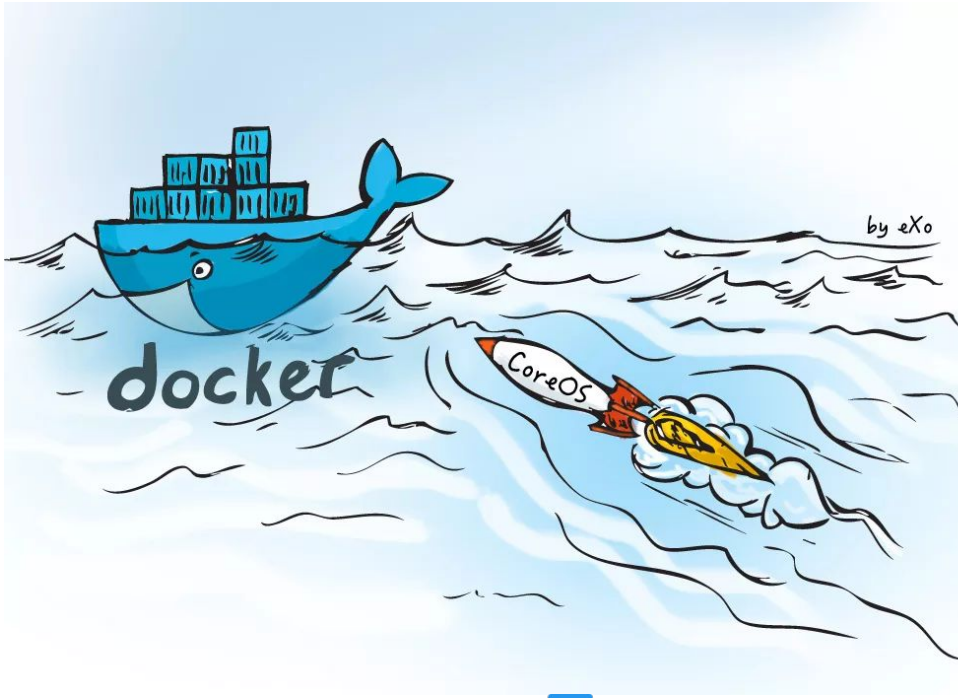
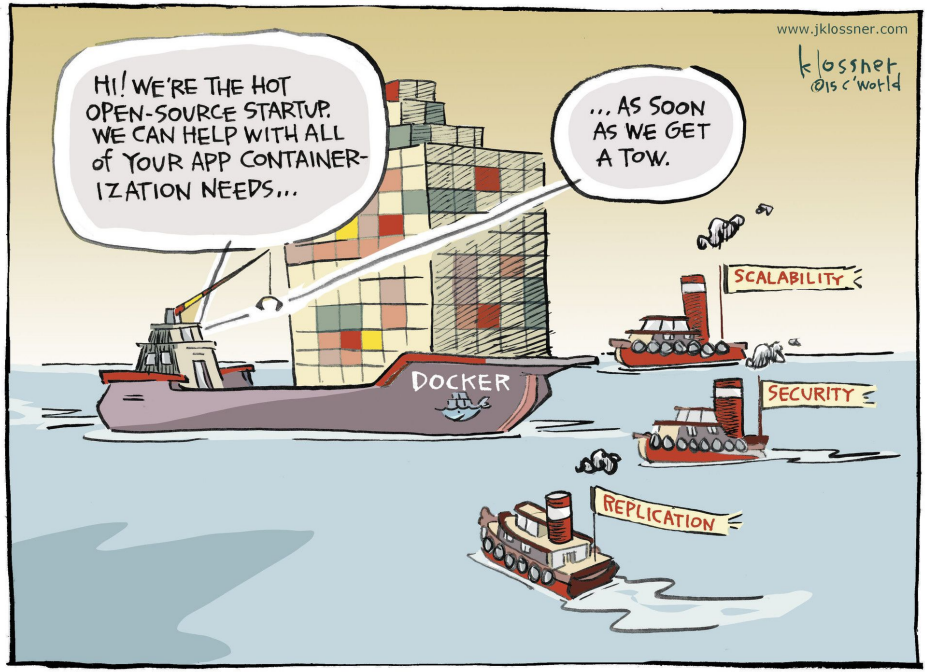
Kubernetes

Tomas Blanarik

VISMA

Containers





docker®

VISMA


Two Containers walk into a bar...
My manager tells me I have image problems
Layer
Layer
Layer
Layer
Layer
Layer
ROELIBOB IS

VISMA

> The director of an orchestra holds the vision for a musical performance and communicates with the musicians in order to coordinate their individual instrumental contributions to achieve this overall vision. As the architect of a system, your job is simply to compose the music (specify the containers to be run) and then hand over control to the orchestra director (container orchestration platform) to achieve that vision.

VISMA


WELL, WE'RE ALREADY DEPLOYING APPS IN CONTAINERS, BUT IT'S HARDLY FIXED EVERYTHING...
MANAGING THEM IS STILL SLOW, INEFFICIENT, AND FULL OF HOLES.
PUTTING APPS IN CONTAINERS IS A GREAT FIRST STEP—
—BUT NOW YOU NEED TO ORCHESTRATE THOSE PUPPIES!
POKE! POKE!
THAT'S WHERE KUBERNETES CAN HELP.

