Connect

What you didn't know about OpenShift

Pilar Bravo

OpenShift Specialist Solution Architect Role Leader

Red Hat

✓ Computer Engineer

✓ Started as J2EE developer

✓ Red Hatter since 2008

✓ Software engineer, Consultant and Solution Architect

✓ Based in Madrid, Spain

Pilar Bravo
OpenShift Specialist Solution Architect Role Leader

I have OpenShift.
Can I ... ?

Administration

Assess and monitor health
I want to be aware of my exposure to issues that can affect service availability, fault tolerance, performance or security.

Recommendations
I want to receive personalised recommendations to avoid risks and get resolutions tailored to my individual clusters

Support
I want to have proactive support so I can avoid problems before they happen.

Knowledge
I want to take advantage of all the knowledge Red Hat has from the thousands of customers using OpenShift.

Red Hat

Administration: Insights Advisor for OpenShift

▸ **Free service leveraging Red Hat experience with supporting and operating OpenShift**

▸ Insights Advisor UI adds **Upgrade risks (Preview)** – ML powered technology to identify potential blockers that could interrupt OCP upgrade flow

▸ **New Insights recommendations** focusing on preventing issues with OpenShift Data Foundation (ODF), OpenShift Cluster Version Operator (CVO) and OpenShift Cluster Autoscaler Operator (CAO)

▸ Insights recommendations available in Hybrid Cloud Console –> **Cluster History** page

https://console.redhat.com/openshift/advisor
https://console.redhat.com/settings/notifications/openshift

5

Security

Sandboxing for testing software
I want to run a containerized workload with known vulnerabilities or to handle an issue in a legacy application.
I want to have administrative control over pods or load custom kernel modules.

Kernel isolation
I want to securely run workloads that require custom kernel tuning and being able to create custom kernel modules.

Noisy neighbour (interference)
I want to run third-party workloads from multiple vendors, (like CNFs) without their custom settings interfering with packet tuning or with sysctl variables set by other apps.

Privileged workloads
I want to securely run workloads that require elevated root privileges, like access to a specific physical device.

Security: OpenShift Sandboxed Containers

Traditional Containers
Isolation by namespaces, cgroups with shared kernel

Container A — PROCESS A — Filter: Seccomp, MAC, CAPS — namespaces

Container B — PROCESS B — Filter: Seccomp, MAC, CAPS — namespaces

Container C — PROCESS C — Filter: Seccomp, MAC, CAPS — namespaces

HOST LINUX KERNEL

CPU   Memory   Network   Storage

Kata Containers
Each container or pod is more isolated in its own lightweight VM

Container A — VIRTUAL MACHINE — PROCESS A — namespaces — GUEST LINUX KERNEL A — HARDWARE VIRTUALIZATION

Container B — VIRTUAL MACHINE — PROCESS B — namespaces — GUEST LINUX KERNEL B — HARDWARE VIRTUALIZATION

Container C — VIRTUAL MACHINE — PROCESS C — namespaces — GUEST LINUX KERNEL C — HARDWARE VIRTUALIZATION

HOST LINUX KERNEL

Red Hat

Virtualization

Modernization
I want to modernize my virtualization workloads, but transforming everything in containers is slow and expensive.

Single Architecture
I want to have a single cloud native platform to be able to host all my workloads.

Vintage Applications
I want to keep my vintage applications as they are. I do not want to migrate them to containers.
I want to keep my Windows VMs as they are for the moment.

