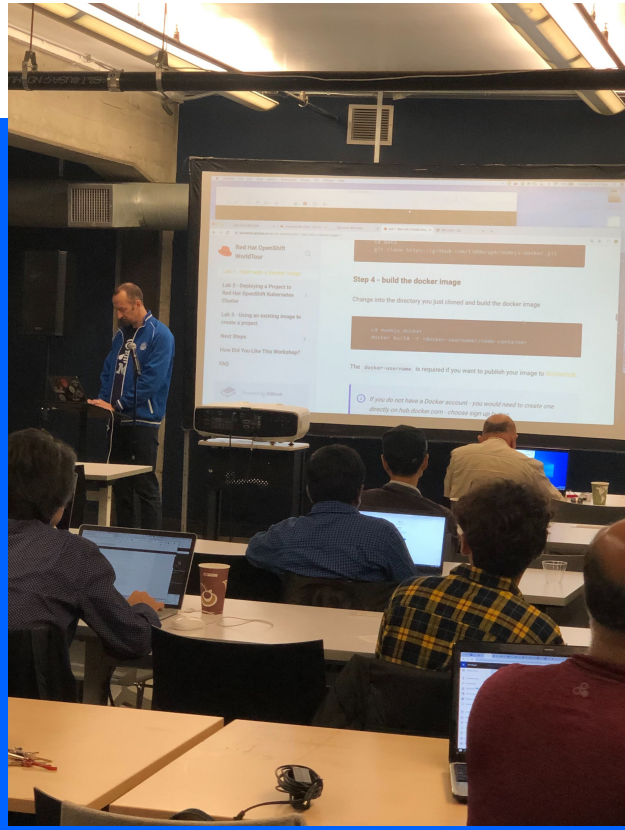
Kafka and Kubernetes OpenShift World Tour

Marek Sadowski

Developer Advocate | IBM San Francisco City Team



ibm.biz/202010-kafka

V1 2020 10 15

IBM Developer

IBM

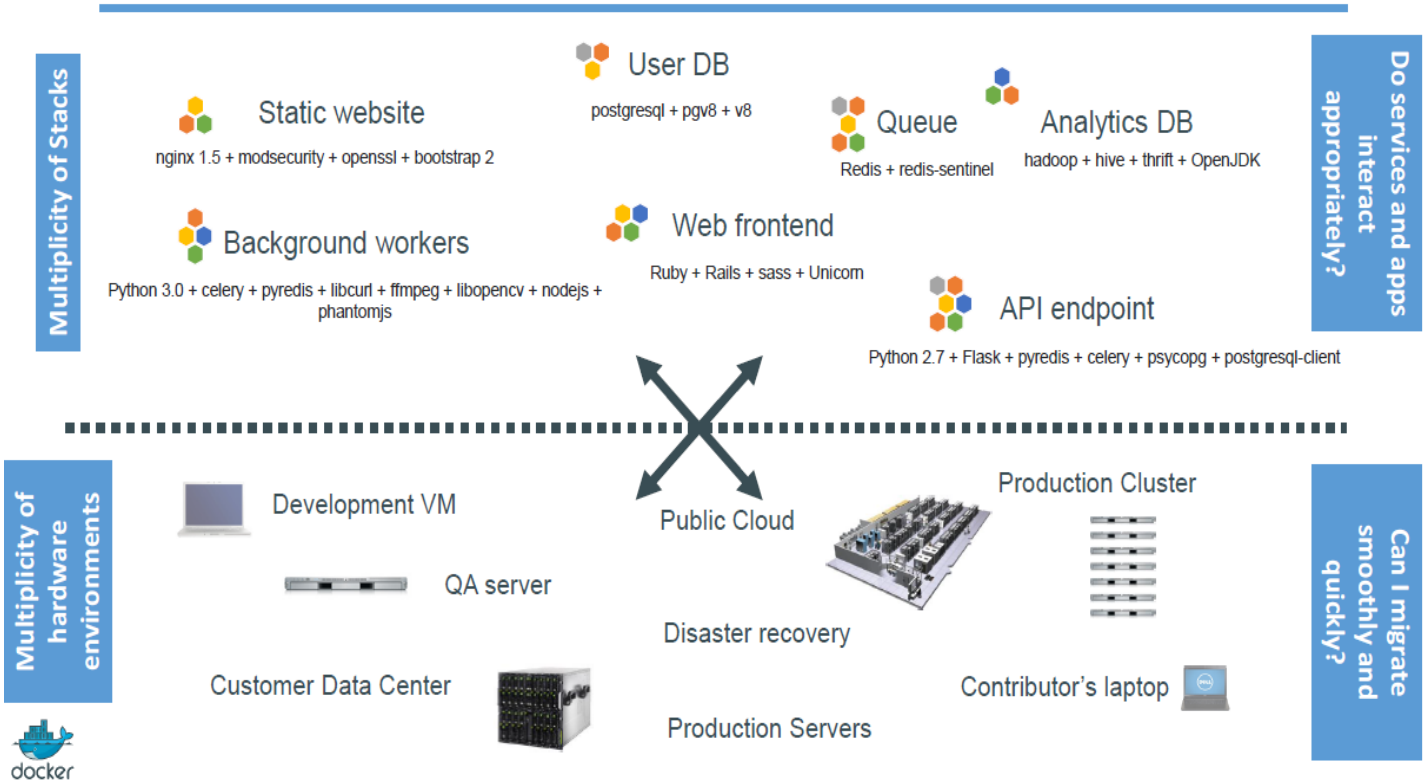
Agenda:

1. Intro to containers
2. containers to host microservices
3. **LAB 0:** Docker build-ship-run
4. orchestration of many containers
5. Kubernetes
6. OKD project - OpenShift
9. Red Hat OpenShift vs Kubernetes
10. **LAB 1:** Kubernetes on OpenShift hands-on
11. **LAB 2:** Kafka on Kubernetes

