

Docker

Docker Introduction

Usage of Docker

Docker architecture

Docker Terminologies

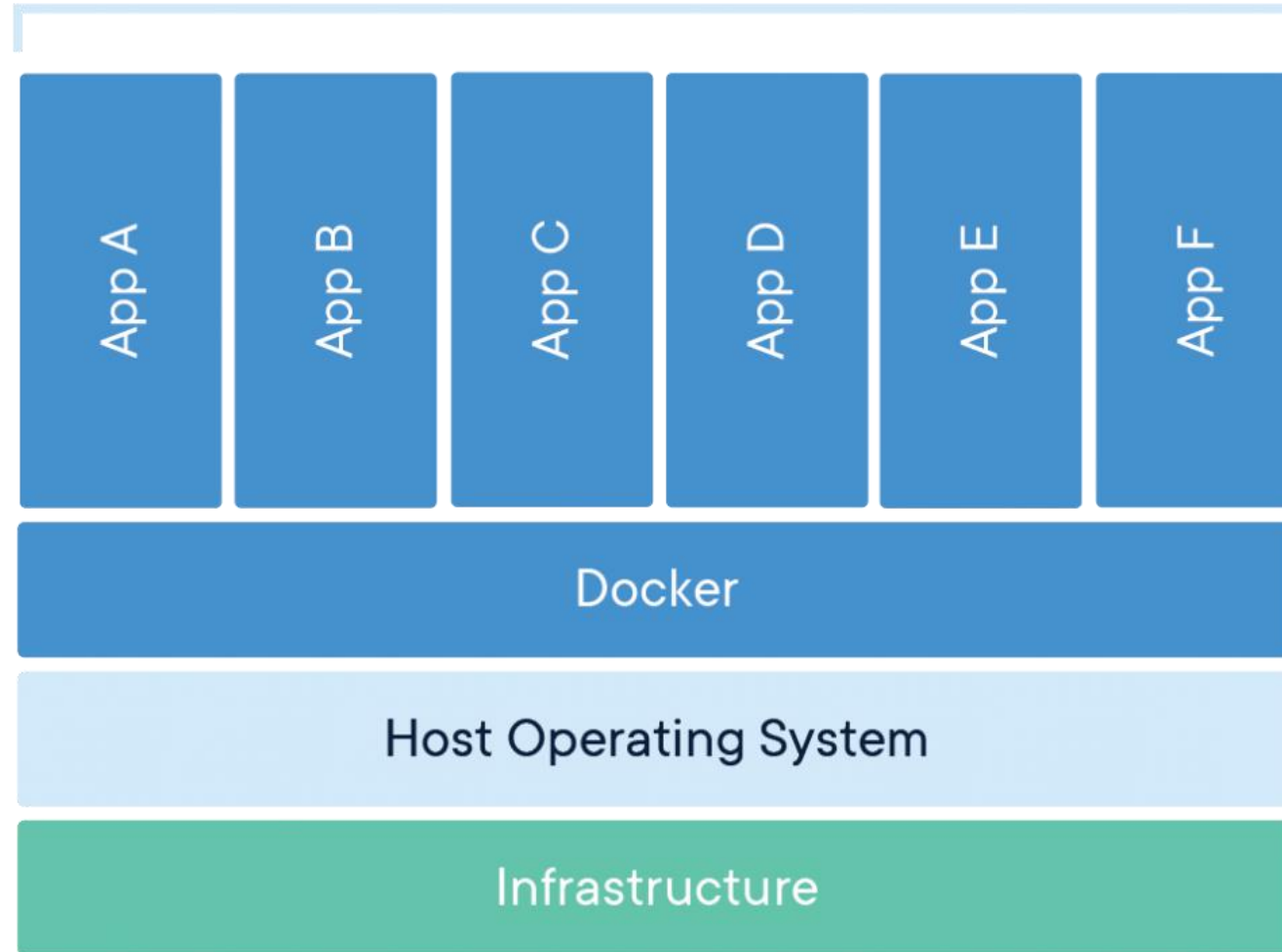
Docker Image Repository

Virtual Machines Vs Docker Containers

Docker Introduction

- Docker is a software that performs OS level virtualization, also known as “containerization”
- Docker is not a virtual machine
- It is a popular tool to build, deploy and run applications using containers
- Apps are packaged in containers that can be run **on any OS**
- Dockers benefits include
 - Works in any machine
 - No compatibility issues
 - Predictable behavior
 - Deploy in seconds
 - Easier to maintain and deploy
 - Works with any language, any OS, any technology.

