# Increase developer productivity on Kubernetes with OpenShift

Martin Östmark

Chief Architect AppDev, Nordics
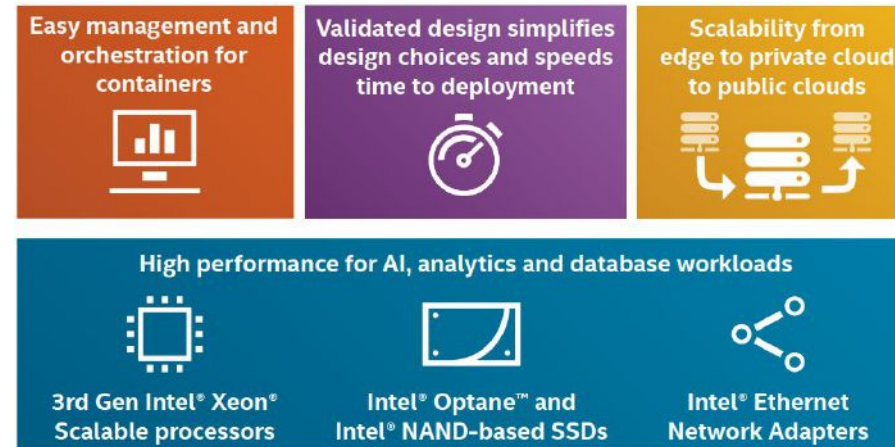
**Red Hat**

# Red Hat OpenShift Reference Architecture

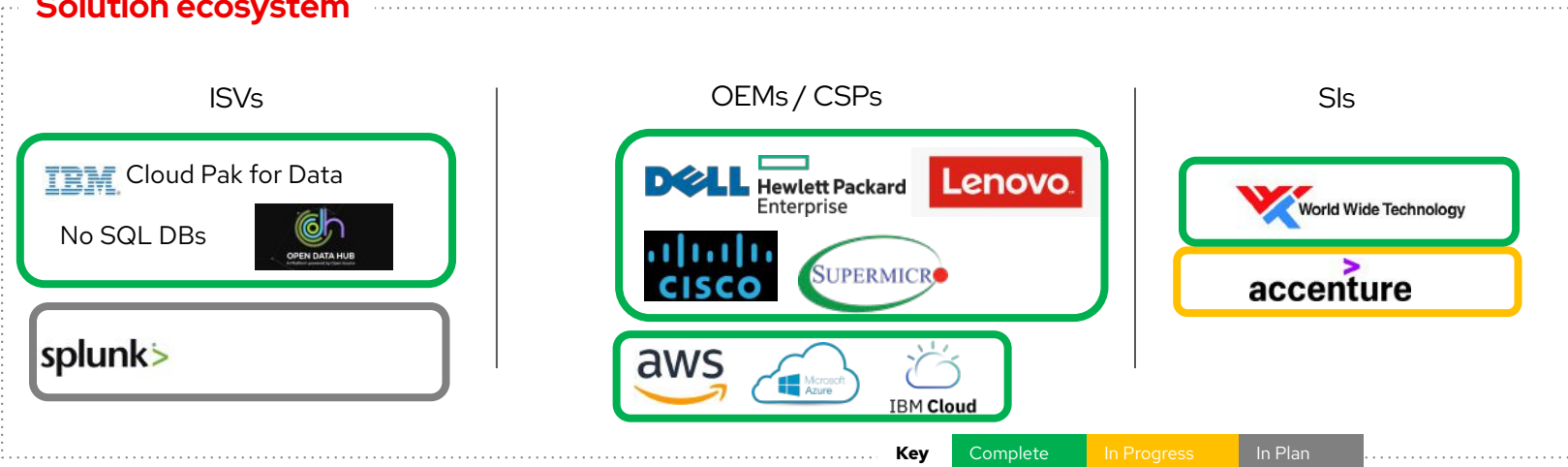## Joint Red Hat and Intel OpenShift Reference Architecture

## Solution overview

**Summary:** The RA enables deployment of performant and low-latency container-based workloads onto different footprints, such as bare metal, virtual, private cloud, public cloud, or a combination of these, in either a centralized data center or at the edge

