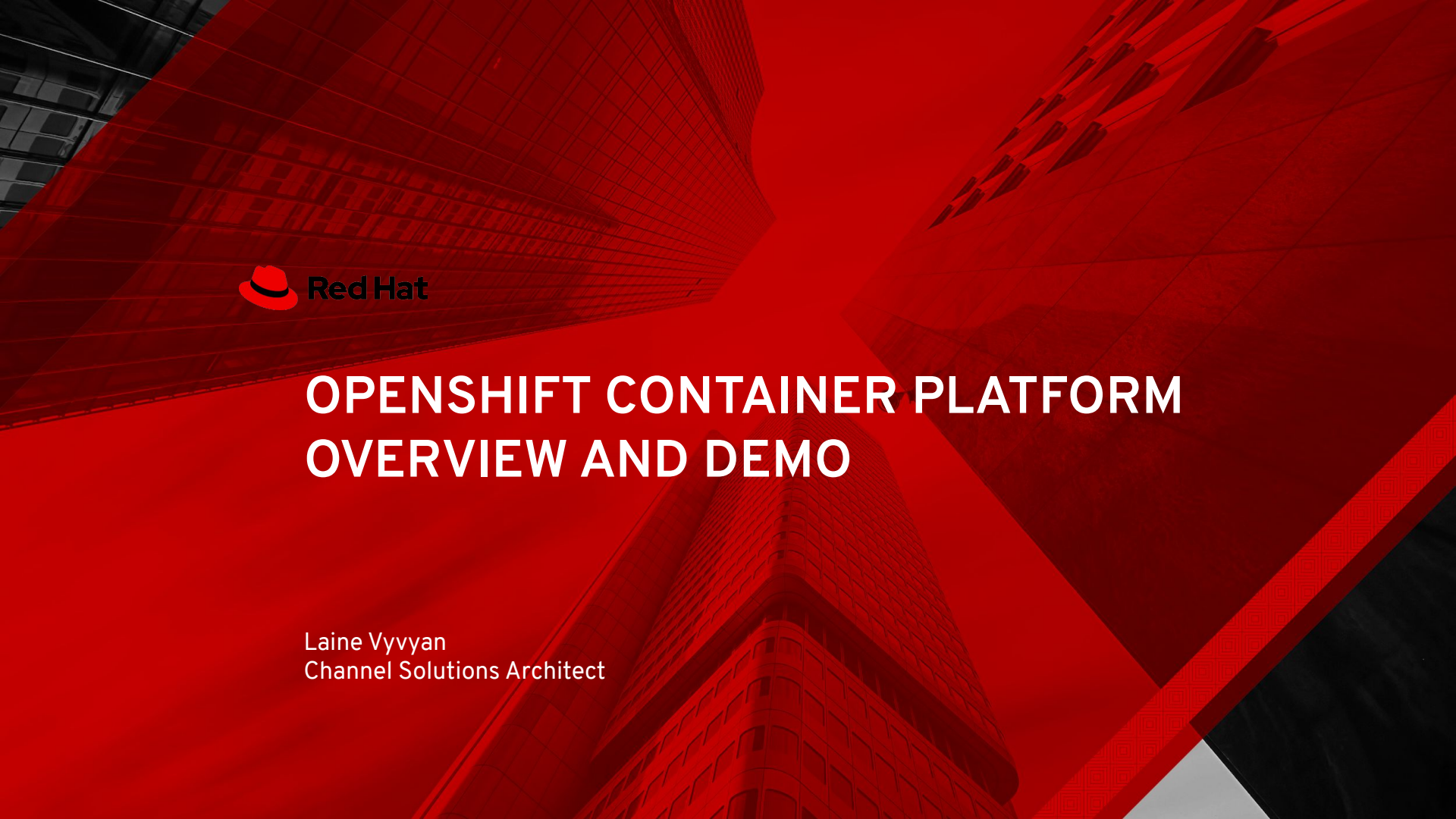


OPENSHIFT CONTAINER PLATFORM OVERVIEW AND DEMO

Laine Vyvyan
Channel Solutions Architect

Hi! I’m Laine Vyvyan.

I’m a Channel Solutions Architect, covering the Midwest.

Me:

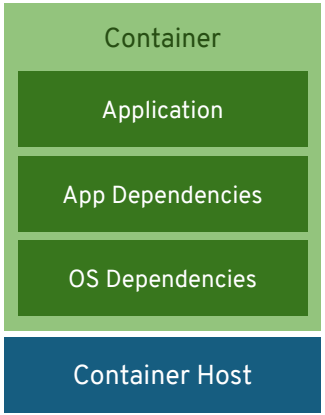
lvyvyan@redhat.com

@lainie_ftw

Red Hat

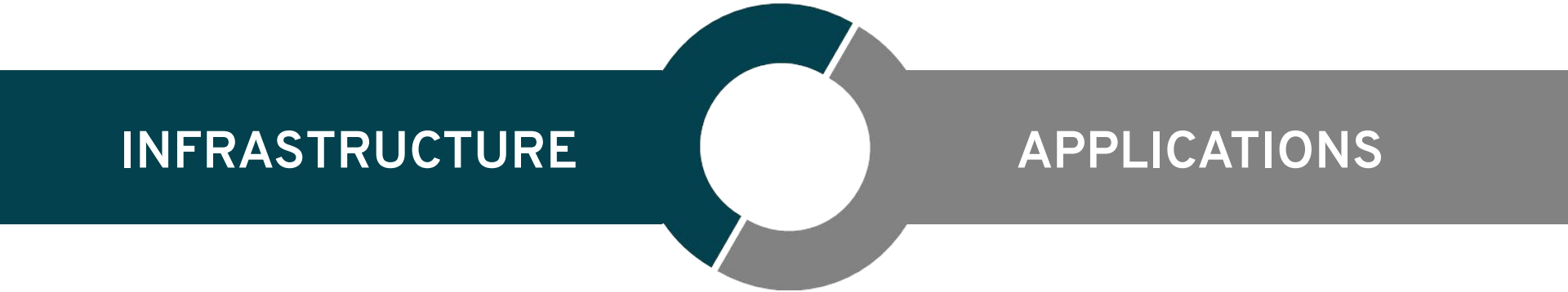
What is a container?

A container is an application, the application's dependencies/libraries/other binaries, and the configuration files that the application needs to run, all bundled into **one portable unit**.



Red Hat

Why Containers?



- Application processes on a shared kernel
- Simpler, lighter, and denser than VMs
- Portable across different environments
- **Dynamic scalability on demand**

- Package apps with all dependencies
- Deploy to any environment in seconds
- Cloud-native application development
- Flexibility with language & runtime

What is Kubernetes?

Kubernetes is open source container orchestration - it automates **deployment**, **scaling**, and **management** of containers.

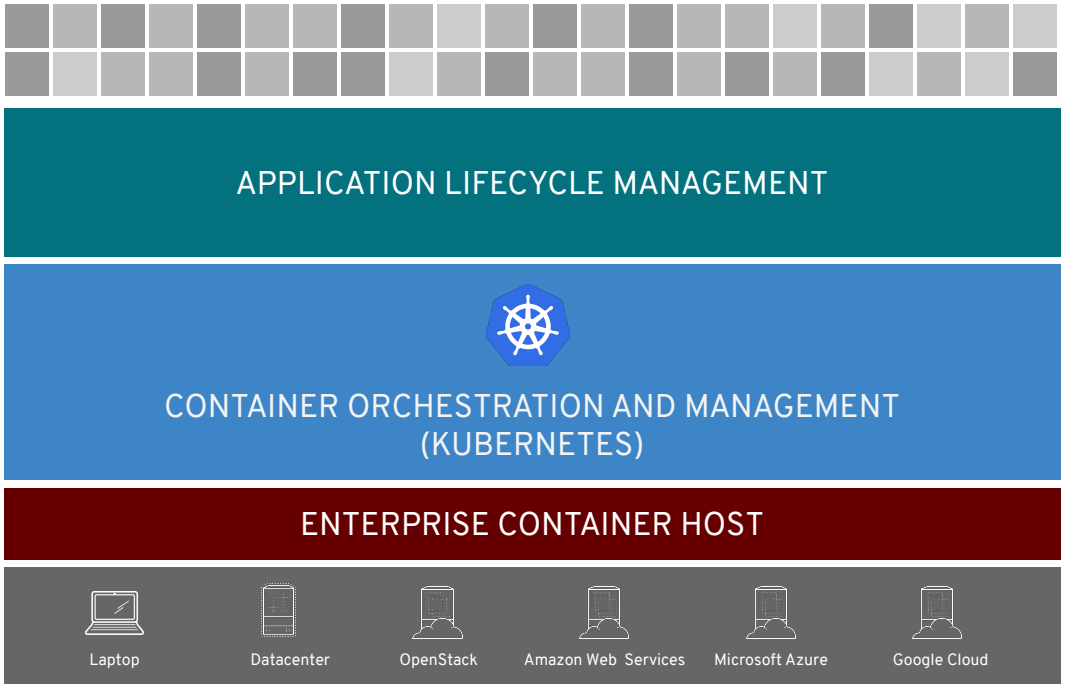
Red Hat

OpenShift 

Red Hat OpenShift is a Kubernetes-based, enterprise-ready container application platform.

