



阿里云

×



CLOUD NATIVE
COMPUTING FOUNDATION

云原生技术公开课

第 03 讲

Kubernetes 核心概念

李响 阿里巴巴资深技术专家



关注“阿里巴巴云原生”公众号
获取第一手技术资料



Agenda

- 什么是 Kubernetes
- Kubernetes 的架构
- Kubernetes 的核心概念与 API
- 尝试一下 Kubernetes 吧

1 什么是 Kubernetes

Kubernetes - 工业级容器编排平台



- **Kubernetes** 源于希腊语，意为 “舵手” 或 “飞行员”
- **k8s** 是通过将 8 个字母 “ubernete” 替换为 8 而导出的缩写

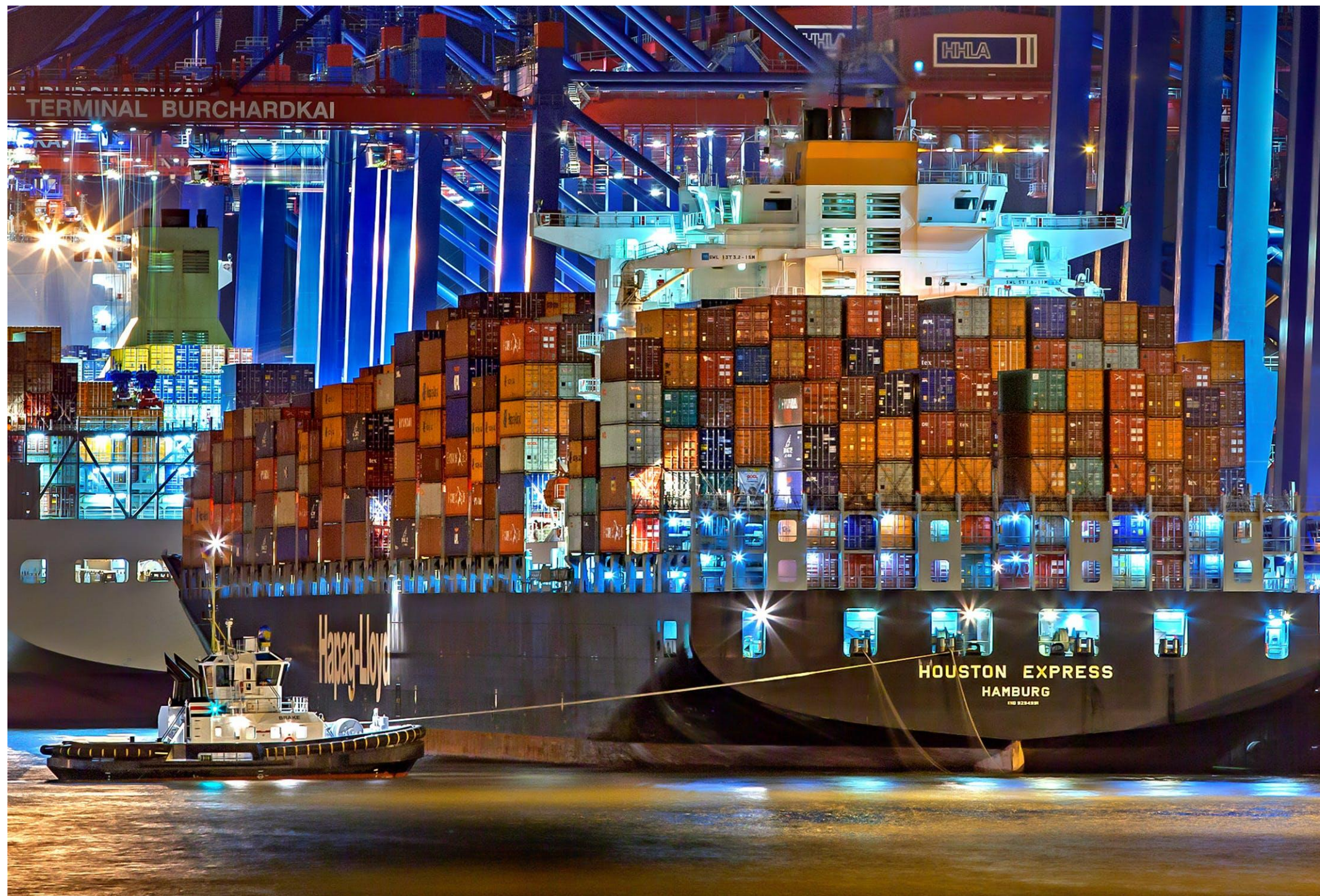


Photo by [Julius Silver](#) from [Pexels](#)

Kubernetes

自动化的容器编排平台

- 部署
- 弹性
- 管理

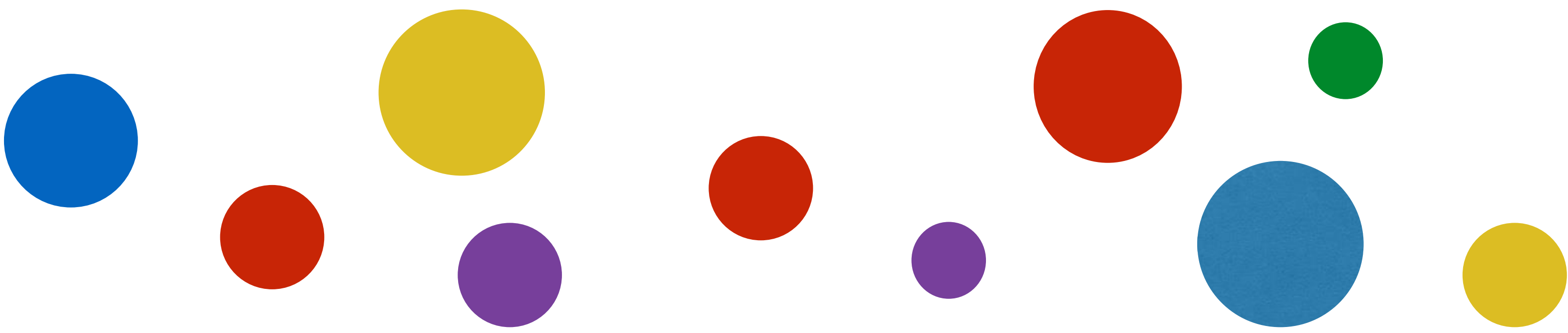


核心功能

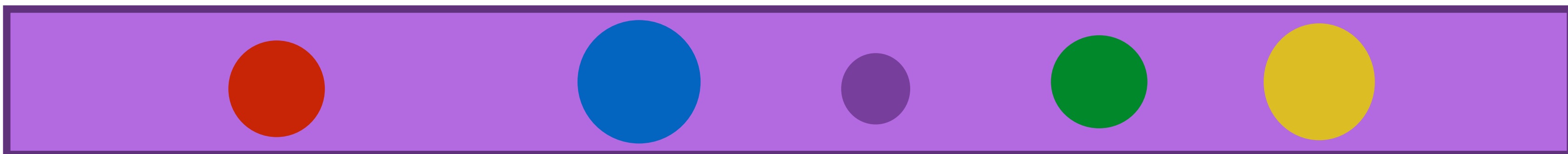
- 服务发现与负载均衡
- 容器自动装箱
- 存储编排
- 自动容器恢复
- 自动发布与回滚
- 配置与密文管理
- 批量执行
- 水平伸缩

