

Kubernetes in 30 minutes

Mar 5, 2019

Jose Carlos Barrales





Netscape Navigator



~~Netscape Navigator~~



Kubernetes



κυβερνήτης

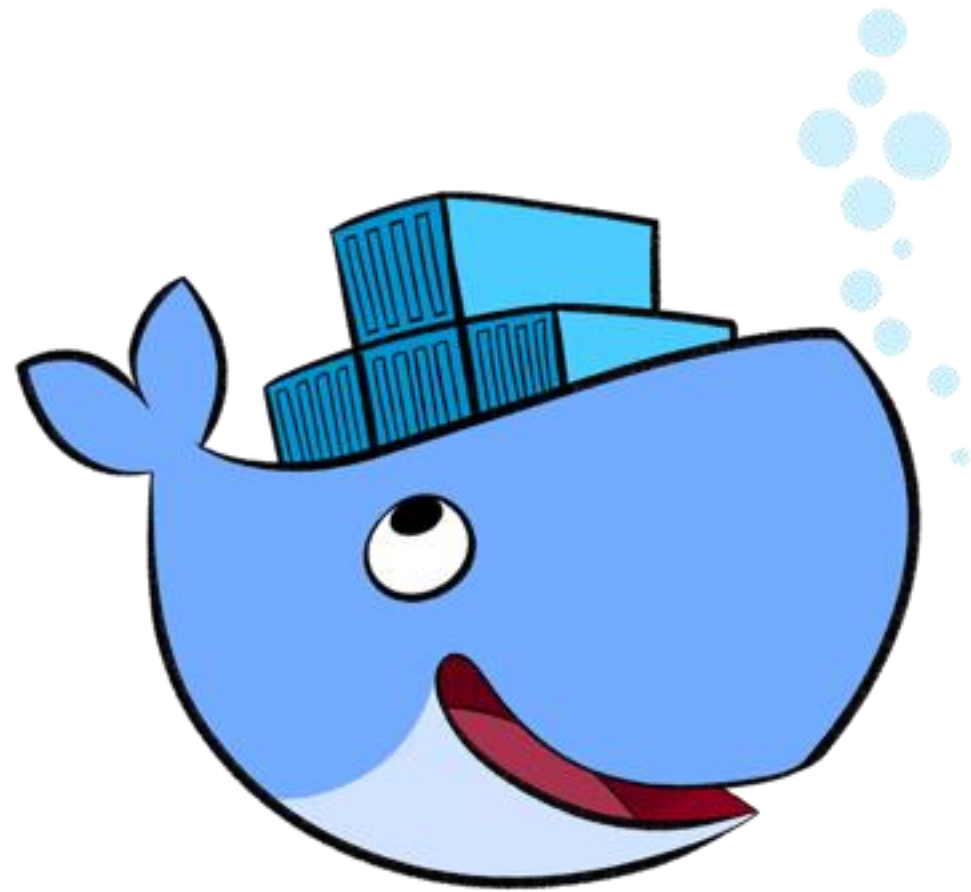


k8s

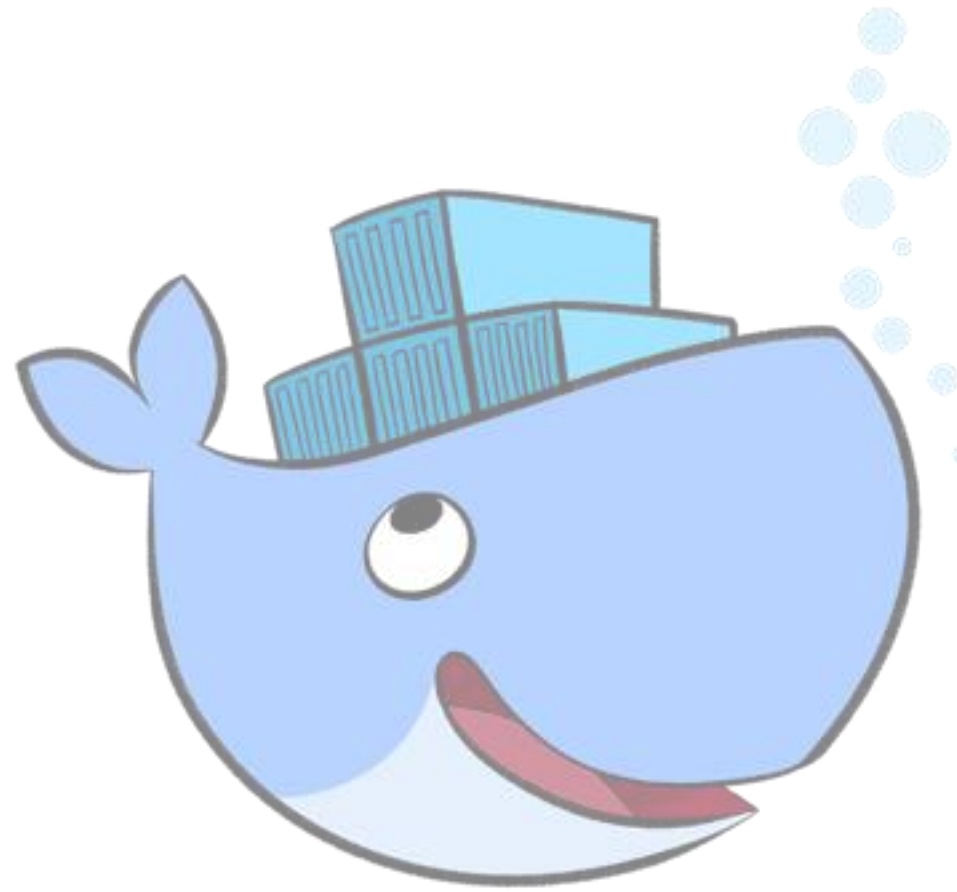
**"Something to do
with
containers...?"**

Assumptions

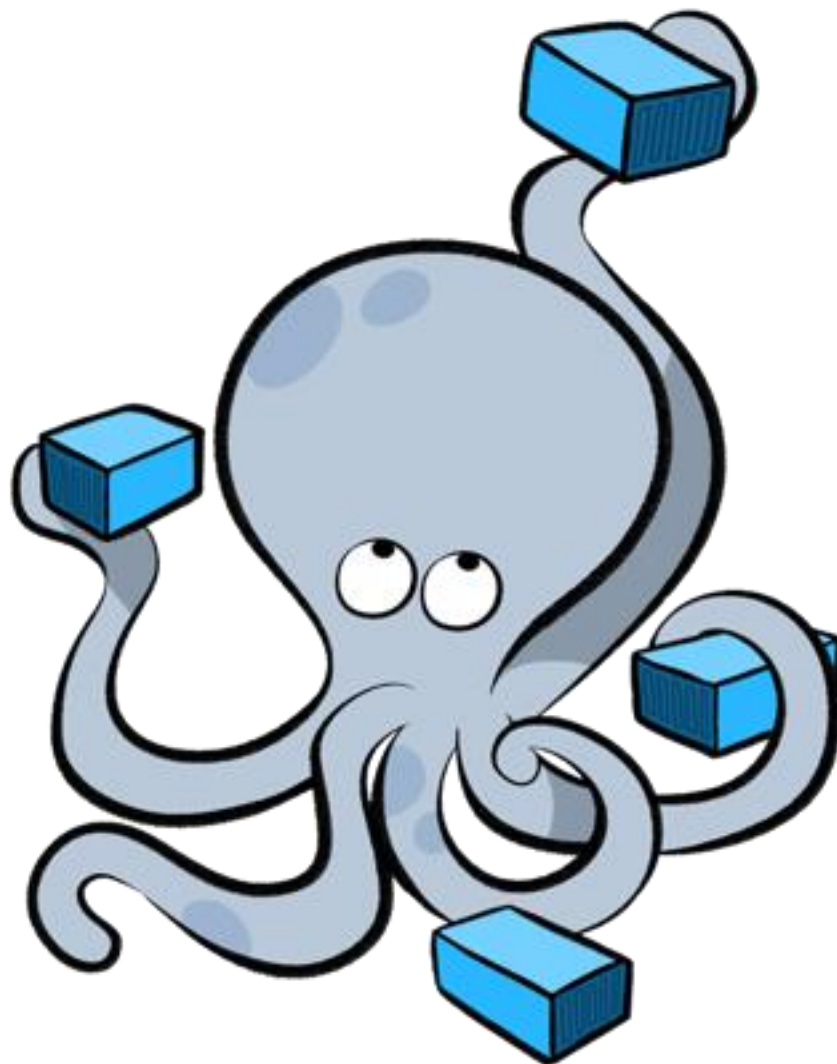
- You know about containers
- You know about orchestration
- You know about the painful art of deploying stuff



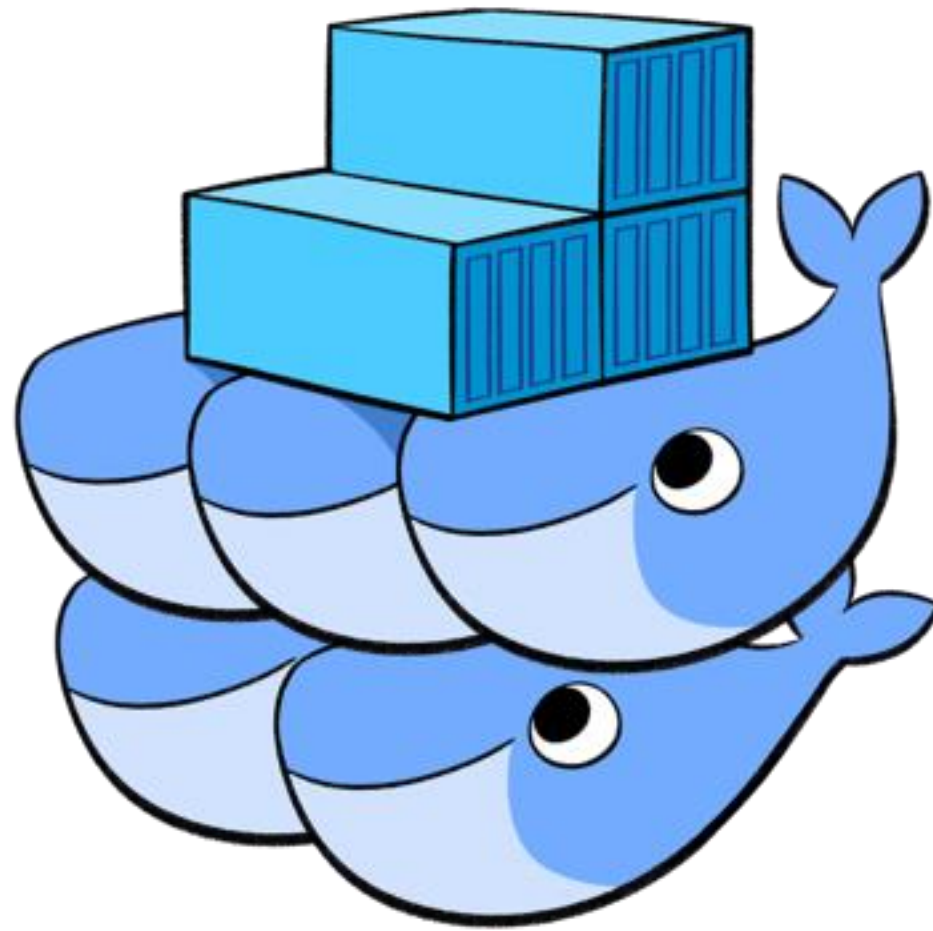
Docker



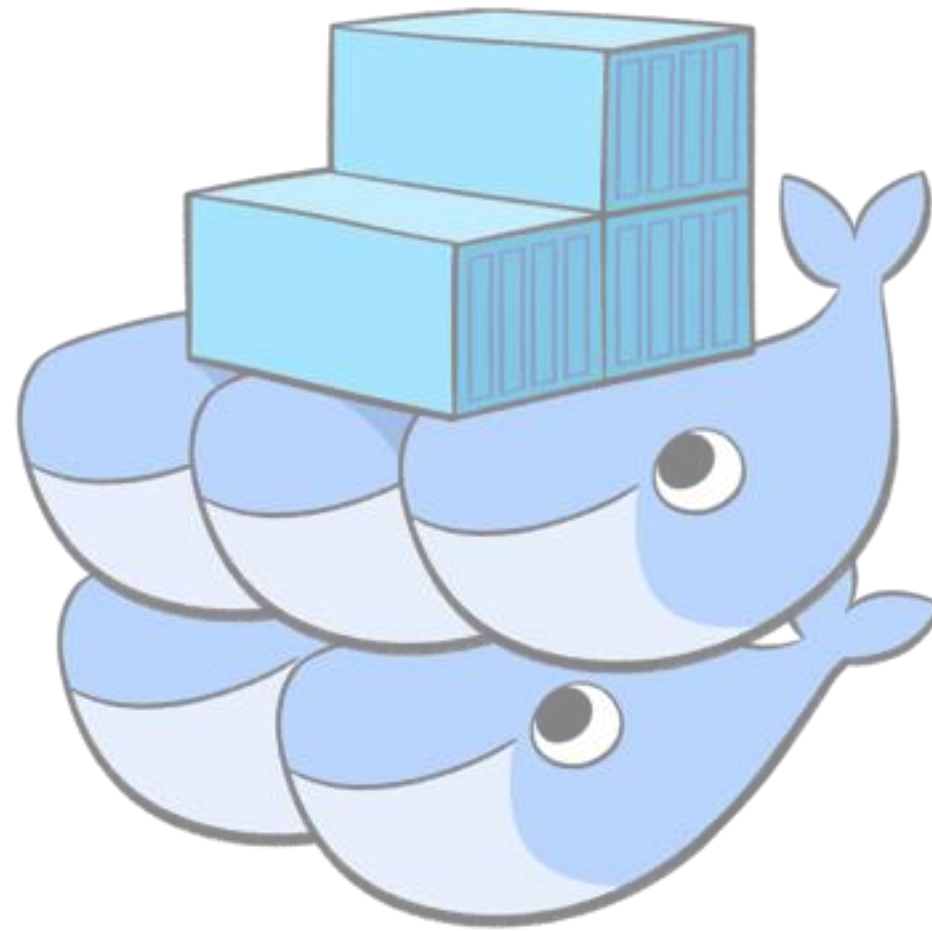
'Random containers doing their own sh*t'



Compose



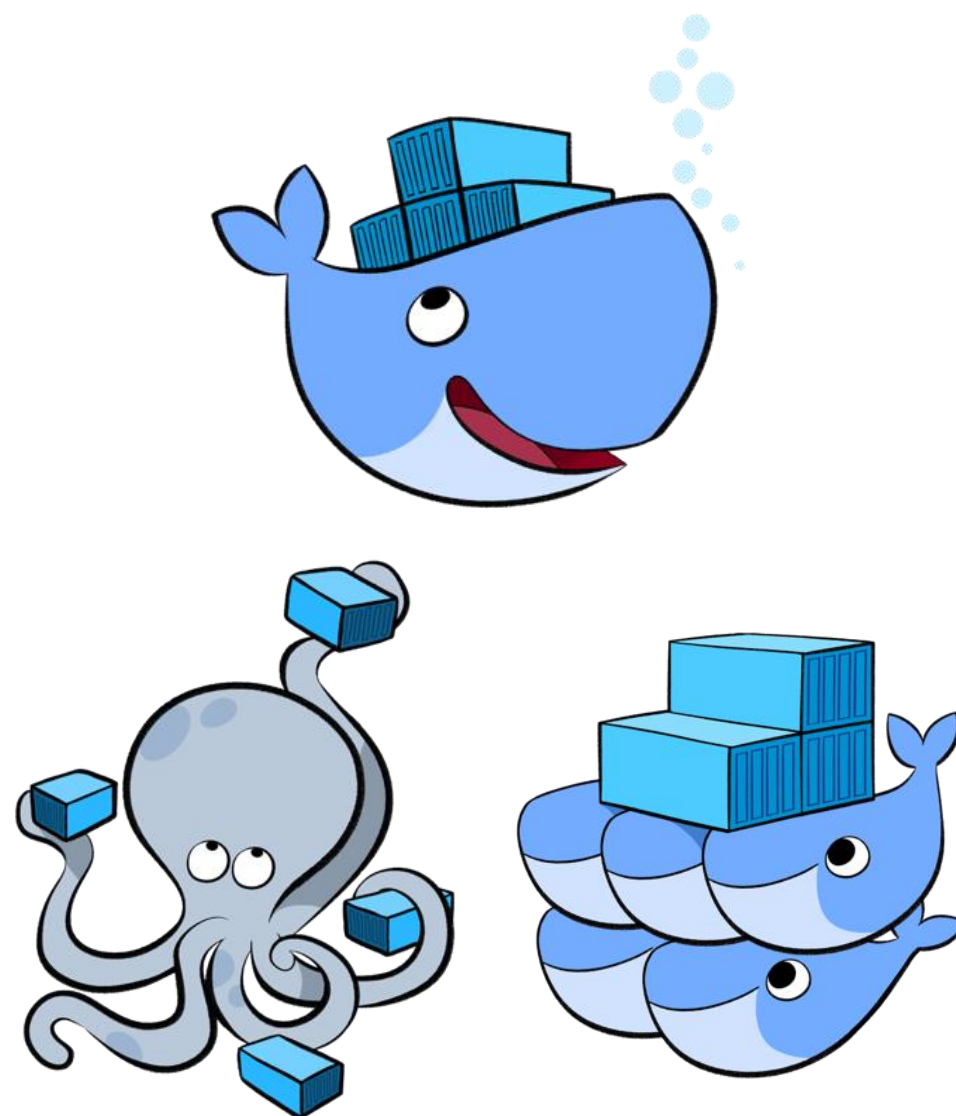
Swarm



**"Clustering and
Discovery"**
(hey, we're getting somewhere!)



=



“Batteries Sold Separately”

“Batteries Included”

