Red Hat Partner Bootcamp

OpenShift Topics

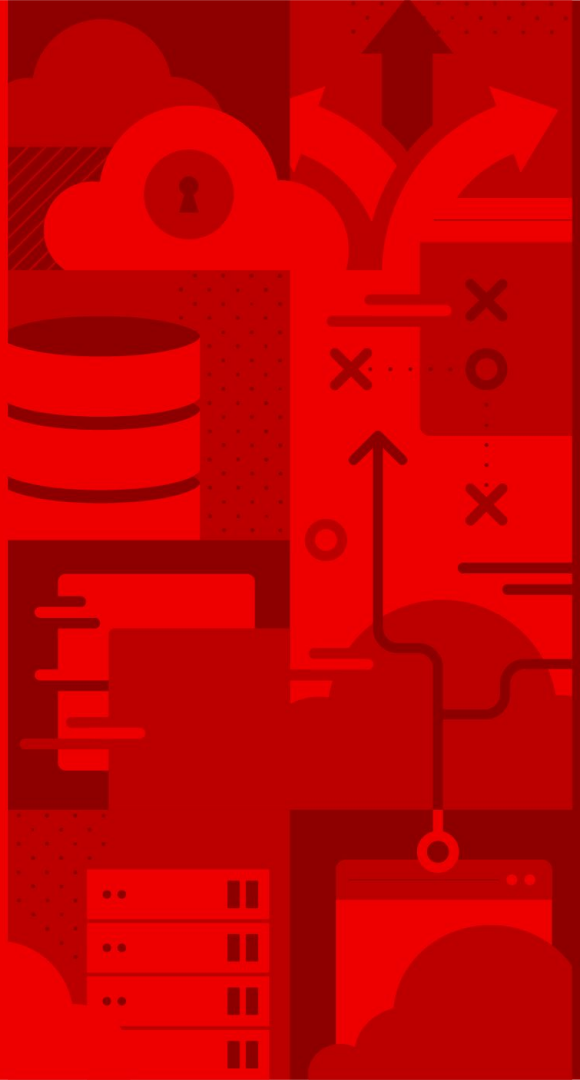
Alfred Bach
Principal Solution
Architect
Red Hat EMEA

Agenda

- Innovation without limitation
- OpenShift Architecture
- OpenShift Container Platform (OCP) Install
- OpenShift Plus

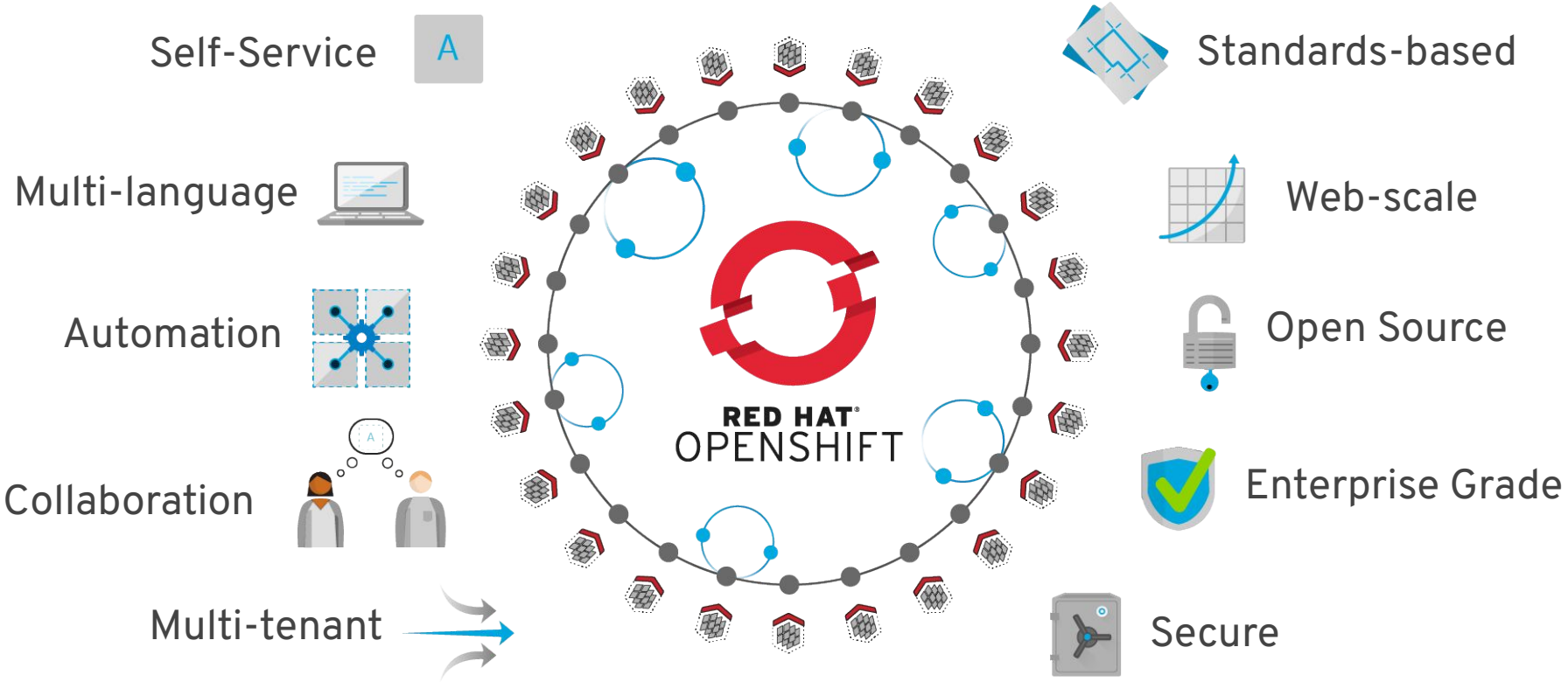
Agenda

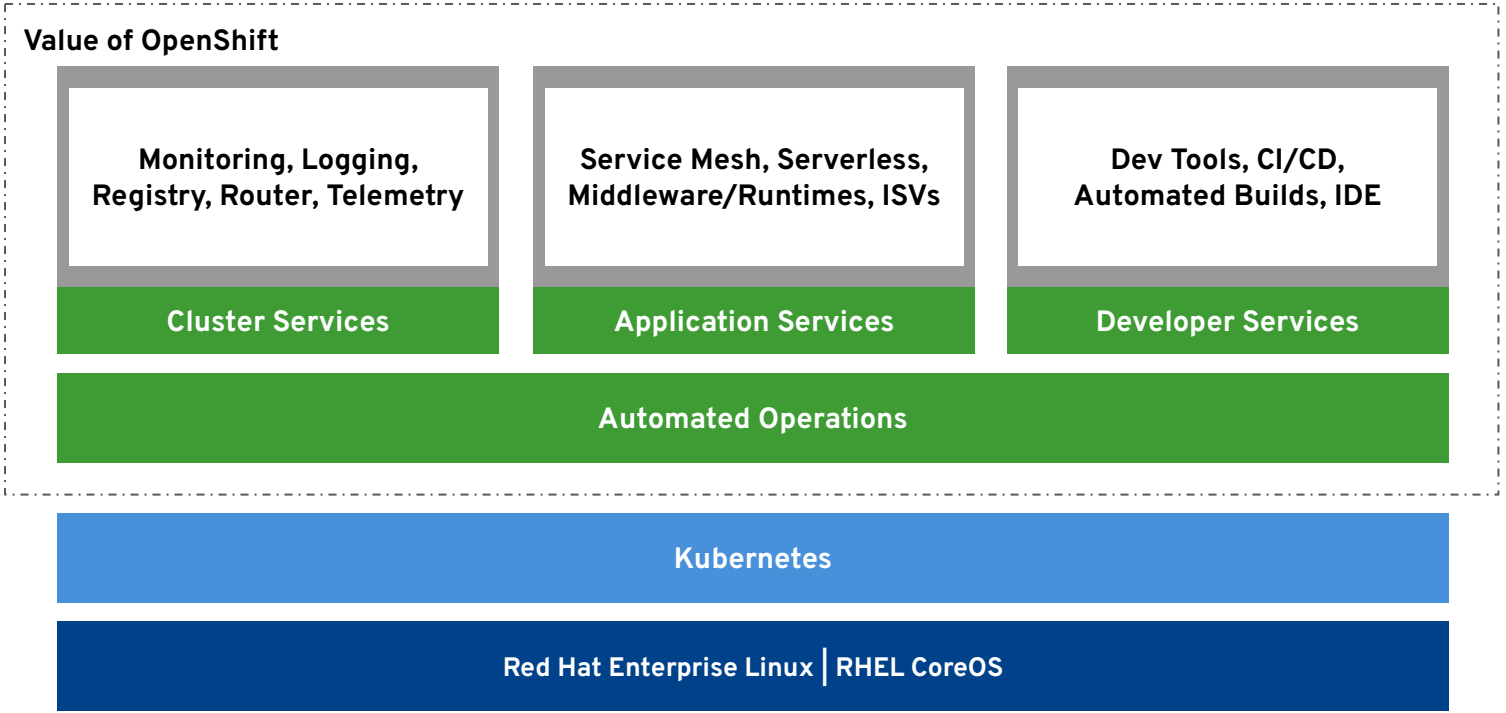
- Innovation without limitation
- **OpenShift Architecture**
- OpenShift Container Platform (OCP) Install
- OpenShift Plus



