OpenShift is a leading hybrid cloud, enterprise Kubernetes application platform, trusted by Many organizations.

Why OpenShift

Accelerate Application Delivery & DevOps

OpenShift helps organizations accelerate development & deployment of critical applications and services.

Customer Momentum

Every day more and more customers are looking into OpenShift. With customers spanning across 14 different industries, it's no suprise OpenShift is gaining traction.

Open Source Innovation Leaders

Red Hat is driving innovation in OpenShift and upstream communities like Docker, Kubernetes, ProjectAtomic & more.

Enterprise Ready

OpenShift provides a complete, enterprise-ready solution. From the operating system, to middleware, to a truely open hybrid cloud. All of which come from a vendor you can trust.

Running Red Hat OpenShift in your organization enables faster time to market, innovation, and enabled DevSecOps

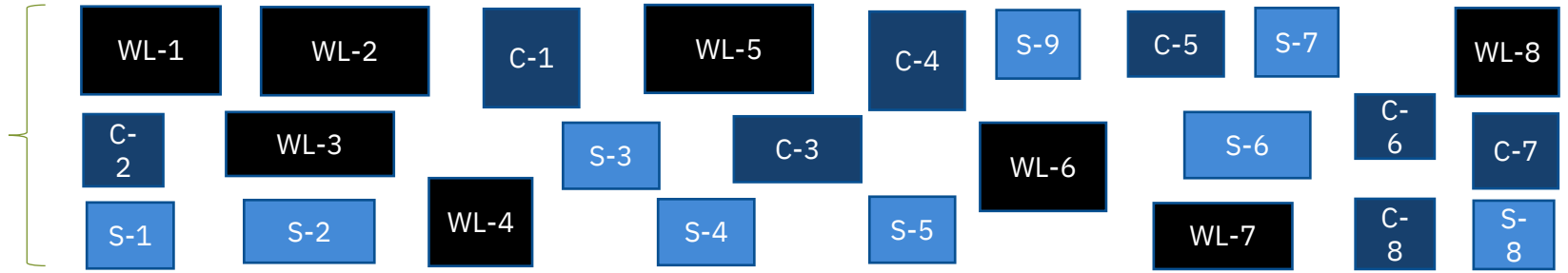
Red Hat OpenShift
Common platform for DevOps; Supports legacy and cloud native

	Developer	Operations	Hybrid multicloud management and portability	Security and controls
BENEFITS	▸ Choice of IDEs ▸ Choice of frameworks ▸ Choice of programming languages ▸ Build and deploy pipelines ▸ Nothing for developers to install	▸ Automated provisioning ▸ Automated installations/upgrades ▸ Autoscaling ▸ Wide range of workloads - 'Stateless' applications - 'Stateful' applications (e.g., DBs) - Containerize applications - VMs (Lift & Shift)	▸ Elastic infrastructure ▸ Choice of IaaS ▸ True application portability	▸ Clients controls implemented and audited ▸ Minimizing the attack potential ▸ Security scanning for vulnerabilities ▸ Encryption everywhere
TYPICAL RESULTS	▸ 20-30% reduction in time app teams spend patching IDE and middleware stack ▸ New developer productive in minutes vs. weeks ▸ Build pipeline 10x faster than current	▸ Create and provision complete, usable, containerized middleware (app servers, MQ, Kafka) in minutes versus days ▸ Upgrade OpenShift Clusters in 2 hours with no user impact ▸ Auto-scaling applications in 3 seconds ▸ Dynamic allocation of storage to containers across all middleware	▸ Single pane of glass enables workloads to move across clouds with ease and no-recompiling ▸ Configurable management console and dashboards, with consolidated logging	▸ Security controls automated on-prem and in the cloud ▸ Applications deployed to multi-clusters/Multiclouds while maintaining strict security posture

OpenShift Container Platform and OpenShift Virtualization

Workloads Types:

- WL — Virtualized Workload
- C — Containerized Workload
- S — Serverless Functions



Import Virtualized Workloads

VMware | RHV | OSP | Cloud | New Workloa

Declarative Co-existence

Virtualized Workloads | Containerized Workloads

RHV: Red Hat Virtualization
OSP: OpenStack Platform

Red Hat OpenShift Virtualization

certified kubernetes

Monitoring | OpenShift Container Storage | Application Services | OperatorHub | Scaling