K8s on GCP



GKE



GCE



Networking

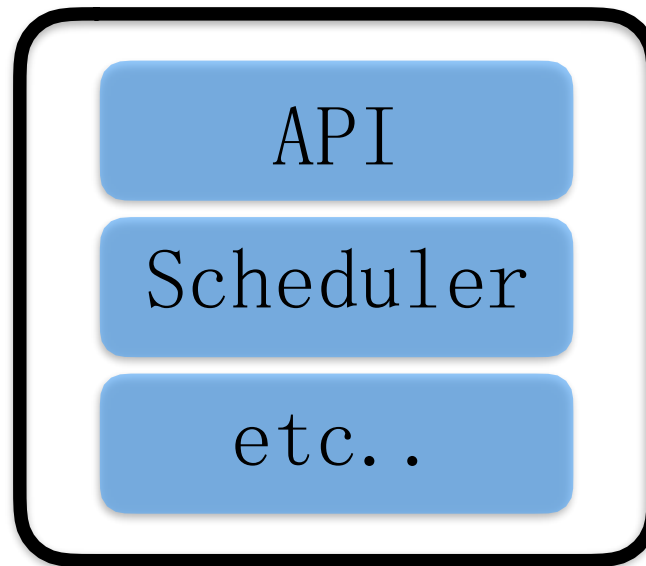
...and others

Basic Concepts

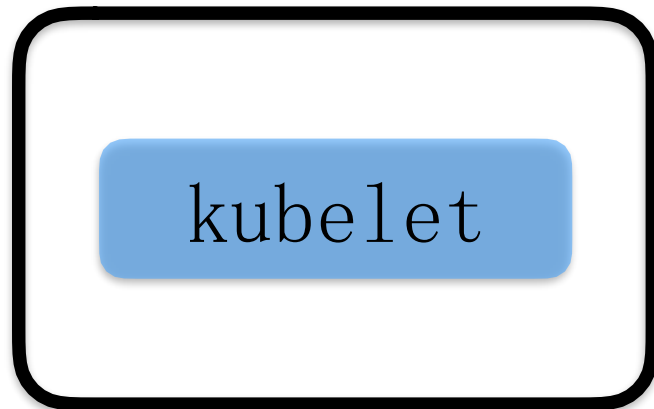
Node

**Hosts running
k8s daemons**

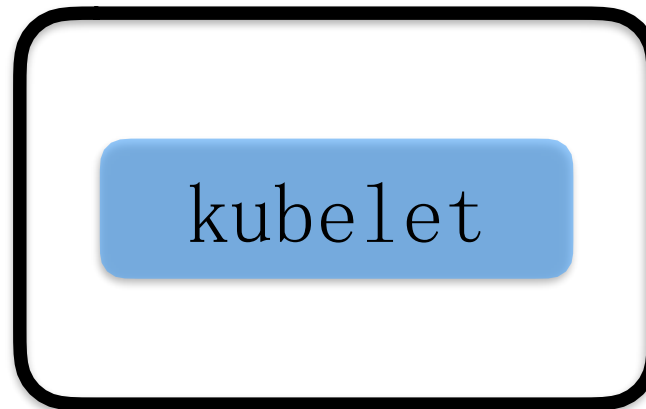
Master



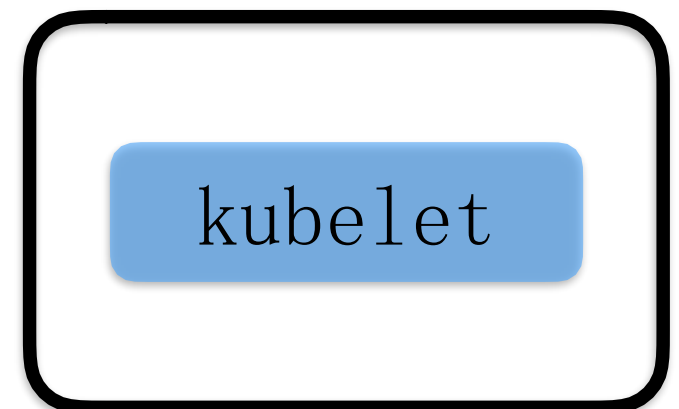
Node



Node



Node

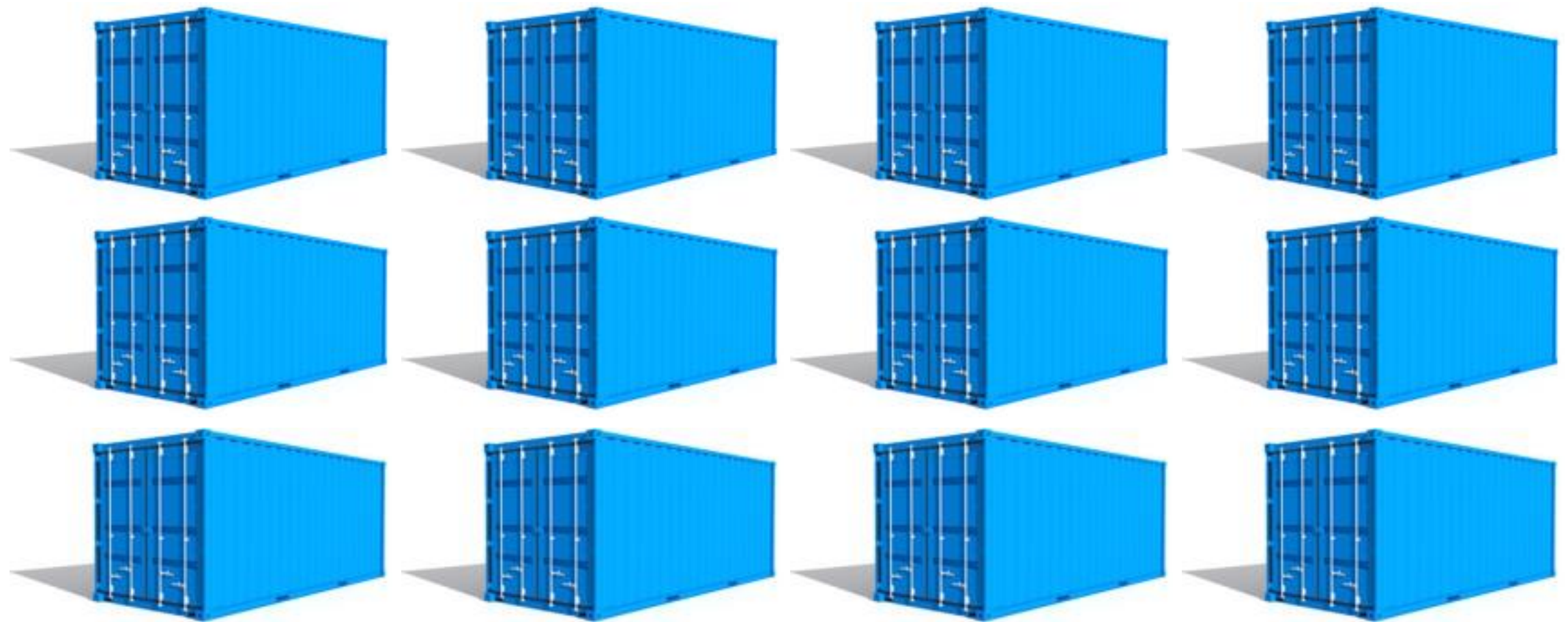


Pod

基本

Basic unit of
deployment in k8s

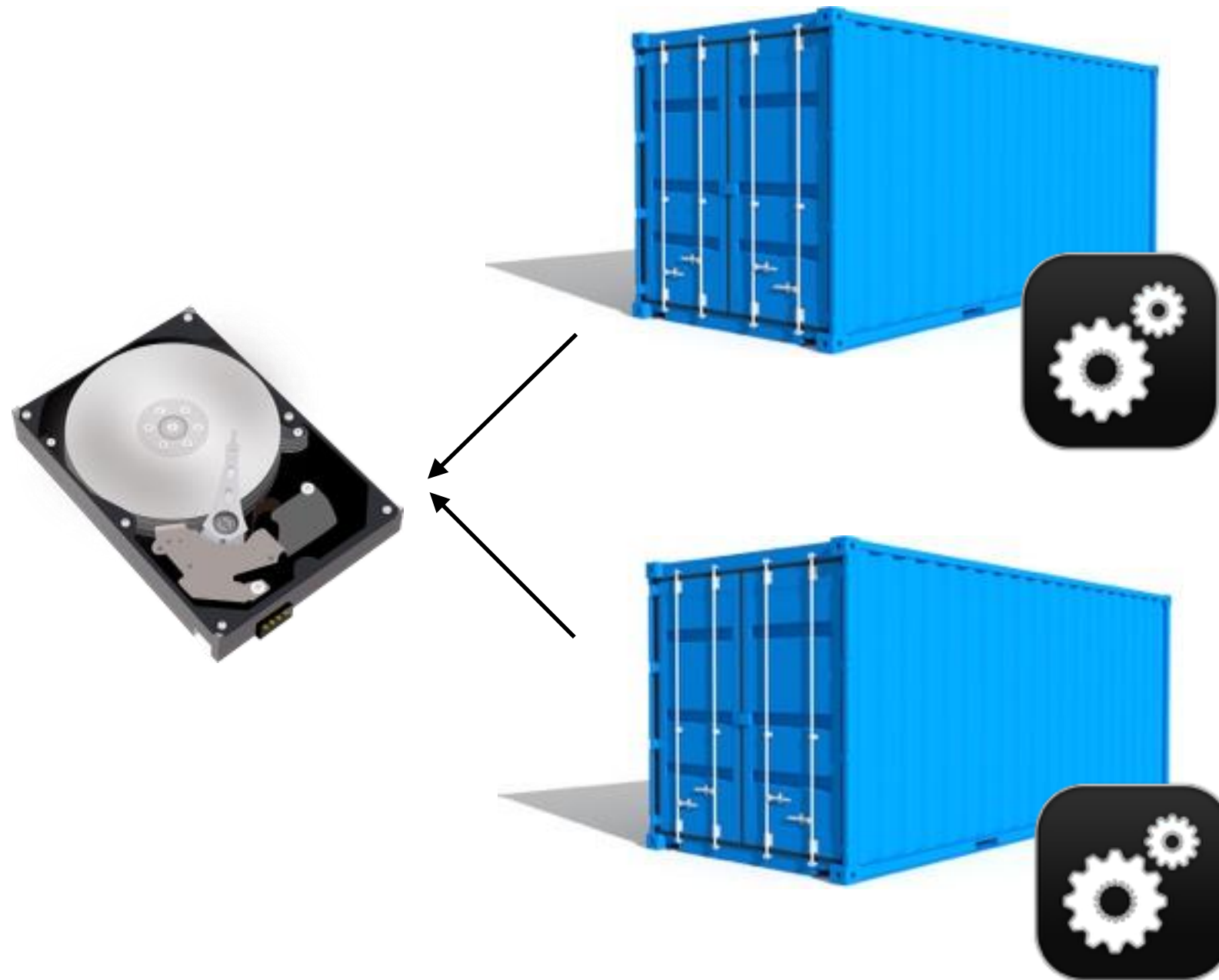
Group of Containers



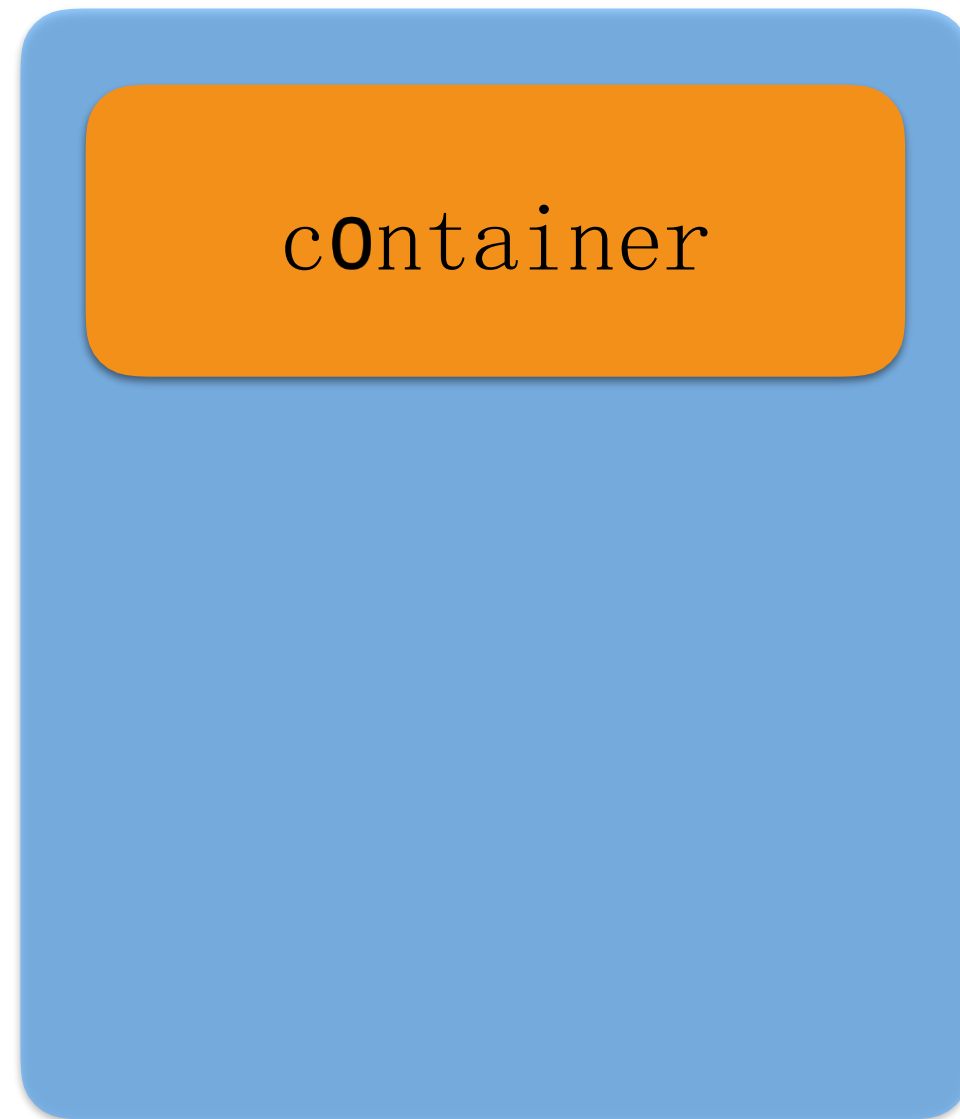
Container configurations



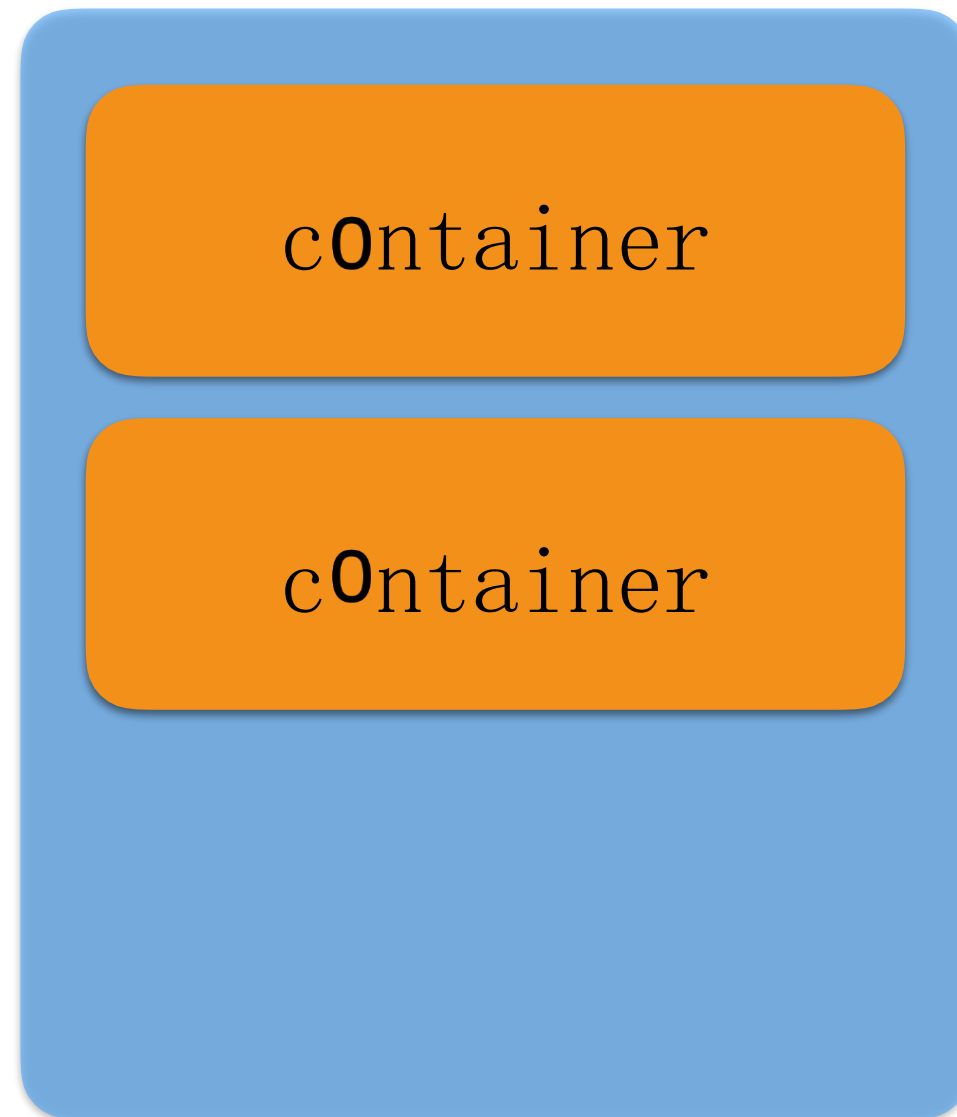
Shared storage



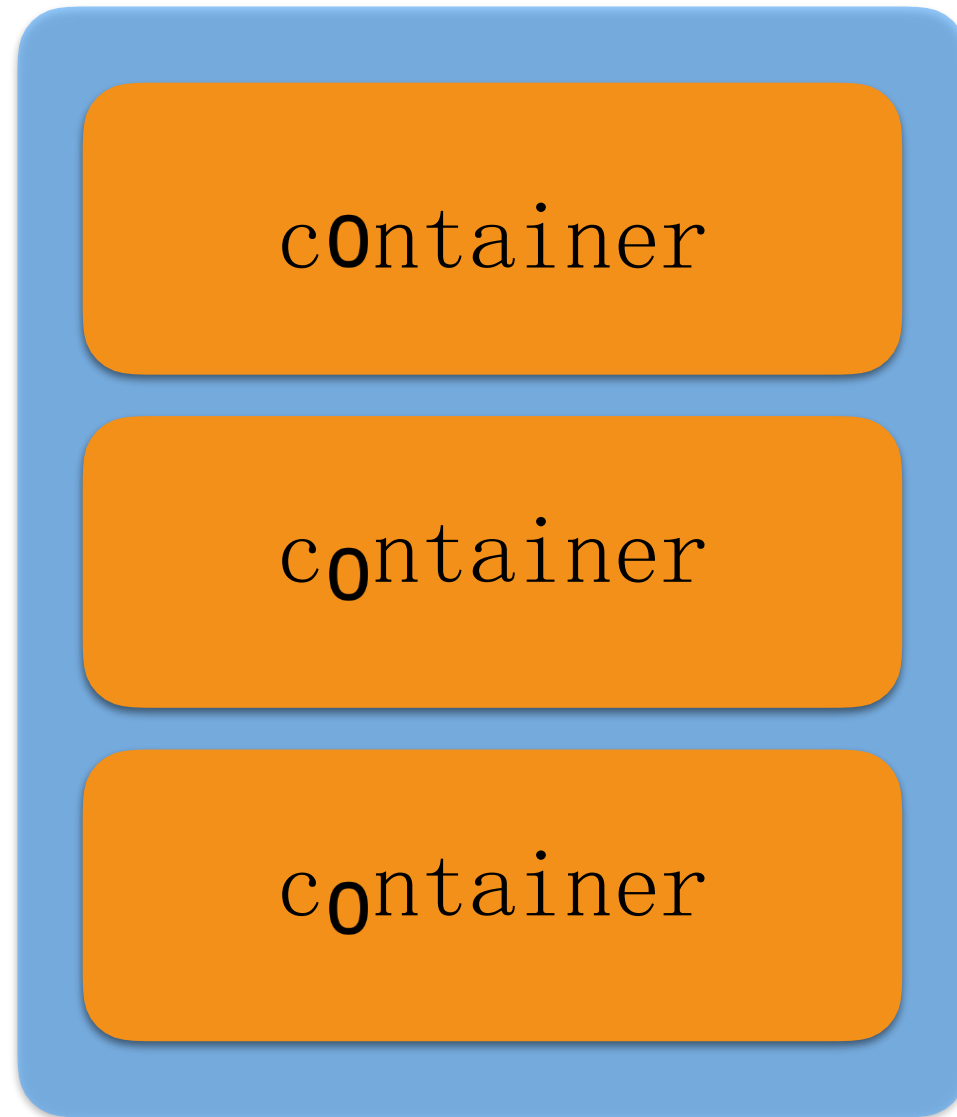
Pod



Pod



Pod



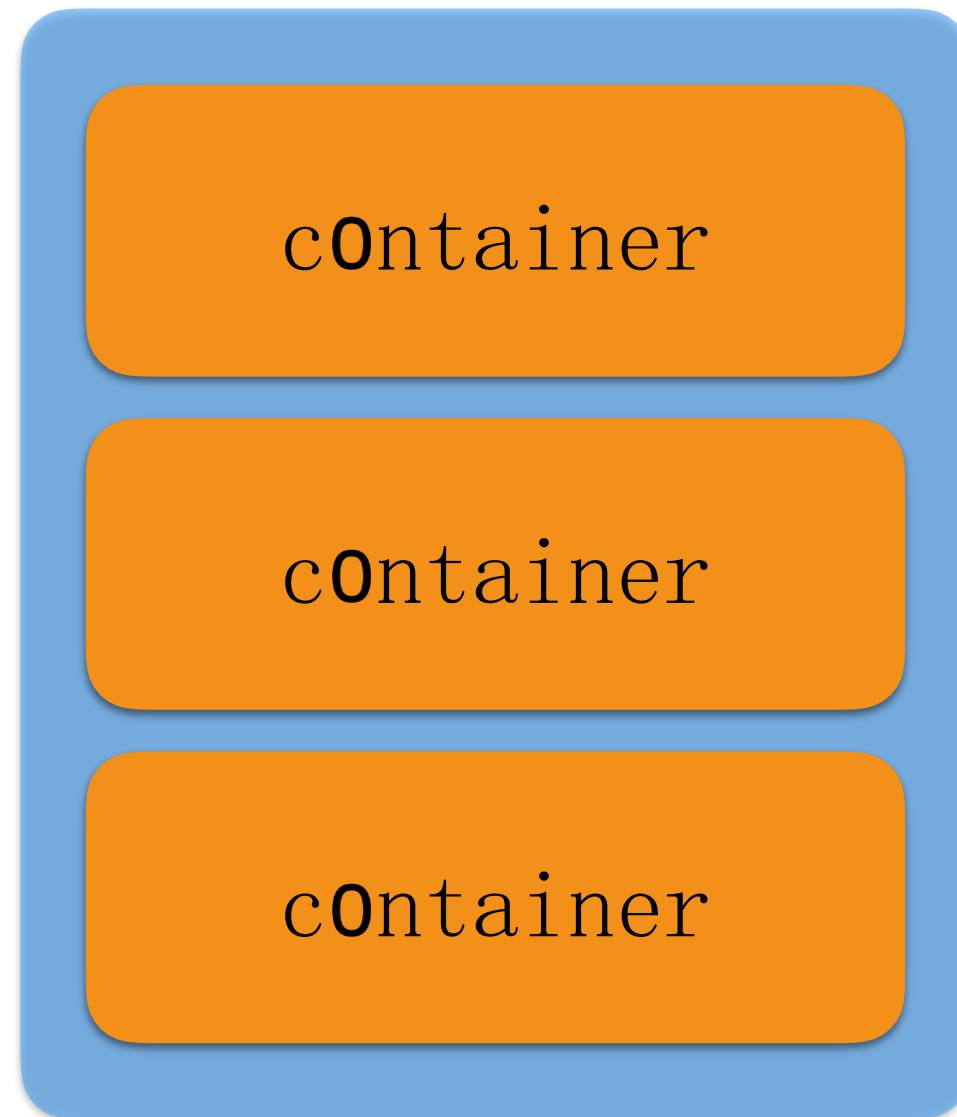
Pod

e. g.

