

Introduction à la conteneurisation avec Docker et Kubernetes



Sébastien Laporte



@seb__laporte



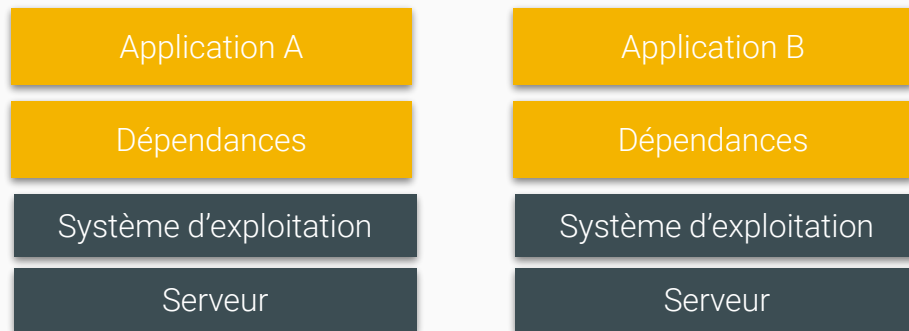
Qu'est ce que la conteneurisation ?

Evolution vers la conteneurisation

1 application / 1 serveur

1 application / 1 VM

Conteneurisation



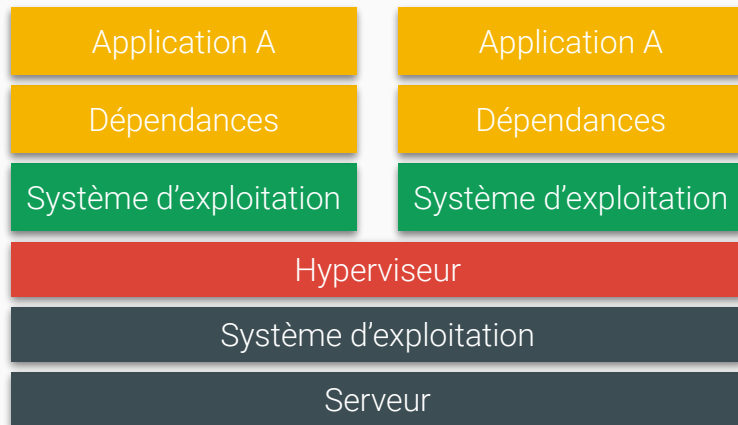
- Serveur dédié à 1 seule application
- Difficile à scaler
- Difficile à faire évoluer
- Forte adhérence à l'infrastructure

Evolution vers la conteneurisation

1 application / 1 serveur

1 application / 1 VM

Conteneurisation



Avantages

- Meilleure utilisation des ressources physiques
- Plus facile à scaler
- Ouverture vers le Cloud

Limitations / inconvénients

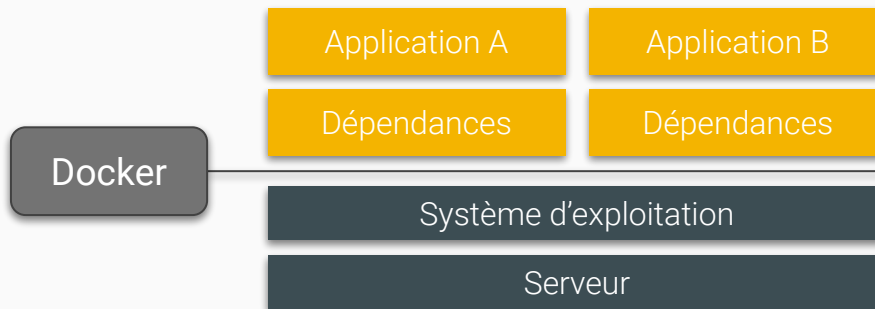
- Chaque VM supplémentaire consomme des ressources physiques avec son OS
- Portabilité de l'application non garantie

Evolution vers la conteneurisation

1 application / 1 serveur

1 application / 1 VM

Conteneurisation



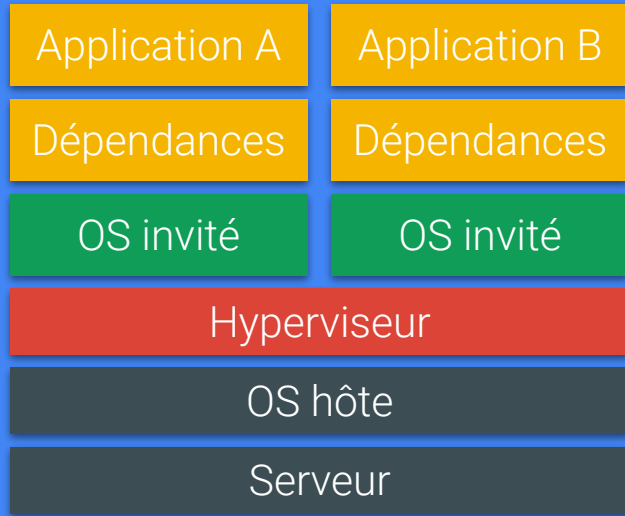
Avantages

- Augmentation de la densité de déploiement
- Démarrage accéléré du service
- Standardisation du packaging de l'application

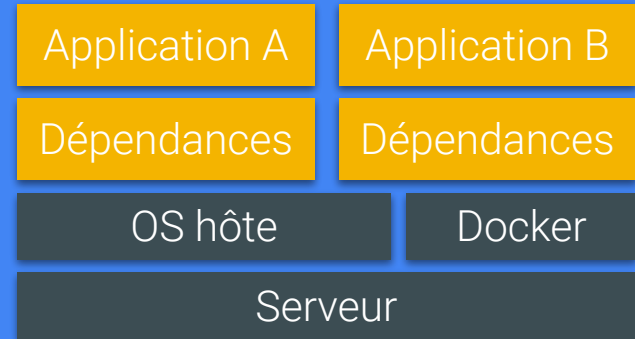
Bénéfices

- Portabilité de l'application
- Déploiement simplifié
- Permet d'augmenter la fréquence de livraison

Virtualisation



Conteneurisation



Virtualisation

- S'appuie sur un **hyperviseur** pour l'accès aux ressources physiques
- Chaque machine virtuelle contient un **système d'exploitation complet**

Avantage

- Très bonne **isolation**

Inconvénients

- Boot d'un système complet au démarrage
- Volumineux
- Potentiel coût de licence

Conteneurisation

- Partage du noyau du système hôte
- Utilise les **cgroups** pour isoler et limiter l'accès aux ressources physiques
- Utilise les **namespaces** pour isoler les conteneurs entre eux

Avantages

- Démarrage immédiat
- Portabilité
- Augmente la densité de déploiement

Inconvénient

- Isolation dépendante du système hôte

Pourquoi la conteneurisation ?



Cas d'usages de la conteneurisation

- Pour le développeur
 - Création de l'environnement simplifié
 - Environnement reproductible
 - Mise en oeuvre d'une stack complexe rapidement et simplement
 - Evite le "ça marche sur mon poste"
- Pour l'opérationnel
 - Montée de version et retour arrière simplifié
 - Montée en charge simplifiée
 - Orienté micro-service
 - Déploiement simplifié

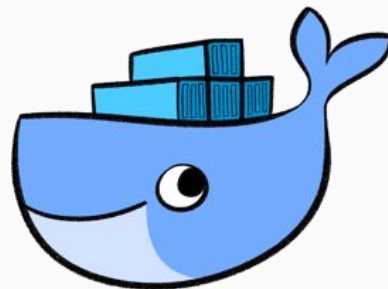
Que faut il savoir pour utiliser Docker ?

Le Dockerfile : la recette

- Fichier texte
- Semblable à un script d'installation

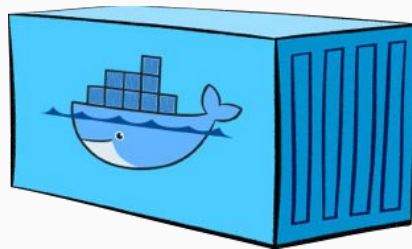
L'image Docker : le packaging unique

- Construite à partir du Dockerfile
- N'embarque pas un système d'exploitation complet
- Contient l'application
- Construit couche par couche (layers)
- Portable
- Utilise un mécanisme de gestion de version

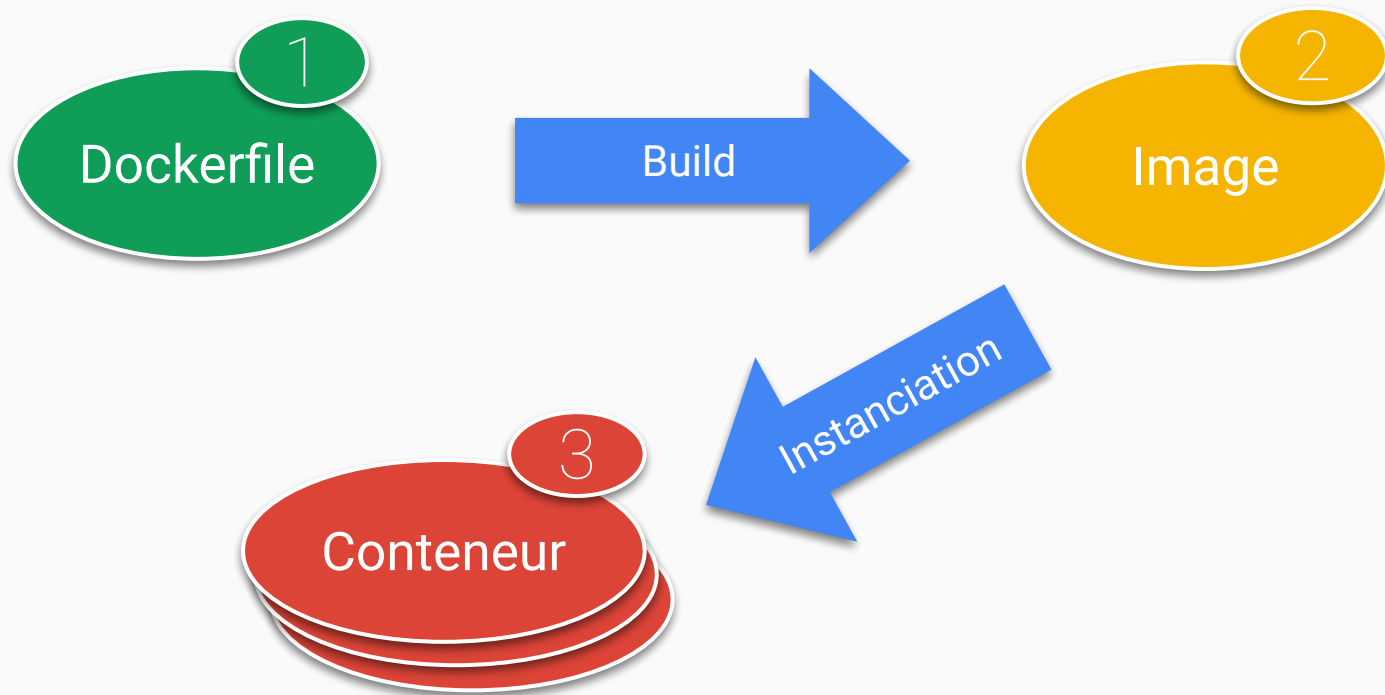


Le conteneur : l'unité de déploiement

- Fait tourner l'application
- Immuable
- Pas de stockage persistant



Création d'un conteneur



Comment automatiser un déploiement
avec Docker ?