Consistent Developer Experience
I want to have the same tools, processes and pipelines for all my applications. I want an unified control and DevOps pipeline.
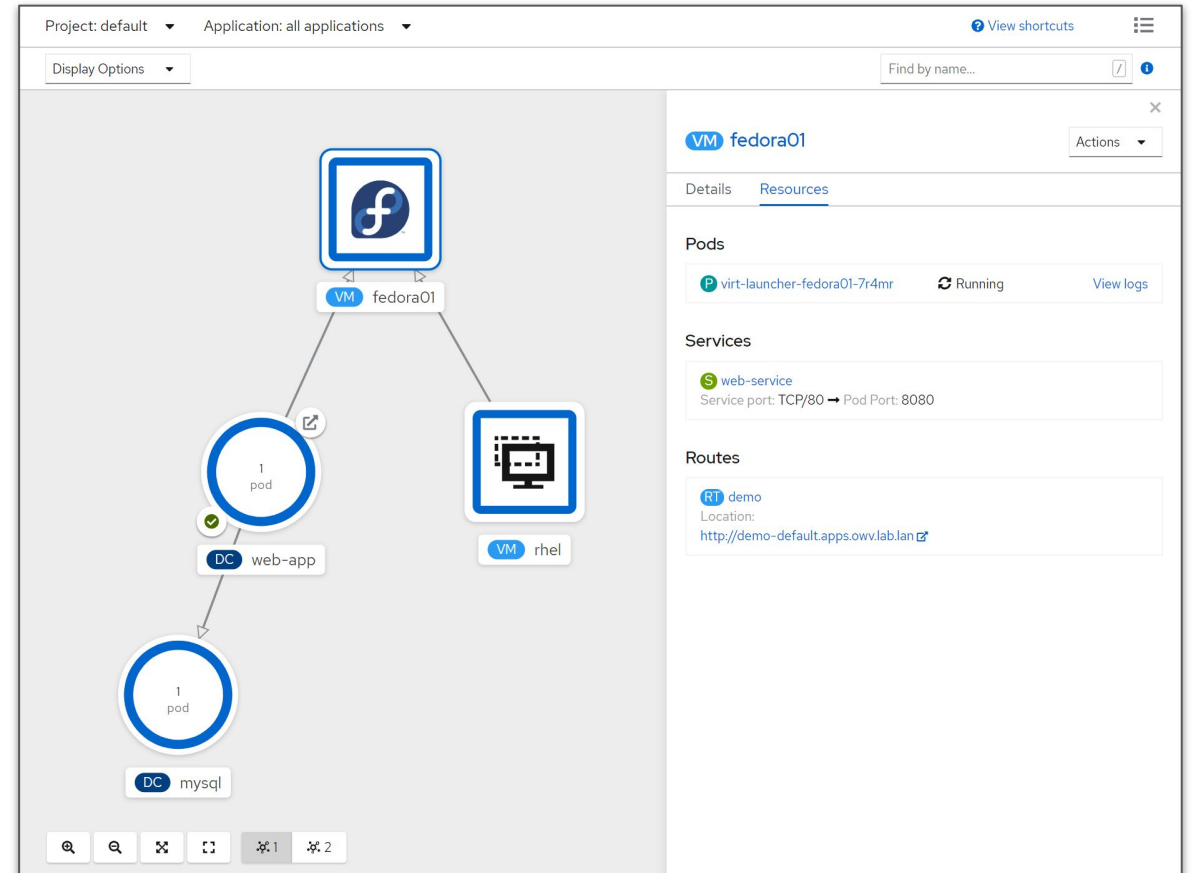
Red Hat

