



Kubernetes

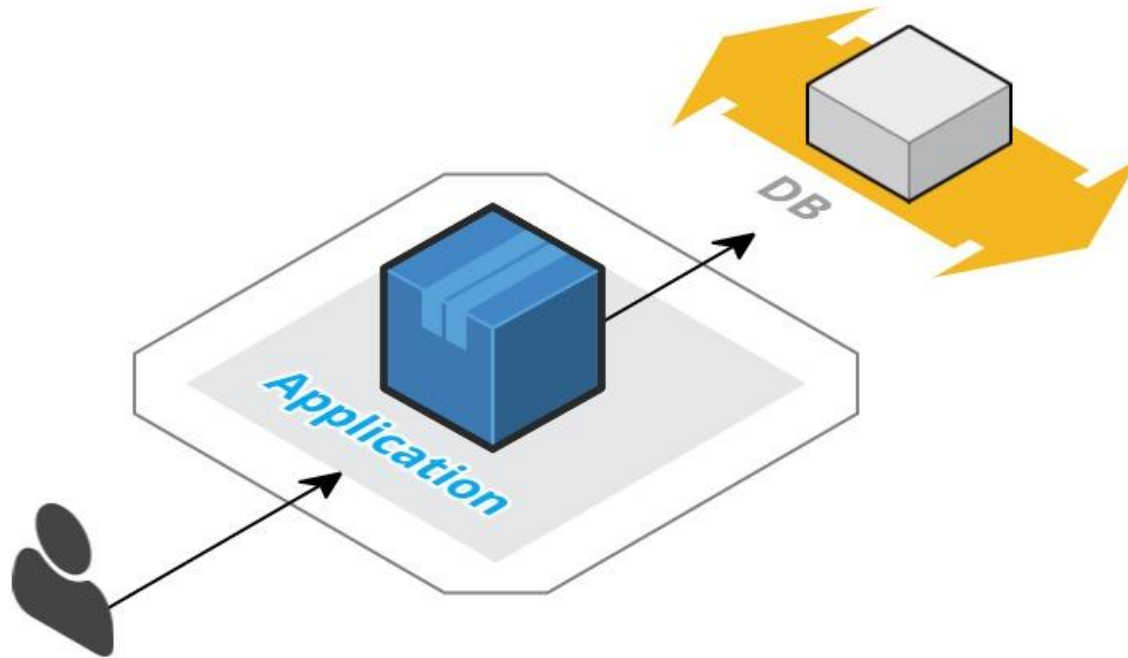
Container orchestration

Joan Marcual

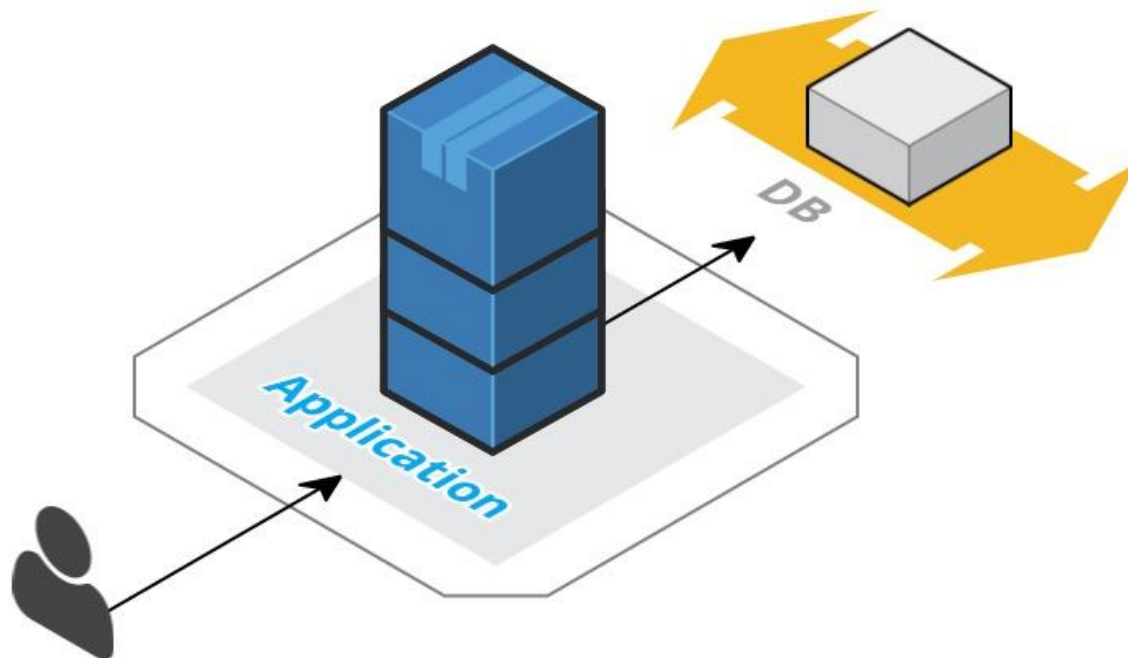
Sergi Alonso

<https://github.com/kiey/CCBDA/tree/master/Research-topic>