Kubernetes - 调度

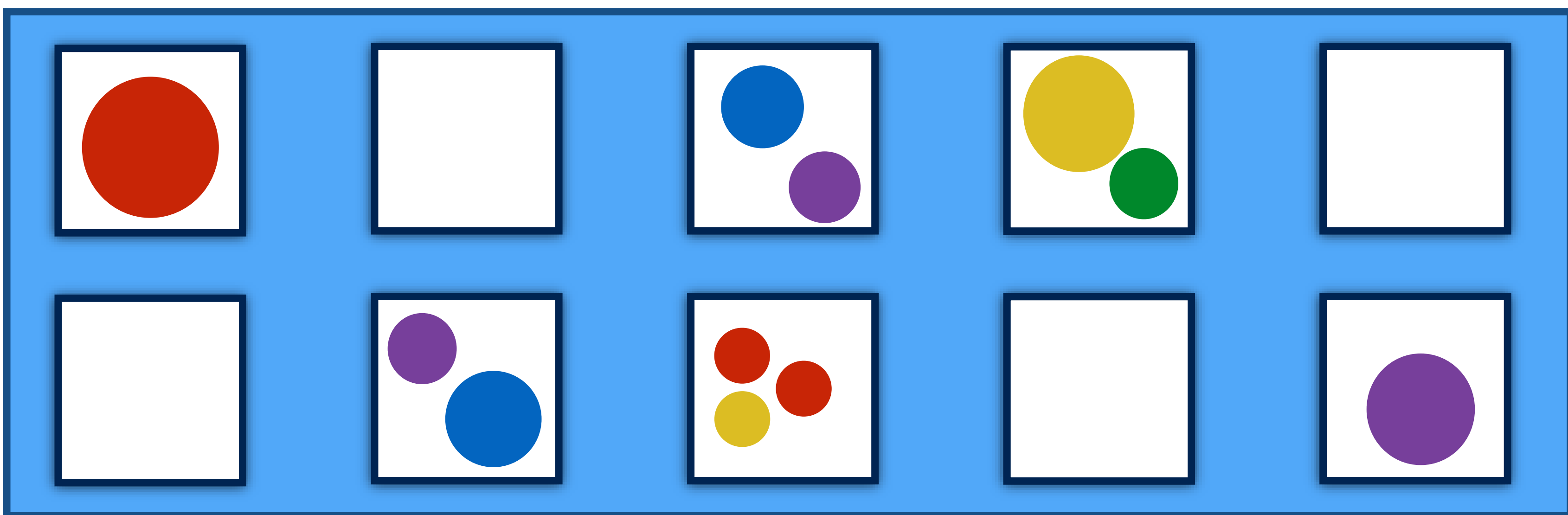
等待调度



正在调度



集群状态



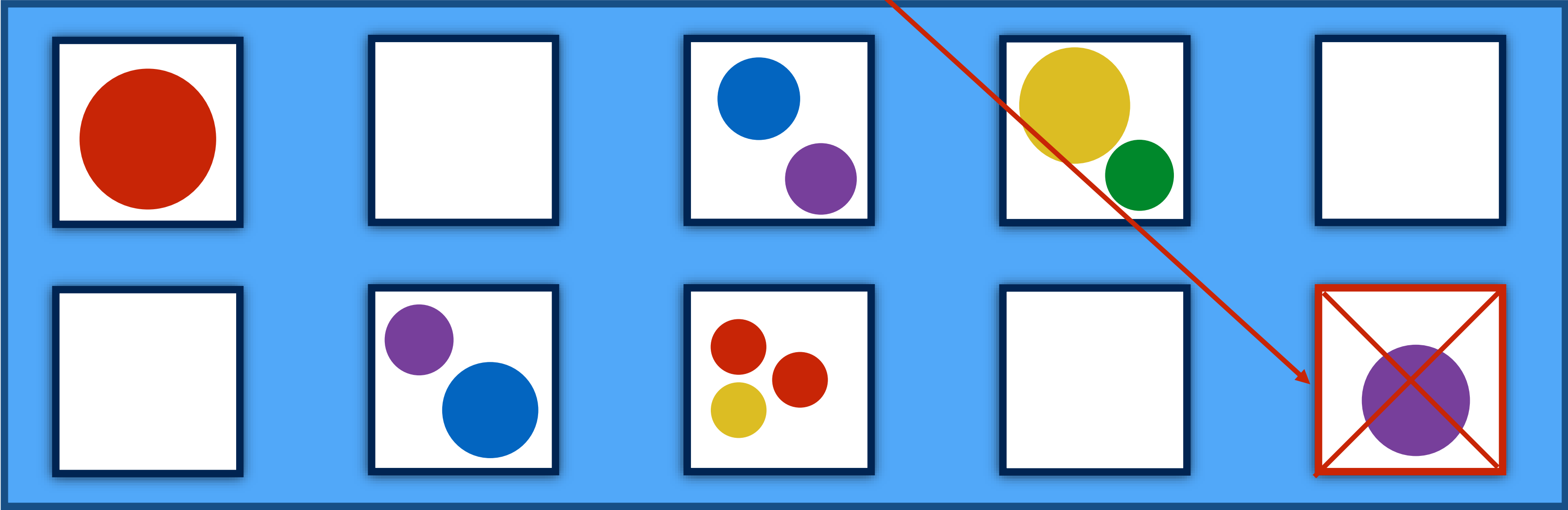
Kubernetes - 自动恢复

状态检查与修复

有一个节点无响应!

