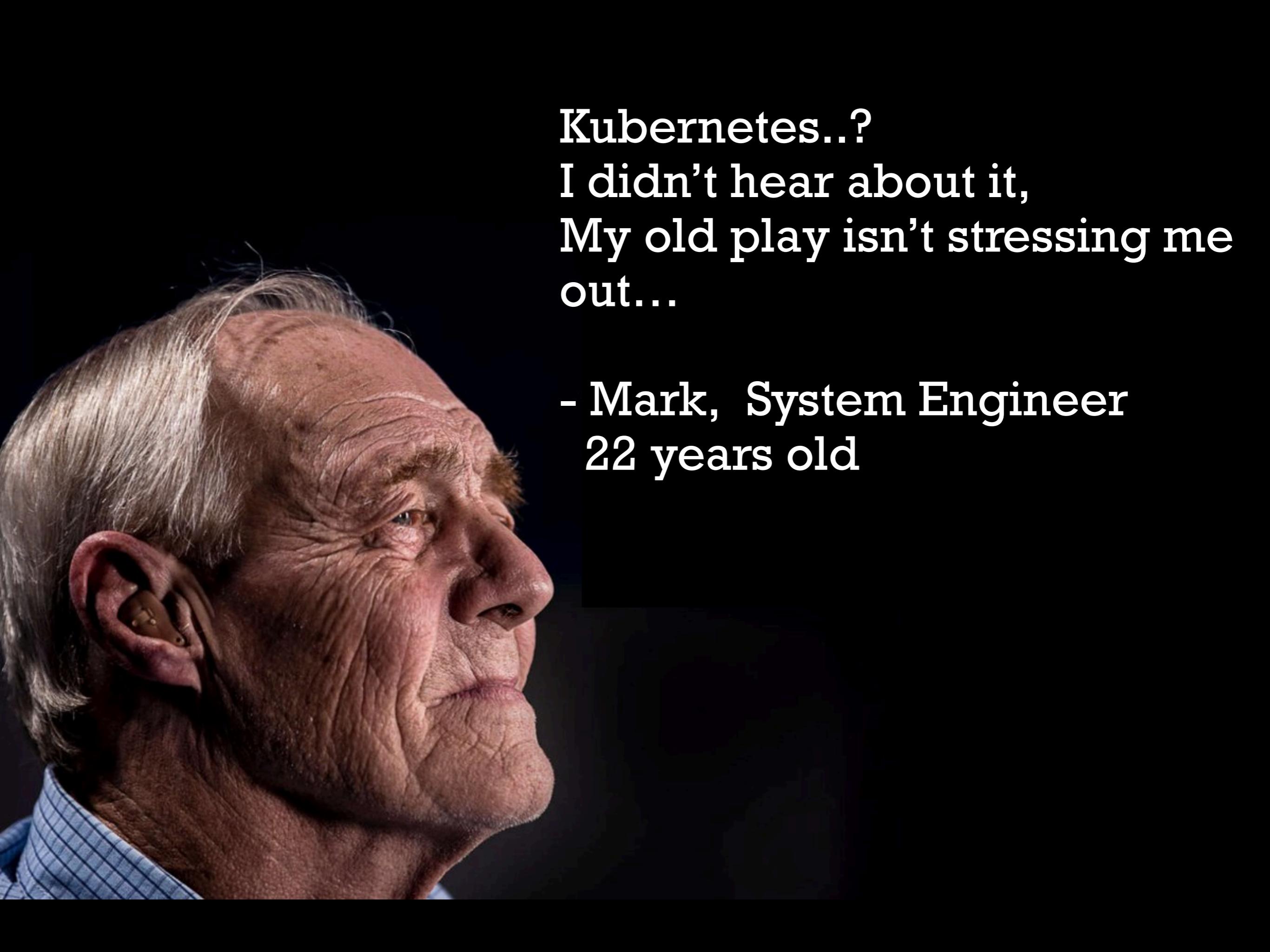


GAME OF KUBERNETES

Uri Shamay
cmpxchg16

```
$ whoami
> Uri Shamay @cmpxchg16
> Proud Father +3
> Extreme HW geek
> Co-Founder of Root Cause Podcast
https://rtcz.io
> Co-Founder of Java-IL Community
> Co-Founder of Tech-Leads-IL Community
> Infrastructure Architect @Next-
Insurance
> More here: http://cmpxchg16.me
```

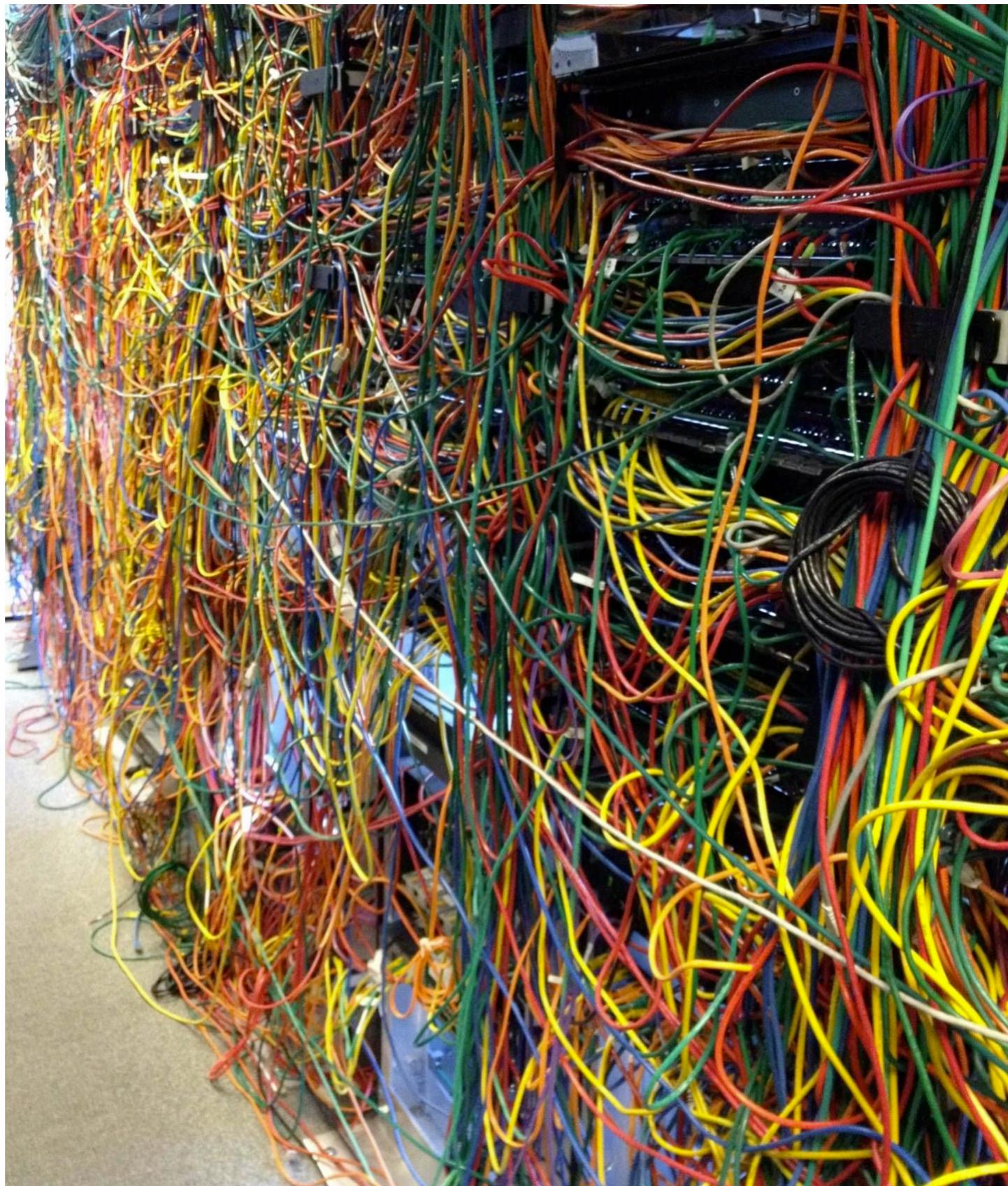


Kubernetes..?
I didn't hear about it,
My old play isn't stressing me
out...

**- Mark, System Engineer
22 years old**

In blazing competitive
world, dev velocity is
must!

