

Kubernetes with Red Hat OpenShift

World Tour

Mangesh Patankar

- Software Engineer & Developer Advocate - IBM

@MangeshPatank



Agenda

- OpenShift – What?
- Kubernetes – Container and Container Orchestration
- Overview of OpenShift Container Platform
- OpenShift Explore
 - Using the oc CLI and the Web Console
 - What is a project? Creating a project in OpenShift
 - Deploying an app from a Docker image
 - What is a Route? Creating a Route
 - Scaling an app, self-healing
 - Logging applications
- IBM Cloud Paks – Introduction
- Hands on

OpenShift

HOSTED SERVICES



Red Hat OpenShift Dedicated

- Private, high-availability Red Hat OpenShift clusters hosted on Amazon Web Services
- Delivered as a hosted service and supported by Red Hat

✓ Supported by Red Hat

[Learn more](#)



Red Hat



- Highly available Red Hat OpenShift clusters hosted on Microsoft Azure
- Delivered as a hosted service jointly engineered, operated, and supported by Red Hat and Microsoft

✓ Supported by Red Hat and Microsoft

[Learn more](#)



Red Hat



- A flexible, fully-managed service of OpenShift on IBM's public cloud
- Delivered as a hosted service and supported by IBM

✓ Supported by Red Hat and IBM

[Learn more](#)

SELF-MANAGED



Red Hat OpenShift Container Platform

- A Kubernetes platform on your own infrastructure designed with security in mind
- Build, deploy and manage your container-based applications consistently across cloud and on-premises infrastructure

✓ Supported by Red Hat

[Learn more](#)



Get started developing and deploying in the public cloud

[Learn more](#)

What is OpenShift?

OpenShift is based on top of **CRI-O/Docker Containers** and the **Kubernetes** Cluster Manager, with added developer and Operation Centric tools.

These tools enable rapid application Development , Deployment And Life Cycle Management.



Tools



Kubernetes



Docker



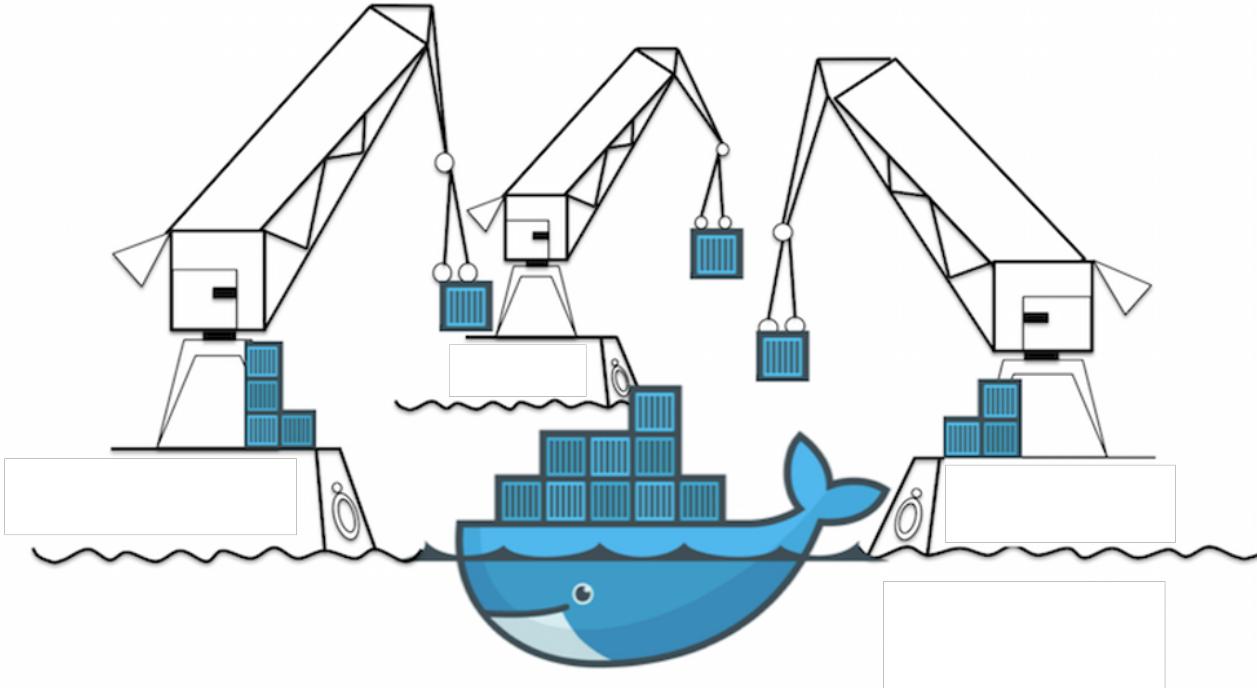
Tools



Kubernetes

Container + Orchestration

Containers



More than 5 yrs

- [Docker](#)
- [CRI-O](#)
- [Containerd](#)
- [Other CRI runtimes: frakti](#)

Why do you need containers?



Libraries

Dependencies

OS

Hardware



Development VM



QA server



Customer Data Center



Public Cloud



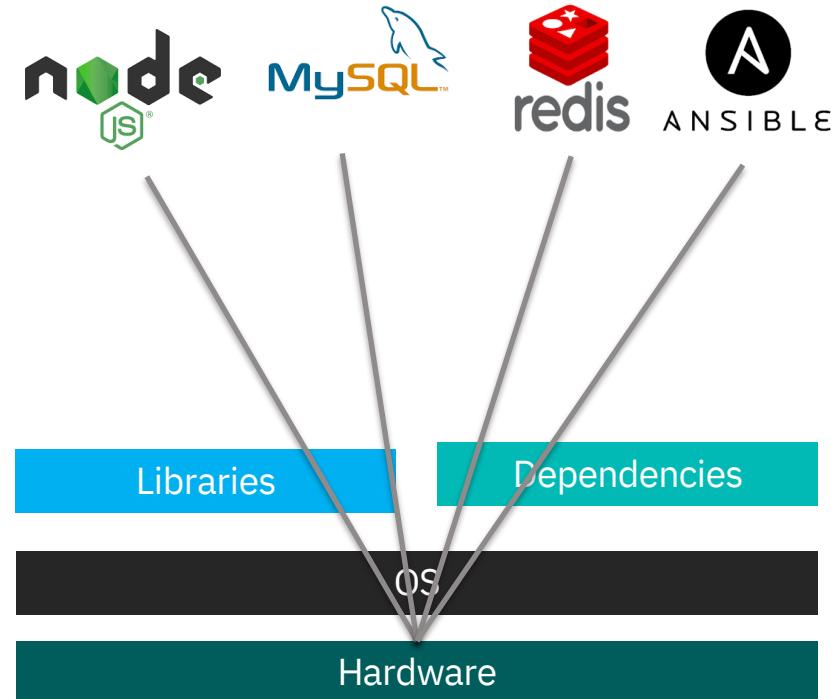
Production Cluster



Contributor's laptop

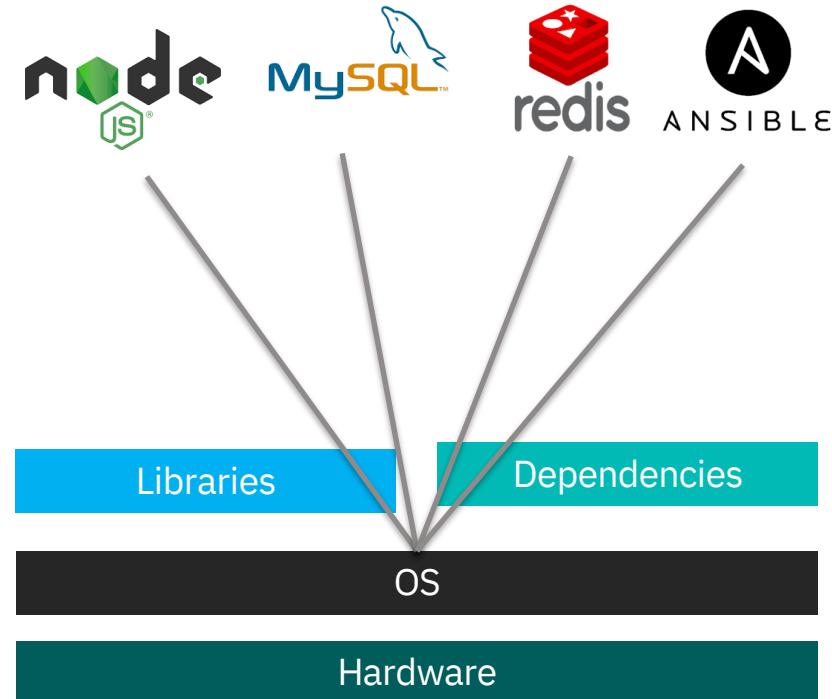
Why do you need containers?