Alerting | Logging | Developer Services | Networking | CRI-O

APIs, Controllers, Registry, Security, Web console, CLI

Virtualized Workloads, Containers and Serverless on a single platform

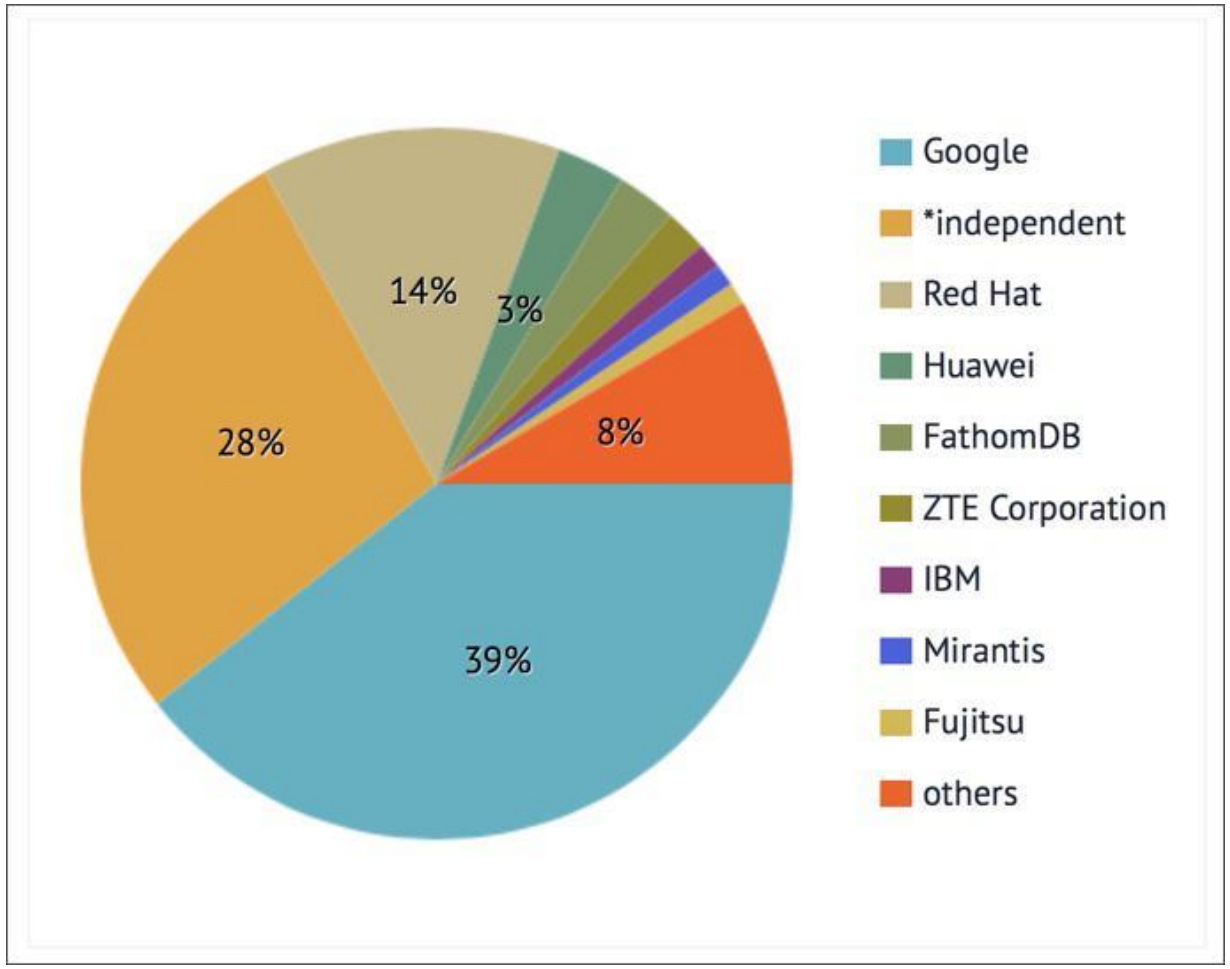
Red Hat OpenShift

Infrastructure

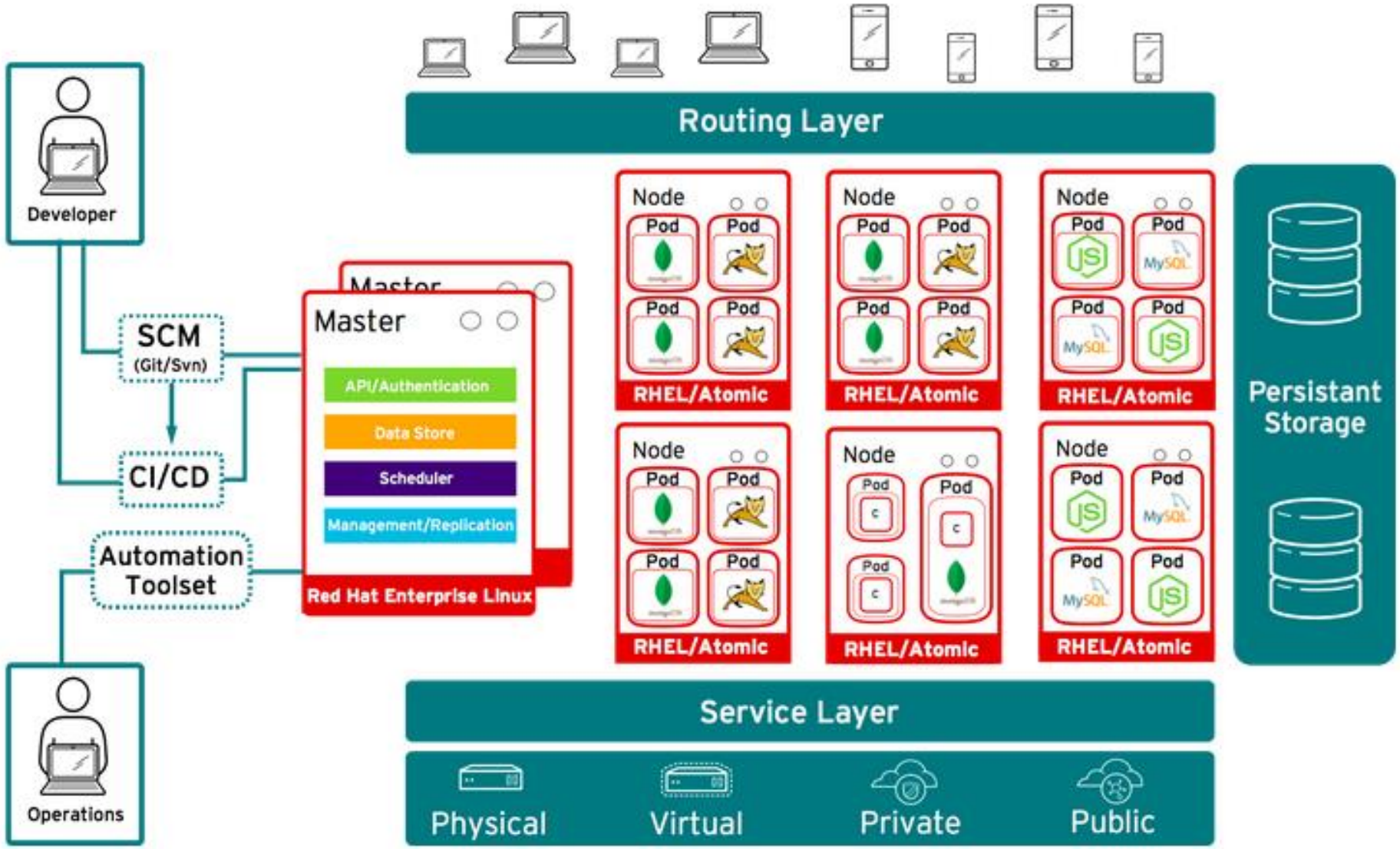
Openshift Deployed …..

Business benefits found by the companies:

- **531%** of the mean **ROI** in 5 years
- **66% increase** in application development lifecycle **velocity**
- **35% reduction in time** for each application developed
- **35% reduction in costs** of the development platform and the IT infrastructure for each application



