateful Set' button is visible at the top left of the main area.

Main controller types:

- Replica Sets
- Stateful Sets
- Daemon Sets
- Jobs (OneTime, Cron)



```
func NewControllerInitializers(loopMode ControllerLoopMode) map[string]
    controllers := map[string]InitFunc{}
    controllers["endpoint"] = startEndpointController
    controllers["replicationcontroller"] = startReplicationControl
```





# WHAT IS A POD?





# IO POD E TU?



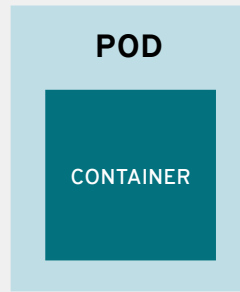
**Dash PODS 3in1**  
Rivoluziona il tuo bucato



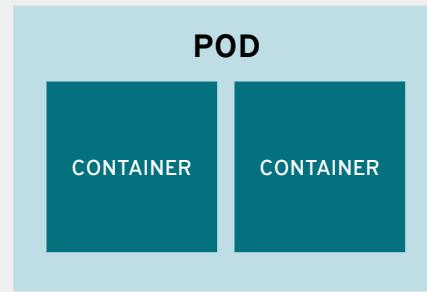
Tenere fuori dalla portata dei bambini. [www.keepcapsfromkids.eu](http://www.keepcapsfromkids.eu)



# CONTAINERS ARE WRAPPED IN PODS WHICH ARE UNITS OF DEPLOYMENT AND MANAGEMENT



IP: 10.1.0.11

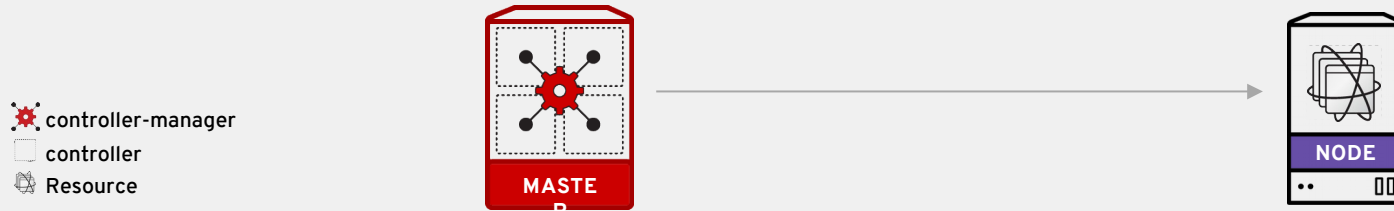


IP: 10.1.0.55



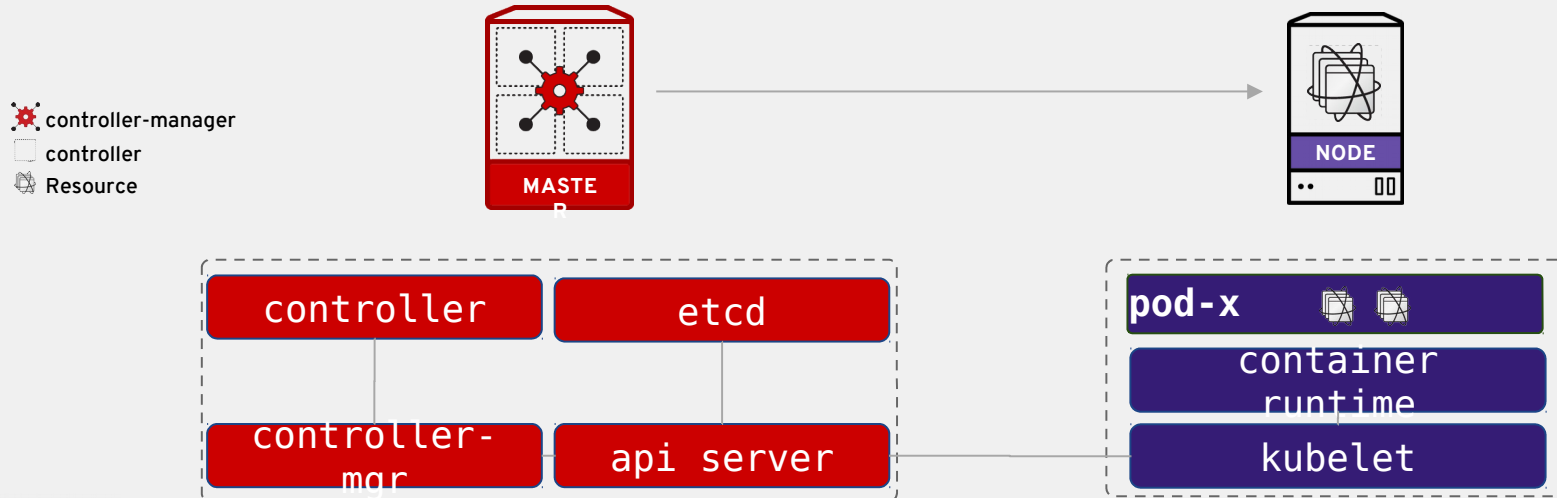
# CONTROLLER & CONTROLLER-MANAGER

- The **controller-manager** is the Master's component that manage the controllers
- A **controller** is a loop that governs the status of kubernetes resources (such as pods) in order to bring it from the current state to the desired state
- Controllers react to **kubernetes events** and define how resources should be orchestrated



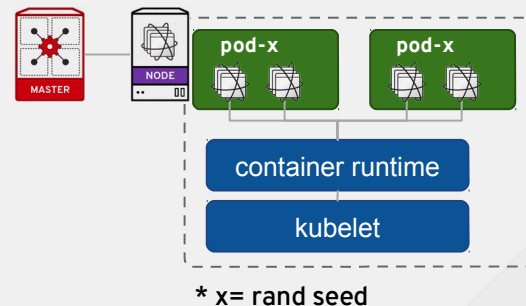
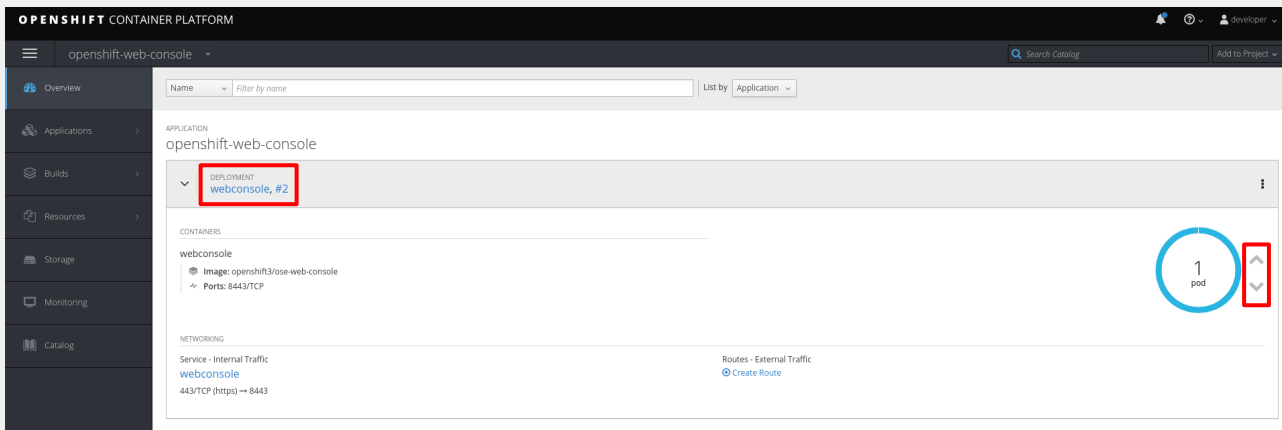
# CONTROLLER & CONTROLLER-MANAGER

- The **controller-manager** is the Master's component that manage the controllers
- A **controller** is a loop that governs the status of kubernetes resources (such as pods) in order to bring it from the current state to the desired state
- Controllers react to **kubernetes events** and define how resources should be orchestrated



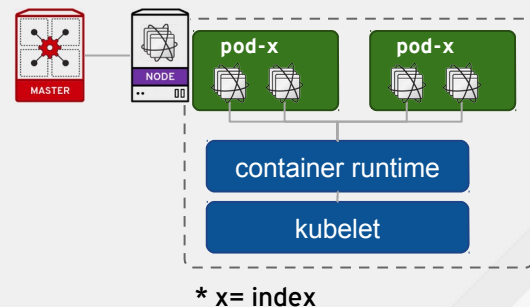
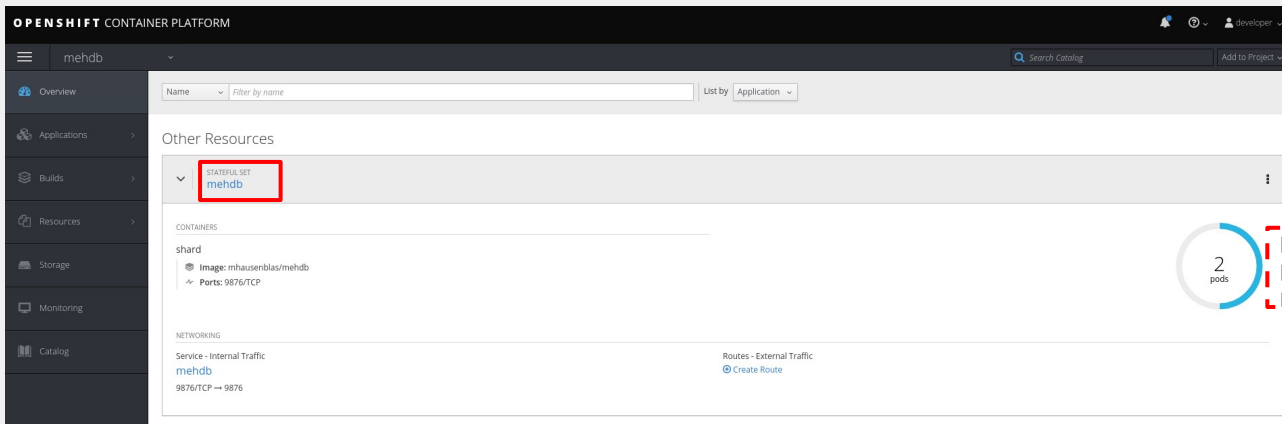
# DEPLOYMENT AND REPLICASET