VISMA

Basic objects

config.yaml

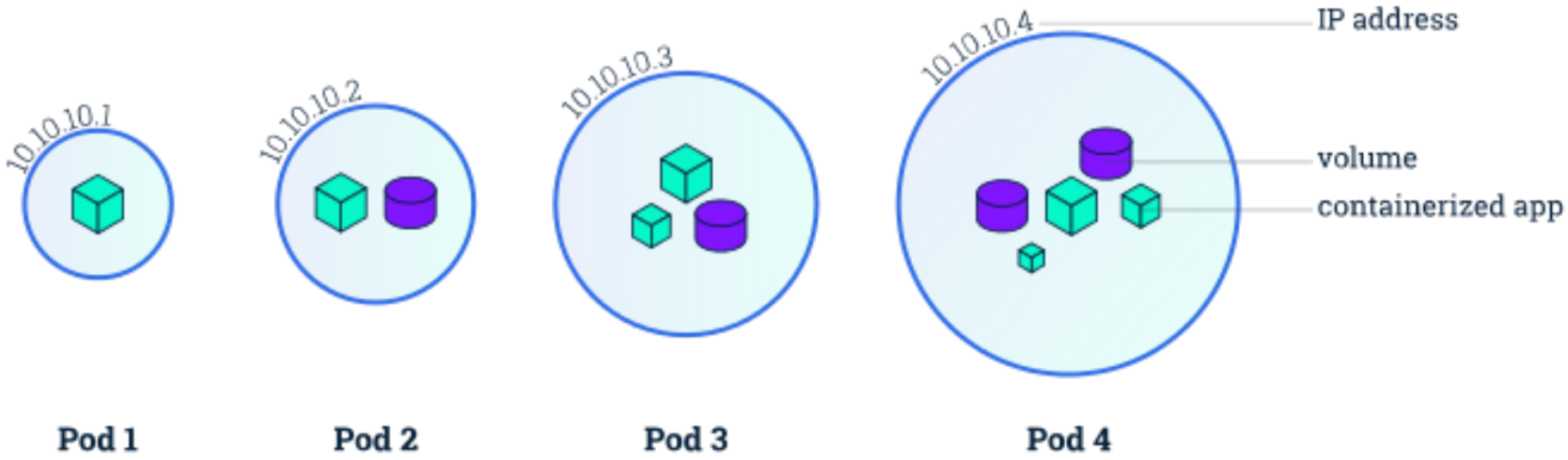
```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name:
  labels:
    app:
spec:
  replicas: 1
  template:
    metadata:
      name:
      labels:
        app:
    spec:
      containers:
      - name:
        image:
        imagePullPolicy: IfNotPresent
      restartPolicy: Always
  selector:
    matchLabels:
      app:
```

object1.yaml

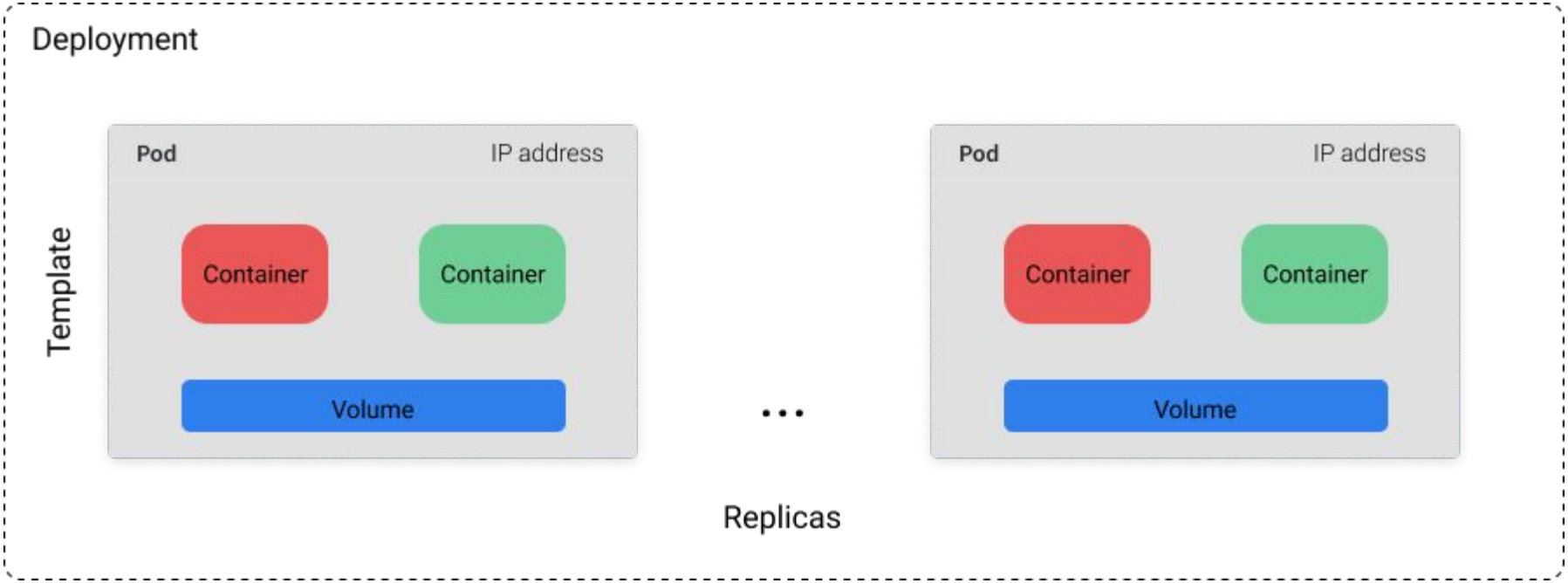
```yaml
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 3
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: hello-world
        image: hello-world:latest
        ports:
        - containerPort: 80
```

