KUBERNETES

microservices run on containers
- kubernetes creates, manages & provides scalibility
- to container.
- K8s provide high availibility.

K8s automate container deploymet, container
- scaling and load balancing.

- It scheduls runs and managed isolated Contaner which are running on virtual physical cloud machine.

cloud - using computing services from cloud provider
cloud native - taking best cloud functionality to develop something better.

* online platform for K8s
- K8s playground
- play with K8s

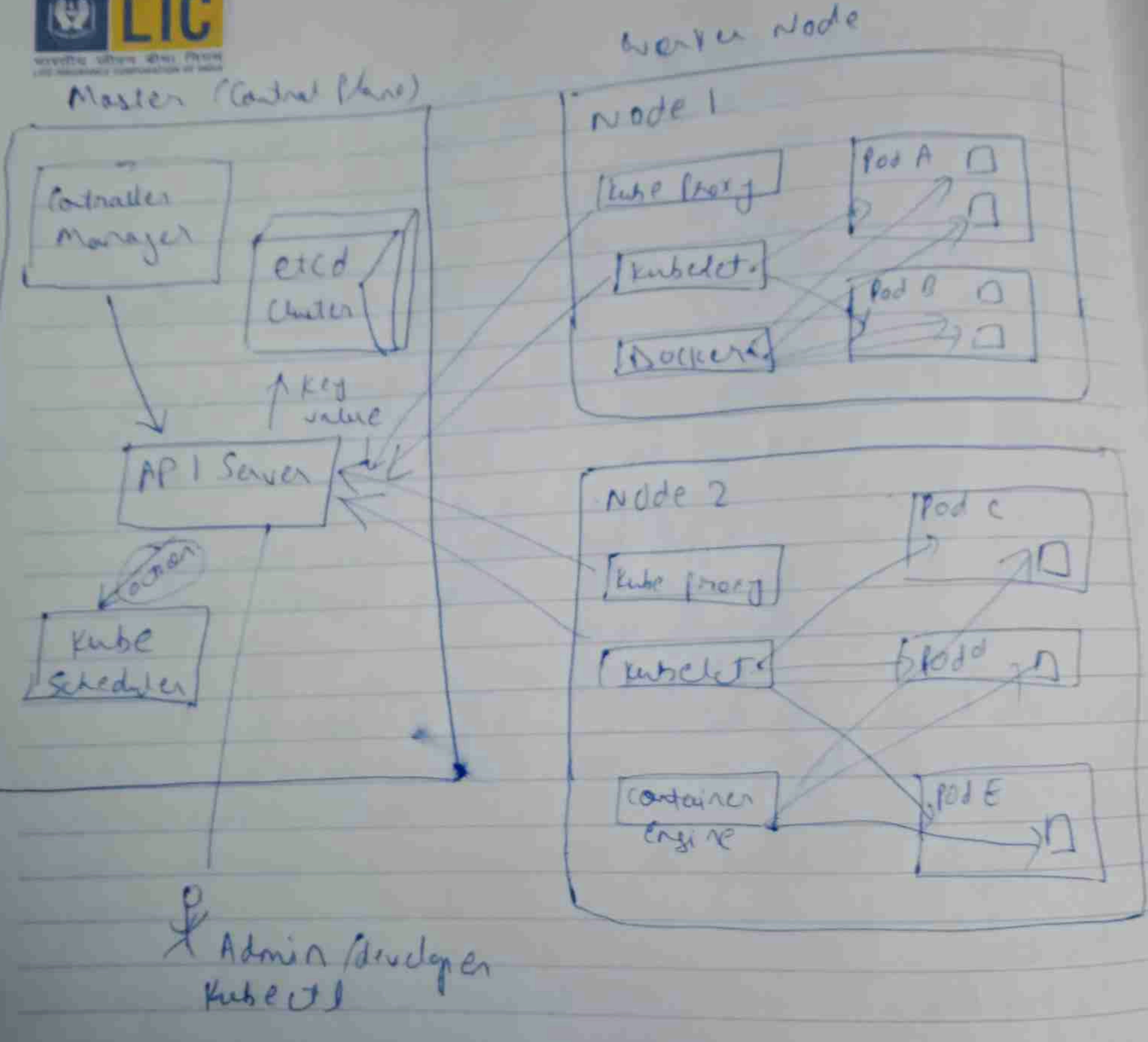
* Local
- minikube
- kubeadm

* Features
- Orchestration
- Load balancing
- Fault tolerance of containers
- Autoscaling
- platform independent
- Rollback
- Batch execution (one time, sequential, parallel)
- Auto Healing
- Health monitoring

- K8s follow master-slave Architecture

cluster → node → Pod → container → App/microservi

ARCHITECHTURE of K8s



* Working with K8s

-> Role of Master Node

- K8s cluster contains containers running on bare metal / VM instance / cloud instance / all mix

- K8s designates one or more of these as master and other as workers.

- The master is now ~~going~~ to run set of K8s processes. These processes will ensure smooth functioning of cluster. These process are called Control plane.

- Can be multi master for high availability.

- Master runs Control plane to run cluster smoothly

(*) Components of control plane

1) Kube - API Server (for all communications)
This API server interacts directly with user. That is we apply yaml or json manifest to Kube Api-server

- This Kube Api server is meant to scale automatically as per load.

- Kube Api server is front end of control plane

2) etcd
stores metadata and status of cluster.

- etcd is consistent and high available store (key value store)

- Source of touch for cluster state (info about state of cluster) etcd has following features :-

1) fully replicated - entire state is available on every node in cluster

2) Secure - Implements automatic TLS with optional client-certificate authentication.

3) fast - Benchmarked at 10,000 writes / second

(3) Kube Scheduler (action karega)

- When users make request for the creation and management of pods, Kube scheduler is going to take action on these request.
- Handles pod creation and management.
- Kube scheduler match / assign any node to create and run pods.
- Scheduler gets the information for h/w configuration from configuration files and schedules the pod on nodes accordingly.

(4) Controller manager Components

Node Controller — detect nod on cloud
Route Controller ~~(load)~~ (setting up network)
Service (load balancing)
Volume controller (mounting volume)

Worker Node

1) Kubelet
Agent running on node , Port 10255
Listen to K8s master
send success / fail report to master

2) Container Engine
works with kubelet.
pulling images
exposing containers on port specified in manifest.

3) Kube proxy
- Assign IP to each pod
- Runs on each node and make ensure that each pod will get its Own unique IP address

POD

- Smallest unit
- Can't start Container without a Pod.
- A cluster is a group of nodes
- A cluster has atleast one worker node and one master node

* Multi container POD
- Share access to memory space
- Connect to each other using local host
- Share access to same volume.

* High level K8s object *

Replication set - Scaling and healing
Deployment - versioning and Rollback
Service - Static (non ephemeral) IP and network
Volume - Non ephemeral storage.