Kubernetes

* Kubernetes → K8s

→ Kubernetes is container management tool or container orchestration tool.

→ Developed by Google labs & later donated to CNCF (Cloud Native Computing Foundation)

→ Open source.

→ Written on Go programming language or Golang.

→ Also it is called as K8s.

* Container ~~mage~~ management (orchestration) tool :-

It is tool or engine automates deploying, scaling & managing containerized application on a group of servers.

Ex: Kubernetes, Docker Swarm, Apache Mesos Marathon etc.

Kubernetes

↓

container management tool

↓

Manages containerized applications → Deploying, Scheduling, Scaling, load balanci

↓

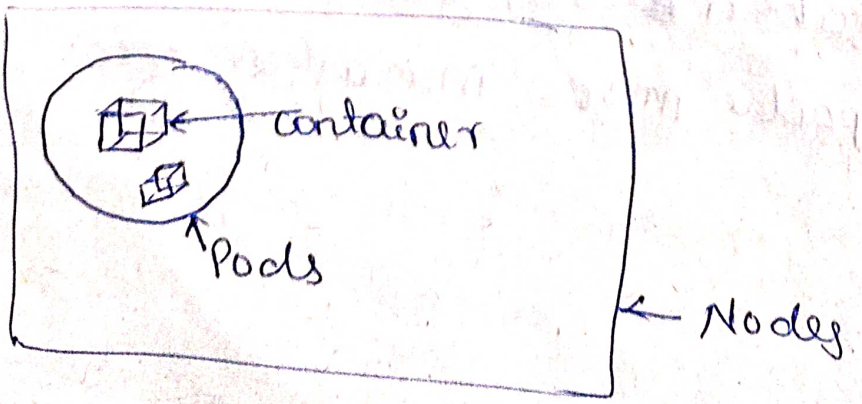
Application available on a container platform like Docker.

* Features of Kubernetes :-

1) Automatic bin packing :—

Based on requirement & available resources kubernetes automatically packages your application & schedules the container.