nginx

app

redis/cache



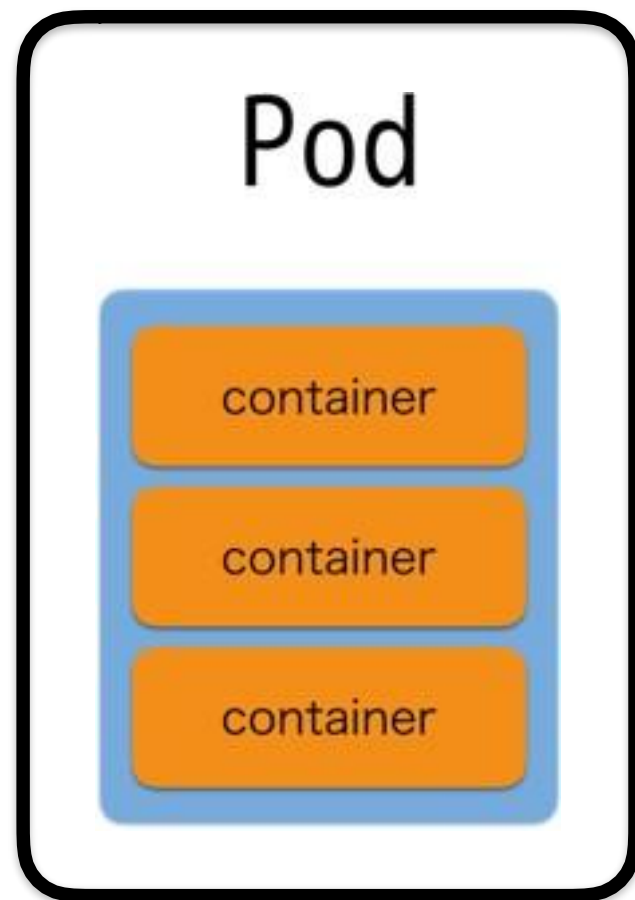
Scheduled together

(“co-scheduled”)

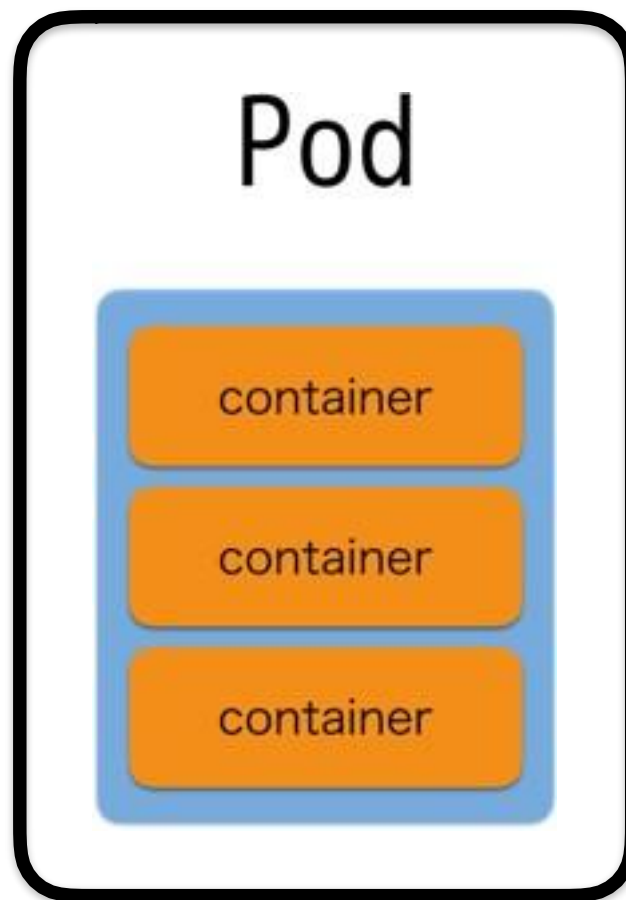
Guaranteed to be on
the same node

("co-located")

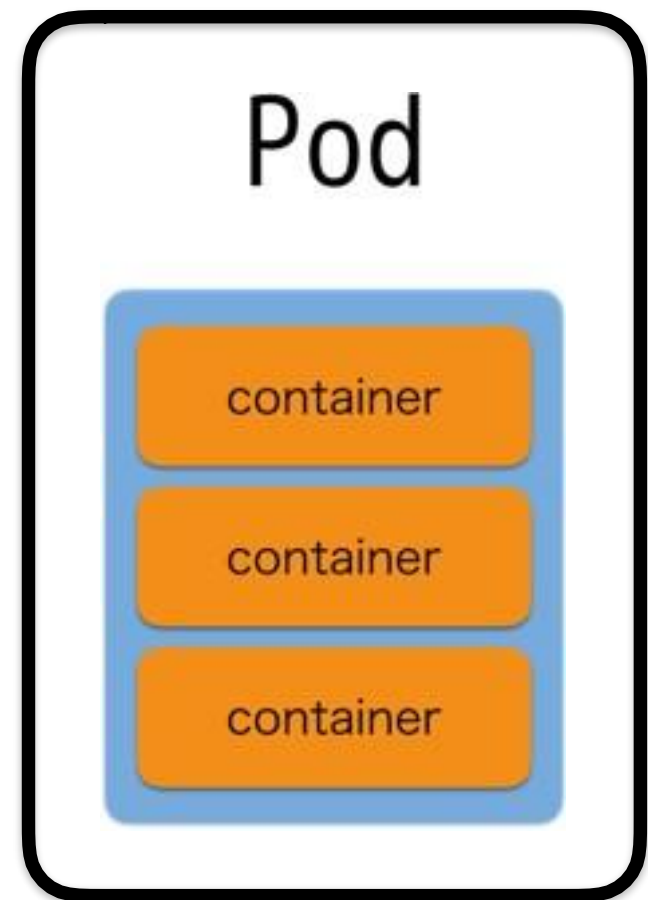
Node



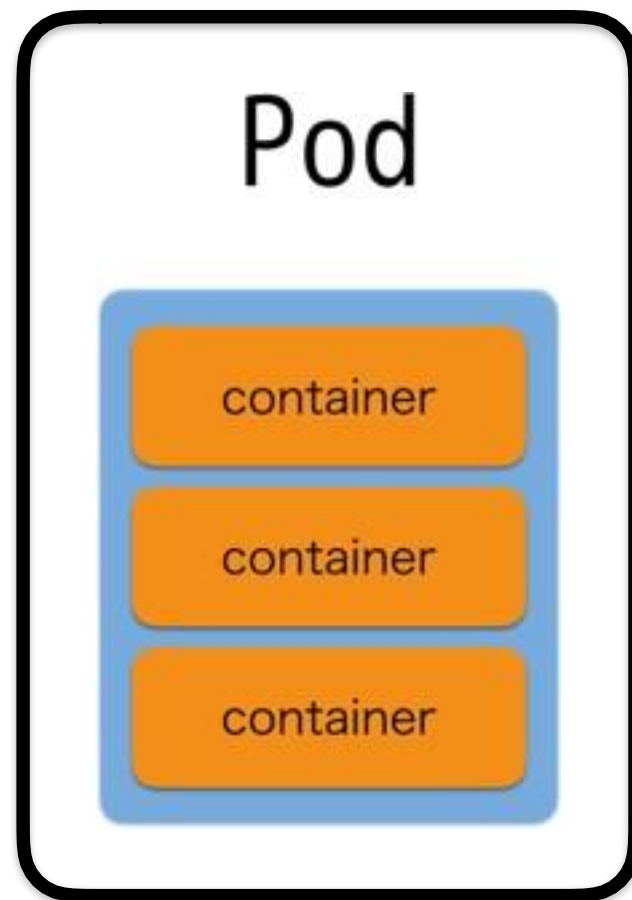
Node



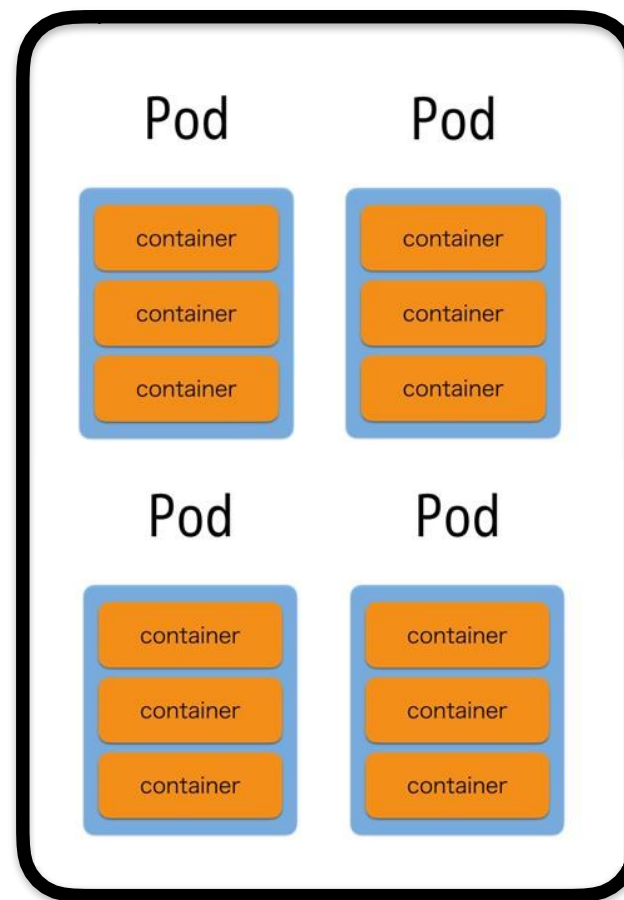
Node



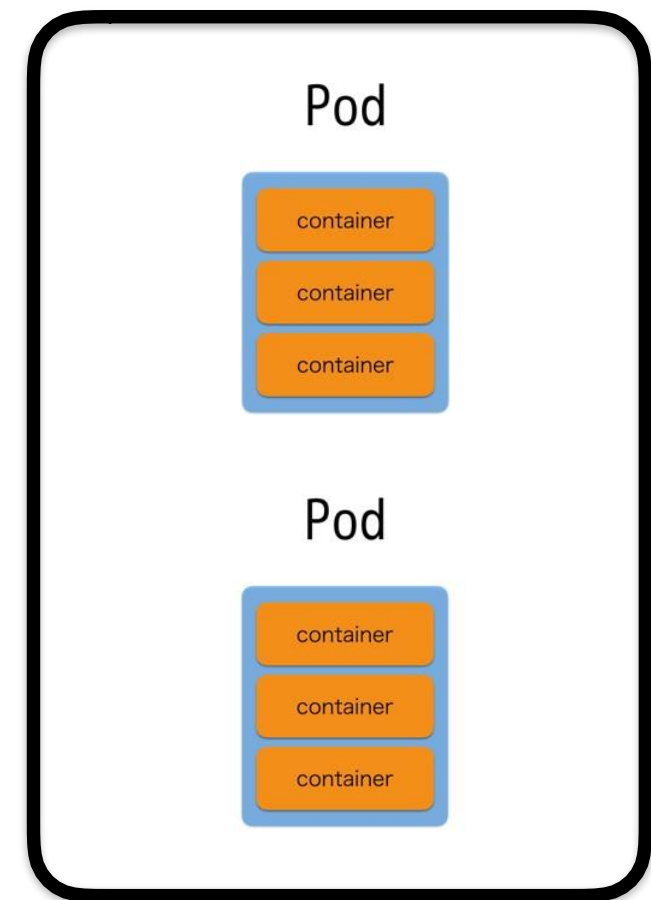
Node



Node

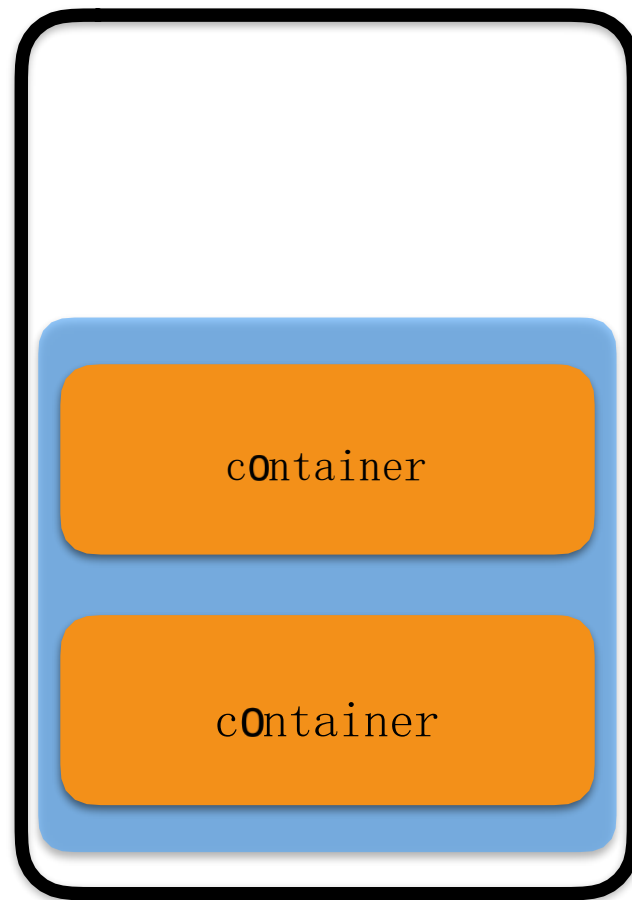


Node

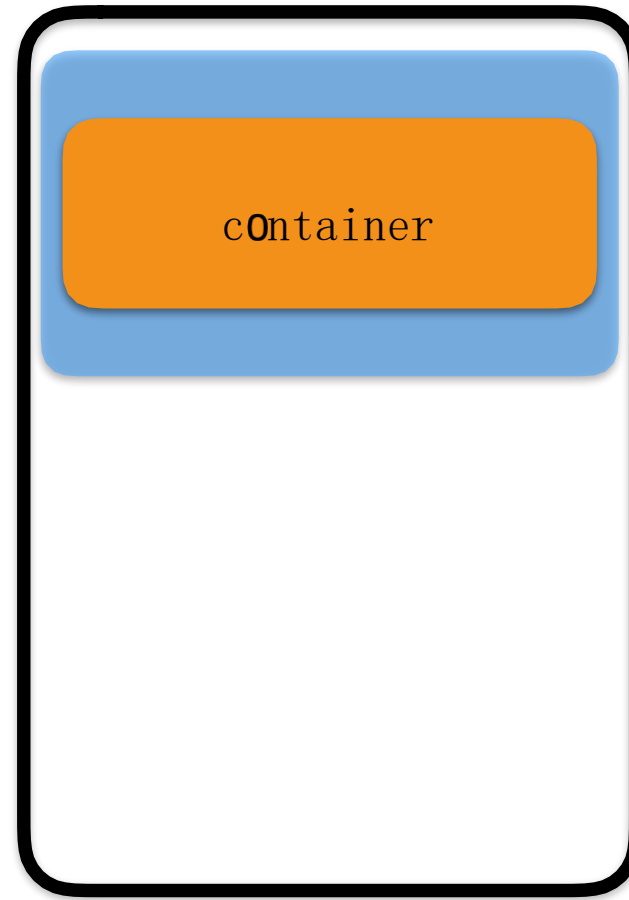


Depends on each node's resource availability
and each pod's resource requirements

