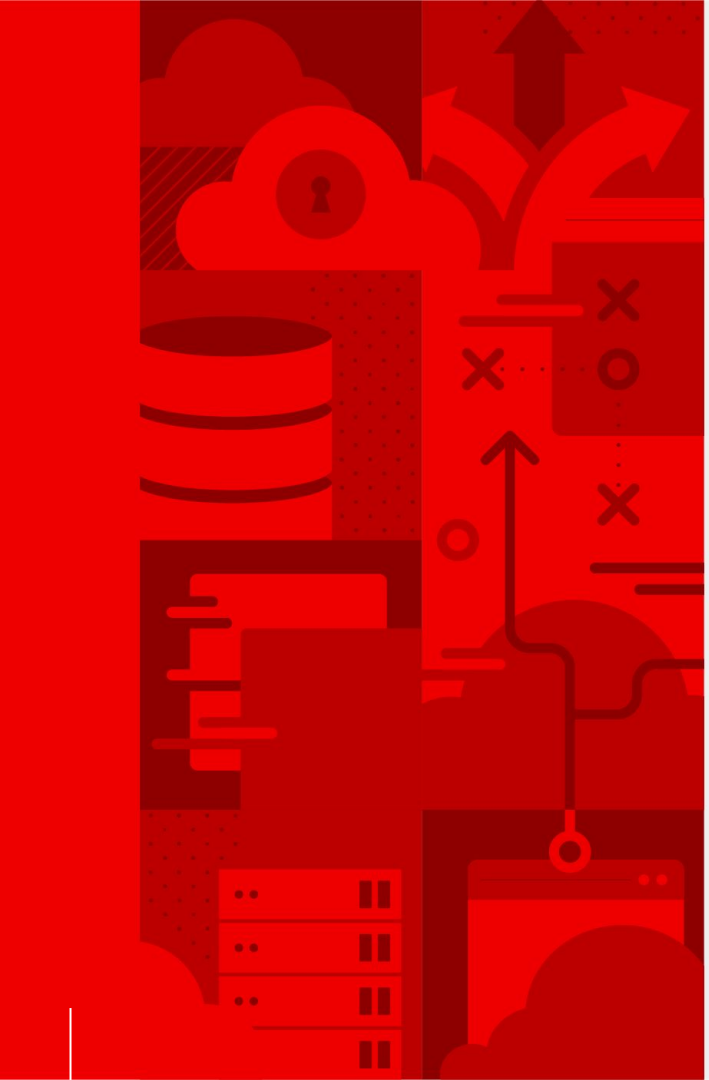


OpenShift Developer

Distance Learning

Introduction to OpenShift for Developers

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat

Self introduction

Name: Wanja Pernath

Email: wpernath@redhat.com

Base: Germany (very close to the Alps)

Role: EMEA Technical Partner Development Manager - OpenShift and MW

Experience: Years of Consulting, Training, PreSales at Red Hat and before

Agenda

Agenda

- What is this DLP all about
- How to use this DLP
- What is your Outcome at the end of this week
- The OpenShift Ecosystem for Developers
  - Basics
  - Advanced topics
- How to demo OpenShift for Developers
- Outlook of Week #2

What is this all about?

Concept

- 4 Weeks of technical training mostly based on (technical) self learning and presentations
- Each week starts with a webinar which talks about the content of the week
- **Result is a working demo and a pitch to present an OpenShift based use case in front of your customers**

Learning Goals

- Getting an overview of the whole OpenShift ecosystem for developers
- Learn about the benefits of OpenShift prior plain Kubernetes
- Learn how to effectively and efficiently demo OpenShift for developers
- Learn to use OpenShift and its ecosystem
- Learn to start coding with OpenShift in your preferred language
- Learn to do proper release management with OpenShift; understand the basics and how to include it into your release management process but also learn how to benefit from OpenShift Pipelines and when to use what
- Understanding and using Operators for releases
- Understand and make use of advanced OpenShift features like Istio and Serverless

Duration and Effort

- Four weeks of work for after hours
- Every week one must have webinar
- One optional webinar to ask questions
- Plan 6-8 hours every week
- The rest: it’s online, it’s self paced, you can decide

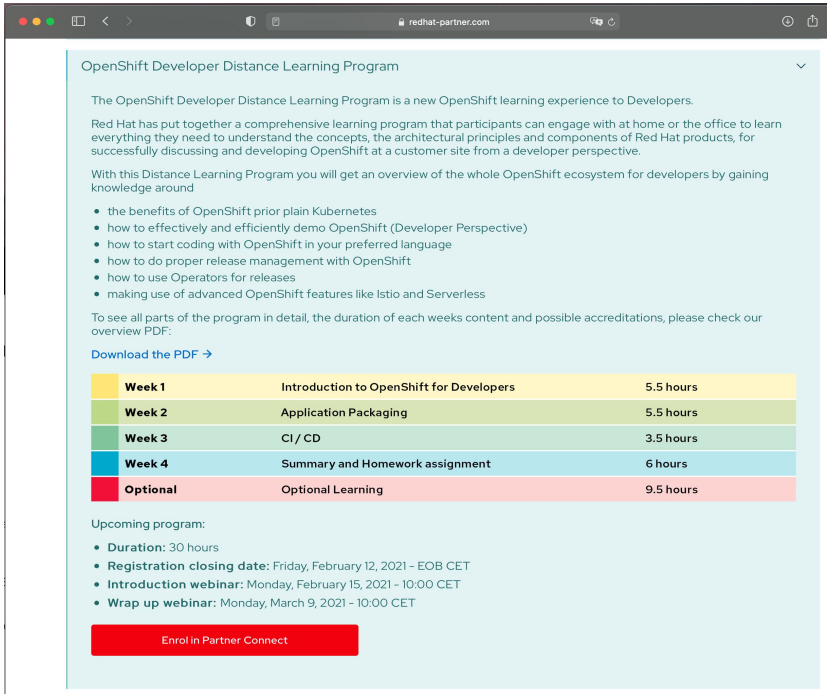
How to work with this DLP

Access your DLP

Go to www.redhat-partner.com

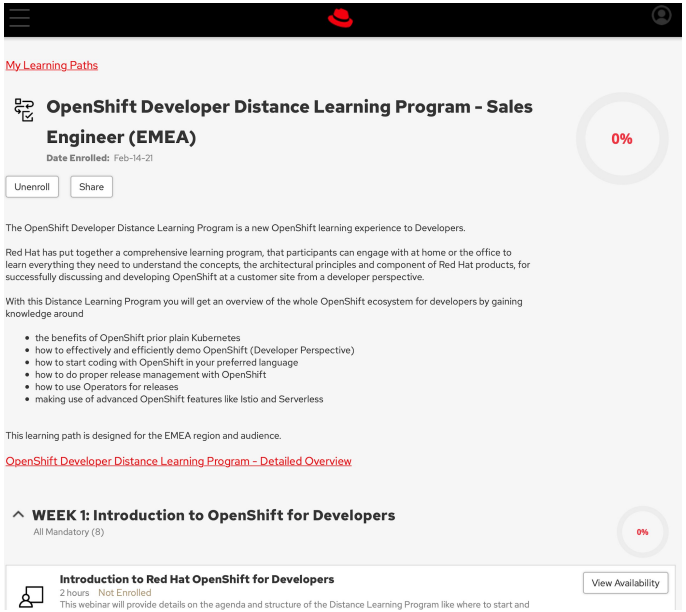
Choose your DLP

Click on Enroll in Partner Connect



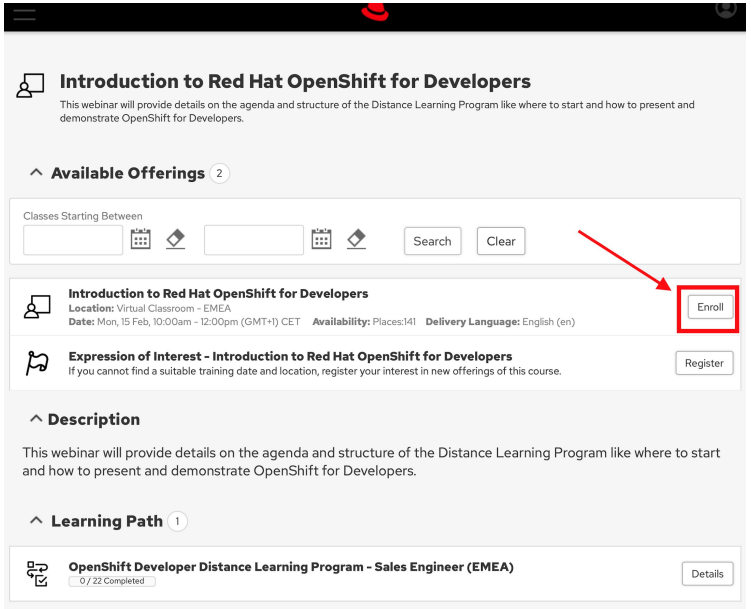
Access your DLP

- OPEN opens and shows the DLP learning path
- You can see the overall progress and the progress by week
- **IMPORTANT:** You have to click on “View availability” on each WEBINAR (this first entry per week) otherwise you won’t get notified by us.



Access your DLP

- And here you have to click on “Enroll” separately for each Webinar
- For all the other entries, you easily click on “Launch”
- A window opens with the training

