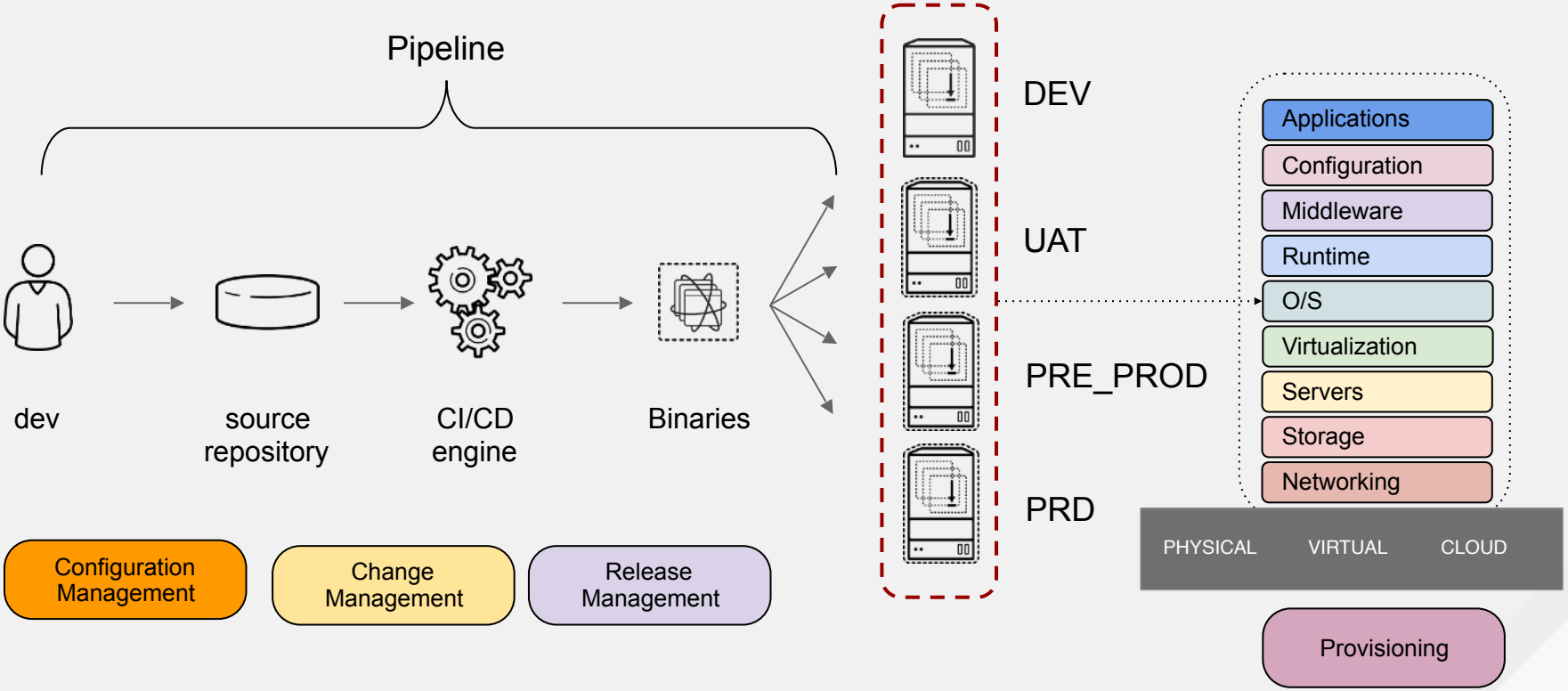
# OPENSHIFT CONTAINER PLATFORM
# Middleware Services

## OVERVIEW

# CHALLENGES !






GENERAL DISTRIBUTION

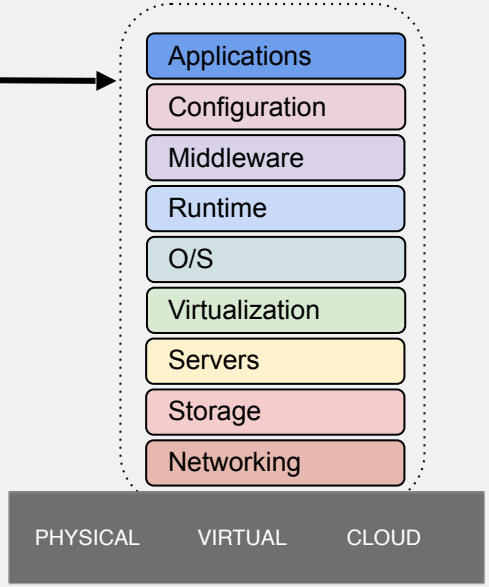
redhat

# CHALLENGES !

Dev

Java | NodeJS | Python | PHP | Perl | Ruby | .NET Core

DBs | NoSql | Caching | Integration | Messaging | Processes Workflows | Business Rules



Ops

## Challenges

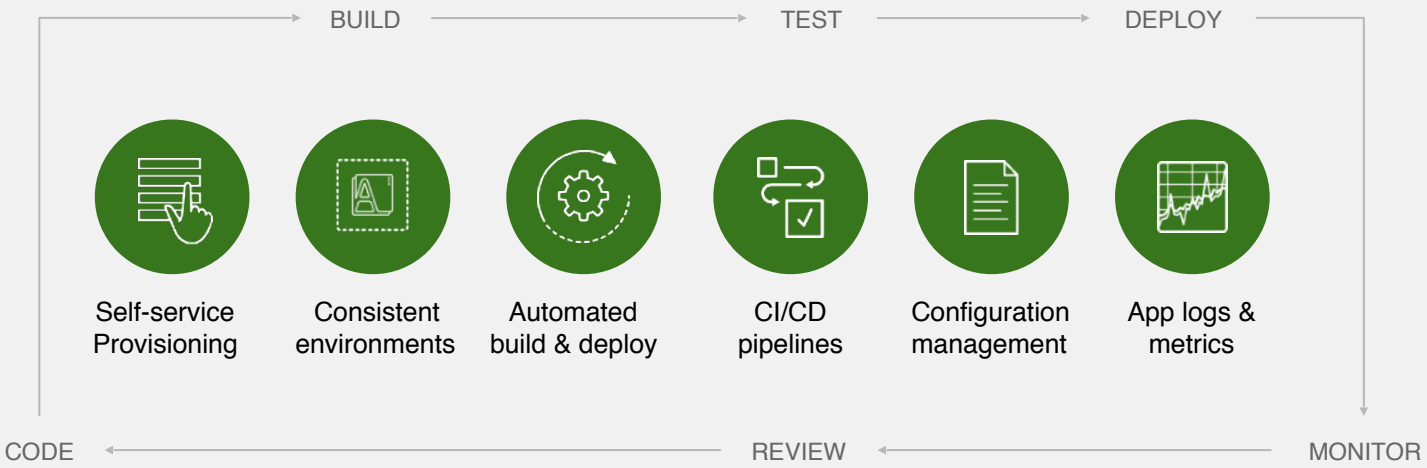
- Coding
- Building Artifacts
- Testing (Unit, Integration, Load)
- Quality Review
- Deployment
- Configuration Management
- Traceability
- Observability
- Infrastructure
- High Availability
- Resiliency
- Security
- Scalability
- Logging
- Metrics
- RBAC

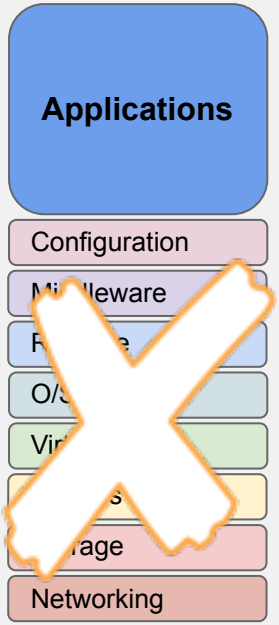


GENERAL DISTRIBUTION

redhat.

# SOLUTION?




**SOMETHING ?**

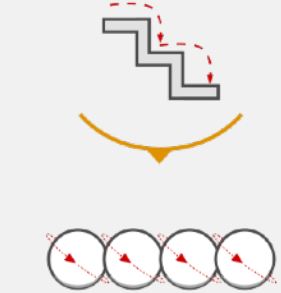
* coming soon

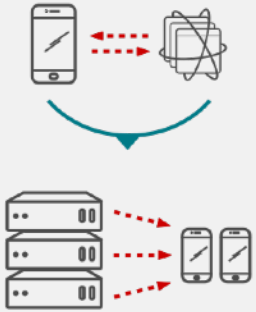


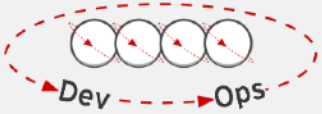
redhat.

# KEY TECHNOLOGY TRENDS

| Development Process | Application Architecture | Deployment & Packaging | Application Infrastructure |
|---|---|---|---|
| Waterfall | Monolithic | Physical Servers | Datacenter |
| Agile | N-Tier | Virtual Servers | Hosted |
| **DevOps** | **Microservices** | **Containers** | **Cloud** |







GENERAL DISTRIBUTION

redhat.

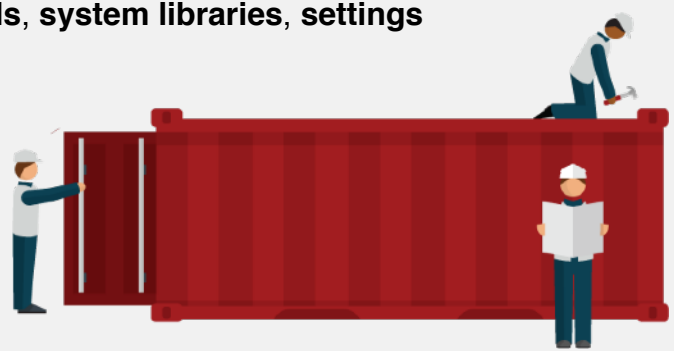
# CONTAINERS

## What is “Container”?

- Easy to deploy and portable across host systems
- Isolates applications on a host operating systems

Adopting a container strategy will allow **applications** to be easily **shared** and **deployed**.

A container image is a **lightweight**, **stand-alone**, executable package of a piece of software that includes everything needed to run it: **code**, **runtime**, **system tools**, **system libraries**, **settings**



Package application & dependencies
Simplify deployment
Speed delivery



GENERAL DISTRIBUTION

redhat.

# MICROSERVICES
## CHARACTERISTICS

1. Componentization
   - Self contained
   - Independently Deployable
   - Independently Upgradable
2. Organized around business capabilities
3. Decentralized data management
4. Smart endpoints and dumb pipes
5. Location Transparency
6. Automation
7. API Focused
8. Decentralized Governance
9. Design for failure

# DEVOPS

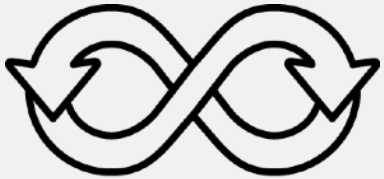
| | |
|---|---|
| Everything as code | Application monitoring |
| Automate everything | Rapid feedback |
| Continuous Integration/Delivery | Rebuild vs. Repair |
| Application is always “releasable” | Delivery pipeline |

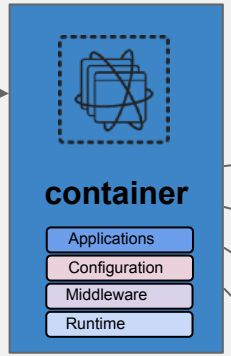


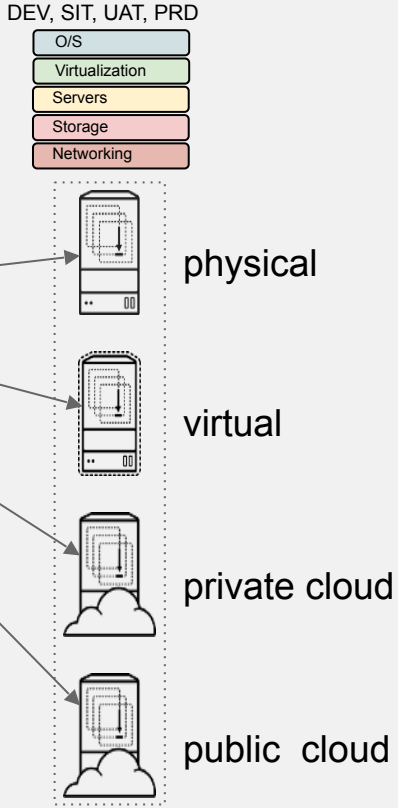
GENERAL DISTRIBUTION

redhat.

# MODERN DELIVERY PIPELINE






- Consistent Infrastructure
- Portable (self contained)
- Independently deployable
- Independently upgradable
- Secure and Isolated
- Automated



GENERAL DISTRIBUTION

redhat.

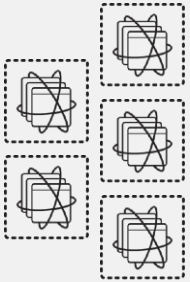
redhat.

# CONTAINERS FOR THE ENTERPRISE

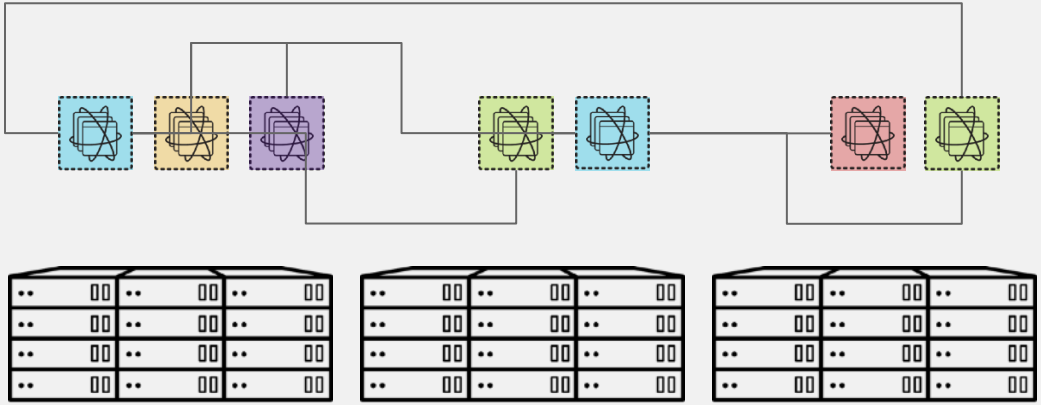
**SECURELY** DELIVERING AND **MAINTAINING** CONTAINER BASED APPS **AT SCALE** CAN BE **COMPLEX**!!!

