kubernetes

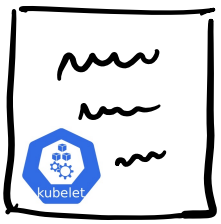
- **What is Kubernetes?**
  - Open sources container orchestration tool.
  - Developed by Google!
  - Manages containerized apps in different environments.

  automation of operational efforts of containerized apps and microservices

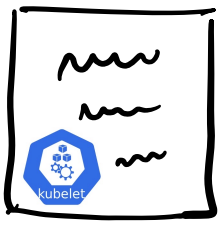
- **What problems does it solve?**
  - Transitions from monolith to microservices
  - Increased usages of containers
  - Unorganized management of containers

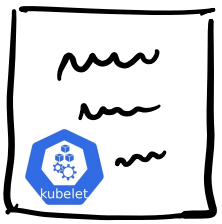
- **Features:**
  - High availability - no downtime
  - Scalability - high performance
  - Disaster recovery - backup and restore

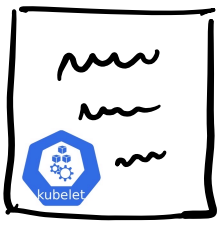
Kubernetes Basic Architecture :



Master node (at least one)







few worker nodes

- Worker nodes have kubelet process running in them!

Kubelet - primary "node agent"
- kubernetes process that makes it possible for the cluster to talk to each other and execute tasks

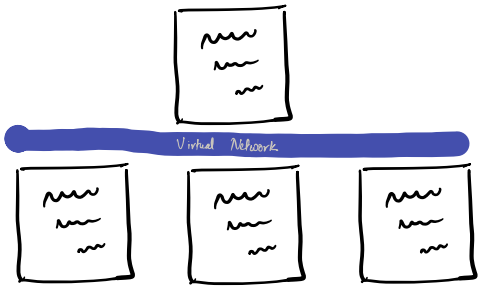
- Worker node has some docker containers running on it.
- Master node - important K8s processes are running

Processes in master node:

1. API server : Entry point to K8s cluster
2. Controller manager: Keeps track of whats happening
3. Scheduler : ensures Pod placements.
   schedules containers based on workload resources
4. etcd : key-value store. Stores configs and states of all containers
   Used in disaster recovery

Virtual Network

Spans all nodes to unite all nodes into one.

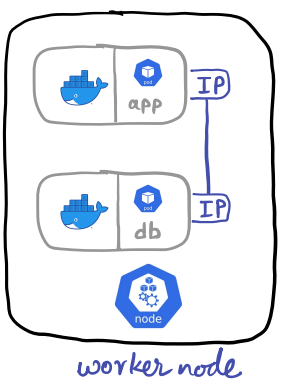


Kubernetes Components

① Pod and node

Pod:
- Smallest unit of K8s
- Abstraction over container
- Creates a layer of running environment on top of containers
- Meant to run one app container in it.
  (can run more)

To communicate between pods, each pod gets an IP address via VN.
- New IP address for each pod, even if it runs the same app.

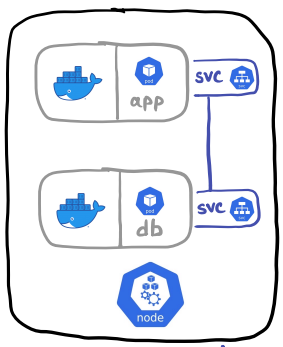


worker node

② Service and Ingress

Its hard to manage IPs on recreation.

- Service is a permanent IP address to each pod
- Service and pod lifecycles are not connected
  meaning when a pod dies, service stays



worker node

- Also has a load balancer
  - catches the request and forwards to the least busy pod.

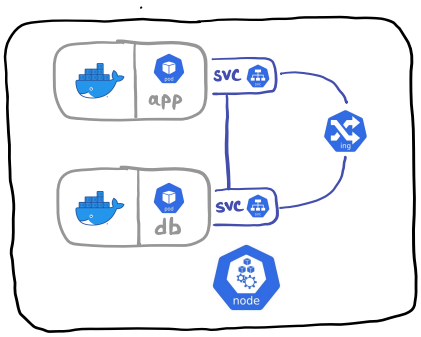
Types of services:

① External : Accessible outside of the node and cluster

② Internal : Accessible in nodes internally.

Ingress:

Prettifies the IP for outside world. Does forwarding inside



worker node

Config Map and Secret:

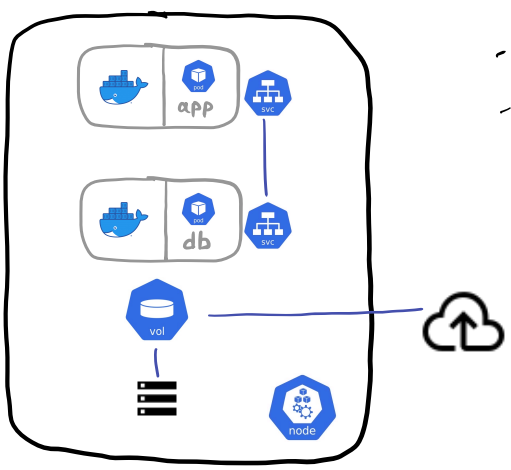
Config map: Configuration data, as plain text

Secret: Same as config map, but secrets, as base64 encoded

- built in security mechanism is not enabled by default.

Volumes:

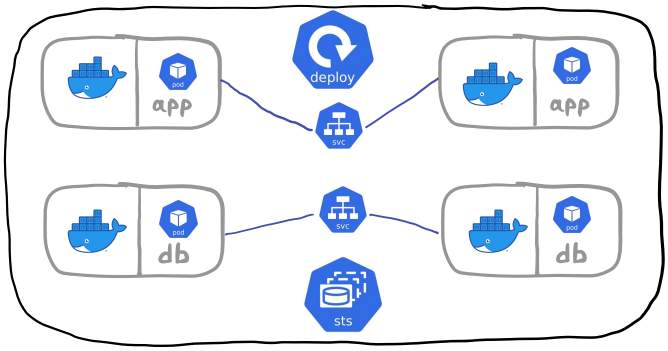
Persistence of data
Attaches a physical drive to pod



- Can be a local storage
- Can be remote
- User solely responsible for managing the state of data on volumes

Deployment Stateful Set

- When the app pod dies, there is a downtime.
- Replicate the node to avoid downtime
- Can share same service



- define blueprint of the pod and # of instances

Deployment is this blueprint.

In practice, user does not create pods. He creates deployments

- Can't replicate db practice via deployment
  because db has state of data. Clones might not share the same data.

Stateful set is for stateful apps such as db
(not easy to deploy, so host data outside)

MINIKUBE AND KUBECTL

Minikube: one node cluster with master and worker processes work in one node in a virtualbox for local testing

Kubectl: K8s cli