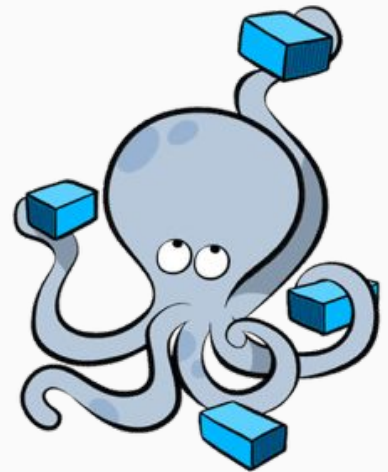


Docker Compose

- Permet d'automatiser la création de conteneurs
- Permet de construire des applications multi-conteneur simplement
- S'appuie sur un fichier texte qui décrit le déploiement
- Permet de gérer une architecture complexe comme une seule entité

Limitations

- Se limite à l'automatisation du déploiement
- N'est pas orienté pour une infrastructure distribuée
- Pas de gestion des droits



Quelle solution pour palier à ces limitations ?

Kubernetes

Qu'est ce que Kubernetes ?

Qu'est ce que ça fait ?

Quelle valeur ajoutée ?

Points de vigilance ?



- Une plate-forme de conteneurs
- Projet Open Source initié par Google
- Gère une flotte de serveurs
- Permet de déclarer l'état attendu d'une application

Kubernetes

Qu'est ce que Kubernetes ?

Qu'est ce que ça fait ?

Quelle valeur ajoutée ?

Points de vigilance ?

- Met à disposition les ressources d'une flotte de serveurs
- S'assure de la bonne santé des applications déployées
- Permet de s'abstraire des spécificités d'une infrastructure

Kubernetes

Qu'est ce que Kubernetes ?

Qu'est ce que ça fait ?

Quelle valeur ajoutée ?

Points de vigilance ?

- Gain de temps pour les opérationnels
 - Surveillance de l'état de santé des applications
 - S'adapte aux changements de l'infrastructure
- Contribue à réduire les coûts d'exploitation
 - Optimisation des ressources des serveurs
 - Ajustement des applications en fonction de la charge

Kubernetes

Qu'est ce que Kubernetes ?

Qu'est ce que ça fait ?

Quelle valeur ajoutée ?

Points de vigilance ?

- Installation / configuration / administration complexe
- Privilégier une solution commerciale on-premise ou Cloud
- Accompagnement recommandé des équipes

Docker en pratique

Le Dockerfile

Exemples

Les instructions

Bonnes pratiques

Application Java Spring Boot

```
FROM openjdk:8-jdk-slim
COPY target/*.jar /opt/app.jar
WORKDIR /opt
EXPOSE 8080
CMD [ "java", "-jar", "app.jar" ]
```

Commande de démarrage

Exposition du port TCP 8080

Position du répertoire de travail

Application Spring Boot

OpenJDK 8

Le Dockerfile

Exemples

Les instructions

Bonnes pratiques

Site web

```
FROM alpine:3.9
RUN apk --update add nginx
RUN mkdir /run/nginx && touch /run/nginx/nginx.pid
COPY nginx.conf /etc/nginx/nginx.conf
ADD www /www
CMD ["nginx"]
```


Le Dockerfile

Exemples

Les instructions

Bonnes pratiques

FROM	Défini l'image de base à utiliser
RUN	Exécute des commandes bash / sh
CMD	Défini la commande à exécuter au démarrage
ADD / COPY	Copie des fichier dans l'image
EXPOSE	Informe d'un port réseau exposés par le conteneur
ENV	Permet de définir des variables d'environnement
VOLUME	Informe d'un répertoire persistant du conteneur

Le Dockerfile

Exemples

Les instructions

Bonnes pratiques

- Privilégier des images de base minimalistes (Alpine)
- 1 seul processus par conteneur
- Démarrer l'application / processus au premier plan
- Logs de l'application sur la sortie standard
- Installer uniquement ce qui est nécessaire
- Limiter le nombre de **layers**
- Organiser les **layers** en fonction du **build**
- Utiliser si besoin le **build multi-stage**
- Éviter l'usage de l'utilisateur **root**

Utilisation de Docker CLI

Gestion des images

Gestion des conteneurs

Gestion des réseaux

Gestion des volumes

docker images	Liste les images
docker build	Crée une image à partir d'un Dockerfile
docker pull	Télécharge une image depuis le Docker hub
docker commit	Crée une image à partir d'un conteneur
docker save	Permet de sauvegarder une image
docker load	Permet de charger une image depuis une archive

Utilisation de Docker CLI

Gestion des images

Gestion des conteneurs

Gestion des réseaux

Gestion des volumes

<code>docker create</code>	Crée un conteneur sans le démarrer
<code>docker run</code>	Crée et démarre le conteneur
<code>docker rm</code>	Supprime un conteneur
<code>docker stop</code>	Arrête un conteneur
<code>docker start</code>	Démarre un conteneur
<code>docker restart</code>	Redémarre un conteneur
<code>docker ps</code>	Liste les conteneur actifs
<code>docker logs</code>	Retourne les logs d'un conteneur
<code>docker inspect</code>	Affiche les propriétés d'un conteneur
<code>docker exec</code>	Exécute une commande dans un conteneur

Utilisation de Docker CLI

Gestion des images

Gestion des conteneurs

Gestion des réseaux

Gestion des volumes

<code>docker network create</code>	Crée un nouveau réseau
<code>docker network rm</code>	Supprime un réseau
<code>docker network ls</code>	Liste les réseaux
<code>docker network inspect</code>	Affiche les informations sur un réseau

Utilisation de Docker CLI

Gestion des images

Gestion des conteneurs

Gestion des réseaux

Gestion des volumes

<code>docker volume create</code>	Crée un nouveau volume
<code>docker volume rm</code>	Supprime un volume
<code>docker volume ls</code>	Liste les volumes
<code>docker volume inspect</code>	Affiche les informations sur un volume

Démo

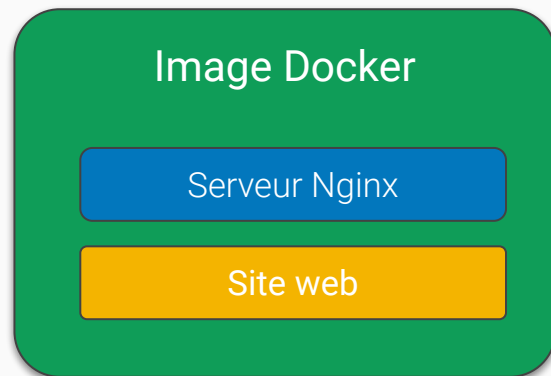
Docker et Docker-Compose

Déploiement d'un site web

Site web statique HTML / CSS

Image Docker : alpine 3.9

Serveur web : Nginx



Kubernetes en pratique

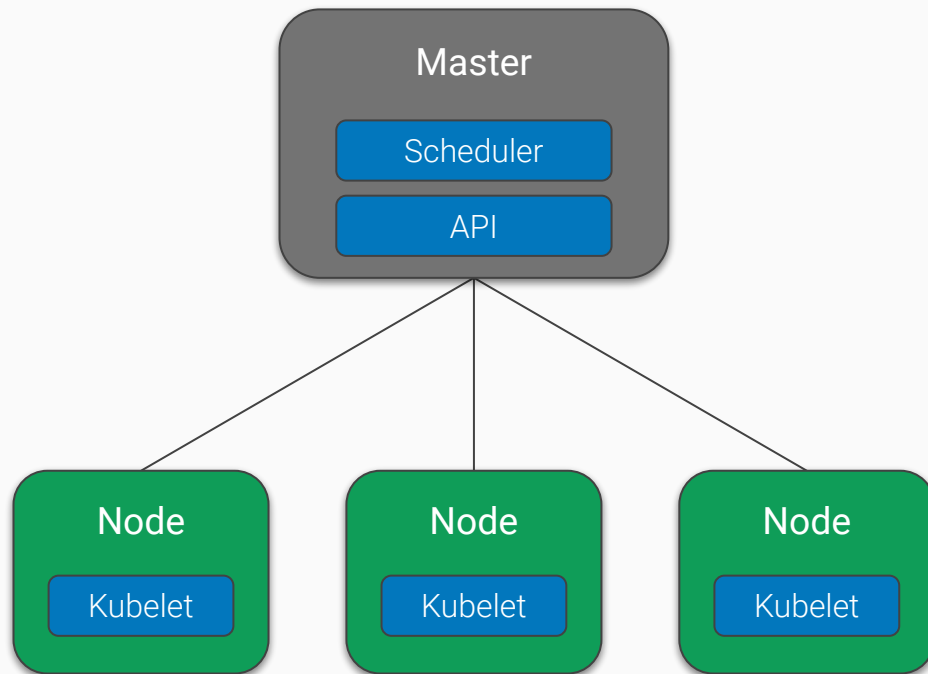
Les ressources Kubernetes

Node

Ingress controller

Persistent volume

Met à disposition des ressources en CPU et mémoire

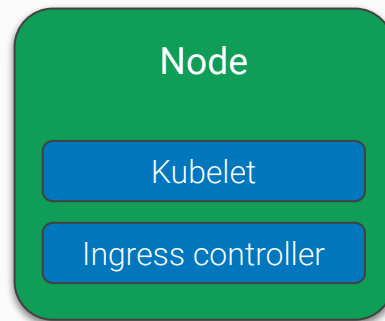


Les ressources Kubernetes

Node

Ingress controller

Persistent volume



- Met à disposition un service de reverse-proxy
- Nombreuse implémentations
 - Nginx
 - HAproxy
 - Istio
 - Traefik
- Possibilité d'utiliser plusieurs implémentations (classes)