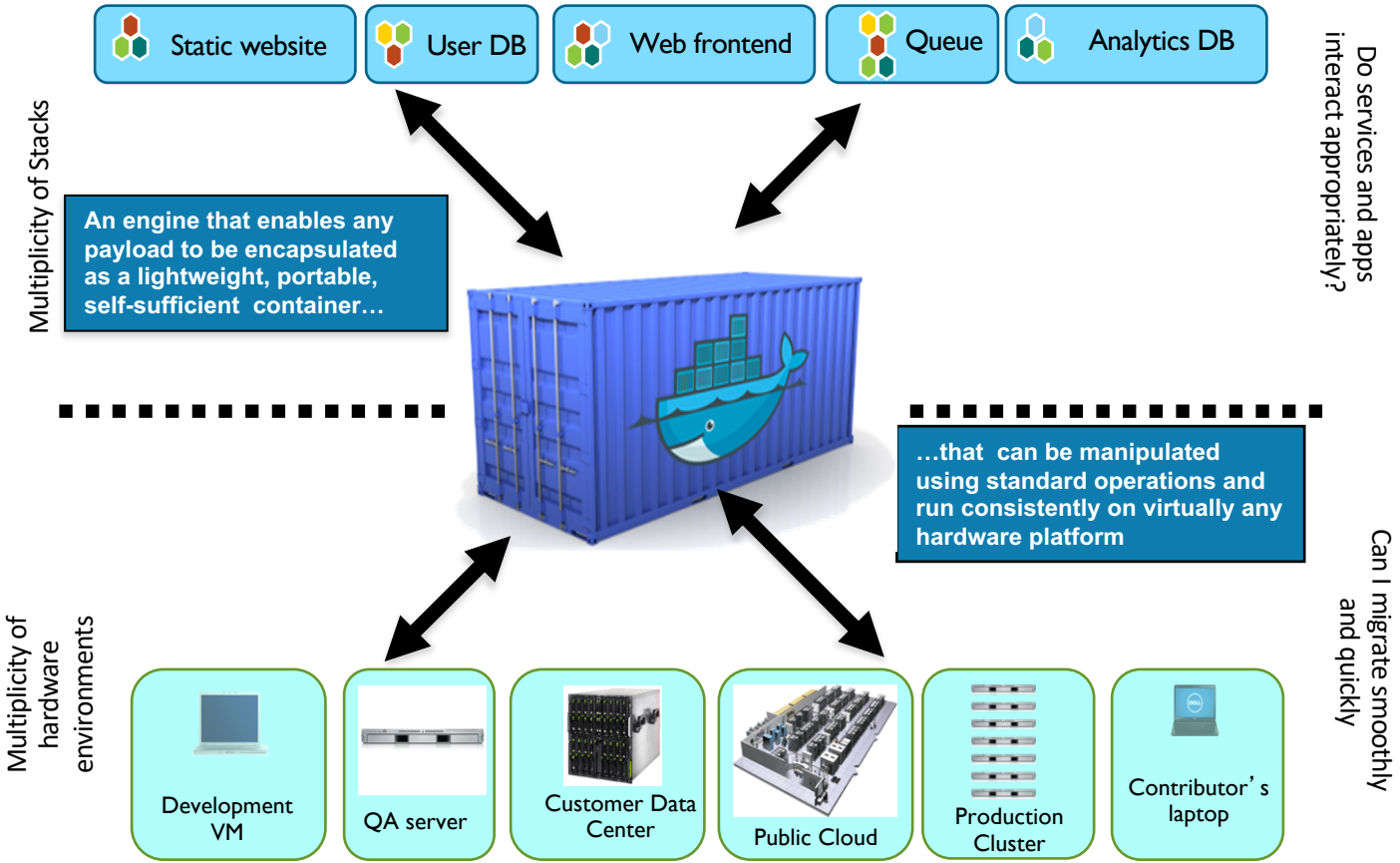
THE POLL

Why Containers?

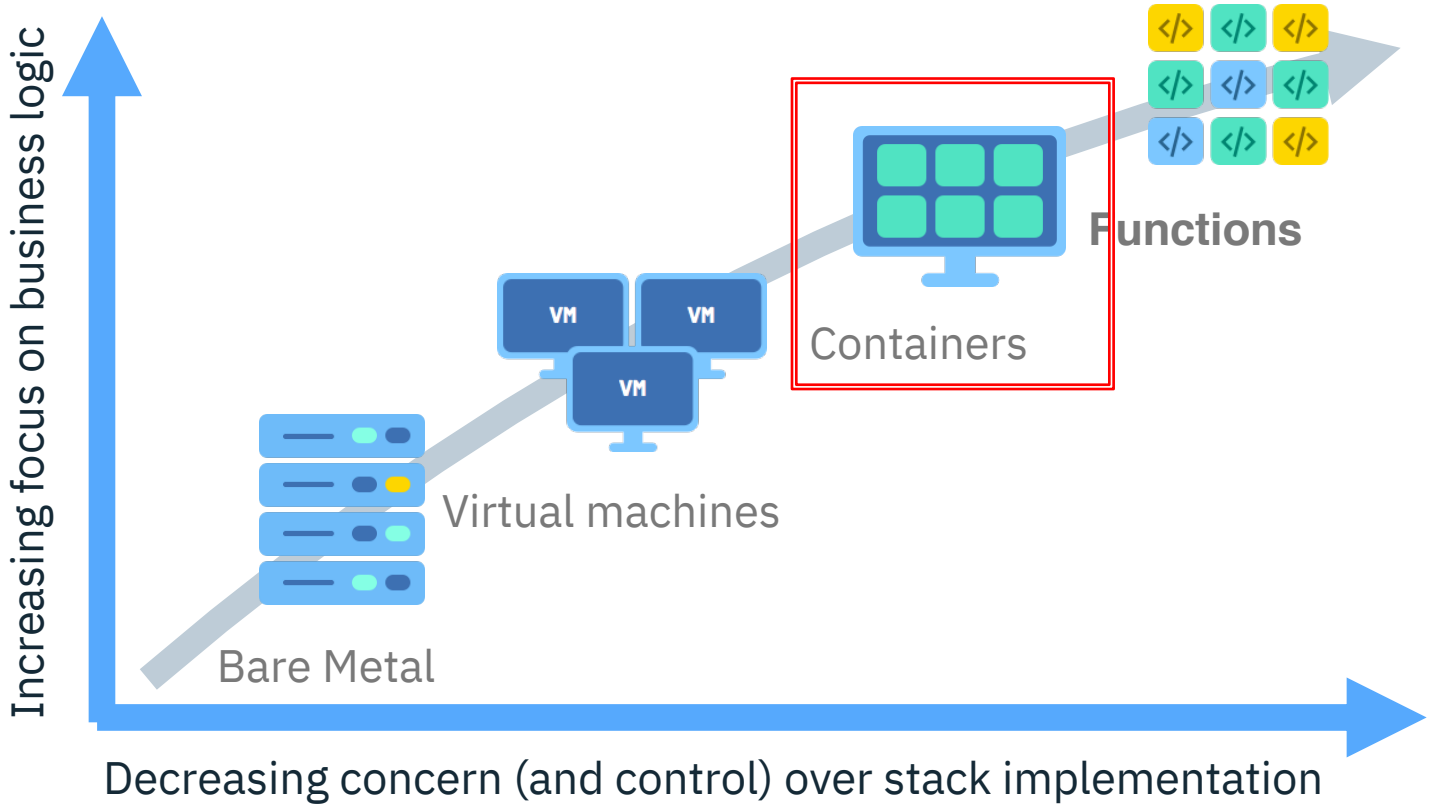
The Challenge



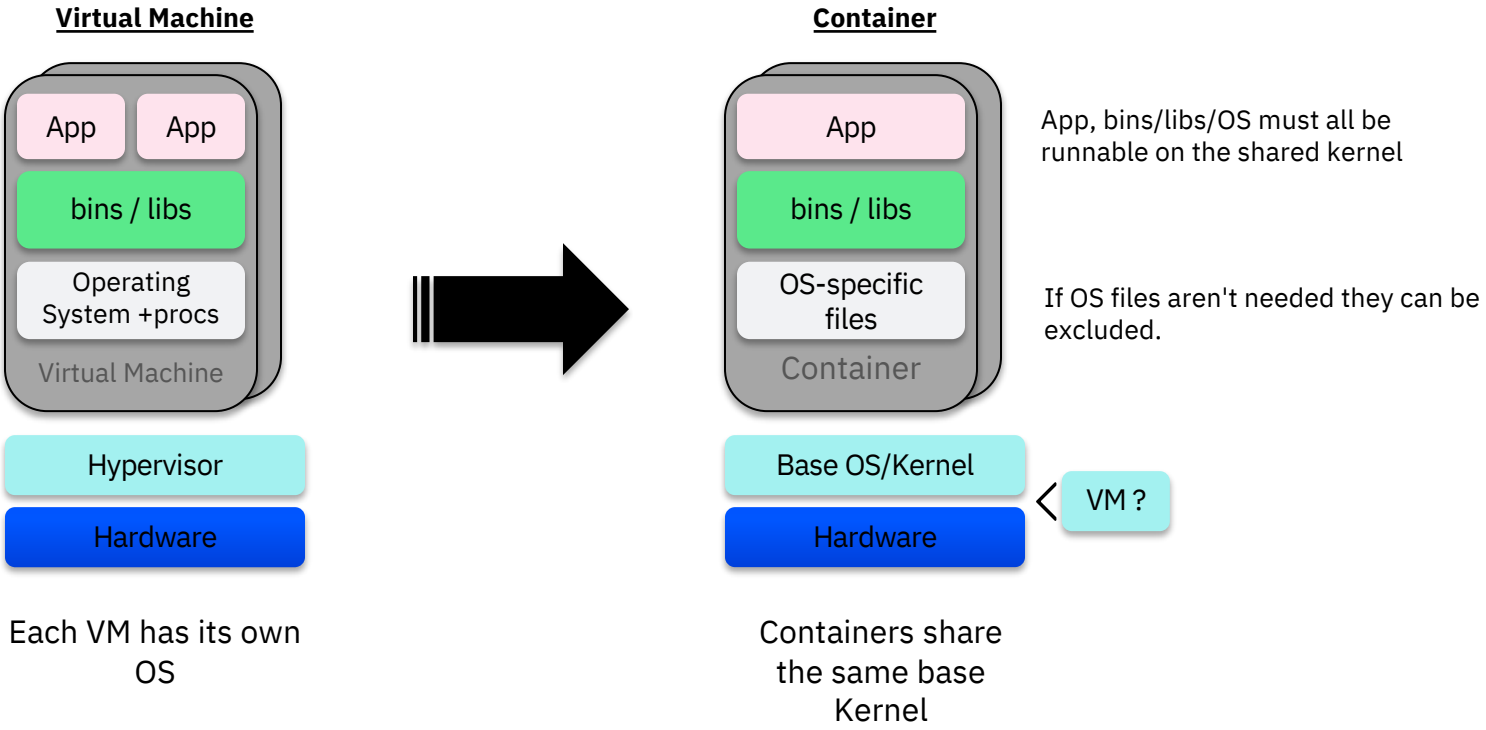
The Solution - A Shipping Container for Code



