



# Orchestration

# Plan



- I. Orchestration ?
- II. Intérêt
- III. Kubernetes, c'est quoi ?
- IV. Ecosystème
- V. Node principal / de travail
- VI. Fonctionnement
- VII. Minikube & Kubectl
- VIII. Manifest
- IX. Pod & Service
- X. ConfigMap & Secret
- XI. Volume
- XII. TP



# Orchestration ?

# Orchestration ?



Processus automatique d'organisation,  
de coordination et de gestion de systèmes  
informatiques complexes, de middleware et de services



# Intérêt



Framework

Gain de temps

Simplicité de gestion

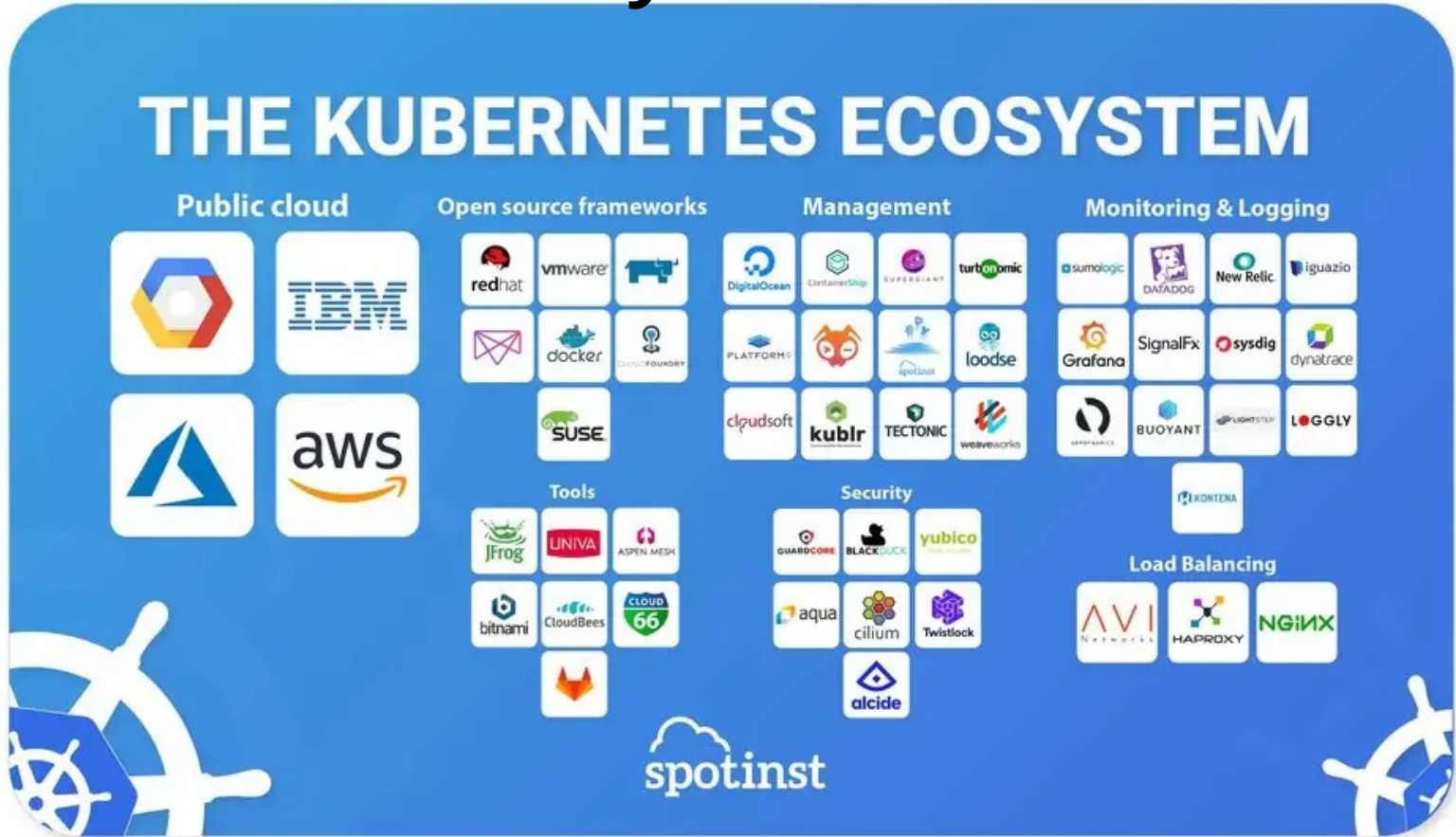
Architecture plus complexe possible

Gestion via manifeste