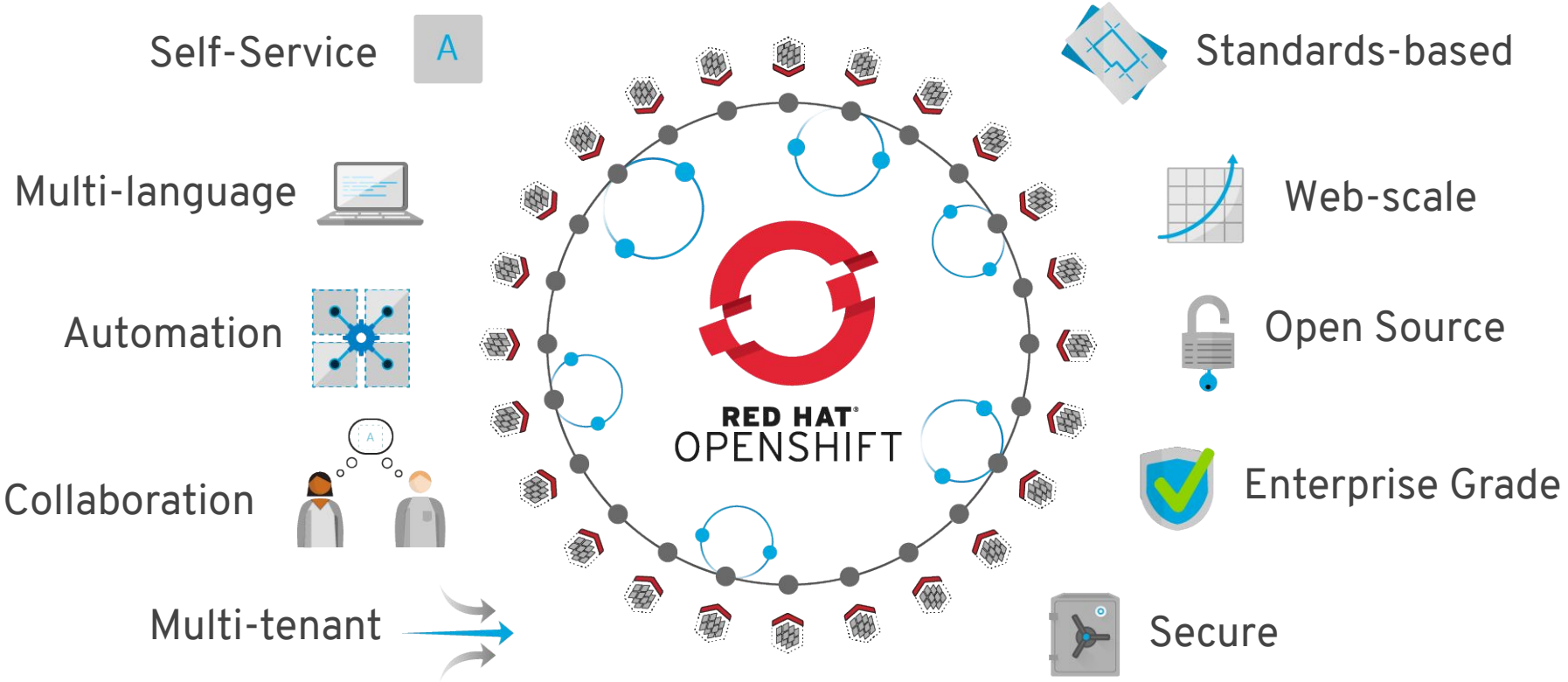


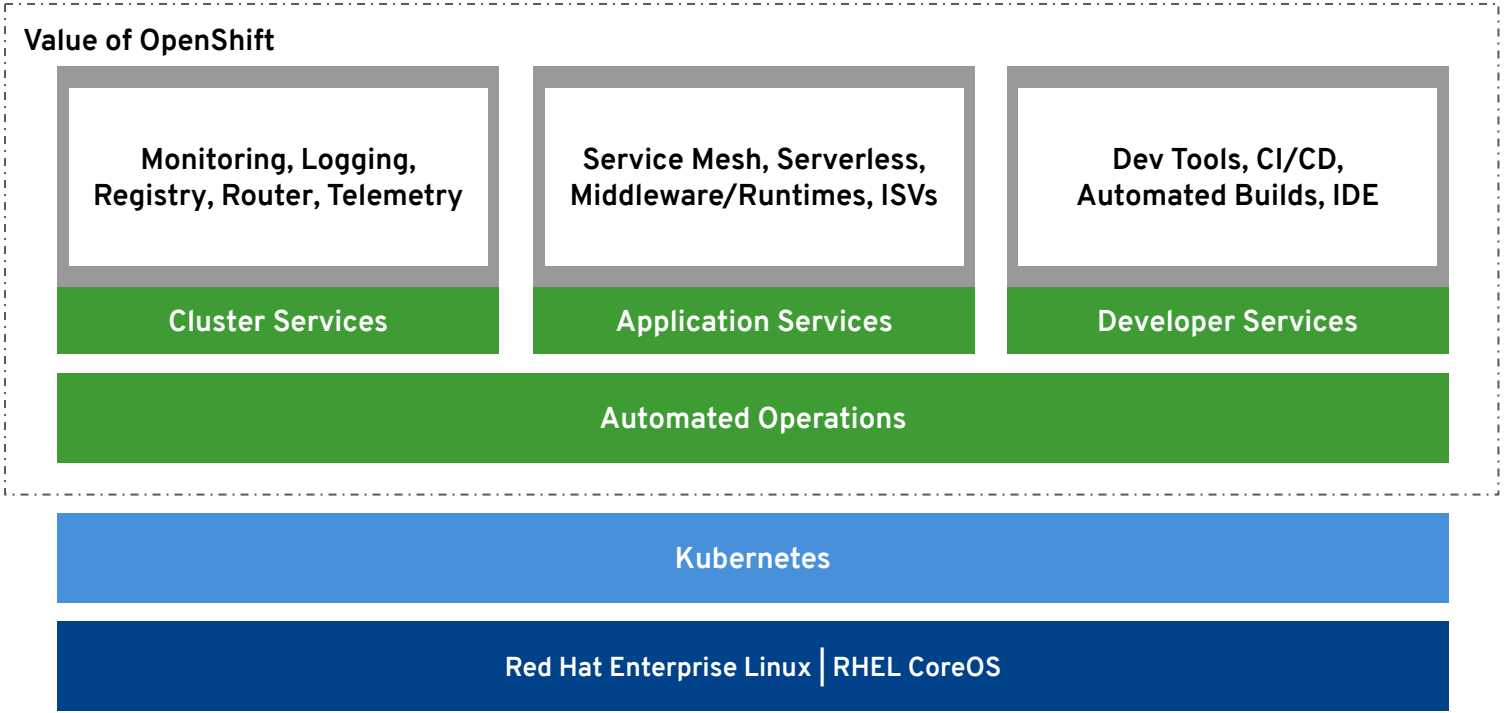
Outcome after Week #1

Week #1 - Outcome

- You've become familiar with the core concepts of OpenShift for Developers
- CLI interfaces with oc and odo
- You can create and deploy apps on OpenShift
- You can use secrets, ConfigMaps and can deploy and use databases

OpenShift Developers Basics

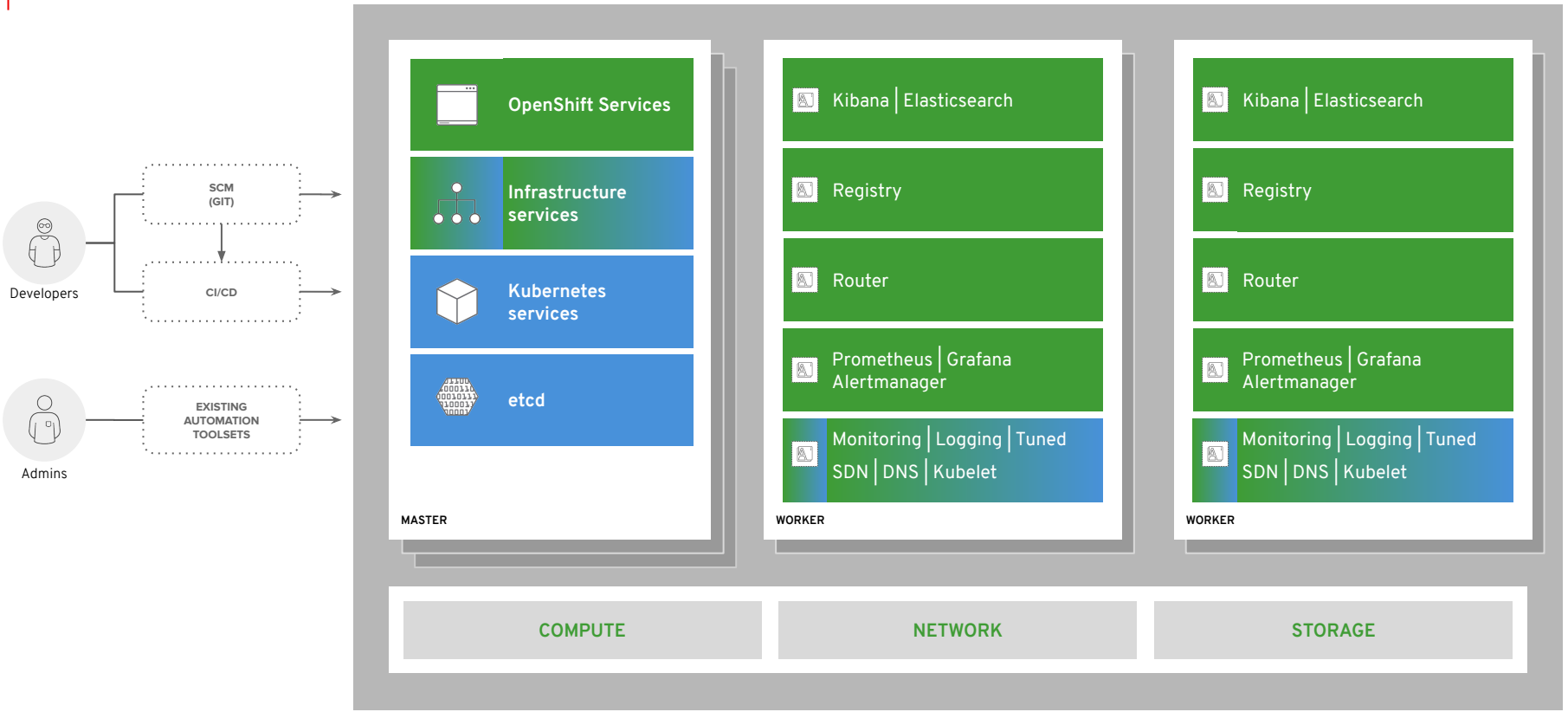

Self-Service
Multi-language
Automation
Collaboration
Multi-tenant
RED HAT
OPENSHIFT
Standards-based
Web-scale
Open Source
Enterprise Grade
Secure



Best IT Ops Experience

CaaS ⟷ PaaS ⟷ FaaS

Best Developer Experience


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

Red Hat

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

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

Greyed logos are not open source

App Definition and Development

Database

Streaming & Messaging

Application Definition & Image Build

Continuous Integration & Delivery

Orchestration & Management

Scheduling & Orchestration

Coordination & Service Discovery

Remote Procedure Call

Service Proxy

API Gateway

Service Mesh

Runtime

Cloud-Native Storage

Container Runtime

Cloud-Native Network

Provisioning

Automation & Configuration

Container Registry

Security & Compliance

Key Management

Cloud

Public

Special

Kubernetes Certified Service Provider

Platform

Certified Kubernetes - Distribution

Certified Kubernetes - Hosted

Certified Kubernetes - Installer

PaaS/Container Service

Kubernetes Training Partner

Observability and Analysis

Monitoring

Logging

Tracing

Chaos Engineering

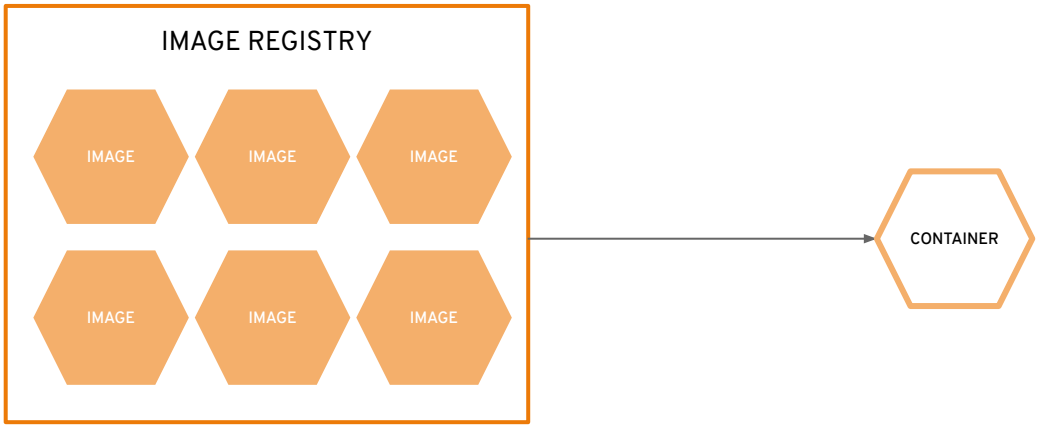
Serverless

Core Concepts

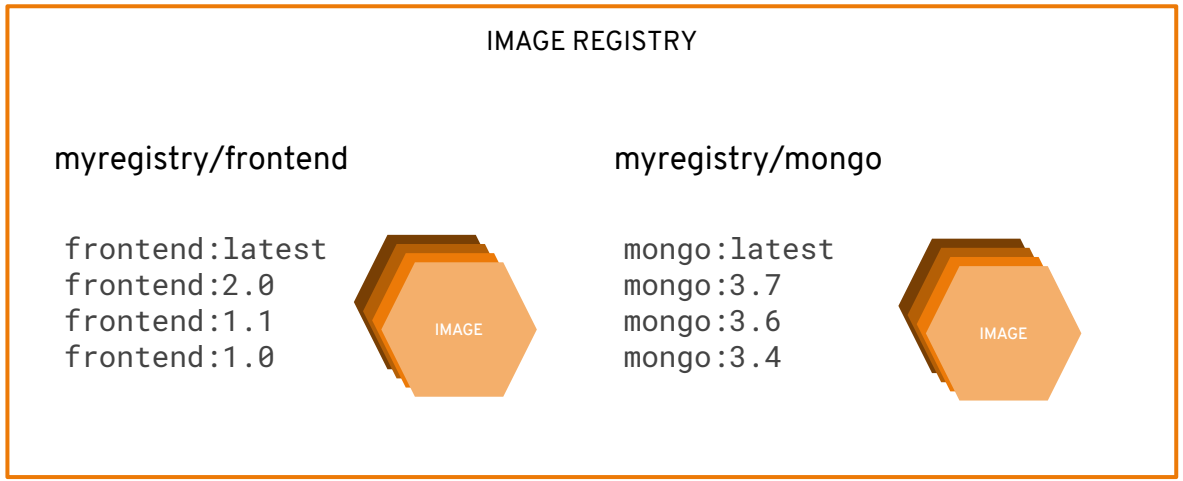
a container is the smallest compute unit

