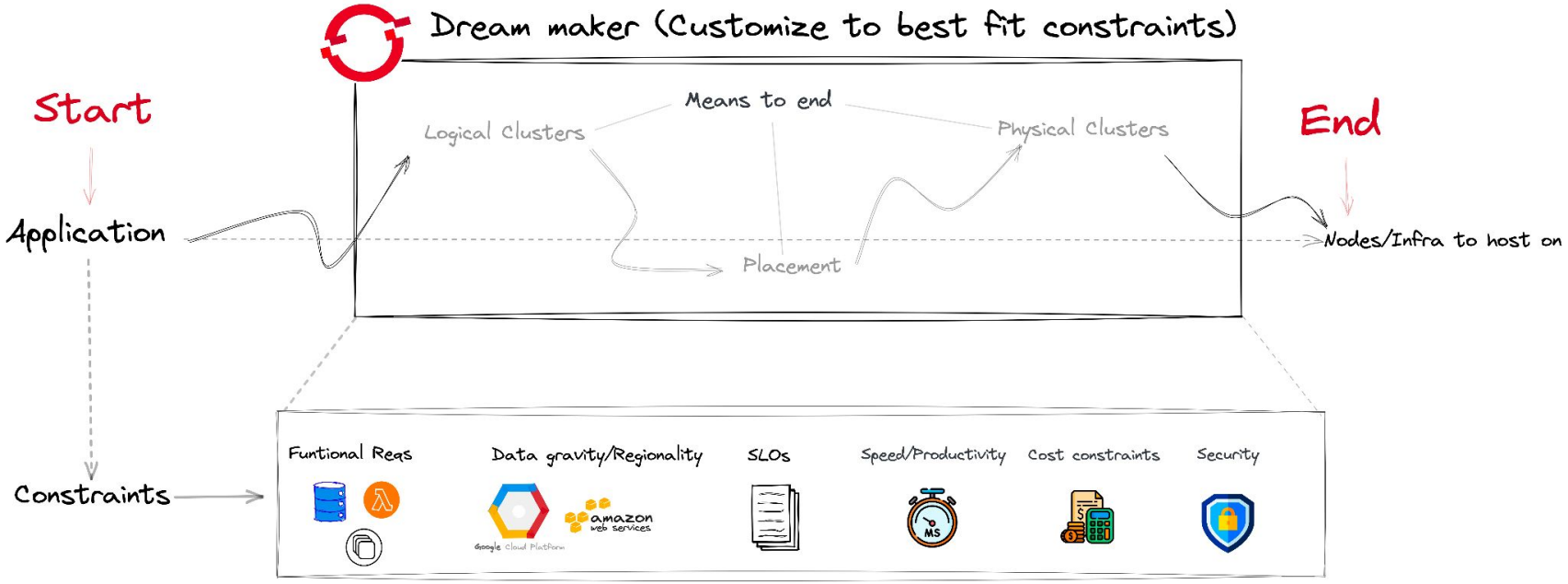
Red Hat OpenShift

Hosted Control Planes (Overview)

Management and Workload Decoupled

Red Hat

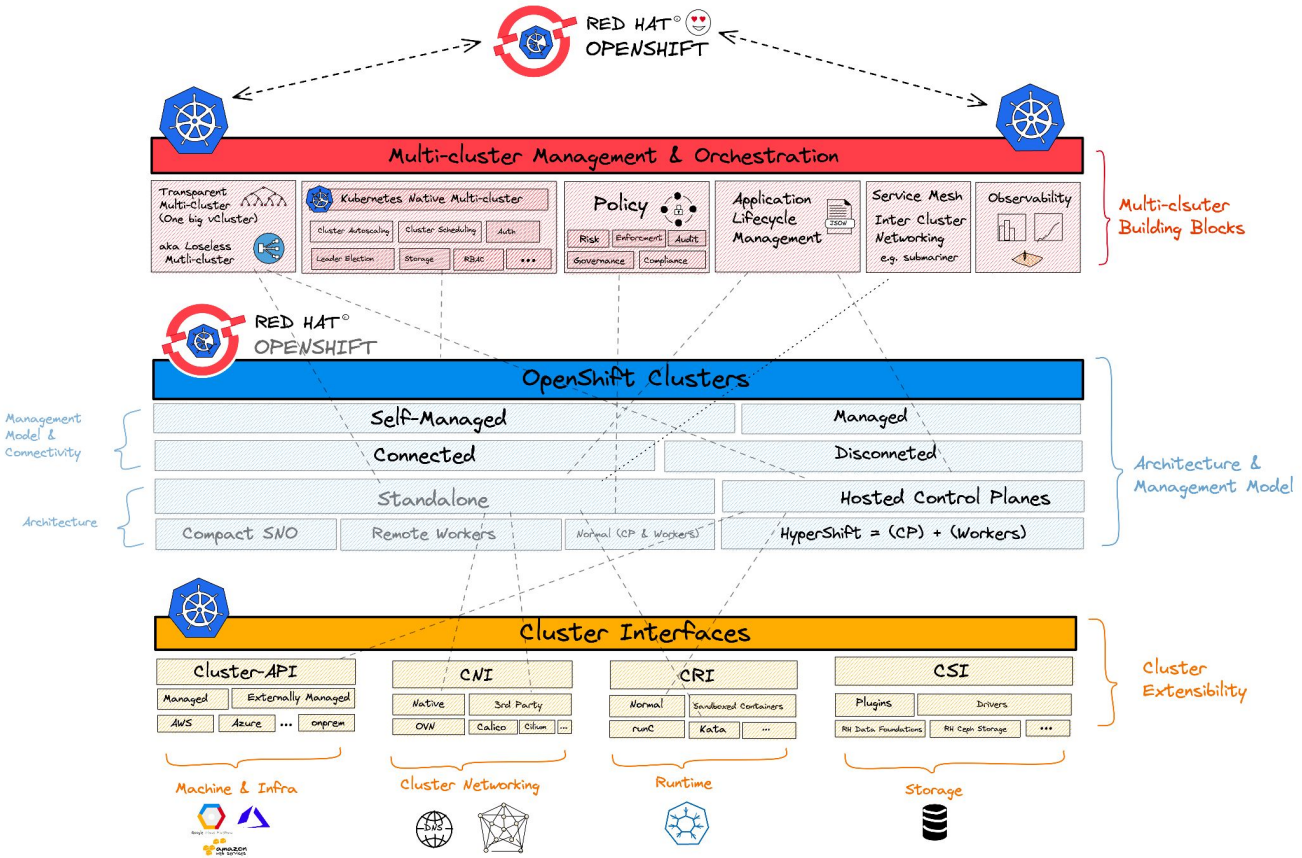
The Big Picture



Users expect applications and services to be available 24/7 — Companies desire more efficient use of cloud resources — Developers expect to deploy code multiple times a day with no downtime — Companies Must comply with Security standards / Regulation