VISMA

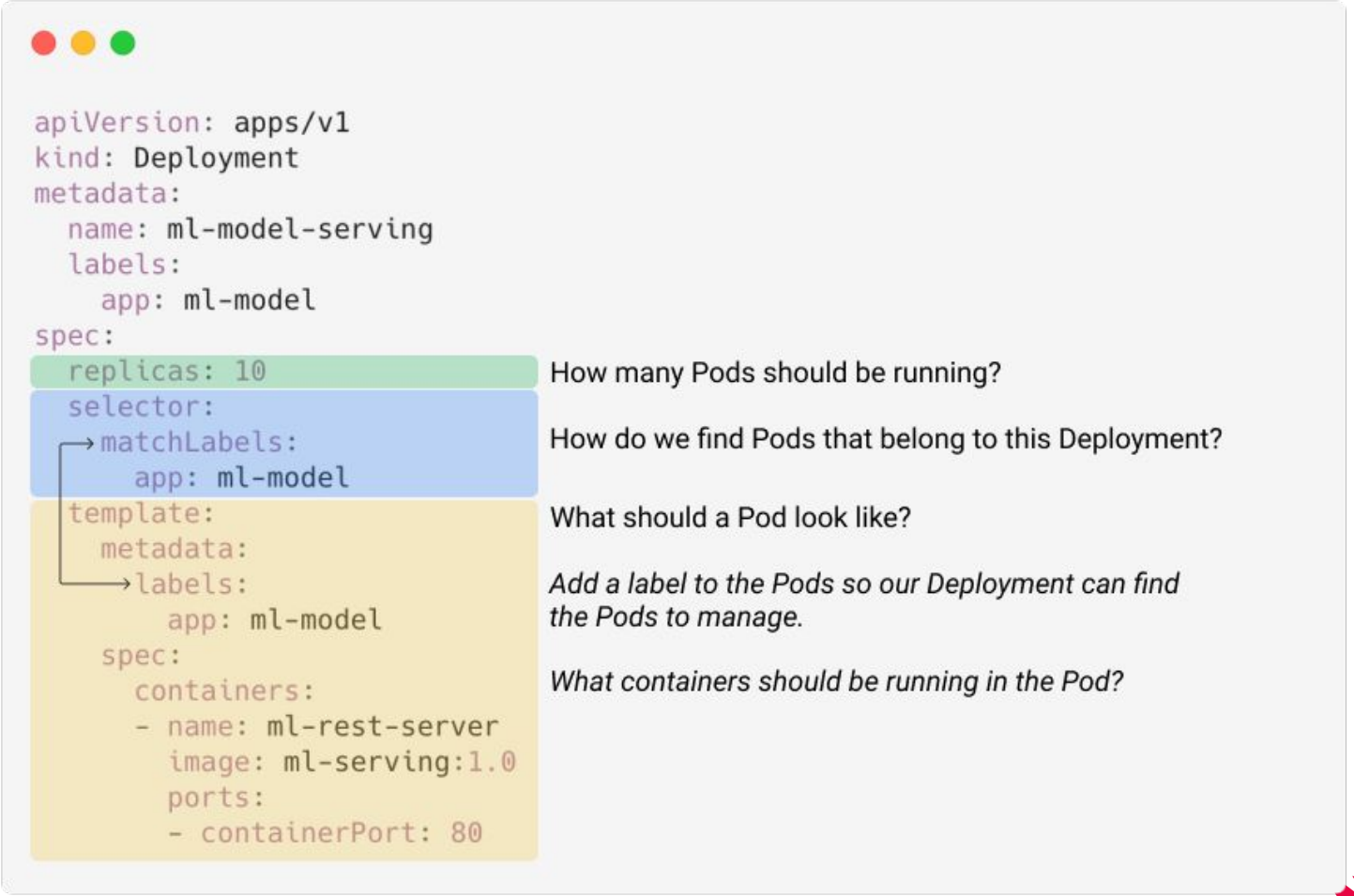
POD


10.10.10.1
10.10.10.2
10.10.10.3
10.10.10.4
IP address
volume
containerized app
Pod 1
Pod 2
Pod 3
Pod 4

