Docker is a container management service.

The keywords of Docker are **develop, ship** and **run** anywhere.

The whole idea of Docker is for developers to easily develop applications, ship them into containers which can then be deployed anywhere.

The initial release of Docker was in March 2013 and since then, it has become the buzzword for modern world development, especially in the face of Agile-based projects.

## Challenges in Software Shipping

How to ship software?

How to ensure versioning between development test and production Test?

Works in one system and does not work on the other system.

Shipping of multiple comodities across multiple locations

All the comodities are put into a container.

All codes can be packed into containers using Docker container and can be shipped into any environment and executed.

## Containers

It is used to build , ship and run the application anywhere.

They are isolated, but share the O/ S and appropriate bins/ libraries

It does not have its own operating system.

Image

It is a read only template

is immutable once it has been created.

Usually derives from the multiple base images that are layers stack on the top of each other.

## Features of Docker

Docker has the ability to reduce the size of development by providing a smaller footprint of the operating system via containers.

With containers, it becomes easier for teams across different units, such as development, QA and Operations to work seamlessly across applications.

We can deploy Docker containers anywhere, on any physical and virtual machines and even on the cloud.

Since Docker containers are pretty lightweight, they are very easily scalable.

## Components of Docker

Docker Engine − It is used for building Docker images and creating Docker containers.

Docker Hub − This is the registry which is used to host various Docker images.

Docker Compose − This is used to define applications using multiple Docker containers.

## Docker Version

To see the version of Docker running, you can issue the following command −

sudo docker version

## Docker Info

To see more information on the Docker running on the system, you can issue the following command −

### Syntax

docker info

Docker Hub is a registry service on the cloud that allows you to download Docker images that are built by other communities. You can also upload your own Docker built images to Docker hub. In this chapter, we will see how to download and the use the Jenkins Docker image from Docker hub.

The official site for Docker hub is − [https://www.docker.com/](https://www.docker.com/community-edition" \l "/add_ons)

## Listing of Containers

One can list all of the containers on the machine via the **docker ps** command. This command is used to return the currently running containers.

Sudo docker ps -a

To stop the dockers

sudo docker stop <id>

To kill the process

sudo docker system prune -y

This command is used to remove Docker images.

docker rmi ImageID

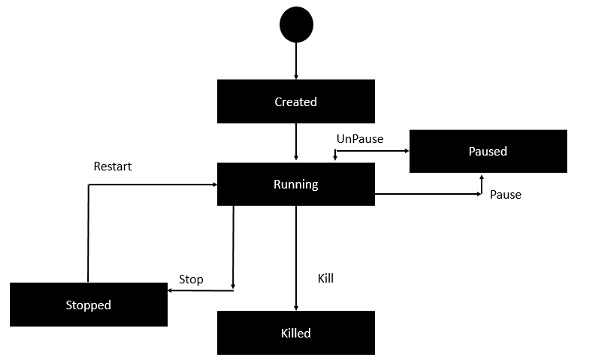
Running of containers is managed with the Docker **run** command. To run a container in an interactive mode, first launch the Docker container.

sudo docker run –it centos /bin/bash

This command is used to stop a running container.

docker stop ContainerID

## Docker – Container Lifecycle



Initially, the Docker container will be in the created state.

Then the Docker container goes into the running state when the Docker run command is used.

The Docker kill command is used to kill an existing Docker container.

The Docker pause command is used to pause an existing Docker container.

The Docker stop command is used to pause an existing Docker container.

The Docker run command is used to put a container back from a stopped state to a running state.

The following image shows the standard and traditional architecture of **virtualization**.

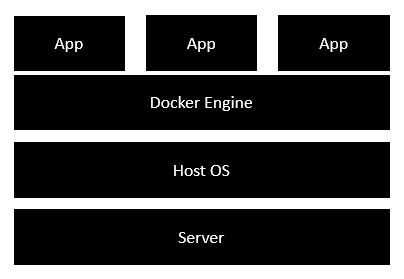
The server is the physical server that is used to host multiple virtual machines.

The Host OS is the base machine such as Linux or Windows.

The Hypervisor is either VMWare or Windows Hyper V that is used to host virtual machines.

You would then install multiple operating systems as virtual machines on top of the existing hypervisor as Guest OS.

You would then host your applications on top of each Guest OS.



The server is the physical server that is used to host multiple virtual machines. So this layer remains the same.

The Host OS is the base machine such as Linux or Windows. So this layer remains the same.

Now comes the new generation which is the Docker engine. This is used to run the operating system which earlier used to be virtual machines as Docker containers.

All of the Apps now run as Docker containers.