INNOVATION THAT MATTERS

CASCON

2016 Oct. 31 - Nov. 2

Cognitive Computing and Cyber Physical Systems

Hands-On: Docker Containers

Build and package a highly scalable microservice application using Docker containers (#402)

Serjik Dikaleh, IBM
Chris Felix, IBM
Dharmesh Mistry, IBM
Salman Moghal, IBM
Ozair Sheikh, IBM

Agenda

Overview

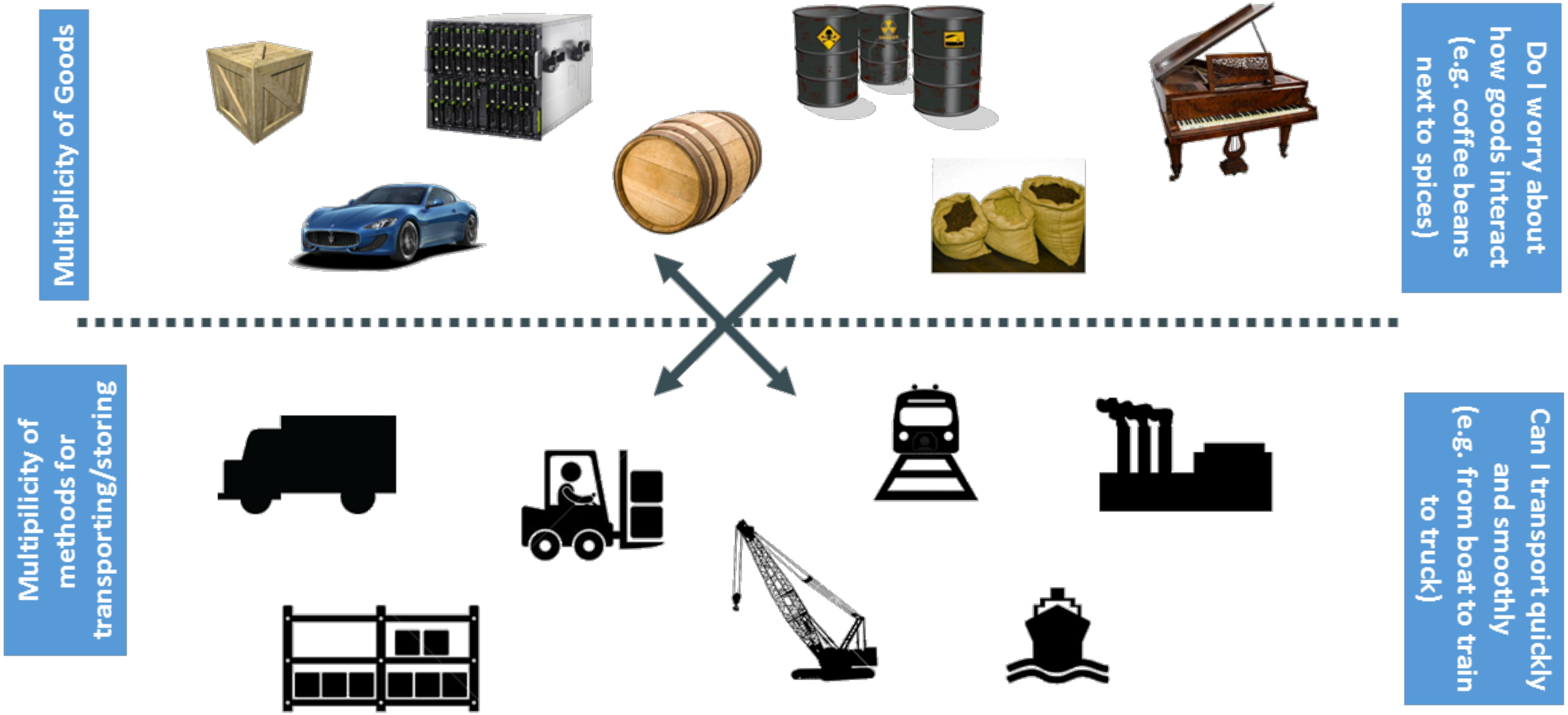
The overview section will provide some overview on containers in general, background and history on containers, talk about IBM’s involvement, and then finally talk about what Docker is.