- A Deployment controller provides declarative updates for Pods and ReplicaSets
- ReplicaSet controller ensures that a specified number of pod replicas are running at any given time
- Recommended to run stateless application



# STATEFULSET

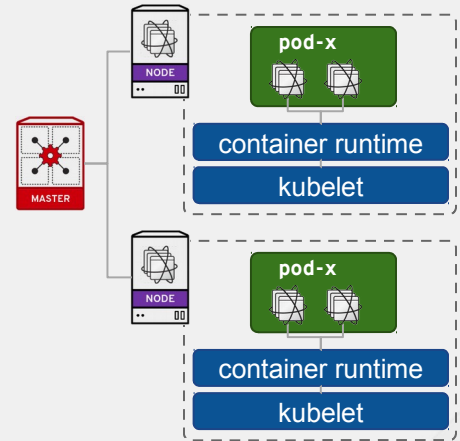
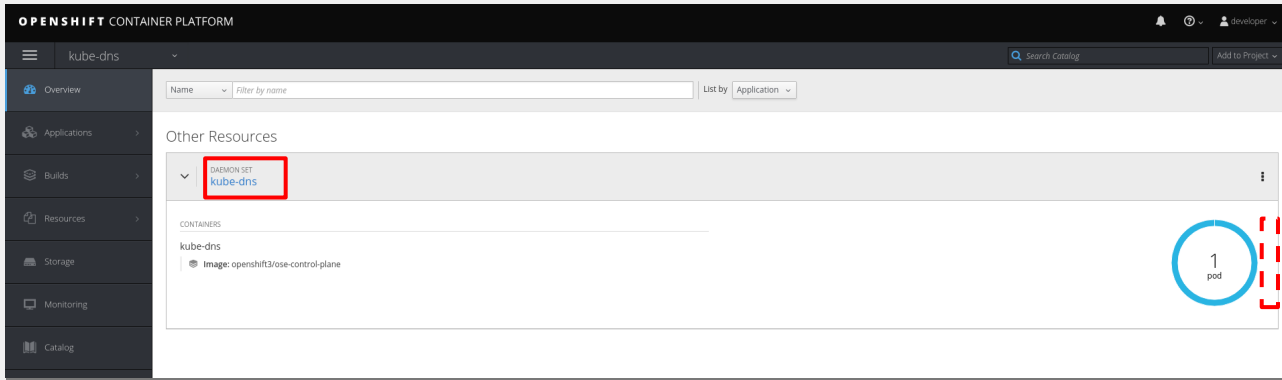
- A stateful set ensure
  - stable resource allocation such as name and storage
  - ordered, graceful deployment, scaling up and termination
- ideal for highly available workloads in a “clustered mode”





# DAEMONSET

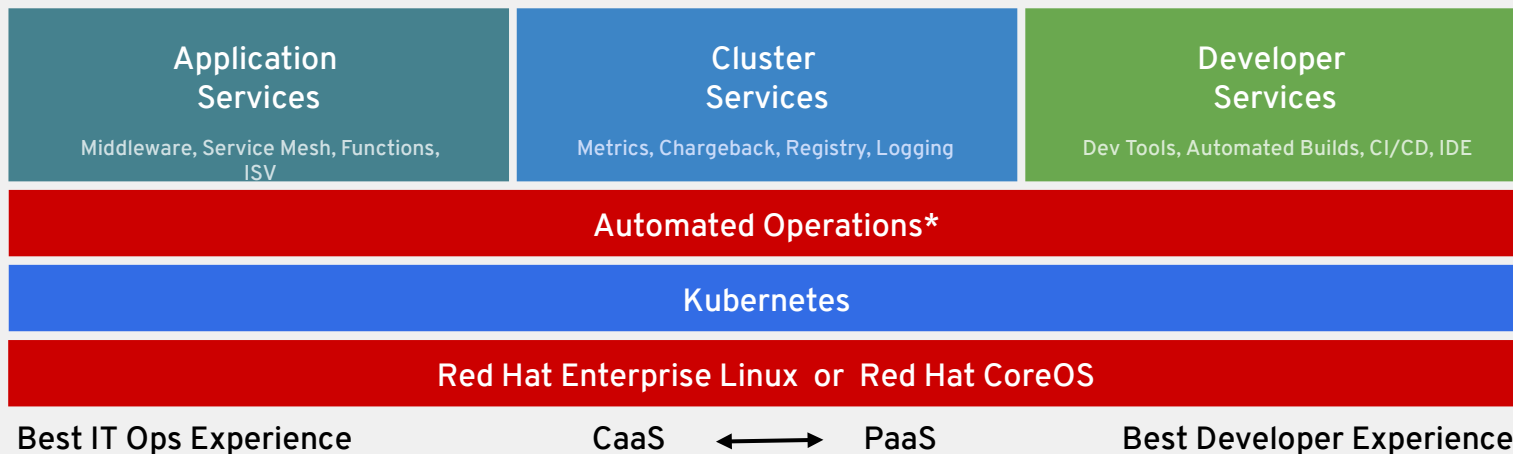
- A daemon set ensure to have **just 1 copy** of a pod on every node
- Daemon set is useful for: Logging Aggregators, Monitoring, Load Balancers / Reverse Proxies / API Gateways, single host batch...



\* x = available node count



# REFERENCE ARCHITECTURE FOR ENTERPRISE KUBERNETES



\*coming soon with OCP 4.0  
(targeted for GA Dec 2018)





# Istio Service Mesh

FOR SERVICE-TO-SERVICE COMMUNICATIONS

**NATALE VINTO**

Specialist Solution Architect

[nvinto@redhat.com](mailto:nvinto@redhat.com)



**#RedHatOSD**

# OPENSIFT SERVICE MESH: ISTIO\*

Istio makes it easy to create a network of deployed services with load balancing, service-to-service authentication, monitoring, and more, helping to avoid operational nightmares.

## POLICY

Grants the ability to write policy that applies to all applications and is not language specific

## ROUTING

Allows for the control of routing flows

## TELEMETRY

Provides the observability needed to manage microservices, such as how services are invoked, communication flows, and points of latency

\* Technology Preview



# ISTIO COMPANION: KIALI & JAEGER

Kiali and Jaeger make the perfect companion for Istio Service Mesh

## VISUALIZATION

Kiali works with Istio to visualize the service mesh topology, features like circuit breakers or request rates.

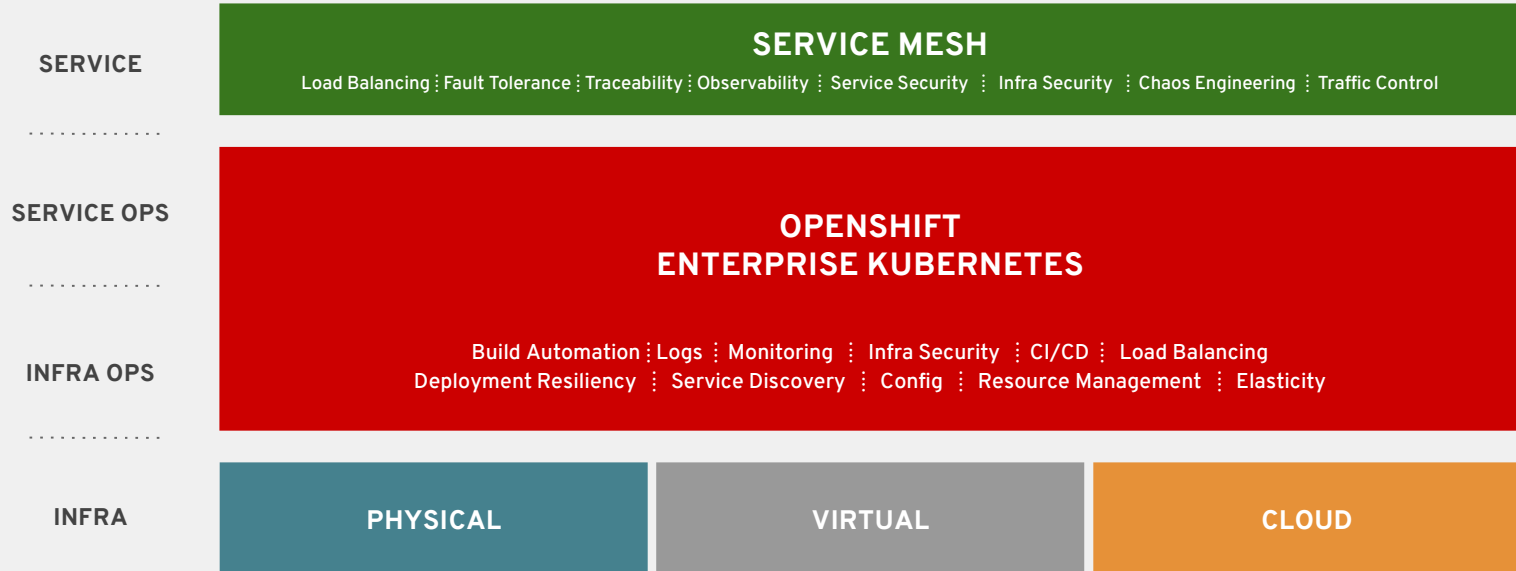
## TRACING

Kiali includes **Jaeger** Tracing, which provides distributed tracing out of the box.