- Right – Required Hardware
- Compatibility/ Dependency
- Long setup time



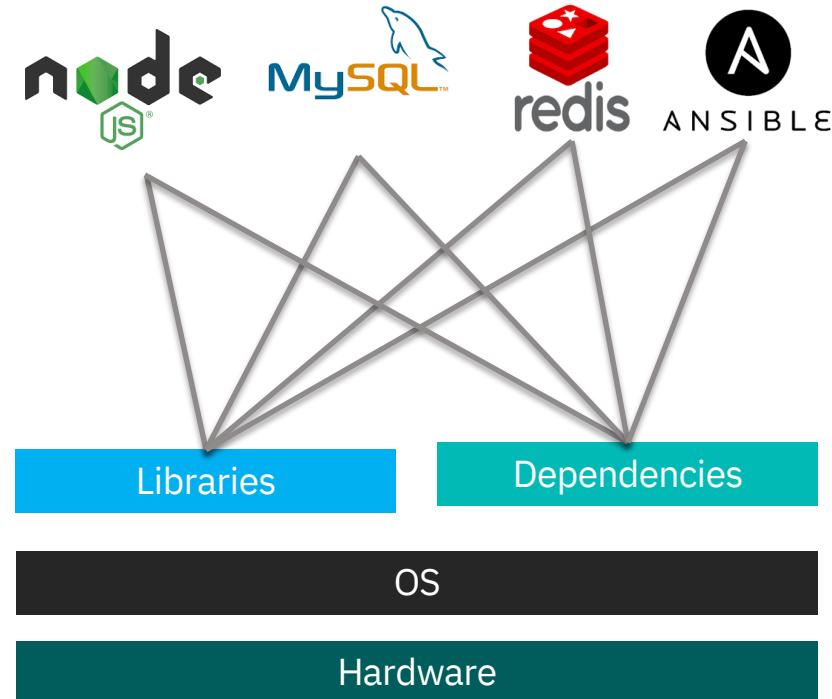
Why do you need containers?

- Compatibility with OS



Why do you need containers?

- Compatibility to work together
- Dependency on particular version supported etc



Why do you need containers?

- Compatibility/ Dependency
- Long setup time
- Different Dev/Test/Prod

Dev

Build

Ship

Multiplicity of hardware environments



Multiplicity of Stacks



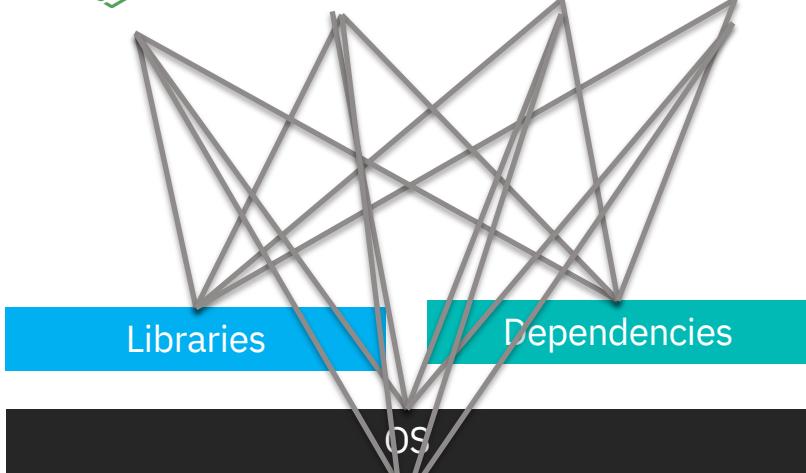
Do services and apps interact appropriately?



Can I migrate smoothly and quickly?