Automatisation des tâches redondantes

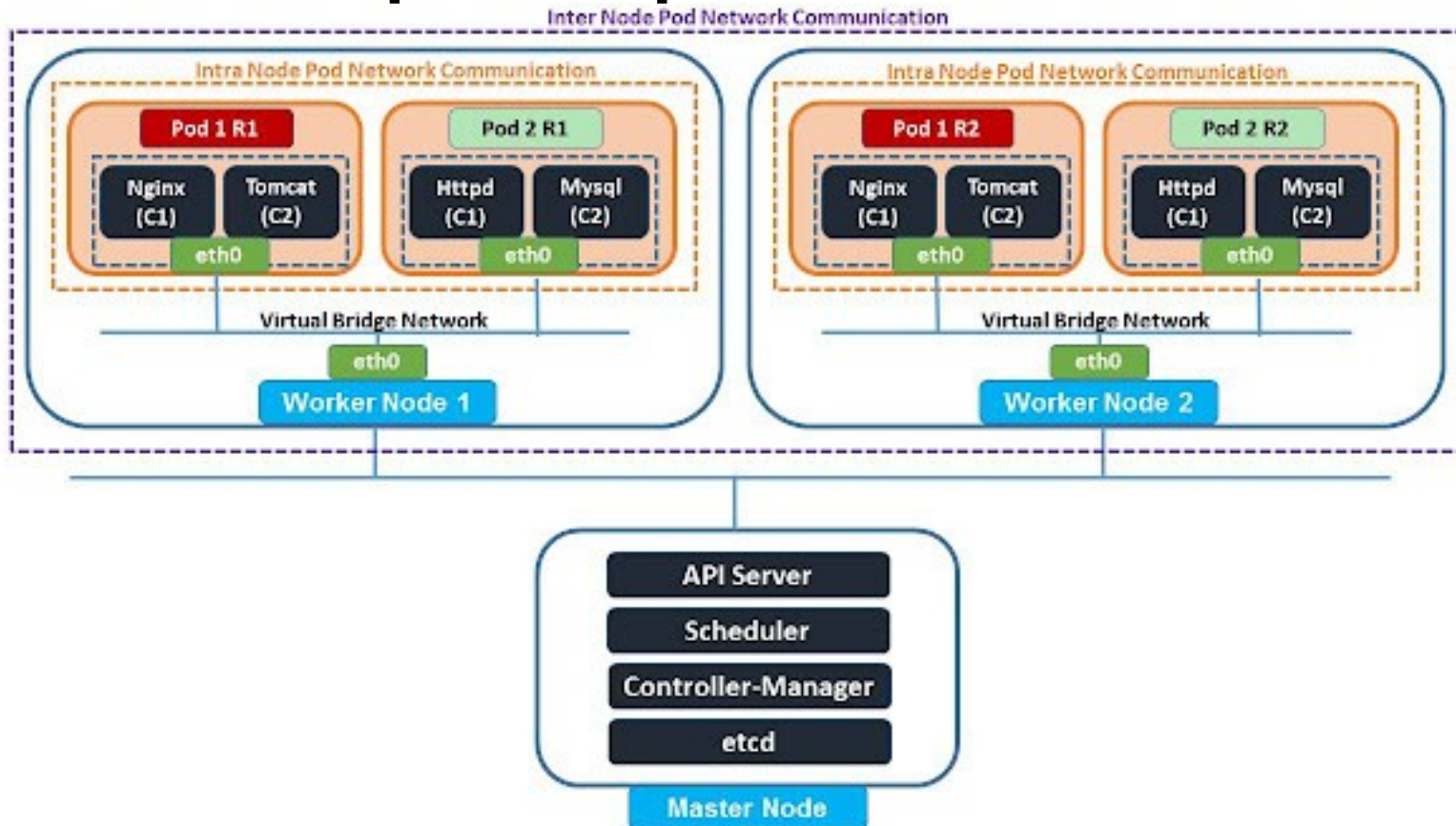


# Kubernetes, c'est quoi ?

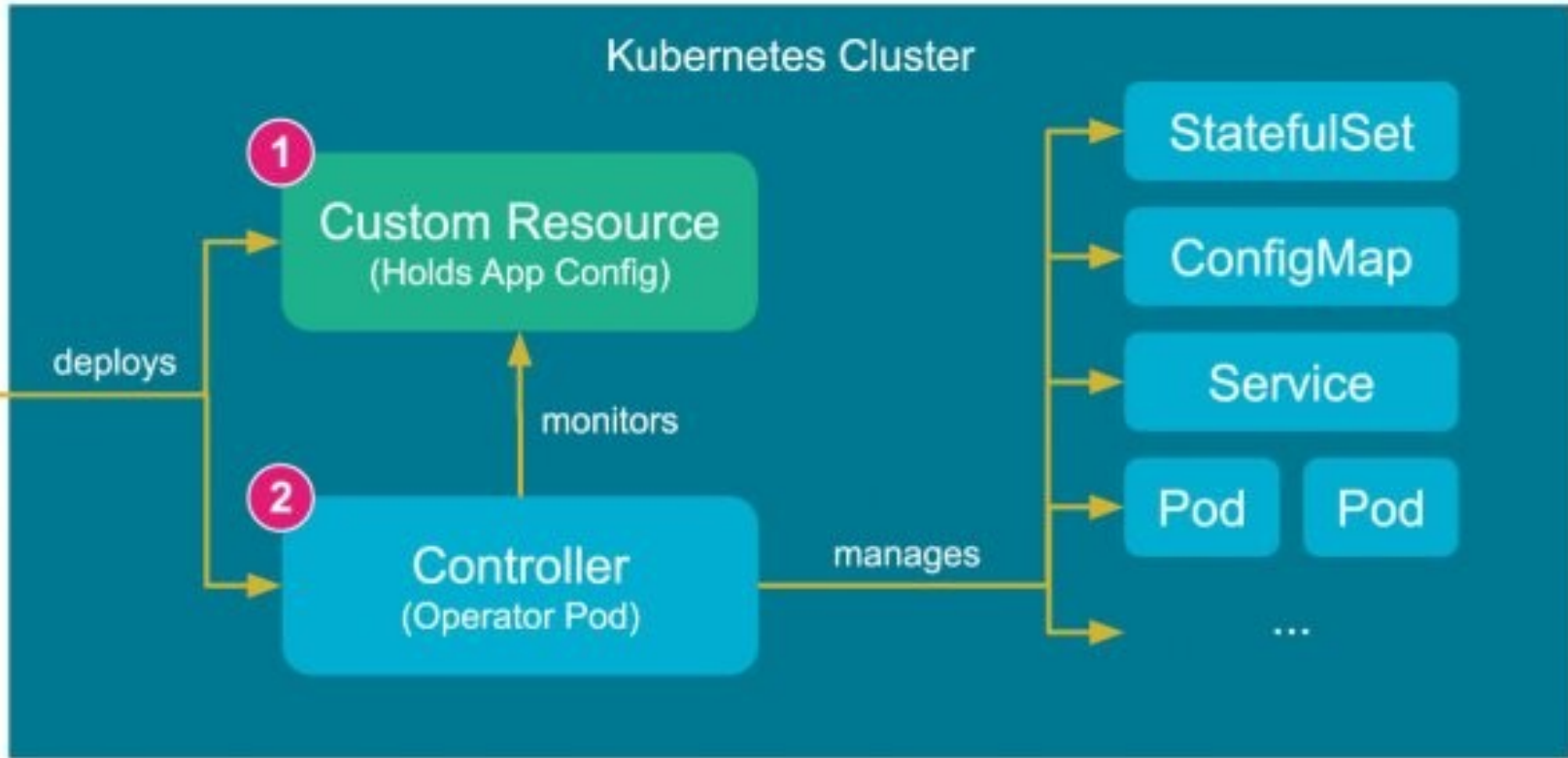




# Node principal / de travail



# Fonctionnement



# Minikube & Kubectl



## **Minikube :**

- permet de lancer des clusters Kubernetes en local
- offre des accès faciliter pour certaines fonctionnalités

## **Kubectl :**

- permet de discuter avec le cluster Kubernetes en cli
- permet d'accéder à n'importe quel cluster

# Manifest



```
kind: Service
apiVersion: v1
metadata:
  name: nginx-service
  namespace: development
  labels:
    app: nginx
  annotations:
    test.io/owner: John Doe
spec:
  type: LoadBalancer
  selector:
    app: nginx
  ports:
    - name: http
      protocol: TCP
      port: 80
```

# Pod & Service

**Pod :**

- regroupe un ou plusieurs conteneurs
- modélise un "hôte logique" spécifique à une application
- durée de vie "aléatoire"

**Conteneurs :**

- une adresse IP
- un espace de ports
- communiquent via localhost
- partagent des volumes
- s'exécutent dans un contexte partagé

**Service :**

- permet de communiquer entre pod
- définit les règles d'exposition d'un ensemble de pod
- dns <service>.default.svc.cluster.local