Functional overview

Red Hat


Self-Service
Multi-language
Automation
Collaboration
Multi-tenant
RED HAT
OPENSHIFT
Standards-based
Web-scale
Open Source
Enterprise Grade
Secure

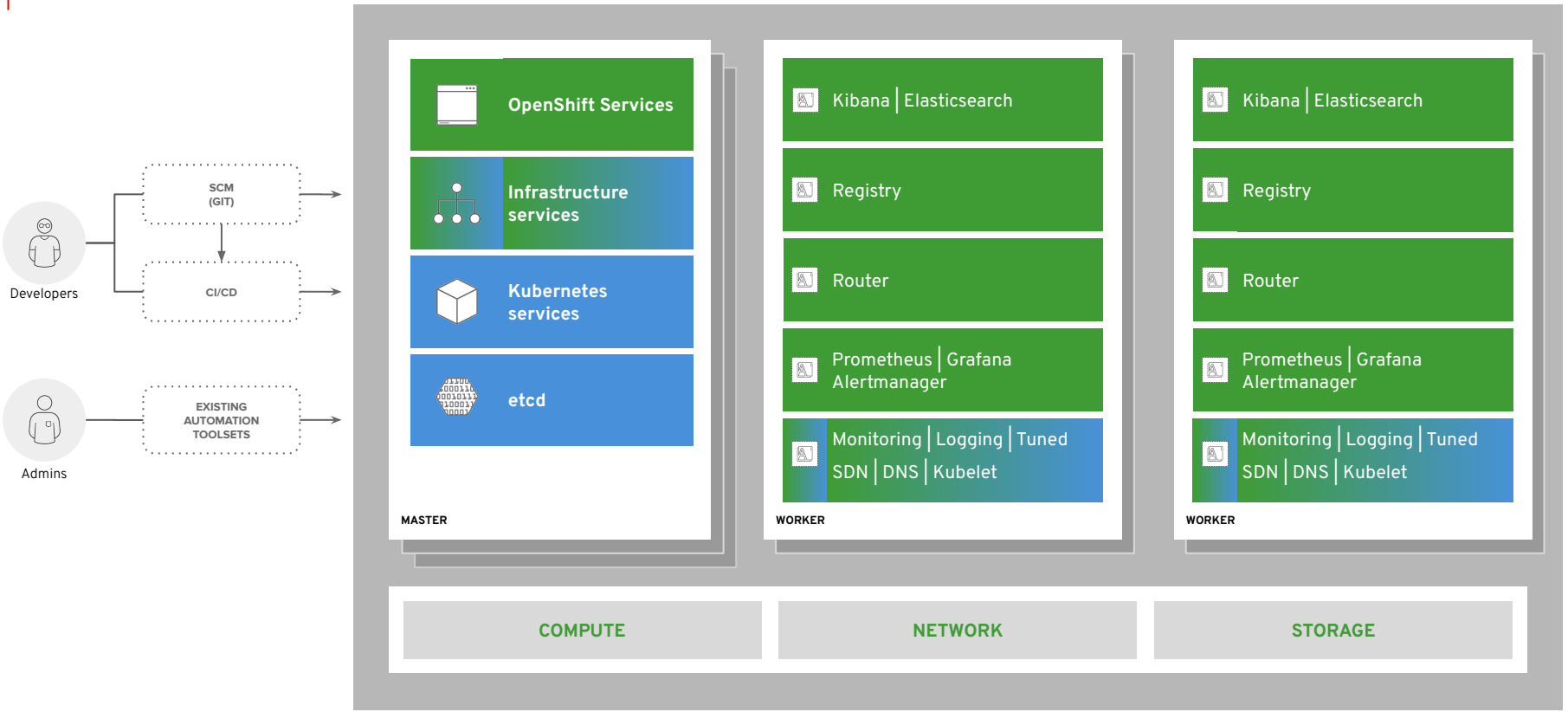


Best IT Ops Experience

CaaS ⟷ PaaS ⟷ FaaS

Best Developer Experience

Red Hat


SCM (GIT)
CI/CD
Developers
EXISTING AUTOMATION TOOLSETS
Admins
OpenShift Services
Infrastructure services
Kubernetes services
etcd
MASTER
Kibana | Elasticsearch
Registry
Router
Prometheus | Grafana Alertmanager
Monitoring | Logging | Tuned
SDN | DNS | Kubelet
WORKER
Kibana | Elasticsearch
Registry
Router
Prometheus | Grafana Alertmanager
Monitoring | Logging | Tuned
SDN | DNS | Kubelet
WORKER
COMPUTE
NETWORK
STORAGE

Red Hat

CNCF Cloud Native Landscape
2019-07-12T03:51:02Z 024e93b

Overwhelmed? Please see the CNCF Trail Map. That and the interactive landscape are at l.cncf.io

