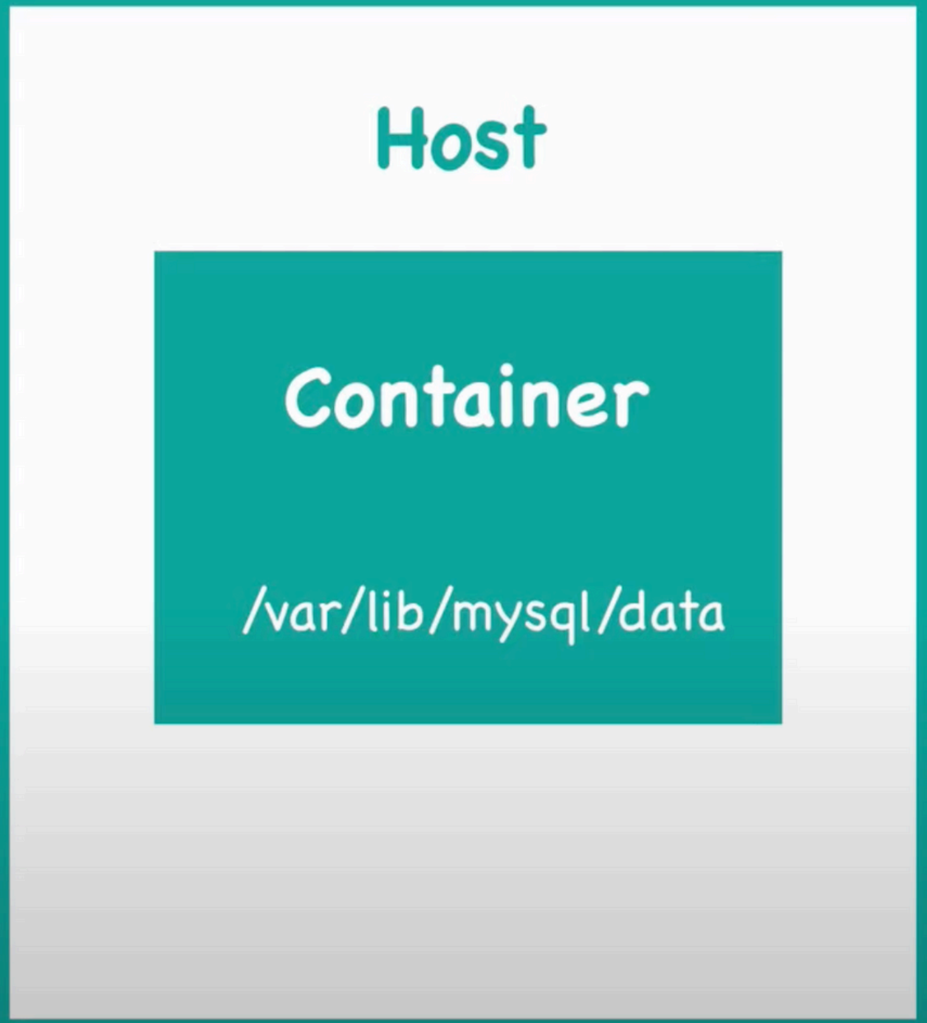
Docker - Day-5

-- Balakrishna

what we will learn today?

- *Docker Volumes.*
- *Docker Volumes Advantages.*
- *Kubernetes Intro, Why Kubernetes.*
- *Kubernetes Architecture.*
- *Kubernetes Fundamentals.*

When do we need Docker volumes?