**Type de service :**

- NodePort
- ClusterIP
- LoadBalancer

# ConfigMap & Secret

**ConfigMap :**

- fichiers de configuration pour les applications

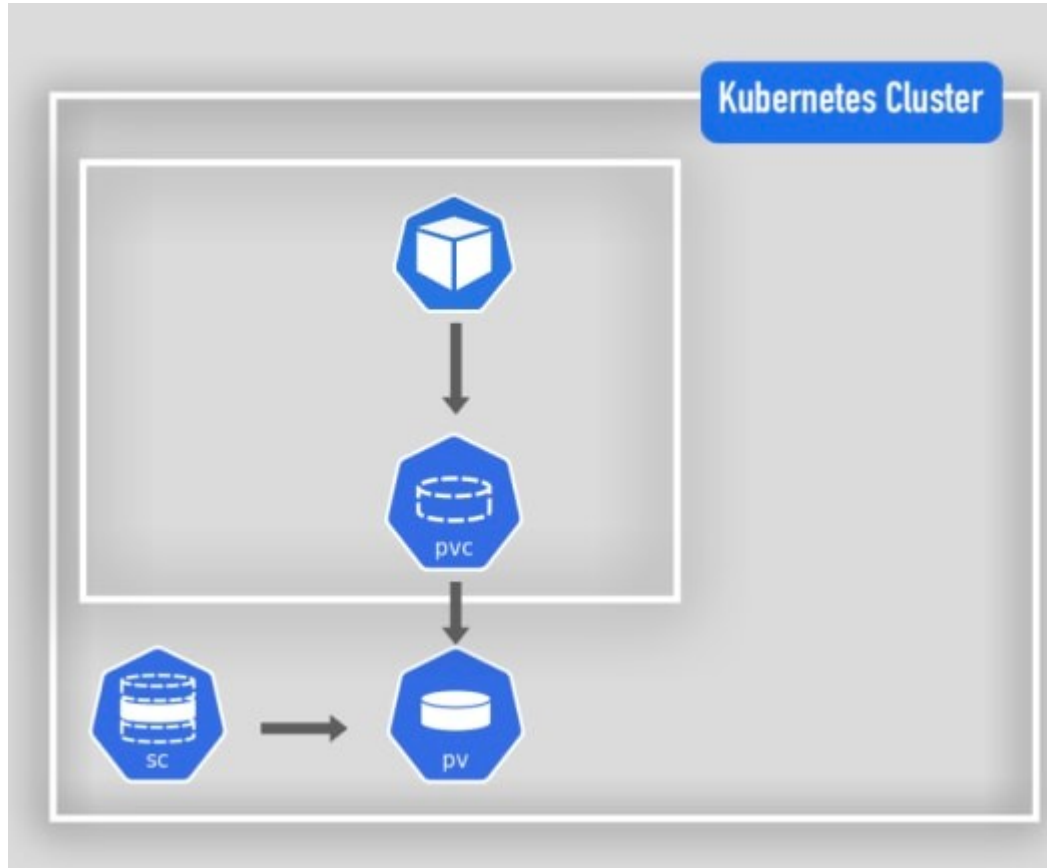
**Secret :**

- permet de stocker des données sensible
- chiffre les données

**Type de secret :**

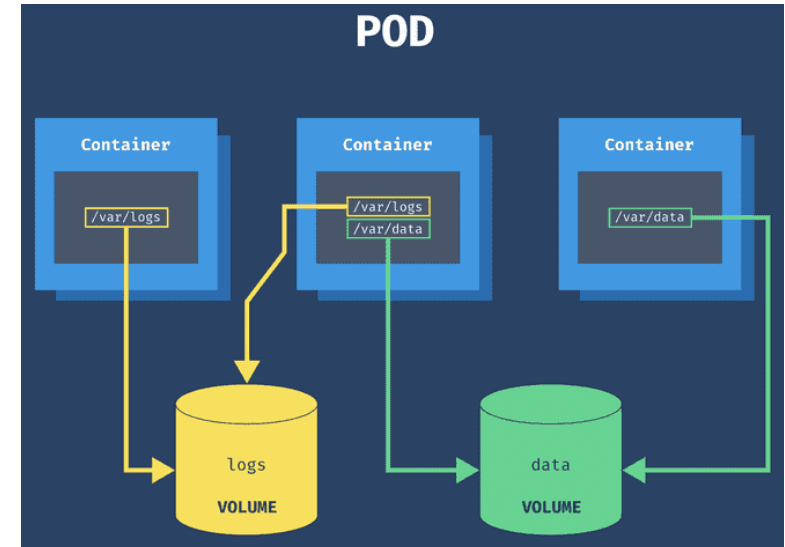
- Opaque
- `kubernetes.io/service-account-token`
- `kubernetes.io/dockercfg`
- `kubernetes.io/basic-auth`
- `kubernetes.io/ssh-auth`
- `kubernetes.io/tls`
- `bootstrap.kubernetes.io/token`

# Volume



## Type de volume :

- configMap
- emptyDir
- persistentVolumeClaim
- secret





# TP





- Monter un cluster Kubernetes
- Déployer les manifestes pod + service
- UI vs cli



- Créer un système de load balancer, est ce possible ?
- Comment augmenter le nb de pod ?



En utilisant le load balancer précédemment mis en place :

- Configurer le pod pour que le fichier html soit placé dans le volume
- Monter un autre pod pour écrire dans ce meme dossier



**Question :**

- Comment écouter les métriques ?
- Comment gérer les logs ?



- I. Orchestration ?
- II. Intérêt
- III. Kubernetes, c'est quoi ?
- IV. Ecosystème
- V. Node principal / de travail
- VI. Fonctionnement
- VII. Minikube & Kubectl
- VIII. Manifest
- IX. Pod & Service
- X. ConfigMap & Secret
- XI. Volume
- XII. TP

# Prochaine étape



Déployer son application