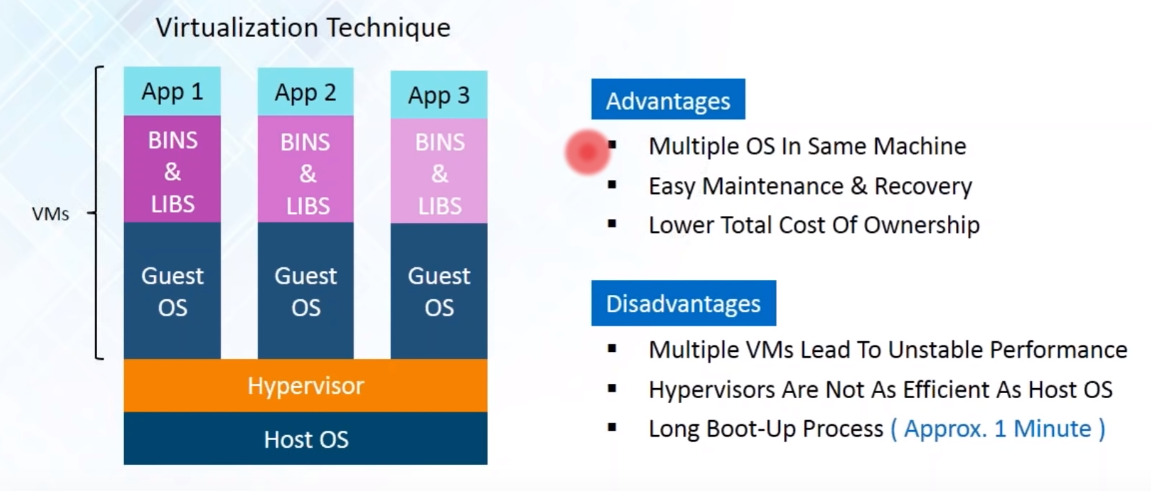
Virtualizations Vs Containerization

**Virtualization**:

Abstraction on the hardware.

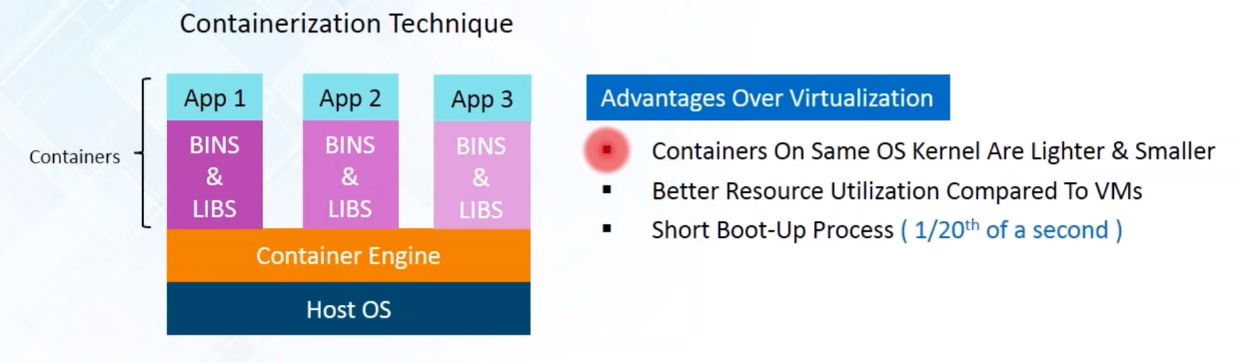
Has its own kernel and own set of libraries.

Scaling is hard.