Pods and Nodes

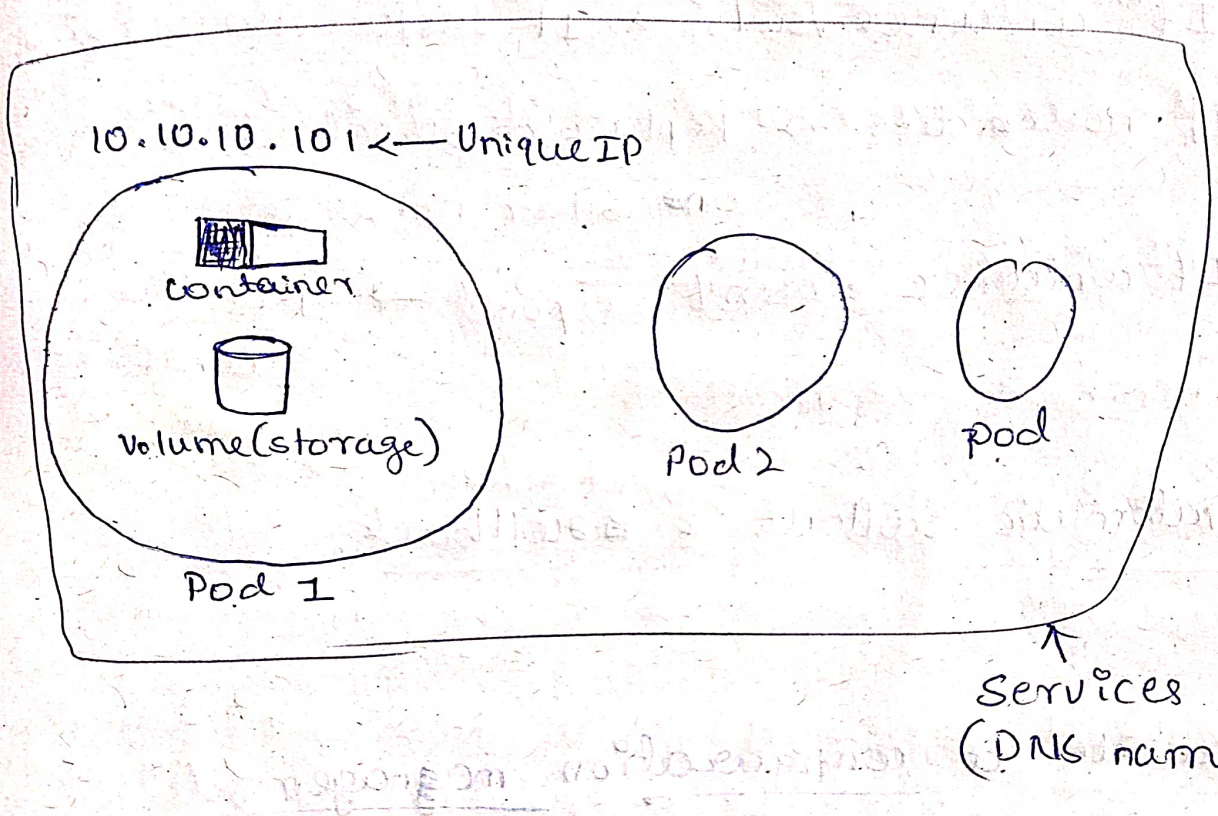


2) Service discovery & load balancing :-

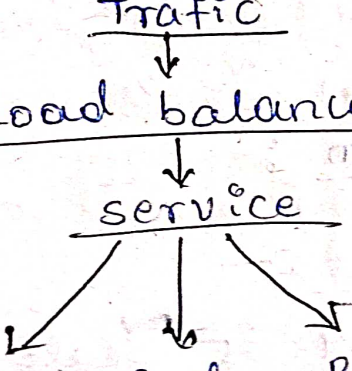
Pods have same set of fn's are abstracted into sets, called services.

Pods contains

1) An application container (one or more)
2) ~~Storage~~ Storage resources
3) A unique network IP.



→ Kubernetes control over network & communication b/w pods & can load load balance across them



3) Storage Orchestration

Kubernetes allows to mount the storage system of your choice, i.e, local, cloud (AWS), Network (NFS)

NFS - Network file s/m

4) Self Healing :—

⇒ If container fails → It restart the container

⇒ If node dies → Replace & reschedule containe one other nodes.

⇒ If container doesn't respond → Kill the container.

5) Automatic rollout & Rollback

6) Secrete & configuration management.

→ Sensitive data like password, keys, tokens are handled using secrets.

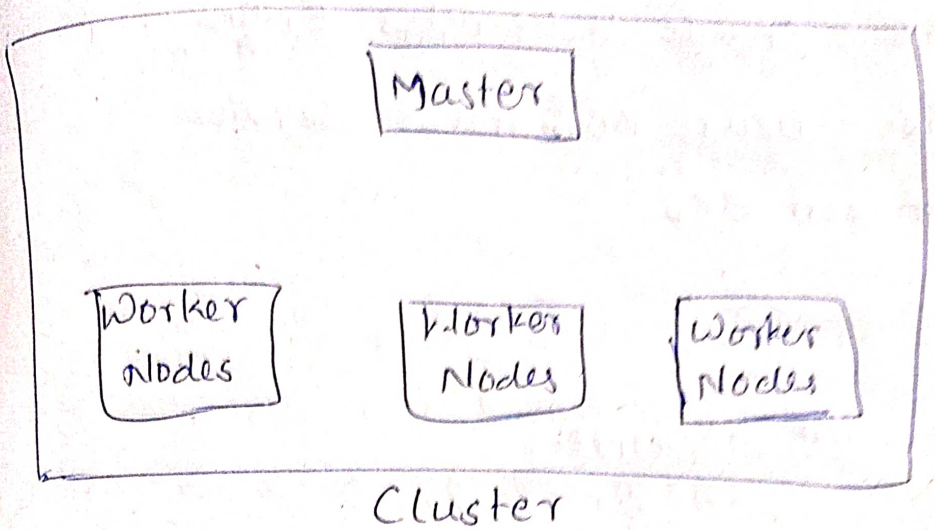
→ Configurations are handled using config map.

7) Batch execution :—

8) Horizontal scaling :—

We can scale up & scale down the containe

Architecture



* Components of master node

1) API server → for all communications
2) Scheduler → Schedules pods on nodes
3) Controller Manager → Runs controllers
4) Etcd → Oper source, distributed key-value database from coreOS.

* kubectl → Command line used to users interact will API.

Hands-on

* Online :- 1) Kubernetes playground (Katacoda)
2) Play with K8s
3) Play with Kubernetes classroom.

Cloud based Kubernetes service.:-

GKE - Google Kubernetes Engine

AKS - Azure Kubernetes Service

Amazon EKS.

* Start with Kubernetes

» launch.sh → Launch cluster

» kubectl cluster-info → Health check

* Step 1 :- Start Minikube

$ minikube version

$ minikube start --wait=false → Start the kubernetes.

$ kubectl get nodes → To view nodes in the cl

$ kubectl creat deployment first-deployment
--image=katacoda/docker-http-server

→ Deployment created.

$ kubectl get pods → Status of deployment

$ kubectl expose deployment first-deployment
--port=80 --type=Nodeport

→ service first-deployment exposed.

$ minikube addons enable dashboard

→ To enable dashboard