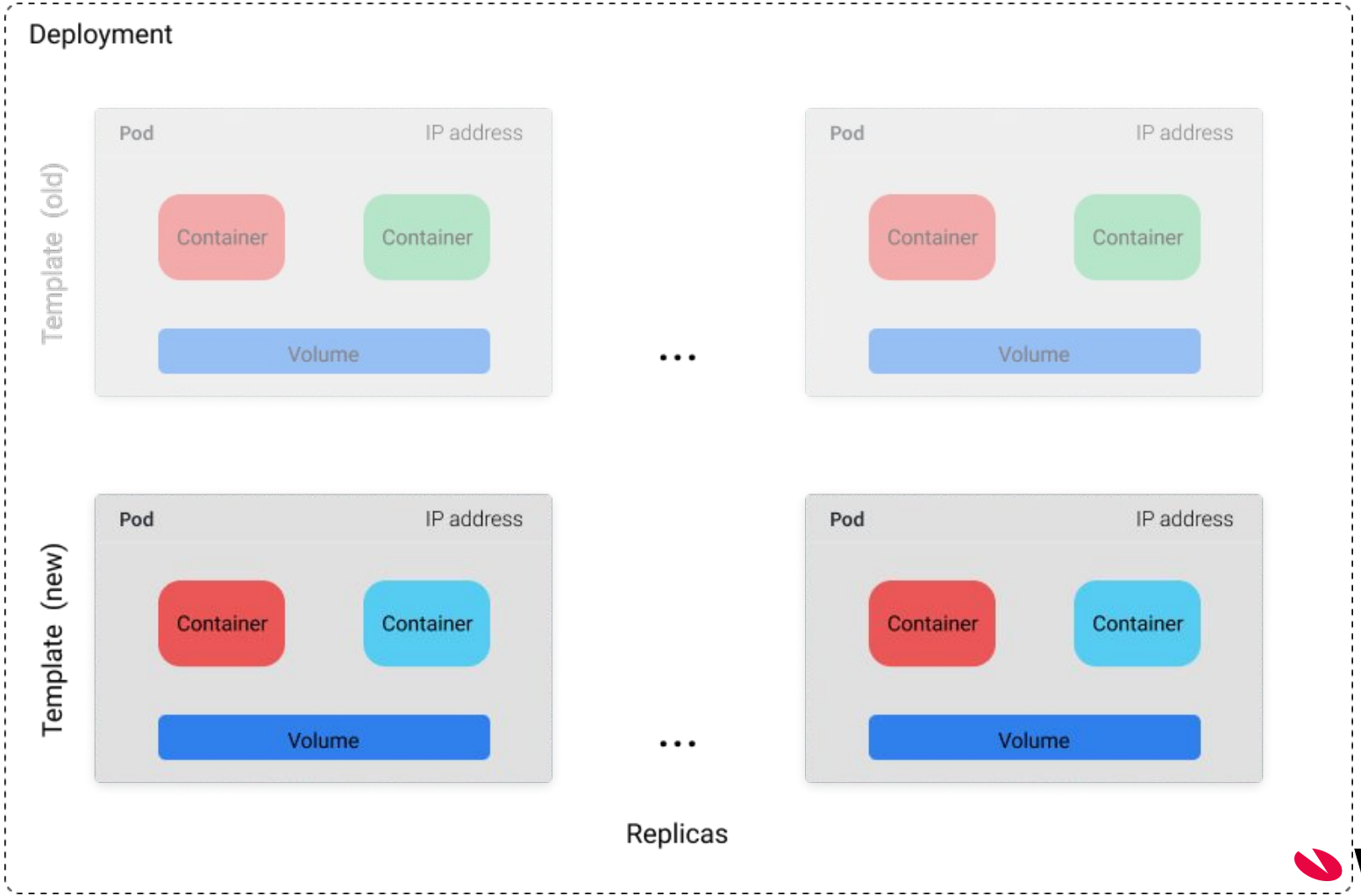
DEPLOYMENT



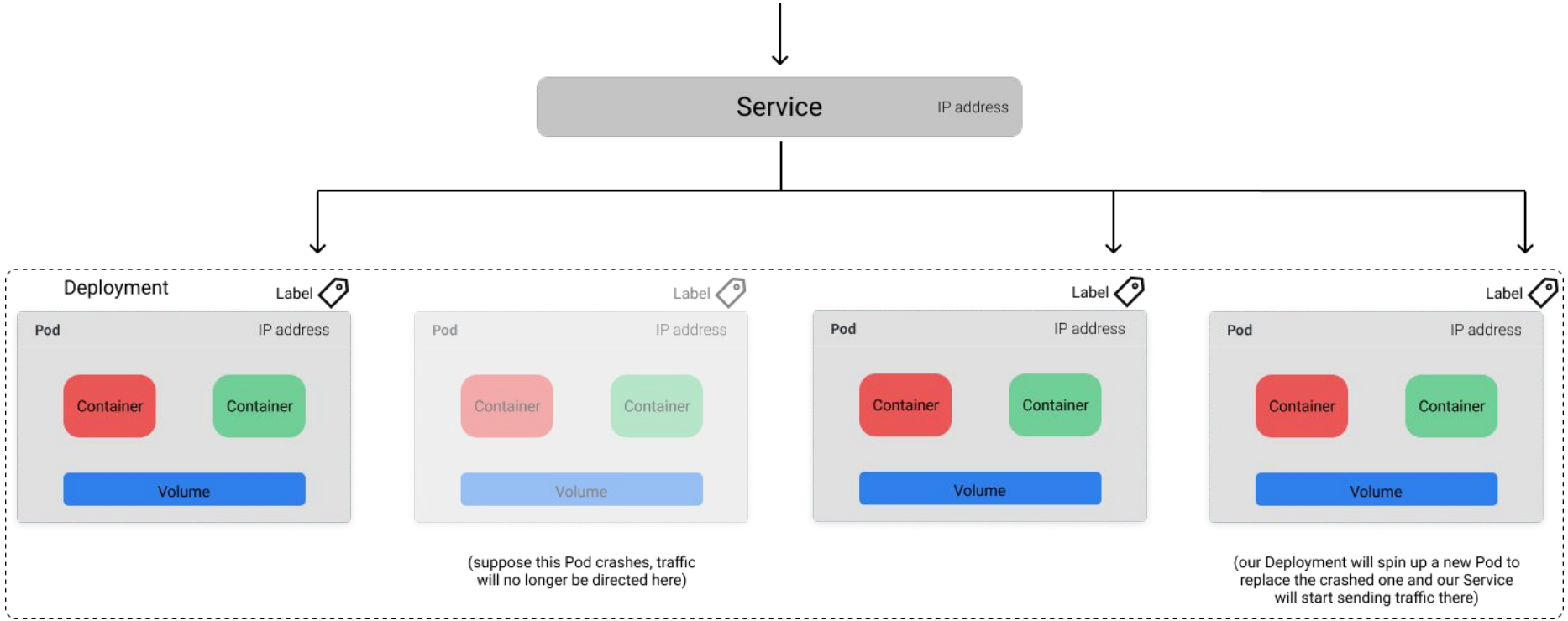
VISMA


apiVersion: apps/v1
kind: Deployment
metadata:
name: ml-model-serving
labels:
app: ml-model
spec:
replicas: 10
How many Pods should be running?
selector:
matchLabels:
app: ml-model
How do we find Pods that belong to this Deployment?
template:
metadata:
labels:
app: ml-model
What should a Pod look like?
Add a label to the Pods so our Deployment can find the Pods to manage.
spec:
containers:
- name: ml-rest-server
image: ml-serving:1.0
ports:
- containerPort: 80
What containers should be running in the Pod?

