Different Parts in a Master Node of Kubernetes

ETCD


ETCD is a distributed reliable key-value store that is Simple, Secure & Fast

key-value store

Key	Value
Name	John Doe
Age	45
Location	New York
Salary	5000

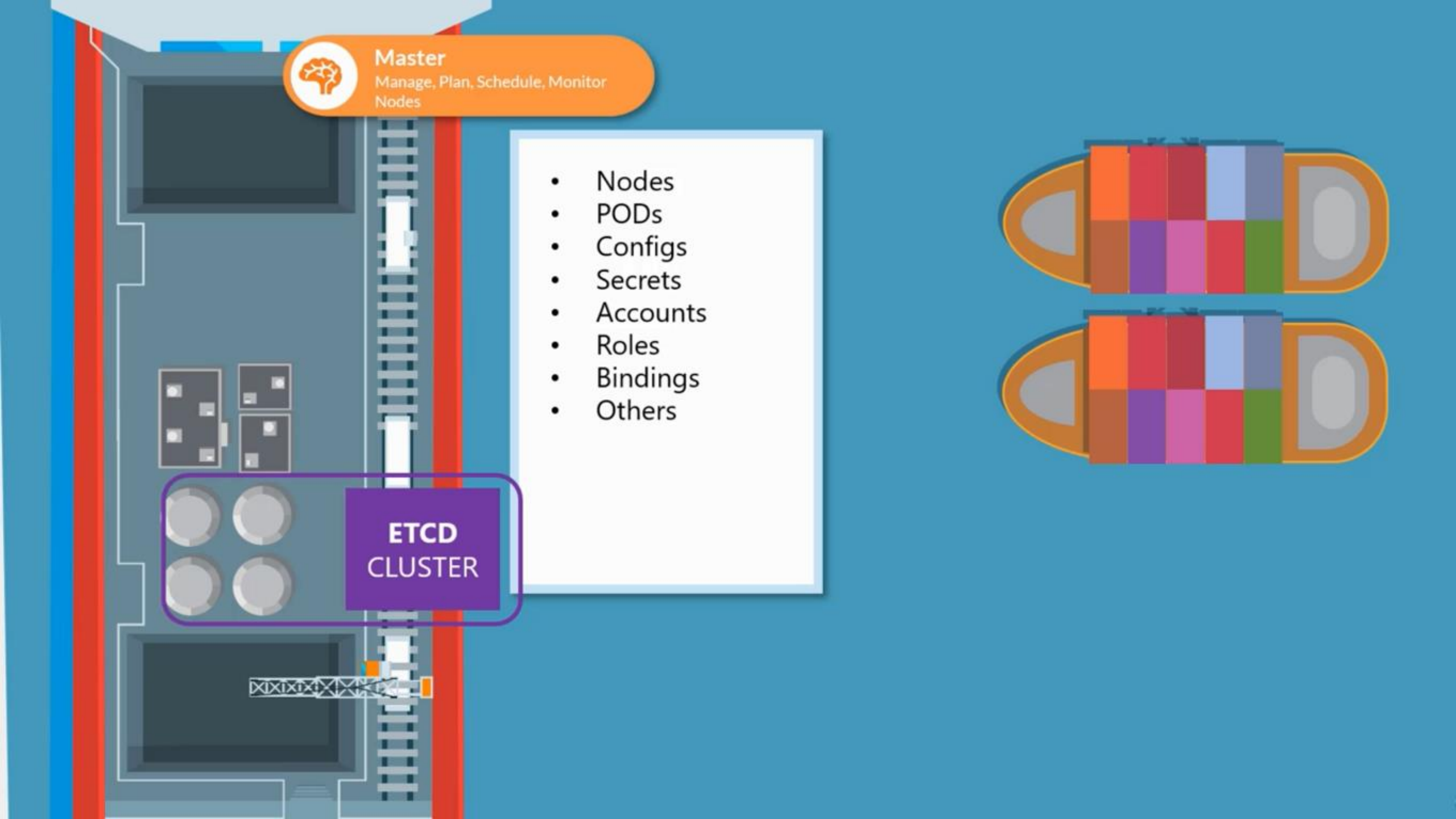
Put Name "John Doe"

Get Name

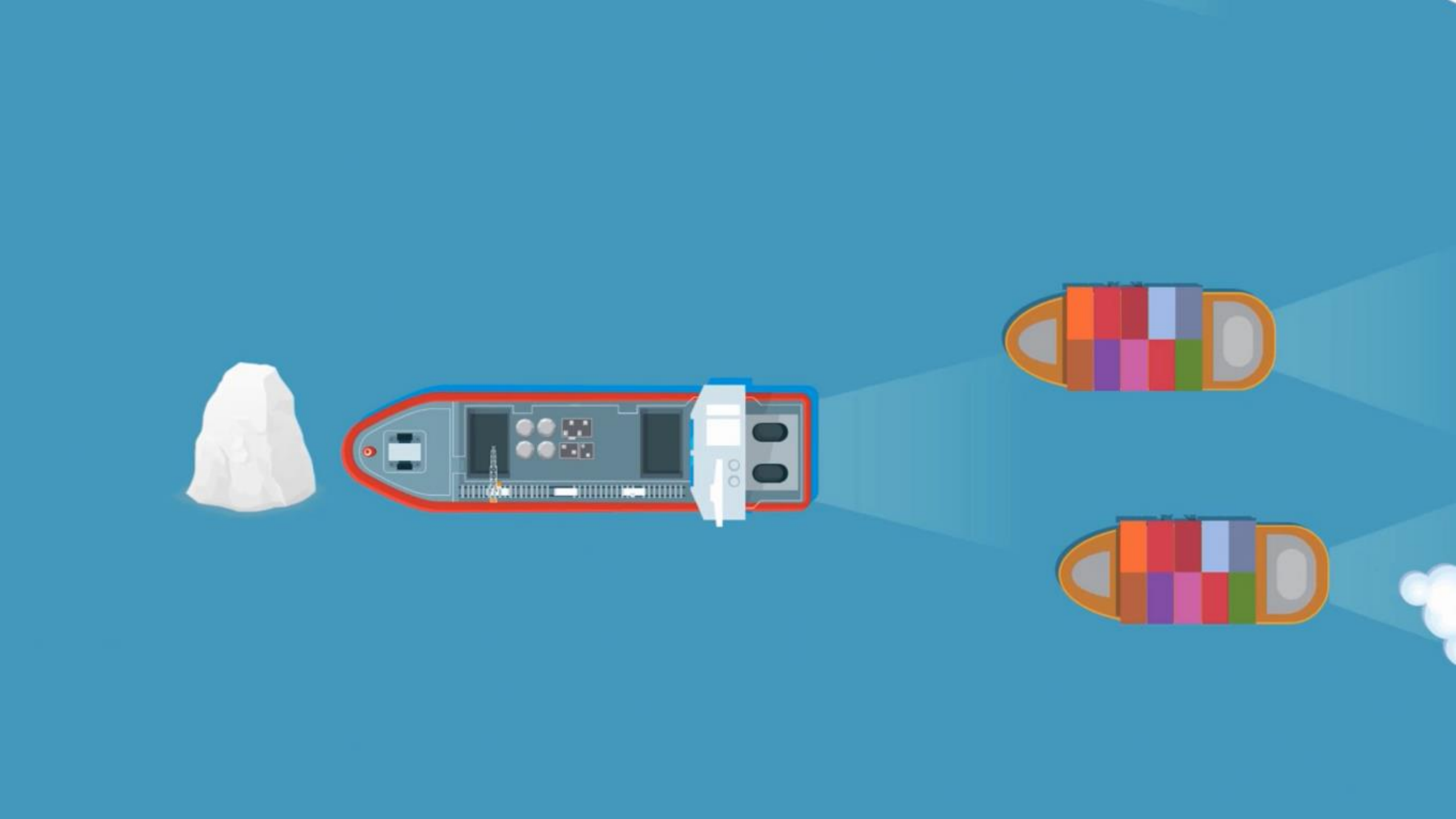
"John Doe"