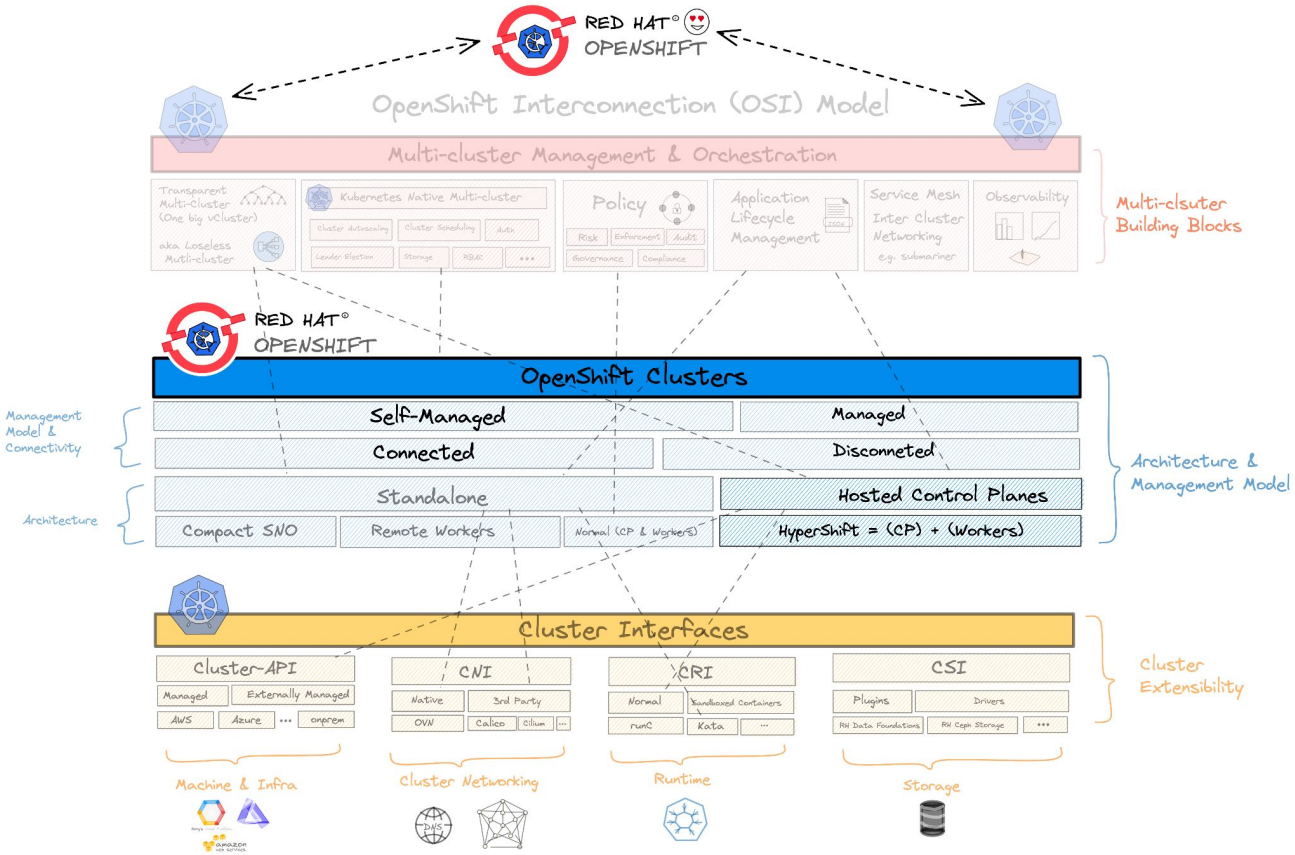
Red Hat

The Big Picture - Dream Maker (aka OpenShift) Tech Stack



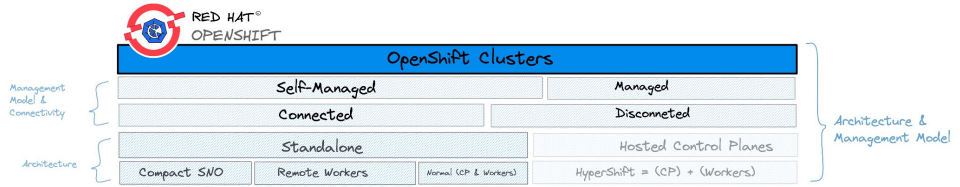
Red Hat

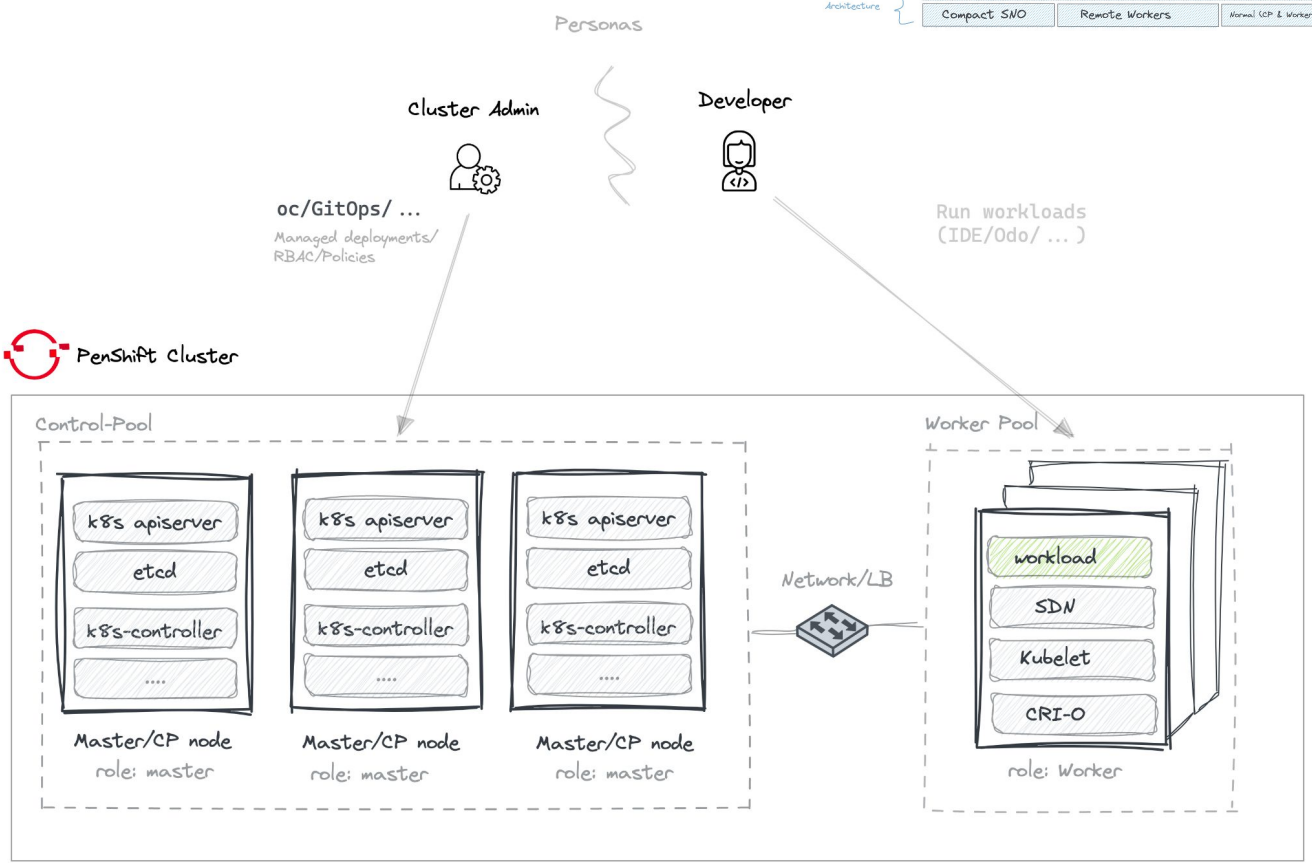
The Big Picture - Tech Stack



Red Hat

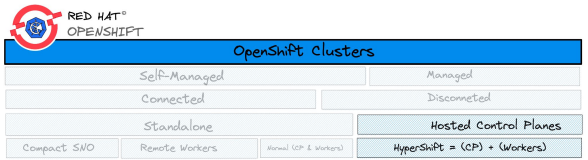
Standalone OpenShift



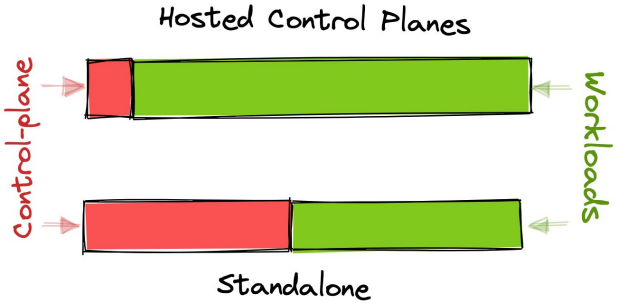


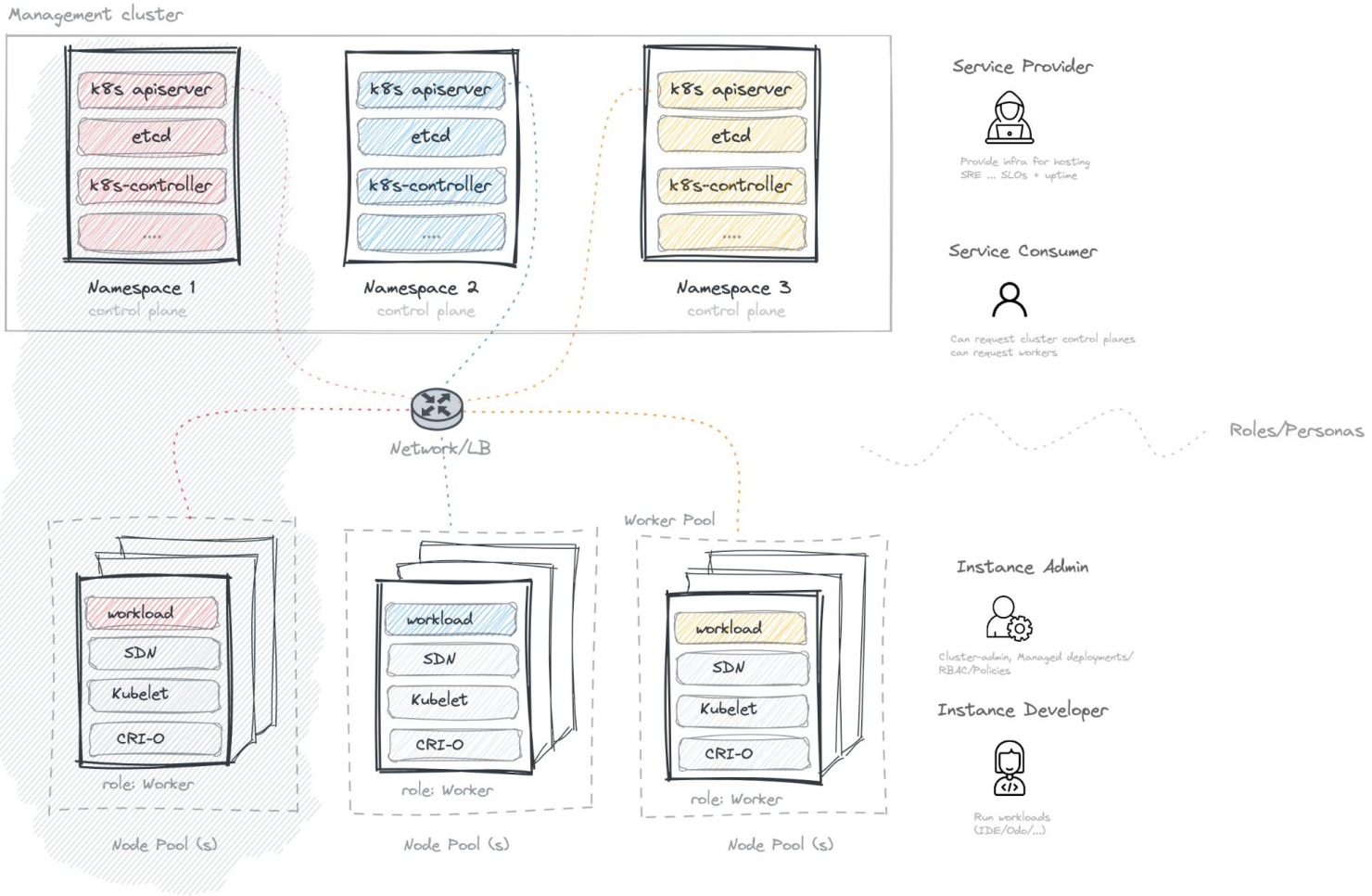
Red Hat

Hosted Control Planes (HyperShift)



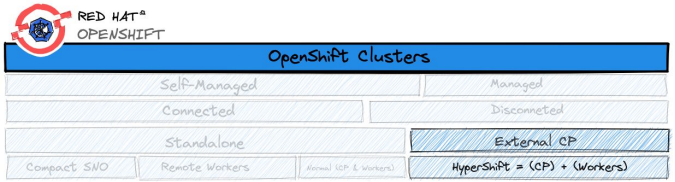
- An **OpenShift** Topology
- Service for **hosting OpenShift control planes** at **scale**
- Solves for **cost** and **time to provision**
- Portable **across clouds**
- Provides **strong separation of concerns** between management and workloads.




Management cluster
k8s apiserver
etcd
k8s-controller
....
Namespace 1
control plane
k8s apiserver
etcd
k8s-controller
....
Namespace 2
control plane
k8s apiserver
etcd
k8s-controller
....
Namespace 3
control plane
Service Provider
Provide infra for hosting
SRE ... SLOs + uptime
Service Consumer
Can request cluster control planes
can request workers
Roles/Personas
Network/LB
Worker Pool
workload
SDN
Kubelet
CRI-O
role: Worker
Node Pool (s)
workload
SDN
Kubelet
CRI-O
role: Worker
Node Pool (s)
workload
SDN
Kubelet
CRI-O
role: Worker
Node Pool (s)
Instance Admin