Containerized Applications



Usage of Docker

- Fast, consistent delivery of your applications
 - Containers are great for continuous integration and continuous delivery (CI/CD) workflows
- Responsive deployment and scaling
 - Docker's container-based platform allows for highly portable workloads.
 - Docker containers can run on a
 - Developer's local machine
 - On physical or virtual machines
 - On cloud providers
 - On a mixture of environments
- Running more workloads on the same hardware

Docker Architecture

- Docker uses a client-server architecture.
- The Docker client talks to the Docker daemon, which does building, running, and distributing your Docker containers.
- The Docker client and daemon can run on the same system, or you can connect a Docker client to a remote Docker daemon.
- The Docker client and daemon communicate using a REST API, over UNIX sockets or a network interface.

Client

`docker build`

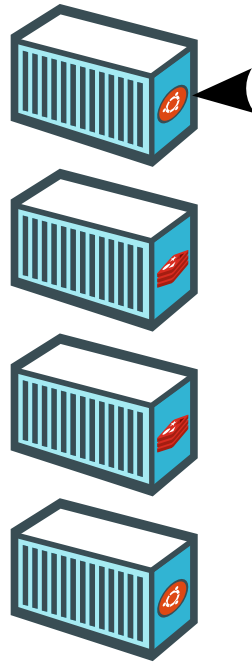
`docker pull`

`docker run`

DOCKER_HOST

Docker daemon

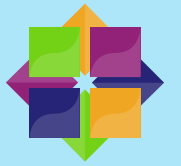
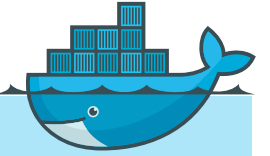
Containers



Images



Registry



Docker Terminologies

- The Docker Daemon
 - It listens for Docker API requests and manages Docker objects such as images, containers, networks, and volumes.
- The Docker Client
 - It is the primary way that many Docker users interact with Docker. The Docker client can communicate with more than one daemon.
- Docker Registries
 - It stores Docker images.
 - Docker Hub is a public registry that anyone can use.
 - Amazon ECR is another example