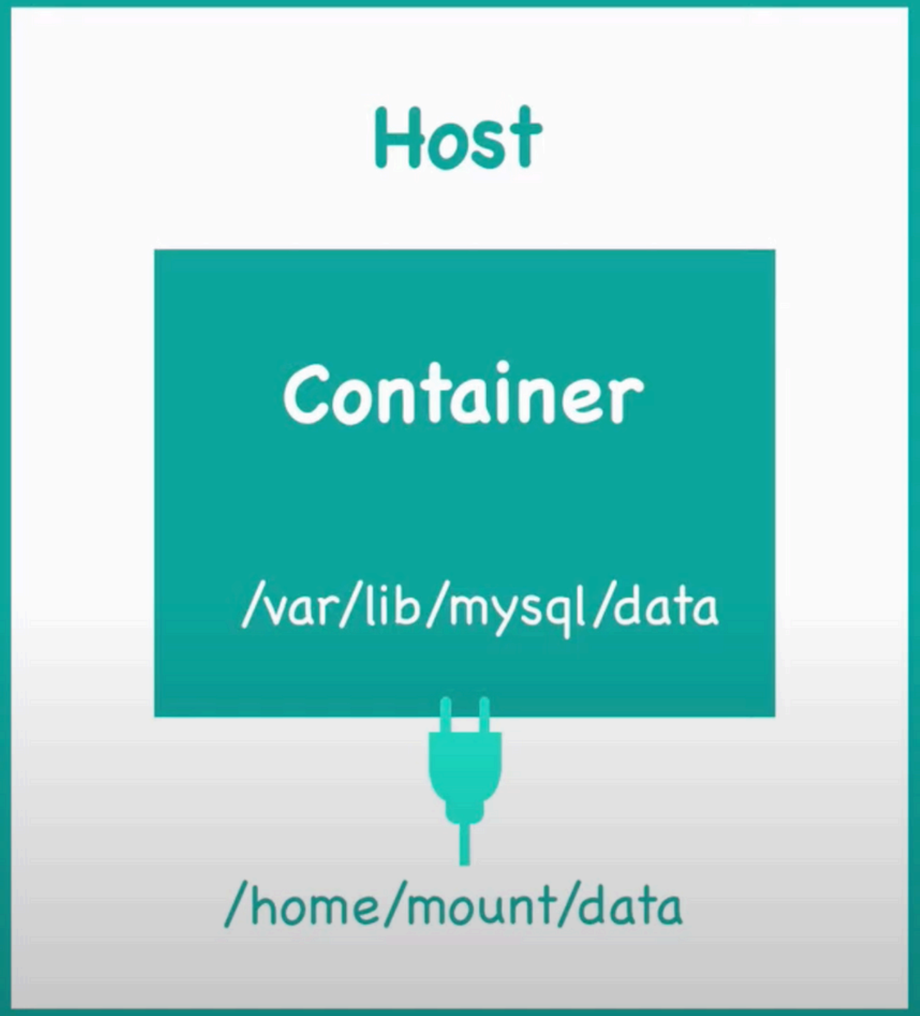


Data is gone!
..when restarting or removing the container!

Virtual File System

Database

What is Docker Volume?



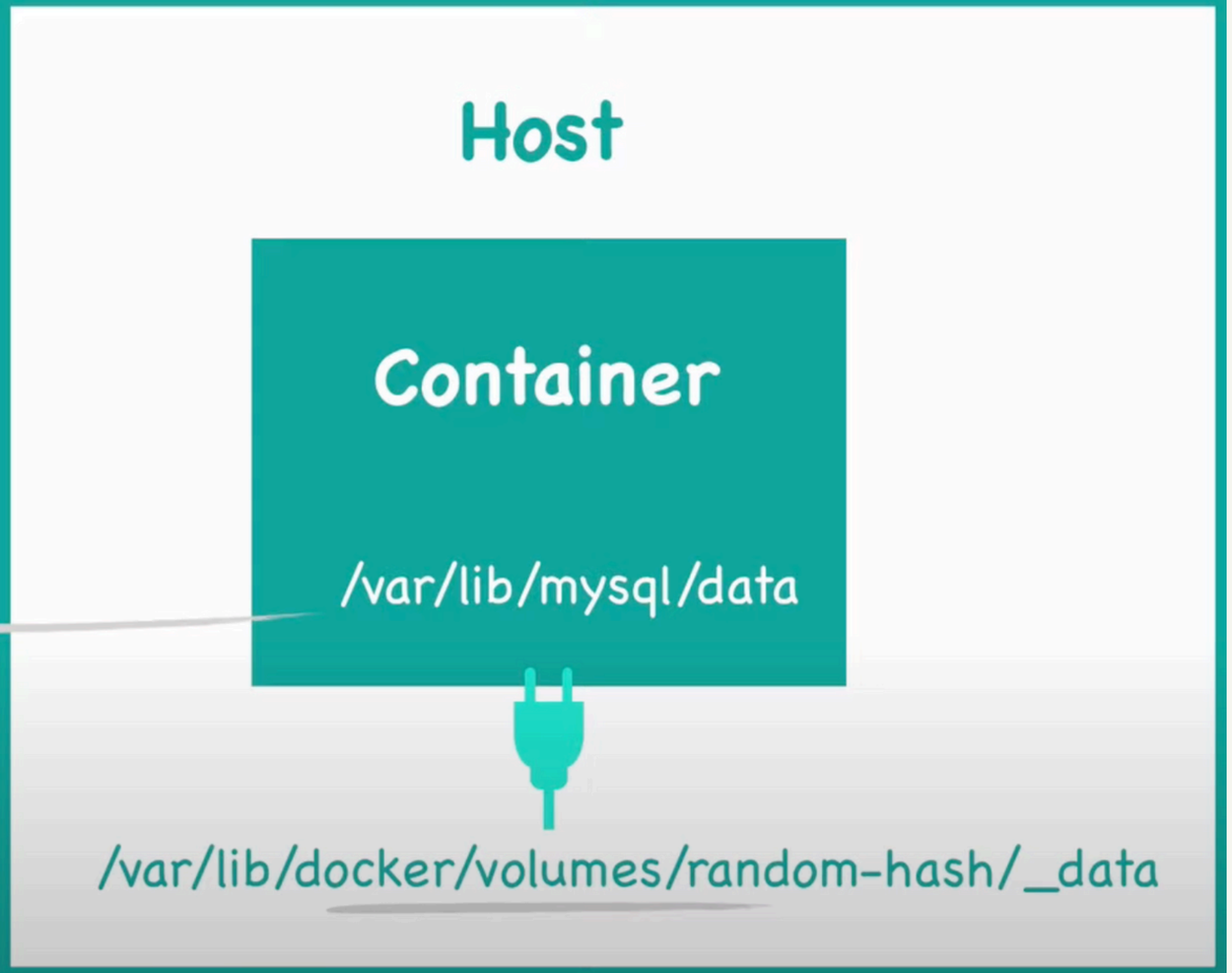
Folder in physical host file system is mounted into the virtual file system of Docker!