Openshift Architecture



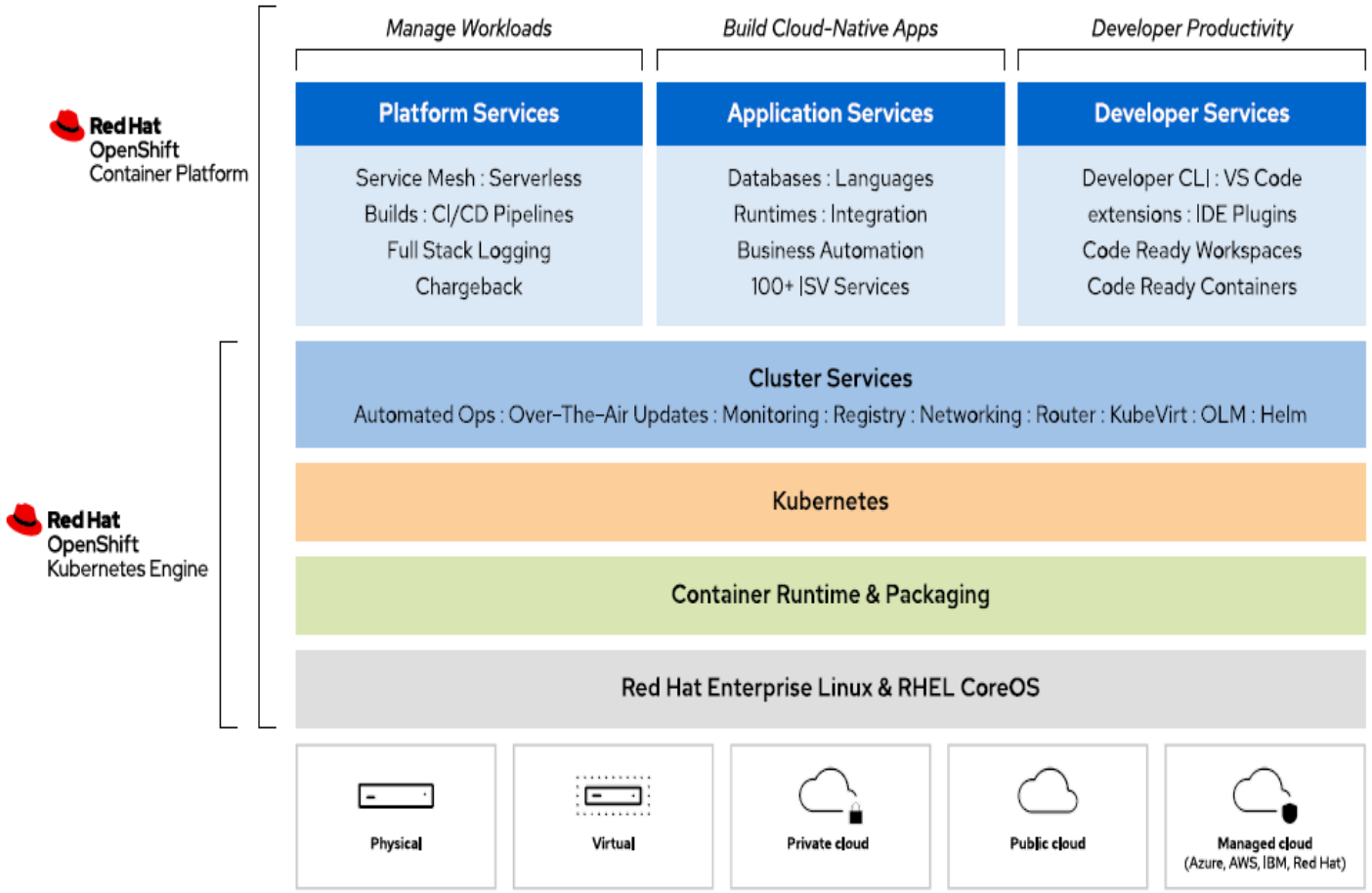


Figure 1.2: Feature comparison between OpenShift Container Platform and OpenShift Kubernetes Engine

OpenShift Features

- High Availability
- Lightweight Operating System
- Load Balancing
- Automating Scaling
- Logging and Monitoring
- Services Discovery
- Storage
- Application Management
- Cluster Extensibility