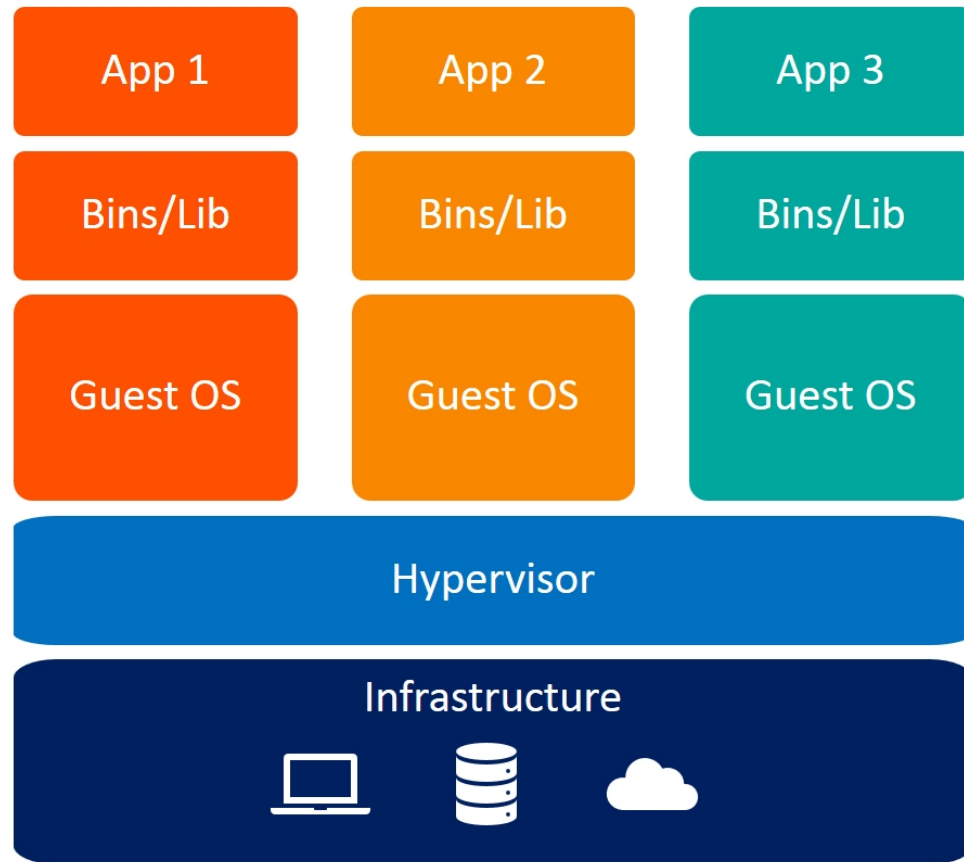
Docker Terminologies

- Docker File
 - A Dockerfile is a text document that contains all the commands a user could call on the command line to assemble an image.
- Docker Images
 - An image is a read-only template with instructions for creating a Docker container.
 - An image is based on another image, with some additional customization.
- Docker Container
 - A container is a runnable instance of an image.
 - You can create, start, stop, move, or delete a container using the Docker API or CLI.

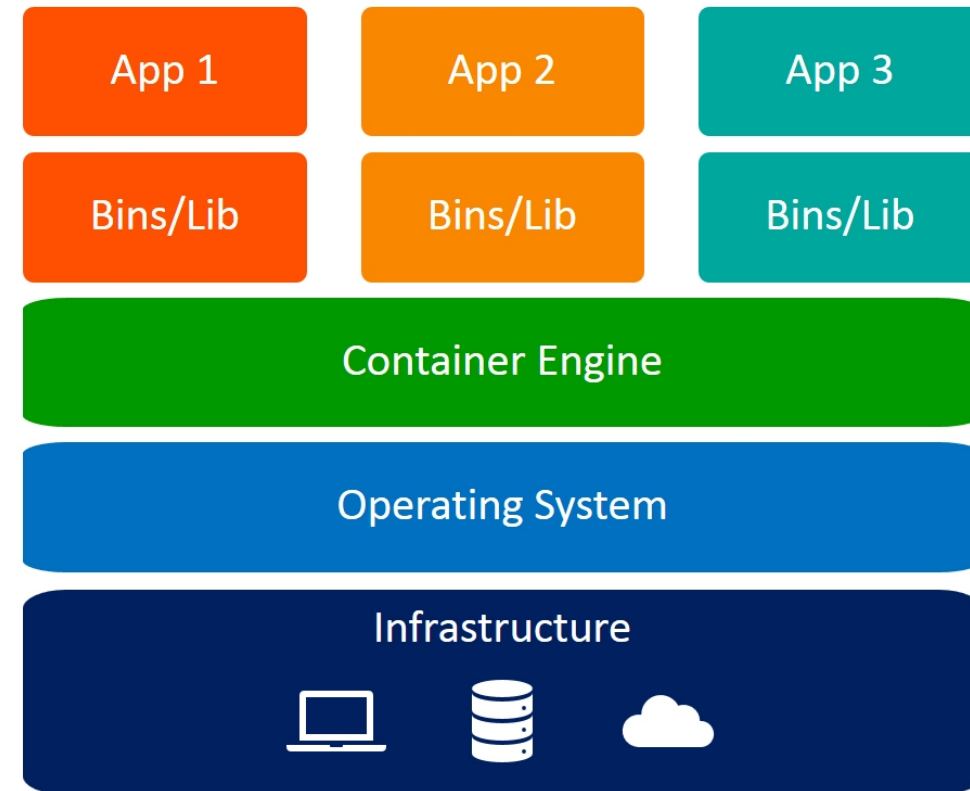
Docker Image Repository

- Docker images are stored in Docker Repositories
- Public Repo: Docker Hub <https://hub.docker.com/>
 - Find base images for many technologies or OS:
 - Ubuntu
 - MySQL
 - NodeJS, Java...
- Private Repo:
 - Amazon ECR (Elastic Container Registry)
 - JFrog
 - Nexus