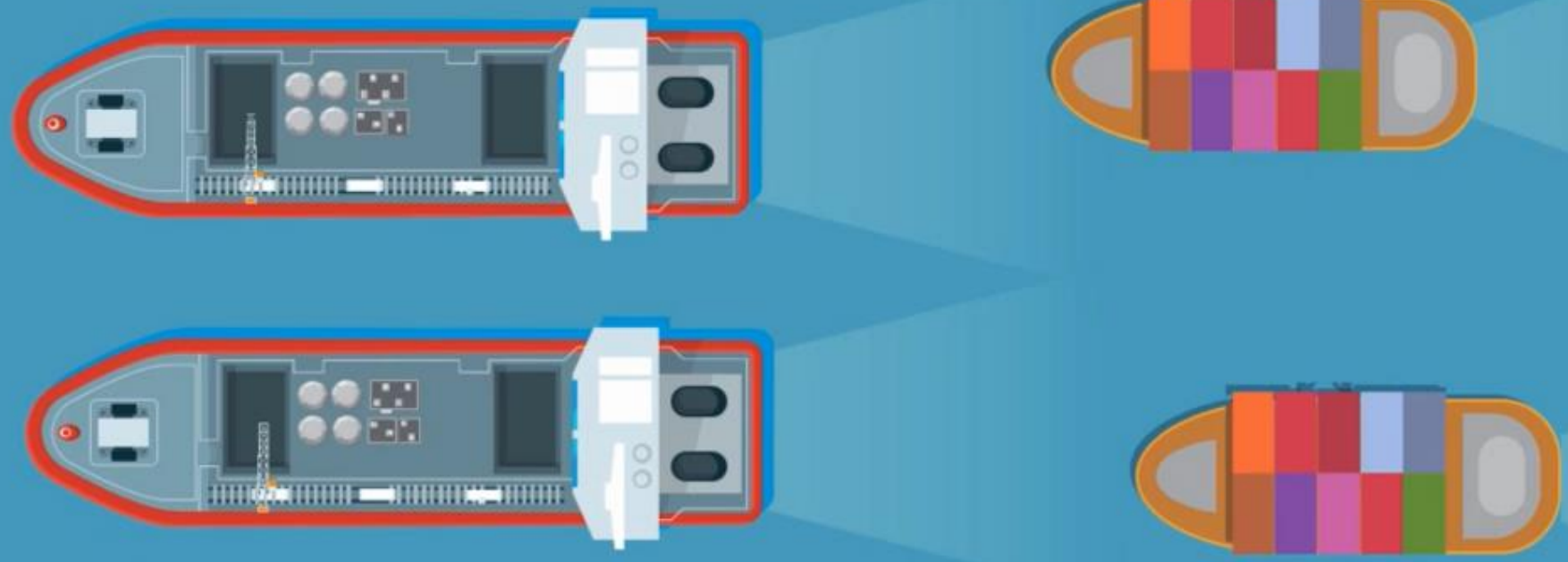
Tabular/Relational Databases

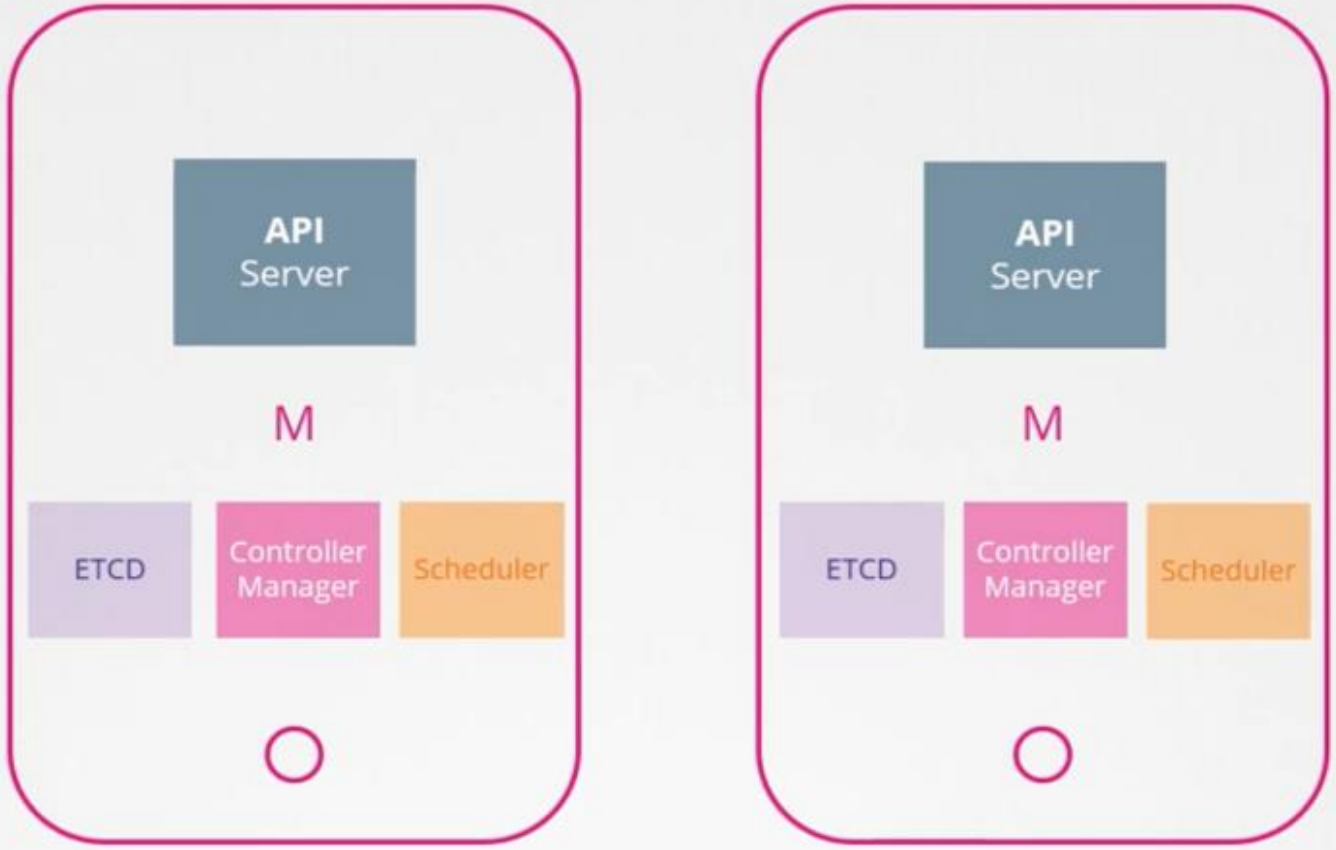
Name	Age	Location
John Doe	45	New York
Dave Smith	34	New York
Aryan Kumar	10	New York
Lauren Rob	13	Bangalore
Lily Oliver	15	Bangalore

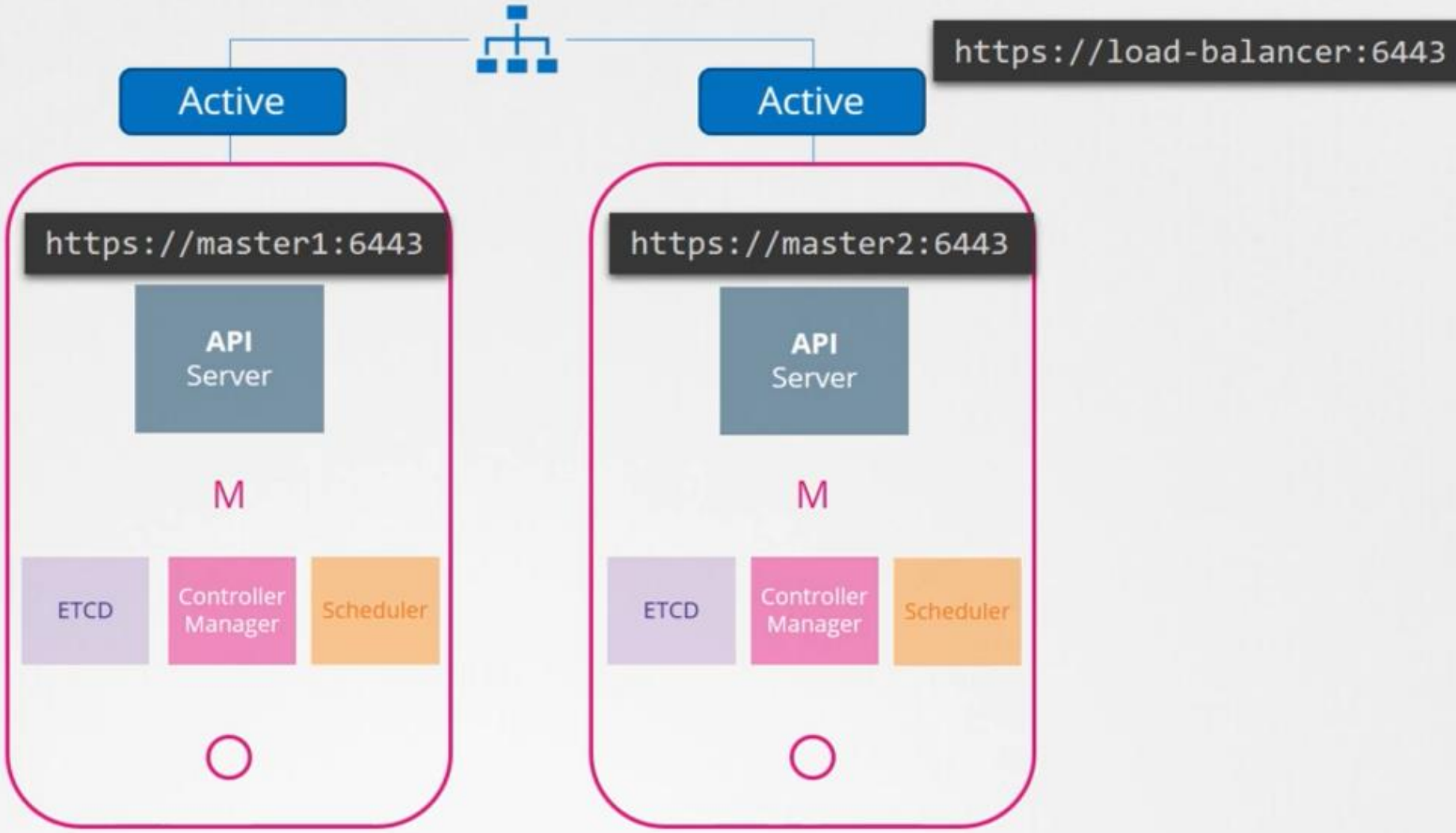

Master
Manage, Plan, Schedule, Monitor Nodes
ETCD
CLUSTER
Nodes
PODs
Configs
Secrets
Accounts
Roles
Bindings
Others

ETCD in HA Environment

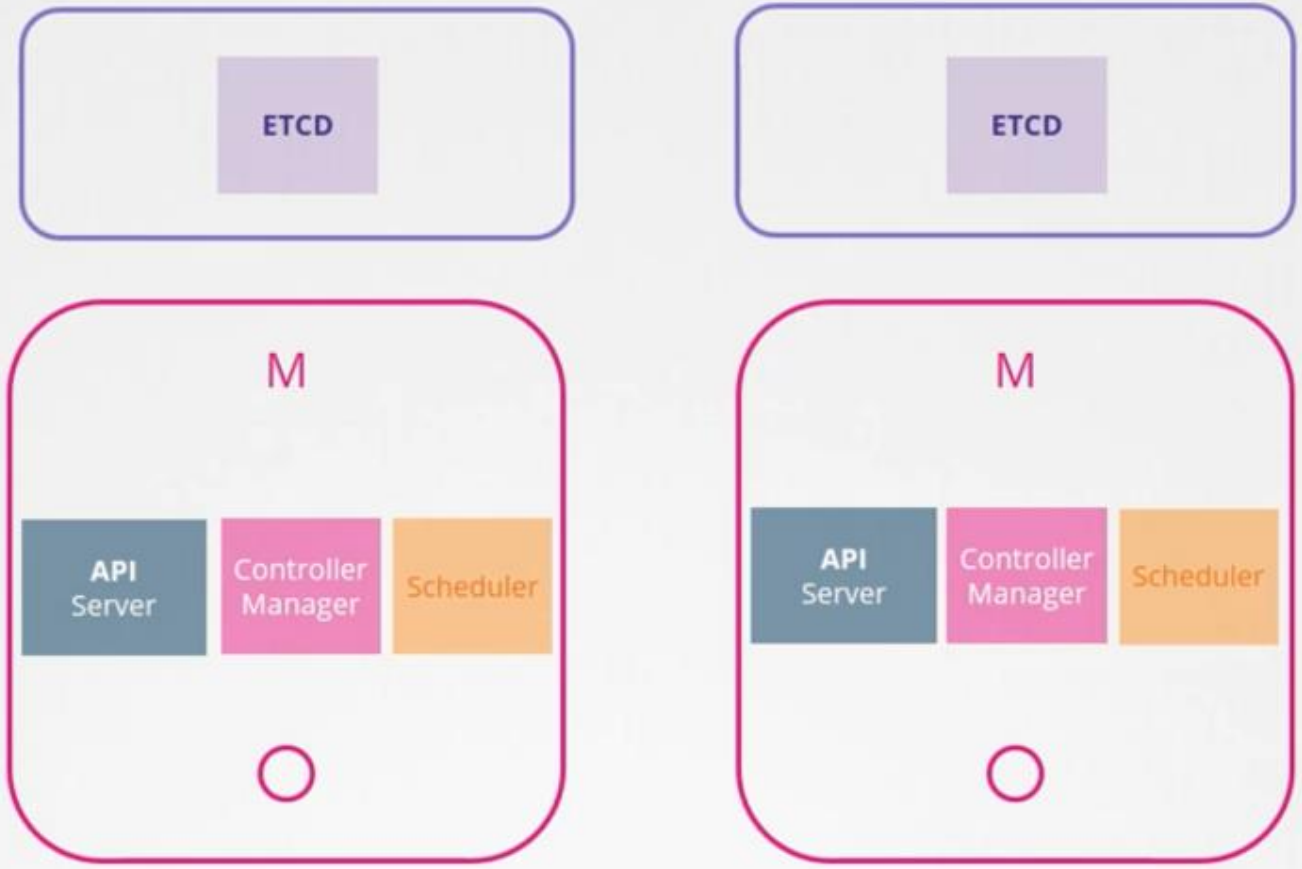





API
Server
M
ETCD
Controller
Manager
Scheduler
API
Server
M
ETCD
Controller
Manager
Scheduler

