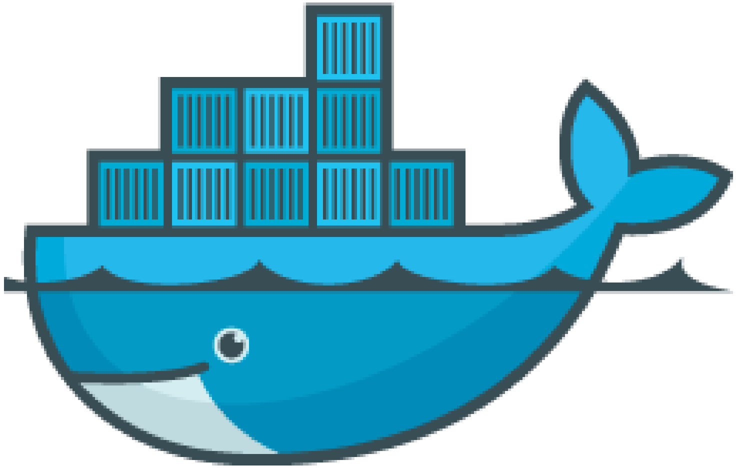


Docker

By

ADITYA PRABHAKARA

Introducing Myself

Aditya S P (sp.aditya@gmail.com)
Freelance trainer and technologist

Boring Stuff about me:

- 14+ years of experience in development and training
- Started with Java, moved to Android and now working on Big Data Technologies

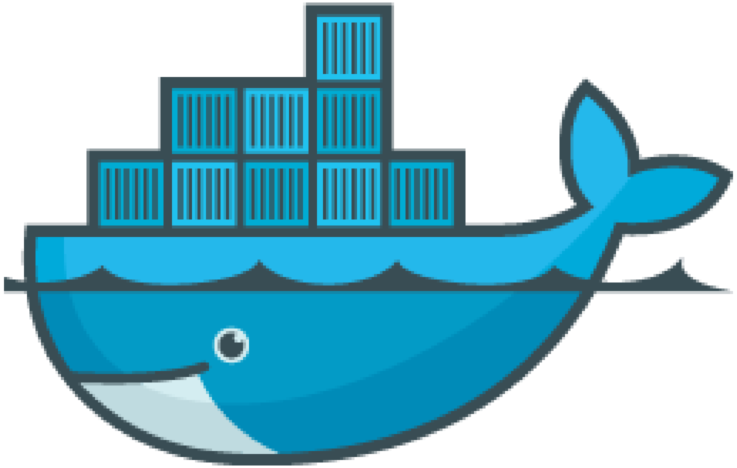
Interesting Things about me:

- Actually Nothing !

Getting to know you

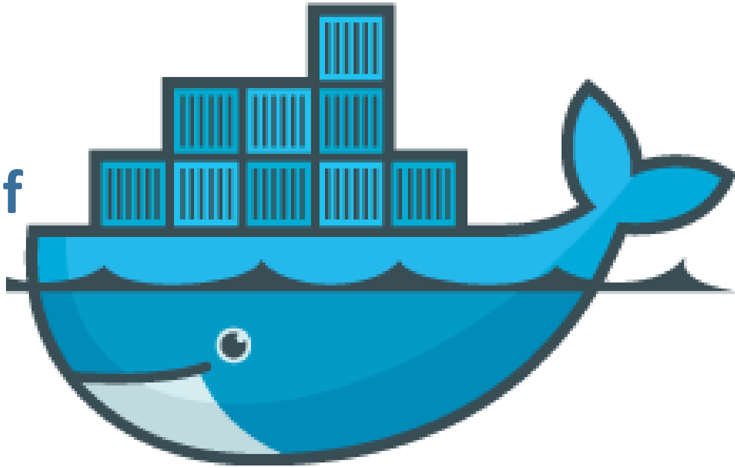
Agenda

- ➢ Introduction to DevOps
- ➢ Dockers
- ➢ Introduction to Kubernetes



Course Objectives

- A good understanding of DevOps
- A good fundamental understanding of Docker
- Where does docker fit in the DevOps Movement
- Understanding of role of Kubernetes



Chapter: Introduction

Docker – Why Now?

- Speed. Speed. Speed.
- Value movement dev-> test-> prod easier and faster
- Portability
- Reduce complexity of developing code for distributed systems
- Reduce complexity of deploying code to the cloud
- For a later time - Docker's founder and CTO Solomon Hykes
- https://www.youtube.com/watch?v=3N3n9FzebAA

Basics first - Virtualization

Docker – Different Versions

- https://www.docker.com/get-docker
- Community Edition and Enterprise Edition
- Stable and Edge

- Stable vs. Edge Cont.
- Edge (beta) released monthly,
- Stable quarterly
- Edge gets new features first, but only supported for a month
- Stable rolls in three months of Edge features

Docker – Setup

- Docker toolbox install

Docker – initial commands

- docker version
  - verified it's working
- docker info
  - most config values
- docker command line structure
- docker (options)

Chapter: Container

Container

- Basic Building block
- Let us get a container running and then we will connect the dots
- Execute the command

```
docker container run nginx
```

First Container Run:What just happened?

Container vs VM

- Not really related with one another but just helps to make some sense

Knowing more about a Container

- docker container stats <container id>
- docker container inspect <container id>
- docker container top <container id>

Interactive Container

- docker container run –it nginx bash
- docker container exec –it <container id>

Try this out !

“alpine” is light weight linux distribution , run an alpine container interactively

Chapter: Images

What is an image

- Application binaries
- Application dependencies
- Some meta data about what to run and how to run
- Not a full fledged OS – No kernel No drivers
- Where are these images stored?

Introduction to docker hub

- What is Docker Hub
- How to find images
- How do we say an image is good!
- Versions of images
- What are official images
- Download images

Working with images

- Pull an image
- Pull based on a tag

Images and layers

- Union file system concept
  - Layers of files and meta data
  - docker image history nginx
  - Saves space as it reuses the layers

Layered Visualization

Image and push

- An image has no real name as such
- It is uniquely identified through user/image:tag
- I can retag an existing image and push to my repository
- Only official images do not have username every other image has a user id behind it

Chapter: Networking

Container Network

- An image has no real name as such
- Each container connected to a private virtual network "bridge"
- Each virtual network routes through NAT firewall on host IP
- All containers on a virtual network can talk to each other without -p
- Best practice is to create a new virtual network for each app:
    - network “my_weblayer" for mysql and php/apache containers
    - network “my_mongo_rest" for mongo and nodejs containers

Docker network commands

- docker network ls
- docker network inspect bridge
  - Check the containers running
  - Check the ip address
- docker network create my_own_network