cluster-admin, Managed deployments/
RBAC/Policies
Instance Developer
Run workloads
(IDE/Odo/...)
OpenShift Cluster

Red Hat

Why HyperShift?



Control plane and workload could run on different providers

Mixed Management & Worker IaaS

Control-plane and Workloads could run on different architectures

Heterogeneous Arch Clusters

Logically centralised management of control-planes

Easier Multi-cluster Management

100s of cluster control planes on the same physical cluster, just a namespace away. Single Pane of Glass, Happy SREs

Easier Operability

What's Possible?

HyperShift Enabler Features

No waiting for machines

Immediate clusters

Multiple Control Planes per node vs 3 nodes for 1 CP

Cheaper Control-planes

Control-Plane as pods, re-use all K8s Goodies

Kubernetes-in-Kubernetes

Isolate tenants, seal away credentials/keys, and prevent users from shooting themselves in the foot

Trust Segmentation & Human Error Reduction

Possible to upgrade Management and WKLOAD separately (e.g., CVE, Patches,...)

Decoupled Life-cycle Management

Red Hat

Short Stories / Use-cases

It takes longer than expected for me to create a cluster

I don't want Workload admins/Developers to have access to control-planes

It takes alot of compute to host my control-plane

- CP as workloads
- Scael, monitor, log, secure,...