Red Hat

OpenShift



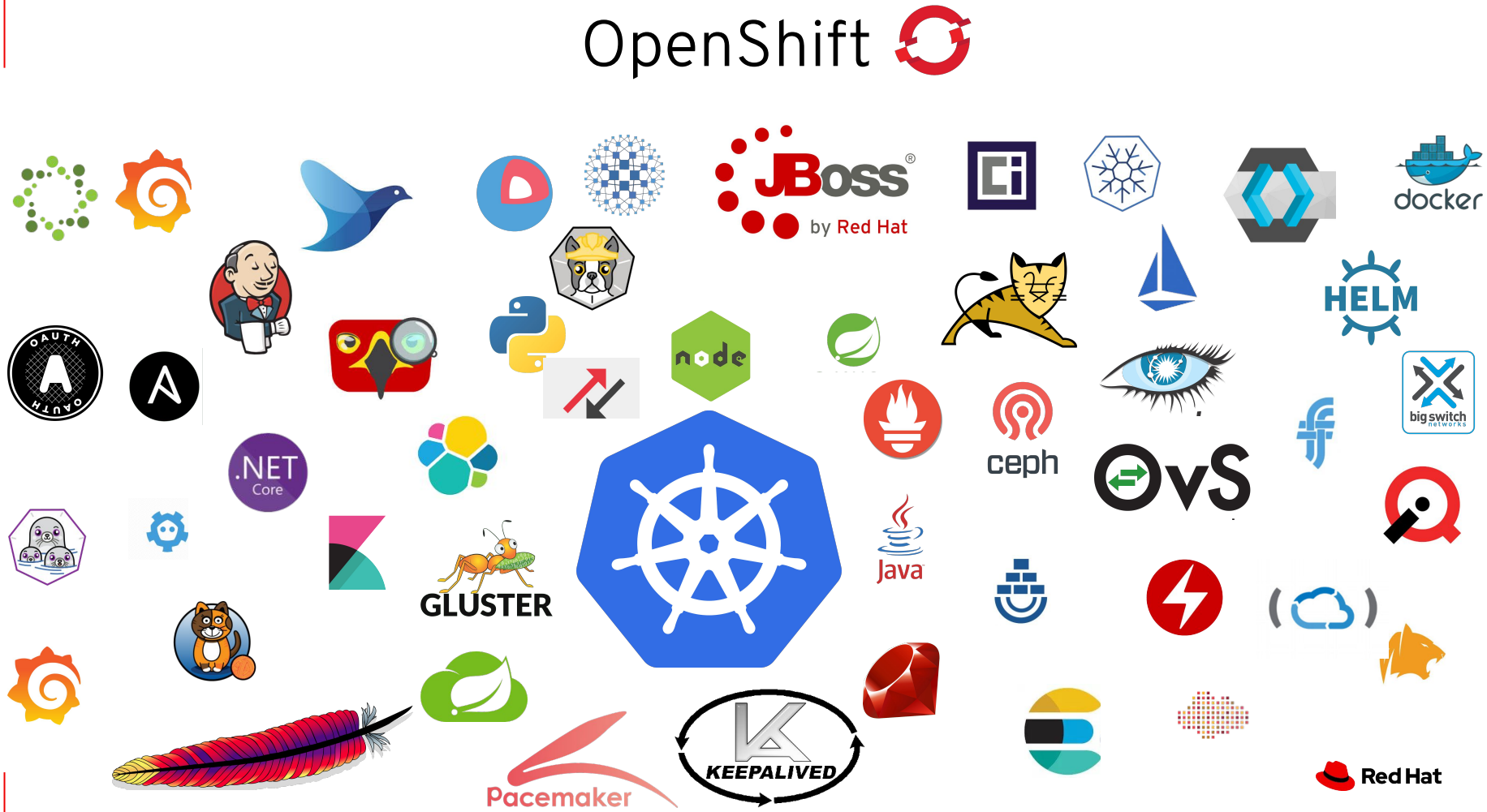
ANY
CONTAINER

RED HAT®
OPENSHIFT

ANY
INFRASTRUCTURE

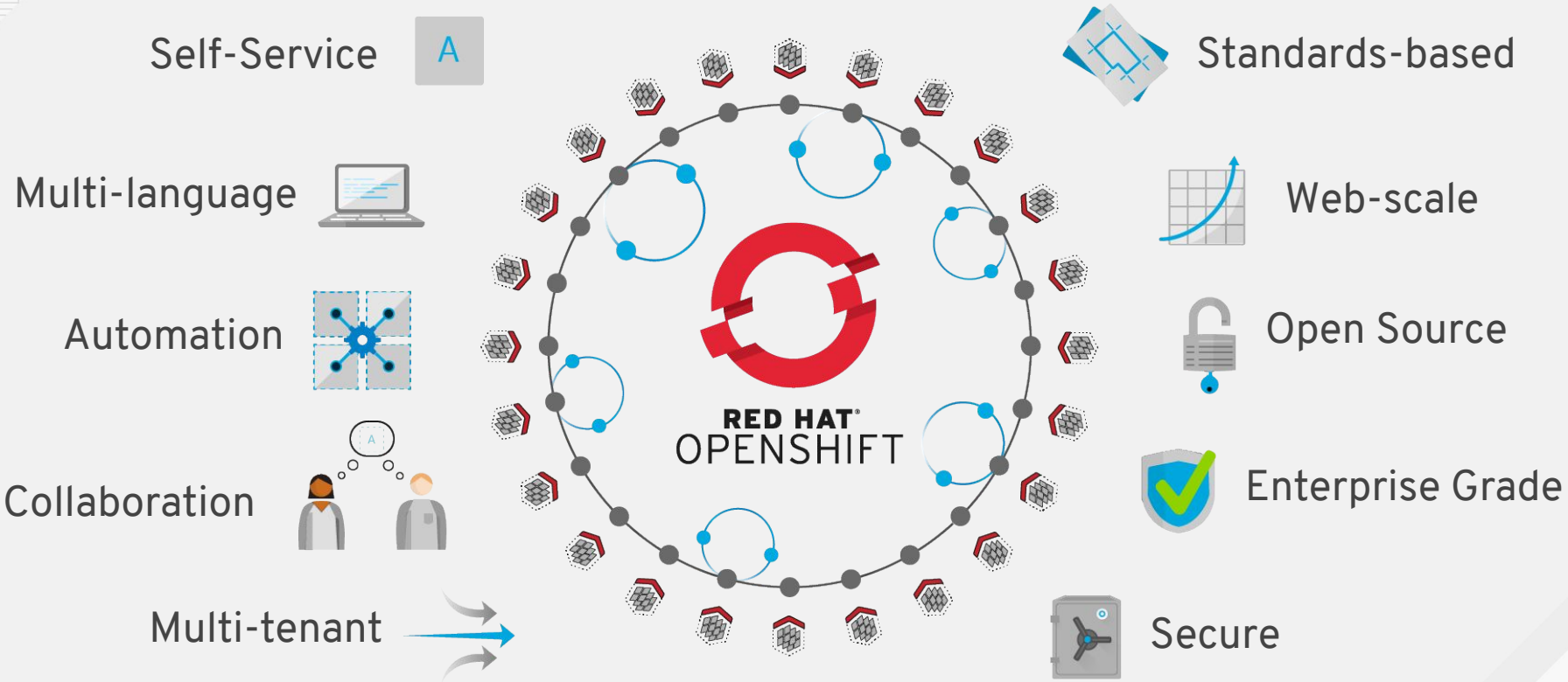
Red Hat

OpenShift

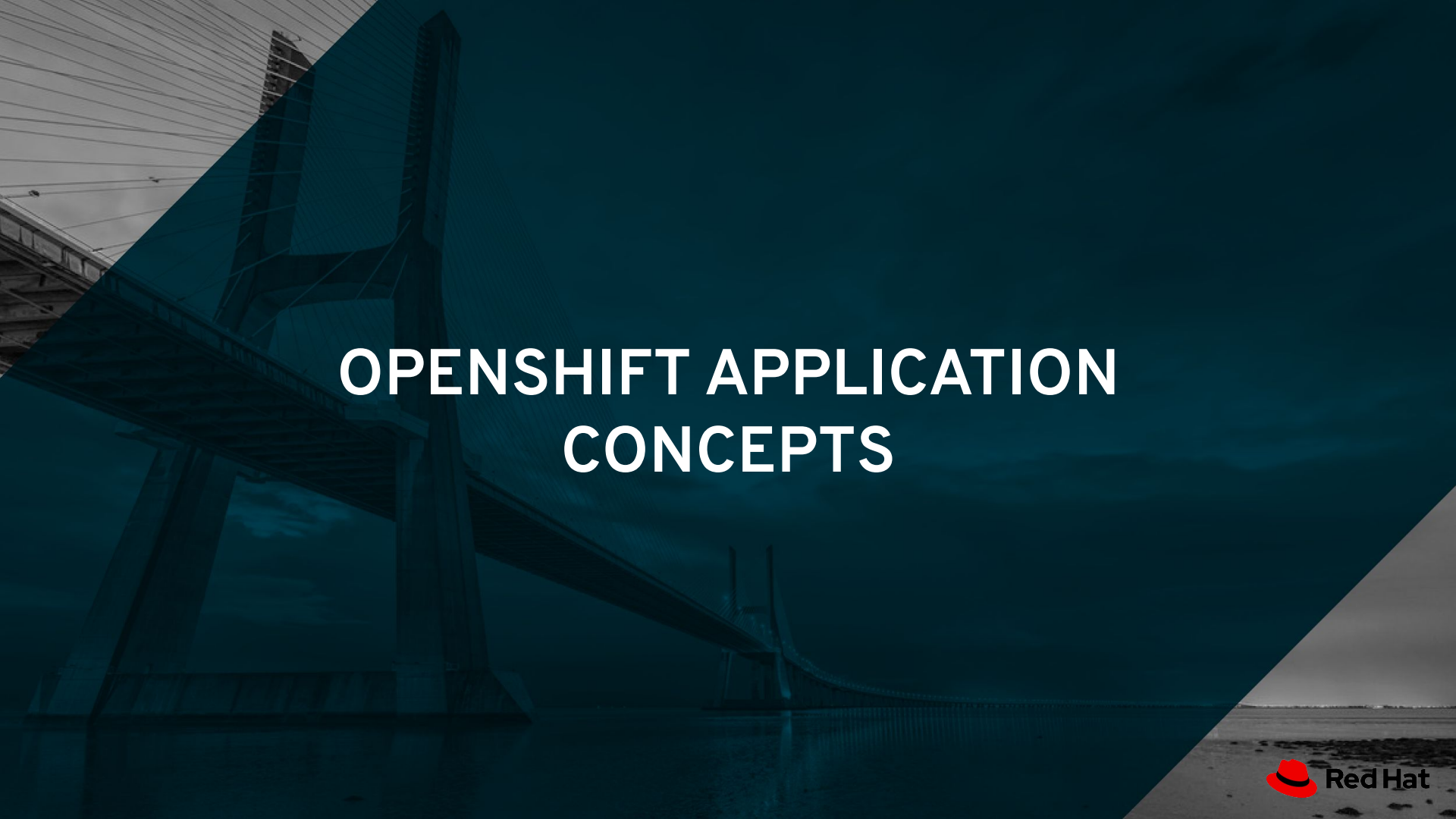

JBoss
by Red Hat
docker
HELM
node
.NET
Core
ceph
OvS
big switch
networks
GLUSTER
Java
KEEPALIVED
Pacemaker
Red Hat

OpenShift

1. 200+ validated integrations
2. 100s of defect and performance fixes
3. 9 year enterprise lifecycle management
4. Security fixes
5. Middleware integration (container images, storage, networking, cloud services, etc)
6. Certified Kubernetes


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

Red Hat


OPENSHIFT APPLICATION
CONCEPTS
Red Hat

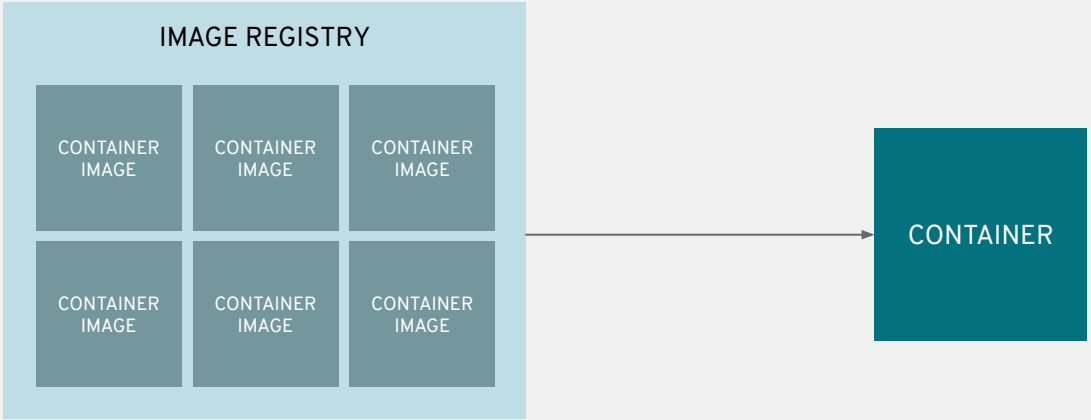
A **container** is the smallest compute unit