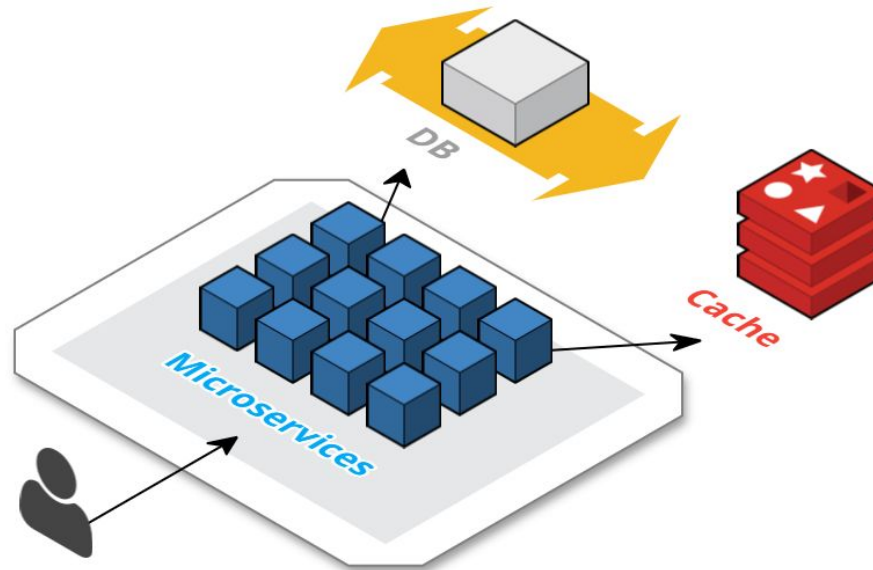
Motivation



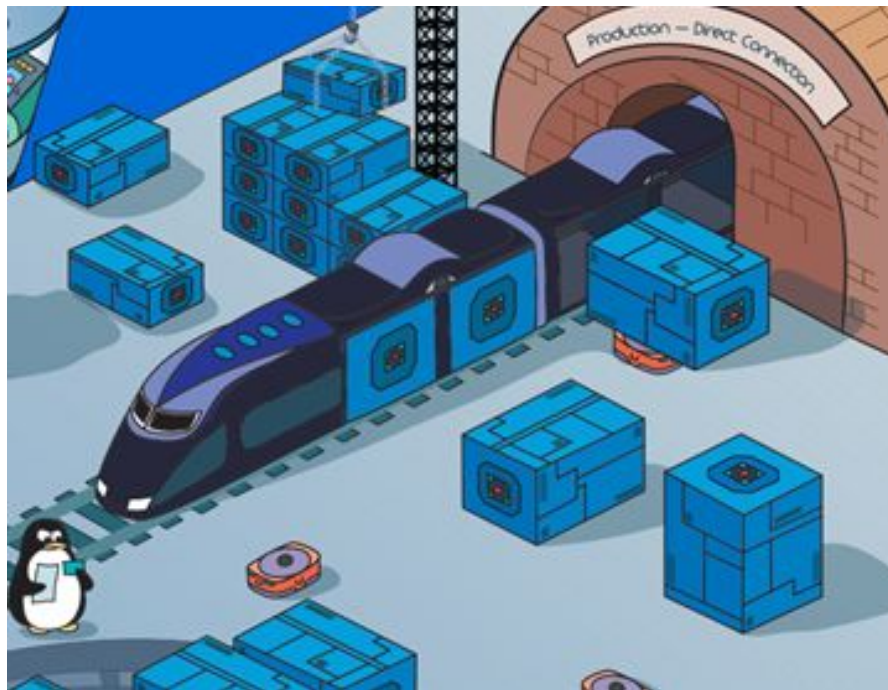
Motivation



Motivation

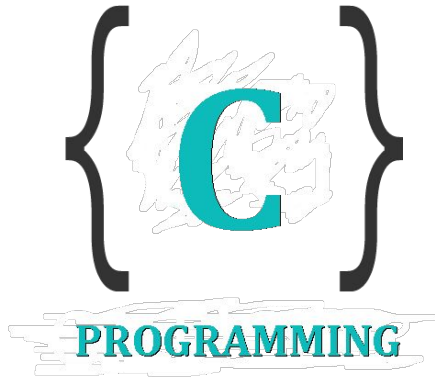


Solution: Orchestrator





Low-level vs High-level

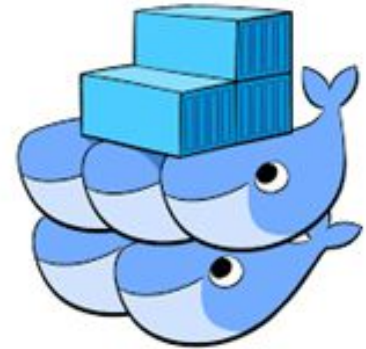


