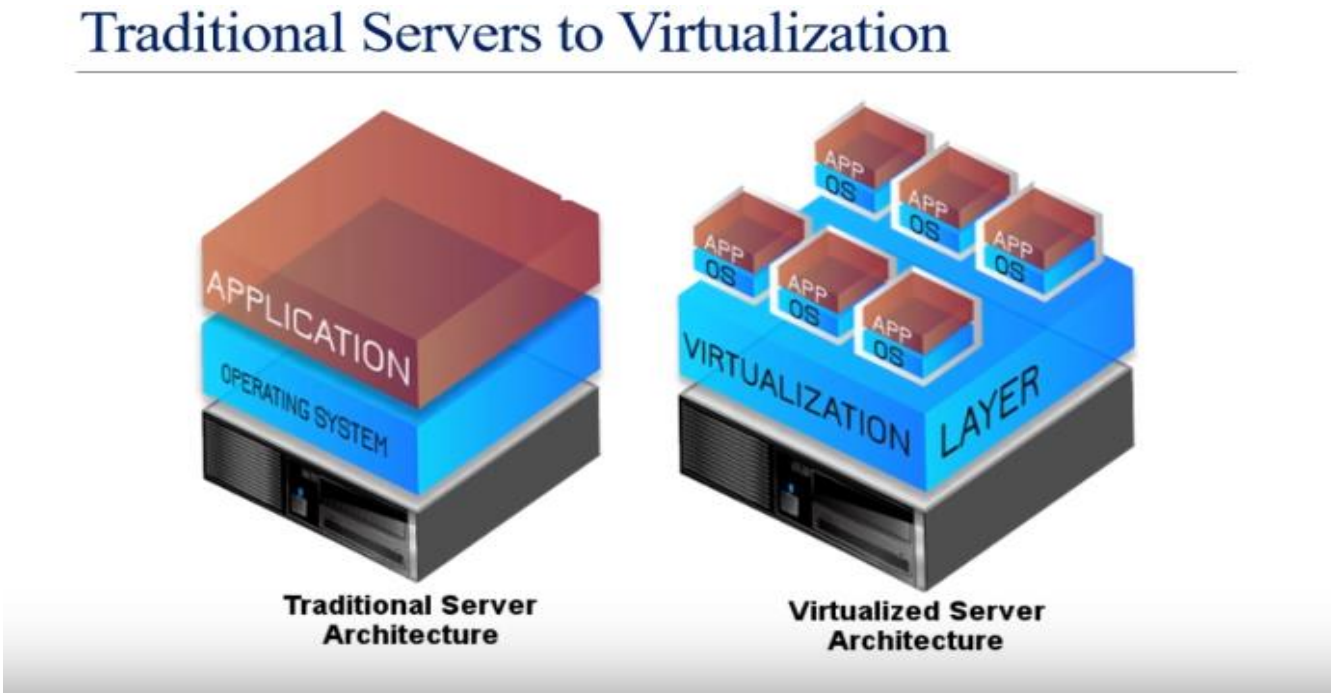
# Docker

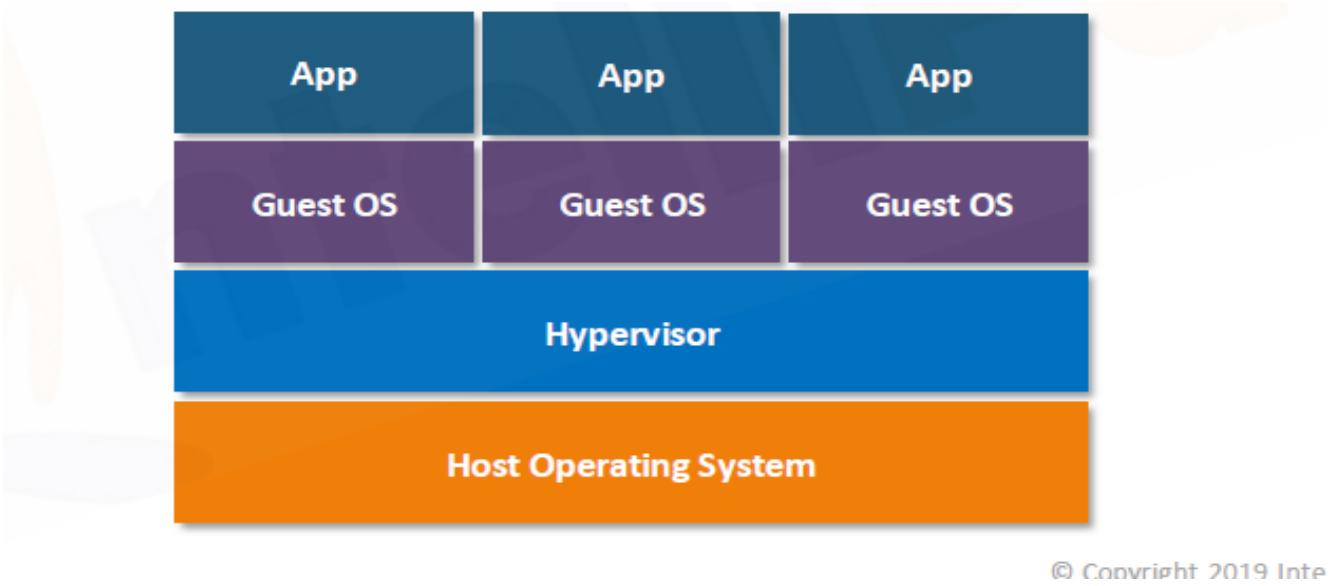
1. What is Virtualization
2. What is Containerization
3. Containerzation Tools
4. Components of Docker
5. What is docker