Red Hat

containers are created from container **images**

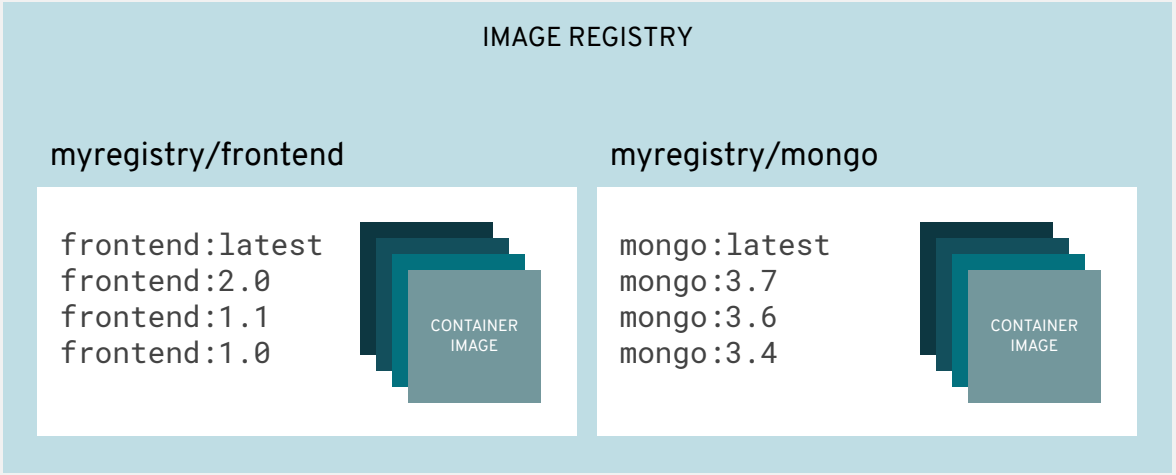
Red Hat

container images are stored in an **image registry**



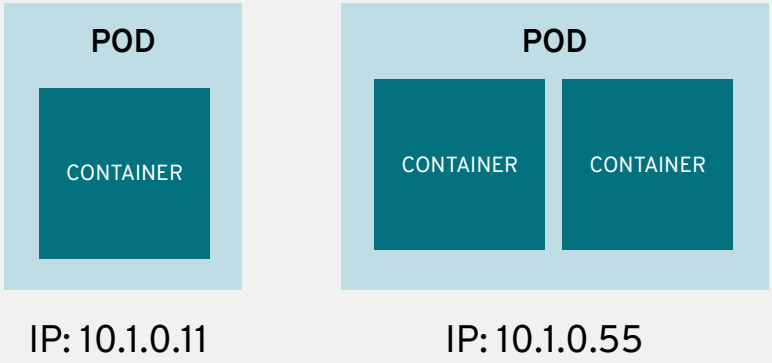
Red Hat

the image registry contains all versions (within default policy) of an image



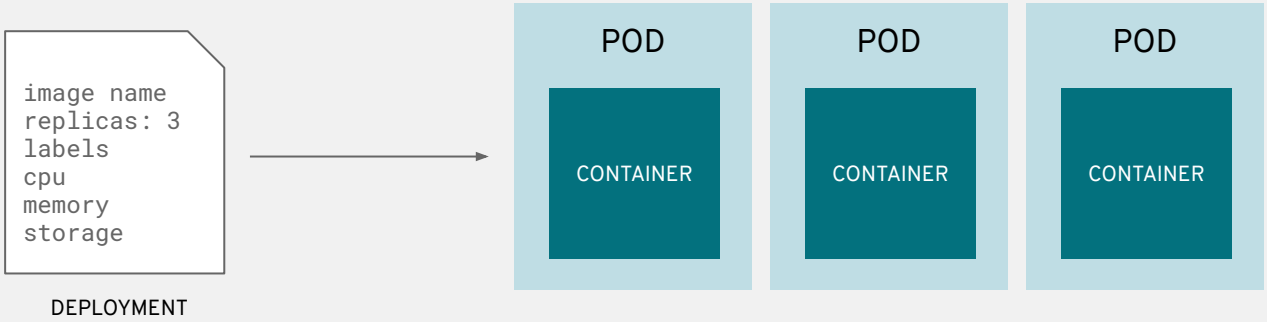
Red Hat

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



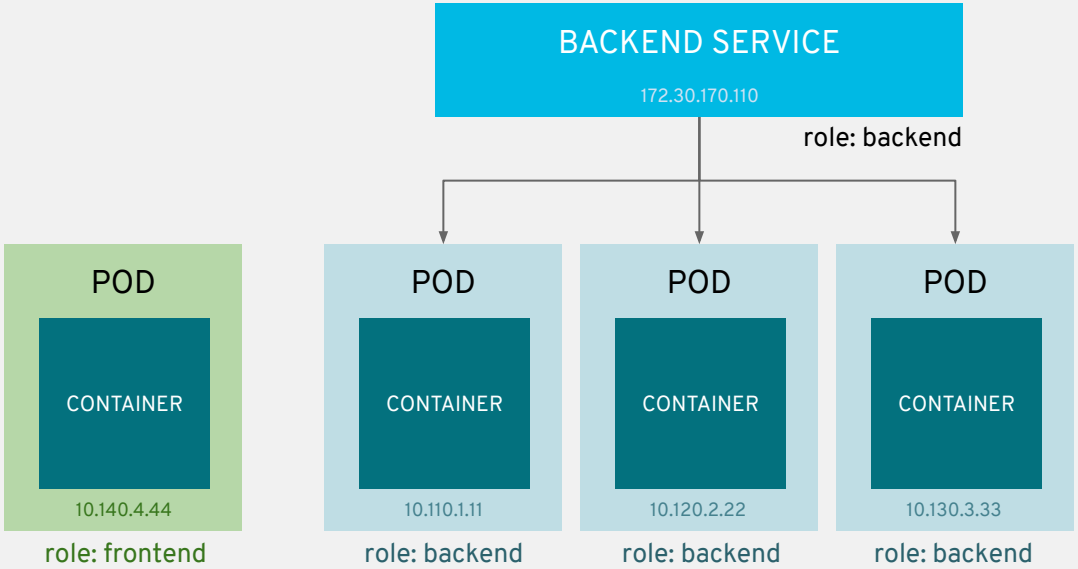
Red Hat

pod configuration is defined in a **deployment config (DC)**



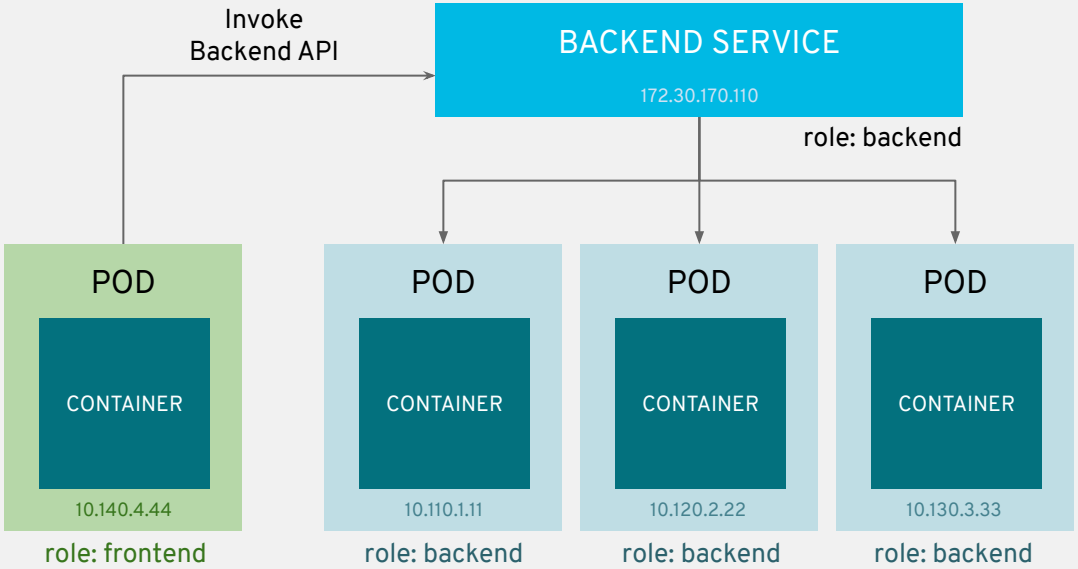
Red Hat

services provide internal load-balancing and service discovery across pods



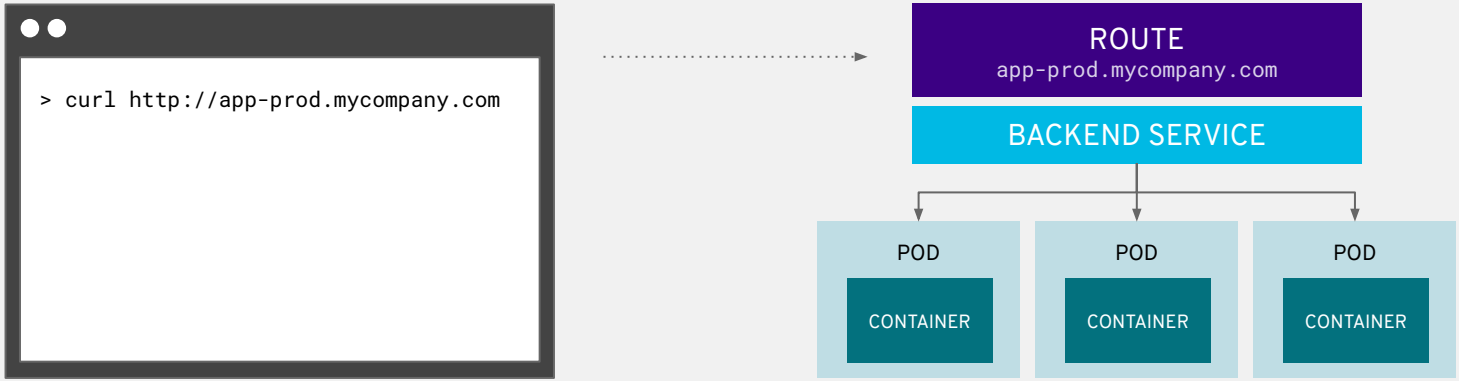
Red Hat

apps can talk to *each other* via services



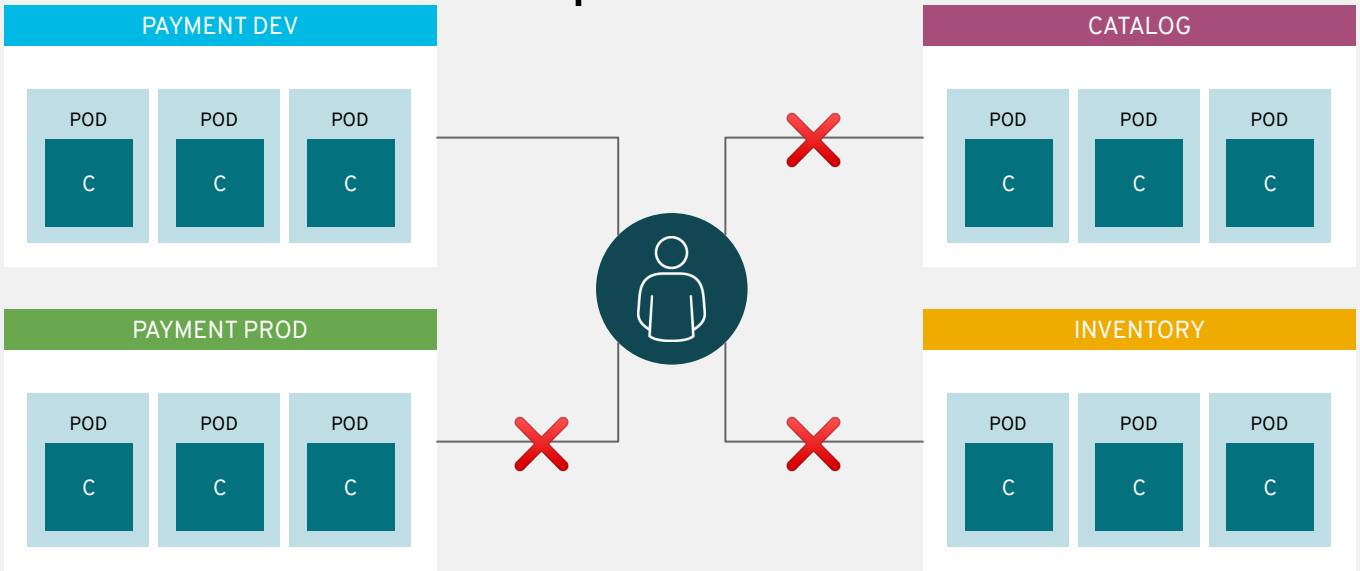
Red Hat

routes open an external door to services, adding them to the external load balancer and providing readable URLs for the app



Red Hat

projects provide logical buckets (and isolation!) in which to contain apps - across environments, teams, groups and departments