VS





Which one?



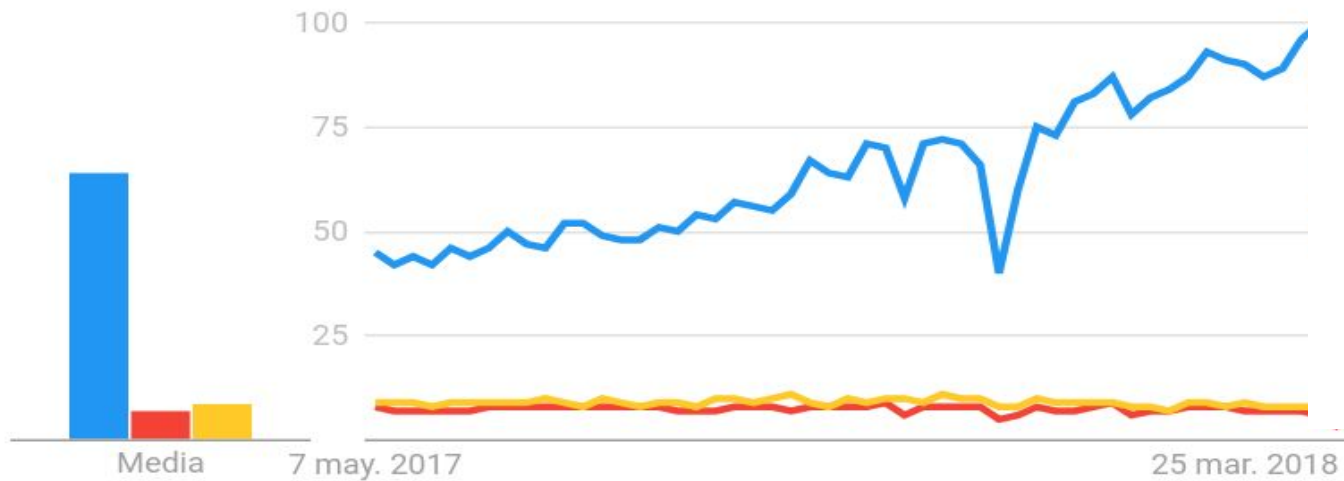


Why kubernetes ?

- Created by Google
- Open Source
- 15+ years
- Huge Community
- Trending

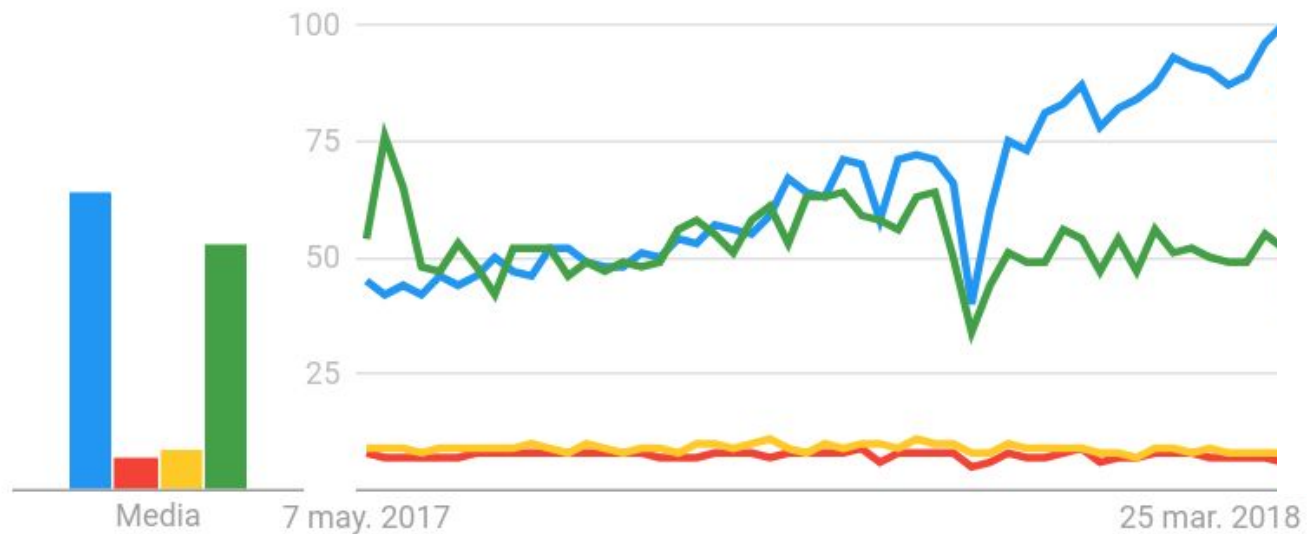
Is it successful?

