



# Red Hat Day Events

January 30, Vancouver

**OpenShift**  
**The Platform for Big Ideas**



**Red Hat**

# DAY-IN-THE-LIFE OF A DEVELOPER with OPENSIFT 4

**Presenter:** Veer Muchandi  
**Title:** Chief Architect - Container Solutions  
**Social Handle:** @VeerMuchandi  
**Blogs:** <https://blog.openshift.com/author/veermuchandi/>

#OpenShift4  
#RedHatEvents

Jan 2020



# Kubernetes Dashboard

Plain Kubernetes

The screenshot displays the Kubernetes Dashboard interface. At the top, the browser address bar shows the URL: `localhost:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/#!/pod?namespace=kube-system`. The dashboard header includes the Kubernetes logo, a search bar, and a '+ CREATE' button. The left sidebar contains navigation options: Nodes, Persistent Volumes, Roles, Storage Classes, Namespace (set to 'kube-system'), Overview, Workloads (Cron Jobs, Daemon Sets, Deployments, Jobs, Pods, Replica Sets, Replication Controllers, Stateful Sets), and Discovery and Load Balancing. The main content area features two line graphs: 'CPU usage' and 'Memory usage', both showing data from 11:10 to 11:24. Below the graphs is a table of running pods.

Name	Node	Status	Restarts	Age	CPU (cores)	Memory (bytes)
✓ <a href="#">kubernetes-dashboard-7b9c7b</a>	minikube	Running	0	27 minutes	0	19.746 Mi
✓ <a href="#">heapster-qhq6r</a>	minikube	Running	0	27 minutes	0	18.004 Mi
✓ <a href="#">influxdb-grafana-77c7p</a>	minikube	Running	0	27 minutes	0	43.926 Mi
✓ <a href="#">kube-scheduler-minikube</a>	minikube	Running	0	20 hours	0.01	11.930 Mi
✓ <a href="#">etcd-minikube</a>	minikube	Running	0	20 hours	0.015	58.445 Mi

Does a developer need to know about nodes, cluster capacity etc?

# Kubernetes Deployments

Plain Kubernetes

```
kind: Deployment
apiVersion: extensions/v1beta1
metadata:
  name: hostname-101-deployment
spec:
  replicas: 3
  selector:
    # Like saying "Make sure there are three pods running
    # with the label app = hostname and version = v101"
    matchLabels:
      app: hostname
      version: v101
  template:
    metadata:
      labels:
        # The `app` label is used by both the service
        # and the deployment to select the pods they operate on.
        app: hostname
        # The `version` label is used only by the deployment
        # to control replication.
        version: v101
    spec:
      containers:
        - name: nginx-hostname
          image: kubegoldenguide/nginx-hostname:1.0.1
          ports:
            - containerPort: 80
```

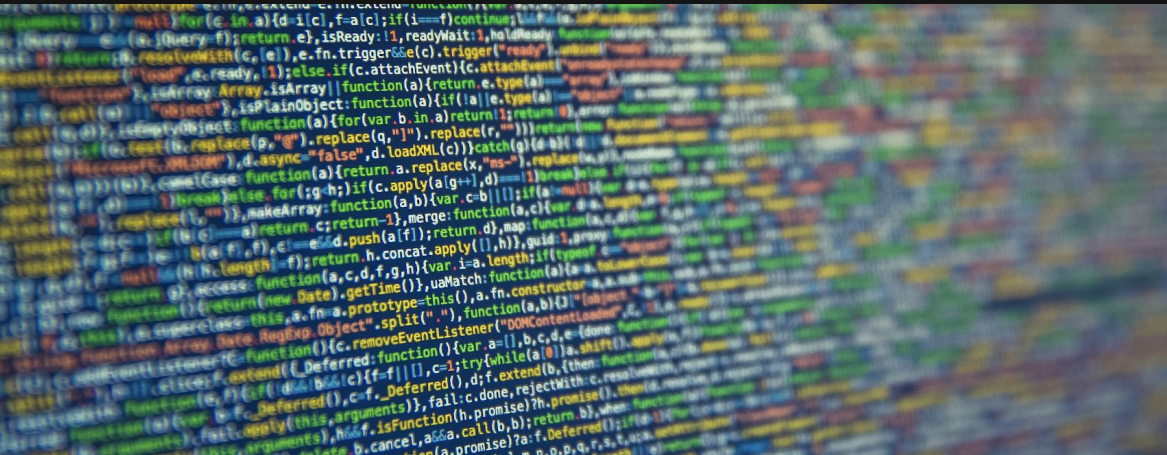
Image Acknowledgements: matthewpalmer.net