Labs

The hands-on portion of the session will focus on two labs and a demo using the example of a microservice running in multiple containers with Docker.

Overview and Background

What are containers?



Multiplicity of Goods

Do I worry about how goods interact (e.g. coffee beans next to spices)

Multipilicity of methods for transporting/storing

Can I transport quickly and smoothly (e.g. from boat to train to truck)

Source: http://slidedeck.io/pointful/docker-intro

What are containers?

Multiplicity of Goods

A standard container that is loaded with virtually any goods, and stays sealed until it reaches final delivery.

Do I worry about how goods interact (e.g. coffee beans next to spices)

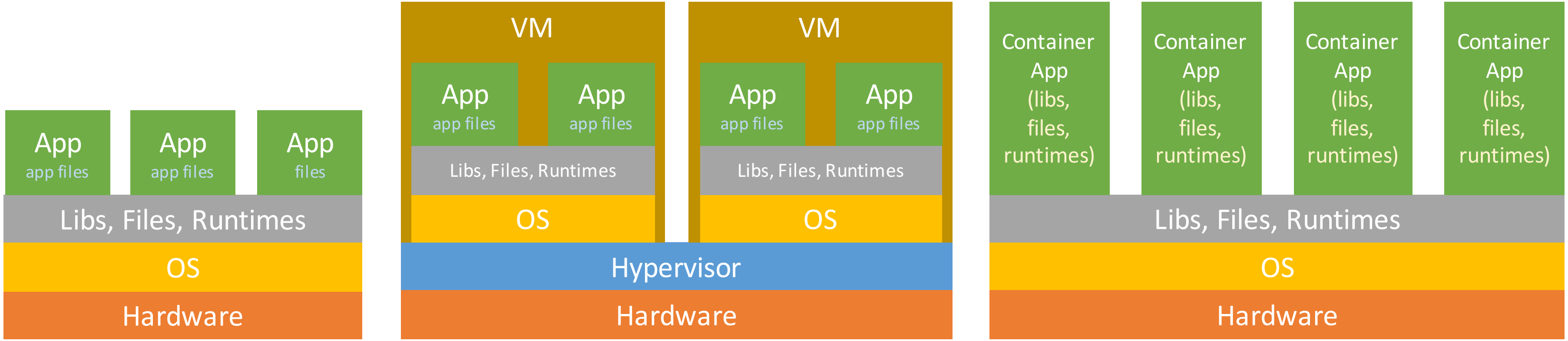
Multiplicity of methods for transporting/storing

…in between, can be loaded and unloaded, stacked, transported efficiently over long distances, and transferred from one mode of transport to another

Can I transport quickly and smoothly (e.g. from boat to train to truck)

Source: http://slidedeck.io/pointful/docker-intro

Background and History



Early days in the Data Center: A dedicated OS for the hardware and several apps deployed in that OS sharing the libraries and runtimes

Virtualization in the Data Centers: A hypervisor enables sharing of the hardware resources across multiple virtual machines with their own definition of CPU, memory, storage and dedicated OS

Modern Data Centers: Each container defines its own sandbox where it loads its required versions of libraries, distinct file system isolation (through layers) and application components and runtimes.

How is IBM involved?