containers are created from container images

container images are stored in an image registry

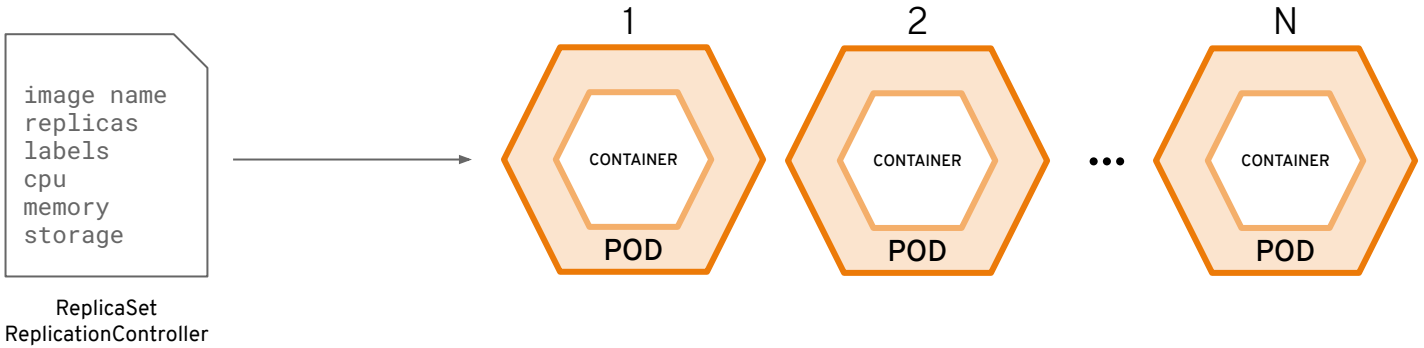


an image repository contains all versions of an image in the image registry

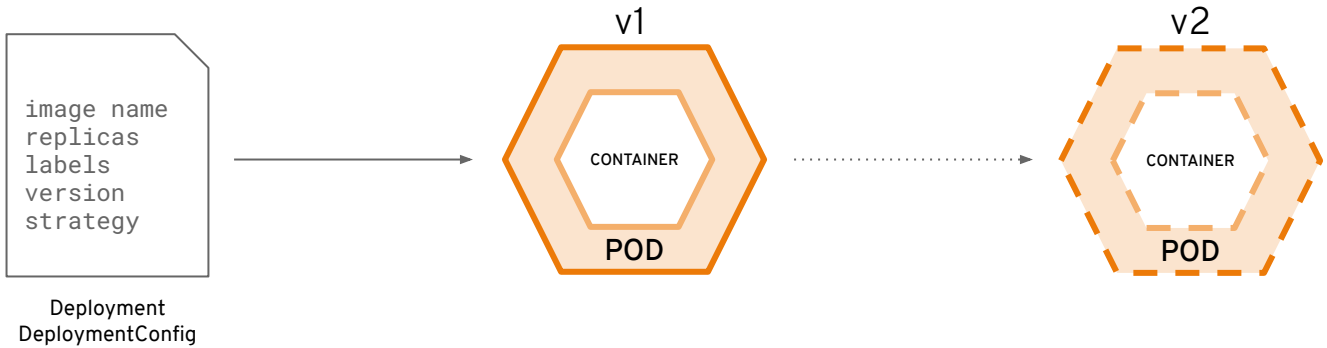


containers are wrapped in pods which are units of deployment and management

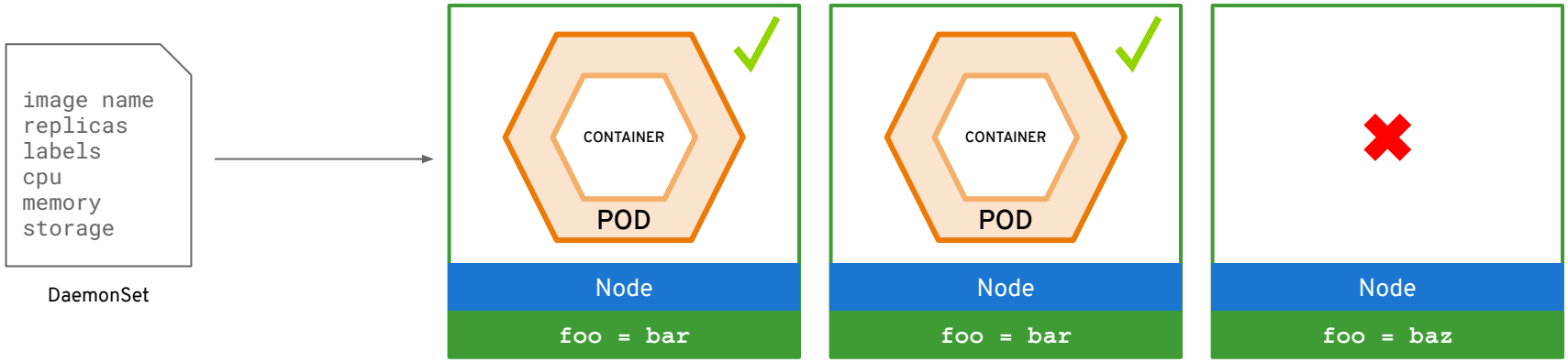
`ReplicationControllers` & `ReplicaSets` ensure a specified number of pods are running at any given time



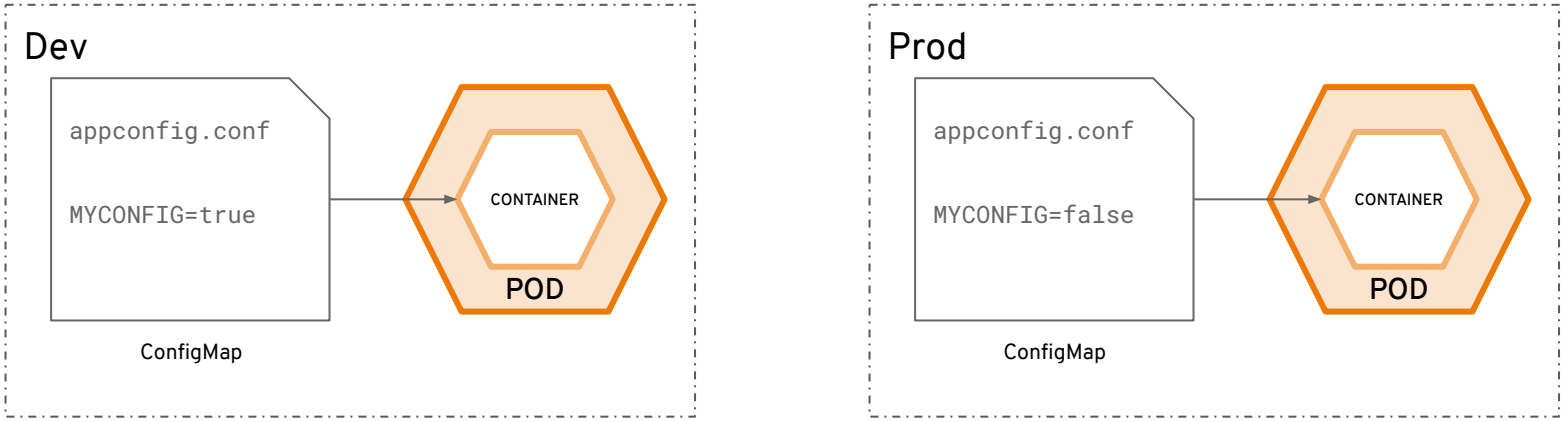
`Deployments` and `DeploymentConfigurations` define how to roll out new versions of Pods



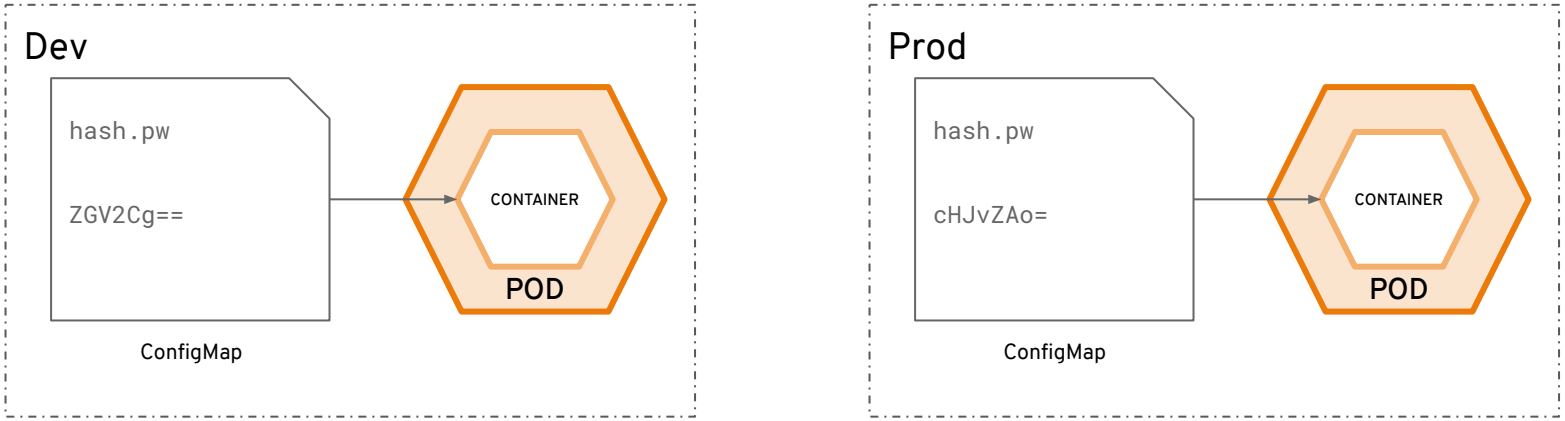
a daemonset ensures that all (or some) nodes run a copy of a pod