WHERE TO GO

# WE NEED A PLATFORM



RED HAT OPENSHIFT



Scheduling

Lifecycle and Health

Discovery

Monitoring

Security

Scaling

Persistence

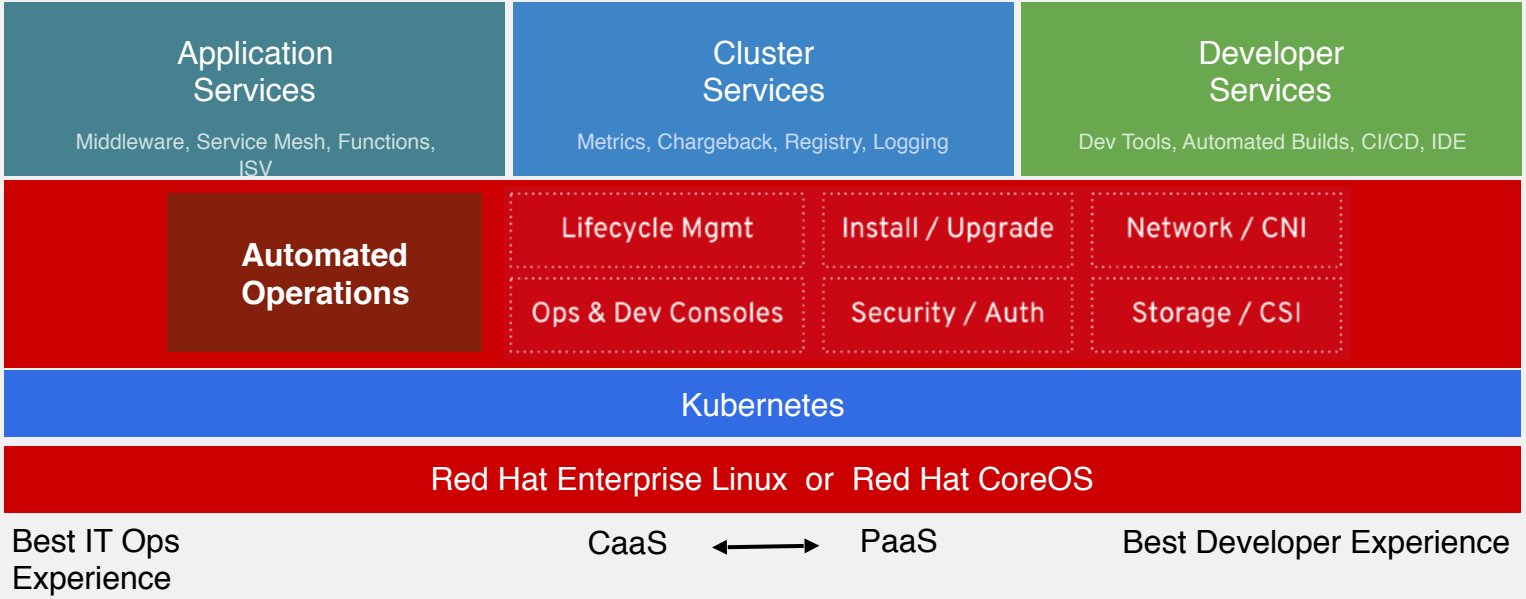
Aggregation



GENERAL DISTRIBUTION

redhat.

# OPENSHIFT CONTAINER PLATFORM




RED HAT OPENSHIFT



redhat

# NOT ENOUGH, THERE IS MORE!

| | |
|---|---|
| Multi-tenancy | Teams and Collaboration |
| Routing & Load Balancing | Quota Management |
| CI/CD Pipelines | Image Build Automation |
| Role-based Authorization | Container Isolation |
| Capacity Management | Vulnerability Scanning |
| Infrastructure Visibility | Chargeback |



GENERAL DISTRIBUTION

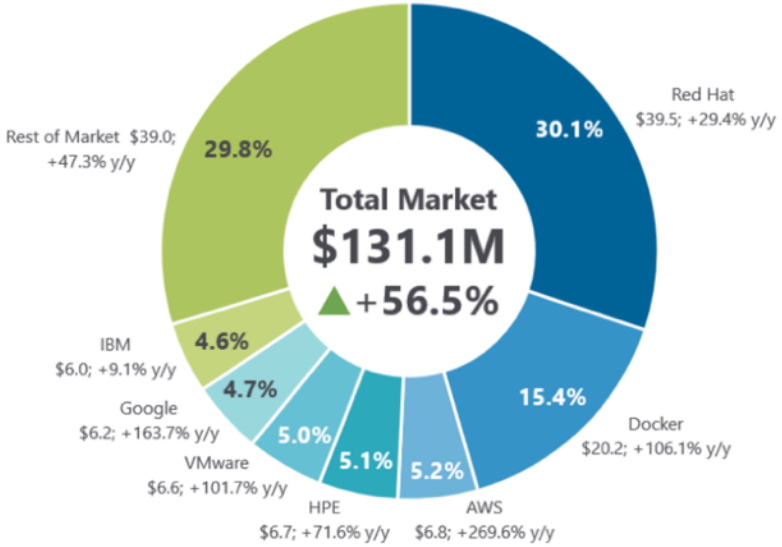
redhat.

# We are the Market Leader !!

IDC MARKET SHARE FIGURE

FIGURE 1

Worldwide Container Infrastructure Software 2017 Share Snapshot

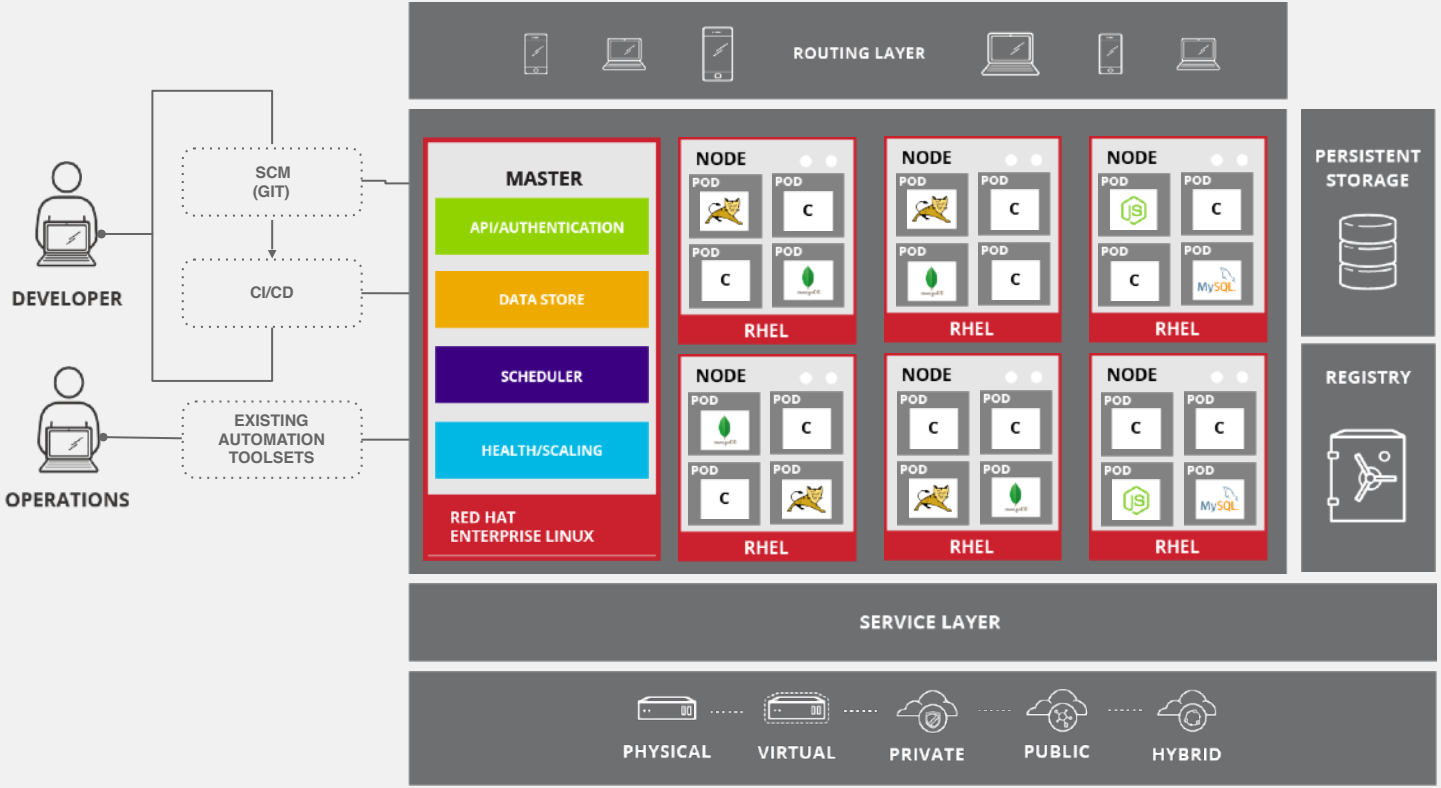



Note: 2017 Share (%), Revenue ($M), and Growth (%)

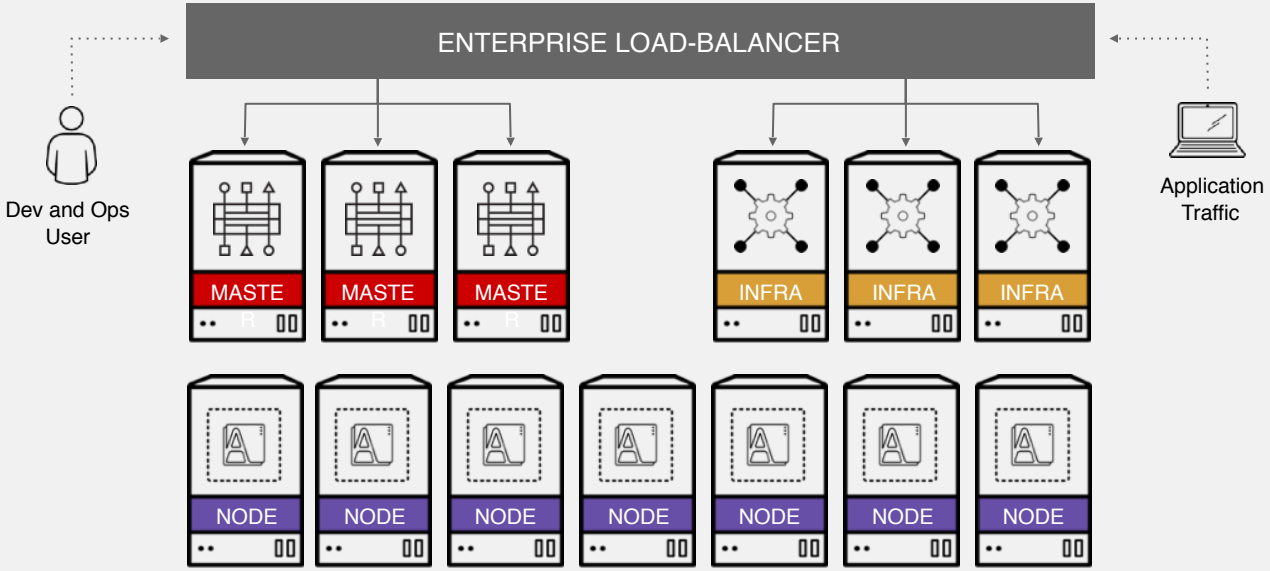


redhat.

# OPENSHIFT ARCHITECTURE



# FULL HIGH-AVAILABILITY ARCHITECTURE



# OPENSHIFT
# WHY

redhat.

# Developer Experience
# PRODUCTIVITY

redhat.

# HOW OPENSHIFT ENABLES DEVELOPER PRODUCTIVITY

BUILD → TEST → DEPLOY

Self-service Provisioning

Consistent environments

Automated build & deploy

CI/CD pipelines

Configuration management

App logs & metrics

CODE ← REVIEW ← MONITOR

| | | |
|---|---|---|
| SPRING & JAVA EE | MICROSERVICES | FUNCTIONS |
| LANGUAGES | DATABASES | APPLICATION SERVICES |
| LINUX | RED HAT OPENSHIFT | WINDOWS* |

* coming soon



redhat.

# TRUE POLYGLOT PLATFORM

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| LANGUAGES | Java | NodeJS | Python | PHP | Perl | Ruby | .NET Core | Third-party Language Runtimes |
| DATABASES | MySQL | PostgreSQL | MongoDB | Redis | | | | Third-party Databases |
| WEB SERVERS | Apache HTTP Server | nginx | Varnish | Phusion Passenger | Tomcat | | | Third-party App Runtimes |
| MIDDLEWARE | Spring Boot | Wildfly Swarm | Vert.x | JBoss Web Server | JBoss EAP | Red Hat A-MQ | Red Hat Fuse | Third-party Middleware |
| | 3SCALE API mgmt | RHDM | RHPAM | Data Virt | RHDG | RH Mobile | RH SSO | Third-party Middleware |

**...and virtually any docker image out there!**

**CrunchyData**
**GitLab**
**Iron.io**
**Couchbase**
**Sonatype**
**EnterpriseDB**
**NuoDB**
**Fujitsu**
and many more



redhat

# APPLICATION SERVICES

## MIDDLEWARE OPTIMIZED FOR Dev and Ops

Infra Monitoring

### RUNTIMES

- RED HAT OPENSHIFT Application Runtimes
- RED HAT DATA GRID
- OpenJDK
- RED HAT AMQ BROKER

### INTEGRATION

- RED HAT FUSE
- RED HAT AMQ
- RED HAT 3SCALE API MANAGEMENT

### AUTOMATION

- RED HAT PROCESS AUTOMATION MANAGER
- RED HAT DECISION MANAGER

App Monitoring

APPLICATION LIFECYCLE MANAGEMENT

CONTAINER ORCHESTRATION AND MANAGEMENT

ENTERPRISE CONTAINER HOST

RED HAT OPENSHIFT

### RED HAT ECOSYSTEM

Red Hat Enterprise Linux, Ansible, Storage, CloudForms, OpenStack



redhat.