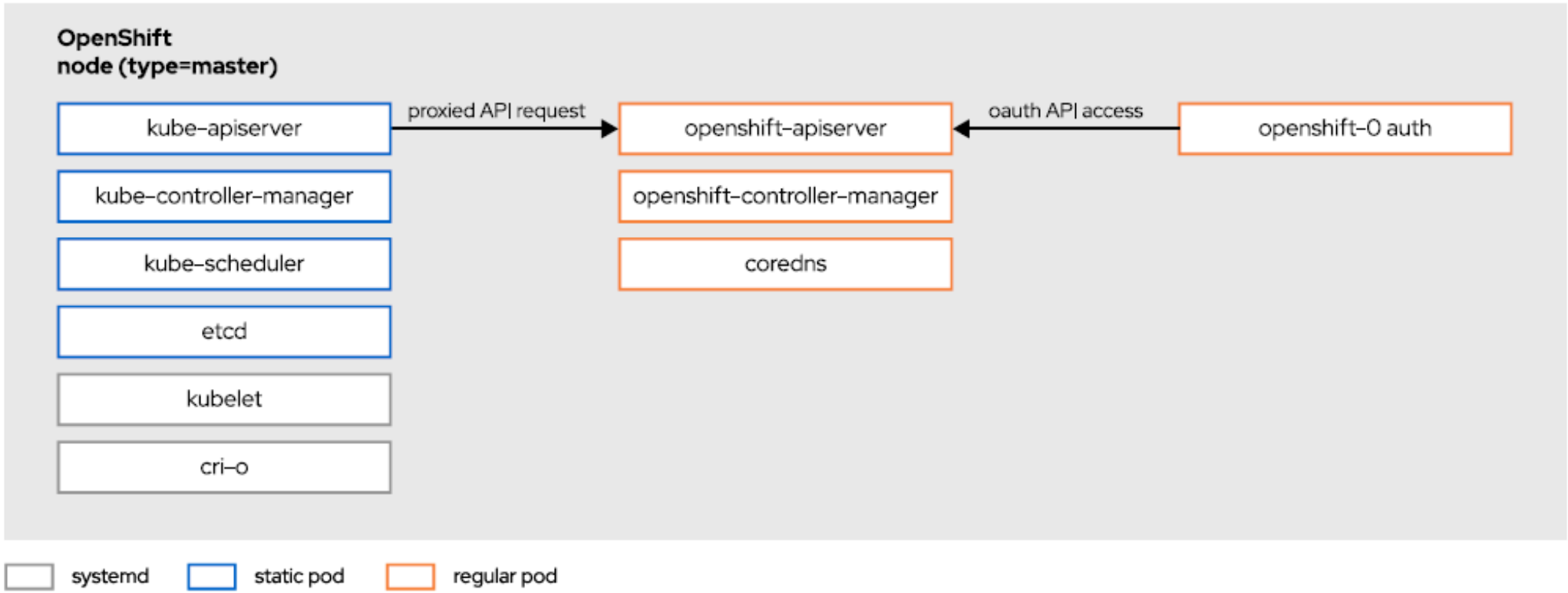
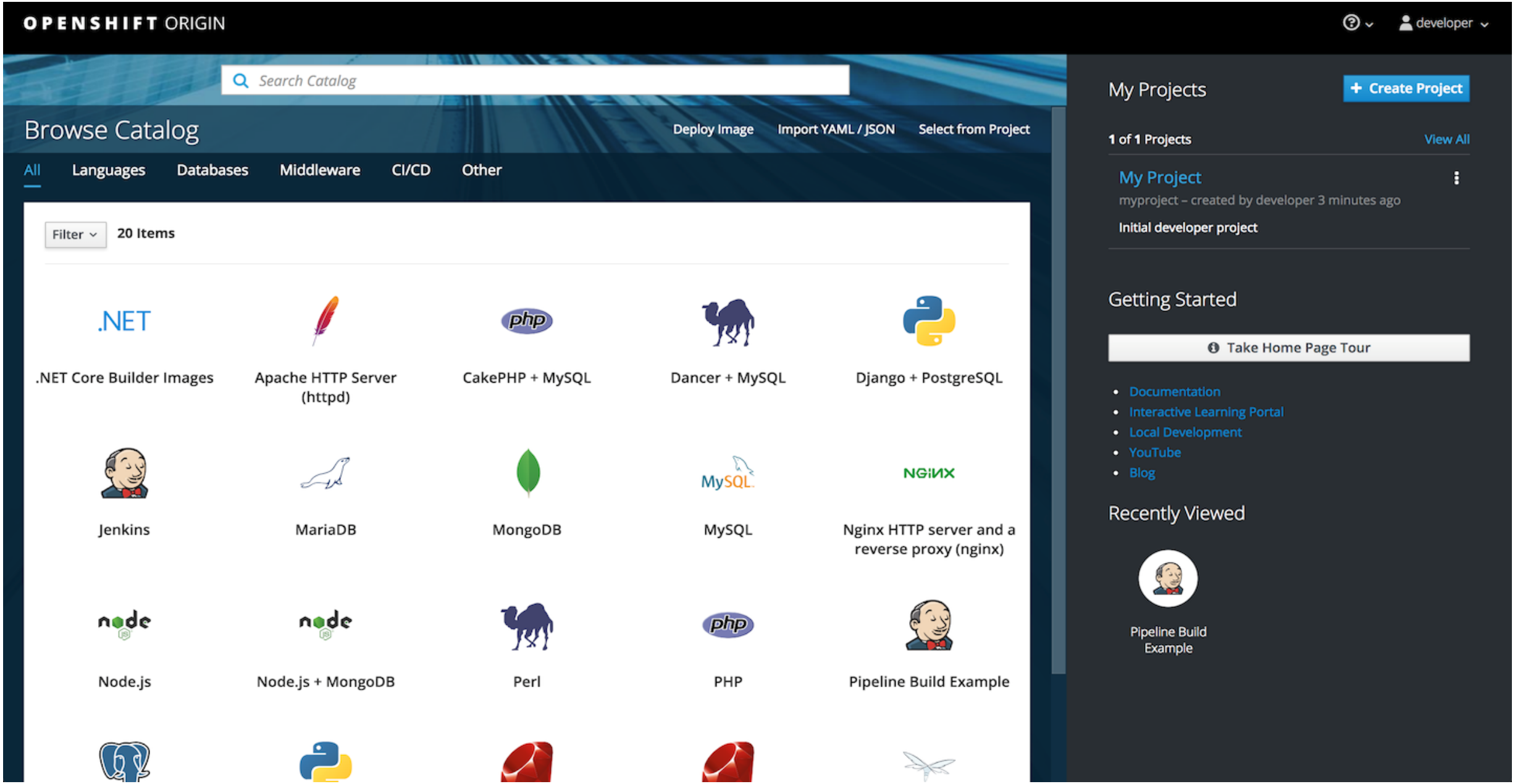
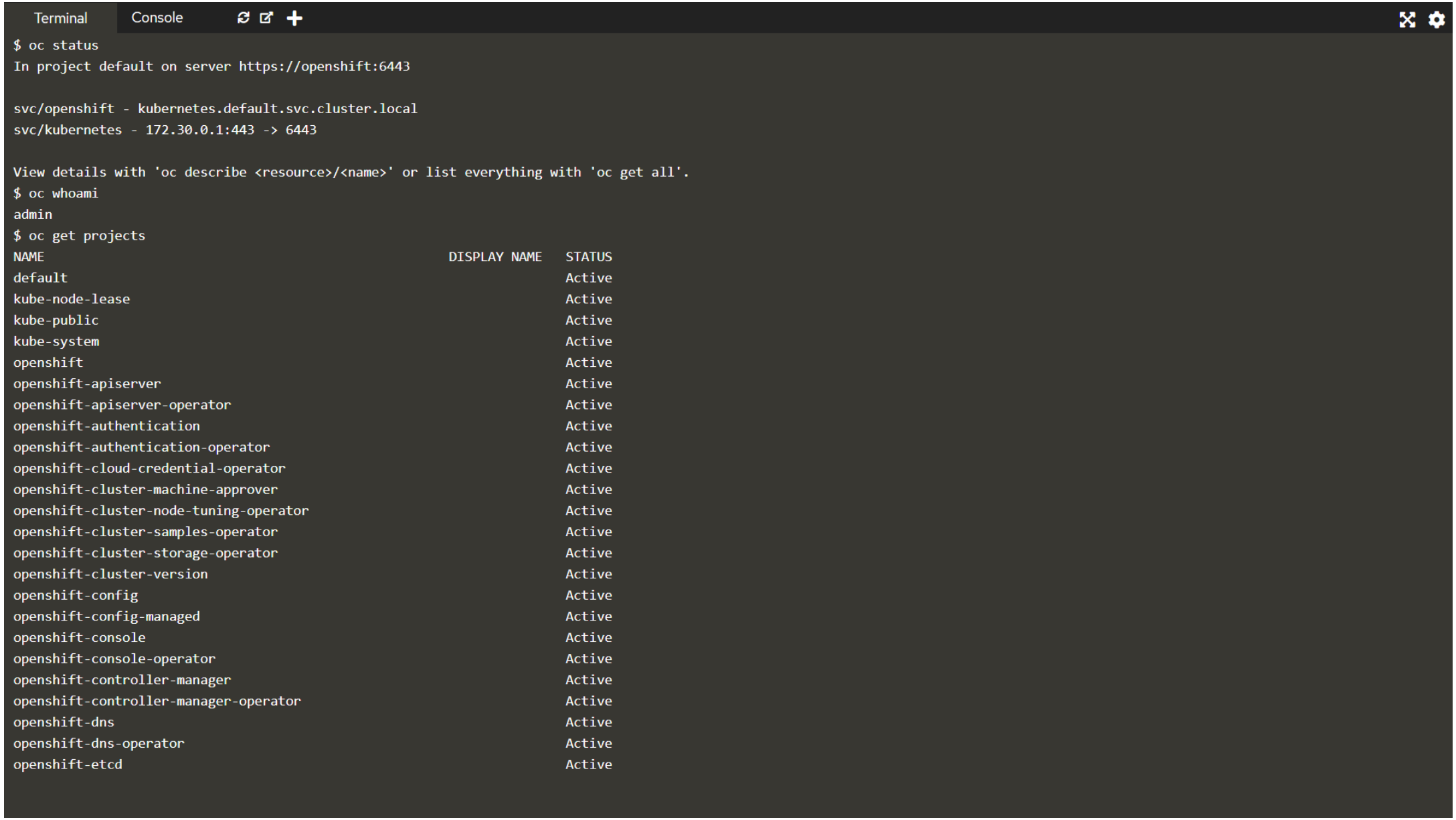