How we got here

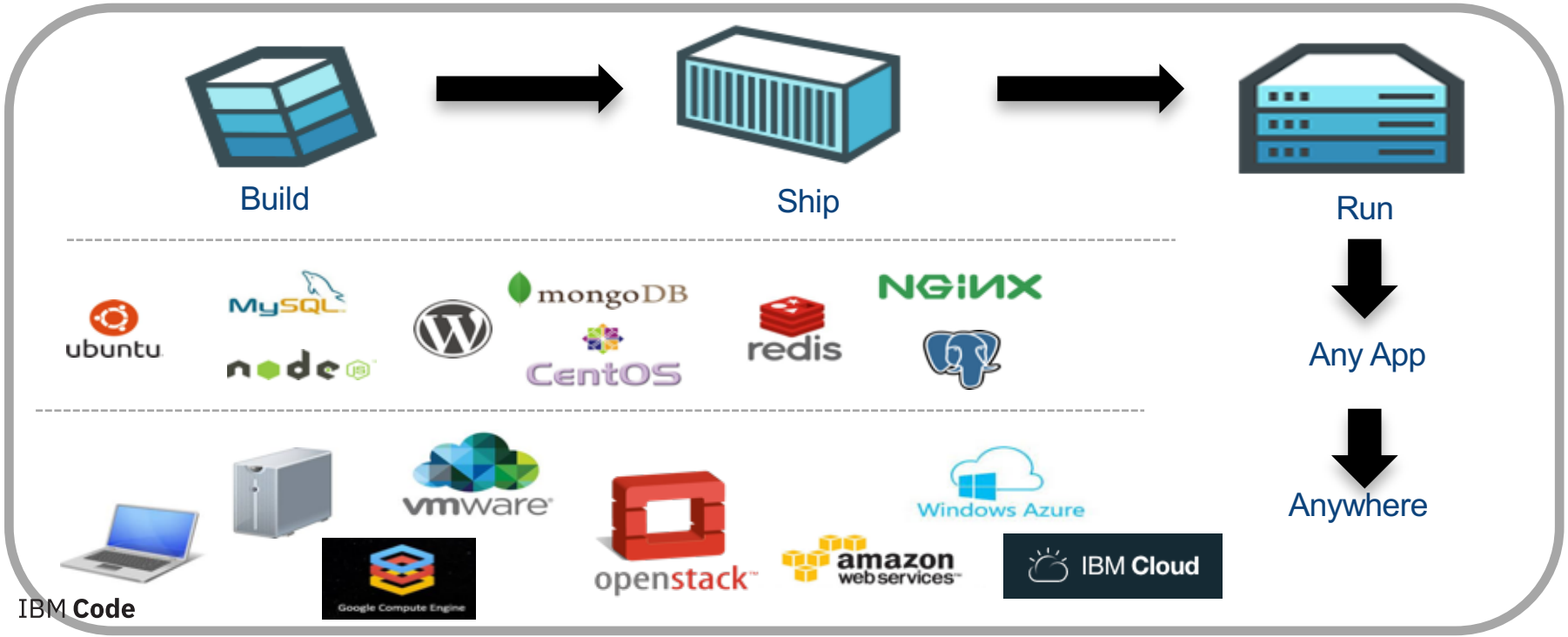


VM vs Container

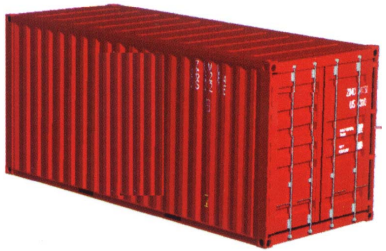


Docker Mission

Docker is an **open platform** for building distributed applications for **developers** and **system administrators**



Dev vs. Ops

Dev	Ops
	
• **Code** • **Libraries** • **Configuration** • **Server runtime** • **OS**	• **Logging** • **Remote access** • **Network configuration** • **Monitoring**

Separation of concerns

A container separates and bridges the Dev and Ops in DevOps

- Dev focuses on the application environment
- Ops focuses on the deployment environment

Docker Component Overview

Docker Engine

- Manages containers on a host
- Accepts requests from clients
  - REST API
- Maps container ports to host ports
  - E.g. 80 → 3582