ANSIBLE



Add layer



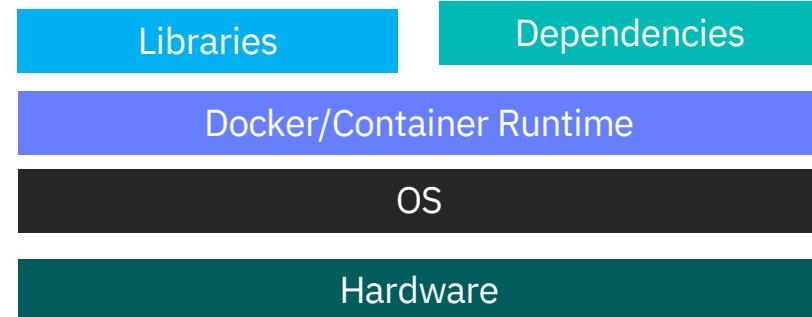
- Remove Compatibility/ Dependency



Container

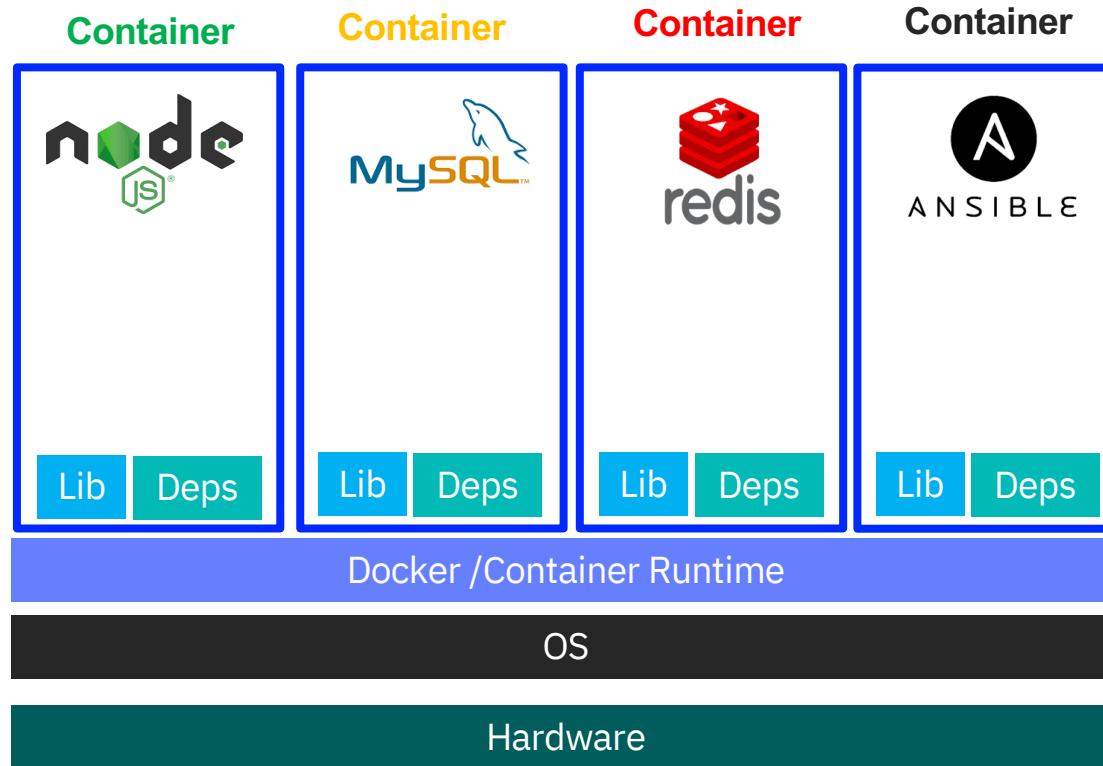


- Remove Compatibility/ Dependency

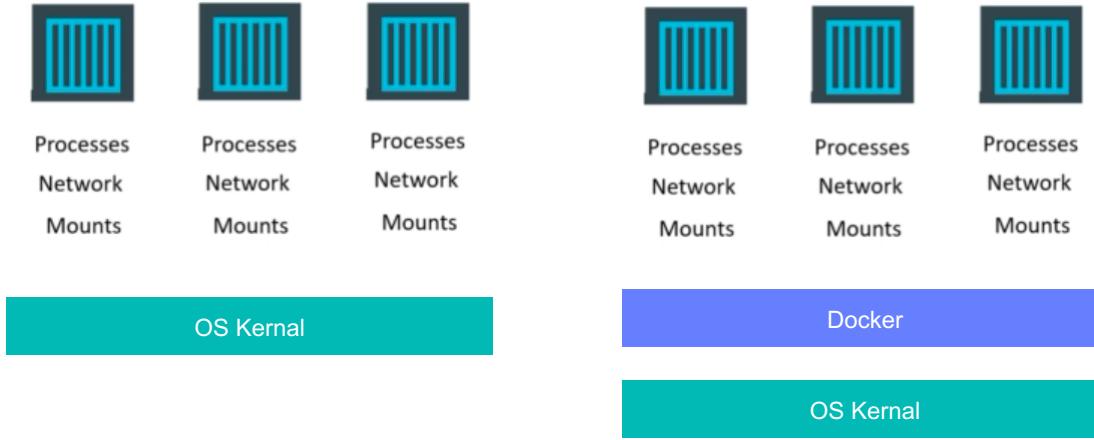


What Container can do?

- Remove Compatibility/Dependency
- Different Dev/Test/Prod
- Scalability
- Polygot Programming



What are containers?



PID - process IDs

USER - user and group IDs

UTS - hostname and domain name

NS - mount points

NET - Network devices, stacks, ports

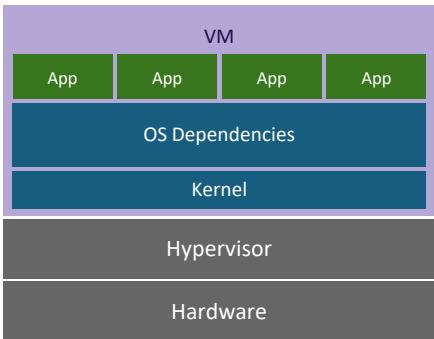
IPC - inter-process communications, message queues

cgroups - controls limits and monitoring of resources

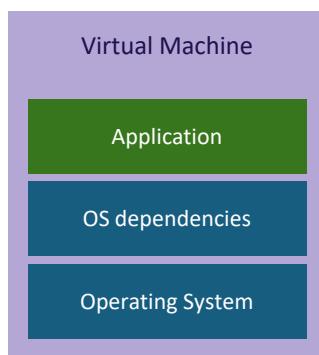
Lxc, lxd, lxcfs

How Containers differ from VMs ?

VIRTUAL MACHINES

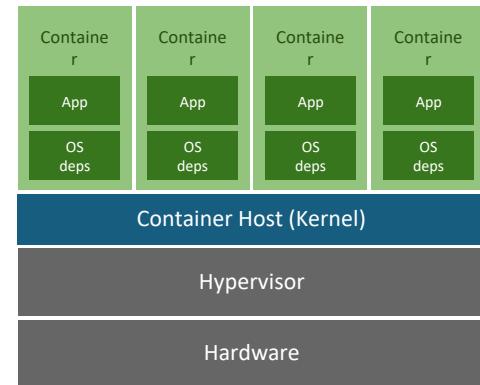


VM isolates the hardware



- + VM Isolation
 - Complete OS
 - Static Compute
 - Static Memory
 - High Resource Usage
- + Container Isolation
 - + Shared Kernel
 - + Burstable Compute
 - + Burstable Memory
 - + Low Resource Usage

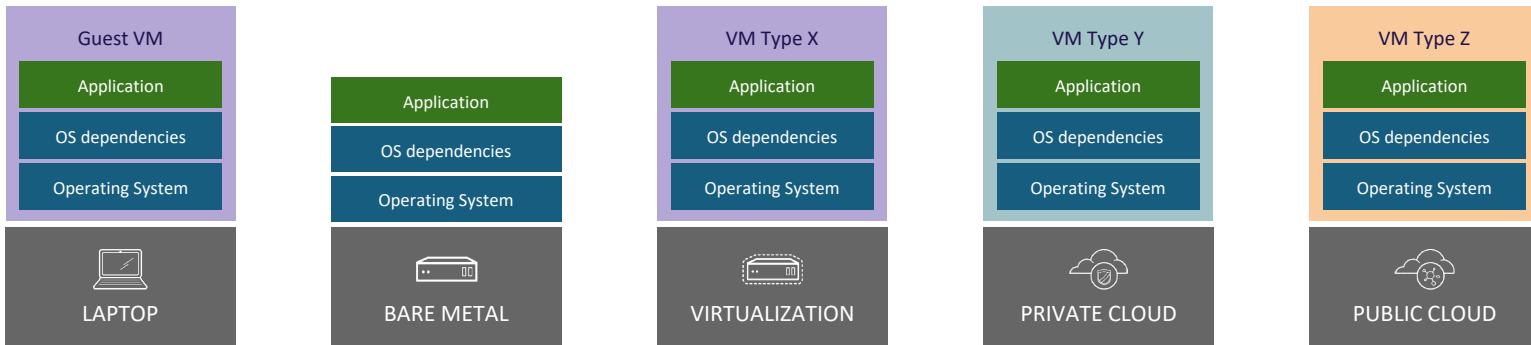
CONTAINERS



Container isolates the process

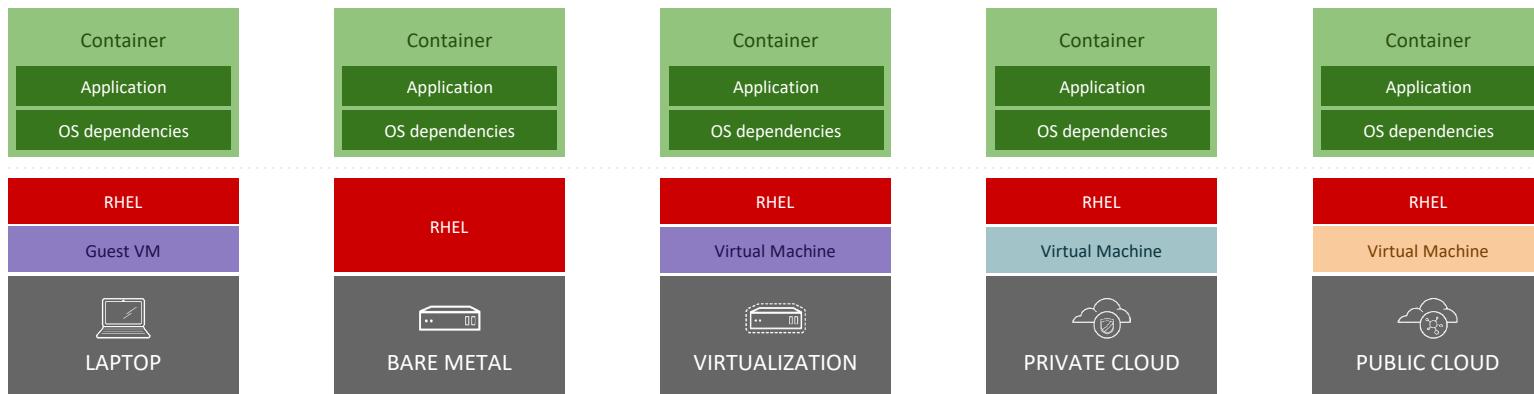
Application Portability with VM

Virtual machines are **NOT** portable across hypervisor and do **NOT** provide portable packaging for applications