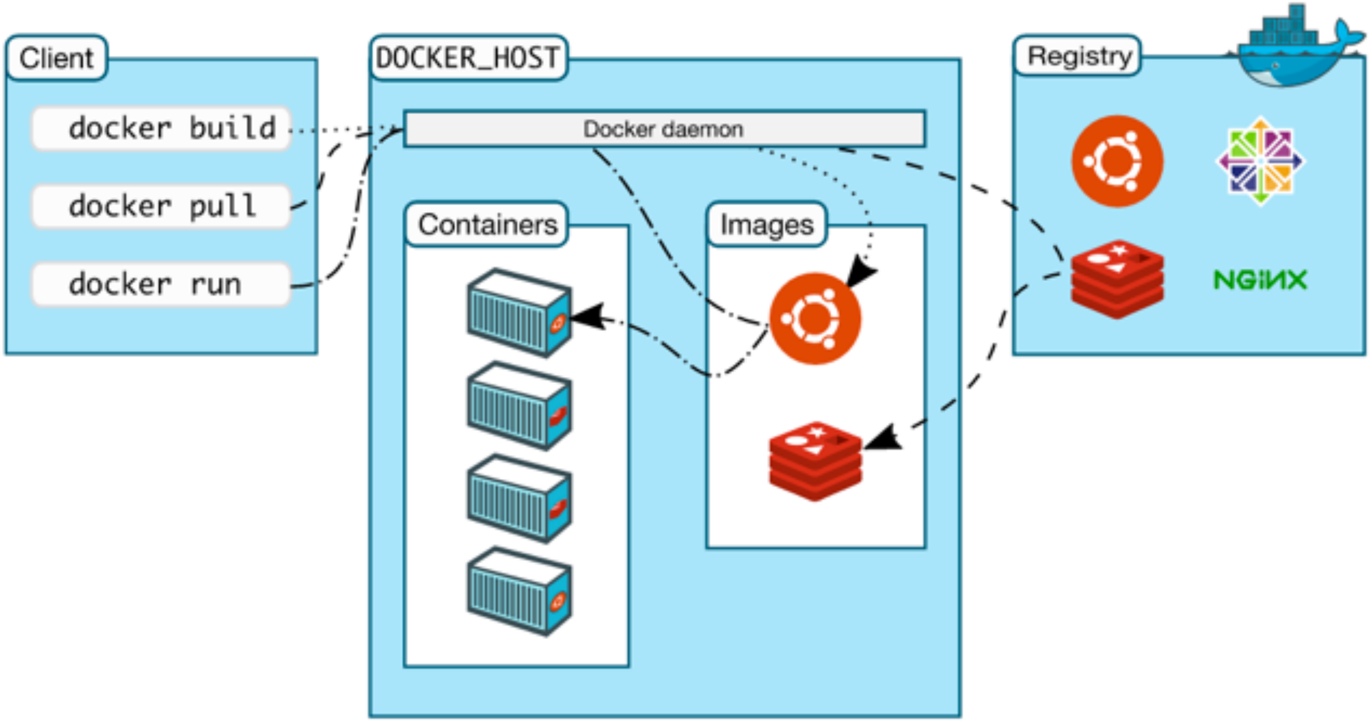
IBM CONTAINERS

- Fully managed **hosted runtime**
- **Integrated** logging, monitoring, load balancing, registry and auto-scaling
- Advanced **security** features
- Built using **Docker** technology

OPEN CONTAINER INITIATIVE

docker

What is Docker?



Source: https://docs.docker.com/engine/understanding-docker/#/what-is-dockers-architecture

Why Containers?