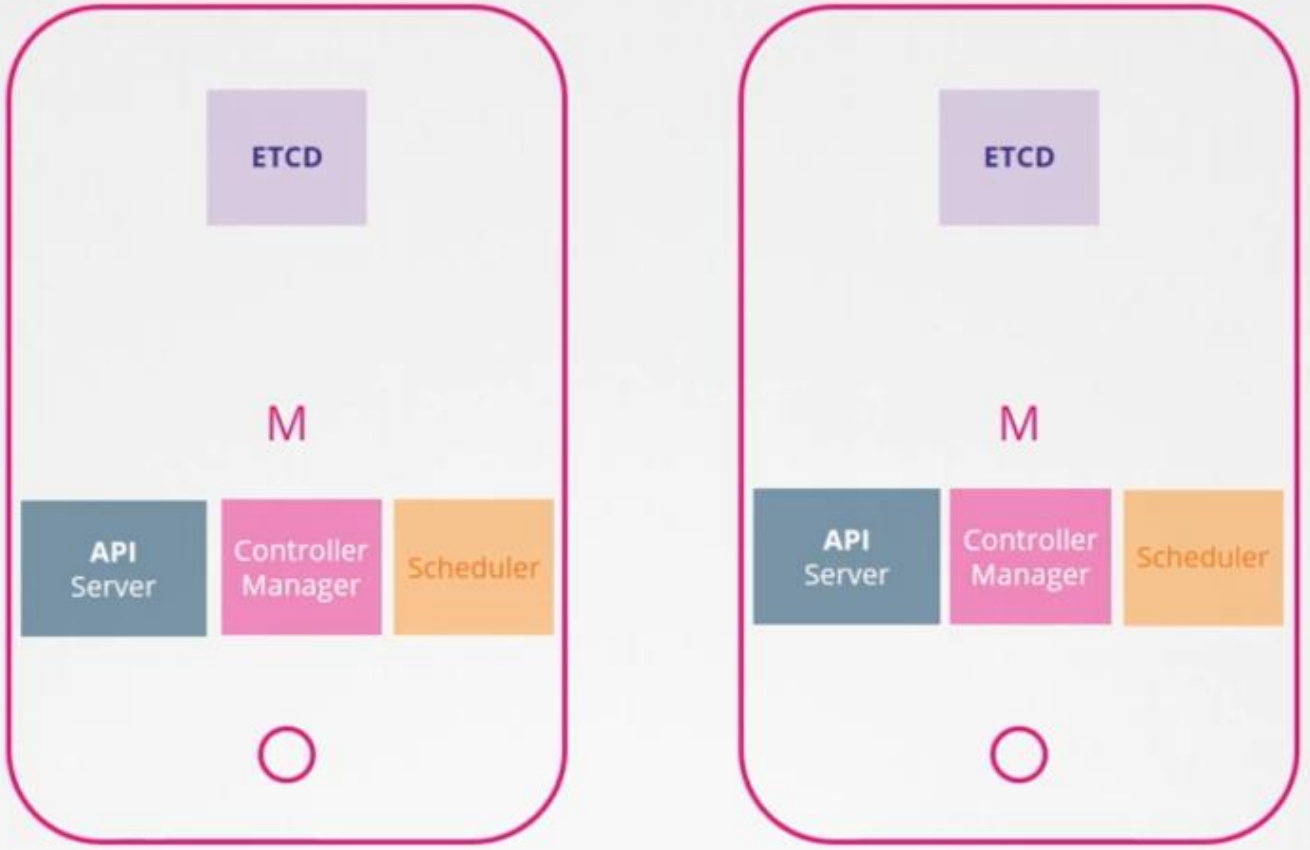

https://load-balancer:6443
Active
Active
https://master1:6443
https://master2:6443
API
Server
API
Server
M
M
ETCD
Controller
Manager
Scheduler
ETCD
Controller
Manager
Scheduler

External ETCD Topology



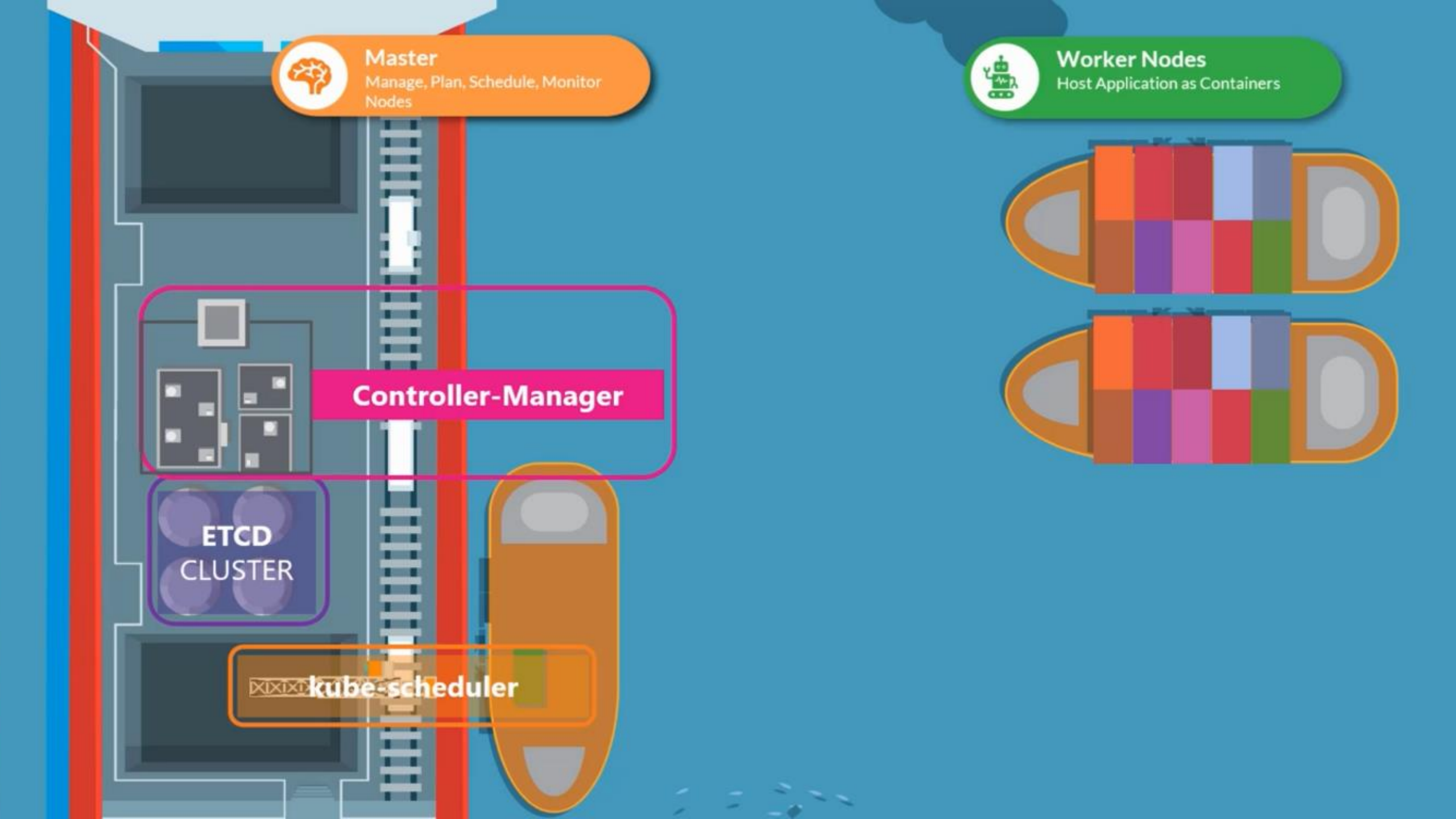
- ✓ Less Risky
- ❖ Harder to Setup
- ❖ More Servers

Stacked Topology



- ✓ Easier to setup
- ✓ Easier to manage
- ✓ Fewer Servers
- ❖ Risk during failures

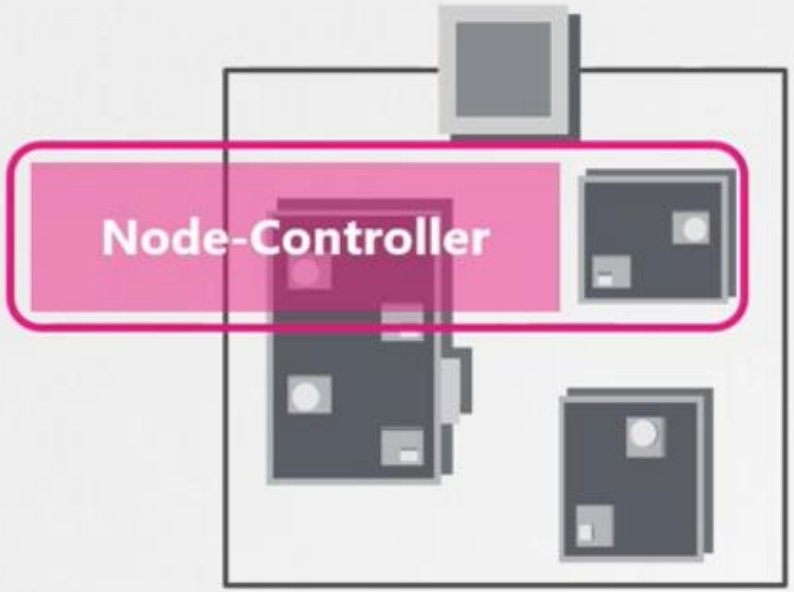
Controller

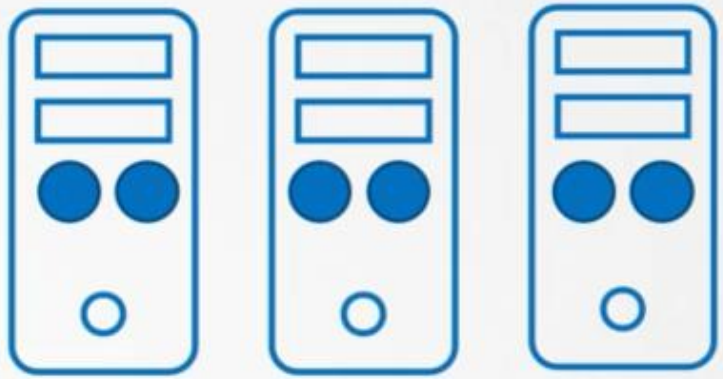

Master
Manage, Plan, Schedule, Monitor Nodes
Worker Nodes
Host Application as Containers
Controller-Manager
ETCD
CLUSTER
kube-scheduler