Virtual File System

Host File System

3 Docker volume types

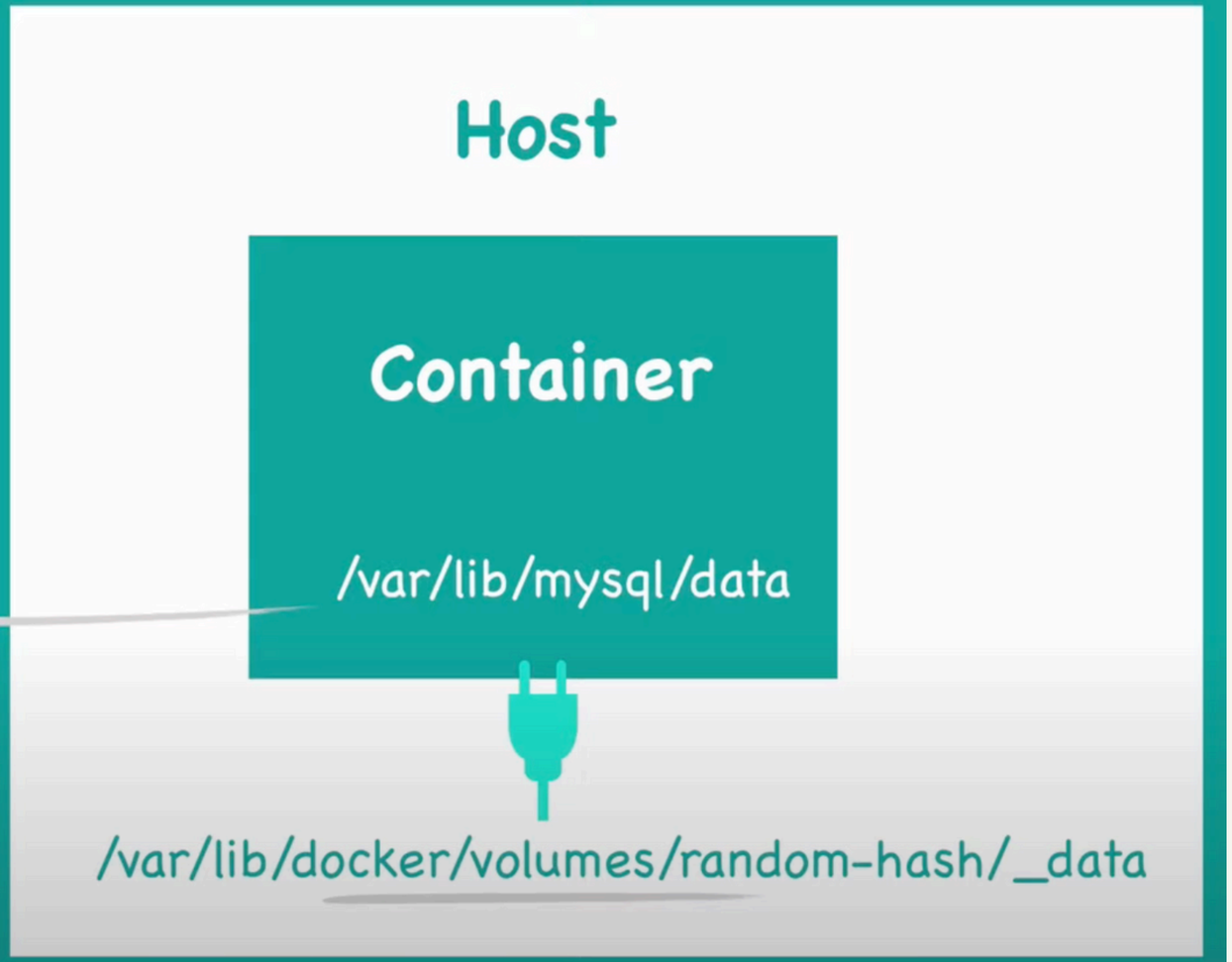
```
docker run
    -v  /var/lib/mysql/data
```