`configmaps` allow you to decouple configuration artifacts from image content

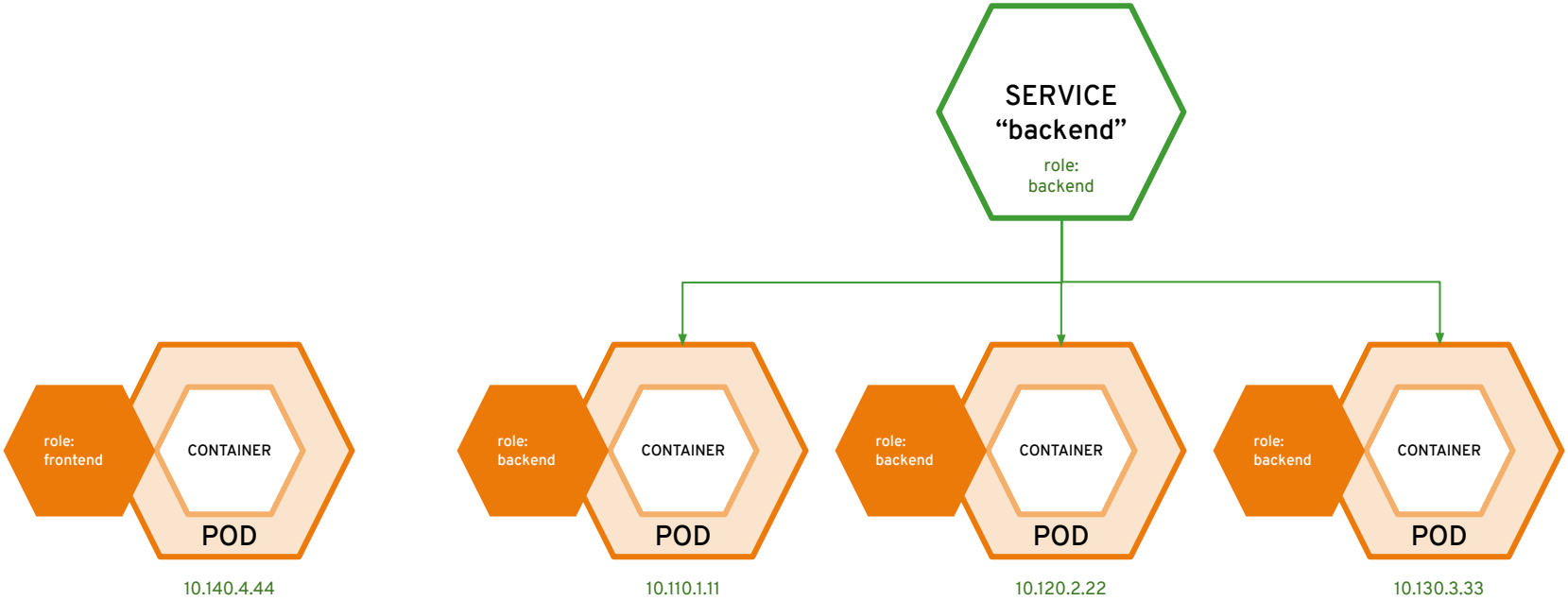


`secrets` provide a mechanism to hold sensitive information such as passwords

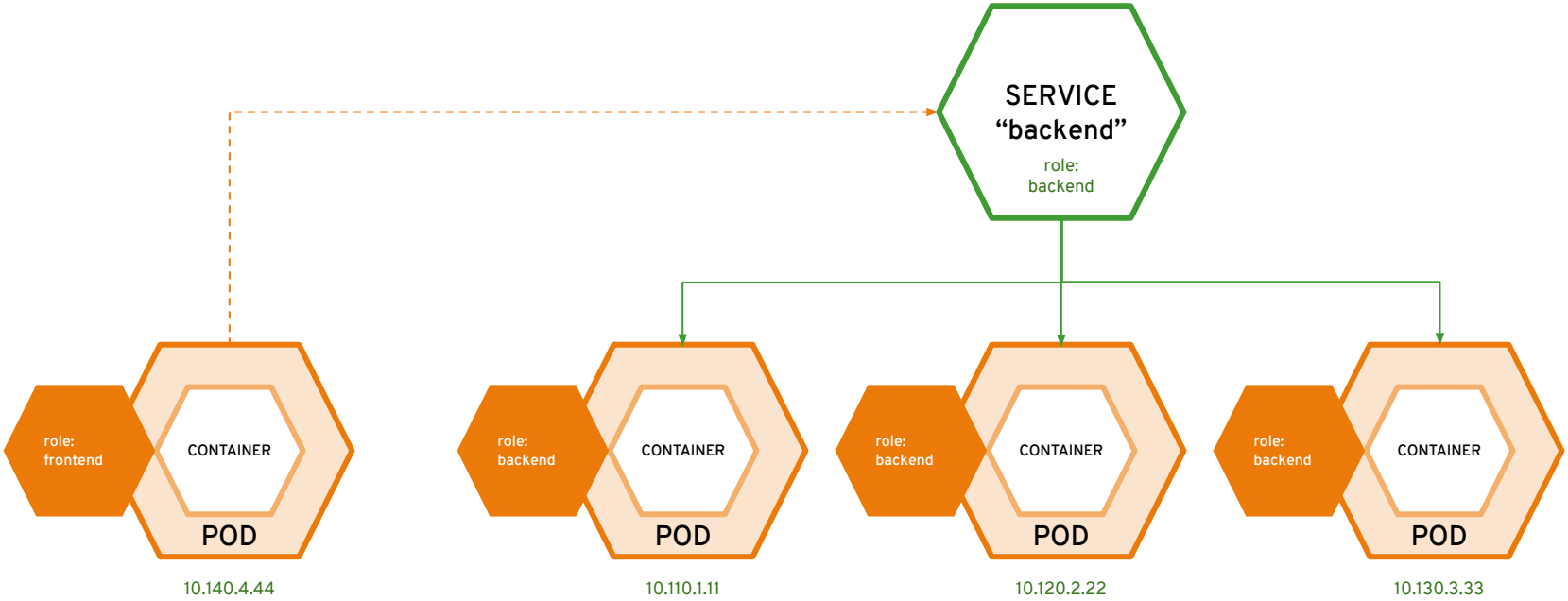


The etcd datastore can be encrypted for additional security
https://docs.openshift.com/container-platform/4.6/security/encrypting-etcd.html

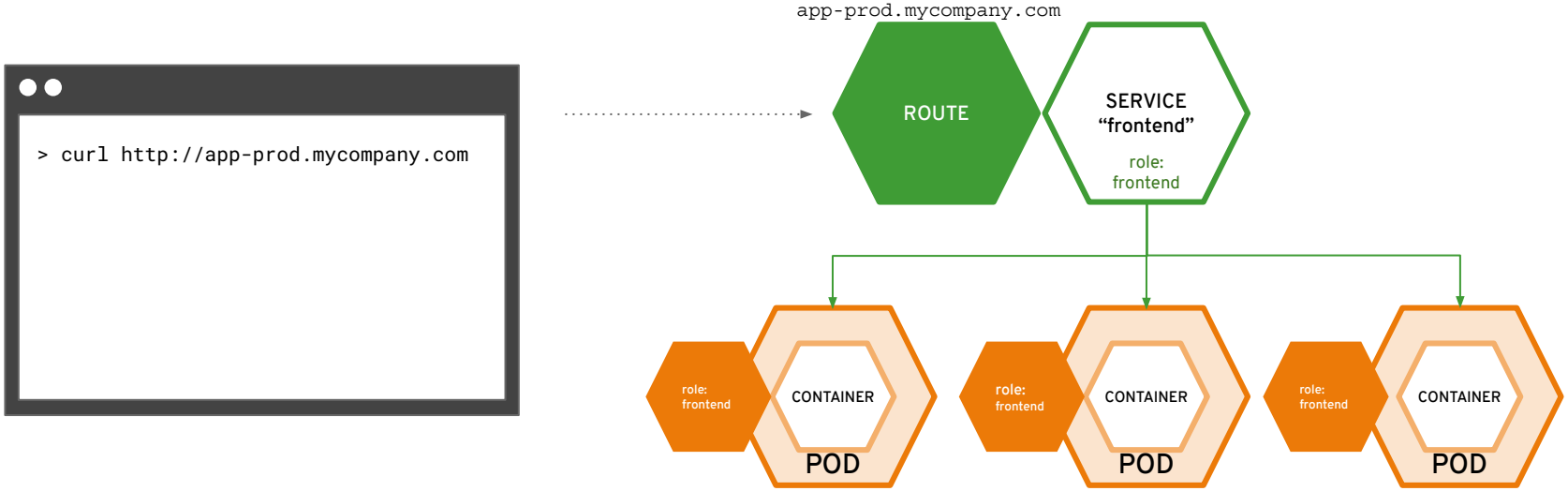
services provide internal load-balancing and service discovery across pods



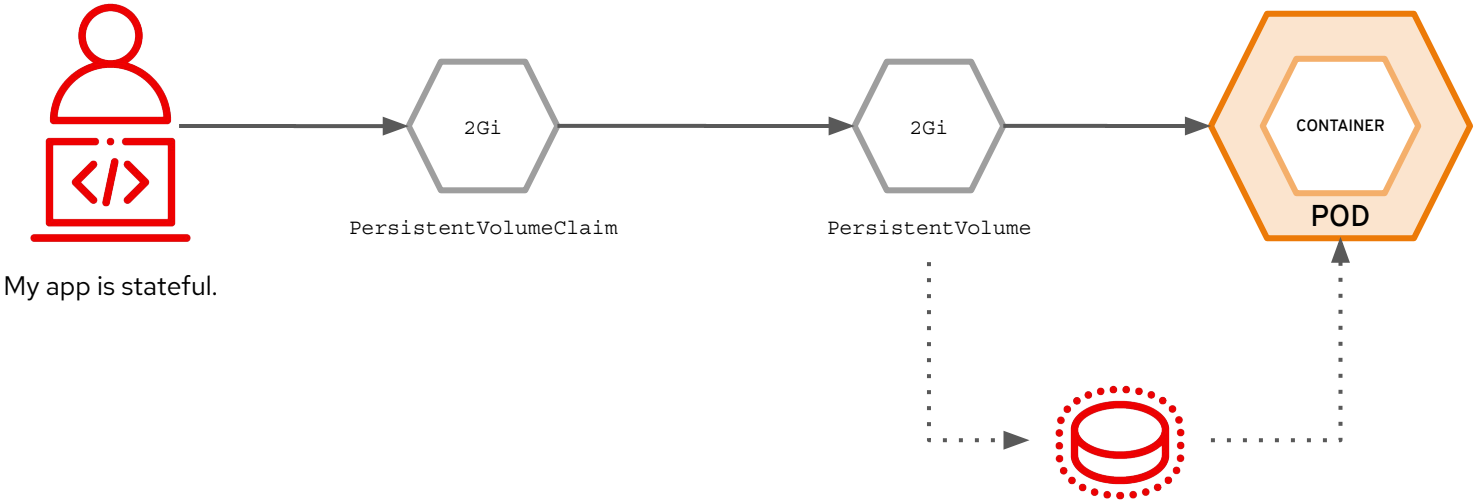
apps can talk to each other via services