# OpenShift Developer focused clients

Web

CLI

IDE



# Developer focused OpenShift Console

Developer focussed clients

The screenshot displays the Red Hat OpenShift Developer Console interface. The top navigation bar includes the Red Hat logo and 'OpenShift Container Platform'. The left sidebar shows navigation options: 'Developer', '+Add', 'Topology' (selected), 'Builds', 'Pipelines', 'Monitoring', and 'Advanced'. The main content area shows the 'Project: bookinfo' and 'Application: all applications'. A 'Display' dropdown is set to '4'. A search bar labeled 'Filter name...' is present. The central part of the screen features a service topology diagram with nodes: 'productpage-v1' (Python icon), 'details-v1' (Ruby icon), 'reviews-v1' (Cartoon character icon), 'reviews-v2' (Cartoon character icon), 'reviews-v3' (Cartoon character icon), and 'ratings-v1' (Node.js icon). Arrows indicate dependencies: 'productpage-v1' depends on 'details-v1', 'reviews-v1', 'reviews-v2', and 'reviews-v3'. 'reviews-v1', 'reviews-v2', and 'reviews-v3' all depend on 'ratings-v1'. A toolbar at the bottom of the diagram includes icons for search, zoom, and other actions.

6

Gives your developers a “home” in OpenShift Kubernetes platform.



# Administrator Console

Developer focused clients

The screenshot displays the Red Hat OpenShift Administrator Console interface. The left sidebar contains navigation options: Administrator, Home, Dashboards, Projects, Search, Explore, Events, Operators, Workloads, Pods, Deployments, Deployment Configs, Stateful Sets, Secrets, Config Maps, Cron Jobs, Jobs, Daemon Sets, Replica Sets, Replication Controllers, Horizontal Pod Autoscalers, Serverless, Networking, Storage, Builds, Service Catalog, Monitoring, Compute, and Nodes.

The main content area is titled "Dashboards" and includes the following sections:

- Details:** Cluster ID: ecd0dd01-dd84-45d7-ae83-73f193f04815, Provider: None, OpenShift Version: 4.2.13.
- Cluster Health:** Cluster is healthy.
- Alerts:** There are 2 different semantic versions of Kubernetes components running.
- Cluster Inventory:** 8 Nodes (8/8), 390 Pods (378/12), 25 PVCs (25/25).
- Cluster Capacity:** CPU (22% used), Memory (32% used), Storage (57% used), Network (0% used).
- Cluster Utilization:** Line chart showing CPU, Memory, and Disk Usage over time.
- Events:** List of recent events, including successful updates and container starts.
- Top Consumers:** Pods by CPU consumption, with kube-apiserver-master1.ocp4.home.ocpcloud.com at the top.

7

Administrators still have their console, to manage the cluster!!



## odo - OpenShift's Dev-Focused CLI

A developer-focused command-line tool for rapid development iterations on OpenShift (inner loop).

Simplifies building of microservices applications on OpenShift.

```
$ odo create wildfly backend
Component 'backend' was created.

$ odo push
Pushing changes to component: 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-myapp.192.168.99.100.nip.io

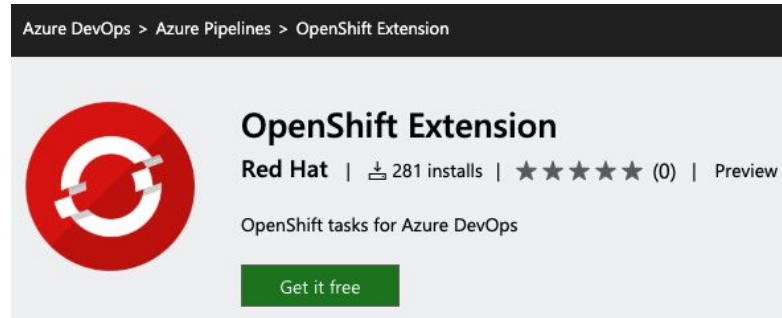
$ odo watch
Waiting for something to change in /dev/frontend
```




# IDEs: OpenShift Deploy Plugins

Red Hat has created plugins to simplify development and deployment to OpenShift from popular IDEs and DevOps Toolchains:

- VS Code
- JetBrains IDEs (e.g. IntelliJ)
- Azure DevOps
- CodeReady Workspaces
- Eclipse IDE



Azure DevOps > Azure Pipelines > OpenShift Extension

 **OpenShift Extension**  
Red Hat | 281 installs | ★★★★★ (0) | Preview

OpenShift tasks for Azure DevOps


[Get it free](#)