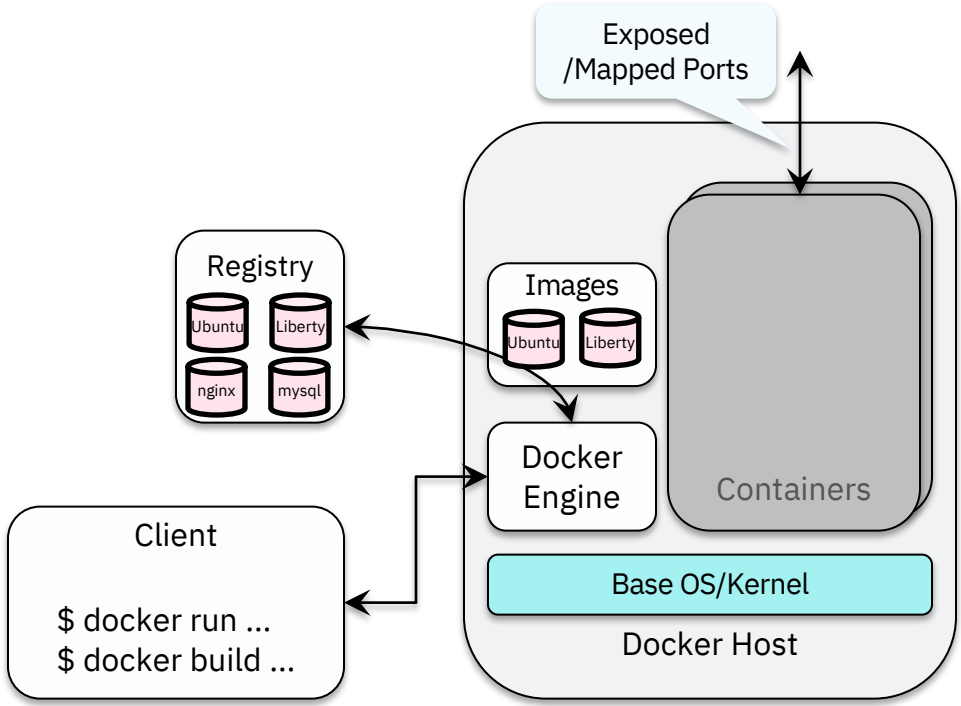
Images

Docker Client

- Drives engine
- Drives "builder" of Images

Docker Registry

- Image DB



Shared / Layered / Union Filesystems

Docker uses a copy-on-write (union) filesystem

New files(& edits) are only visible to current/above layers

Layers allow for reuse

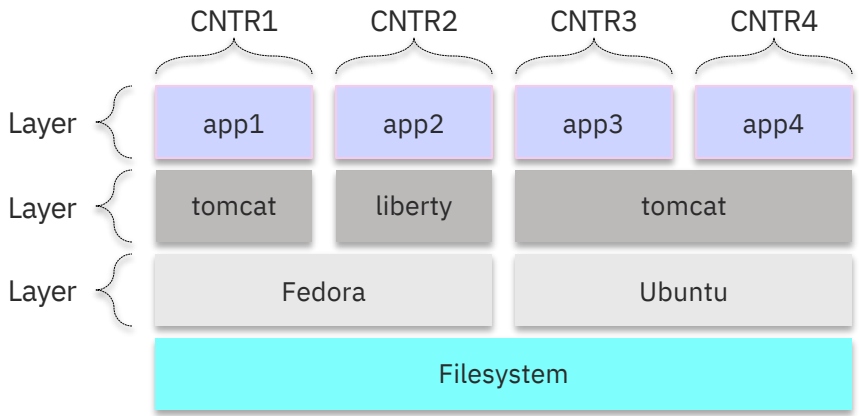
– More containers per host

– Faster start-up/download time

Images

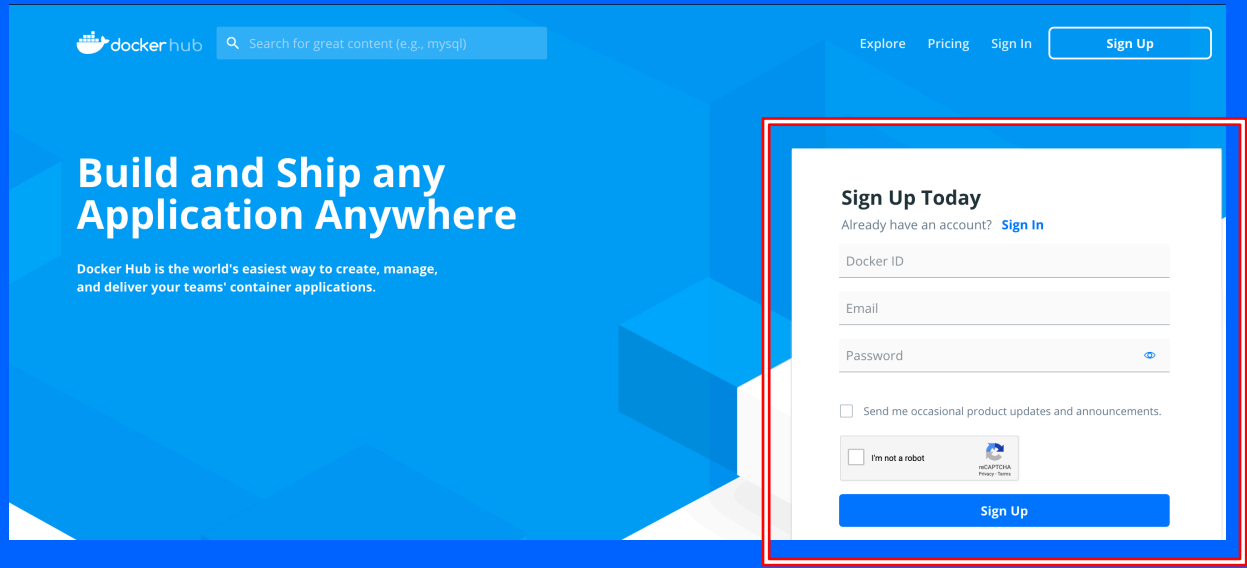
– Tarball of layers

Think: Transparencies on projector



Hands-on

Download FREE Docker Desktop and use your terminal on your computer
or just sit back and watch



Our First Container

```
$ docker run ubuntu echo Hello World
Hello World
```

What happened?

- Docker created a directory with a "ubuntu" filesystem (image)
- Docker created a new set of namespaces
- Ran a new process: `echo Hello World`
  - Using those namespaces to isolate it from other processes
  - Using that new directory as the "root" of the filesystem (`chroot`)
- That's it!
  - Notice as a user I never installed "ubuntu"
- Run it again - notice how quickly it ran

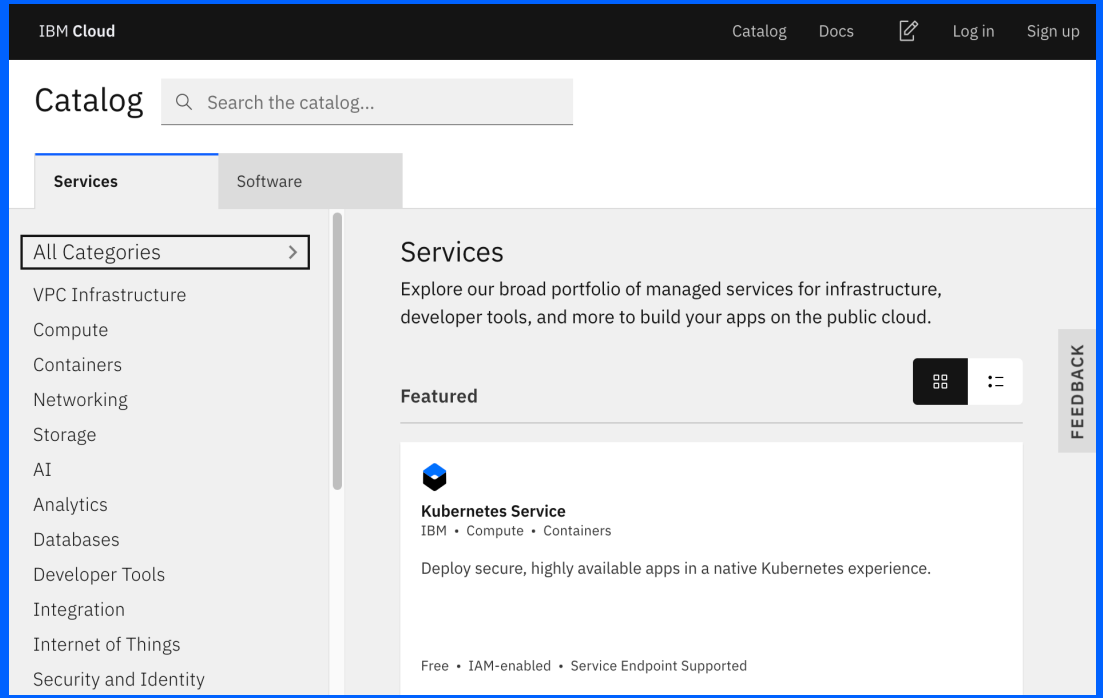
```
Mareks-MBP:~ mareksadowski$ docker run ubuntu echo Hello World
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
6b98dfc16071: Pull complete
4001a1209541: Pull complete
6319fc68c576: Pull complete
b24603670dc3: Pull complete
97f170c87c6f: Pull complete
Digest: sha256:5f4bdc3467537cbbe563e80db2c3ec95d548a9145d64453b06
Status: Downloaded newer image for ubuntu:latest
Hello World
Mareks-MBP:~ mareksadowski$ docker run ubuntu echo Hello World
Hello World
Mareks-MBP:~ mareksadowski$
```

LAB 0
building a translation service with Watson:

http://ibm.biz/202010-kafka-lab0

IBM Cloud Catalog

https://cloud.ibm.com/catalog



Second part – the orchestration of successful API deployment

Containers

Everyone’s container journey starts with one container....

Containers

At first the growth is easy to handle....