**Purpose:** A general purpose OpenShift reference architecture to showcase the best of Intel and Red Hat products with key workloads



Easy management and orchestration for containers

Validated design simplifies design choices and speeds time to deployment

Scalability from edge to private cloud to public clouds

High performance for AI, analytics and database workloads

3rd Gen Intel® Xeon® Scalable processors

Intel® Optane™ and Intel® NAND-based SSDs

Intel® Ethernet Network Adapters

## Intel enabling status

- Intel® Xeon (2nd Gen – Cascade Lake, 3rd Gen – Ice Lake)
- Intel Optane (PMEM, SSD); Columbiaville

## Collateral

- Intel OpenShift RA for 4.6
- Intel OpenShift Solution Brief for 4.6
- Red Hat: OpenShift Ref Arch – Multiple OEMs
- Dell: OpenShift Offering
- HPE: OpenShift Offering
- Cisco: OpenShift Offering
- Lenovo: OpenShift Offering
- Supermicro: OpenShift Offering
- Penguin Computing: OpenShift Offering

## Solution ecosystem

ISVs

OEMs / CSPs

SIs

IBM Cloud Pak for Data

No SQL DBs

OPEN DATA HUB

splunk>

DELL  Hewlett Packard Enterprise  Lenovo

CISCO  SUPERMICRO

aws  Microsoft Azure  IBM Cloud

World Wide Technology

accenture
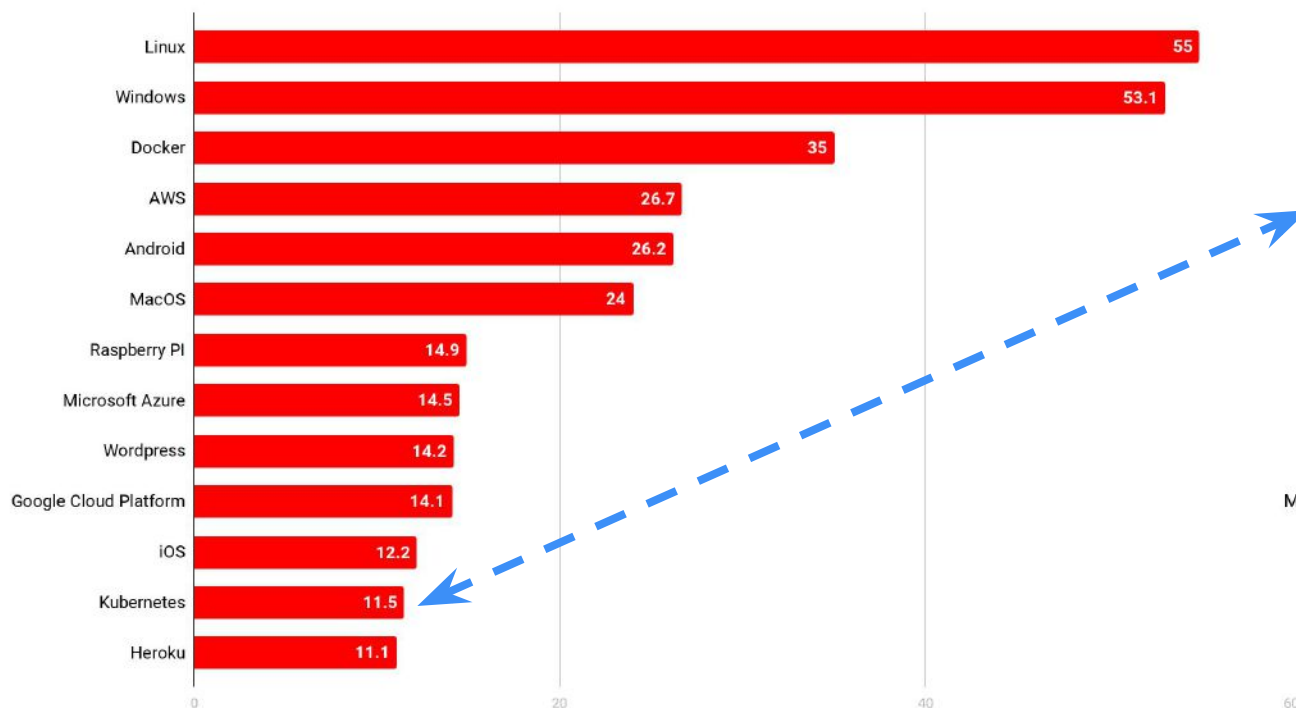
Key    Complete    In Progress    In Plan