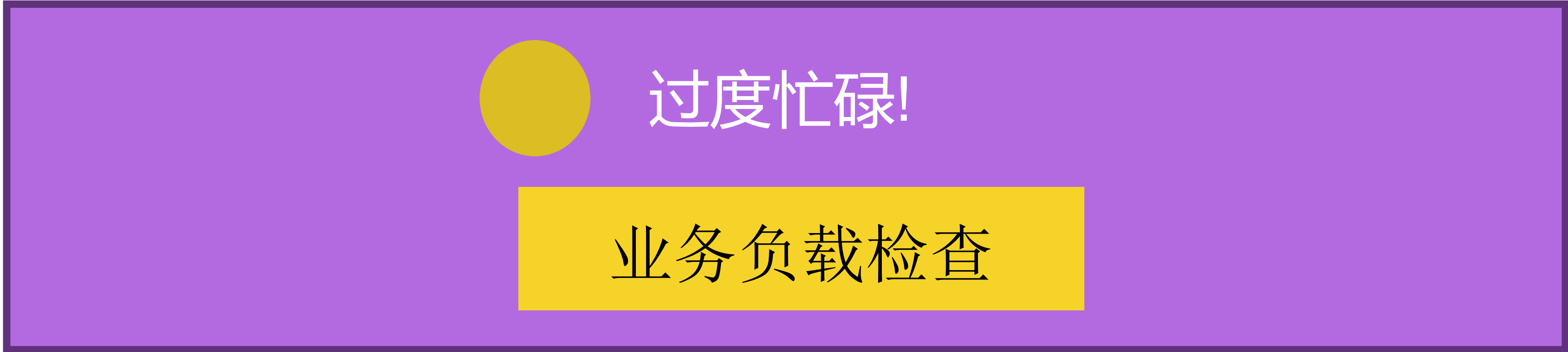
节点健康检查

集群状态

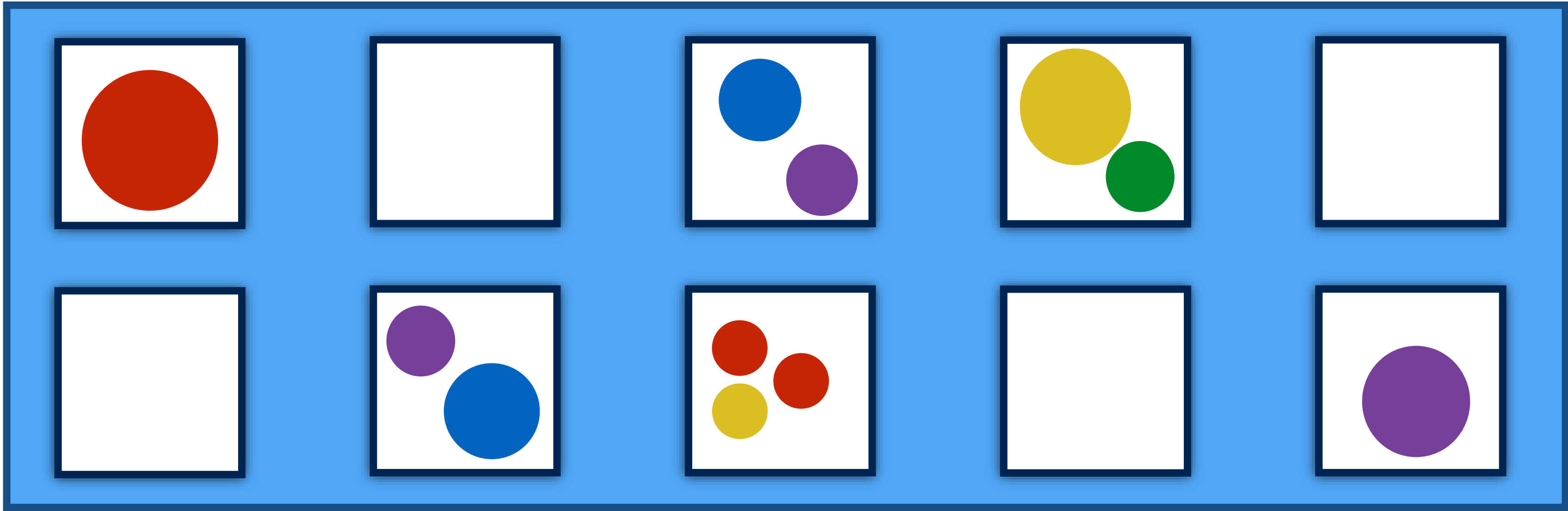


Kubernetes - 水平伸缩

状态检查与修复

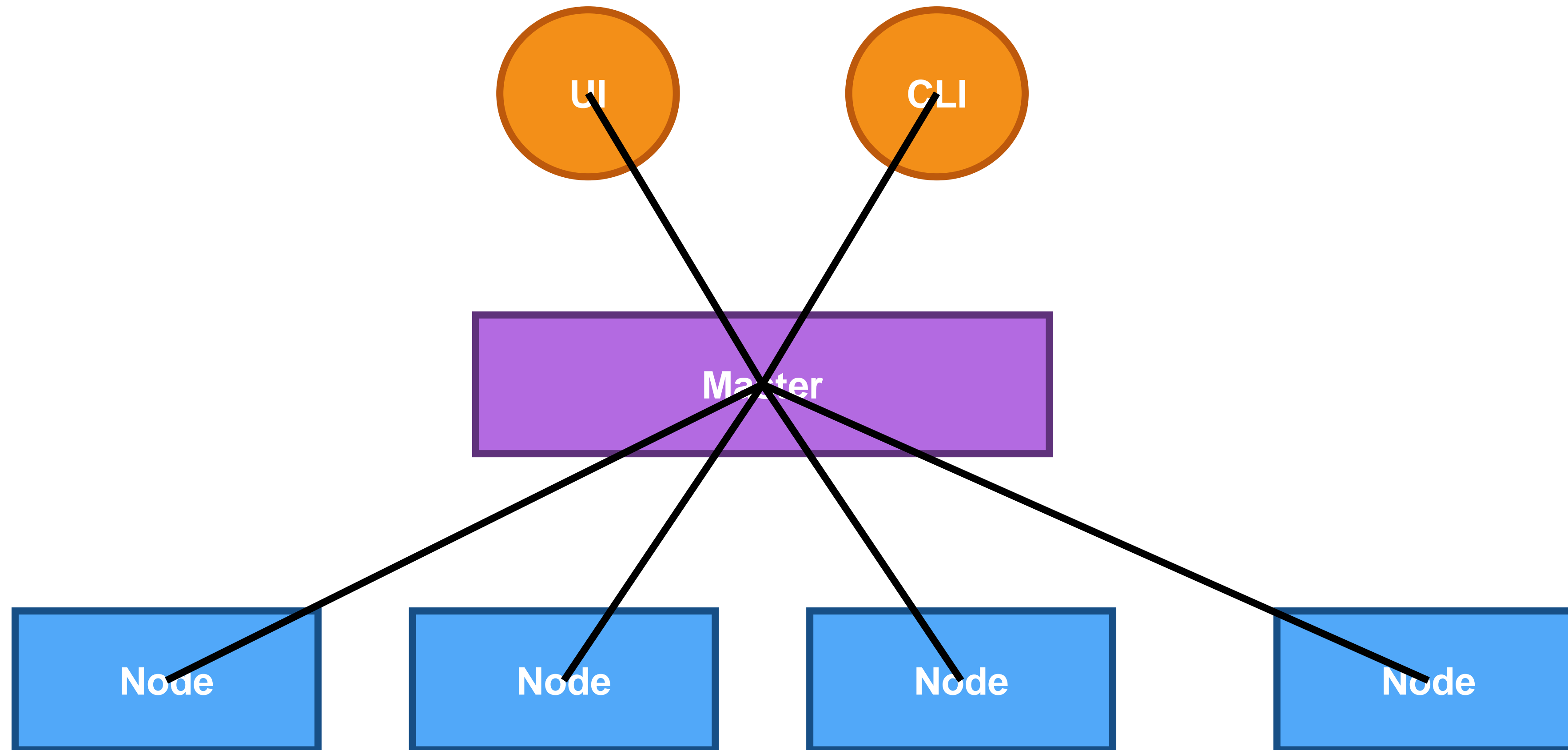


集群状态