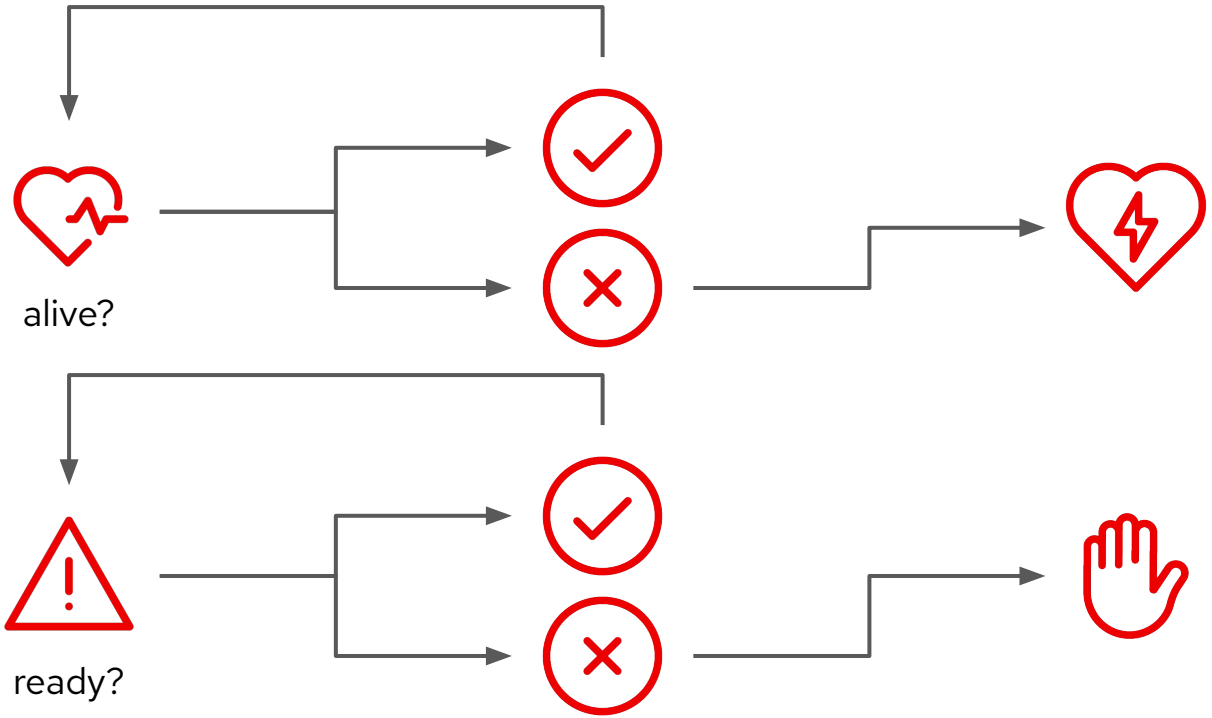
`routes` make services accessible to clients outside the environment via real-world urls



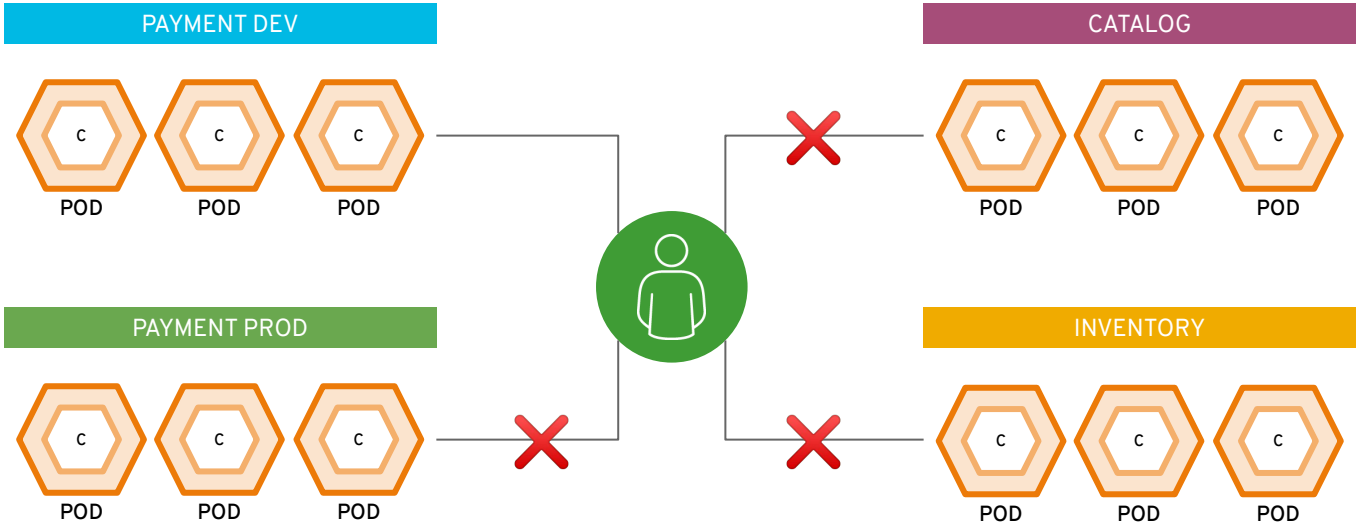
Persistent Volume and Claims



Liveness and Readiness



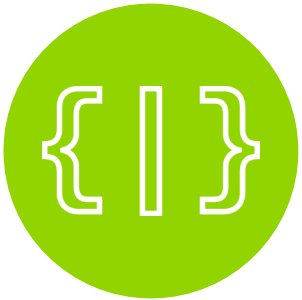
projects isolate apps across environments, teams, groups and departments