# Agenda

- Kubernetes solves all your problems!?
- Spend more time developing
- Easily navigate the many components of your applications.
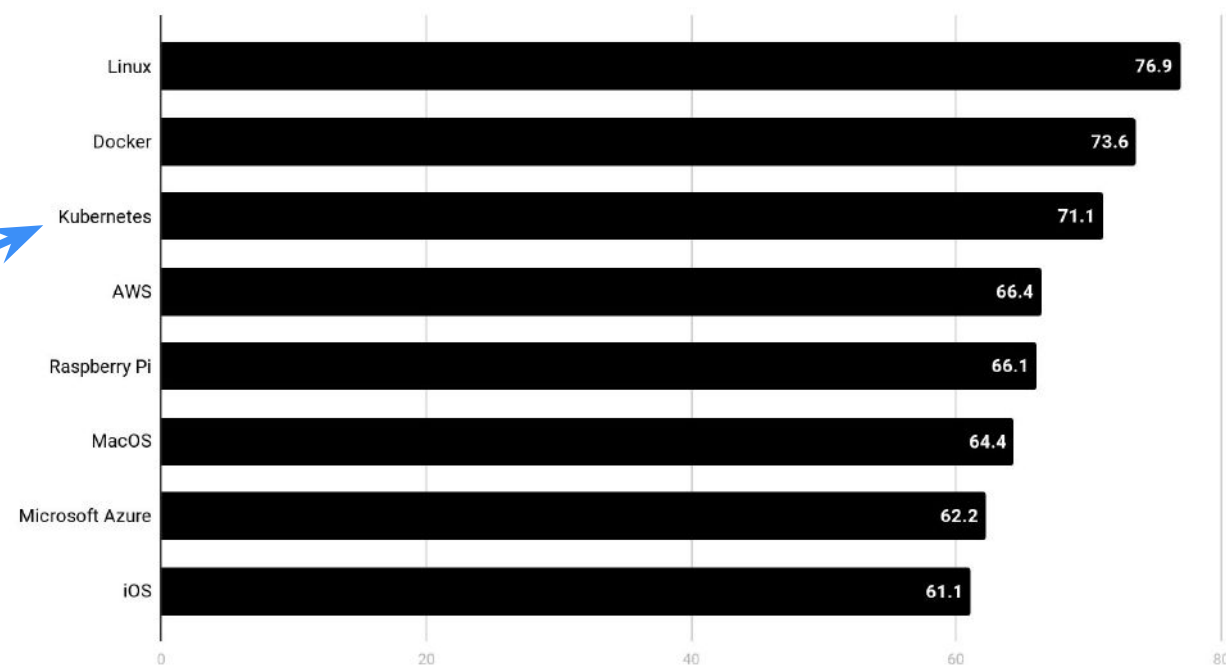- Get advanced services, without the hassle.
- Recap

# Kubernetes solves all your problems!?

Red Hat

# Kubernetes and Developers

**Platform Adoption****

| Platform | Value |
|----------|-------|
| Linux | 55 |
| Windows | 53.1 |
| Docker | 35 |
| AWS | 26.7 |
| Android | 26.2 |
| MacOS | 24 |
| Raspberry PI | 14.9 |
| Microsoft Azure | 14.5 |
| Wordpress | 14.2 |
| Google Cloud Platform | 14.1 |
| iOS | 12.2 |
| Kubernetes | 11.5 |
| Heroku | 11.1 |

**Most Loved Platforms****

| Platform | Value |
|----------|-------|
| Linux | 76.9 |
| Docker | 73.6 |
| Kubernetes | 71.1 |
| AWS | 66.4 |
| Raspberry Pi | 66.1 |
| MacOS | 64.4 |
| Microsoft Azure | 62.2 |
| iOS | 61.1 |

## Kubernetes tried by most businesses

Red Hat

# Complexities preventing adoption...

**Ask a Developer**

"What do you think about when you hear the word **Kubernetes** ?

