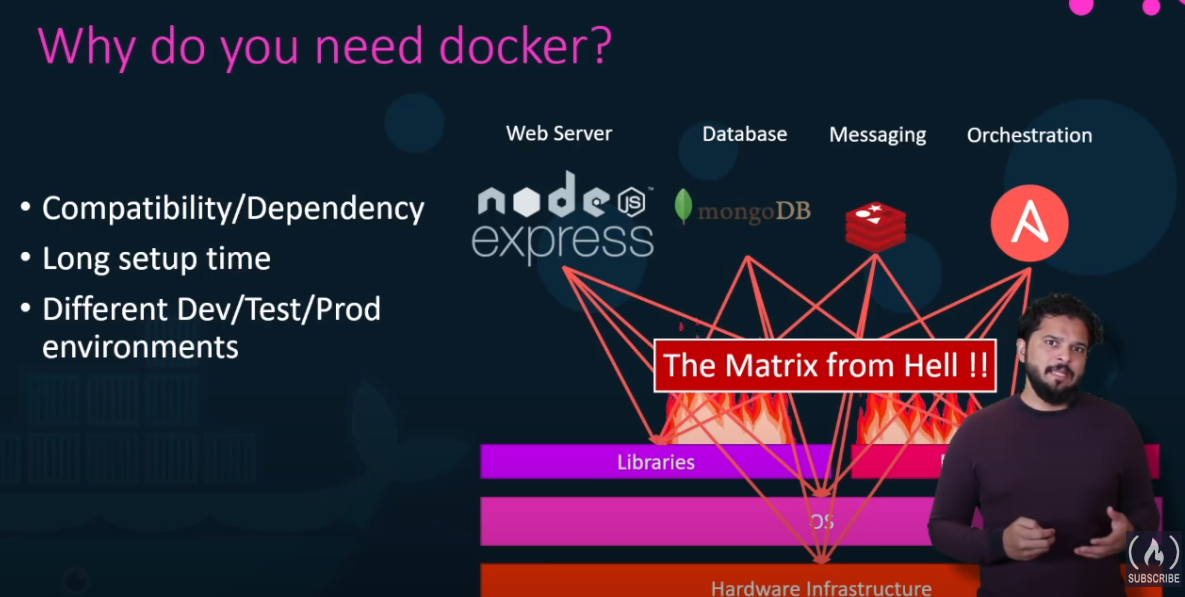
**Docker**

**Why do you need Docker**



1. Compatibility/Dependency :

For a project , we may need different components like sepcific webserver (Node Js Express) , DB (Mongo DB), messaging system(Amazon MQ), Orchestration (eg: Ansible) etc . We need to consider/setup below things according to Compatibility/Dependency