Red Hat

BUILD & DEPLOY

BUILD AND DEPLOY CONTAINER IMAGES

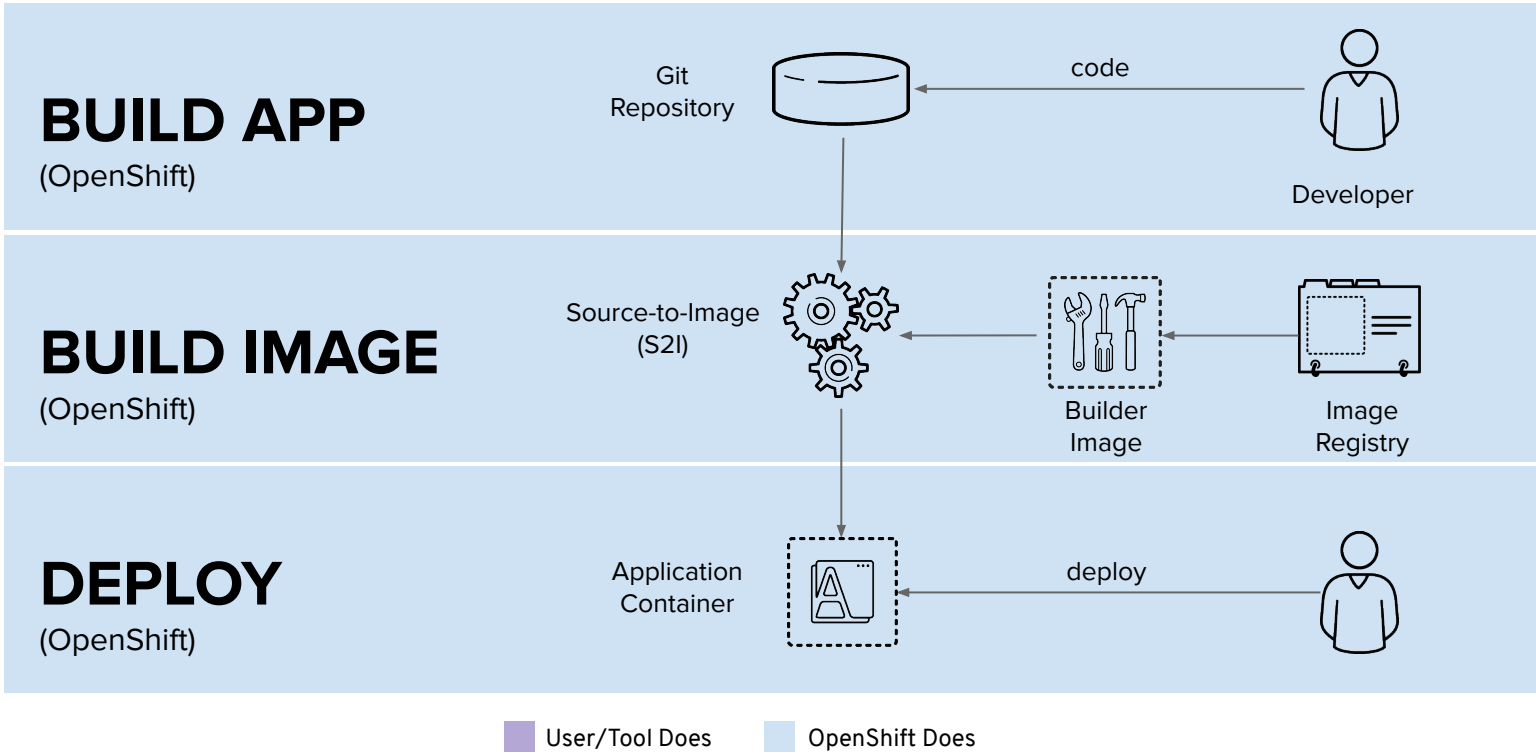


DEPLOY YOUR SOURCE CODE

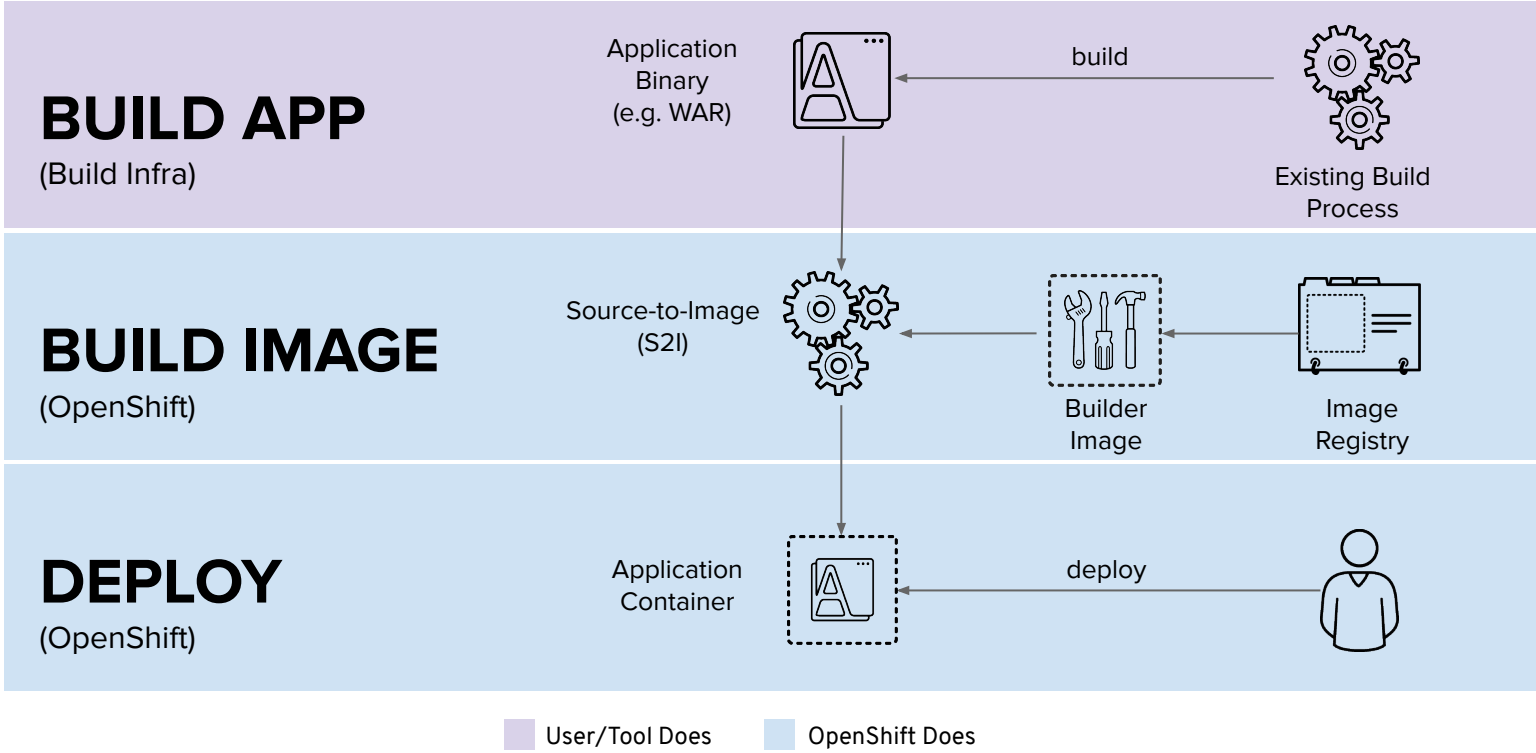
DEPLOY YOUR APP BINARY

DEPLOY YOUR CONTAINER IMAGE

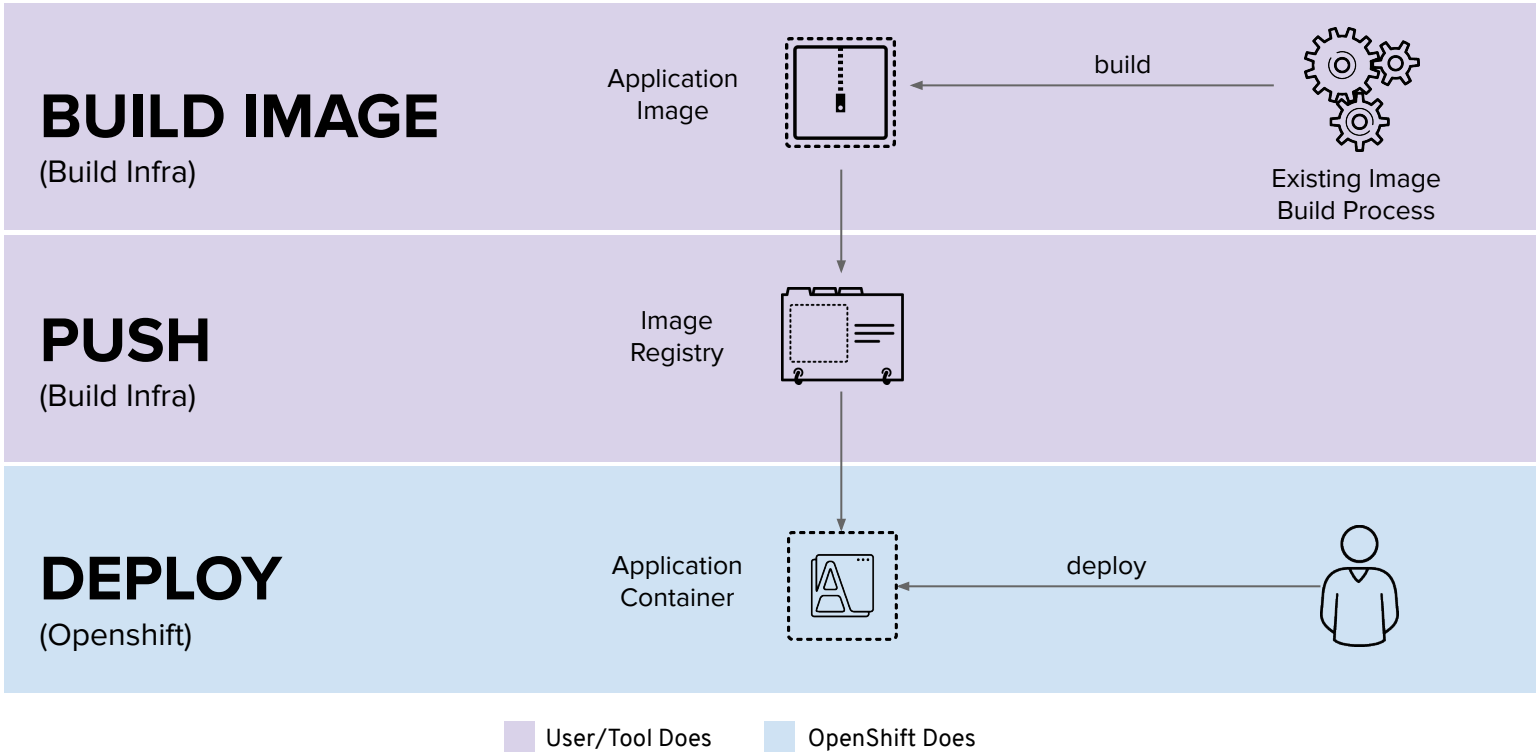
DEPLOY SOURCE CODE WITH SOURCE-TO-IMAGE (S2I)



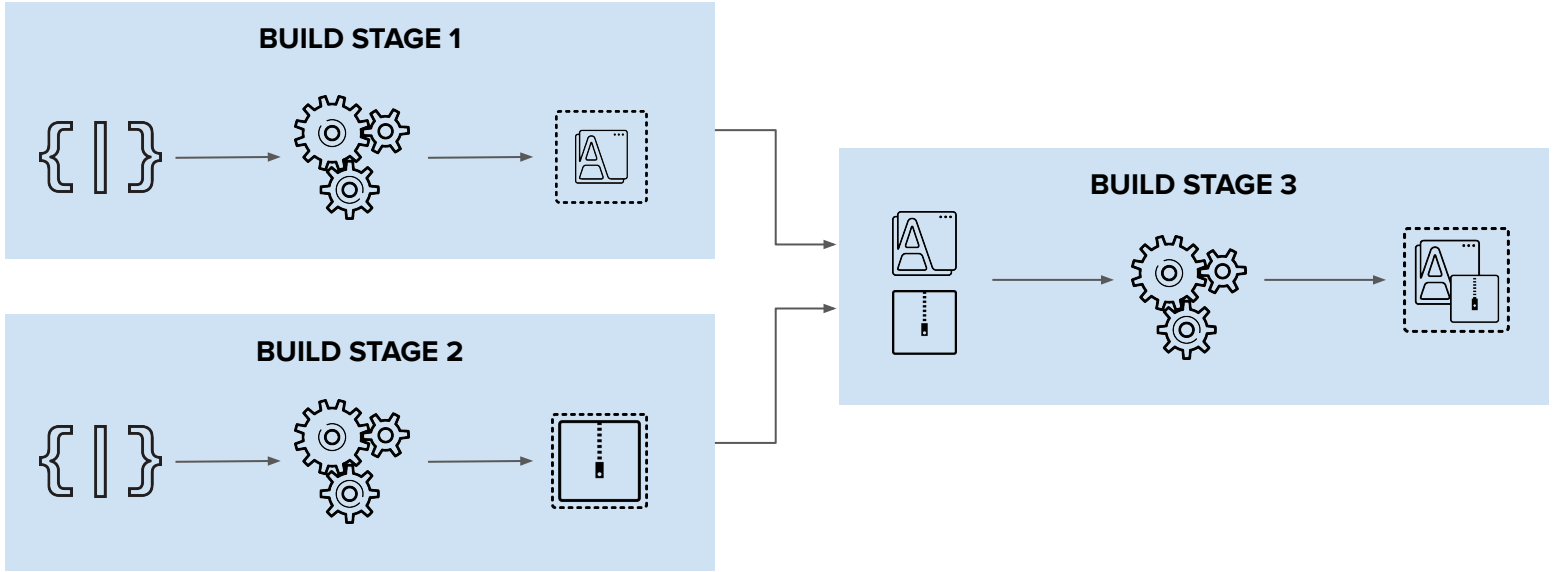
DEPLOY APP BINARY WITH SOURCE-TO-IMAGE (S2I)



DEPLOY DOCKER IMAGE

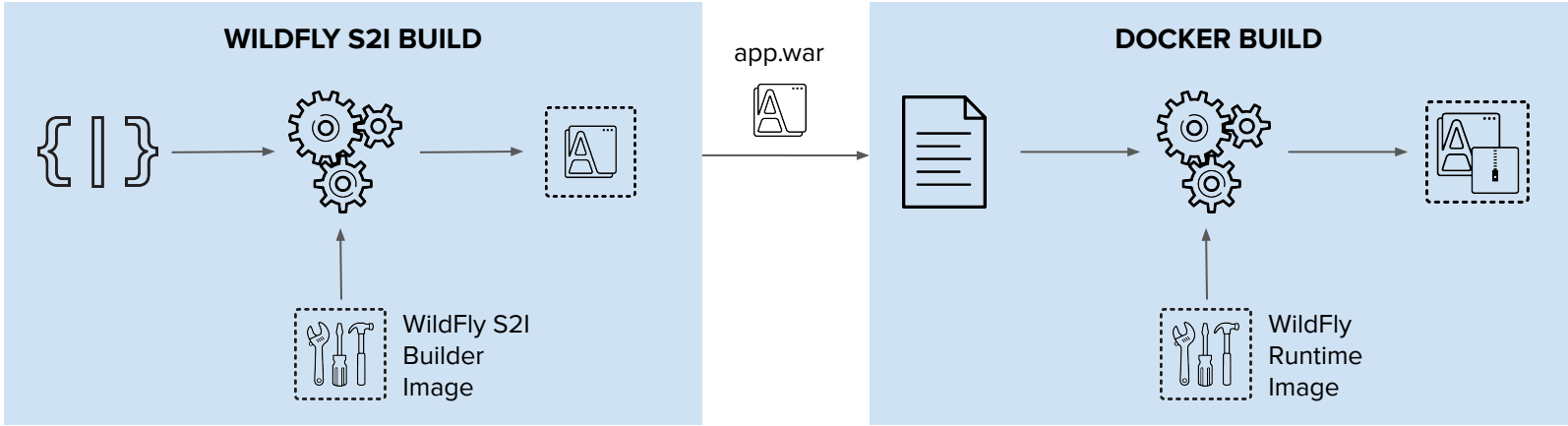


BUILD IMAGES IN MULTIPLE STAGES



EXAMPLE: USE ANY RUNTIME IMAGE WITH SOURCE-TO-IMAGE BUILDS

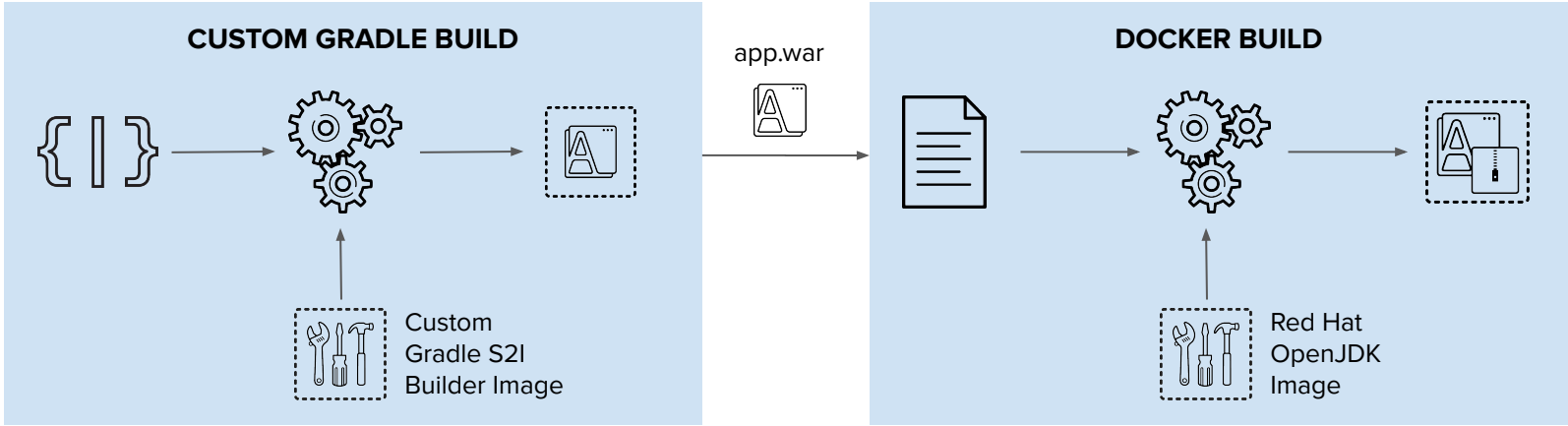
Use Source-to-Image to build app binaries and deploy on lean vanilla runtimes