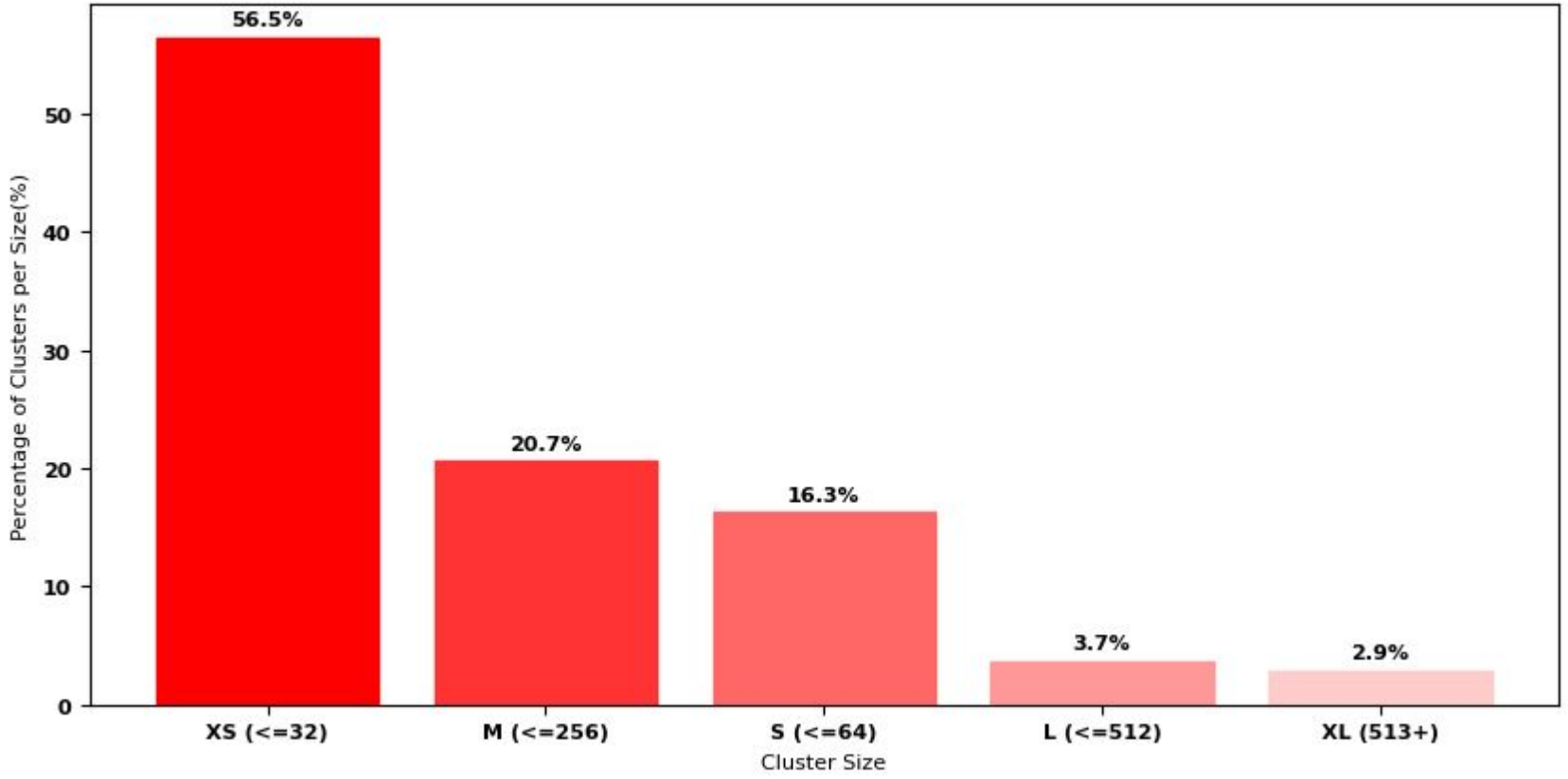
I have one large server, but It's too much to host just one cluster (Wasted resources)

Prevent CRD conflicts between tenants.

- Same operator, different version

I want more space for workloads on my cluster

Cluster Sizes Trending Down, Cluster Count UP!

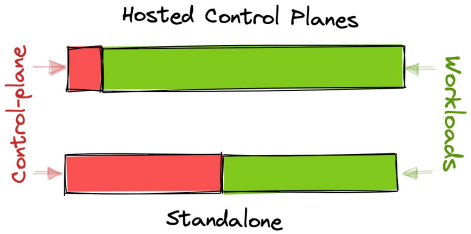


Red Hat

Why HCP?

HCP

- Supported OpenShift topology
- Reduced infrastructure costs / densification
- Faster cluster creation
- Strong separation between control and workload
- Support multi-arch / multi-env
- Centralized management in a "Managed" model