Node



Node



Node

Node

This will NEVER happen

container

container

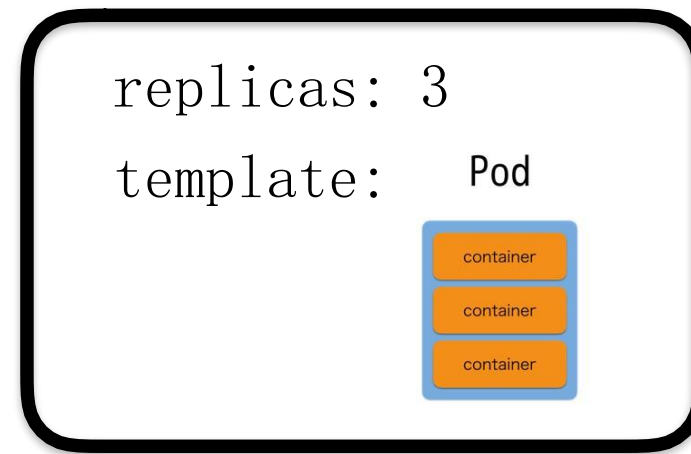
Miscellaneous

- Each pod has its own IP address
- Pods are expected to be stateless

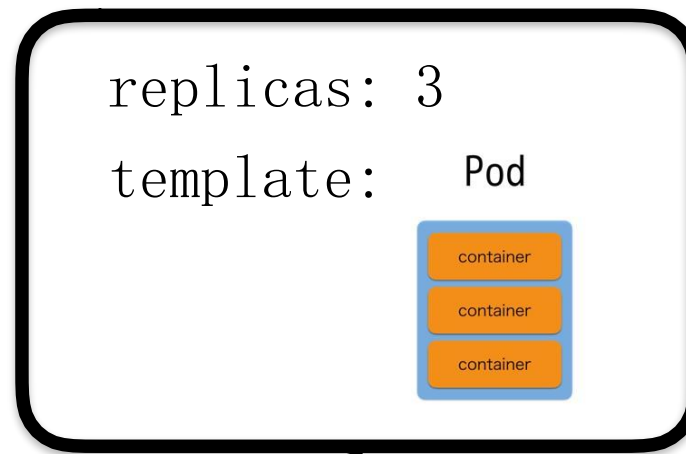
Replica Set

**Keepstrack of
Pod replicas**

Replica Set



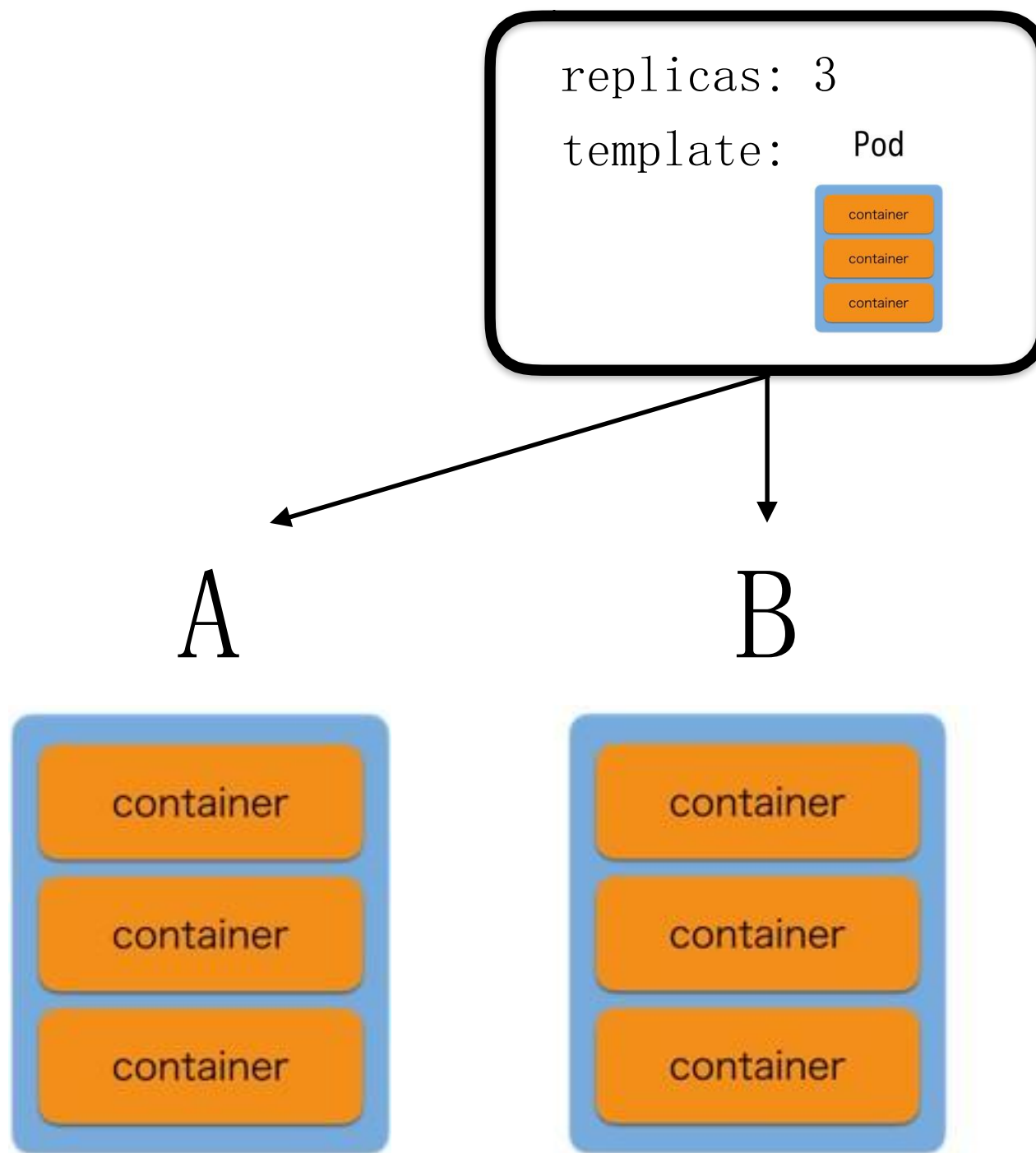
Replica Set



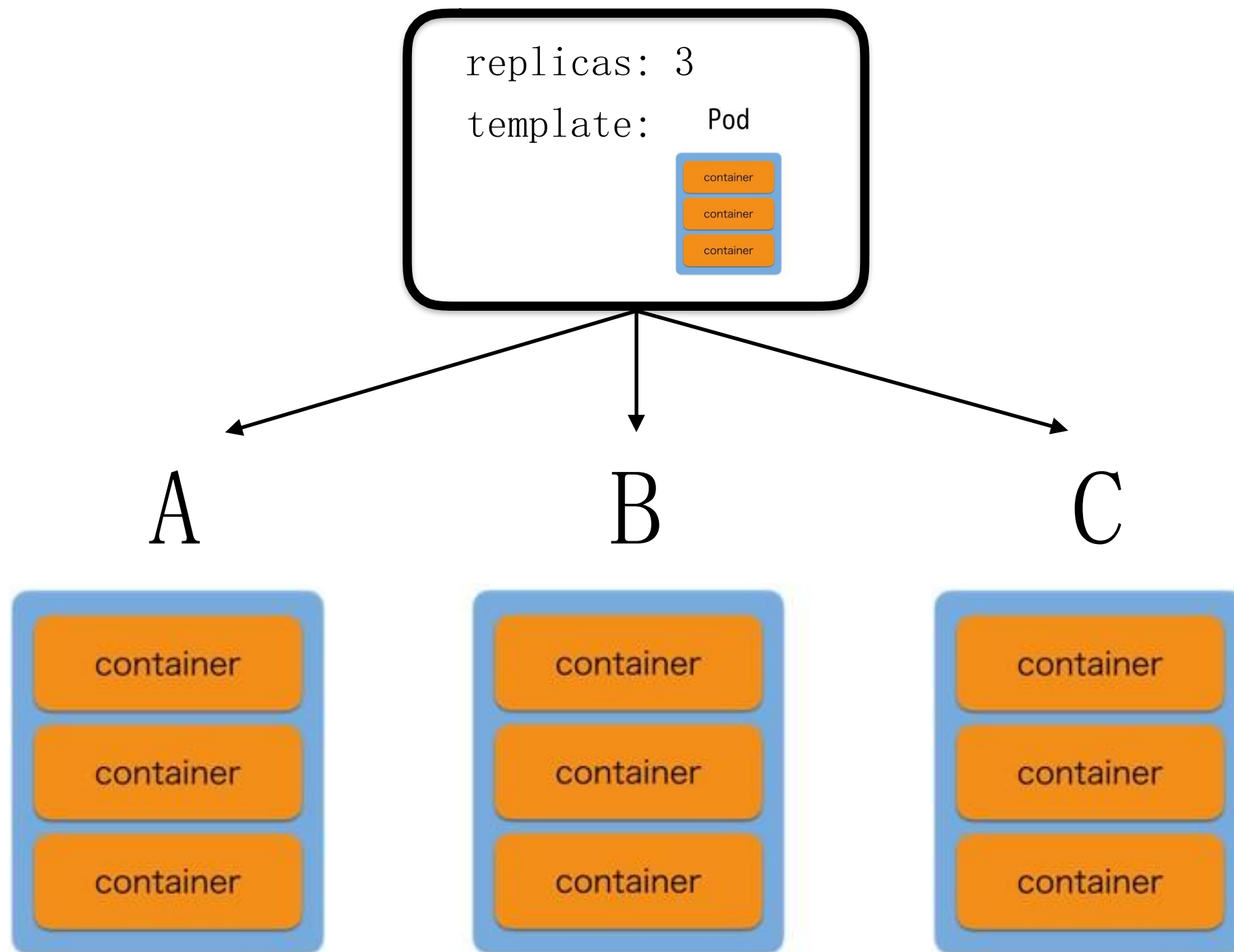
A



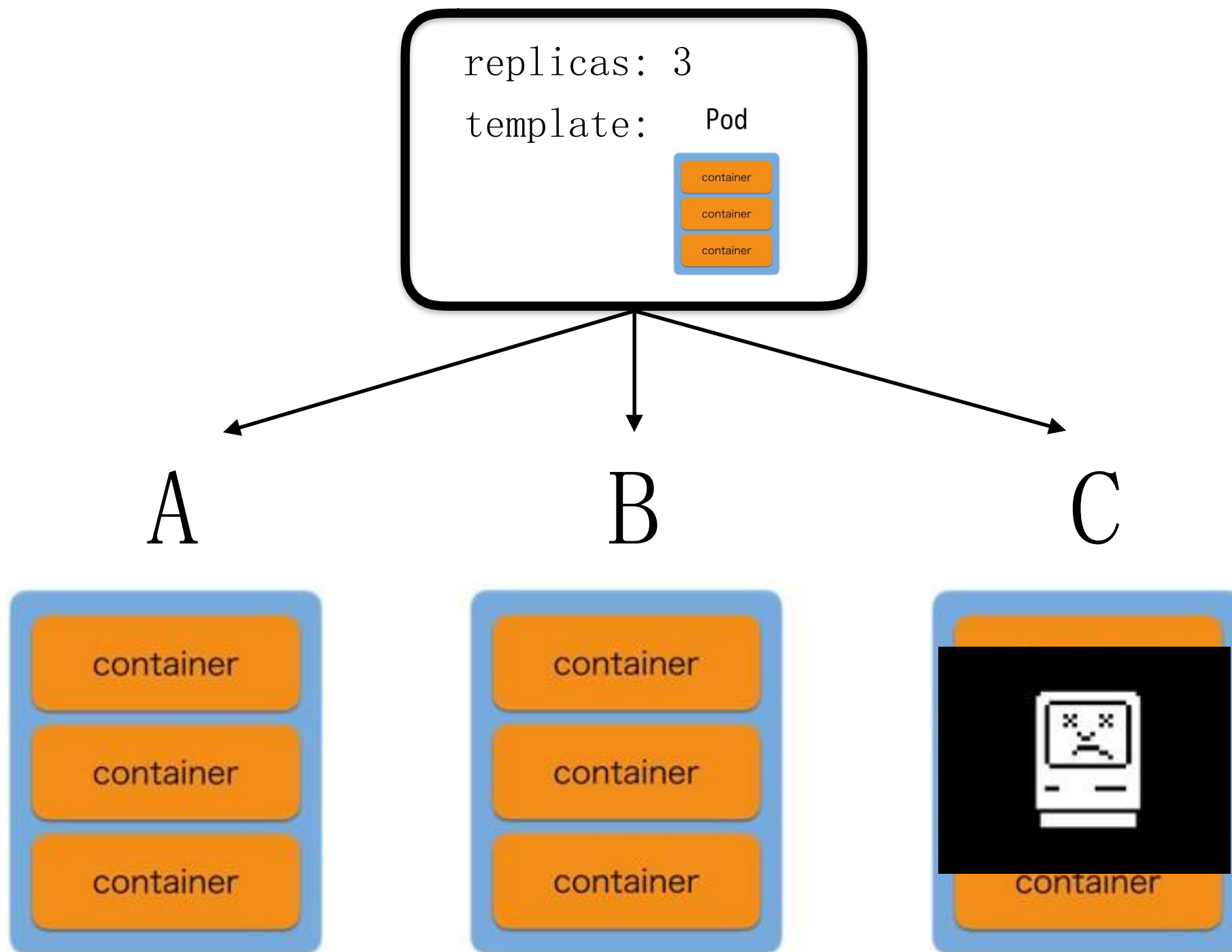
Replica Set



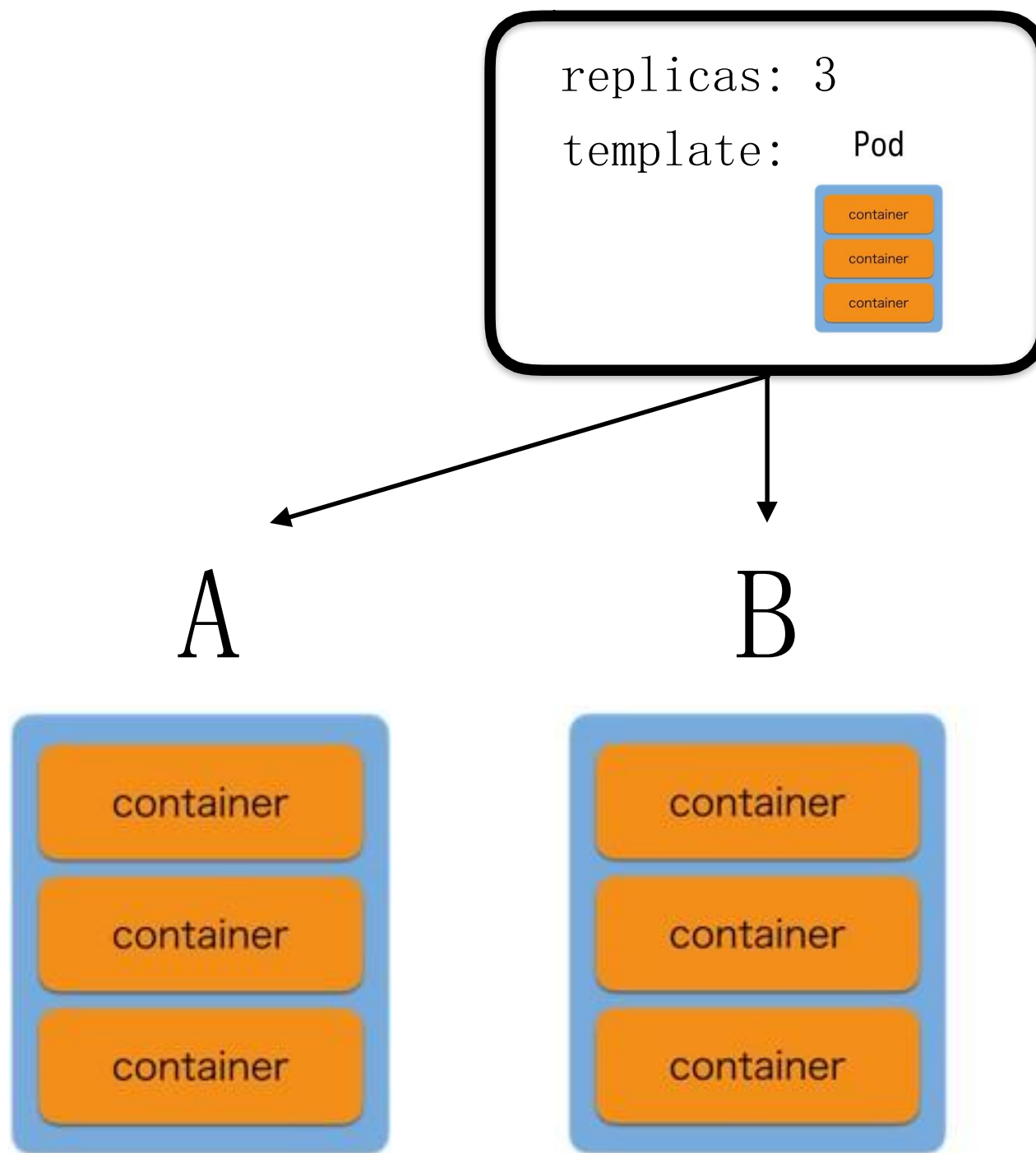
Replica Set



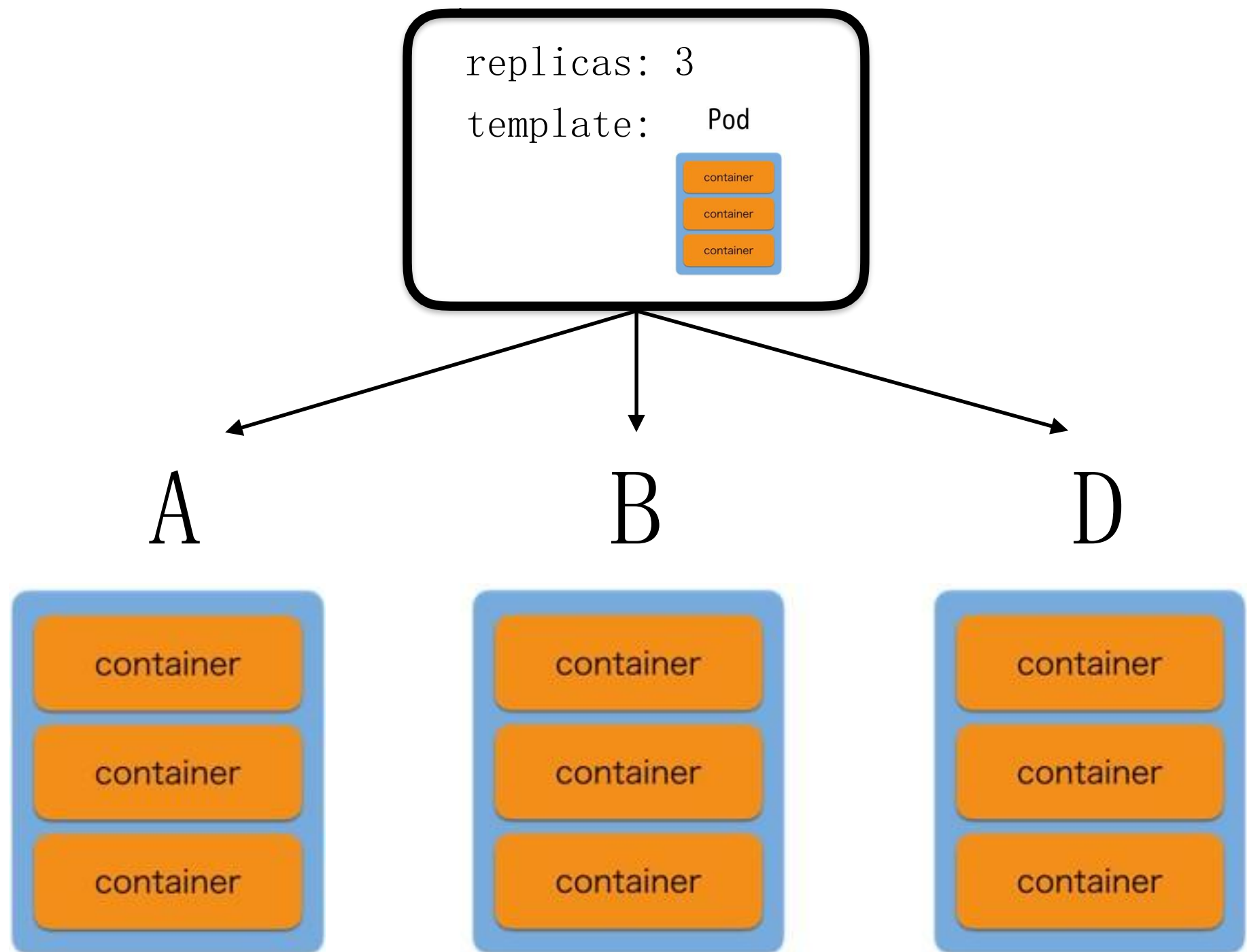
Replica Set



Replica Set



Replica Set



Deployment

**Manages Replica Set
state transitions**

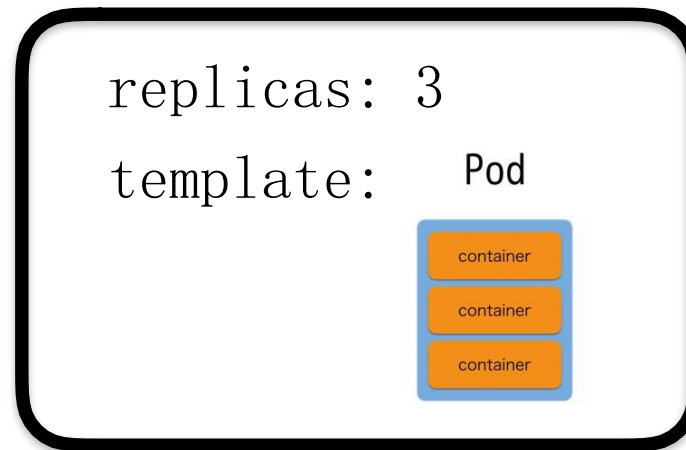
Deployment

replicas: 3

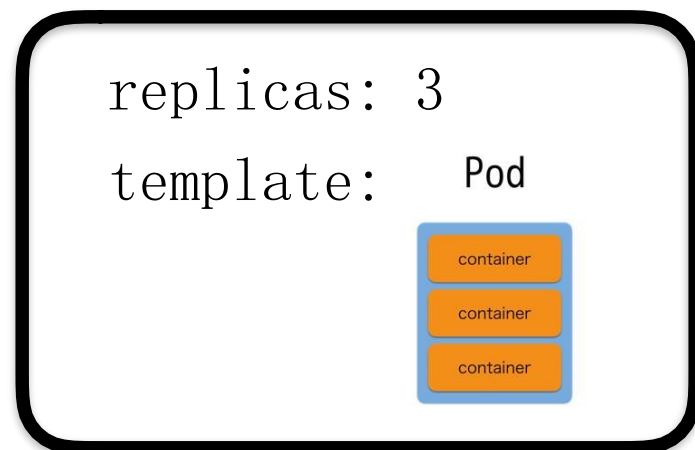
template: Pod



Deployment



Replica Set A



Deployment

Change some settings,
please!

replicas: 3
template: Pod

container
container
container

Replica Set A

replicas: 3
template: Pod

container
container
container

Deployment

Change some settings,
please!

