



**Red Hat**  
Developer

---



**Red Hat**

## **Developer Webinar Series 2020**

[developers.redhat.com/webinars/](https://developers.redhat.com/webinars/)

# Developer Productivity on Kubernetes with OpenShift



**Ryan Jarvinen**

Developer Advocate Red Hat  
@RyanJ



**Markus Eisele**

Developer Adoption Lead Red Hat  
@myfear

**Q: What is OpenShift?**



**DISTRIBUTIONS**

# Distributions by Red Hat

- RHEL 8 for Enterprise Productivity
- RHEL CoreOS for Container Hosts
- Universal Base Images for Containerized Workloads
- Fedora IoT for Edge Enablement
- Fedora for reliable access to upstream bits  
(also my favorite desktop OS)

**OpenShift is a complete Cloud Platform  
built around **Kubernetes****



# OpenShift: Kubernetes as a distributable platform for cloud productivity

## Q: What is the value to **Developers**?

1. What is Kubernetes?
2. What Kubernetes is not?
3. *What does success with Kubernetes typically look like?*

# 1. What is **Kubernetes**?



# Hard Multi-Tenancy in Kubernetes

Friday, May 18, 2018

Kubernetes is the new kernel. We can refer to it as a “cluster kernel” versus the typical operating system kernel. This means a lot of great things for users and a lot of the same challenges we have already faced with operating system kernels. One of which being privilege isolation. In Kubernetes, we refer to this as multi-tenancy, or the dream of being able to isolate tenants of a cluster.

**Kubernetes is the new kernel**

<https://blog.jessfraz.com/post/hard-multi-tenancy-in-kubernetes/>



Kelsey Hightower ✓  
@kelseyhightower

**Kubernetes is the new systemd**

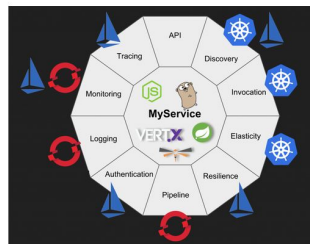
Kubernetes is not the kernel; it's systemd.

16:57 - 25 ene. 2019 <https://twitter.com/kelseyhightower/status/1088828102480781313>

## Why Kubernetes is The New Application Server

By Rafael Benevides June 28, 2018

+10 rating, 16 votes



**Kubernetes is the new Application Server**

this article, I'm going to ask some provocative questions to make my case for *Why Kubernetes is the new application server.*

<https://developers.redhat.com/blog/2018/06/28/why-kubernetes-is-the-new-application-server/>

Home > Containers > Kubernetes

**Kubernetes is the new OS**

INSIDER

## Sorry, Linux. Kubernetes is now the OS that matters

Linux is just plumbing. The real OS—the real value—is with Kubernetes



By Matt Asay  
InfoWorld | NOV 20, 2018

<https://www.infoworld.com/article/3322120/kubernetes/sorry-linux-kubernetes-is-now-the-os-that-matters.html>



# Concepts

## Concepts

▼ Overview

[What is Kubernetes](#)

Kubernetes Components

The Kubernetes API

▶ Working with Kubernetes Objects

▶ Cluster Architecture

▶ Containers

▶ Workloads

▶ Services, Load Balancing, and Networking

▶ Storage

▶ Configuration

▶ Security

▶ Policies

## What is Kubernetes



This page is an overview of Kubernetes.

- [Going back in time](#)
- [Why you need Kubernetes and what can it do](#)
- [What Kubernetes is not](#)
- [What's next](#)

Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation. It has a large, rapidly growing ecosystem. Kubernetes services, support, and tools are widely available.

The name Kubernetes originates from Greek, meaning helmsman or pilot. Google open-sourced the Kubernetes project in 2014. Kubernetes builds upon a decade and a half of experience that Google has with running production workloads at scale, combined with best-of-breed ideas and practices from the community.