Anonymous Volumes

3 Docker volume types

```
docker run
   -v  name:/var/lib/mysql/data
```



Named Volumes

Automatically created by Docker

Kubernetes

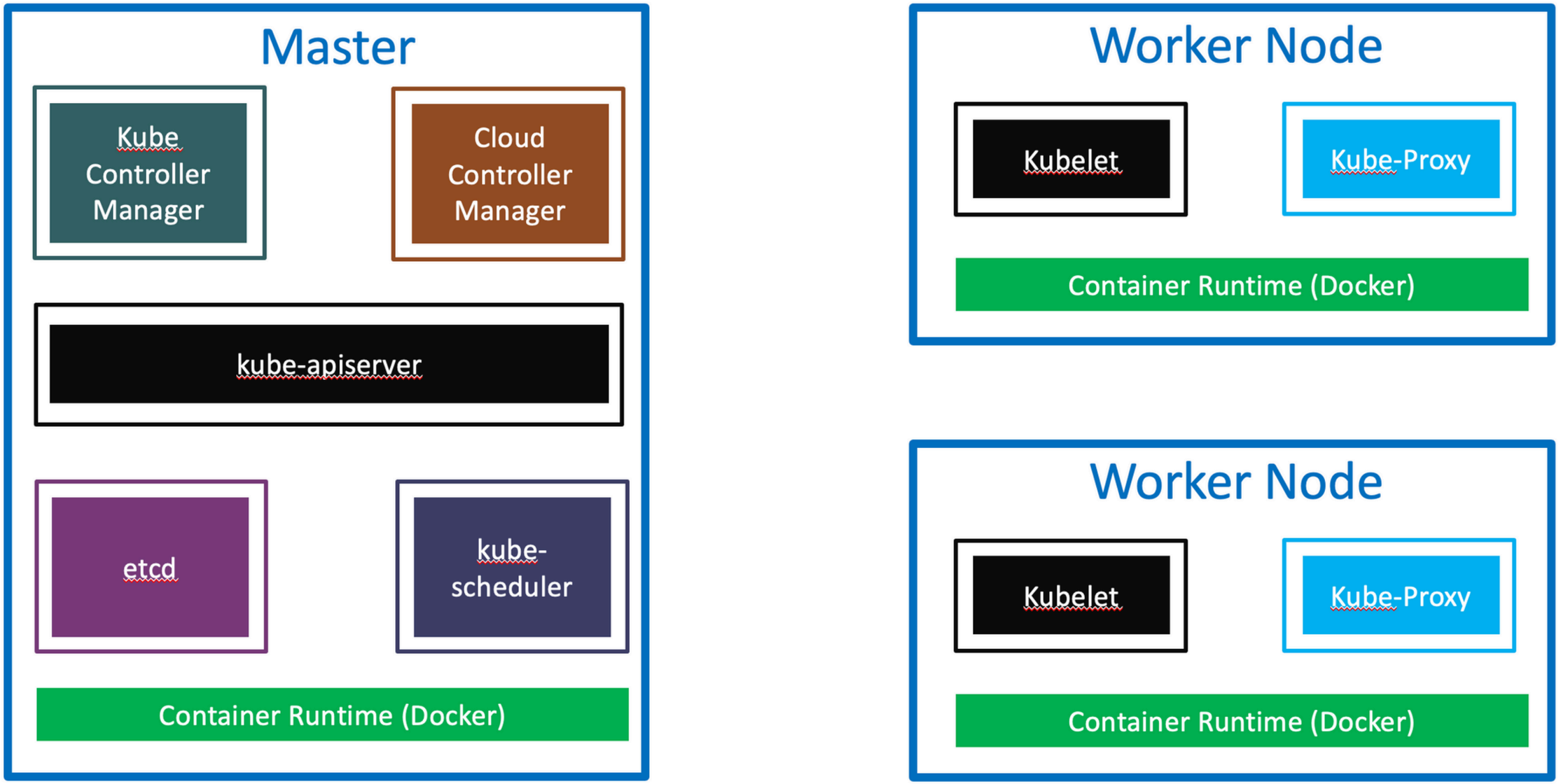
Defination:

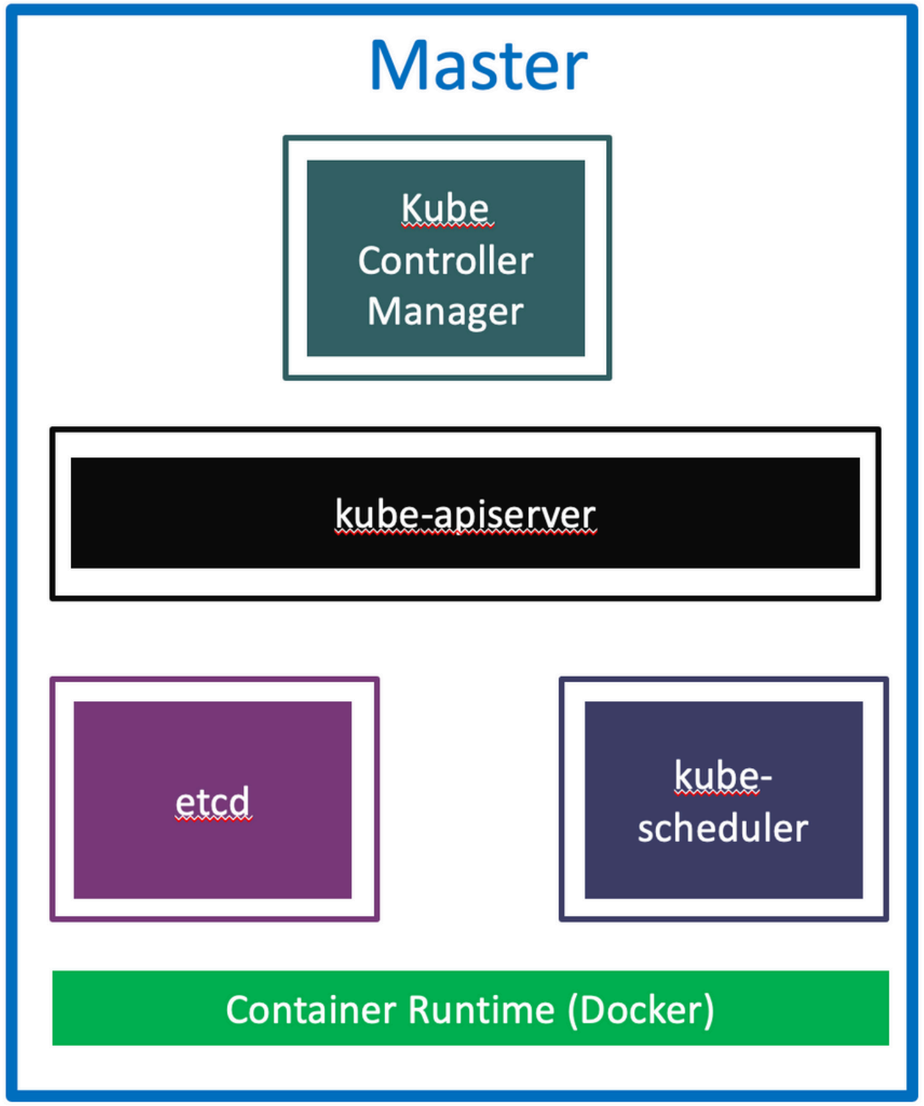
Kubernetes (also known as K8s) is an open-source container orchestration platform that automates the deployment, scaling, management, and operation of containerized applications.

