Red Hat OpenShift

OpenShift Container Platform

Workshop 2019

Name

Red Hat

Topics

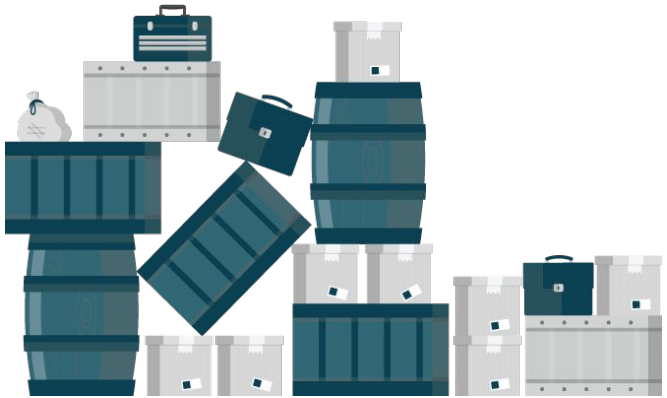
- Modern Challenges
- What's Kubernetes?
- What is OpenShift?
- OpenShift Architecture Review
- Reference Architecture
- Secure Software Supply Chain & Automated CI/CD Pipeline

Modern Challenges

The Problem

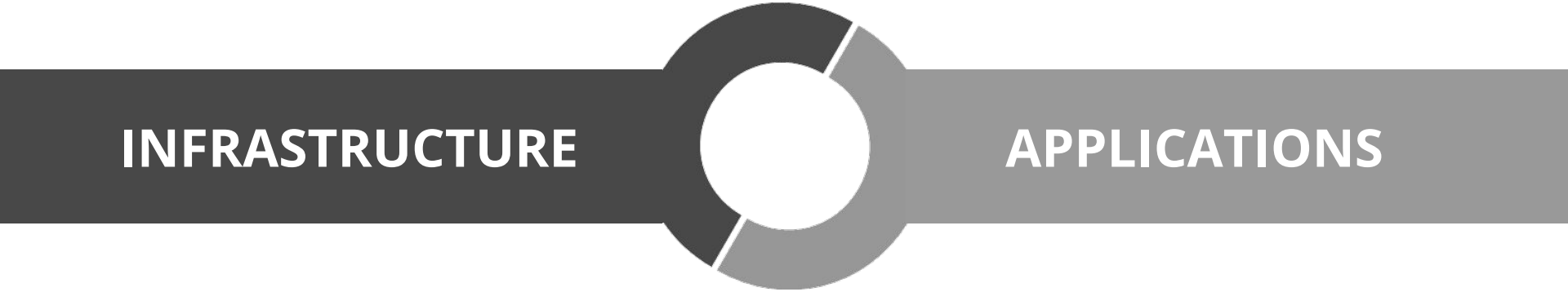
Applications require complicated installation and integration every time they are deployed leading to

- Slow service delivery
- Reduced service quality
- Frequent down times



What Are Containers?

It Depends Who You Ask



- Sandboxed application processes on a shared Linux OS kernel **(multi-tenancy)**
- Simpler, lighter, and denser than virtual machines **(density)**
- Portable across different environments **(portability)**

- Package my application and all of its dependencies **(encapsulation)**
- Deploy to any environment in seconds and enable CI/CD **(ephemerality)**
- Easily access and share containerized components **(standardization)**

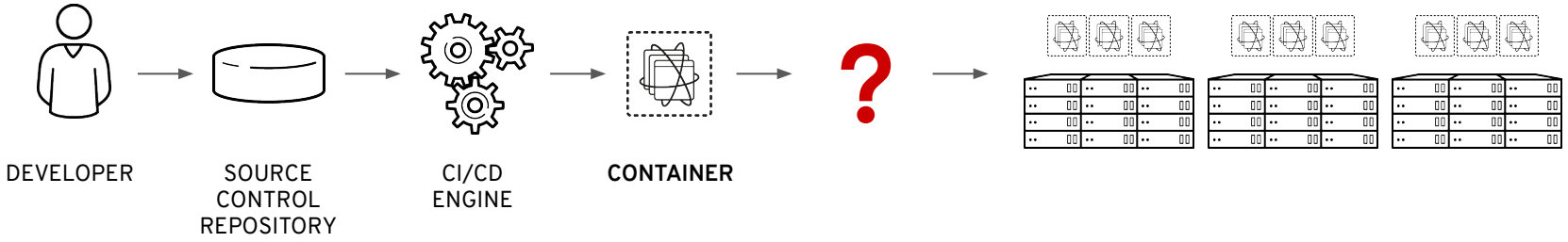
I ❤ Containers

(most developers do)

But Scale Brings Complexity

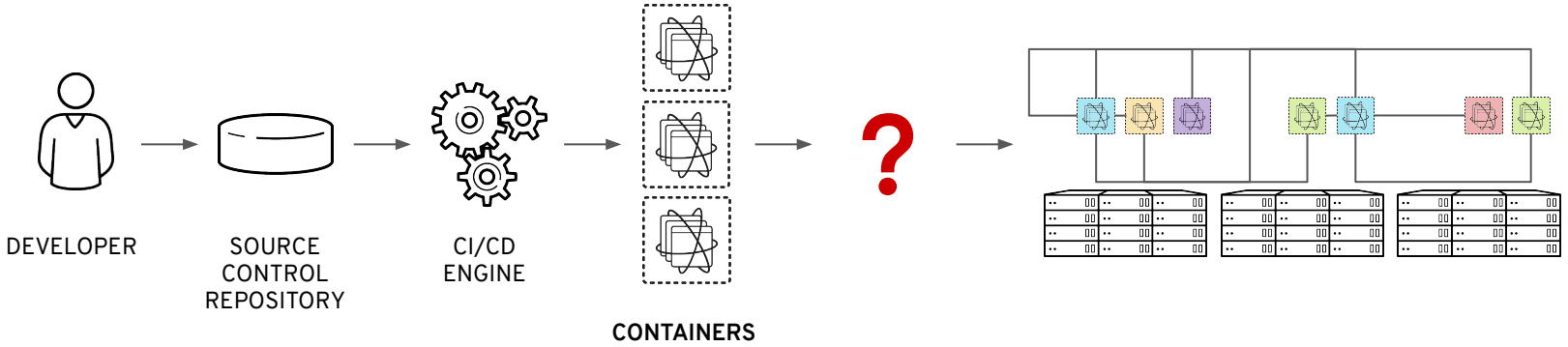
Red Hat

Application Defined In **One** Container



Red Hat

Applications Defined In **Multiple** Containers

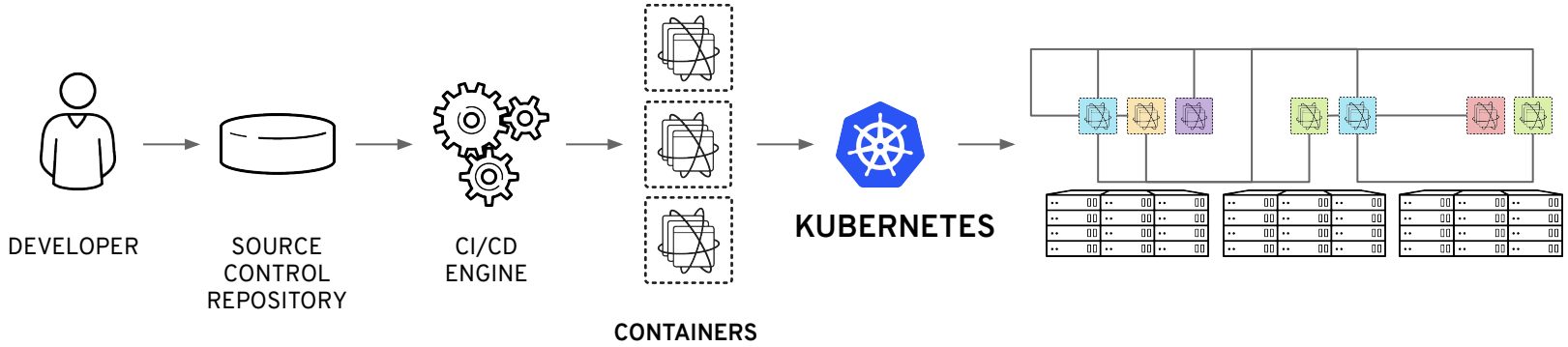


What Do You Need To Operationalize Containers?

Red Hat

What's Kubernetes

What Does The **Community** Have To Offer?



“AN OPEN-SOURCE CONTAINER-ORCHESTRATION SYSTEM FOR AUTOMATING DEPLOYMENT, SCALING AND MANAGEMENT OF CONTAINERIZED APPLICATIONS”*

*WIKIPEDIA JANUARY 2019

Red Hat

KUBERNETES IS THE API OF THE CLOUD

- Robust scheduling
- Auto-scale
- Self-healing
- Stateless and Stateful
- Automated Deployments

By any objective measure, the industry has converged on Kubernetes as the container orchestration engine of choice

Red Hat

Red Hat Has Been A **Kubernetes Leader** Since Day 1

K8S 1.0

K8S 1.0

Microsoft Azure K8S 1.6

amazon web services docker MESOSPHERE K8S 1.8

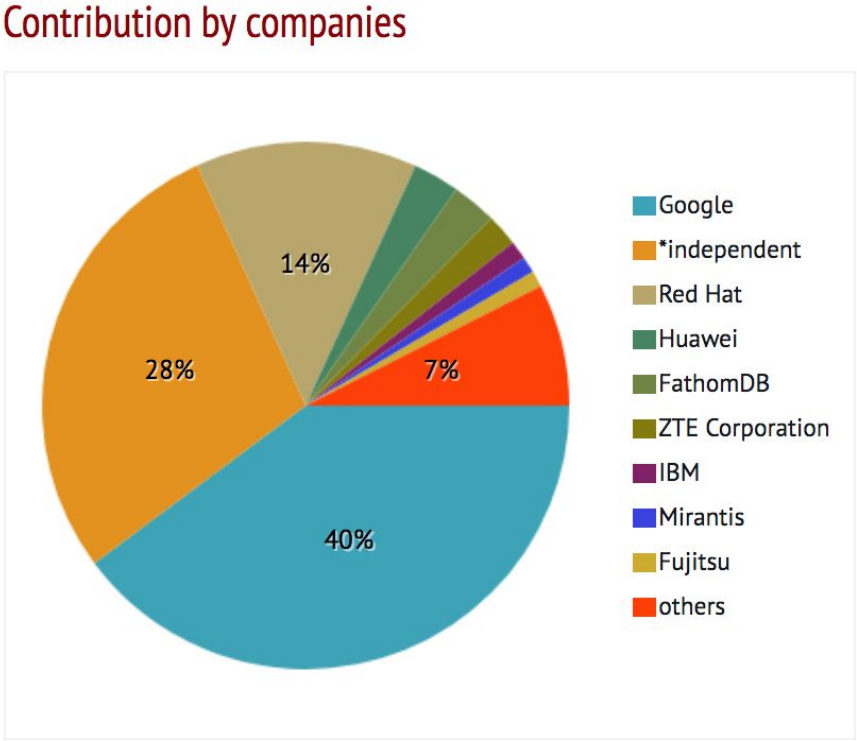
Pivotal vmware K8S 1.9

1.0 1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9 .....

2015 2016 2017 2018

Red Hat

Kubernetes Project Contributions



#1 - Google - 41,649
#2 - Red Hat - 14,410
#6 - IBM - 1,230
#9 - CoreOS - 964*
#10 - Microsoft - 728
#13 - VMware - 433
#15 - Intel - 400
#23 - Cisco - 192
#26 - Pivotal - 141
#41 - Oracle - 36
#56 - Docker - 14
Amazon/AWS - ?

* Most CoreOS commits were done using personal email addresses (Independent)

Red Hat

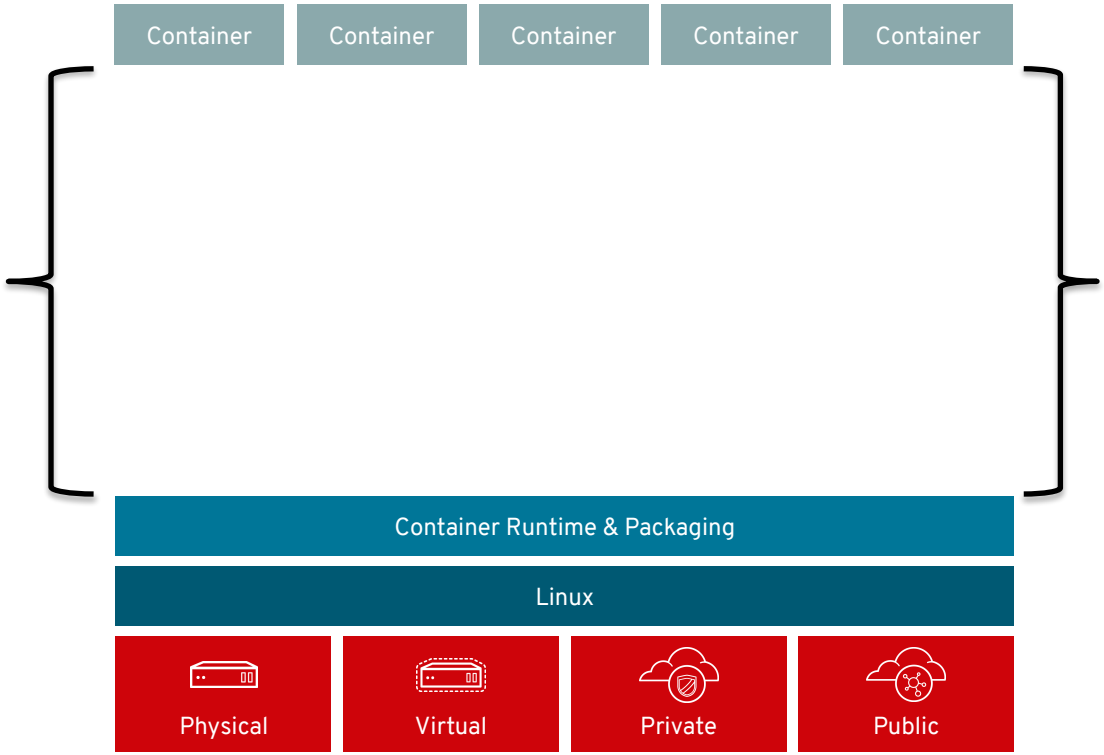
RED HAT CHAIRS ABOUT HALF THE KUBERNETES SPECIAL INTEREST GROUPS