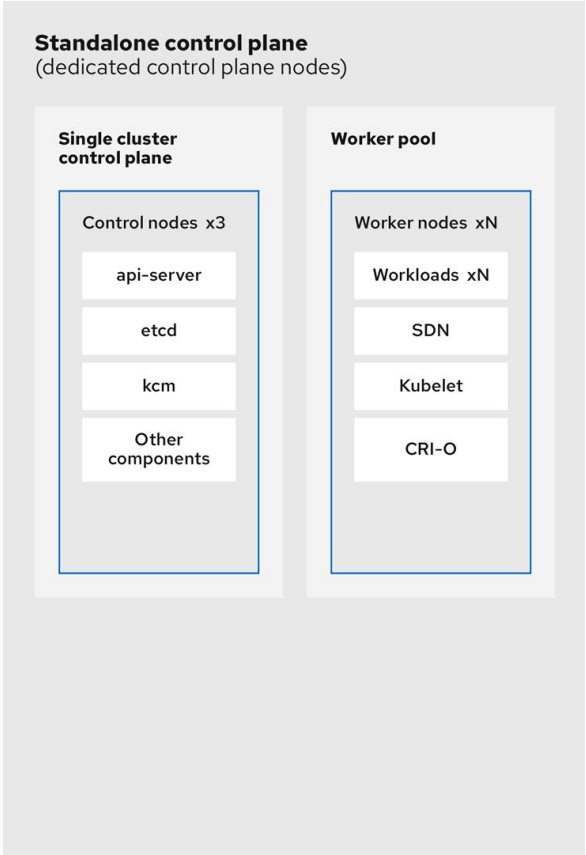
Red Hat

Red Hat OpenShift

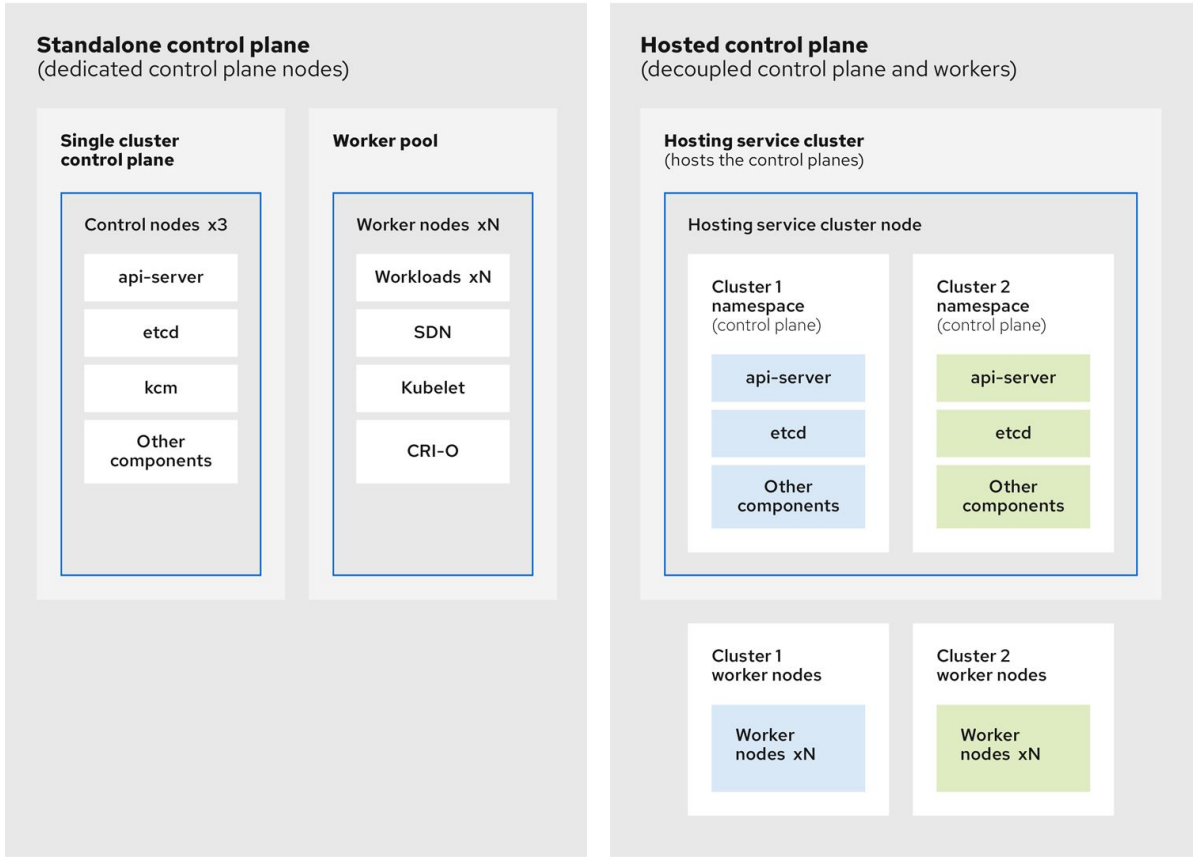
HCP Architecture & Support

Red Hat

Standalone Control Plane



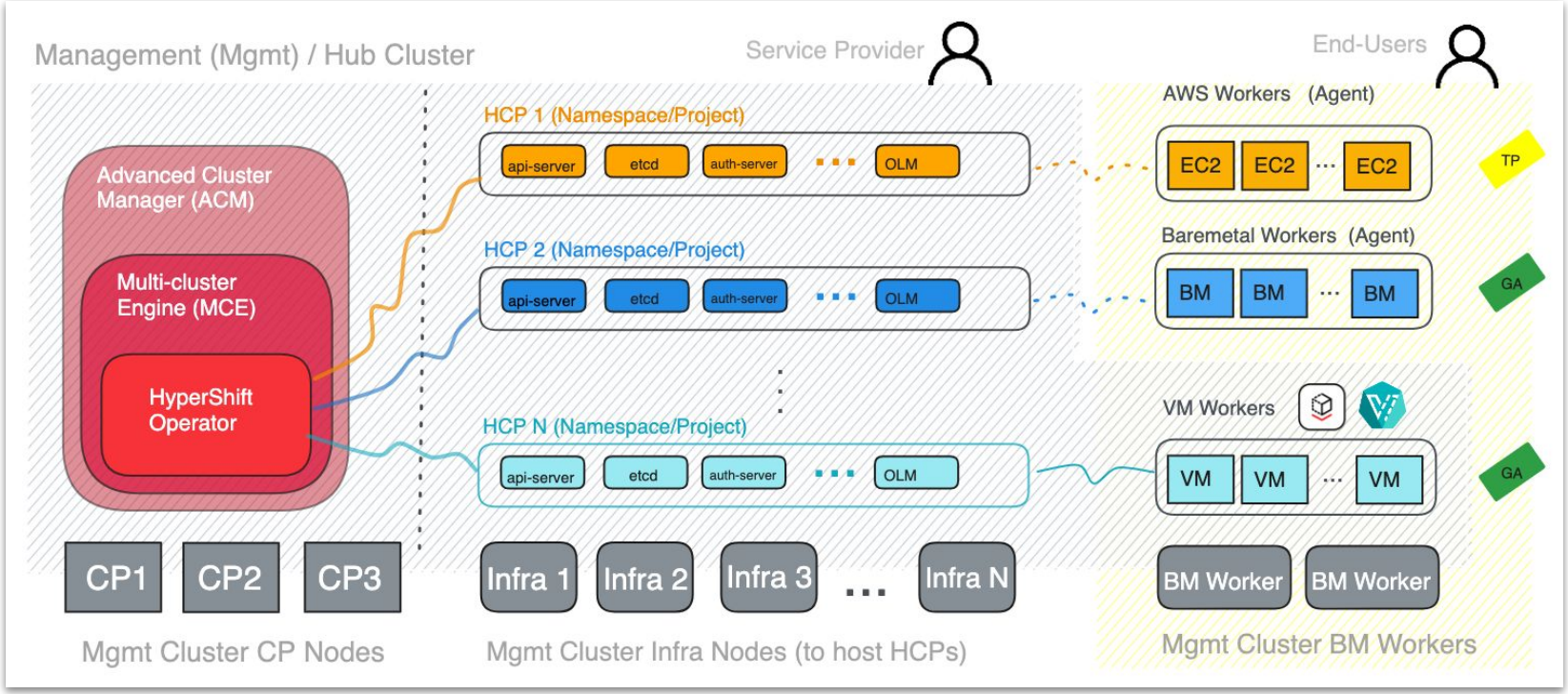
Hosted Control Plane



Management

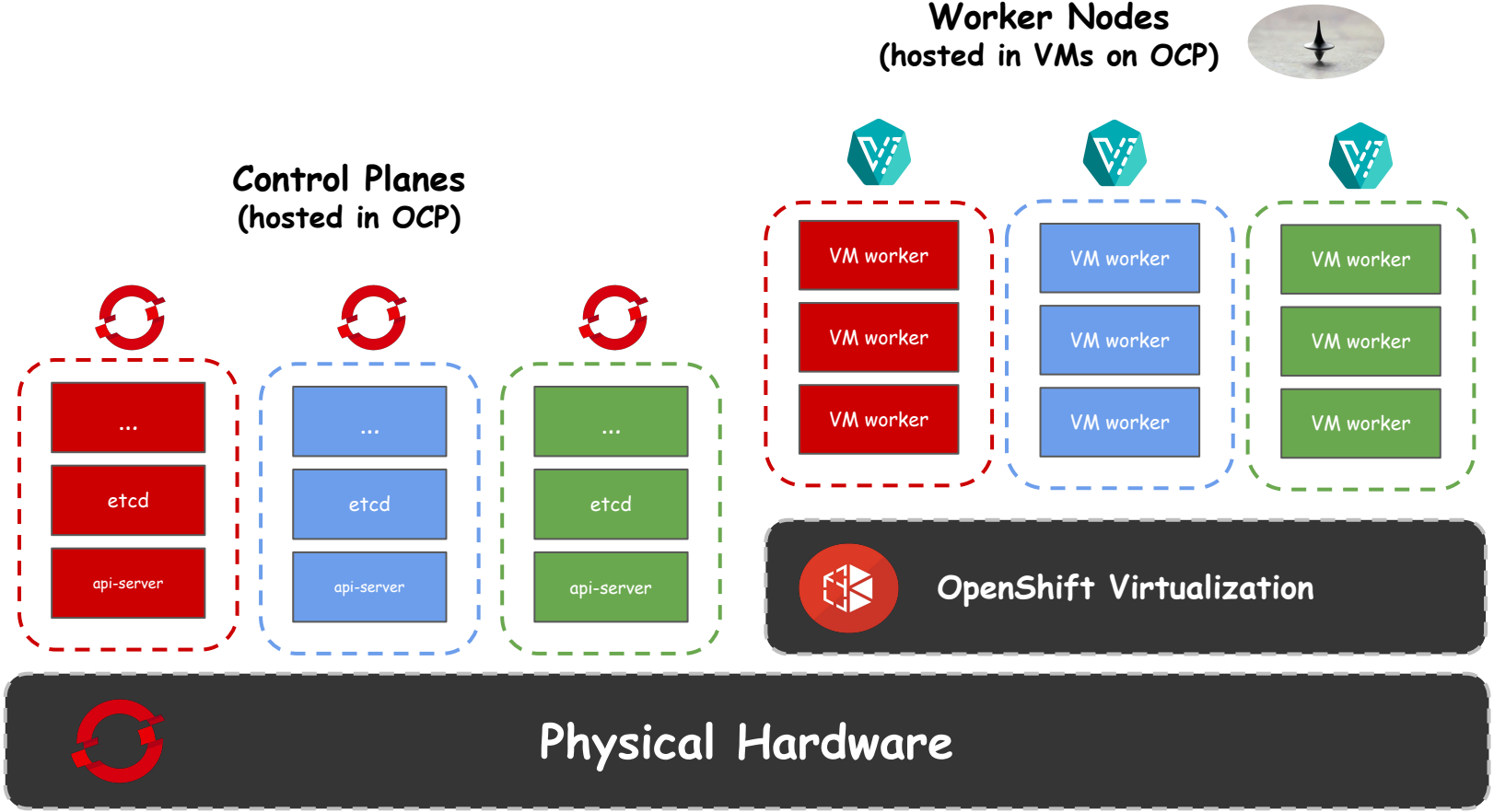
- Hosted Control Plane:
  - Management of the cluster happens from the hosting cluster, allowing a clearer breakdown of responsibilities/personas
  - Interaction happens through the following Kubernetes objects:
    - HostedCluster which represents the control plane.
    - NodePool which represents the workers.
    - MachineConfigs which are embedded as part of the NodePool specification.