## What is Virtulization ?

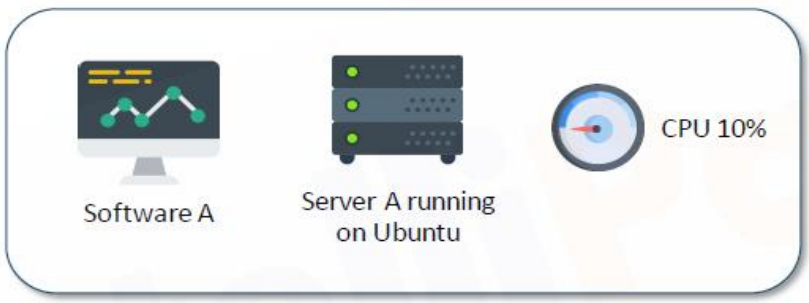
**Virtualization is the process of running multiple virtual system or resources on top of a single physical machine.**

**These resources could be a sorage device , network or even an operating system .**





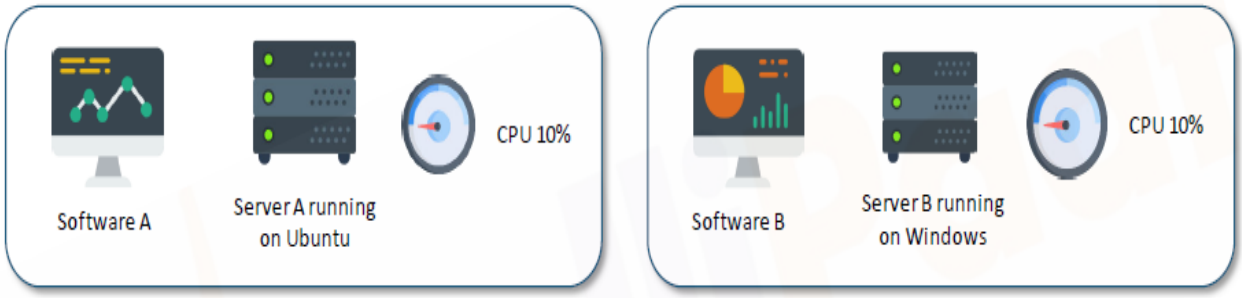

## Problems before Virtualization




**Imagine Software A running on Server A which has Ubuntu running on it .**

**This software can only run in the Ubuntu environment.**

## Problems before Virtuallization




**Some time later, we needed software B which can only run on Windows.**

**Therefore, we had to buy and run a Server B which had Windows running it. The software took only 10 % of the CPU resources.**

## Problems  before Virtualization

- Buying servers was expensive.
- Resources were not being utilized at their full potential.
- The process of getting any software up and running was time consuming.
- Disaster recovery was difficult.

## After Virualization


**Windows and Ubuntu OS now are running on the same server in parallel using the Virtualization technology.**

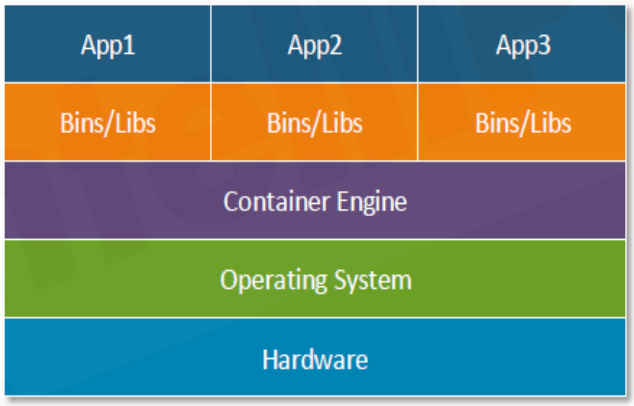
**This accounts for better CPU utilization and cost saving!**

## Advantages of Virtualization

- It results in reduced spending.
- Resources are utilized more efficiently.
- Process of getting software up and running is shorter.
- Easier backup and disaster recovery is available.