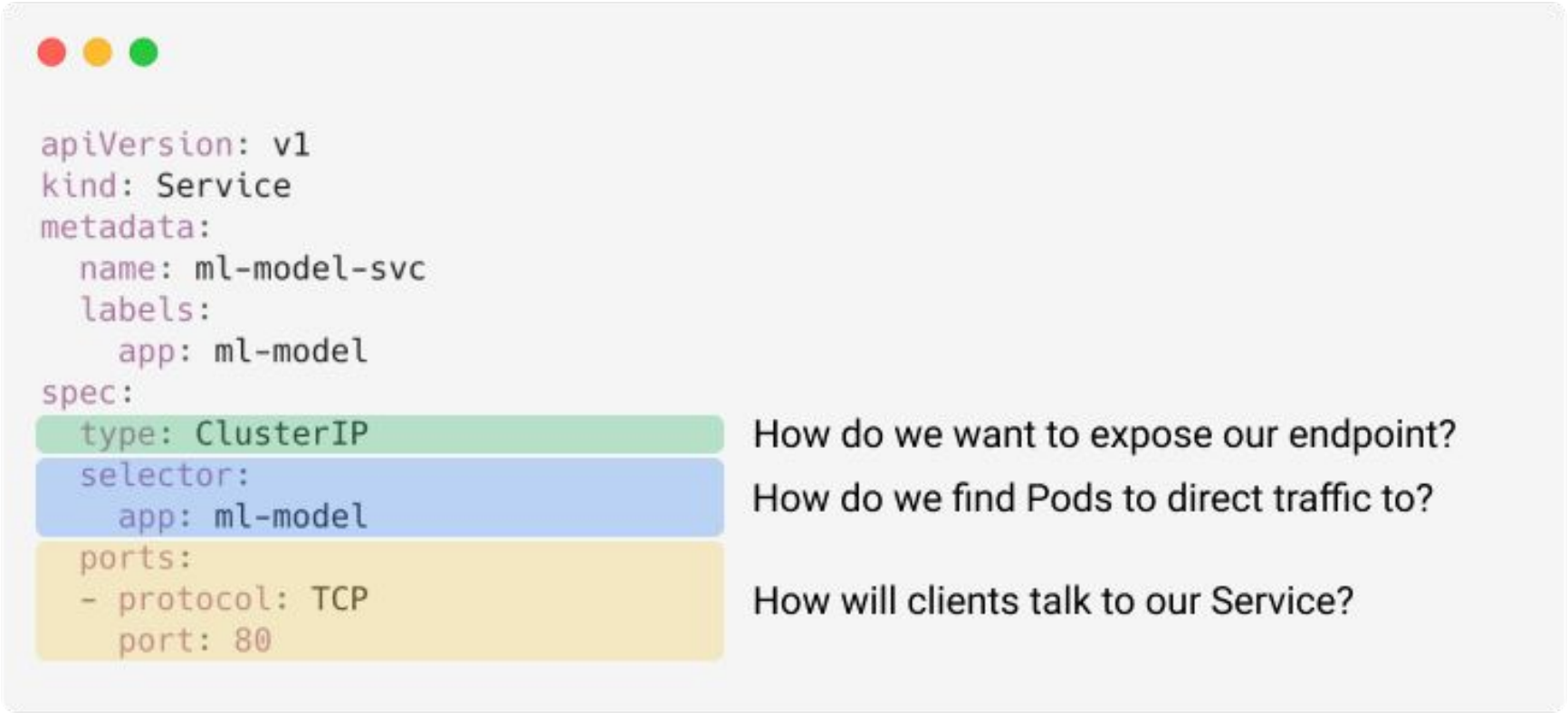
VISMA


Deployment
Template (old)
Pod
IP address
Container
Container
Volume
Pod
IP address
Container
Container
Volume
Template (new)
Pod
IP address
Container
Container
Volume
Pod
IP address
Container
Container
Volume
Replicas