Docker versus Virtual Machines



Virtual Machines



Containers

Virtual Machines Vs Docker Containers

Virtual Machines

- Need more resources
- Process isolation is done at hardware level
- Separate Operating System for each VM
- VMs can be customized
- Takes time to create Virtual Machine
- Booting takes minutes Booting is done within seconds

Docker Containers

- Less resources are used
- Process Isolation is done at Operating System level
- Operating System resources can be shared within Docker
- Custom container setup is easy
- Creation of docker is very quick
- Booting is done within seconds

Why Docker?



Speed



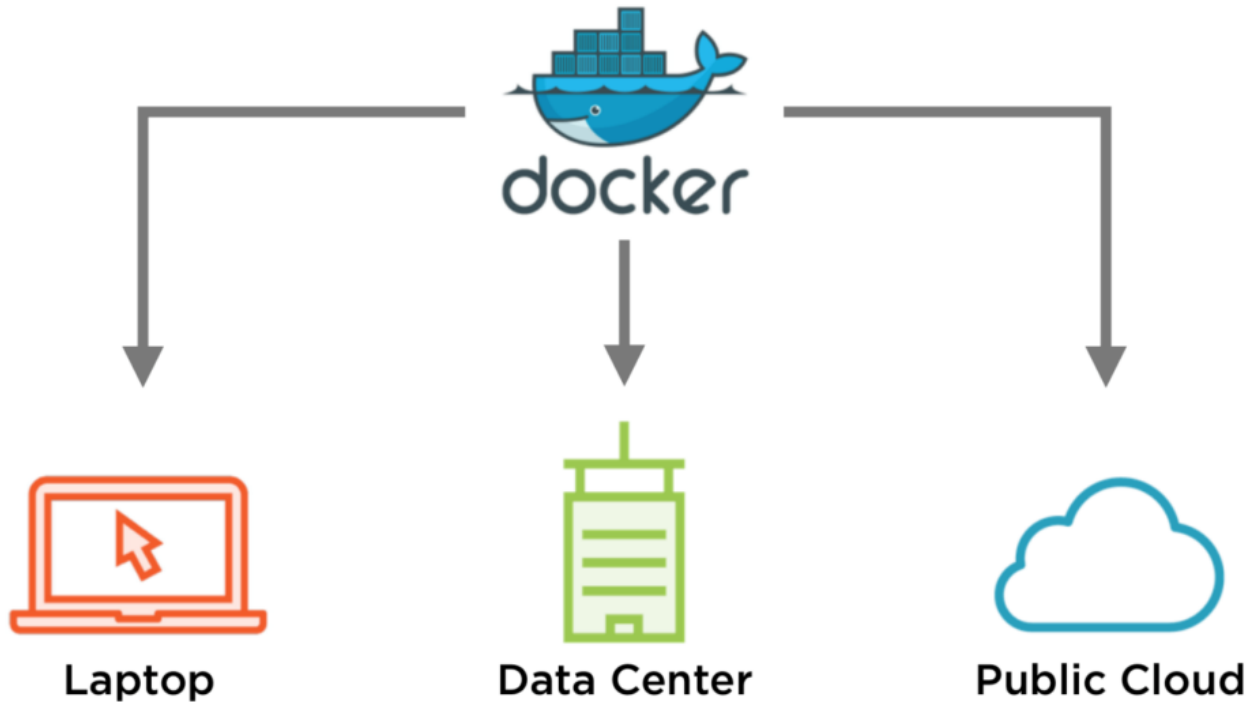
Portability



Automation

- Docker containers are very easy to deploy in any cloud platform.
- It can get more applications running on the same hardware when compared to other technologies
- It makes it easy for developers to quickly create, ready-to-run containerized applications and it makes managing and deploying applications much easier.
- There is no limitation on running Docker as the underlying infrastructure can be your laptop or else your Organization's Public / Private cloud space

Docker Anywhere...



- The purpose of Docker is to help developers and dev-ops team in becoming more productive and less error prone.
- Setup and deployment of new projects becomes much more easier and time efficient with the help of Docker.