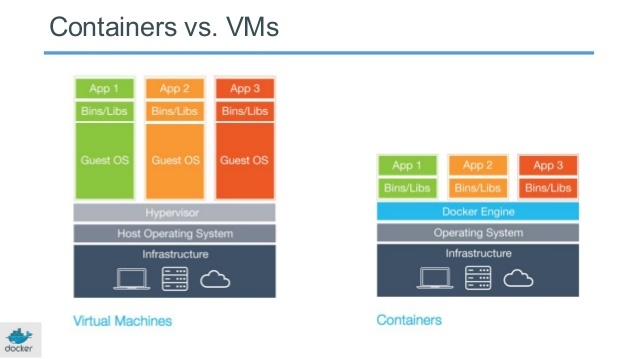
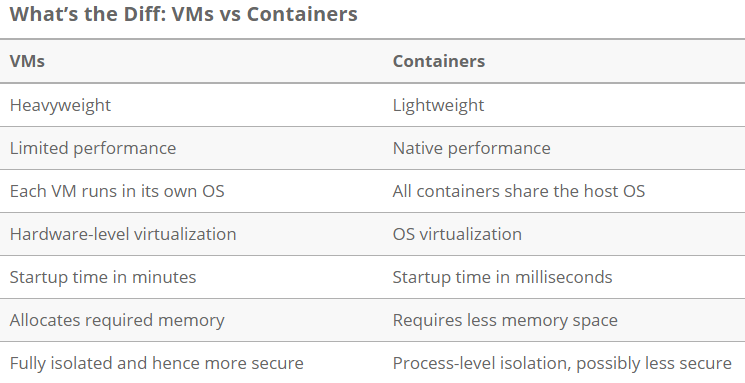
Docker : 

Before Dockers, VM’s were popular as it’s such a huge revolution in Server’s World



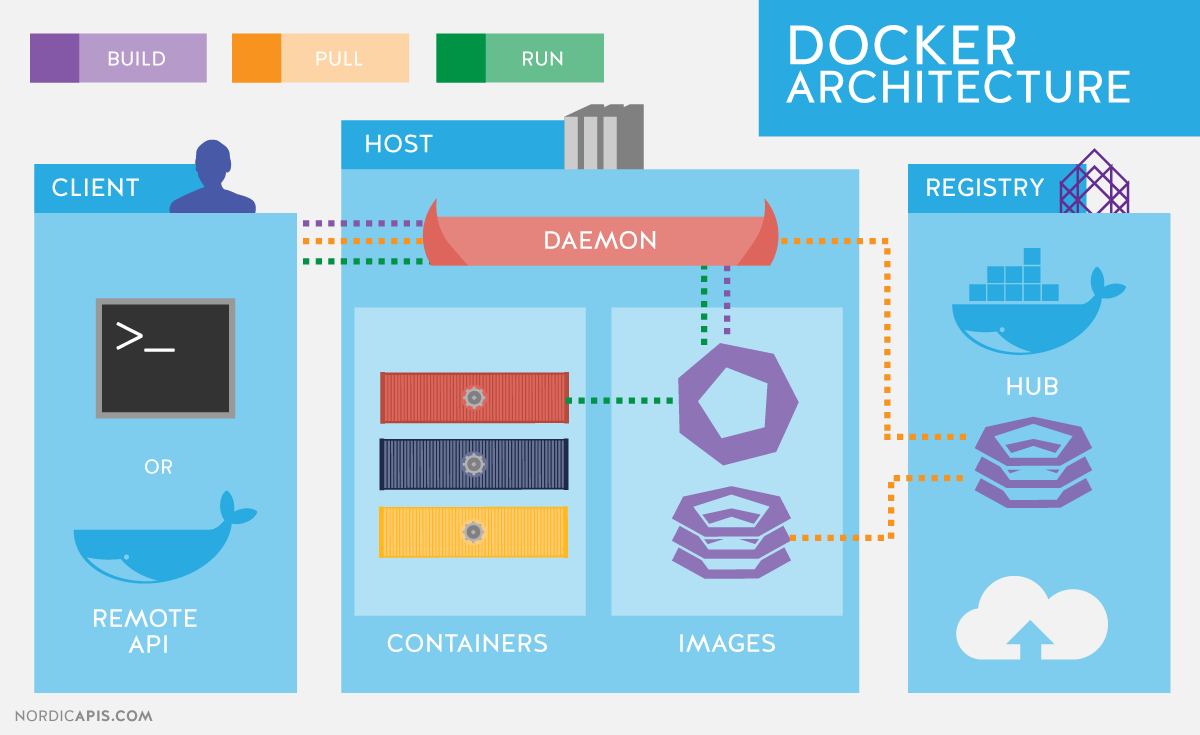
We can see few differences, which can draw a line between these 2 technologies



What is Docker?