Virtualization: OpenShift Virtualization

➔ Virtual machines
  ◆ Running in containers, managed as Pods
  ◆ Using the KVM hypervisor
➔ Scheduled, deployed, and managed by Kubernetes
➔ OpenShift Virtualization uses KVM, the Linux kernel hypervisor
➔ KVM is a core component of the Red Hat Enterprise Linux kernel
  ◆ KVM has 10+ years of production use: Red Hat Virtualization, Red Hat OpenStack Platform, and RHEL all leverage KVM, QEMU, and libvirt
➔ QEMU uses KVM to execute virtual machines
➔ `libvirt` provides a management abstraction layer
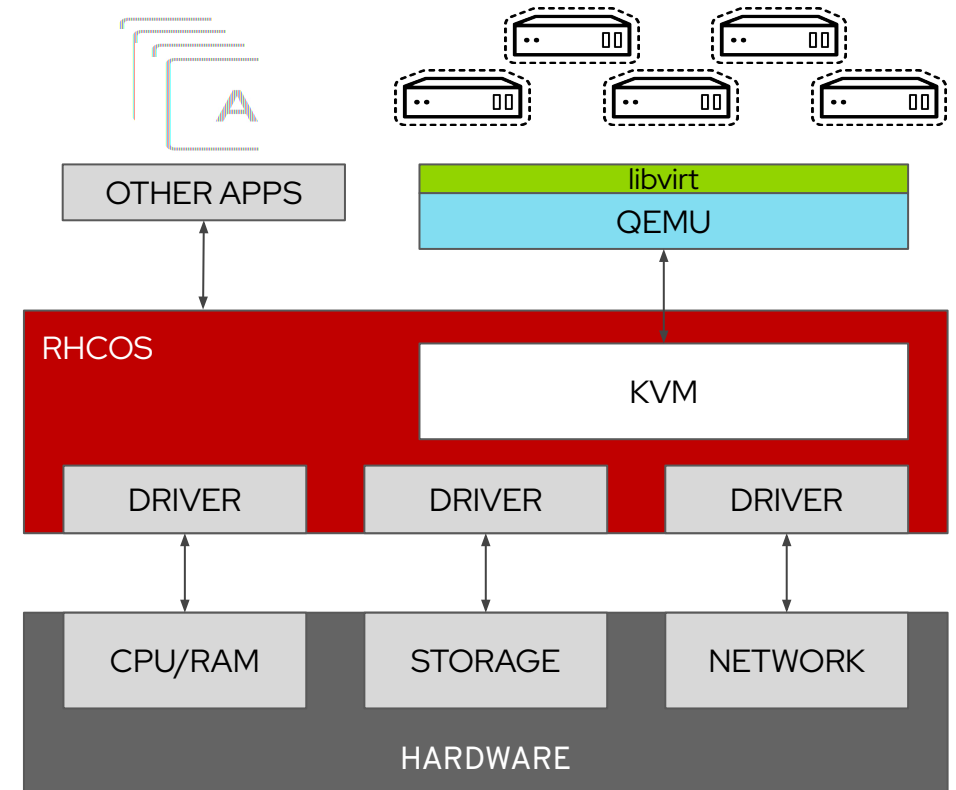
Virtualization: OpenShift Virtualization

➔ Virtual Machines connected to pod networks are accessible using standard Kubernetes methods:
  ◆ Service
  ◆ Route
  ◆ Pipelines
  ◆ etc.
➔ 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

Virtualization: OpenShift Virtualization

Migration Toolkit for Virtualization

➔ Easy to use UI

➔ Mass migration of VMs from VMware, Red Hat Virtualization and OpenStack to OpenShift (Tech Preview)

➔ VM data pre-copied before shutdown (Warm Migration) for VMware and RHV migrations

➔ VM validation service: Run checks on VM configuration to avoid migration issues

➔ Parallelized VM conversion

◆ Maximize throughput

➔ Migration Network Selection

◆ Avoid impact on other running workloads

Flexibility

Agility
It takes longer than expected to create a cluster. I want to be able to create clusters quickly, even as part of a pipeline as temporary clusters.