Containers

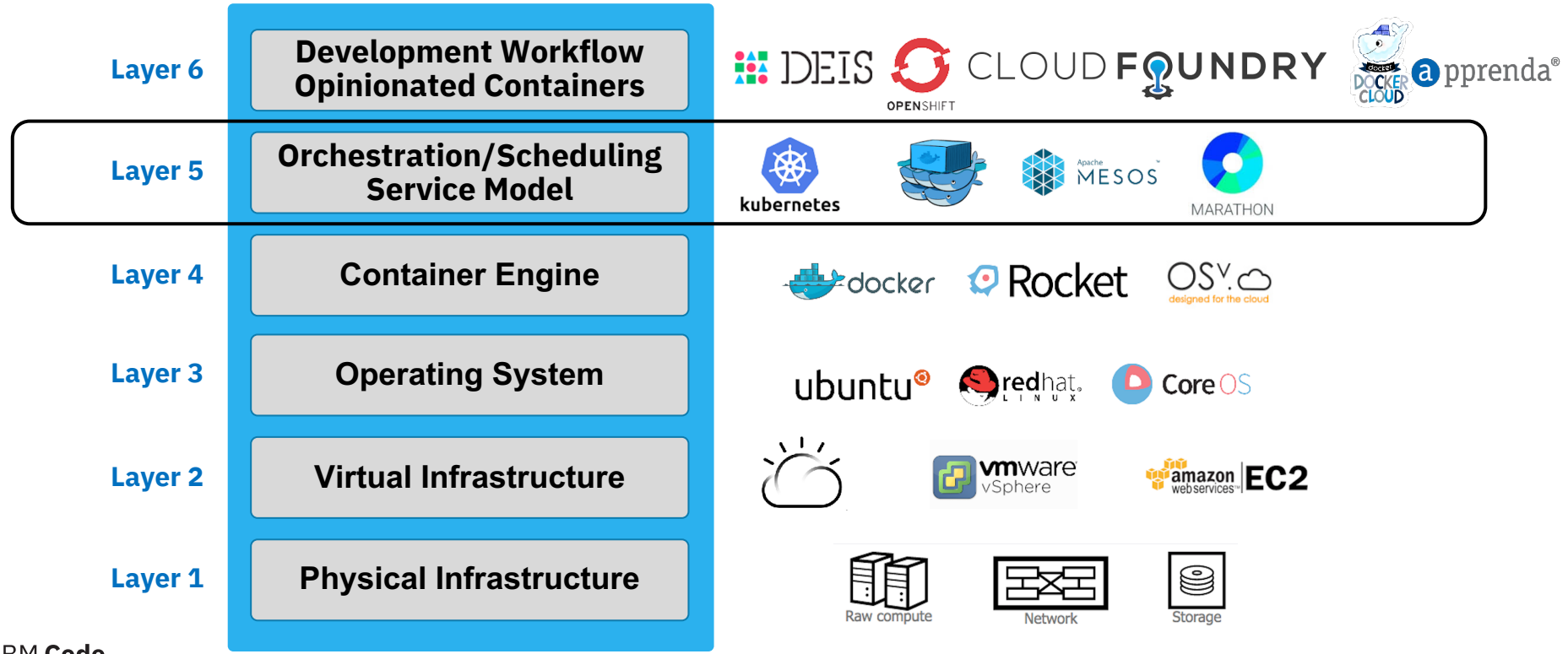
But soon it is overwhelming... chaos reigns

Container Orchestration

Allows users to define how to coordinate the containers in the cloud when the multi-container packaged application is deployed.

- Scheduling
- Cluster management
- Service discovery
- Provisioning
- Monitoring
- Configuration management

Container Ecosystem Layers

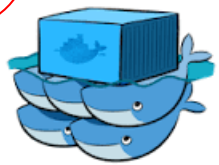


Kubernetes – Mesos – Marathon – Docker Swarm

Kubernetes is a cluster manager for containers (large deployments)

MESOS is a distributed system kernel that will make your cluster look like one giant computer system to all supported frameworks and apps that are build to be run on mesos.

(Kubernetes can be run on Mesos)



Swarm has a YAML-based deployment model, auto-healing of clusters, overlay networks with DNS, high-availability through the use of multiple masters, and network security using TLS with a Certificate Authority (1-10 containers)

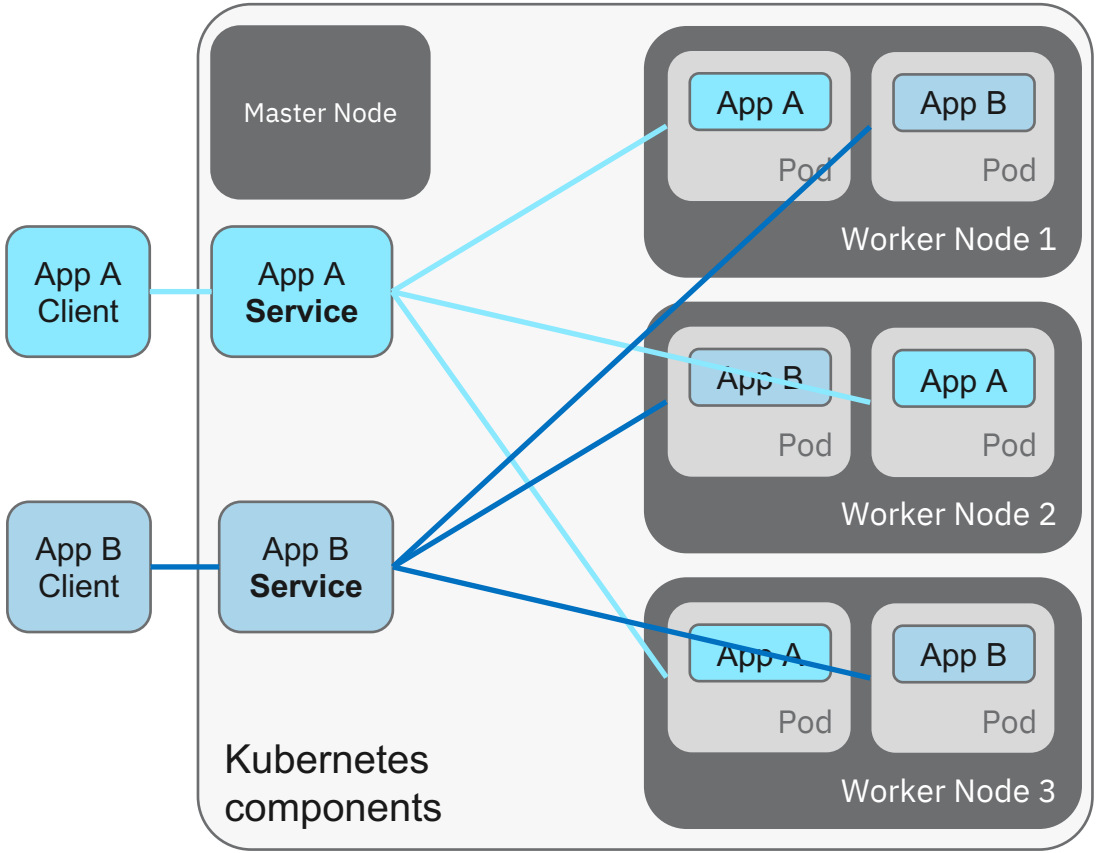
Marathon is a cluster-wide init and control system for running Linux services in cgroups and Docker containers, and it is run on top of Mesos

What is Kubernetes?

- Container orchestrator
- Manage applications, **not machines**
- Designed for extensibility
- Open source project managed by the Linux Foundation

Kubernetes Architecture: Workloads

- Container
  - Packaging of an app
- Pod
  - Unit of deployment
- Service
  - Fixed endpoint for 1+ pods



