Kubernetes

Kubernets is a open source orchestration tool, developed by Google.

Kubernetes take care of automatic deployment

Auto scaling of deployed apps

Distribution of load

Monitoring and health check

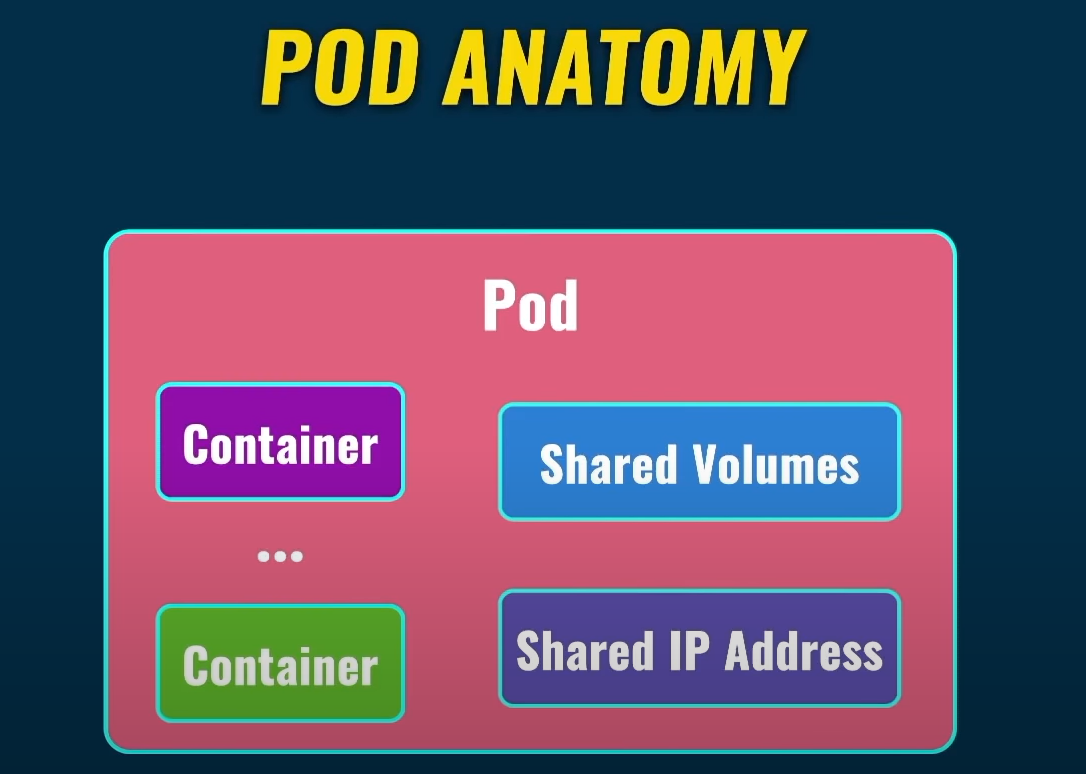
Replacement of failed containers

**POD** is the smallest unit in the kubernetes world.

In the docker world **container** is the smallest unit

Containers are created inside of the pod.

Inside the pod there are many containers, shared IP address, shared volumes.



