

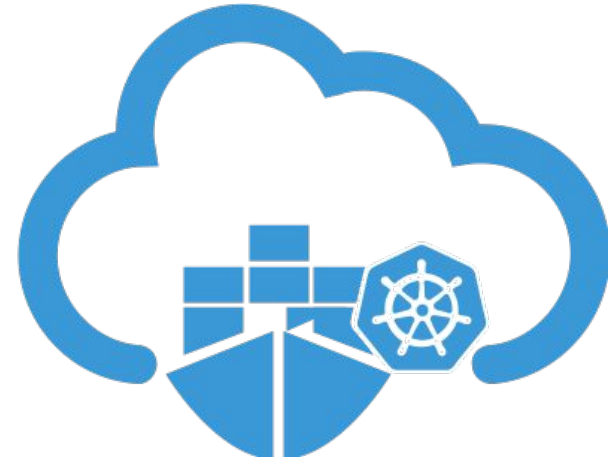


Kubernetes

Fannie LE FEL
Mathieu SALLES


Sommaire

- A. Présentation de la technologie
- B. Objectif du tutoriel
- C. Configurations
- D. Code source et script
- E. Conclusion



A. Présentation de la technologie

1. Depuis quand existe Kubernetes ?

- 2003-2004 : publication du système Borg (gestion de clusters)

- 2013 : Kubernetes est la version open source de Omega (management de clusters évolutifs)
La communauté s'agrandit



- 2015 : Kubernetes 1.0
La communauté s'agrandit
1ère édition de la plus grandes conférences technologiques d'entreprise



- 2016 : Kubernetes 1.2 et 1.3
Nouvel outil : Minikube
Le plus gros déploiement de Kubernetes
Prise en charge par Windows Server



- 2017 : La communauté s'agrandit



Kubernetes 1.6, 1.7, 1.8, 1.9
Présentation d'outils



- 2018 : Kubernetes 1.10, 1.11



2. Qu'est-ce que Kubernetes ?

Projet open source créé par



détenu par la



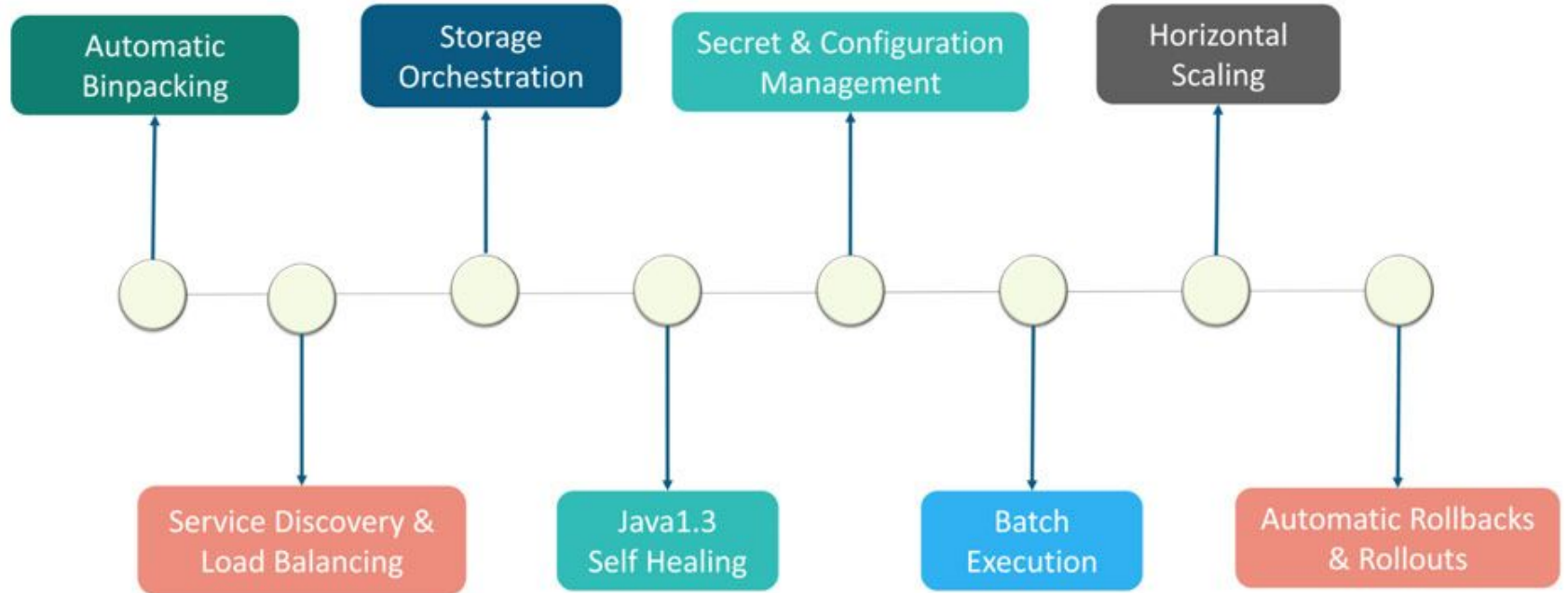
CLOUD NATIVE
COMPUTING FOUNDATION

- Orchestrateur de conteneurs
- Automatisation du déploiement
- Automatisation de la gestion d'applications multi-conteneurs
- Exécution et coordination des applications conteneurisées sur un cluster de machines
- Gestion entière du cycle de vie des applications