- Containers
- YAML
- Virtualization
- Infrastructure
- Abstraction
- Complexity

**DIY K8s strategy – maintenance problems**

kubernetes

Documentation   Blog   Training   Partners   Community   Case Studies   English ⌄   v1.18 ⌄

# Concepts

HOME   GETTING STARTED   **CONCEPTS**   TASKS   TUTORIALS   REFERENCE   CONTRIBUTE

Search 🔍

**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
▶ Scheduling
▶ Cluster Administration

# What is Kubernetes?

This page is an overview of Kubernetes.

- **Going back in time**
- **Why you need Kubernetes and what it can 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 combines over 15 years of Google's experience running production workloads at scale with best-of-breed ideas and practices from the community.

## Going back in time

# 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.

# Complexities preventing adoption...

**Ask a Developer**

"What do you think about when you hear the word **Kubernetes** ?

- Containers
- YAML
- Virtualization
- Infrastructure
- Abstraction
- Complexity

What do you really want from **Kubernetes** ?

- Infrastructure as Code
- Innovation & Cloud Native Tools
- Flexibility
- Extensibility
- Portability
- Community

**Not much focus and support yet**

# Developer interest

**"I JUST WANT TO CODE"**                    **"I WANT TO BE A KUBE EXPERT"**

Simple / Opinionated                           Powerful / Flexible

...but all developers want:

➔ Choice of tools, not "corporate dictated" tools

➔ Ability to consume cloud services and deploy anywhere

➔ Kubernetes...but easier

➔ To be able to switch contexts instantly

# How to easily deploy things on OpenShift

# Deploy an application on Kubernetes

## Common process

Build the application binary → Build the container image → Push to a container image repository → Deploy the container image

Create access to the service → Configure ingress

Install and configure a load balancer
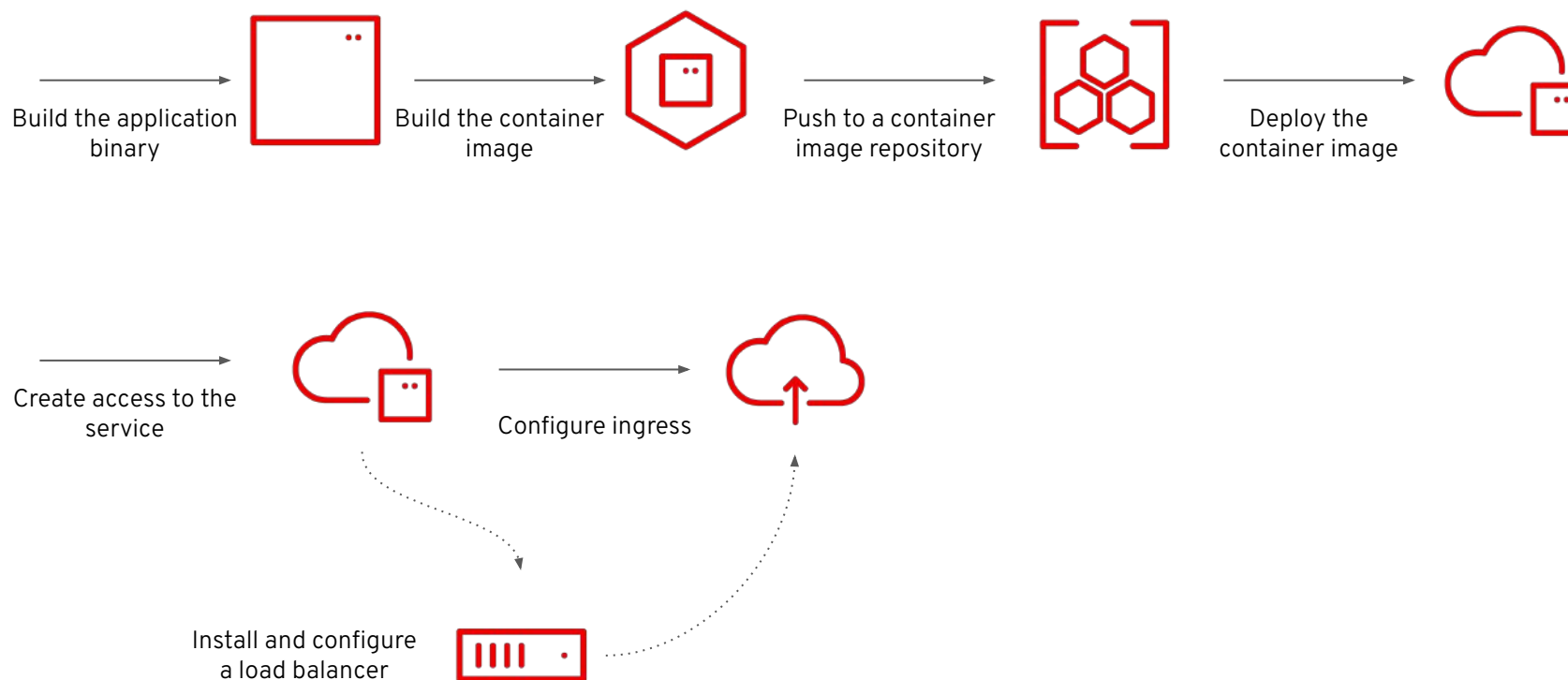
# What happened? Source-to-Image on OpenShift explained