Key Functions:

- Automated Deployment – Define once, deploy anywhere.
- Self-healing – Restarts failed containers, replaces dead pods, reschedules workloads.
- Scaling – Auto-scale up/down based on load.
- Load Balancing – Distributes traffic across services.
- Rolling Updates/Rollbacks – Safely update applications without downtime.
- Secret & Config Management – Inject configs and secrets without rebuilding images.

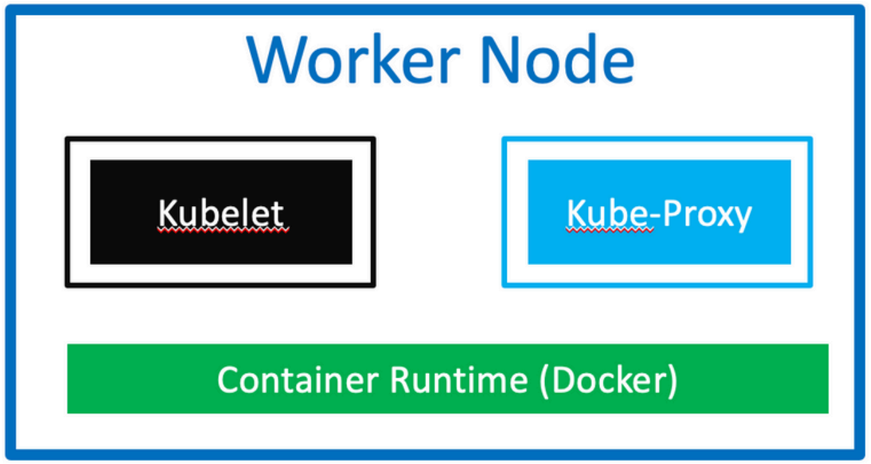
Kubernetes Architecture





- kube-controller-manager
  - Controllers are responsible for noticing and responding when nodes, containers or endpoints go down. They make decisions to bring up new containers in such cases.
  - Node Controller: Responsible for noticing and responding when nodes go down.
  - Replication Controller: Responsible for maintaining the correct number of pods for every replication controller object in the system.

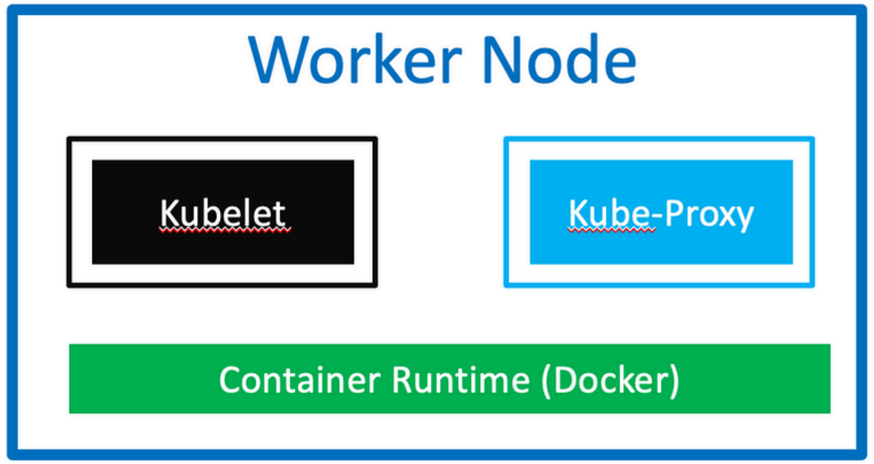
Kubernetes Architecture – Worker Nodes



- Container Runtime
  - Container Runtime is the underlying software where we run all these Kubernetes components.
  - We are using Docker, but we have other runtime options like rkt, container-d etc.