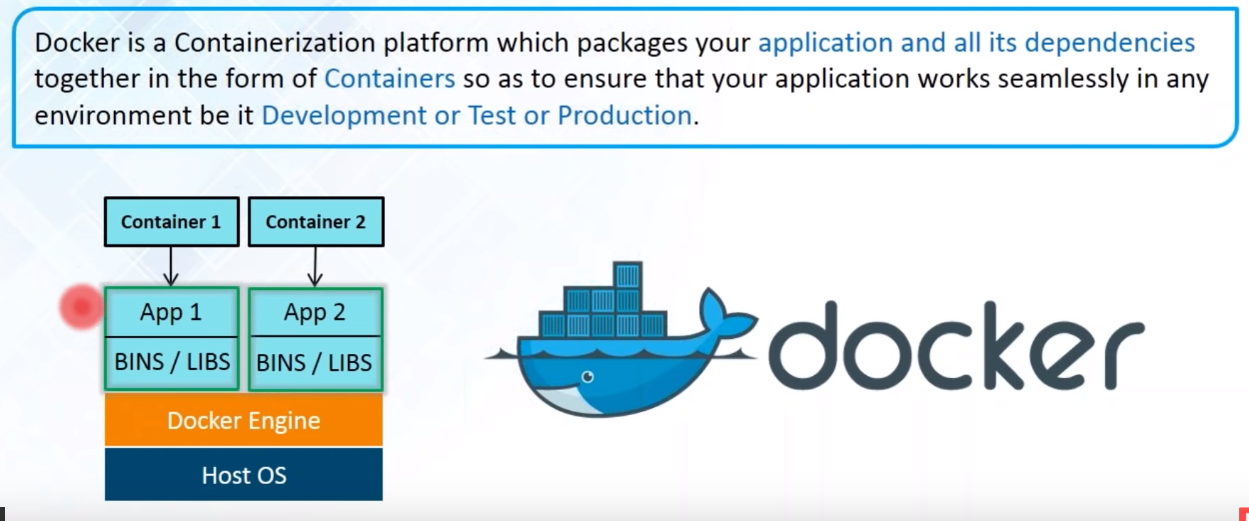
**Containerization:**

Kernel and Libraries are handled by host OS



Can have different environment or libraries for different applications.

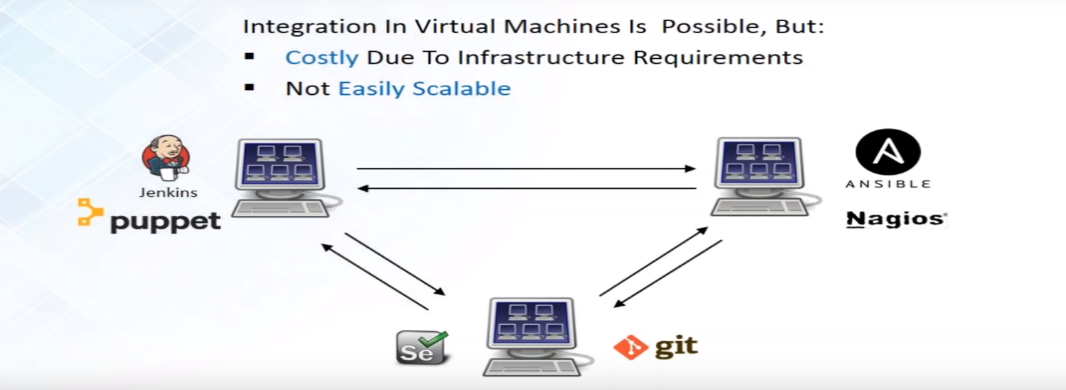
What is Docker?