# RED HAT APPLICATION RUNTIMES

NON-RESTRICTIVE DEVELOPMENT FOR THE CLOUD

LAUNCH SERVICE

| | | |
|---|---|---|
| JAVA SE<br>OPENJDK | JAVA EE<br>JBOSS EAP | JAVA WEB<br>JBOSS WS |
| SERVERLESS | SPRING<br>SPRING BOOT | JAVASCRIPT<br>NODE.JS |
| MICROPROFILE<br>THORNTAIL | REACTIVE<br>VERT.X | RED HAT OPENSHIFT Application Runtimes |

DISTRIBUTED DATA
DATA GRID

MESSAGING
AMQ BROKER

AUTHENTICATION RED HAT SSO

Optimized for OpenShift / Kubernetes Services with pre-configured Missions and Boosters
Integration with RH Developer, CI/CD tools, Security Services
Available Application Migration Toolkit
Python, Go and .Net also supported by Red Hat (with a different SLA)

Facilitate cloud native app development ON THE CLOUD:

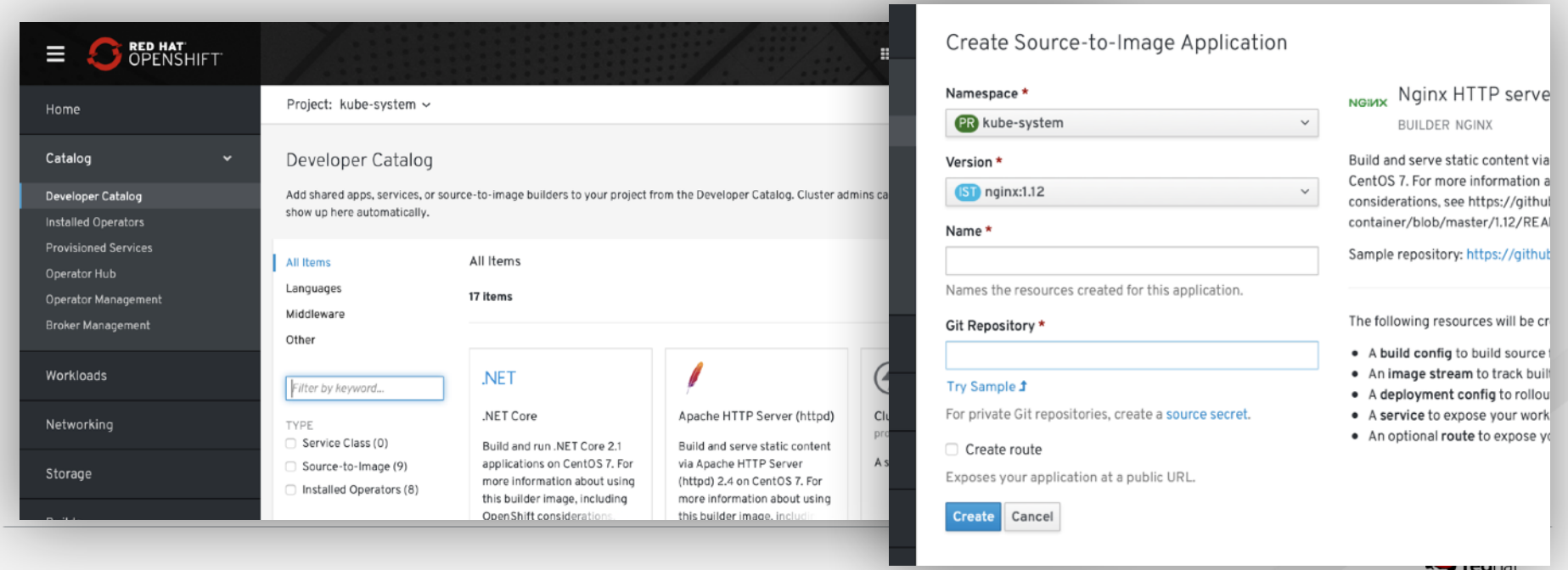
- ✓ Faster getting started
- ✓ Simplify container dev
- ✓ Automate DevOps
- ✓ Standardize tools/processes
- ✓ Fully supported JDK



redhat.

# DEVELOPER CATALOG
# SELF-SERVICE

- Entry point for a developer to access all services available to them
- Merges all capabilities from Operators, Service Catalog, Brokers, and S2I



# CROSS TECHNOLOGY DEVELOPMENT

## CONSISTENT EXPERIENCE

- Consistent provisioning process (**Service Catalog**)
- Consistent Build and Deployment process (**S2I**)
- Dependencies hassle free application delivery (**Container Image**)
- Consistent Configuration Management (**IS, DC, ConfigMap, Secret**)
- Cross technology services resiliency (**self-healing, Autoscaling**)
- Cross technology health check (**Readiness and Lifeness Probs**)
- Cross technology deployment strategies (**Rolling, Canary, ..etc**)
- Cross technology Microservices platform (**Istio**)
- Cross technology Application Logs (**EFK**)
- Cross technology Web Based IDE (**Code Ready Workspaces**)
- Commonly used runtimes and Frameworks (**Spring Boot, Thorntail, Wildfly Swarm, Vert.x, Open Liberty**, ...etc)

# Application Journey - Lifecycle Management

Red Hat DevStudio
Red Hat CodeReady Workspaces
Fabric8

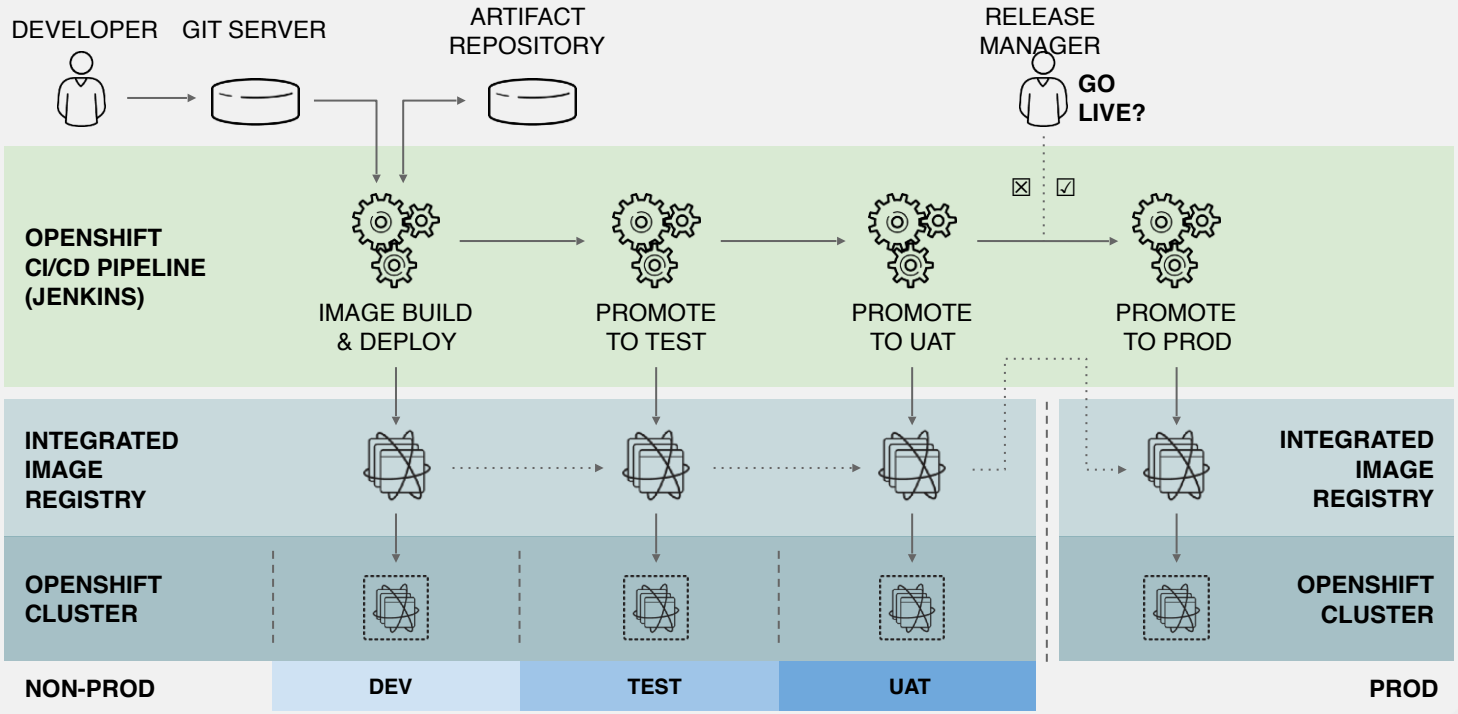
BC, S2I
Built-in Jenkins
Built-in pipeline

Red Hat Base Images
Runtimes
App Services

OCR
Quay

Jenkins
Skopeo

DC, RC, ConfigMap, Secrets, Service, Readiness/Liveness Probes




Deployment Configuration/Plan

Configuration Management

Probes and Health Check

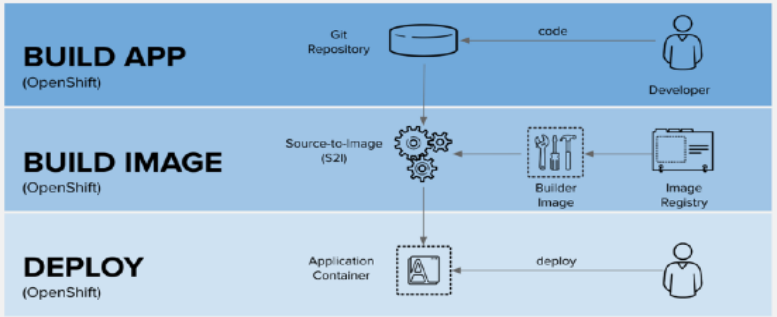
Service (Workload) Discovery



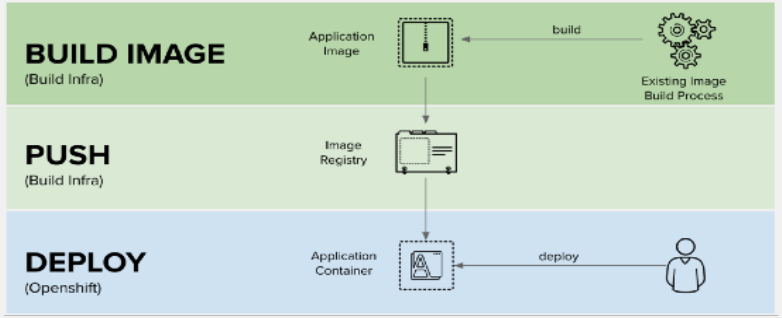
redhat

# Maximum Flexibility

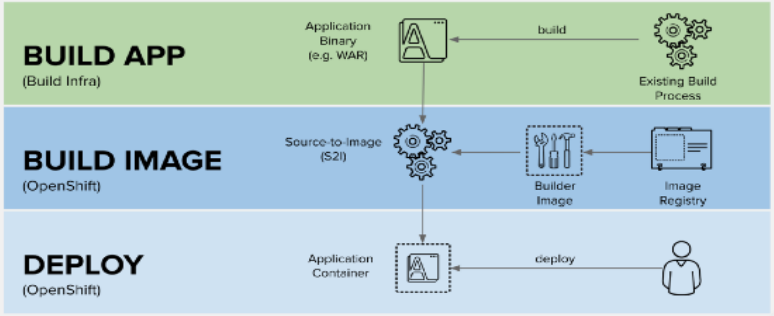
## Build From Source



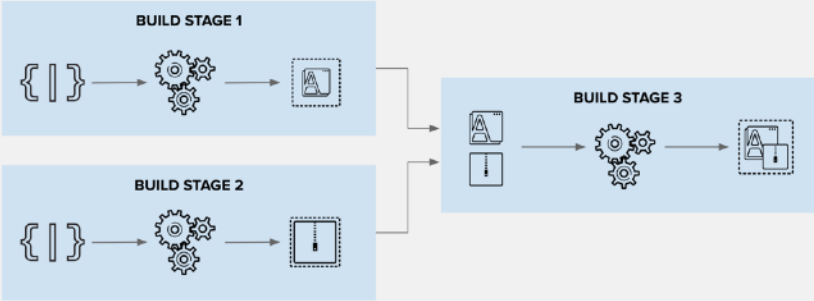

## Build From Image




## Build From Binary




## Multi Stage Build



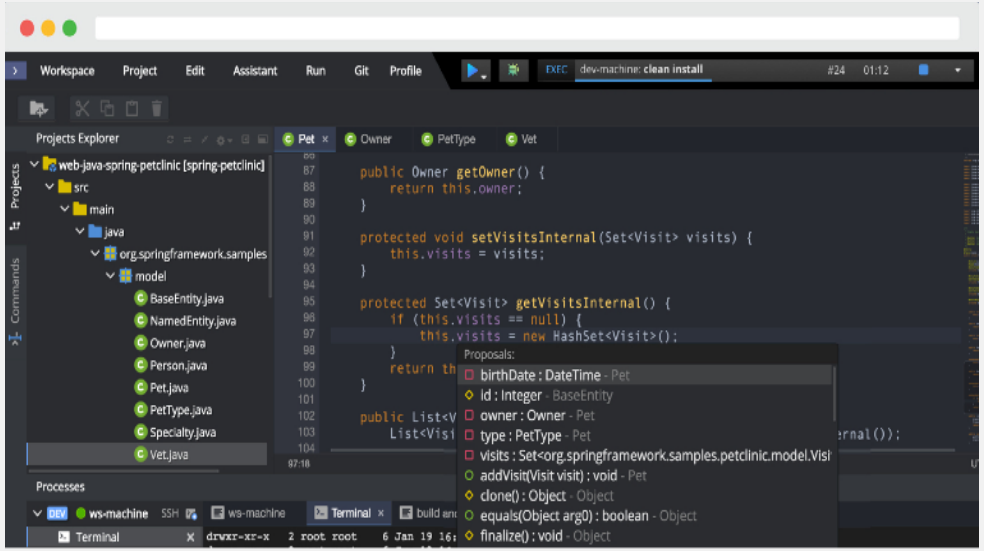

**Still not enough, You can go for S2I Custom Build**



redhat

# CodeReady Workspaces

- Browser-based Web IDE + Dev Environment in pods
- Red Hat supported Eclipse Che
- Bundled with OCP/OSD SKU
- Available on OCP and OSD
- Enabled via an operator
- RHEL 8-based stacks (tools and runtimes)




Eclipse Che

# odo: A CLI FOR DEVELOPERS

odo is a new CLI for OpenShift that is tailored for developer syntax and workflows.

Goal is to make it simple for a developer to create an app, add components (like a database) and expose it without needing to know Kubernetes.

odo is a affectionately called "OpenShift DO!"

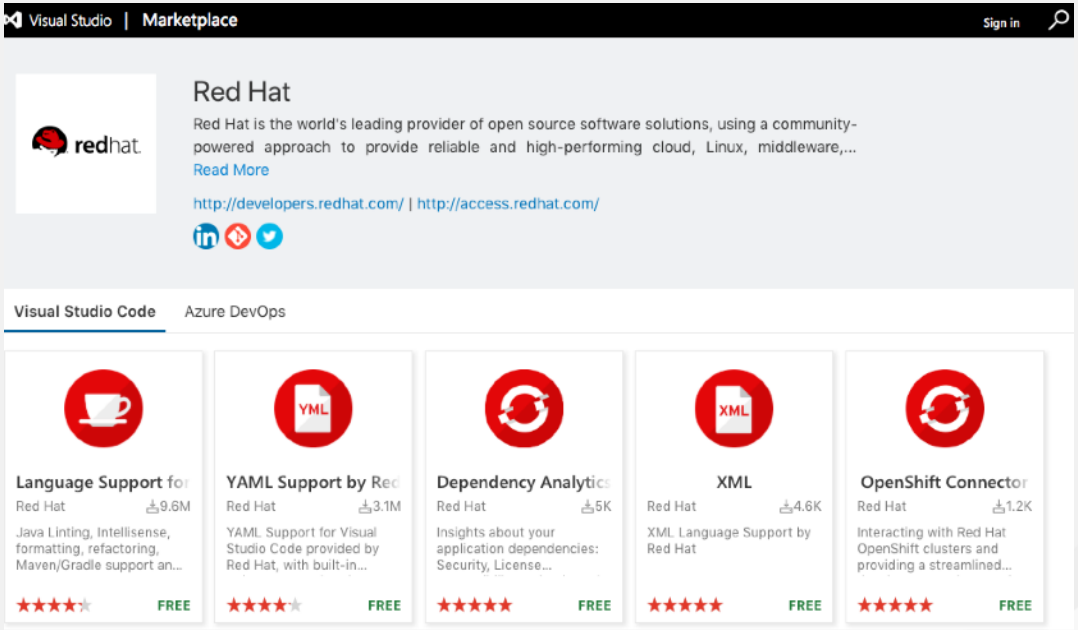
```
> odo create wildfly backend
Component 'backend' was created.
To push source code to the component run 'odo push'

> odo push
Pushing changes to component: backend

> odo storage create backend-store --path /data --size 100M
Added storage backend-store to backend

> odo create php frontend
Component 'frontend' was created.
To push source code to the component run 'odo push'

> odo push
Pushing changes to component: frontend

> odo url create
frontend - http://frontend-myproject.192.168.99.100.nip.io

> odo watch
Waiting for something to change in /Users/tomas/odo/frontend
```

# Visual Studio Code Plugins

Red Hat plugins for VSCode add IDE superpowers for **Java**, **Kubernetes** YAML and **Fuse** XML.