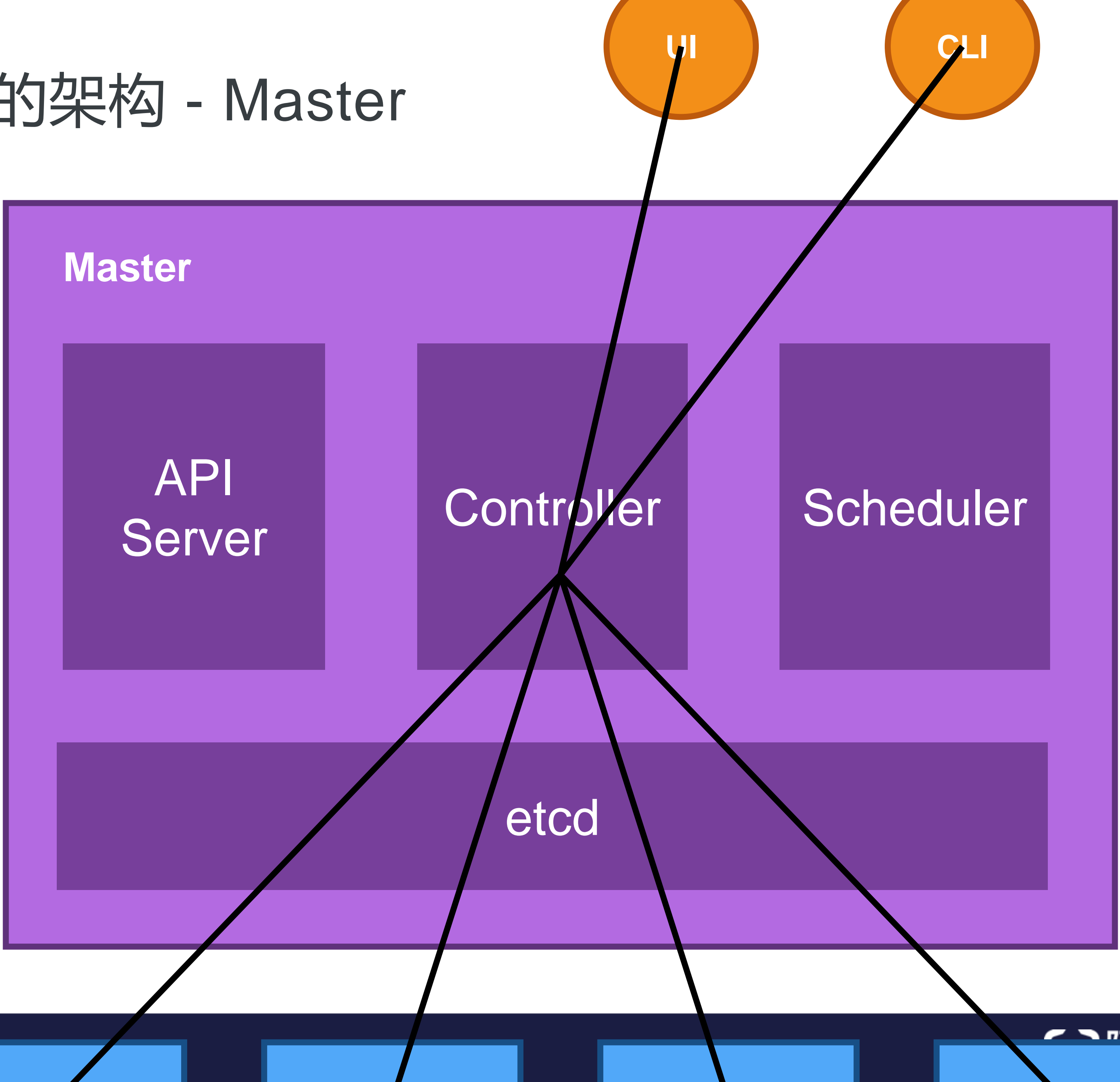


2 Kubernetes 的架构

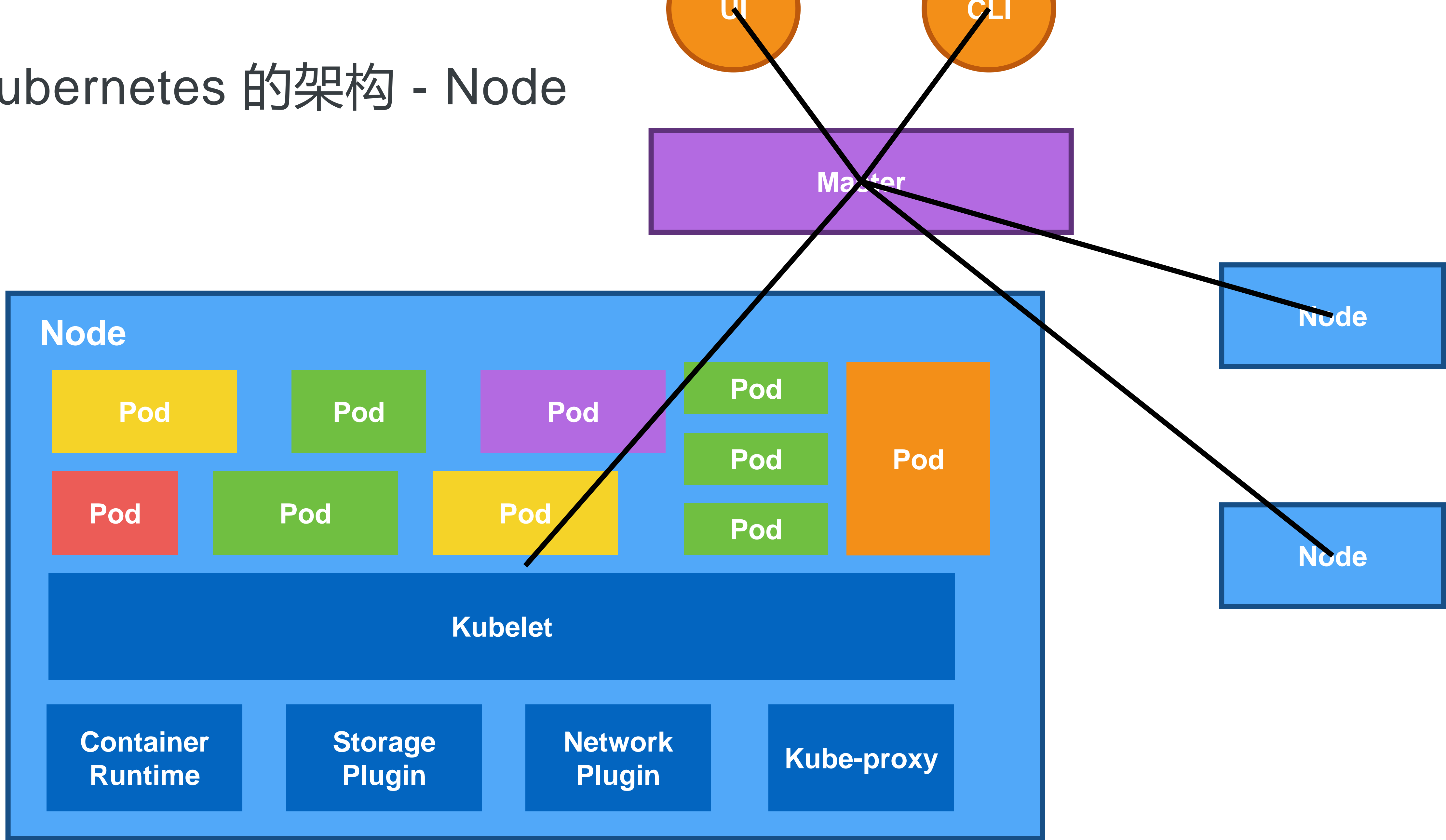
Kubernetes 的架构



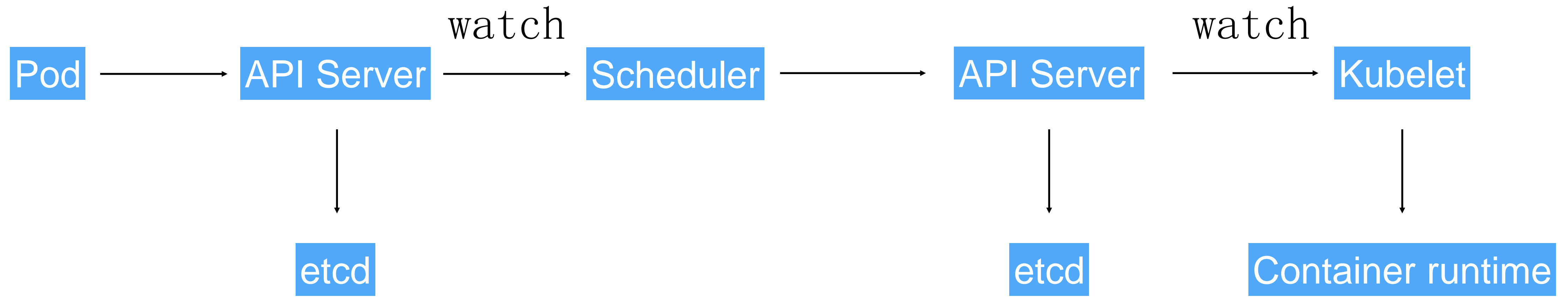
Kubernetes 的架构 - Master



Kubernetes 的架构 - Node