 **OpenShift Connector**  
by Red Hat



Visual Studio Code > Other > OpenShift Connector

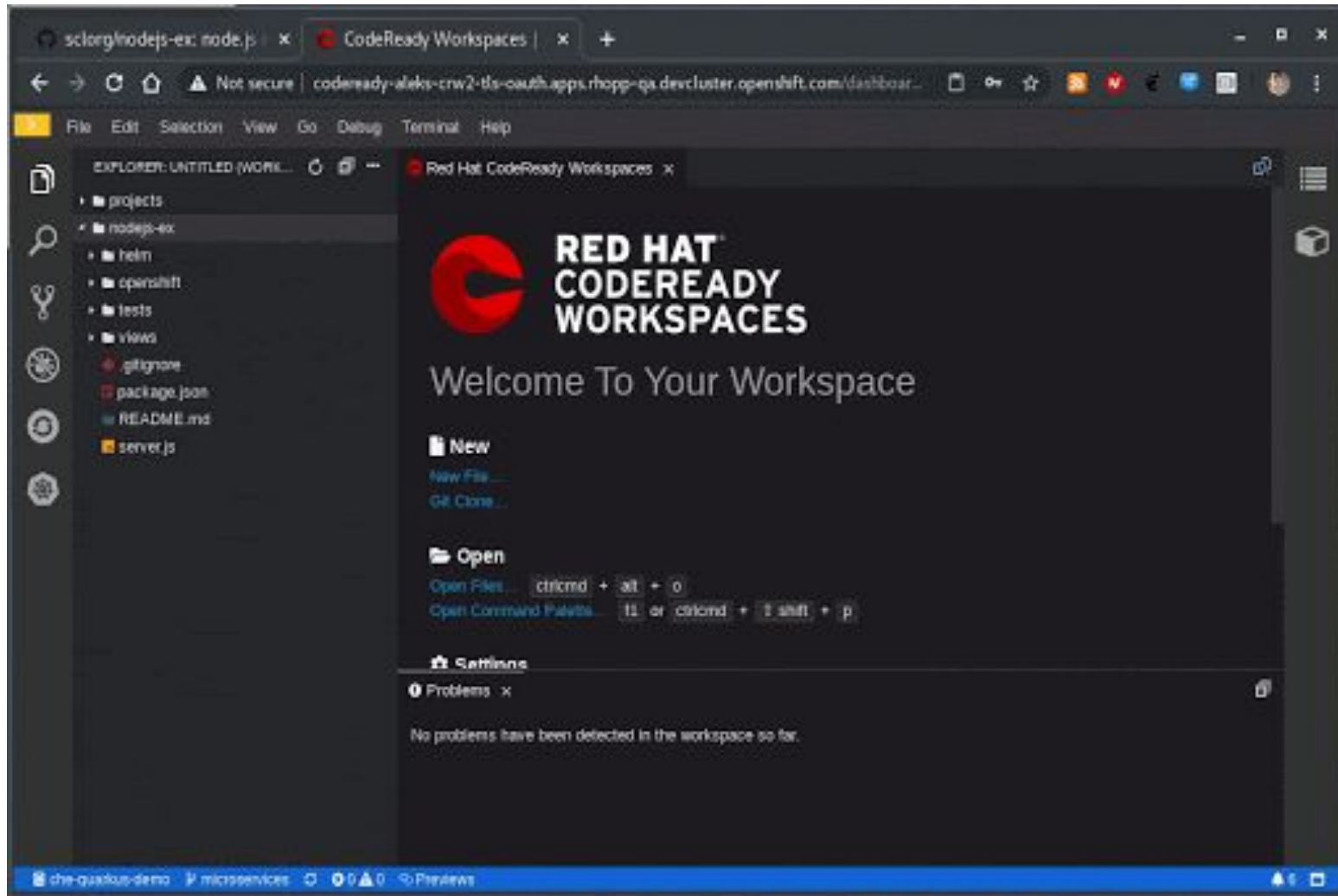
 **OpenShift Connector** Preview  
Red Hat | 7,589 installs | 22,799 downloads | ★★★★★ (2) | Free

Interacting with Red Hat OpenShift clusters and providing a streamlined developer experience using Visual Studio Code

[Install](#) [Trouble Installing?](#)

# Code Ready Workspaces

Developer focused clients



## CodeReady Workspaces 2.x

- Workspace includes runtimes, project files, source code and Cloud native IDE
- Based on Eclipse Che 7
- IDE uses VSCode plugins
- Native integration with OpenShift
- Build and deploy locally and to OpenShift