**One container per pod is the most common use case**

Each pod must be located in the same server.one pod .. one server

**Kubernetes cluster**

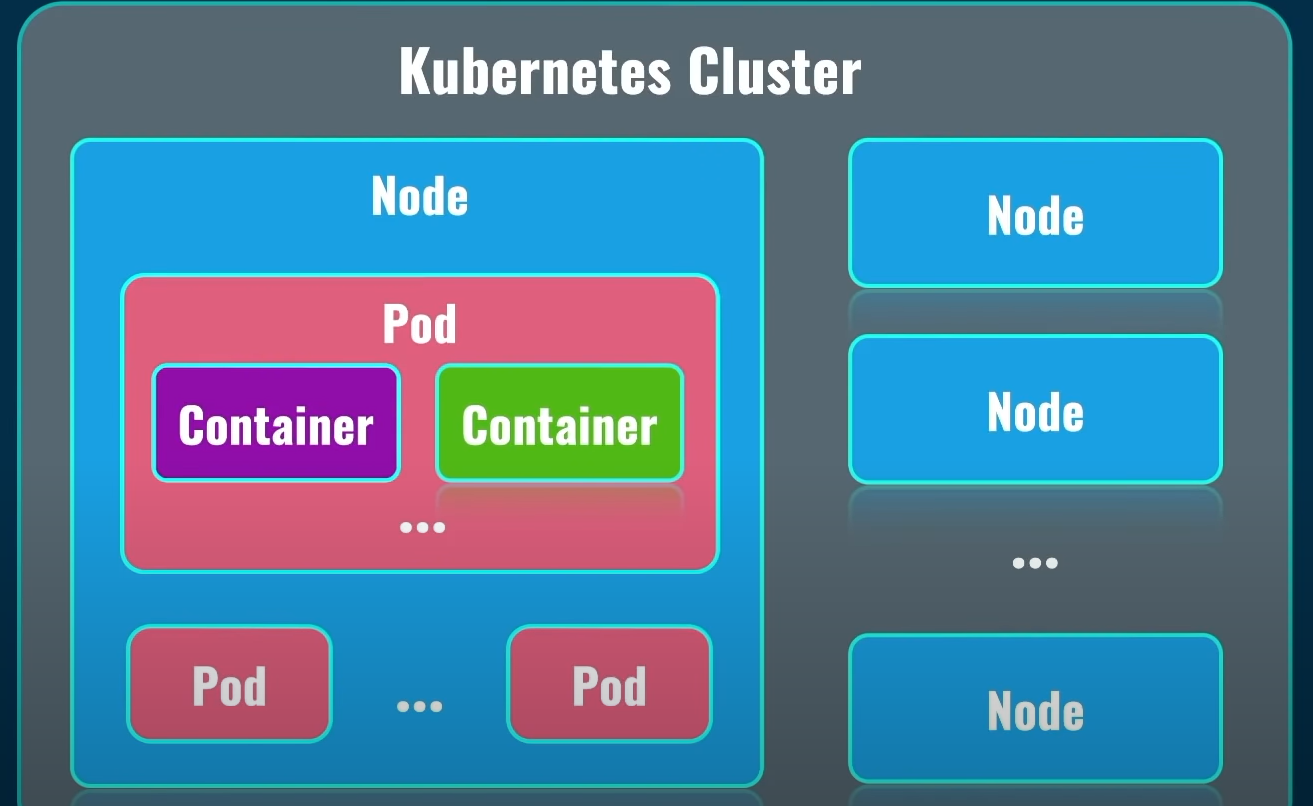
Kubernetes cluster Consist of nodes, having multiple nodes.

**Node** is a server either a metal or virtual server

Nodes which belongs to same Kubernetes cluster are located close to each other in order to perform all jobs efficiently.

