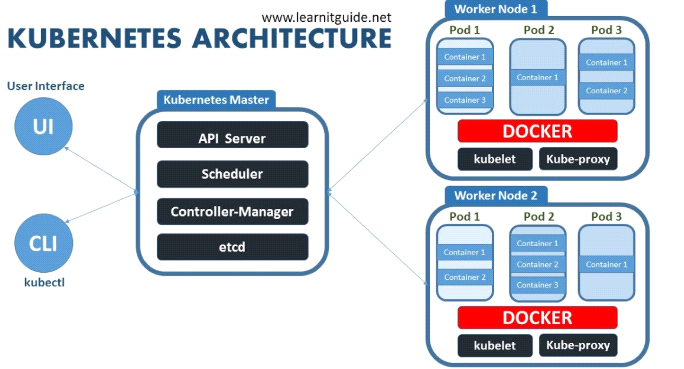
**What is Kubernetes**

Kubernetes is container management and Orchestration Engine provides the features nothing but managing/Deploying/Linking/comunicating/Autoscaling/loadbalcing/Upgrading of containers.

**Features: -**

* Automatic Bin Packing
* Service discovery and load balancing
* Storage Orchestration
* Self-healing
* Batch Execution
* Secret & Configuration Management
* Horizontal scaling
* Automatic Rollback & Rollouts
* High Availability

**High-level Architecture of Kubernetes:**



**Master Node:** Responsible for Managing the Kubernetes cluster.

It is the entry point for all administrative tasks we can Manage Node with CLI/API/GUI or DASH Board

Can be more than one master Node for Load balancing in cluster out of many only one will be the leader

**Master Node components: -**

* ETCD
* Scheduler
* Controller manager
* API server

**ETCD** – It is a data base. Which stores information in key value based. It stores the following information.

* configuration data of cluster
* Present state of the cluster.
* Each components state of cluster.

Any object can query the etcd to know the cluster state it is single source of the truth of all components & Nodes of the cluster.

**Scheduler –** It is a program on master node which performs the scheduling tasks like launching pods on worker nodes based on resource availability (It maintain resources artifacts in Kubernetes cluster)

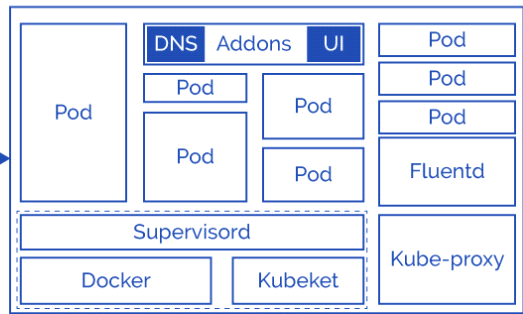
**Controller Manager –** It’s responsible for co-ordination & health of the entire cluster it is going to be conductor and co-ordinator ensuring that nodes are up and running and pods are behaving right way and maintain desired state of the Pods.

* Node controller: Responsible for on boarding new nodes and responding when nodes goes down.
* Replication controller: Responsible for maintaining the desired number of pods running all the time.

**API Server –** API server is kind of gatekeeper for entire cluster. It is Responsible for orchestrating all operations within the cluster. It exposes the Kubernetes APIs to nodes, it is the front-end for the Kubernetes control plane.

The Kubernetes API server validates and configures data for the API objects which include pods, services, replication controllers, and others.

**Worker Node components: -**



* Kubelet
* Docker Engine
* Kube-Proxy
* Pod
* fluentd
* Addons

**Kubelet :**

The main Kubernetes agent registers node with cluster runs on every worker node, talks to API server

Update the configuration of the Node to the etcd of master server.

Initiate pods in worker node based on the specification

Exposes end point on: 10255

**Docker Engine –** It’s a platform to run containers present on each Worker Node, pulling images, stopping and starting containers.

Kublet & Docker Engine together part of supervisord layer/process.

**Kube-Proxy –**It is core networking element responsible for entire network configuration & management. It routes the traffic to appropriate containers based on IP address and port number of the incoming request. In other words, we can say it is used for port translation.

**Pod** : Pod is basic deployment unit in Kubernetes, runs with single/Multiple containers. Containers share same IP address communicates with localhost each other’s with IPC (Inter Process communication).

**Container:** It provide the run time environment for the micro service applications contain application libraries & binaries

**Fluentd:** is responsible for Logging of the all Node components talking to central logging mechanism if you configure

**Addons:** Addons are optional for the Node but it makes admin life easier ex: DNS,Calico ,flannel.