Restrict administrator access
As an administrator, I don't want the cluster users to have access to the control plane.
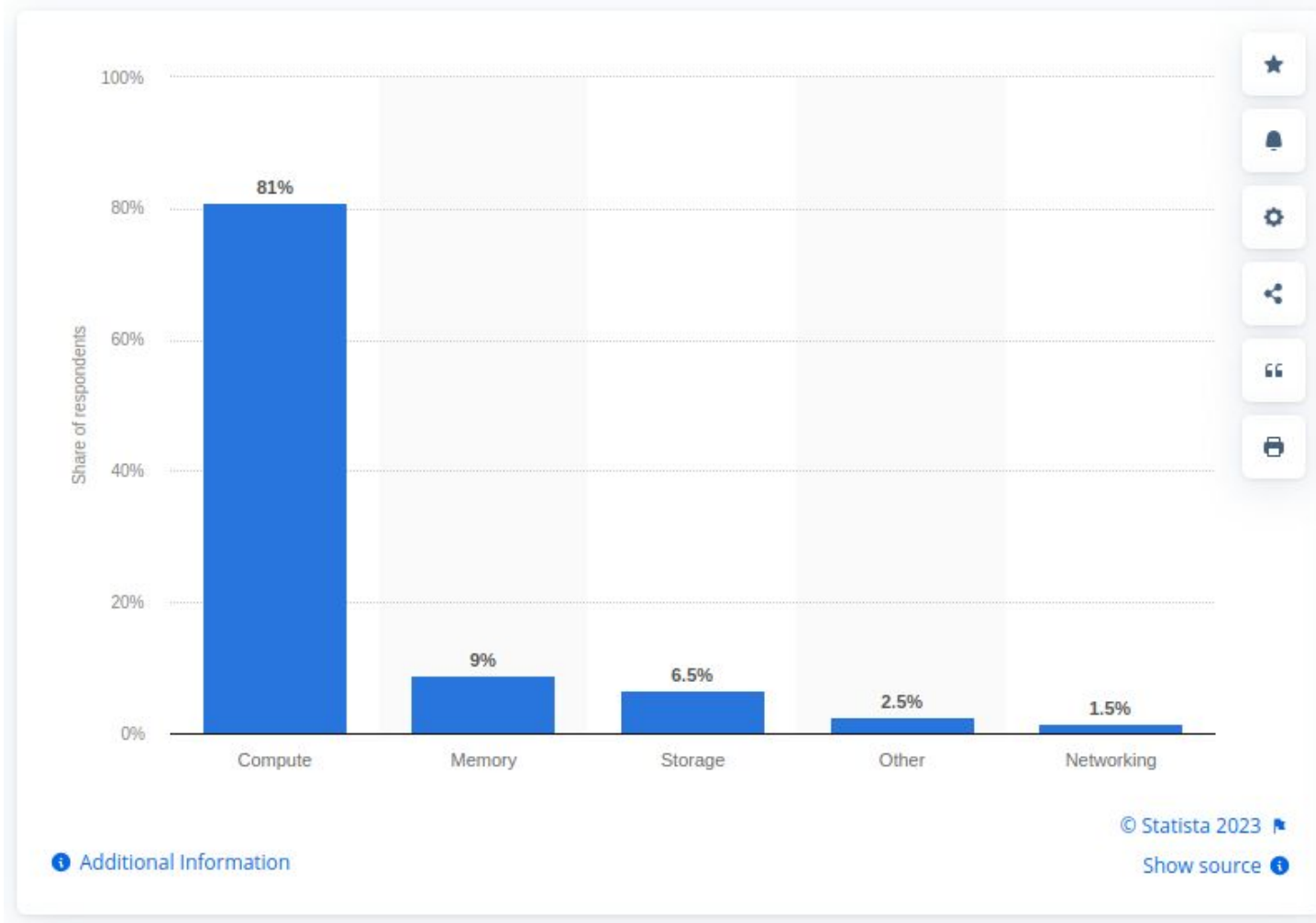
Resource Usage
It takes a lot of compute to run my control plane, and I am wasting resources most of the time. I want to optimize resource usage.

Lack of space
I want more space for the workloads in my cluster, instead of provisioning more hardware.