Figure 1.3: Architecture of an OpenShift control plane node

How Openshift Looks Like .. Live Demo

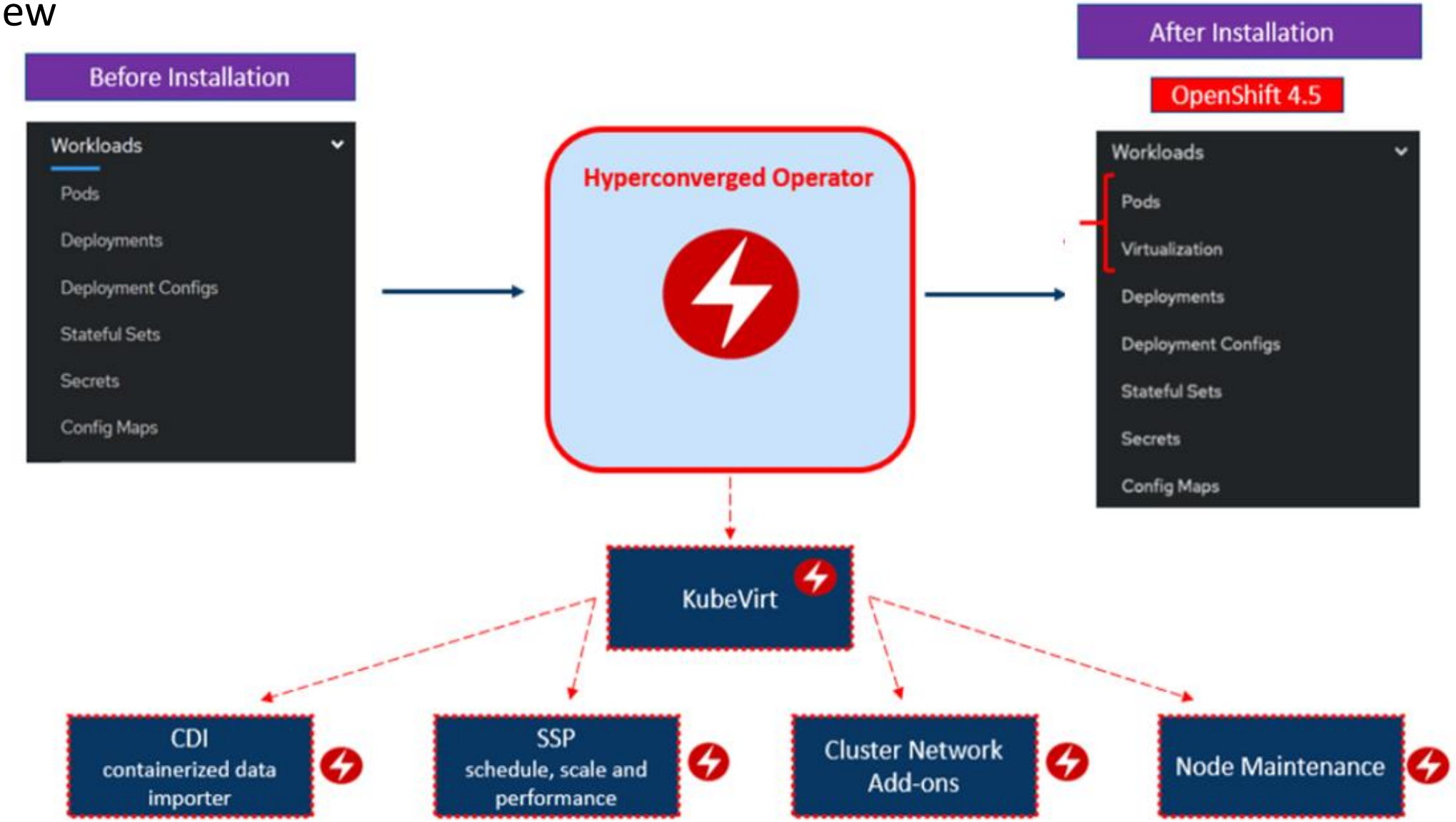


How Openshift Looks Like .. Live Demo

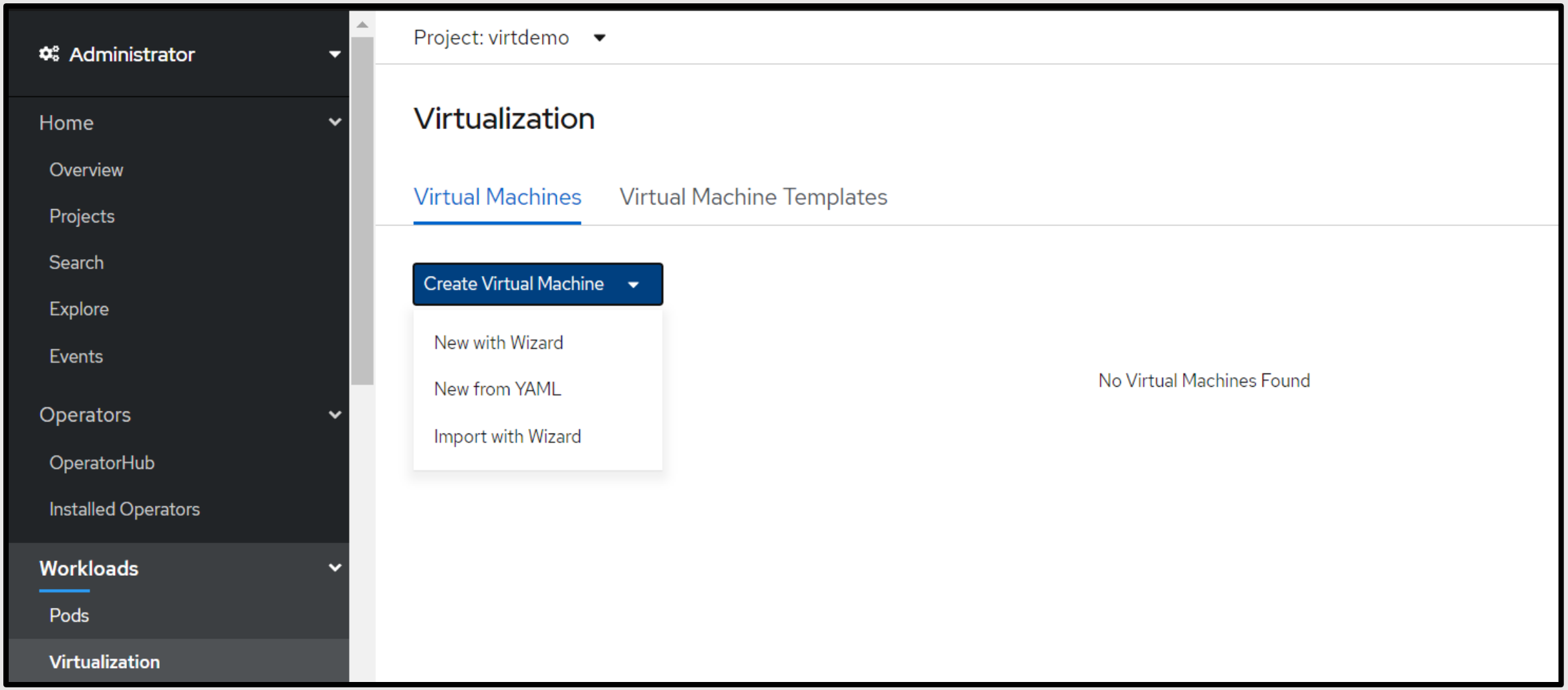


Operator View

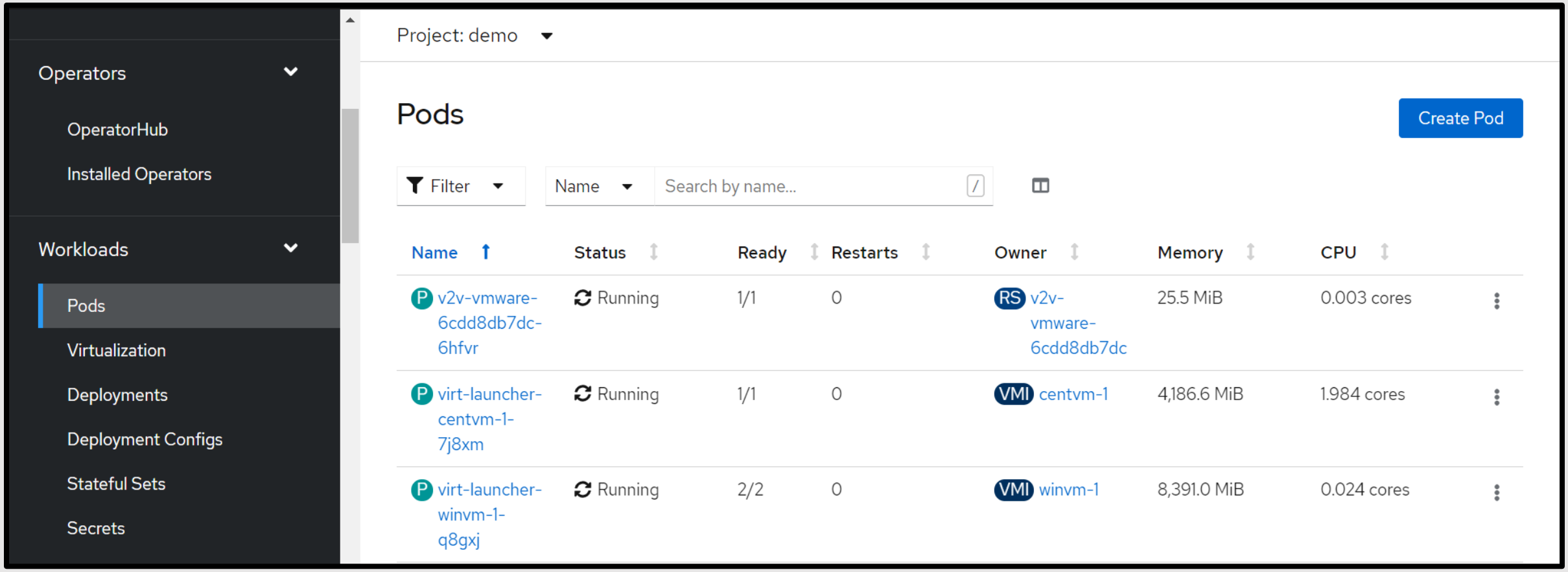
The diagram here shows the Operator view of the Red Hat OCP console.