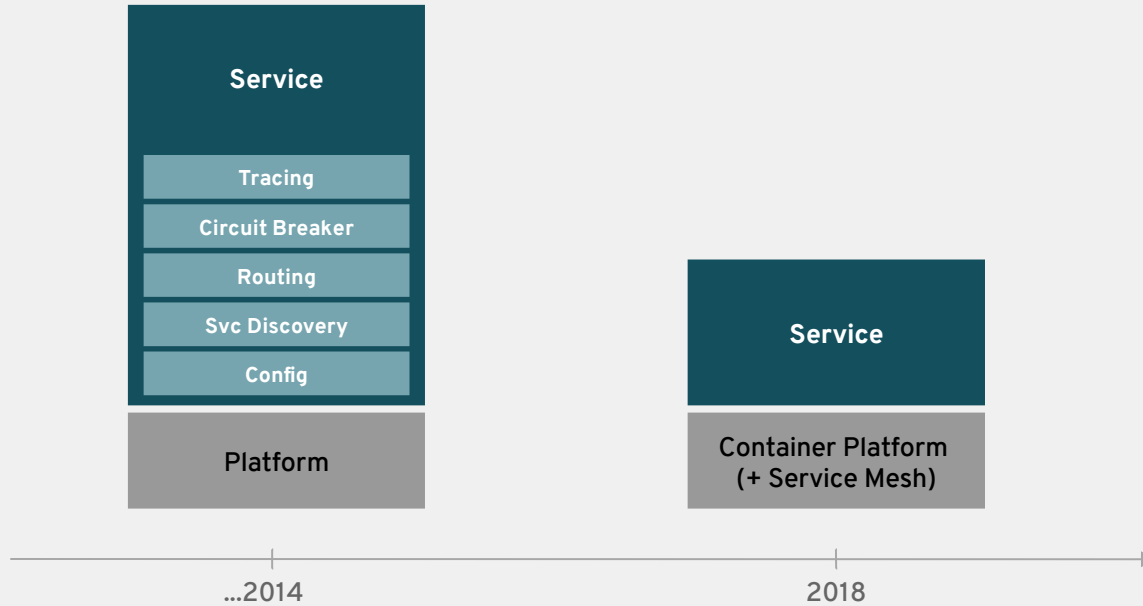




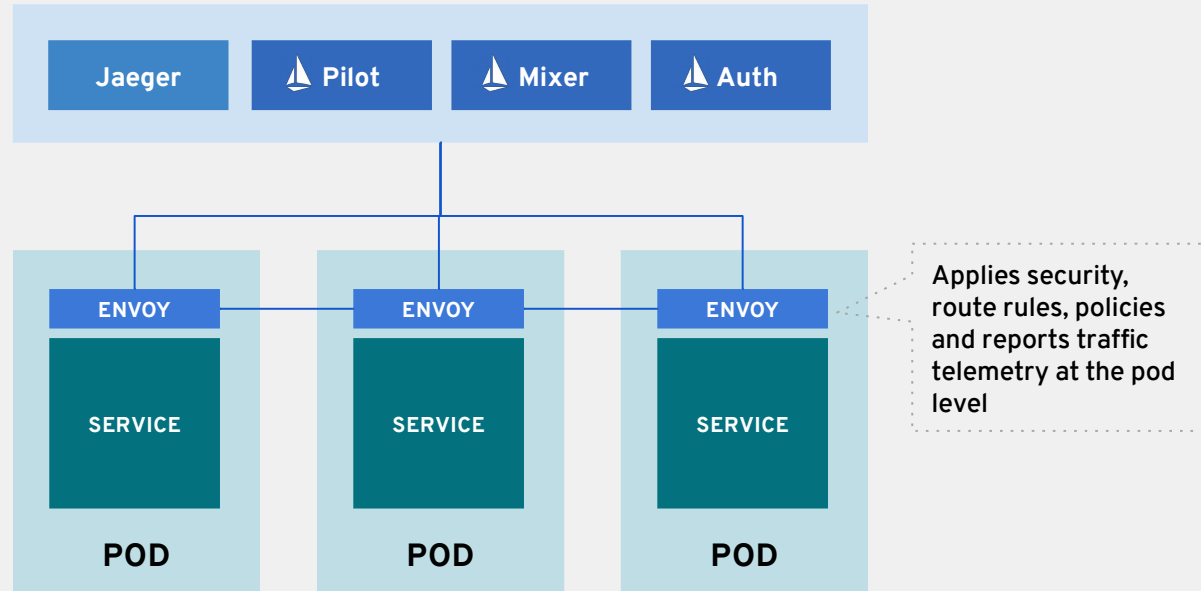
# SERVICE MESH ARCHITECTURE



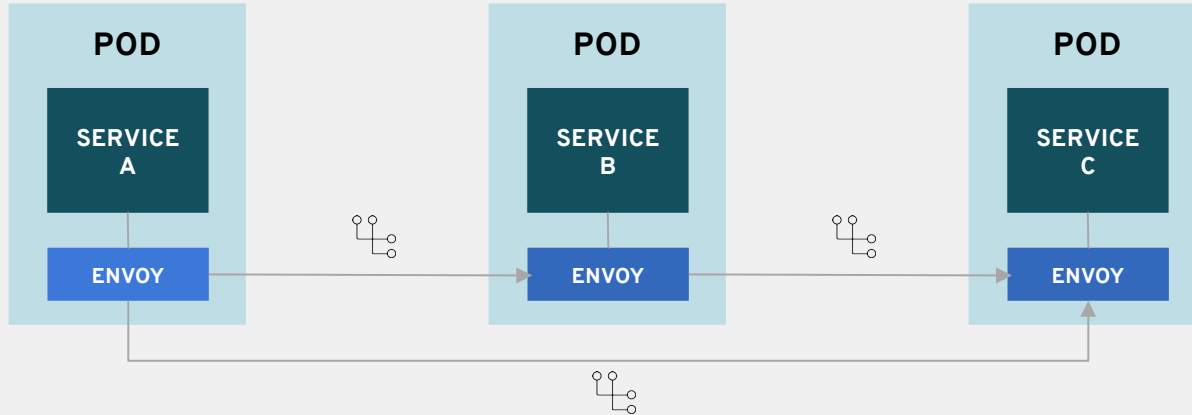
# MICROSERVICES EVOLUTION



# SERVICE MESH ARCHITECTURE



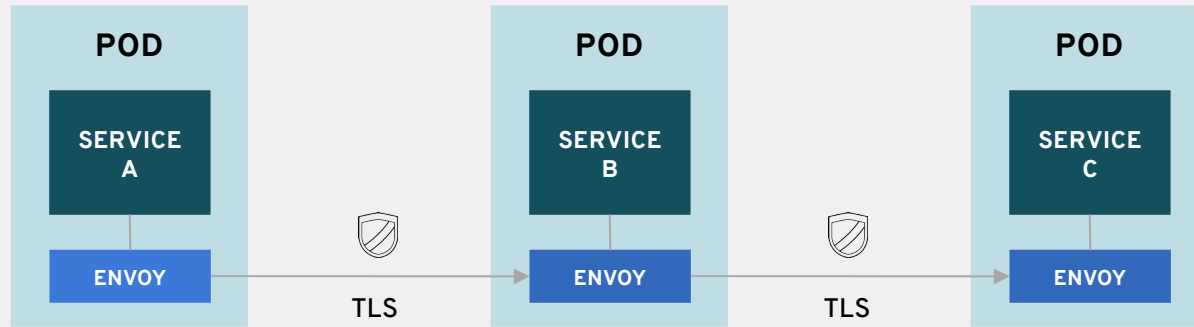
# CIRCUIT BREAKERS WITH ISTIO



transparent to the services



# SECURE COMMUNICATION WITH ISTIO

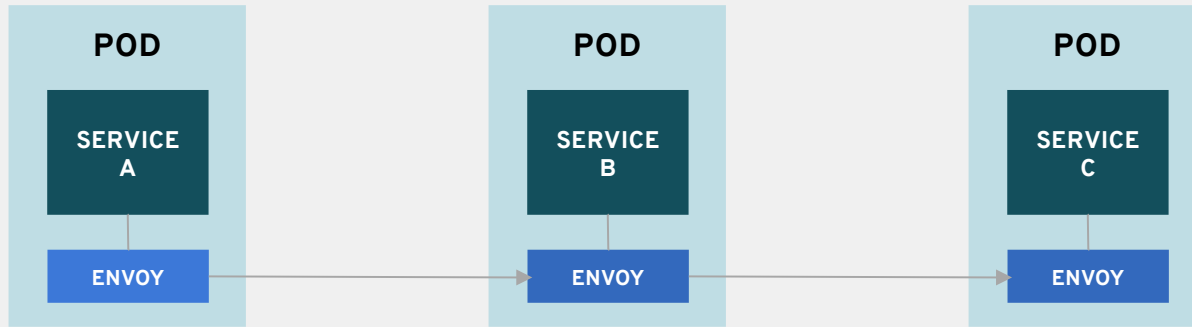


mutual TLS authentication, transparent to the services





# DISTRIBUTED TRACING WITH ISTIO & JAEGER



discovers service relationships and process times,  
transparent to the services



# **DEMO TIME: Istio Internals**

YouTube video player interface showing a video player, description, and related videos.