- Created a build pod for building the app

- Created an OpenShift Build config

- Pulled the builder image into OpenShift's internal container registry.

- Cloned the application repo locally

- Compiled the source code and created the artifact(s) to run the application

- Created a new container image with the compiled application and pushed this container image into the internal container registry

- Created a Kubernetes Deployment with specifications around pods, service spec etc.

- Kicked off a deploy of the container image.

- Removed the build pod.

- Exposed the service in the route (ingress)

# Recap -
# how to easily deploy things on OpenShift
# *So that you can spend more time developing*

Red Hat

# How to use the Developer Perspective

**Topology view**

# Recap -
# How to use the Developer Perspective
*So you can easily navigate the many components of your applications*

**Red Hat**

# Easily deploy developer services on OpenShift

Red Hat

Recap -
Easily deploy developer services on OpenShift

*So you can use modern day features, without the hassle*

Red Hat

We showed you how to easily deploy applications on OpenShift
*So you can spend more time developing*

We showed you how to use the developer perspective
*So you can easily overview and manage the many components of your applications.*

We  showed you how to easily deploy developer services on OpenShift

*So you can use modern day features, without the hassle*

Red Hat

# Developer Productivity in Scope

| | Kubernetes | OpenShift |
|---|---|---|
| **Developer Productivity** | Out of scope | ✔ [Developer Perspective](#) (app-centric topology view, live terminals, logs, and stats), Routes (easy hostname records), Developer Catalog (Operator-backed, Admin configurable) |
| **Inner Loop** | Out of scope | ✔ CodeReady Workspaces (hosted IDE w/ VScode plugin support), **odo** ("cf push" style code promotion between commits) |
| **Builds, Pipelines** | Out of scope | ✔ Tekton pipelines, Source-to-Image S2I, Buildah (runs builds on cluster, producing container images that **do not require root privs**) |
| **Container Registry** | Out of scope | ✔ Quay w/ Clair for static image analysis (on-cluster) |
| **Logging** | Out of scope | ✔ Prometheus (on-cluster, Developer Perspective dashboard includes PromQL and live data) |
| **Service Mesh** | Out of scope | ✔ Istio, Kiali, Jaeger (on-cluster) |
| **Serverless** | Out of scope | ✔ Knative (on-cluster, Developer Perspective dashboard integration) |
| **Advanced Workloads & Marketplace** | CRDs (v1.12+ recommended) | ✔ Dashboard support for CRDs, Disk storage (volumes and claims), Operator Marketplace (on-cluster), Helm3, KubeVirt (VMs), GPUs |

Red Hat

# The Red Hat Developer Program

Technical developer content that educates; access to free software for development









## Free Products

Free access to downloadable and SaaS software for use in development and pre-production environments.
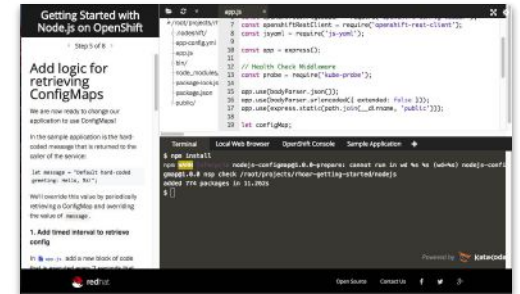
## Red Hat DevNation

Technical talks and hands-on labs delivered virtually and physically. Book a private session for your organization.