replicas: 2
template: Pod

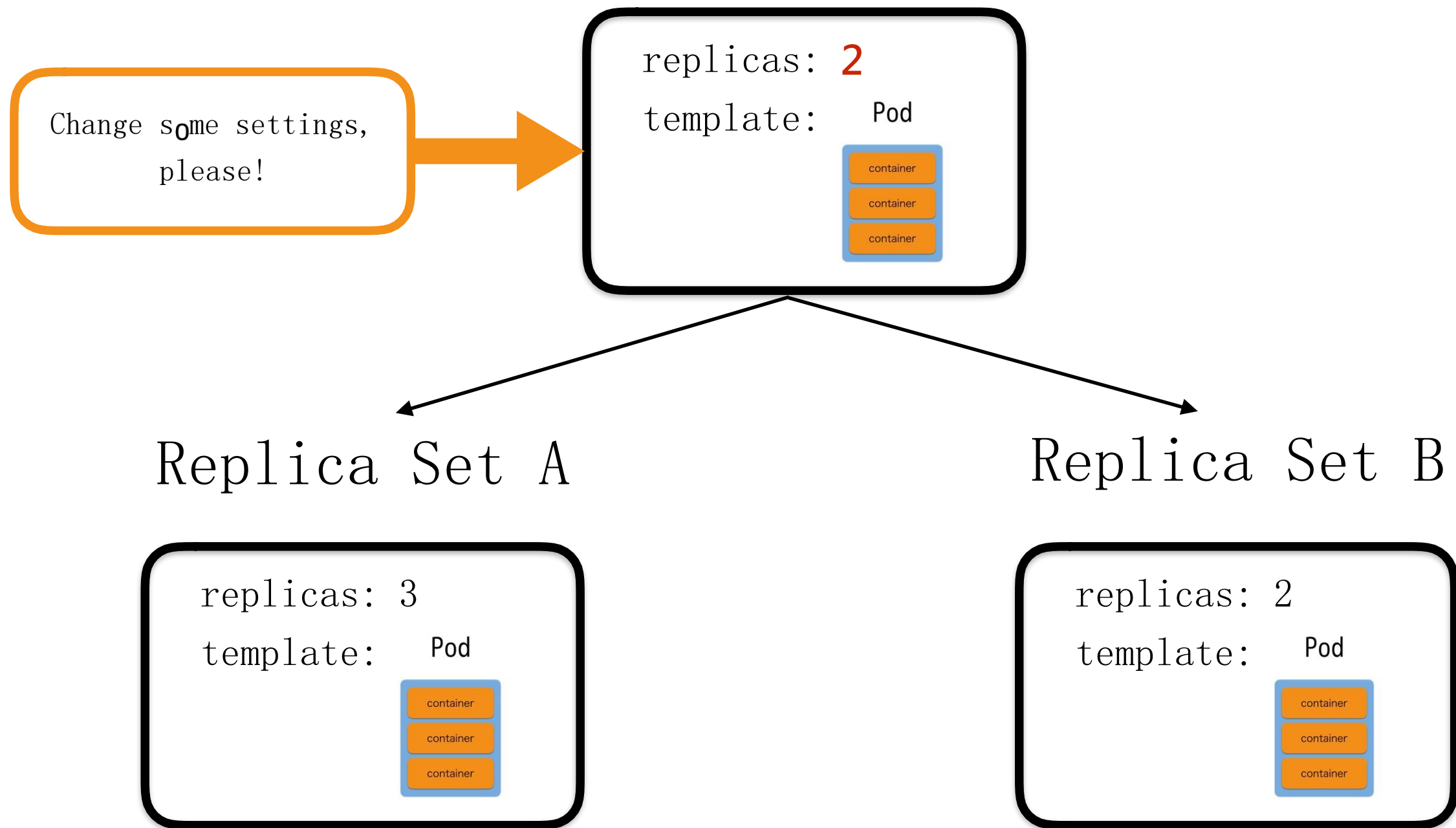


Replica Set A

replicas: 3
template: Pod



Deployment



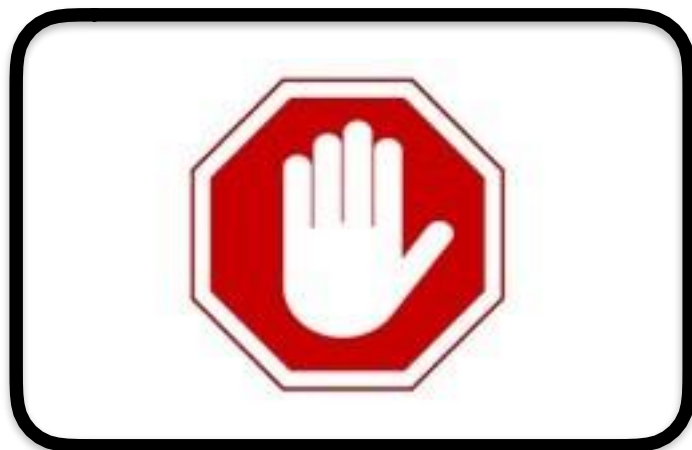
Deployment

Change some settings,
please!

replicas: 2
template: Pod



Replica Set A



Replica Set B

replicas: 2
template: Pod



Deployment

Change some settings,
please!

replicas: 2
template: Pod



Replica Set B

replicas: 2
template: Pod



Keeps track of
state change history

Deployment

```
replicas: 3
```

```
template: Pod
```



Deployment

replicas: 3

template: Pod



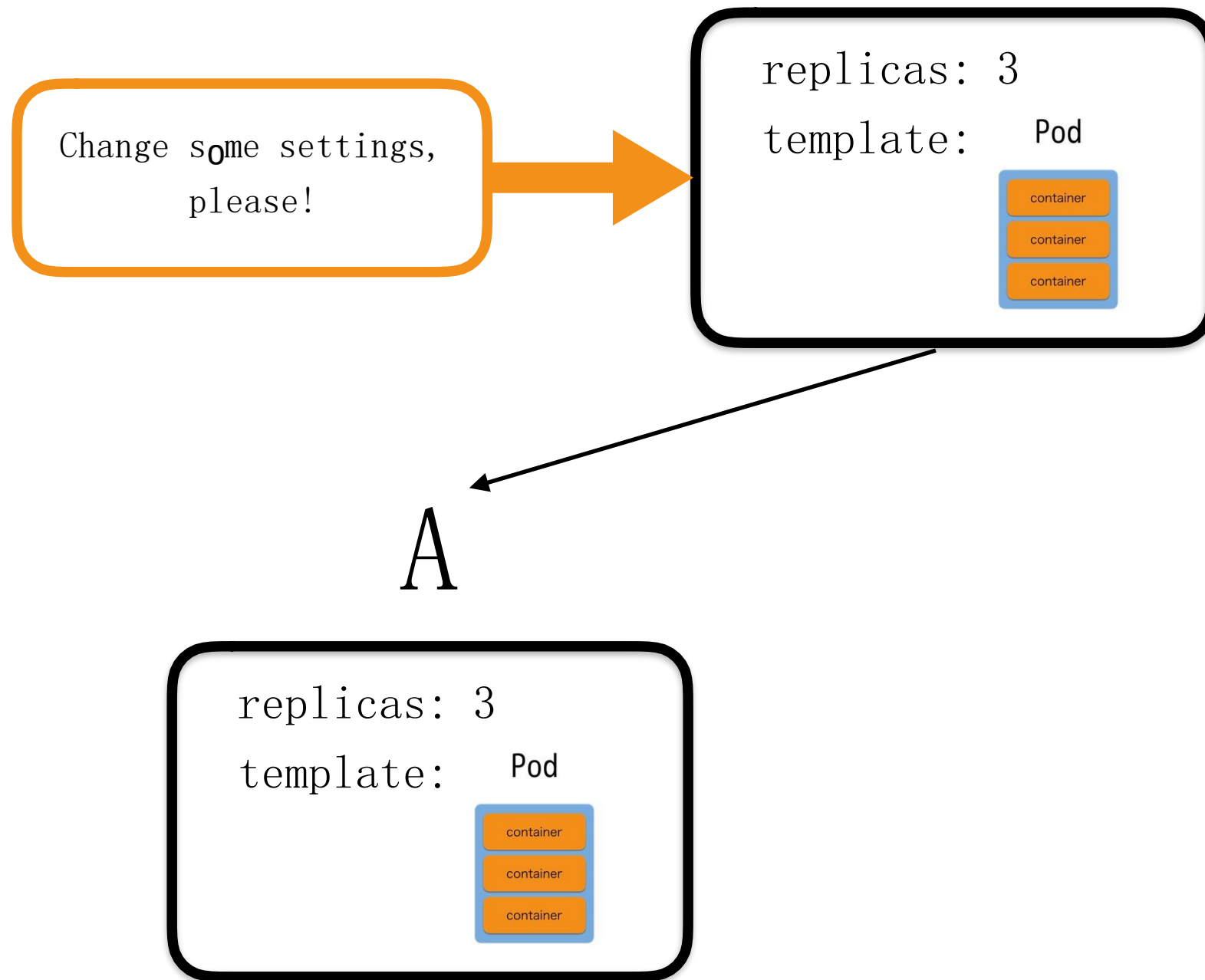
A

replicas: 3

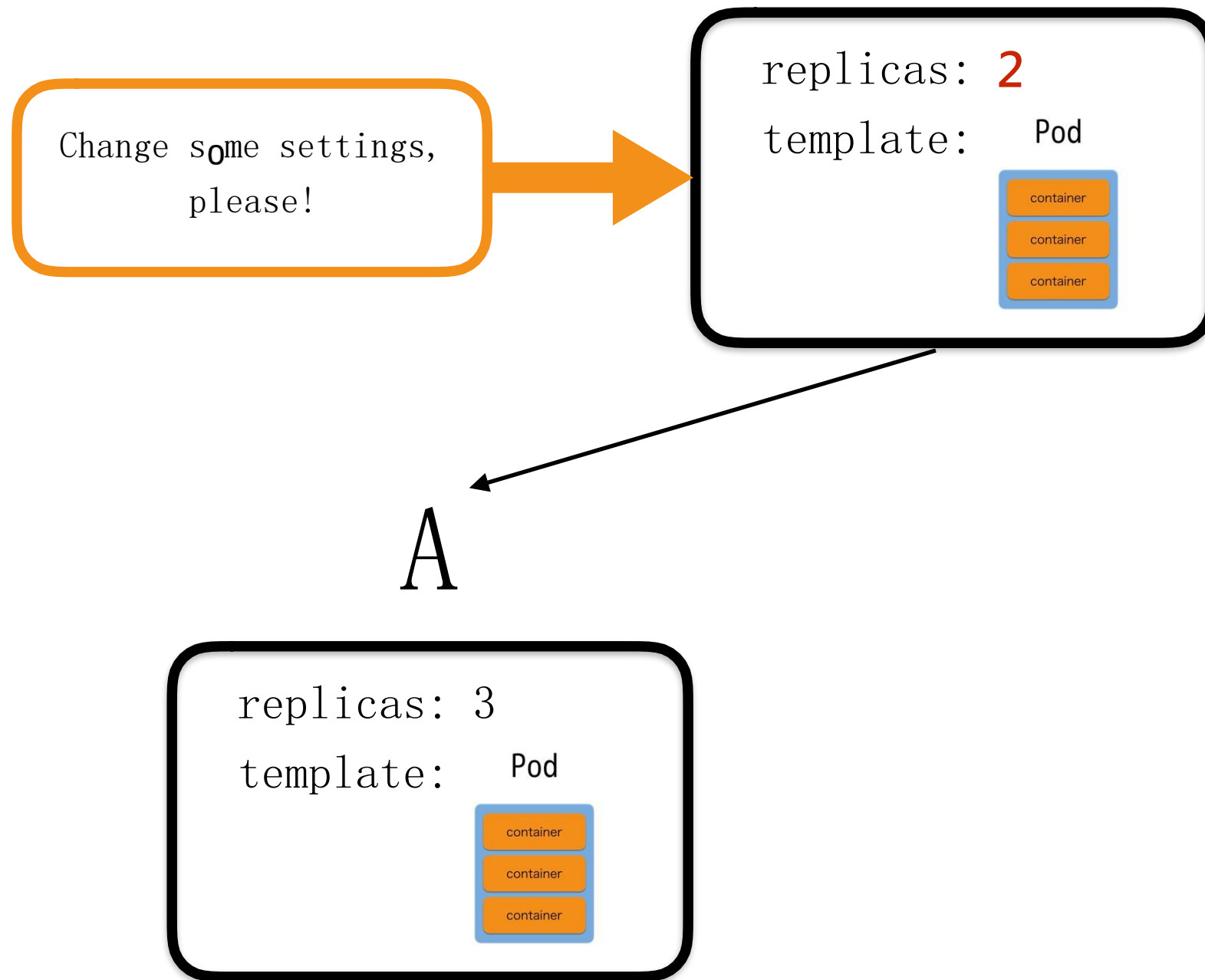
template: Pod



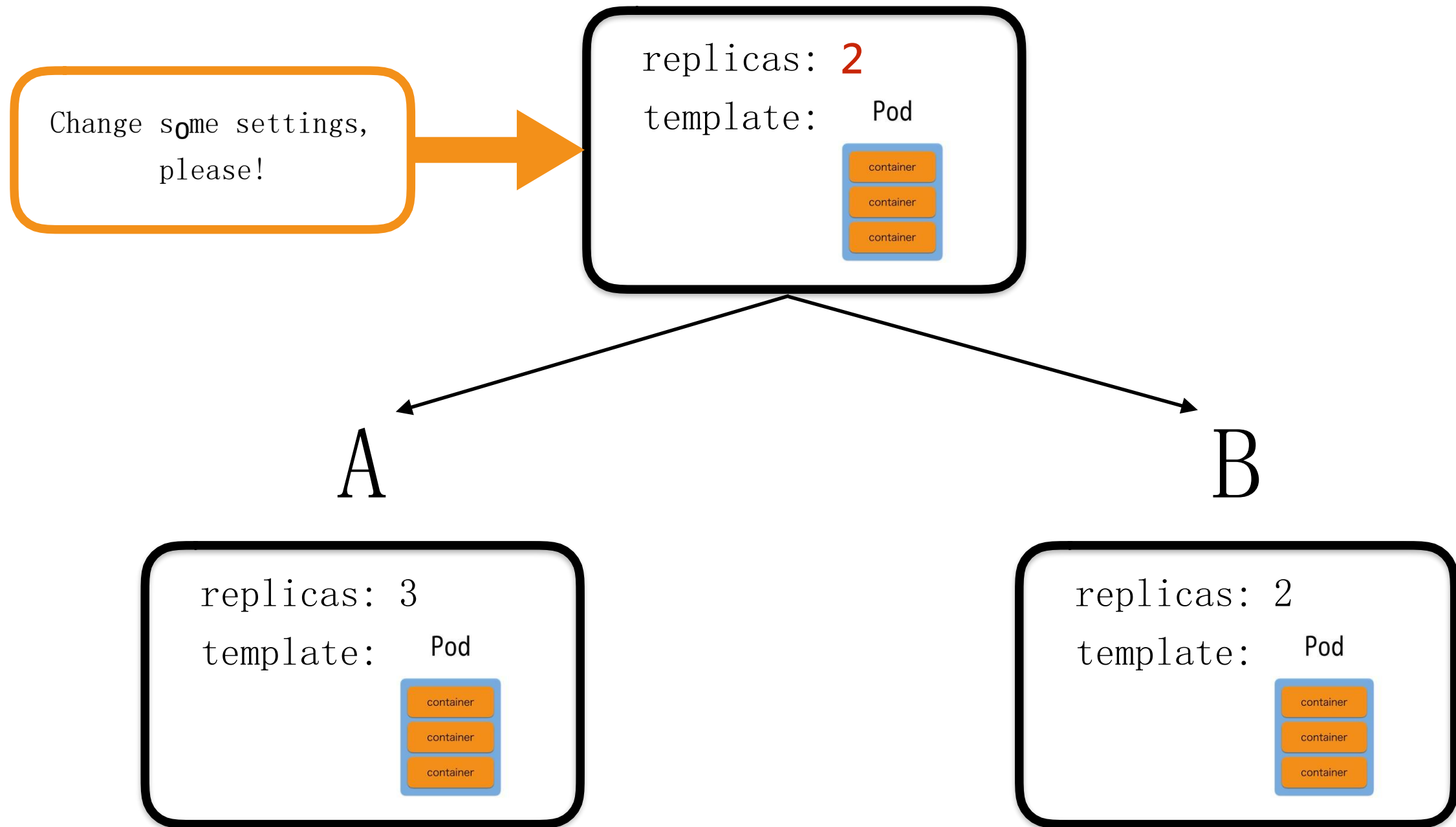
Deployment



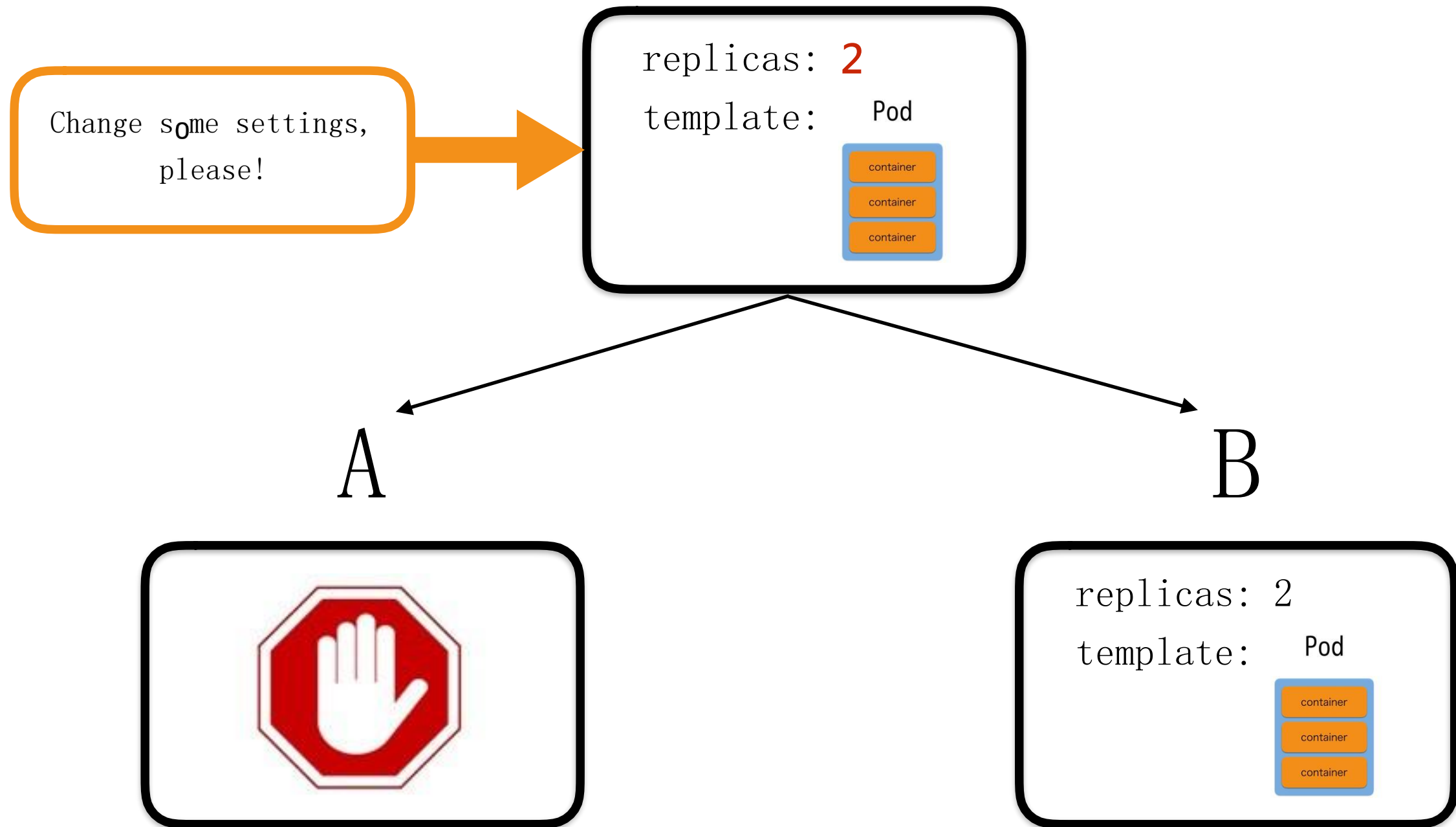
Deployment



Deployment



Deployment



Deployment

Change some settings,
please!

replicas: 2
template: Pod



B

replicas: 2
template: Pod



Deployment

Oh wait, I actually
didn't want to do that...

replicas: 2
template: Pod



B

replicas: 2
template: Pod



Deployment

Oh wait, I actually
didn't want to do that...

replicas: **3**
template: Pod



B

replicas: 2
template: Pod



Deployment

Oh wait, I actually
didn't want to do that...