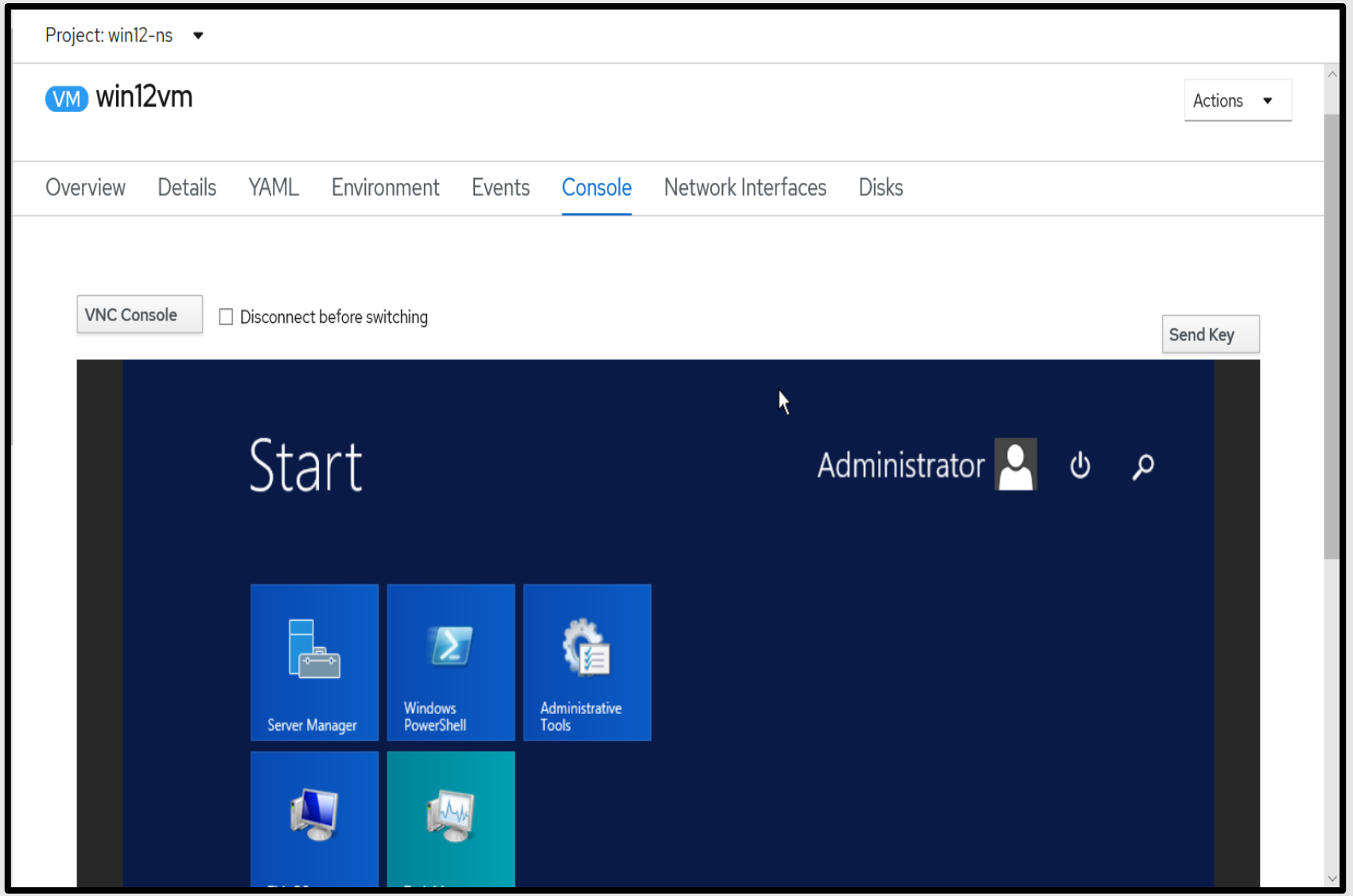
Web Console Interface - Virtualization



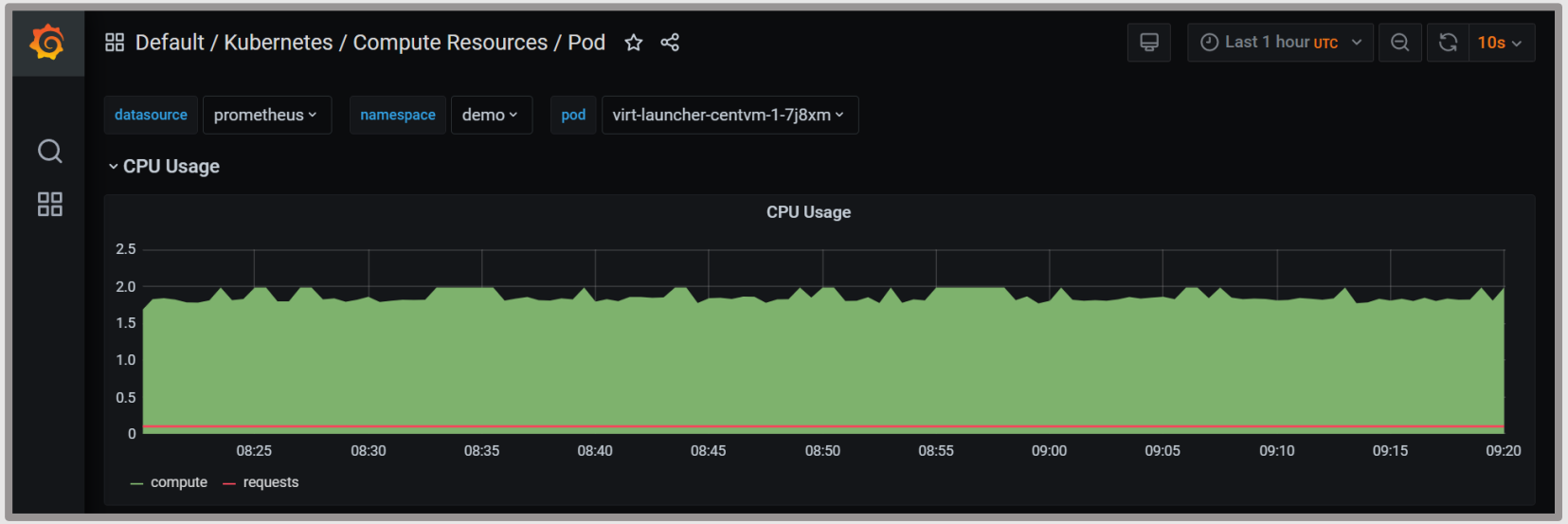
Web Console Interface - Pods

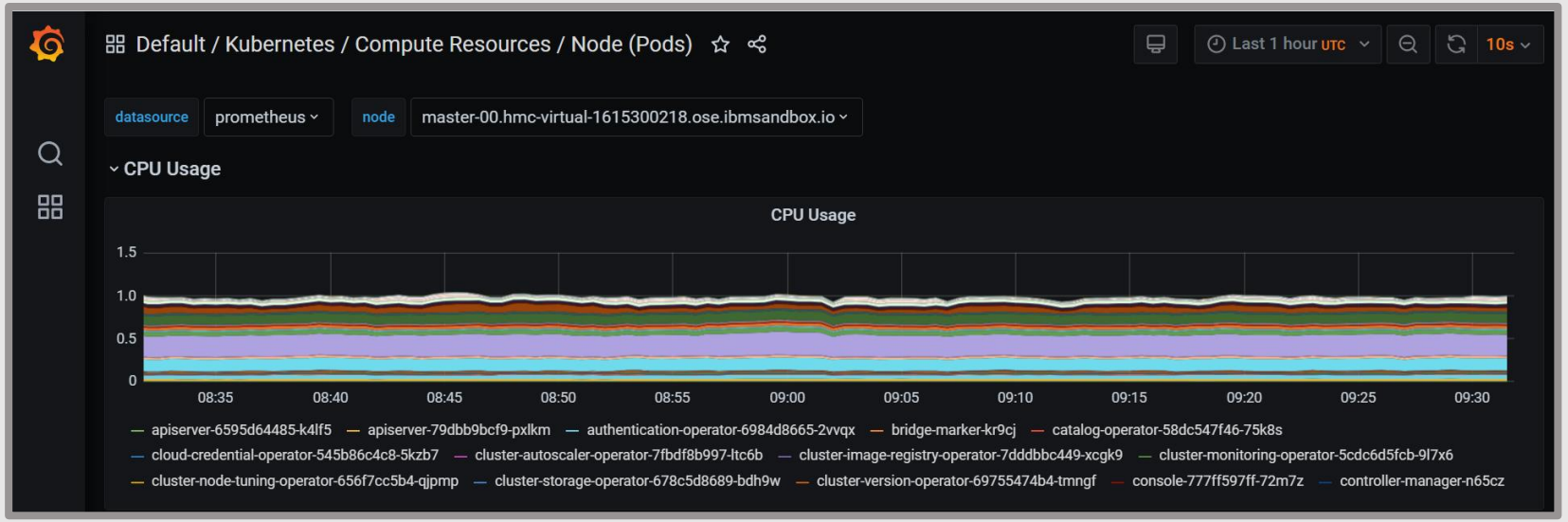


Web Console Interface - VM

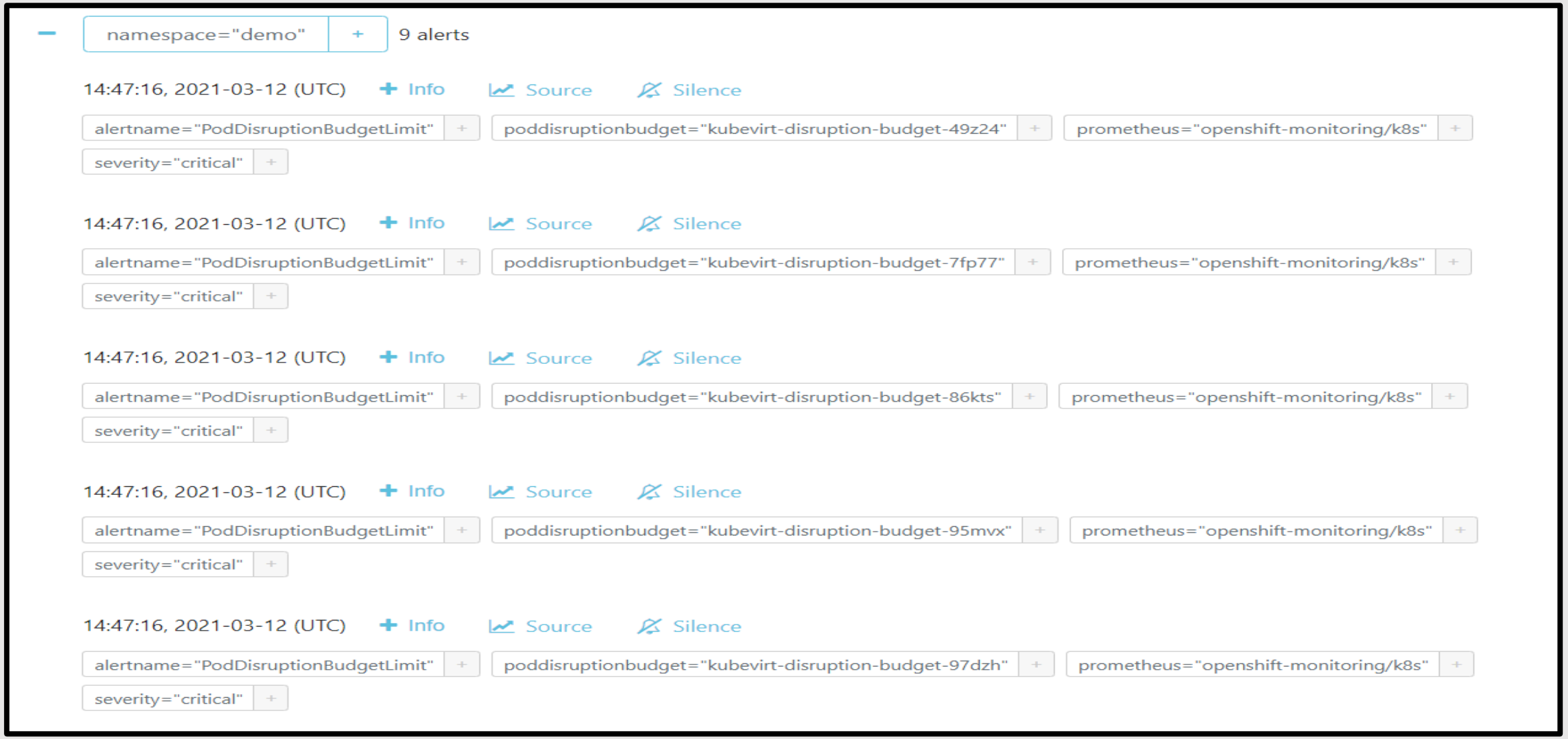


Monitoring

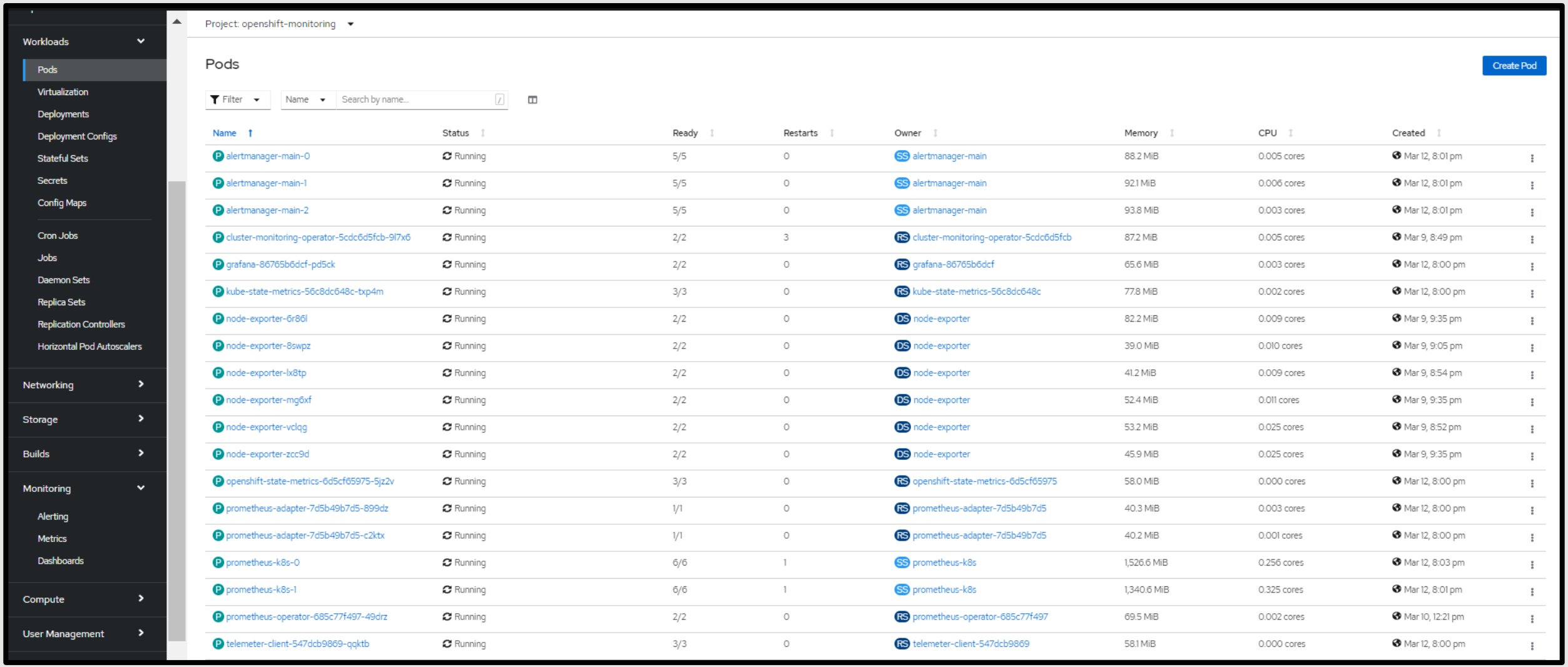

Default / Kubernetes / Compute Resources / Pod
Last 1 hour UTC
10s
datasource
prometheus
namespace
demo
pod
virt-launcher-centvm-1-7j8xm
CPU Usage
CPU Usage
2.5
2.0
1.5
1.0
0.5
0
08:25
08:30
08:35
08:40
08:45
08:50
08:55
09:00
09:05
09:10
09:15
09:20
compute
requests


Default / Kubernetes / Compute Resources / Node (Pods)
Last 1 hour UTC
10s
datasource
prometheus
node
master-00.hmc-virtual-1615300218.ose.ibmsandbox.io
CPU Usage
CPU Usage
1.5
1.0
0.5
0
08:35
08:40
08:45
08:50
08:55
09:00
09:05
09:10
09:15
09:20
09:25
09:30
apiserver-6595d64485-k4lf5
apiserver-79dbb9bcf9-pxlkm
authentication-operator-6984d8665-2vvqx
bridge-marker-kr9cj
catalog-operator-58dc547f46-75k8s
cloud-credential-operator-545b86c4c8-5kzb7
cluster-autoscaler-operator-7fbdf8b997-ltc6b
cluster-image-registry-operator-7dddbbc449-xcgk9
cluster-monitoring-operator-5cdc6d5fcb-9l7x6
cluster-node-tuning-operator-656f7cc5b4-qjpmp
cluster-storage-operator-678c5d8689-bdh9w
cluster-version-operator-69755474b4-tmngf
console-777ff597ff-72m7z
controller-manager-n65cz

Alerting


namespace="demo"