## What is Containerization?

**Application containerization is an OS-level virtualization method used to deploy and run distributed applications without launching an entire virtual machine (VM) for each app.**

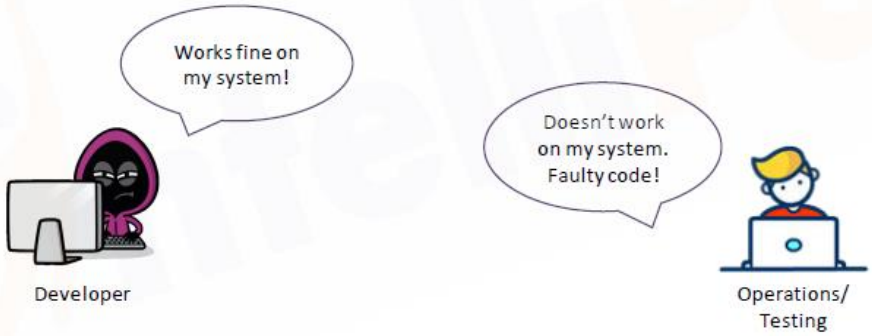





## Problems before Containerization

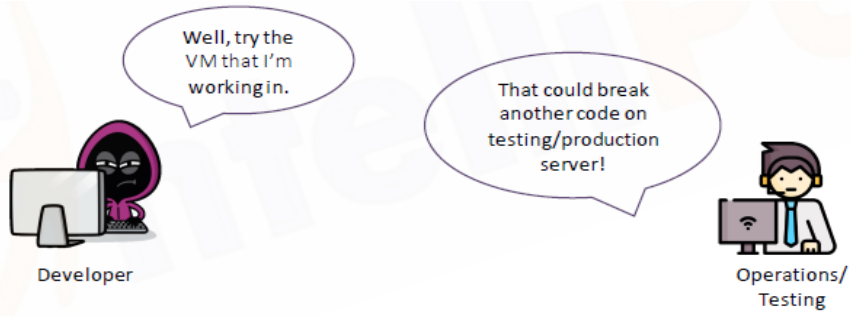
**Developers when run the code on their system, it would run perfectly.**

**But the same code would not run on the operation's system.**




**The problem was with the environment the code was being run in .**

**Well, a simple answer could be, why not give the same VM to the operations/testing team to run the code.**



## Advantanges of Containers

- Containers are not resource hungry.
- They are lightweight and hence portable.
- They are developer friendly and can be configured through the code.