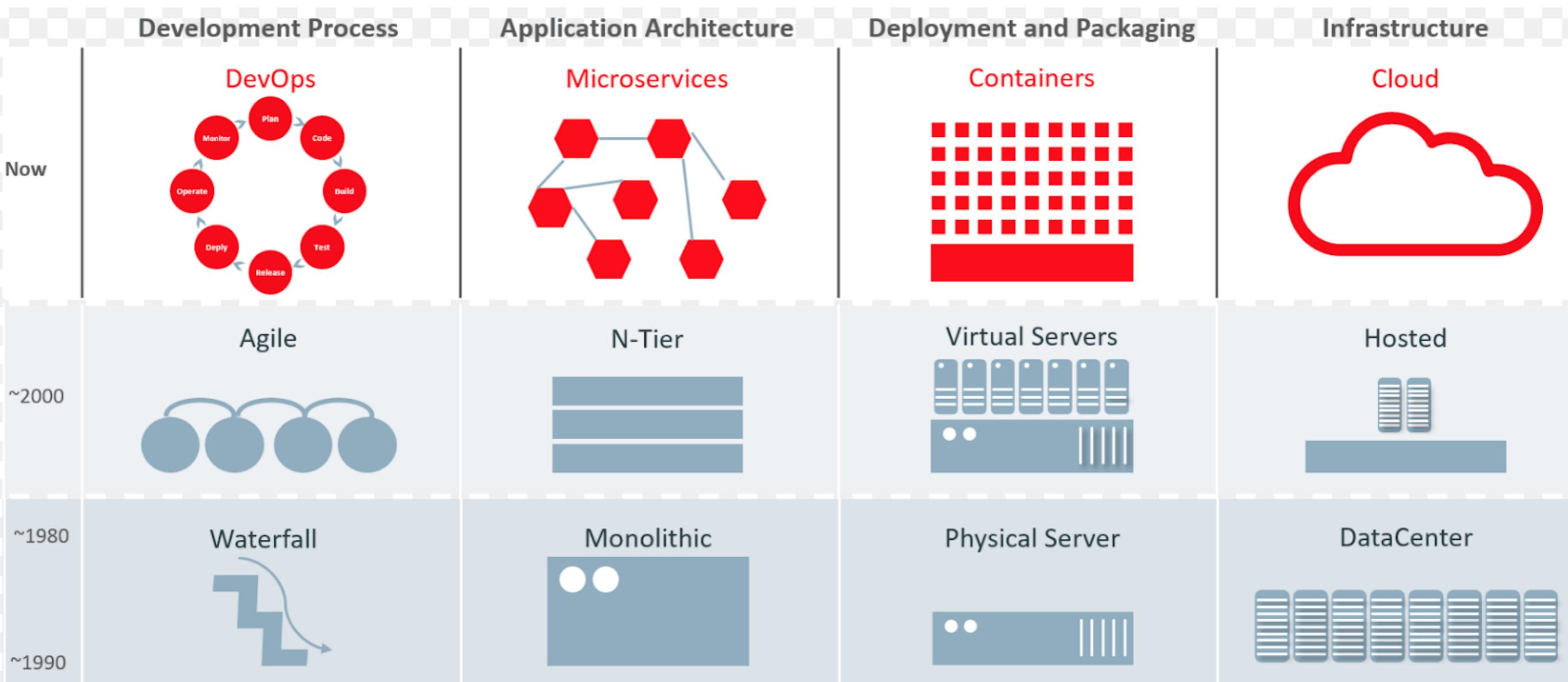
Motivation :: let's look at history



Operations Recipe

- Server admin racked a 2U server ~ hours
- OS + patches ~ hours
- Application admin
 - Load app service
 - Config proxy
 - SSL/TLS Certificate

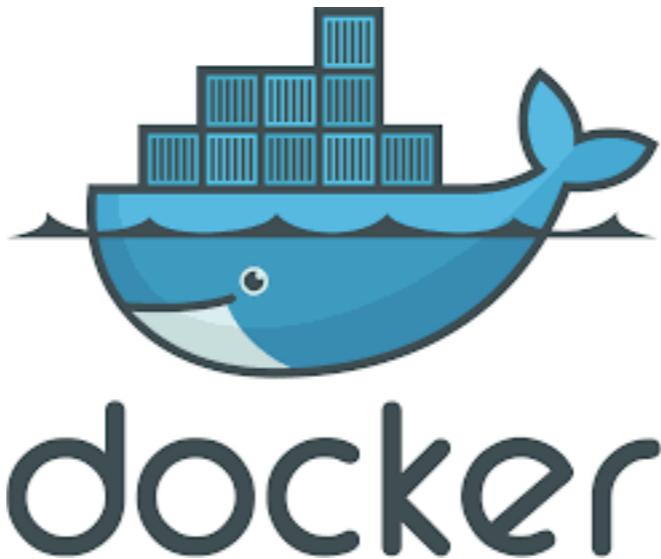
~ hours
- Network Engineer
 - Provision firewall ports + subnet
- Sys admin put storage ~ hours



**Everyone want an
automatic &
repeatable &
resilient system,
how do I get such?**

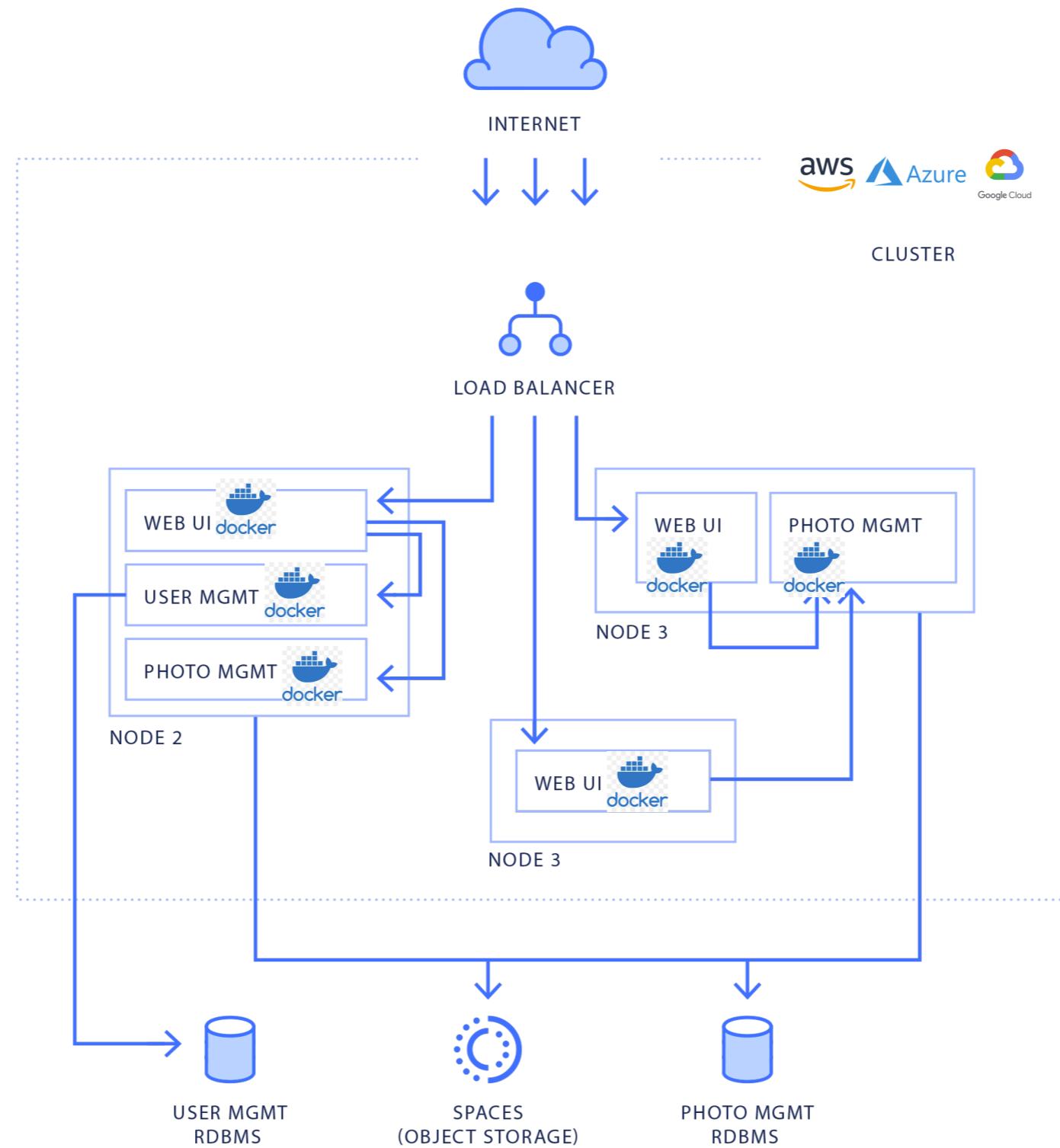
What every system needs :: cross-cutting concerns

- Service discovery and load balancing
- High availability
- Auto scaling in/out
- Automated rollouts and rollbacks
- Self-healing
- ETC (Automatic bin packing & Secret and configuration management)