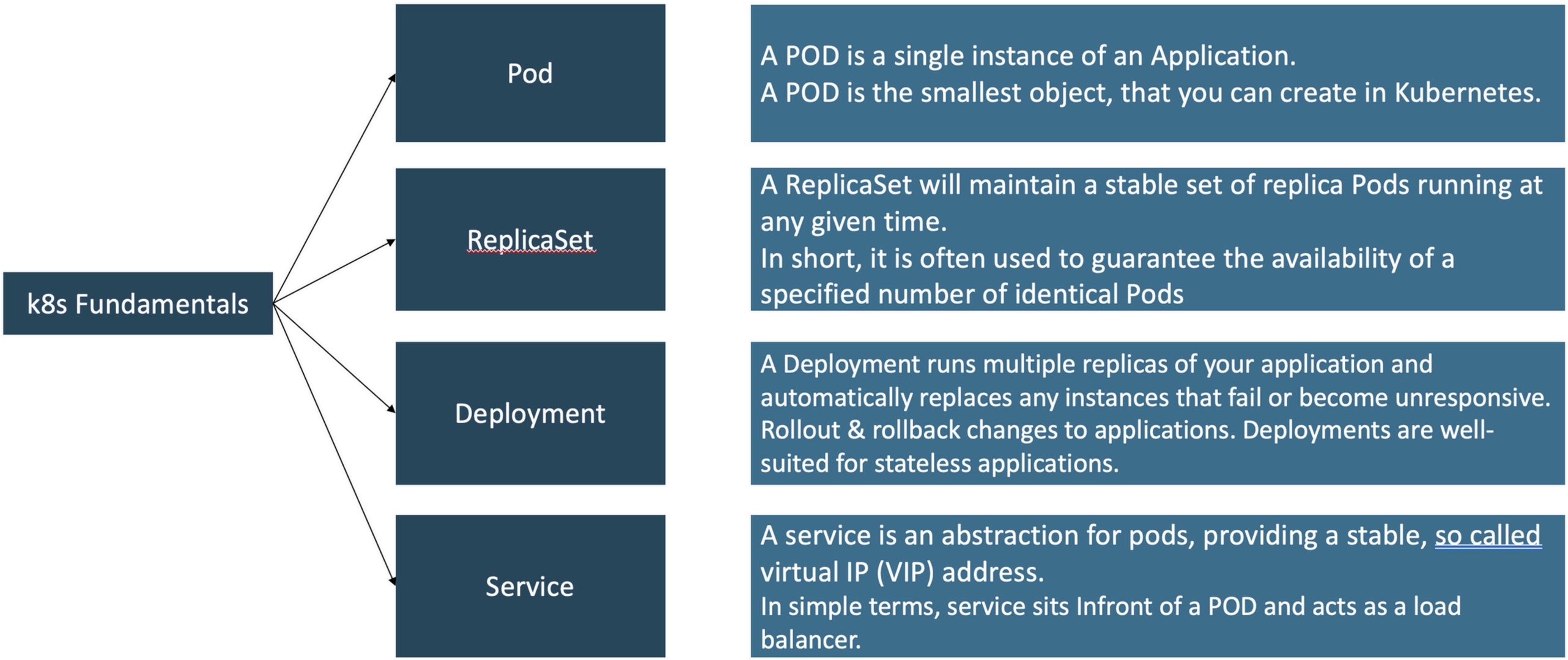
- Kubelet
  - Kubelet is the agent that runs on every node in the cluster
  - This agent is responsible for making sure that containers are running in a Pod on a node.
- Kube-Proxy
  - It is a network proxy that runs on each node in your cluster.
  - It maintains network rules on nodes
  - In short, these network rules allow network communication to your Pods from network sessions inside or outside of your cluster.