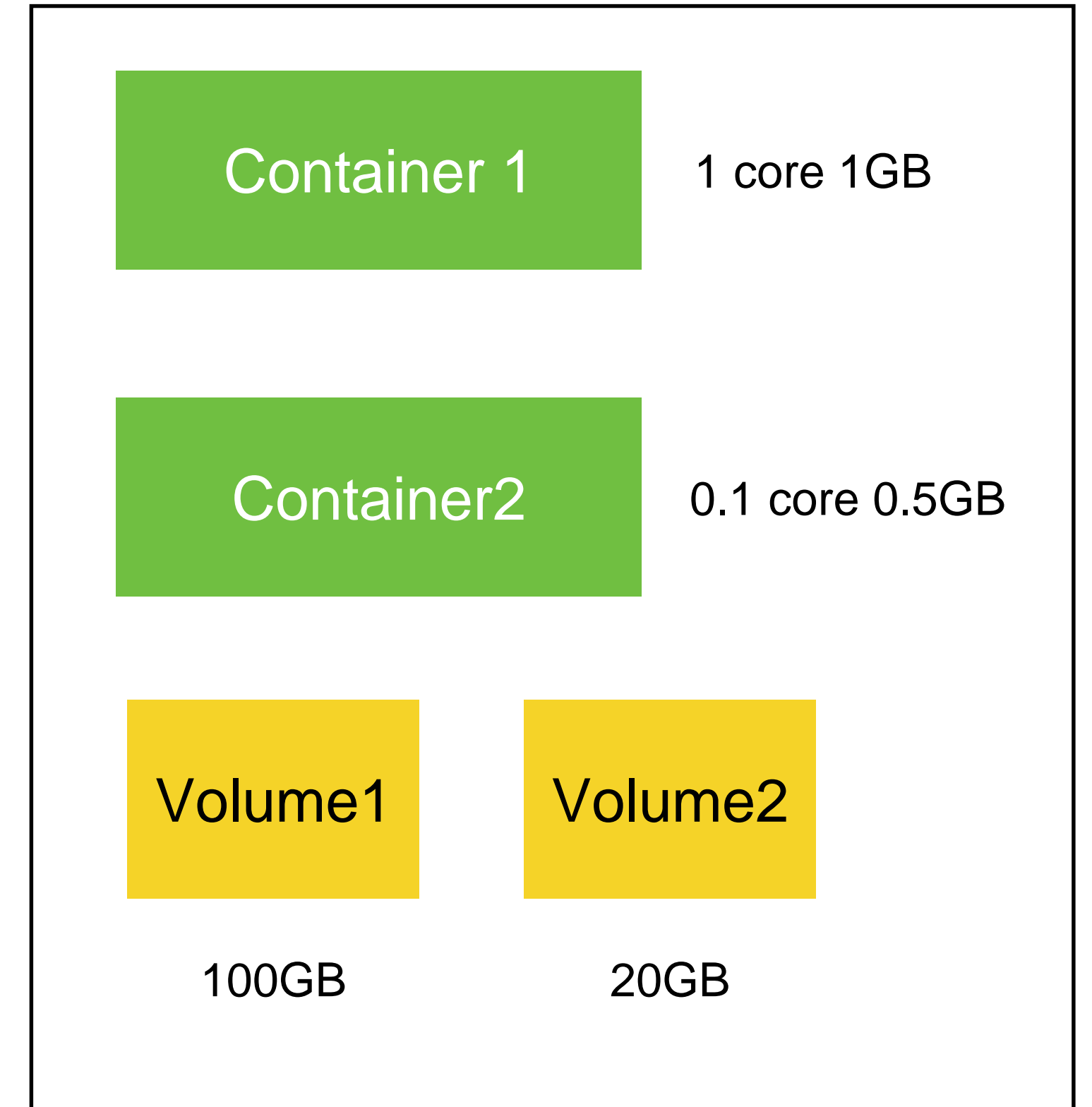
Kubernetes 的架构 - 例子



3 Kubernetes 的核心概念与 API

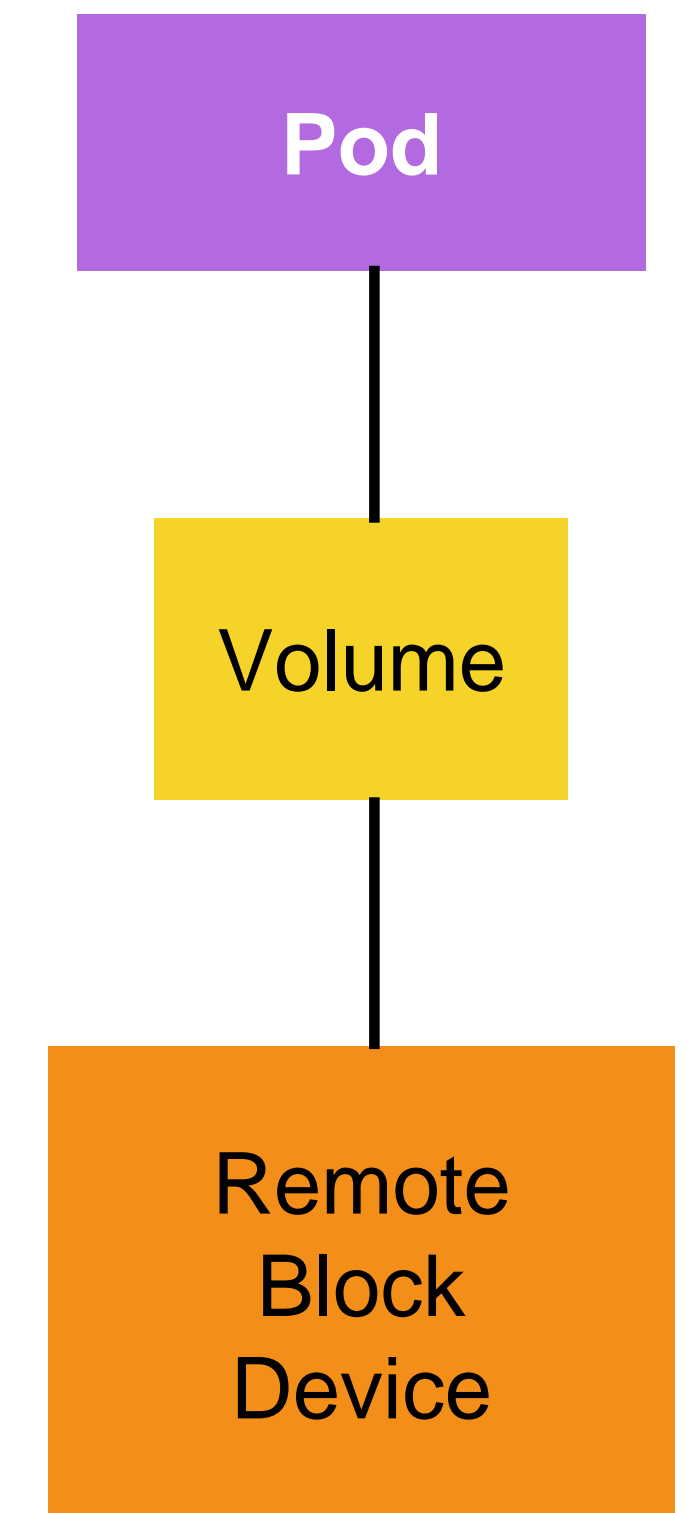
核心概念 - Pod

- 最小的调度以及资源单元
- 由一个或者多个容器组成