Greyed logos are not open source

App Definition and Development

Database

Streaming & Messaging

Application Definition & Image Build

Continuous Integration & Delivery

Orchestration & Management

Scheduling & Orchestration

Coordination & Service Discovery

Remote Procedure Call

Service Proxy

API Gateway

Service Mesh

Runtime

Cloud-Native Storage

Container Runtime

Cloud-Native Network

Provisioning

Automation & Configuration

Container Registry

Security & Compliance

Key Management

Cloud

Public

Special

Kubernetes Certified Service Provider

Platform

Certified Kubernetes - Distribution

Certified Kubernetes - Hosted

Certified Kubernetes - Installer

PaaS/Container Service

Kubernetes Training Partner

Observability and Analysis

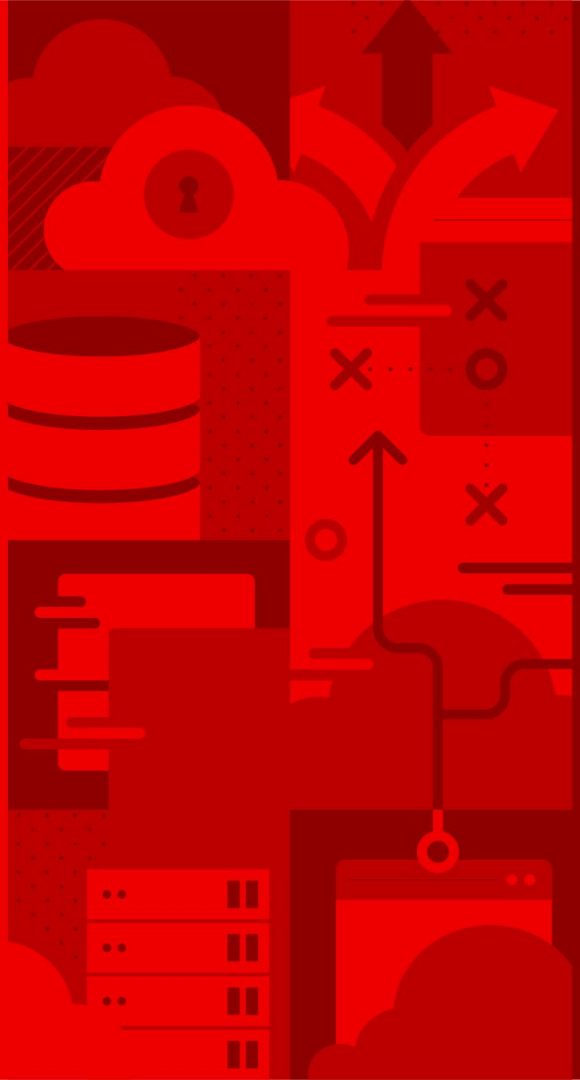
Monitoring

Logging

Tracing

Chaos Engineering

Serverless

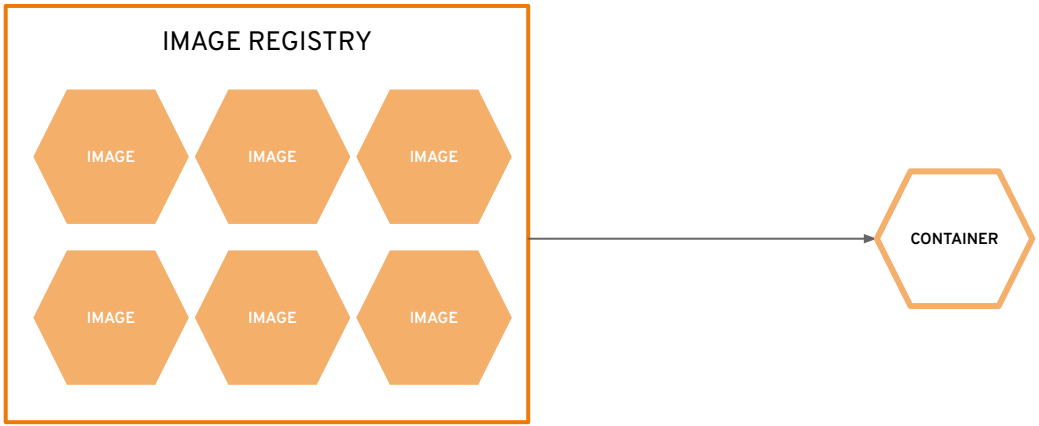


OpenShift and Kubernetes core concepts

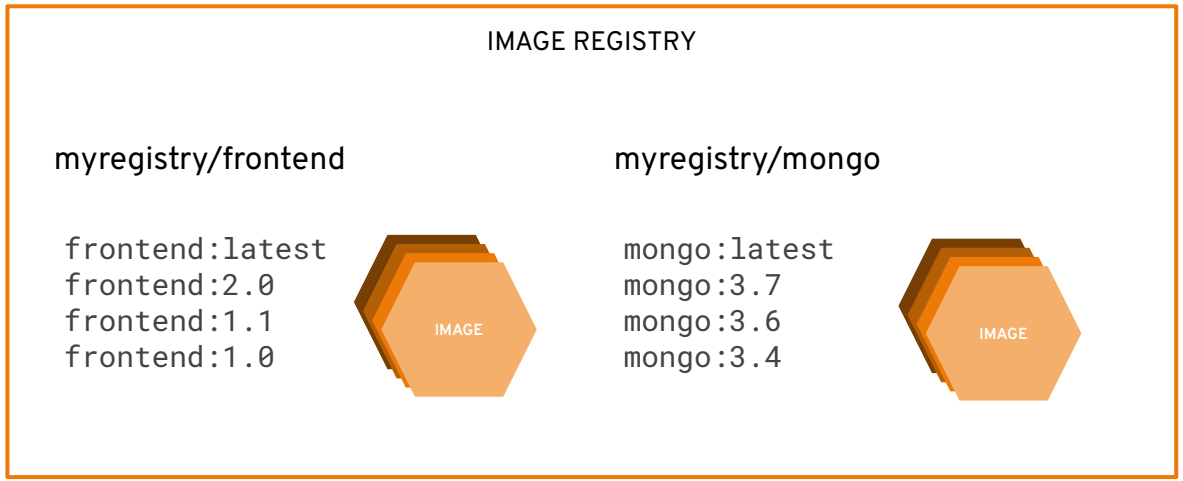
a container is the smallest compute unit

containers are created from container images

container images are stored in an image registry

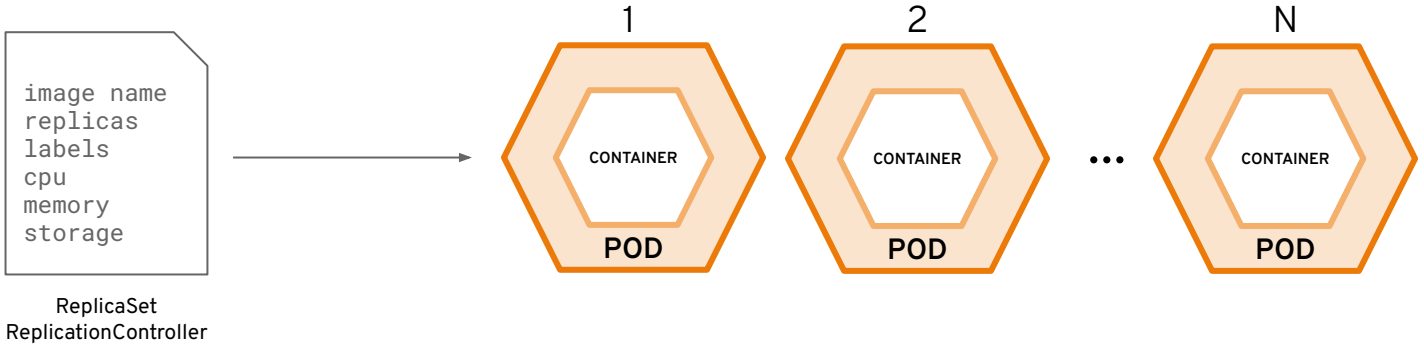


an image repository contains all versions of an image in the image registry

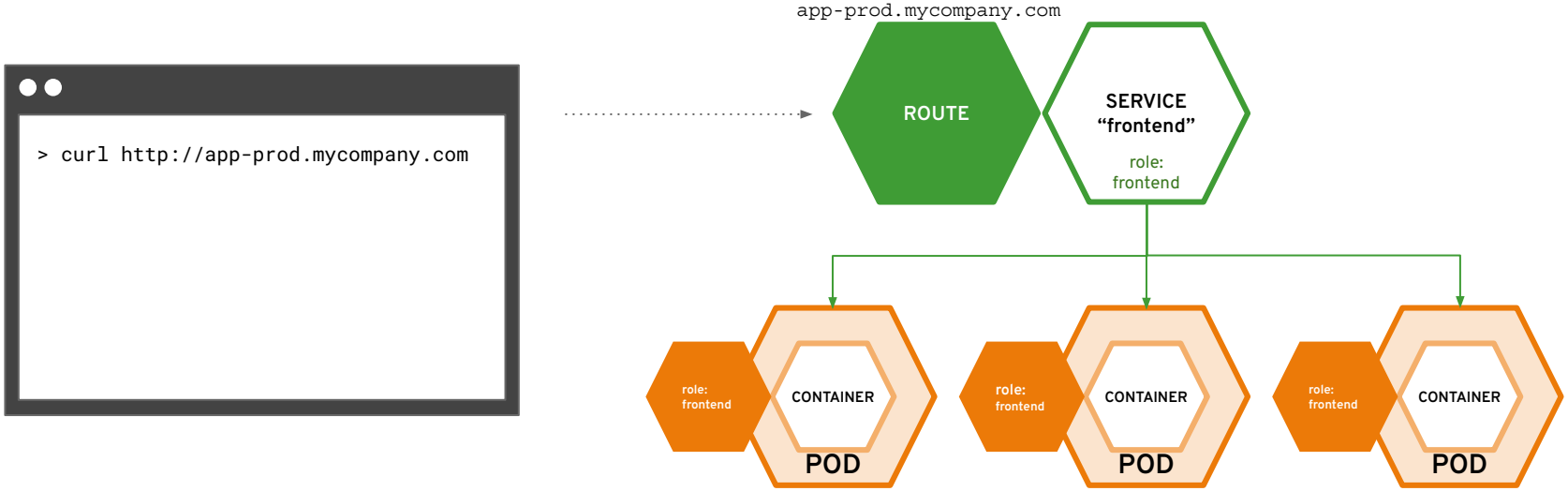


containers are wrapped in pods which are units of deployment and management

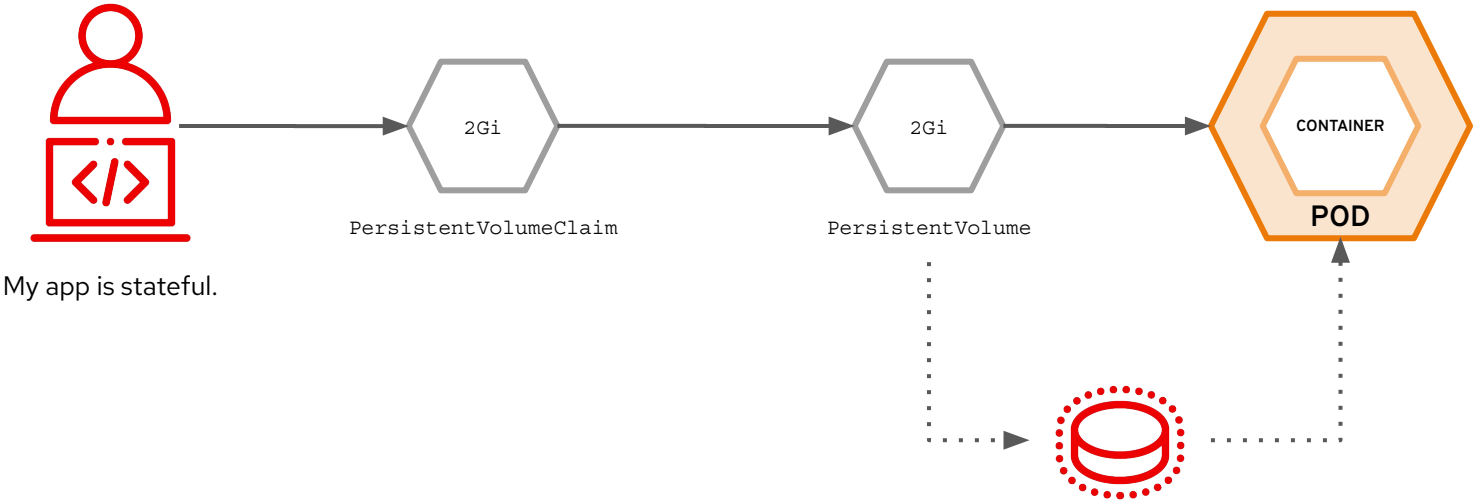
ReplicationControllers & ReplicaSets ensure a specified number of pods are running at any given time