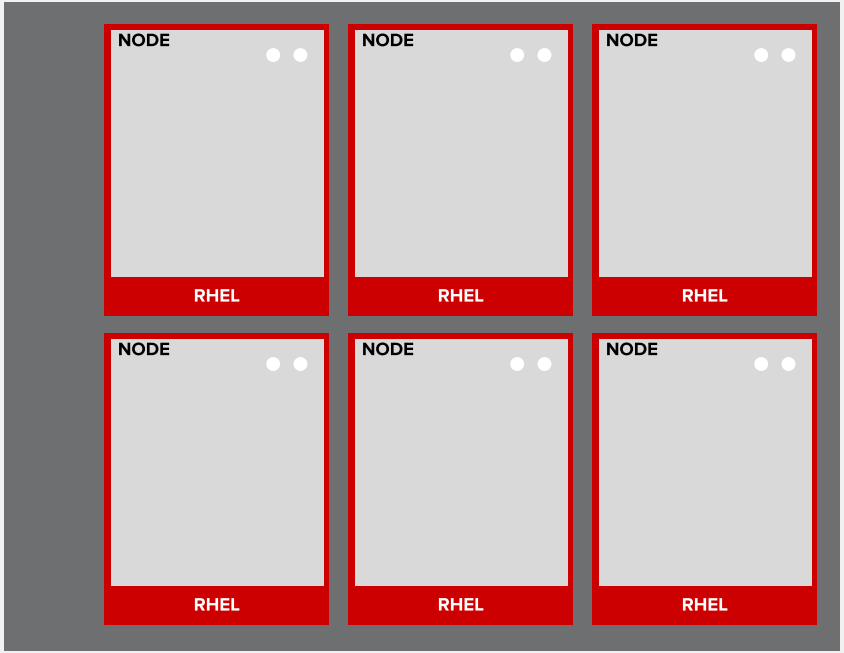
Red Hat


OPENSHIFT CLUSTER ARCHITECTURE
Red Hat

YOUR CHOICE OF INFRASTRUCTURE

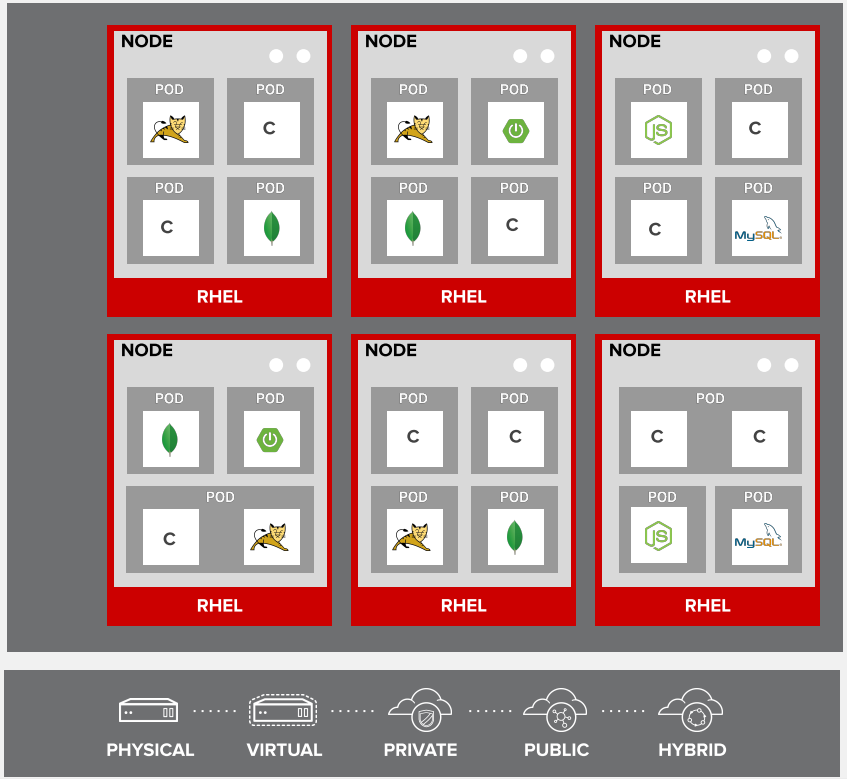
Red Hat

NODES:
RHEL* INSTANCES WHERE...