# CodeReady Containers

*OpenShift on your Laptop*

```
$ crc setup
Prepare your machine for running OpenShift

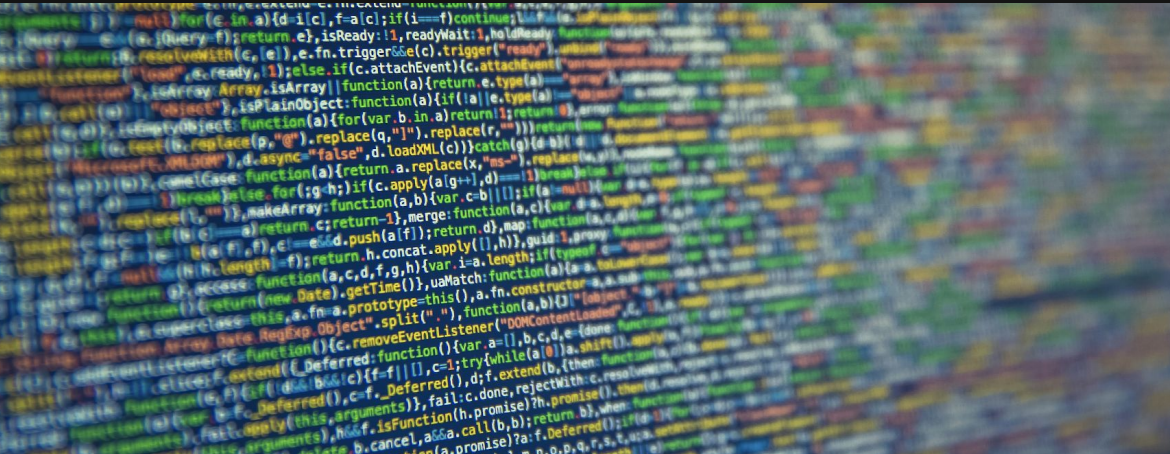
$ crc start -b crc-hyperkit-4.2.0.crcbundle
Start with the Hyperkit 4.2 bundle

$ crc status
Get the status of the cluster
```

Provides a pre-built development environment based on **Red Hat Enterprise Linux** and **OpenShift** for quick container-based application development. Use with OpenShift on-premises or cloud.

- Based on OpenShift 4.x
- Linux (libvirt)
- Windows (Hyper-V)
- MacOS (Virtualbox)

# OpenShift Serverless

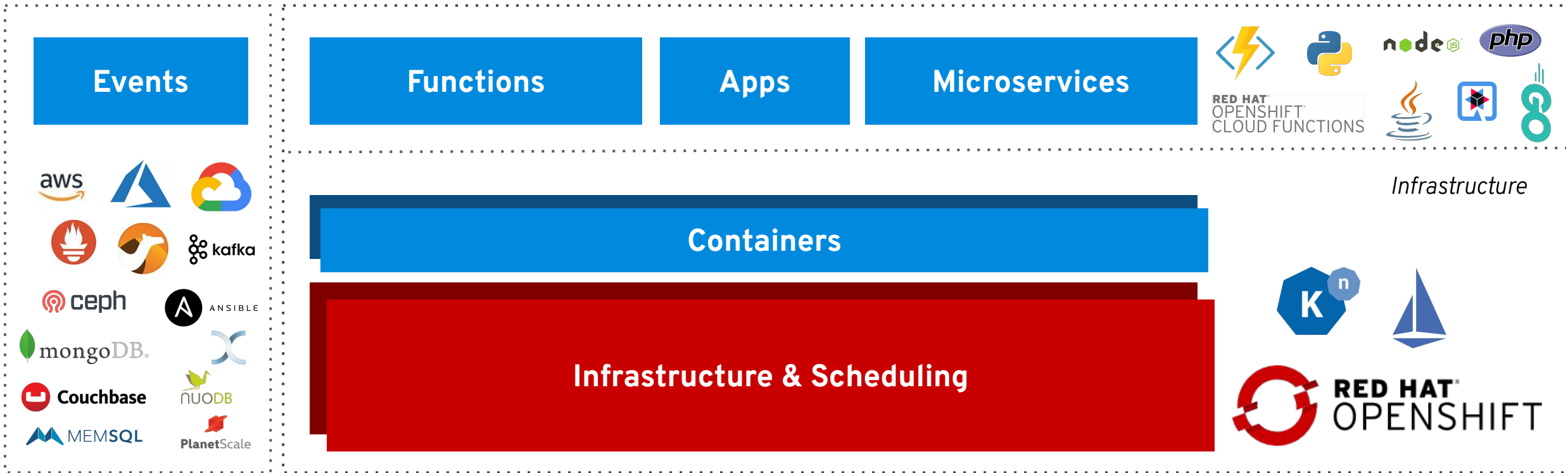


OpenShift Serverless helps developers deploy and run applications that will scale up or scale to zero on-demand.

Applications are packaged as OCI compliant Linux containers that can be run anywhere.

**Tech Preview on OCP 4.2**

# Microservices, Functions and Apps + Events



OperatorHub.io

## Knative Client (*kn*)

Create a new service with 1 instance running all the time (no scale to zero) and limiting memory consumption

```
kn service create myService --image=.. --min-scale=1 --max-scale=100 --limits-memory=100m
```

Update a service with multiple Revisions to send 50% of traffic to each version

```
kn service update myService --traffic myService-rev1=50,myService-rev2=50
```

Update a service with multiple Revisions to send 10% of traffic while 90% goes to Revision 2.

```
kn service update myService --traffic myService-rev1=10,myService-rev2=90
```

For more <https://github.com/knative/client>

# Knative Serverless on Console

OpenShift Serverless

Select the resource type to generate

- Deployment

apps/Deployment

A Deployment enables declarative updates for Pods and ReplicaSets.