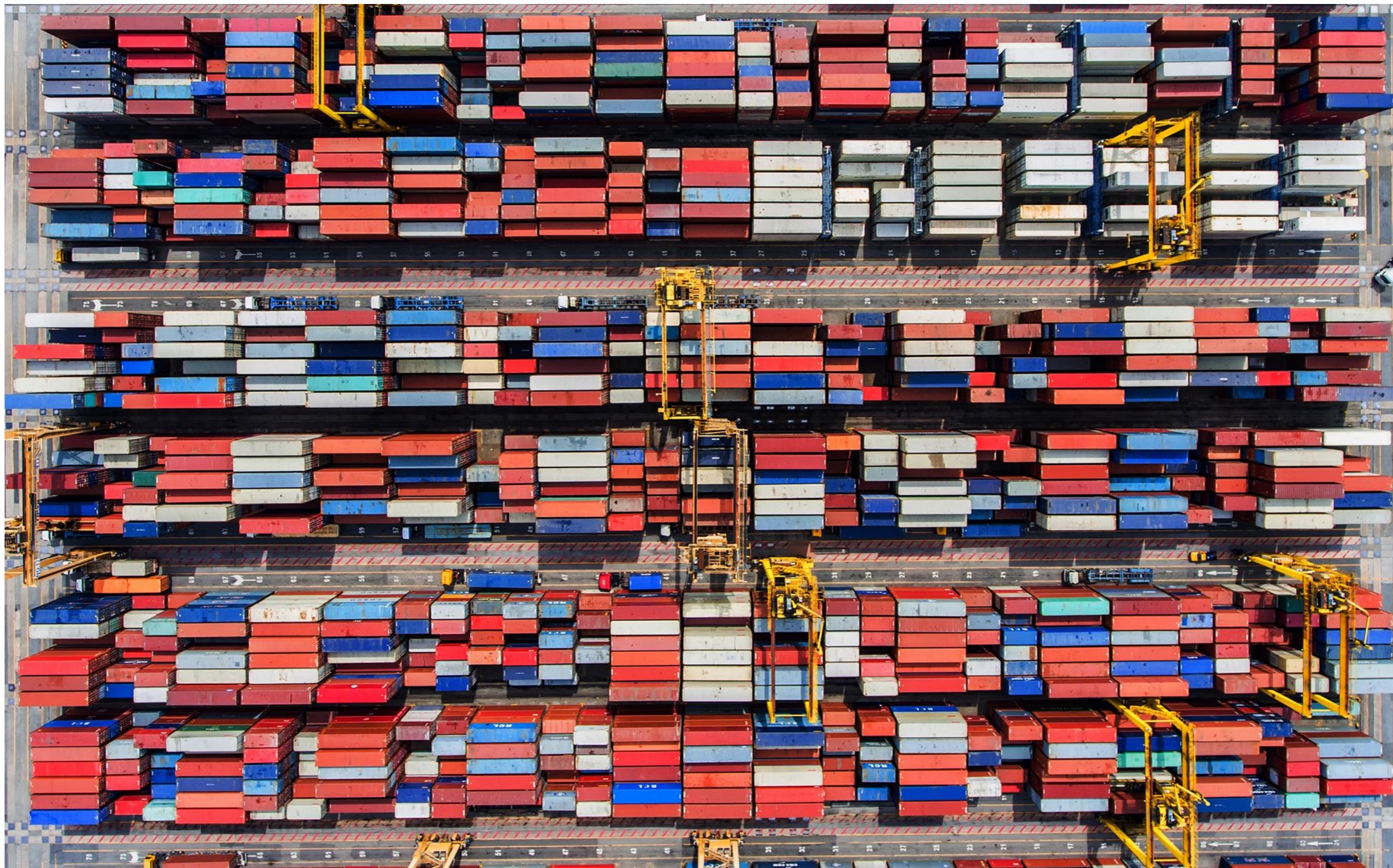


If container is the smallest execution unit, how do we orchestrate multiple containers?

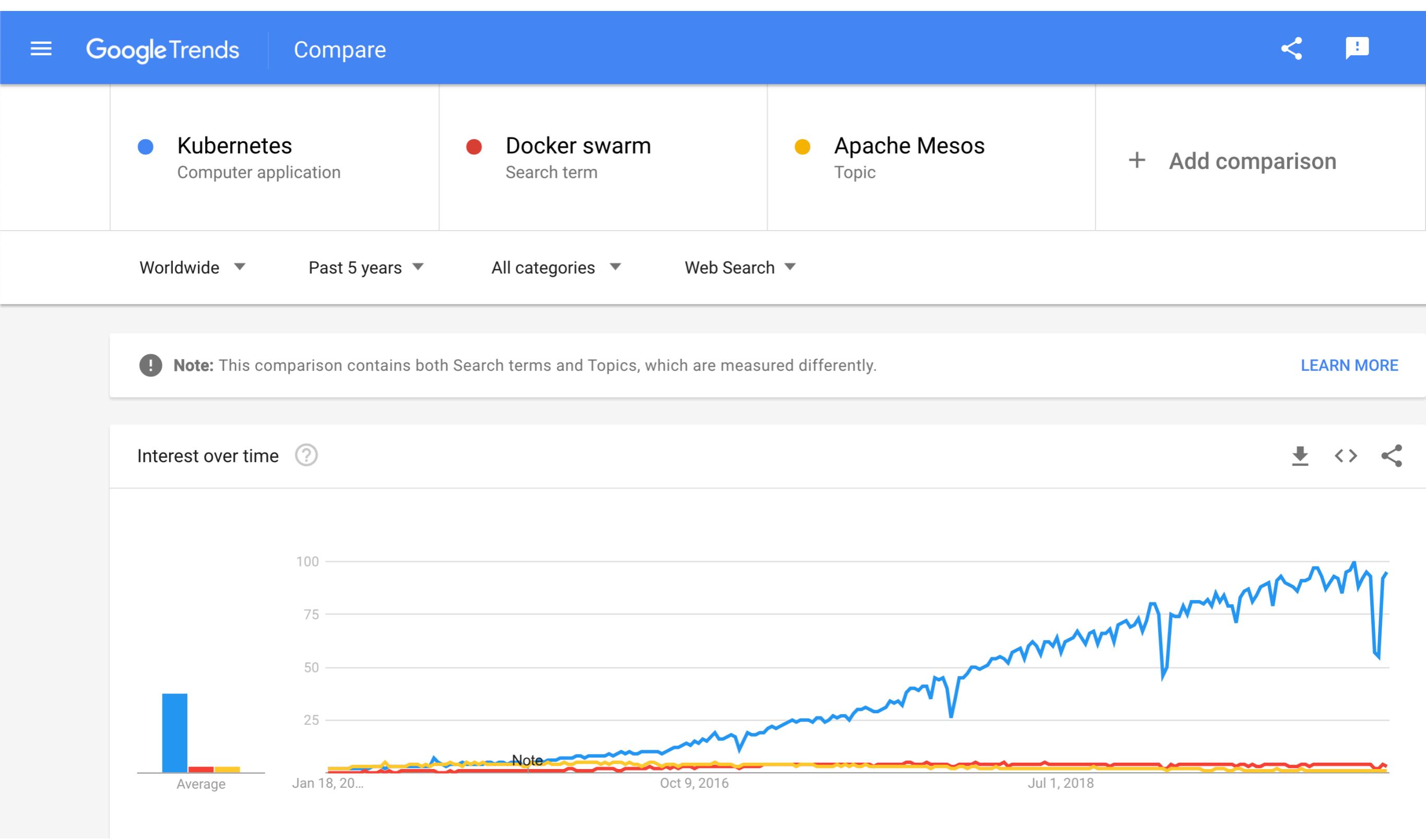
Desired State



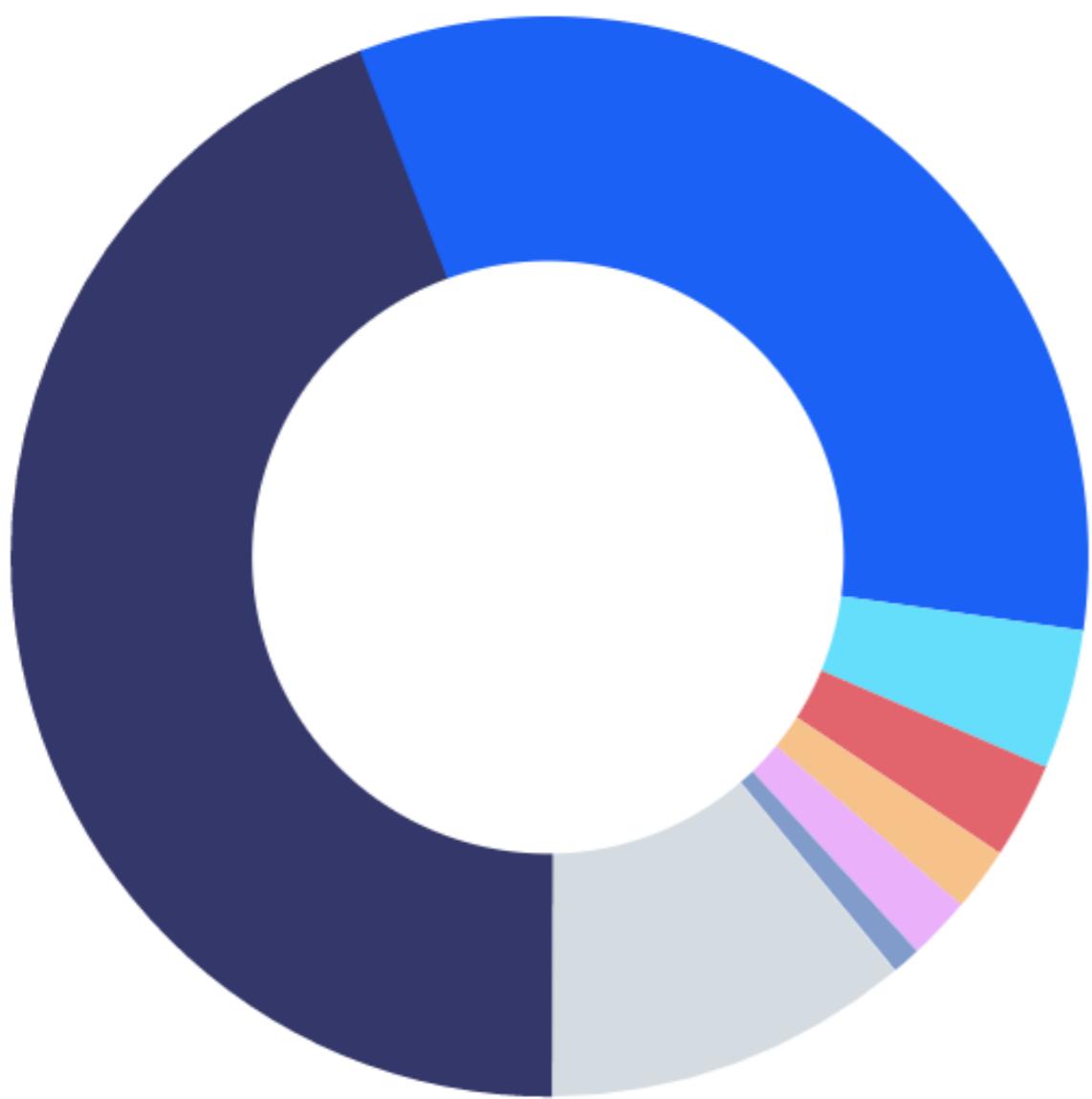
Cluster management software



Cluster management software become popular



Which container orchestration platform do you primarily use?



- Kubernetes 42%
- Docker Swarm 35%
- OpenShift by Red Hat 5%
- Apache Mesos 3%
- Marathon 2%
- Nomad by HashiCorp 2%
- CoreOS Tectonic 1%
- Other 12%

What is Kubernetes?