SERVICE

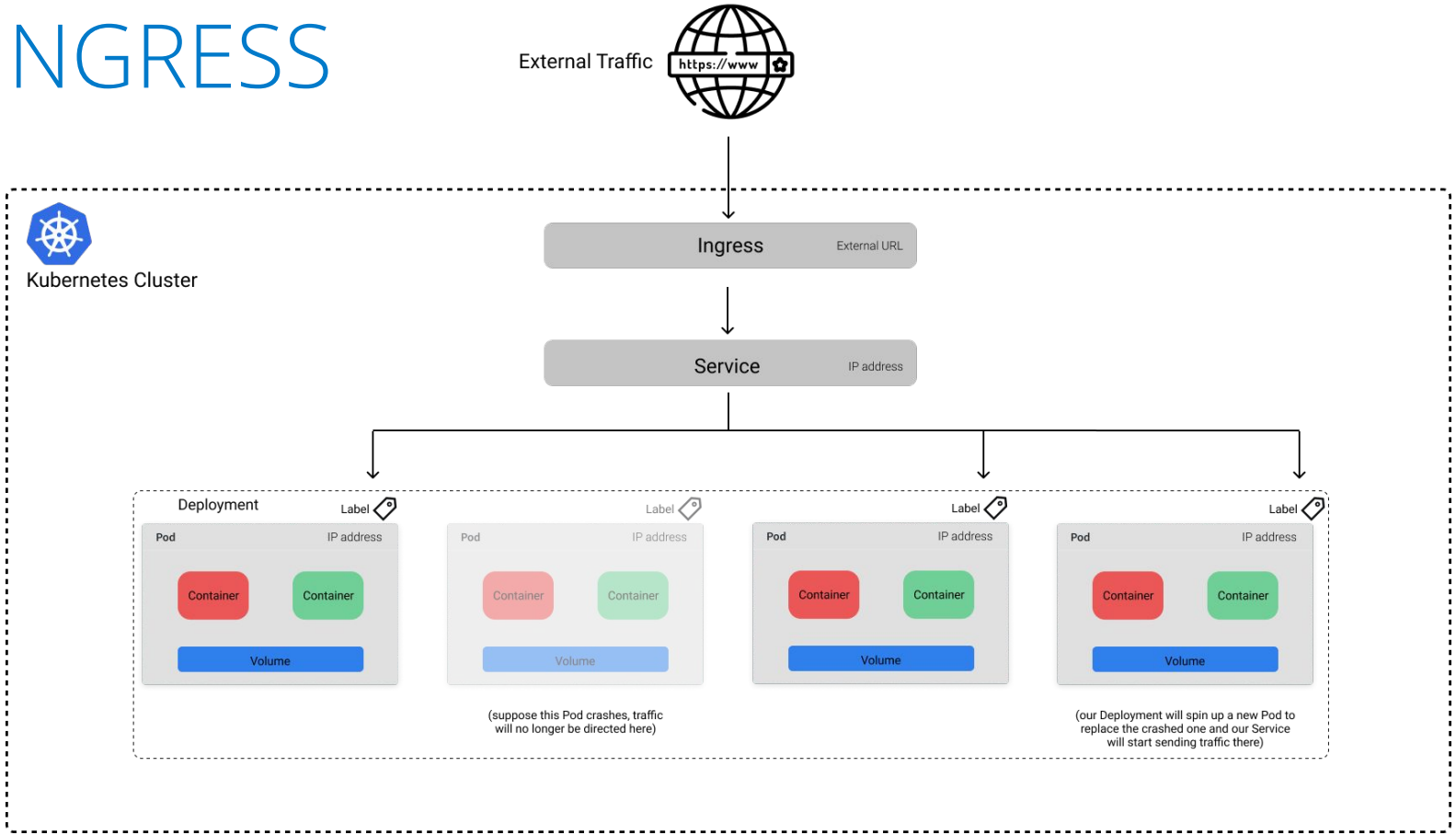


VISMA

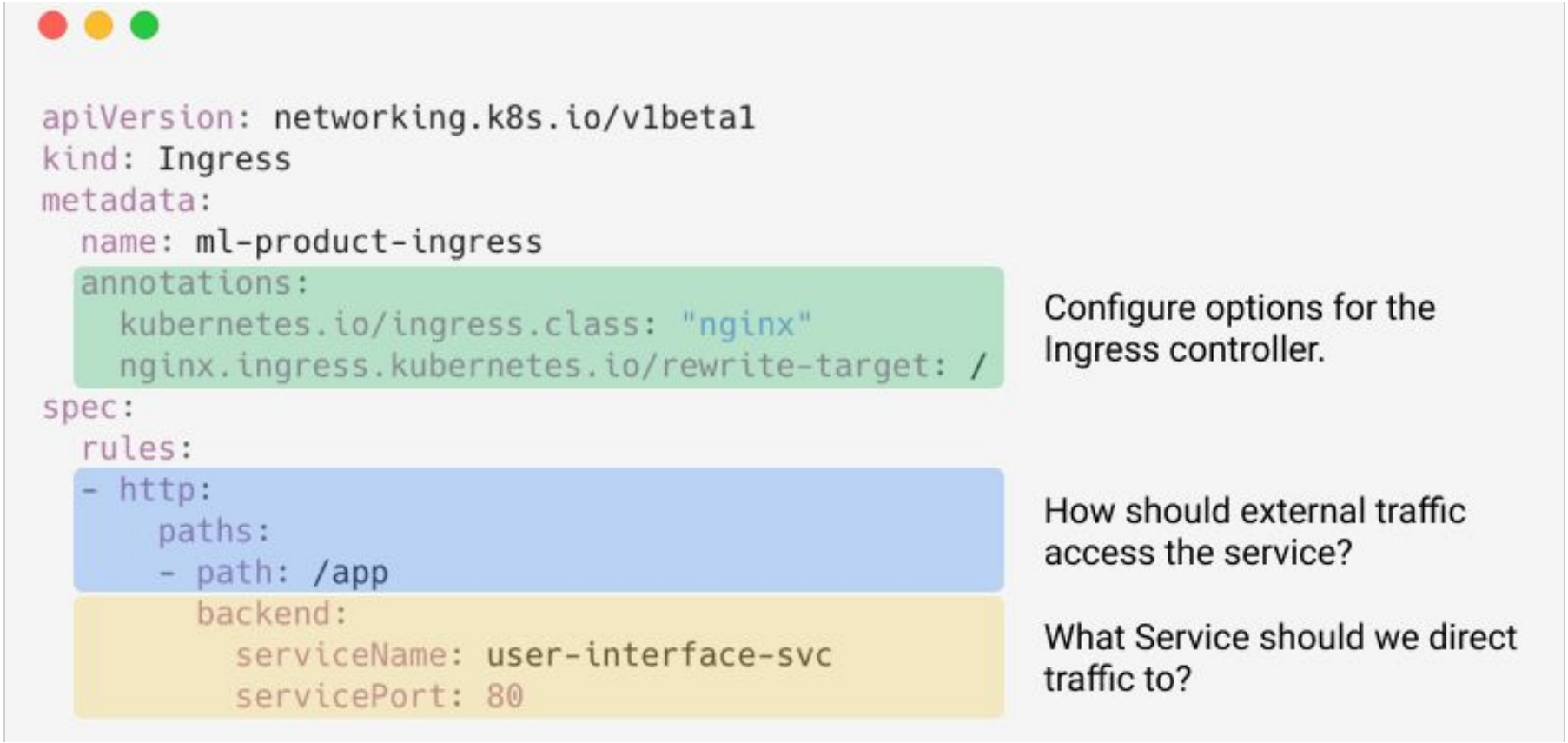

apiVersion: v1
kind: Service
metadata:
name: ml-model-svc
labels:
app: ml-model
spec:
type: ClusterIP
selector:
app: ml-model
ports:
- protocol: TCP
port: 80
How do we want to expose our endpoint?
How do we find Pods to direct traffic to?
How will clients talk to our Service?

VISMA

INGRESS



VISMA


apiVersion: networking.k8s.io/v1beta1
kind: Ingress
metadata:
name: ml-product-ingress
annotations:
kubernetes.io/ingress.class: "nginx"
nginx.ingress.kubernetes.io/rewrite-target: /
spec:
rules:
- http:
paths:
- path: /app
backend:
serviceName: user-interface-svc
servicePort: 80
Configure options for the Ingress controller.
How should external traffic access the service?
What Service should we direct traffic to?

VISMA

JOB

```yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: model-train-job
spec:
  template:
    spec:
      containers:
      - name: ml-model-train
        image: ml-development:1.2.1
        command: ["python",  "train.py"]
      restartPolicy: Never
  backoffLimit: 5
```

How should we create the Pod for this Job?

How many times should we try to complete this Job before quitting?