Docker is a tool designed to make it easier to create, deploy, and run applications by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package.

Docker Architecture:



**1) Docker Client:**

The Docker client provides a command line interface (CLI) that allows you to issue build, run, and stop application commands to a Docker daemon.

**2) Docker Host (Daemon)**

The Docker host provides a complete environment to execute and run applications

the daemon is responsible for all container-related actions and receives commands via the CLI or the REST API.

It stores/manages images and containers

**3) Docker Images:**

Docker Image is a read-only template/file (Set of instructions) comprised of multiple layers which is source to create container

Docker images are the "source code" for our containers; we use them to build containers.

**4) Docker Container:**

Run time instance of the docker image called

You can create, start, stop, move, or delete a container using the Docker API or CLI

By default, a container is relatively well isolated from other containers and its host machine

**5) Docker Registry/Hub:**

A Docker registry is a Cloud Repository to store/share Docker images. The registry allows Docker users to pull images locally, as well as push new images to the registry (Web/Cloud repository)

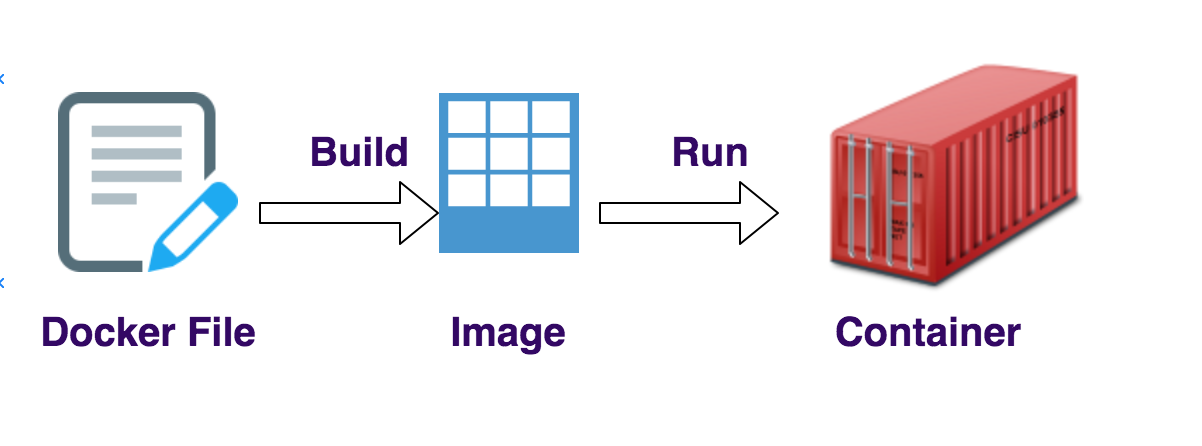
Docker stores the images we build in registries. There are public and private registries.