17 of 40 GROUPS

API MACHINERY	AWS	APPS	ARCHITECTURE	AUTH	AUTO SCALING
AZURE	BIG DATA	CLI	CLUSTER LIFECYCLE	CLUSTER OPS	CONTRIBUTOR EXPERIENCE
DOCS	INSTRUMENTATION	MULTI CLUSTER	NETWORK	NODE	ON-PREM
OPENSTACK	PRODUCT MANAGEMENT	RELEASE	SCALABILITY	SCHEDULING	SERVICE CATALOG
STORAGE	TESTING	UI	WINDOWS	APP DEF	CLUSTER API
CONTAINER IDENTITY	KUBEADM ADOPTION	RESOURCE MANAGEMENT	IOT EDGE	POLICY	

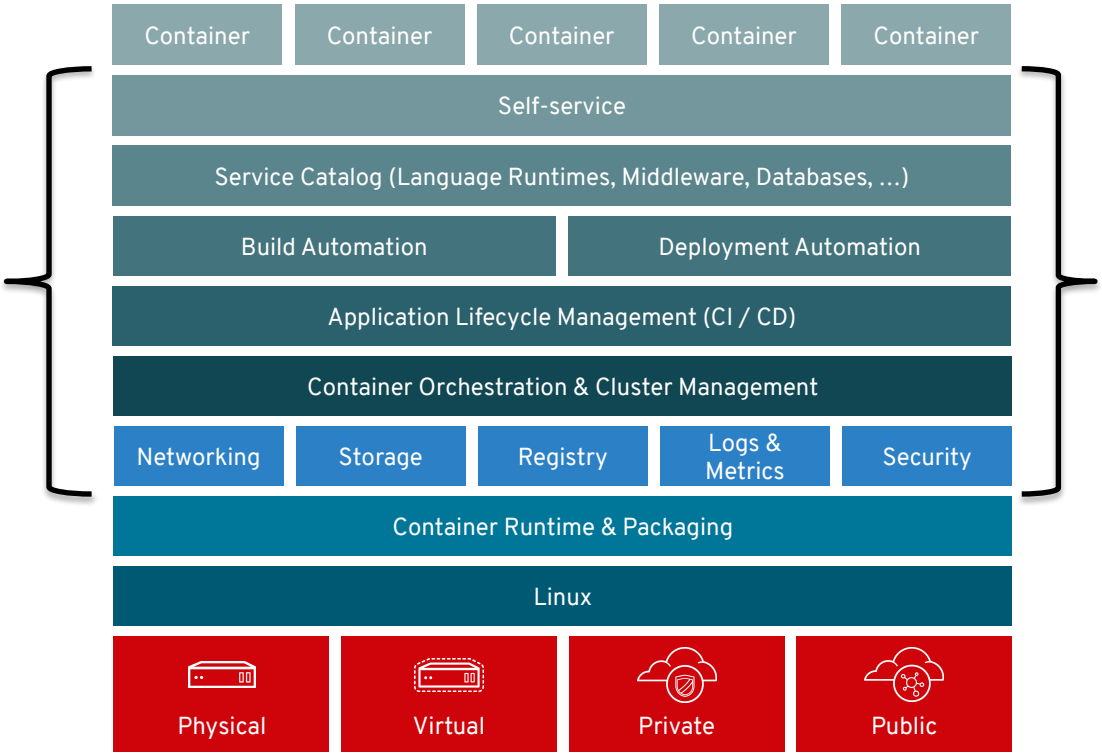
Red Hat

What Do You Need To Operationalize Containers?



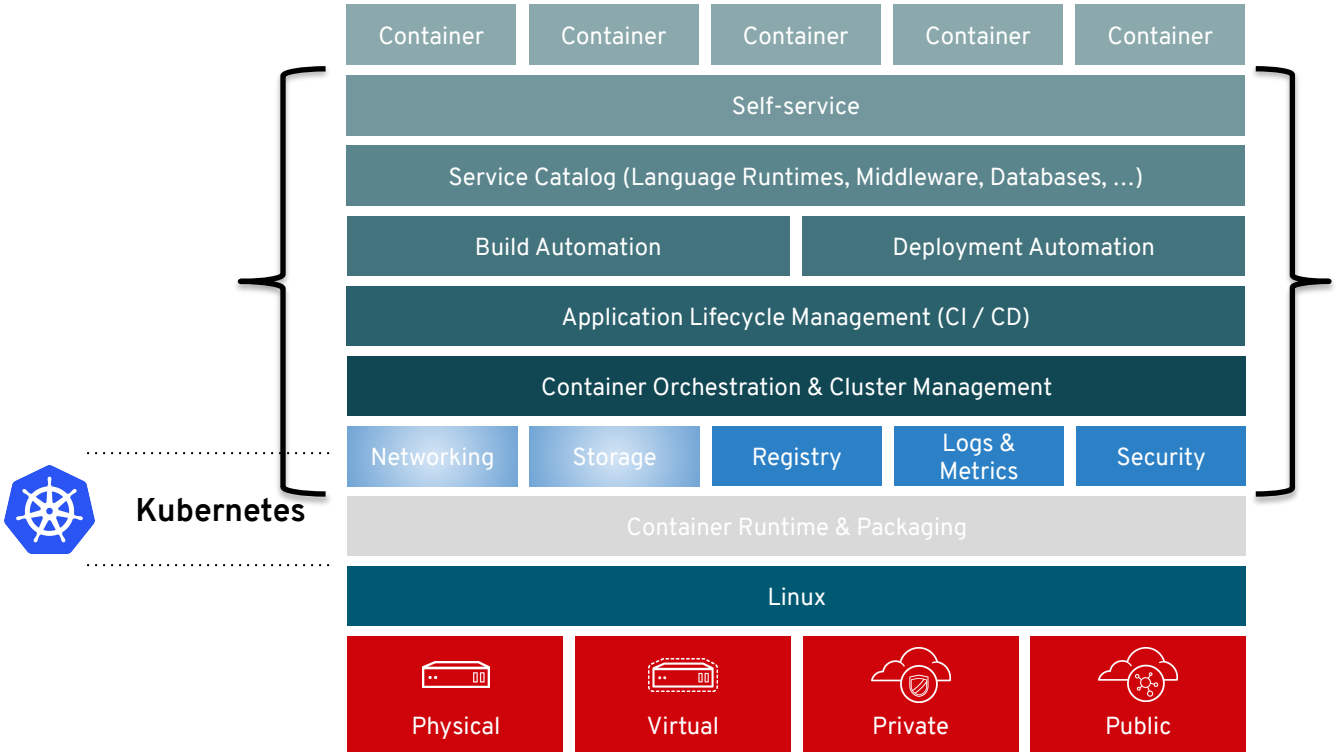
Red Hat

What Do You Need To Operationalize Containers?



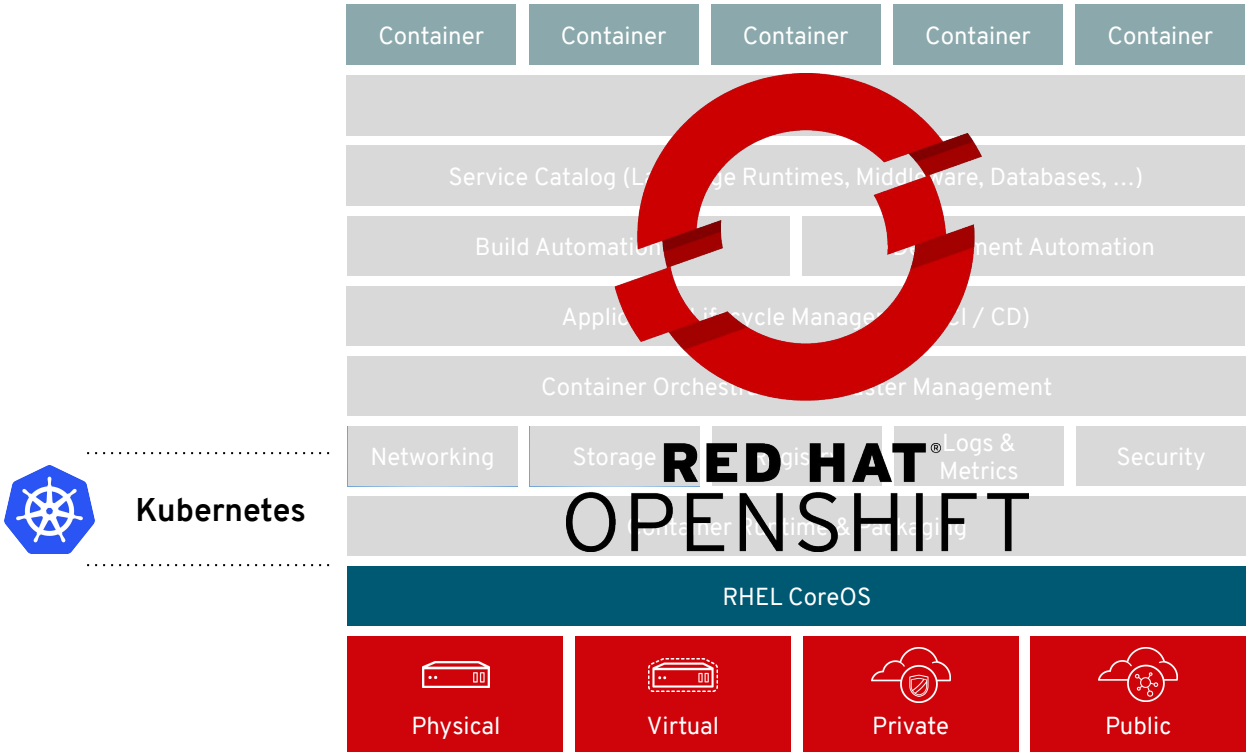
Red Hat

Kubernetes?



Red Hat

Kubernetes + OpenShift?



Red Hat

What’s Openshift

Project vs. Product?

Vs.

Kubernetes + OpenShift?

okd

- OpenShift Commons
- 100+ Integrations
- Align time with OSS trunk
- Integrate OSS projects
- Partner integration platform
- No-cost validations for innovation

OPENSHIFT CONTAINER PLATFORM

OPENSHIFT DEDICATED

AZURE RED HAT OPENSHIFT

OPENSHIFT ONLINE

Red Hat

DEPLOYMENT STRATEGIES

ORCHESTRATION | CLUSTER SERVICES

SECURITY | STORAGE | NETWORK

CONTAINER | CONTAINER | CONTAINER | CONTAINER | CONTAINER

SELF-SERVICE

OPENSHIFT SERVICES | SERVICE CATALOG

OPENSHIFT BUILDS | OPENSHIFT DEPLOYMENTS

APPLICATION LIFECYCLE AUTOMATION

ORCHESTRATION | CLUSTER SERVICES

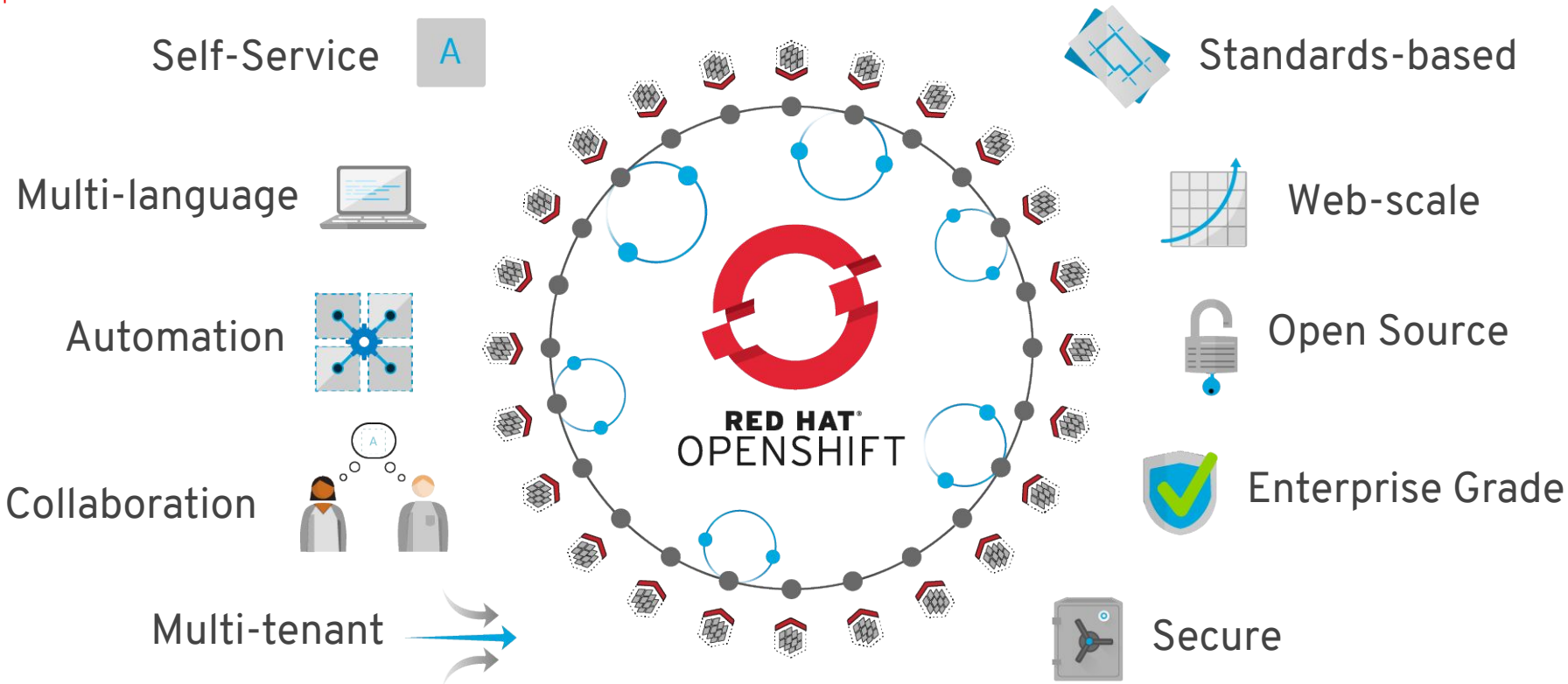
SECURITY | TELEMETRY | STORAGE | NETWORK | REGISTRY

PACKAGING FORMAT

RED HAT ENTERPRISE LINUX | ATOMIC HOST

PHYSICAL | VIRTUAL | PRIVATE | PUBLIC

OpenShift is full industry standard kubernetes, but we provide the surrounding components too that are required for a full deployment of a container platform.