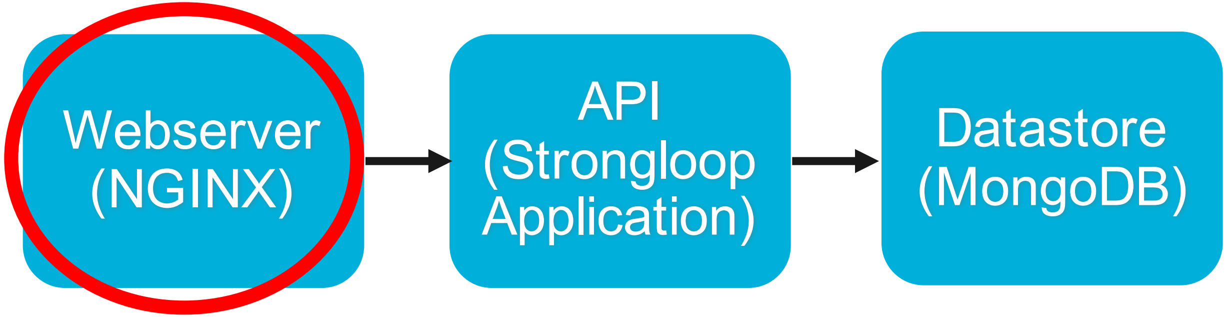
- Portable runtime environment for your applications
- No need to worry about missing dependencies, packages, and other concerns during deployments
- Each app is run in it’s own isolated container, and as a result you can run different versions of libraries for each app
- You can automate testing, integration, packaging of your application
- Consistent and repeatable process for app lifecycle
- No more inconsistencies between dev, test, production
- Speeds up continuous deployment and continuous integration

Labs and Demo

Today you will go through two labs and a demo to build a containerized microservice. This is a good real world example of how containers could be used.

Lab 1 – Basic Docker

- In lab exercise 1 you will go through the fundamentals of Docker to create a container for a simple web server. You will learn about Docker images, containers, and basic command line interface interactions to work with Docker. This container will be the starting point of our multi container solution for our microservice
  - https://github.com/cloud-coder/docker-lab-2016/blob/master/part-1/lab-instructions.md
- Timeframe: 30 mins



Webserver (NGINX)

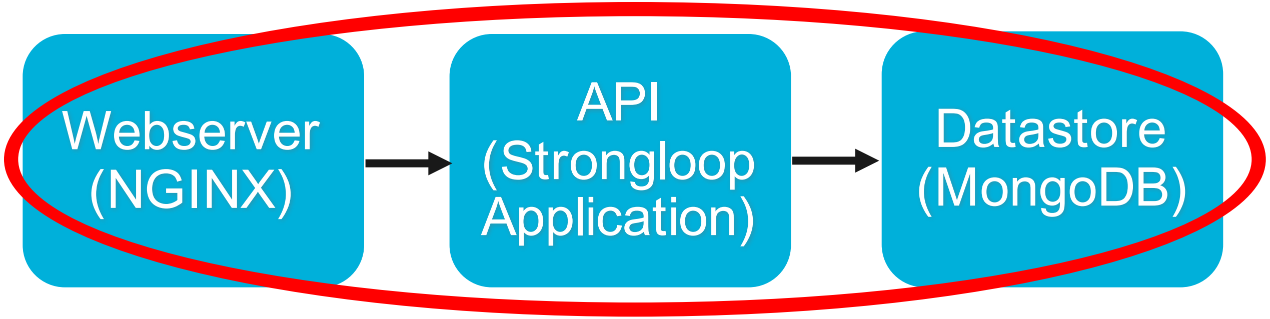
API (Strongloop Application)

Datastore (MongoDB)

Lab 2 - Docker Compose

- In this lab, we will take the first lab a step further. You will go through using Docker compose to build multiple containers to work together for your micro service. You will use Compose to create three containers; a container for your web server, one for your database and one for your service.
  - https://github.com/cloud-coder/docker-lab-2016/blob/master/part-2/lab-instructions.md

- Timeframe: 30 mins



Demo – Docker Swarm

- The third part will be a live demo of using Docker Swarm for clustering. This will turn a group of VMs into a single virtual Docker engine.
  - https://github.com/cloud-coder/docker-lab-2016/tree/master/part-3
  - URL has steps and links to videos to see the demo
- Timeframe: 45 minutes

Lab Information

- GitHub URL: https://github.com/cloud-coder/docker-lab-2016
- Login to VM
  - User: student
  - Password: cas2016con