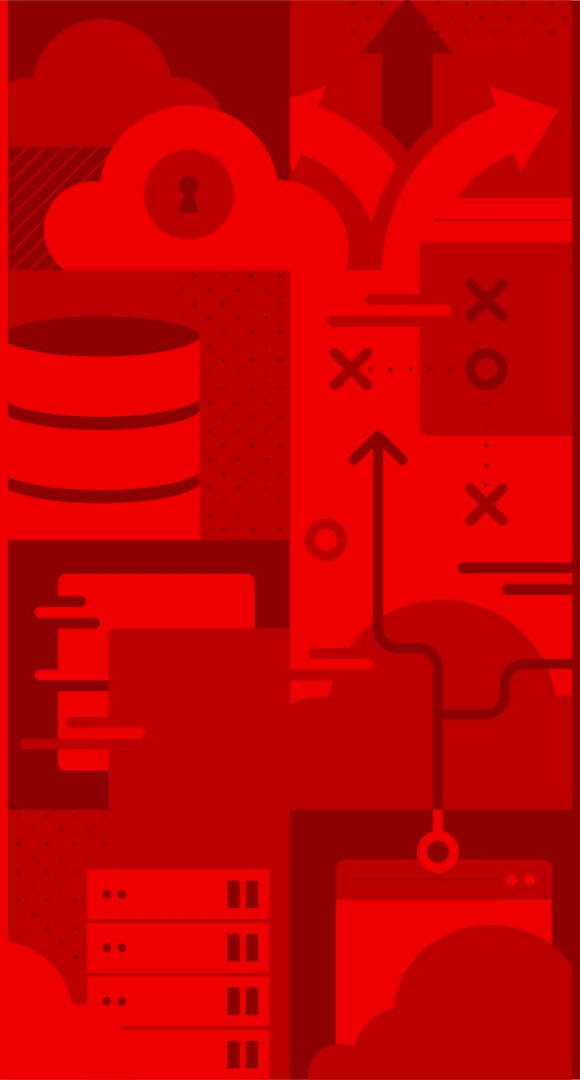


`routes` make services accessible to clients outside the environment via real-world urls