● kubernetes ● Docker swarm ● Mesos

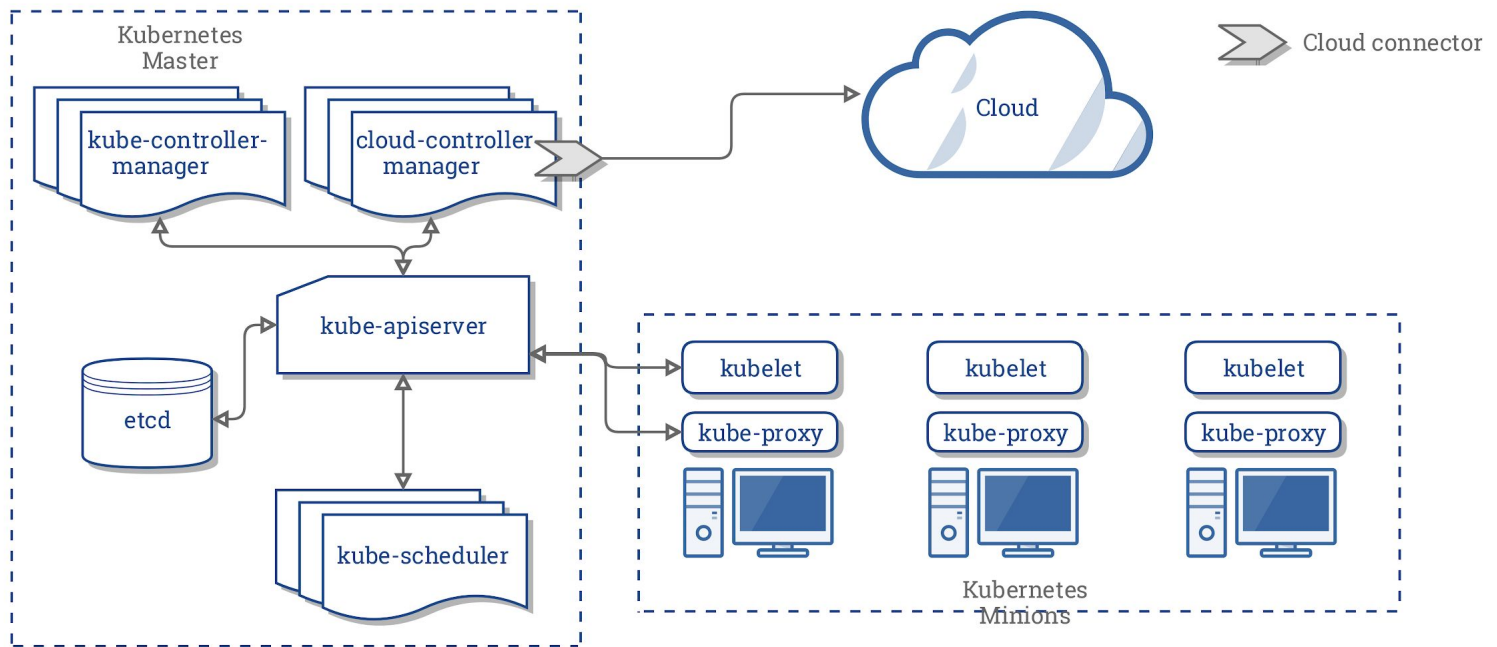


Is it successful?