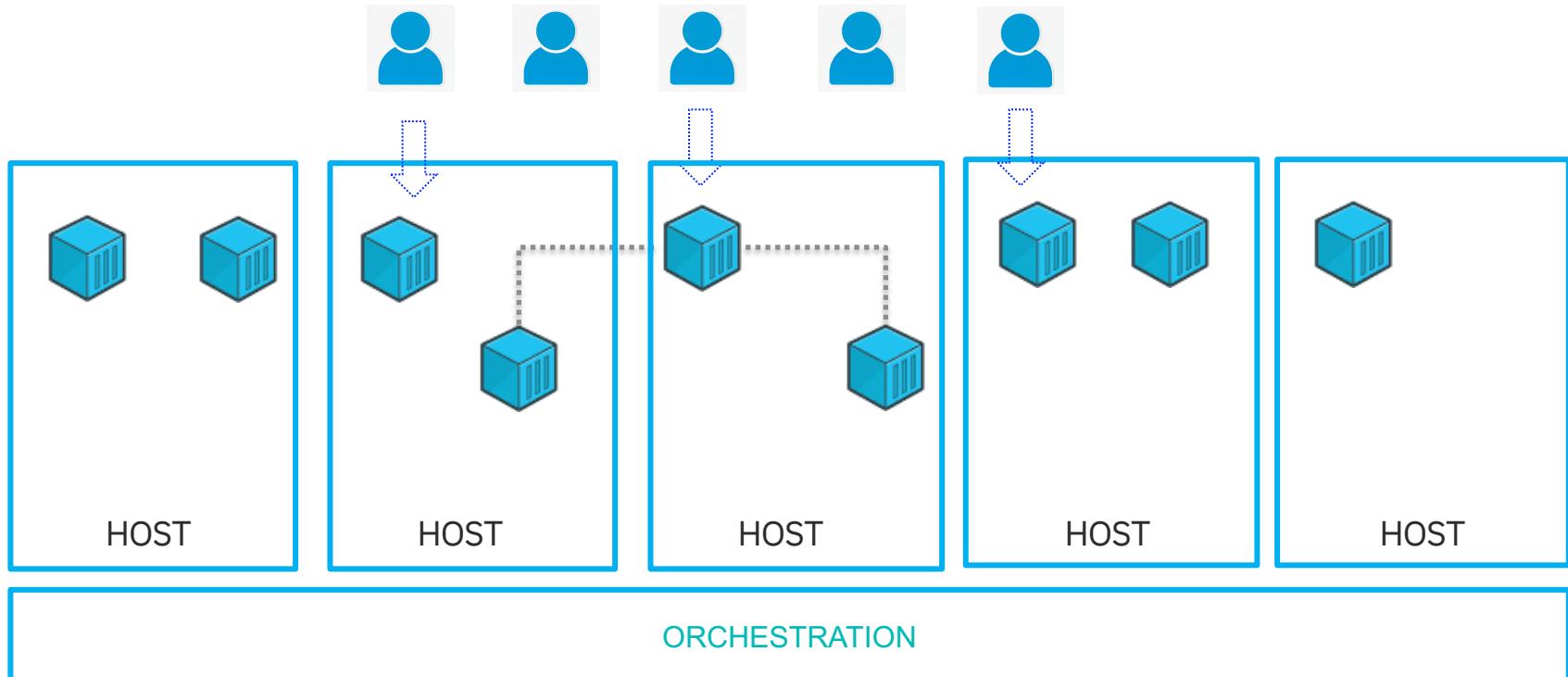


Application Portability with Containers

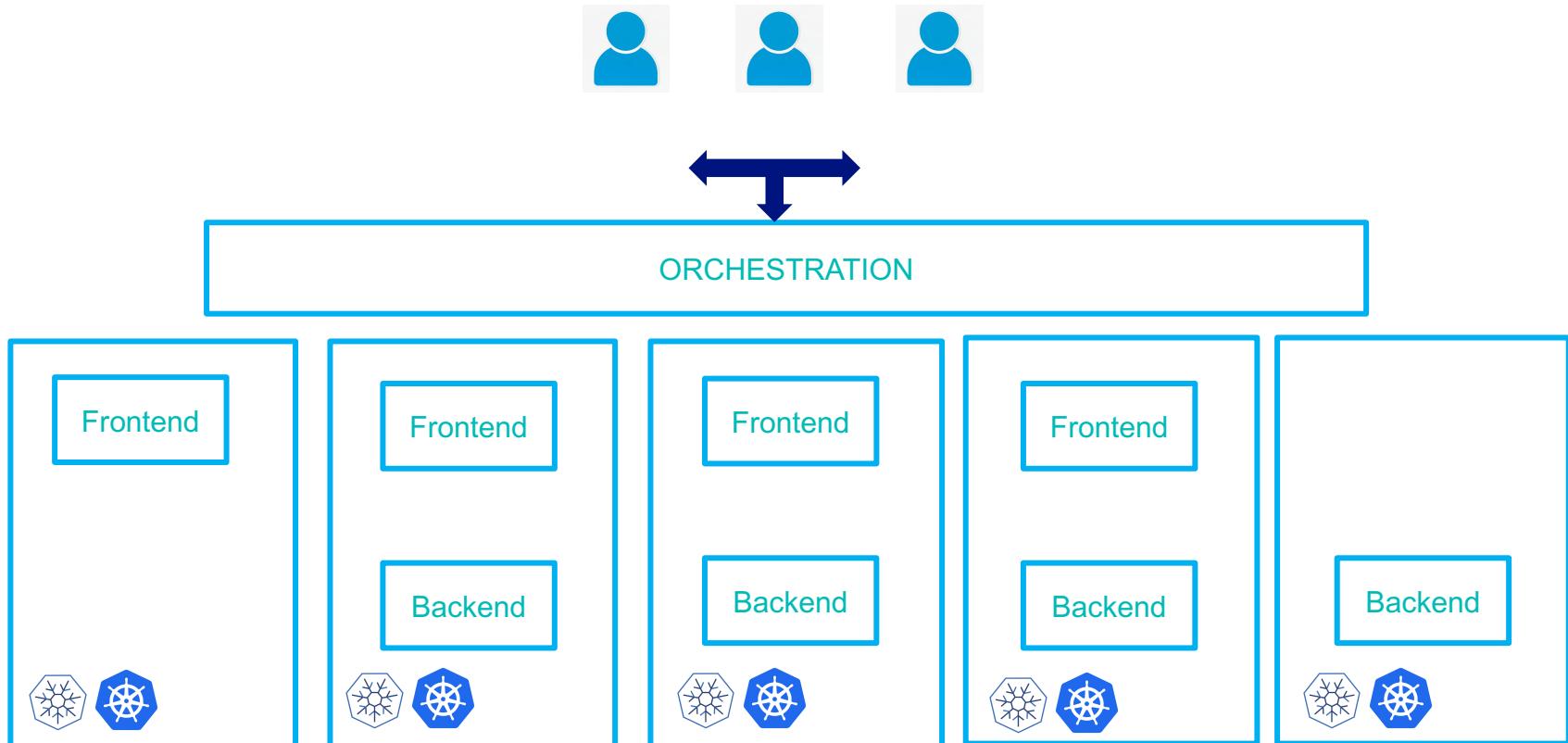
RHEL Containers + RHEL Host = Guaranteed Portability
Across Any Infrastructure



