

# Kubernetes

Nived Varma

Microsoft Certified Trainer

Expert Azure Architect

# Kubernetes Fundamentals

**Pod, ReplicaSet, Deployment & Service**



# Kubernetes - Fundamentals

k8s Fundamentals

Pod

A POD is a single instance of an Application.  
A POD is the smallest object, that you can create in Kubernetes.

ReplicaSet

A **ReplicaSet** will maintain a stable **set of replica Pods** running at any given time.  
In short, it is often used to guarantee the availability of a specified number of identical Pods

Deployment

A **Deployment** runs multiple **replicas of your application** and automatically **replaces any instances that fail or become unresponsive**.  
**Rollout & rollback changes to applications**. Deployments are well-suited for stateless applications.

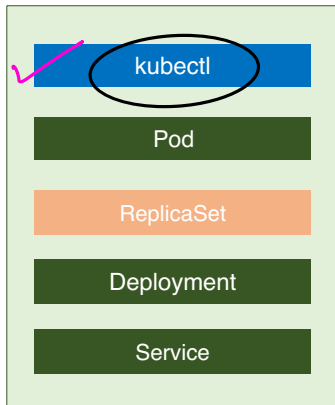
Service

A service is an **abstraction for pods**, providing a stable, so-called virtual IP (VIP) address.  
In simple terms, **service sits Infront of a POD and acts as a load balancer**.

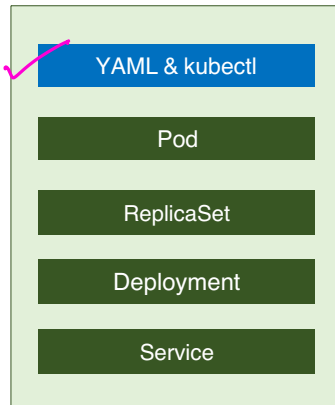
# Kubernetes - Imperative & Declarative

## Kubernetes Fundamentals

Imperative



Declarative



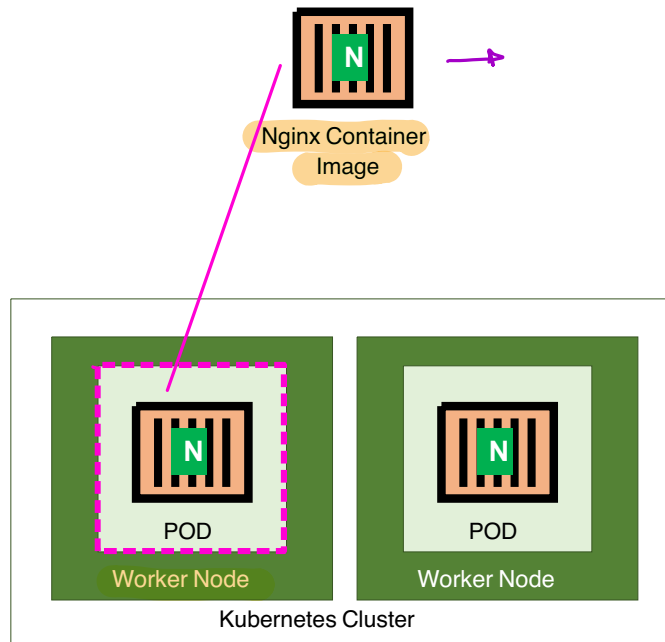
→ XAML

# Kubernetes POD



# Kubernetes - POD

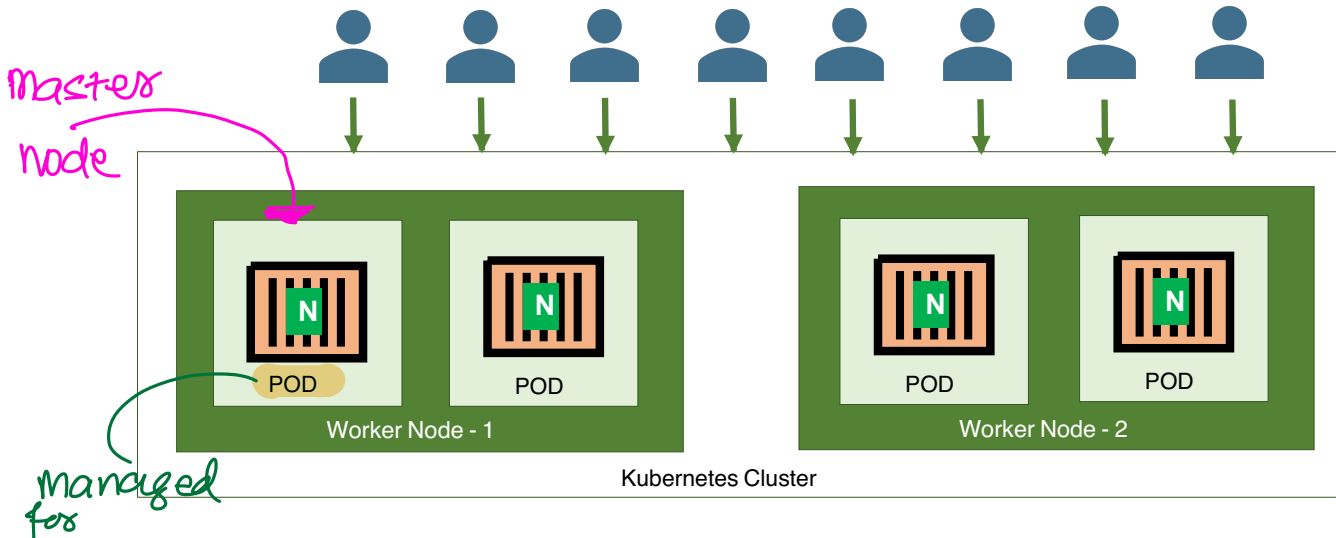
- With Kubernetes our core goal will be to **deploy our applications** in the form of **containers** on **worker nodes** in a k8s cluster.
- Kubernetes **does not** deploy containers directly on the **worker nodes**.
- Container is **encapsulated** in to a Kubernetes Object named **POD**.
- A POD is a **single instance** of an **application**.
- A POD is the **smallest object** that we can create in Kubernetes.

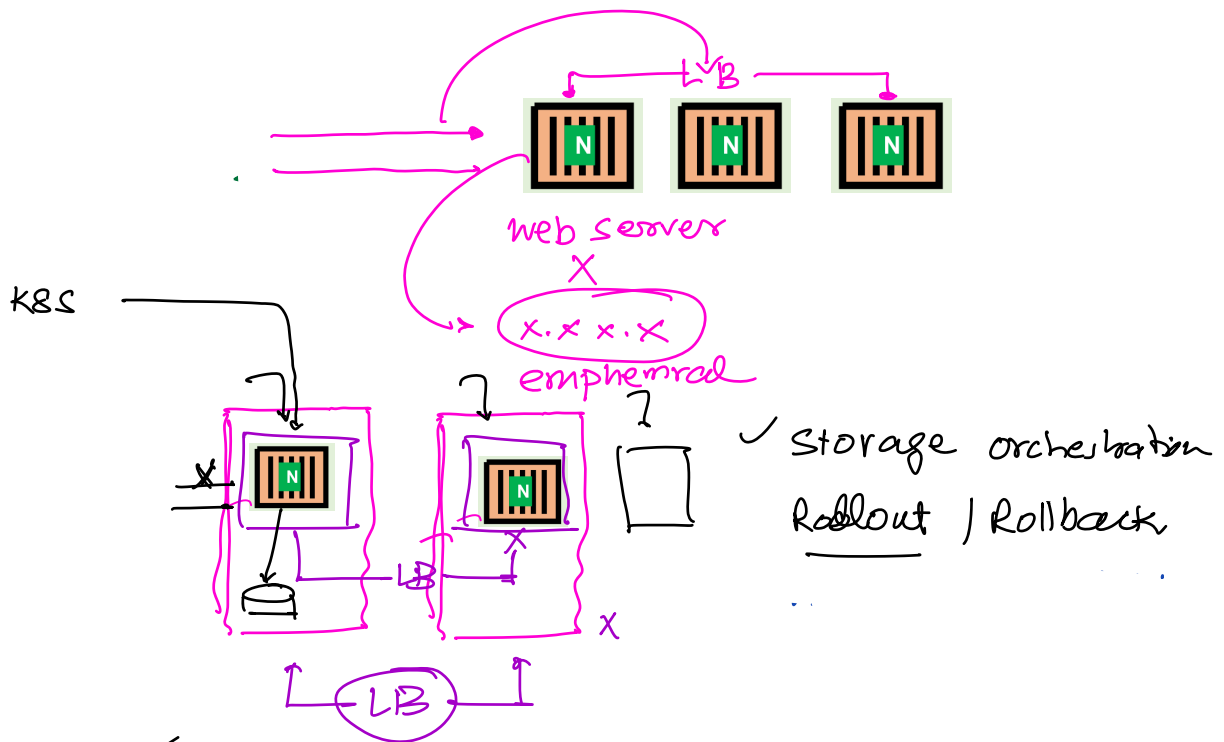


# Kubernetes - POD

- PODs generally have **one to one** relationship with **containers**.
- To scale up we **create** new **POD** and to scale down we **delete** the **POD**.

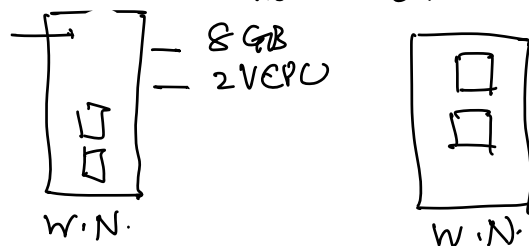
**K8S** - **orchestrator**





✓ Automated Bin Packing

How much CPU/RAM - containers.

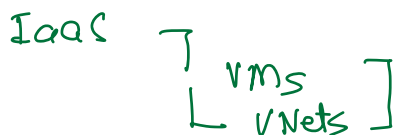


✓ Self-Healing



✓ Secret & Config. management

Health State



K8S is not all inclusive  
PaaS → combination of  
IaaS + PaaS

↳ Master Node

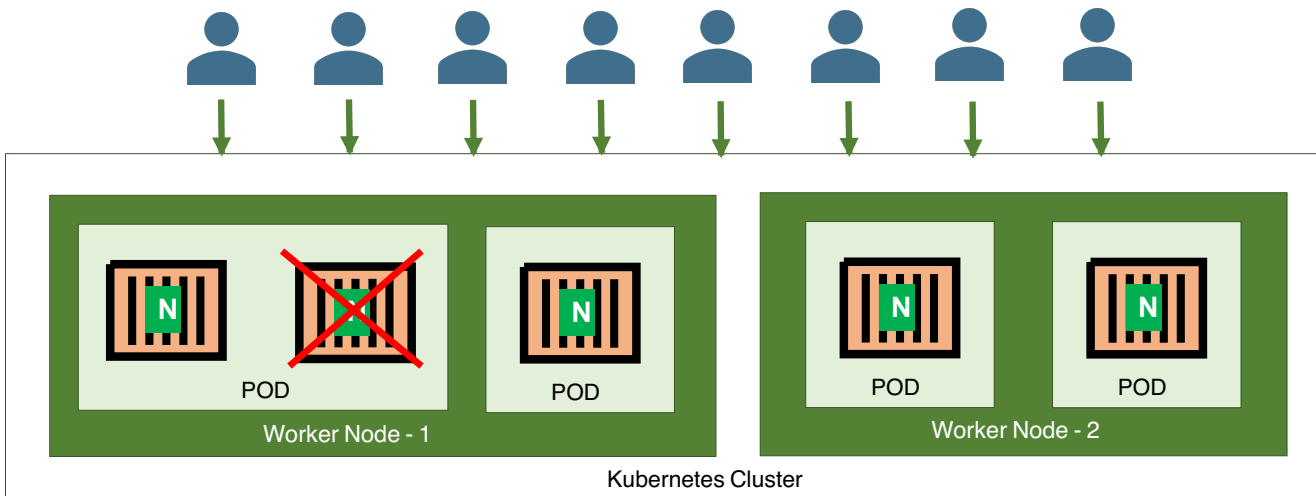


K8S is not?

- ✓ Does not limit the types of app<sup>n</sup> supported
- ✓ Does not deploy source code. & does not build app<sup>n</sup>.
- ✓ Does not provide application-level services.
- ✓ Does not dictate logging, monitoring, alerting systems.

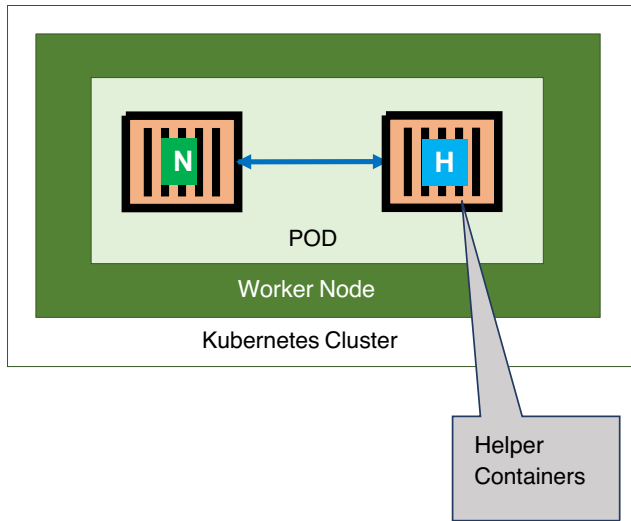
# Kubernetes – PODs

- We **cannot have** multiple containers of **same kind** in a **single POD**.
- **Example:** Two NGINX containers in single POD serving same purpose is **not recommended**.



# Kubernetes – Multi-Container Pods

- We can have multiple containers in a single POD, provided **they are not of same kind**.
- **Helper Containers (Side-car)**
  - **Data Pullers**: Pull data required by Main Container
  - **Data pushers**: Push data by collecting from main container (logs)
  - **Proxies**: Writes static data to html files using Helper container and Reads using Main Container.
- **Communication**
  - The two containers can easily communicate with each other easily as they share same **network space**.
  - They can also easily share **same storage space**.
- Multi-Container Pods is a **rare use-case** and we will try to focus on core fundamentals.

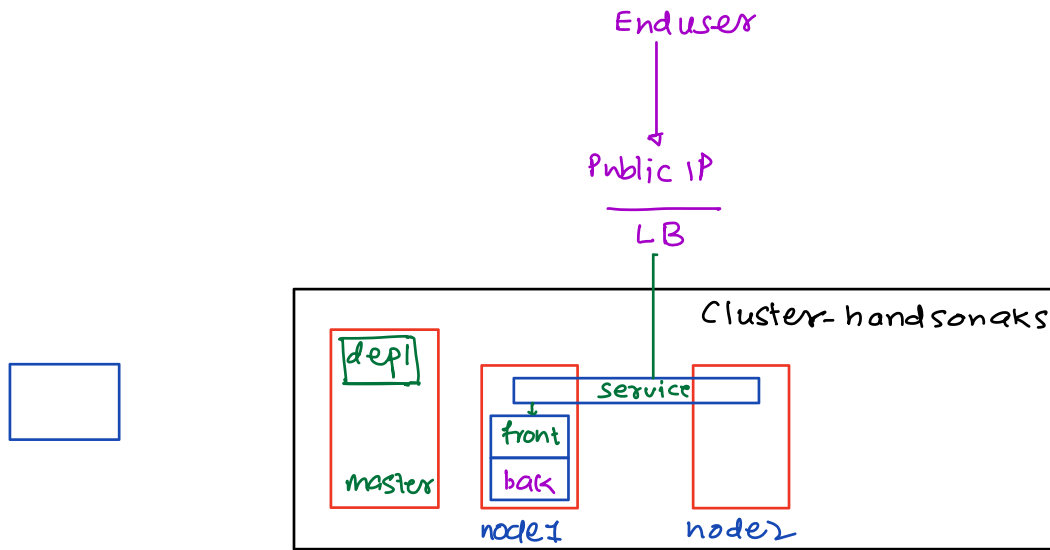


# Kubernetes

## PODs

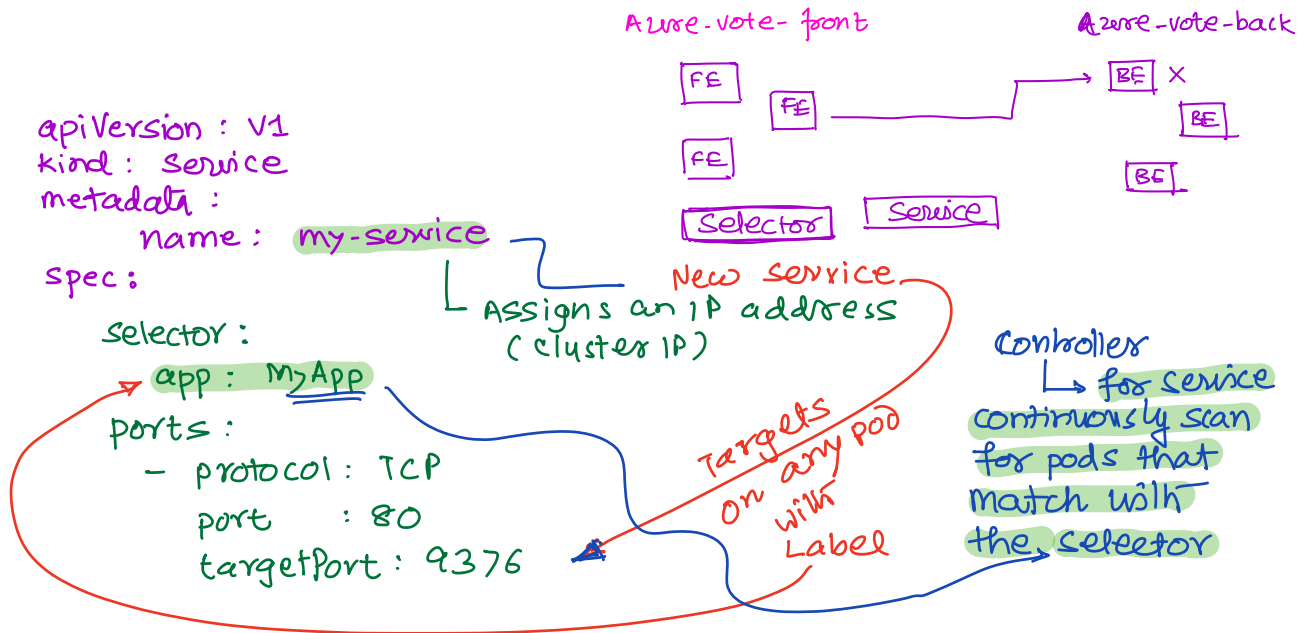
## Demo

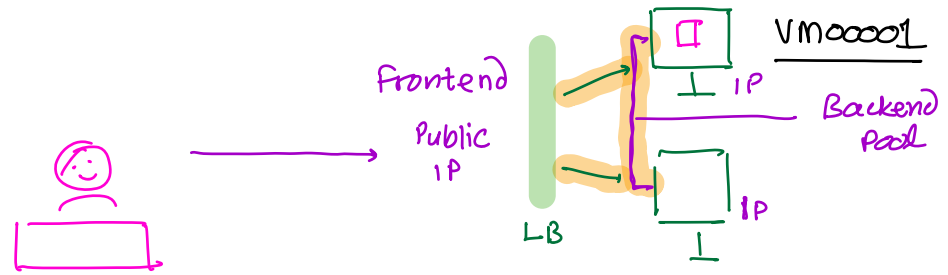




Service → Network service to expose an application running on a set of pods.

K8S → Gives pods their own IP addresses and a single DNS name for set of pods.





K8S → assumes that pods can communicate with other pods regardless of which host they land.

K8S → Private IP — (cluster IP)