replicas: 3
template: Pod



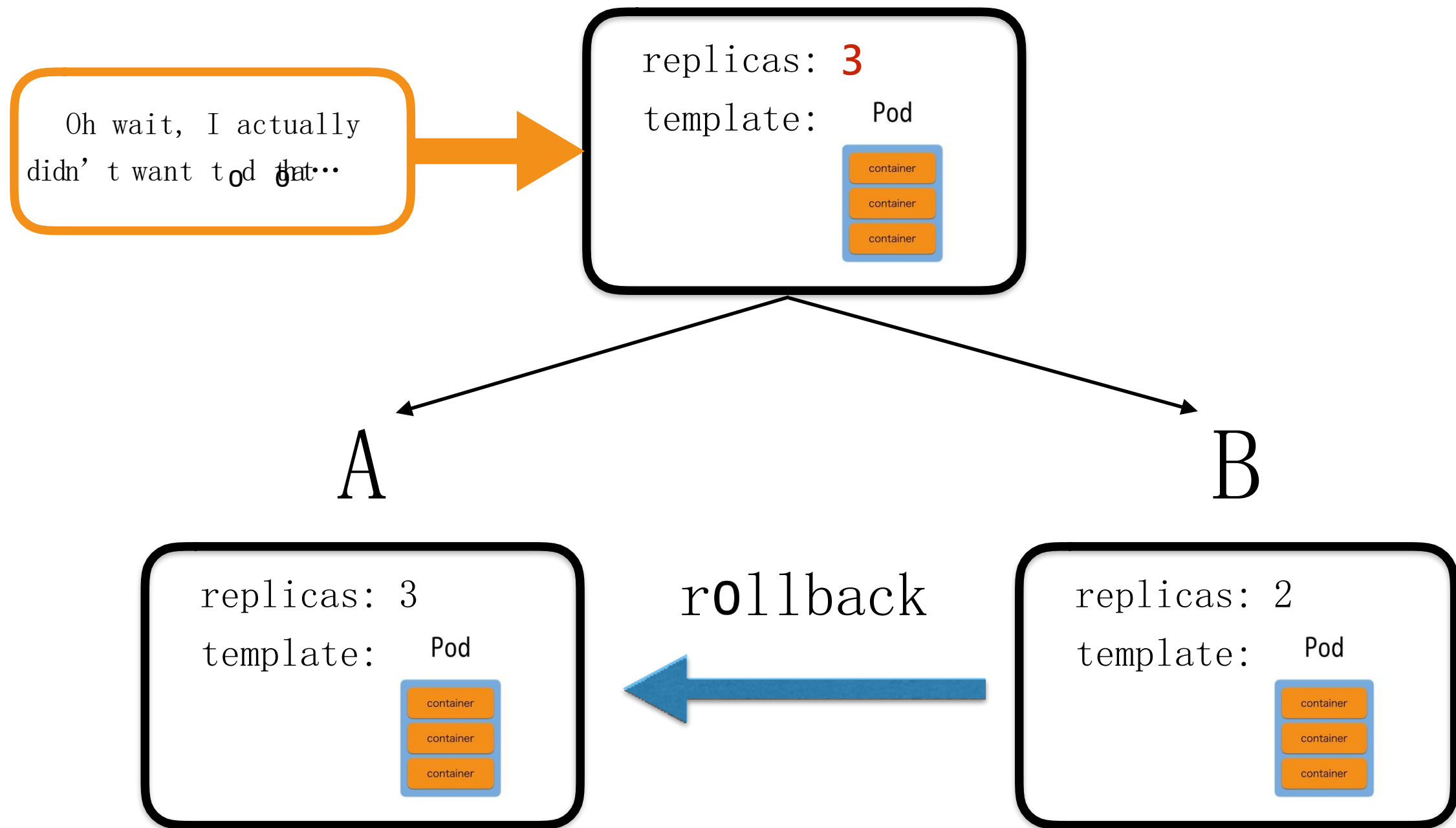
B

rollback

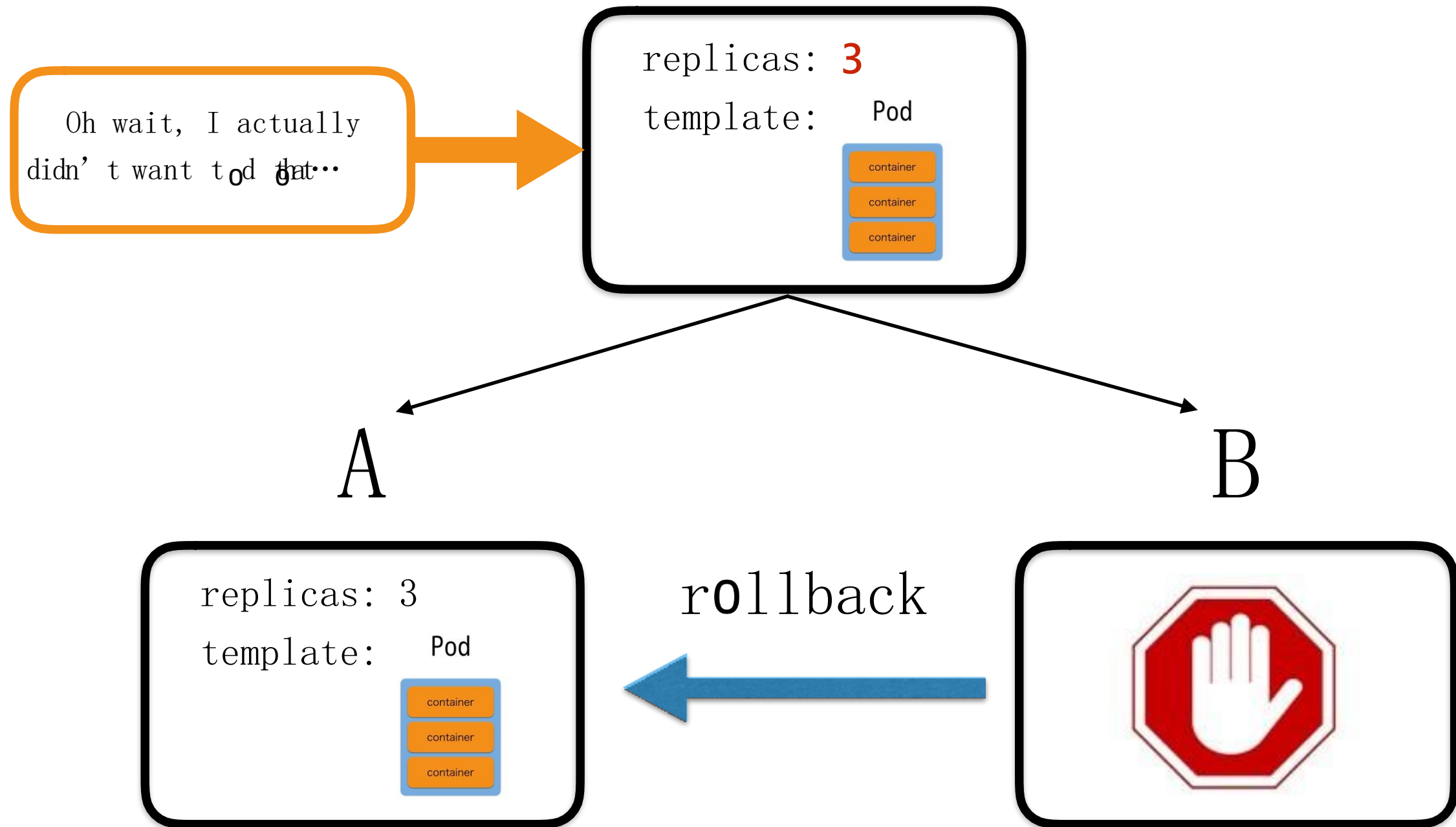
replicas: 2
template: Pod



Deployment



Deployment



Deployment

Oh wait, I actually
didn't want to do that...

replicas: **3**
template: Pod



A

replicas: 3
template: Pod



Services

Logical set of Pods

(and ways to access them)