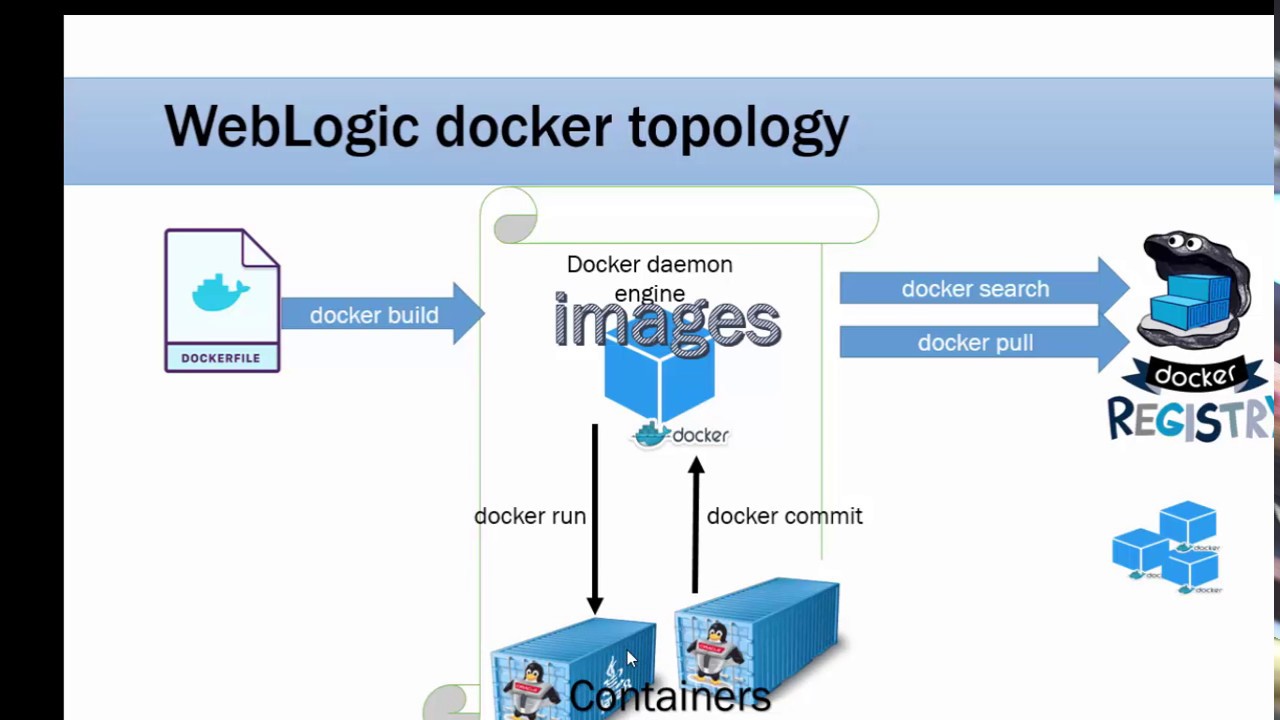
Docker company has public registry called [Docker hub](https://hub.docker.com/), where you can also store images privately. Therefore, we can use those existing images by just pulling in to local machine

6)Docker File:

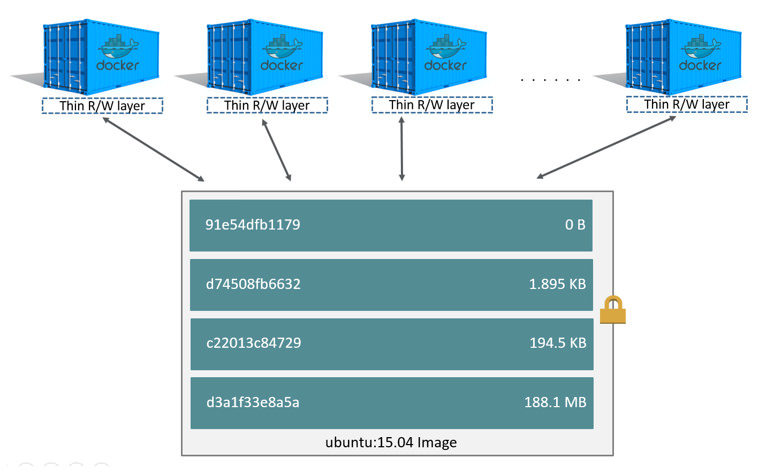
Docker file is a text document that contains all the commands a user could call on the command line to assemble an image.

**Relationship B/W Docker file, Image, Container, registry can see as below:**





Relationship b/w Image and container:



Docker images are made up of one or more read-only layers

Each layer is only set of differences from the layer before it, the layers are stacked on top of each other.

For example, when you create new container, you add up new writable layer on top of the underlying layers. The layer is often called the "container layer". All changes made to the running container such as writing new files, modifying existing files and deleting files are written to this this writable container layer

Note: difference b/w container and an image is the top writable layer. All writes to the container that add new or modify existing data are stored in this writable layer. When the container is deleted, the writable layer is also deleted. The underlying image will be unchanged. Because each container will be having its own writable layer and all the changes are stored in this container layer, Multiple containers can share access to the same underlying image, Yet have their own data state..

Docker uses storage drivers to manage the contents of the image layers and the writable container layer.