The **OpenShift** plugin allows developers to quickly connect and deploy to OpenShift instances locally or remotely.

**Dependency Analytics** adds license and CVE package alerts.



# OPENSHIFT SERVICE MESH

Jaeger

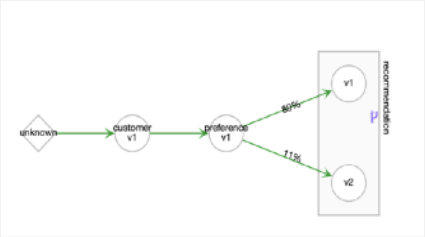
Secure

Istio

Prometheus

Observe

Observe




Connect

Control

redhat.

# IDC White Paper | The Business Value of Red Hat OpenShift

## Application Development and Business Operations Impact

| Number of applications/ major features developed per year | Application development life cycle (weeks) | Annual user productivity gained per 100 application developers | Annual additional revenue per 100 application developers |
|---|---|---|---|
| 36% | 66% | 1,156 | $1.65 |
| More | Faster | Hours | Million |

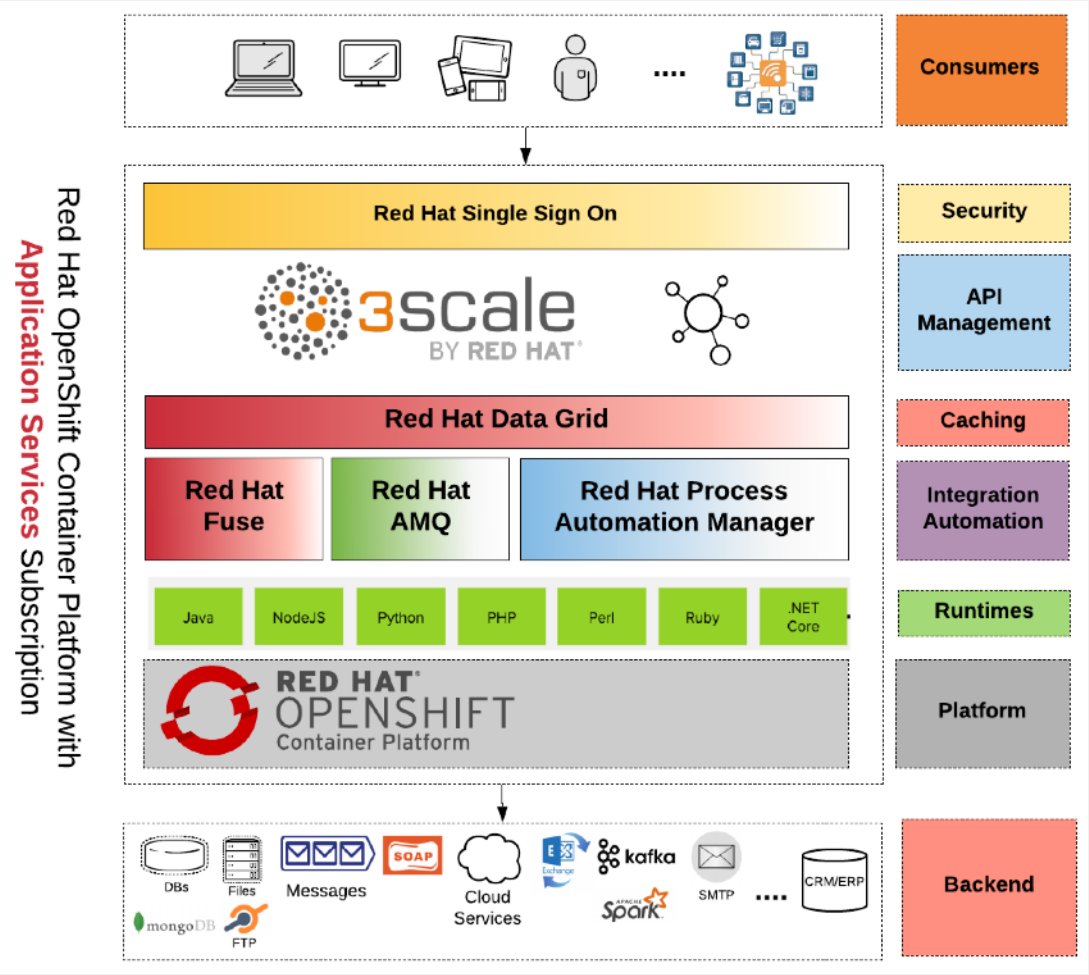
## ROI Summary of Red Hat OpenShift

| 5 Year ROI | Average Annual Benefits per 100 Developers | Payback Period |
|---|---|---|
| 531% | $1.29M | 8 MONTHS |

IDC. www.idc.com | Document #US41845816

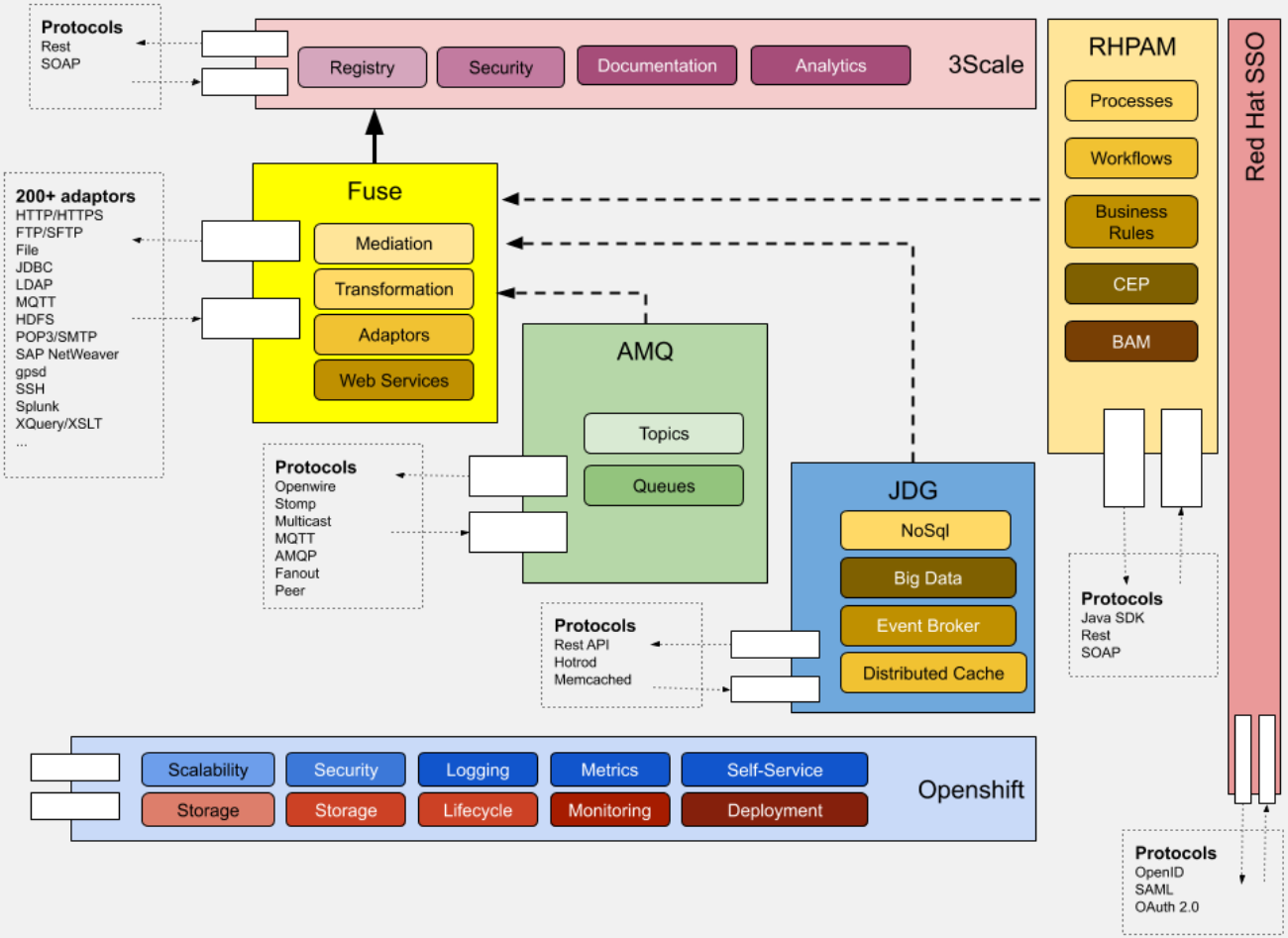
# Architecture View

redhat.


Red Hat OpenShift Container Platform with
Application Services Subscription
....
Consumers
Red Hat Single Sign On
Security
3scale
BY RED HAT
API Management
Red Hat Data Grid
Caching
Red Hat Fuse
Red Hat AMQ
Red Hat Process Automation Manager
Integration Automation
Java
NodeJS
Python
PHP
Perl
Ruby
.NET Core
Runtimes
RED HAT
OPENSHIFT
Container Platform
Platform
DBs
Files
Messages
SOAP
Cloud Services
kafka
Spark
SMTP
....
CRM/ERP
mongoDB
FTP
Backend



redhat

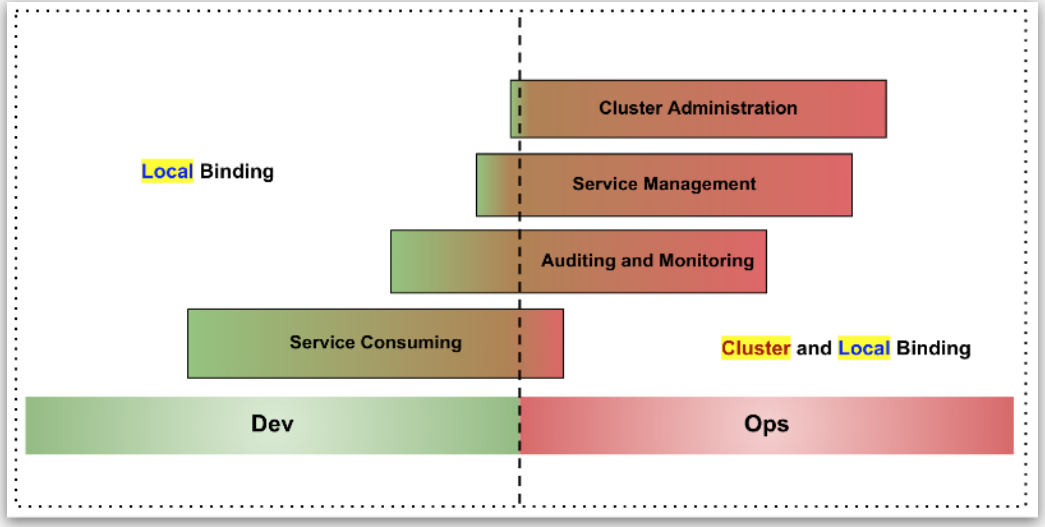

Protocols
Rest
SOAP
Registry
Security
Documentation
Analytics
3Scale
RHPAM
Processes
Workflows
Business Rules
CEP
BAM
Red Hat SSO
200+ adaptors
HTTP/HTTPS
FTP/SFTP
File
JDBC
LDAP
MQTT
HDFS
POP3/SMTP
SAP NetWeaver
gpsd
SSH
Splunk
XQuery/XSLT
...
Fuse
Mediation
Transformation
Adaptors
Web Services
AMQ
Topics
Queues
Protocols
Openwire
Stomp
Multicast
MQTT
AMQP
Fanout
Peer
JDG
NoSql
Big Data
Event Broker
Distributed Cache
Protocols
Rest API
Hotrod
Memcached
Protocols
Java SDK
Rest
SOAP
Scalability
Security
Logging
Metrics
Self-Service
Storage
Storage
Lifecycle
Monitoring
Deployment
Openshift
Protocols
OpenID
SAML
OAuth 2.0

# The Initial Setup
# And
# Day2 Ops

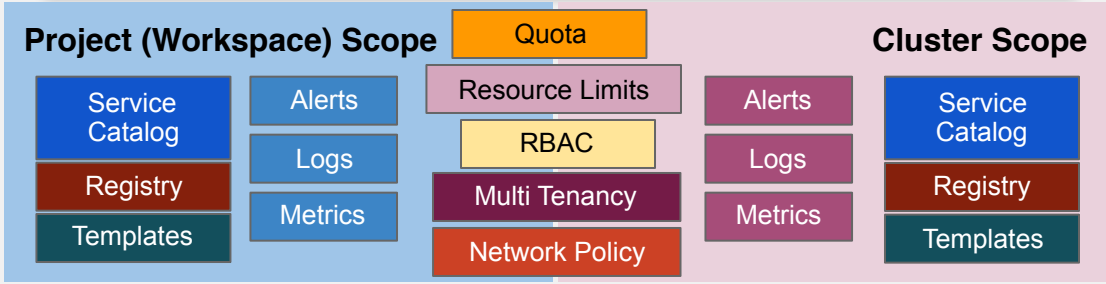
redhat.

# Application Journey - Touch points

- Build/Test/Deploy
- App images
- App templates
- Provisioning
- Team/Collaboration
- Project RBAC




- Cluster setup
- Maintenance/upgrade
- Capacity management
- Cluster resiliency
- Infrastructure
- Tenants
- Network policy
- RBAC






redhat

# AUTOMATED SETUP - Ansible

## PREREQUISITES

- Hosts
- Network
- Firewall
- Proxy
- Storage
- LB
- DNS

## ONE INVENTORY FILE

- [OSEv3:children]
- [OSEv3:vars]
- [masters]
- [etcd]
- [lb]
- [glusterfs]
- [nodes]
- [bastion]

masters
nodes
etcd
lb
bastion

# Registry Configuration
#OCS App Storage
#Proxy
# Metrics deployment
# Logging deployment
# Service brokers
# Prometheus

List of hosts
Labels

## ONE PREREQUISITE PLAYBOOK

$ **ansible-playbook** \
-i /path/to/inventory \
playbooks/**prerequisites.yml**

## ONE INSTALL PLAYBOOK

$ **ansible-playbook** \
-i /path/to/inventory \
playbooks/**deploy_cluster.yml**

Still Complex!