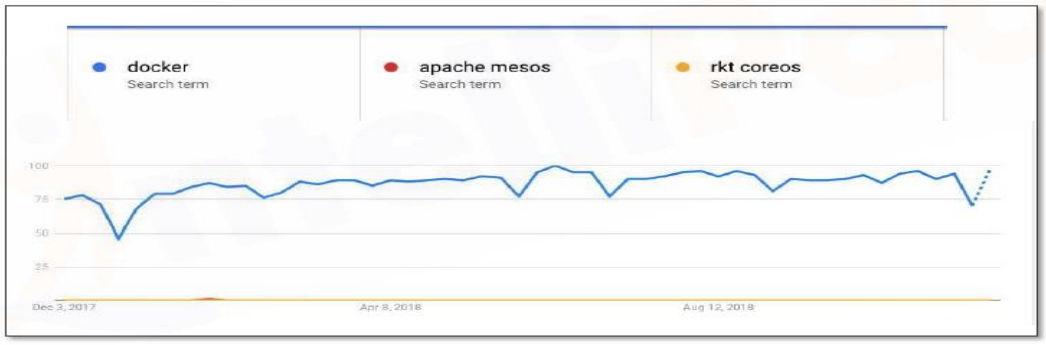
## Containerization Tools

docker

MESOS

rkt

**Docker is clearly the most famous among them all !**

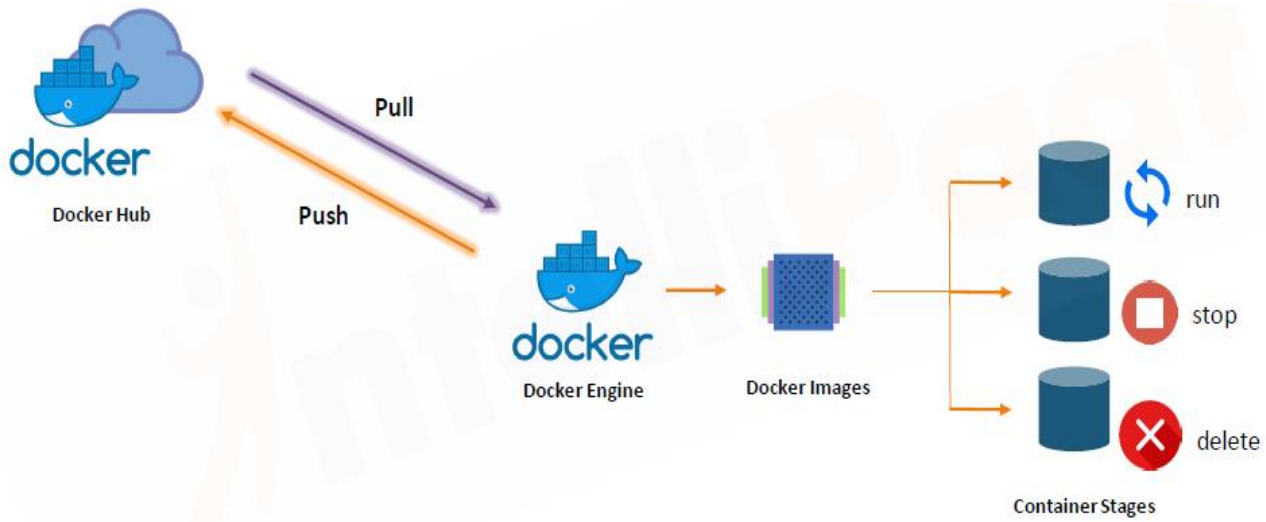


## What is Docker

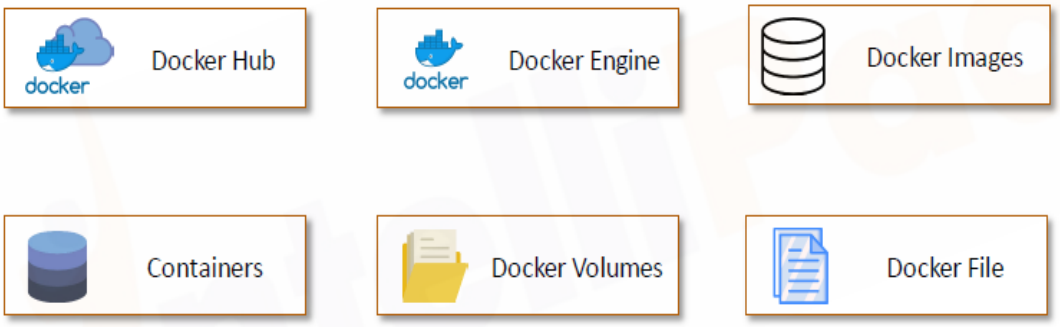
**Docker is a computer program that performs operating-system-level virtualization, also known as "containerization" . It was first released in 2013 and is developed by Docker, Inc.**

**Docker is used to run software packages called "containers" .**

## Docker Container Life Cycle




## Compenets of Docker Ecosystem



## Docker Hub

- Docker Hub is a central public docker registry.
- It can store custom docker images.
- The service is free, but your images would be public.
- It requires username/password.

## Dockeer Engine

- Docker Engine is the heart of the docker ecosystem.
- It is responsible for managing your container runtimes.
- It works on top of operating system level.
- It utilizes the kernel of the underlying OS.

## Docker Images

- Docker Image is like the template of a container.
- It is created in layers.
- Any new changes in the image results in creating a new layer.
- One can launch multiple containers from a single docker image.

## Containers

- Container size is in the order of MBs whereas VM size is in GBs. Hence containers starts almost immediately.

- A Docker Container is a lightweight software environment.
- It works on top of the underlying OS kernel.
- It is small in size and therefore is highly portable.
- It is created using the docker image.



## Docker Volumes

- Docker Containers cannot persist data.
- To persist data in containers, we can use Docker Volume.
- A Docker Volume can connect to multiple containers simultaneously.
- If not created explicitly, a volume is automatically created when we create a container.

## Dockerfile

- Dockerfile is a YAML file, which is used to create custom containers
- It can include commands that have to be run on the command line
- This Dockerfile can be used to build custom container images