● kubernetes ● Docker swarm ● Mesos ● cloud computing

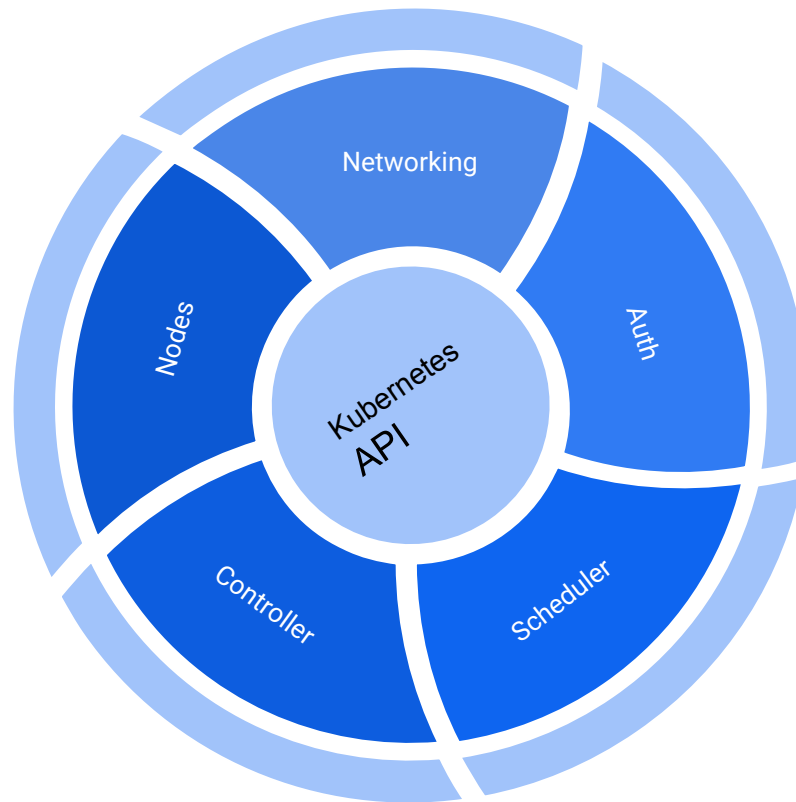


Architecture





Everything is done through the API

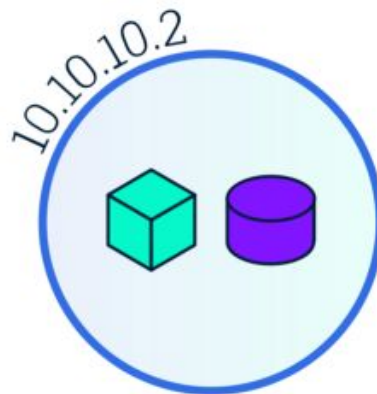




