

Our app/software used to run on a single server in olden times. Whenever we needed to scale our app or users load used to increase on our app we then had to increase the no. of server eventually, but this was not a pretty good idea as it would have increased the cost of buying new hardware , cost of storing the hardware and also maintaining it as well.

This problem was solved by a company named VMWare. How=> Virtual Machines

What are virtual machines and what they do?

When we talked about one application on one server. VMs solved this problem.

The problem here is Vms require their own Operating system. Even though it was far better than one app on one system, still os was a problem in this scenario.

Flaws of virtual machine

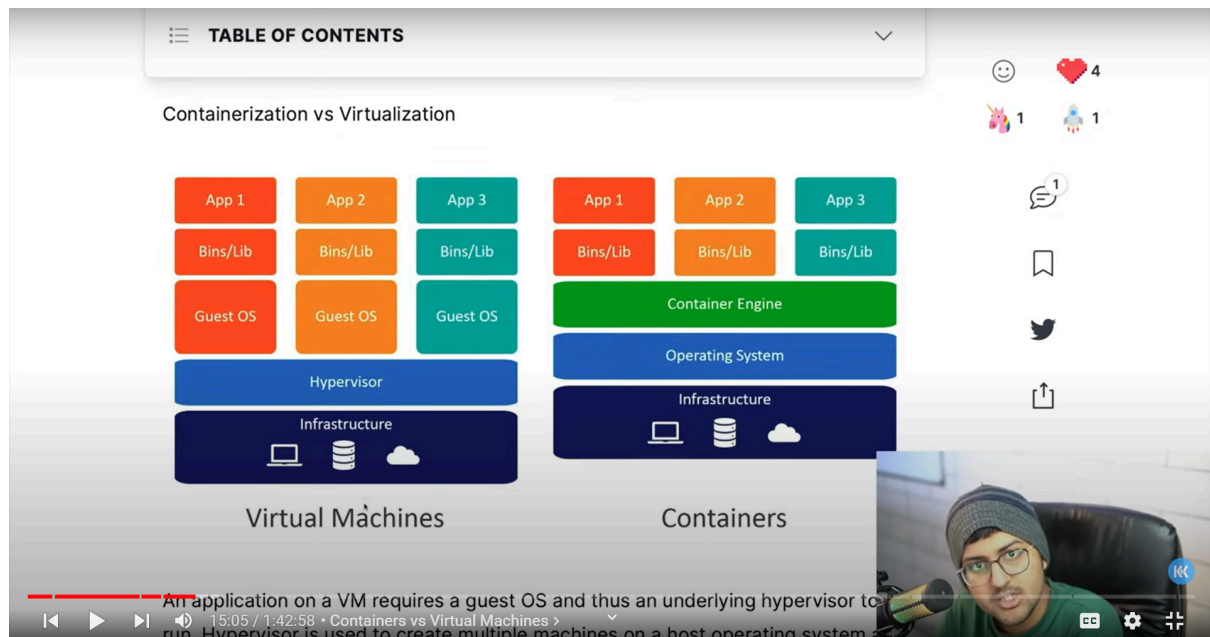
1. Having multiple projects thus operating systems => now we would have windows as well as an linux based os stored on the hard disk.
2. Slow speed of virtual machines. They were not superfast.
3. Dedicated cpu , ram for particular OS.

Dual booting - Dual booting is the process of installing and running two different operating systems (OS) on a single computer, allowing users to choose which OS to use when starting up the computer

Docker made popular containers popular for companies like google and all , they were using containers prior to that as well but docker made it more popular.

Difference between containers and VMs

Containers are like sort of what VMs are doing but we need not require additional Os for that.



Microsoft worked with docker to run docker on windows

To run docker on windows

We need to install docker desktop

Most of the containers previously that became famous were linux based.

If we have a windows app that is containerized that will not run on a linux based docker/ container and vice-versa. Thus windows container will require windows and for linux app linux.

If we are using windows we need to use WSL.

For Mac: VMs & Docker Desktop

For Linux: As it is.

Now let us learn about what is docker how it helps us

Docker helps us too run our app in isolated environments, means container will not know what is happening outside it even in the OS.

Docker allows us to do all these things.

1. Docker runtime => runtime basically allows us to start and stop containers. Runtime is of two types.
 - a) low level runtime - nc => to work with OS and starts and stops the container.
 - a. Container d => lies on higher level of hierarchy, manages the run, how to interact your container with the network, this kind of stuff is done by this.