- 定义容器运行的方式（Command、环境变量等）
- 提供给容器共享的运行环境（网络、进程空间）



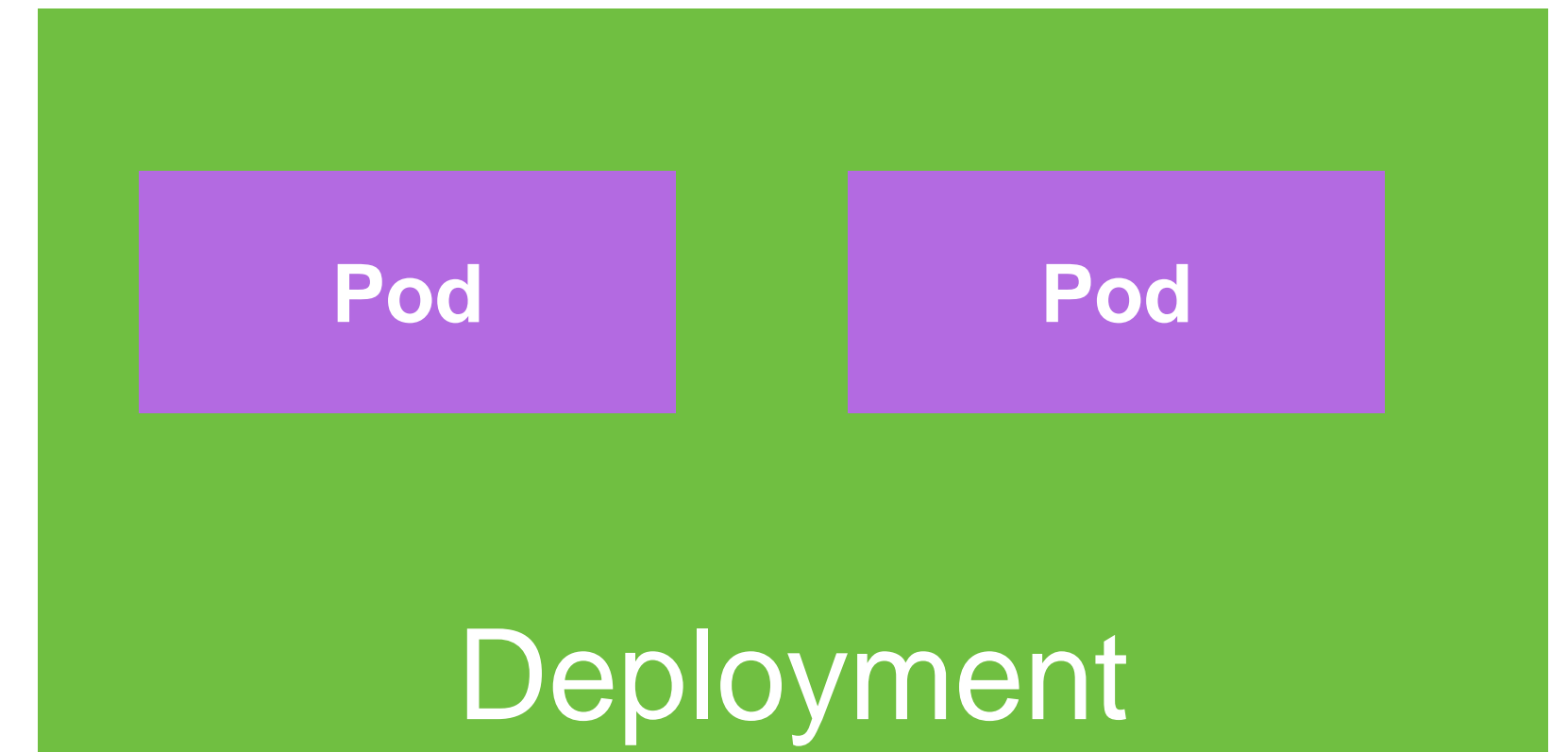
核心概念 - Volume

- 声明在 Pod 中的容器可访问的文件目录
- 可以被挂载在 Pod 中一个（或者多个）容器的指定路径下
- 支持多种后端存储的抽象
 - 本地存储、分布式存储、云存储...