## What Kubernetes is not

Kubernetes is not a traditional, all-inclusive PaaS (Platform as a Service) system. Since Kubernetes operates at the container level rather than at the hardware level, it provides some generally applicable features common to PaaS offerings, such as deployment, scaling, load balancing, logging, and monitoring. However, Kubernetes is not monolithic, and these default solutions are optional and pluggable. Kubernetes provides the building blocks for building developer platforms but preserves user choice and flexibility where it is important.

Kubernetes:

- Does not limit the types of applications supported. Kubernetes aims to support an extremely diverse variety of workloads, including stateless, stateful, and data-processing workloads. If an application can run in a container, it should run great on Kubernetes.
- Does not deploy source code and does not build your application. Continuous Integration, Delivery, and Deployment (CI/CD) workflows are determined by organization cultures and preferences as well as technical requirements.
- Does not provide application-level services, such as middleware (for example, message buses), data-processing frameworks (for example, Spark), databases (for example, MySQL), caches, nor cluster storage systems (for example, Ceph) as built-in services. Such components can run on Kubernetes, and/or can be accessed by applications running on Kubernetes through portable mechanisms, such as the [Open Service Broker](#).
- Does not dictate logging, monitoring, or alerting solutions. It provides some integrations as proof of concept, and mechanisms to collect and export metrics.
- Does not provide nor mandate a configuration language/system (for example, Jsonnet). It provides a declarative API that may be targeted by arbitrary forms of declarative specifications.
- Does not provide nor adopt any comprehensive machine configuration, maintenance, management, or self-healing systems.
- Additionally, Kubernetes is not a mere orchestration system. In fact, it eliminates the need for orchestration. The technical

*“Kubernetes provides the building blocks for building developer platforms”*

*“Kubernetes is not a traditional all-inclusive PaaS (Platform-as-a-Service) system”*

**∴ Kubernetes is not** a developer platform



What does **success**  
typically look like?



# Developers want tools that help maximize **productivity**

- Performant
- Secure
- Consistently available
- Stable and compatible with related tools
- Faster than ever before

**Developers will refuse to adopt tooling that gets in their way**



## Developers demand **freedom** to innovate

- Choice of architectures
- Choice of programming languages
- Choice of databases
- Choice of application services
- Choice of development tooling
- Choice of build and deployment automation

**Developers desire the freedom to apply tools to problems in any locality**

# Kubernetes as a Distribution:

## An Integrated Platform Experience focused on Cloud Productivity

	Kubernetes	OpenShift
<b>Developer Productivity</b>	<i>Out of scope</i>	<a href="#">Developer Perspective</a> (app-centric topology view, live terminals, logs, and stats), Routes (easy hostname records), Developer Catalog (Operator-backed, Admin configurable)
<b>Inner Loop</b>	<i>Out of scope</i>	Odo ("cf push" style code promotion between commits), CodeReady Workspaces (hosted IDE w/ VScode plugin support)
<b>Builds, Pipelines</b>	<i>Out of scope</i>	Tekton pipelines, Buildah (runs builds on cluster, producing container images that <b>do not</b> require root privs)
<b>Container Registry</b>	<i>Out of scope</i>	Quay w/ Clair for static image analysis (on-cluster)
<b>Logging</b>	<i>Out of scope</i>	Prometheus (on-cluster, Developer Perspective dashboard includes PromQL and live data)
<b>Service Mesh</b>	<i>Out of scope</i>	Istio (on-cluster)
<b>Serverless</b>	<i>Out of scope</i>	Knative (on-cluster, Developer Perspective dashboard integration)
<b>Advanced Workloads &amp; Marketplace</b>	CRDs (v1.12+ recommended)	Dashboard support for CRDs, Disk storage (volumes and claims), Operator Marketplace (on-cluster), Helm3, KubeVirt (VMs), GPUs



# Welcome to OperatorHub.io

OperatorHub.io is a new home for the Kubernetes community to share Operators. Find an existing Operator or list your own today.