**Video Title:** [https://youtu.be/\\_SKOXBaKgk8](https://youtu.be/_SKOXBaKgk8)

**Video Description:** [Faded text]

**Related Videos:** [List of video thumbnails and titles]



# Prometheus Cluster Monitoring

PROVIDING ALERTS ALSO FOR OPENSIFT  
CONTAINER STORAGE

**NATALE VINTO**

Specialist Solution Architect  
nvinto@redhat.com

**CARLOS TORRES**

Specialist Solution Architect  
ctorres@redhat.com



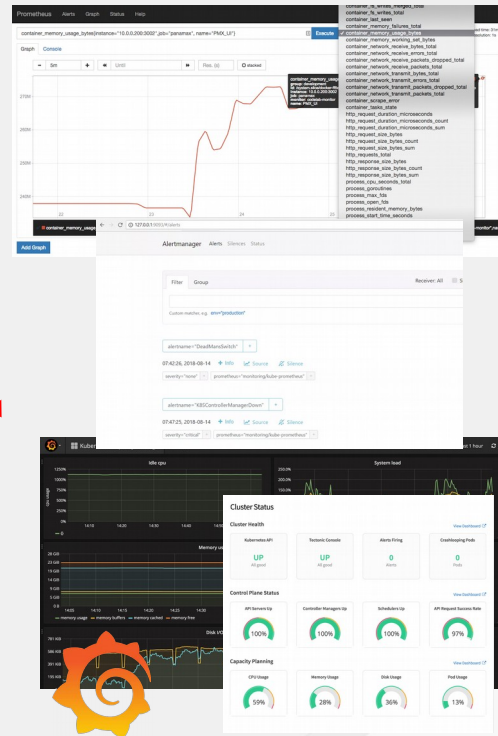
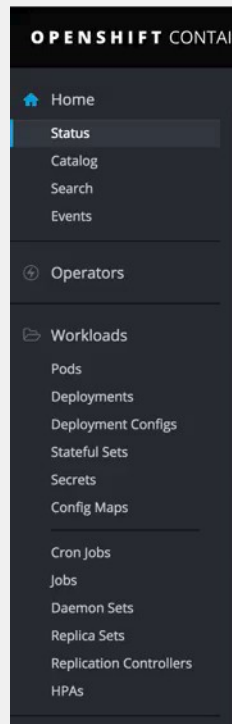
#RedHatOSD

# COMPREHENSIVE MONITORING SUITE

The stack includes three distinct UIs:

- **Alertmanager** UI to manage alerts which have been fired
- **Prometheus** UI for querying and plotting any metrics
- **Grafana** to browse cluster-level dashboards

All UIs are accessible directly via the new admin console under the “Monitoring” menu.



**DEMO TIME:**  
**Cluster Console - EventFeed**

- Home
- Operators
- Workloads
- Pods
- Deployments
- Deployments Configs
- Services
- Config Maps
- Daemon Sets
- Jobs
- Horizontal Pod Autoscalers
- Vertical Pod Autoscalers
- Network Policy
- Network
- Storage

Project: redhat-test

## Deployments

Create Deployment

Filter Deployments by Name

NAME	HEALTHY	LABELS	STATUS	REPLICATOR
app-01	1/1	app-01	1 of 1 pods	app-01
app-02	2/2	app-02	2 of 2 pods	app-02
app-03	1/1	app-03	1 of 1 pods	app-03
app-04	1/1	app-04	1 of 1 pods	app-04
app-05	1/1	app-05	1 of 1 pods	app-05
app-06	1/1	app-06	1 of 1 pods	app-06
app-07	1/1	app-07	1 of 1 pods	app-07
app-08	1/1	app-08	1 of 1 pods	app-08
app-09	1/1	app-09	1 of 1 pods	app-09
app-10	1/1	app-10	1 of 1 pods	app-10

<https://youtu.be/MG-2s11uoPI>

**DEMO TIME:**  
**Cluster Console - Monitoring**



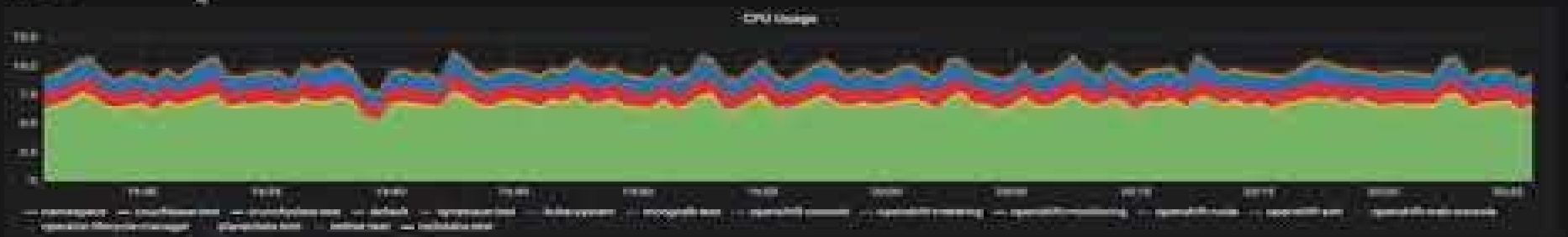
Cluster: prometheus

Headlines

CPU Requests Commitment	CPU Limits Commitment	Memory Requests Commitment	Memory Limits Commitment
54.12%	54.38%	30.6%	18.82%

<https://youtu.be/4aREWqD-V3c>

CPU



CPU Usage

Component	CPU Usage	CPU Requests	CPU Requests %	CPU Limits	CPU Limits %
prometheus	0.15	-	-	-	-
ingress-nginx	0.17	0.28	79.41%	0.43	40.70%
certification	0.07	0.42	276.92%	0.20	188.23%



# OpenShift Container Storage

SOFTWARE DEFINED STORAGE FOR YOUR  
KUBERNETES

**CARLOS TORRES**

Specialist Solution Architect  
ctorres@redhat.com



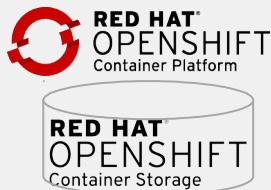
#RedHatOSD

# RED HAT OPENSIFT CONTAINER STORAGE

Flexible deployment with the same user experience and features

## Converged = in containers

Persona: DevOps, App Architects



- Highly scalable, scale app+storage, start small and scale fast
- Storage life cycle managed by OCP

## Independent = for containers

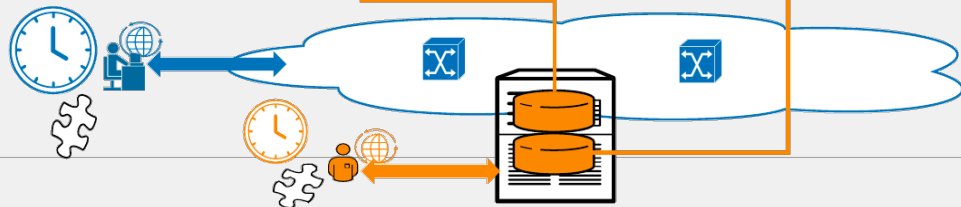
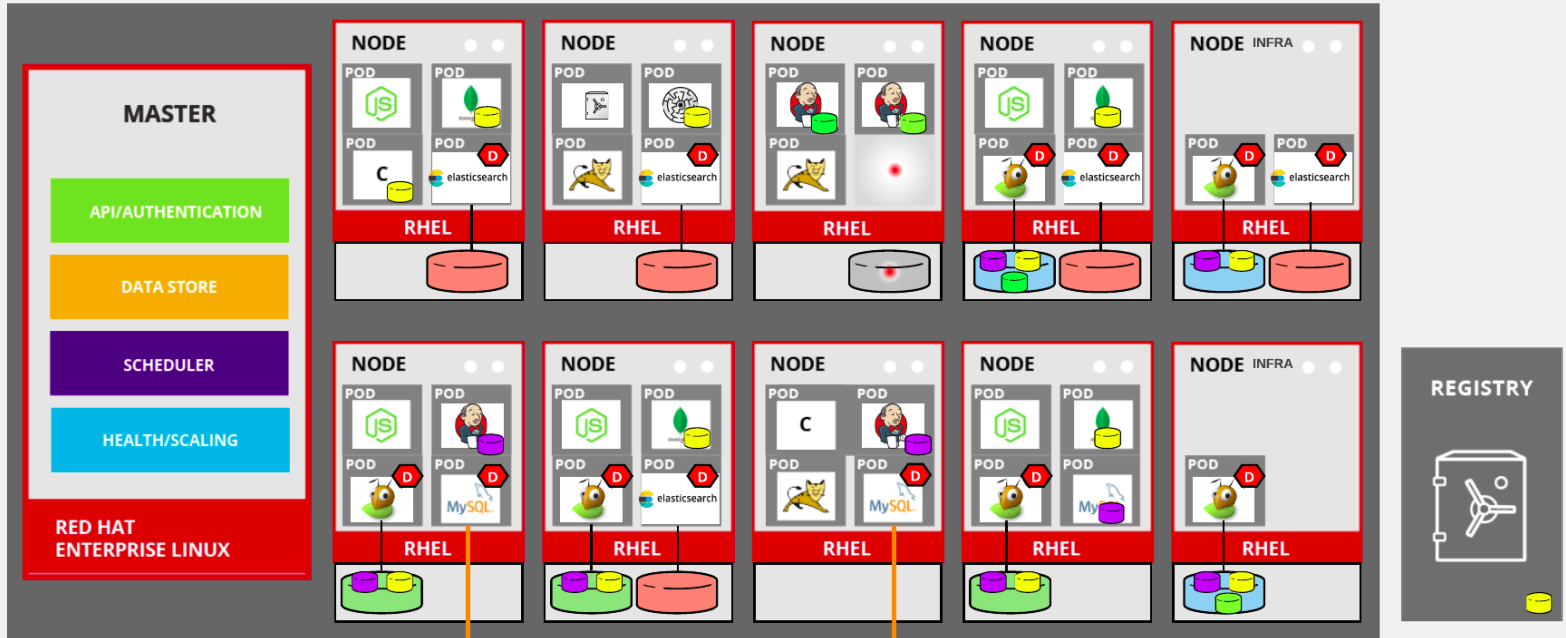
Persona: Storage Admins, Infrastructure Admins