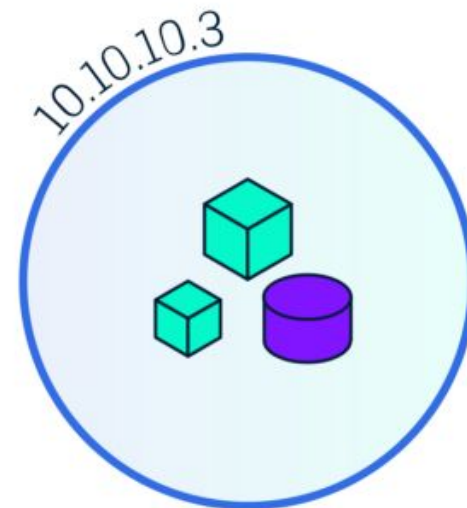
Pods



Pod 1



Pod 2

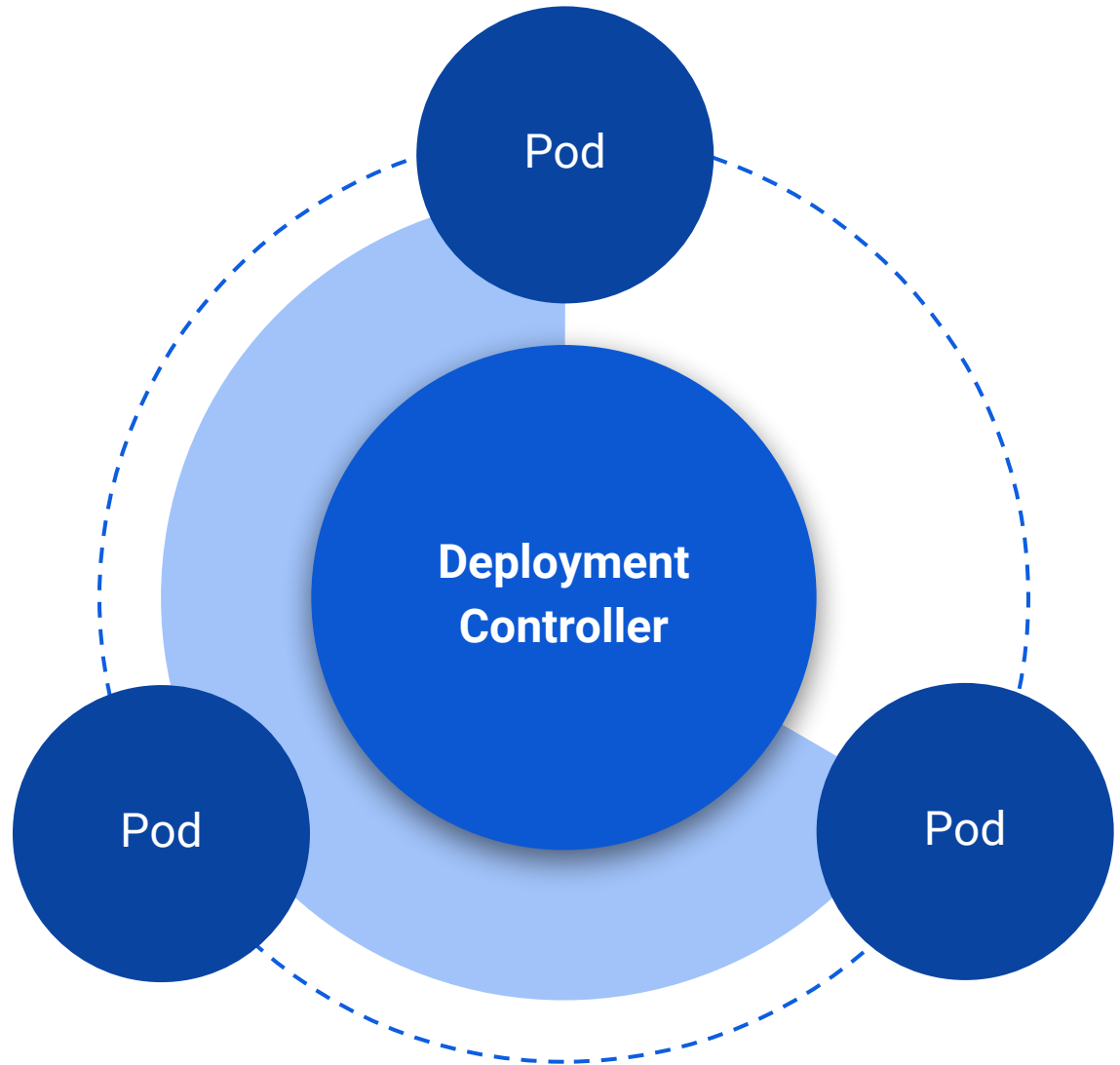


Pod 3

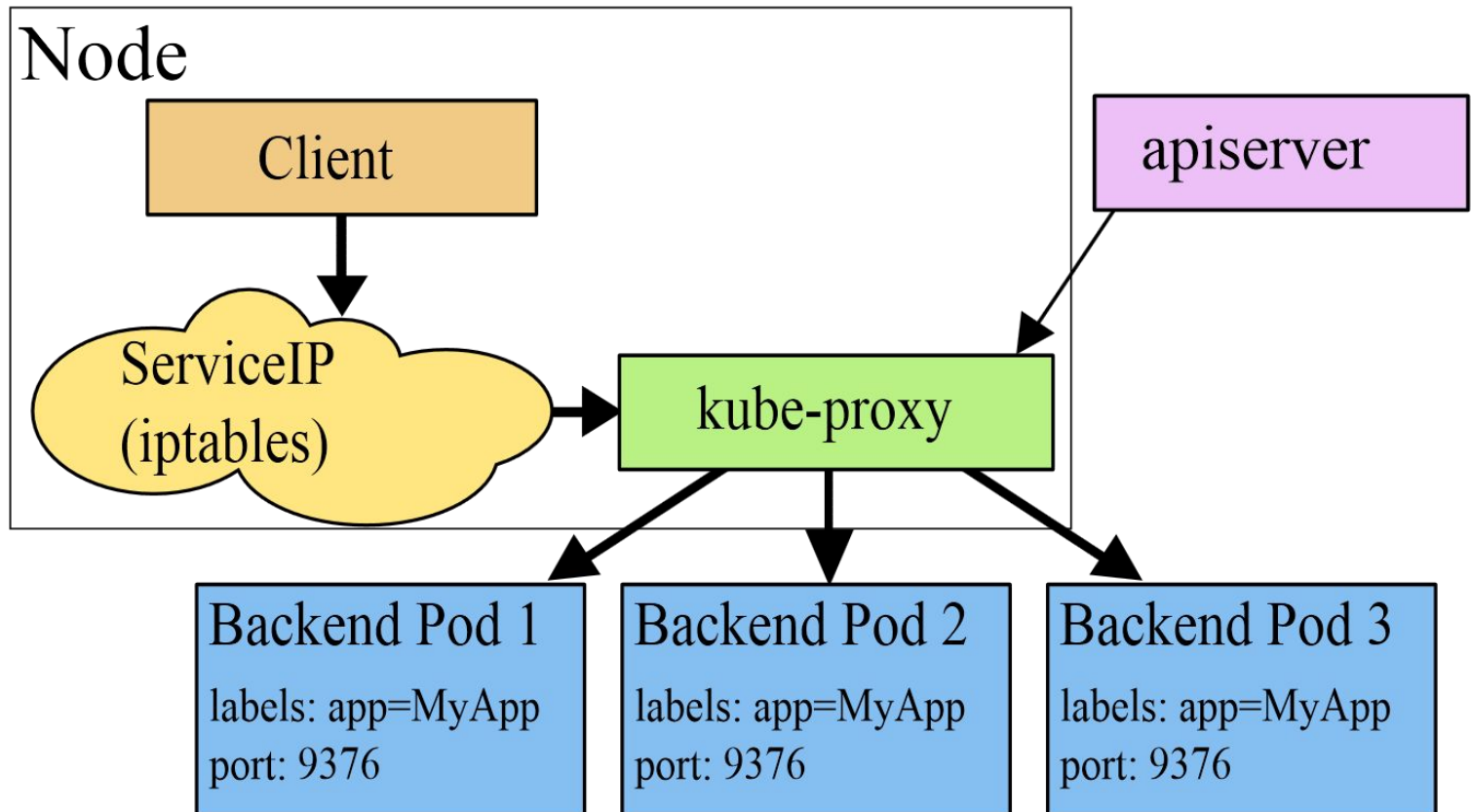


Controllers

- Supervision
- Replication
- Self-healing
- Update rolling