## CATEGORIES

109 ITEMS

VIEW  ▾

SORT A-Z ▾

- AI/Machine Learning
- Application Runtime
- Big Data
- Cloud Provider
- Database
- Developer Tools
- Integration & Delivery
- Logging & Tracing
- Monitoring
- Networking
- OpenShift Optional
- Security
- Storage
- Streaming & Messaging

## PROVIDER

- Altinity (1)



**Akka Cluster Operator**  
provided by Lightbend, Inc.

Run Akka Cluster applications on Kubernetes.



**Altinity ClickHouse Operator**  
provided by Altinity

ClickHouse Operator manages full lifecycle of ClickHouse



**Anchore Engine Operator**  
provided by Anchore Inc.

Anchore Engine - container image scanning service for policy-based security, best



**Apache CouchDB**  
provided by IBM

Apache CouchDB is a highly available NOSQL database for web and mobi



**Apache Spark Operator**  
provided by radanalytics.io

An operator for managing the Apache Spark clusters and



**API Operator for Kubernetes**  
provided by WSO2

API Operator provides full



**APIcast**  
provided by Red Hat

APIcast is an API gateway built on top of NGINX. It is part of



**Appranix CPS Operator**  
provided by Appranix, Inc

The Appranix CPS operator enables you to back up and

```
if (top !== self) {
function calcWidth() {
  var wW = 0;
  if (typeof window.innerWidth == 'number') {
    wW = window.innerWidth;
  } else if (document.documentElement && document
  } else if (document.body && document.body.c
  wW = document.documentElement.scrollH
  if (sH = document.documentElement.scrollHeight
  var wH = window.innerHeight || document
  sW = !document.all && (sH > wH)
  'menu', 'wid
```

**Ability to focus on `{$dayjob}`**

```
if (top !== self) {
function calcWidth() {
var wW = 0;
if (typeof window.innerWidth == 'number') {
wW = window.innerWidth;
} else if (document.documentElement && document
body && document.clientWidth) {
wW = document.documentElement.clientWidth;
} else if (document.body && document.body.c
lientWidth) {
wW = document.body.clientWidth;
}
if (sH = document.documentElement.scrollHeight
|| document
innerHeight) {
var wH = window.innerHeight || document
innerHeight;
var sW = !document.all && (sH > wH)
? 'menu', 'wid
```

**“Kubernetes **does not**  
deploy source code or  
build your application”**



VMs? Containers?  
All I want to do is program!  
Jeesh.



```
if (top != self)
function calcWidth() {
  var wW = 0;
  if (typeof window.innerWidth == 'number') {
    wW = window.innerWidth;
  } else if (document.documentElement && document
    wW = document.documentElement.clientWidth;
  } else if (document.body && document.body.c
    wW = document.body.clientWidth;
  }
  if (sH = document.documentElement.scrollHeight
  var wH = window.innerHeight || document
  var sW = !document.all && (sH > wH)
    'menu', 'wid
```

**Q: What does **iterative development** look like on OpenShift?**

*INTRODUCING:*

# OpenShift-Do (aka. odo)

a developer-focused command line interface



**odo** source code and releases:

```
https://github.com/redhat-developer/odo
```



## Command Line Options

	kubectl	oc	odo
The community standard for interacting with Kubernetes API resources	X		
Support for CustomResourceDefinitions and kubectl plugins for extended functionality	X	X	
A drop-in replacement for kubectl that adds support for OpenShift-specific resources types		X	
Credentials management and cluster login support: <code>oc login</code>		X	X
Enhanced support for Kubernetes namespaces		X	X
Automatically sync code changes into a cluster: <code>oc rsync</code> , <code>odo watch</code>		X	X
Ability to preview code changes on the cluster: <code>odo push</code> (comparable to <code>cf push</code> )			X
Enhanced support for application components and label management. Quickly adopt best-practices as you work			X