Container Orchestration



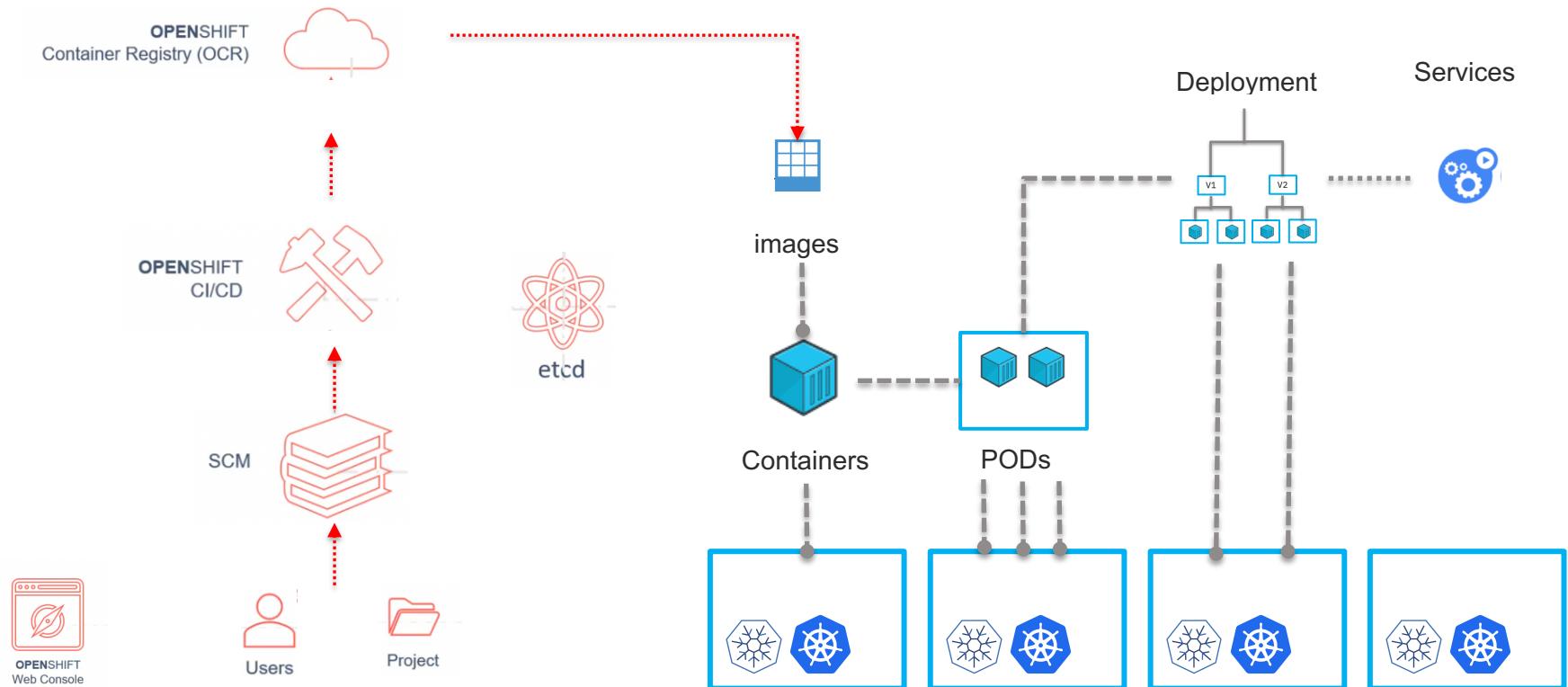
Orchestration – k8s



What is K8s?

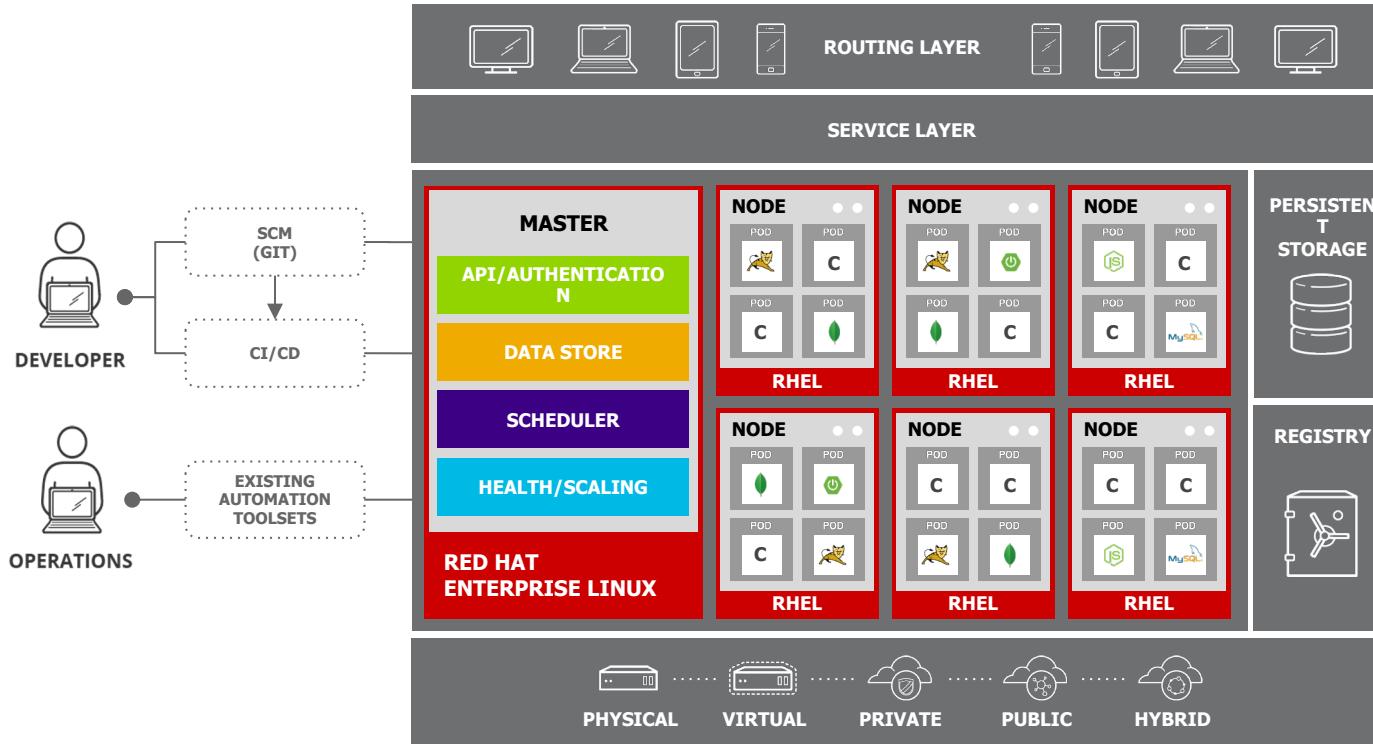
- Based on Google's Borg & Omega - Open Source - Container Orchestrator
- Open Governance
 - Cloud Native Compute Foundation
- Adoption by Enterprise
 - RedHat, Microsoft, IBM and Amazon
- Help to automate DevOps – Deployment, scaling and management of containerized apps
- Helps organize container in logical units(pods, nodes)
- Helps in resource monitoring and logging

OpenShift Components



Kubernetes

OpenShift Architecture





DevOps Tools and User Experience

Web Console, CLI, REST API, SCM integration

Containerized Services

Auth, Networking, Image Registry

Runtimes and xPaaS

Java, Ruby, Node.js and more

Kubernetes

Container orchestration
and management

Etcd

Cluster state and configs

OpenShift Kubernetes Extensions

Docker

Container API and packaging format

RHEL

Container optimized OS

OpenShift Explore

Using the oc CLI and the Web Console

What is a project? Creating a project in OpenShift

Deploying an app from a Docker image

What is a Route? Creating a Route

Scaling an app, self-healing

Logging applications

CLI

Console

CFI

Play



```
Apples-MacBook-Pro:~ apple$ oc  
OpenShift Client
```

This client helps you develop, build, deploy, and run your applications on any OpenShift or Kubernetes compatible platform. It also includes the administrative commands for managing a cluster under the 'adm' subcommand.

To create a new application, login to your server and then run new-app:

```
oc login https://mycluster.mycompany.com  
oc new-app centos/ruby-25-centos7~https://github.com/sclorg/ruby-ex.git  
oc logs -f bc/ruby-ex
```

This will create an application based on the Docker image 'centos/ruby-25-centos7' that builds the source code from GitHub. A build will start automatically, push the resulting image to the registry, and a deployment will roll that change out in your project.