Controller

Watch Status

Remediate Situation

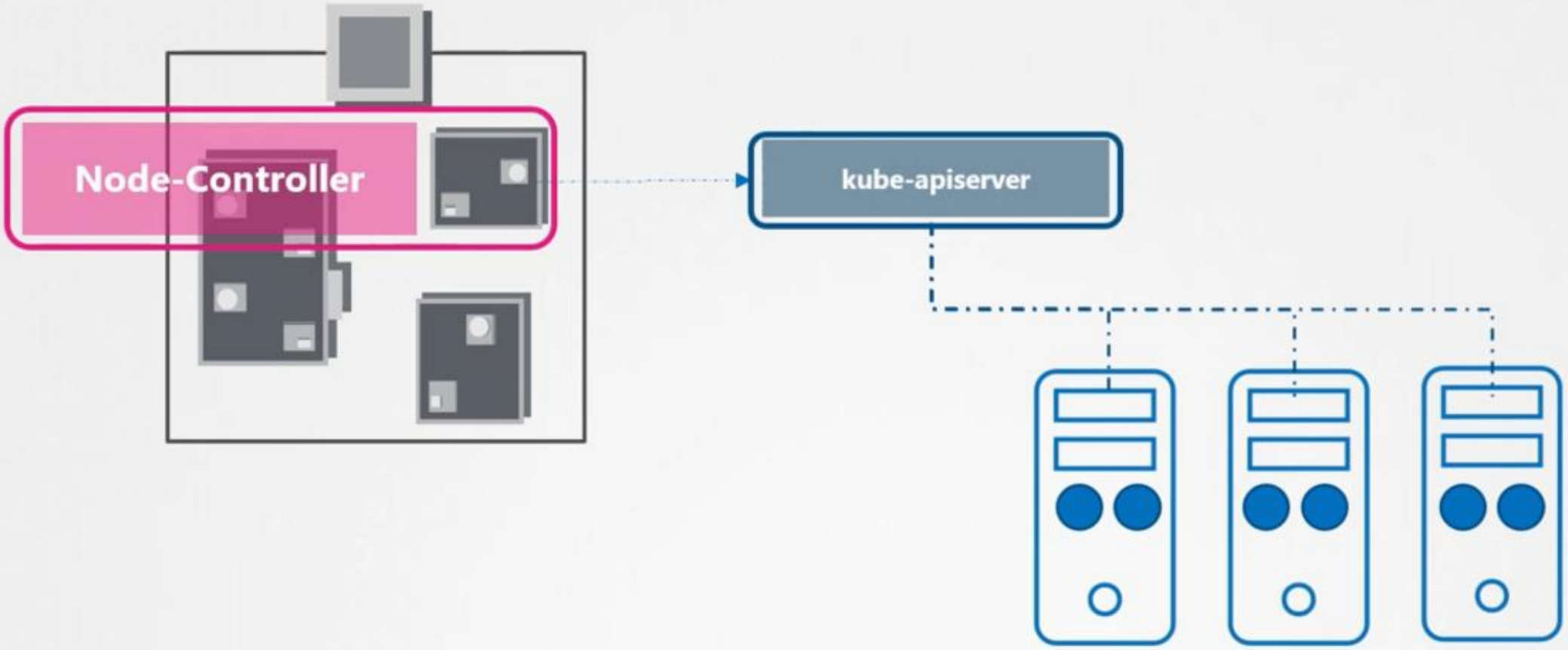




Controller

Watch Status

Remediate Situation

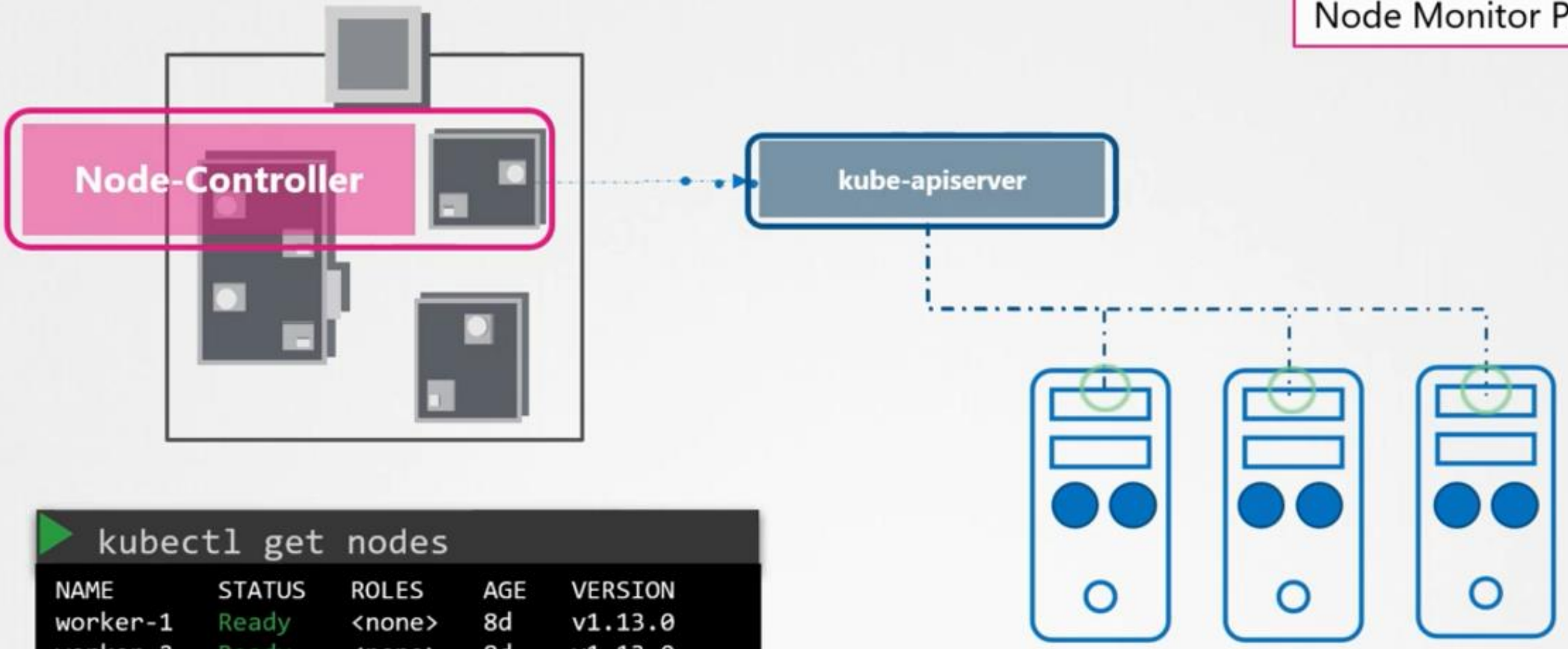


Controller

Watch Status

Remediate Situation

Node Monitor Period = 5s




```
kubectl get nodes
NAME       STATUS   ROLES    AGE   VERSION
worker-1   Ready    <none>   8d    v1.13.0
worker-2   Ready    <none>   8d    v1.13.0
```

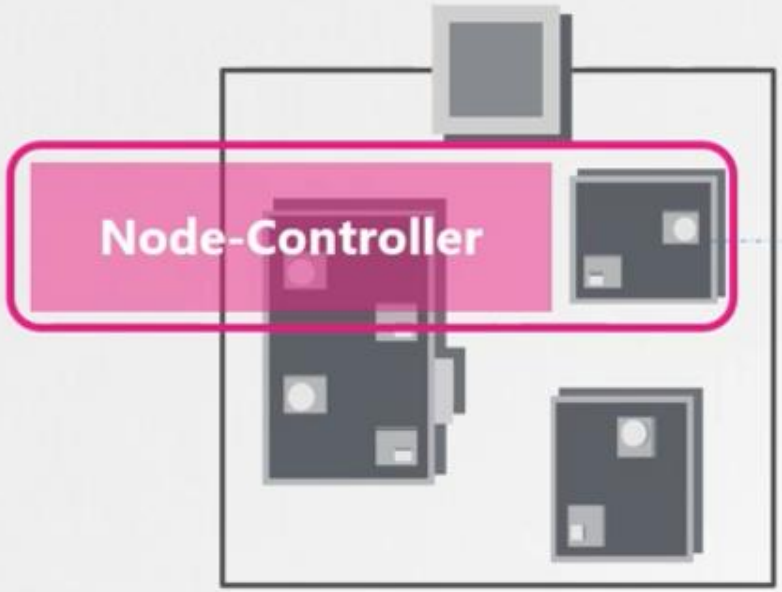
Controller

Watch Status

Remediate Situation

Node Monitor Period = 5s

Node Monitor Grace Period = 40s



kube-apiserver


```
kubectl get nodes
NAME       STATUS     ROLES    AGE   VERSION
worker-1   Ready      <none>   8d    v1.13.0
worker-2   NotReady   <none>   8d    v1.13.0
```