9 alerts
14:47:16, 2021-03-12 (UTC)
Info
Source
Silence
alertname="PodDisruptionBudgetLimit"
poddisruptionbudget="kubevirt-disruption-budget-49z24"
prometheus="openshift-monitoring/k8s"
severity="critical"
14:47:16, 2021-03-12 (UTC)
Info
Source
Silence
alertname="PodDisruptionBudgetLimit"
poddisruptionbudget="kubevirt-disruption-budget-7fp77"
prometheus="openshift-monitoring/k8s"
severity="critical"
14:47:16, 2021-03-12 (UTC)
Info
Source
Silence
alertname="PodDisruptionBudgetLimit"
poddisruptionbudget="kubevirt-disruption-budget-86kts"
prometheus="openshift-monitoring/k8s"
severity="critical"
14:47:16, 2021-03-12 (UTC)
Info
Source
Silence
alertname="PodDisruptionBudgetLimit"
poddisruptionbudget="kubevirt-disruption-budget-95mvx"
prometheus="openshift-monitoring/k8s"
severity="critical"
14:47:16, 2021-03-12 (UTC)
Info
Source
Silence
alertname="PodDisruptionBudgetLimit"
poddisruptionbudget="kubevirt-disruption-budget-97dzh"
prometheus="openshift-monitoring/k8s"
severity="critical"

Monitoring - Pods



Value Proposition on a page – 15 seconds pitch

MANAGEMENT CONSISTENCY

OPERATIONAL EFFICIENCY

DEVELOPMENT CONSISTENCY

LEGACY WITH MODERNIZATION FIRST APPROACH

FUTURE-PROOF SKILLSET MODERNIZATION

IN-BROWSER IDE

Kubernetes and OpenShift

Kubernetes Terminology

Term	Definition
Node	A server that hosts applications in a Kubernetes cluster.
Master Node	A node server that manages the **`control plane`** in a Kubernetes cluster. Master nodes provide basic cluster services such as APIs or controllers.
Worker Node	Also named **`Compute Node`**, worker nodes execute workloads for the cluster. Application pods are scheduled onto worker nodes.
Resource	Resources are any kind of component definition managed by Kubernetes. Resources contain the configuration of the managed component (for example, the role assigned to a node), and the current state of the component (for example, if the node is available).
Controller	A controller is a Kubernetes process that watches resources and makes changes attempting to move the current state towards the desired state.
Label	A key-value pair that can be assigned to any Kubernetes resource. Selectors use labels to filter eligible resources for scheduling and other operations.
Namespace	A scope for Kubernetes resources and processes, so that resources with the same name can be used in different boundaries.

OpenShift Terminology

Term	Definition
Infra Node	A node server containing infrastructure services like monitoring, logging, or external routing.
Console	A web UI provided by the RHOCP cluster that allows developers and administrators to interact with cluster resources.
Project	OpenShift's extension of Kubernetes' namespaces. Allows the definition of user access control (UAC) to resources.