Les ressources Kubernetes

Node

Ingress controller

Persistent volume

- Met à disposition du stockage persistant
- Nombreuse implémentations
 - AWS Elastic Block Store
 - Azure File / Azure Disk
 - Vsphere Volume
 - Ceph FS
 - iSCSI
 - NFS
- Possibilité d'utiliser plusieurs implémentations (classes)

Les objets Kubernetes

Pod

Replica set

Deployment

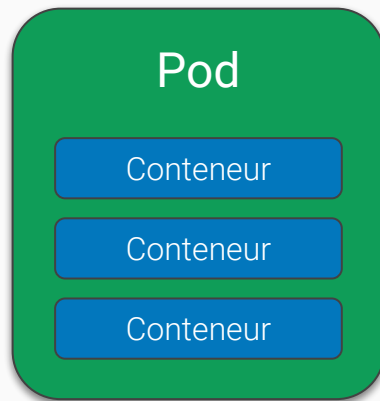
Service

Secret

Config map

Ingress

Persistent volume claim



nginx

application

redis / cache

- Unité de déploiement
- Contient un groupe de conteneurs
- Partagent le stockage
- Partagent la même interface réseau (même IP)
- Ordonnés ensembles (co-scheduled)
- Déployés sur le même node
- Ephémère

Les objets Kubernetes

Pod

Replica set

Deployment

Service

Secret

Config map

Ingress

Persistent volume claim

Replica Set

Replicas : nombre de pods

Template : définition du pod

- Défini le nombre de **pods** à déployer
- S'assure du nombre de réplicas

Les objets Kubernetes

Pod

Replica set

Deployment

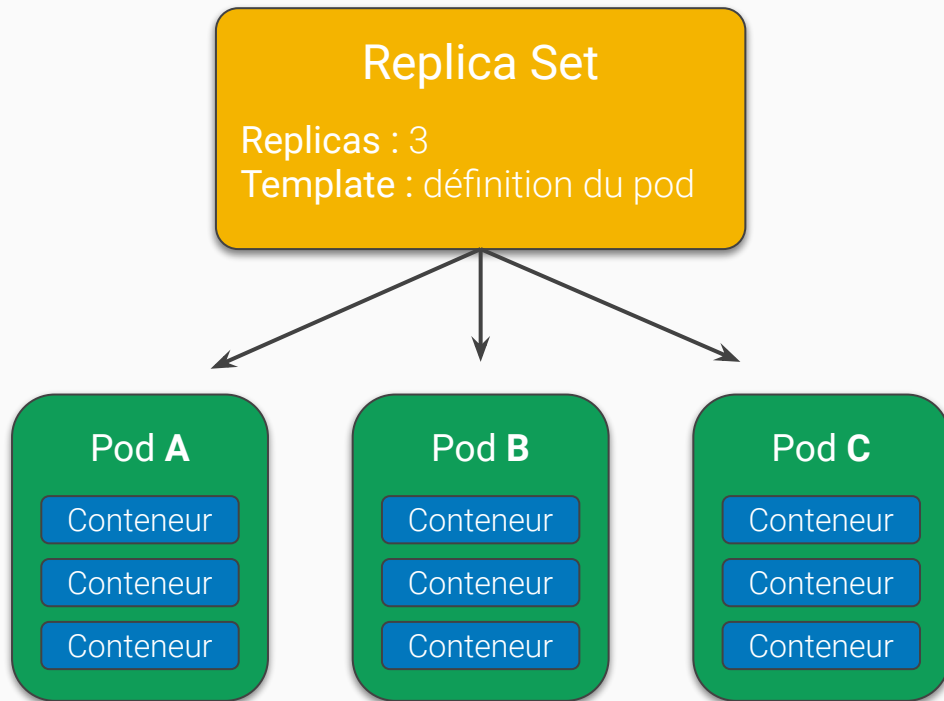
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

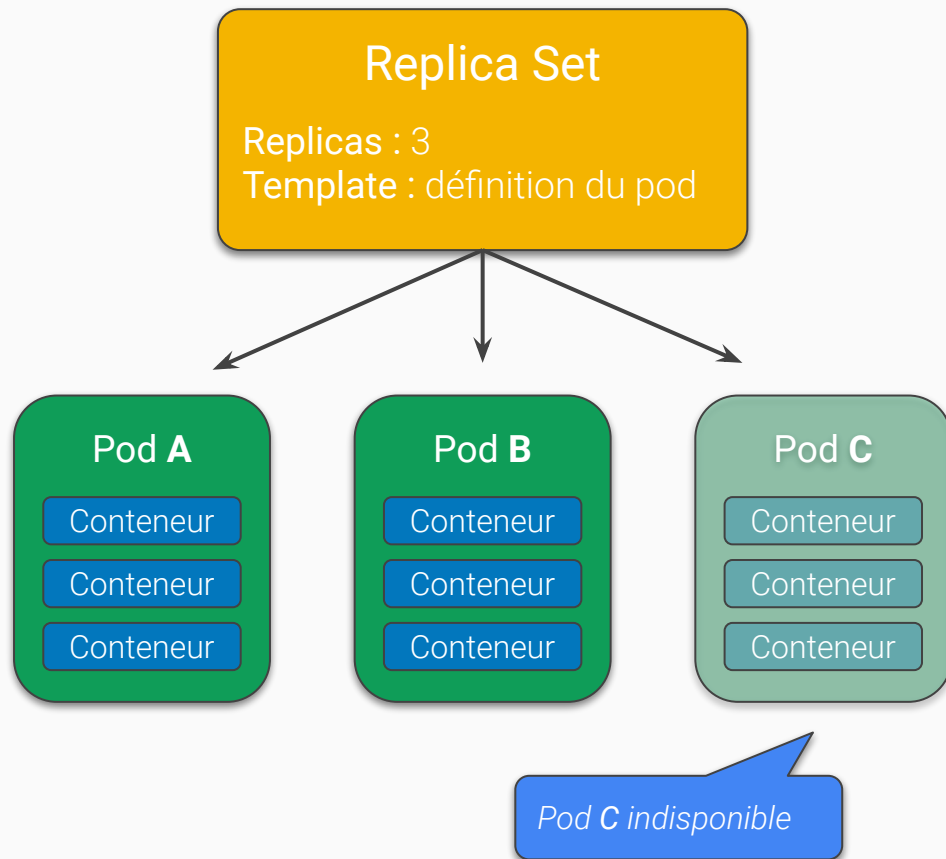
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

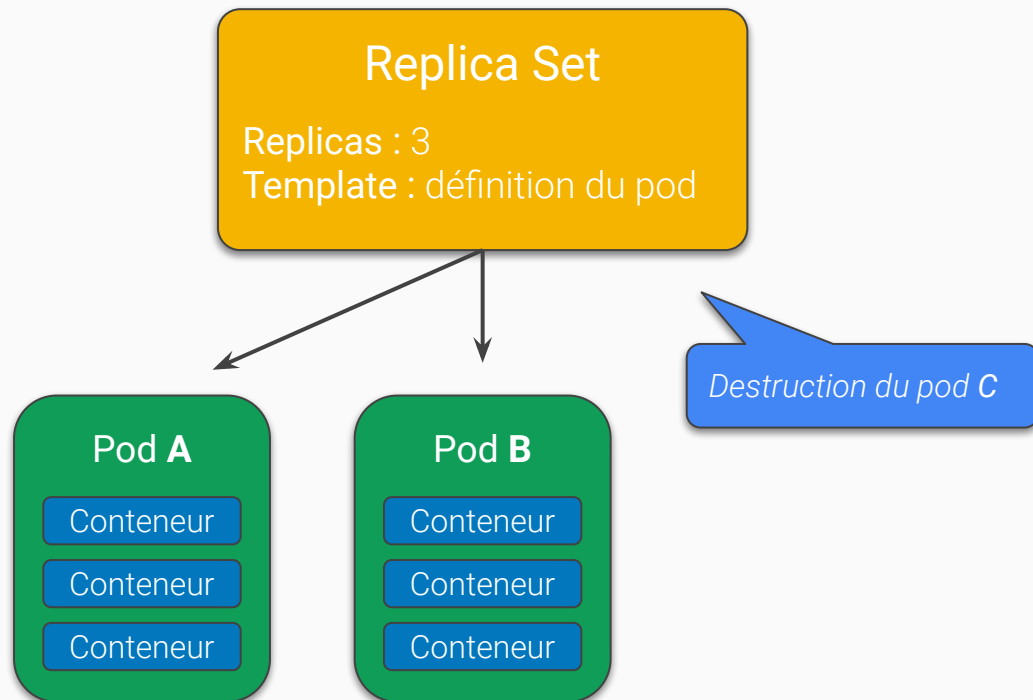
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

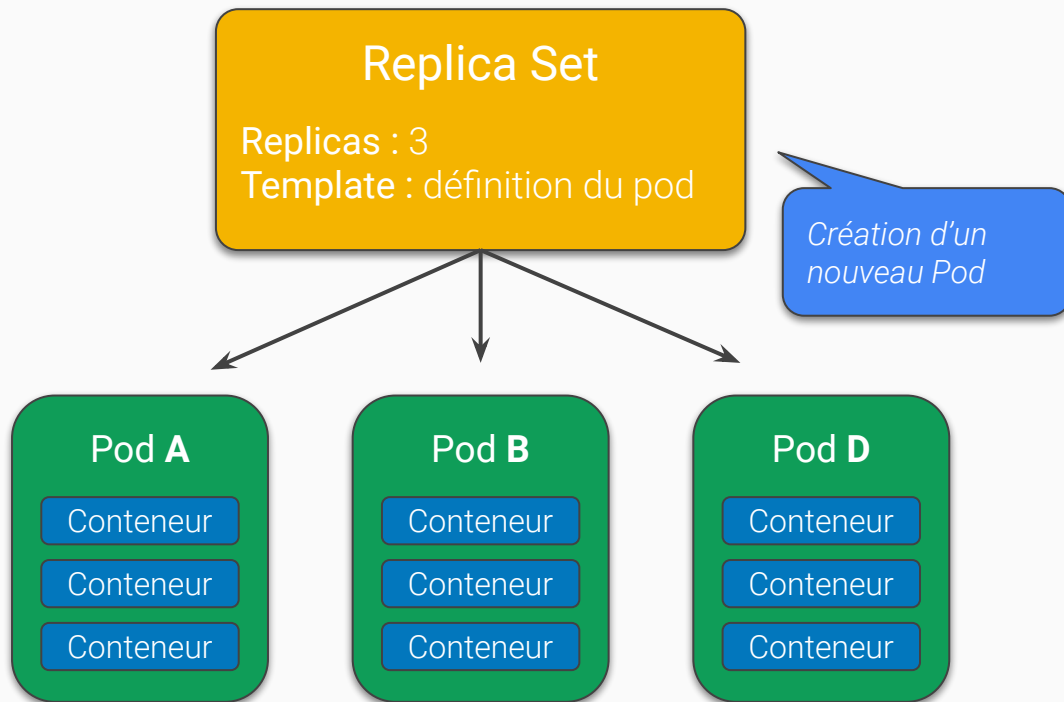
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