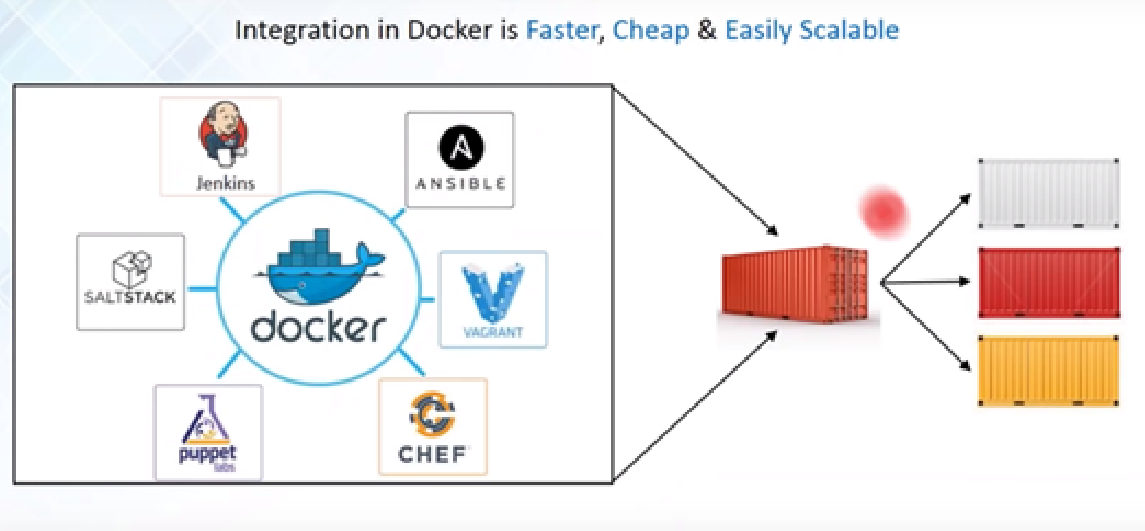
Number of systems can be scaled.

Ram allocated to VM cannot be taken back. Remaining cannot be allocated to additional VMs

In virtualization



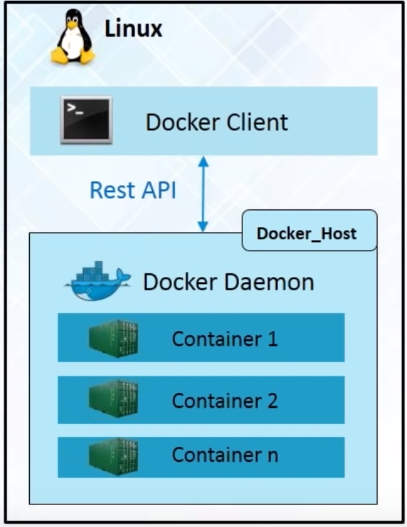
In Docker



Docker plays important role in Testing and Deployment phase.

Docker connection with a Daemon (computer program that runs as a background process)

Combination of REST API, Socket IO and TCP.



**Docker Images**:

Read Only Template used to create containers.

**Docker Registry**:

Is a storage component for Docker images.

Daemon stores all docker images in the docker registry. It can be local or In cloud.

**Docker Hub**:

Similar to github public cloud repository.

**Basic Docker Commands**

**To pull image from Docker Hub**

docker pull <image name>

**To run Docker Image**

docker run <image name>

**To list down all image in the system**

docker images

**To list down all running containers**

docker ps

**To list down all container (Running or not)**

docker ps -a

**Dockerfile**

Contains instructions to build a docker image

FROM 🡪 Base image from which the container is built.

RUN 🡪 The command that needs to be executed on the image.

**Build an image**

docker build -t <image name :tag> <dockerfile directory>

**Run an Image**

docker run –name <container name> -p <host port>:<container port> <image name>

**Inside Dockerfile**

FROM ubuntu

RUN <installation commands> <separated by &&>

COPY <path of file to be copied> <destination path>

EXPOSE <port no> port number the service has to run

**Building a container and run from docker file**

docker build -t <user specified image name>:1.0 <directory path where ur docker file is present or . >

docker run -p <host port>:<container port> --name <”application name”> <user specified image name>:1.0

**Start Container id**

docker start <container id>

docker stop <container id>

**Delete container**

docker rm <container id>

**Yaml file**

wordpress:

image:<image name from docker hub>

links:

* <dependency 1 container name>: <user specified name>

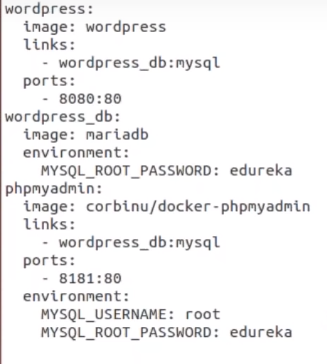
ports:

* <host port>:<container port>

<dependency 1 container name>:

Image: <image name from docker hub>

Environment:



**Running :**

Docker compose up -d

**Docker Compose**

Run different containers