**Docker:**

Docker is a set of platform as a service products that use OS-level virtualization to deliver software in packages called containers.

Docker Installation:

https://docs.docker.com/desktop/install/debian/

Docker Engine: That manages the containers.

Docker file: Text document that contains all the command that user can call on the command line to assemble the image.

Image: Template to create the container.

Container: Running instance of the docker image. Container hold the entire package to run the application.

Docker Hub .com for the docker Images.

Docker Commands:

docker -v , --version

docker images

docker pull <Image\_name>

docker run <Image\_name>

docker search <Image\_name> = search the image

docker ps -a = show all the running containers

docker run --env MY\_SQL\_PASSWORD=test

docker run --detach, docker run -d

docker run --name <Container\_Name> -d <Image\_name> = create Container and run the given image

For open the image in interactive mode for use in the local

docker run -it -d <Image\_name>

docker exec -it <Container\_ID>

docker inspect <Container\_ID>

docker stop <Container\_ID>

docker rm <Container\_ID>

==> Remove the Images

docker rmi <Image\_Id>

==> Delete all the images:

docker images -a

docker rmi $(docker images -a -q)

==============================================================================

Commands For Image creation in Docker

------------------------------------------------------------------------------

docker login

docker commit

docker push

docker copy

docker logs

docker volume

docker logout

docker build -t <Image\_name> <DockerFilePath>

========================================================================

**Kubernetes:**

Kubernetes open source container orchestration tool

Develop by google

Helps to manage containerized application in different environments (local machine, Colud machine, Virtual machine).

**What problem Kubernetes solve and what are the tasks.**

As we have microservices it increases the containers. Suppose Each container have one microservice. For Managing the service containers we require a Kubernetes.

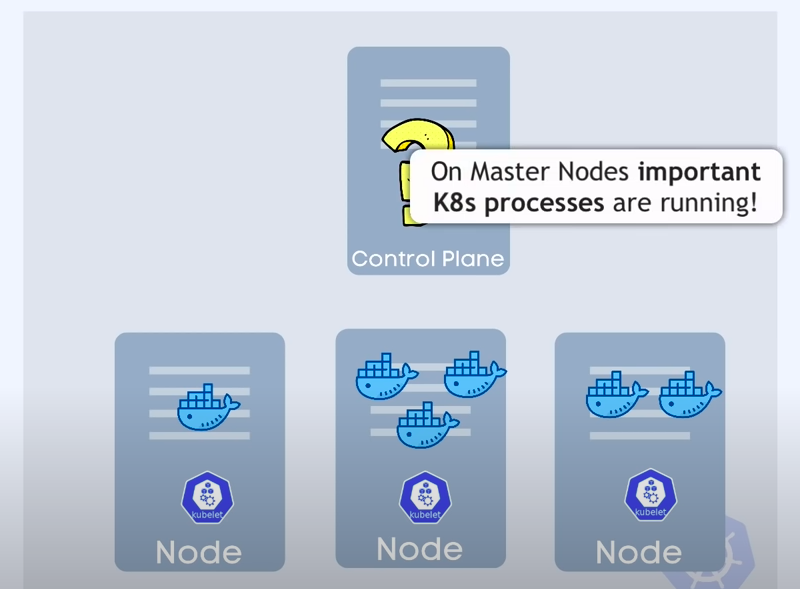
What orchestration tool (like Kubernetes) offeeers [Features of the Kubernetes]:

High Availability no downtime.

Scalability or high performance.

Disaster Recovery: Backup and restore.

Kubernates Architecture:

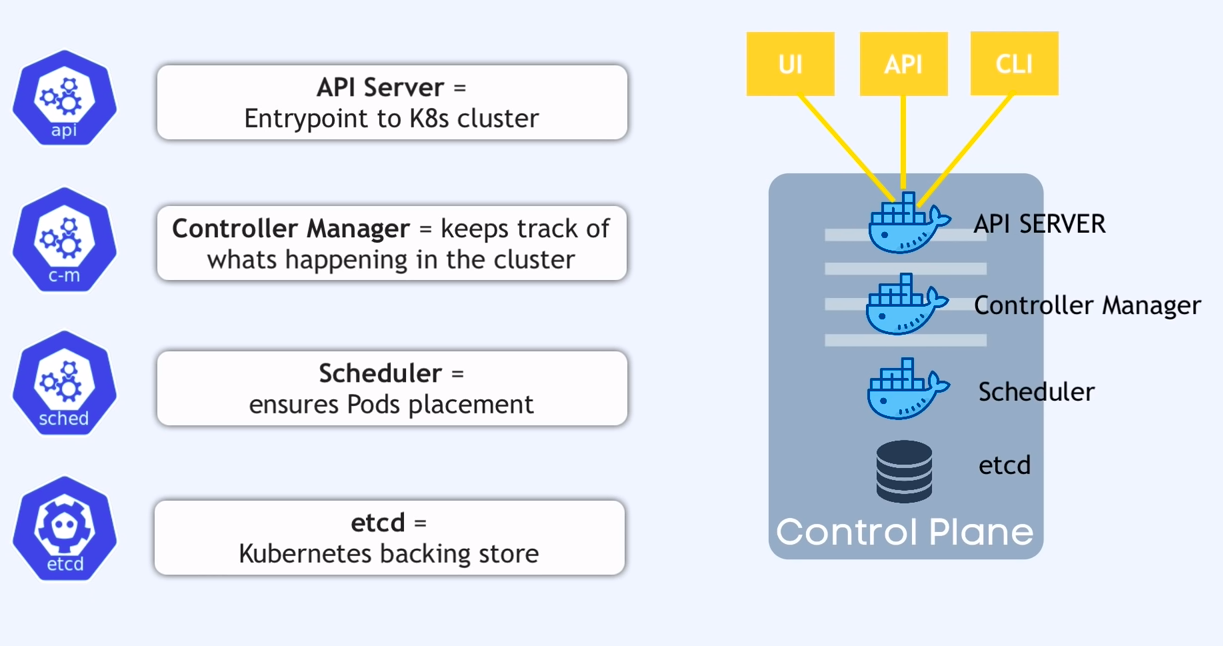


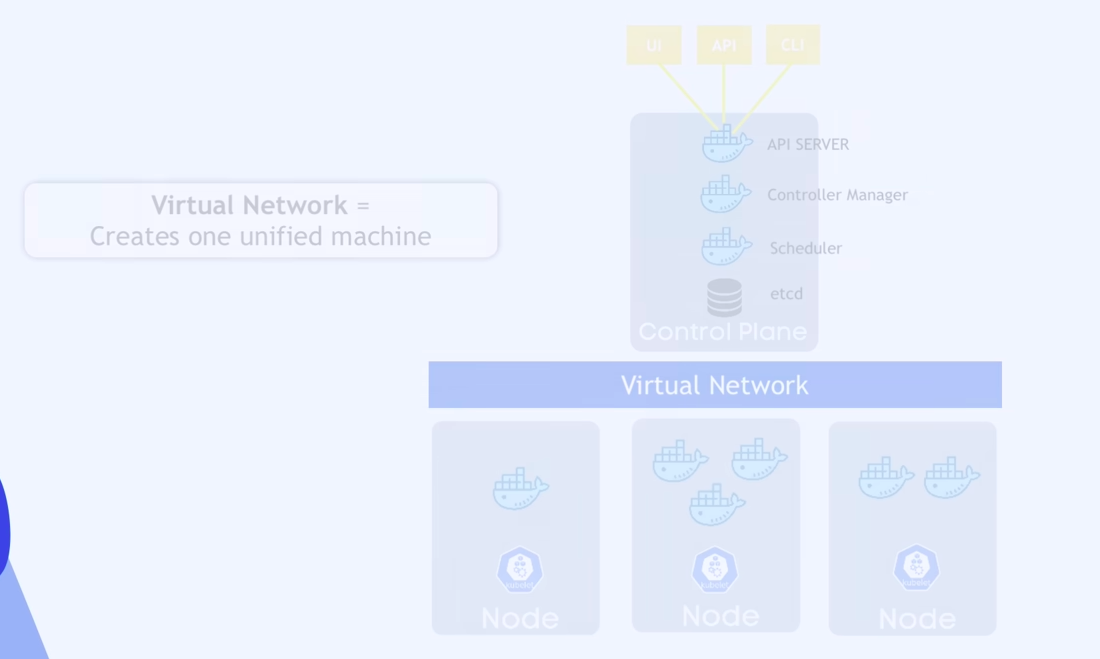
In that we have many Node with on parent node running in top of all the nodes.