Controller

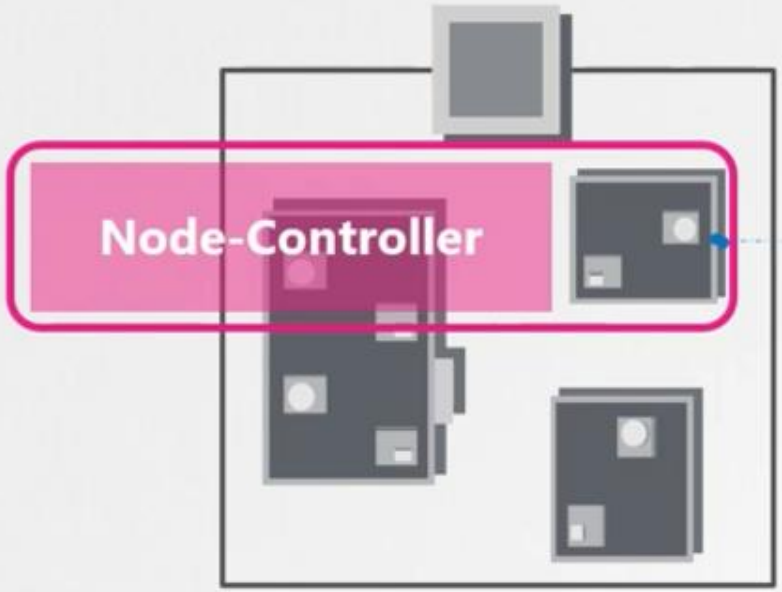
Watch Status

Remediate Situation

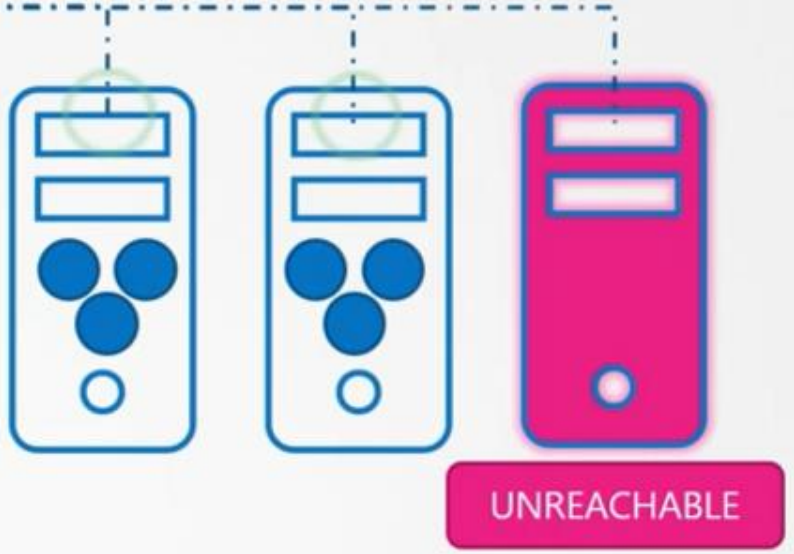
Node Monitor Period = 5s

Node Monitor Grace Period = 40s

POD Eviction Timeout = 5m

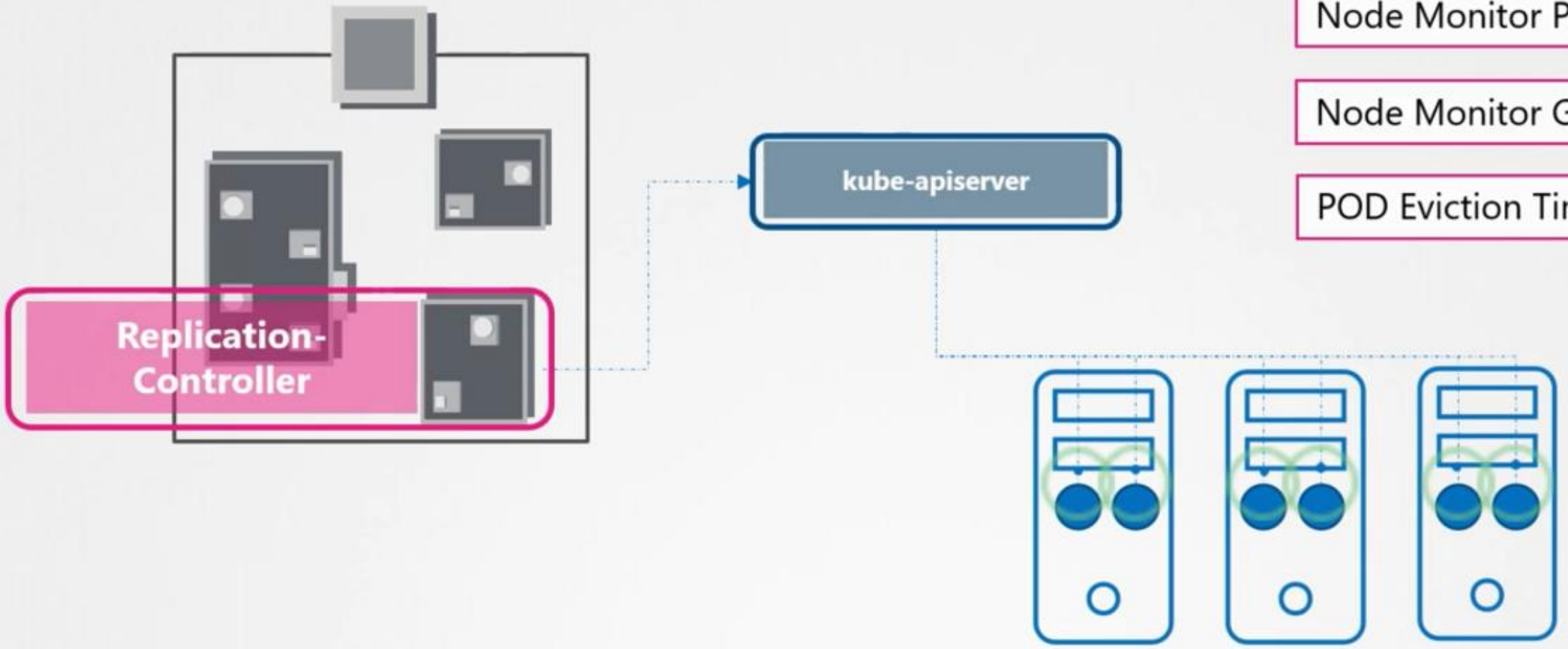


kube-apiserver




```
kubectl get nodes
NAME       STATUS     ROLES    AGE   VERSION
worker-1   Ready      <none>   8d    v1.13.0
worker-2   NotReady   <none>   8d    v1.13.0
```