Services





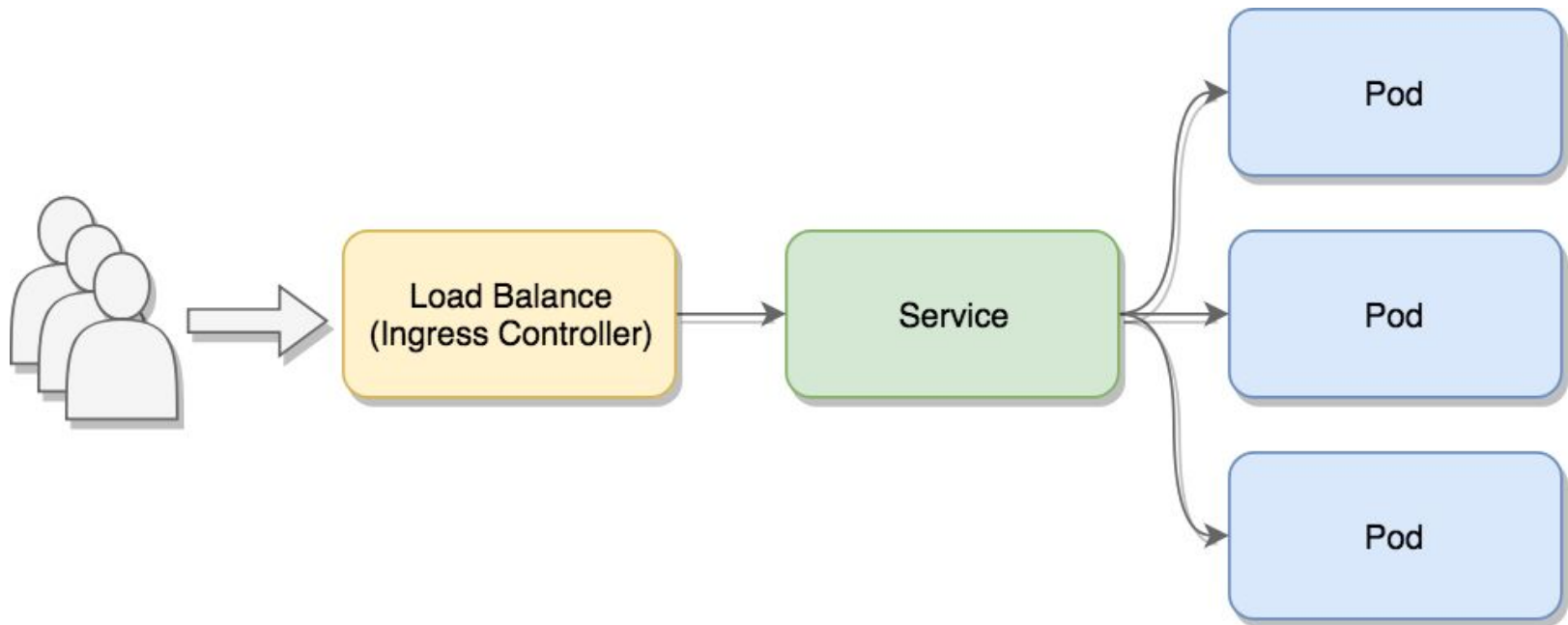
Keep it simple: YAML

```
kind: Pod
metadata:
  labels:
    app: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.7.9
    ports:
    - containerPort: 9376
```