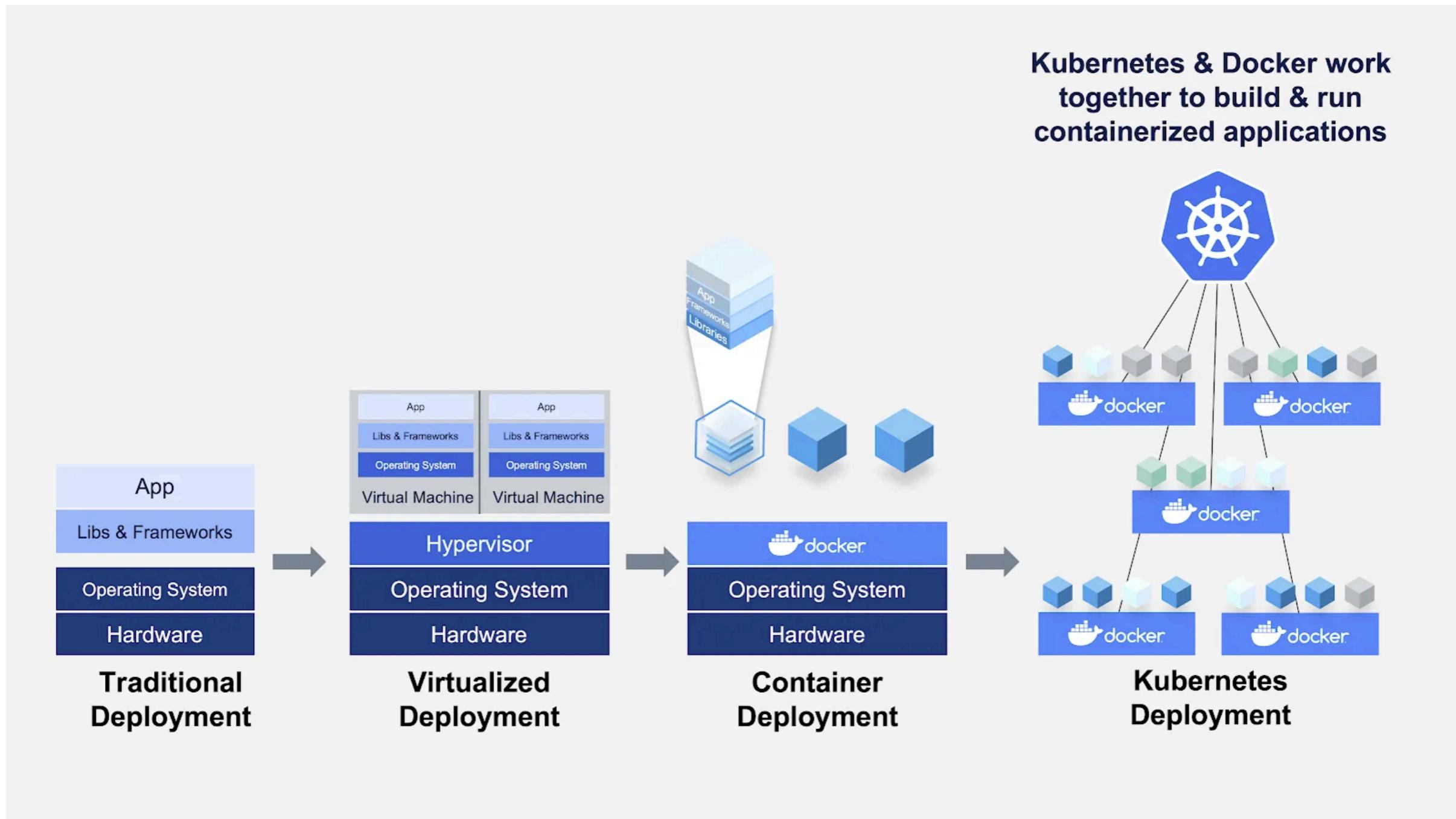
Kubernetes (commonly stylized as k8s) is an open-source container-orchestration system for automating application deployment, scaling, and management.

Kubernetes History

- First announced by Google in mid-2014
- Development and design are heavily influenced by Google's Borg system
- Start as Java project, abandoned Java very soon, move to Go

Why do I need it...?
Because infrastructure
automation & orchestration are
not just for Google ^_(ツ)_/^

Kubernetes look on data center like a big computer



#Benefit: cross-cutting concerns implementation

- Service discovery and load balancing
- High availability
- Auto scaling in/out
- Automated rollouts and rollbacks
- Self-healing
- ETC (Automatic bin packing & Secret and configuration management)

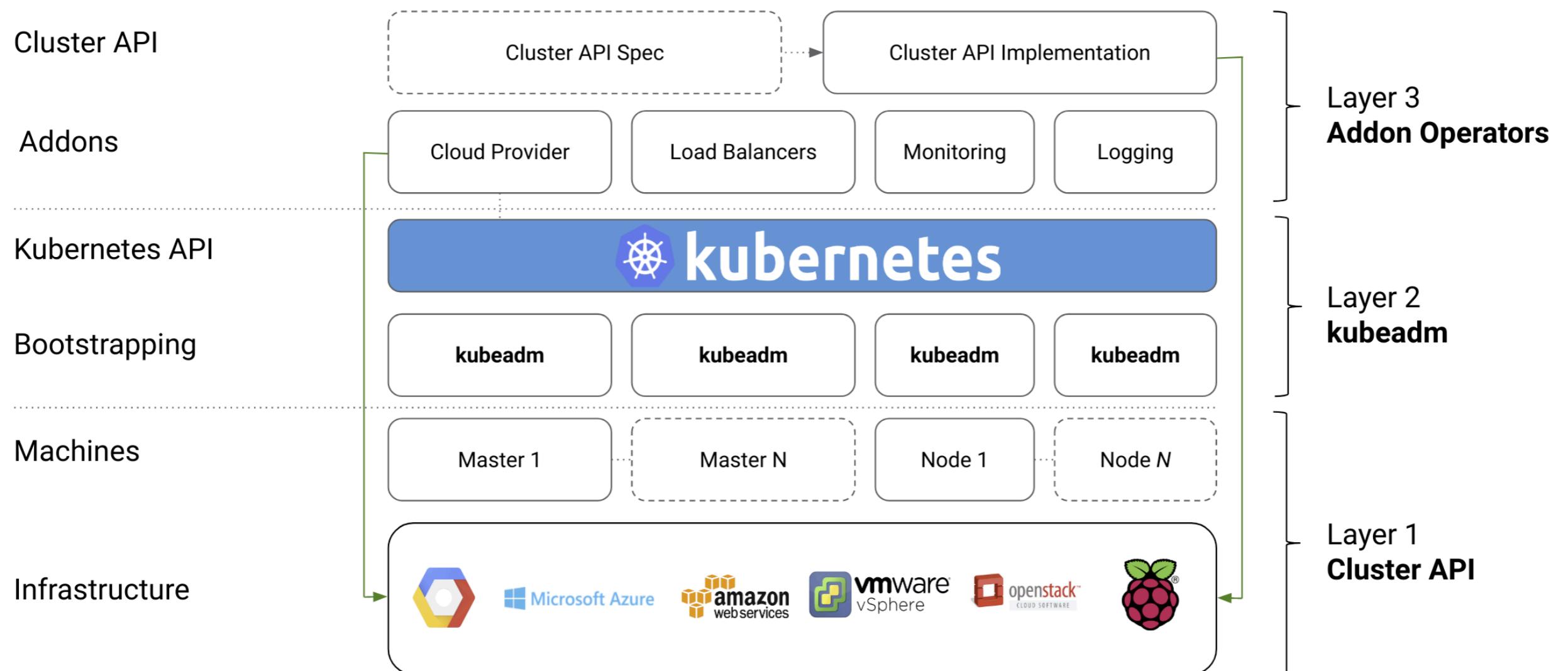


#Benefit: Infrastructure As Code

```
# service.yaml
apiVersion: v1
kind: Service
metadata:
  name: hello-kubernetes
spec:
  type: LoadBalancer
  ports:
  - port: 80
    targetPort: 8080
  selector:
    app: hello-kubernetes
```