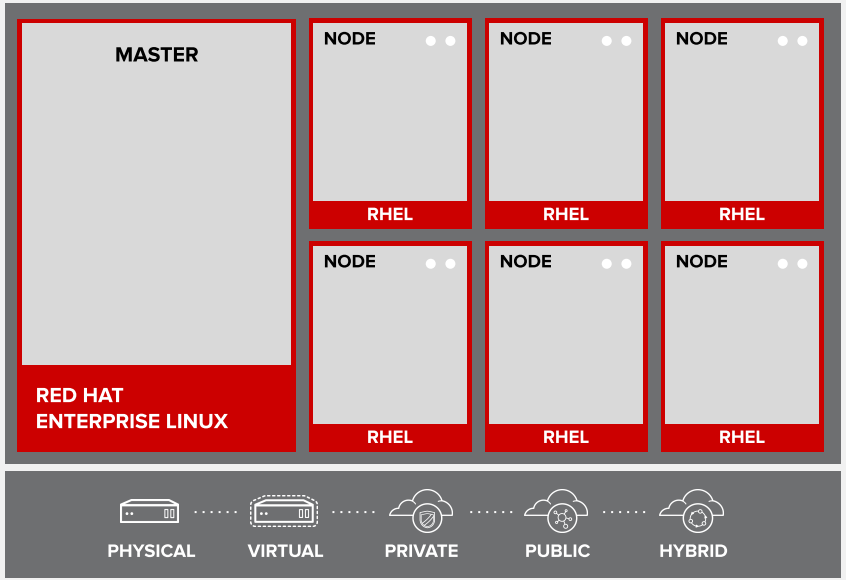
Red Hat

NODES:
RHEL* INSTANCES WHERE PODS RUN



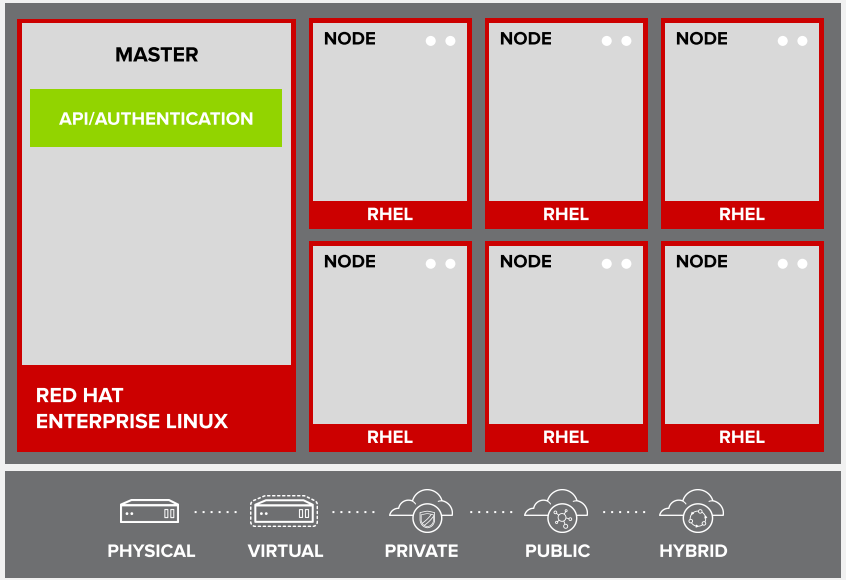
Red Hat

MASTERS ARE THE CONTROL PLANE. THEY HANDLE THINGS LIKE…



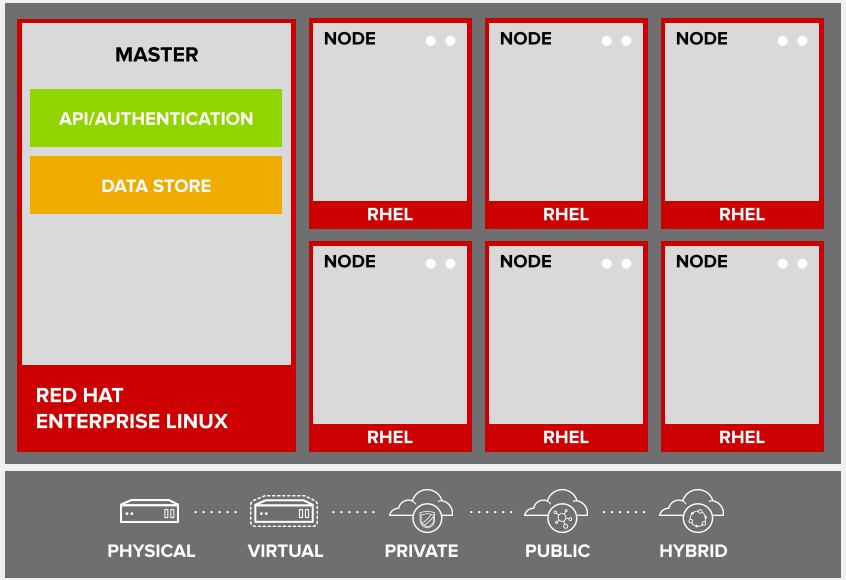
Red Hat

API AND AUTHENTICATION



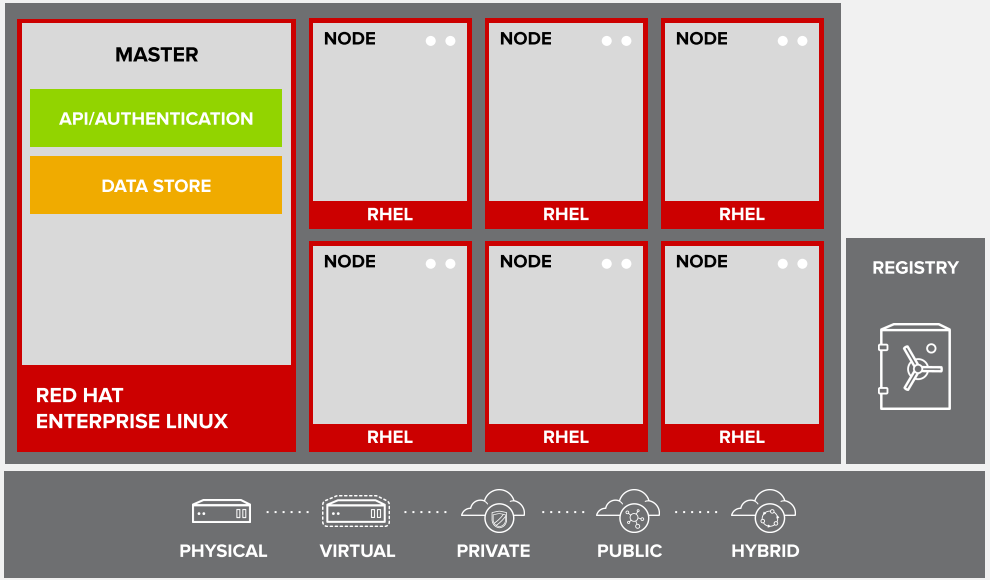
Red Hat

DESIRED AND CURRENT STATE WITHIN THE CLUSTER



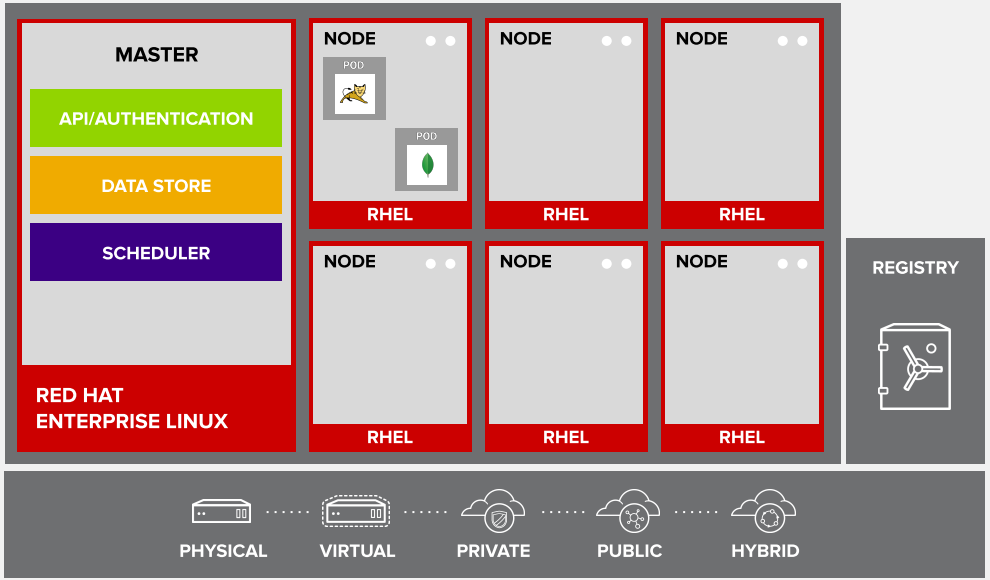
Red Hat

INTEGRATION WITH THE REGISTRY



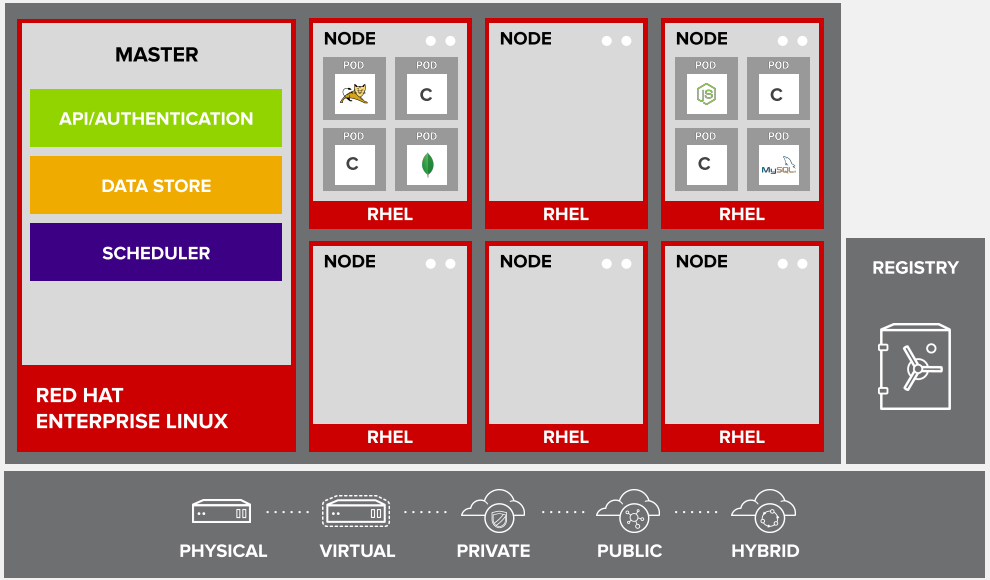
Red Hat

ORCHESTRATION AND SCHEDULING



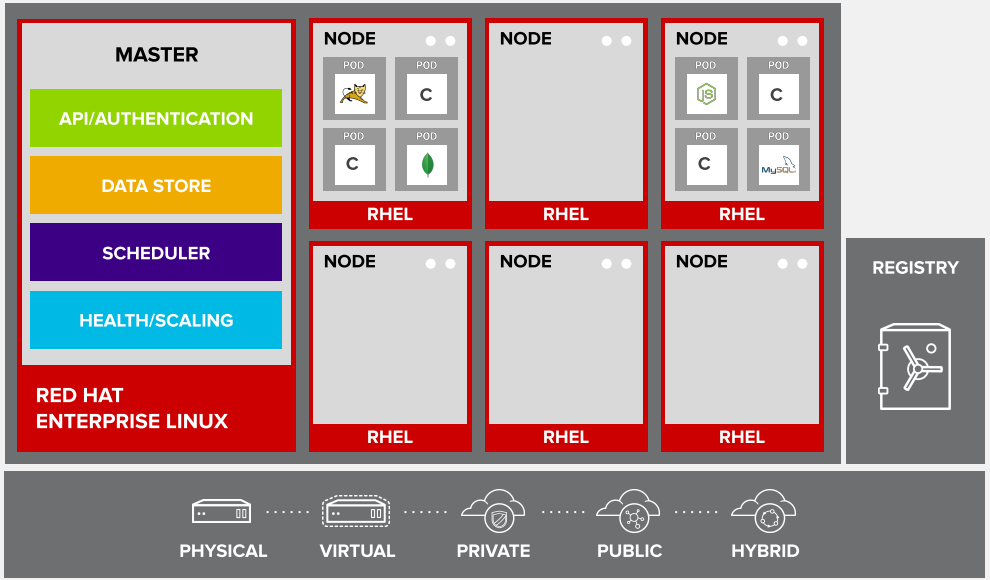
Red Hat

PLACEMENT BY POLICY



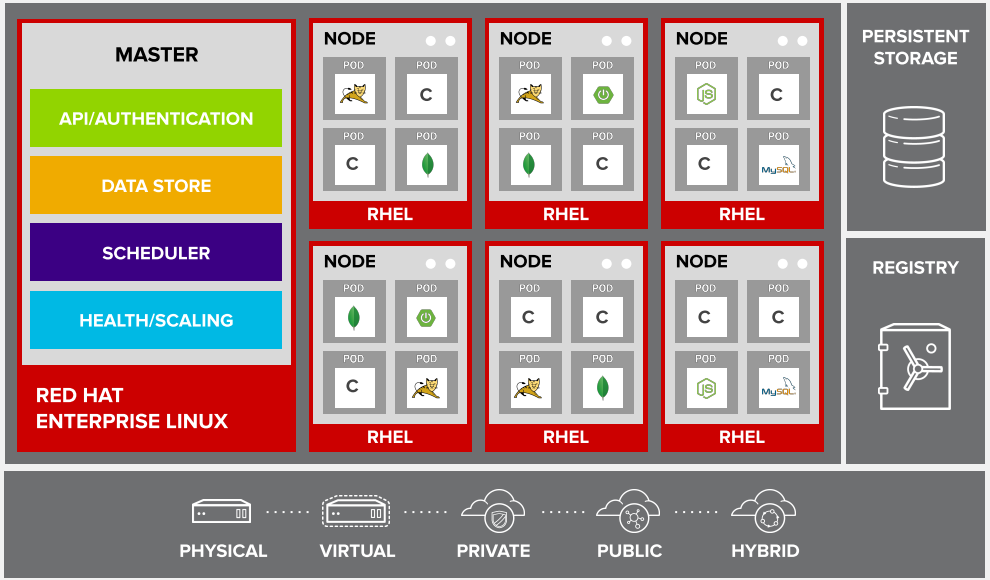
Red Hat