Kubernetes

Intelligent Scheduling

Self-healing

Horizontal scaling

Service discovery & load balancing

Automated rollouts and rollbacks

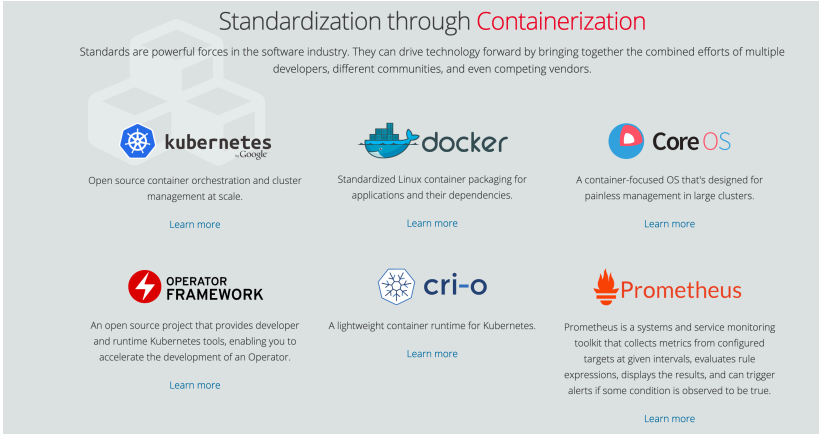
Secret and configuration management

ORIGIN COMMUNITY DISTRIBUTION OF KUBERNETES

okd

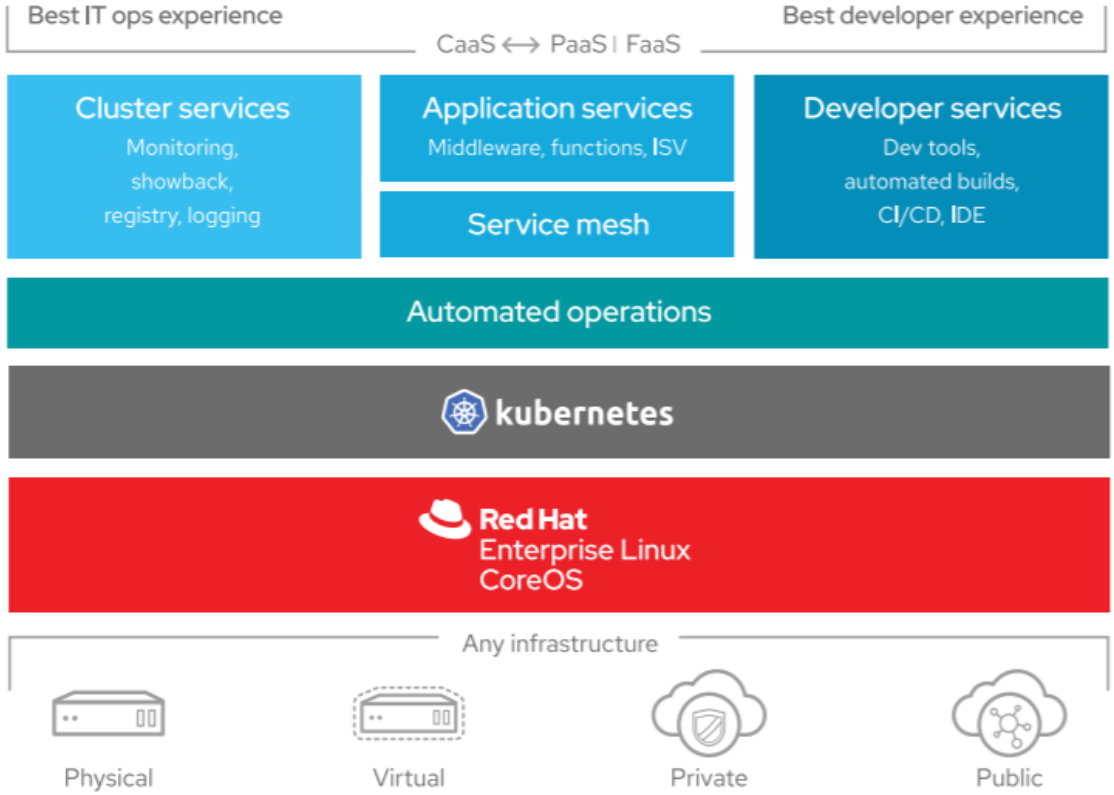
The Origin Community Distribution of Kubernetes that powers Red Hat OpenShift.

Built around a core of OCI container packaging and Kubernetes container cluster management, OKD is also augmented by application lifecycle management functionality and DevOps tooling. OKD provides a complete open source container application platform.

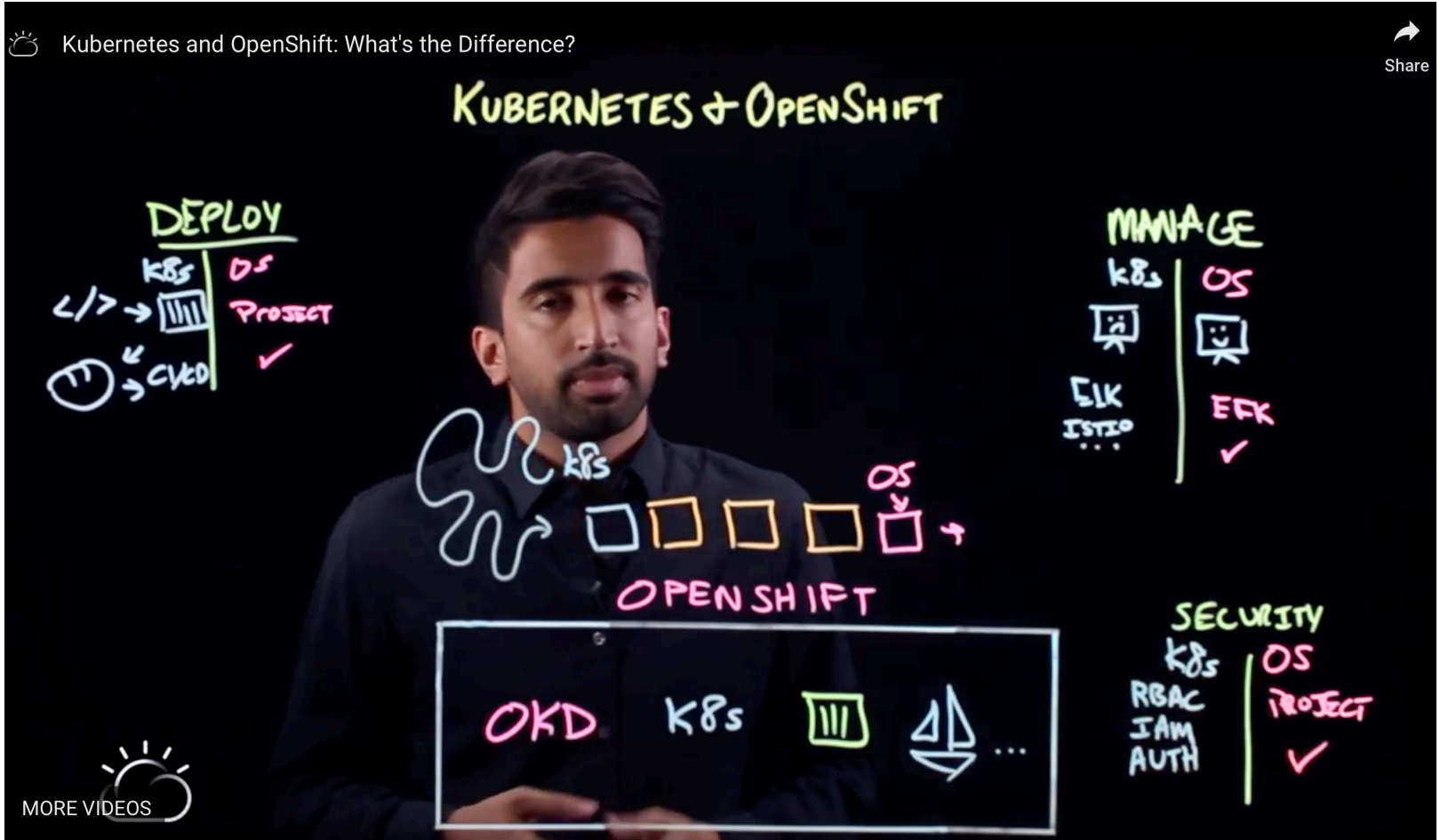


Open Source | Secure | Fully Certified | Portable

RED HAT® OPENSHIFT



#IBMDeveloper


Kubernetes and OpenShift: What's the Difference?
Share
KUBERNETES & OPENSHIFT
DEPLOY
k8s
OS
PROJECT
CI/CD
MANAGE
k8s
OS
ELK
ISTIO
EFK
k8s
OS
OPENSHIFT
OKD
k8s
SECURITY
k8s
OS
RBAC
IAM
AUTH
PROJECT
MORE VIDEOS