RED HAT ANSIBLE Tower

# AUTOMATED DAY 2 OPS

ADD NODE PLAYBOOK

$ **ansible-playbook** \
-i /path/to/inventory \
playbooks/openshift-node/**scaleup.yml**

Cluster Monitoring PLAYBOOK

$ **ansible-playbook** \
-i /path/to/inventory \
playbooks/**deploy_cluster.yml**

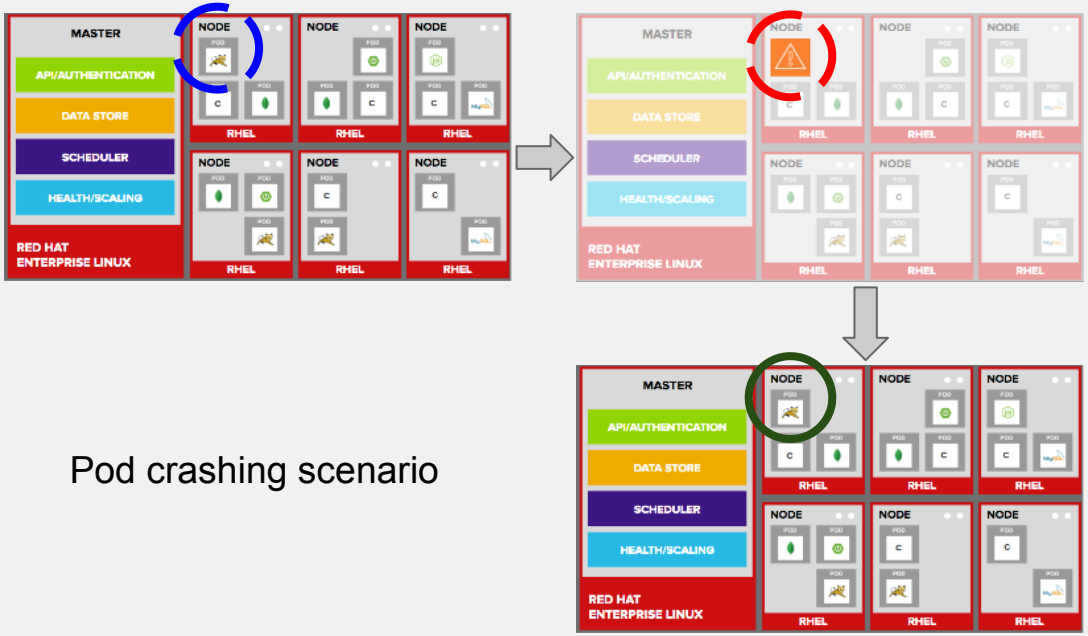
Cluster Metrics PLAYBOOK

$ **ansible-playbook** \
-i /path/to/inventory \
playbooks/**deploy_cluster.yml**

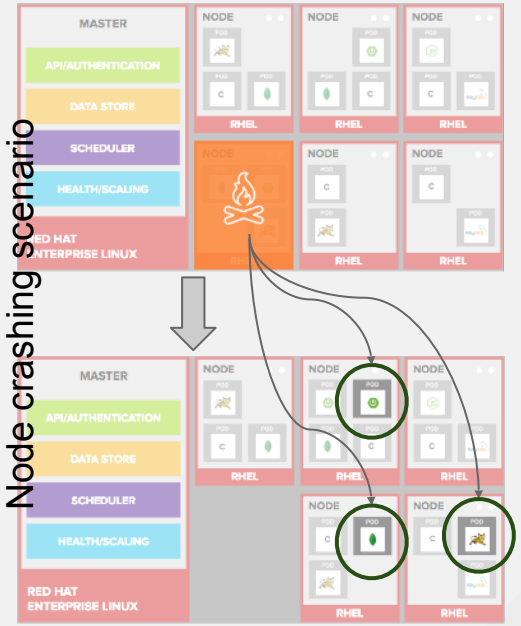
**… and more**

# SELF-HEALING

## PODS AND NODES




Pod crashing scenario




Node crashing scenario

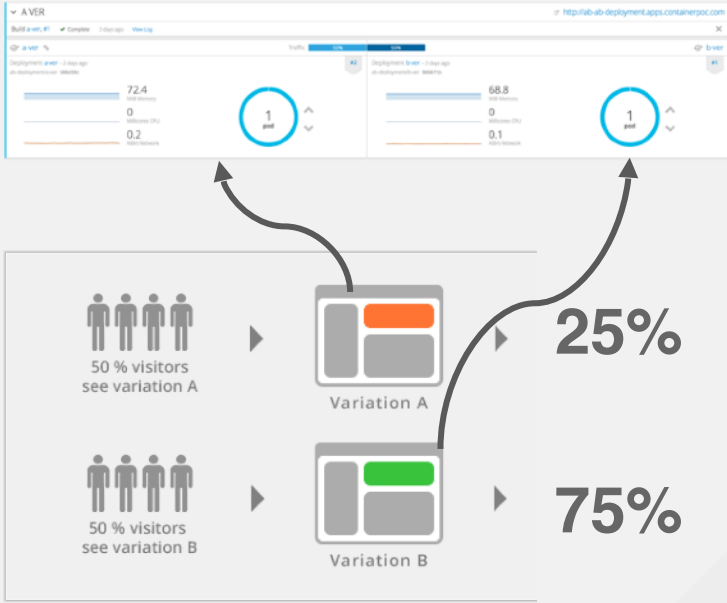


redhat

# DEPLOYMENT STRATEGIES

## Rolling, Recreate, Blue/Green, A/B

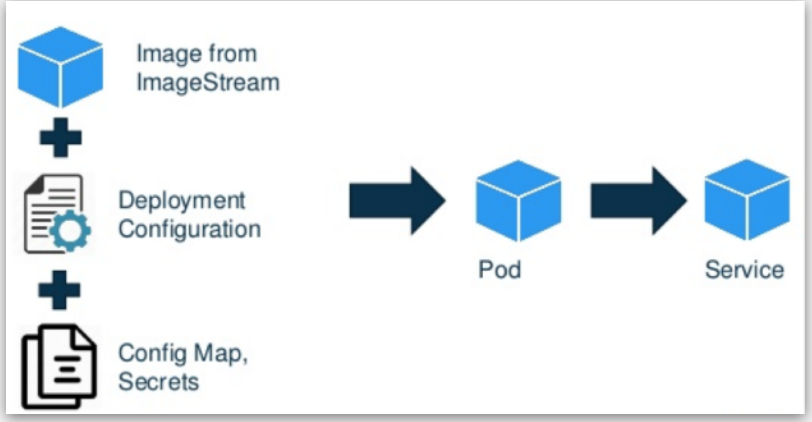
- Support for zero downtime deployment
- Canary deployment
  - A/B
  - Blue/Green
- Custom deployment strategy
  - Pre deployment hooks
  - Post deployment hooks
- Deployment triggers



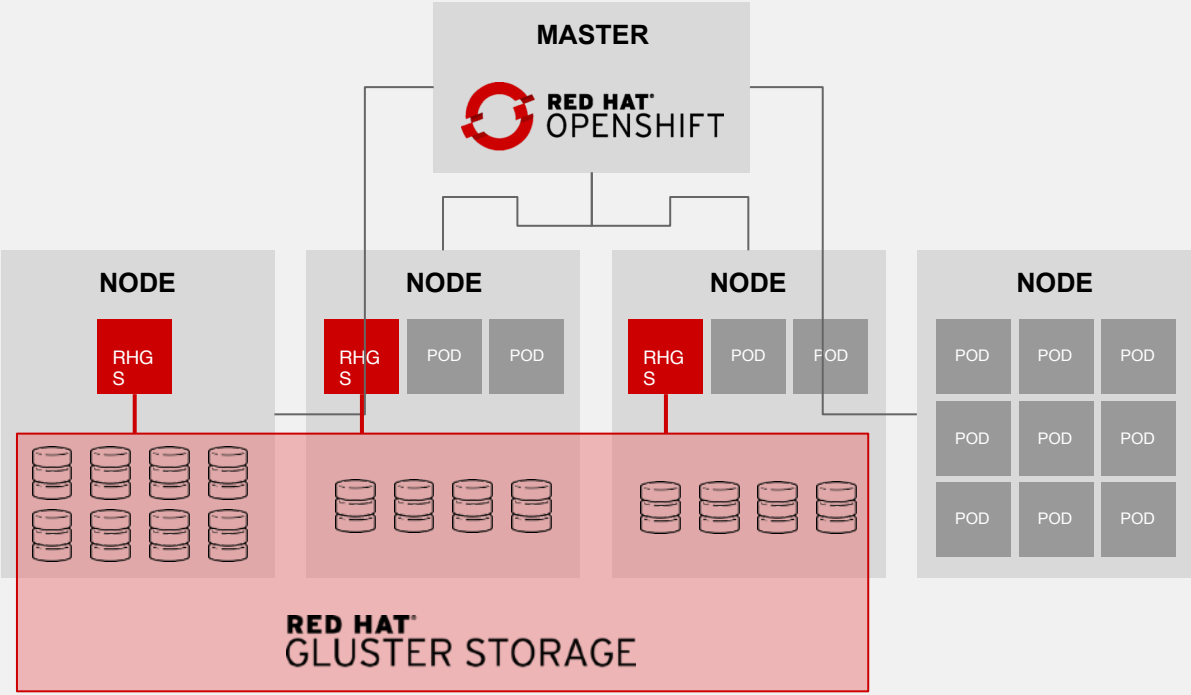
# CONFIGURATION MANAGEMENT

## ConfigMap, Secret, Environment Parameters, Image Promotion

- The ConfigMap object provides mechanisms to inject containers with configuration data.
- The ConfigMap API object holds key-value pairs of configuration data that can be consumed in pods.
- Configuration data can be consumed in pods in a variety of ways. A ConfigMap can be used to:
  - Populate the value of environment variables.
  - Set command-line arguments in a container.
  - Populate configuration files in a volume.
- Secrets provides data encryption for data at rest and in motion



# CONTAINER-NATIVE STORAGE



# OPENSHIFT 4 THEMES

**Coming This Month**

## AUTOMATED OPERATIONS

Integrate CoreOS technology for a better install, config and upgrade experience.
Bring over-the-air upgrades to the platform.

## IMMUTABLE INFRASTRUCTURE

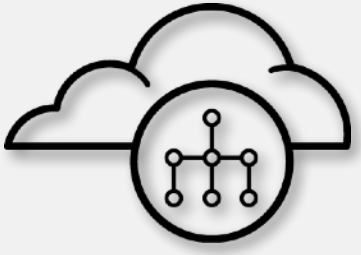
Introduce Red Hat CoreOS as an immutable OS option.
Drive “infrastructure as code” throughout the platform.

## OPERATOR HUB

Provide a curated list of Kubernetes-native workloads certified for OpenShift. Easily installed from on the Console.

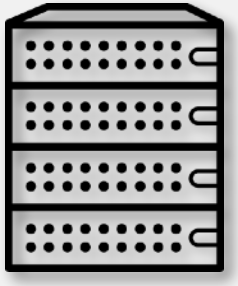
# The Installation Experience

**Kubernetes Operator** (No Ansible Installer anymore)



OPENSHIFT by Red Hat + redhat COREOS

**Installer Provisioned Infrastructure (IPI)**
Simplified opinionated
“Best Practices”
single cluster provisioning



OPENSHIFT by Red Hat + redhat COREOS + RED HAT ENTERPRISE LINUX

**User Provisioned Infrastructure (UPI)**
Customer managed
resources & infrastructure
single cluster provisioning



redhat

# Installer Provisioned Infrastructure (IPI)

Day 1: openshift-install - Day 2: Operators

User managed

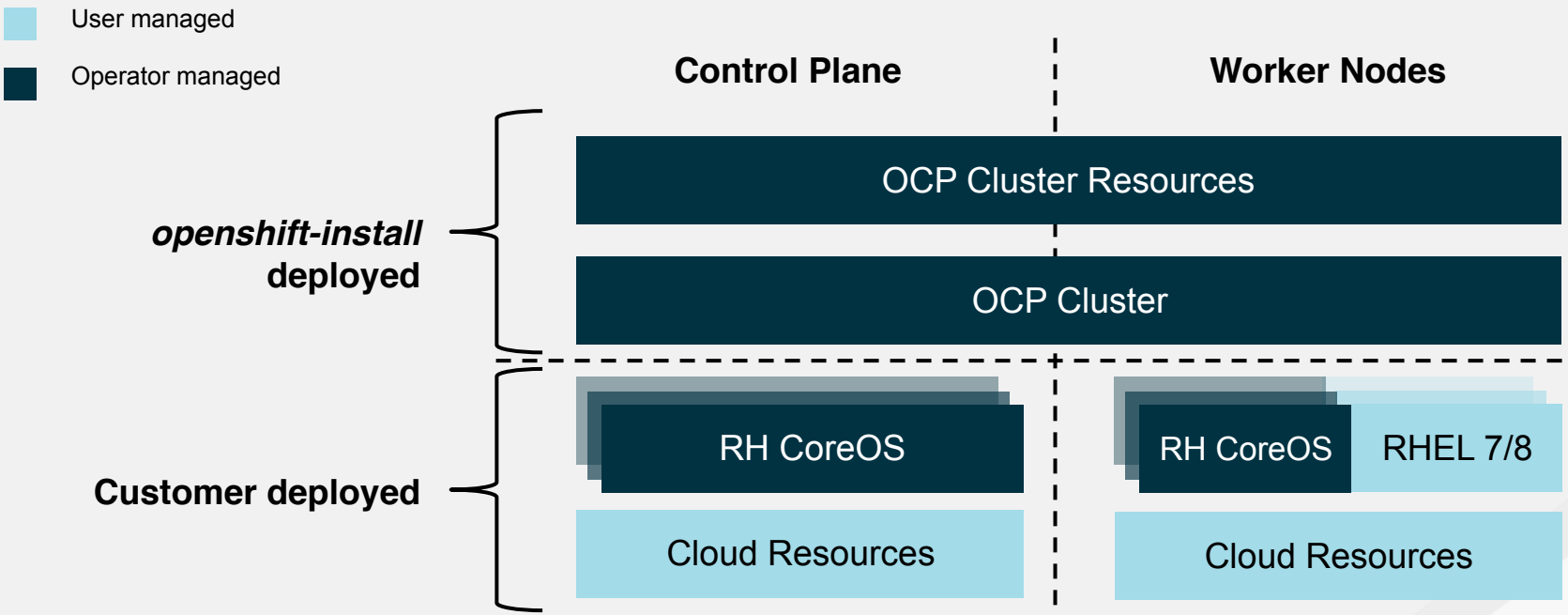
Operator managed

***openshift-install* deployed**

| Control Plane | Worker Nodes |
|---|---|
| OCP Cluster Resources | |
| OCP Cluster | |
| RH CoreOS | RH CoreOS |
| Cloud Resources | Cloud Resources |