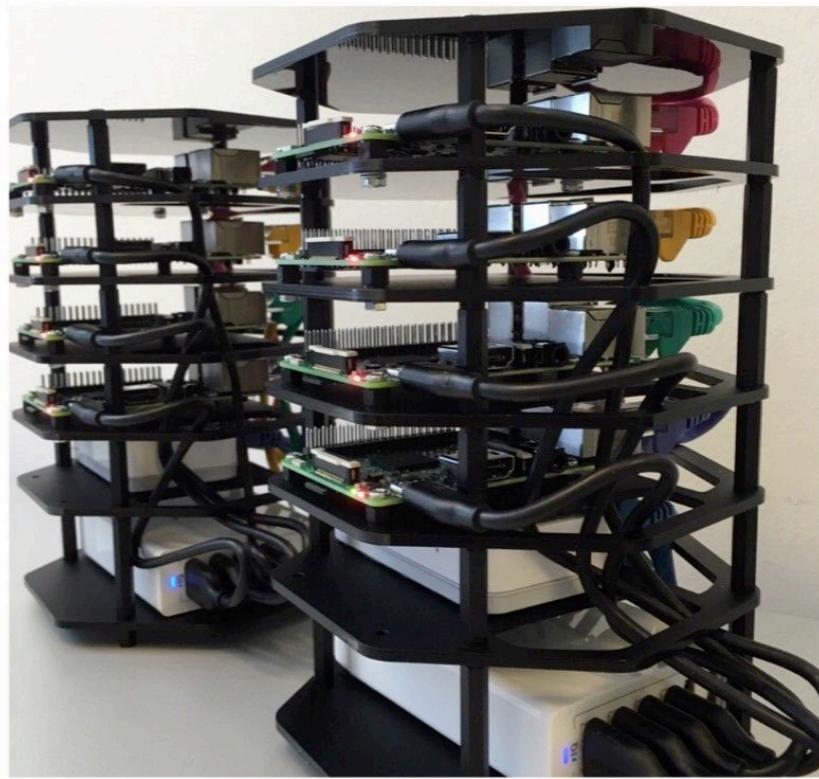
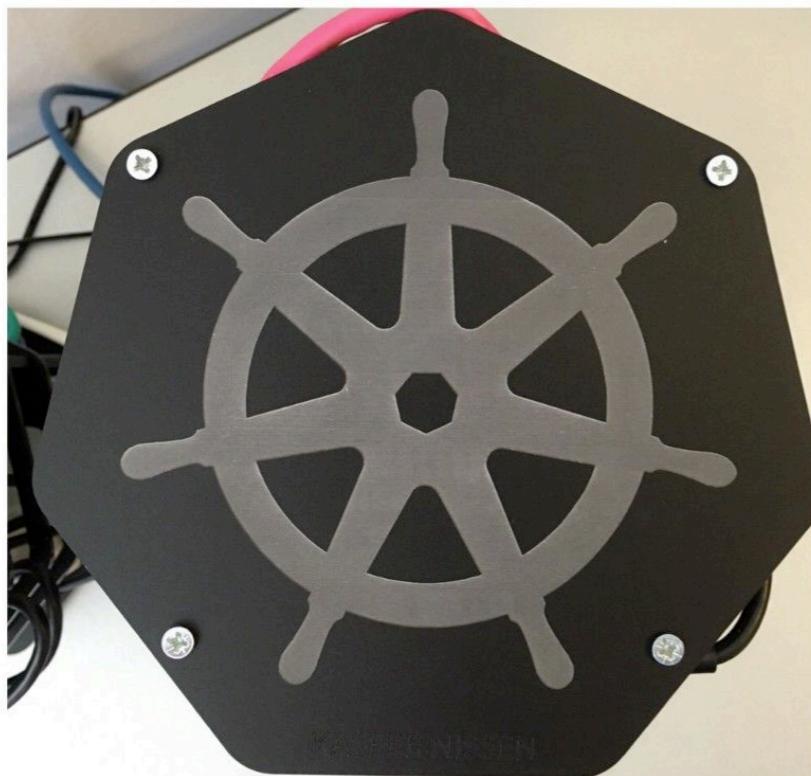
```
# deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: hello-kubernetes
spec:
  replicas: 3
  selector:
    matchLabels:
      app: hello-kubernetes
  template:
    metadata:
      labels:
        app: hello-kubernetes
    spec:
      containers:
      - name: hello-kubernetes
        image: hello-kubernetes
        ports:
        - containerPort: 8080
```

#Benefit: Cloud Native = no vendor lock



#Benefit:
automate the complexity of heterogeneous environments

Kubernetes cluster on raspberry pi



As the Cloud Operating
System, Kubernetes has more,
too much for one session :D

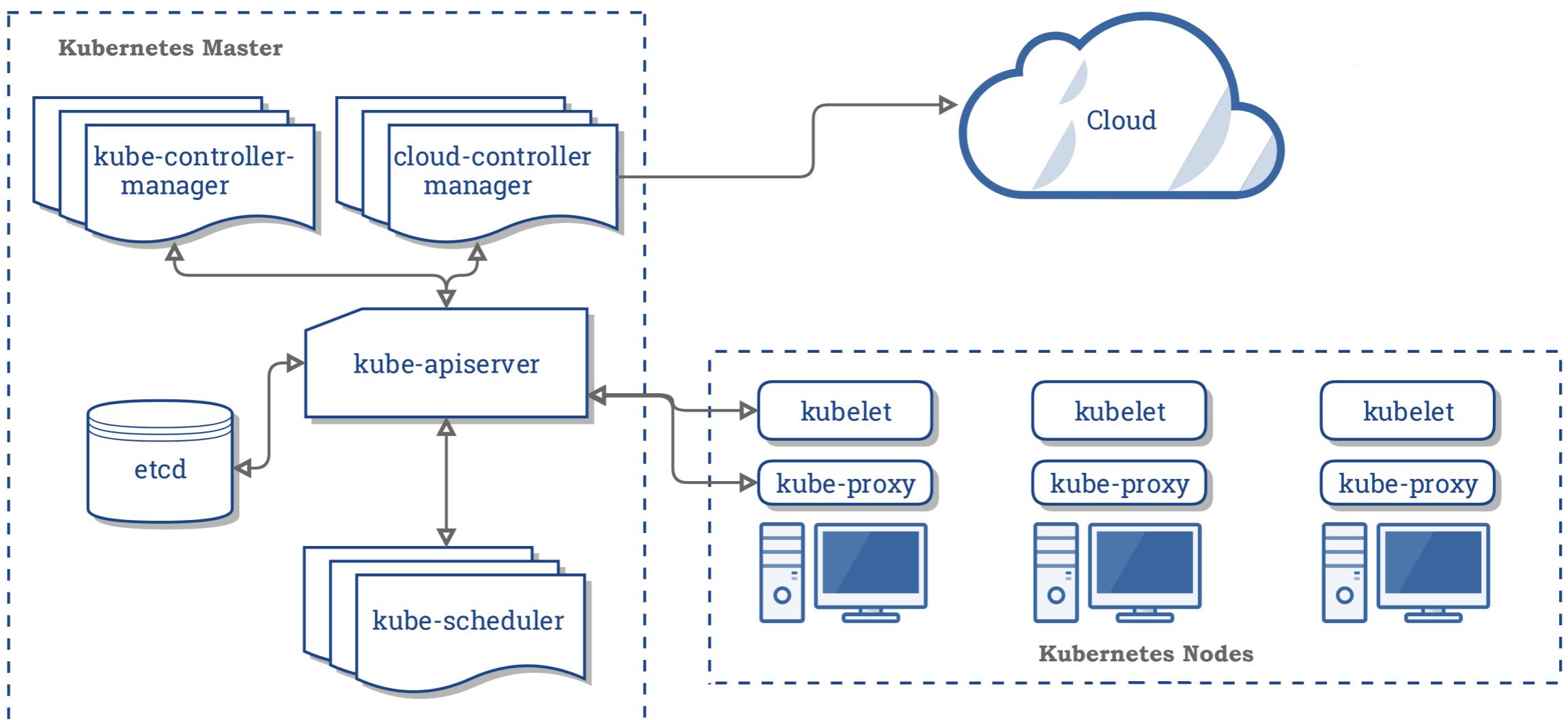
**Want an algorithm to
maximise ROI of
Kubernetes...**