Controller



Watch Status

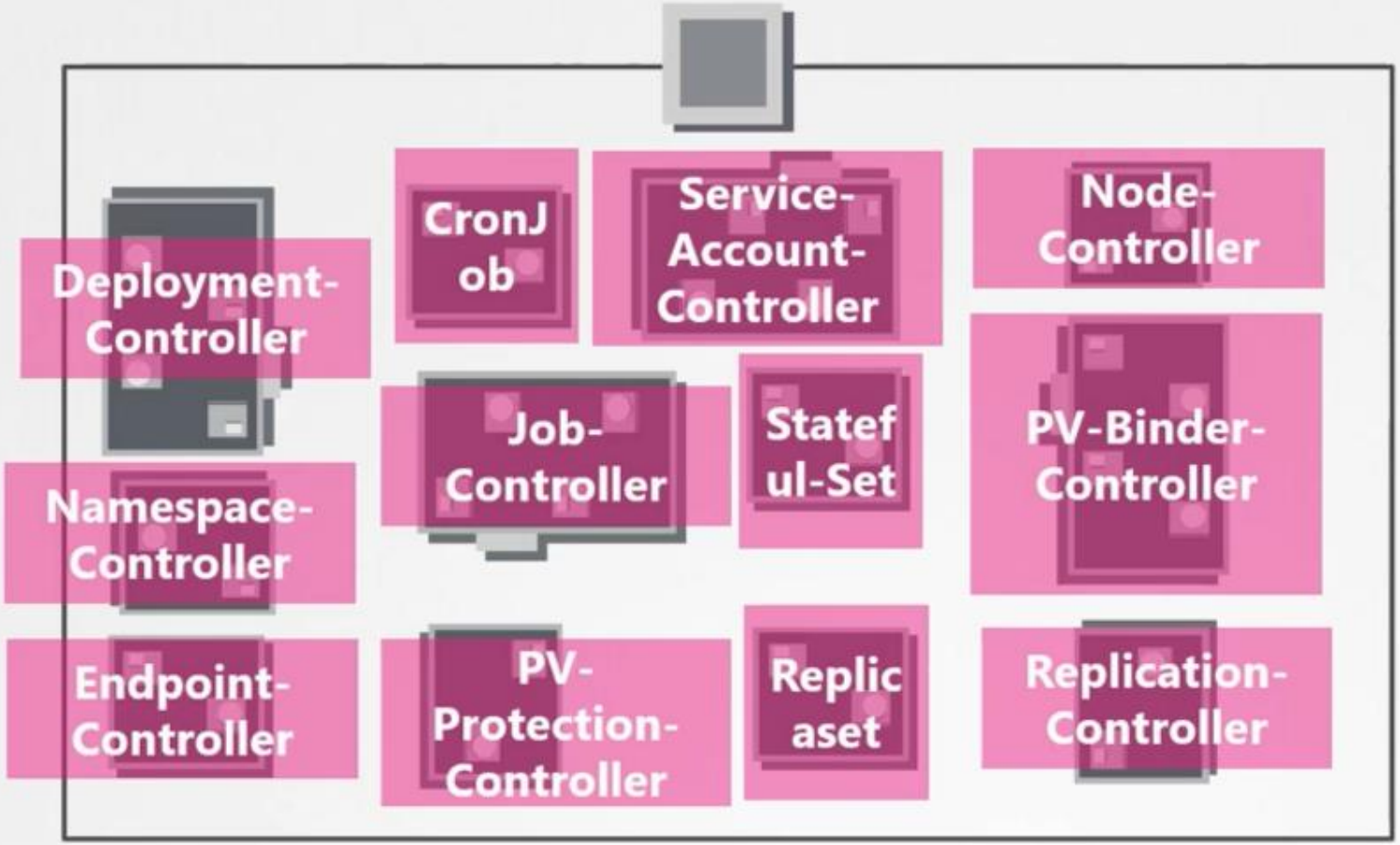
Remediate Situation

Node Monitor Period = 5s

Node Monitor Grace Period = 40s

POD Eviction Timeout = 5m

Controller



Watch Status

Remediate Situation

Node Monitor Period = 5s

Node Monitor Grace Period = 40s

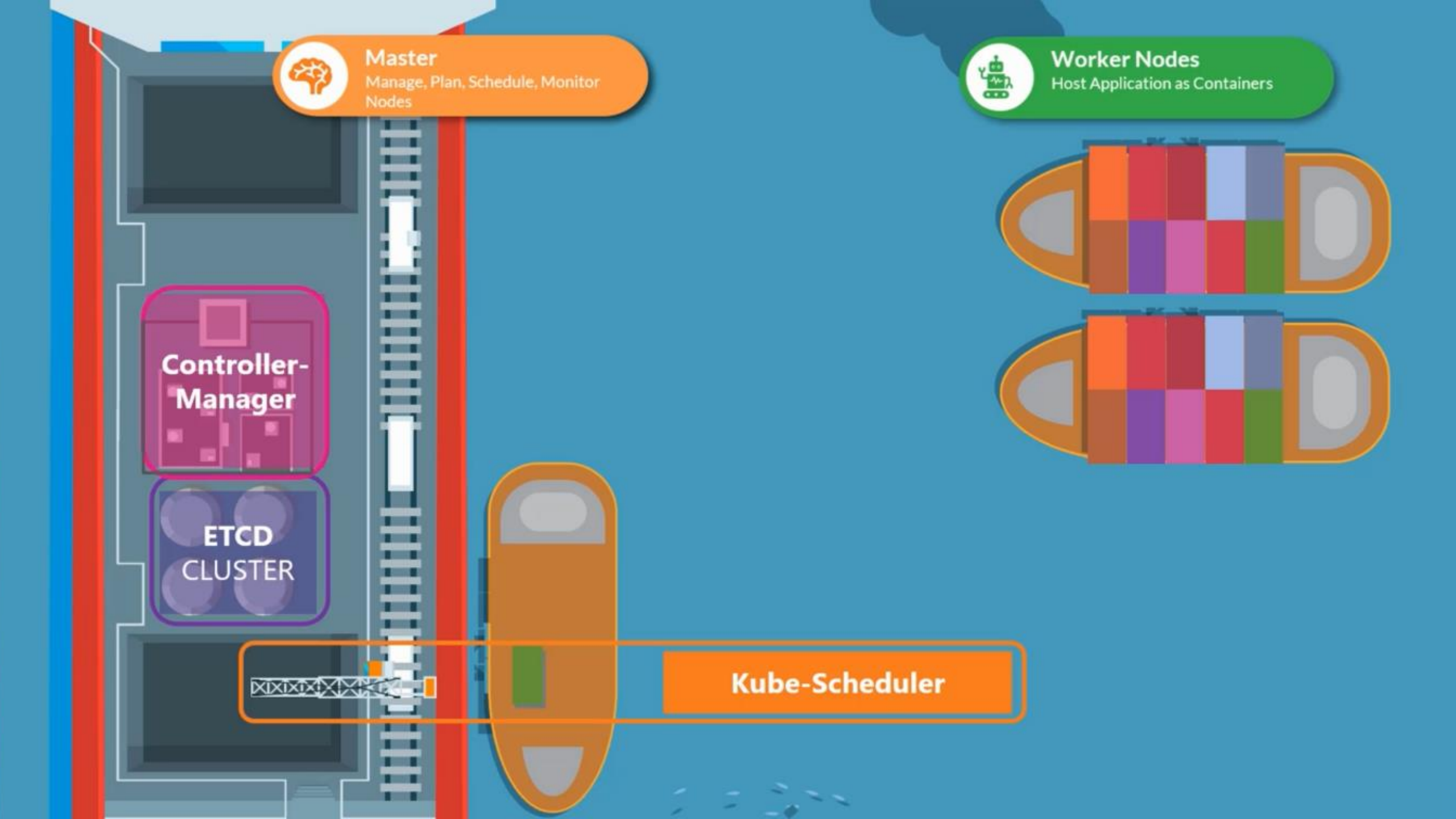
POD Eviction Timeout = 5m

View kube-controller-manager - kubeadm

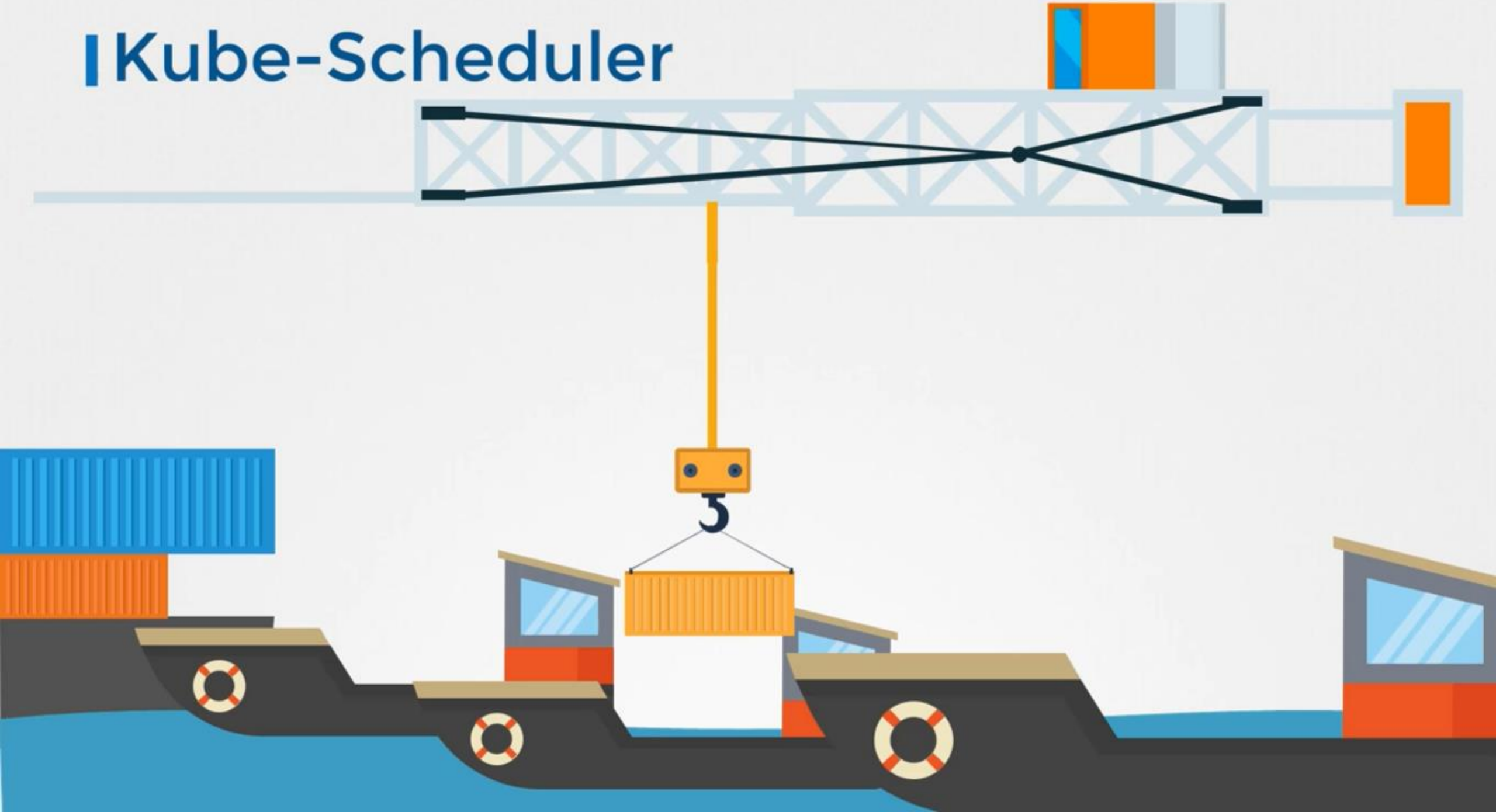
```
kubectl get pods -n kube-system
```

NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
kube-system	coredns-78fcdf6894-hwrq9	1/1	Running	0	16m
kube-system	coredns-78fcdf6894-rzhjr	1/1	Running	0	16m
kube-system	etcd-master	1/1	Running	0	15m
kube-system	kube-apiserver-master	1/1	Running	0	15m
kube-system	kube-controller-manager-master	1/1	Running	0	15m
kube-system	kube-proxy-lzt6f	1/1	Running	0	16m
kube-system	kube-proxy-zm5qd	1/1	Running	0	16m
kube-system	kube-scheduler-master	1/1	Running	0	15m
kube-system	weave-net-29z42	2/2	Running	1	16m
kube-system	weave-net-snmdl	2/2	Running	1	16m

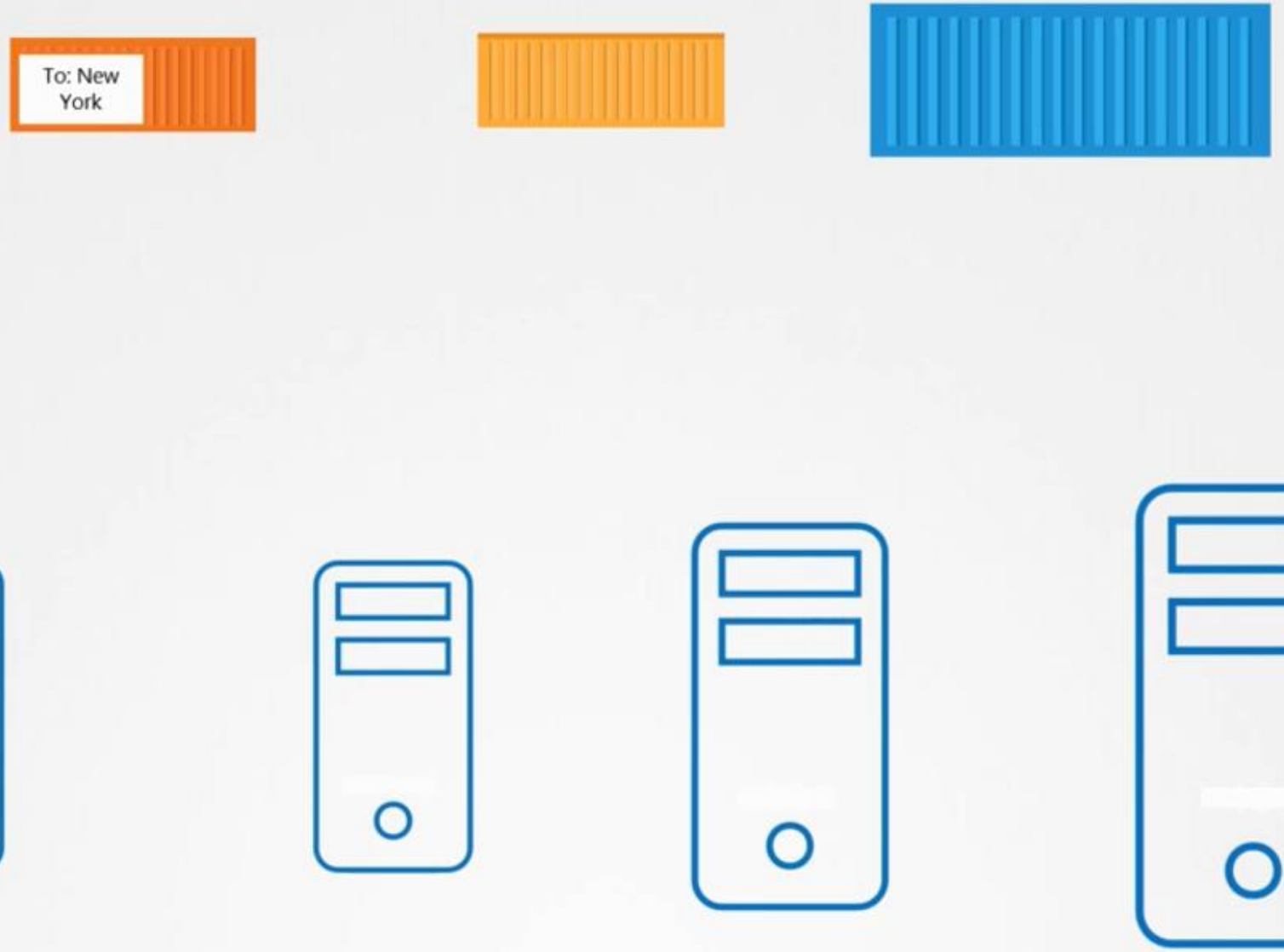
Kube Scheduler


Master
Manage, Plan, Schedule, Monitor Nodes
Worker Nodes
Host Application as Containers
Controller-Manager
ETCD CLUSTER
Kube-Scheduler