Raw Pod Access

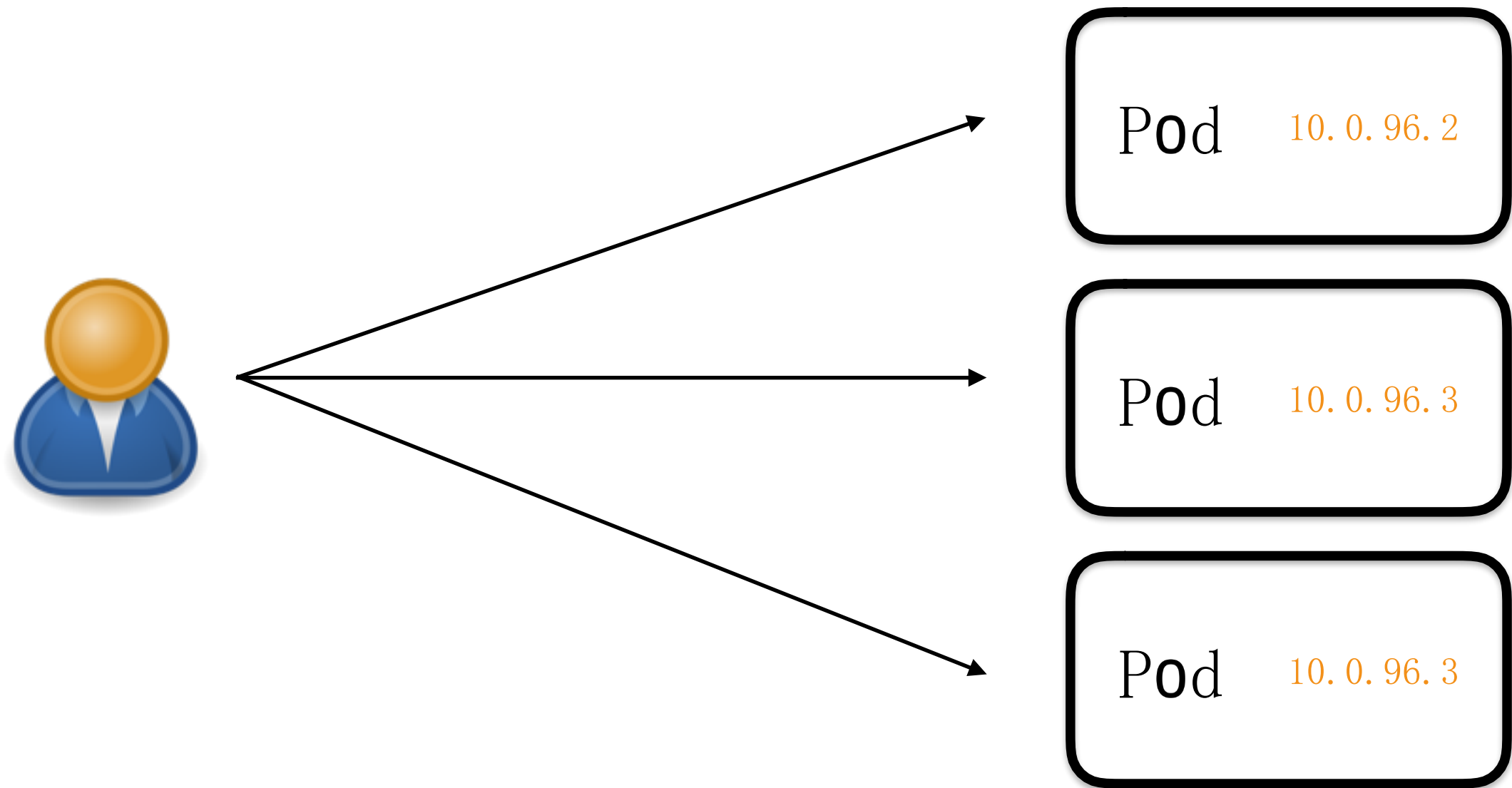


Pod 10.0.96.2

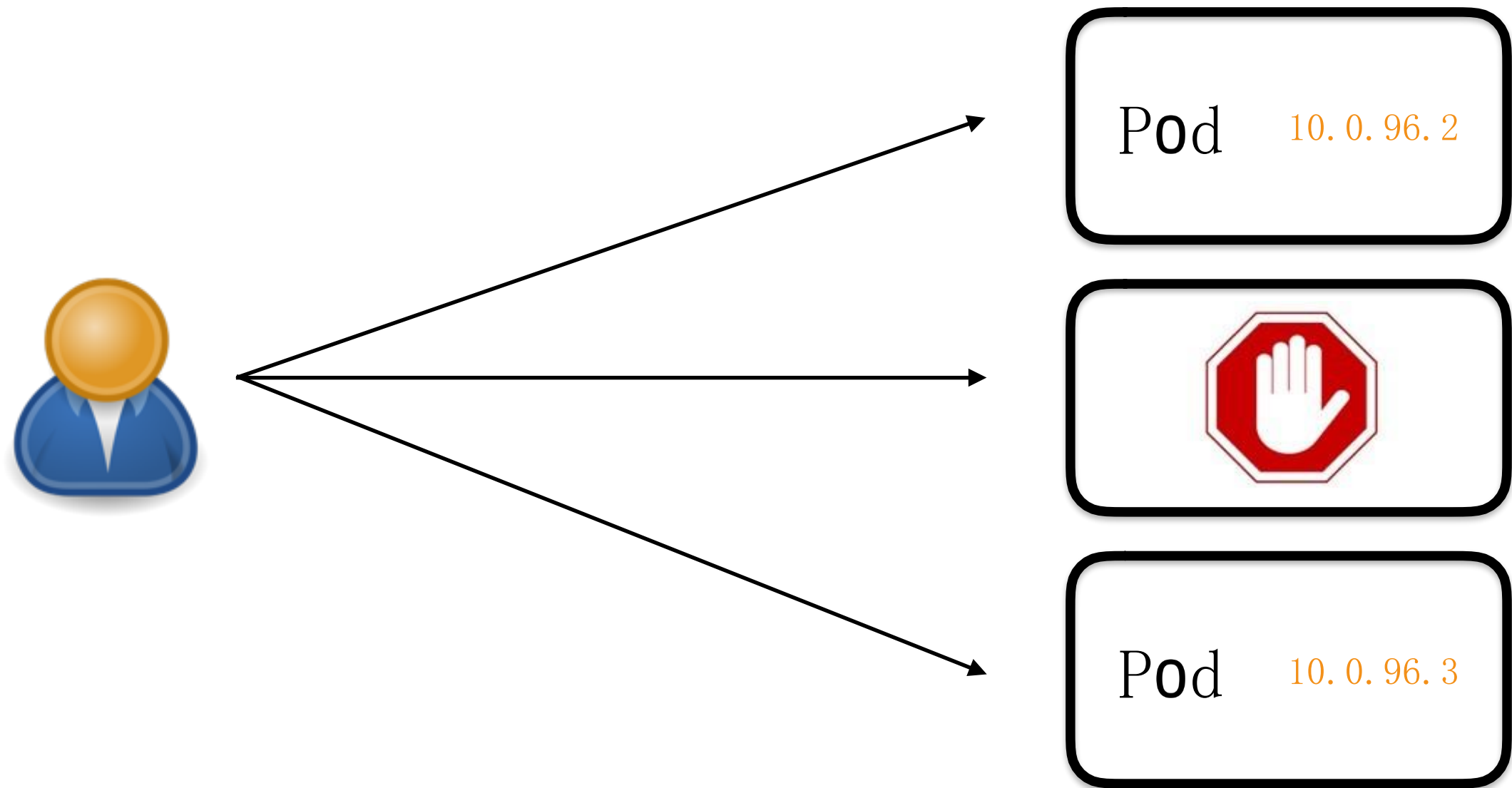
Pod 10.0.96.3

Pod 10.0.96.3

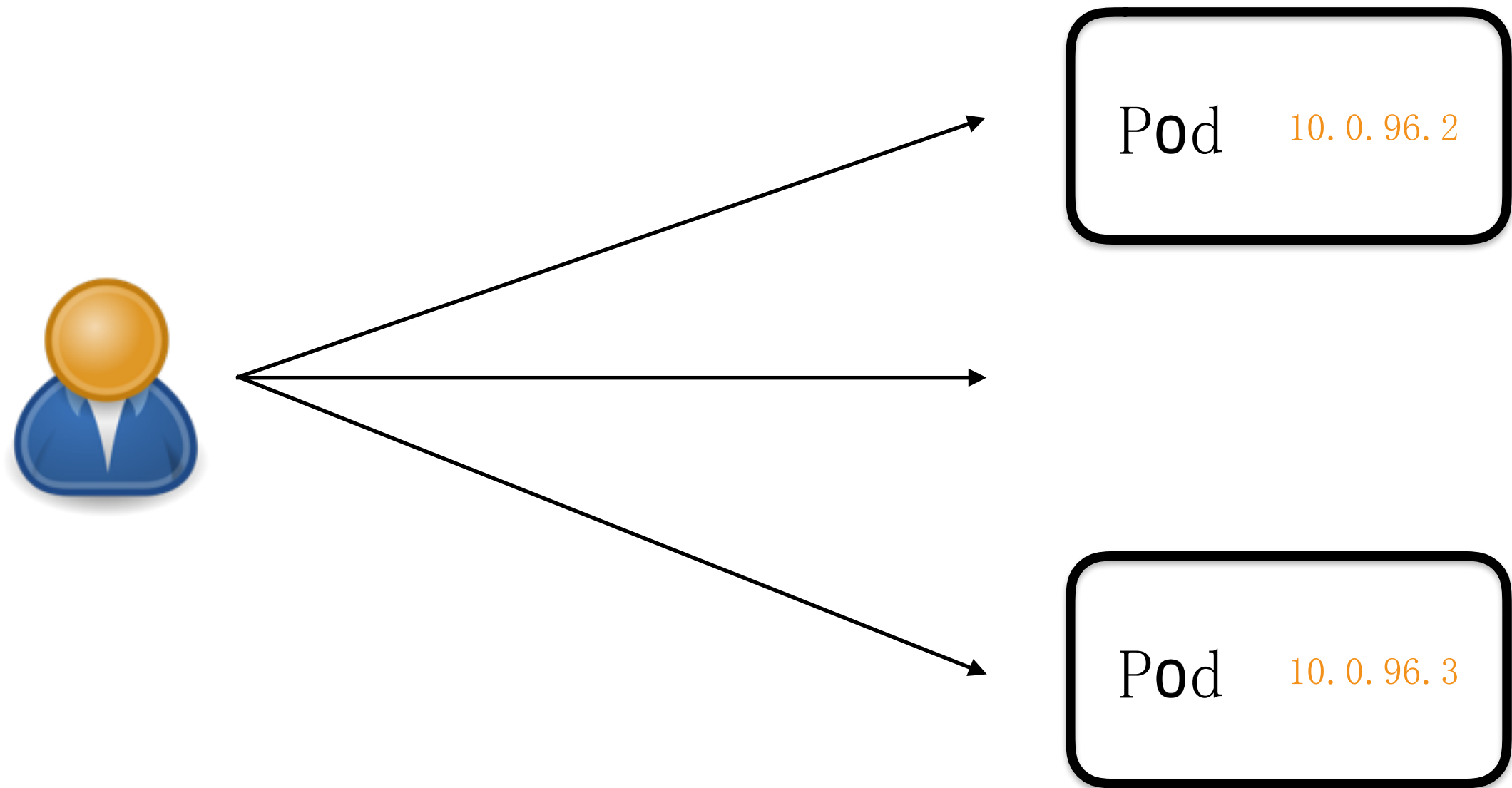
Raw Pod Access



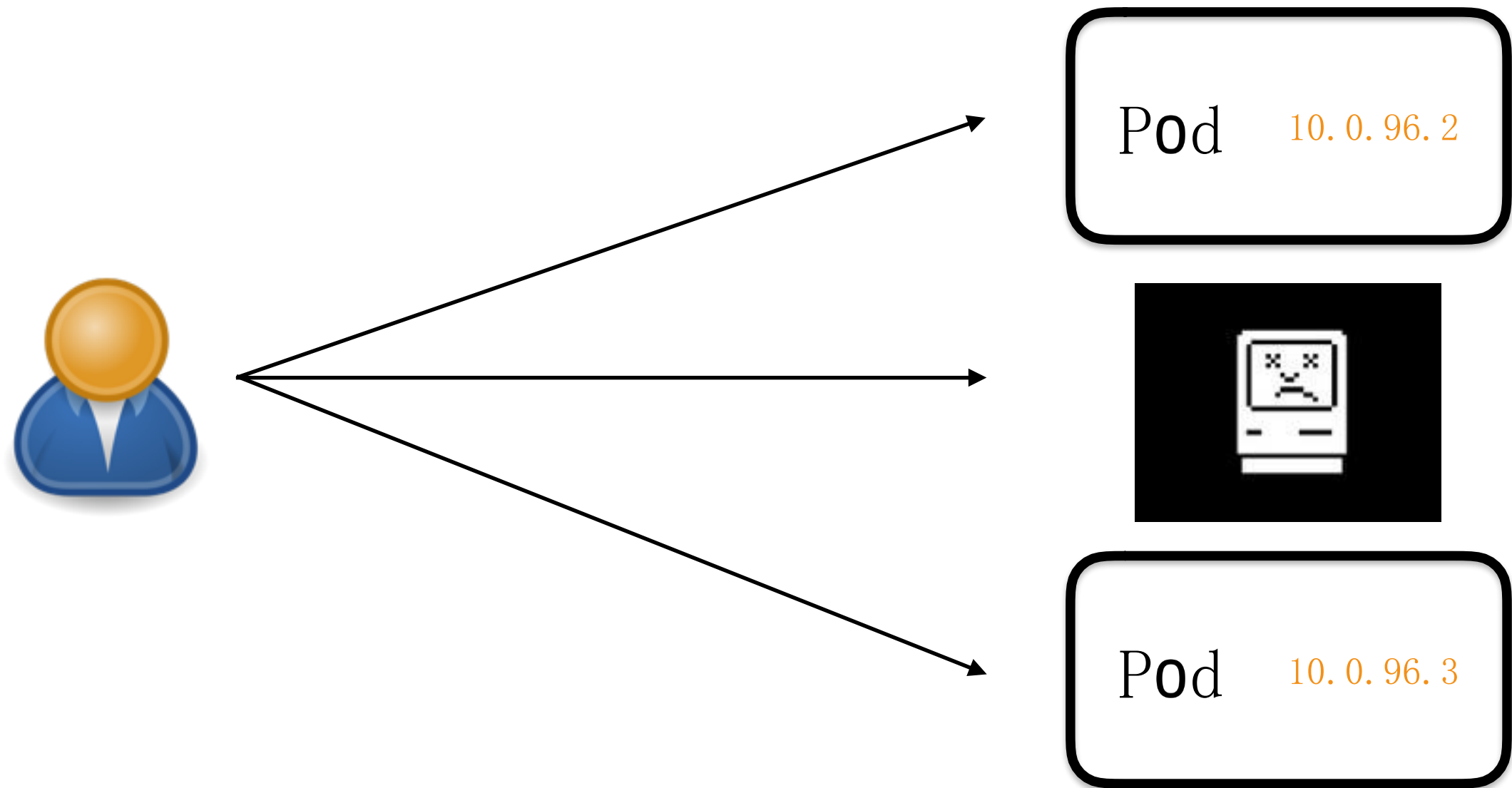
Raw Pod Access



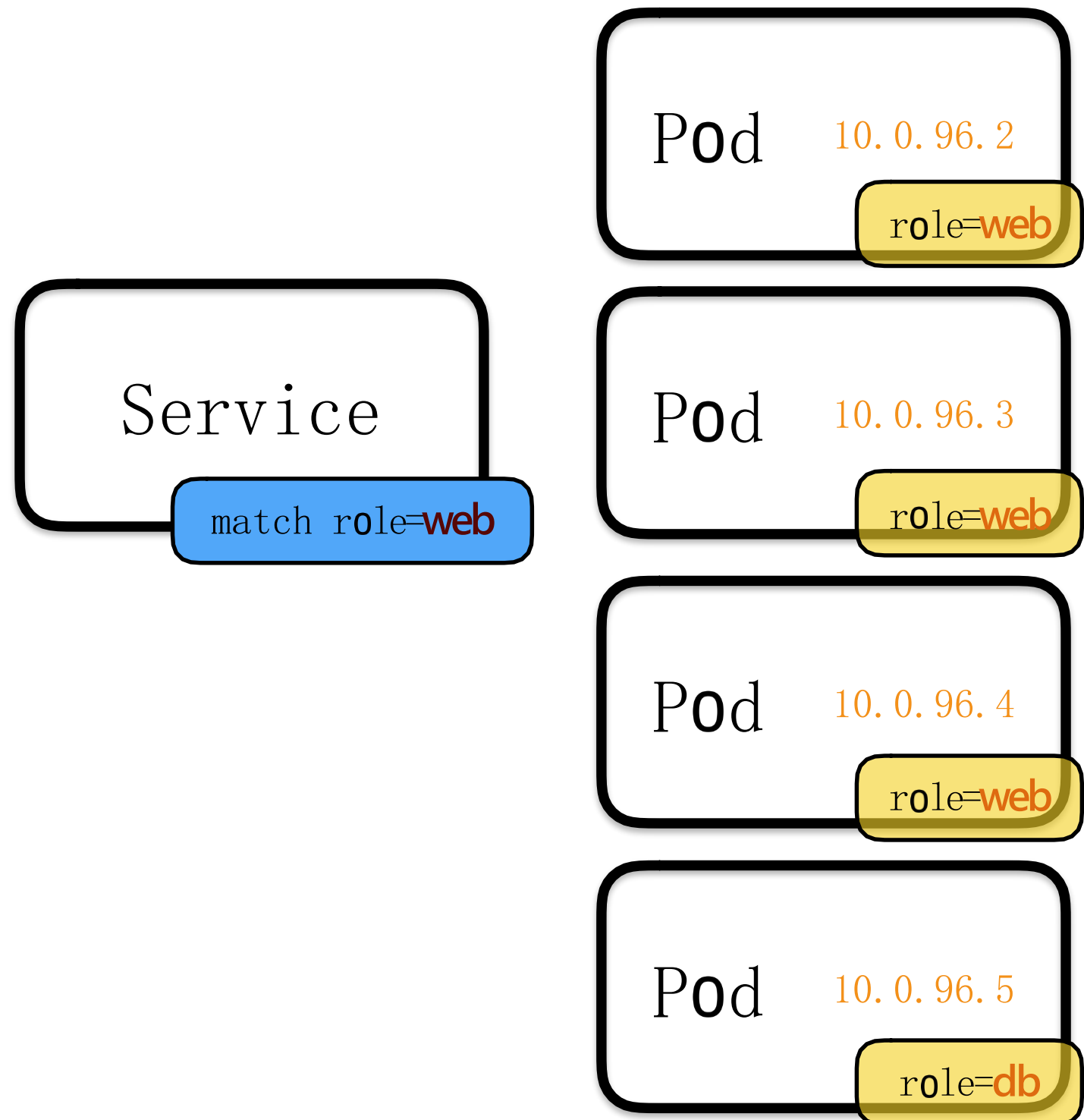
Raw Pod Access



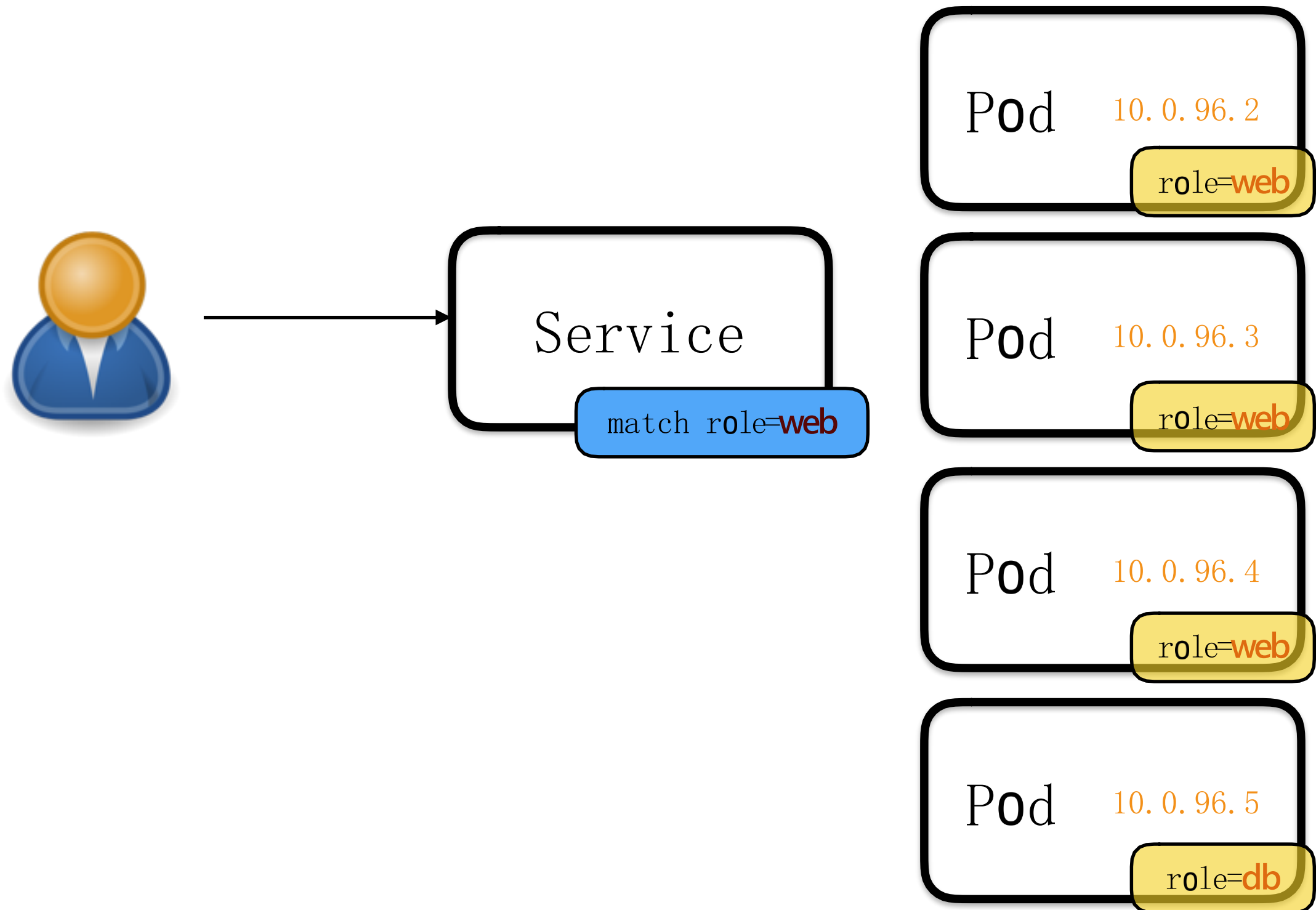
Raw Pod Access



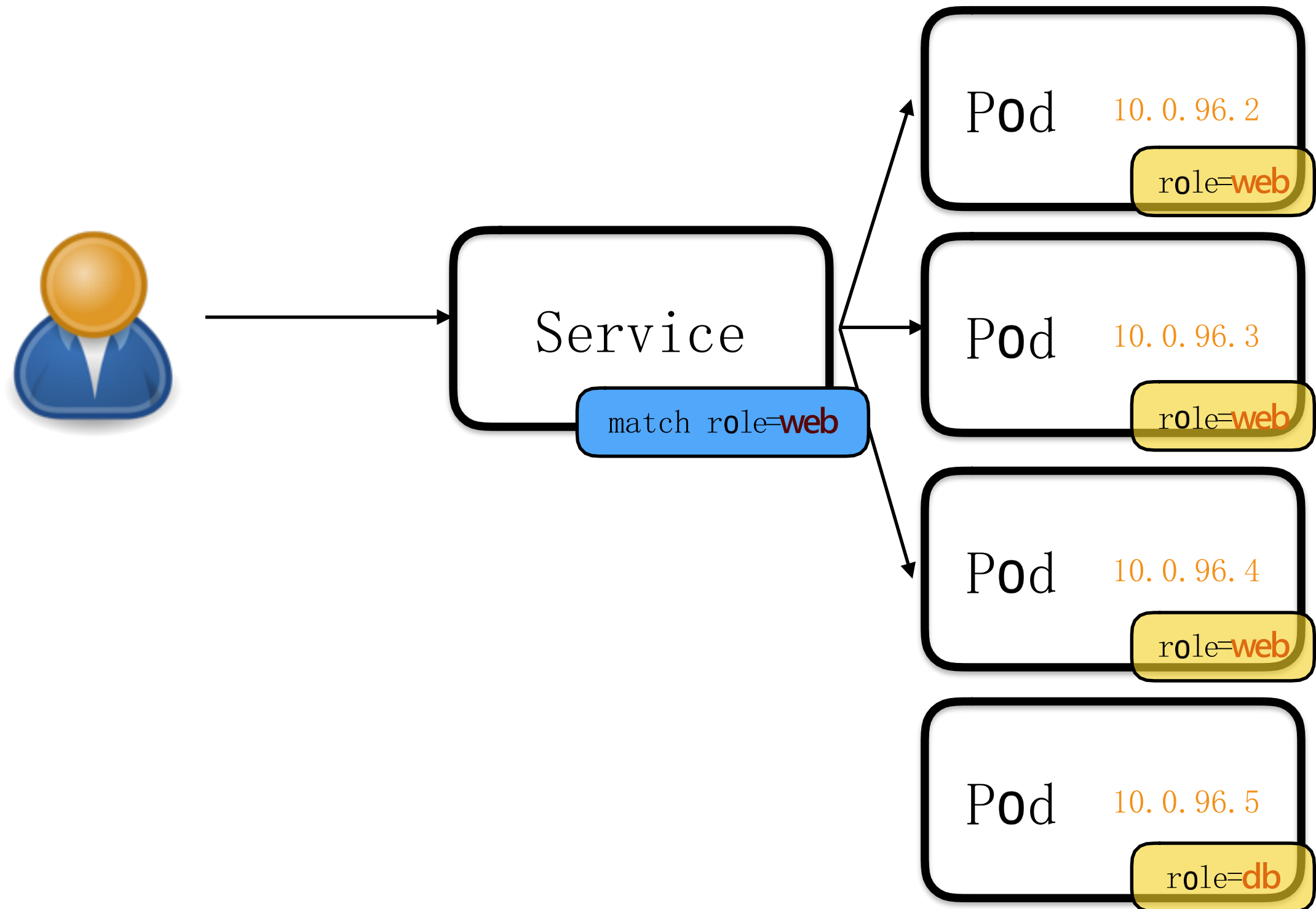
Access Via Service



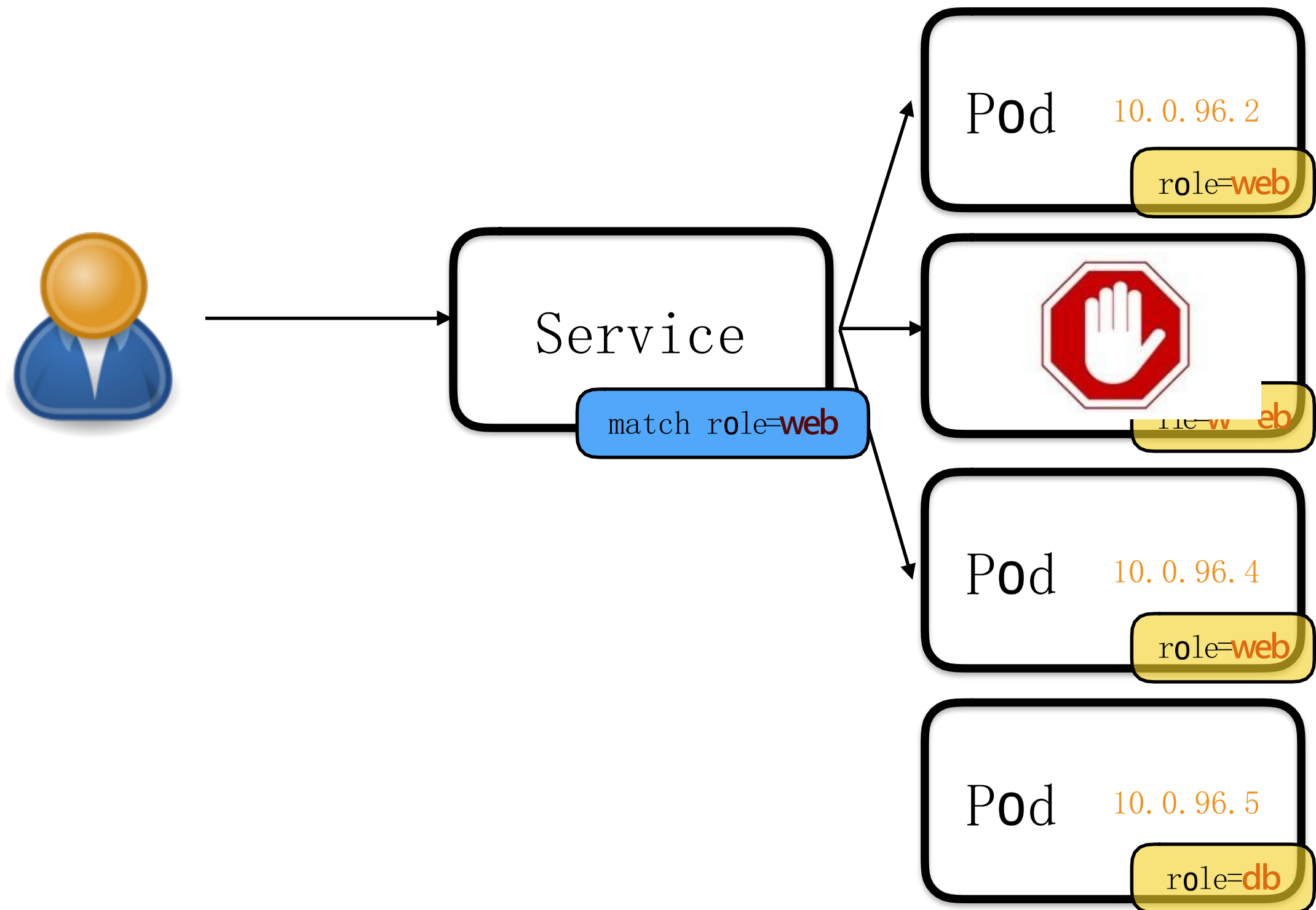
Access Via Service



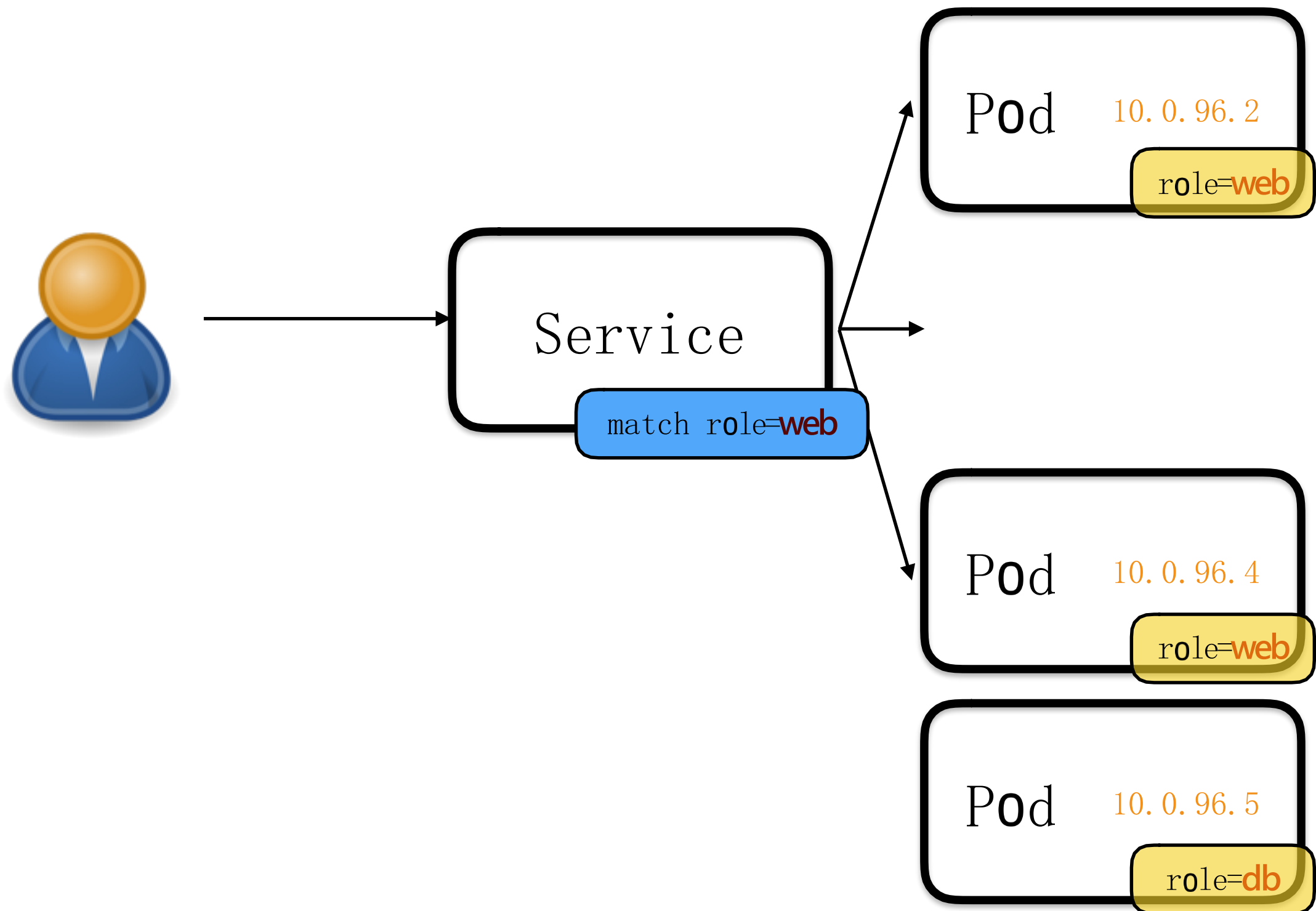
Access Via Service



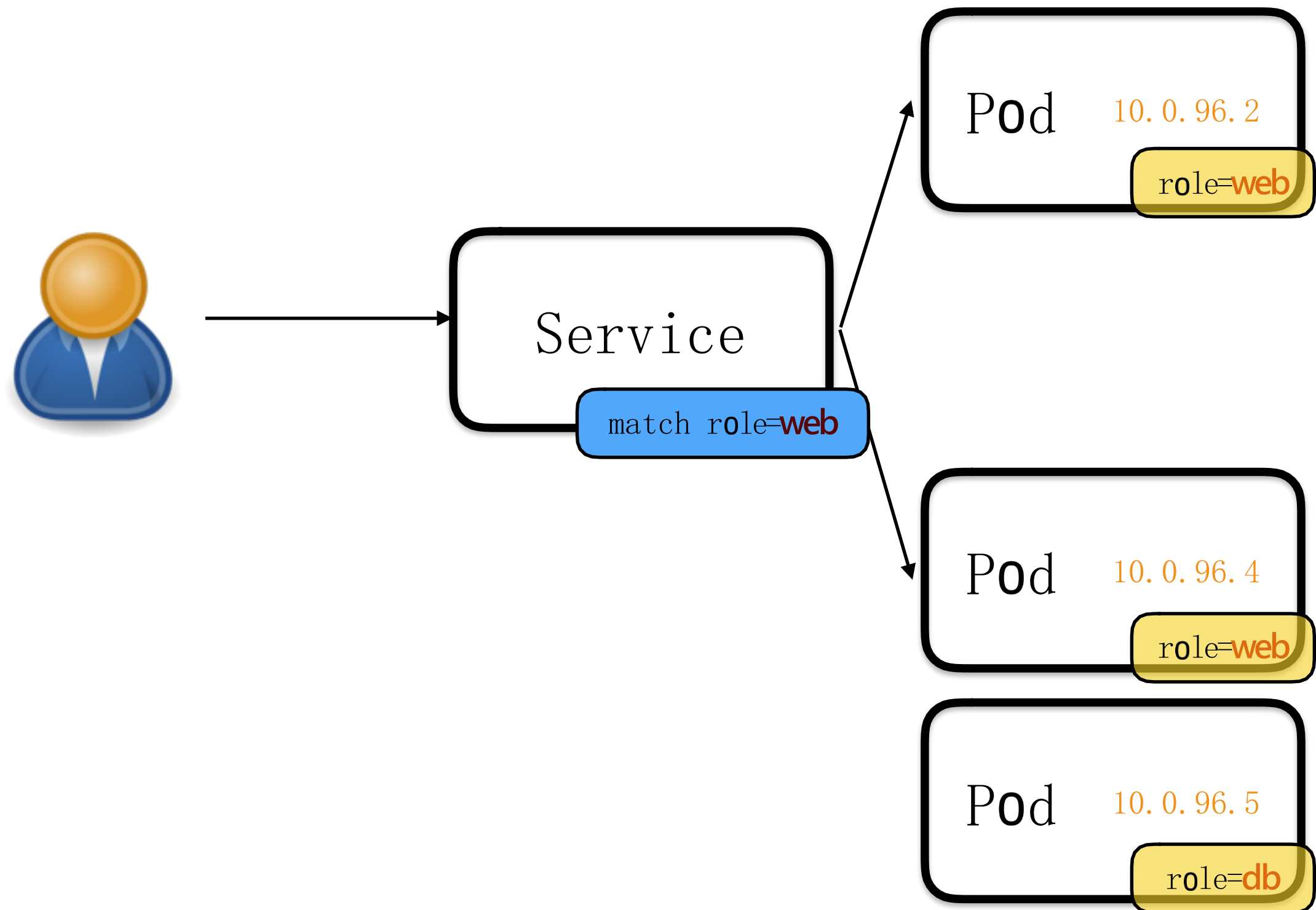
Access Via Service



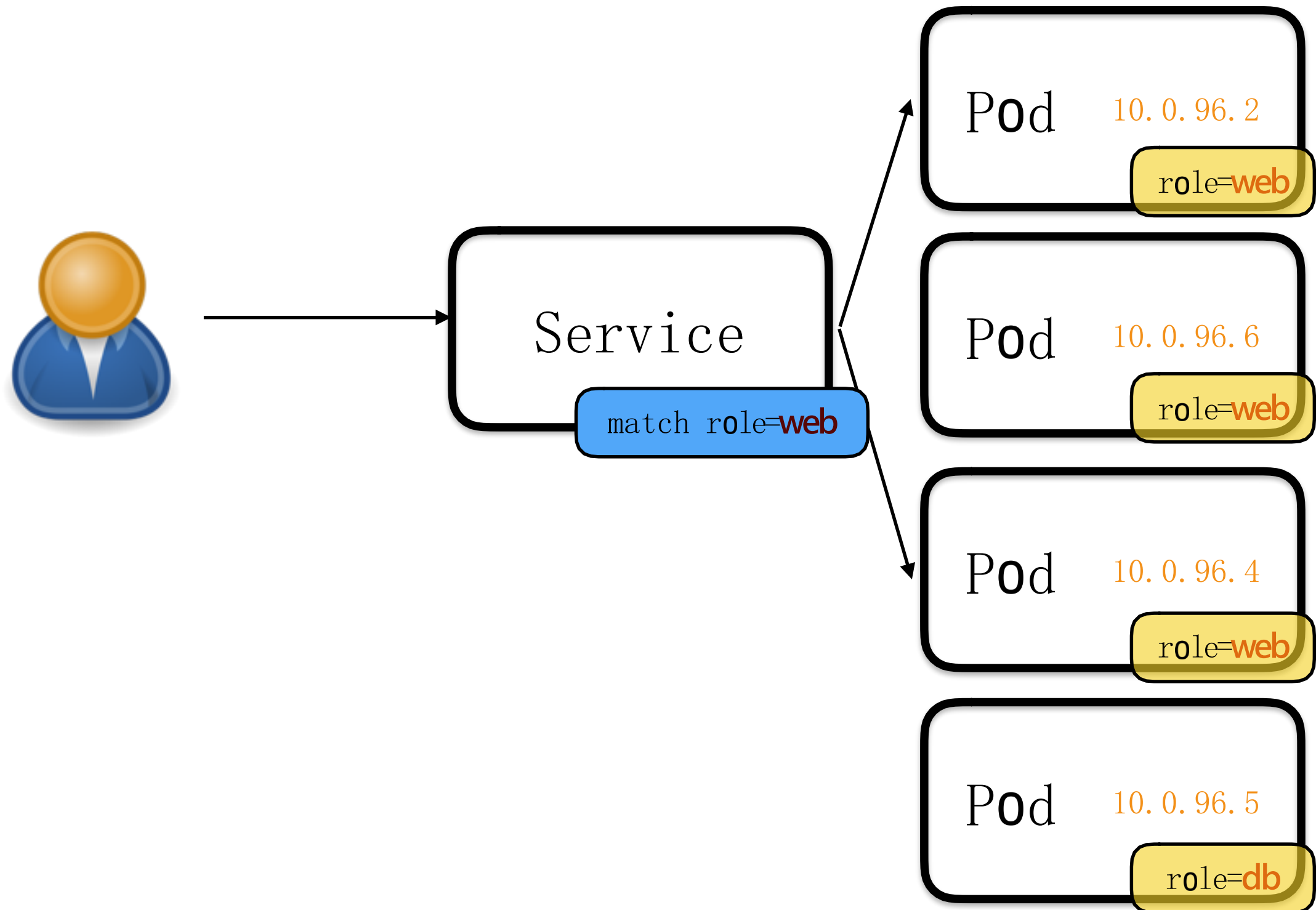
Access Via Service



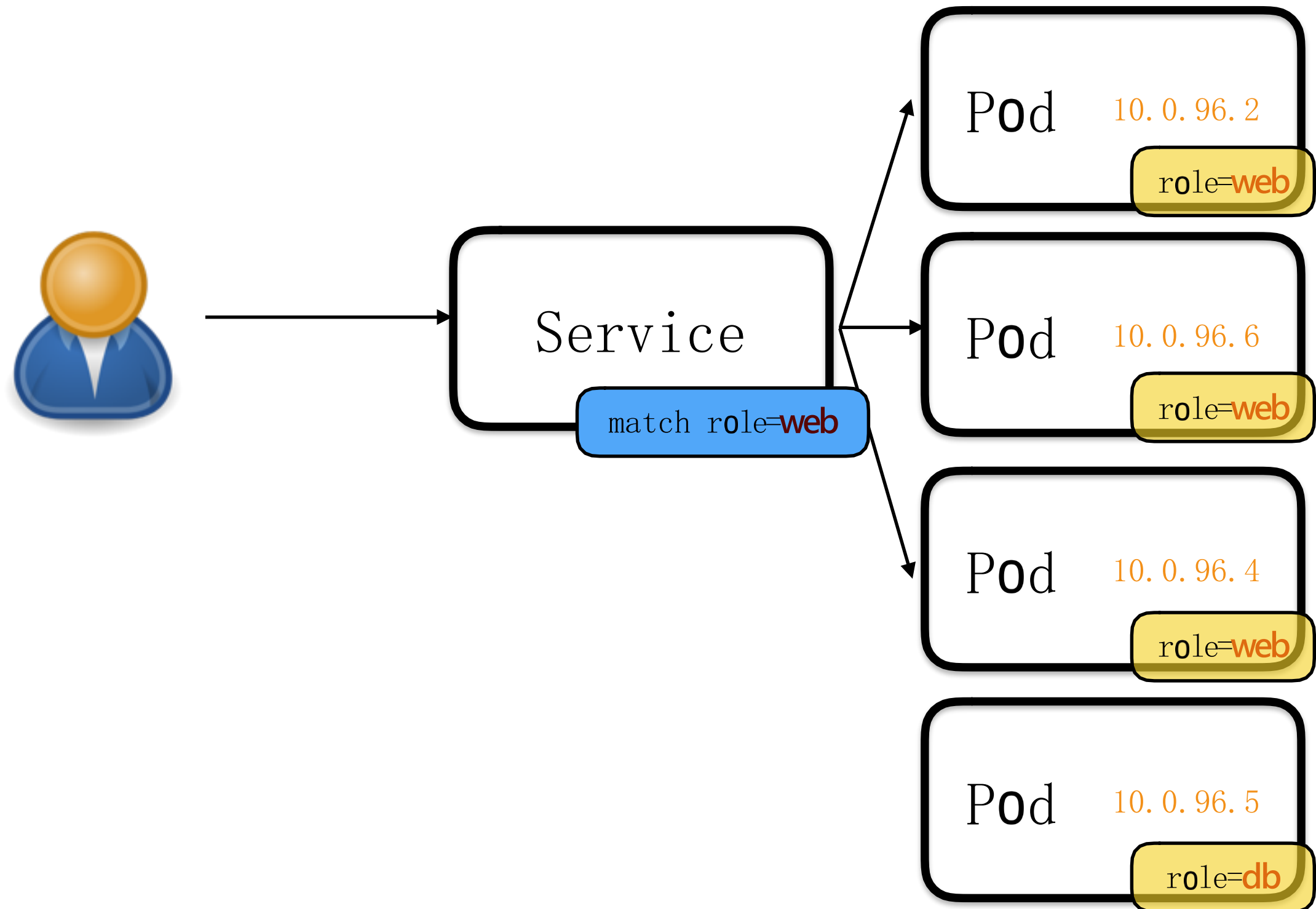
Access Via Service



Access Via Service



Access Via Service



Secrets

Store pieces of
data in
k8s

e.g. Identity Information

(securely)

(...in the future)

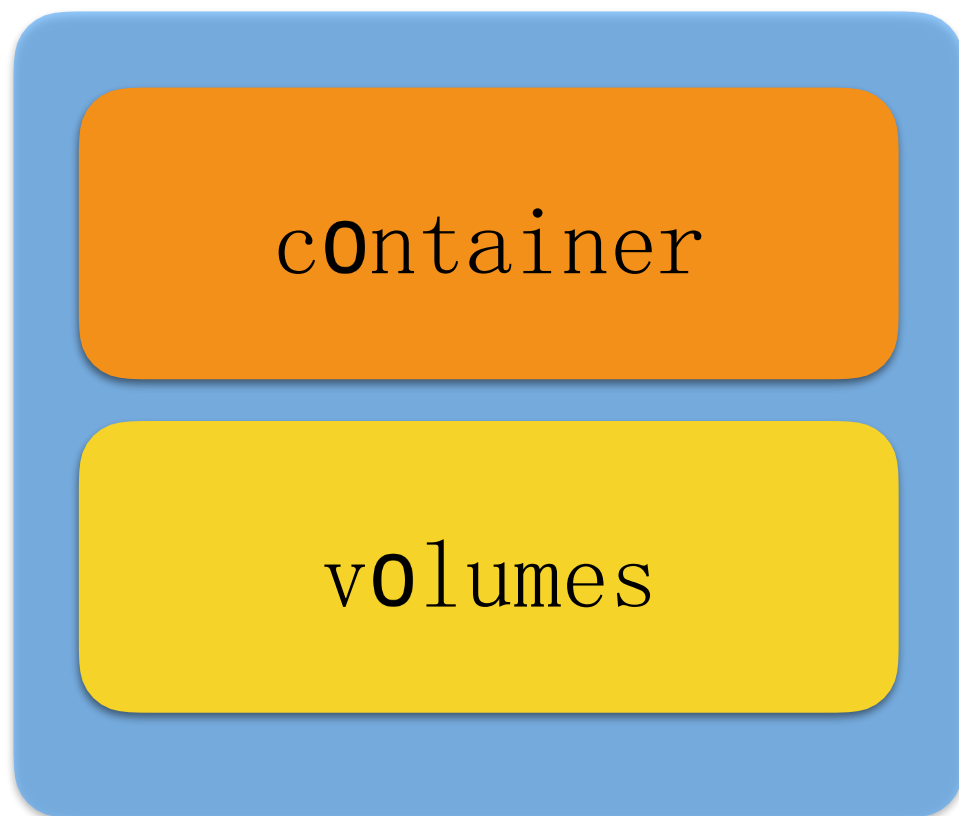
Secret