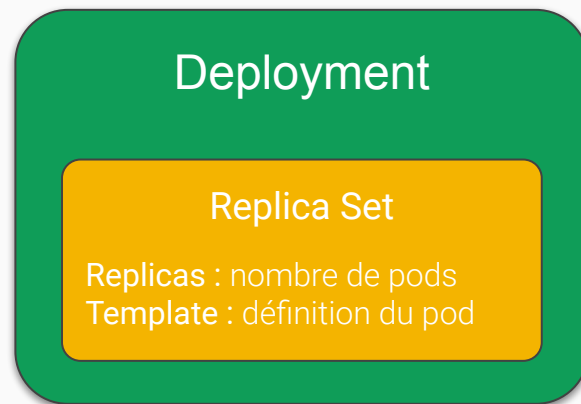
Service

Secret

Config map

Ingress

Persistent volume claim



- Gère les transitions du **Replica Set**
- Garde l'historique des modifications
- Gère les modifications du **template**
- Permet le retour arrière

Les objets Kubernetes

Pod

Replica set

Deployment

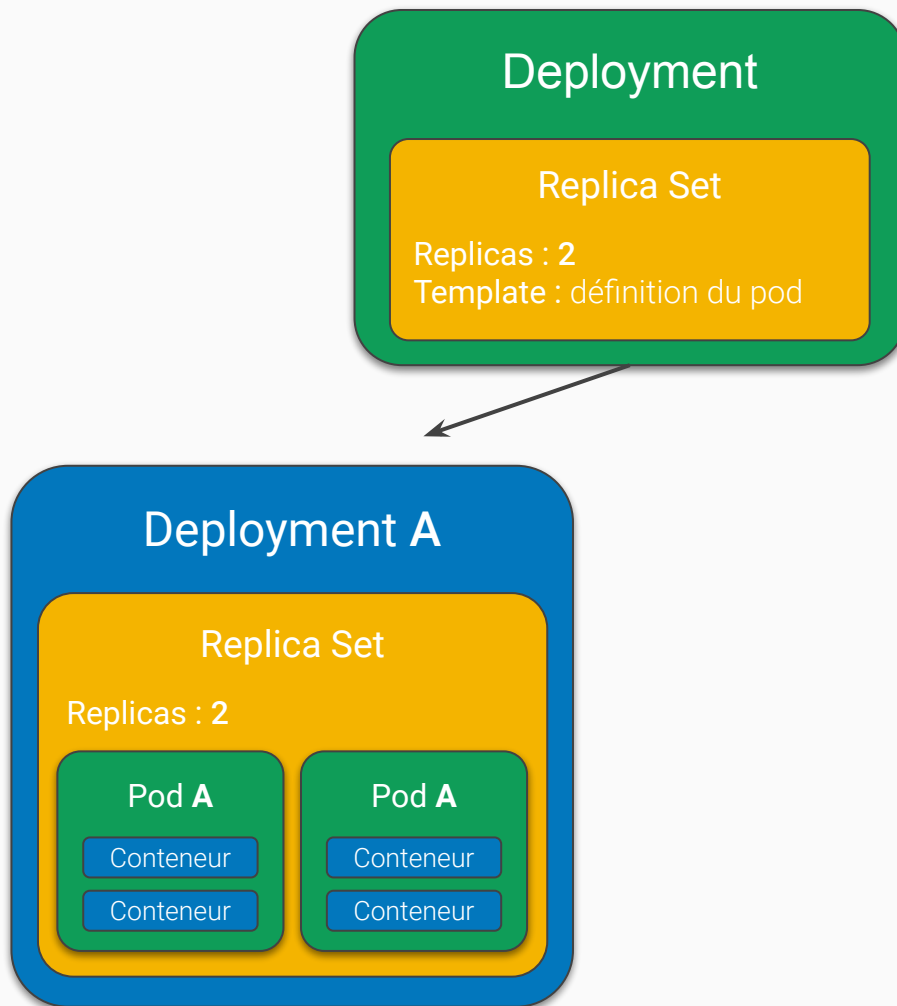
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

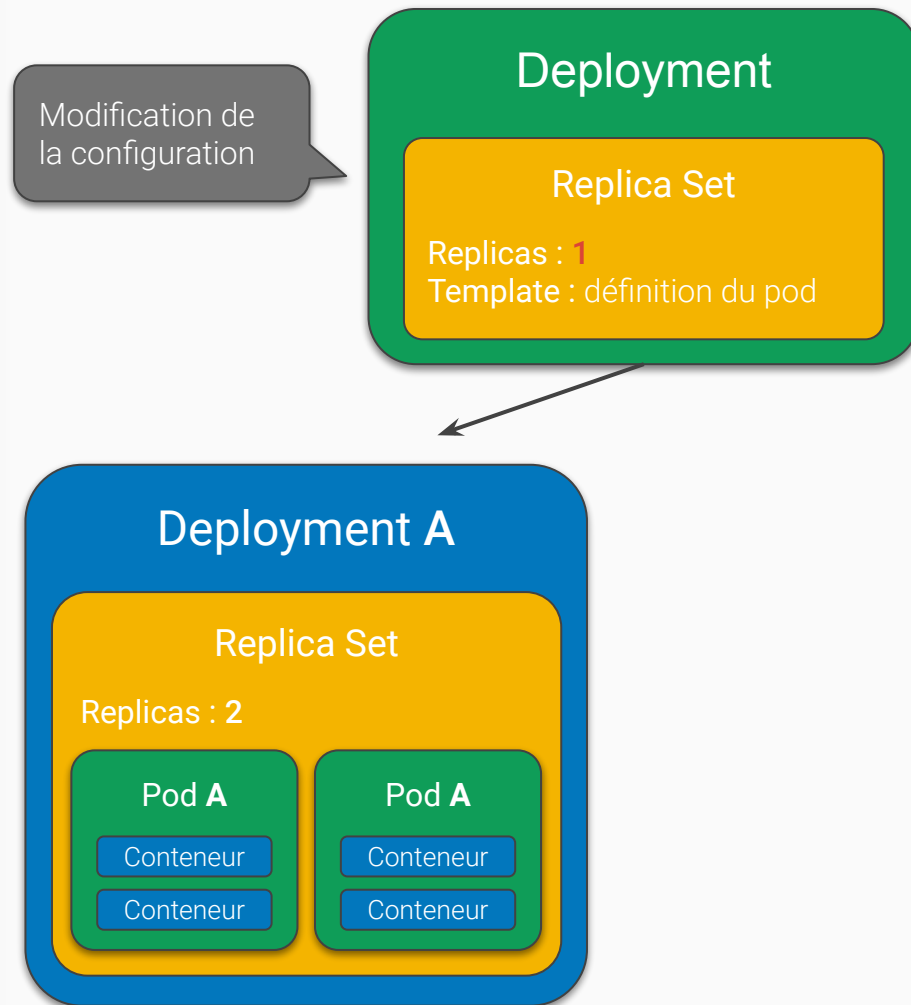
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

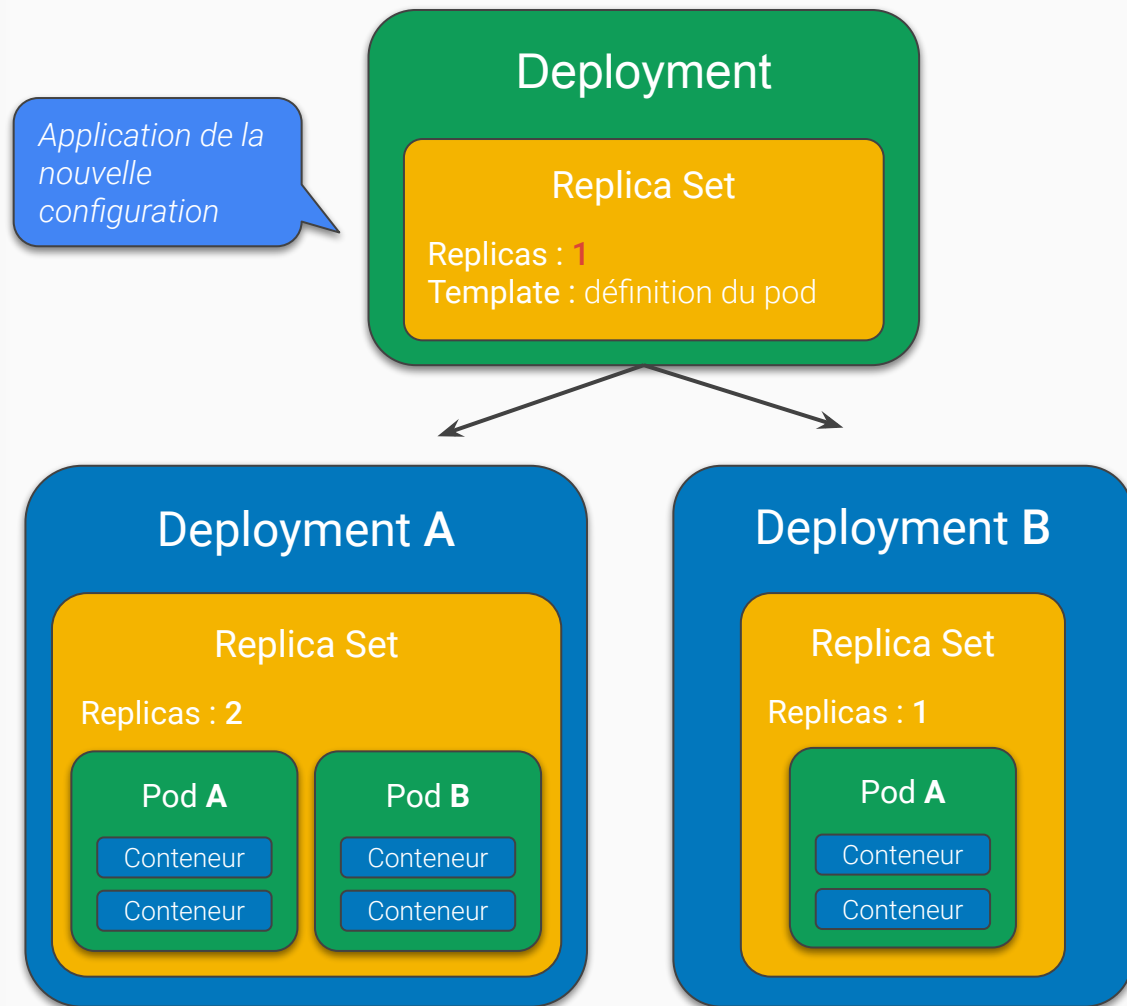
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