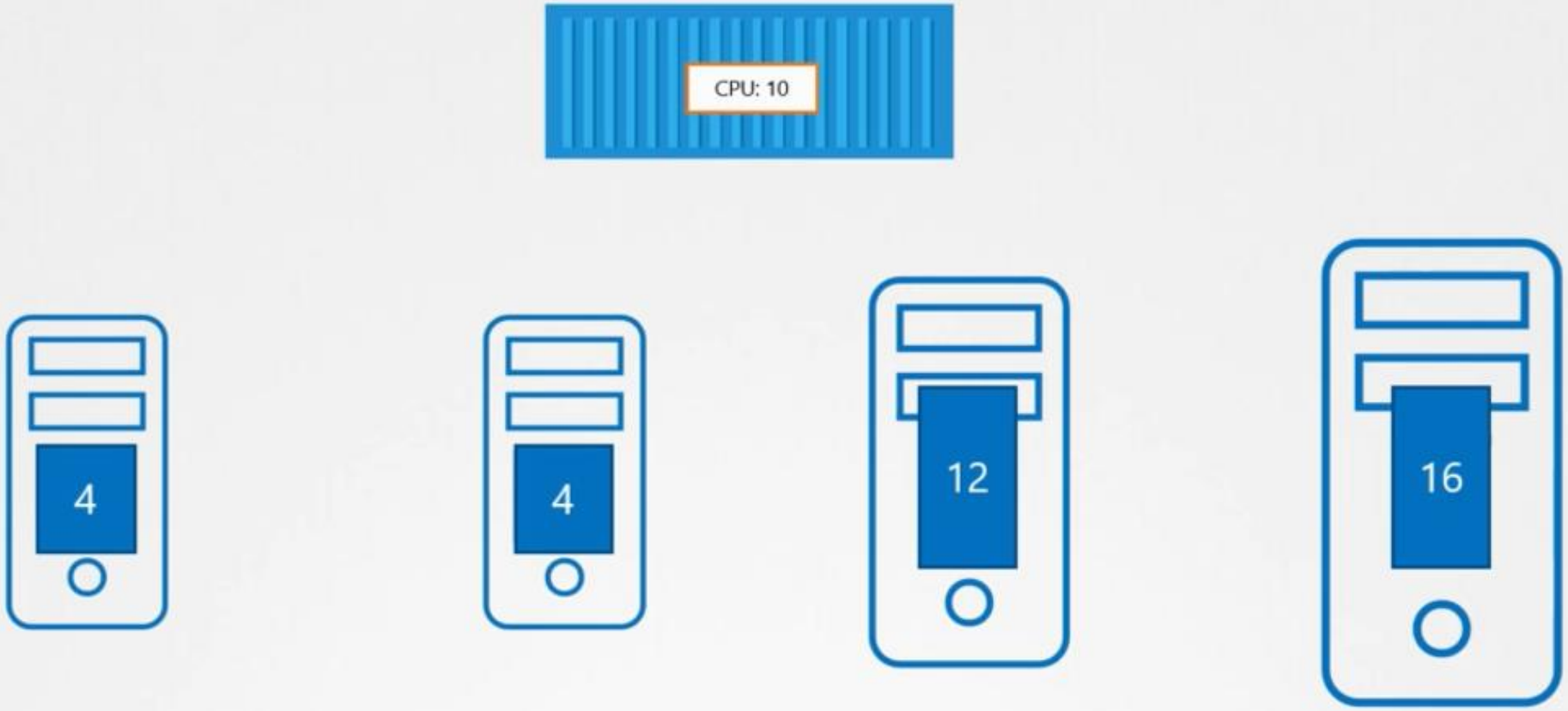
Kube-Scheduler



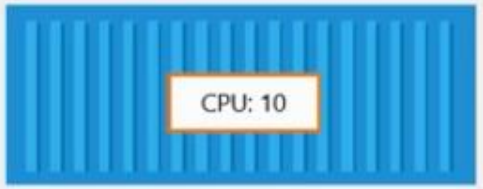
Kube-Scheduler



Kube-Scheduler

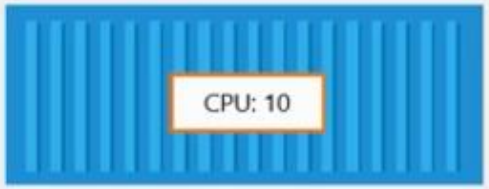


Kube-Scheduler



1. Filter Nodes

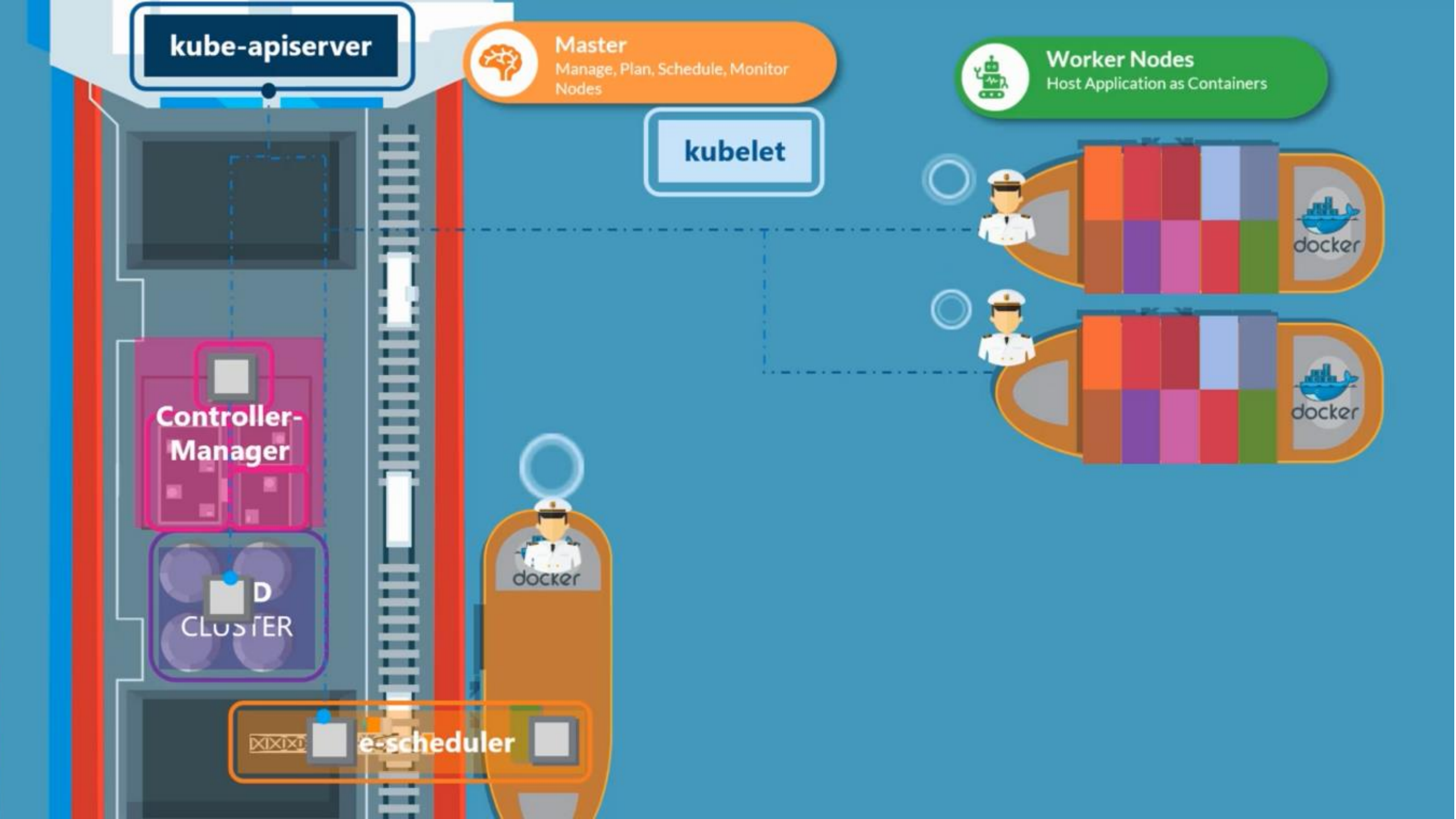
Kube-Scheduler



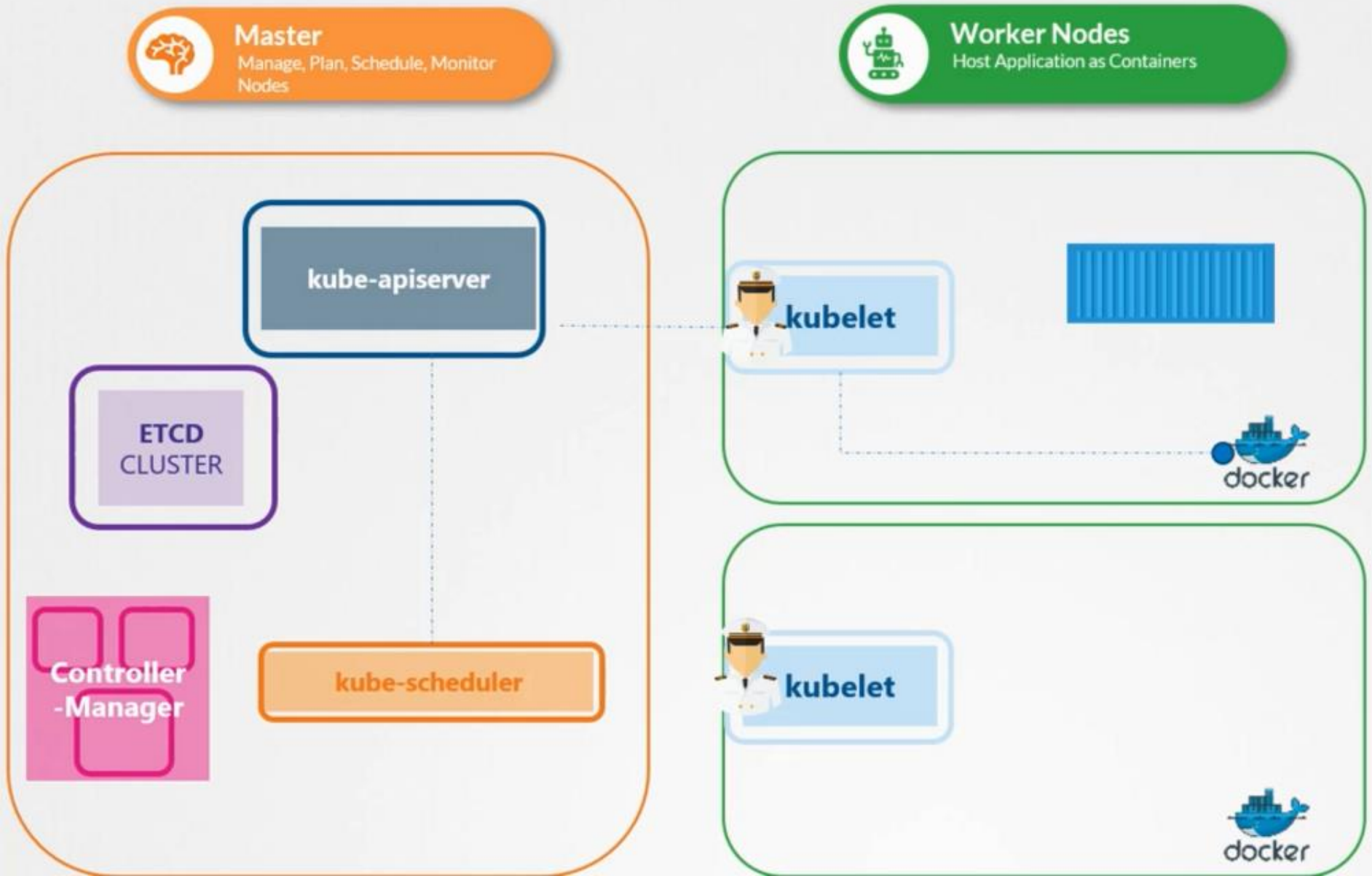
1. Filter Nodes

2. Rank Nodes

Kubelet


kube-apiserver
Master
Manage, Plan, Schedule, Monitor Nodes
kubelet
Worker Nodes
Host Application as Containers
docker
docker
Controller-Manager
D
CLUSTER
docker
e-scheduler