Lab 1: follow steps here: http://ibm.biz/202010-kafka-lab1

URL: https://oskafka.mybluemix.net

Key: oslab

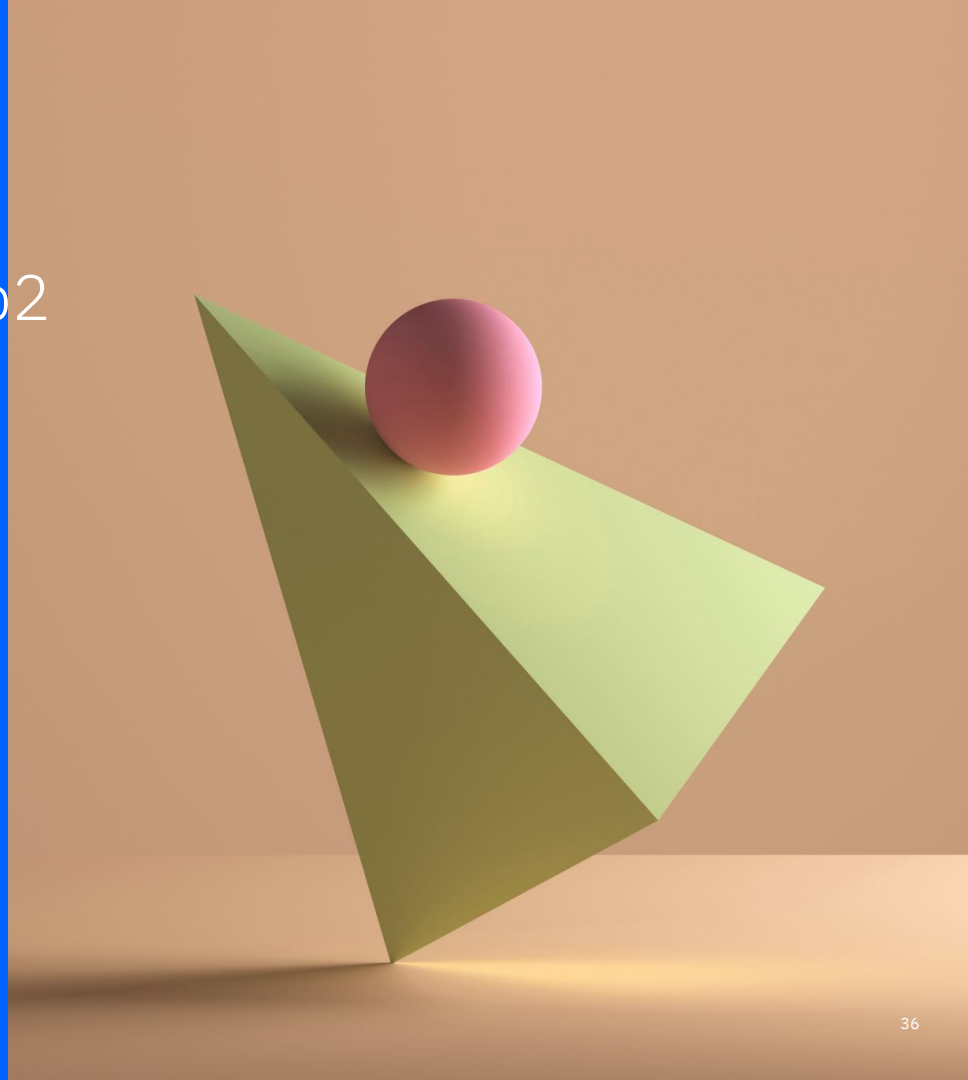
Stretch LAB with Red Hat

- https://medium.com/@blumareks/running-a-microservice-containers-on-kubernetes-straight-from-a-github-ff73047e877b

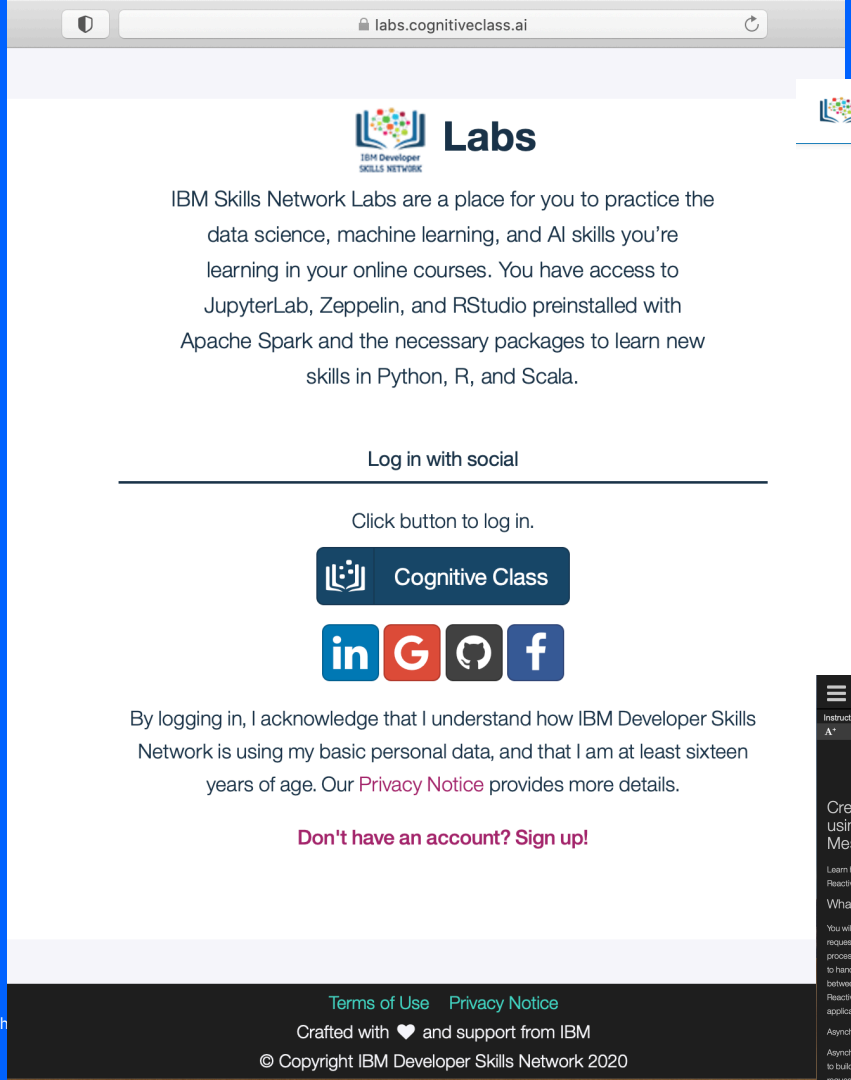
LAB 2: Kafka and Kubernetes

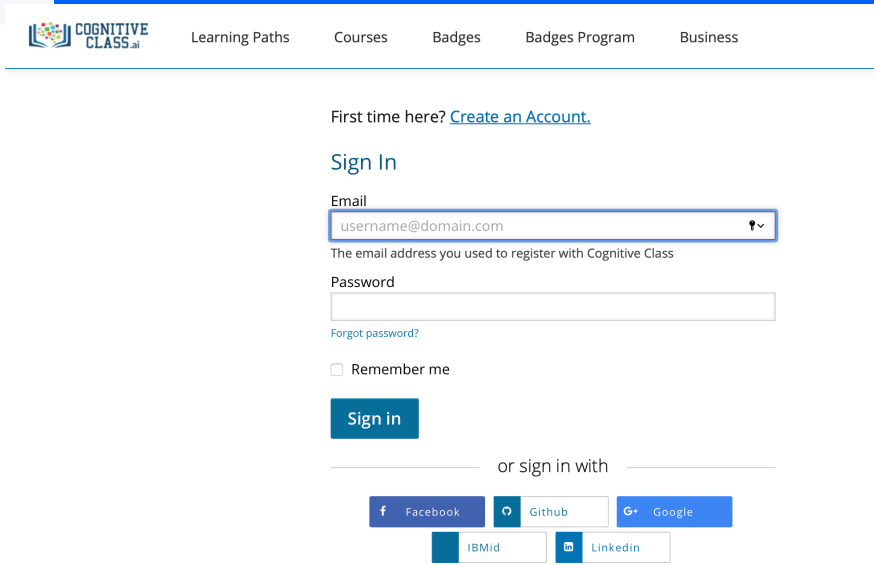
ibm.biz/202010-kafka-lab2

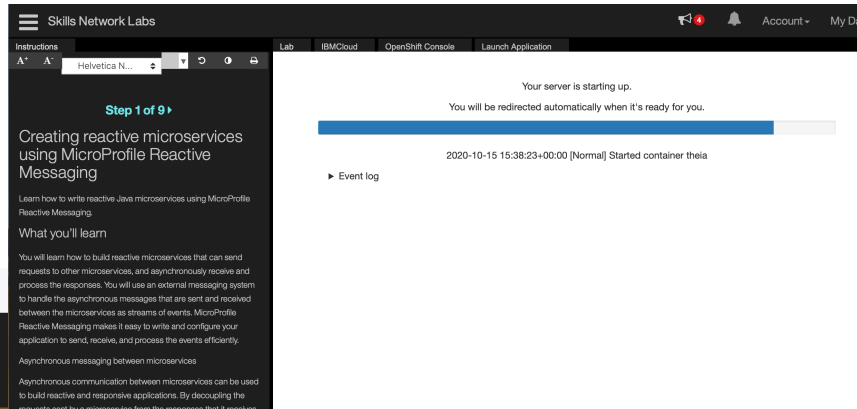
http://ibm.biz/202010-kafka-lab2-env



LAB2 :