key1 → base64 value

key2 → base64 value

key3 → base64 value

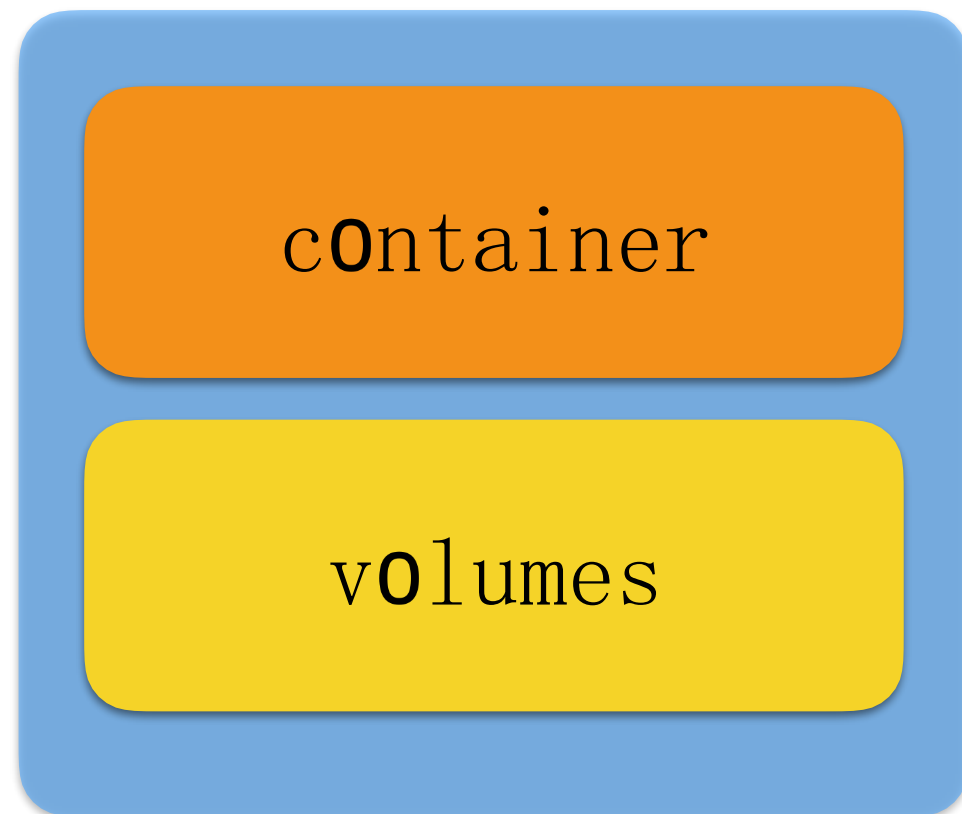
Pod



```
container:  
  volumeMounts:  
    - name: certificates  
      mountPath: /etc/ssl/certs
```

```
volumes:  
  - name: certificates  
    secret:  
      secretName: ca-certificates
```

Pod



```
env:  
  - name: foo-secret  
    valueFrom:  
      secretKeyRef:  
        name: foo  
        value: secret-value
```

ConfigMaps

**Same as Secrets
(Unprotected)**

Ingress

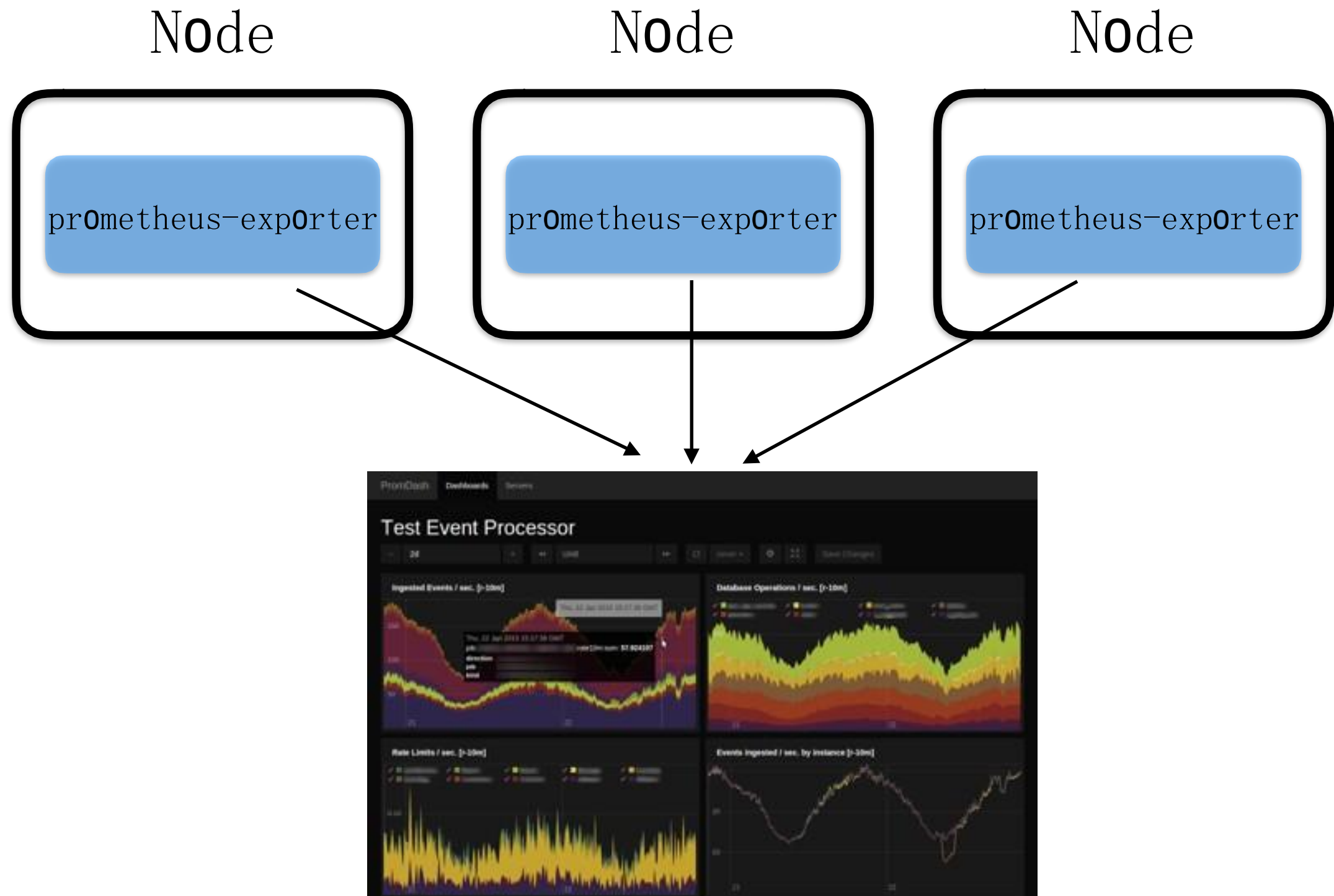
**Services are for within
the cluster
only**

(external IPs allowed, but don't use that to serve
external requests)

**Inbound connections to
internal cluster
services
(New since 1.2)**

DaemonSets

Ensure nodes run a copy of a Pod



Sh*tSets

~~Sh*tSets~~

StatefulSets

Use only when **really** needed

...and with **really** I mean never

Keeps unique IDs in replicas
($x-0$, $x-1$, $x-2$,...)

Questions?