1. OS

2. Hardware / infrastructure

3. Libraries

1. Long Setup time

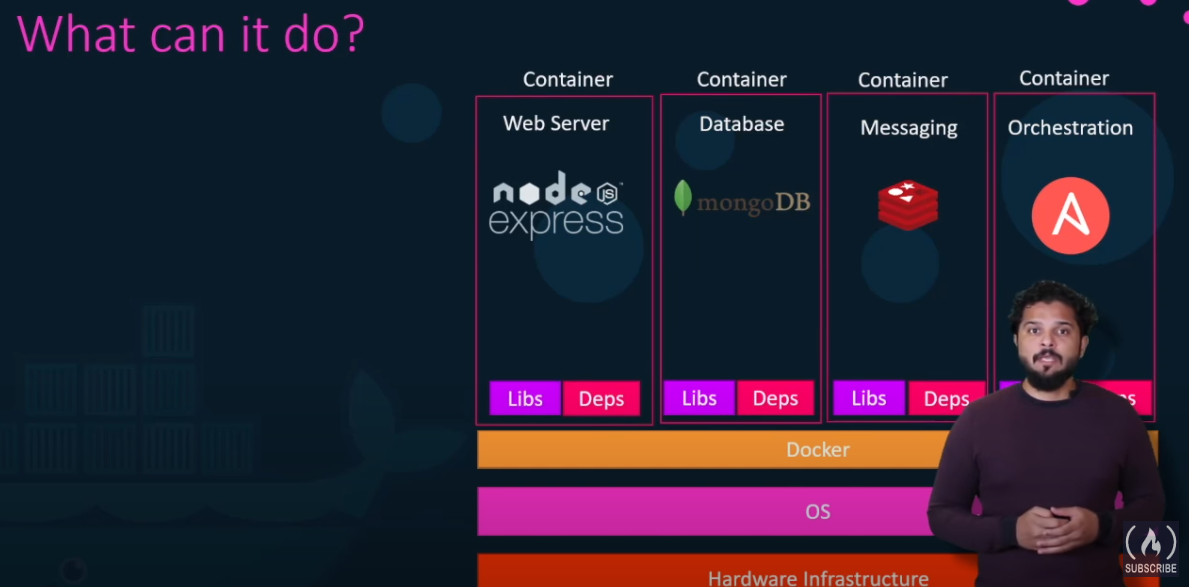
For every new developer or tester , the above set will consume much amount of time.

1. Different dev/test/prod environments

Developing , building and shipping our applications will be a complex and difficult process

By using Docker :

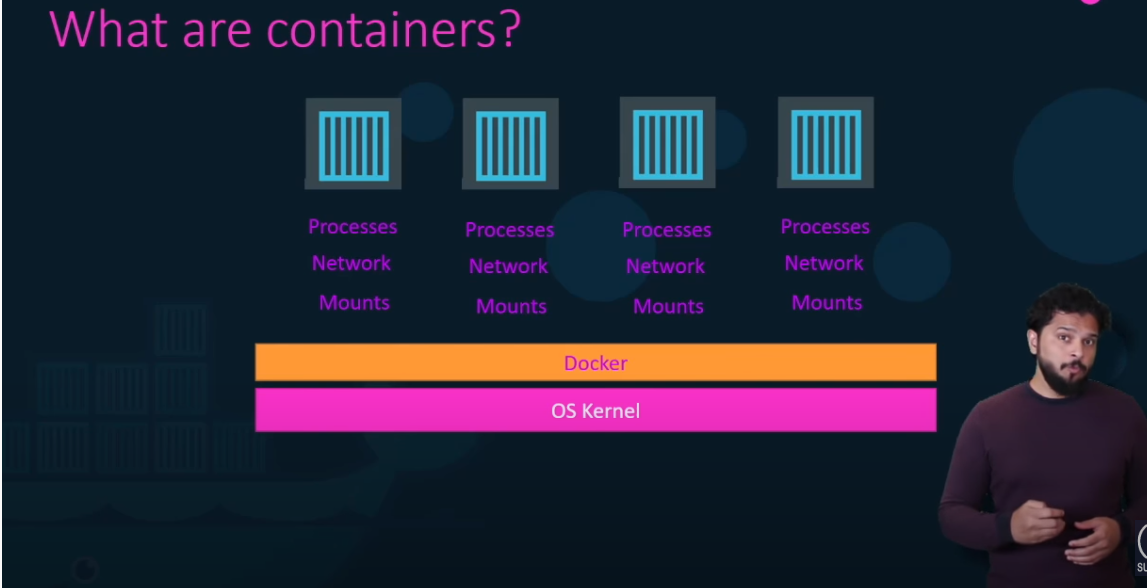
You can pack each of these components into separate containers with all there dependencies and libraries



**What are containers**

Containers are completely Isolated environments with their own processes, services, network interfaces , mounts but shares the same OS kernel. This means If a docker is installed on an Ubuntu OS , it can run docker containers with any OS of Linux kernel since host OS shares the same kernel. So it means that you can’t run a windows docker container on a Linux kernel.