Here Control plane is the main node and another is the child nodes.

Each node have many docker containers running inside it and have one kublets inside the each node.

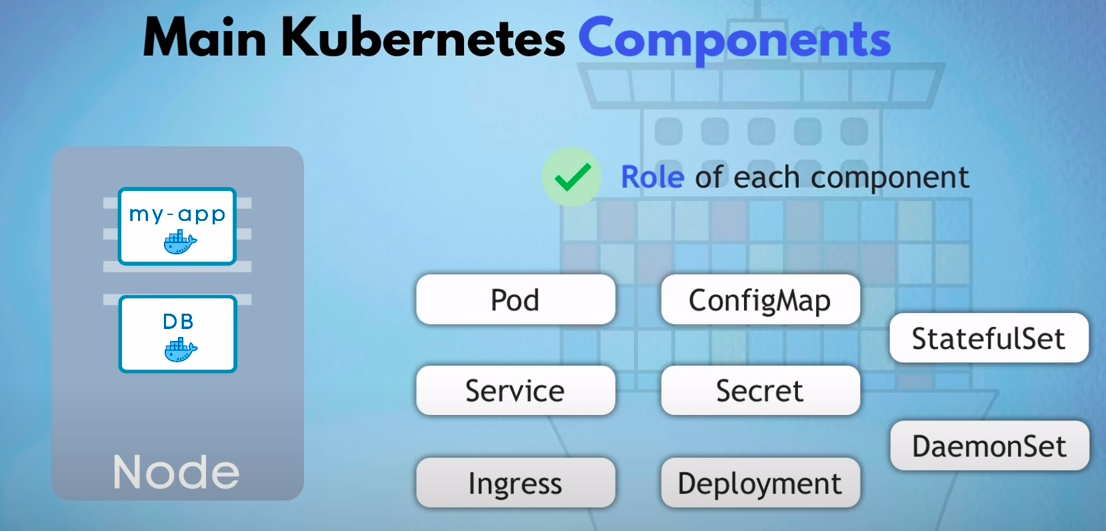
What have inside the Control Plane parent node?





Master node have two node copy if one node fails then the service can’t sttoped in production. We have more clustered.

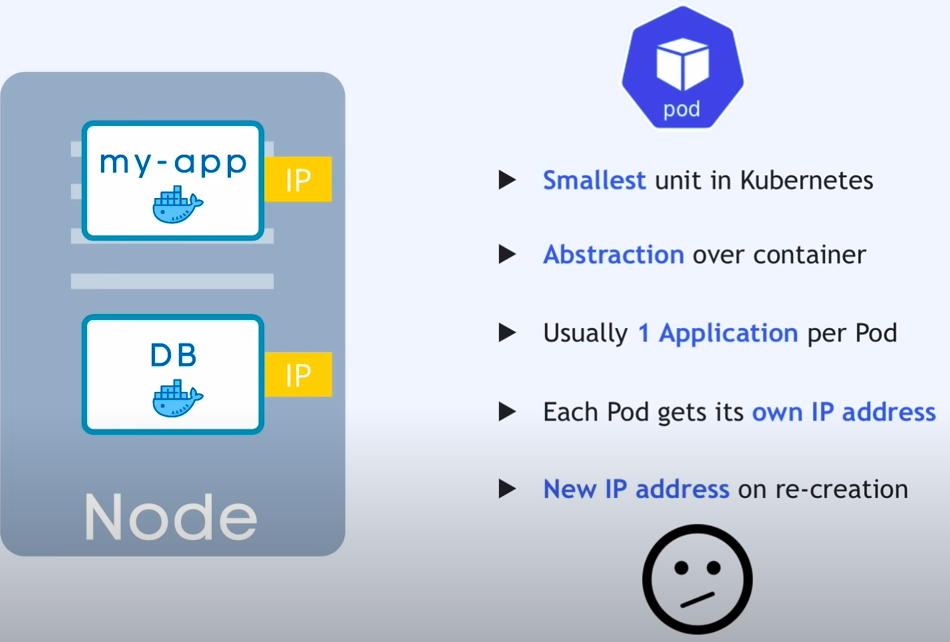
Kubernates Componnents:



Create the demo with the 2 docker container my-app and database container.

**Node and Pods:**

Node is the image or the virtual machine. (1 application per pod)



**Service & Ingress:**