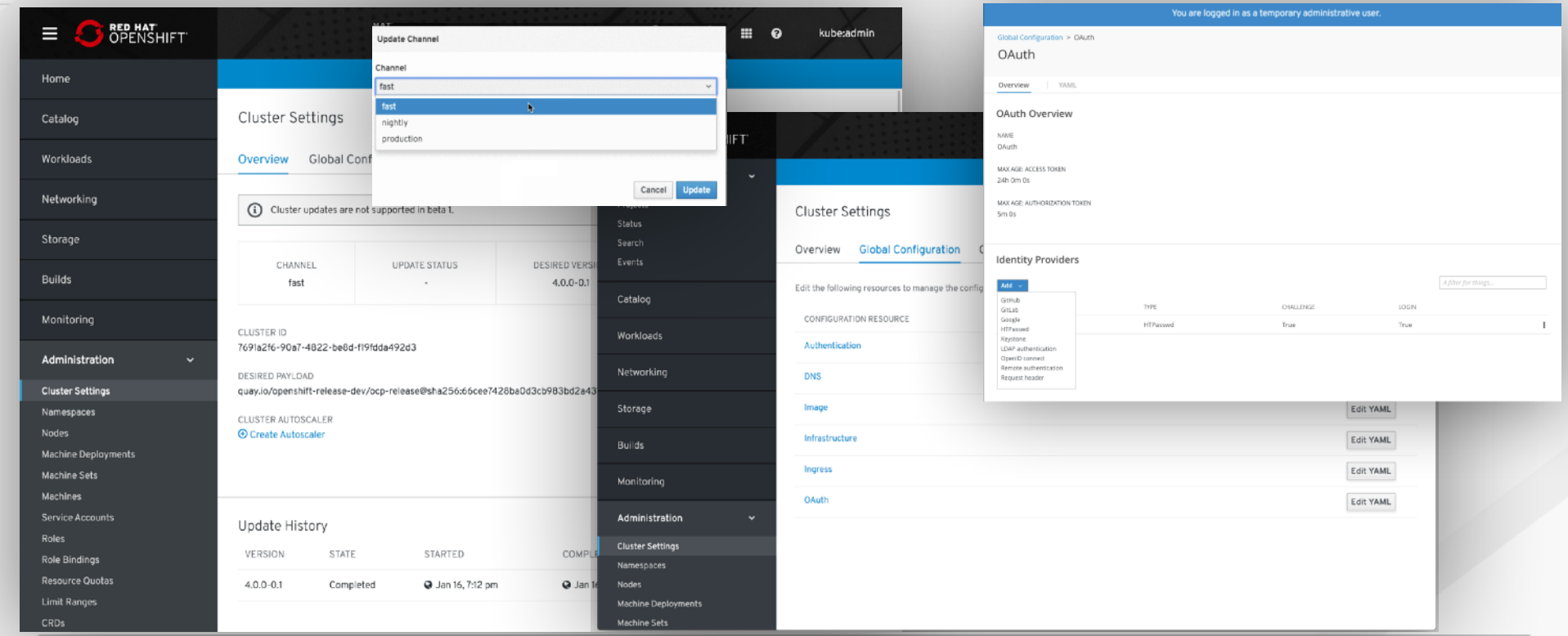
# User Provisioned Infrastructure (UPI)

Day 1: openshift-install - Day 2: Operators + Customer Managed Nodes & Infra



# DAY 2 OPS

## Cluster Management

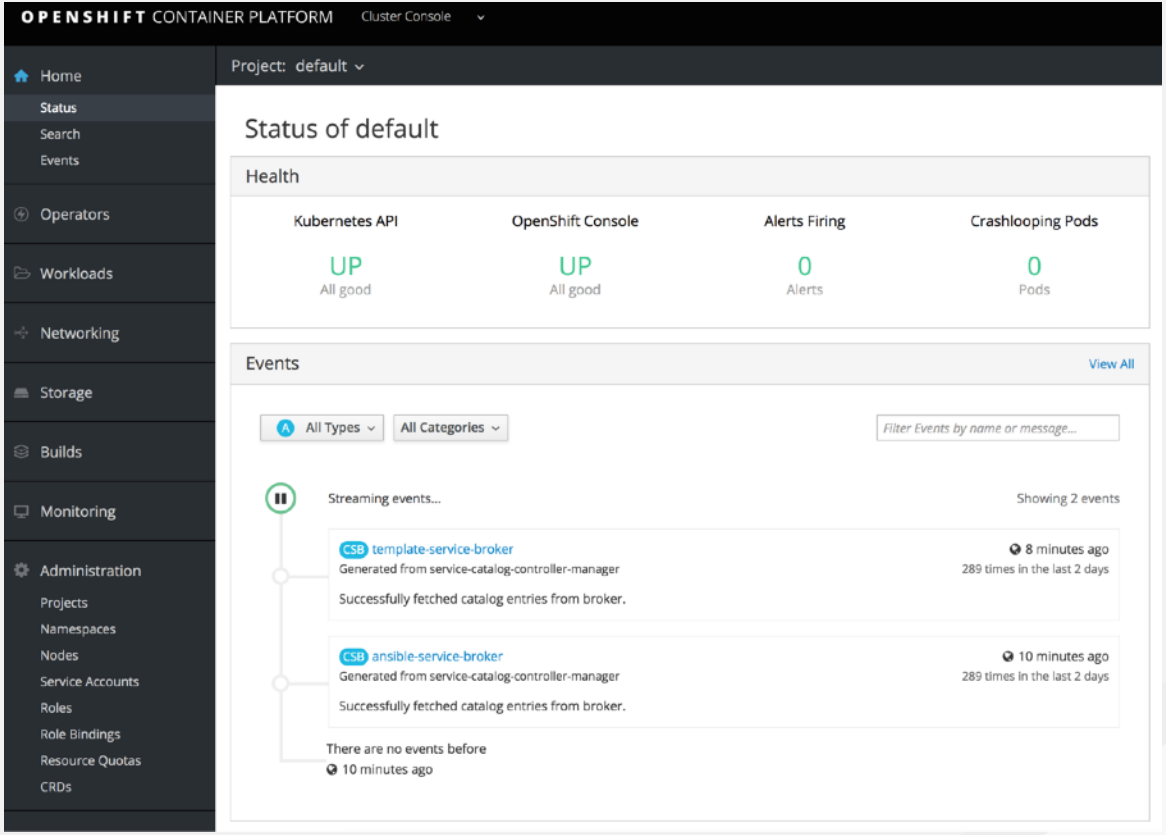



redhat.

# OPENSHIFT CLUSTER CONSOLE

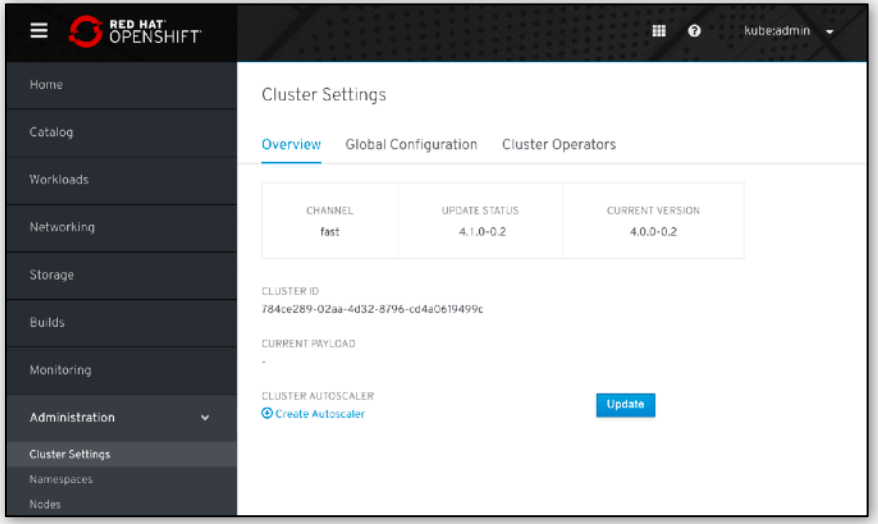
Cluster Management

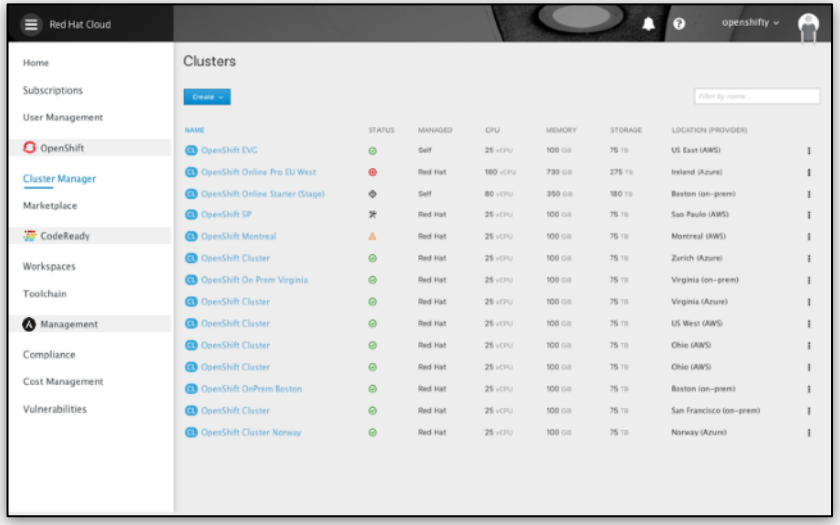
- Overall Health
- Utilization
- Cluster Nodes
- Storage
- Networking
- Projects
- Workloads
- Built-in Dashboards
- Custom Dashboards
- RBAC
- More...



# DAY 2 OPS

## Over-the-air Updates

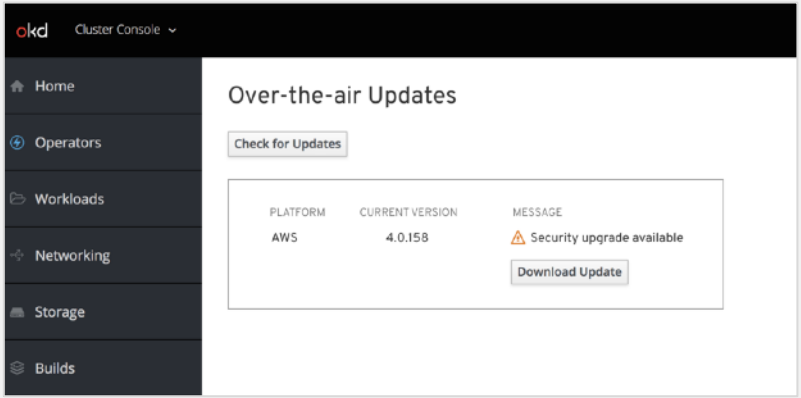



- "Over-the-air" updates can be performed from either OpenShift Cluster Console: "Administration→Cluster Settings" menu or Red Hat Cloud web interface at [https://cloud.openshift.com](https://cloud.openshift.com)
- Updates images are comprised of top level controller manifests, roles, and other resources necessary to update a cluster to a particular version
  - Bundled as a container image to avoid the need for a separate content delivery mechanism

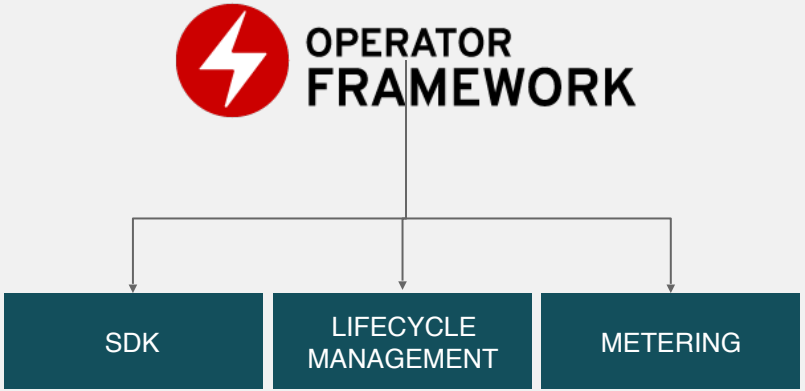
# OVER-THE-AIR UPDATES

- OpenShift retrieves list of available updates
- Admin selects the target version
- OpenShift is updated over the air
- Auto-update support



# OPERATOR FRAMEWORK

Operators codify operational knowledge and workflows to automate lifecycle management of containerized applications with Kubernetes




**Operators are only targeting the platform, But also all workloads running on top of it!**

# OPERATOR HUB COMMUNITY
# CERTIFIED & SUPPORTED OPENSHIFT OPERATORS

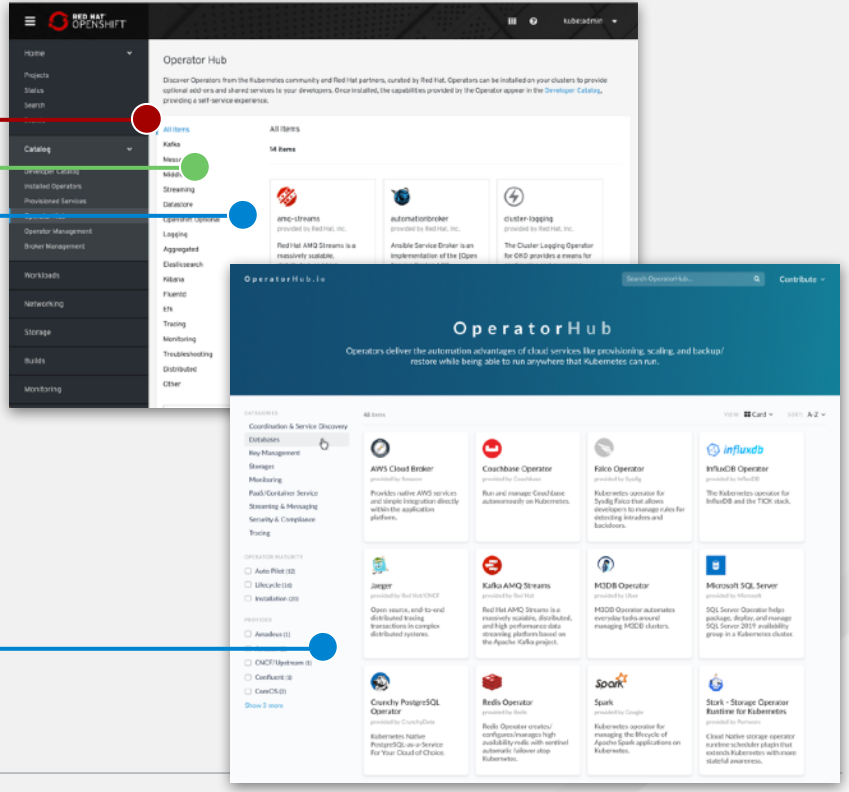
- Accessible to admins only
- Discovery/install of all optional components and apps
- Upstream and downstream content
- ISV partners will support their Operators

TYPES OF OPERATORS

Red Hat Products

ISV Partners

Community



# Open Standards

redhat.