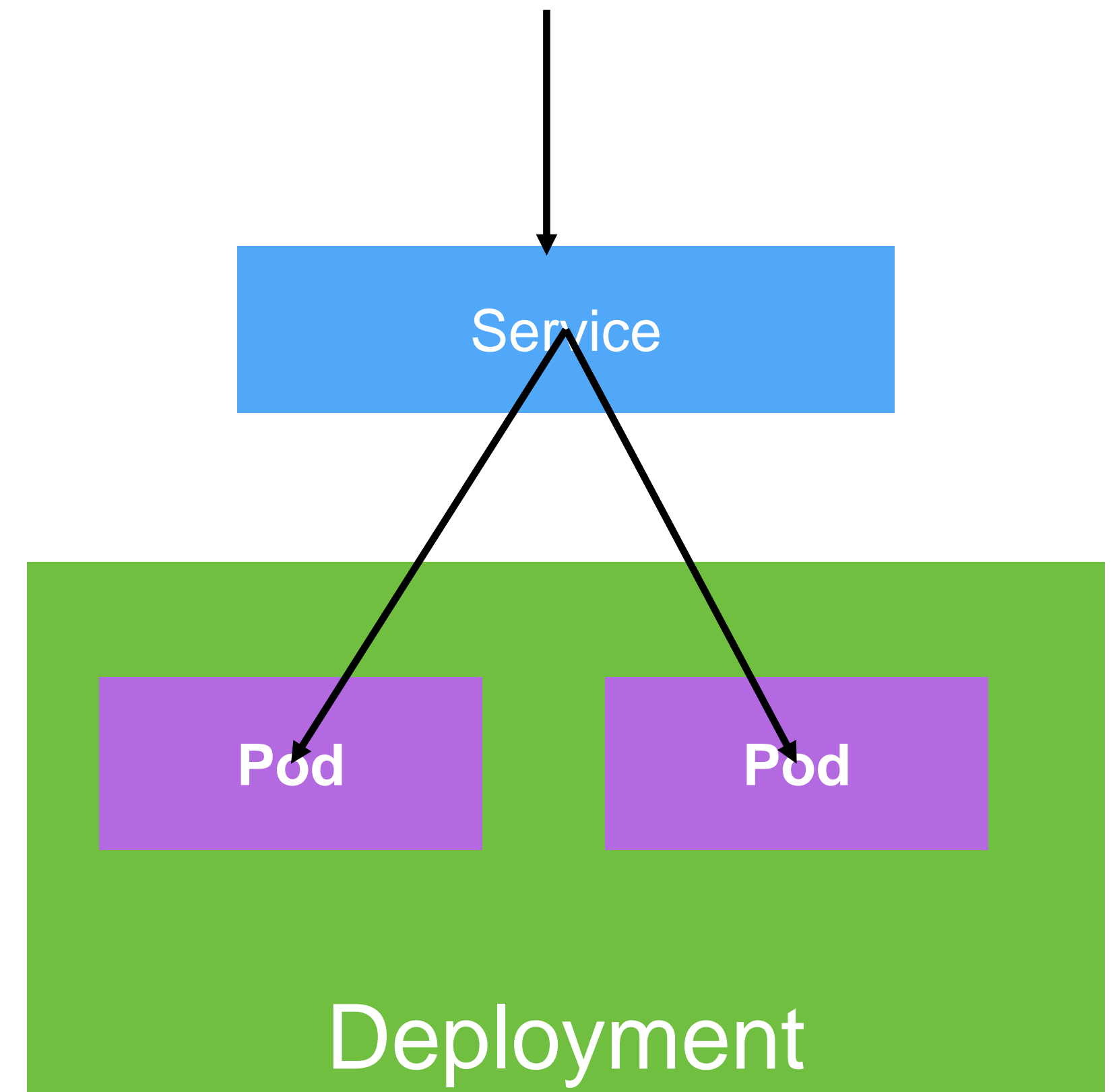
核心概念 - Deployment

- 定义一组 Pod 的副本数目、版本等
- 通过控制器（Controller）维持 Pod 的数目
 - 自动恢复失败的 Pod
- 通过控制器以指定的策略控制版本
 - 滚动升级、重新生成、回滚等



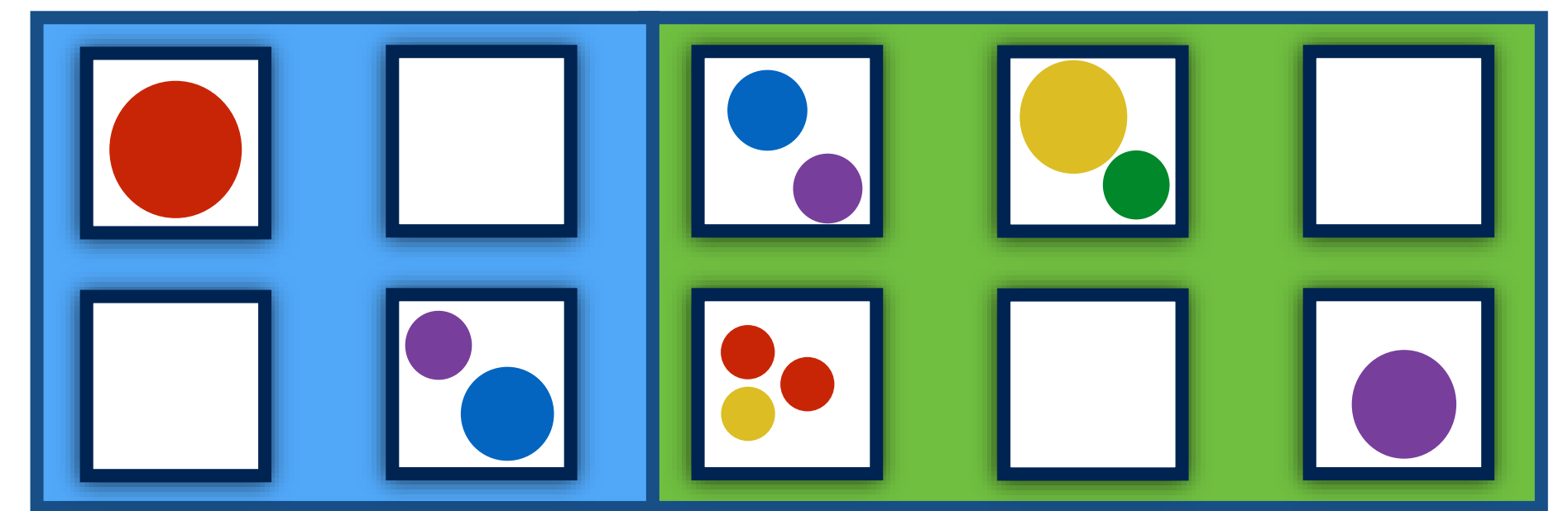
核心概念 - Service

- 提供访问一个或多个 Pod 实例的稳定访问地址
- 支持多种访问方式实现
 - ClusterIP
 - NodePort
 - LoadBalancer



核心概念 - Namespaces

- 一个集群内部的逻辑隔离机制（鉴权、资源额度）
- 每个资源都属于一个 Namespace
- 同一个 Namespace 中的资源命名唯一
- 不同 Namespace 中的资源可重名