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

Openshift Is Kubernetes
For The Enterprise

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

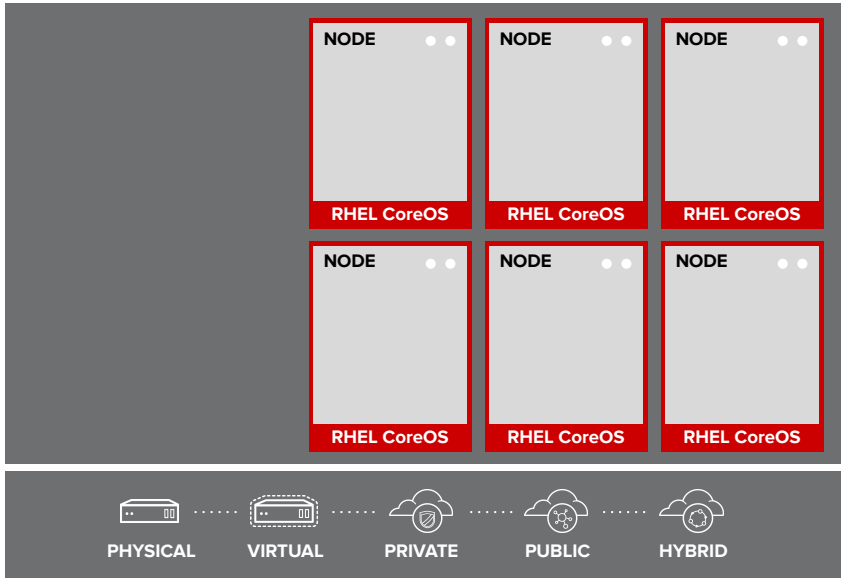
Red Hat

OpenShift Architecture

Your Choice Of Infrastructure

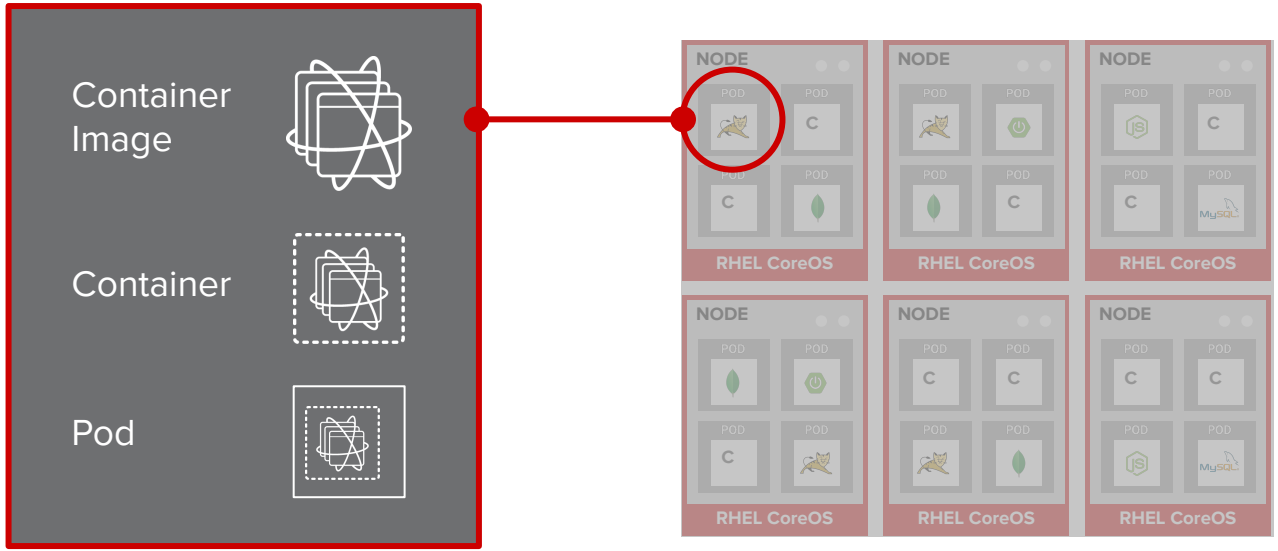
Red Hat

RHEL Is Where Apps Run



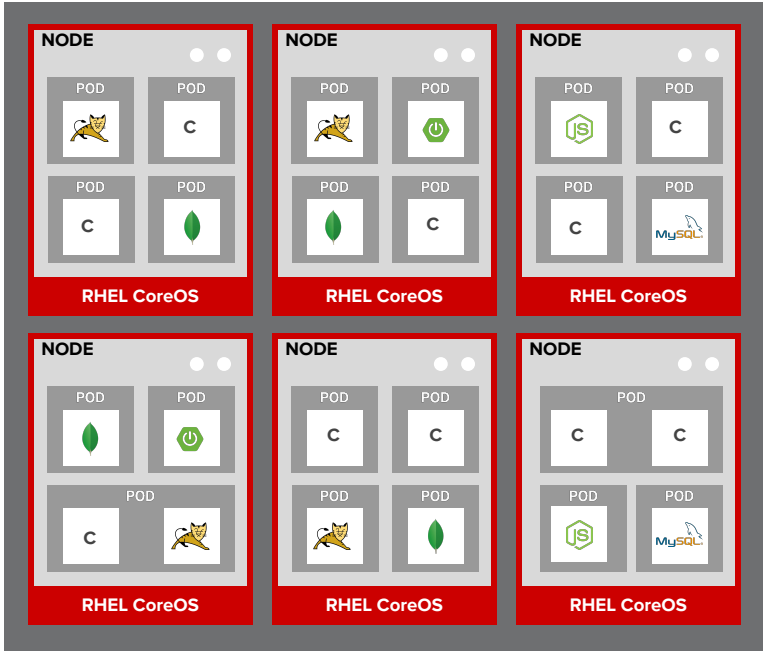
Red Hat

Apps Run In Containers



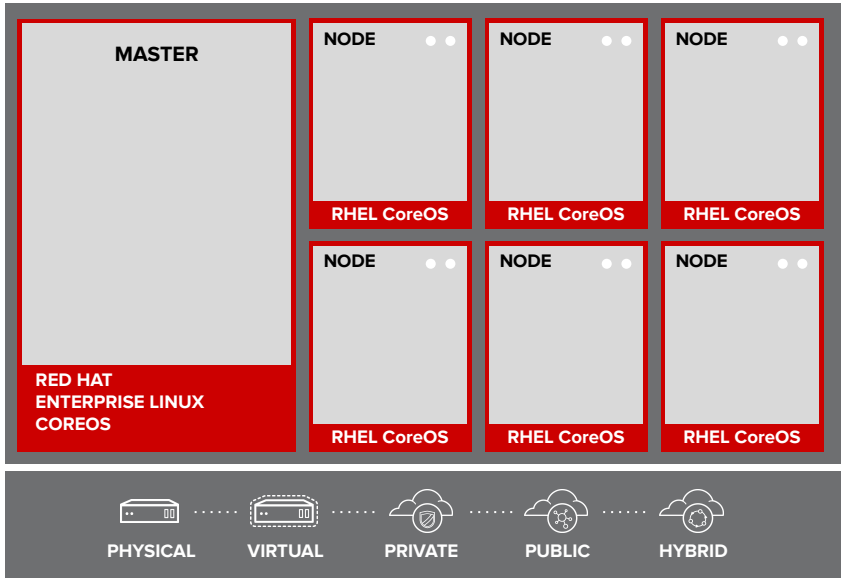
Red Hat

Pods Are The Unit of Orchestration



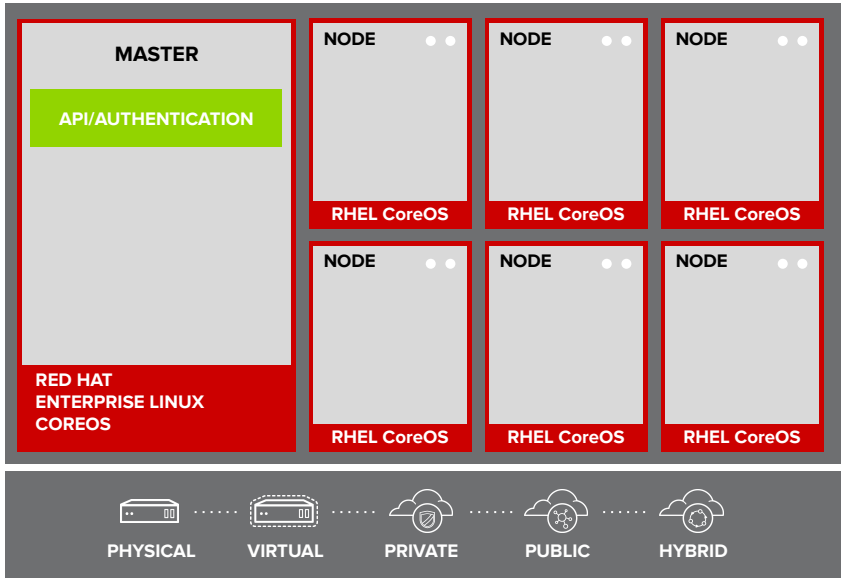
Red Hat

Masters Are The Control Plane



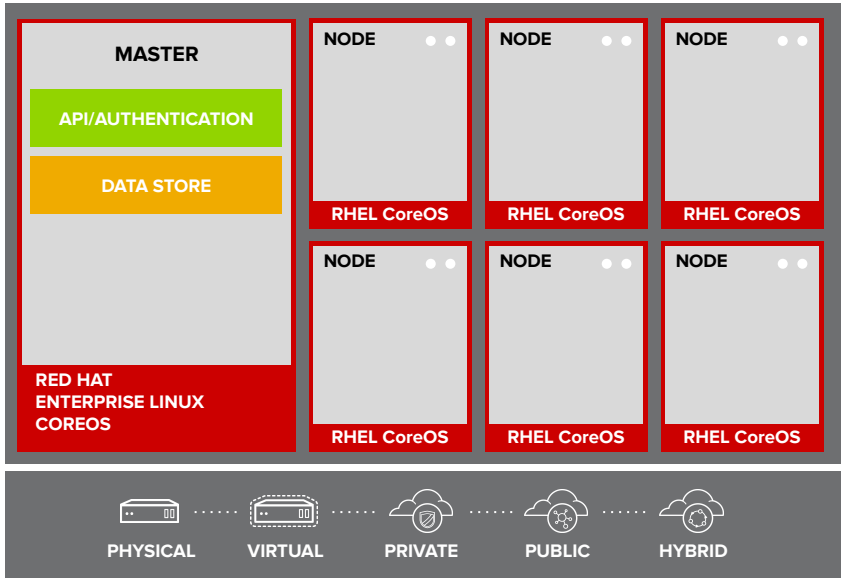
Red Hat

API And Authentication



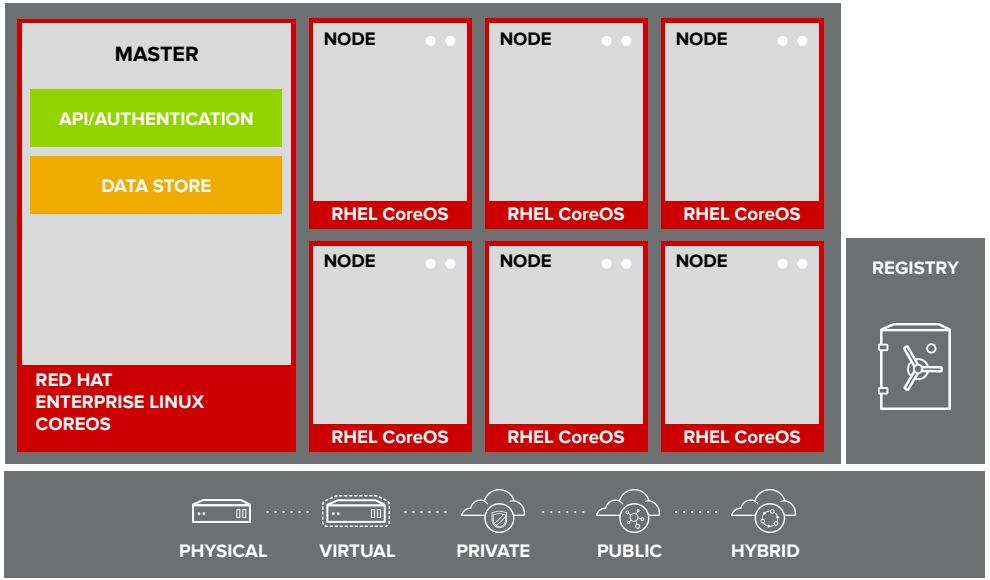