Architecture Overview



Red Hat

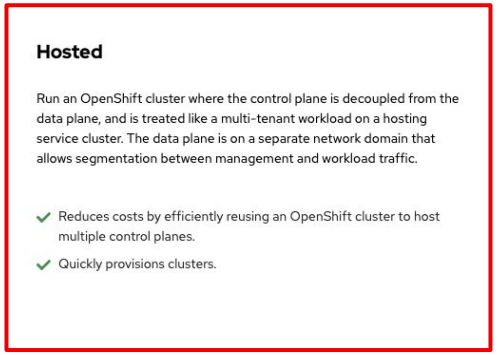
The OpenShift Virtualization Provider


Worker Nodes
(hosted in VMs on OCP)
Control Planes
(hosted in OCP)
...
etcd
api-server
...
etcd
api-server
...
etcd
api-server
VM worker
VM worker
VM worker
VM worker
VM worker
VM worker
VM worker
VM worker
VM worker
OpenShift Virtualization
Physical Hardware

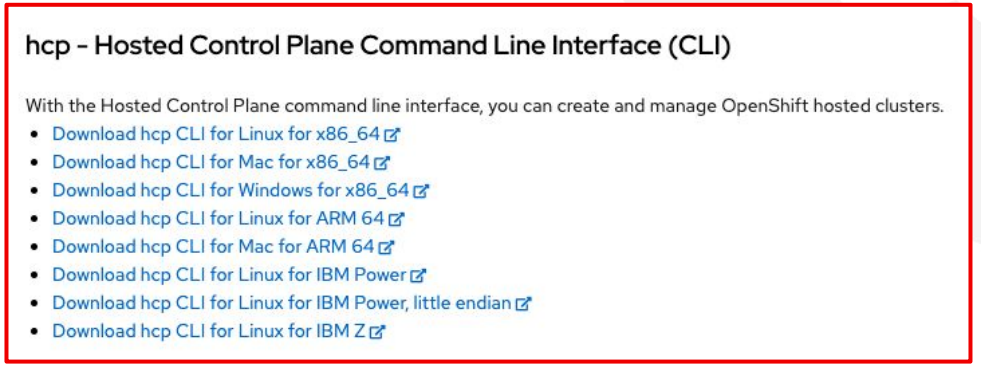
Red Hat

Create HCP cluster

Via ACM WebUI



Dedicated CLI



Contributor: David Martini

Red Hat

Create HCP cluster

Time Provisioning = ~10 min

```
export CLUSTER_NAME=hcp01
export PULL_SECRET="./pull-secret"
export SSH_KEY="./dm_key.pub"
export MEM="8Gi"
export CPU="4"
export WORKER_COUNT="3"
export BASE_DOMAIN=drkspace.fr
export CP_DEPLOYMENT_MODE="SingleReplica"
export INFRA_DEPLOYMENT_MODE="SingleReplica"
```

Variables definition

```
hcp create cluster kubevirt \
--name $CLUSTER_NAME \
--release-image $RELEASE_IMAGE \
--node-pool-replicas $WORKER_COUNT \
--pull-secret $PULL_SECRET \
--ssh-key $SSH_KEY \
--memory $MEM \
--cores $CPU \
--control-plane-availability-policy $CP_DEPLOYMENT_MODE \
--infra-availability-policy $INFRA_DEPLOYMENT_MODE
```