- Highly scalable, independent scalability from OCP platform
- Adaptive to in-place BC/DR strategies



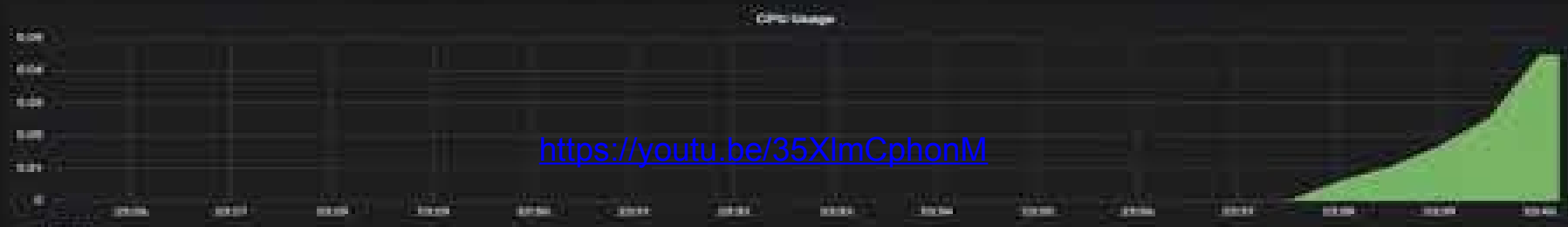
# OPENSSHIFT FULL INTEGRATION



-  glusterfs-fuse
-  gluster-block

**DEMO TIME:**  
**Monitoring - OpenShift Container  
Storage**

K8s / Compute Resources / Pod



<https://youtu.be/35XImCphonM>

CPU Quota

- Container
- Pod

CPU Quota

CPU Usage  
0.04

Memory Usage

Memory Usage

# RHOCS: ANSIBLE ADVANCED DEPLOYMENT

Converged playbooks already available

Deployment workflow	Registry	Metrics	Logging	Applications
Deploying Red Hat Openshift Container Storage in Converged Mode				✓
Deploying Red Hat Openshift Container Storage in Converged Mode with Registry	✓			
Deploying Red Hat Openshift Container Storage in Converged Mode with Logging and Metrics		✓	✓	
Deploying Red Hat Openshift Container Storage in Converged mode for Applications with Registry, Logging, and Metrics	✓	✓	✓	✓



<https://red.ht/2DaKPzg>



#RedHatOSD





# Openshift Ansible Service Broker

And the road to Kubernetes Operators!

**ALESSANDRO ARRICHELLO**

Solution Architect

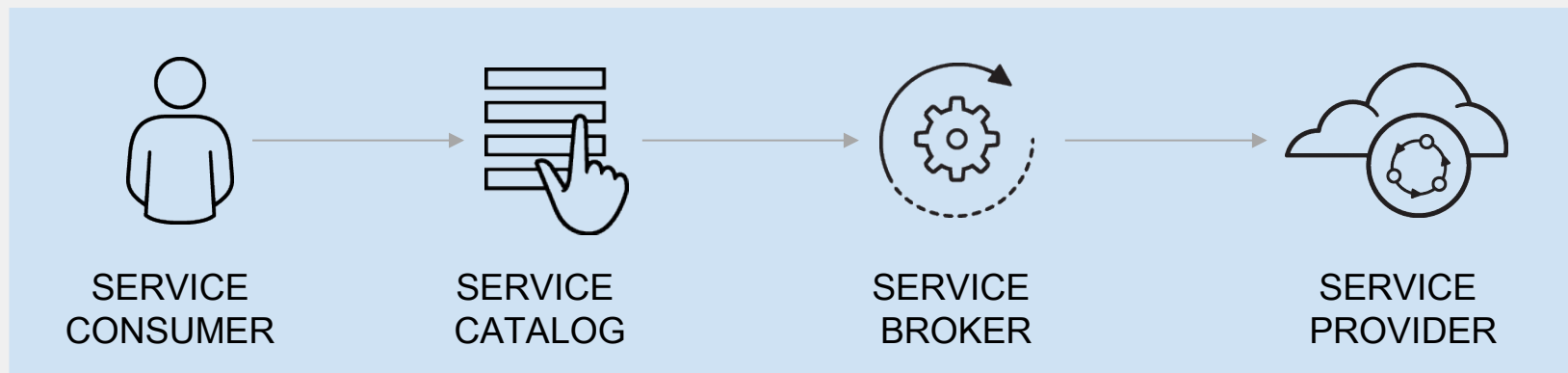
ale@redhat.com



#RedHatOSD



# WHAT IS A SERVICE BROKERAGE?



Automated, Standard and Consistent





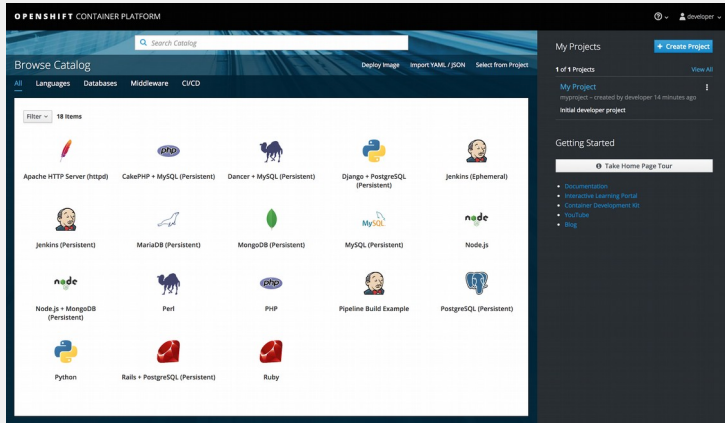
# OPEN SERVICE BROKER API™

A multi-vendor project to standardize how services are consumed on cloud-native platforms across service providers



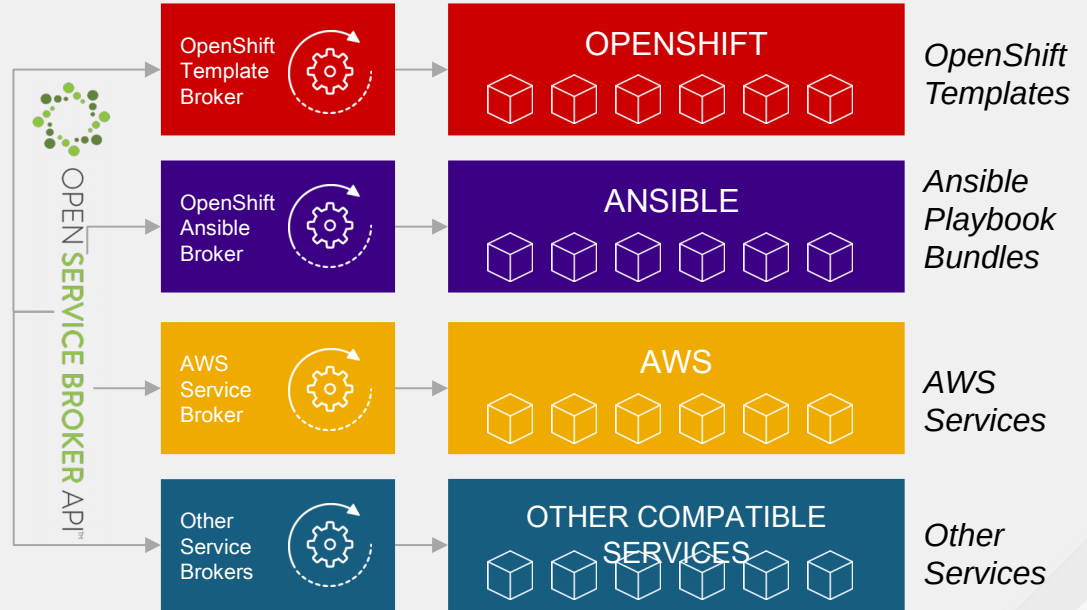
# BROKERAGE WITH OPENSHIFT

## SERVICE CATALOG



## Broker

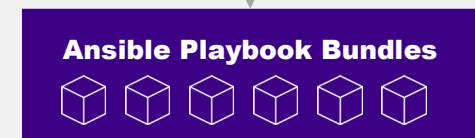
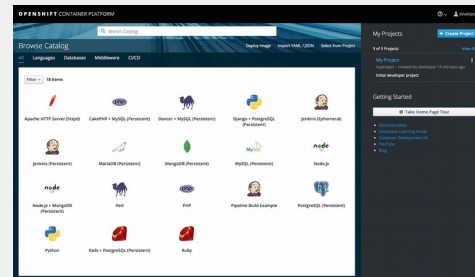
## Service



# OPENSIFT ANSIBLE BROKER

Anything you can do with Ansible, you can do with the Ansible Broker

- Use Ansible on OpenShift to
  - Deploy containerized applications
  - Manage external components (e.g. Oracle database)
  - Provision cloud services (e.g. AWS RDS)
  - Orchestrate multi-service solutions
  - Manage dependencies or other logics on deployments (e.g. database initialization)