Red Hat

Desired And Current State



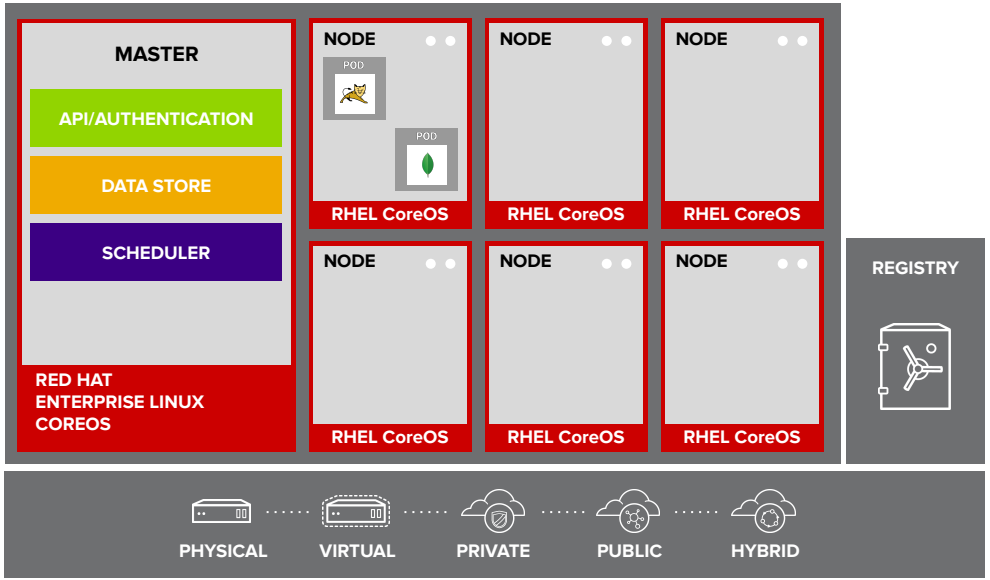
Red Hat

Integrated Container Registry



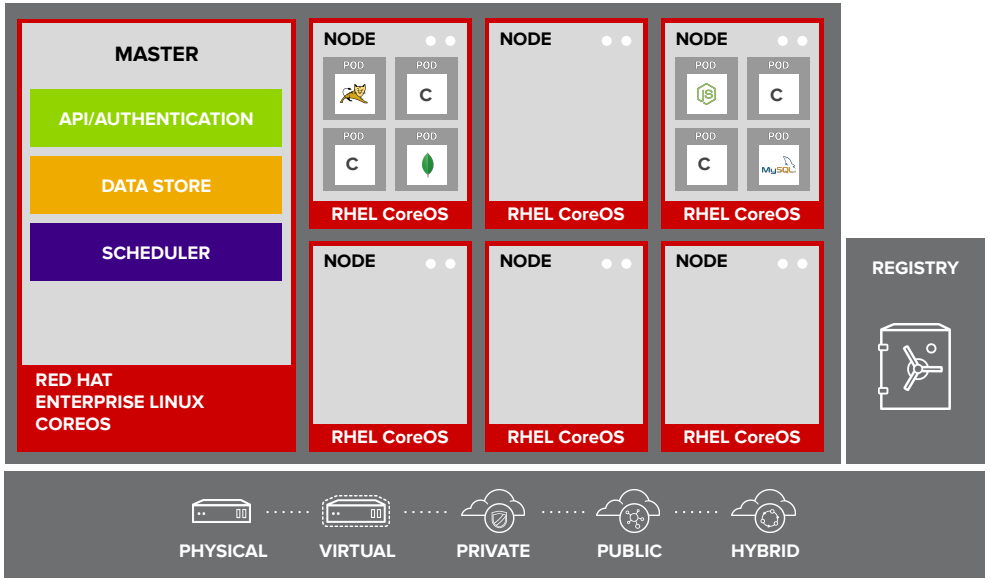
Red Hat

Orchestration And Scheduling



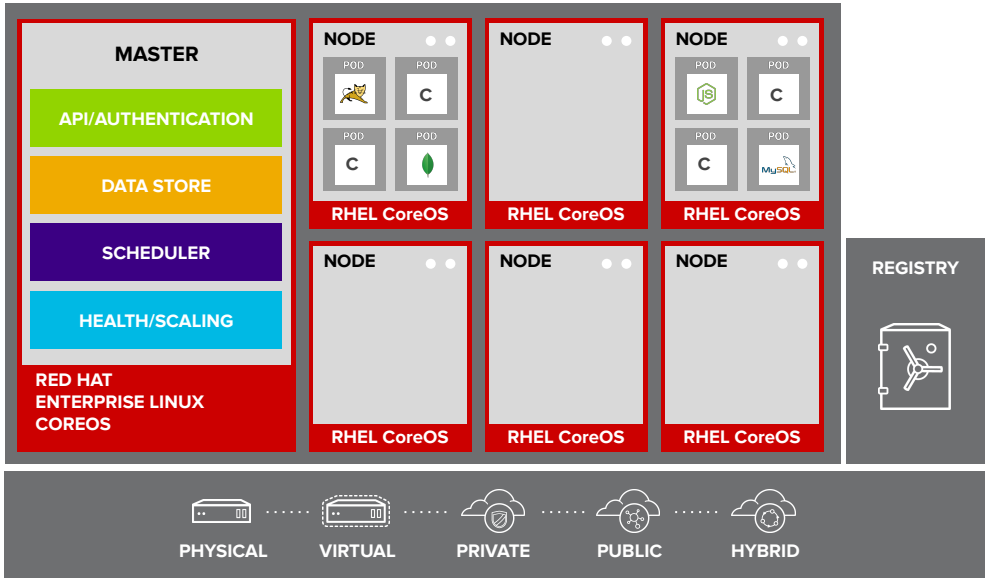
Red Hat

Placement By Policy



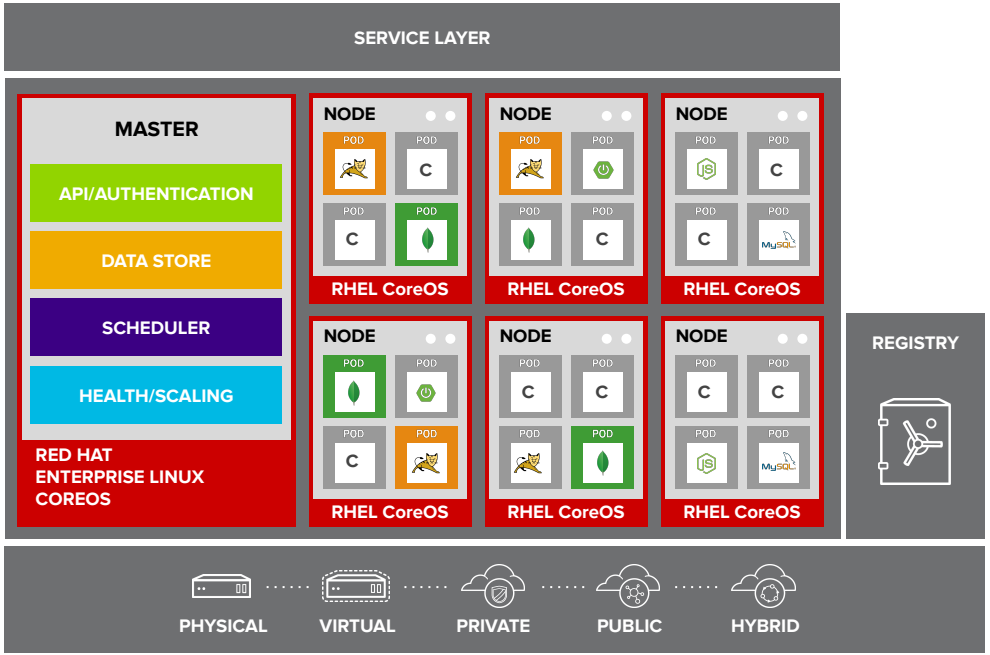
Red Hat

Autoscaling Pods



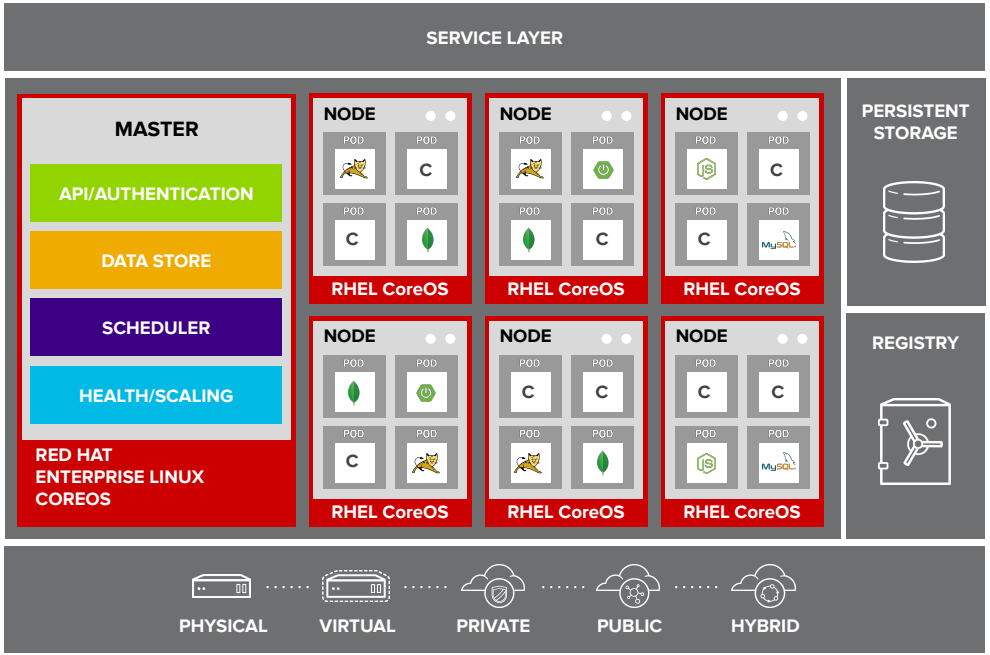
Red Hat

Service Discovery



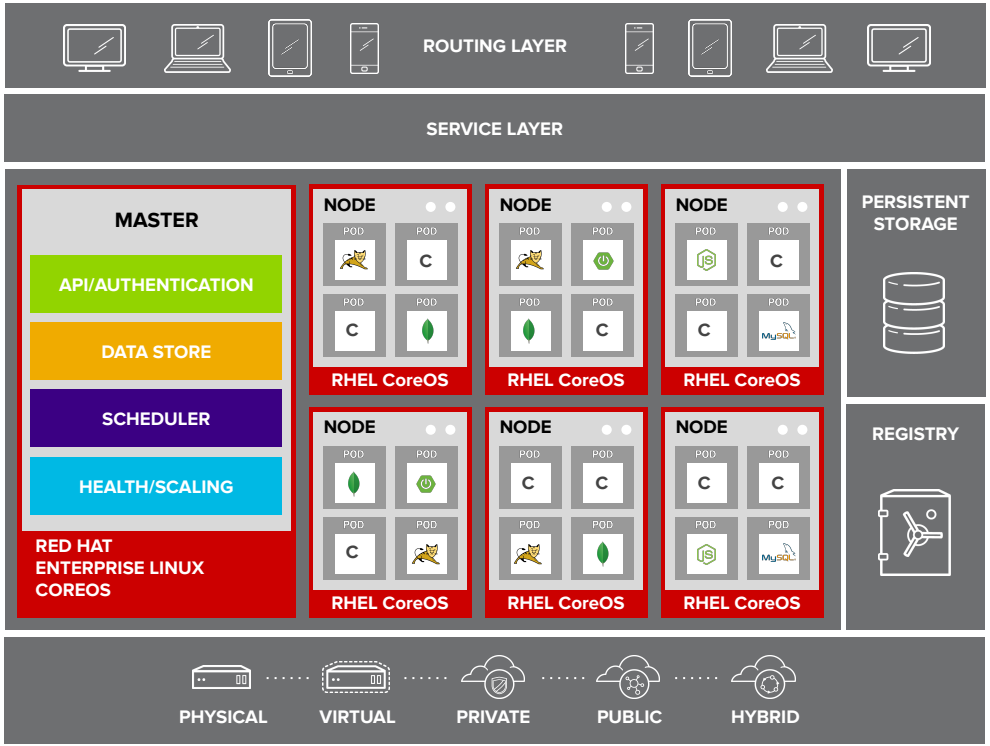
Red Hat

Persistent Data In Containers



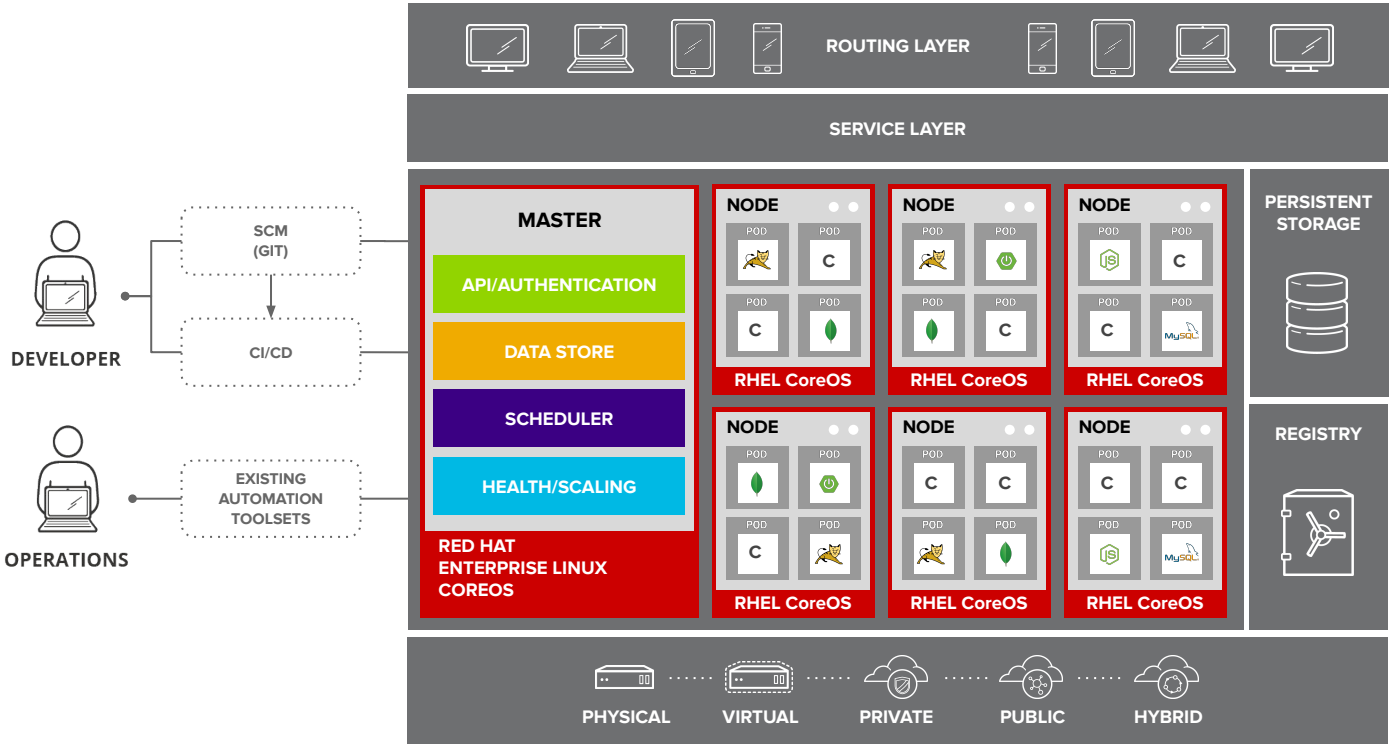
Red Hat

Routing And Load Balancing



Red Hat

Access Via Web CLI and API



Red Hat

Reference Architecture

Recommended Minimum Requirements

Master Host (x3)