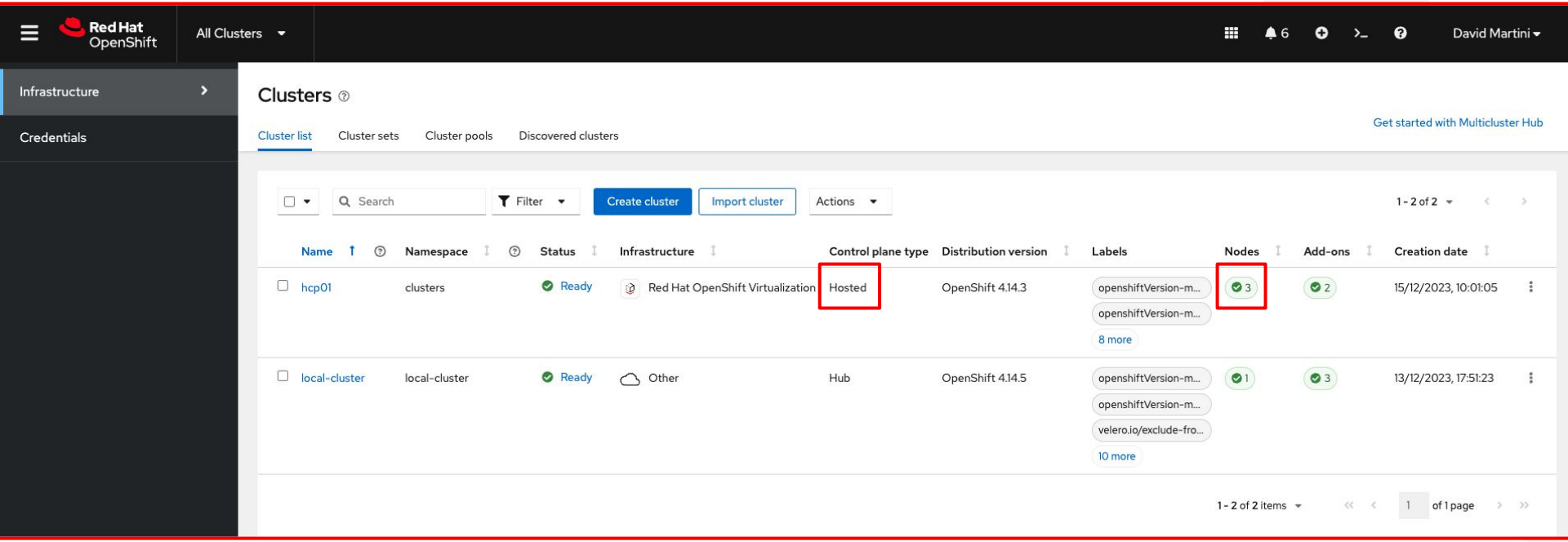
Cluster creation

https://github.com/davmartini/redhat-techs/tree/main/openshift/hcp

Contributor: David Martini

Red Hat

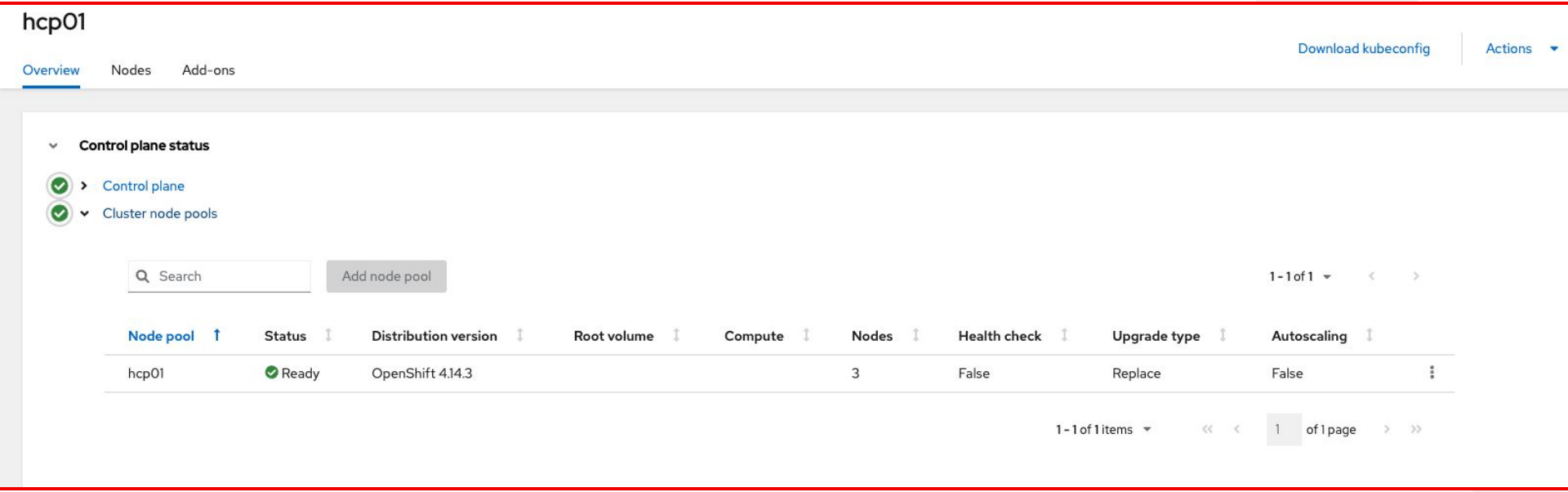
Create HCP cluster



Contributor: David Martini

Red Hat

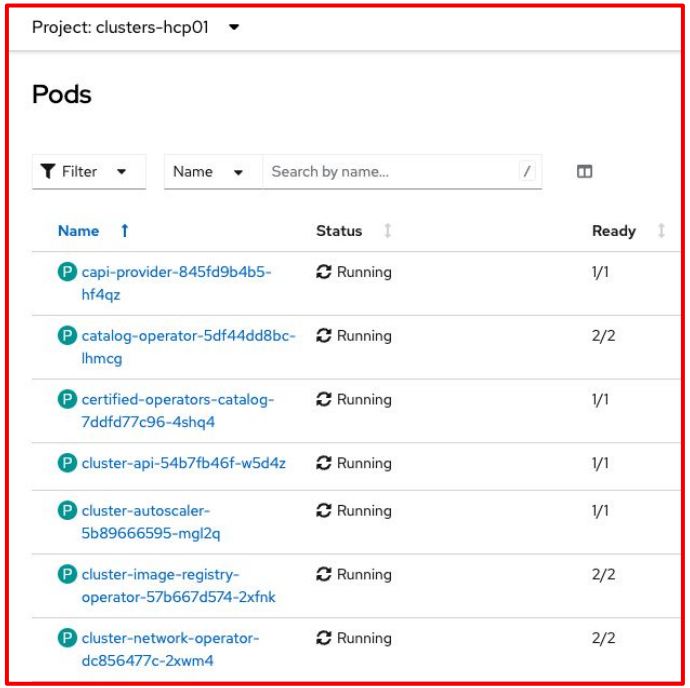
Create HCP cluster



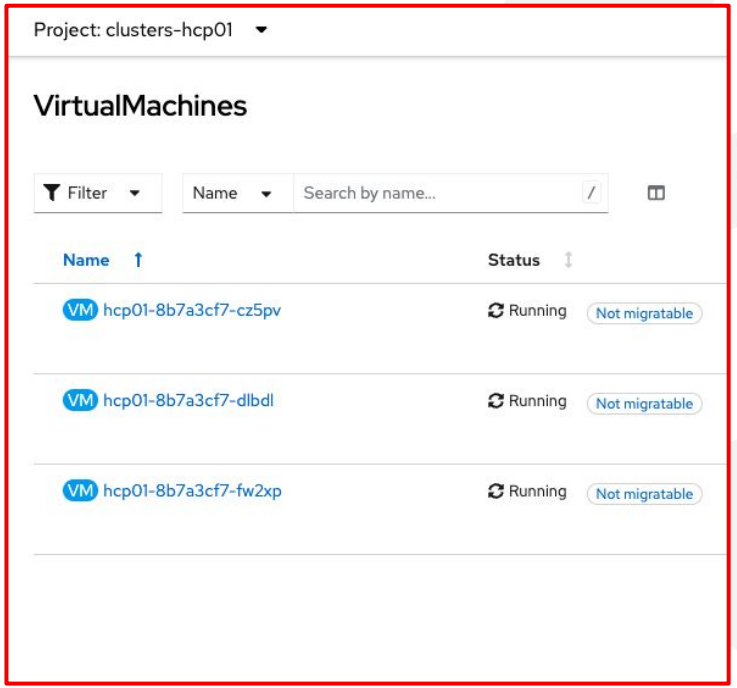
Contributor: David Martini

Red Hat

HCP cluster on Management Cluster



Control Plane Pods



Data Plane VMs

Contributor: David Martini

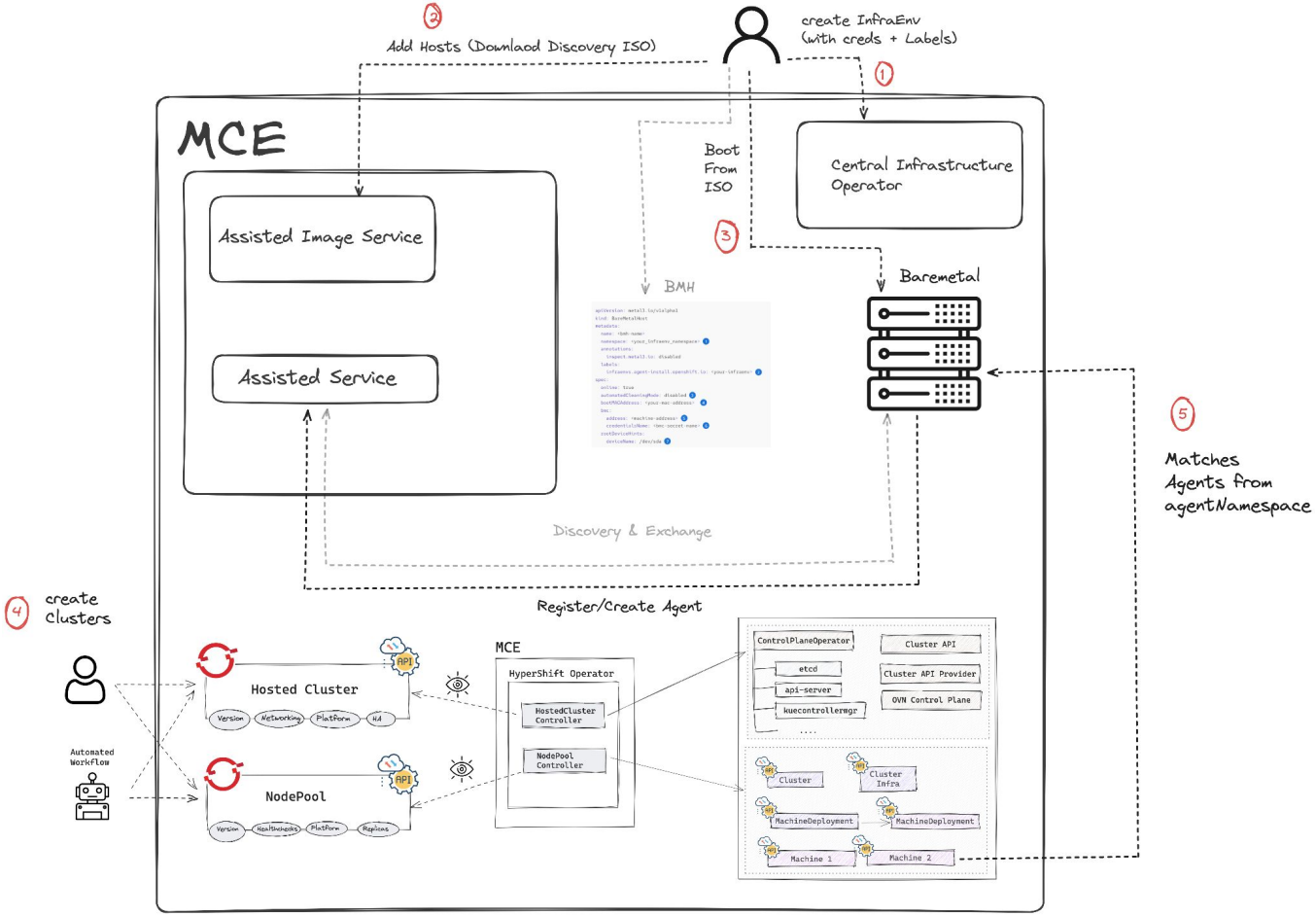
Red Hat

Red Hat OpenShift

The Agent Provider

Red Hat

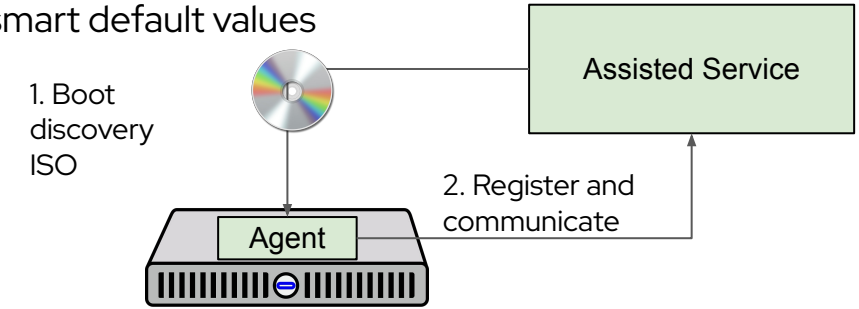
Deep-Dive Cluster-API-Agent Flow



Red Hat

Assisted Installer Overview

- Most installers take the desired configuration as input and attempt to install. If the installation fails, the administrator gathers logs to understand the cause, possibly changes the inputs, and tries again.
- The Assisted Installer is different:
  - An installation service provides a central point of contact, from defining the configuration, to monitoring progress, to downloading logs.
  - The administrator downloads a *discovery ISO* from the service and boots hosts with it.
  - Each host runs an agent process that communicates with the installation service (discovery phase).
  - The service gathers information from the agents regarding the actual environment and uses it to
    - Validate inputs *before* the installation begins
    - Suggest smart default values