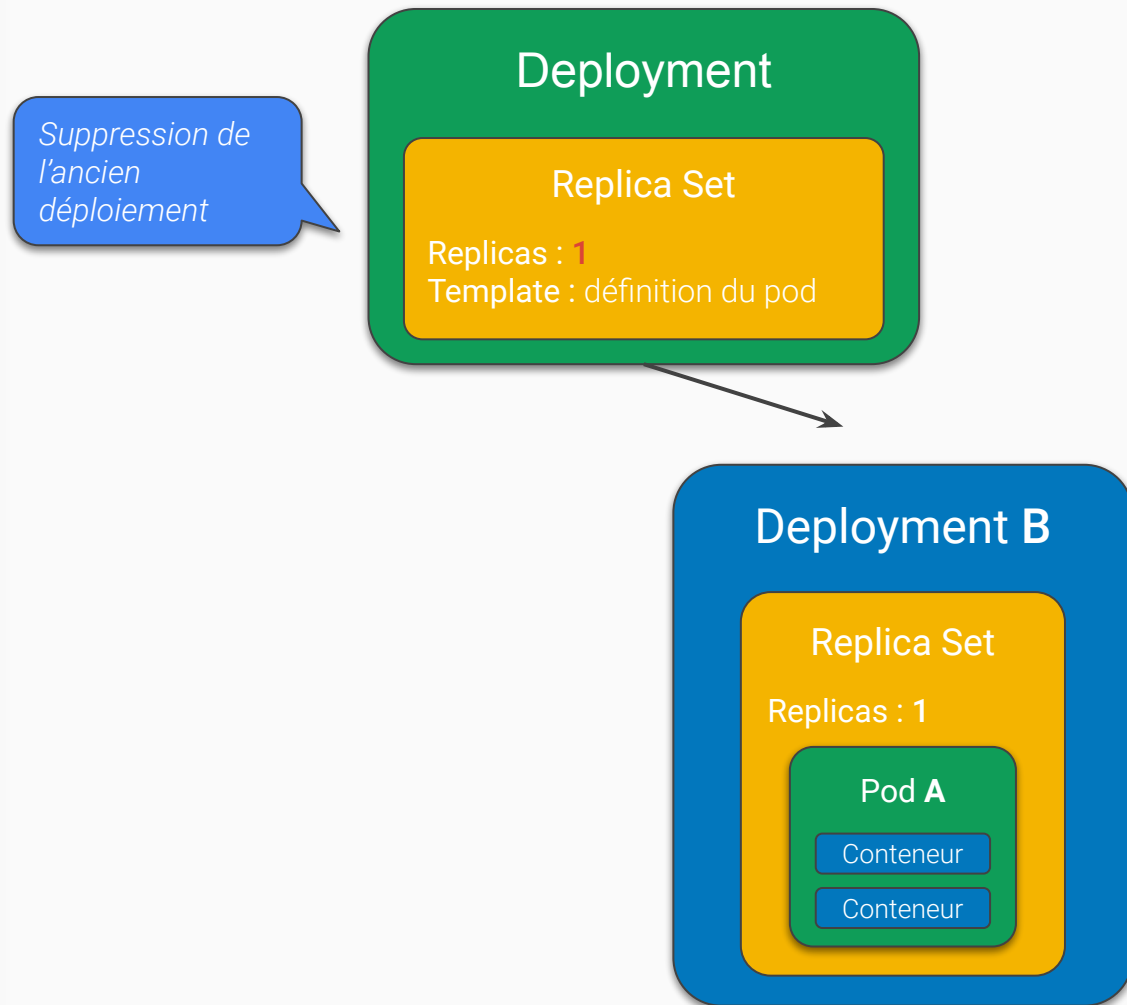
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

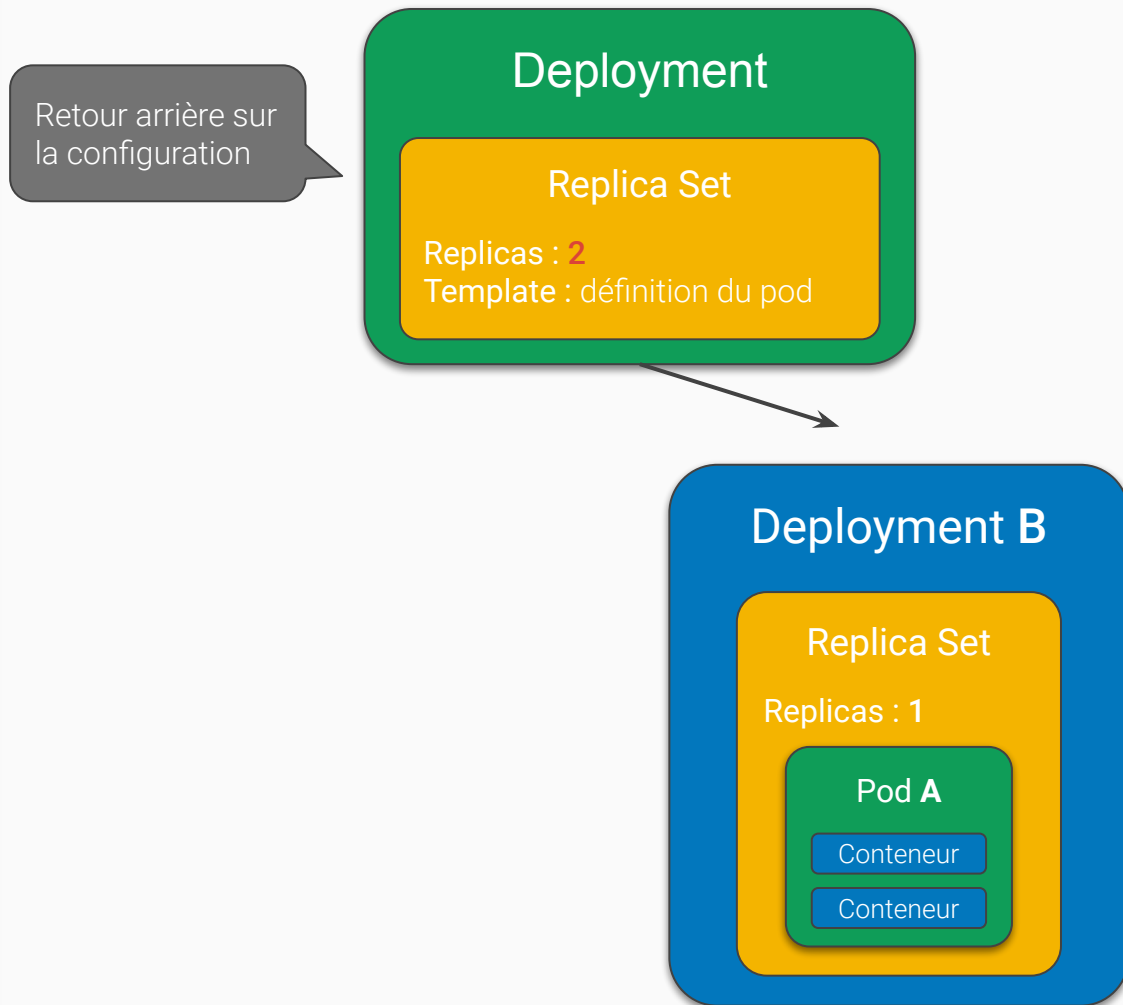
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

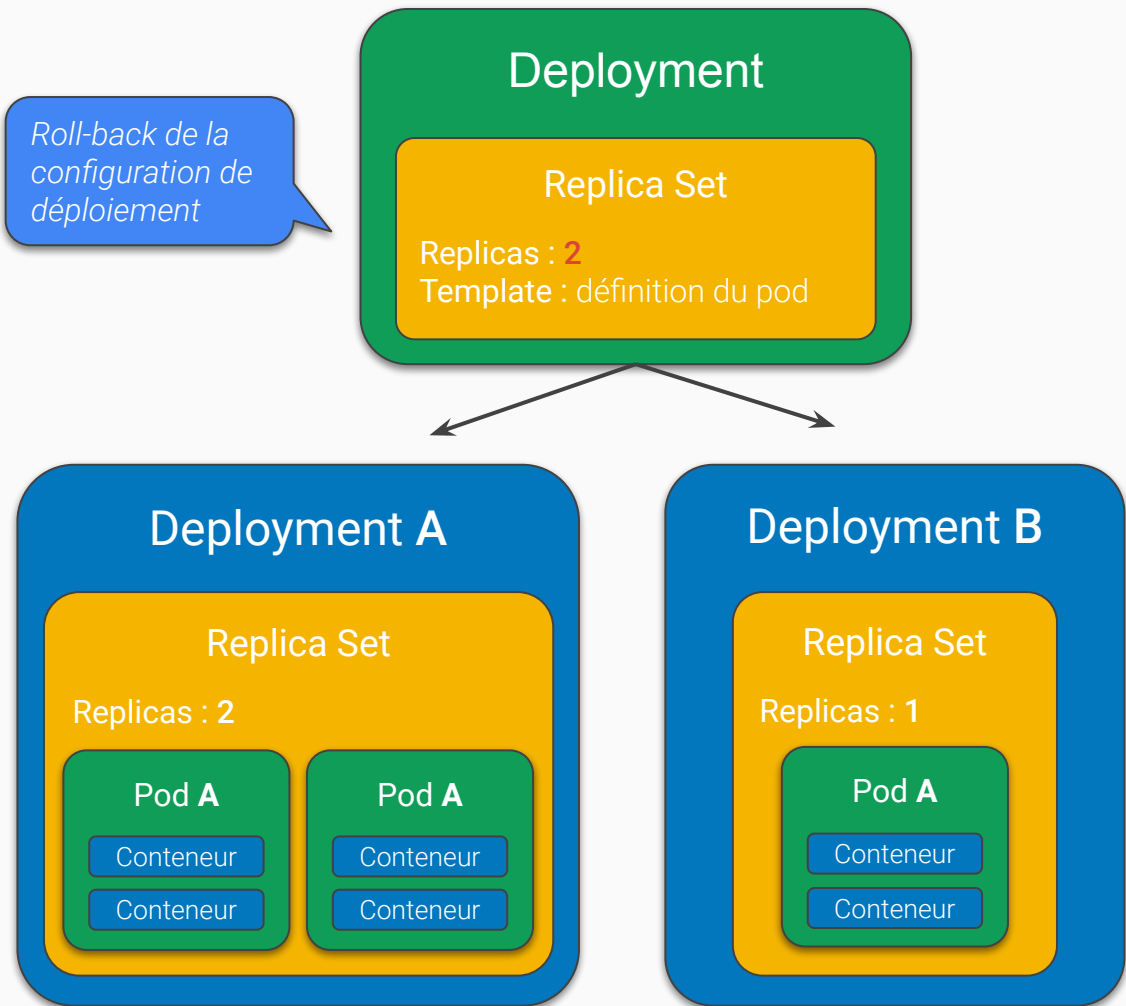
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