- Deployment Config

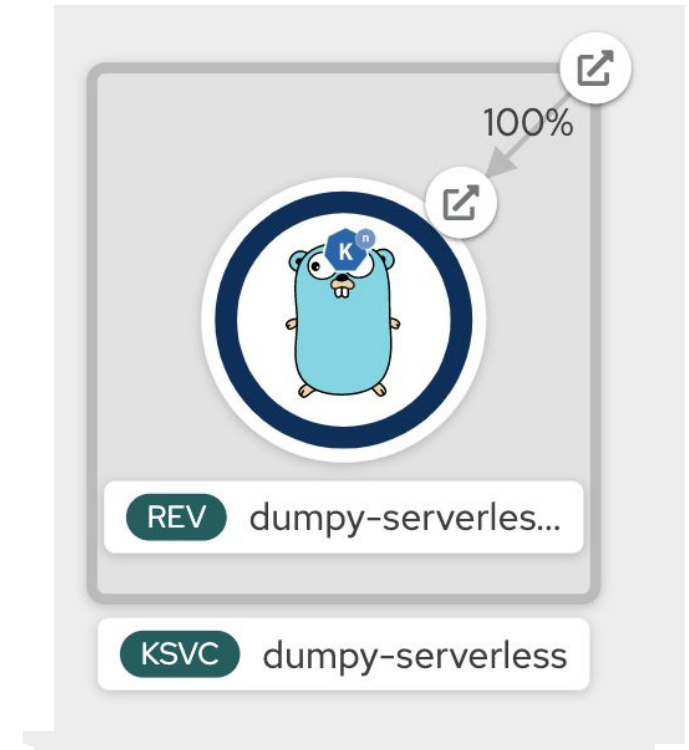
apps.openshift.io/DeploymentConfig

A Deployment Config defines the template for a pod and manages deploying new images or configuration changes

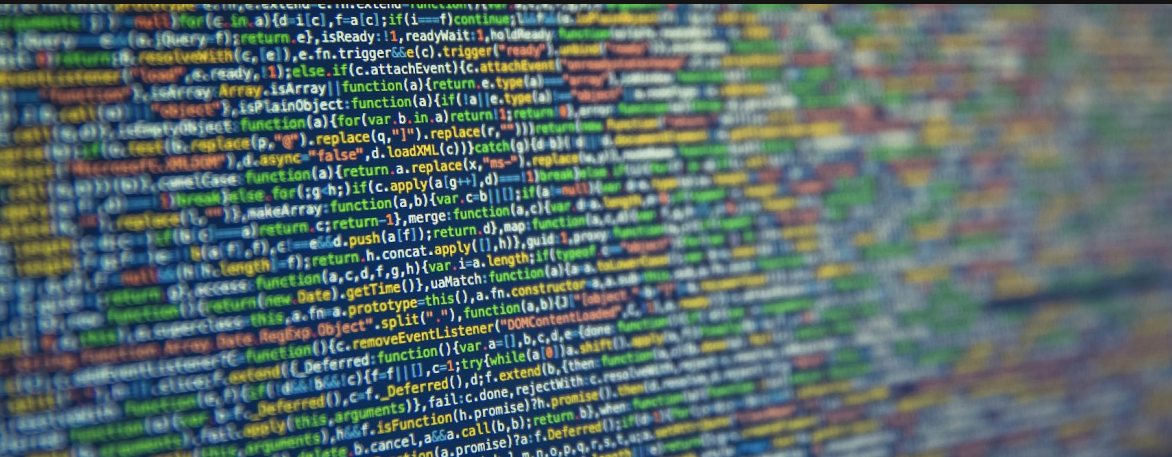
- Knative Service Tech Preview

serving.knative.dev/Service

A Knative Service enables scaling to zero when idle



# OpenShift Pipelines



OpenShift Pipelines provides a Cloud-Native CI/CD experience based on Tekton

- Container based
- Serverless
- Designed for DevOps

Tech Preview on OCP 4.2





An open-source project for providing a set of shared and standard components for building Kubernetes-style CI/CD systems