Red Hat

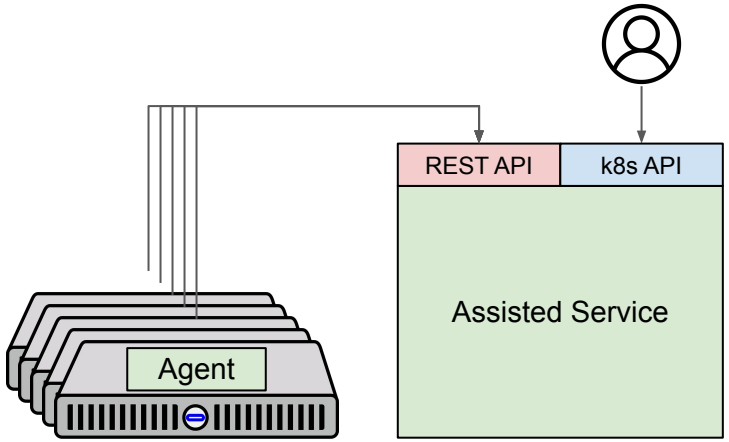
Infrastructure Operator APIs

The service exposes two kinds of APIs:

1. REST: used by agents, as well as users for downloading images, logs, fetching events, etc.
2. Kubernetes-native (Custom Resources): used by users/automation for provisioning

The operator defines 3 CRDs:

- **AgentClusterInstall** (an extension of Hive's ClusterDeployment)
- **Agent** (represents an agent running on a discovered host)
- **InfraEnv** (represents the infrastructure environment encapsulated in a Discovery ISO)



Persona 1: Infrastructure Administrator

The Infrastructure Administrator performs three main tasks:

1. Defines one or more InfraEnvs
2. Downloads the discovery image from an InfraEnv and boots hosts with it (this may be automated with external tooling)
    a. Once a host boots with the discovery image, the agent running on it will register with the service, creating an Agent CR
3. Optionally sets properties on the Agents (e.g., hostname, installation disk) and approves them for use

Persona 2: Cluster Creator

The Cluster Creator:

1. Defines a HyperShift HostedCluster of type “agent”
    a. Specifies the namespace where the relevant Agent CRs are located
    b. As a side effect, an operator called cluster-api-provider-agent is installed in the HostedCluster’s namespace
2. Defines a HyperShift NodePool
3. Scales up the NodePool, which creates CAPI Machines
    a. For each CAPI Machine, cluster-api-provider-agent will find a suitable Agent (unused, approved, matching specified labels). Once it finds a suitable Agent the Assisted Service will run additional validations and, if successful, install the host.

Red Hat OpenShift

Failure Modes & DR

Red Hat

Failure Scenarios

Failure	Result
Loss of management cluster worker	Hosted control plane API is **still available.** Hosted cluster data plane is **still available.** Impacted hosted control plane member is **rescheduled**.
Loss of management cluster availability zone	Hosted control plane API is **still available but degraded.** Hosted cluster data plane is **still available.** Impacted hosted control plane **maintains quorum.**
Loss of management cluster control plane	Hosted control plane API is **still available.** Hosted cluster data plane is **still available.**
Loss of management cluster control plane and workers	Hosted control plane API is **not available.** Hosted cluster data plane is **still available.**

See a live demo of these situations!

Failure Scenarios Compact Nodes (Not Recommended)

Failure	Result
Loss of management cluster worker	Hosted control plane API is **still available but degraded.** Hosted cluster data plane is **still available.** Impacted hosted control plane member is **rescheduled**.
Loss of management cluster availability zone	Hosted control plane API is **still available but degraded.** Hosted cluster data plane is **still available.** Impacted hosted control plane **maintains quorum.**
Loss of management cluster control plane	Hosted control plane API is **not available.** Hosted cluster data plane is **still available.**
Loss of management cluster control plane and workers	Hosted control plane API is **not available.** Hosted cluster data plane is **still available.**

See a live demo of these situations!

Roadmap

Red Hat OpenShift

Project

Hypershift

Product

Hosted Control Plane (HCP)

GA

OCP 4.14 / ACM 2.9 / MCE 2.4

Release date : **11/2023**

GA BM

GA OCP Virt

TP AWS

Red Hat

Contributor: David Martini

Red Hat OpenShift

Hypershift → Hosted Control Plane (HCP)

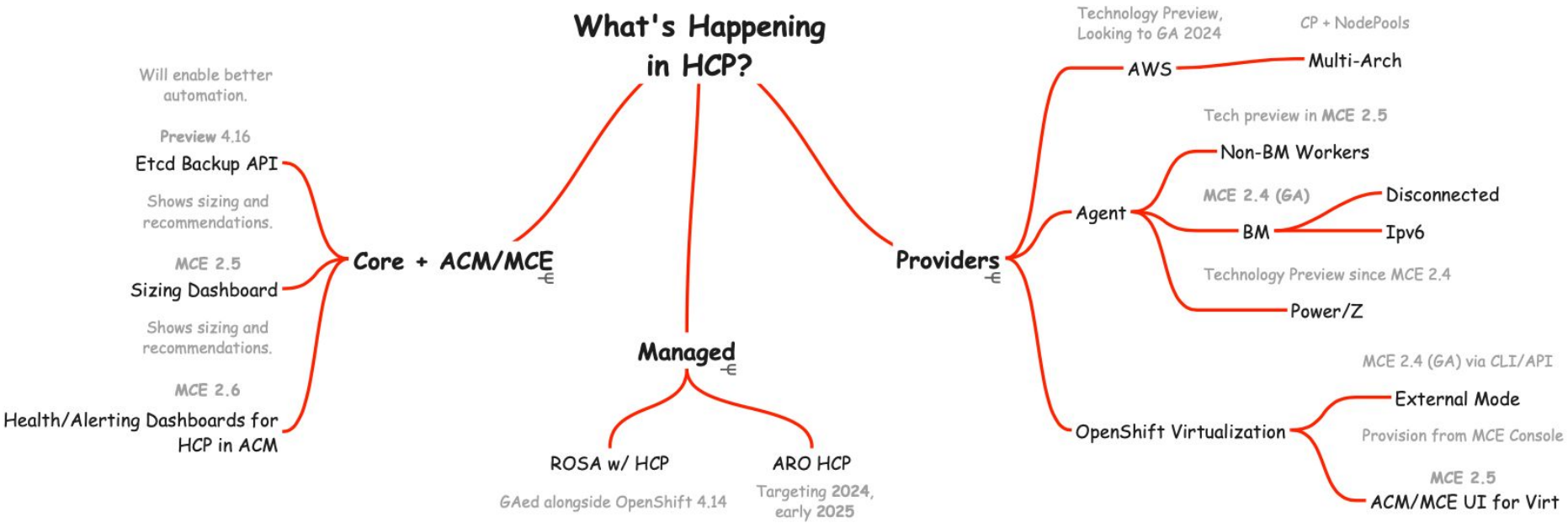
GA

Red Hat OpenShift Service on AWS (ROSA)

Release date : 4/11/2023

Contributor: David Martini

What's Happening in HCP?



Red Hat

Thank you!