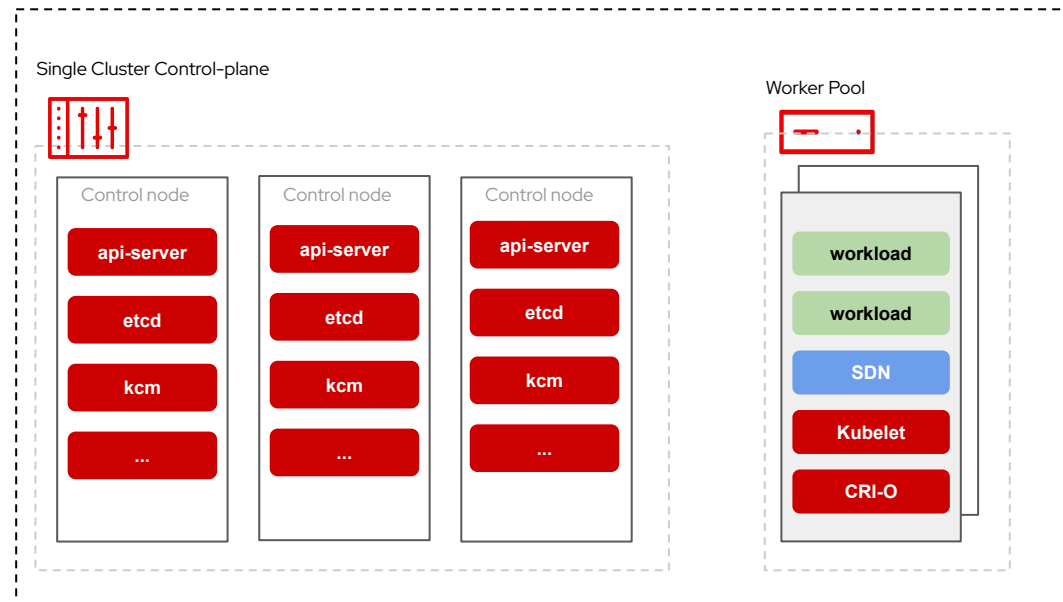Red Hat

Flexibility

Source: Statista

Flexibility: OpenShift Hosted Control Planes

Standalone OpenShift

Control-Plane (CP) + Workers

Standalone OpenShift **Cluster** (dedicated CP nodes)

Single Cluster Control-plane

Control node
- api-server
- etcd
- kcm
- ...

Control node
- api-server
- etcd
- kcm
- ...

Control node
- api-server
- etcd
- kcm
- ...

Worker Pool
- workload
- workload
- SDN
- Kubelet
- CRI-O

Low CAPEX and OPEX costs
(bundling of CPs + CP as pods)

Central Management of CPs
(easy operation & maintenance)

Multi-arch support
(e.g. CP x86, workers ARM)*****

HyperShift

Control-Plane (CP) **+** **Workers**

HyperShift **Clusters** (decoupled CP and workers)

Management Cluster (Hosts Control Planes)

Cluster 1 Namespace
(control-plane)
- api-server
- etcd
- kcm
- ...

Cluster 2 Namespace
(control-plane)
- api-server
- etcd
- kcm
- ...

Cluster 3 Namespace
(control-plane)
- api-server
- etcd
- kcm
- ...

Cluster 1 workers
- Worker
- worker

Cluster 2 workers
- Worker
- worker

Cluster 3 workers
- Worker
- worker
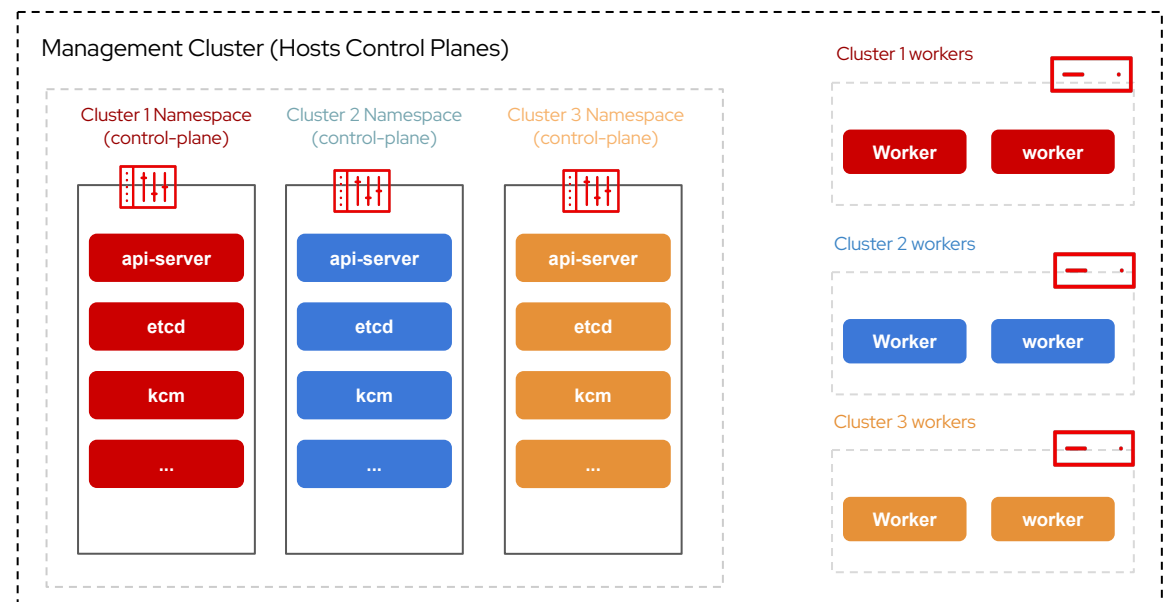
Network & Trust
segmentation