Don't overfit for Kubernetes - start gradually in dev/staging env's



Q&A

Kubernetes Architecture



Managed vs DIY

CONTAINERS



KOPS



EKS



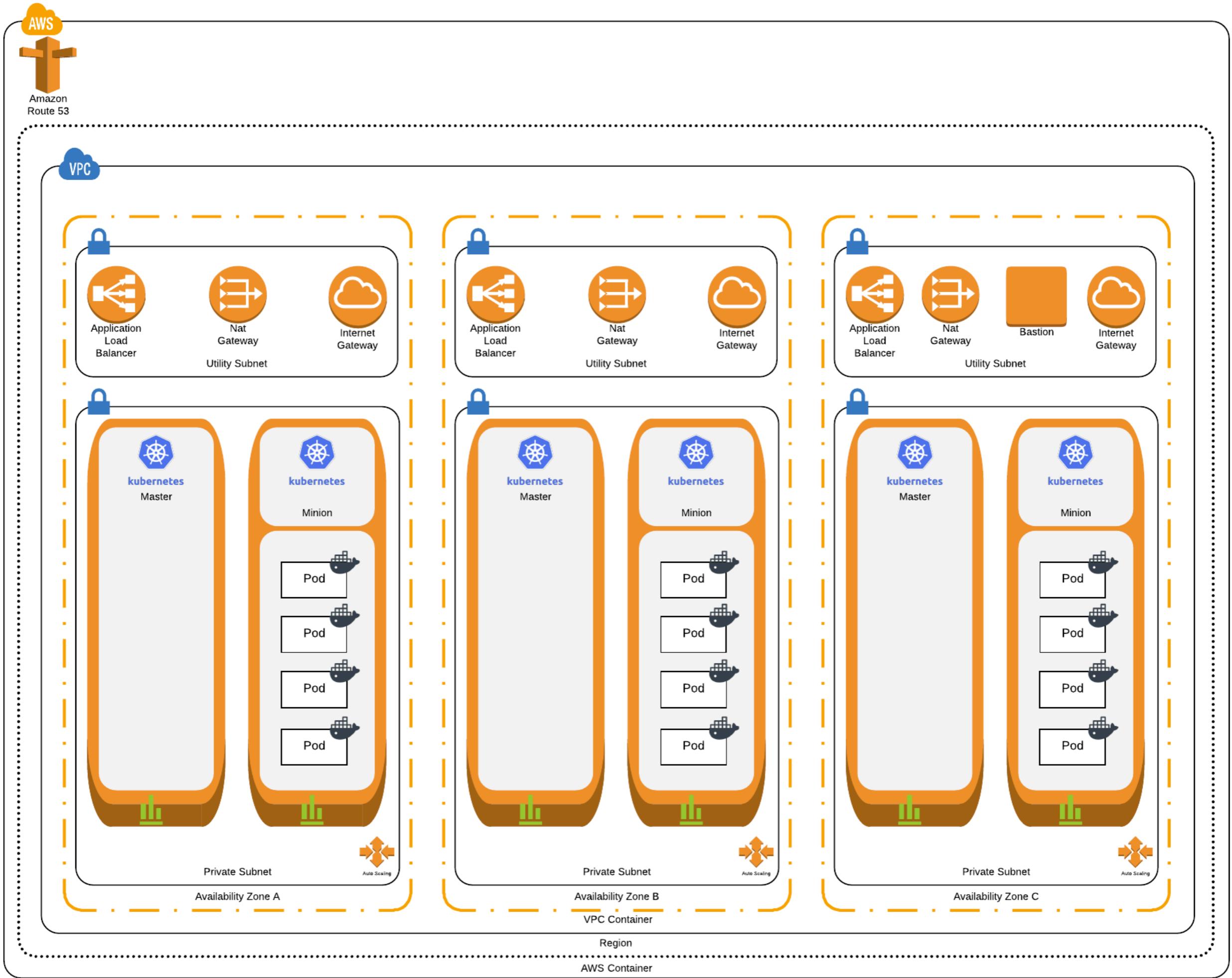
GKE



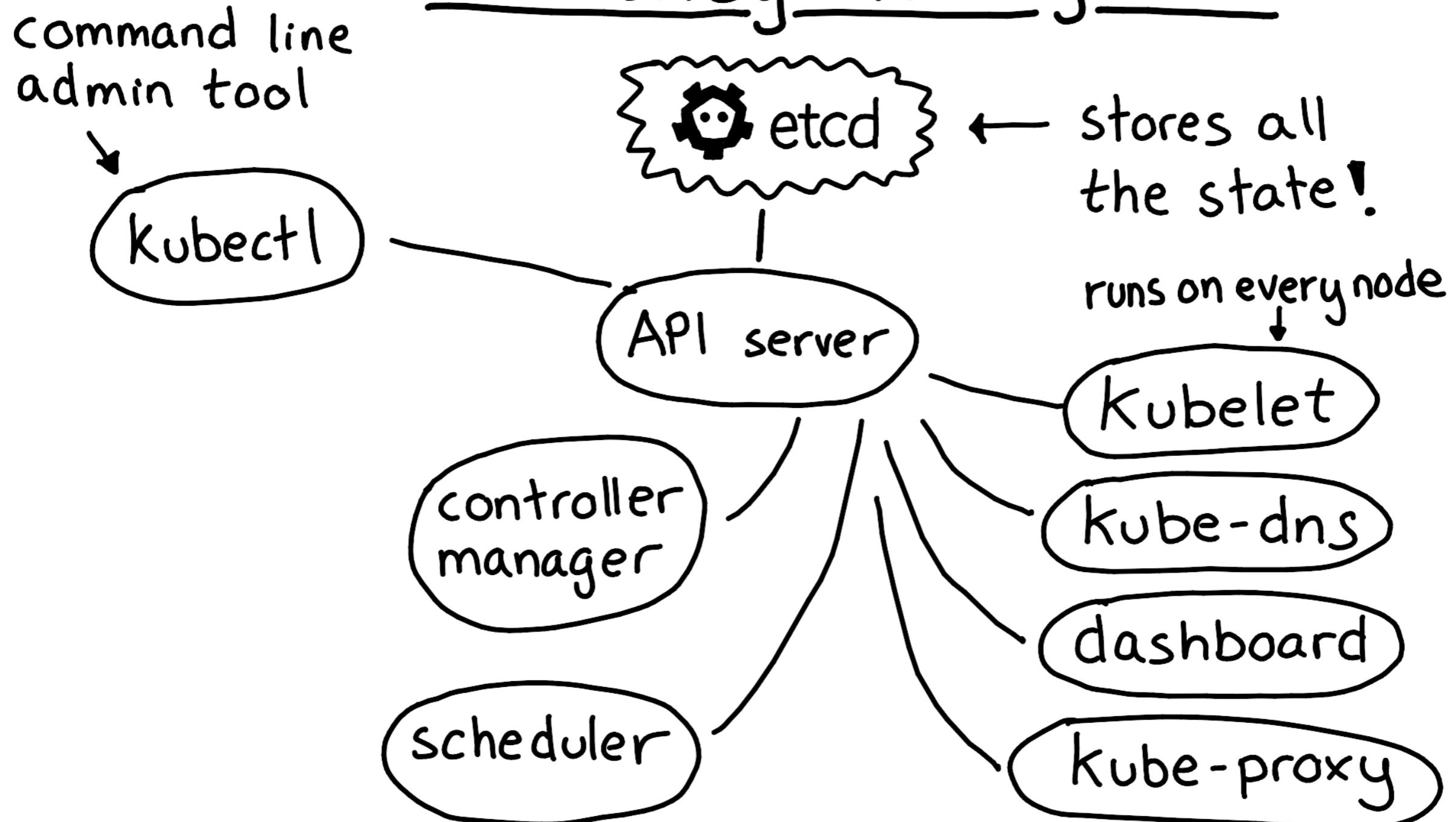
AKS



kubernetes



How they fit together



Kubernetes Component

Automation

Pod

Lowest unit of encapsulation

Services

Deployment of Application Services

Network Policies

Network Configuration

Deployments

Automatic services distribution

Ingress

Load Balancer

ReplicaSet