Once your application is deployed, use the status, describe, and get commands to see more about the created components:

```
oc status  
oc describe deploymentconfig ruby-ex  
oc get pods
```

To make this application visible outside of the cluster, use the expose command on the service we just created to create a 'route' (which will connect your application over the HTTP port to a public domain name).

```
oc expose svc/ruby-ex  
oc status
```

You should now see the URL the application can be reached at.

To see the full list of commands supported, run 'oc --help'.

More Details

```
Usage:
  oc [flags]

Basic Commands:
  types      An introduction to concepts and types
  login      Log in to a server
  new-project Request a new project
  new-app    Create a new application
  status     Show an overview of the current project
  project    Switch to another project
  projects   Display existing projects
  explain    Documentation of resources
  cluster    Start and stop OpenShift cluster

Build and Deploy Commands:
  rollout    Manage a Kubernetes deployment or OpenShift deployment config
  rollback   Revert part of an application back to a previous deployment
  new-build  Create a new build configuration
  start-build Start a new build
  cancel-build Cancel running, pending, or new builds
  import-image Imports images from a Docker registry
  tag        Tag existing images into image streams

Application Management Commands:
  get        Display one or many resources
  describe  Show details of a specific resource or group of resources
  edit       Edit a resource on the server
  set        Commands that help set specific features on objects
  label      Update the labels on a resource
  annotate   Update the annotations on a resource
  expose     Expose a replicated application as a service or route
  delete    Delete one or more resources
  scale      Change the number of pods in a deployment
  autoscale  Autoscale a deployment config, deployment, replication controller, or replica set
  secrets   Manage secrets
  serviceaccounts Manage service accounts in your project

Troubleshooting and Debugging Commands:
  logs      Print the logs for a resource
  rsh       Start a shell session in a pod
  rsync     Copy files between local filesystem and a pod
  port-forward Forward one or more local ports to a pod
  debug     Launch a new instance of a pod for debugging
  exec      Execute a command in a container
  proxy     Run a proxy to the Kubernetes API server
  attach    Attach to a running container
  run       Run a particular image on the cluster
```

OpenShift Web Console

OPENSHIFT CONTAINER PLATFORM Service Catalog ▾

My Projects + Create Project

Search Catalog

Browse Catalog

All Languages Databases Middleware CI/CD Other

Deploy Image Import YAML / JSON Select from Project

Filter ▾ 40 Items

 .NET	 .NET	 .NET	 .NET		
.NET Core	.NET Core + PostgreSQL (Persistent)	.NET Core Example	.NET Core Runtime Example	3scale-gateway	amp-apicast-wildcard-router
 amp-pvc	 Apache HTTP Server	 Apache HTTP Server (httpd)	 CakePHP + MySQL	 CakePHP + MySQL (Ephemeral)	 Dancer + MySQL
 Dancer + MySQL (Ephemeral)	 Django + PostgreSQL	 Django + PostgreSQL (Ephemeral)	 Jenkins	 Jenkins (Ephemeral)	 MariaDB
 MariaDB (Ephemeral)	 MongoDB	 MongoDB (Ephemeral)	 MySQL	 MySQL (Ephemeral)	 Nginx HTTP server and a reverse proxy

My Projects

5 of 34 Projects

View All

demoproj created by IAM#mapatank@in.ibm.com 12 hours ago

kabanero-samples created by IAM#mapatank@in.ibm.com 4 months ago

kabanero created by system:serviceaccount:default:default a month ago

kabanero-samples2 created by IAM#mapatank@in.ibm.com 25 days ago

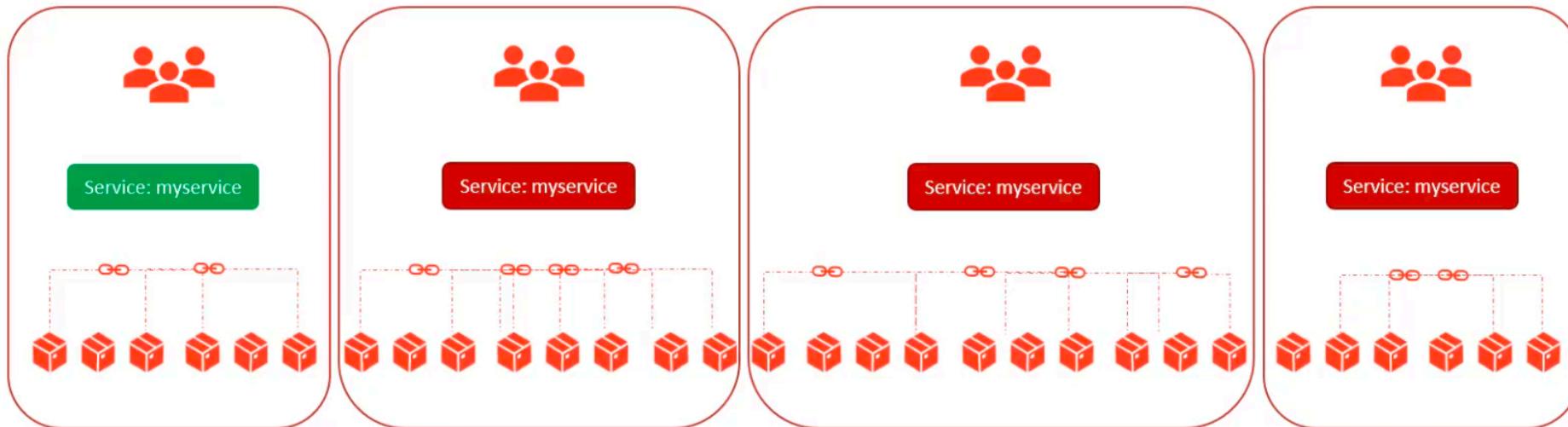
blog blog - created by IAM#mapatank@in.ibm.com 4 months ago