- **CPU** - 4 vCPU
- **RAM** - 16 GB
- **DISK** - 40 GB

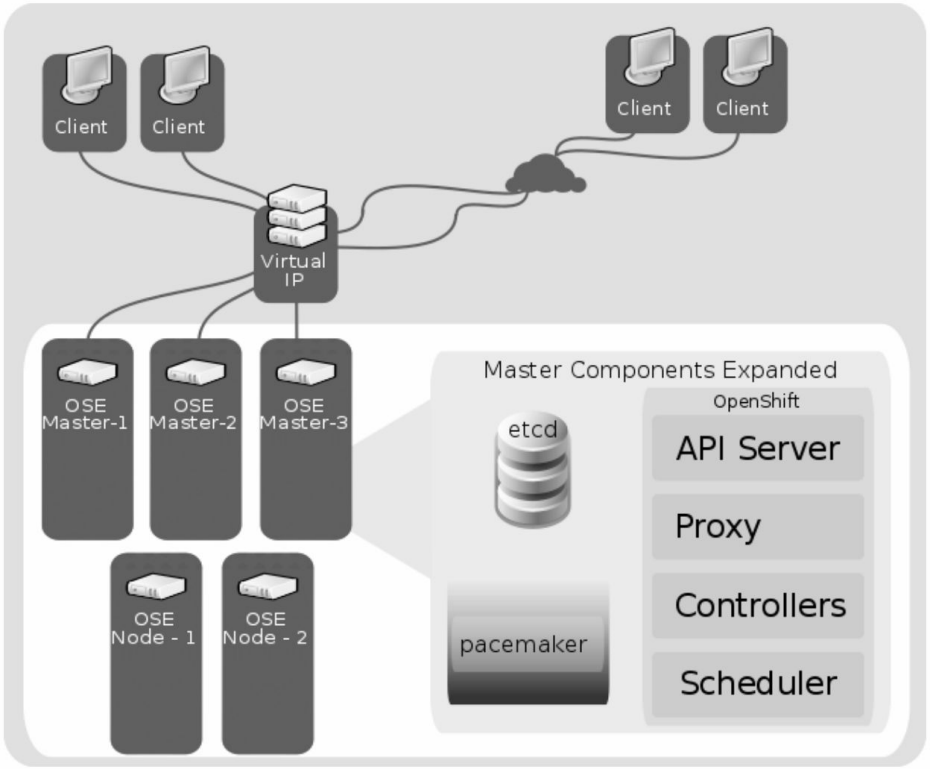
Node Host (x2)

- **CPU** - 1 vCPU
- **RAM** - 8 GB
- **DISK** - 15 GB

Reference

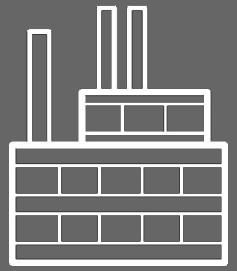
Red Hat

OpenShift Highly Available Deployment



Red Hat

Automated CI/CD Pipeline

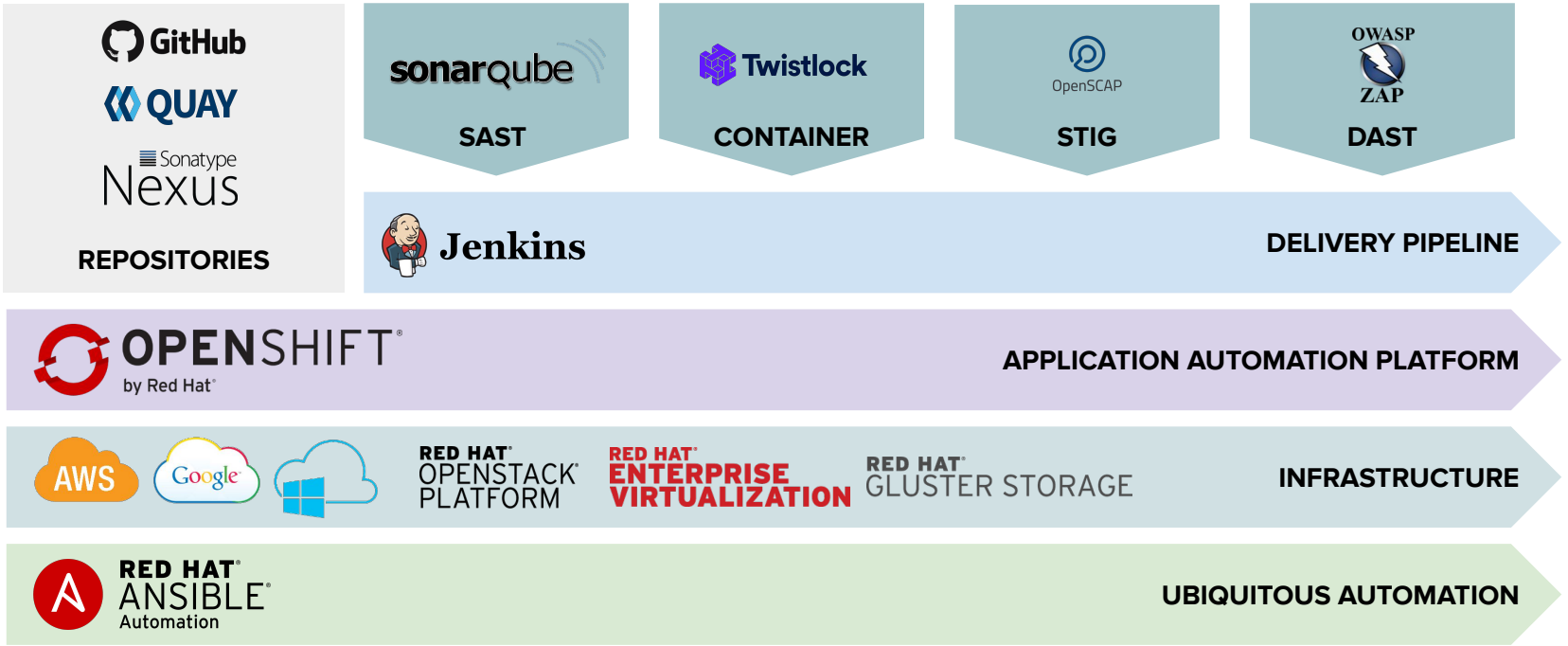


The Secure Software Factory

An opinionated approach to DevSecOps

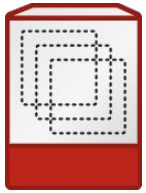
Red Hat

The Secure Software Factory



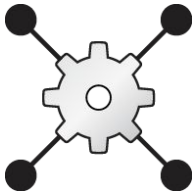
Red Hat

The Key Enabling Technologies Are Integrated Into Openshift



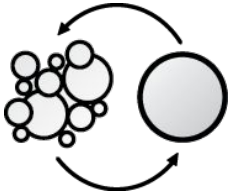
CONTAINERS

New paradigm unlocked by immutability, image layers, process isolation, portability



CONTAINER ORCHESTRATION

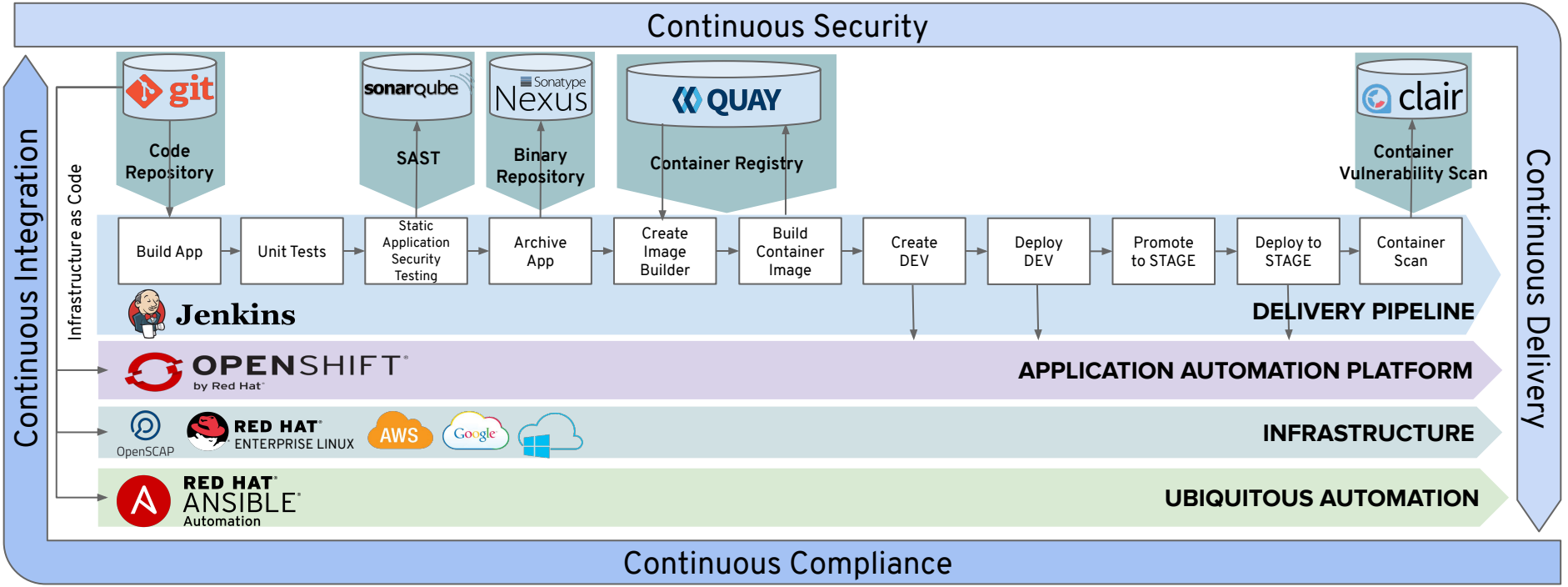
Operating containerized apps in production using a declarative configuration paradigm



CI/CD PIPELINE TOOLS/AUTOMATION

Build, package, and quality assurance processes codified and available on-demand

The Secure Software Factory



Red Hat

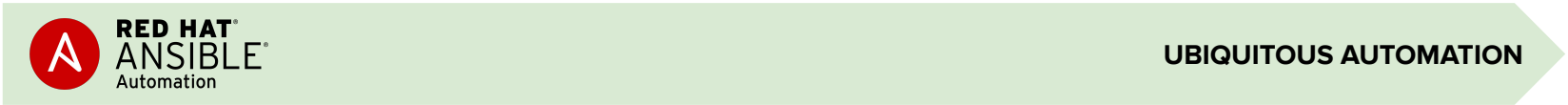
What Behaviors Does This Approach Promote?