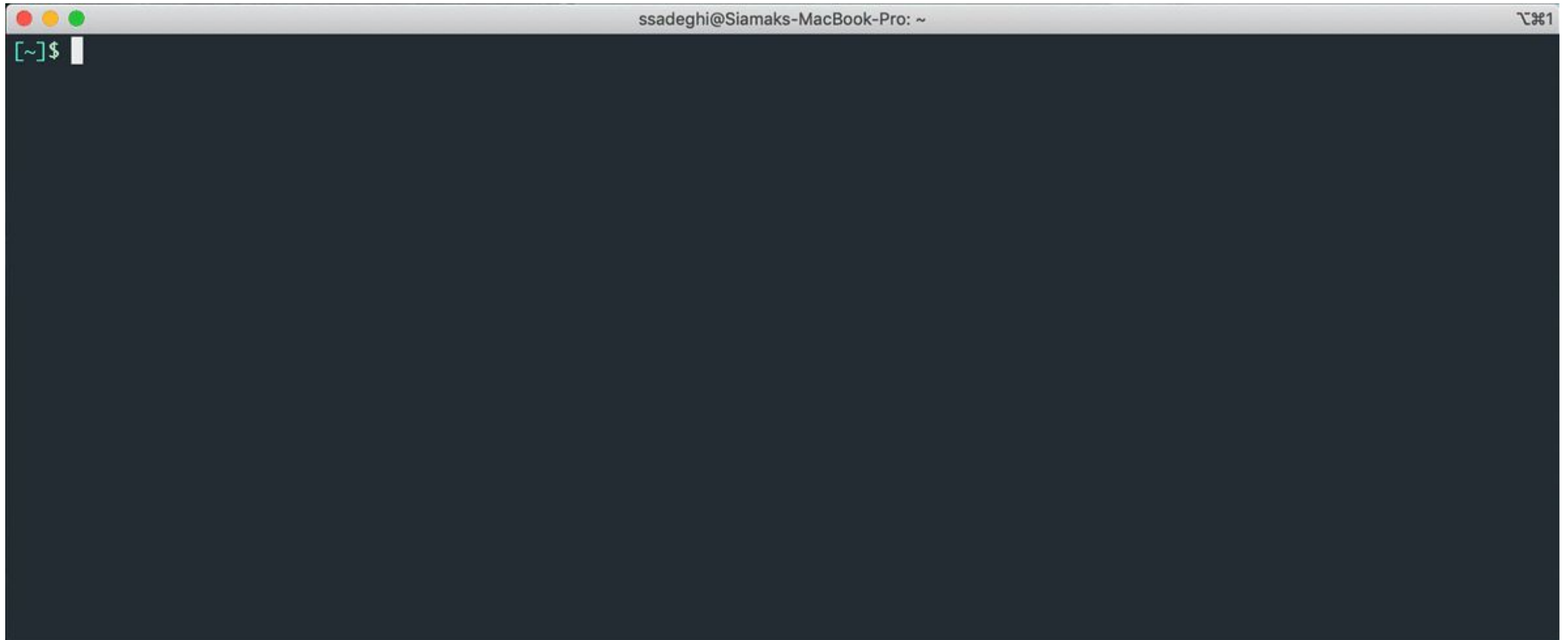


CD.FOUNDATION

Governed by the Continuous Delivery Foundation

Contributions from Google, Red Hat, Cloudbees, IBM, Pivotal and many more

# Manage Pipelines with Tekton CLI



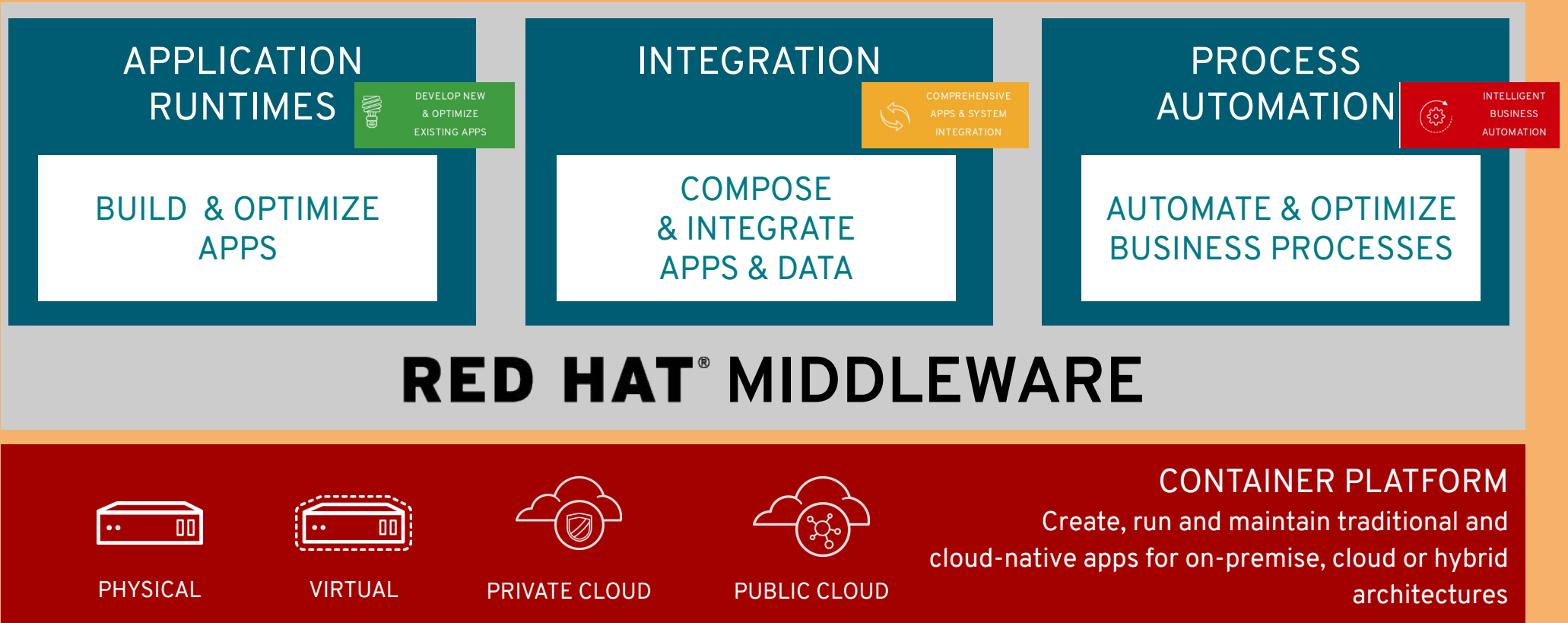
# Pipeline Visualisation in **Developer** perspective