AUTOSCALING PODS



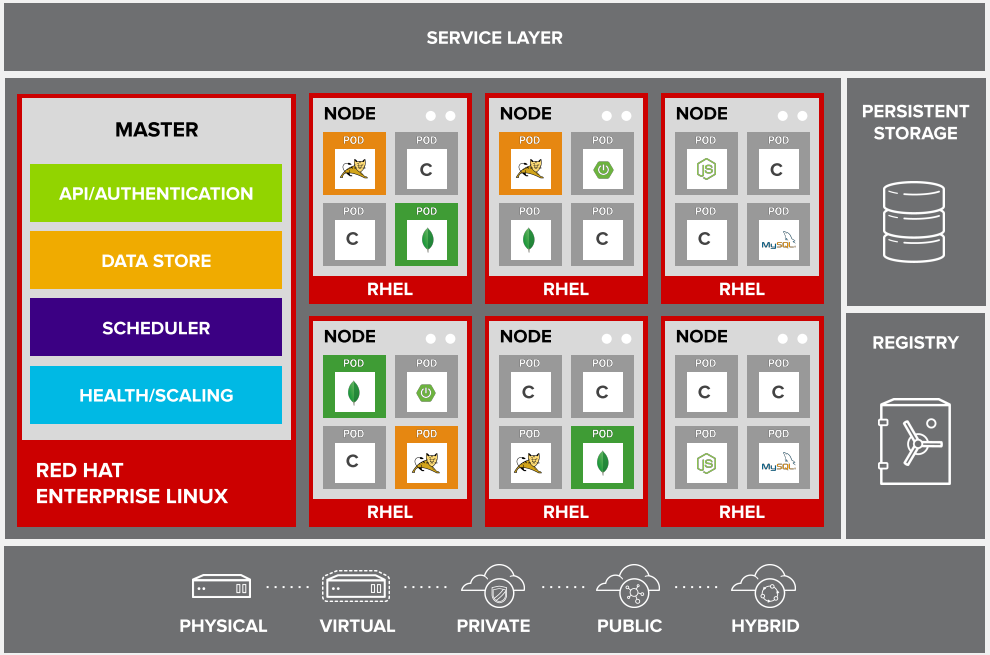
Red Hat

PERSISTENT DATA IN CONTAINERS



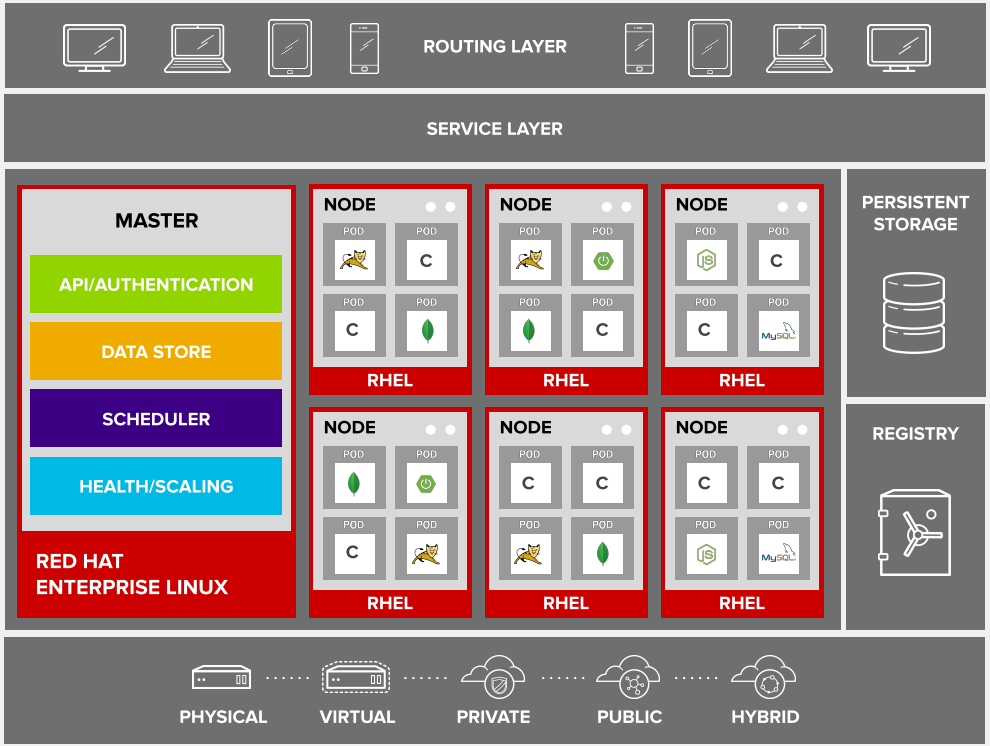
Red Hat

ON TOP OF EVERYTHING IS THE SERVICE LAYER



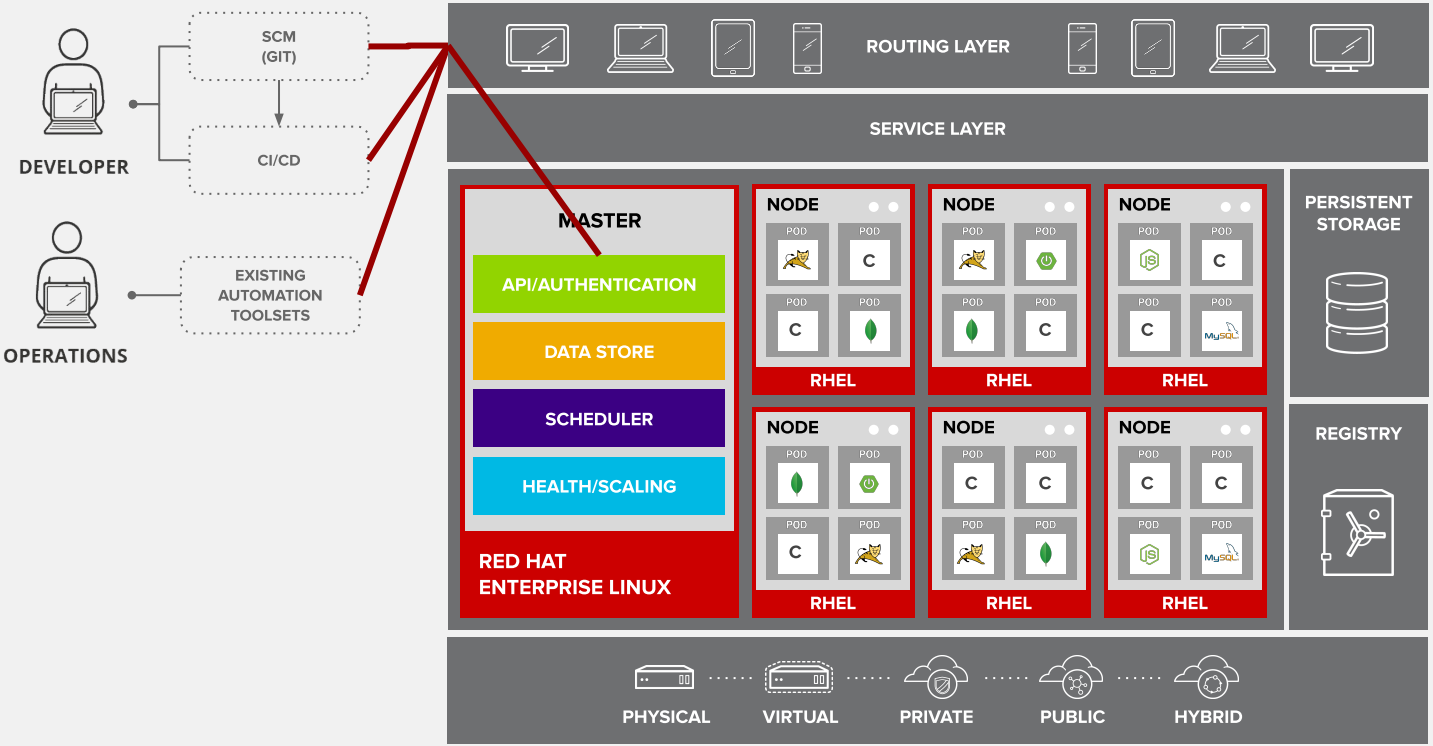
Red Hat

AND ROUTING AND LOAD-BALANCING



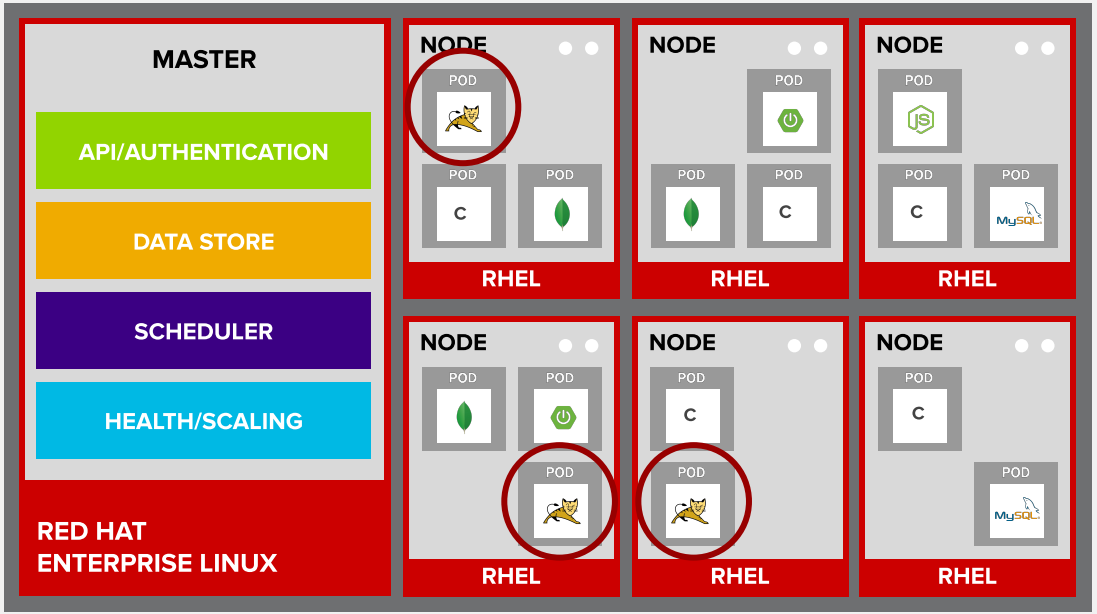
Red Hat

ACCESS VIA WEB, CLI, IDE AND API



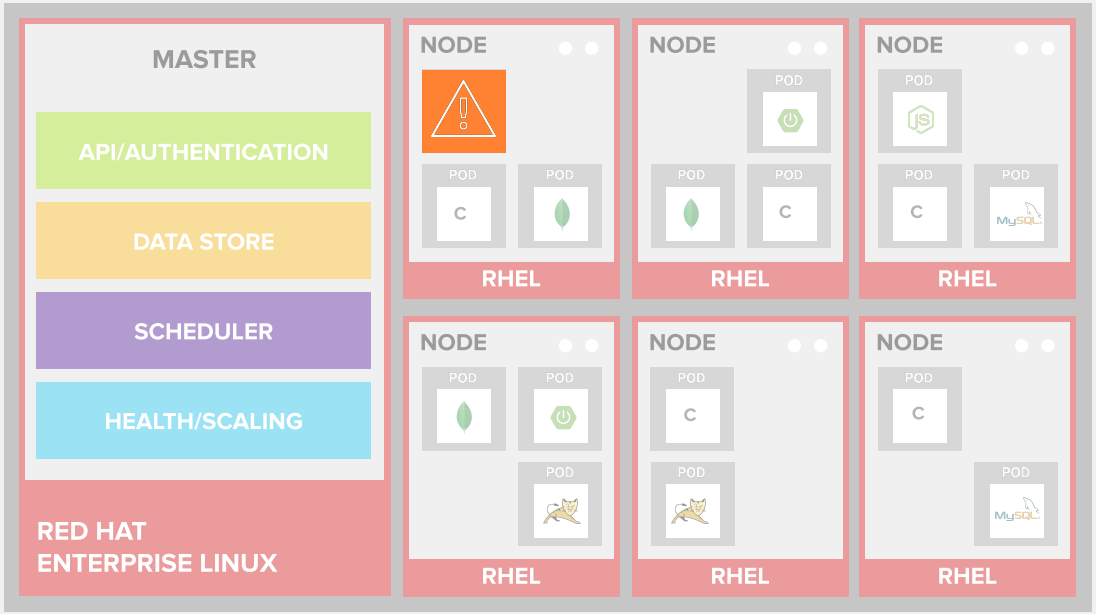
Red Hat

AUTO-HEALING FAILED PODS



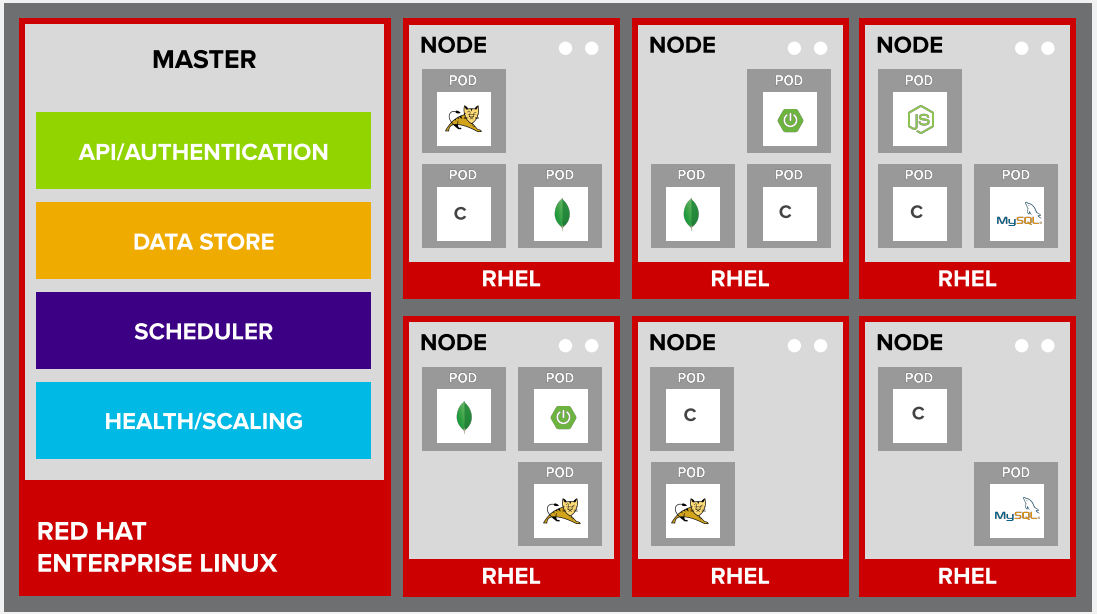
Red Hat

AUTO-HEALING FAILED PODS



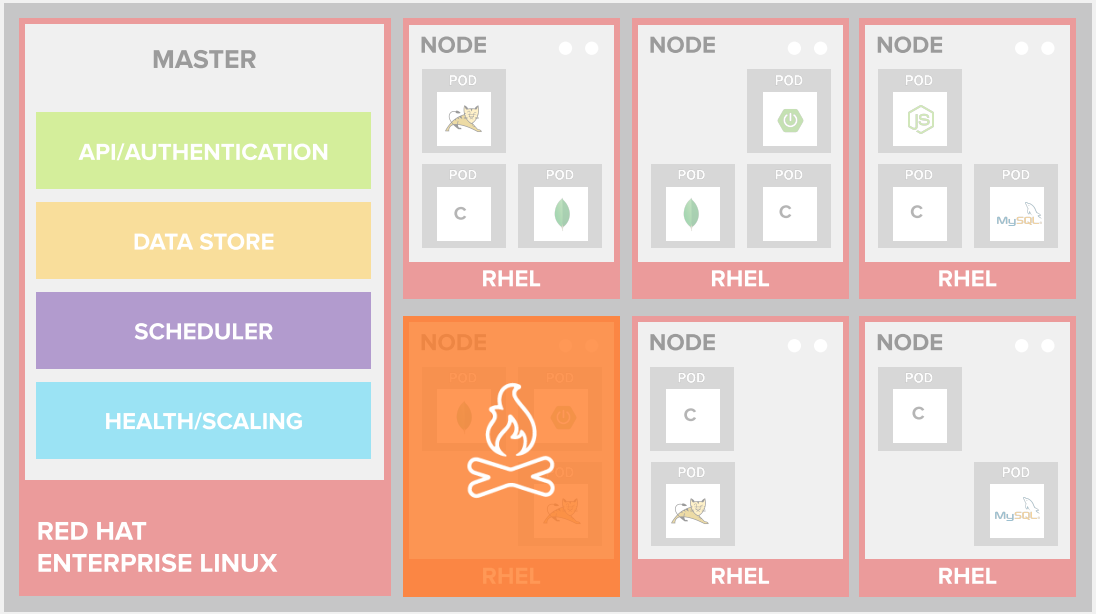
Red Hat

AUTO-HEALING FAILED PODS



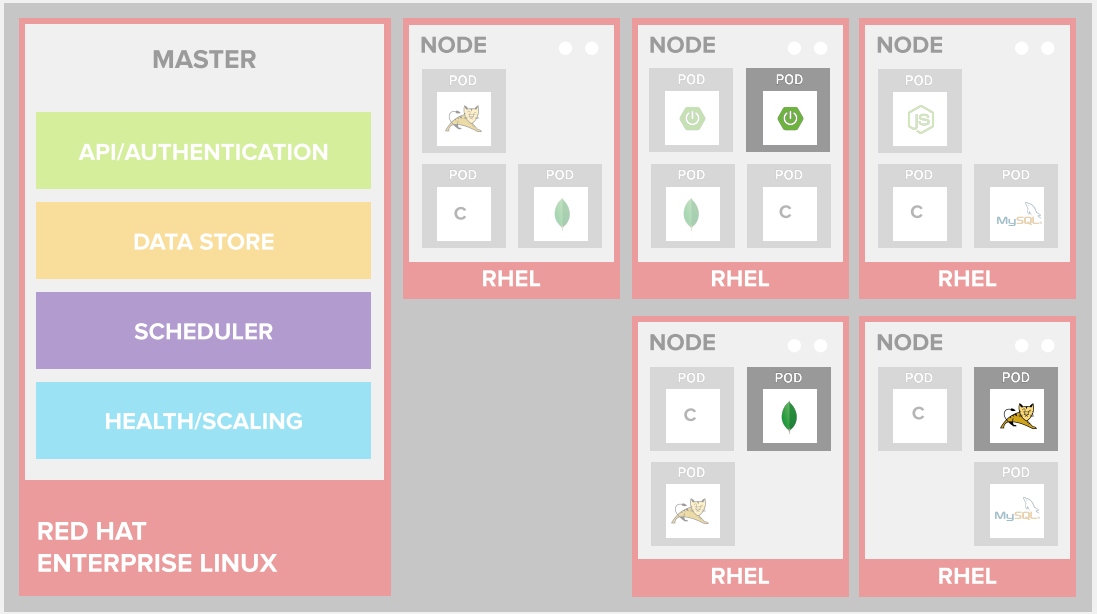
Red Hat