Mixed Iaas For CP and
Workers*****

Fast cluster bootstrapping
(CP as Pods)

Hosted Control Planes + Virtualization

I am running OpenShift on VMs
I want to have a single cloud native platform to be able to host all my workloads without entitlements needed for my bare metal hardware.
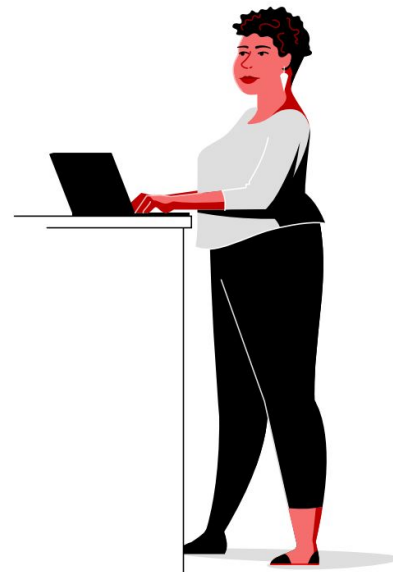
I am running OpenShift on bare metal
I want to host multiple OpenShift clusters and increase utilization.

I provide dedicated clusters
I want to faster cluster provisioning, offload administration to cluster admins, improve cluster isolation and be able to run different OpenShift versions.