The screenshot displays the Red Hat OpenShift Container Platform interface in the Developer perspective. The top navigation bar includes the Red Hat logo and the text 'OpenShift Container Platform'. On the right, there are icons for a grid, a help question mark, and a user profile 'siamak'. The left sidebar contains navigation options: 'Developer', '+ Add', 'Topology', 'Builds', 'Pipelines' (highlighted), and 'Advanced'. The main content area shows 'Project: Project01' and 'Pipeline Run Details' for 'pipelinerun01a' in a 'Running' state. A 'Tech Preview' badge and an 'Actions' dropdown are visible. Below this, there are tabs for 'Overview', 'YAML', and 'Logs'. The 'Overview' tab shows a 'Pipeline Run Overview' diagram with five steps: 'code compile', 'compile & test', 'unit test', 'security check', and 'image build'. The first four steps are completed with green checkmarks, while 'image build' is in progress with a refresh icon. Below the diagram, metadata is listed: Name (pipelinerun01a), Namespace (NS project01), Labels (app=dummy-mongo-pod-test, bap.me/environment=dev, bap.me/track=experimental, bap.me/tier=backend), Annotations (0 Annotations), and Created At (Aug 8, 4:00 pm).

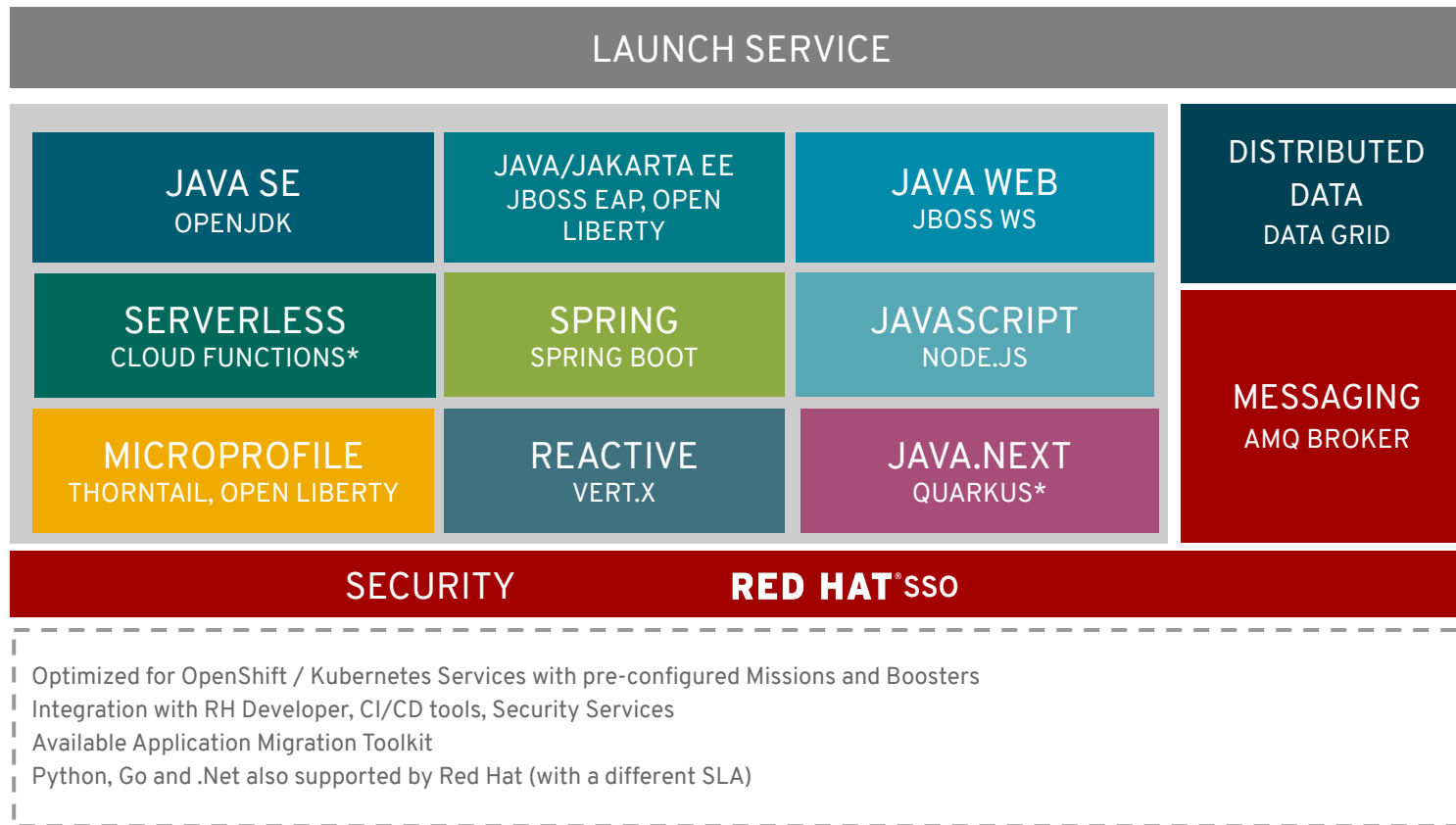
# Red Hat AppDev Solutions

CREATE THE APPLICATION LANDSCAPE CUSTOMERS NEED

Application Modernization  
and Migration



# Red Hat Runtimes

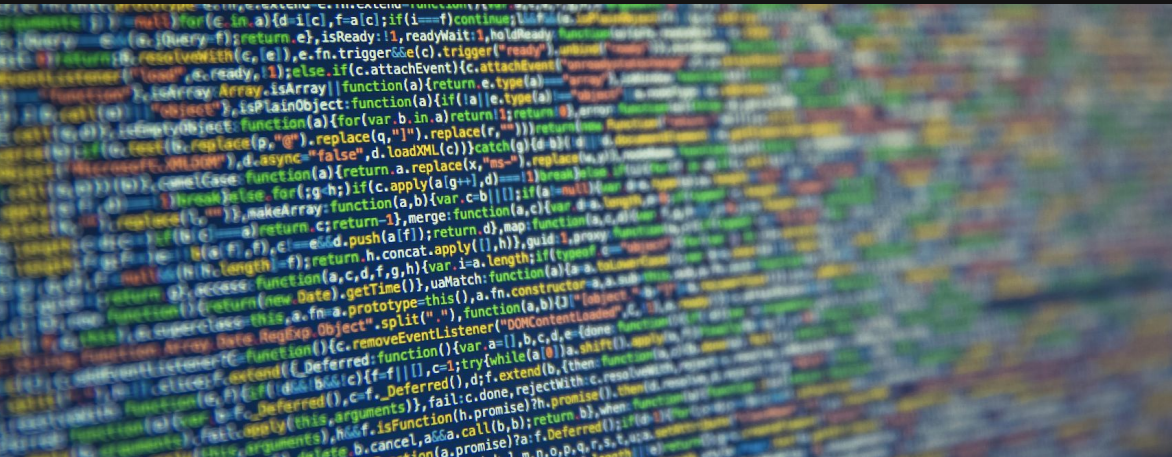