## Industry Leading Experts

E-books, quick reference guides, cheat sheets, and how-to articles and videos. Learn new technologies from the experts.

## Labs and Trials

Faster, easier product evaluation and adoption through step-by-step tutorials and labs. Free trials for dev teams.

*Sign up for free at developers.redhat.com*
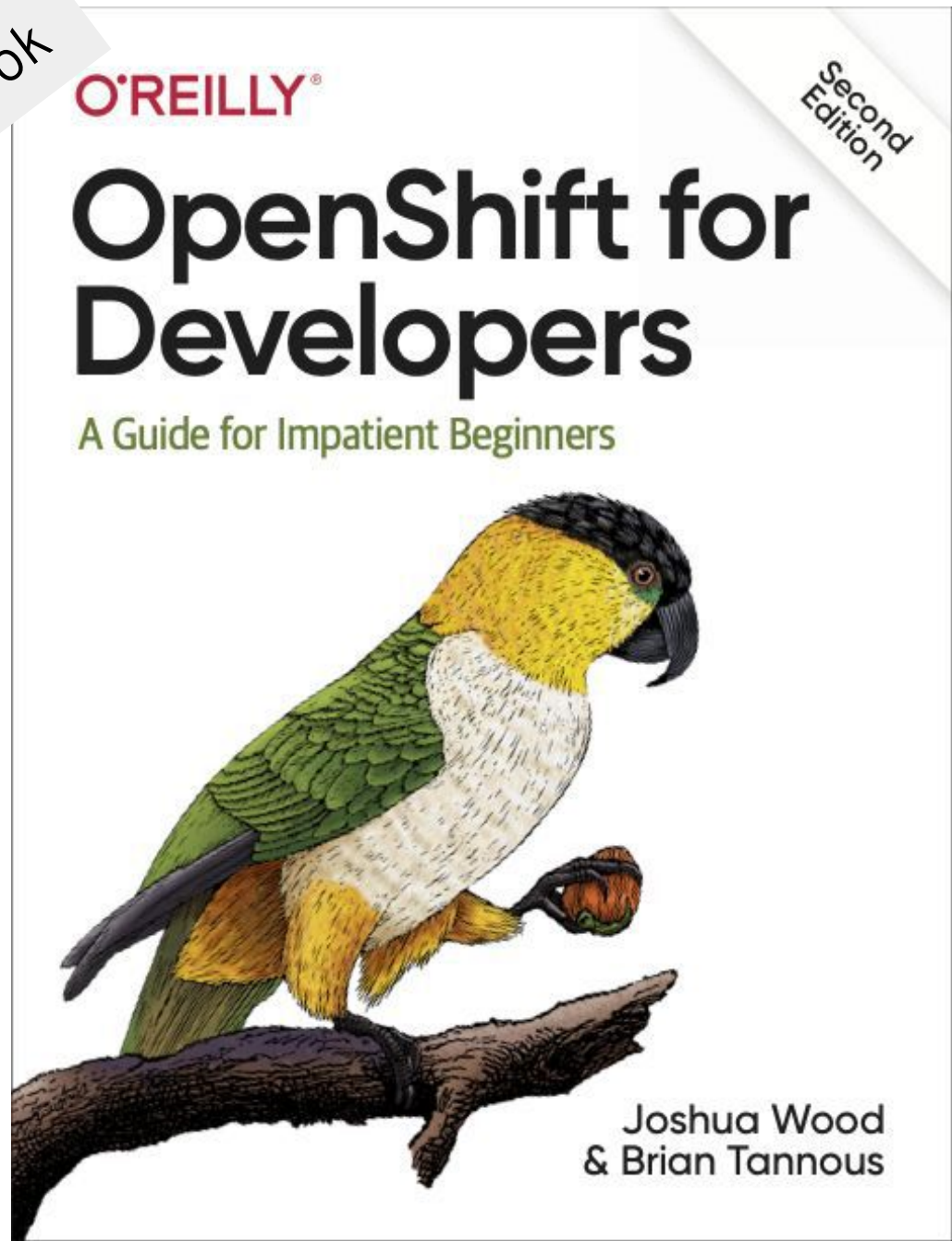
Red Hat

# Developer Sandbox

Get **free access** for renewable **30 days** to a self-service, cloud-hosted **Kubernetes** experience with **Developer Sandbox** for **Red Hat OpenShift**.