A lightweight, OCI-compliant container runtime

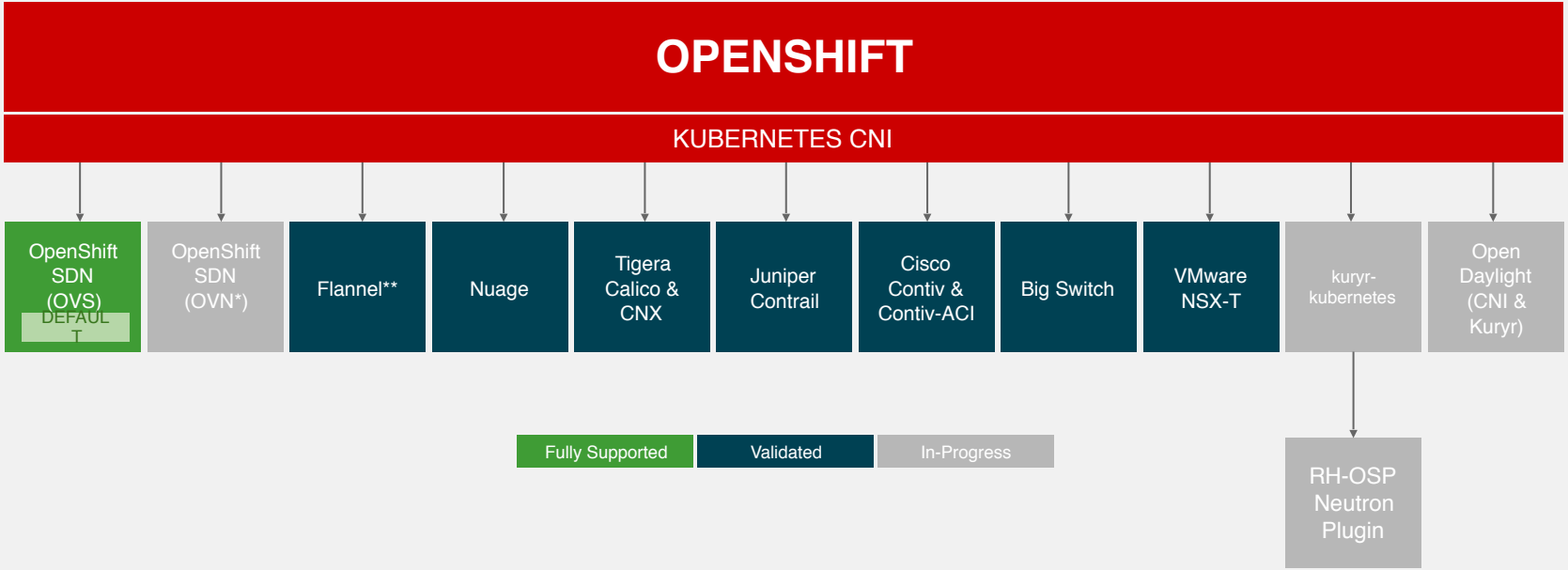
| Minimal and Secure Architecture | Optimized for Kubernetes | Runs any OCI-compliant image (including **docker**) |
|---|---|---|

Optional runtime in OCP 3.10, default OCP 3.11+



redhat.

# OPENSHIFT NETWORK PLUGINS




* Coming as default in OCP 4.1
** Flannel is minimally verified and is supported only and exactly as deployed in the OpenShift on OpenStack reference architecture

# PERSISTENT STORAGE

- Persistent Volume (PV) is tied to a piece of network storage
- Provisioned by an administrator (static or dynamically)
- Allows admins to describe storage and users to request storage
- Assigned to pods based on the requested size, access mode, labels and type

Container Storage Interface (CSI) - Pluggable Storage Interface (Persistent volume plugin system)

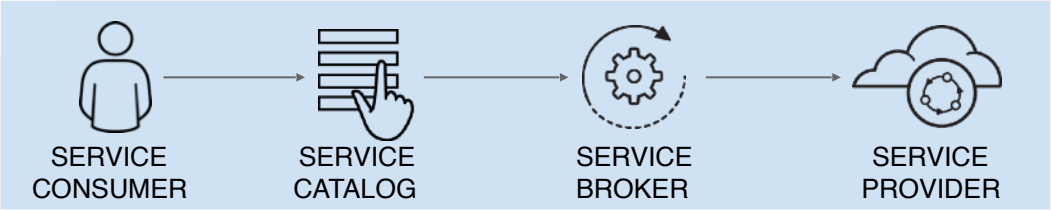
| | | | | | |
|---|---|---|---|---|---|
| NFS | OpenStack Cinder | iSCSI | Azure Disk | AWS EBS | FlexVolume |
| GlusterFS | Ceph RBD | Fiber Channel | Azure File | GCE Persistent Disk | VMWare vSphere VMDK |
| | | NetApp Trident* | Container Storage Interface (CSI)** | | |

* Shipped and supported by NetApp via TSANet
** Tech Preview

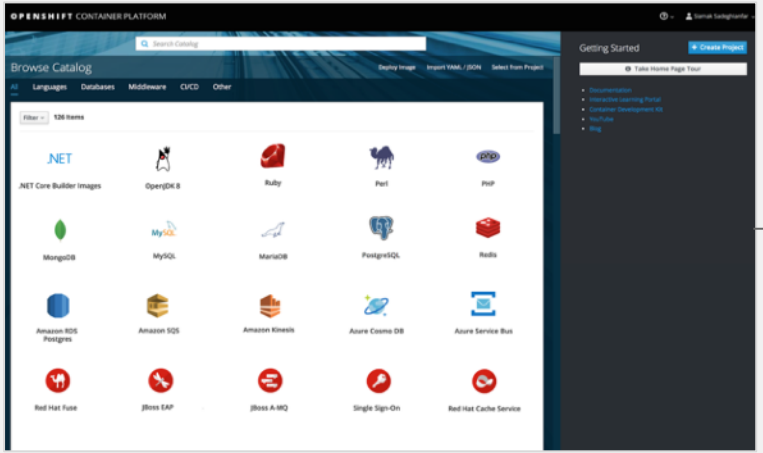


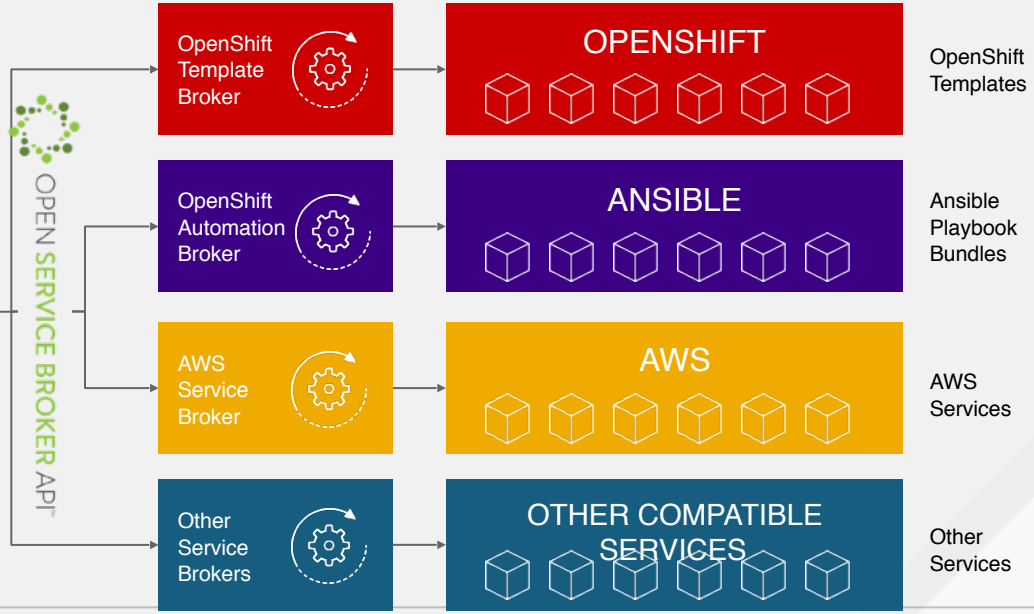
redhat

# SERVICE BROKER -OPENSHIFT SERVICE CATALOG




Automated, Standard and Consistent



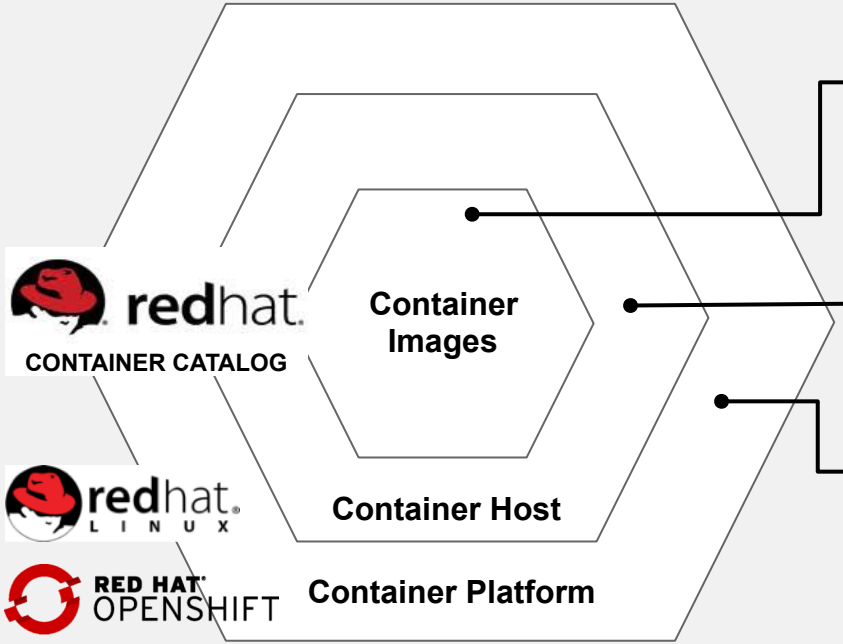






redhat

# SECURITY

# Security - Defense in Depth




Trusted registry, Image scanning, signing, and blueprinting (**reproduction**)

SELinux, Cgroups, Kernel Namespaces, Seccomp

Authentication, RBAC, Quota, Resource Limits, Limit Ranges, Audit, Trusted Registry, Control Image Availability, Image Scanning, Secret Management

# SECURITY IN ACTIONS

- RBAC
- Secret Management
- Quotas - Project Level
- Resource Limit - Pod Level
- Limit Ranges
- Control Image Availability
- Image Scanning
- OS Level (SELinux, Cgroups, Seccomp)
- Network Isolation



redhat.

# PCI DSS Product applicability Guide

**Greater security and compliance**

- **PCI DSS product applicability guide** to help customers understand how the Payment Card Industry Data Security Standard (PCI DSS) apply to the Red Hat OpenShift Container Platform
- Red Hat engaged **Coalfire Systems, Inc.**, a respected Payment Card Industry Qualified Security Assessor (QSA) company, to conduct an independent technical assessment of OpenShift Container Platform running on RHEL and/or Atomic Host. The applicability guide examines the PCI DSS through the eye of a QSA and identifies where the various requirements apply, or do not apply, to the overall solution. Overall, Red Hat and Coalfire came to the conclusion that OpenShift could be configured and deployed in a way that would satisfy the PCI DSS, and we produced a Product Applicability Guide (PAG) to help you understand these opinions.
- Find the guide on https://www.redhat.com/en/resources/openshift-pci-product-applicability-guide-datasheet

# SUPPORT MODEL

redhat.

# EXTENDED ENTERPRISE SUPPORT

- Red Hat provides Enterprise support up to **9 years**
- The 9 years support includes three support phases

| Phase 1 Full Support (5 years) | | | | | Phase 2 Maintenance Support (2 years) | | Phase 3 Extended Life Phase (2 years) | |
|---|---|---|---|---|---|---|---|---|
| Year 1 | Year 2 | Year 3 | Year 4 | Year 5 | Year 6 | Year 7 | Year 8 | Year 9 |

This life cycle encompasses stated time periods for each major version of OpenShift. The life cycle for each major version of OpenShift is split into production phases, each identifying the various levels of maintenance over a period of time from the initial release date. The production phases reference a major version of the product, **though multiple minor (X.Y) versions of the product may be supported at the same time**.

Compare this to PKS support - End of General Support - that lasts for 9 month per release!

Check Kubernetes latest Vulnerability - CVE-2018-1002105 Privilege escalation - and the fact that PKS users have to upgrade/ migrate to specific newer version to get the fix!!!

https://pivotal.io/jp/security/cve-2018-1002105

Check Red Hat support for all Openshift (Kubernetes) releases still covered by enterprise support

https://access.redhat.com/security/vulnerabilities/3716411

# FULL STACK SUPPORT

- Red Hat Support covers from OS up till application services
- Openshift support includes enterprise support for **RHSCL (With no Extra Cost)**
  - **Jenkins**
  - **MariaDB**
  - **MongoDB**
  - **MySQL**
  - **PostgreSQL**
  - **Apache httpd**
  - **nginx**
  - **Varnish Cache**
  - **Redis**
  - **Perl, Python, Ruby, PHP**
  - **Git, Mvn**



redhat.

# RED HAT SERVICES FOR OPENSHIFT ADOPTION

## RED HAT OPEN INNOVATION LABS

### EXPERIMENT
Rapidly build prototypes, do DevOps, and be agile.

### CATALYZE INNOVATION
Bring modern application development back to your team.

### IMMERSE YOUR TEAM
Work side-by-side with experts in a residency-style engagement.

**TO SHOW YOUR TEAMS HOW OPENSHIFT AND MODERN DEVELOPMENT PRACTICES CAN DRIVE INNOVATION: START WITH A 4- TO 12-WEEK LABS RESIDENCY**

## RED HAT CONTAINER ADOPTION PROGRAM

### FRAMEWORK FOR SUCCESSFUL CONTAINER ADOPTION AND IT TRANSFORMATION:

**Mentoring, training, and side-by-side collaboration** to:

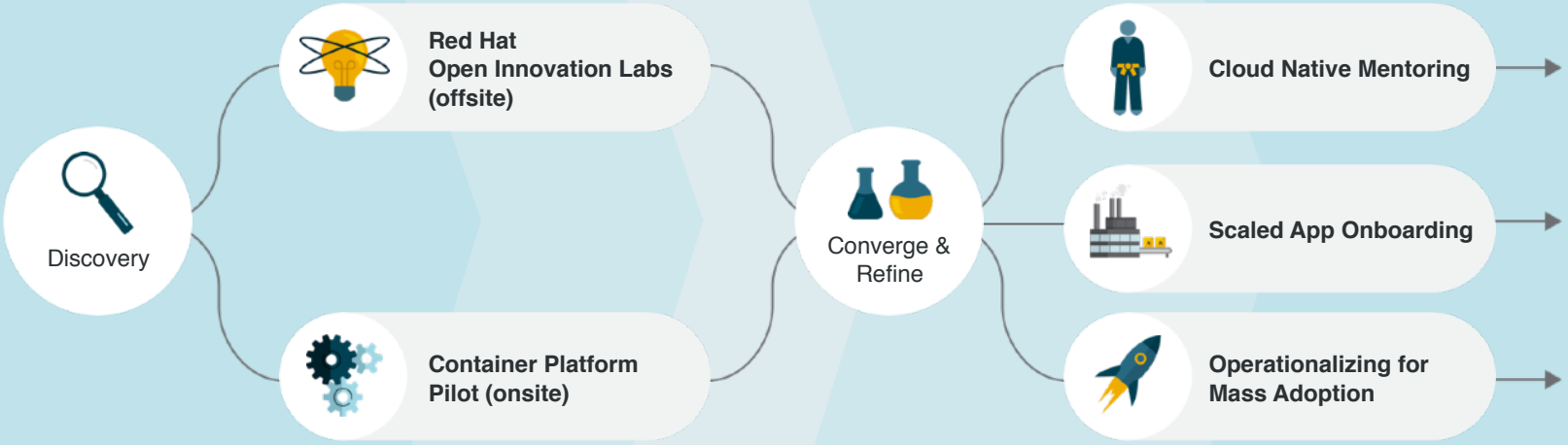
- Create a production platform and team to run it
- Create end-to-end container-driven deployment automation
- Scale application onboarding expertise
- Guide new Kubernetes-native development
- Align business with IT through **included Red Hat Open Innovation Labs**

**TO BEGIN A COMPREHENSIVE PROGRAM (INCLUDING OPEN INNOVATION LABS): START WITH THE 12-WEEK RED HAT CONSULTING CONTAINER PLATFORM PILOT**



redhat.