read more on https://blog.openshift.com/chaining-builds/

EXAMPLE: USE ANY BUILD TOOL WITH OFFICIAL RUNTIME IMAGES

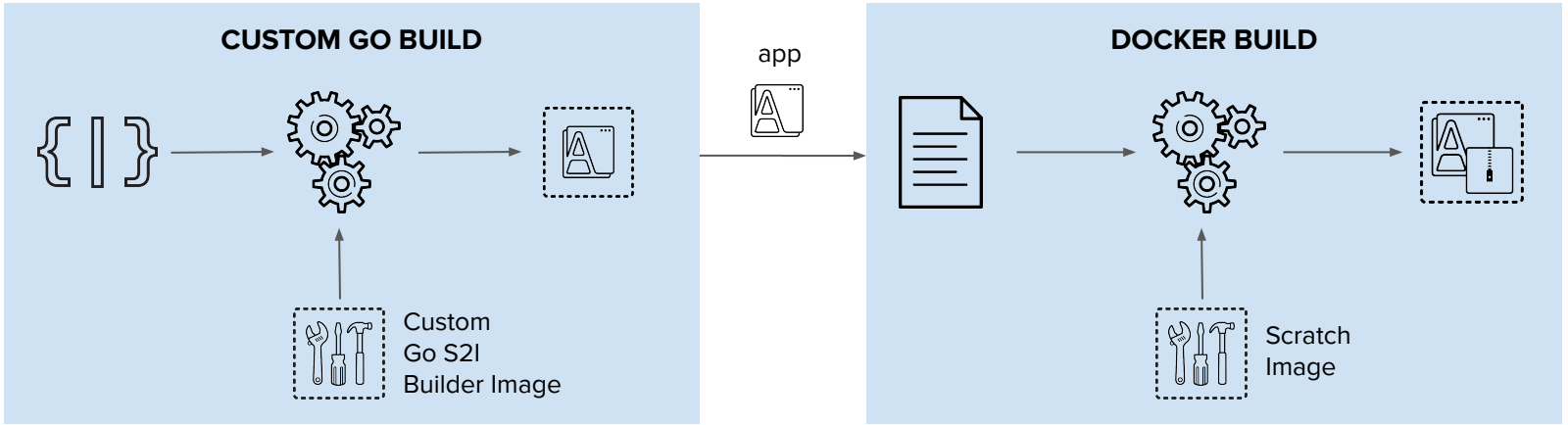
Use your choice of build tool like Gradle and deploy to official images like the JDK image



read more on https://blog.openshift.com/chaining-builds/

EXAMPLE: SMALL LEAN RUNTIMES

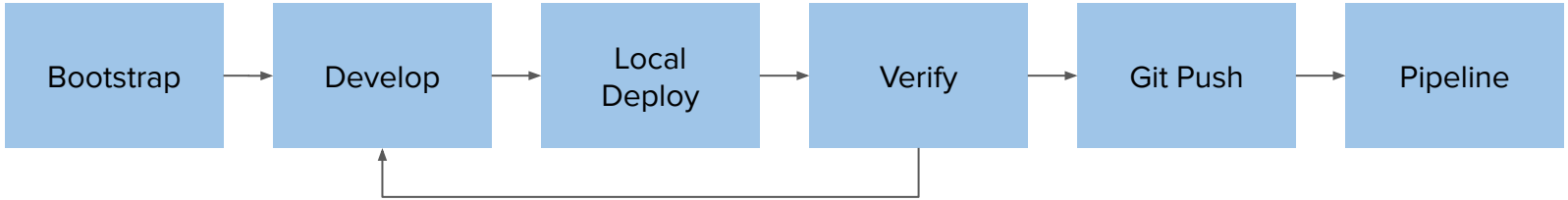
Build the app binary and deploy on small scratch images



read more on https://blog.openshift.com/chaining-builds/

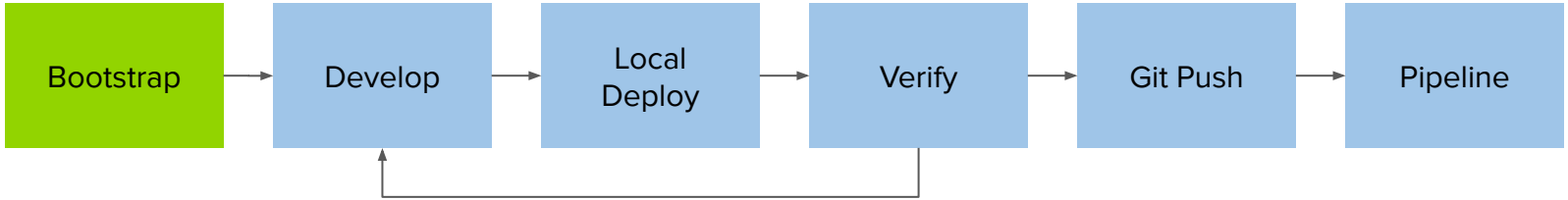
DEVELOPER WORKFLOW

LOCAL DEVELOPMENT WORKFLOW



RED HAT®
OPENSHIFT
Container Platform

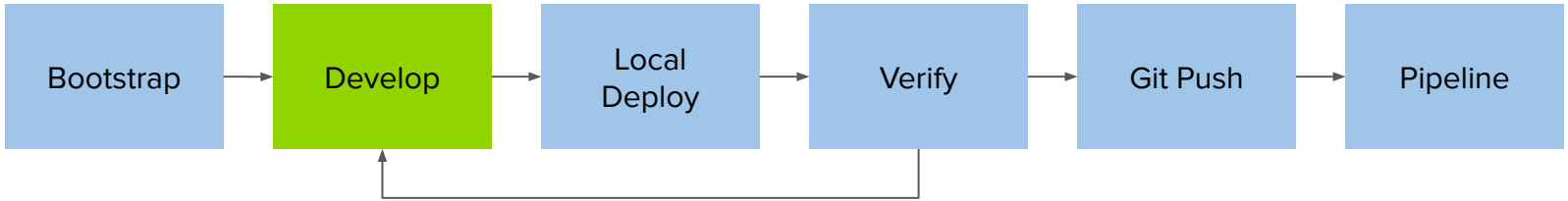
LOCAL DEVELOPMENT WORKFLOW



BOOTSTRAP

- Pick your programming language and application runtime of choice
- Create the project skeleton from scratch or use a generator such as
  - Maven archetypes
  - Quickstarts and Templates
  - OpenShift Generator

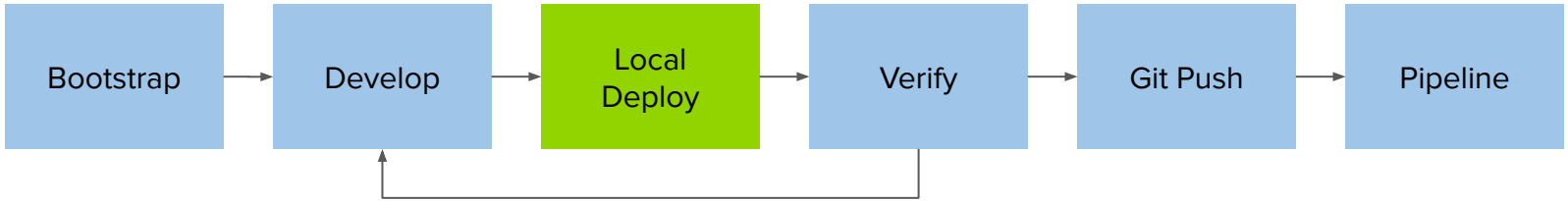
LOCAL DEVELOPMENT WORKFLOW



DEVELOP

- Pick your framework of choice such as Java EE, Spring, Ruby on Rails, Django, Express, ...
- Develop your application code using your editor or IDE of choice
- Build and test your application code locally using your build tools
- Create or generate OpenShift templates or Kubernetes objects

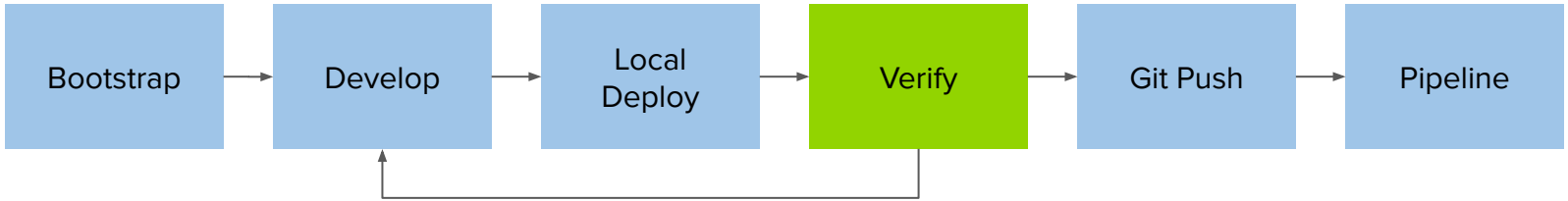
LOCAL DEVELOPMENT WORKFLOW



LOCAL DEPLOY

- Deploy your code on a local OpenShift cluster
  - Red Hat Container Development Kit (CDK), minishift and oc cluster
- Red Hat CDK provides a standard RHEL-based development environment
- Use binary deploy, maven or CLI rsync to push code or app binary directly into containers

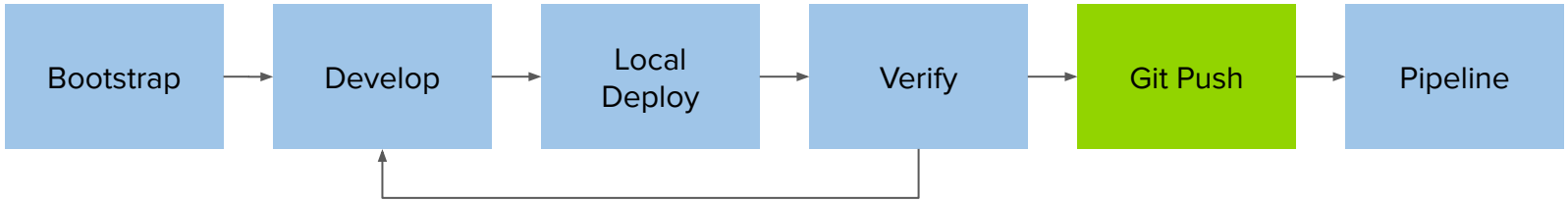
LOCAL DEVELOPMENT WORKFLOW



VERIFY

- Verify your code is working as expected
- Run any type of tests that are required with or without other components (database, etc)
- Based on the test results, change code, deploy, verify and repeat

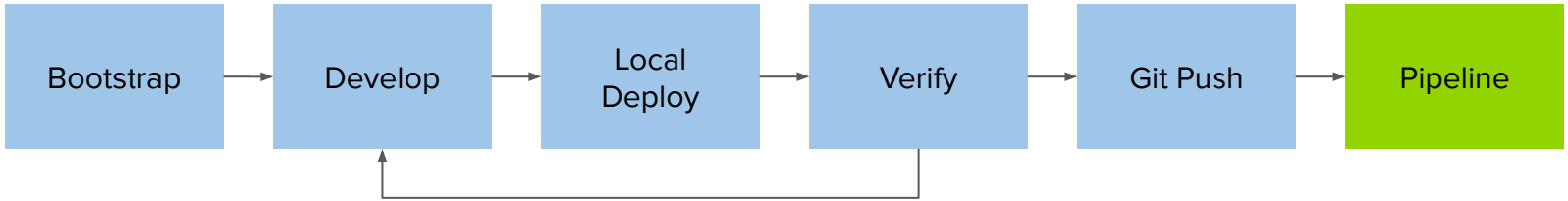
LOCAL DEVELOPMENT WORKFLOW



GIT PUSH

- Push the code and configuration to the Git repository
- If using Fork & Pull Request workflow, create a Pull Request
- If using code review workflow, participate in code review discussions