**odo** examples

# Iterative Development on the Command Line

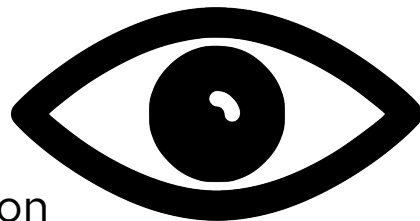
## **odo push**

- Push local sources to initiate hosted builds and deployments
- Push local binary build artifacts into remote containers
- Decouple pushes from commits



## **odo watch**

- Watch local sources and/or binary artifacts for updates, then replicate changes into a hosted container
- Synchronize HTML, CSS, and interpreted language sources to achieve tighter loops with production-grade instrumentation



# Application Components

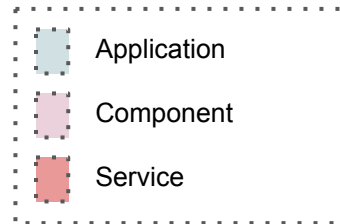
Use local files and artifacts to preview changes while `odo` provides enhanced support for application-component labels and context management

```
$ odo create <component_type> [component_name] [--local] [local_path]
odo create nodejs
odo create nodejs frontend
odo create nodejs frontend --local ./frontend

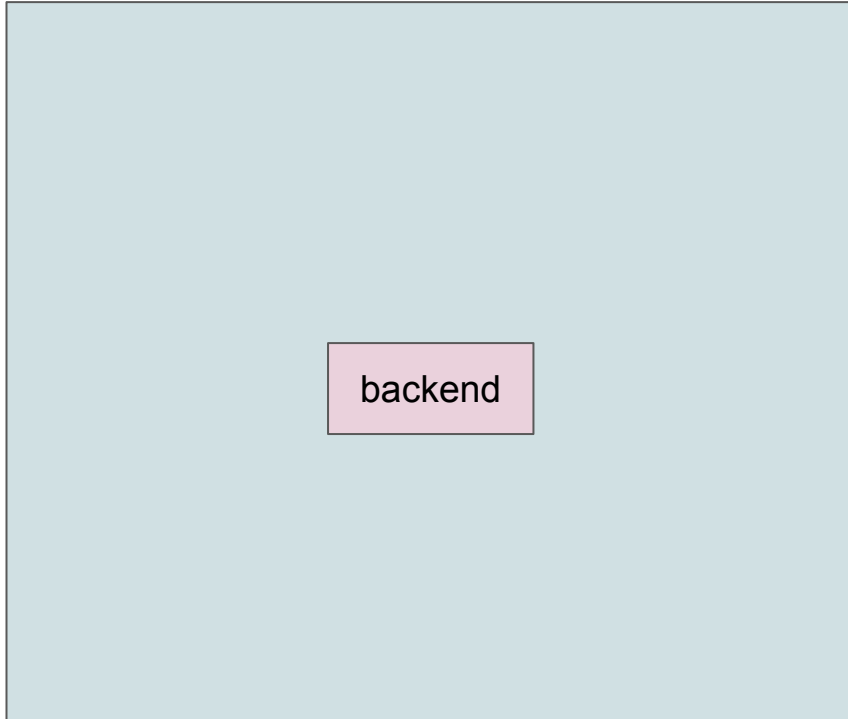
$ odo create <component_type> [component_name] [--binary binary_path]
odo create wildfly backend --binary ./target/myapp.war
```