Red Hat

Hosted Control Planes + Virtualization

Increase Utilization of Infrastructure

- Reduce unused and underutilized infrastructure
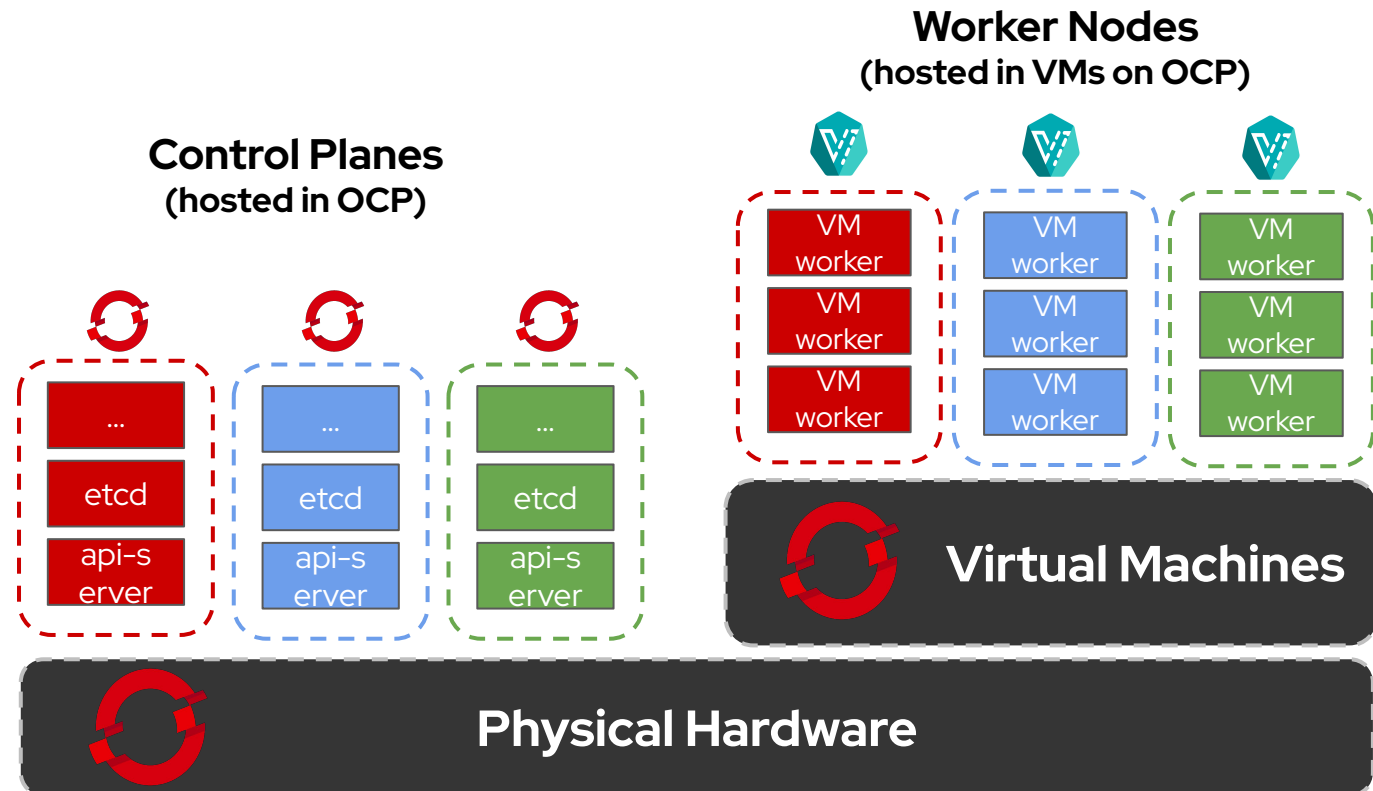- Increase bare metal node utilization by deploying multiple hosted clusters.

Reduce Dependency on Legacy Virtualization

- Eliminate legacy hypervisor hosting your container platform.
- Underlying virtualization layer is included with hosted OpenShift cluster entitlements (no separate licensing needed)

Worker Nodes
(hosted in VMs on OCP)

Control Planes
(hosted in OCP)

...	...	...
etcd	etcd	etcd
api-server	api-server	api-server

VM worker / VM worker / VM worker
VM worker / VM worker / VM worker
VM worker / VM worker / VM worker

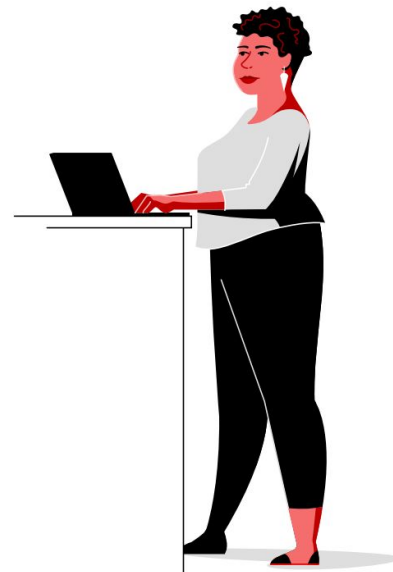**Virtual Machines**

Physical Hardware

Edge

I have different platforms with different resources and locations
Can I work the same way, with the same deployment model, everywhere? Can I move my workloads wherever it makes more sense?

My apps have various components
Can I upgrade each one separately? Can I standardize the way I work in the datacenter across all my devices?
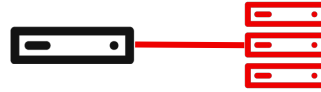
I need to create real time applications
Can I use OpenShift for that? Can I move my applications closer to the users? Can I enable new business models with OpenShift?