Persistent Volume and Claims





OpenShift 4 Architecture

Red Hat

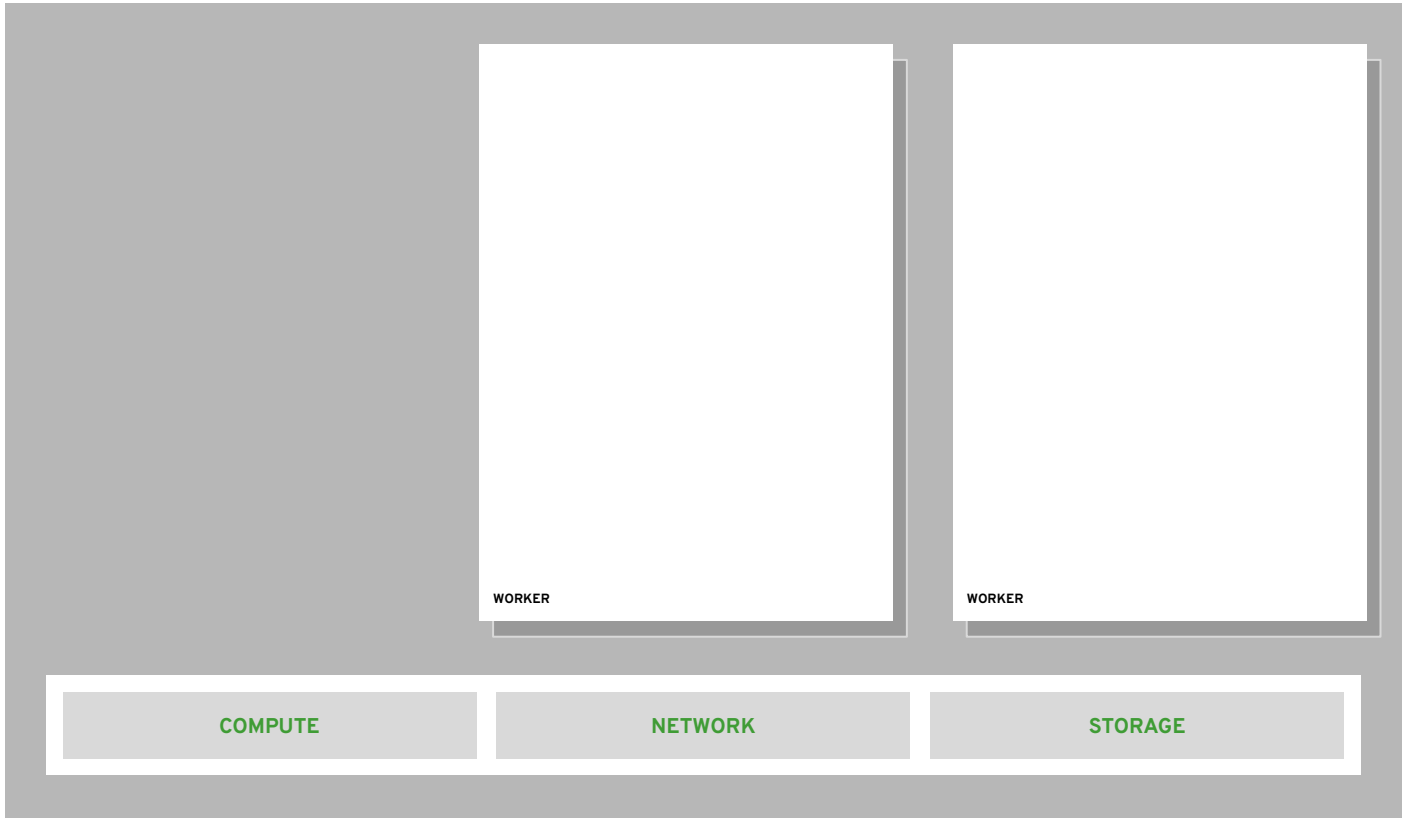
your choice of infrastructure

COMPUTE

NETWORK

STORAGE

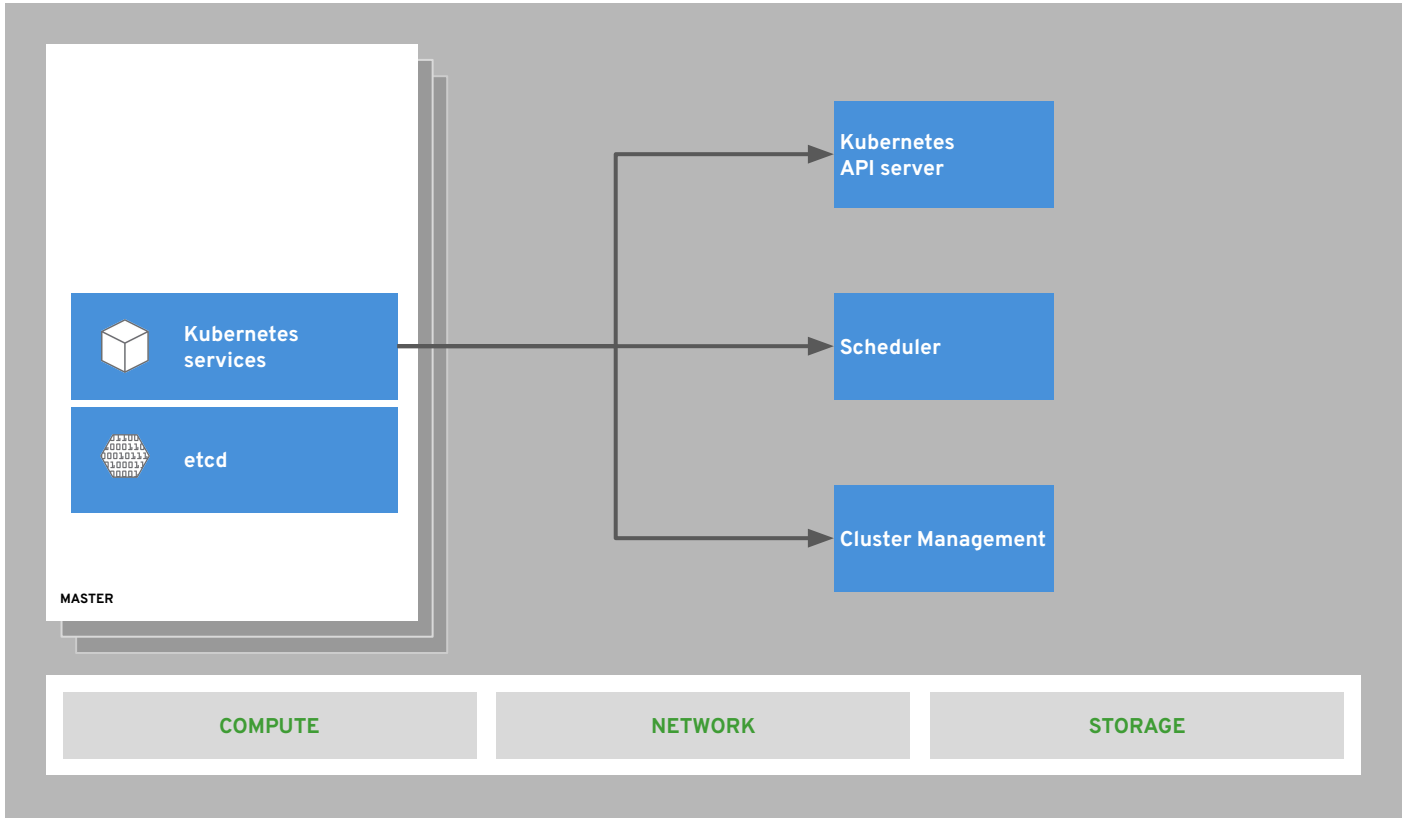
workers run workloads



masters are the control plane

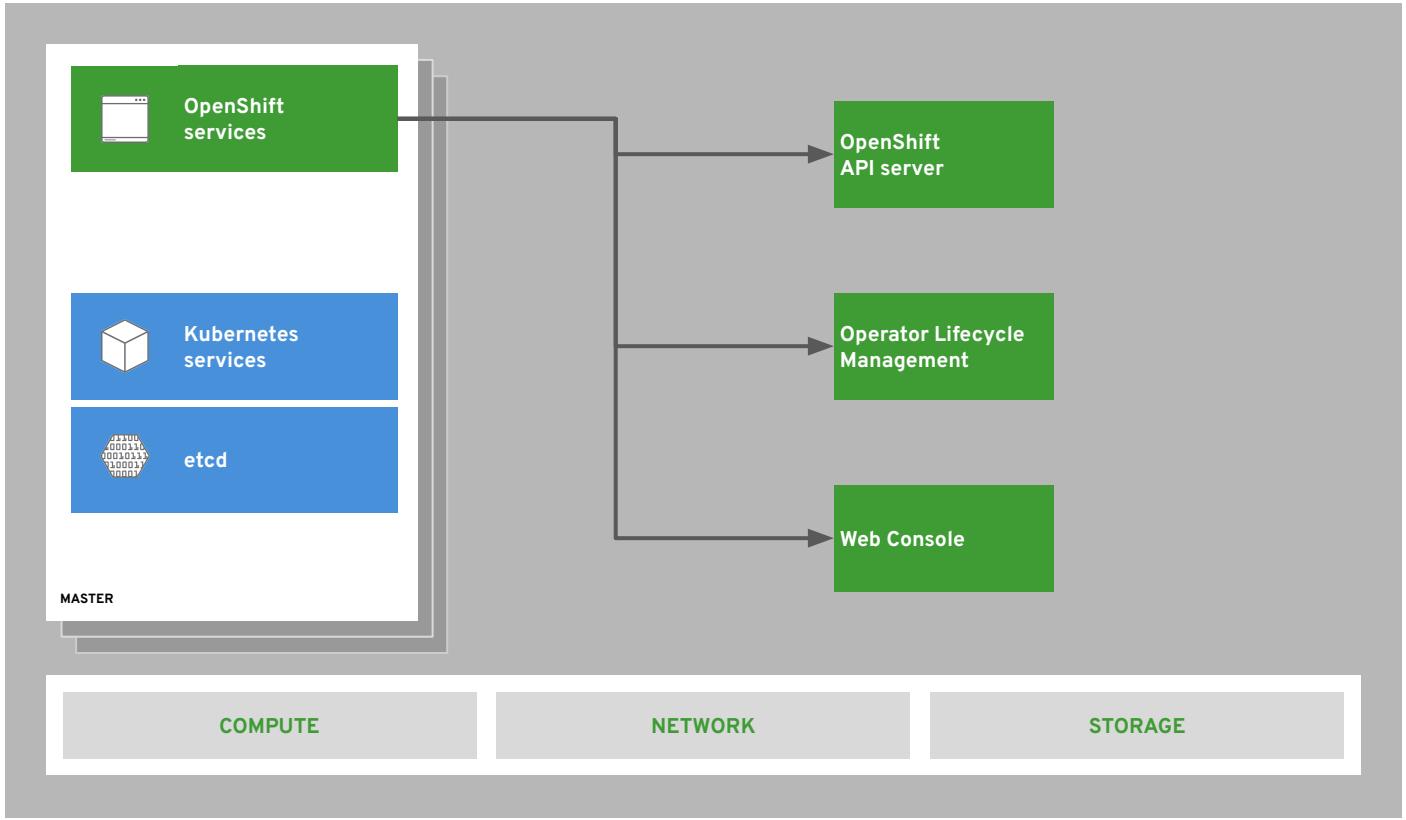
state of everything

core kubernetes components

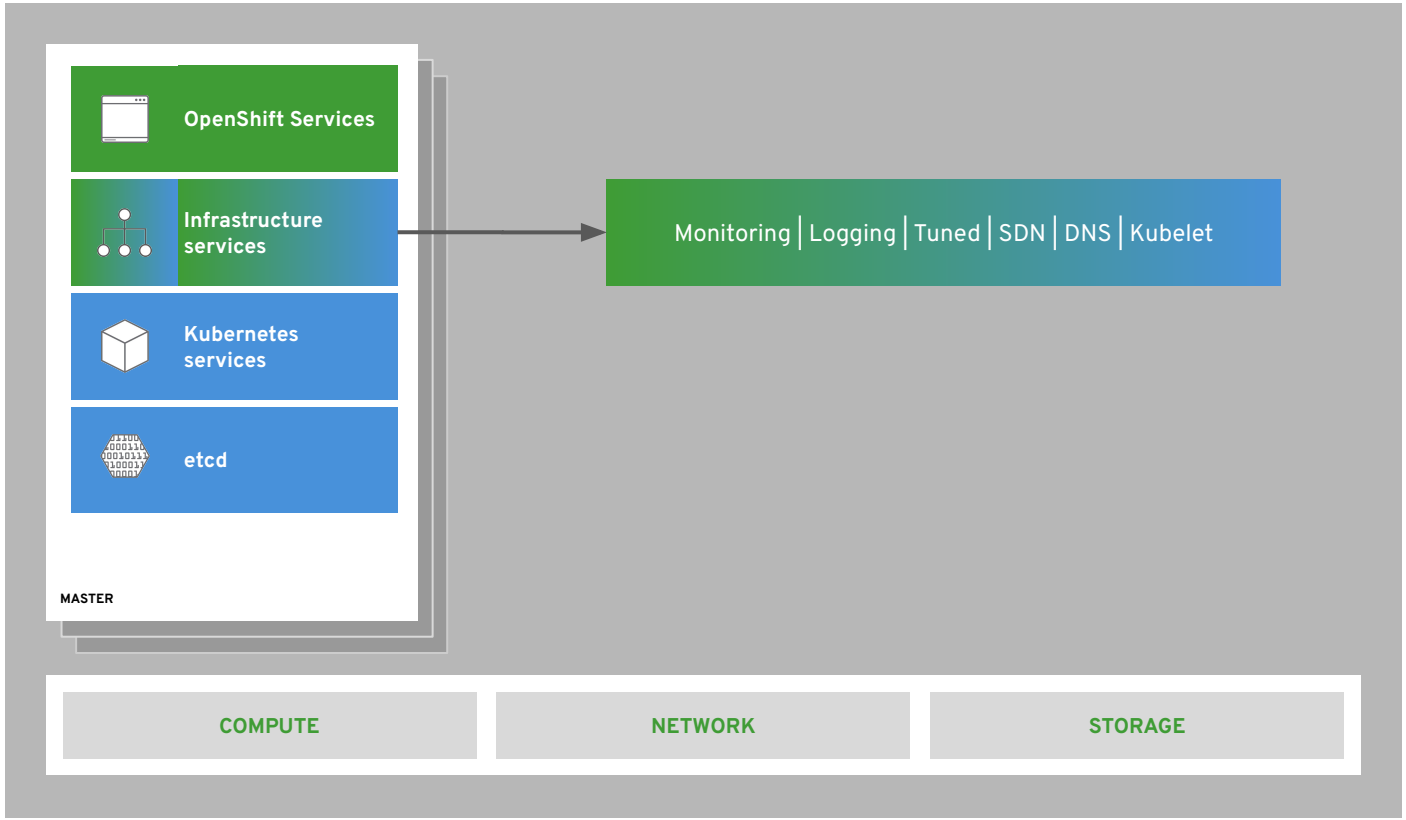


Red Hat

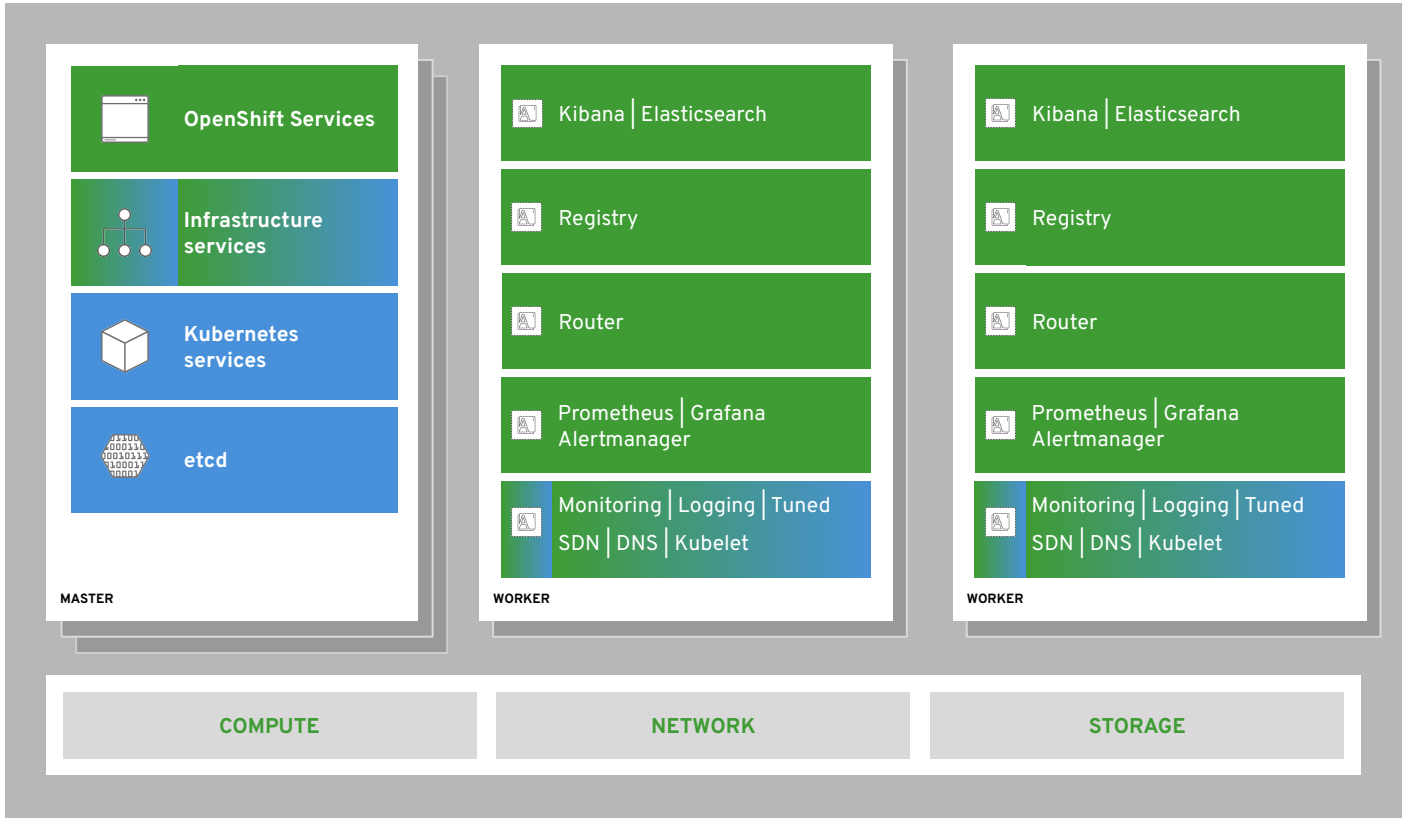
core OpenShift components



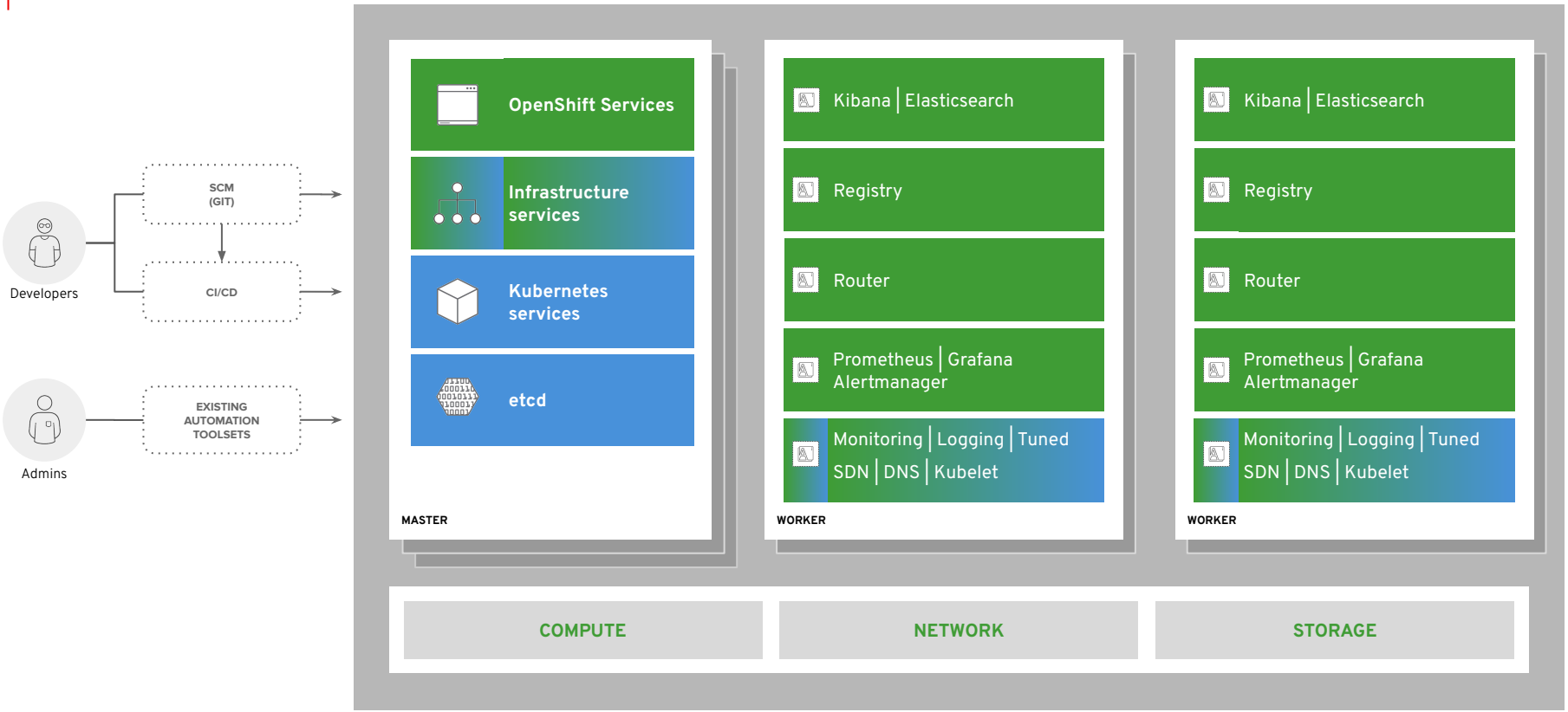