# Create an Application context

```
$ odo application create odo-demo
```

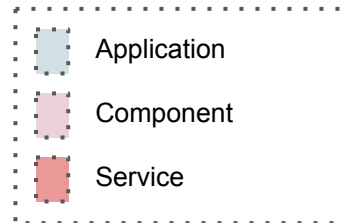


# Add a Component to an Application context

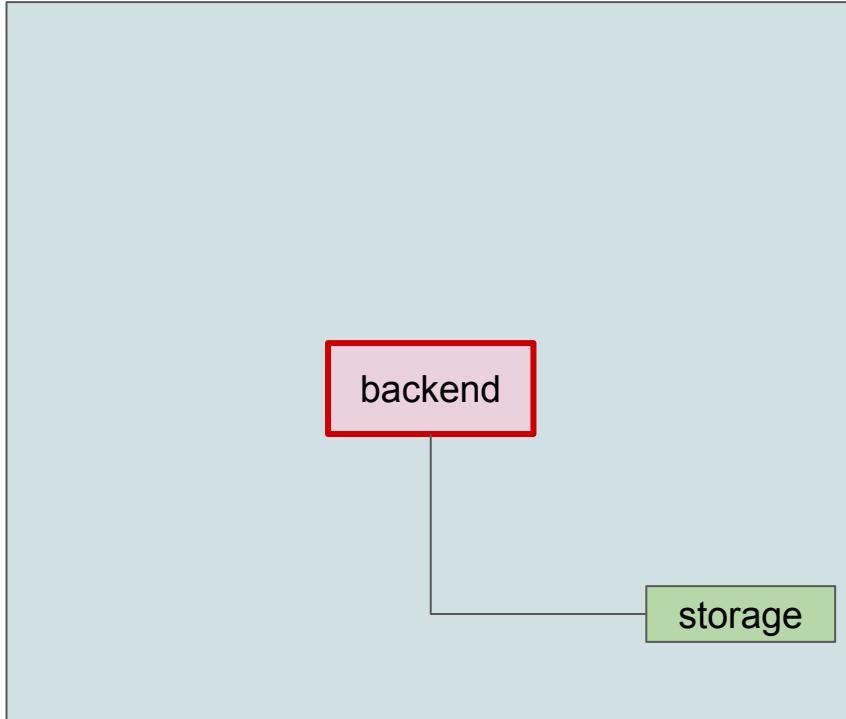


```
$ odo create wildfly
```

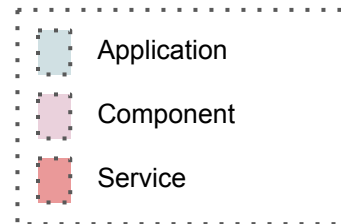
- From local (Iterative development)
- From binary (Iterative development)
- From git (cloud based development)



