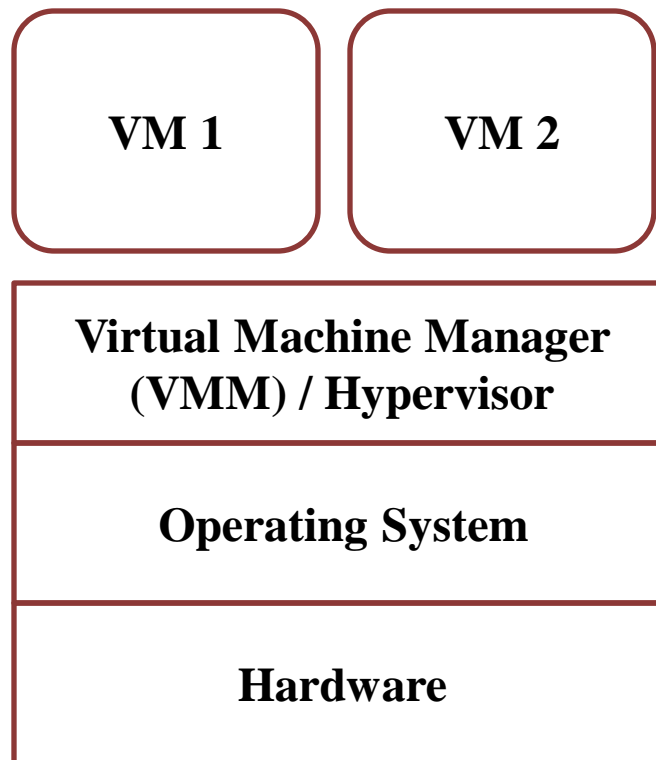


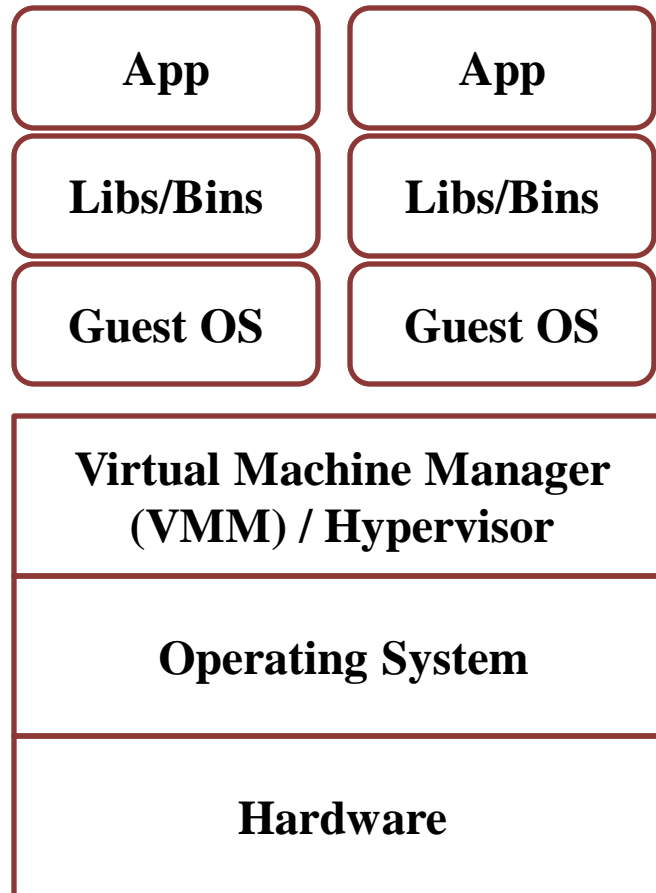
# Containers

Dr. Amit Praseed

# Review of Virtual Machines



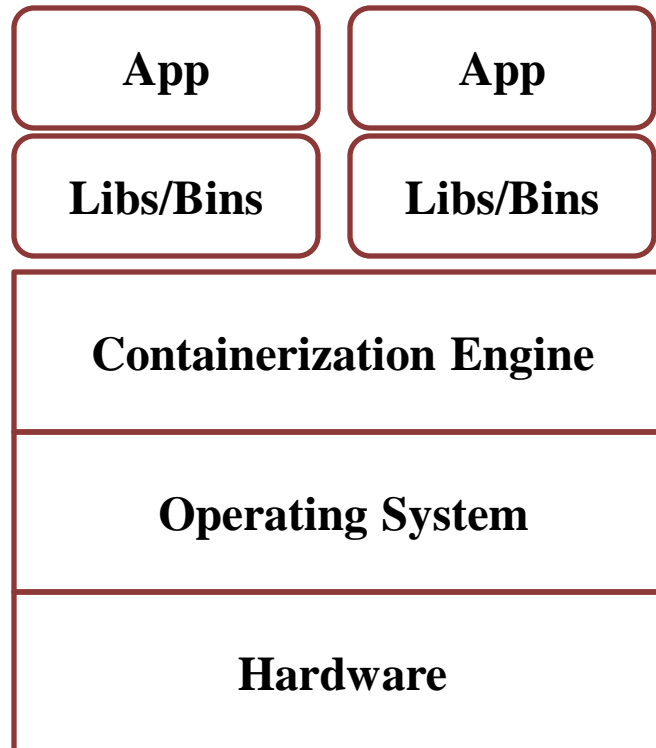
# Review of Virtual Machines



# What is a Container?

- Containerization involves
  - bundling an application together with all of its related configuration files, libraries and dependencies
  - run in an efficient and bug-free way across different computing environments
- Solves a major problem in software development
  - “It works on my system”
  - Containers bundle all related files and libraries with the application, so it can run anywhere