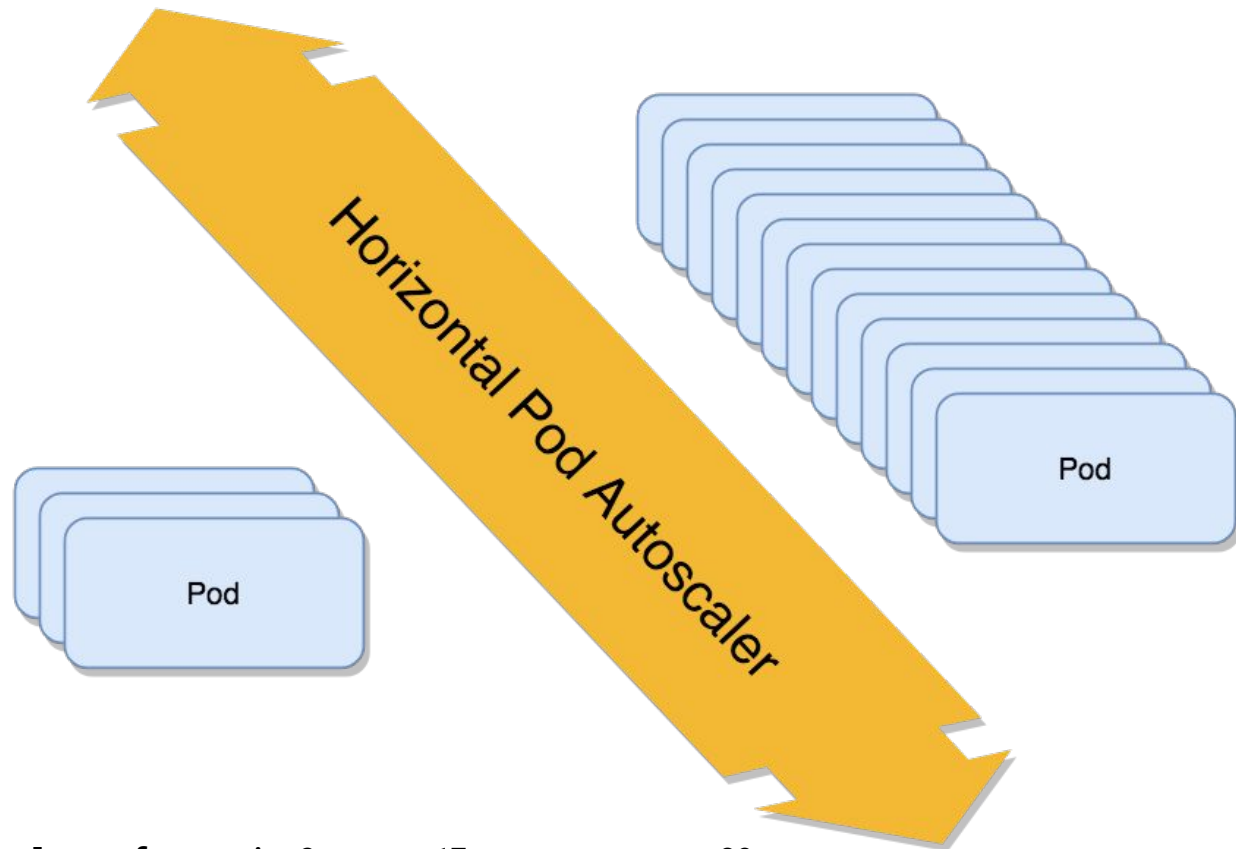
```
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    [...]
```

```
kind: Service
apiVersion: v1
metadata:
  name: web-service
spec:
  selector:
    app: MyWebApp
  ports:
  - protocol: TCP
    port: 80
    targetPort: 9376
```


Load Balancing



Auto-scaling



```
$ kubectl autoscale rc foo --min=2 --max=17 --cpu-percent=80
```



Author's experience



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación



Conclusions

- It's good
- It's mature
- It's easy
- Everybody uses it
- De facto standard
- Huge community
- Vendor support
- Seriously, it rocks