

# 1. Overview

27 February 2022 14:47

- [GitHub - cloudacademy/intro-to-k8s: Assets used for the production of the Introduction to Kubernetes course on Cloud Academy](#)
- [GitHub - kubernetes-sigs/metrics-server: Scalable and efficient source of container resource metrics for Kubernetes built-in autoscaling pipelines.](#)

cluster → more than one node / compute / system → come together on a network to perform task

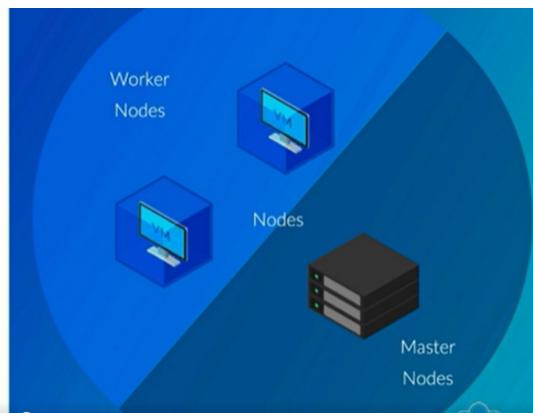
In k8 we can have clusters from cloud / legacy (vpc)

- controlled | Manged (version is depending on the provider)  
we control but time consuming  
AKS → azure  
EKS → aws

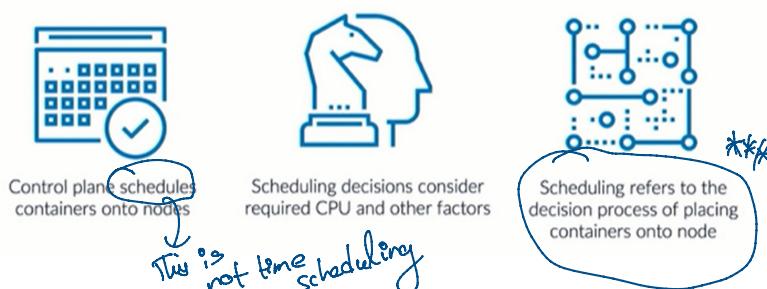
other options similar to k8 is Docker Swarm mode / EKS / DO / GKE

## Kubernetes Architecture

- Cluster refers to all of the machines collectively and can be thought of as the entire running system
- Nodes are the machines in the cluster
- Nodes are categorized as workers or masters
- Worker nodes include software to run containers managed by the Kubernetes control plane
- Master nodes run the control plane
- The control plane is a set of APIs and software that Kubernetes users interact with
- The APIs and software are referred



## Scheduling

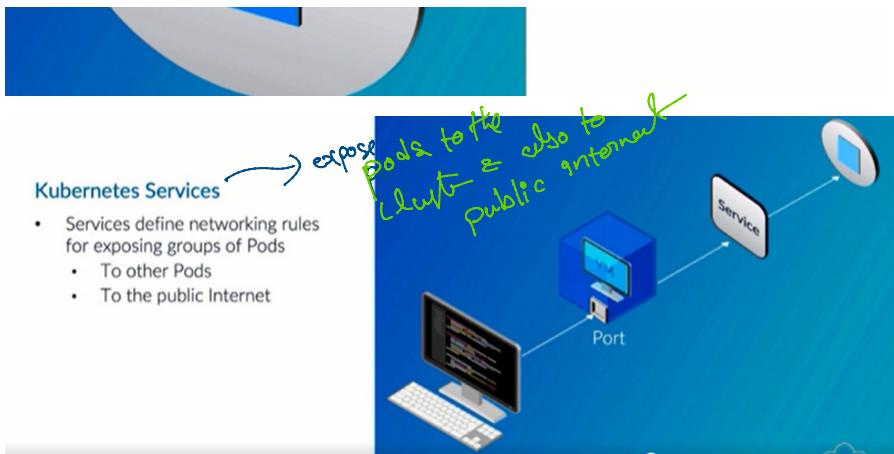


Pods → Group of containers (one or more)  
↳ all containers in pod run on the same node.



## Kubernetes Pods

- Groups of containers
- Pods are the smallest building block in Kubernetes



**Kubernetes Deployments** → control rollback & rollout of pods.

- Manage deploying configuration changes to running Pods
- Horizontal scaling

→ Kubernetes API Server



 The API Server is a master component that acts as the frontend for the cluster

## Interacting with Kubernetes: Method 1

### REST API



It is possible but not common to work directly with the API server



You might need to if there is no Kubernetes client library for your programming language

## Interacting with Kubernetes: Method 2

### Client Libraries



Handles authenticating and managing individual REST API requests and responses



Kubernetes maintains official client libraries



Community-maintained libraries when no official library exists

## Interacting with Kubernetes: Method 3

kubectl



Issue high-level commands that are translated into REST API calls



Works with local and remote clusters

kubectl

Most commonly used method is this one.



Kubernetes success correlates with kubectl skill



Manages all different types of Kubernetes resources, and provides debugging and introspection features



kubectl commands follow an easy to understand pattern



### Example kubectl commands

- kubectl create to create resources (Pods, Services, etc.)
- kubectl delete to delete resources
- kubectl get to get list of resources of a given type
  - kubectl get pods
- kubectl describe to print detailed info about a resource(s)
  - kubectl describe pod server
- kubectl logs to print container logs

→ commands can be used directly (or)  
you can also refer to a file called manifest  
where all resources names are declared

## Interacting with Kubernetes: Method 4

Web Dashboard

→ optional  
not all clusters will have this