DOCKER CLASS-1

DOCKER

MONOLITHIC:

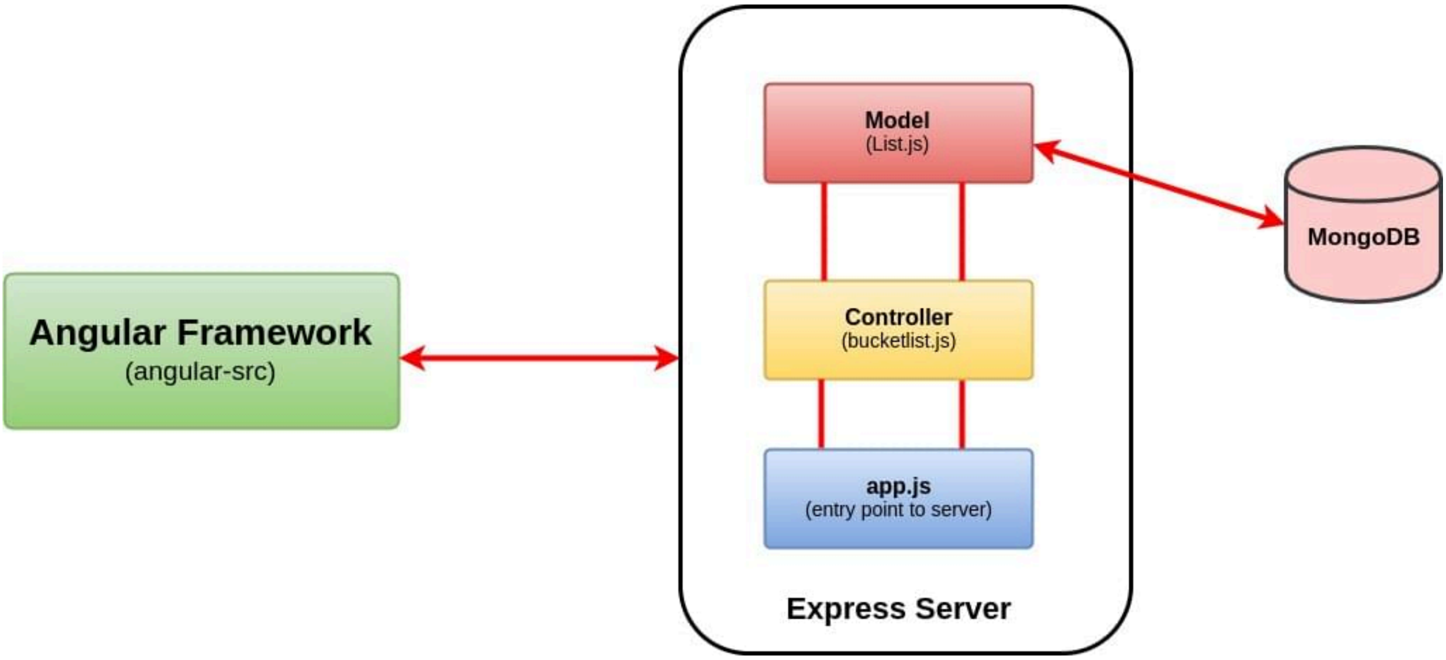
If an application contains N number of services (Let's take Paytm has Money Transactions, Movie Tickets, Train tickets, etc..) If all these services are included in one server then it will be called Monolithic Architecture. Every monolithic Architecture has only one database for all the services.

MICRO SERVICES:

If an application contains N number of services (Let's take Paytm has Money Transactions, Movie Tickets, Train tickets, etc..) if every service has its own individual servers then it is called microservices. Every microservice architecture has its own database for each service.

WHY DOCKER:

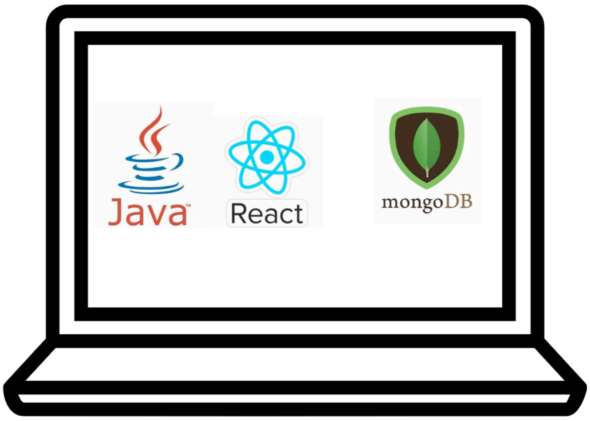
let us assume that we are developing an application, and every application has front end, backend and Database.