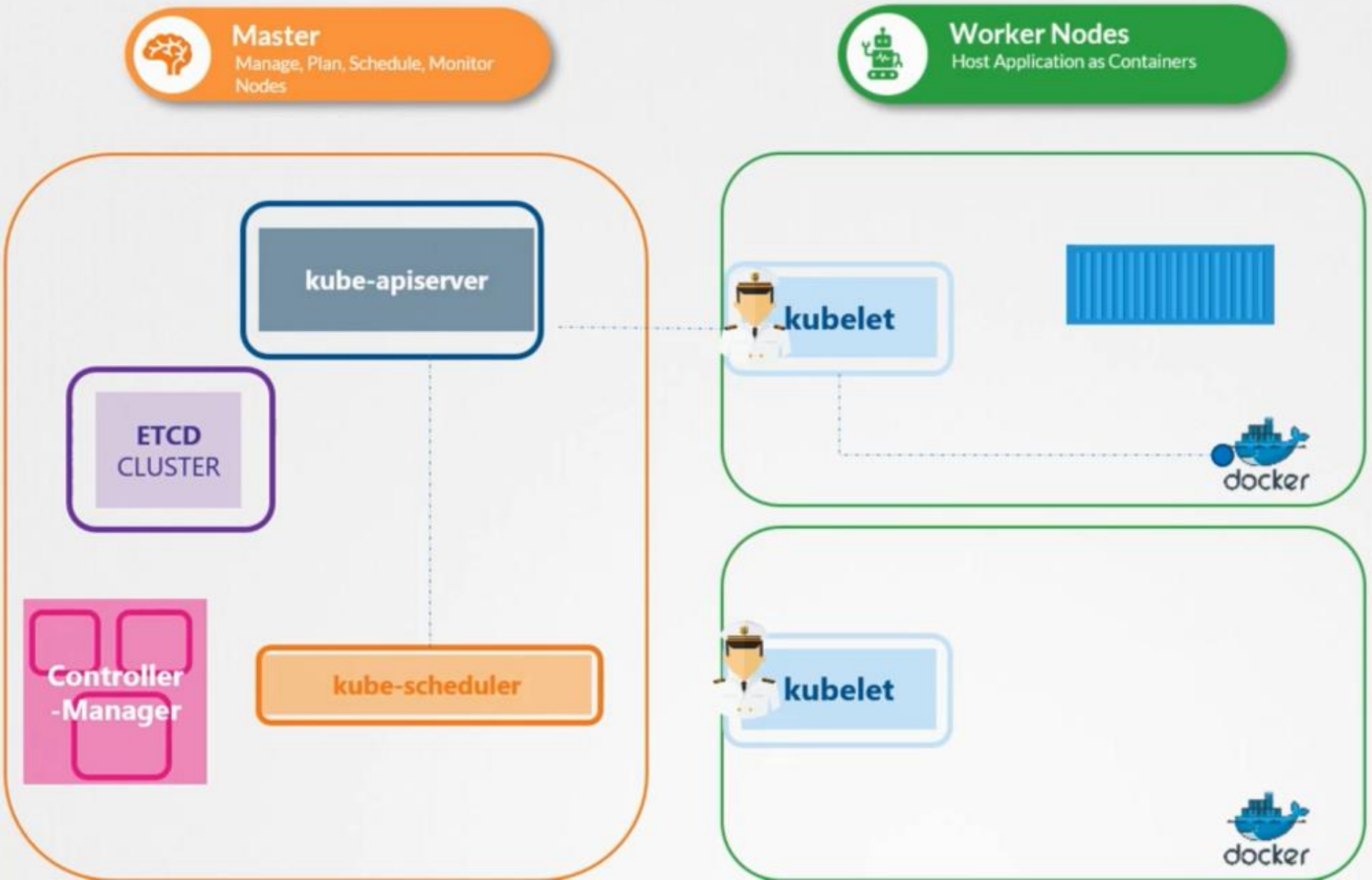
Kubernetes Architecture



Register Node

Create PODs

Kubernetes Architecture

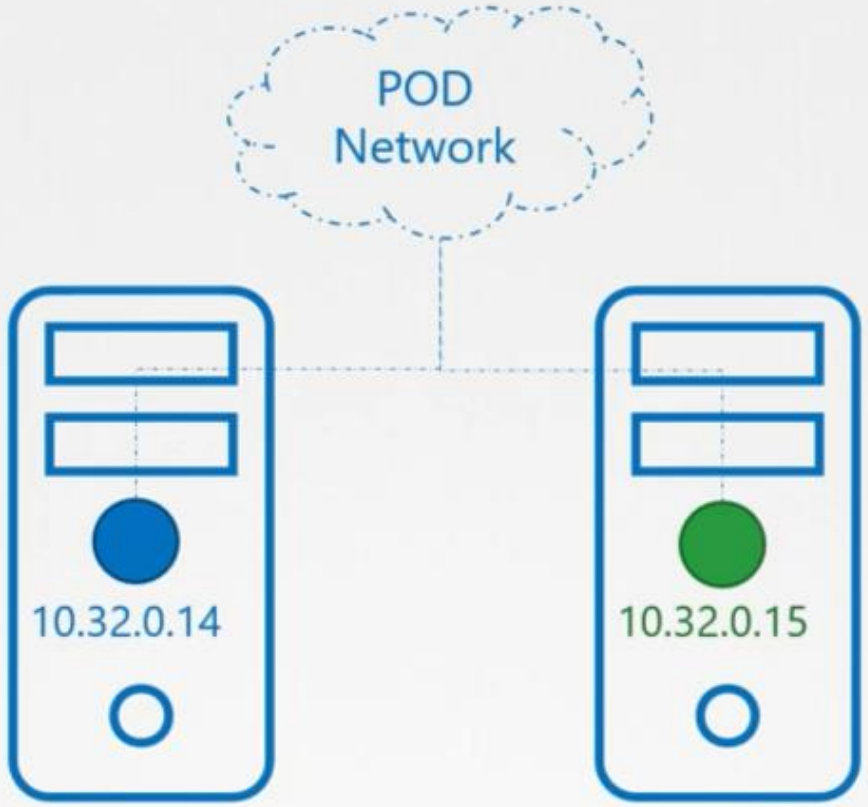


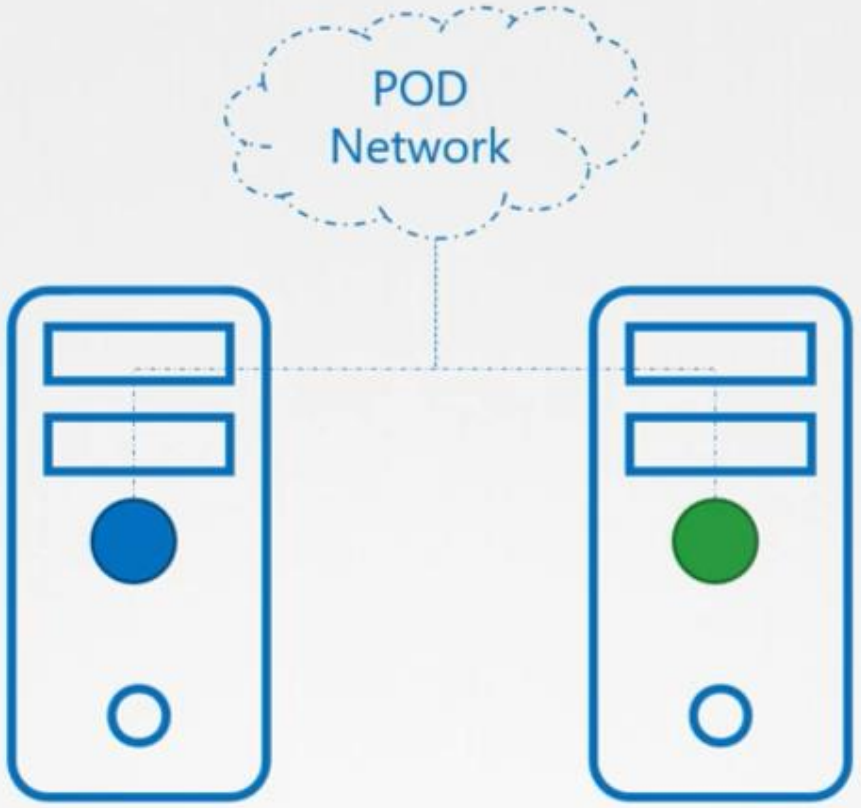
Register Node

Create PODs

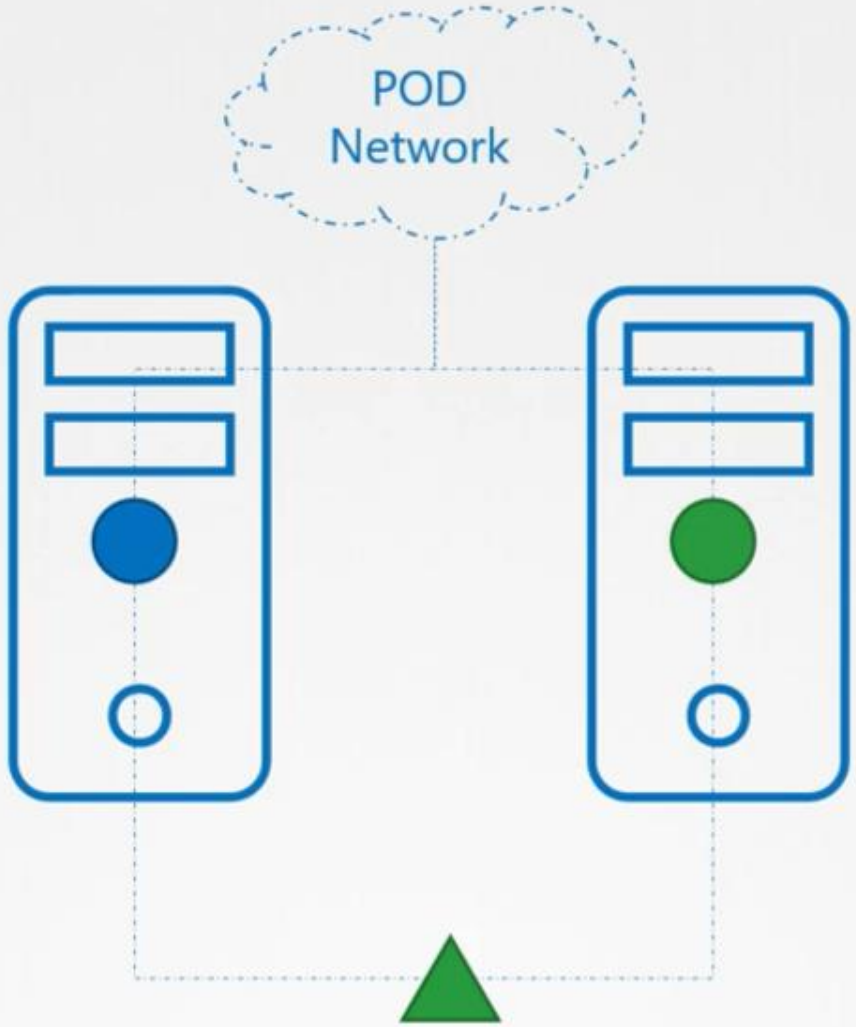
Monitor Node & PODs

Kube Proxy

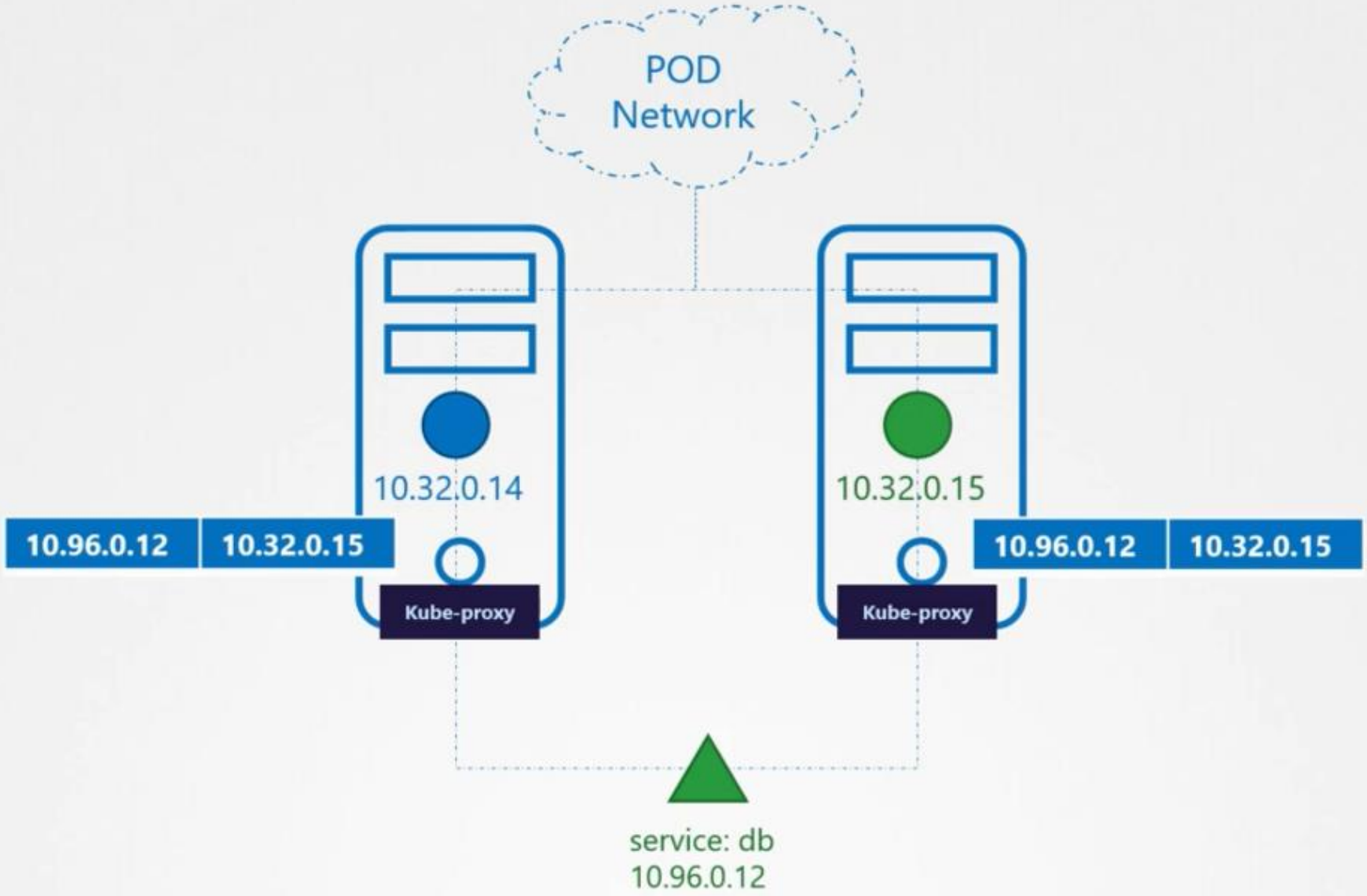

POD
Network
10.32.0.14
10.32.0.15



service: db
10.96.0.12


POD
Network
service: db
10.96.0.12

Kube-proxy


POD
Network
10.32.0.14
10.96.0.12
10.32.0.15
Kube-proxy
10.32.0.15
10.96.0.12
10.32.0.15
Kube-proxy
service: db
10.96.0.12

All slides have been taken from “Kubernetes for beginners” course