- Small batch sizes
- Early, frequent testing (TDD)
- Ability to deliver changes quickly
- Supply chain assurance (BOM)

The Secure Software Factory

Red Hat

The Secure Software Factory



Red Hat

The Value Of Trusted Content

Red Hat Registry Stats

- 227 repositories
- 2,169 images
- 1+ TB storage

Red Hat Security Statistics 2016

- 97 critical RHSA
- 286 important RHSA
- 100% fixed in <1d

Red Hat Customer Portal Stats 2016

- 13,100,000 visitors
- 2,400,000 searches
- 108,300,000 views

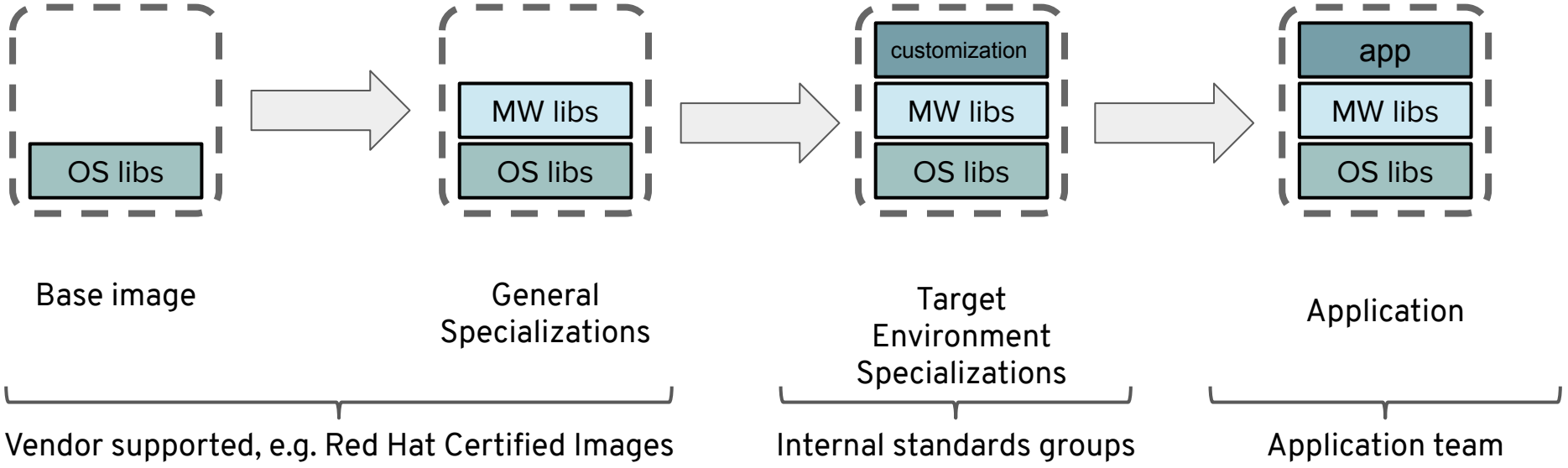
- Image Documentation
- Image Advisories

RED HAT® CONTAINER CATALOG

- Container Health Index
- Extensive Image Metadata

Red Hat

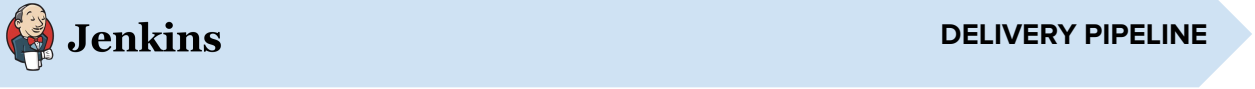
Standardize The Build Chain With Quality Parts

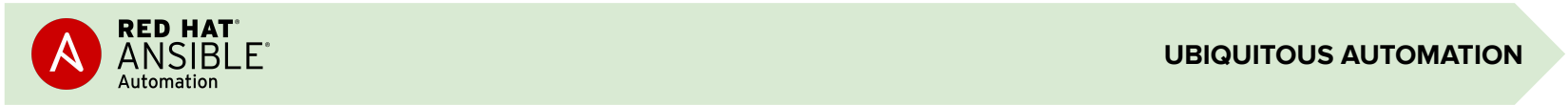


The Golden Image

Red Hat

The Secure Software Factory

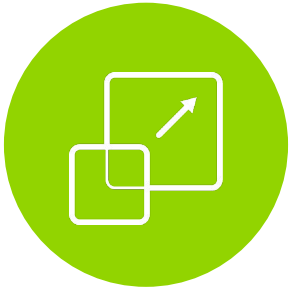




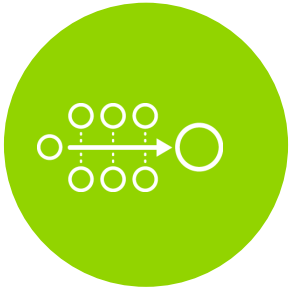
Red Hat

Openshift Loves CI/CD

JENKINS-AS-A SERVICE
ON OPENSHIFT



HYBRID JENKINS INFRA
WITH OPENSHIFT

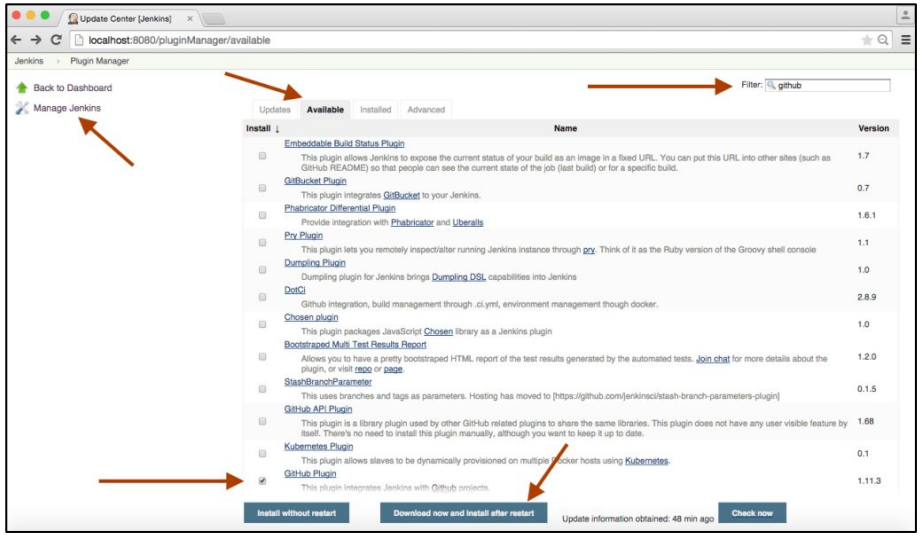


EXISTING CI/CD
DEPLOY TO OPENSHIFT

Red Hat

Jenkins Plugin

- The most fundamental part of a Pipeline
- Tell Jenkins *what* to do, and serve as the basic building block for both Declarative and Scripted Pipeline syntax



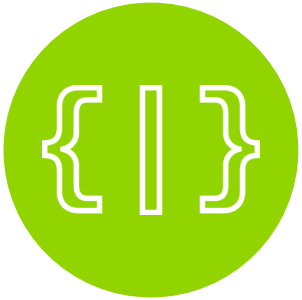
Red Hat

```
kind: BuildConfig
apiVersion: v1
metadata:
  name: sample-pipeline
  labels:
    Name: sample-pipeline
spec:
  triggers:
  - type: GitHub
    github:
      secret: secret101
  - type: Generic
    generic:
      secret: secret101
strategy:
  type: JenkinsPipeline
  jenkinsPipelineStrategy:
    jenkinsfile: |-
      node('maven') {
      stage 'build'
            openshiftBuild(buildConfig: 'ruby-sample-build', showBuildLogs:'true')
            stage 'deploy'
            openshiftDeploy(deploymentConfig: 'frontend')
      }
```

OPENSHIFT PIPELINES IN WEB CONSOLE

Red Hat

Build And Deploy Container Images



DEPLOY YOUR SOURCE CODE

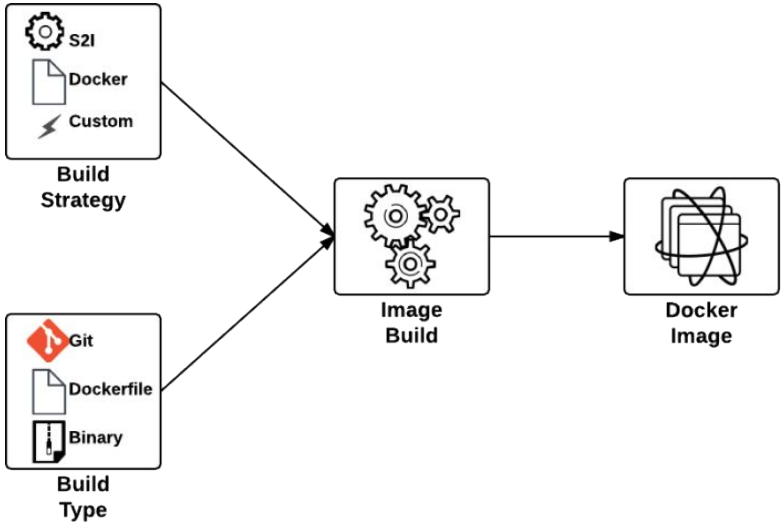
DEPLOY YOUR APP BINARY

DEPLOY YOUR CONTAINER IMAGE

Red Hat

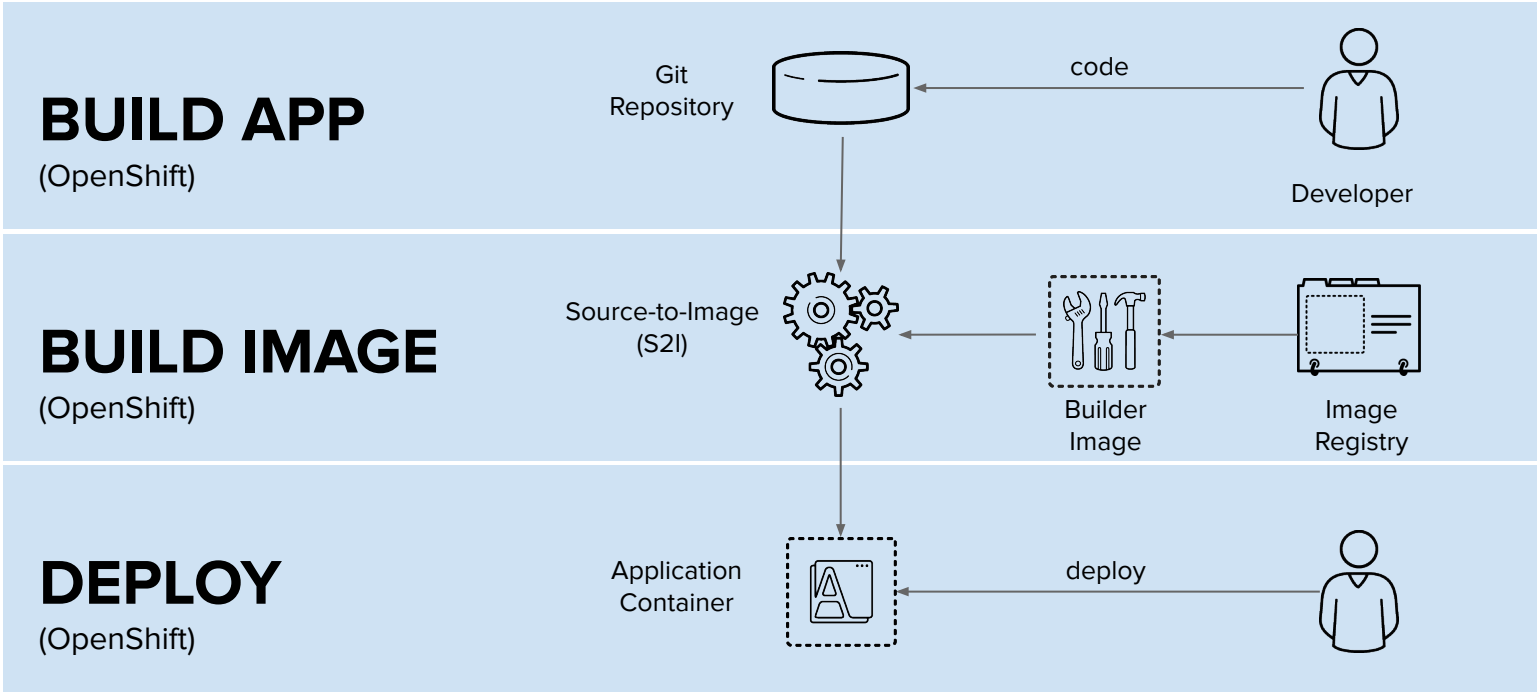
Openshift Build Strategies

- Source: use source code from git repository or Dockerfile as the build input
- Binary: Streaming content in binary format from a local file system to the builder
- Image: Additional files can be provided to the build process via images. Files will copy from source image to destination image.