OpenShift at the Edge

**OPENSHIFT
SINGLE NODE**

**OPENSHIFT
REMOTE WORKER NODES**

**OPENSHIFT
3-NODE**

Definition

OpenShift deployment on a single box (**C**ontrol Plane + **W**orker) with fair amount of resources.	**OpenShift C**ontrol Plane resides in a central location ("good enough" network required), whereas **W**orkers are distributed sharing the control plane.	**OpenShift C**ontrol Plane and **W**orkers reside on the same node. HA setup with 3 servers.

Examples

In-vehicle field operations	Telco 5G far edge – RAN	Telco 5G near edge & MEC
Telco 5G sparsely populated areas	IoT / data collection gateways	Edge AI & Data pipelining
In-field single server operations	Visual Inspection using Machine Vision	Smart manufacturing
Disconnected environments	Critical Workload requiring High Availability	Remote office
		Disconnected clusters

18

Red Hat

Artificial Intelligence

Tested and supported AI/ML tools
I want to use tools like JupyterHub, TensorFlow and PyTorch in OpenShift, fully upgraded, tested and supported.

Freedom of choice
I want to extend OpenShift with more services of my choice. Including not only Red Hat technology but partners, like Anaconda, IBM, Seldon, etc...
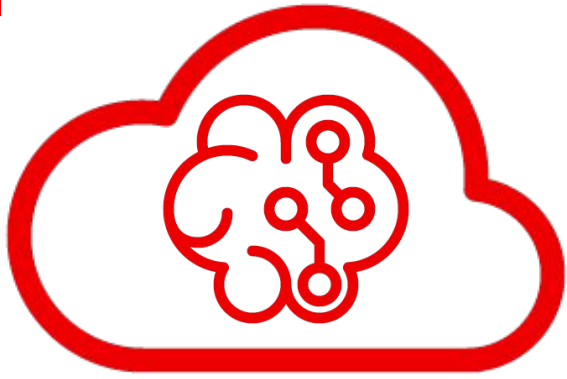
I want to start fast and scale quickly
As a data scientist, I want extra resources to start fast and scale, with things like integrated GPU support with NVIDIA and CUDA runtimes.

Develop best practices
I want to use the same deployment model for all my applications, integrate regular apps with AI/ML models. Also control which data I am uploading to the cloud.

Red Hat

AI for the Open Hybrid Cloud

Train, serve, monitor and manage the lifecycle of AI/ML models and applications, from experiments to production

Red Hat OpenShift

Red Hat OpenShift Data Science

Red Hat OpenShift AI

Expands upon the proven capabilities of Red Hat OpenShift and Red Hat OpenShift Data Science, to:

▸ Provide a unified platform for data scientists, application developers and IT Ops

▸ Scale to handle workload demands of foundation models (volume of data, duration of training run, size of model, acceleration required, and scalability).

▸ Deliver consistency, ease-of-use, and cloud-to-edge deployment options.

▸ Underlying platform for training, serving and tuning foundation models for IBM watsonx.ai and Red Hat Ansible Lightspeed with Watson Code Assistant

Red Hat | intel.

Red Hat Summit

Connect

Thank you

in linkedin.com/company/red-hat   f facebook.com/redhatinc

▶ youtube.com/user/RedHatVideos   🐦 twitter.com/RedHat

Red Hat

Resources

Resources

[Insights Advisor for OpenShift](#) – Red Hat Hybrid Cloud Console
[Red Hat Insights Upgrade Risks for Red Hat OpenShift](#) – Red Hat Blog

[Learn about OpenShift Sandboxed Containers](#) – Red Hat Blog
[Kata Containers Project](#)

[Migrate your VMs to OpenShift Virtualization using Migration Toolkit for Virtualization 2.4](#) – Red Hat Blog

[Hosted Control Planes](#)

[Nested OpenShift using OpenShift Virtualization](#) – Red Hat Blog