internal and support infrastructure services



integrated routing



dev and ops via web, cli, API, and IDE



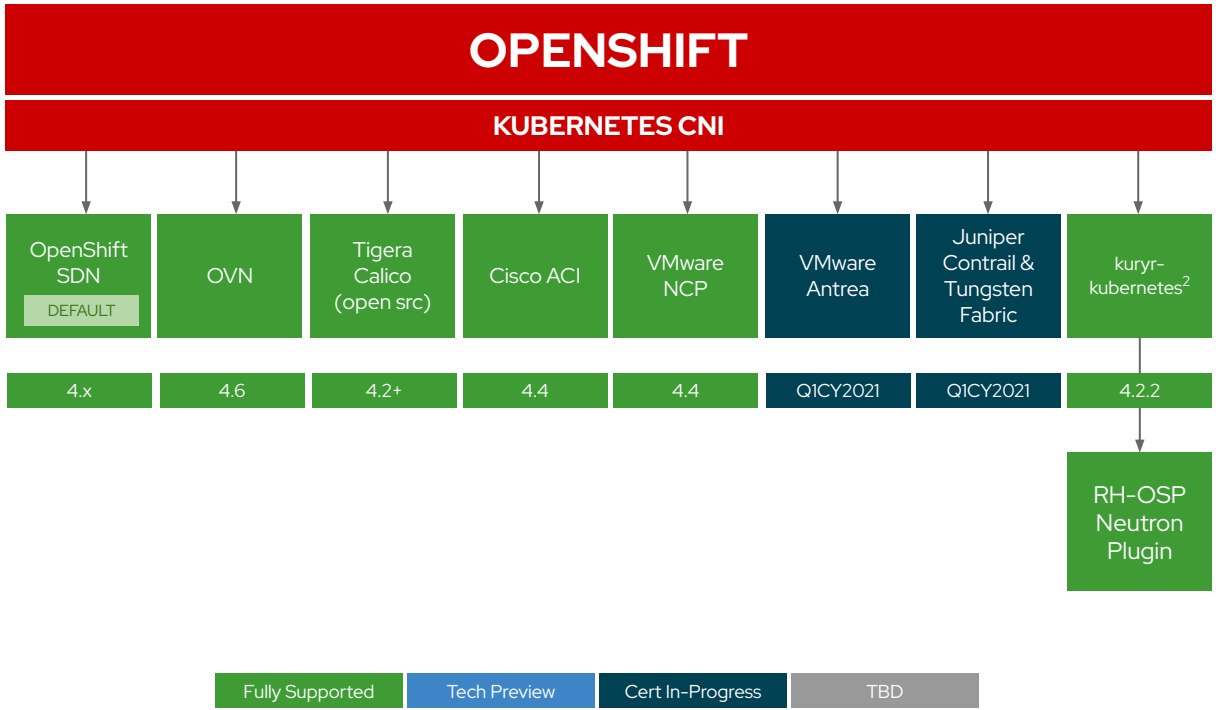
Networking

A pluggable model for network interface controls in kubernetes

OpenShift Networking Plug-ins

3rd-party Kubernetes CNI plug-in certification primarily consists of:

1. Formalizing the partnership
2. Certifying the container(s)
3. Certifying the Operator
4. Successfully passing the same Kubernetes networking conformance tests that OpenShift uses to validate its own SDN