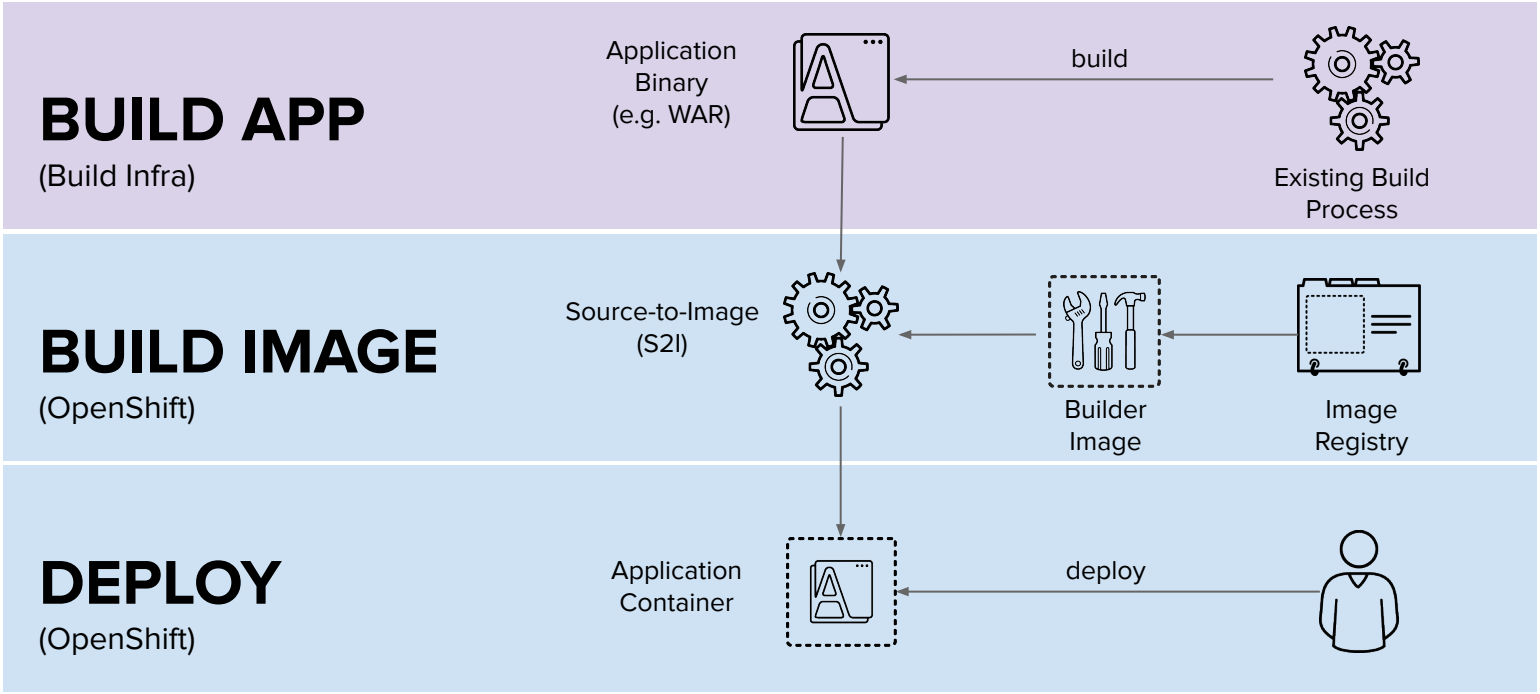


Red Hat

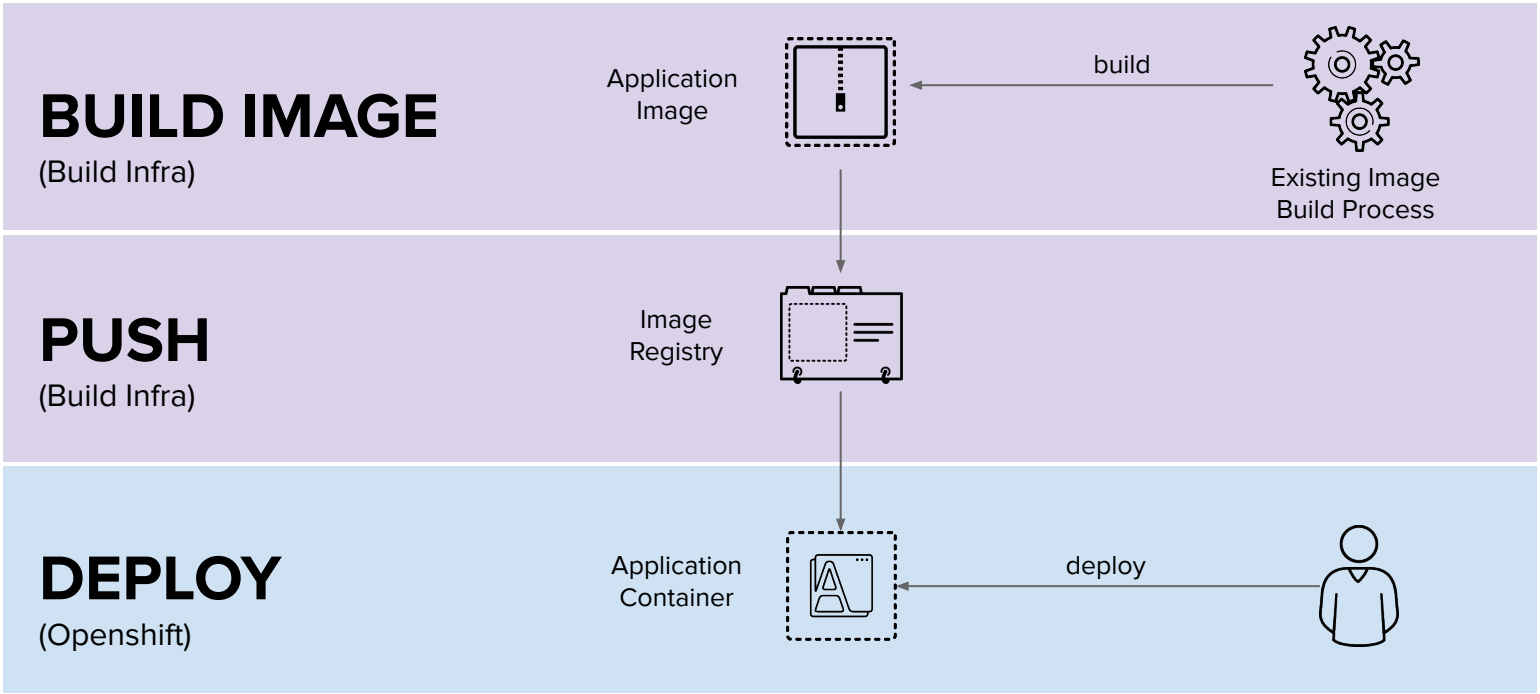
Source Code Deployment



App Binary Deployment



Container Image Deployment

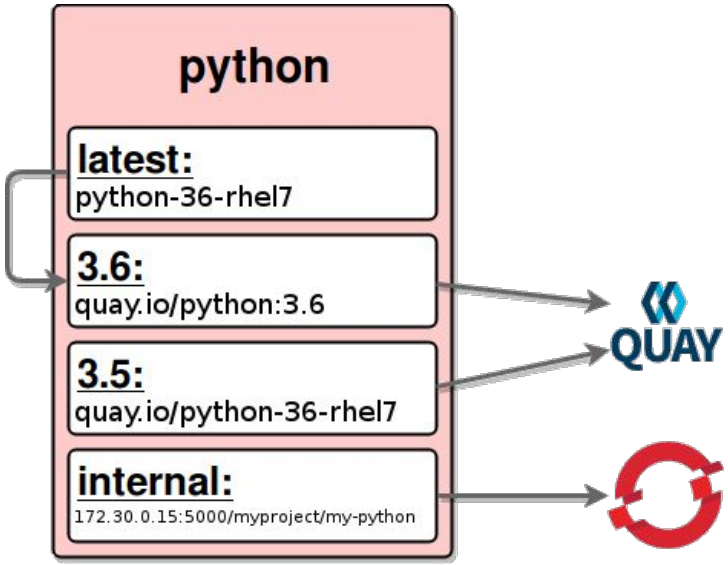


Openshift Build Steps Include

- Trigger a build in OpenShift
- Verify a build succeeded
- Trigger a deployment
- Scale a deployment up/down
- Verify a deployment succeeded
- Verify a service is accessible
- Tag an image
- Create Resource via YAML/JSON
- Delete any resource
- Cancel a build
- Cancel a deployment

What Are Image Streams?

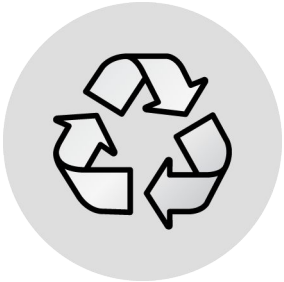
- Contains all of the metadata information about any given image that is specified in the Image Stream specification
- Does not contain the actual image data
- Ultimately points either to an external registry, e.g. registry.access.redhat.com, quay.io, OpenShift's internal, etc.



Red Hat

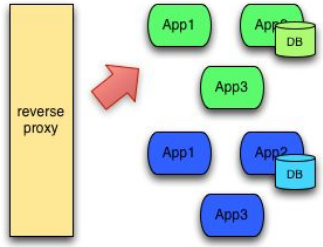
Openshift Deployment Strategies

Painless deployments with zero/reduced downtime through automation



ROLLING DEPLOYMENTS

A rolling deployment slowly replaces instances of the previous version of an application with instances of the new version of the application.



BLUE/GREEN DEPLOYMENTS

A blue/green deployment is a software deployment strategy that relies on two identical production configurations that alternate between active and inactive.

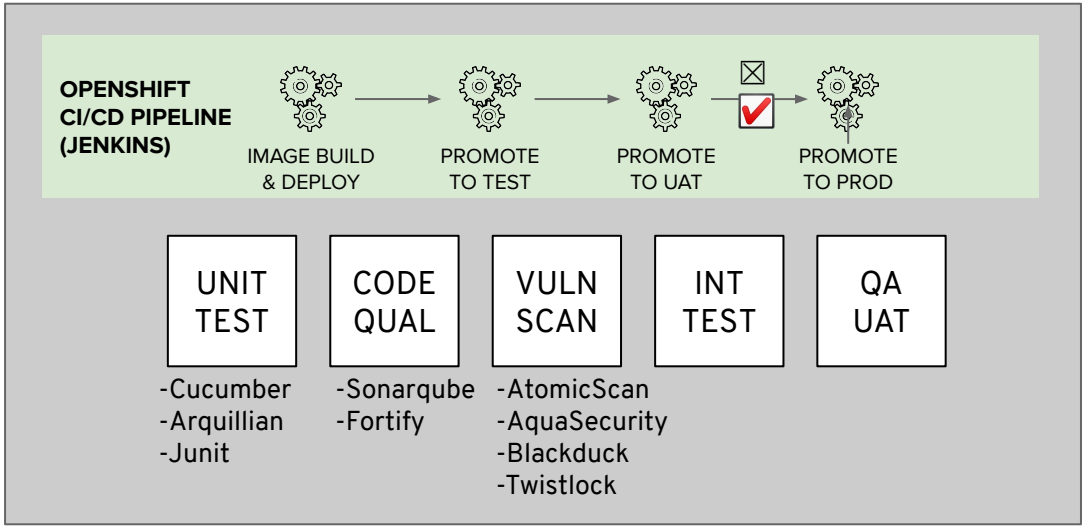
A/B DEPLOYMENTS

A/B testing (sometimes called split testing) is comparing two versions of a web page to see which one performs better.

Red Hat

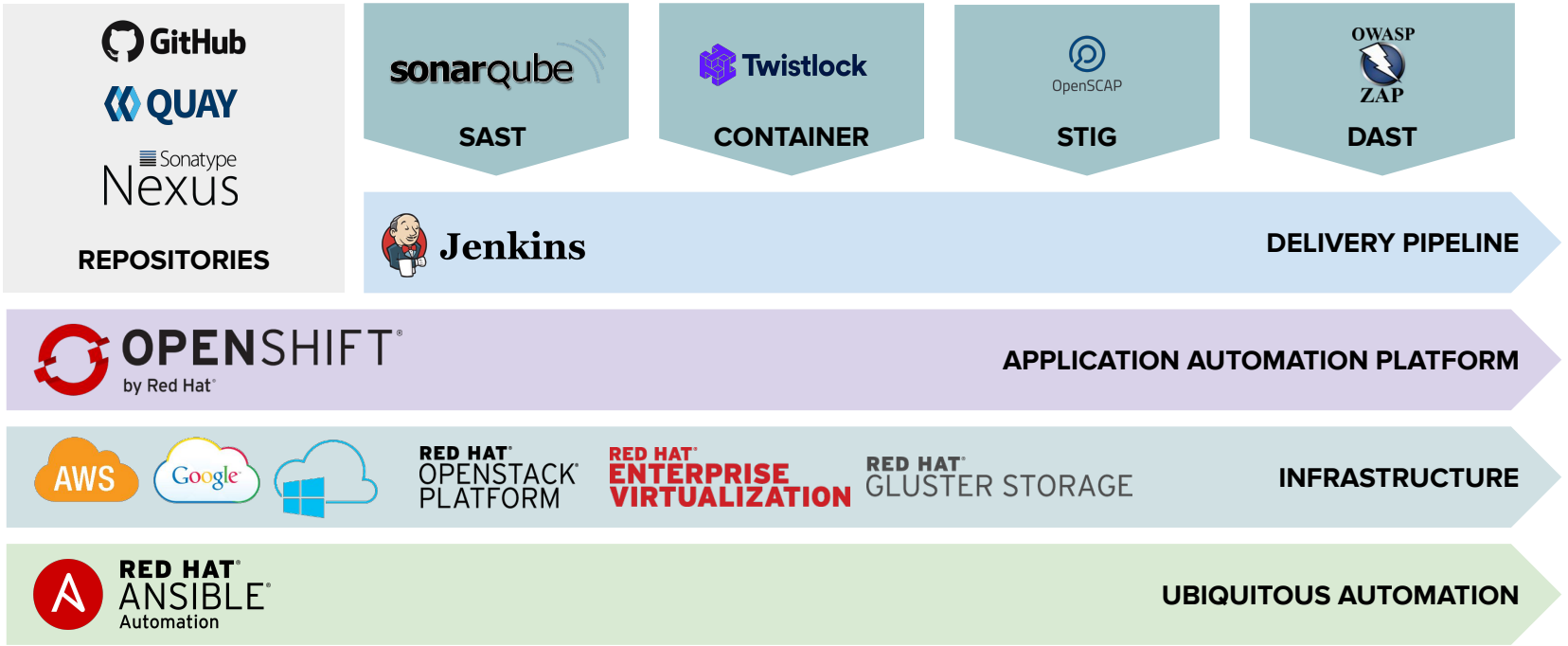
Continuous Integration Must Include Gates

- Integrate security testing into your build / CI process
- Use automated policies to flag builds with issues
- Trigger automated rebuilds
- Sign your custom container images
- Design for separation of concerns



Red Hat

The Secure Software Factory

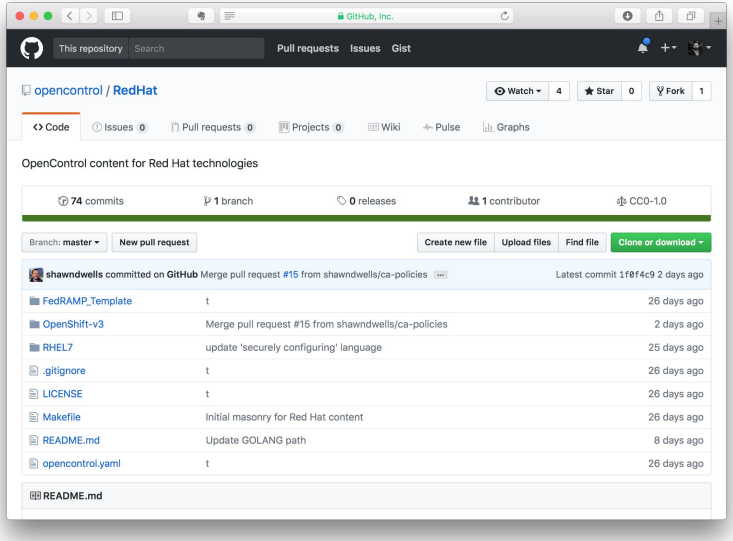


Red Hat

Prove It: Open Control

CODE DRIVEN PRODUCTION OF TAILORED COMPLIANCE DOCUMENTATION

- Open Source, community driven security compliance documentation
- Generative from your pipelines
  - Templates for multiple compliance targets and profiles
  - A full documentation suite: CONOPS, SRTM, CRM, etc
  - Linked to implementation tools: SCAP Content, Ansible Playbooks, etc
- Still very early implementation/in development