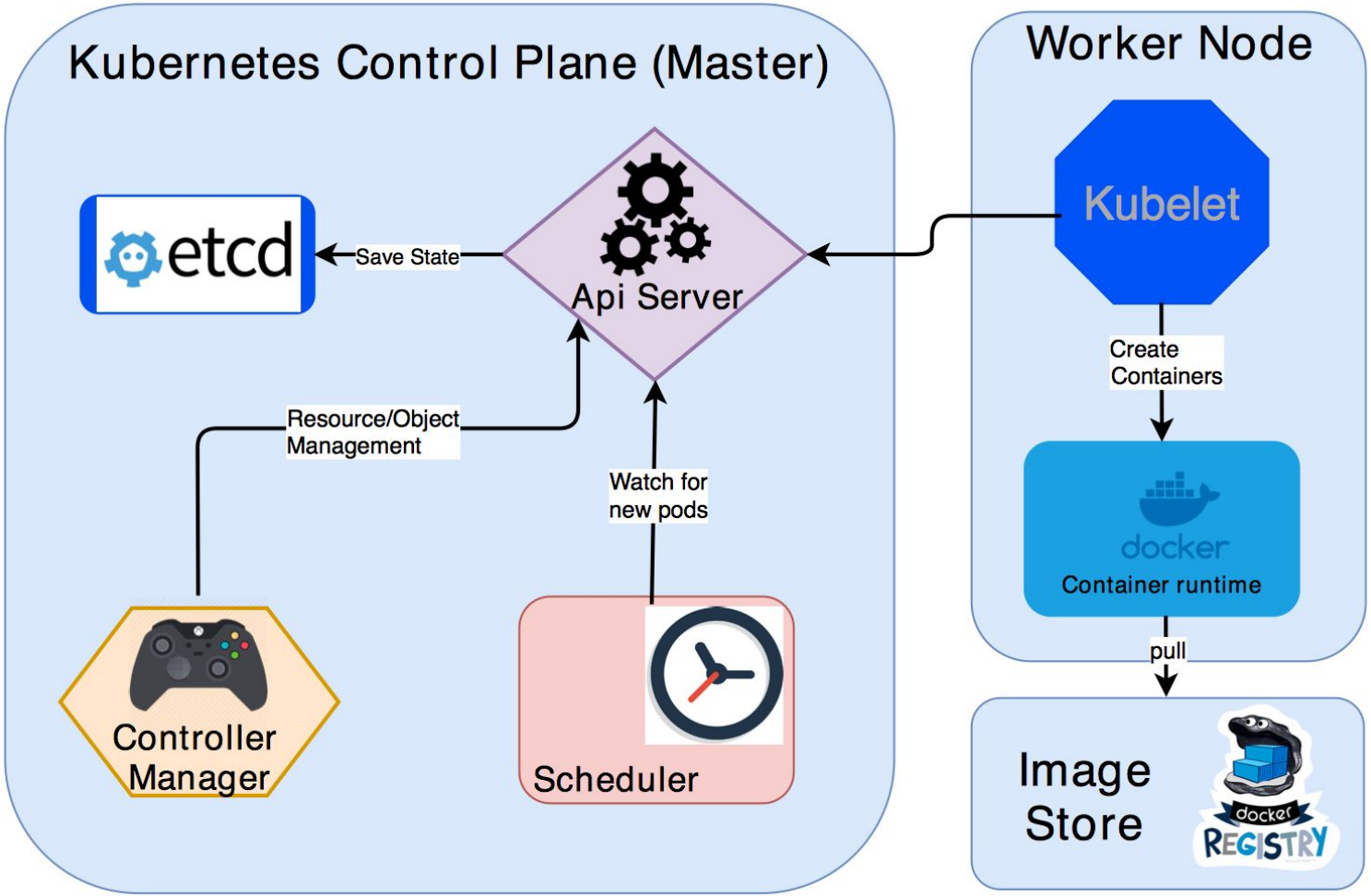
VISMA

… and more

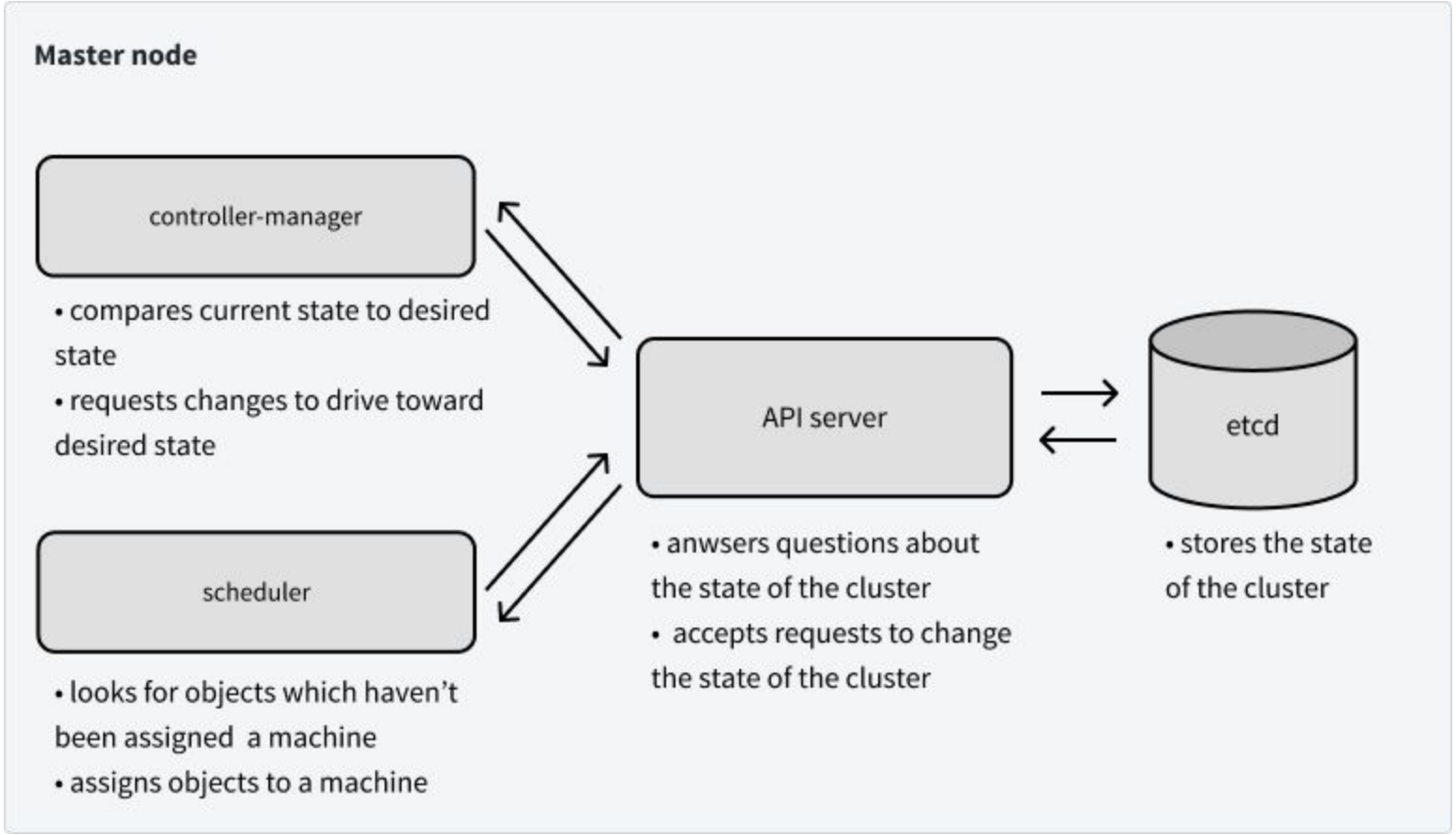
VISMA

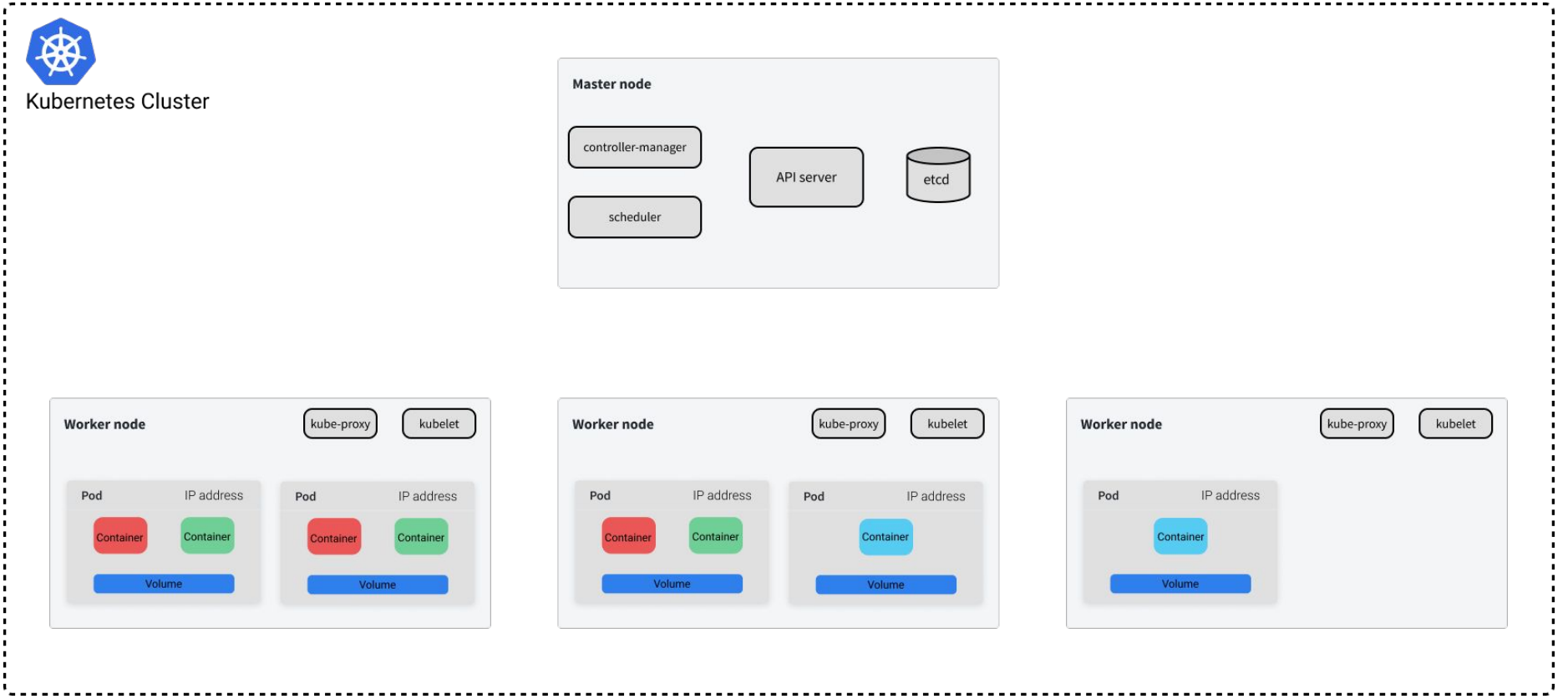
How? K8S Control Plane

VISMA


Kubernetes Control Plane (Master)
etcd
Save State
Api Server
Resource/Object Management
Watch for new pods
Controller Manager
Scheduler
Worker Node
Kubelet
Create Containers
docker
Container runtime
pull
Image Store
docker
REGISTRY

VISMA


Master node
controller-manager
• compares current state to desired state
• requests changes to drive toward desired state
scheduler
• looks for objects which haven't been assigned a machine
• assigns objects to a machine
API server
• anwsers questions about the state of the cluster
• accepts requests to change the state of the cluster
etcd
• stores the state of the cluster


Kubernetes Cluster
Master node
controller-manager
scheduler
API server
etcd
Worker node
kube-proxy
kubelet
Pod
IP address
Container
Container
Volume
Pod
IP address
Container
Container
Volume
Worker node
kube-proxy
kubelet
Pod
IP address
Container
Container
Volume
Pod
IP address
Container
Volume
Worker node
kube-proxy
kubelet
Pod
IP address
Container
Volume

VISMA

Respect

Reliability

Innovation

Competence

Team spirit

VISMA