Replication of containers

ReplicationController

Failover

Volumes

Storage across containers

Namespaces

Isolated Environments

ConfigMaps and Secrets

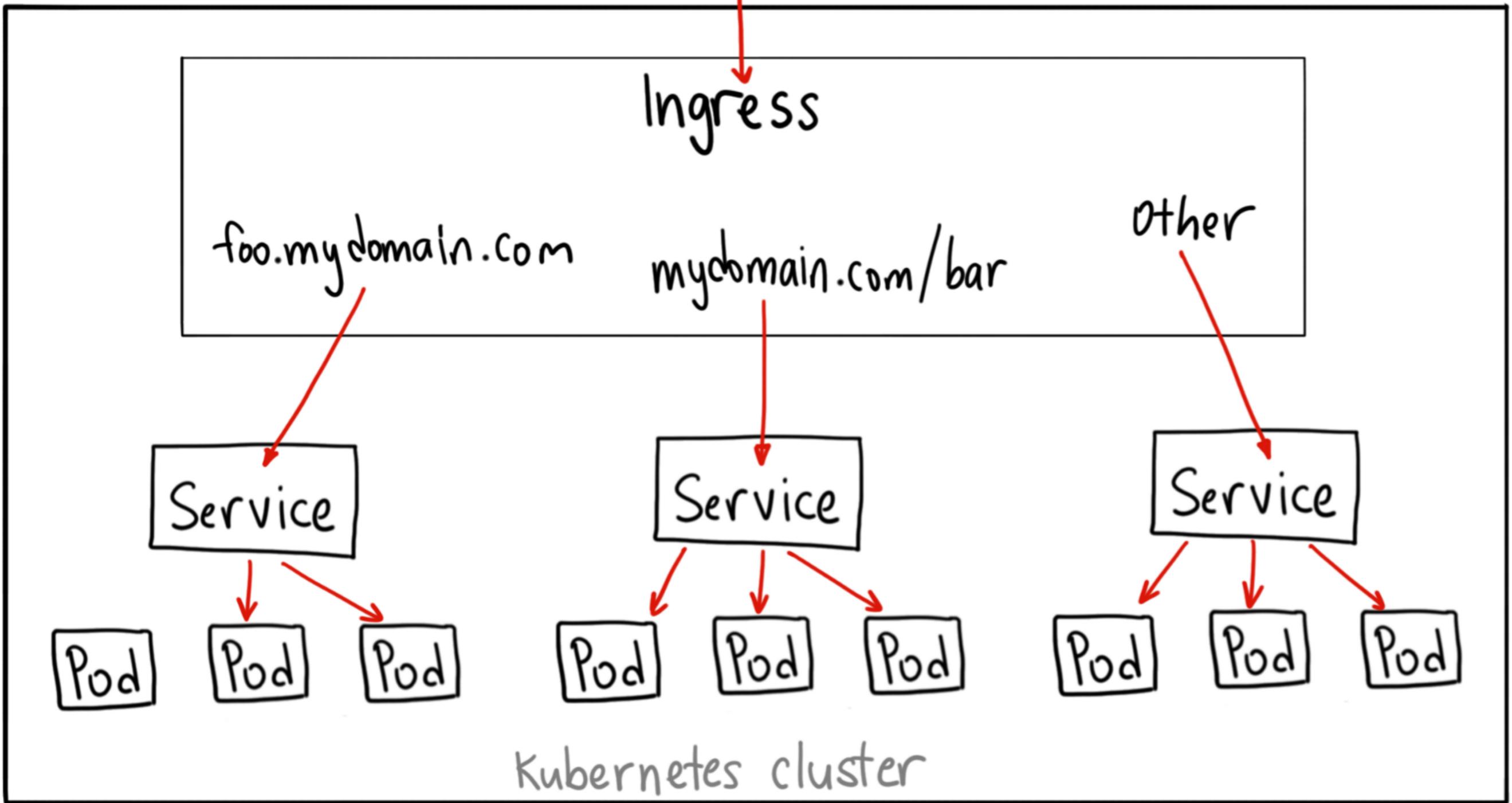
Decouple configs artifacts from code

StatefulSets

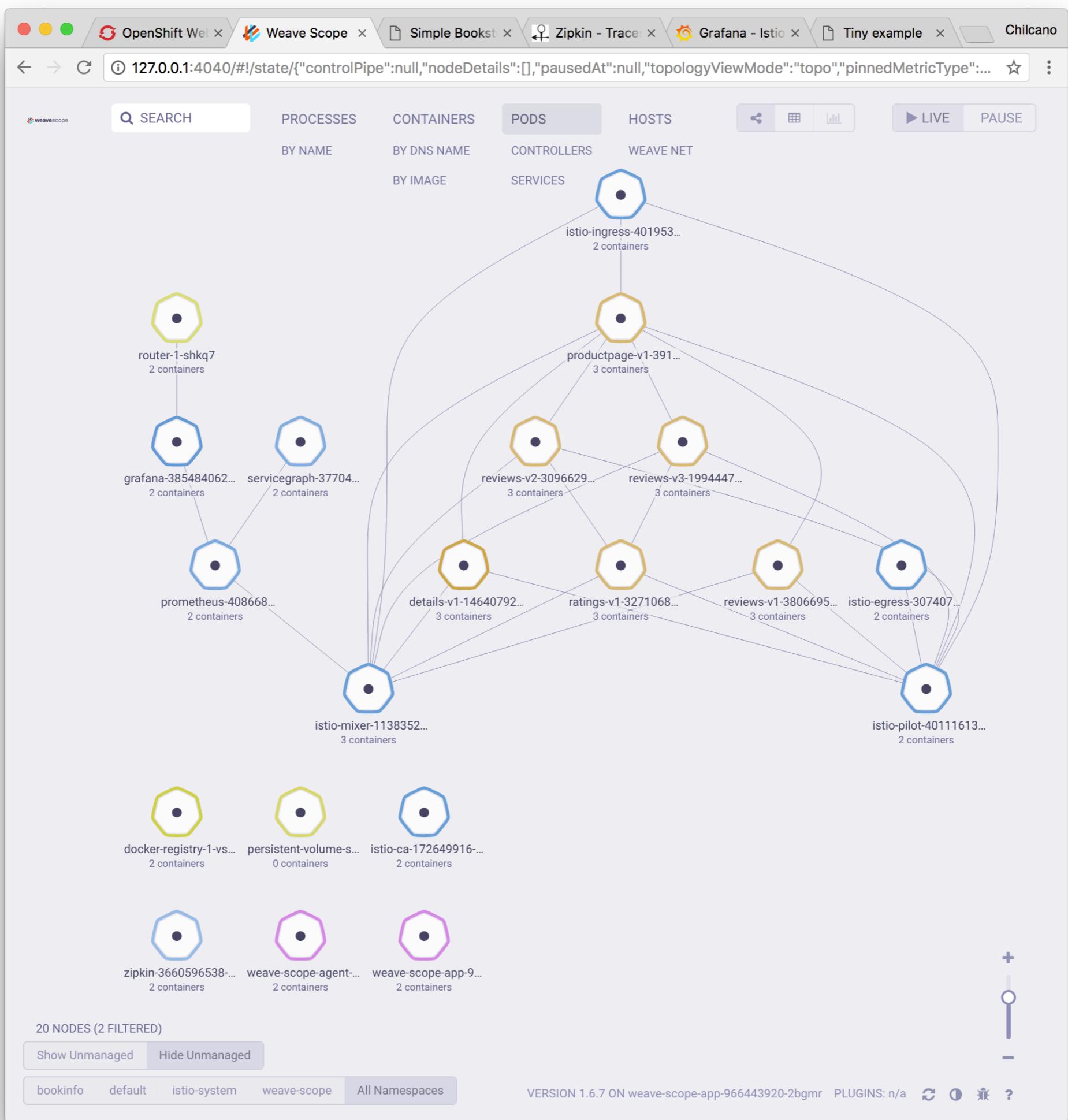
Manage stateful applications

DaemonSets

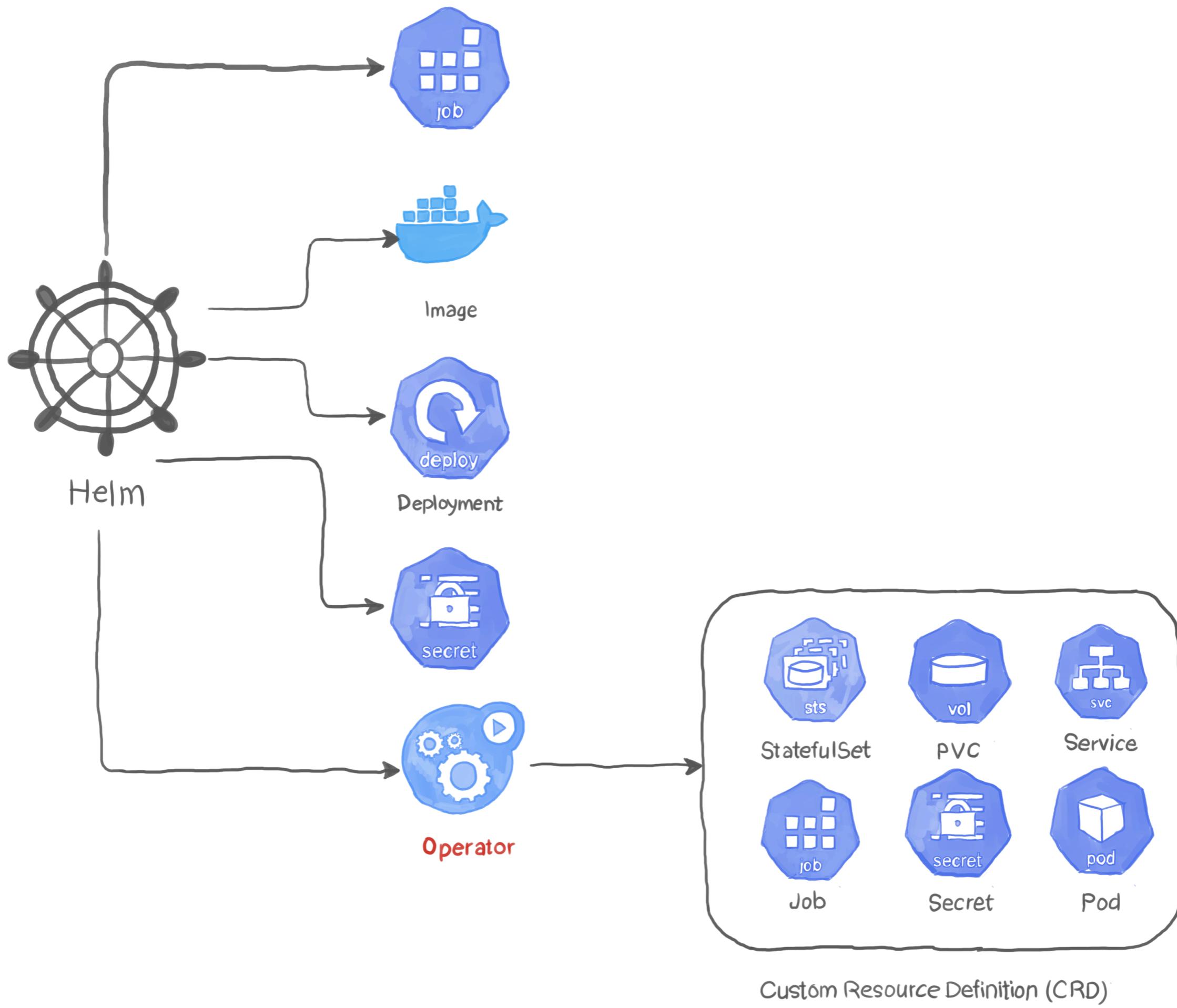
All Nodes run a copy of a Pod



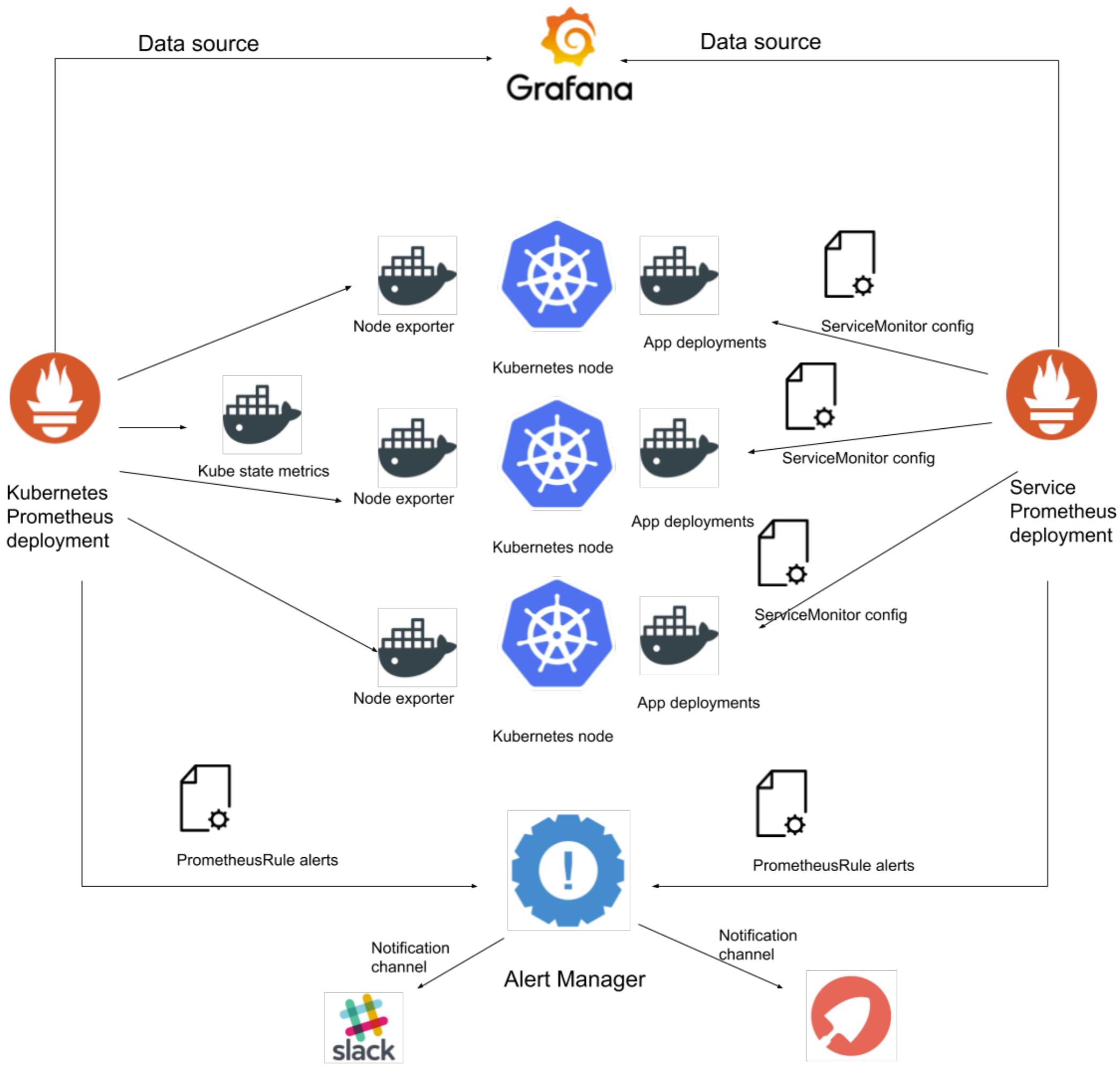
```
kubectl apply -f "https://  
cloud.weave.works/k8s/scope.yaml"
```