# CONTAINER ADOPTION PROGRAM




**ESTIMATED PAYBACK TIME: 17 months [1]**
**TOTAL THREE-YEAR BENEFITS NPV: $10.1 M (USD) [1]**

[1] The Total Economic Impact of Red Hat Consulting's Container Adoption Program and Open Innovation Labs, Forrester, 2018.



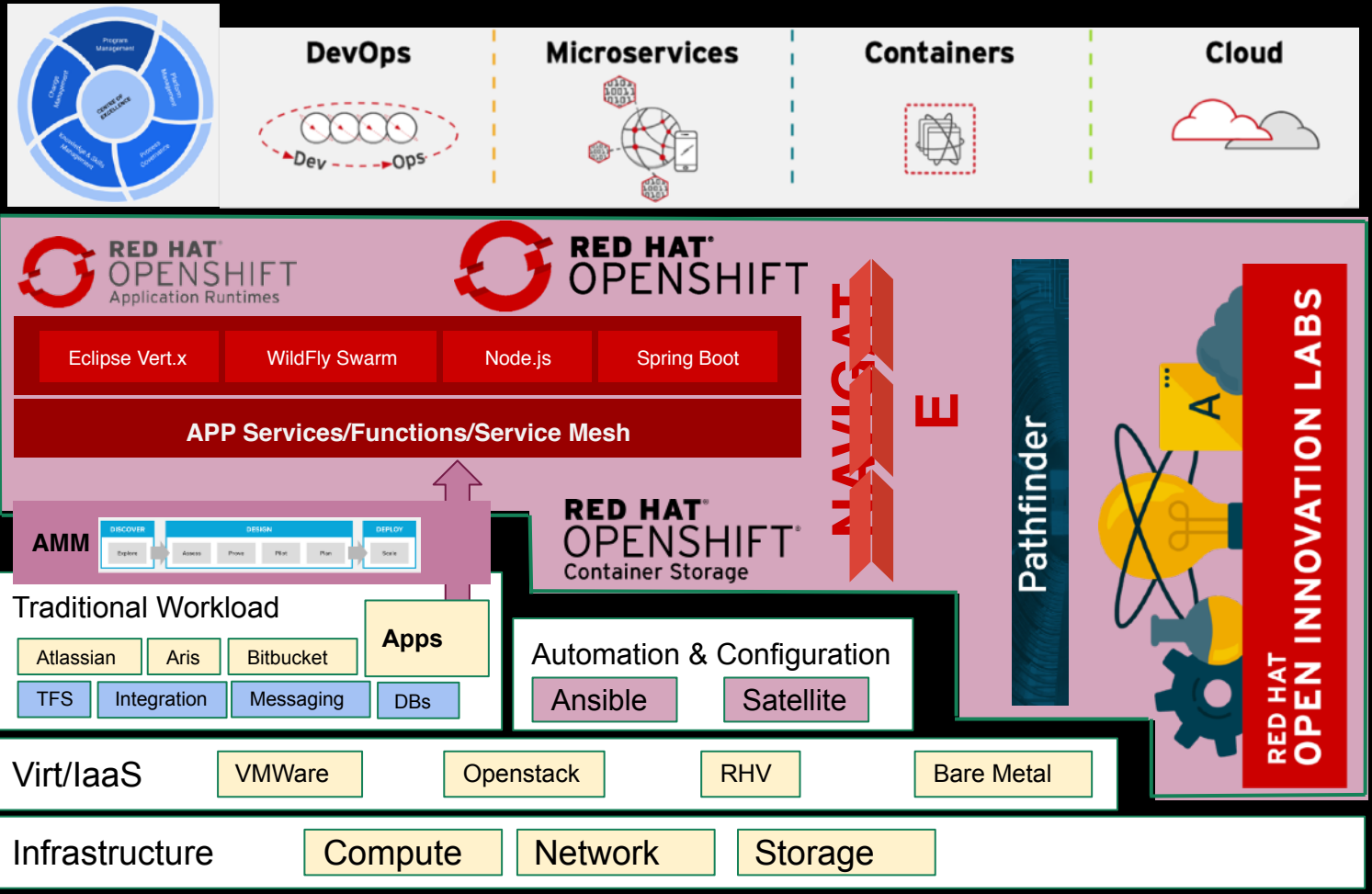
redhat.

# Training Track

| Course Name | Description |
| --- | --- |
| Intro to containers, Kubernetes, and Red Hat OpenShift (DO180) | Build core knowledge and skills in managing containers through hands-on experience with Docker, Kubernetes, and OpenShift Container Platform. |
| Red Hat OpenShift Administration I (DO280) | Red Hat OpenShift Administration I (DO280) prepares the system administrator to install, configure, and manage OpenShift Container Platform instances. |
| Red Hat OpenShift Administration II (DO380) | Red Hat OpenShift Administration II teaches you how to build robust clusters that provide high availability and the ability to run many applications. |
| Red Hat OpenShift Development I: Containerizing Applications (DO288) | Red Hat OpenShift Container Platform provides developers an enterprise-ready solution for developing and deploying containerized software applications. |
| Red Hat OpenShift Development II: Creating Microservices with Red Hat OpenShift Application Runtimes (DO292) | Red Hat OpenShift Development II: Creating Microservices with OpenShift Application Runtimes – Onsite<br>Red Hat OpenShift Development II: Creating Microservices with Red Hat OpenShift Application Runtimes (DO292) teaches how to develop and deploy microservices. |



redhat.


DevOps
Dev
Ops
Microservices
Containers
Cloud
RED HAT OPENSHIFT Application Runtimes
RED HAT OPENSHIFT
Eclipse Vert.x
WildFly Swarm
Node.js
Spring Boot
APP Services/Functions/Service Mesh
AMM
DISCOVER
DESIGN
DEPLOY
RED HAT OPENSHIFT Container Storage
NAVIGATE
Pathfinder
RED HAT OPEN INNOVATION LABS
Traditional Workload
Atlassian
Aris
Bitbucket
Apps
TFS
Integration
Messaging
DBs
Automation & Configuration
Ansible
Satellite
Virt/IaaS
VMWare
Openstack
RHV
Bare Metal
Infrastructure
Compute
Network
Storage

# SIMPLIFIED SUBSCRIPTION MODEL

redhat.

# SIMPLE SUBSCRIPTION MODEL

- A unified Hybrid Cloud Subscription
- Core/vCPU based Subscription Model
- 1 or 3 Year Subscription Term
- Comprehensive (Platform and Platform with App Services)
- OS is included
- Cost Free Master and Infrastructure nodes



redhat.

# SIEM AND APM TOOLS

redhat.

# OPTIONS

- Built-in
  - EFK
  - Prometheus
  - Grafana
- Third Party
  - Splunk (https://splunkbase.splunk.com/app/3836/)
  - AppDynamics APM (https://www.appdynamics.com/solutions/cloud-monitoring/openshift/)
  - Instana APM (https://blog.openshift.com/deploying-instana-apm-natively-openshift-environment/, https://www.instana.com/supported-integrations/openshift-monitoring/)
  - New Relic APM (https://blog.openshift.com/using-new-relic-to-monitor-applications-on-openshift/)



redhat.

# MICROSERVICES
# ISTIO SERVICE MESH

redhat.

# WHAT YOU NEED FOR MICROSERVICES?

- Visibility & Reporting
- Resilience & Fault Tolerance
- Routing & Traffic Control
- Identity & Security
- Policy Enforcement

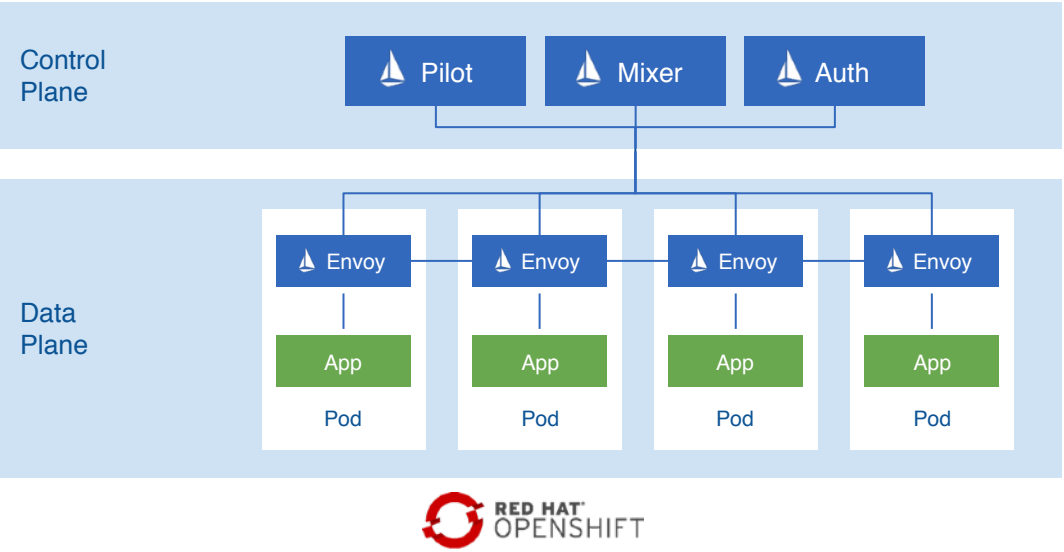
→ Istio



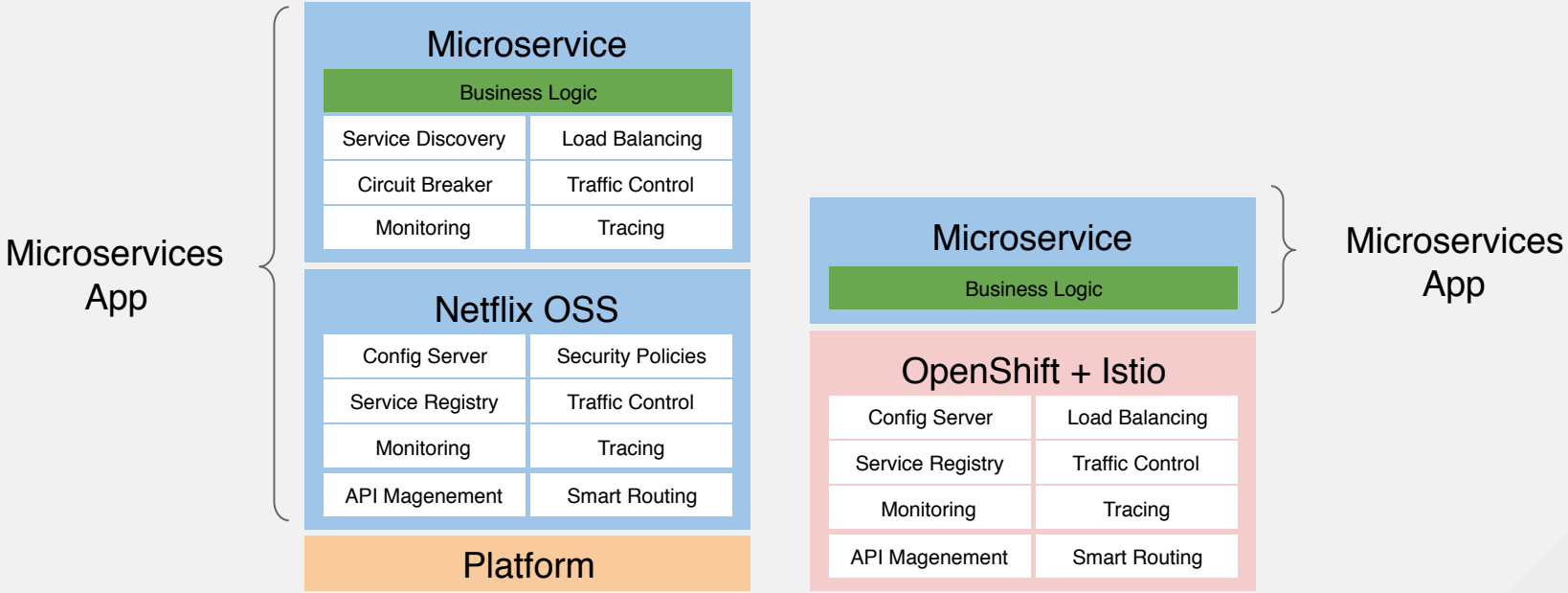
redhat.

# WHAT IS ISTIO?

a service mesh to connect, manage, and secure microservices



# NETFLIX OSS VS ISTIO



# istio explained

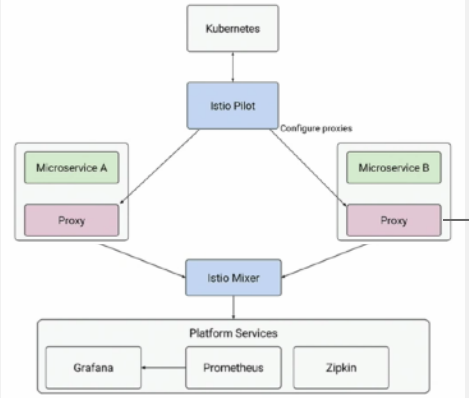
A complete framework for **connecting, securing, managing and monitoring** services

Secure and monitor traffic for microservices **and** legacy services

**Multi-environment and multi-platform**, but Kubernetes first

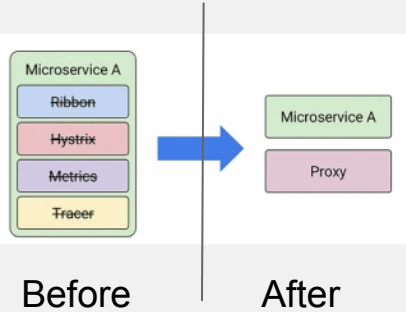
Control Plane

Service to Service Communication
Routing Rules
Retries
Circuit Breaker
Performance Monitoring
Tracing




envoy

Dynamic service discovery
Load balancing
TLS termination
HTTP/2
gRPC proxying
Circuit breakers
Health checks
Traffic split
Fault injection




Before | After



redhat

redhat.

# THANK YOU

plus.google.com/+RedHat

facebook.com/redhatinc

linkedin.com/company/red-hat

twitter.com/RedHatNews

youtube.com/user/RedHatVideos