**https://developers.redhat.com/developer-sandbox**

```
[your@sandbox ~]$ lscpu
RAM: 7GB
Storage: 15GB
Time limit: 30 days
Awesome: YES
```

free e-book

**O'REILLY®**

Second Edition

# OpenShift for Developers

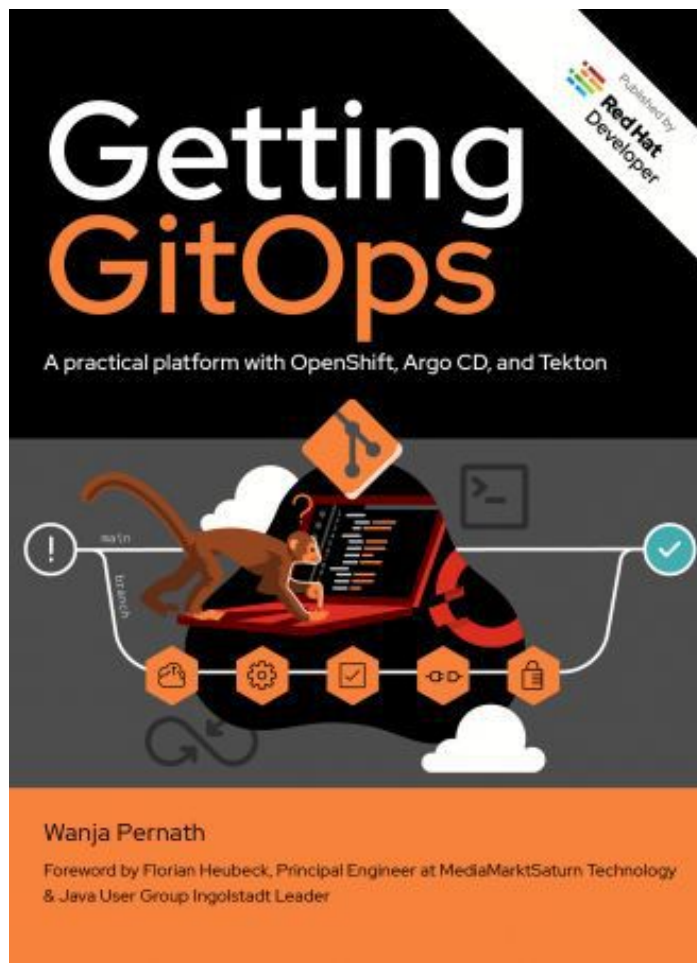## A Guide for Impatient Beginners

Joshua Wood
& Brian Tannous

Download
https://red.ht/3IxJCzY

Red Hat

# Getting GitOps: A practical platform with OpenShift, Argo CD, and Tekton



This book covers:

- How to install and use Quarkus for Java development
- How to configure an application to use a PostgreSQL database in a Kubernetes environment
- Basic Kubernetes files
- OpenShift Templates
- The Kustomize configuration management tool
- The Docker, Podman, Buildah, and Skopeo build tools
- Basic Helm charts and subcharts
- Kubernetes Operators
- CI/CD with Tekton
- CI/CD with OpenShift Pipelines
- GitOps with Argo CD
- Tekton security

https://developers.redhat.com/e-books/getting-gitops-practical-platform-openshift-argo-cd-and-tekton

# 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.

**Corporate**: linkedin.com/company/red-hat
**Developer**: linkedin.com/showcase/red-hat-developer/

**Corporate:** youtube.com/user/RedHatVideos
**Developer**: youtube.com/channel/UC7noUdfWp-ukXUIAsJnSm-Q

**Corporate:** @RedHat
**Developer:** @rhdevelopers