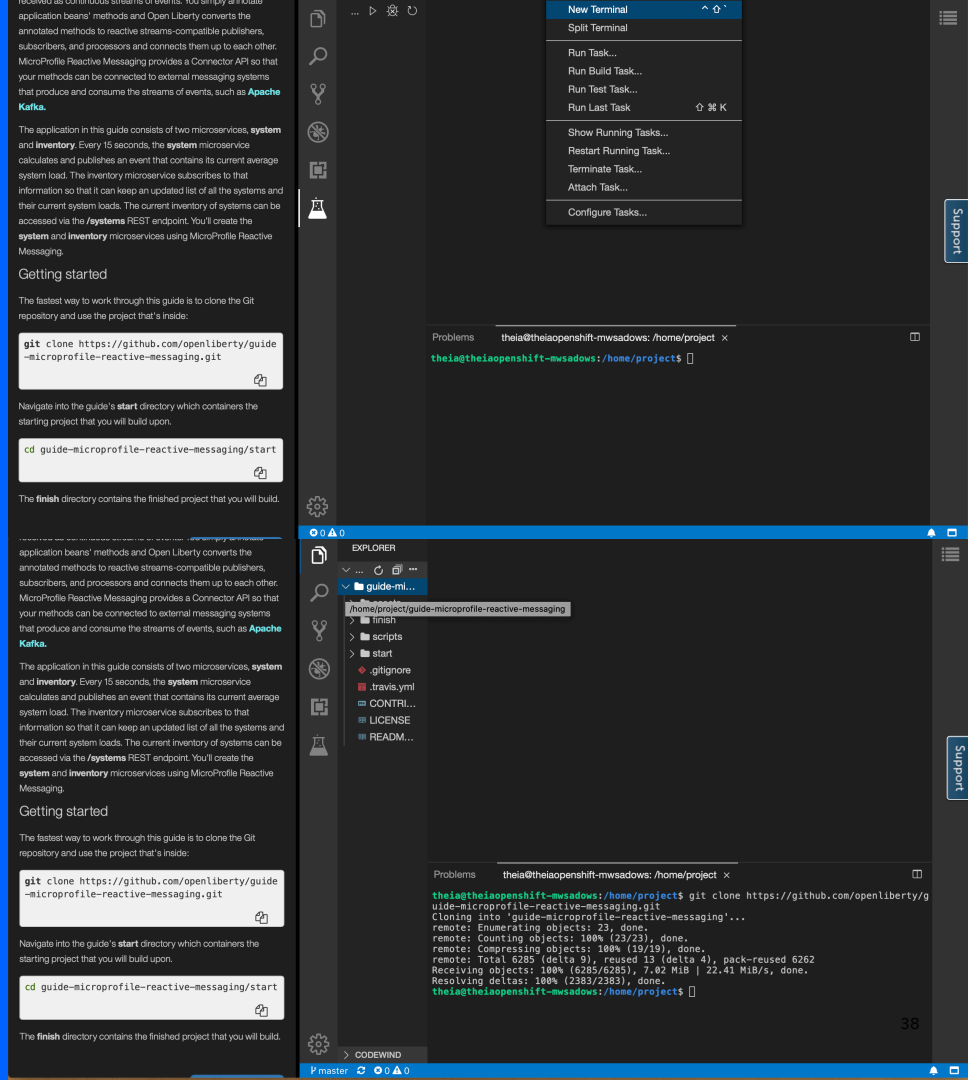

labs.cognitiveclass.ai
Labs
IBM Developer SKILLS NETWORK
IBM Skills Network Labs are a place for you to practice the data science, machine learning, and AI skills you're learning in your online courses. You have access to JupyterLab, Zeppelin, and RStudio preinstalled with Apache Spark and the necessary packages to learn new skills in Python, R, and Scala.
Log in with social
Click button to log in.
Cognitive Class
By logging in, I acknowledge that I understand how IBM Developer Skills Network is using my basic personal data, and that I am at least sixteen years of age. Our Privacy Notice provides more details.
Don't have an account? Sign up!
Terms of Use
Privacy Notice
Crafted with ♥ and support from IBM
© Copyright IBM Developer Skills Network 2020


COGNITIVE CLASS.ai
Learning Paths
Courses
Badges
Badges Program
Business
First time here? Create an Account.
Sign In
Email
username@domain.com
The email address you used to register with Cognitive Class
Password
Forgot password?
Remember me
Sign in
or sign in with
Facebook
Github
Google
IBMid
Linkedin

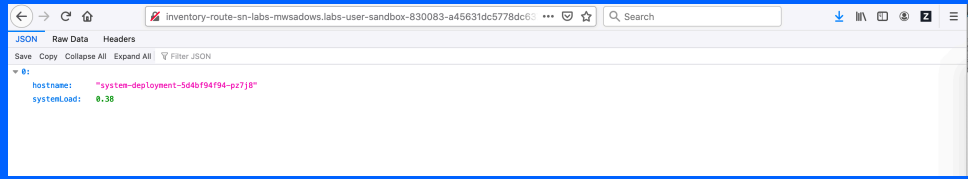

Skills Network Labs
Account
Instructions
Lab
IBMCloud
OpenShift Console
Launch Application
Step 1 of 9
Creating reactive microservices using MicroProfile Reactive Messaging
Learn how to write reactive Java microservices using MicroProfile Reactive Messaging.
What you'll learn
You will learn how to build reactive microservices that can send requests to other microservices, and asynchronously receive and process the responses. You will use an external messaging system to handle the asynchronous messages that are sent and received between the microservices as streams of events. MicroProfile Reactive Messaging makes it easy to write and configure your application to send, receive, and process the events efficiently.
Asynchronous messaging between microservices
Asynchronous communication between microservices can be used to build reactive and responsive applications. By decoupling the
Your server is starting up.
You will be redirected automatically when it's ready for you.
2020-10-15 15:38:23+00:00 [Normal] Started container theia
Event log

LAB2: doing the lab

The new terminal – scroll for instructions



LAB 2: tips (use OpenShift console to find out the route to the app)




```
Problems    theia@theiaopenshift-mwsadows: /home/project/guide-microprofile-reactive-messaging/start ×

theia@theiaopenshift-mwsadows:/home/project/guide-microprofile-reactive-messaging/start$ sed -i 's=$NA
MESPACE_NAME='"$NAMESPACE_NAME"'=g' openshift.yaml
theia@theiaopenshift-mwsadows:/home/project/guide-microprofile-reactive-messaging/start$ oc apply -f o
penshift.yaml
deployment.apps/system-deployment created
deployment.apps/inventory-deployment created
deployment.apps/kafka-deployment created
deployment.apps/zookeeper-deployment created
service/system-service created
service/kafka-service created
service/inventory-service created
service/zookeeper-service created
route.route.openshift.io/inventory-route created
theia@theiaopenshift-mwsadows:/home/project/guide-microprofile-reactive-messaging/start$ oc get routes
NAME              HOST/PORT
                                        PATH    SERVICES            PORT    TERMINATION    WILDCA
RD
inventory-route   inventory-route-sn-labs-mwsadows.labs-user-sandbox-830083-a45631dc5778dc6371
theia@theiaopenshift-mwsadows:/home/project/guide-microprofile-reactive-messaging/start$
theia@theiaopenshift-mwsadows:/home/project/guide-microprofile-reactive-messaging/start$
```

Thank you!

Join me for My office hours – use slack:

https://join.slack.com/t/biz-on-cloud/shared_invite/zt-hop799ua-MmK32III9VS6xxqMfkzrlg

Marek Sadowski

Developer Advocate

@blumareks