kubernetes

Why kubernetes ?

Container Clusters

- What if we have 10s, 100s, 1000s of running containers on multiple VMs?
- How to deploy, scale, restart, manage all of these containers?
- What problems do they solve?
  - Management
    - Metrics
    - Health checks
    - Security
  - Abstraction of hardware
  - Networking
  - Scheduling
  - Scaling
  - Deployment
    - Rollbacks
    - Zero-downtime / blue-green
  - Service discovery

A Brief Kubernetes History

- “K8s”
- Evolved out of Borg (Google’s internal container cluster)
- Open sourced ~2014
- Grew in popularity, open source velocity increased
- Now the most popular container cluster (most cloud platforms have some sort of managed K8s offering)
- Features added regularly and frequently
- Cloud Native / CNCF - Kubernetes, Prometheus, Fluentd

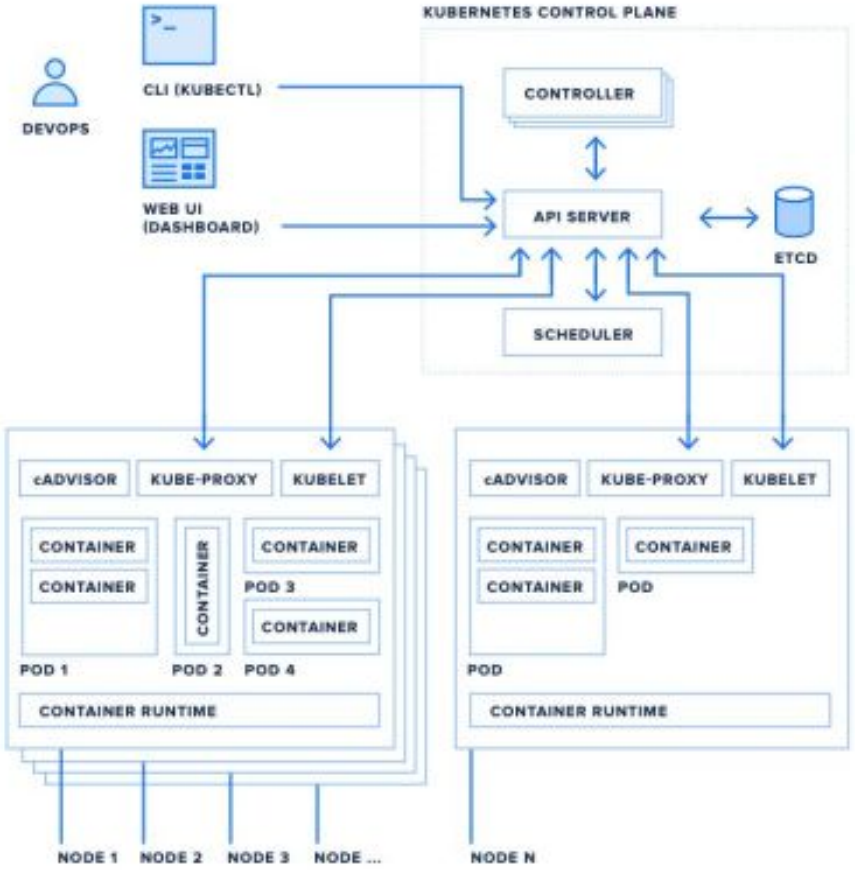
Kubernetes Architecture

- Client side :- CLI(KUBECTL)

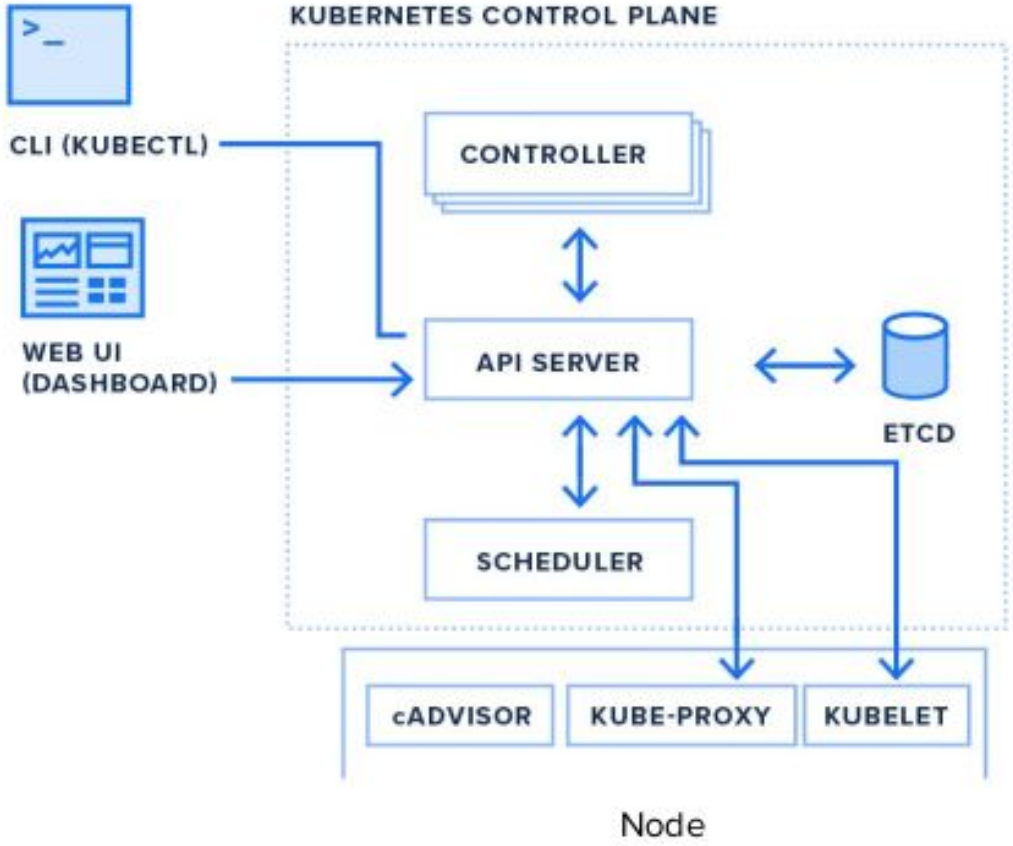
  WEB UI(DASHBOARD)

- Kubernetes Control plane or master node

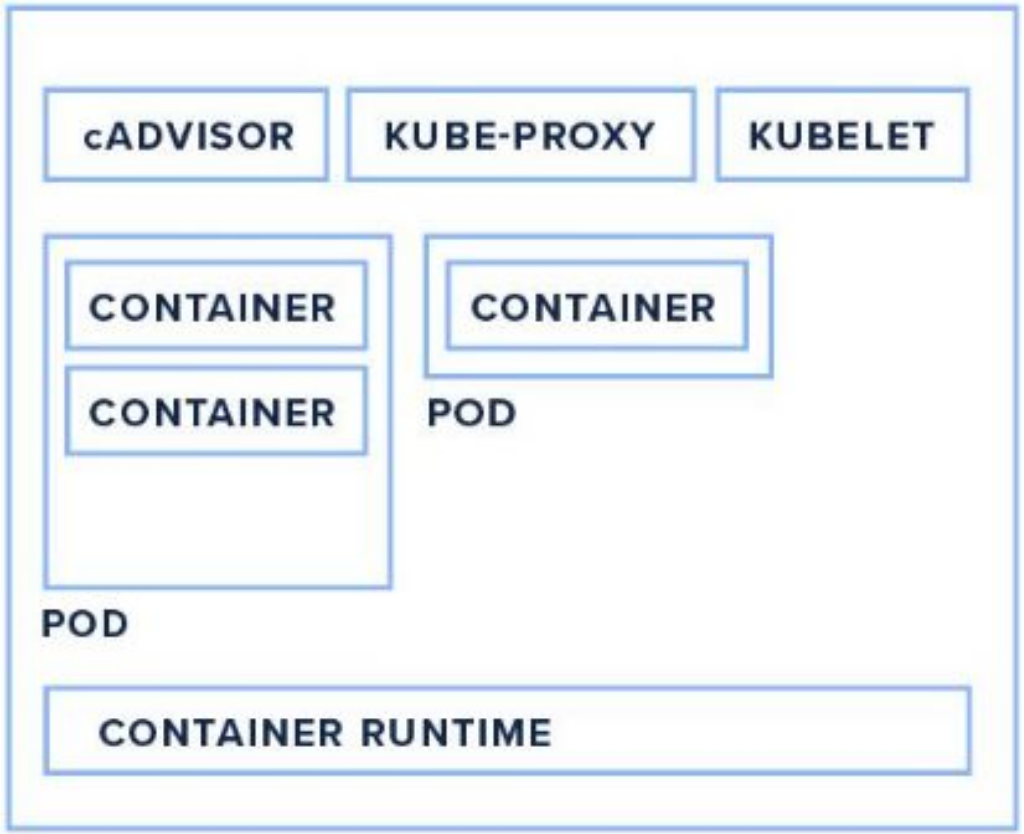
- Minions node or worker node



Kubernetes control plane



Kubernetes worker nodes



Kubernetes installation

Single Node

- Docker desktop
- Minikube

Custom kubernetes

- Kubeadm
- Kubespray

Cloud

- AWS - EKS
- Azure - AKS
- Google - GKE

Some K8s commands

Minikube start

Minikube stop

Kubectl version

Kubectl get

Kubectl apply

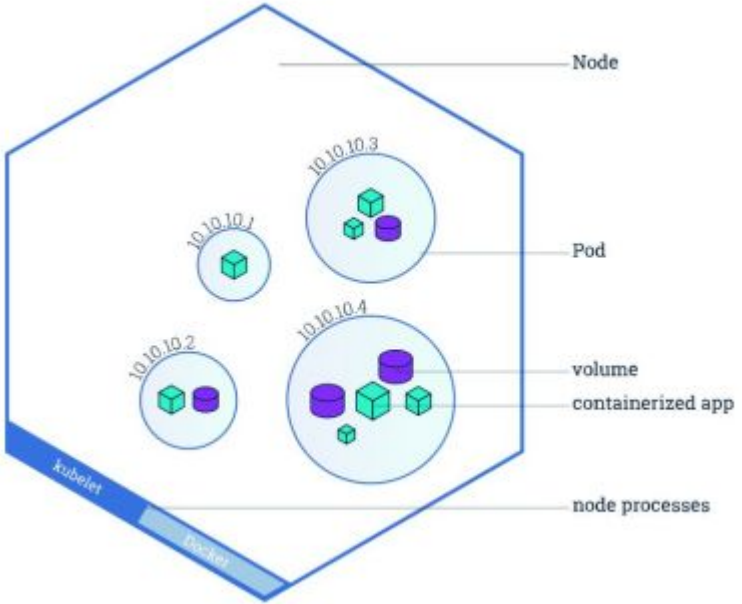
Kubectl create

Kubectl delete

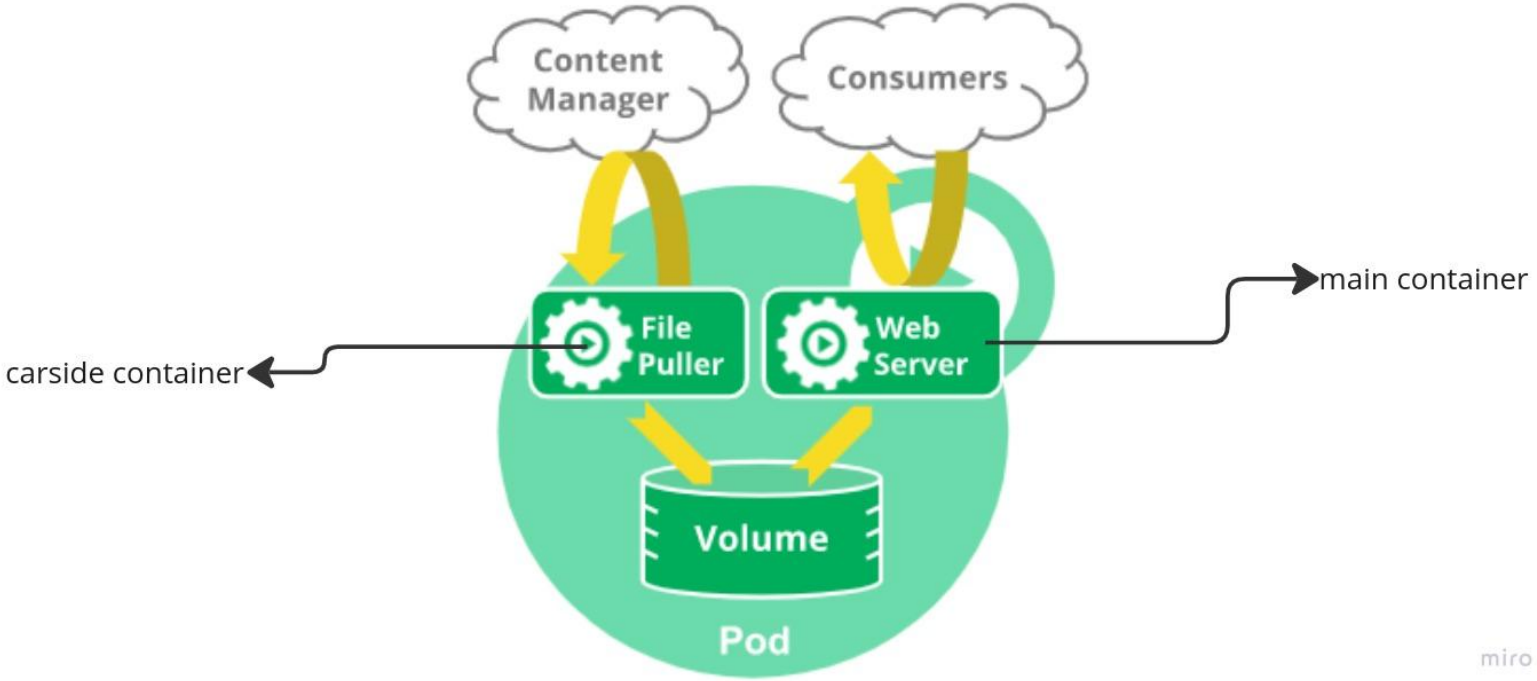
K8s components

PODS

- **Pods** are the smallest deployable units of computing that you can create and manage in Kubernetes.
- A **Pod** is a group of one or more containers.
- **Pods** that run a single container.
- **Pods** that run multiple containers that need to work together.
- **Pod** containers share resources
  - Storage
  - Network (localhost)
  - Always run on the same Node



Multiple containers in single pod



Create a container

Create a yaml file eg :- ak.yaml

```yaml
apiVersion: v1
kind: Pod
metadata:
 name: akpod1 # name of your any kind type
spec: # to create env
 containers:
 - name: akc1
   image: aakashgaur57/apache_ka_code
   ports:
   - containerPort: 80
```