If Kubernetes is the Helmsman,
Helm is the Meta one



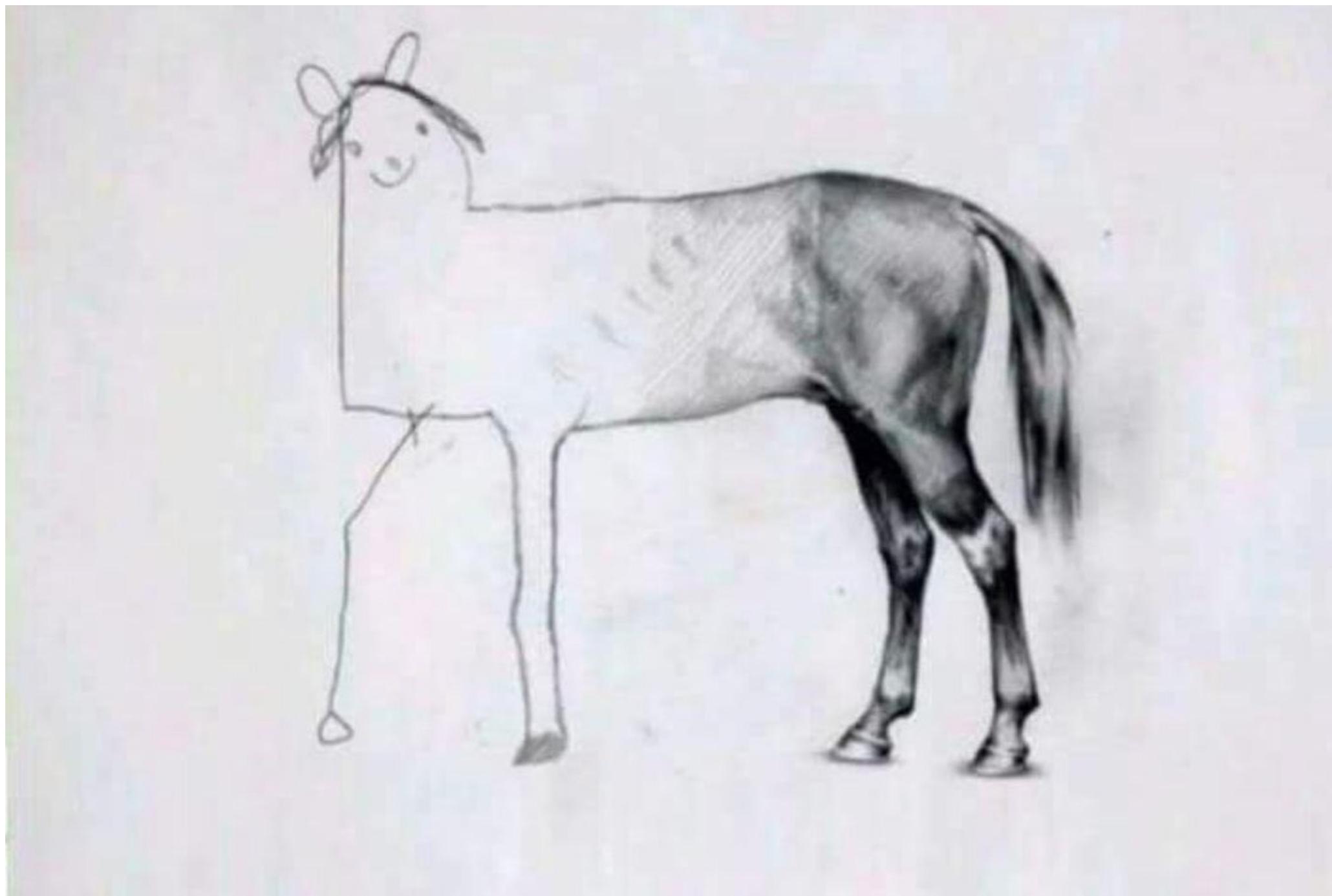
```
$ helm install stable/prometheus-operator  
--name prometheus-operator --namespace  
monitoring
```



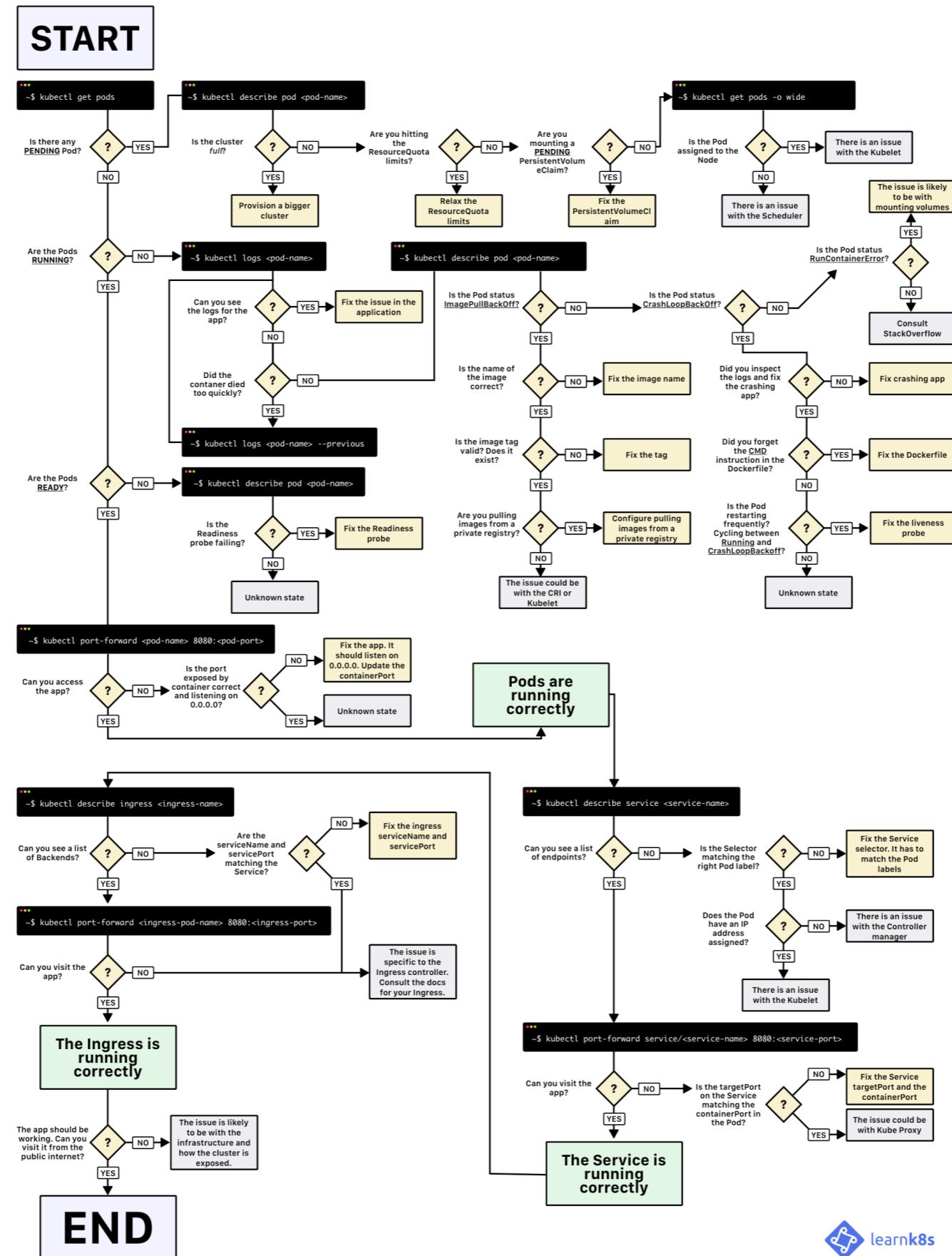
Challenges

**#challenge:
It easily works!
VS**

Production grade & secured cluster



#challenge: Complexity



#challenge: Performance

```
$ whoami  
ubuntu
```

```
$ whereami  
In a Docker container. Inside a  
Kubernetes cluster. Running in a VM.  
On top of a Hypervisor. In someone  
else's datacenter.
```

```
$ howdidigethere  
No. Fucking. Clue.
```

Live Demo

Resources

- All in one - <https://kubernetes.io/>

- Kubernetes presentations:

<https://github.com/cncf/presentations/tree/master/kubernetes>

- Awesome Kubernetes sources

<https://github.com/ramitsurana/awesome-kubernetes>

- <https://learnk8s.io/>

- Kubernetes Failure Stories

<https://k8s.af/>

Q&A