AUTO-HEALING PODS FOR FAILED *NODES*



Red Hat

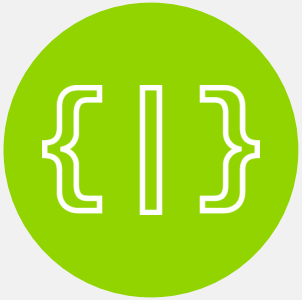
AUTO-HEALING PODS FOR FAILED *NODES*



Red Hat

CONTAINER IMAGE BUILD AND DEPLOY STRATEGIES

BUILD AND DEPLOY CONTAINER IMAGES

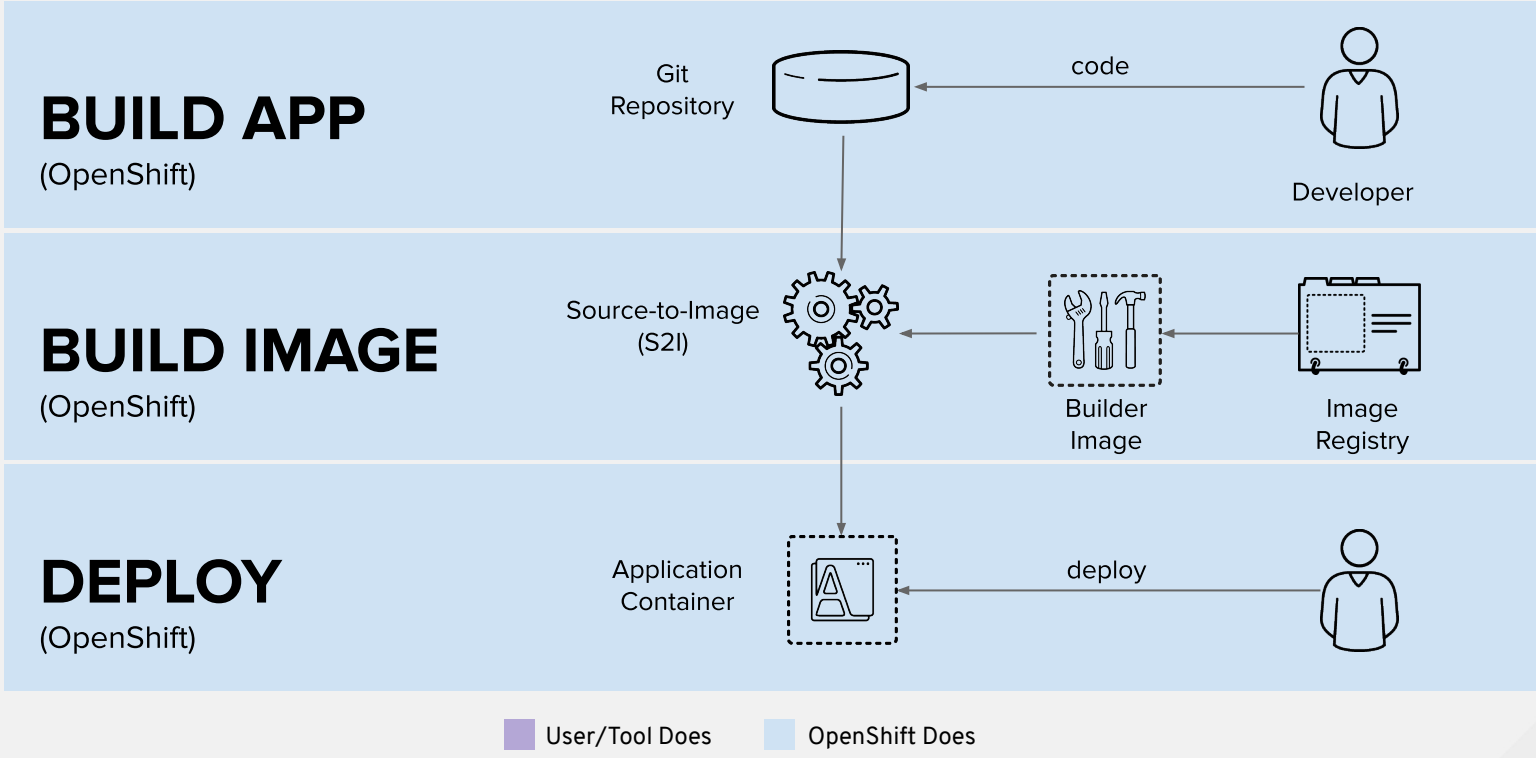


DEPLOY YOUR SOURCE CODE

DEPLOY YOUR APP BINARY

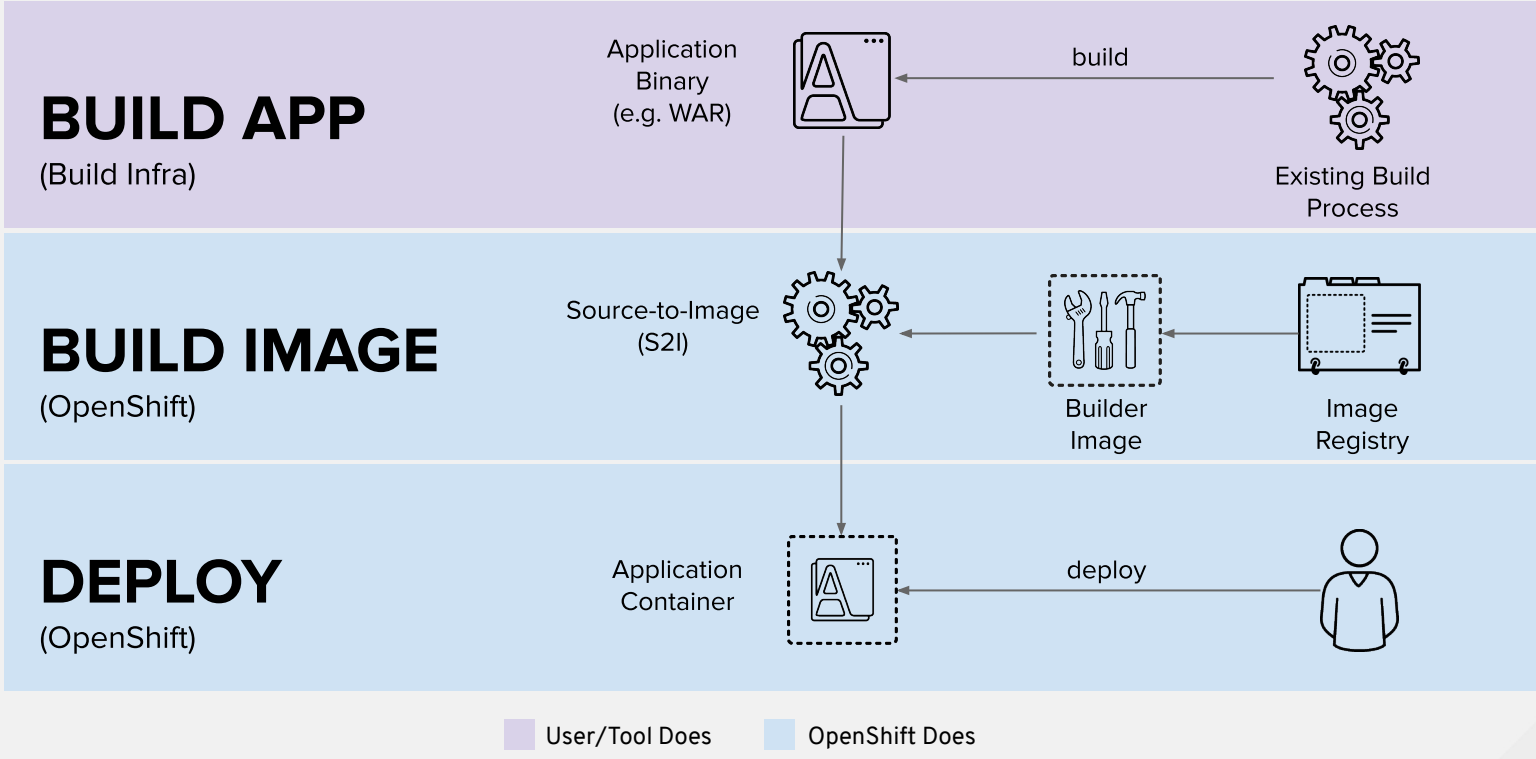
DEPLOY YOUR CONTAINER IMAGE

DEPLOY **SOURCE CODE** WITH SOURCE-TO-IMAGE (S2I)



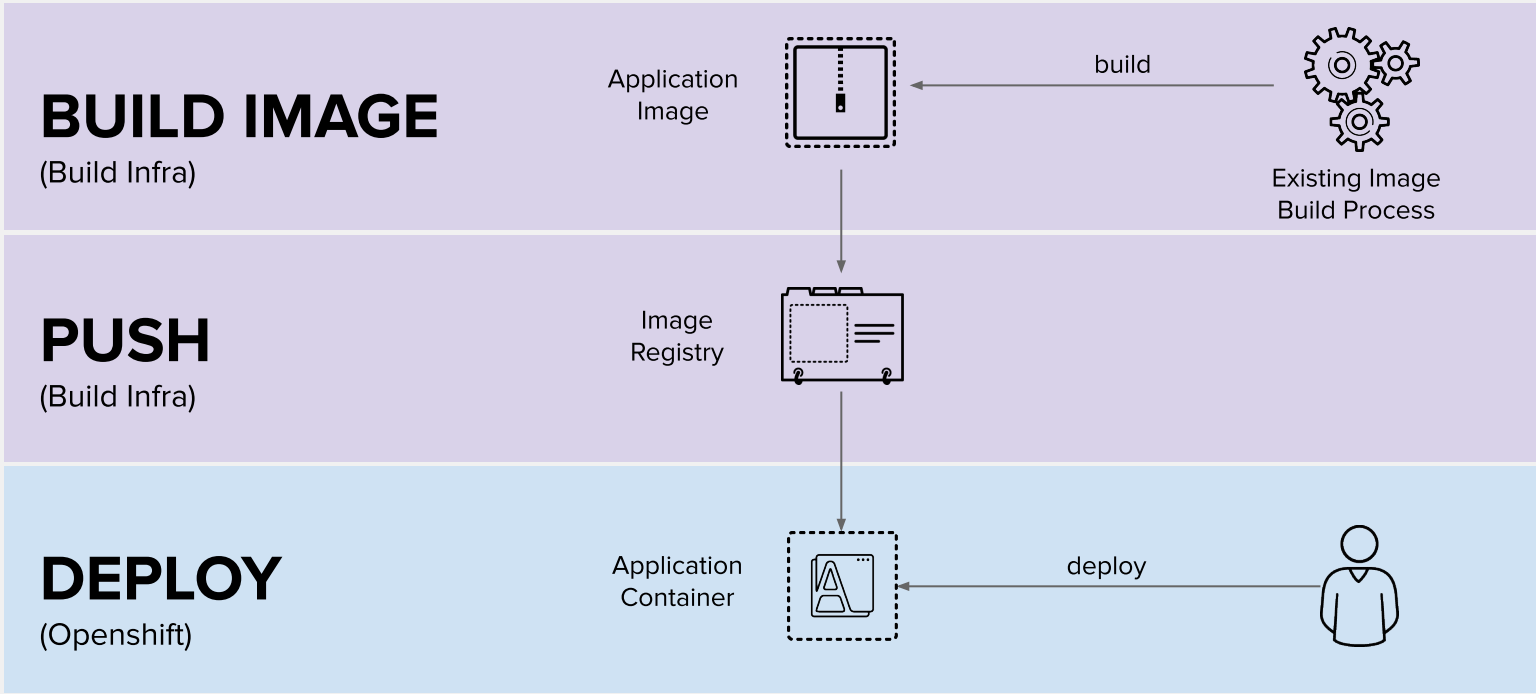
Red Hat

DEPLOY **APP BINARY** WITH SOURCE-TO-IMAGE (S2I)



Red Hat

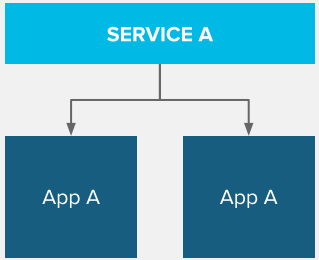
DEPLOY **DOCKER IMAGE**



Red Hat

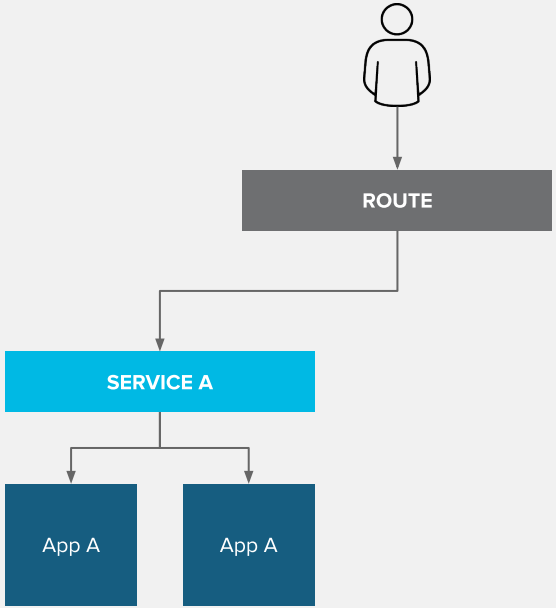
(ALMOST) DEMO TIME!