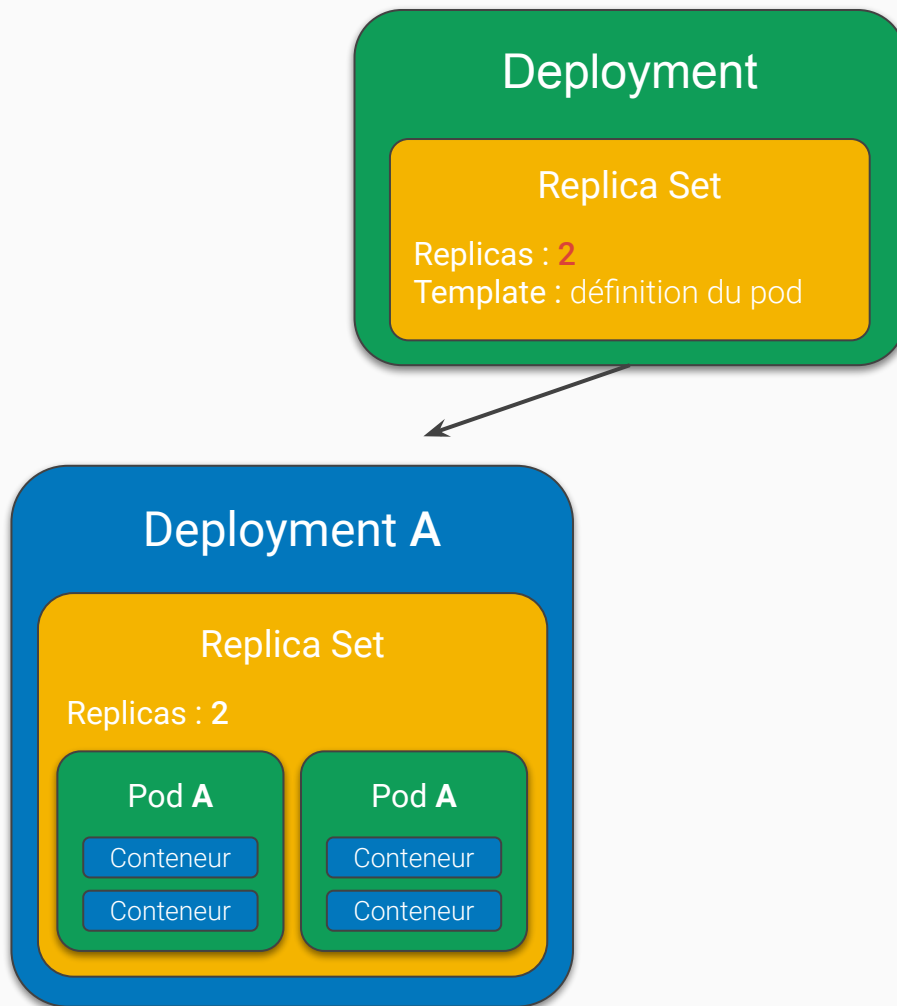
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

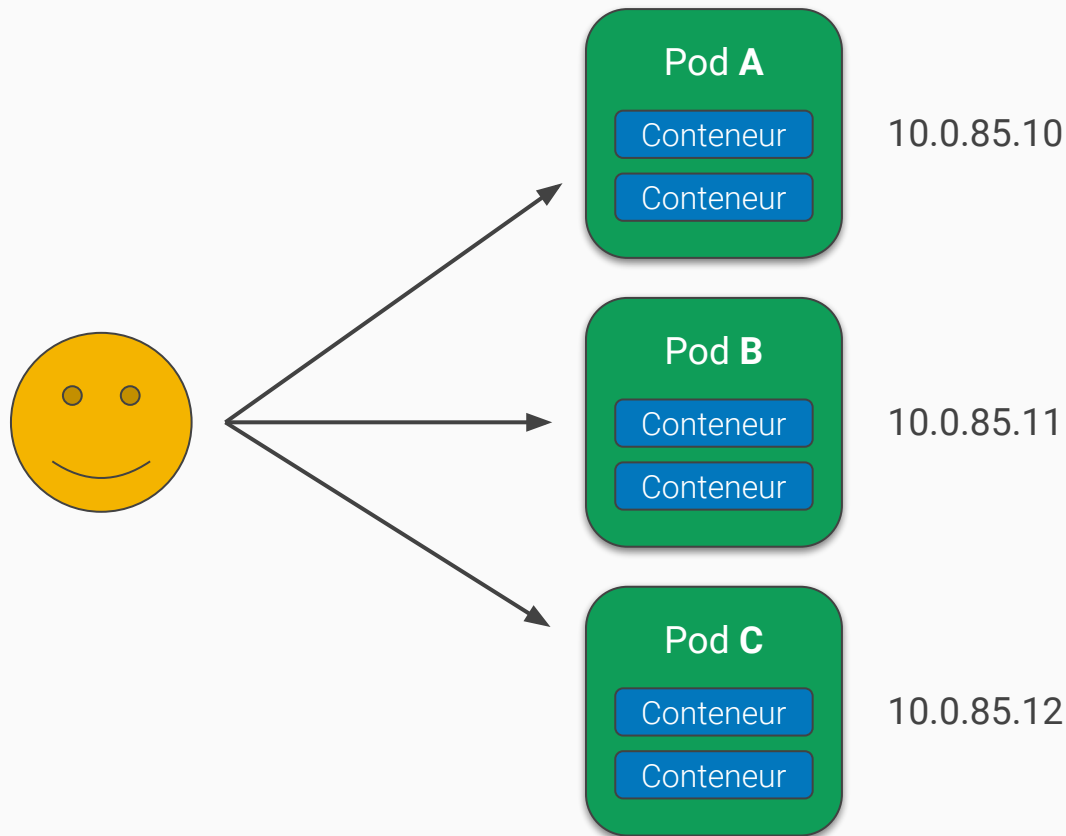
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

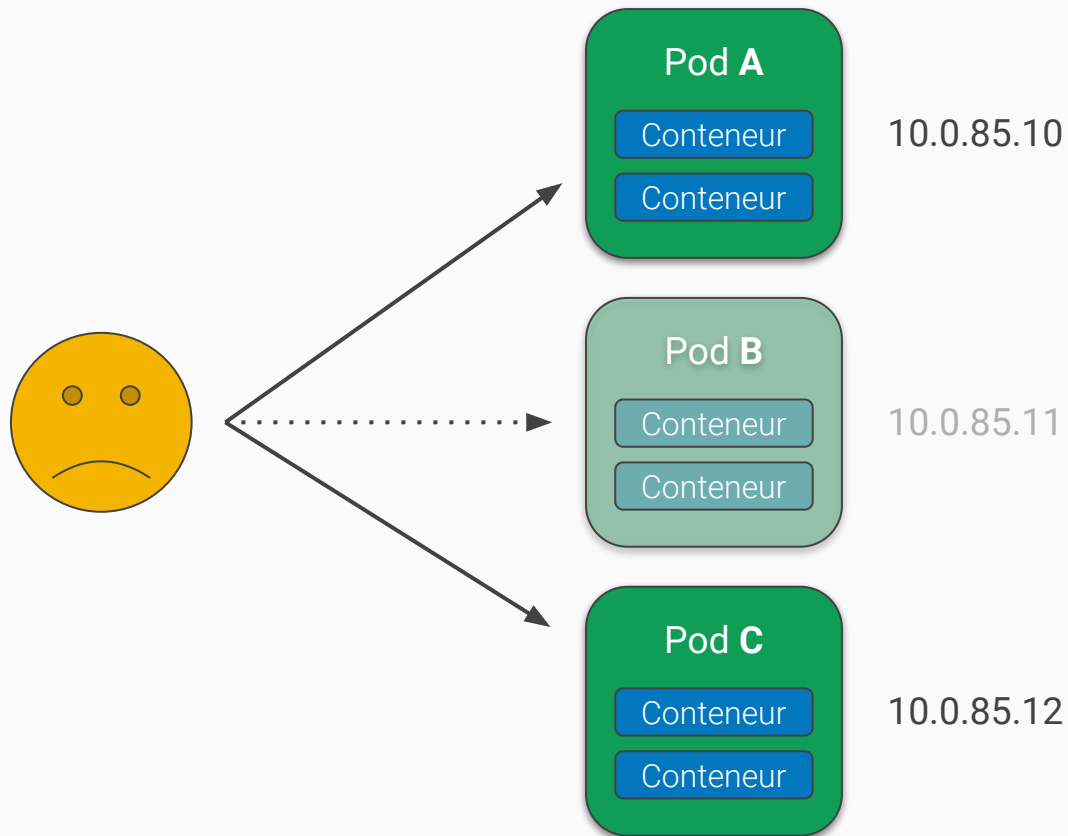
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