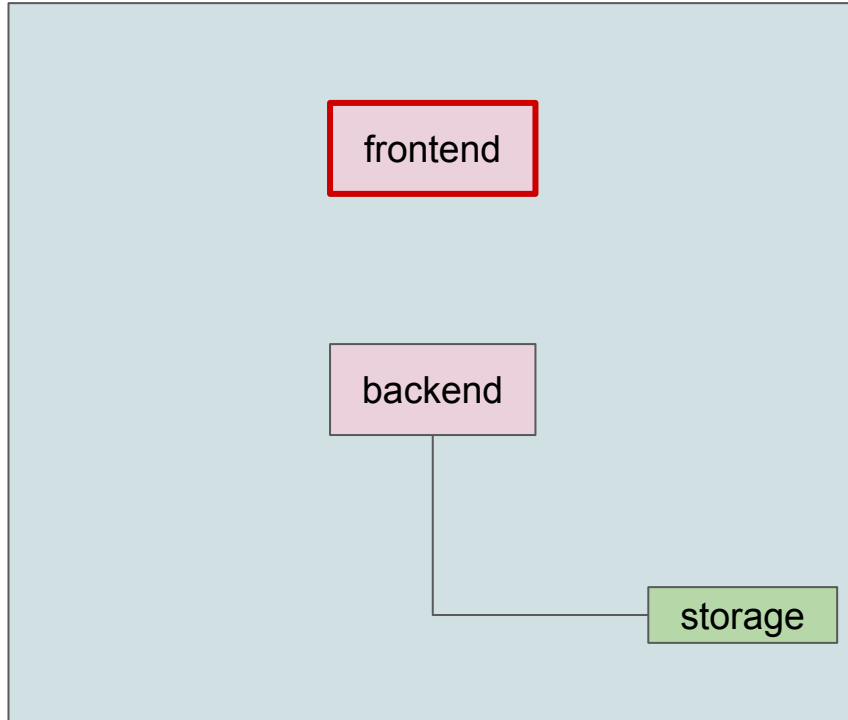
# Add Storage to the current Component



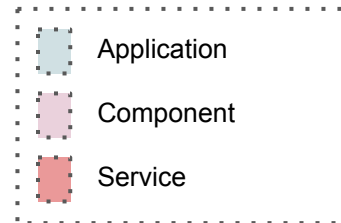
```
$ odo storage create  
  --path=/opt/my-app/data  
  --size=1Gi
```