blog

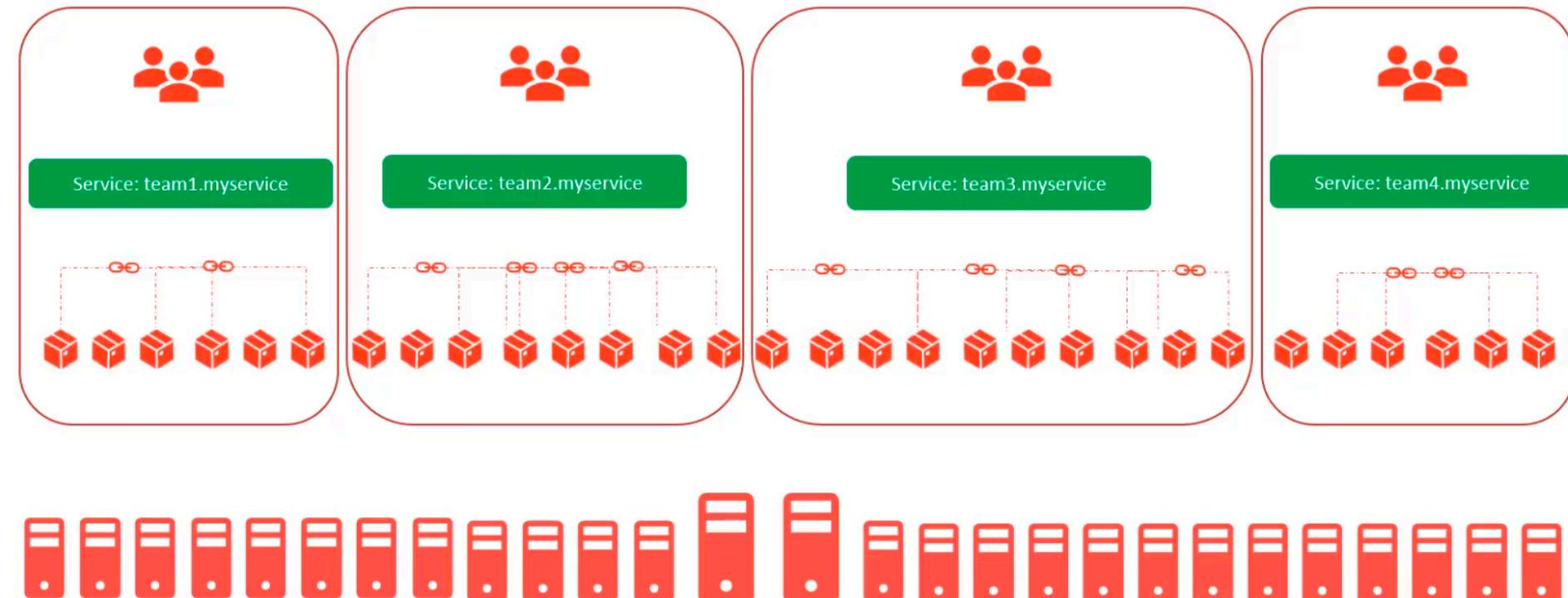
Recently Viewed

Node.js MongoDB

Projects



Isolation with Projects



Deploy an Image

OPENSHIFT CONTAINER PLATFORM Service Catalog

Node.js

Information Configuration Results

* Add to Project demoproj

Version 10 — latest

* Application Name nodejsdemo

* Git Repository <https://github.com/openshift/nodejs-ex.git>

Try Sample Repository ↗

If you have a private Git repository or need to change application defaults, view advanced options.

Cancel < Back Create

Projects

+ Create Project

View All

amp-pvc Apache HTTP Server

Dancer + MySQL (Ephemeral)

MariaDB (Ephemeral)

NGINX

PostgreSQL

PostgreSQL (Ephemeral)

Python

Rails + PostgreSQL (Ephemeral)

Rails

Redis (Ephemeral)

Ruby

system

WildFly

Objects

oproj

nero-samples

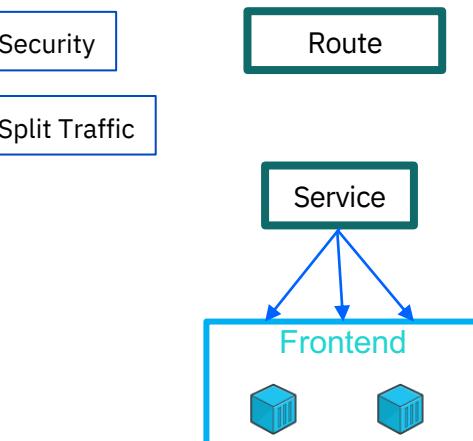
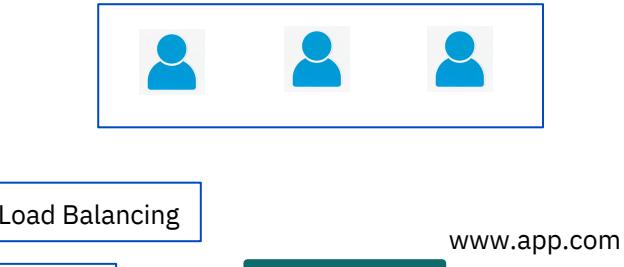
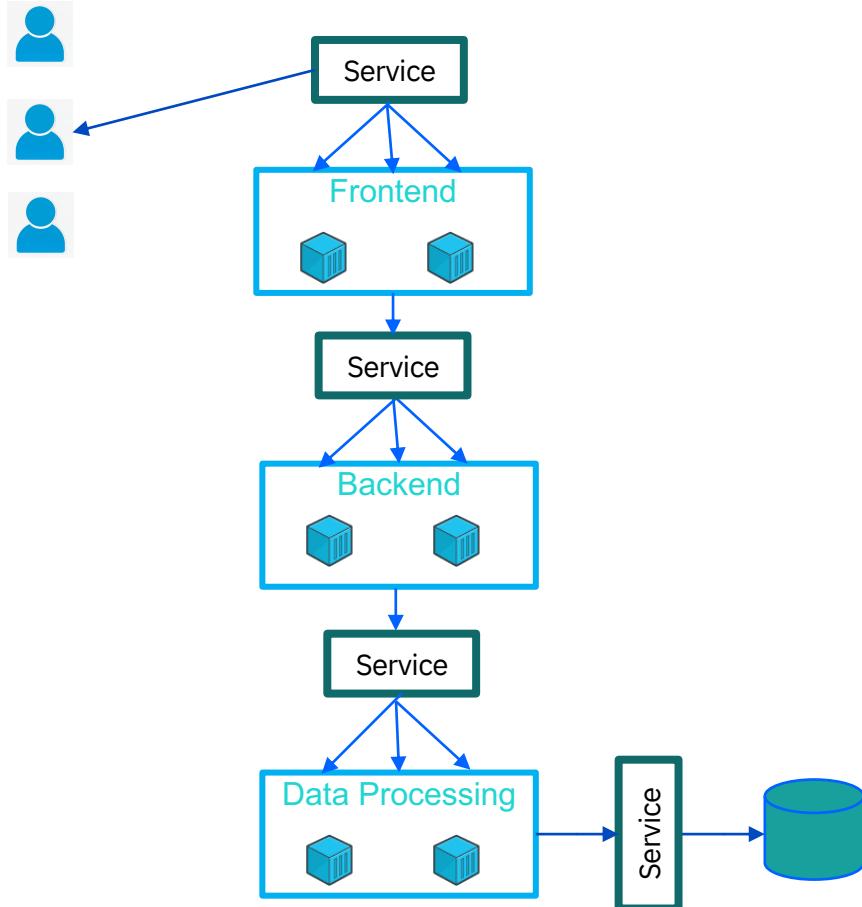
nero

nero-samples2

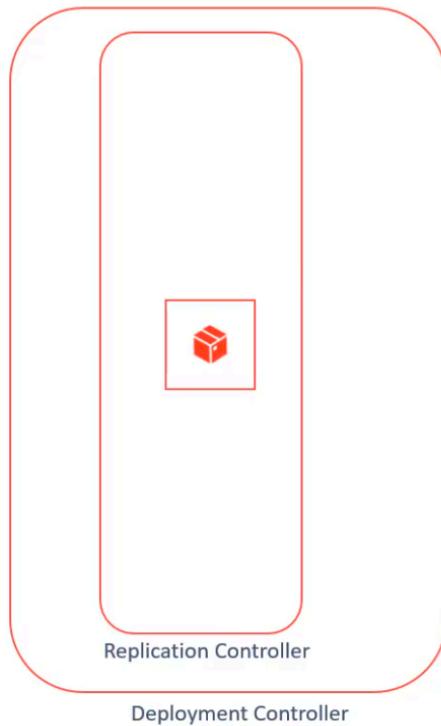
Recently Viewed

nodejsdemo MongoDB

Routes - Services

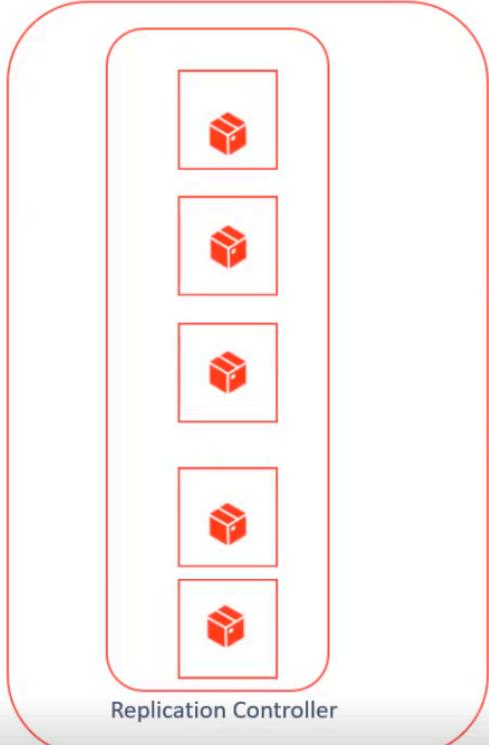


Scale



```
spec:  
  replicas: 1
```

Scale (Contd..)