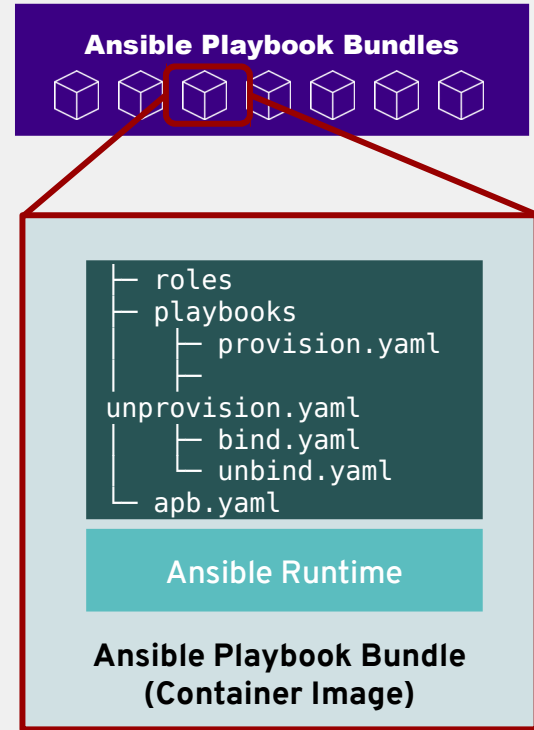


TP available from version 3.6  
GA from 3.7



# ANSIBLE PLAYBOOK BUNDLES (APB)

- Packaged as a container image
- Embed Ansible runtime
- Use named playbooks for actions
- Fulfill Service Catalog dynamically with services and parameters
- Provide a command line tool to manage APBs



# APB CREATION WORKFLOW



Site Reliability  
Engineer



playbooks and  
\$vars



APB image



Service Catalog  
update

INIT

CUSTOMIZATIO  
N

PREPARE AND  
BUILD

PUSH



<https://developers.redhat.com/blog/2018/05/23/customizing-an-openshift-ansible-playbook-bundle/>

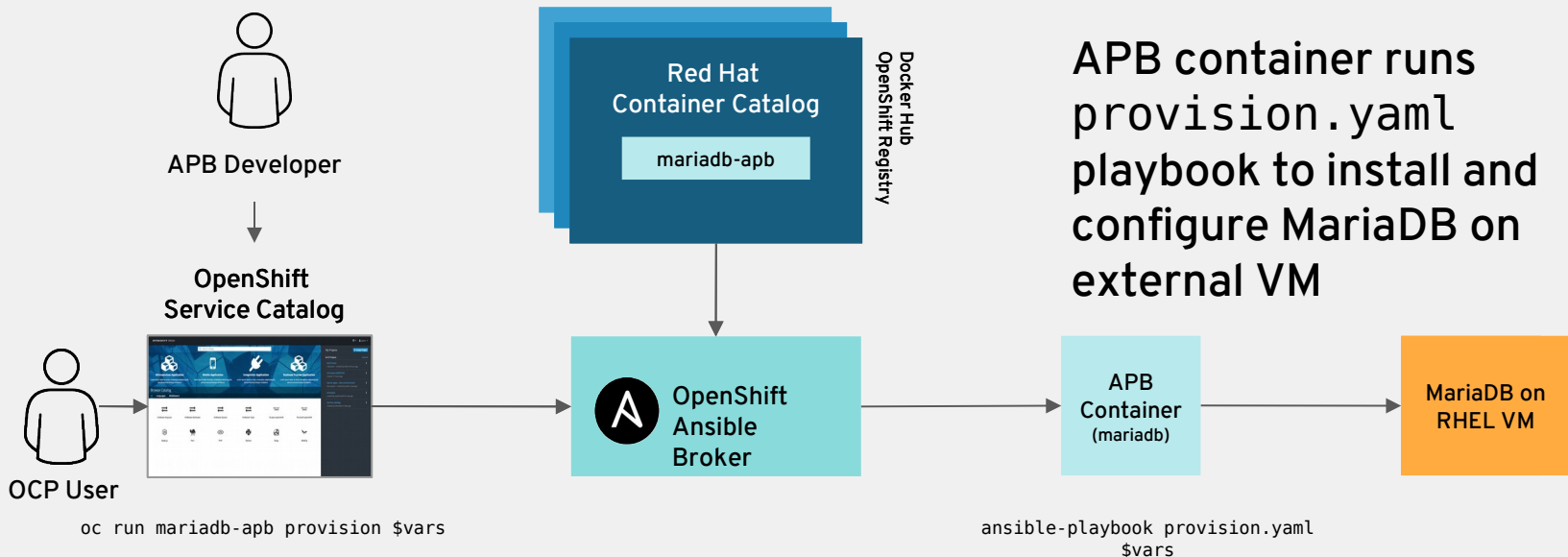


#RedHatOSD



**DEMO TIME:**  
**MariaDB Provisioning on Remote  
RHEL**

# OPENSHIFT APB MARIADB REMOTE PROVISIONING





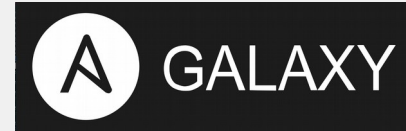
```
root@ip-10-0-1-10:~# kubectl get pods --all-namespaces
```

NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
default	docker-registry-2-4q1ak	1/1	Running	0	21d
default	registry-console-1-ds9qz	1/1	Running	0	21d
default	router-2-5m9cc	1/1	Running	0	20d
default	router-2-cfp92	1/1	Running	0	20d
default	router-2-fvj8l	1/1	Running	0	20d
kube-service-catalog	apiserver-g1gkc	1/1	Running	5	21d
kube-service-catalog	controller-manager-7w8fp	1/1	Running	10	21d
karatregistry-aw1ed6-deployment-web-prod-2kq7k	app-3cc890ef-d172-4b19-866c-3384191136a2	1/1	Running	0	22s
openshift-ansible-service-broker	api-server-1-49qg	1/1	Running	11	16d
openshift-ansible-service-broker	api-server-1-49qg	1/1	Running	5	21d
openshift-infra	kubernetes-controller-1-7ms78	1/1	Running	0	20d
openshift-infra	kubernetes-controller-2-f9zqs	1/1	Running	0	20d
openshift-infra	heapster-b14g8	1/1	Running	0	20d
openshift-template-service-broker	apiserver-t547r	1/1	Running	5	21d
openshift-template-service-broker	apiserver-w2aha	1/1	Running	5	21d
openshift-template-service-broker	apiserver-z9wqz	1/1	Running	5	21d
openshift-web-console	webconsole-c9cf77469-bzj18	1/1	Running	7	21d
openshift-web-console	webconsole-c9cf77469-j1j8d	1/1	Running	7	21d
openshift-web-console	webconsole-c9cf77469-ksnqt	1/1	Running	7	21d
test	my-app-1-buz1f	0/1	Completed	0	20m
test	my-app-1-xy9xg	1/1	Running	1	20m

```
root@ip-10-0-1-10:~#
```

# APB INTEGRATION WITH ANSIBLE GALAXY

Support discovering/running APB sources published to [Ansible Galaxy](#) from the OpenShift Ansible Service Broker.



## How it works:

- APB's can be now be created right from `mazer` command line tool using the `init` command and then pushed to Ansible Galaxy.
- Broker should now be able to discover and provision APB-based services published to Ansible Galaxy and also make them available in the service catalog.



# What's Next? Operators!

# KUBERNETES OPERATORS

THE EASE OF THE CLOUD EVERYWHERE

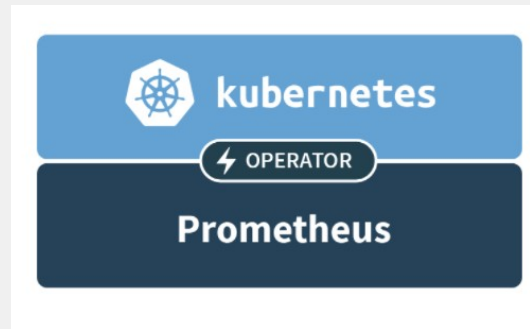
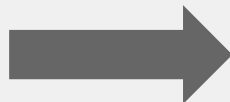
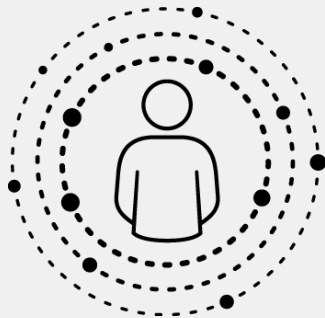


## OPERATOR

- encode human operational knowledge
- **automatically patch, upgrade, recover, and tune apps and services**
- Kubernetes-native
- Purpose-built for a specific application or service



# ENCODING AND AUTOMATING OPS KNOWLEDGE WITH OPERATORS



## WITHOUT OPERATORS

### REACTIVE

- Continually checks for anomalies
- Alert humans for response
- Requires manual change to fix

## WITH OPERATORS

### PROACTIVE

- Continually adjusts to optimal state
- Automatically acts in milliseconds



# OPERATOR FRAMEWORK

An open source toolkit to manage application instances on  
Kubernetes in an automated, scalable way