Inside of the node there are multiple pods,

**KUBERNETES CLUSTER ARCHTICTURE OVERVIEW**

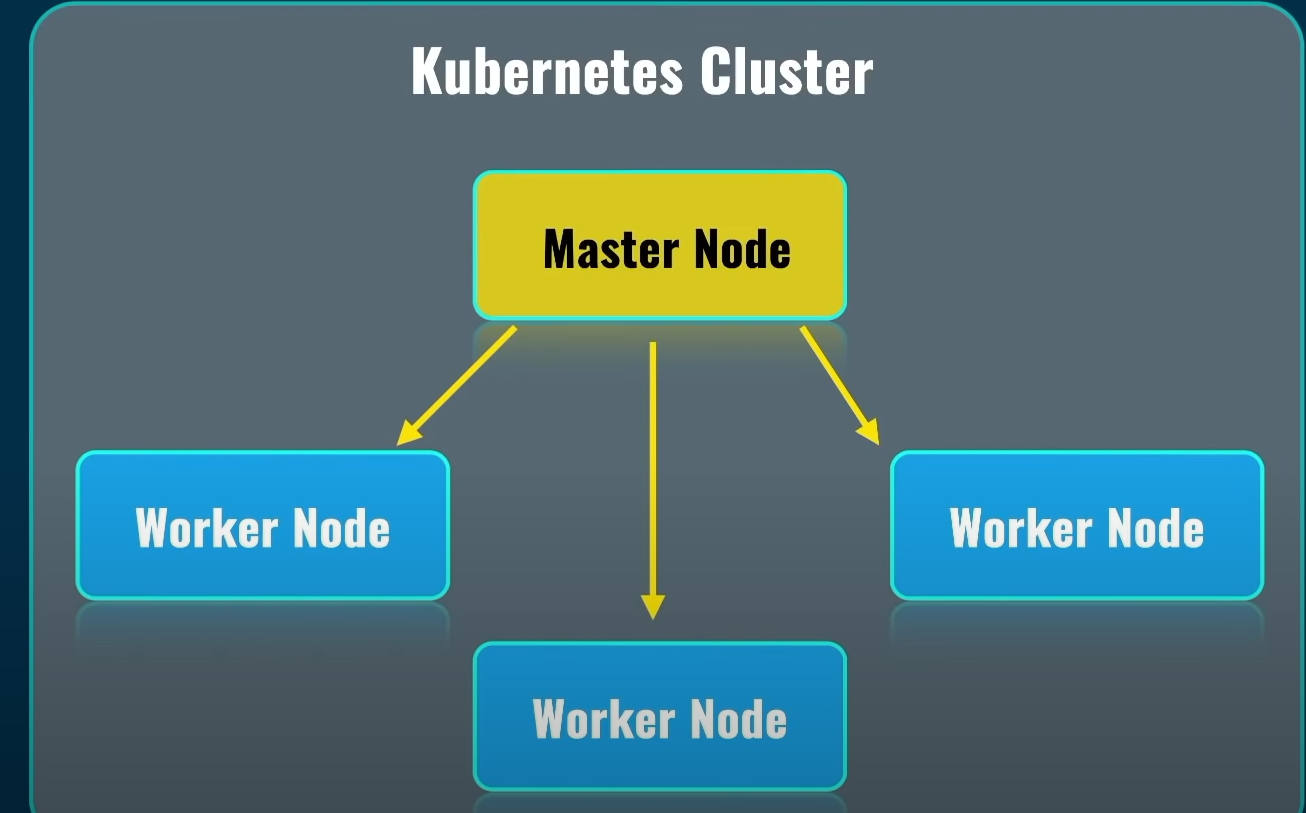


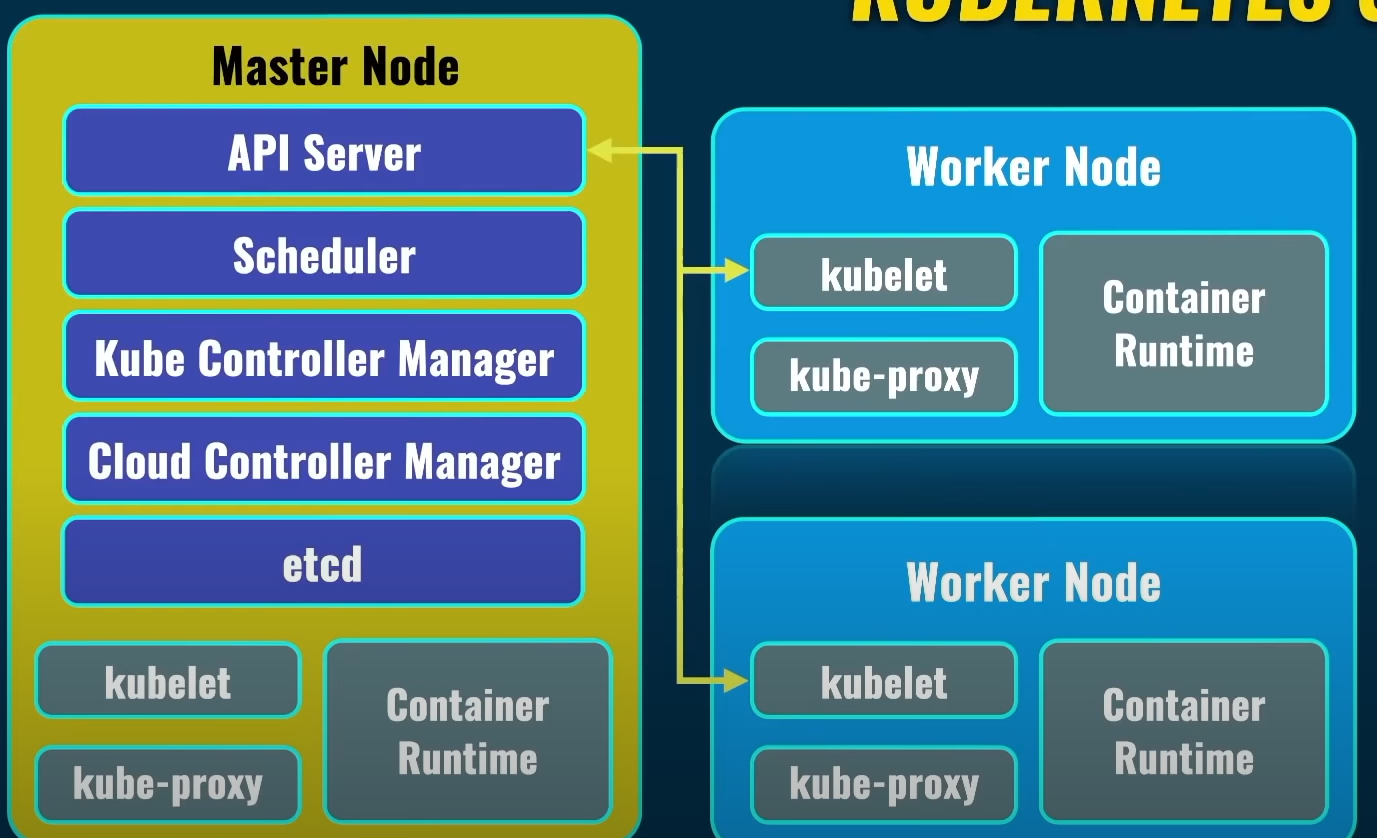
**In kubernetes cluster we have master node and worker node. Master node manages the worker node like distributing the load to worker nodes.**

All boards which are related to our applications are deployed on worker nodes

Master nodes runs only on system pods, which are responsible for actual work of kubernetes cluster in general.

Kubernetes cluster is actually control plane,it does not run our client apps.





**Kubelet, kube proxy and container run time** present in each node of kubernetes cluster

**Container runtime** runs actual containers inside of each node, there are such container runtimes as Docker CRI or container D. There is also such service as Kubelet and such service on each worker node communicates with API server service on the master node.

**API server service** is main point of communication between different nodes in the kubernetes world.

**Kube-proxy** which is present in each node is responsible for network communication inside of each node and between nodes.

**Scheduler** which is present in master node is responsible for planning and distribution of load between different nodes in the cluster.

**Kube controller Manger:** This single point which control everything actually in kubernetes cluster and it controls actually what happens on each nodes on the cluster.

The **cloud controller manager** lets you link your cluster into your cloud provider's API, and separates out the components that interact with that cloud platform from components that only interact with your cluster.

**ETCD** is a service which actually stores all logs related to operation of kubernetes cluster and stored as key value pairs.

Other services like DNS service which is responsible for names resolution in kubernets cluster.

By using API server we can manage entire kubernetes cluster. It’s done by Kubectl or kube control.

**Kube control(Kubectl)** is a separate command linetool which allows you to connect to specific k8s cluster and manages it remotely.