To develop the application we need install those dependencies to run to the code.

So i installed Java11, ReactJS and MongoDB to run the code.

After some time, i need another versions of java, react and mongo DB for my application to run the code.

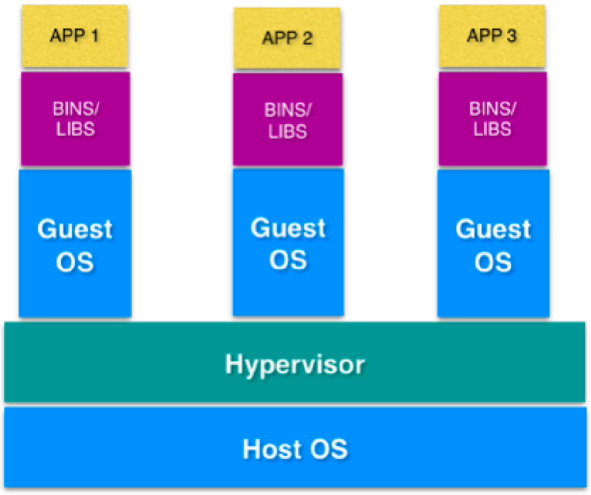


So its really a hectic situation to maintain multiple versions of same tool in our system. To overcome this problem we will use virtualization.

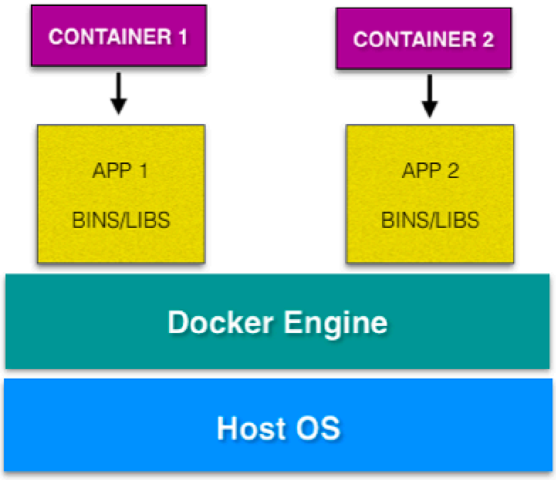
VIRTUALISATION:

It is used to create a virtual machines inside on our machine. in that virtual machines we can hots guest OS in our machine.

by using this Guest OS we can run multiple application on same machine. Hypervisor is used to create the virtualisation.



VIRTUAL MACHINE ARCHITECTURE



DOCKER ARCHITECTURE

DRAWBACKS:

- It is old method.
- If we use multiple guest OS then the performance of the system is low.

CONTAINERIZATION: It is used to pack the application along with its dependencies to run the application.

CONTAINER:

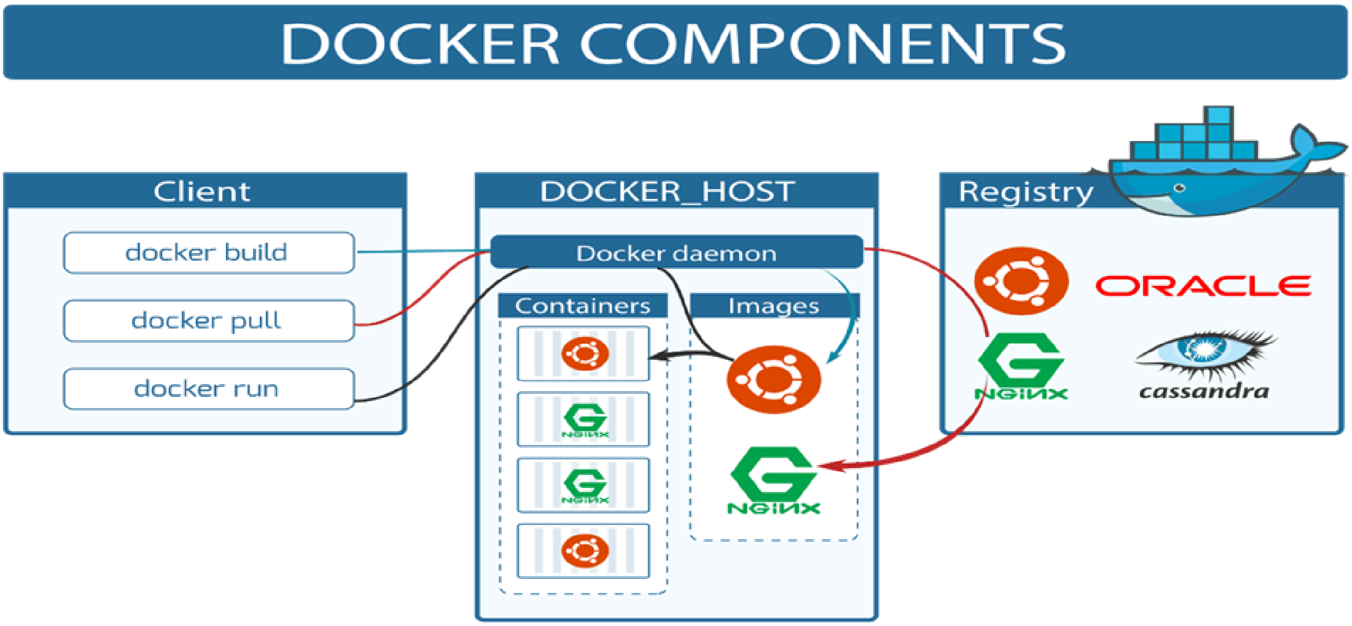
- Container is nothing but, it is a virtual machine which does not have any OS.
- Docker is used to create these containers.
- A container is like a lightweight, standalone package that contains everything needed to run a piece of software.
- It includes the application code, runtime, system libraries, and dependencies.
- To create a container we use docker.

DOCKER

- It is an open source centralized platform designed to create, deploy and run applications.
- Docker is written in the Go language.
- Docker uses containers on host O.S to run applications. It allows applications to use the same Linux kernel as a system on the host computer, rather than creating a whole virtual O.S.

- We can install Docker on any O.S but the docker engine runs natively on Linux distribution.
- Docker performs O.S level Virtualization also known as Containerization.
- Before Docker many users face problems that a particular code is running in the developer's system but not in the user system.
- It was initially released in March 2013, and developed by Solomon Hykes and Sebastian Pahl.
- Docker is a set of platform-as-a-service that use O.S level Virtualization, where as VM ware uses Hardware level Virtualization.
- Container have O.S files but its negligible in size compared to original files of that O.S.

DOCKER ARCHITECTURE:



DOCKER CLIENT:

It is the primary way that many Docker users interact with Docker. When you use commands such as docker run, the client sends these commands to docker daemon, which carries them out. The docker command uses the Docker API.

DOCKER HOST:

Docker host is the machine where you installed the docker engine.

DOCKER DAEMON:

Docker daemon runs on the host operating system. It is responsible for running containers to manage docker services. Docker daemon communicates with other daemons. It offers various Docker objects such as images, containers, networking, and storage.

DOCKER REGISTYR:

A Docker registry is a place where Docker images are stored and can be easily shared. It serves as a centralized repository for Docker images, allowing users to upload, download, and manage container images.

POINTS TO BE FOLLOWED:

- You cant use docker directly, you need to start/restart first (observe the docker version before and after restart)
- You need a base image for creating a Container.
- You cant enter directly to Container, you need to start first.
- If you run an image, By default one container will create.

DOCKER BASIC COMMANDS:

- To install docker in Linux : `yum install docker -y`
- To see the docker version : `docker --version`
- To start the docker service : `service docker start`
- To check service is start or not : `service docker status`
- To check the docker information : `docker info`
- To see all images in local machine : `docker images`
- To download image from docker hub to local : `docker pull image name`
- To create a container : `docker run -itd --name cont-name -p 8081:80 shaikmustafa/dm`
- To start container : `docker start container name`
- To see all containers : `docker ps -a`
- To see only running containers : `docker ps` (ps: process status)
- To stop the container : `docker stop container name`