# How are Containers Organized?



# Containerization in One Image



# Containers vs VMs

## Containerization

- Isolates and encapsulates a single application
- Lightweight
- Migration is very simple
- OS Level Virtualization
- *Technically* can only run on same / similar underlying OS as the container

## Virtual Machines

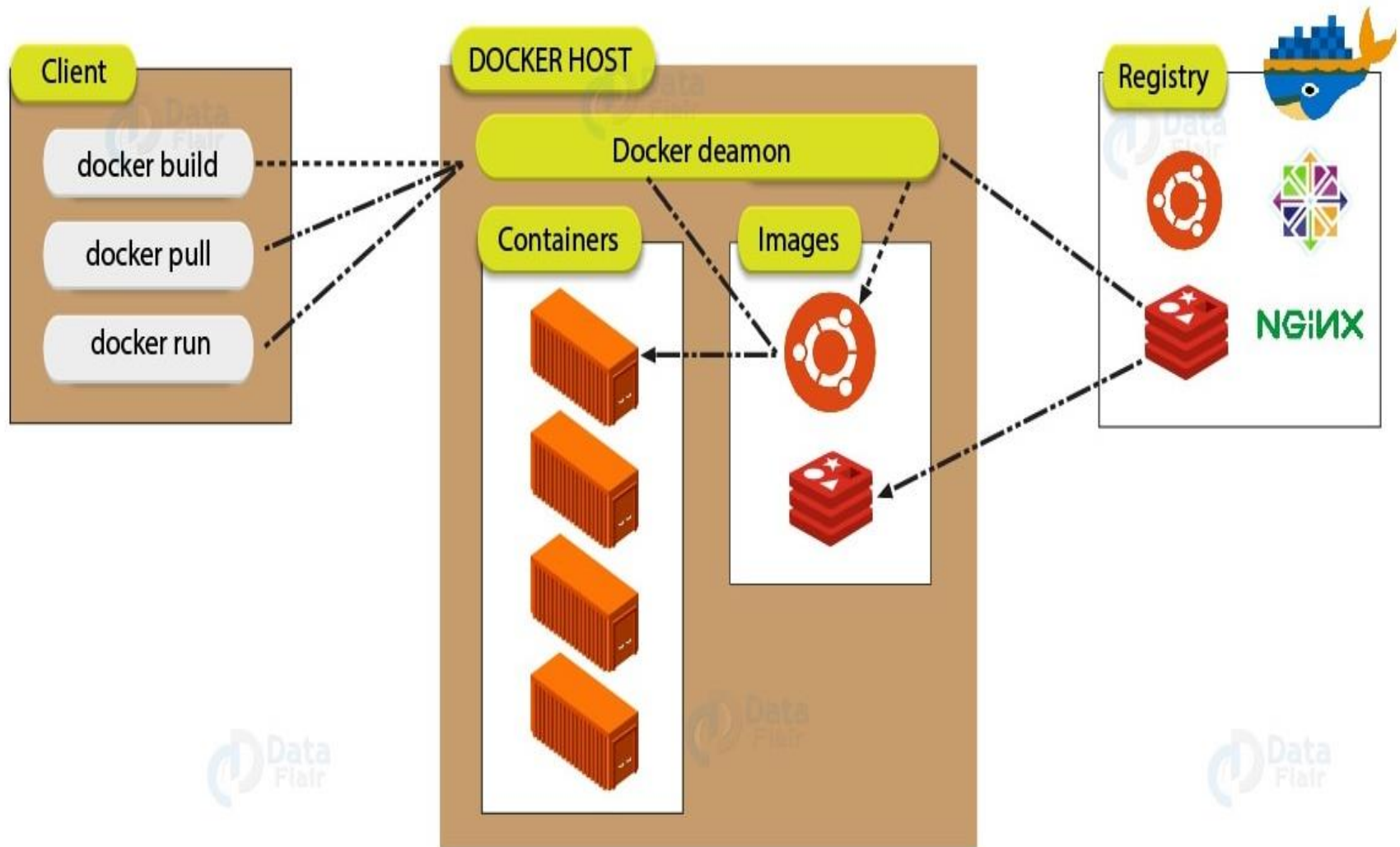
- Isolates and encapsulates an entire OS
- Heavyweight
- Migration is comparatively difficult
- Hardware level virtualization
- Can be run on any underlying OS with proper hypervisor

# Docker

- Docker is an open-source engine that automates the deployment of applications into containers
  - adds an application deployment engine on top of a virtualized container execution environment.
  - provides a lightweight and fast environment
- Features of Docker
  - An easy and lightweight way to model reality
  - A logical segregation of duties
  - Fast, efficient development life cycle
  - Encourages service orientated architecture



# Docker Architecture



# Basics of Docker

- Docker Images : Images are the “source code” of containers. It encompasses a particular application environment
- Registry : Docker images can be published to public or private registries, so they can be easily reused
- Docker Containers : A container is a live instance of a Docker image