Red Hat

Security Content Automation Protocol

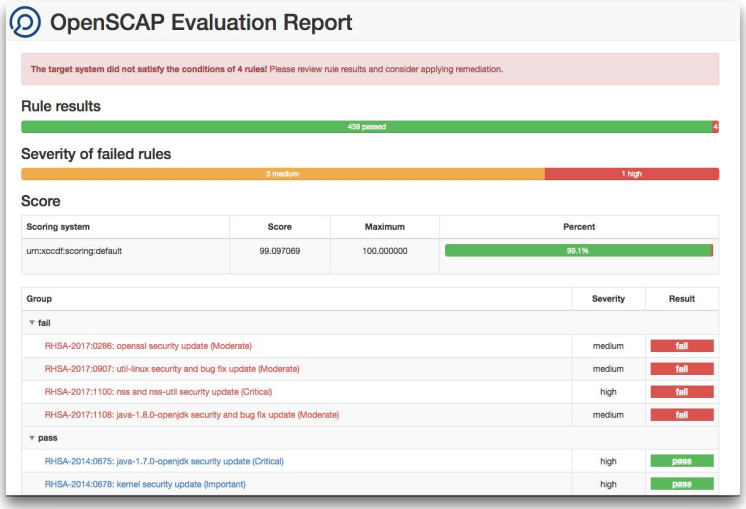
REPLACING OUTDATED, MANUAL PROCESSES WITH

- Group of standards designed to automate management, assessment, and policy compliance
- Many components such as CVE, CCE, XCCDF, OVAL
- Open source implementation is OpenSCAP (https://open-scap.org)
- SCAP Workbench GUI
- RHEL STIG XCCDF profile shipped with SCAP Security Guide (SSG)
- Run container specific SCAP content and scans with AtomicScan from Cloudforms

OpenSCAP BASE

SCAP SECURITY GUIDE

SCAP WORKBENCH



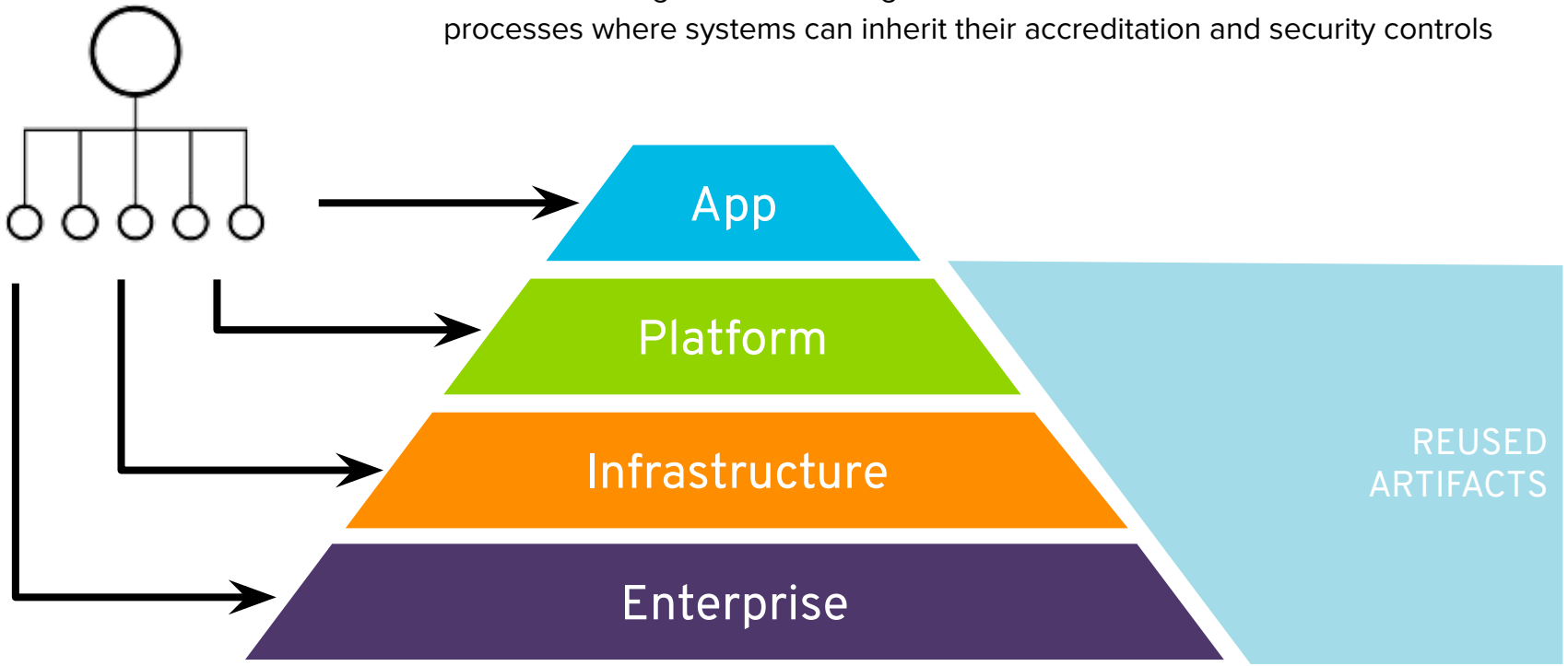
Red Hat

HOW DOES THE FACTORY HELP ME WITH SECURITY?

Red Hat

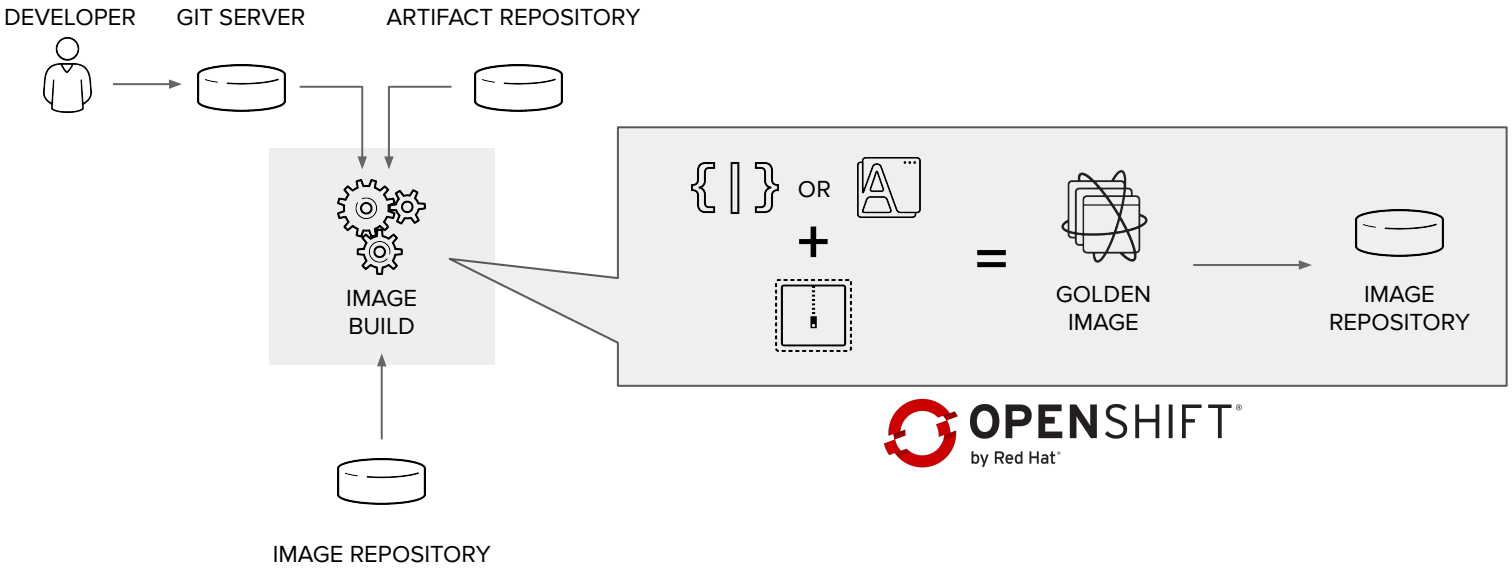
The Security Inheritance Model

Cloud technologies are heralding in a new era of **landlord-tenant** accreditation processes where systems can inherit their accreditation and security controls



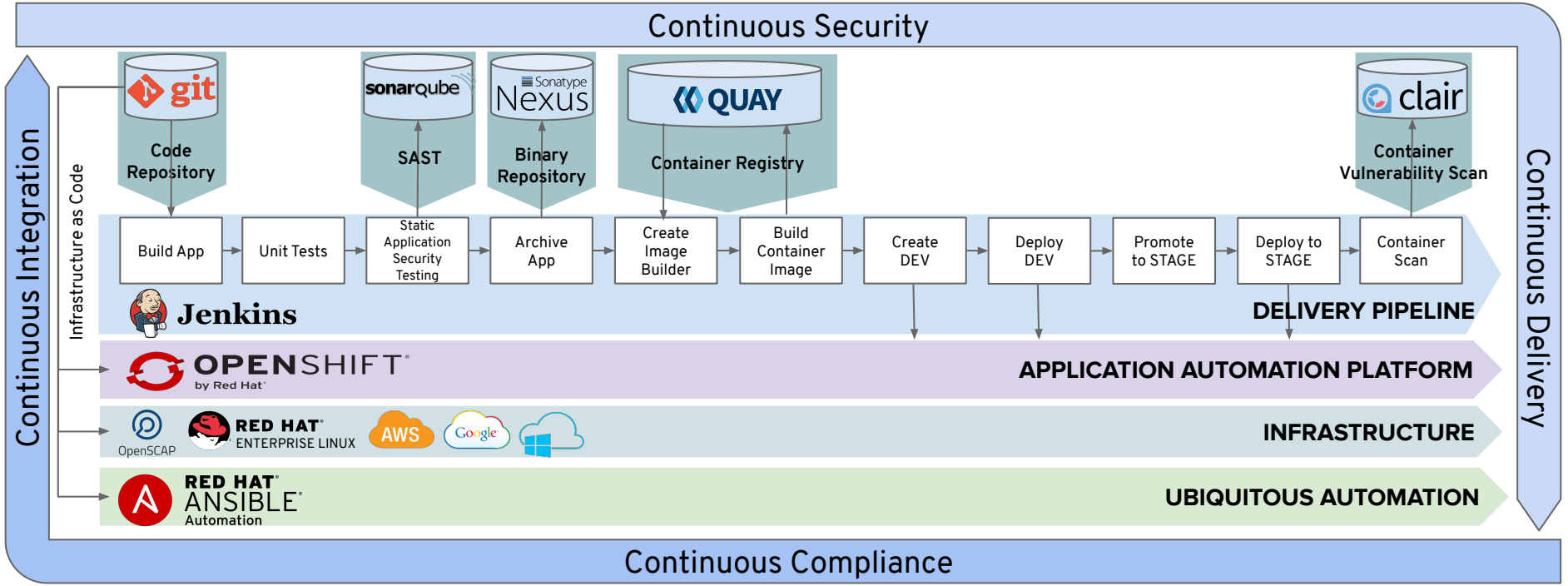
Red Hat

Building Applications On Openshift



Red Hat

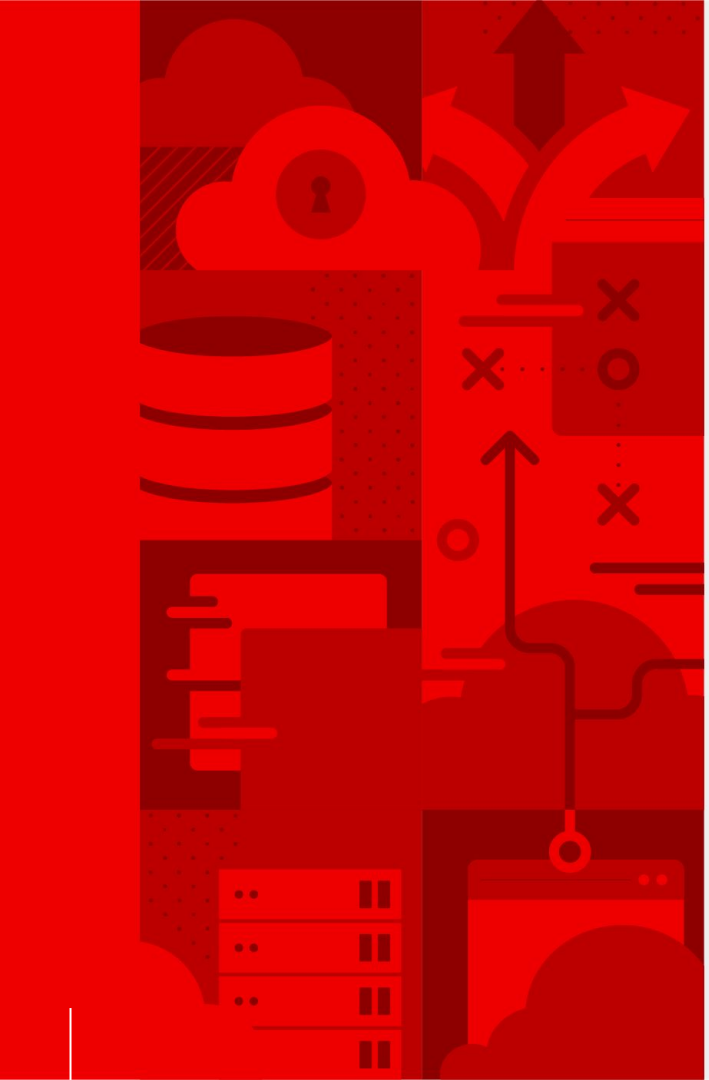
The Secure Software Factory



Red Hat

Summary Of Benefits

- Inherit security/compliance from OpenShift and trusted content repositories (e.g. Red Hat Container Registry)
- Standardize and automate app build process and dependency updates
- OpenShift has the flexibility to integrate your existing tools and governance



Thank You

OpenShift - Empower developers to innovate and ship faster with the leading hybrid cloud, enterprise container platform

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat