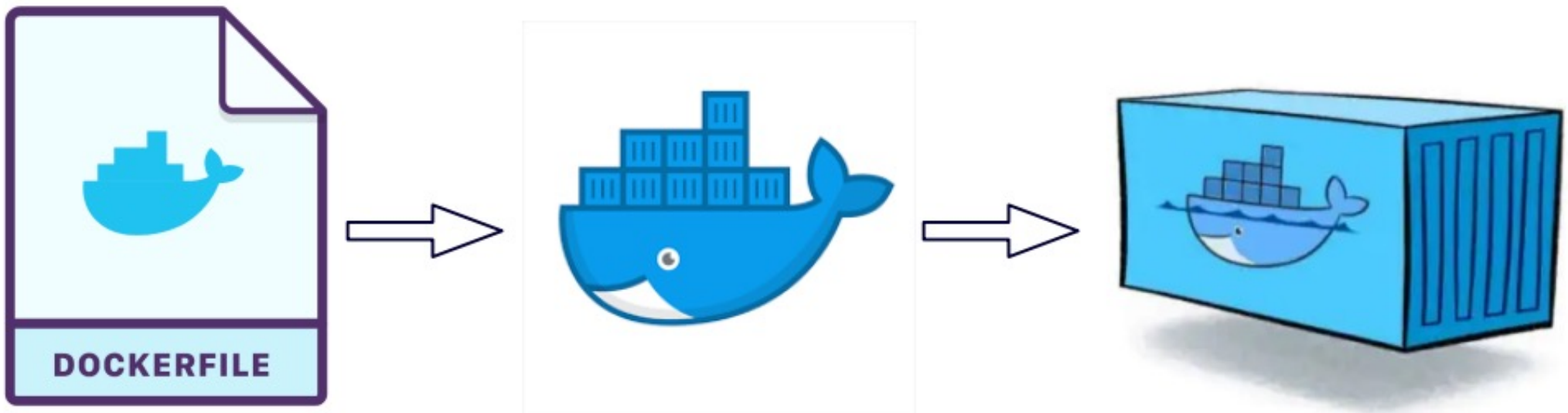
kubernetes

Present By Amit Ganvir

# What is Kubernetes

- Kubernetes also known as K8s
- Free to use
- Kubernetes is an open source project
- It is for automating deployment, scaling, and management of containerized applications
- cluster together groups of hosts running Linux® containers, and Kubernetes helps you easily and efficiently manage those clusters
- Kubernetes clusters can span hosts across on-premise, public, private, or hybrid clouds
- Kubernetes is the best orchestration tool
- Beside similarly we have RedHat Openshift, AWS-ECS/EKS and Google Kubernetes Engine(GKE)

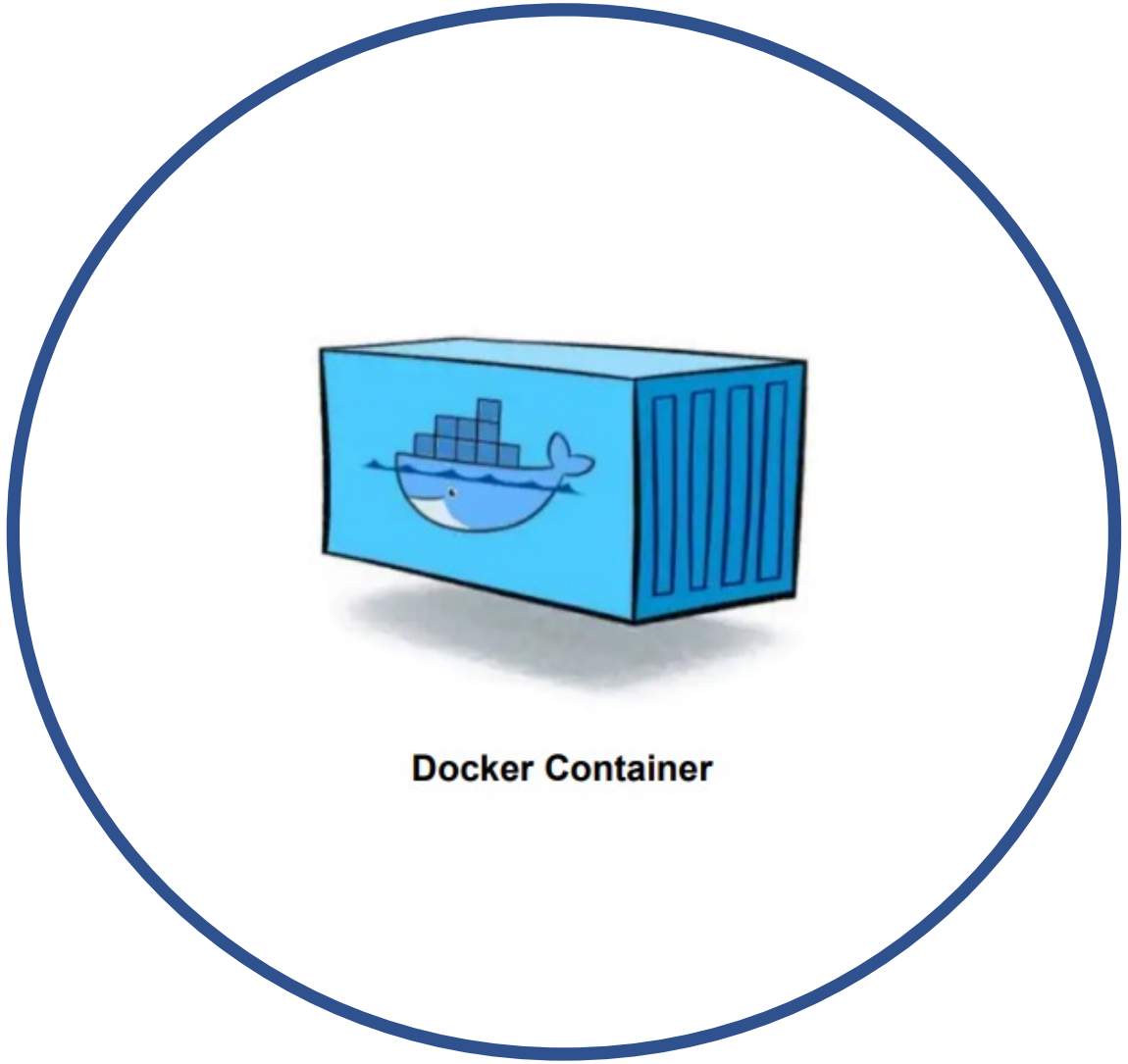



Docker file

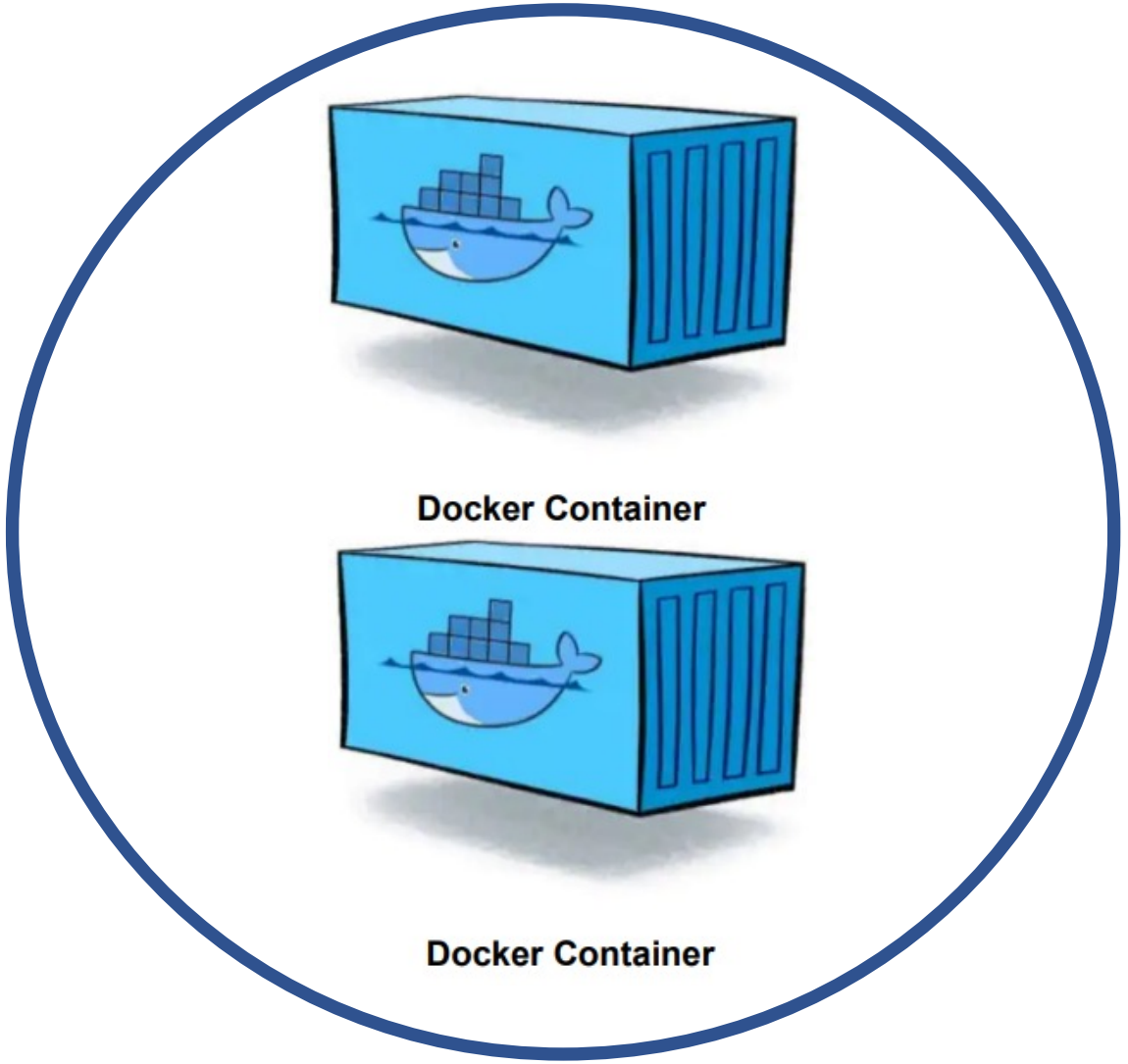
Docker Image

Docker Container

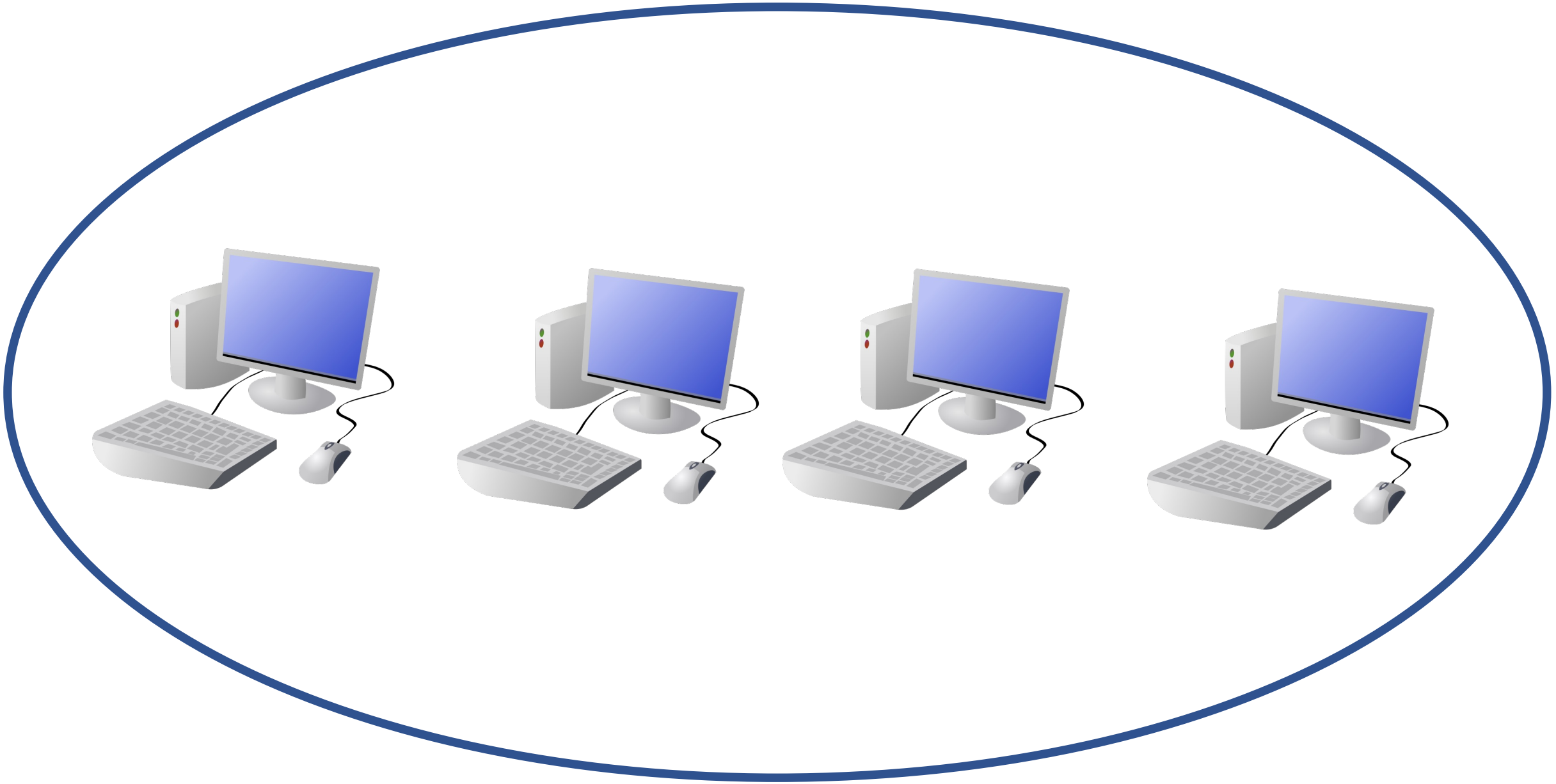
# POD




POD1

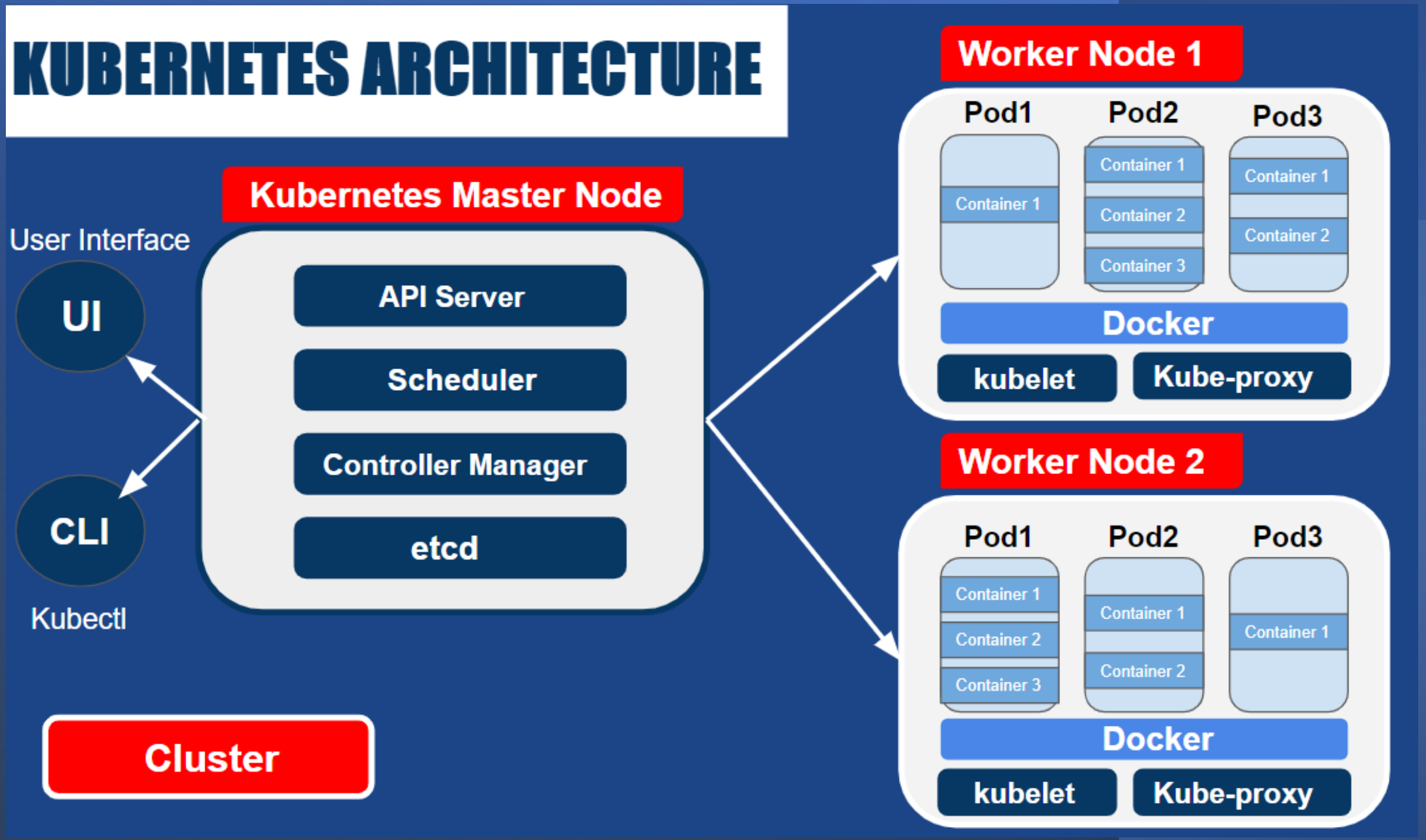



POD2

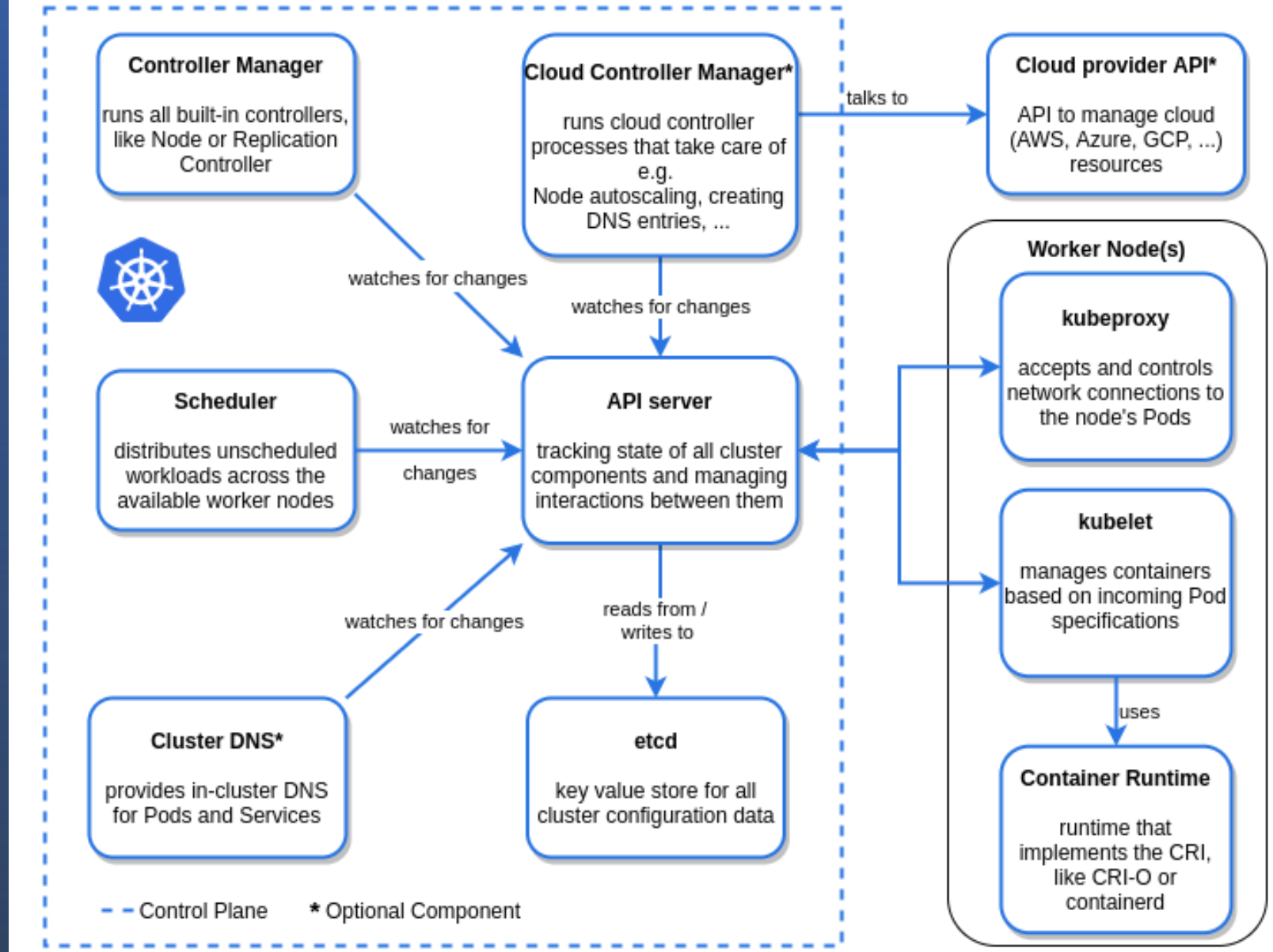


Cluster

# KUBERNETES ARCHITECTURE



# Kubernetes Architecture



# What is kubectl

It basically provides the CLI to run commands against the Kubernetes cluster with various ways to create and manage the Kubernetes component.

**What are the advantages of Kubernetes?**

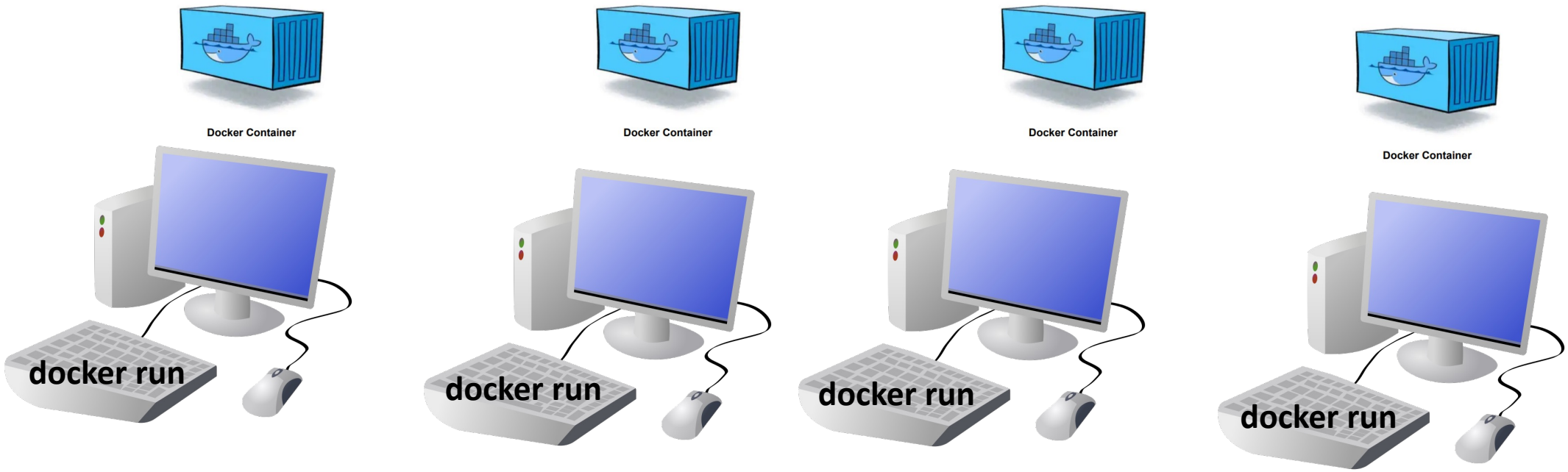
- Automated Scheduling
- Self-Healing Capabilities
- Automated Rollback and rollouts
- Horizontal Scaling and Load balancing

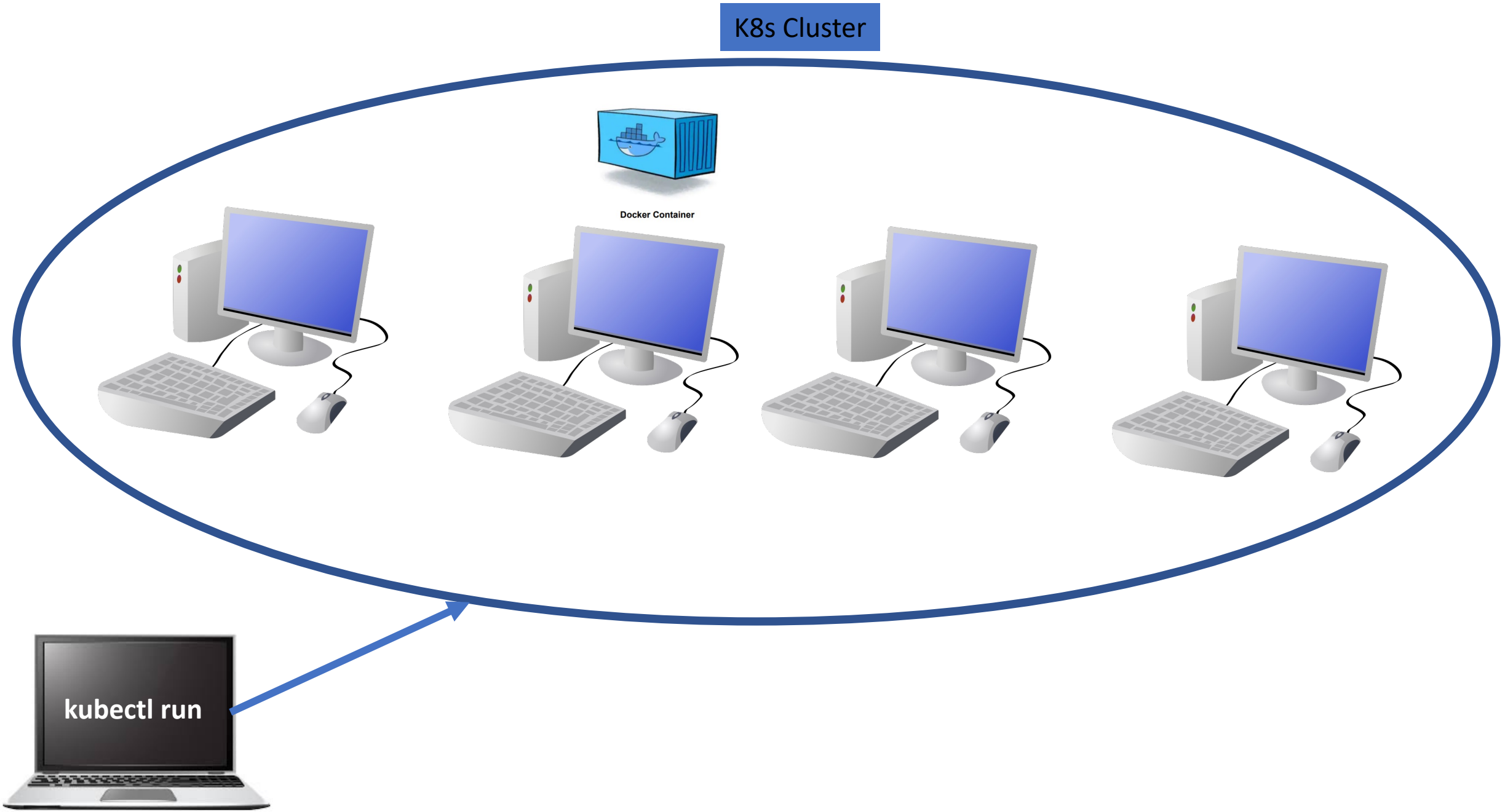
**What are the different types of services in Kubernetes?**

1. Cluster IP (which is Default service)
2. Node Port
3. Load Balancer
4. External Name (For FQDN/DNS Records name)

**Kubernetes Port range?**

- The default Loadbalancer/node port range for Kubernetes is 30000 – 32767


Docker Container
Docker Container
Docker Container
Docker Container
docker run
docker run
docker run
docker run


K8s Cluster
Docker Container
kubectl run

# Setup k8s cluster

- https://github.com/amitganvir23/devops-session/blob/master/k8s-1-21-setup.sh
- Master will Required at least 4 GB RAM, 2 Core CPU (t2.medium)
- Worker will Required at least 1 GB RAM, 1 Core CPU (t2.micro)

To check your nodes

```
root@ip-172-31-24-184:~# kubectl get nodes
NAME                 STATUS   ROLES                  AGE   VERSION
ip-172-31-24-184     Ready    control-plane,master   22m   v1.21.1
ip-172-31-83-28      Ready    <none>                 20m   v1.21.1
root@ip-172-31-24-184:~#
```

To check all the namespace (project)

```
root@ip-172-31-24-184:~# kubectl get ns
NAME                 STATUS   AGE
default              Active   25m
kube-node-lease      Active   25m
kube-public          Active   25m
kube-system          Active   25m
root@ip-172-31-24-184:~#
```

TO run Nginx container on cluster

```
root@ip-172-31-24-184:~# kubectl run nginx --image=nginx
```

Check Pods

```
root@ip-172-31-24-184:~# kubectl get pods
```

# How to access k8s cluster from Clientside

Copy ‘admin.conf’ contents from Master Node

```
root@ip-172-31-24-184:~# cat /etc/kubernetes/admin.conf
```

Paste ’admin.conf’ contents on Client machine

```
root@ip-172-31-83-28:~# mkdir  ~/.kube
root@ip-172-31-83-28:~# vim  ~/.kube/config
```

Verify k8s cluster access

```
root@ip-172-31-83-28:~# kubectl get nodes
NAME                 STATUS     ROLES                  AGE     VERSION
ip-172-31-24-184     Ready      control-plane,master   22m     v1.21.1
ip-172-31-83-28      Ready      <none>                 20m     v1.21.1
root@ip-172-31-24-184:~#
```

# Creating YAML File

A Kubernetes manifest (Definition) is a YAML file

| | |
|---|---|
| API version of k8s to create Objects (1) | apiVersion: |
| Type of Objects (2) | kind: |
| Data of the Objects follow with (3) | metadata: |
| Specification to add additional Information (4) | spec: |

# Creating YAML File

A Kubernetes manifest (Definition) is a YAML file

API version of k8s to create Objects (1)
Type of Objects (2)
Data of the Objects follow with (3)

Specification to add additional Information (4)

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    app: nginx
spec:
  containers:
  - image: nginx
    name: nginx
```

# Creating YAML File

Deployment manifest (Definition YAML file

API version of k8s to create Objects (1)
Type of Objects (2)
Data of the Objects follow with (3)
Specification to add additional Information (4)
Template (a)
Replicas for num of pods to launch (b)
Selector (c)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: nginx
  name: nginx
spec:
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - image: nginx
        name: nginx
  replicas: 1
  selector:
    matchLabels:
      app: nginx
```

# Different types of Services in k8s

1. Cluster IP (which is Default service)
2. Node Port (To access app outside of the cluster)
3. Load Balancer (To access app outside of the cluster)
4. External Name (For FQDN/DNS Records name)

# Creating YAML File

Service manifest (Definition YAML file

| | |
|---|---|
| API version of k8s to create Objects (1) | apiVersion: v1 |
| Type of Objects (2) | kind: Service |
| Data of the Objects follow with (3) | metadata: |
| | name: nginx-svc |
| | labels: |
| | app: nginx-svc |
| Specification to add additional Information (4) | spec: |
| Type of Service (a) | type: LoadBalancer |
| Ports (b) | ports: |
| | - port: 80 |
| | name: nginx |
| | targetPort: 80 |
| | nodePort: 30001 |
| Selector (c) | selector: |
| | app: nginx |

# K8s Dashboar

Copy ‘admin.conf’ contents from Master Node

```
# kubectl apply -f https://raw.githubusercontent.com/kubernetes/dashboard/v2.7.0/aio/deploy/recommended.yaml
```

Replace ClusterIP to Nodeport

```
# kubectl -n kubernetes-dashboard edit svc kubernetes-dashboard
```

Check your Nodeport

```
# kubectl -n kubernetes-dashboard get svc
```

Default Admin-token for default namespace

```
# kubectl -n kube-system describe secret $(kubectl -n kube-system get secret | awk '/^deployment-controller-token-/{print $1}') | awk '$1=="token:"{print $2}'
```

Browser: https://<NodeIP>:<NodePort>

# Data Availabe in Github Repository

https://github.com/amitganvir23/devops-session

https://www.aquasec.com/cloud-native-academy/kubernetes-101/kubernetes-dashboard/

https://devopscube.com/setup-kubernetes-cluster-kubeadm

# Thank You!

Present By **Amit Ganvir**
amitganvir6@gmail.com