Replication Controller

APPLICATION
simple-webapp <https://simple-webapp-docker-my-webapplication.192.168.99.100.nip.io>

DEPLOYMENT CONFIG
simple-webapp, #1

CONTAINERS

simple-webapp

- Image: my-webapplication/simple-webapp 64ad4d7 212.2 MB
- Build: simple-webapp, #1
- Source: Add new file 1c42c7e
- Ports: 8080/TCP

NETWORKING

Service - Internal Traffic

simple-webapp

8080/TCP (8080-tcp) → 8080

Routes - External Traffic

<https://simple-webapp-docker-my-webapplication.192.168.99.100.nip.io>

Route simple-webapp-docker, target port 8080-tcp

Traffic Split

simple-webapp	90%
simple-webapp-docker	10%

Deployment Controller

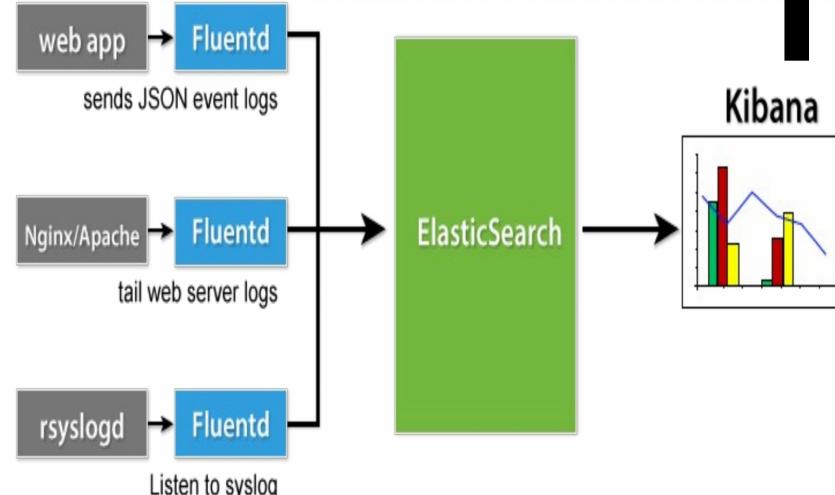
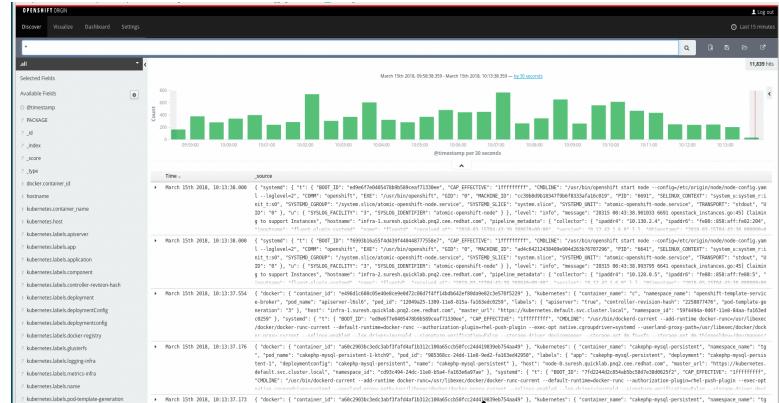
Logging

OpenShift Container Platform provides a solution called *EFK*.

EFK stands for: *Elasticsearch*, *Fluentd*, and *Kibana*



- **Elasticsearch**, an open source search engine and object store that provides a distributed RESTful API for logs
 - **Fluentd**, a data collector project that gathers logs from the application nodes and sends them to the Elasticsearch service
 - **Kibana**, a web interface for Elasticsearch
-
- ...and **Curator**, for the maintenance of Elasticsearch on a per-project basis.



IBM Containerized Software's on OpenShift Container Platform

Cloud Paks – *Enterprise-ready Containerized Software*

A faster, more secure way to move your core business applications to any cloud through enterprise-ready containerized software solutions

IBM containerized software

Packaged with Open Source components, pre-integrated with the common operational services, and secure by design



Container platform and operational services

Logging, monitoring, security, identity access management



Complete yet simple

Application, data and AI services, fully modular and easy to consume

IBM certified

Full software stack support, and ongoing security, compliance and version compatibility

Run anywhere

On-premises, on private and public clouds, and in pre-integrated systems

Cloud Paks – Pre-integrated for cloud use cases

Reduce dev time up to 84%*

Cloud Pak for Applications

Build, deploy, and run applications

IBM containerized software



Container platform and operational services 

Make data ready for AI in days

Cloud Pak for Data

Collect, organize, and analyze data

IBM containerized software



Container platform and operational services 

Eliminate 33% of integration cost

Cloud Pak for Integration

Integrate applications, data, cloud services, and APIs

IBM containerized software



Container platform and operational services 

Reduce manual processes up to 80%*

Cloud Pak for Automation

Transform business processes, decisions, and content

IBM containerized software



Container platform and operational services 

Reduce IT op expense by up to 75%*

Cloud Pak for Multicloud Management

Multicloud visibility, governance, and automation

IBM containerized software



Container platform and operational services 

NEW

Includes Red Hat OpenShift w/no functional restrictions



IBM Cloud Pak System



IBM Cloud



AWS



Azure



Google Cloud



Edge



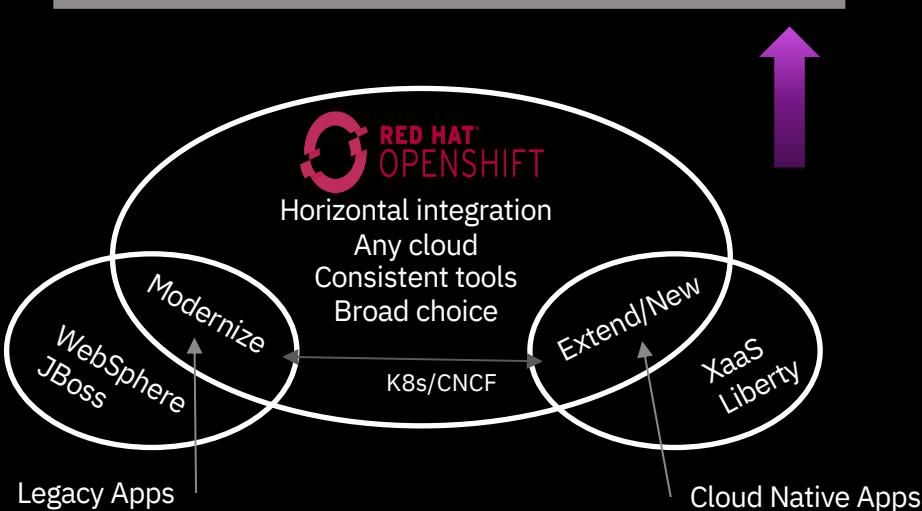
Private



Systems

Cloud Pak for Applications “raises the bar”

Cloud Pak for Applications delivers enterprise value above the OpenShift platform



Cloud Pak for Applications' enterprise value:

- IBM Cloud Pak: enterprise-ready, open, faster, and more secure
- Simplifies creation of best-of-breed cloud-native architecture
- Delivers new & existing middleware and modernization tools
- Reduces developer overhead, especially for choice/compliance/audit
- Reduces complexity of managing apps at scale

Kubernetes with Red Hat OpenShift

World Tour

THANK YOU

Mangesh Patankar

@MangeshPatank

Next Hands on Lab





Kubernetes with OpenShift

World Tour



Courses and tutorials (earn badges)
<https://cognitiveclass.ai>



Check out the IBM Developer website
<https://developer.ibm.com/>
Code Patterns



Join the community on Slack
<http://ibm.biz/dev-community>

Join our Meetup groups

Bangalore: <https://bit.ly/ibm-blr>

Delhi and NCR: <https://bit.ly/ibm-delhi>

Mumbai: <https://bit.ly/ibm-mumbai>

Pune: <https://bit.ly/ibm-pune>

Hyderabad: <https://bit.ly/ibm-hyd>

Chennai: <https://bit.ly/ibm-chennai>

Back Up Slides