# Create additional Application Components, swap context

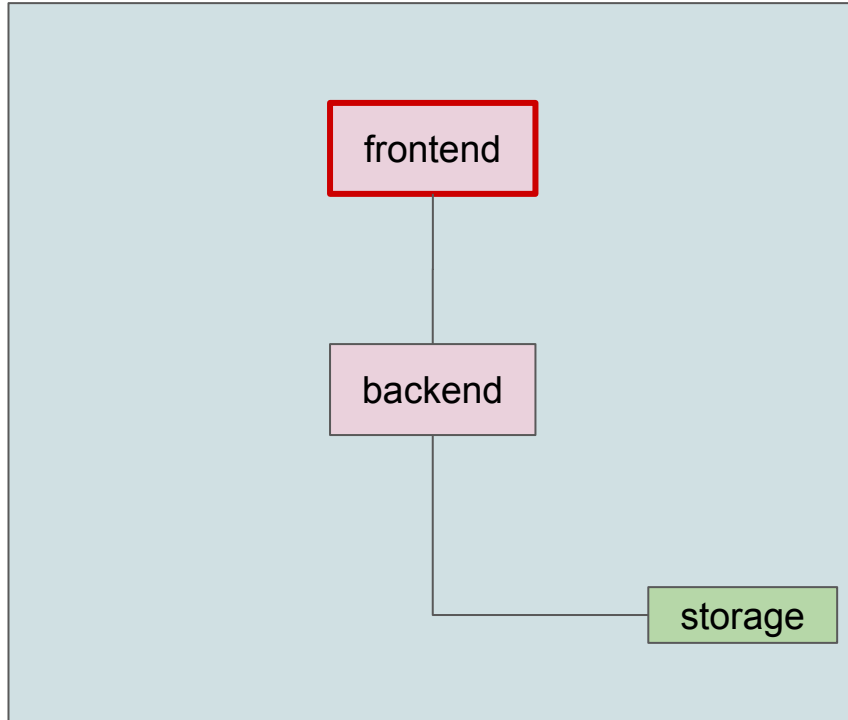


```
$ odo create httpd frontend
```

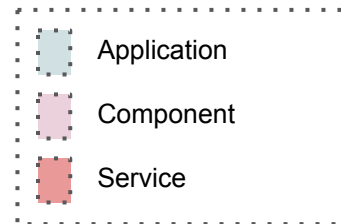




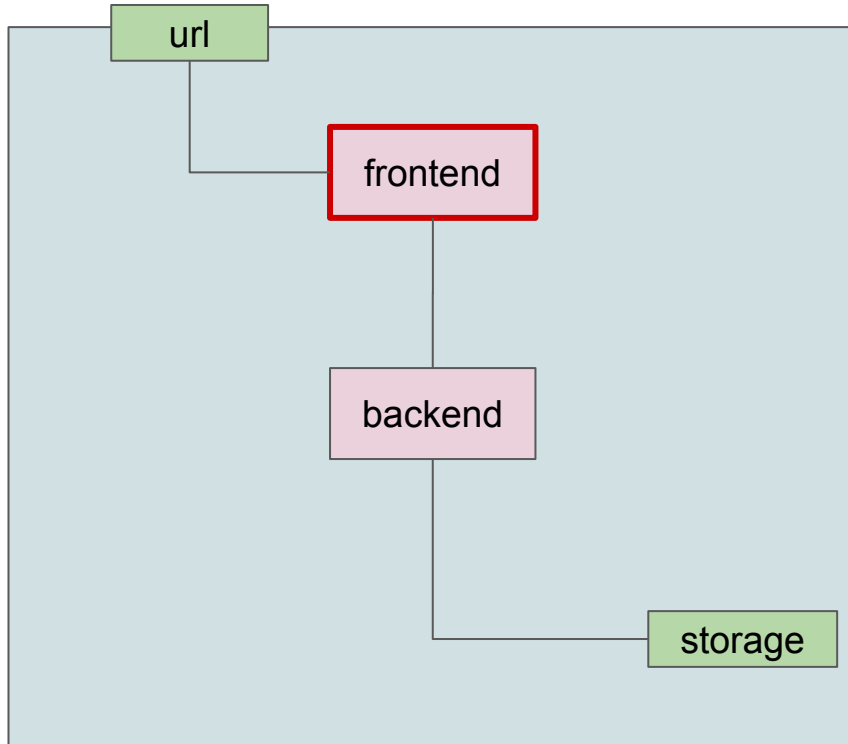
# Define relationships between Application Components



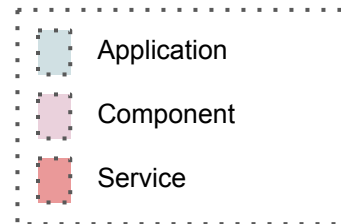
```
$ odo link backend
```



# Setup DNS and establish external access to a Service



```
$ odo url create
```





## Interactive Learning Portal

Our Interactive Learning Scenarios provide you with a pre-configured OpenShift® instance, accessible from your browser without any downloads or configuration. Use it to experiment, learn OpenShift and see how we can help solve real-world problems.

Foundations of  
OpenShift

START COURSE

Building Applications On  
OpenShift

START COURSE

Subsystems,  
Components, and  
Internals

START COURSE

OpenShift Playgrounds

START COURSE

Service Mesh Workshop  
with Istio

START COURSE

Building Operators on  
OpenShift

START COURSE

AI and Machine Learning  
on OpenShift

Other ways to interact with the system:

Link to

Odo push

Odo watch

Oc watch

Git commit && git push ->

    GitHub -> webhook -> buildConfig -> deployment

IDE support

Che w/ VSCode plugins

Docker build && docker tag && docker push -> deploymentConfig

Oc exec (attach to terminals)

All available on [learn.openshift.com](https://learn.openshift.com):

- Operator SDK with Helm



**Looking for  
Continuous  
Productivity?**



Looking for a  
**Developer  
Platform?**



# OpenShift 4

The simplicity and automation of the public cloud, delivered as an enterprise hybrid cloud platform.

[GET STARTED](#)

## A smarter Kubernetes platform

OpenShift 4 introduces automated installation, patching, and upgrades for every layer of the container stack from the operating system through application services. Try OpenShift 4 today and experience the ease of push-button cluster and service management.



# Install OpenShift Container Platform 4

- Clusters
- Subscriptions
- Documentation
- Support Cases
- Cluster Manager Feedback


## Select an infrastructure provider



Run on Amazon Web Services



Run on Microsoft Azure



Run on Google Cloud Platform



Run on VMware vSphere




Run on Red Hat OpenStack

Developer Preview



Run on Red Hat Virtualization



Run on Bare Metal

IBM Z  
IBM LinuxONE™

Run on IBM Z



Run on Laptop

Powered by Red Hat CodeReady Containers

# Questions?

*We want your feedback!*

<https://groups.google.com/forum/#!forum/openshift-dev-users>

# Thank You!



**Red Hat**  
Developer



**Red Hat**

**April 2 16:00 CET**

**Monolith to Microservices:  
The journey to the new normal**

[developers.redhat.com/webinars/](https://developers.redhat.com/webinars/)