Product Manager: Marc Curry

OpenShift SDN

An Open

vSwitch-based

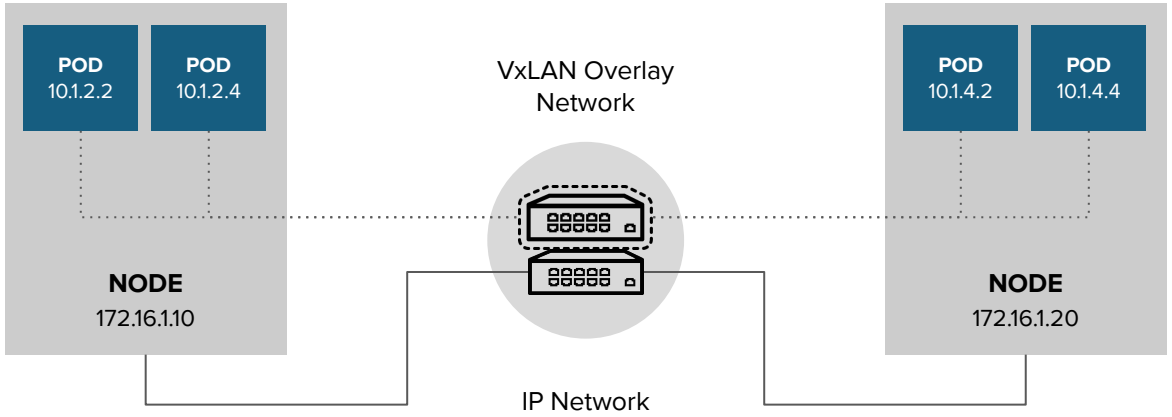
Software Defined

Network for

kubernetes

Red Hat

OpenShift SDN high-level architecture



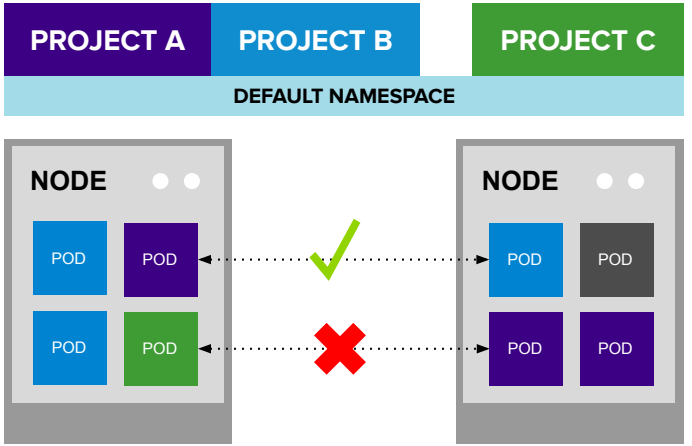
OpenShift SDN “flavors”

OPEN NETWORK **(Default)**

- All pods can communicate with each other across projects

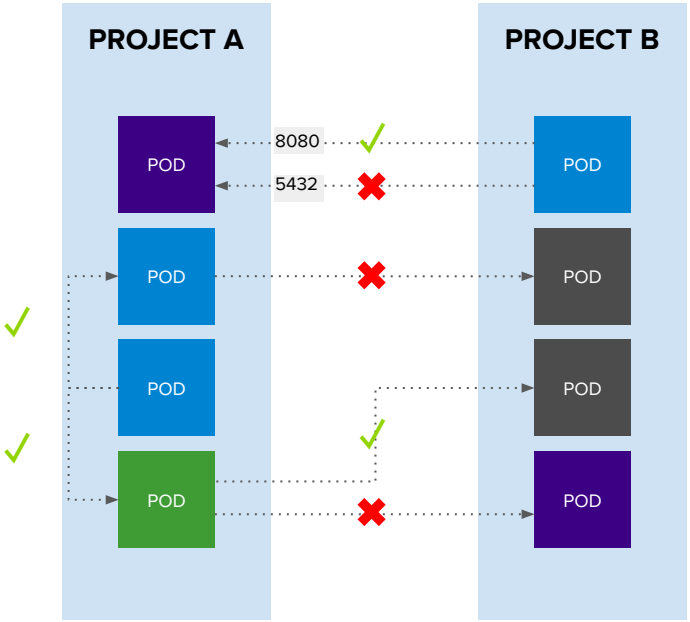
MULTI-TENANT NETWORK

- Project-level network isolation
- Multicast support
- Egress network policies



Multi-Tenant Network

NetworkPolicy



Example Policies

- Allow all traffic inside the project
- Allow traffic from green to gray
- Allow traffic to purple on 8080

```
apiVersion: extensions/v1beta1
kind: NetworkPolicy
metadata:
  name: allow-to-purple-on-8080
spec:
  podSelector:
    matchLabels:
      color: purple
  ingress:
  - ports:
    - protocol: tcp
      port: 8080
```

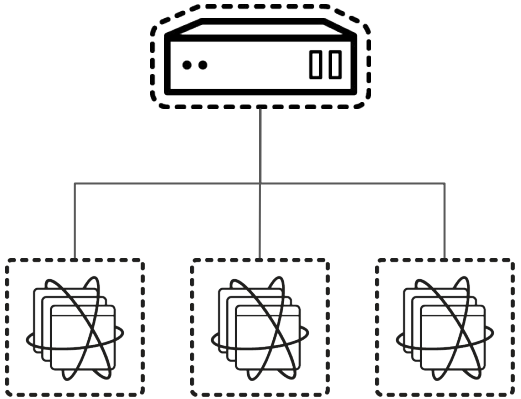
routes and ingress

How traffic enters the cluster

Red Hat

Routing and Load Balancing

- Pluggable routing architecture
  - HAProxy Router
  - F5 Router
- Multiple-routers with traffic sharding
- Router supported protocols
  - HTTP/HTTPS
  - WebSockets
  - TLS with SNI
- Non-standard ports via cloud load-balancers, external IP, and NodePort

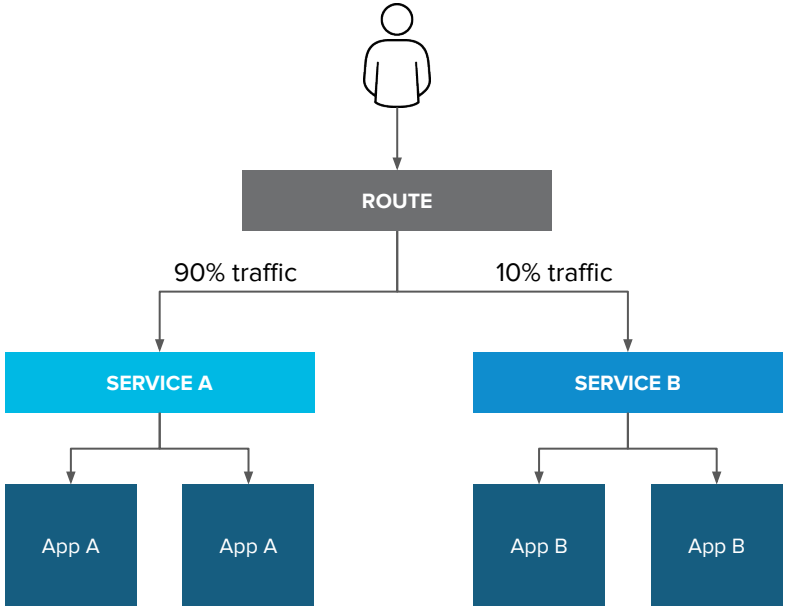


Routes vs Ingress

Feature	Ingress	Route
Standard Kubernetes object	X	
External access to services	X	X
Persistent (sticky) sessions	X	X
Load-balancing strategies (e.g. round robin)	X	X
Rate-limit and throttling	X	X
IP whitelisting	X	X
TLS edge termination	X	X
TLS re-encryption	X	X
TLS passthrough	X	X
Multiple weighted backends (split traffic)		X
Generated pattern-based hostnames		X
Wildcard domains		X

Router-based deployment methodologies

Split Traffic Between Multiple Services For A/B Testing, Blue/Green and Canary Deployments