Facilitate cloud native app development  
ON THE HYBRID CLOUD:

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

# Quarkus

## Supersonic Subatomic Java



A Kubernetes Native Java stack tailored for GraalVM & OpenJDK HotSpot, crafted from the best of breed Java libraries and standards



**INSTANA**

1st Operator  
written in Quarkus

 **GitHub**

★ Star

2,297

🍴 Fork

348

# The community is growing



Quarkus in Worldwide  
11 results



**4,617** Followers

**VERT.X**

Eclipse Vert.x



Hibernate



RESTEasy



Apache Camel



Eclipse MicroProfile



Netty



DynamoDB



My Faces



Kubernetes



OpenShift



Jaeger



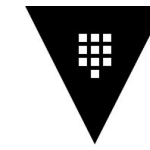
Prometheus



Apache Kafka



Infinispan

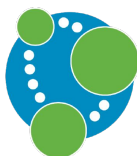


HashiCorp  
**Vault**

 **OmniFaces**



Flyway



Neo4j



mongoDB.  
MongoDB

**MQTT**.ORG

MQTT



KeyCloak



Apache Tika



# DEMO

- Edit and Debug using CRW
- Push changes to Git Repo
- Build and Deploy using OpenShift Pipelines/Tekton
- Deploy as Serverless Service
- Local App Dev using CRC
- Using OpenShift Plugin with Local IDE
  - Using odo



# Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.



[linkedin.com/company/red-hat](https://www.linkedin.com/company/red-hat)



[youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)



[facebook.com/redhatinc](https://www.facebook.com/redhatinc)



[twitter.com/RedHat](https://twitter.com/RedHat)