BLUE/GREEN



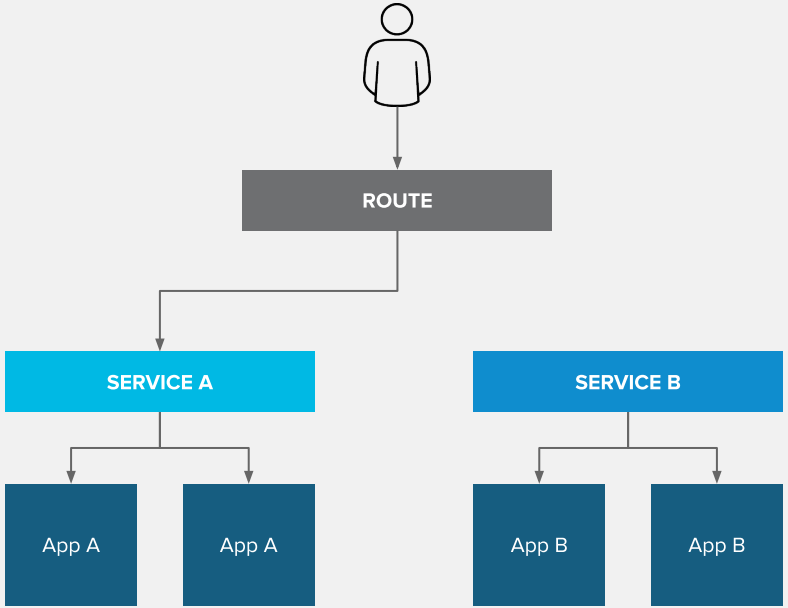
Red Hat

BLUE/GREEN



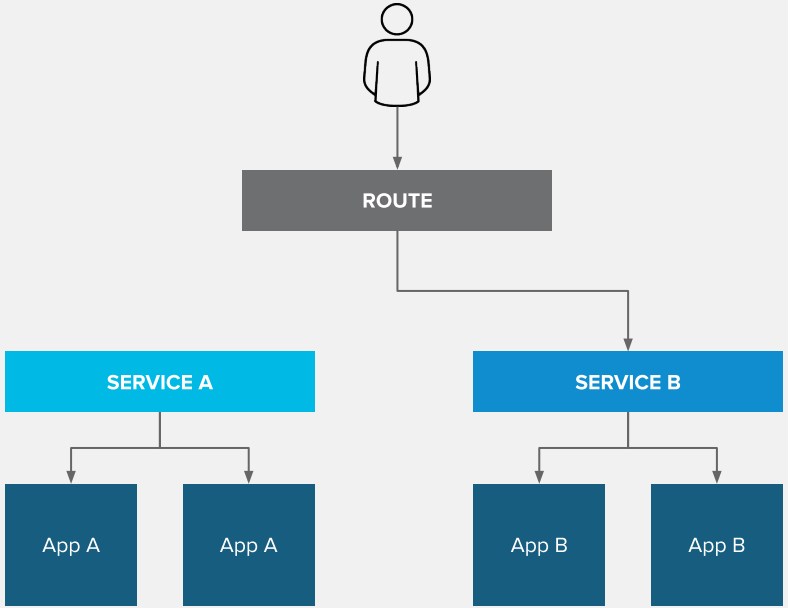
Red Hat

BLUE/GREEN



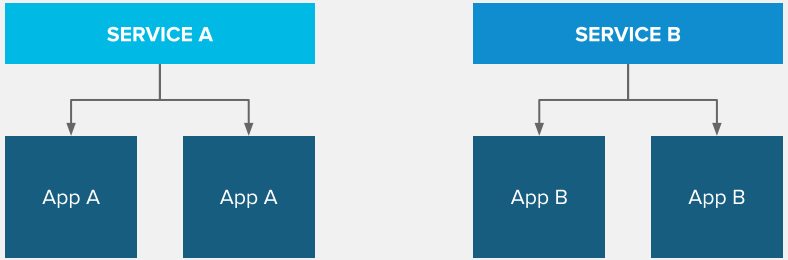
Red Hat

BLUE/GREEN



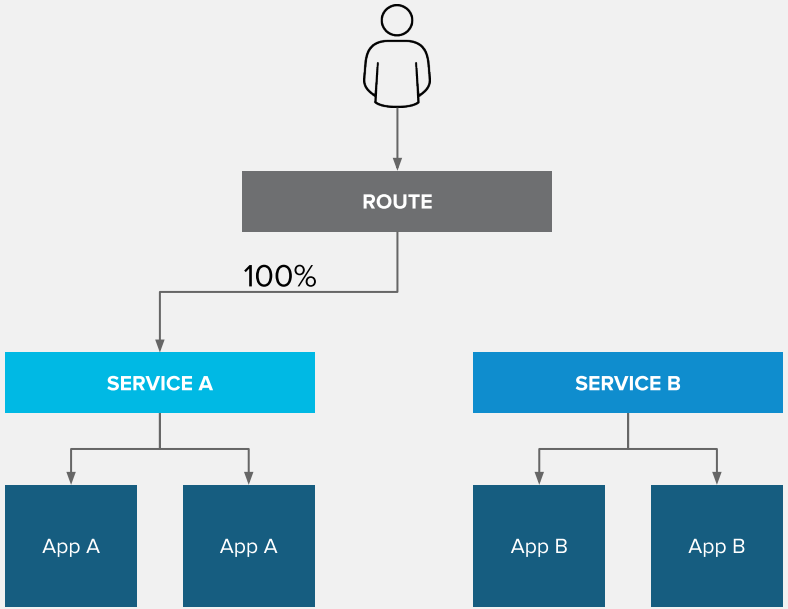
Red Hat

A/B



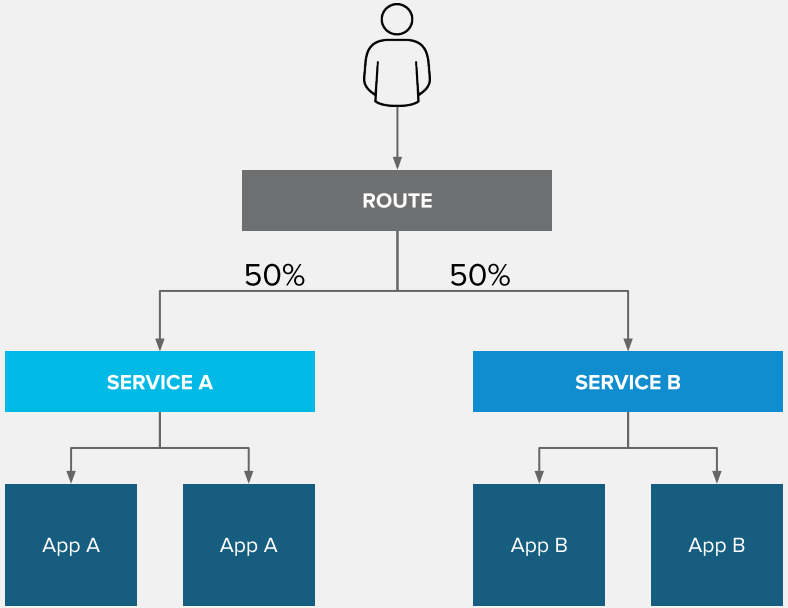
Red Hat

A/B



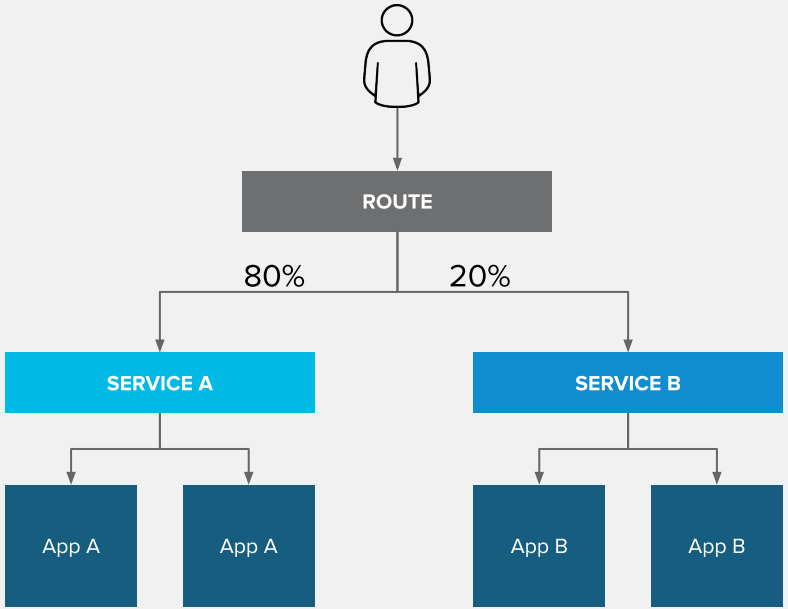
Red Hat

A/B



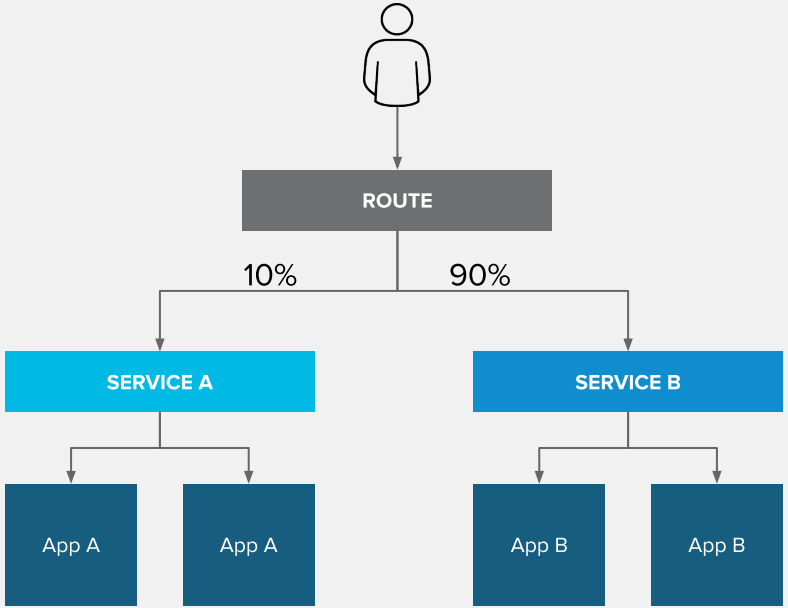
Red Hat

A/B



Red Hat

A/B



Red Hat

(ACTUALLY) DEMO TIME!

redhat.

THANK YOU

plus.google.com/+RedHat

facebook.com/redhatinc

linkedin.com/company/red-hat

twitter.com/RedHatNews

youtube.com/user/RedHatVideos