LOCAL DEVELOPMENT WORKFLOW

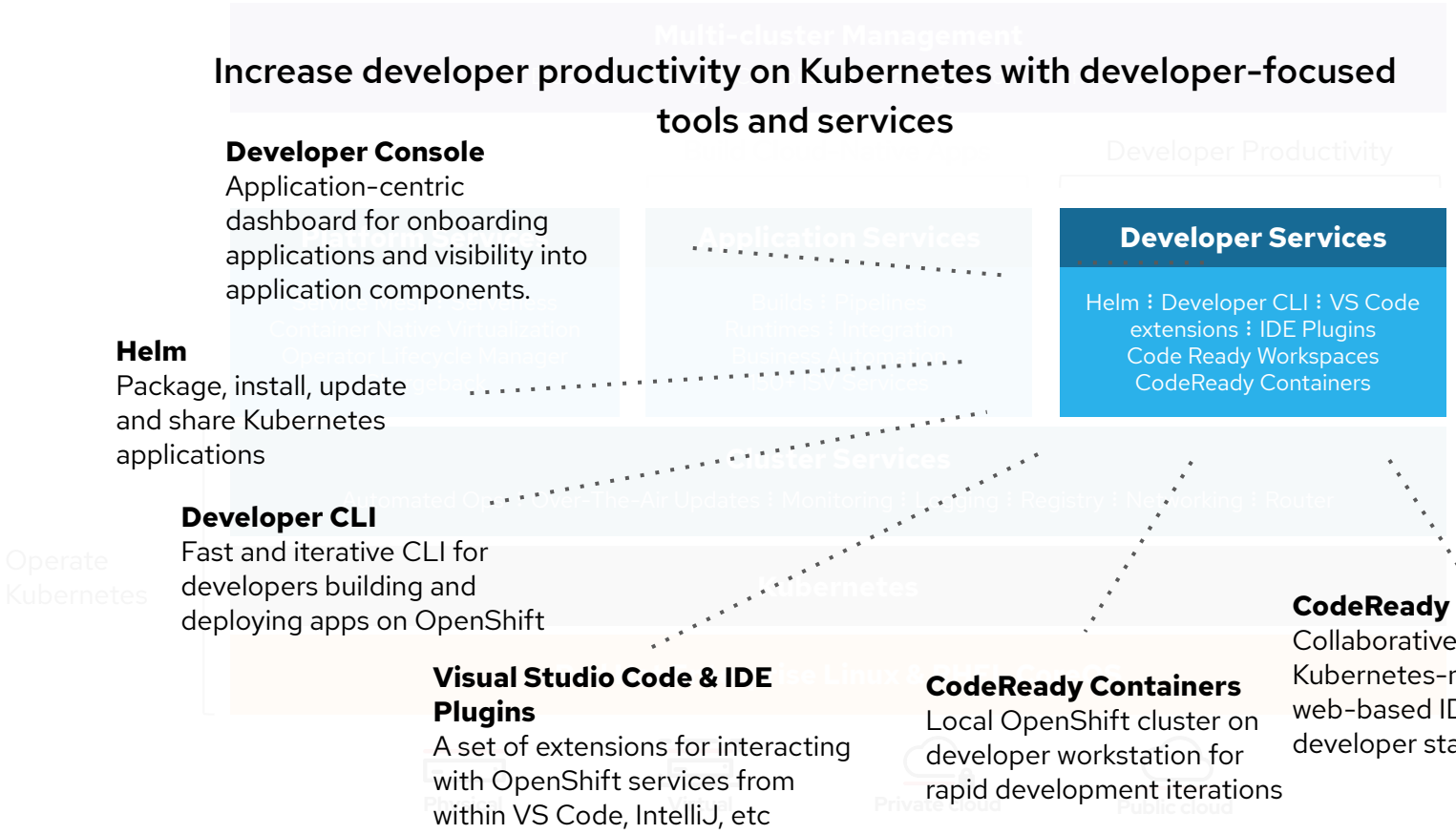


PIPELINE

- Pushing code to the Git repository triggers one or multiple deployment pipelines
- Design your pipelines based on your development workflow e.g. test the pull request
- Failure in the pipeline? Go back to the code and start again

OpenShift Developer Ecosystem

OpenShift Container Platform

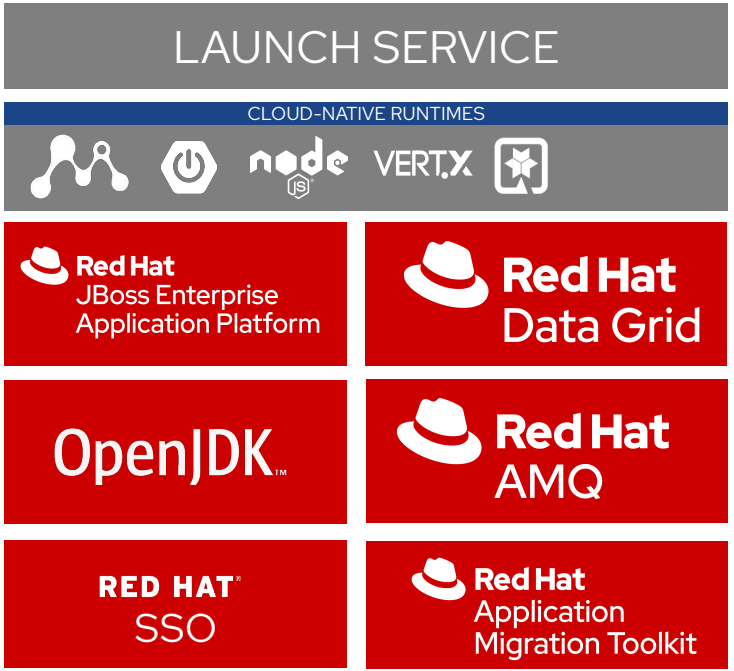


Red Hat

Red Hat Runtimes

- **Quarkus GA** - Supersonic, Subatomic Java. Native support coming in Q3 via Mandrel.
- **Red Hat SSO** - Support for OpenShift secret Vaults, WebAuthn protocol.
- **Data Grid 8.1** - Cross-site cluster support
- **Z-Series support for Runtimes** - Use existing Z & Cloud Pak investment with Runtimes & OpenShift.
- **JBOSS EAP** expansion pack for MicroProfile
- **Spring Boot 2.2** - New AMQ Starters, GA of Reactive support and Kubernetes Java annotations.

[What's New in Red Hat Runtimes [Q2]]

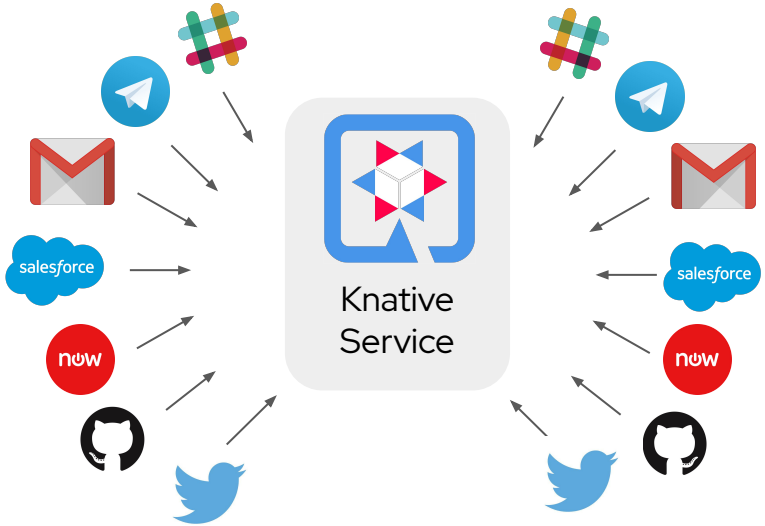


Product Manager: Karena Angell (on behalf of the Red Hat Application Services team)

Red Hat

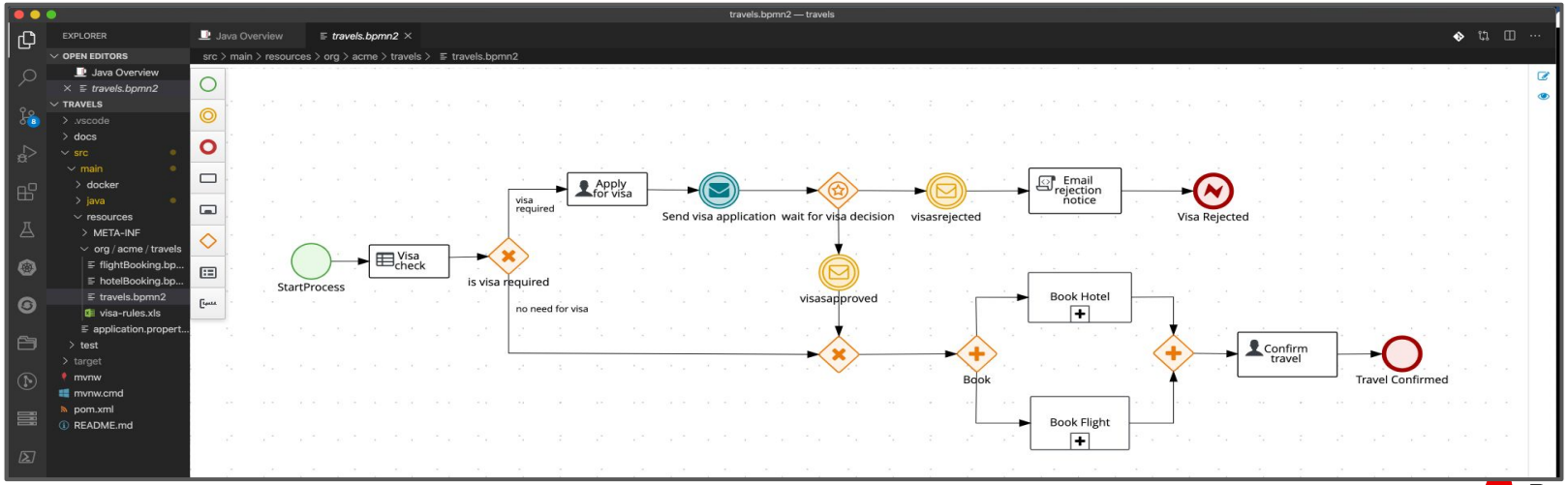
Red Hat Integration

- **Service Registry GA** - schema and API registry to support Kafka and API-based workloads on OpenShift.
- **Mirror Maker 2.0** - major evolution in data replication for Kafka on OpenShift now at full support in Strimzi
- **Camel K for Serverless (TP)** - leverage the huge Camel connector catalog to drive events into your OpenShift Serverless applications based on Camel K and Knative Eventing.



Red Hat Process Automation

- **Kogito -** moves to developer preview for next-gen business automation based on Quarkus
- **Trusty AI -** first MVP of Trusty AI in the next Kogito developer preview release. With runtime dashboards and metrics.
- **Red Hat Business Automation Bundle -** released in the Visual Studio Code Marketplace as Developer Preview



Product Manager: Karena Angell (on behalf of the Red Hat Application Services team)

Red Hat

DEMO

Outlook on week #2

Outlook

- After you've developed your App, you need to find a way to package and distribute it
- We are going to have a look on
  - Templates
  - Helm 3
  - Operators
- Kick-Off Webinar will be upcoming Monday, 7th of June. We are going to start at 10:00 am CEST

THANK YOU