Service

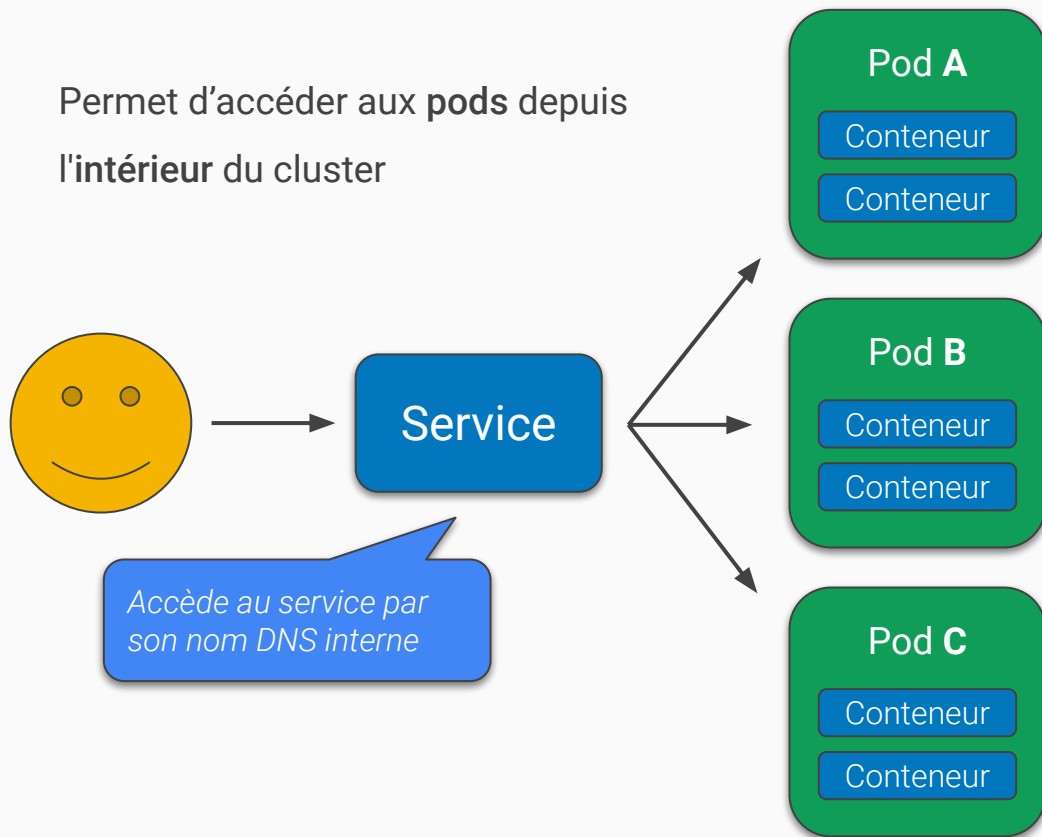
Secret

Config map

Ingress

Persistent volume claim

Permet d'accéder aux **pods** depuis
l'**intérieur** du cluster



Les objets Kubernetes

Pod

Replica set

Deployment

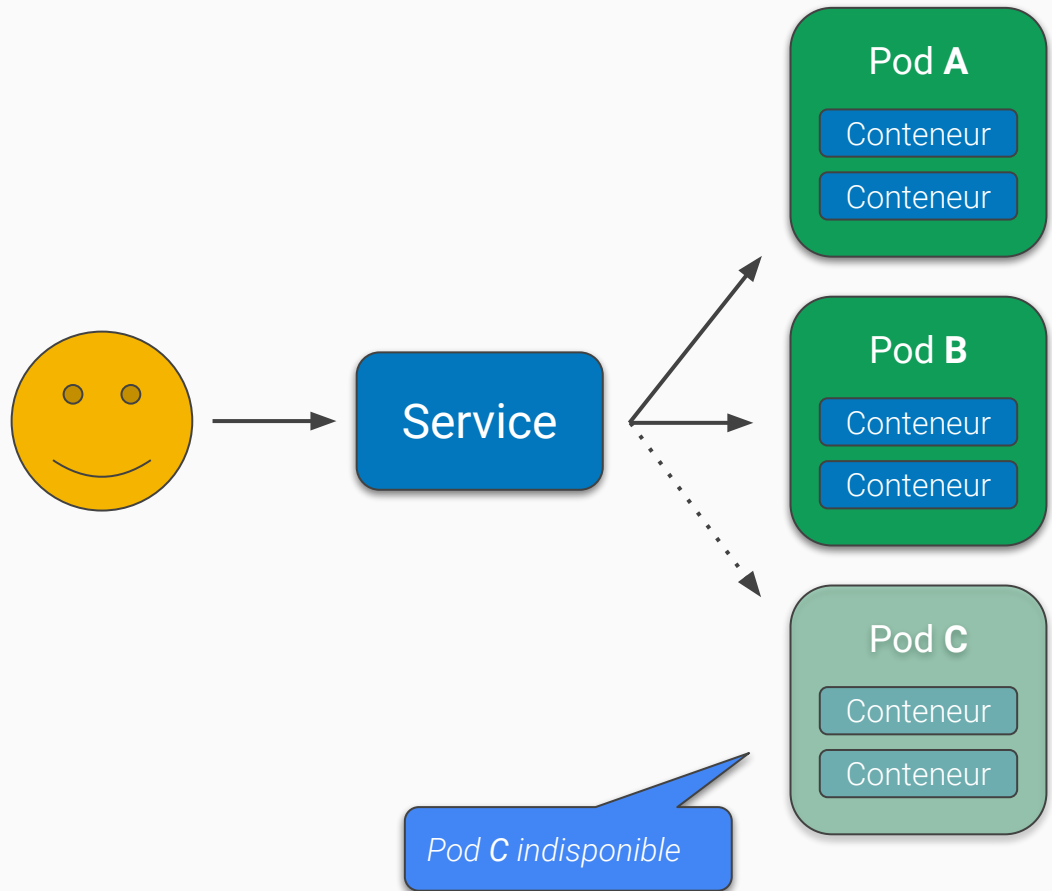
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

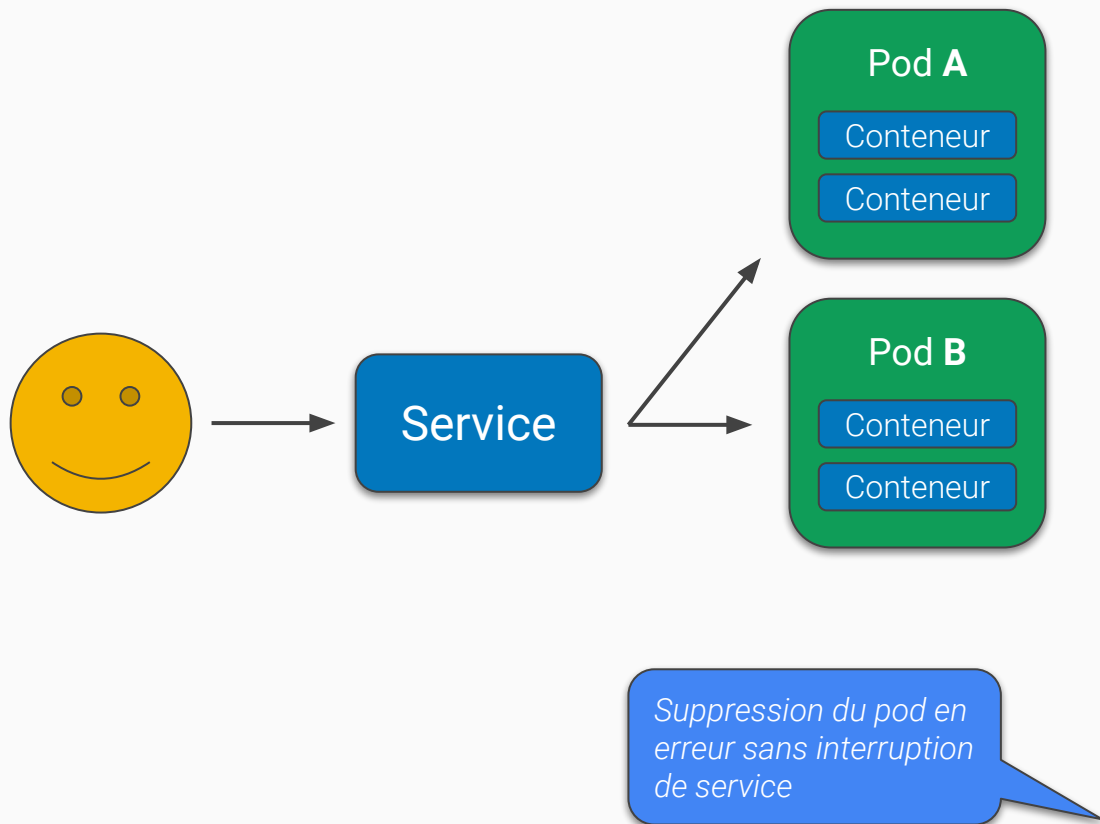
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

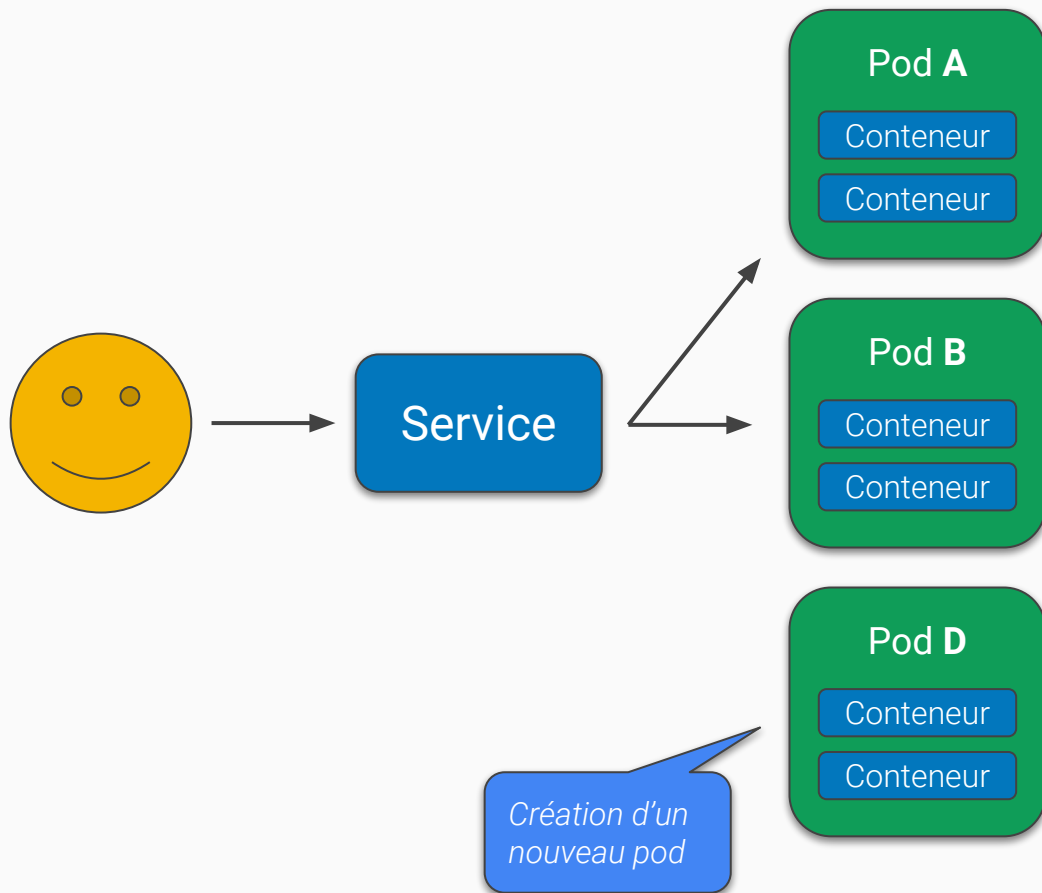
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