Build Operators without  
specialized knowledge of  
the Kubernetes API



Install, update, and manage  
Operators and their dependencies



Enable usage reporting  
for Operators

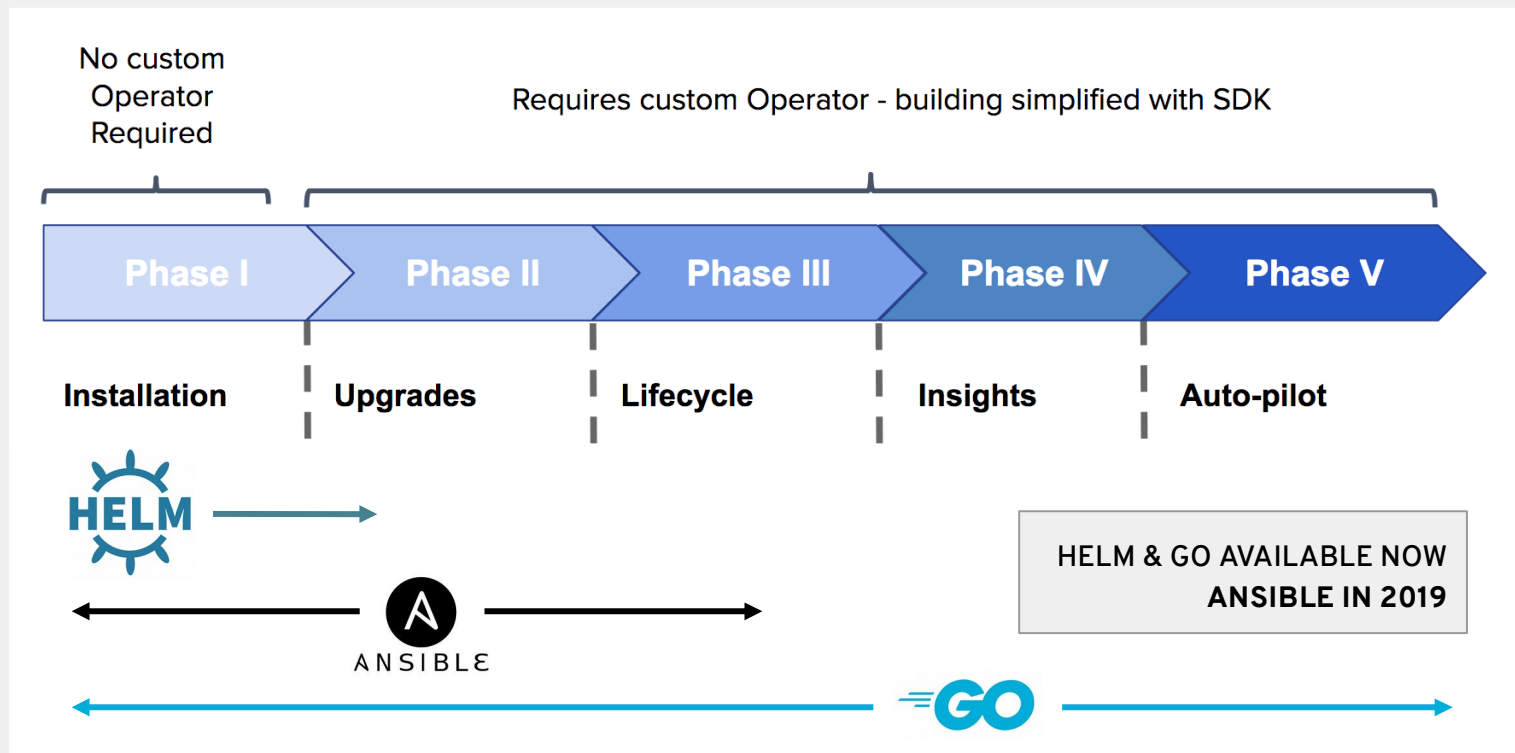
<https://github.com/operator-framework>



#RedHatOSD



# OPERATOR IMPLEMENTATION PATHS



# OPERATORS IN PREVIEW IN OCP 3.11

**APPLICATION OPERATORS**  
DEVELOPER PREVIEW

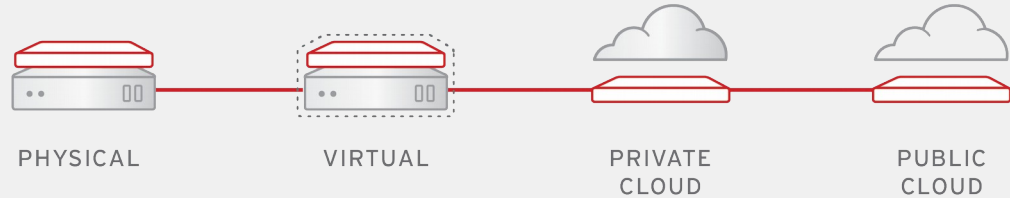


**OPERATOR LIFECYCLE  
MANAGER (OLM)**  
TECH PREVIEW

Install, manage, and upgrade Operators and their dependencies



Portable application services  
across any infrastructure





# PORTABLE HYBRID CLOUD SERVICES WITH ISV OPERATORS



**60+ Certified ISV Operators in Red Hat Early Access Program**



#RedHatOSD





# Container-native Virtualization

THE FUTURE OF VIRTUALIZATION!

**FEDERICO SIMONCELLI**

CNV Engineering Manager

[fsimonce@redhat.com](mailto:fsimonce@redhat.com)

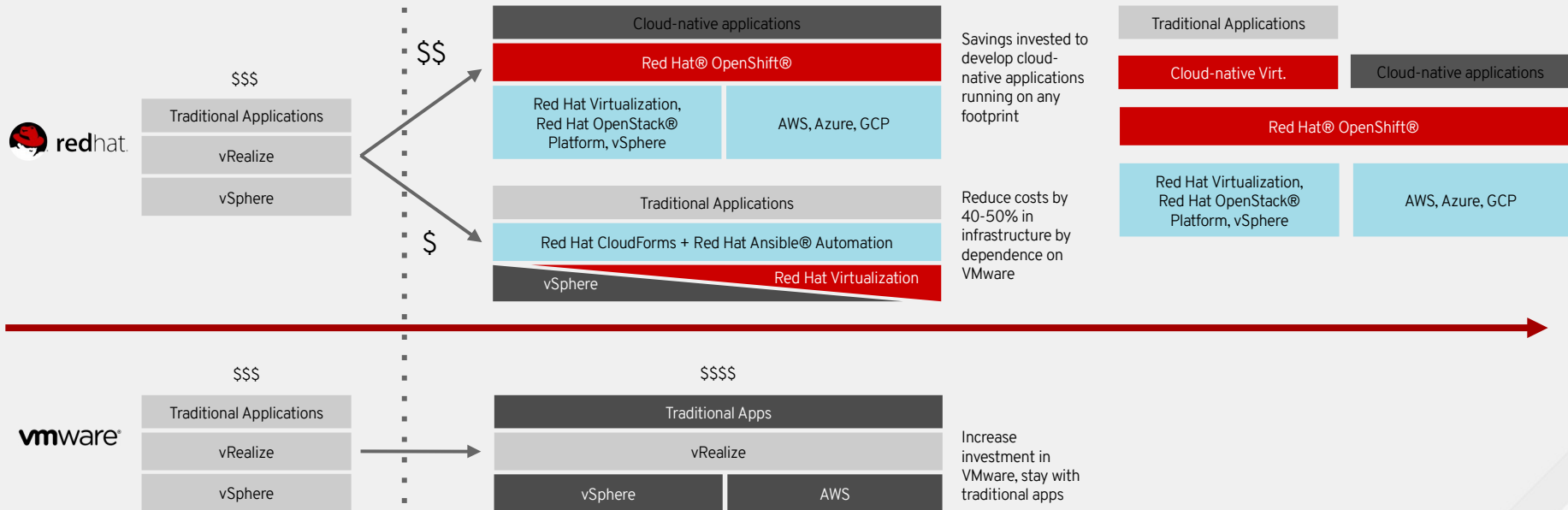


#RedHatOSD

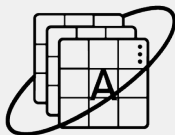
## PRESENT STATE

## TRANSFORMATION

## FUTURE OF VIRTUALIZATION



# CONTAINERS AND VIRTUAL MACHINES



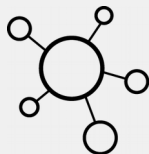
## CONTAINER INFRASTRUCTURE AND ORCHESTRATION

Containerized applications and Kubernetes container orchestration as provided by OpenShift are becoming the standard for new applications.



## VIRTUALIZED WORKLOADS

Virtualized workloads are not going anywhere fast! Business reasons (cost, time to market) and technical reasons (different or older operating system)



## BARE-METAL RESURGENCE

Increasingly customers are pursuing bare-metal clusters for net new business functionality being built in containers.

**As the technology mix changes, you will reach a tipping point where containers are the default but some workloads are still more suited to run as VMs**



# COMPONENTS OF CNV



- **KubeVirt**  
The virtual machine operator  
<https://github.com/kubevirt/kubevirt/>
- **Containerized Data Importer (CDI)**  
Importing disks  
<https://github.com/kubevirt/containerized-data-importer>
- **OpenShift Web Console**  
With UI extensions  
<https://github.com/openshift/origin-web-console>
- **Containerized Virt-v2v**  
Importing a whole virtual machine  
<https://github.com/kubevirt/v2v-job>

**Leverages tried and trusted RHEL & RHV (KVM) virtualization capabilities.**



# Container-native Virtualization Demo

[http://kubevirt.io/get\\_kubevirt/](http://kubevirt.io/get_kubevirt/)

## Pre-requisites:

- kubectl
- minikube/minishift

## Notes:

- Yes, we're running nested virt here - fine for getting started!
- Using upstream bits, for now, in product preview coming!



```

sgordon@kubevirt-minishift-demo/ $> # Let's look at the new pods our KubeVirt CRDs are running in the kube-system namespace.
sgordon@kubevirt-minishift-demo/ $> oc get crds
NAME                                     AGE
datavolumes.cdi.kubevirt.io            3h
openshiftwebconsoleconfigs.webconsole.operator.openshift.io 3h
virtualmachineinstancepresets.kubevirt.io 3h
virtualmachineinstancereplicaset.kubevirt.io 3h
virtualmachineinstances.kubevirt.io    3h
virtualmachines.kubevirt.io            3h
sgordon@kubevirt-minishift-demo/ $> # The CDI controller runs in the default namespace.
sgordon@kubevirt-minishift-demo/ $> oc get pods -n default
NAME                                READY   STATUS    RESTARTS   AGE
cdi-deployment-767b445c45-wp7pb    1/1     Running   0           3h
docker-registry-1-2gqht             1/1     Running   0           3h
persistent-volume-setup-658qq      0/1     Completed 0           3h
router-1-nn7qx                      1/1     Running   0           3h
sgordon@kubevirt-minishift-demo/ $> # Our own namespace is as expected empty right now.
sgordon@kubevirt-minishift-demo/ $> oc get pods
No resources found.
sgordon@kubevirt-minishift-demo/ $> oc get all
No resources found.
sgordon@kubevirt-minishift-demo/ $> # Lets look at a VM definition
sgordon@kubevirt-minishift-demo/ $> vim fedora-vm.yaml
sgordon@kubevirt-minishift-demo/ $> # Let's now create the VM
sgordon@kubevirt-minishift-demo/ $> kubectl create -f fedora-vm.yaml
virtualmachine.kubevirt.io "fedora-vm" created
sgordon@kubevirt-minishift-demo/ $> # The VirtualMachine object is the persistent representation of our virtual machine.