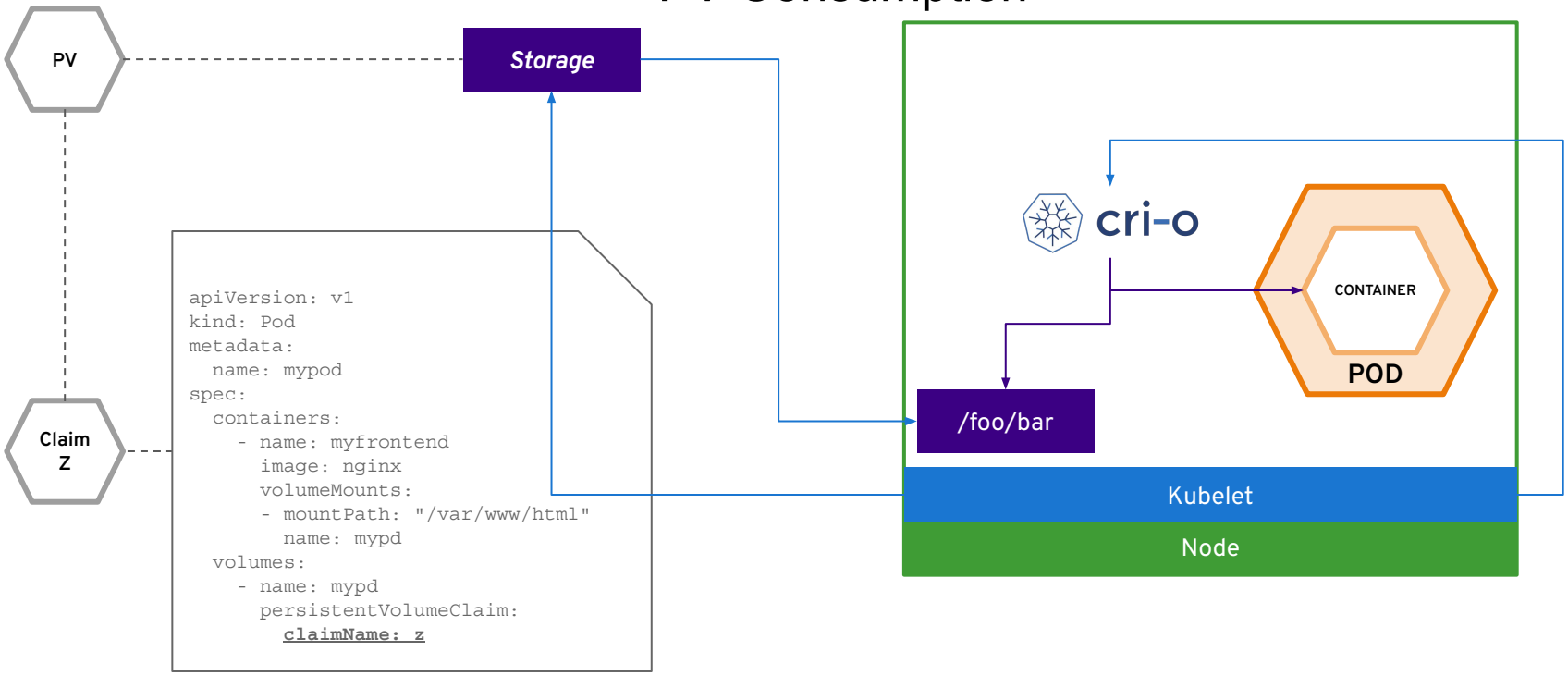


Red Hat

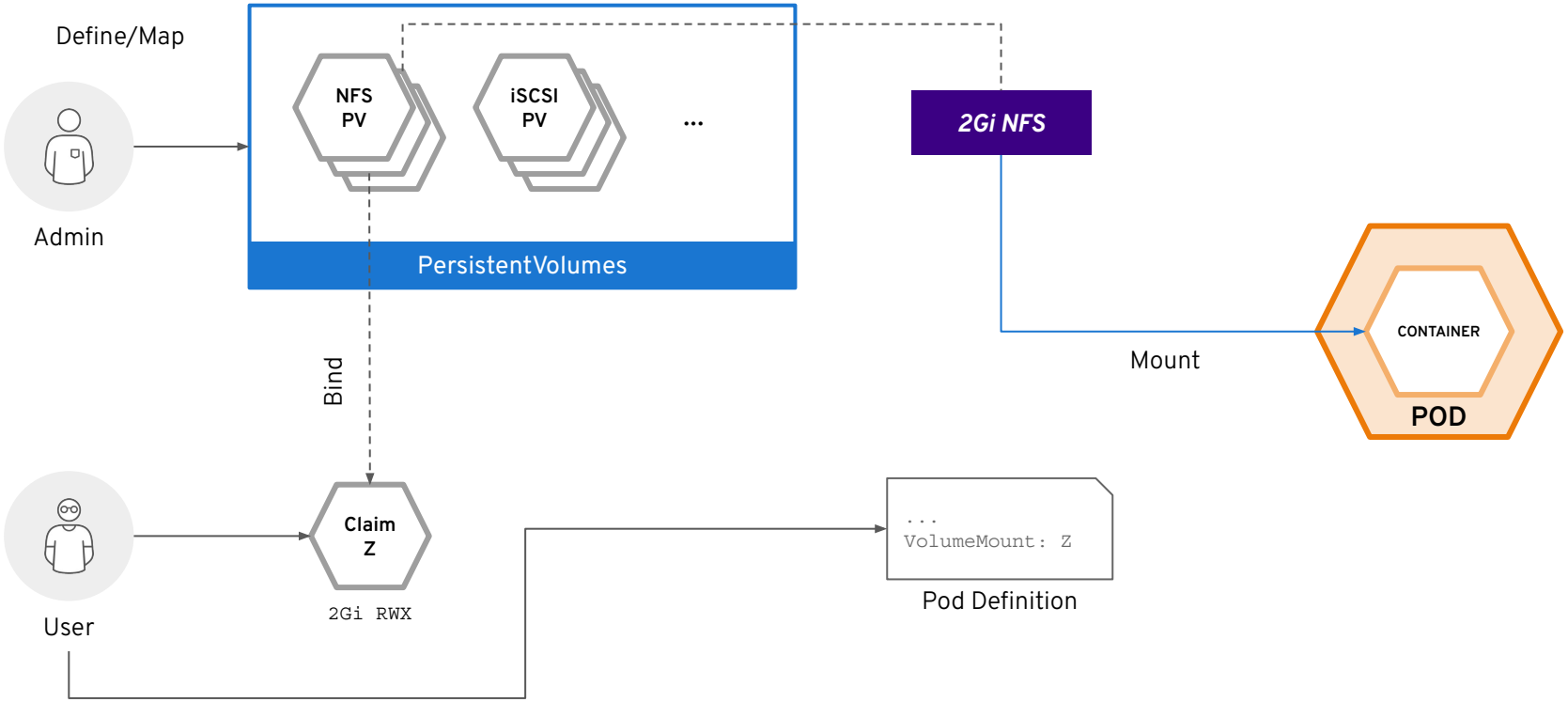
Persistent Storage

Connecting real-world storage to your containers to enable stateful applications

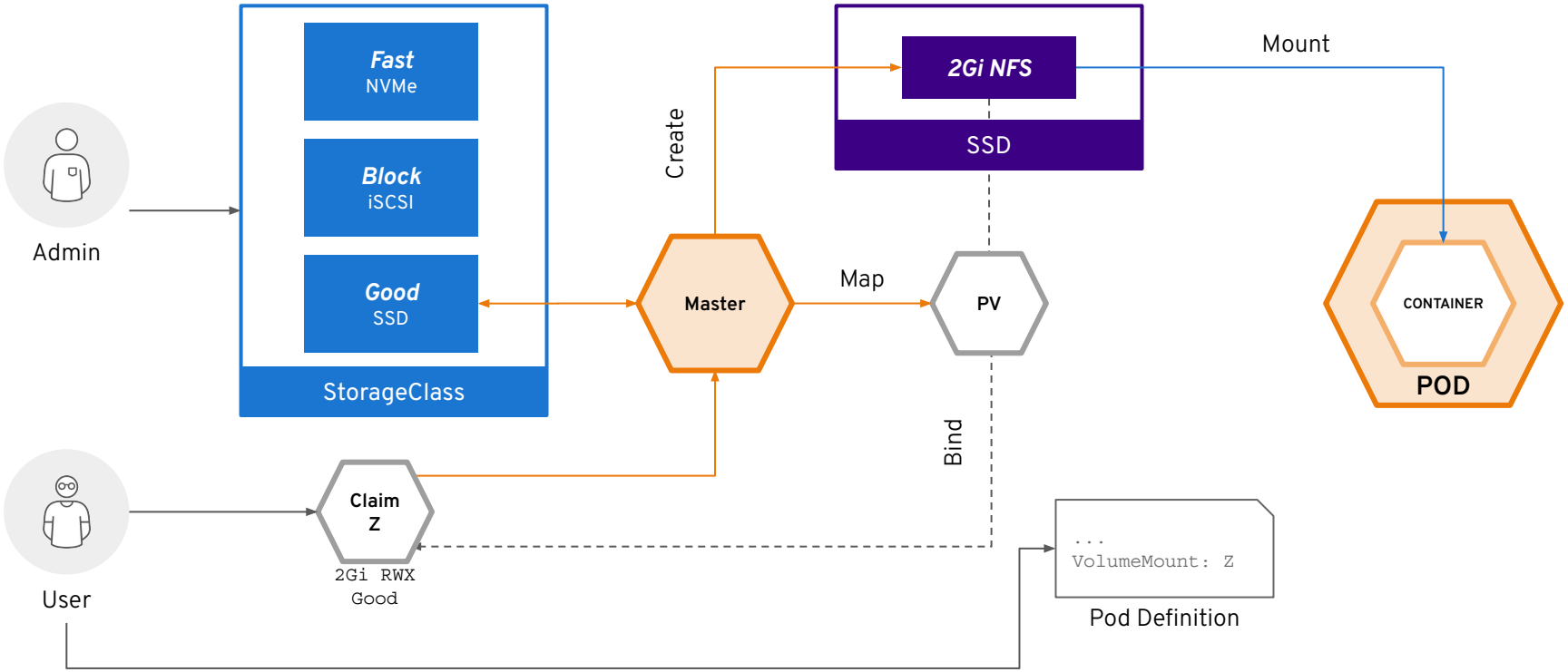
PV Consumption



Static Storage Provisioning



Dynamic Storage Provisioning



linkedin.com/company/red-hat

youtube.com/user/RedHatVideos

facebook.com/redhatinc

twitter.com/RedHat

Red Hat