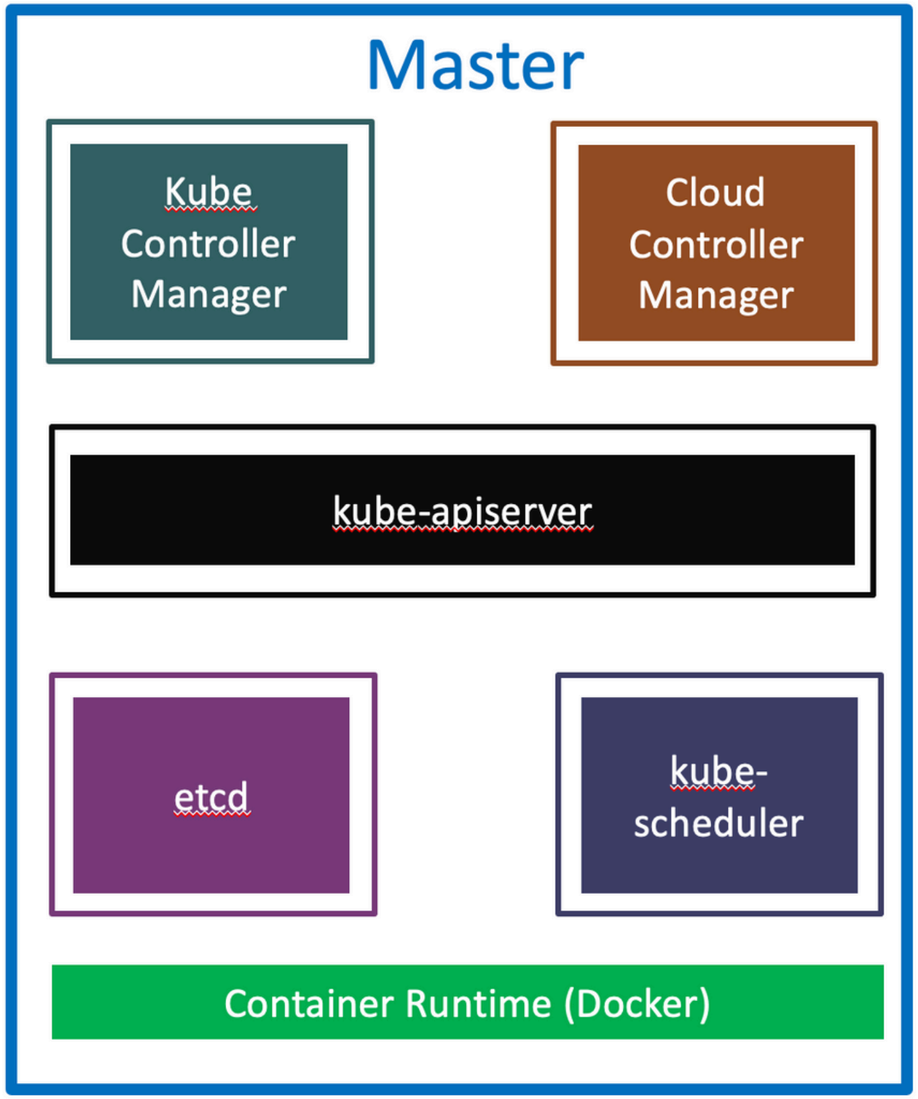
Kubernetes Architecture – Worker Nodes



- Container Runtime
  - Container Runtime is the underlying software where we run all these Kubernetes components.
  - We are using Docker, but we have other runtime options like rkt, container-d etc.

- Kubelet
  - Kubelet is the agent that runs on every node in the cluster
  - This agent is responsible for making sure that containers are running in a Pod on a node.
- Kube-Proxy
  - It is a network proxy that runs on each node in your cluster.
  - It maintains network rules on nodes
  - In short, these network rules allow network communication to your Pods from network sessions inside or outside of your cluster.


k8s Fundamentals
Pod
ReplicaSet
Deployment
Service
A POD is a single instance of an Application.
A POD is the smallest object, that you can create in Kubernetes.
A ReplicaSet will maintain a stable set of replica Pods running at any given time.
In short, it is often used to guarantee the availability of a specified number of identical Pods
A Deployment runs multiple replicas of your application and automatically replaces any instances that fail or become unresponsive. Rollout & rollback changes to applications. Deployments are well-suited for stateless applications.
A service is an abstraction for pods, providing a stable, so called virtual IP (VIP) address.
In simple terms, service sits Infront of a POD and acts as a load balancer.



- kube-apiserver
  - It acts as front end for the Kubernetes control plane. It exposes the Kubernetes API
  - Command line tools (like kubectl), Users and even Master components (scheduler, controller manager, etcd) and Worker node components like (Kubelet) everything talk with API Server.
- etcd
  - Consistent and highly-available key value store used as Kubernetes' backing store for all cluster data.
  - It stores all the masters and worker node information.
- kube-scheduler
  - Scheduler is responsible for distributing containers across multiple nodes.
  - It watches for newly created Pods with no assigned node, and selects a node for them to run on.

Thank You!!

Keep upskilling!!!