```

<https://youtu.be/0H5SbrpiH1Q>

# ROADMAP THEMES

(What's missing today?)

## Supportability

- Simplify upgrade process
- Debug tooling support (sosreports, Insights)
- Broad provider support

## Production Workloads

- Layer-2 Networking
- Live Migration
- Upload image as Template
- Guest agent introspection

## Embrace the Platform

- Operators for all
- Integrated VM management
- Metrics and monitoring

Container-native Virtualization is **not** a drop-in replacement for traditional virtualization today.

**Technology Preview access in an upcoming release of OpenShift.**







# OpenShift Container Platform 3.11

WHAT'S NEW?

**ALESSANDRO ARRICHIELLO**

Solution Architect  
ale@redhat.com

**PIERLUIGI SFORZA**

Solution Architect  
psforza@redhat.com

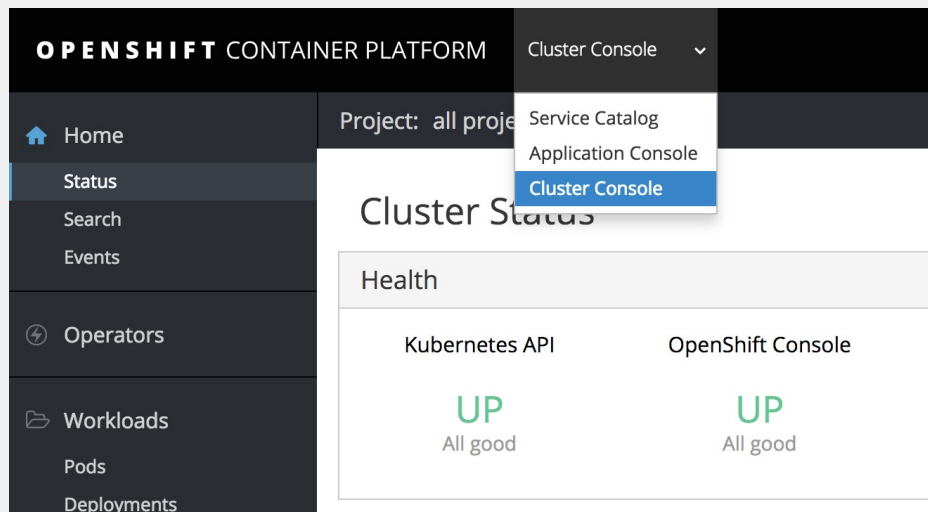


#RedHatOSD

# NEW ADMIN-FOCUSED CONSOLE

Users have a choice of experience based on their role or technical abilities

- Admin/CaaS experience with heavy exposure to Kubernetes
- AppDev/PaaS experience with standard OpenShift UX
- Sessions are not shared across the Consoles but credentials are
- Both hosted on cluster, in `openshift-console` and `openshift-webconsole` namespaces



# ACCESS CONTROL MANAGEMENT

## Visual management of the cluster's RBAC Roles and RoleBindings

- Track down users and service accounts with a specific Role
- View cluster-wide or namespaced bindings
- Visually audit a Role's verbs and objects

Project admins can self-manage roles and bindings scoped to their namespace

The screenshot displays the OpenShift Container Platform Cluster Console interface. The top navigation bar shows 'OPENSIFT CONTAINER PLATFORM' and 'Cluster Console'. The left sidebar contains navigation options: Home, Operators, Workloads, Networking, Storage, Builds, Monitoring, and Administration. The main content area is divided into two sections: 'Role Bindings' and 'Rules'.

**Role Bindings Section:**

- Project: all projects
- Buttons: Create Binding, Filter Role Bindings by role or subject...
- Summary: 62 Cluster-wide Role Bindings, 60 Namespace Role Bindings, 206 System Role Bindings
- Table with columns: NAME, ROLE REF, SUBJECT KIND, SUBJECT NAME, NAMESPACE
- Visible entries include: admin, admin-0, alertmanager-main, amostreams.v1.0.0.beta-role, cluster-admin, registry-admin, management-admin, default, management-admin, management-admin, management-admin.

**Rules Section:**

- Buttons: Add Rule, Create Binding
- Summary: Overview, YAML, Role Bindings
- Table with columns: NAME, ROLE REF, SUBJECT KIND, SUBJ
- Visible entries include: cluster-admin, cluster-admin-0, cluster-admins, couchbase-admin, olm-operator-binding-operator-lifecycle-manager.

**Rules Detail View:**

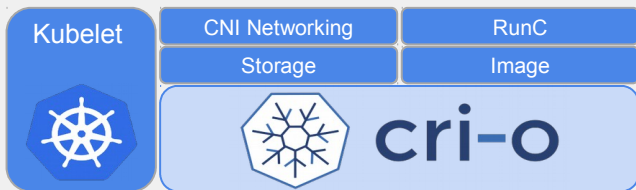
- Buttons: Overview, YAML, Role Bindings
- Buttons: Add Rule, Create Binding
- Table with columns: ACTIONS, RESOURCES
- ACTIONS: create, delete, deletecollection, get, list, patch, update, watch
- RESOURCES: secrets, serviceaccounts



# CRI-O / BUILDAH / PODMAN



- Becoming the default for partners
- Crictl for node debugging and troubleshooting
- Podman for image tagging & management
- Continues to mature with OpenShift online, customer, and community deployments

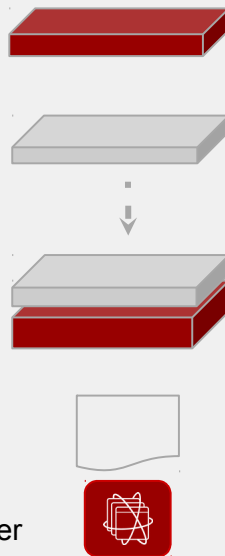


Start from an existing image or from scratch

Generate new layers and/or run commands on existing layers

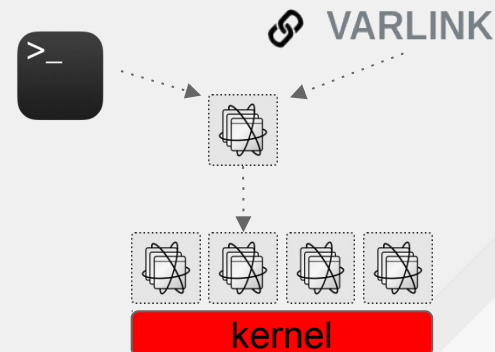
Commit storage and generate the image manifest

Deliver image to a local store or remote OCI / docker registry



Podman is planned to GA with RHEL 7.6.

A daemon-less CLI/API for running, managing, and debugging OCI containers and pods



# REFERENCE ARCHITECTURE GUIDES

**Release:** ocpsupplemental-3.11 (in 4-6 weeks after 3.11 GA)

Since 3.10, Reference Architecture Implementation guides are now part of the OpenShift product documentation (<https://docs.openshift.com>).

Documentation for deploying OCP 3.11 on: *(not live yet)*

- [OpenShift 3.11 on Red Hat OpenStack Platform \(RHOSP\)](#)
- [OpenShift 3.11 on Amazon Web Services \(AWS\)](#)
- [OpenShift 3.11 on Microsoft Azure](#)
- [OpenShift 3.11 on VMware vSphere](#)
- [OpenShift 3.11 on Google Cloud Platform \(GCP\)](#)
- [OpenShift 3.9 on Red Hat Virtualization 4 \(RHV\)](#) *(update in progress)*



# LOCAL DEVELOPMENT

## CDK 3.6

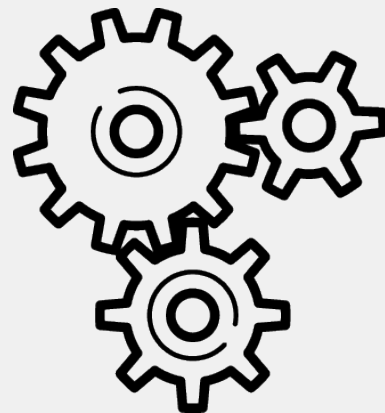
- OpenShift Container Platform v3.10.45 (and update to 3.11)
- Based on Minishift 1.24

## Minishift 1.24

- Configuration used to start a profile is not saved
- Provide a way to modify the kube-apiserver config same as openshift-apiserver.
- Do not apply templates in xpaas addon one by one
- Local proxy server to handle proxy issues. (technology preview)

## kubectl

- We always shipped kubectl for Linux on the master's file system, but now we will offer it in the [oc client downloads](#)








... so you want to do  
containers and Kubernetes?



#RedHatOSD





**When faced with two or more alternatives that  
deliver roughly the same value:  
Take the path that makes future changes easier.**

*Dave Thomas  
Author of Manifesto for  
Agile Software Development*



#RedHatOSD







# GRAZIE PER L'ATTENZIONE

**ALESSANDRO ARRICHIELLO**  
Solution Architect  
ale@redhat.com

**PIERLUIGI SFORZA**  
Solution Architect  
psforza@redhat.com



#RedHatOSD