Namespace1

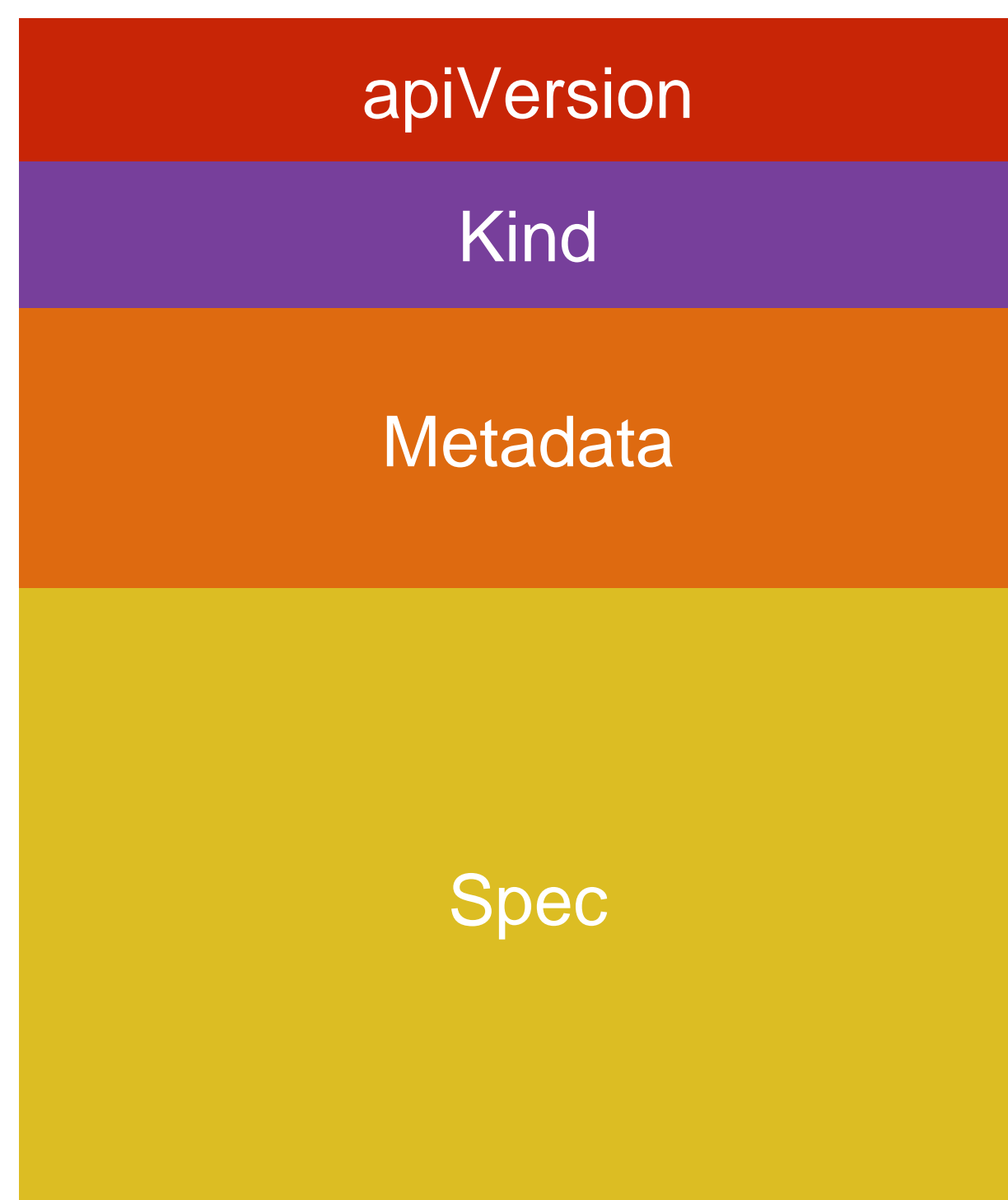
Namespace2

API - 基础知识

HTTP + JSON/YAML

- kubectl
- UI
- curl

/api/v1/namespaces/\$NAMESPACE/pods/\$NAME



```
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    name: nginx
spec:
  containers:
  - name: nginx
    image: nginx
    ports:
    - containerPort: 80
```


API - Label

- 一组 Key:Value
- 可以被 selector 所查询
 - select color=red
- 资源集合的默认表达形式
 - 例如 Service 对应的一组 Pod

```
apiVersion: v1
kind: Pod
metadata:
  name: apple
  labels:
    color: red
spec:
  containers:
  - name: nginx
    image: nginx
    ports:
    - containerPort: 80
```

```
apiVersion: v1
kind: Pod
metadata:
  name: banana
  labels:
    color: yellow
spec:
  containers:
  - name: nginx
    image: nginx
    ports:
    - containerPort: 80
```

```
apiVersion: v1
kind: Pod
metadata:
  name: strawberry
  labels:
    color: red
spec:
  containers:
  - name: nginx
    image: nginx
    ports:
    - containerPort: 80
```

4 尝试一下 Kubernetes 吧!

安装 Minikube

1. 安装 VirtualBox

`https://www.virtualbox.org/wiki/Downloads`

2. 安装 MiniKube (中国版)

`https://yq.aliyun.com/articles/221687`

3. 启动 MiniKube

```
minikube start --vm-driver virtualbox
```

以 macOS 为例子，其它操作系统请访问：

<https://kubernetes.io/docs/tasks/tools/install-minikube/>

尝试一些命令吧！

1. 提交一个 nginx deployment

```
kubectl apply -f https://k8s.io/examples/application/deployment.yaml
```

2. 升级 nginx deployment

```
kubectl apply -f https://k8s.io/examples/application/deployment-update.yaml
```

3. 扩容 nginx deployment

```
kubectl apply -f https://k8s.io/examples/application/deployment-scale.yaml
```

以 macOS 为例子，其它操作系统请访问：

<https://kubernetes.io/docs/tasks/tools/install-minikube/>

谢谢观看

THANK YOU



关注“阿里巴巴云原生”公众号
获取第一手技术资料