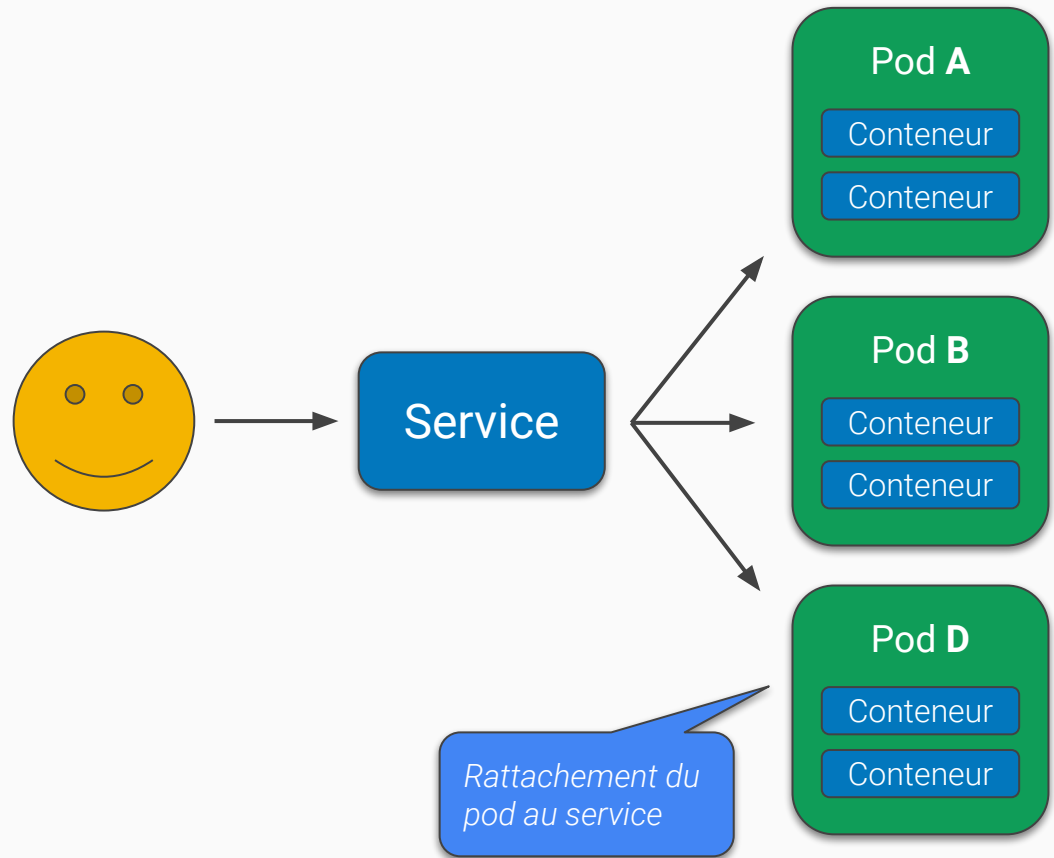
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

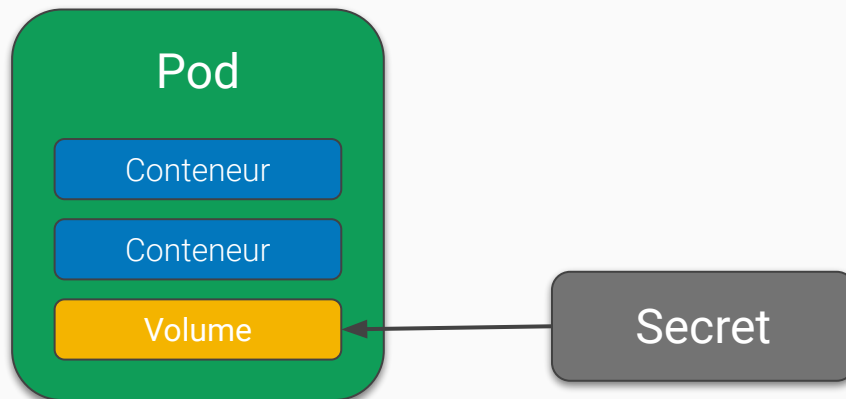
Service

Secret

Config map

Ingress

Persistent volume claim



- Permet de stocker des **informations sensibles**
- Exploitable via des **volumes** sur des conteneurs
- Encodé en **base64**
- Sous forme de fichier
- Sous forme de clé - valeur

Les objets Kubernetes

Pod

Replica set

Deployment

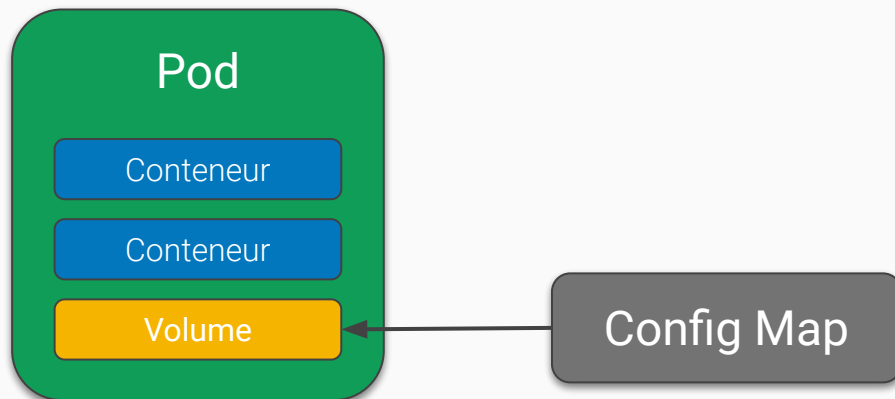
Service

Secret

Config map

Ingress

Persistent volume claim



- Permet de stocker de la configuration
- Exploitable via des **volumes** sur des conteneurs
- Sous forme de fichier
- Sous forme de clé - valeur

Les objets Kubernetes

Pod

Replica set

Deployment

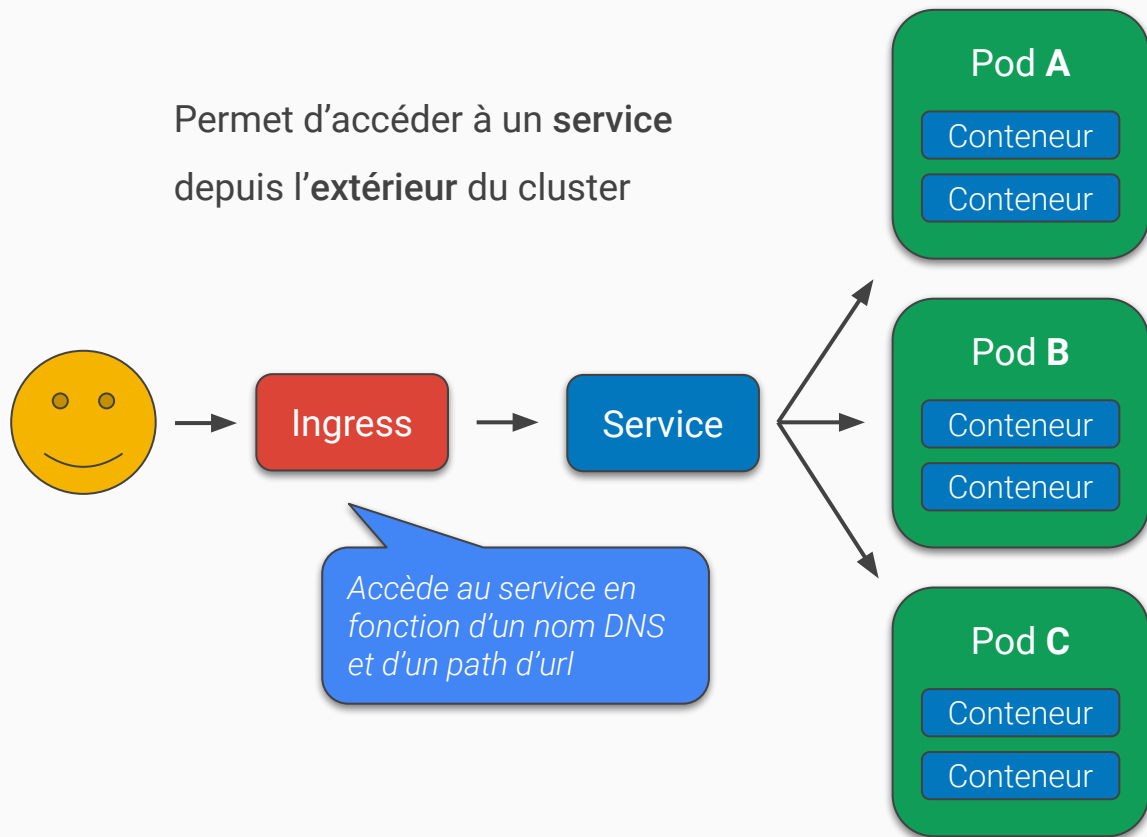
Service

Secret

Config map

Ingress

Persistent volume claim



Les objets Kubernetes

Pod

Replica set

Deployment

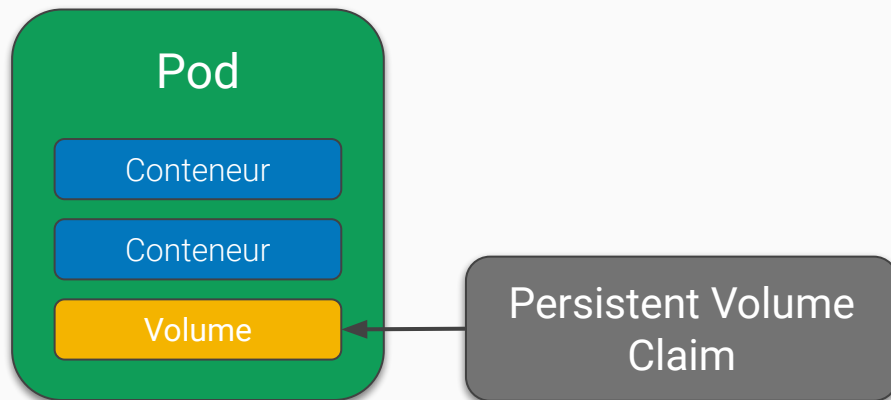
Service

Secret

Config map

Ingress

Persistent volume claim



- Permet l'obtention d'un espace de stockage
- Défini la quantité de stockage
- Défini la politiques d'accès : lecture seule, écriture

Démo

Kubernetes





<http://bit.ly/seblaporte-tadx-container>

Merci

Avez-vous des questions ?



@seb__laporte