There is an exception , you can run a linux docker container on Windows. But its not actually running on DOS kernel. In turn, Windows here create a linux virtual machine and then run docker container with linux over it.



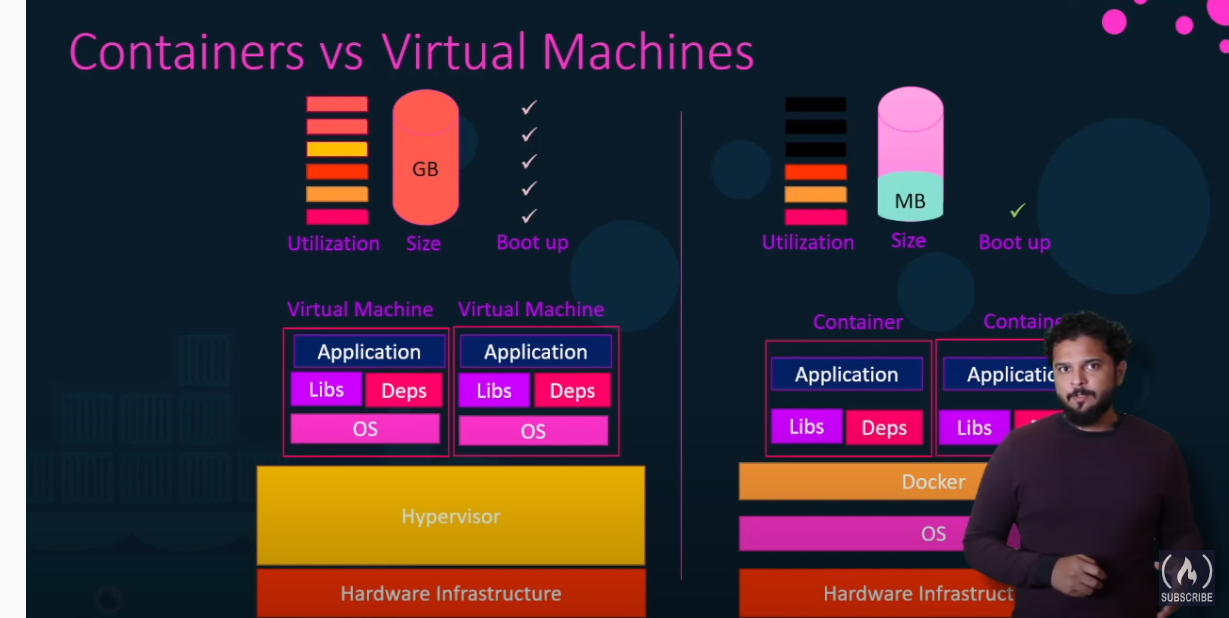
Docker is not meant for virtualizing and running different systems/OS on a same hardware. For that purpose , if you want to isolate a complete system virtual machines are the best way. They uses hypervisors like VMware for this purpose

Docker is meant for package and containerize applications and to ship them and run them any where anytime you want

**Container v/s Virtual Machines**

1. VMs are meant for complete system Isolation while Dockers are meant for Isolation of applications sharing the same OS kernel and resources
2. VMs need a Hypervisor like VMware to host the guest OS and to Isolate the guest OS completely from the host OS. On other hand, docker doesn’t need since it share the same OS kernel
3. Size of each VMs will be heavy (in GBs) since its having an entire OS while Dockers are light weight (MBs)
4. Bootup time is more for a VM since it need to boot up the OS while for a docker its very less

Most of the real time project uses a hybrid system where Containers in VMs



**Docker in Devops**

1. Usually when a developer completes development , they do the packaging of the application and handover to the devops team with an instruction file . Any issue with deployment , devops team need to communicate with developer to resolve the issue
2. On the other hand with dockers, developers and operation team works hand in hand to transform the instruction guide into a **docker file** with both of their requirement. And this docker file is used to create the **docker image** of the application . Now they can run any number of instance of this docker image in any environment they needed.