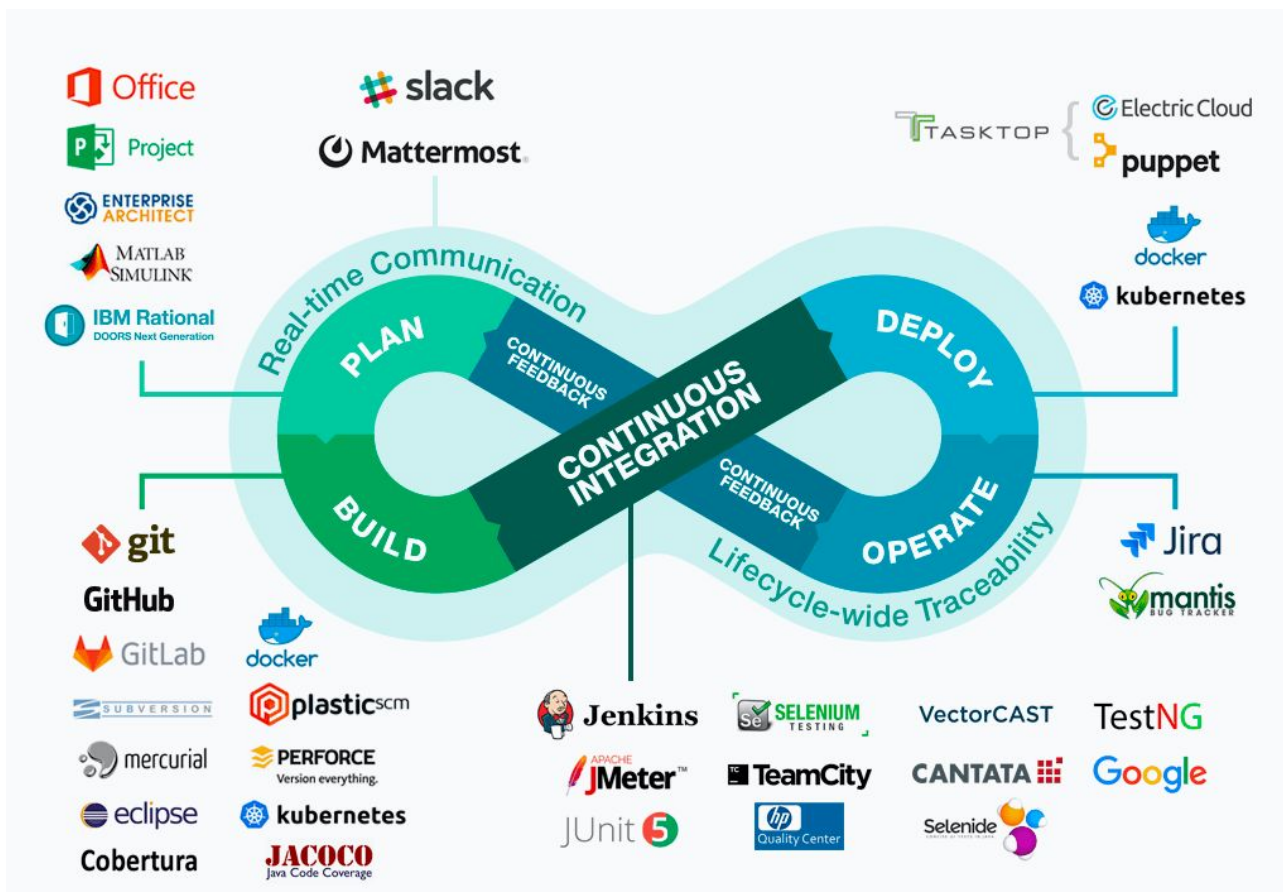
Leader sur le marché de l'orchestration de conteneurs



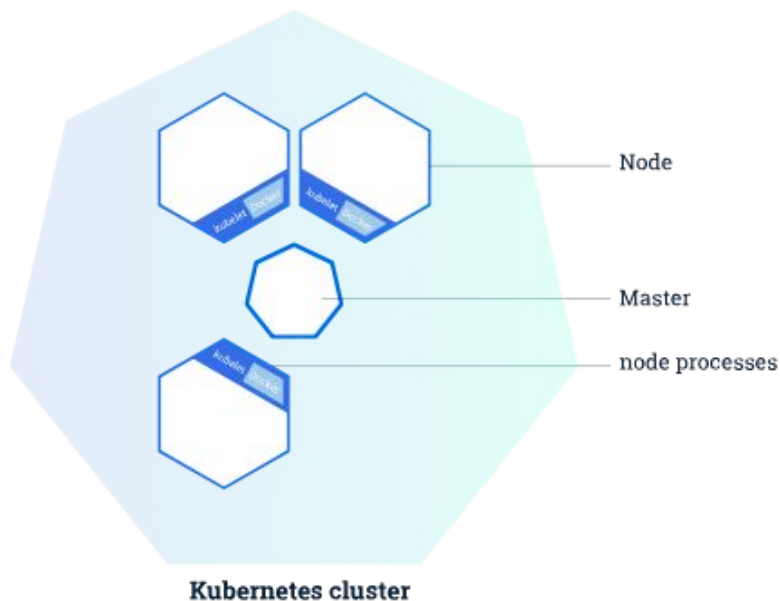
3. Quelles sont les fonctionnalités de Kubernetes ?



4. Pour qui est fait Kubernetes ?



5. Comment fonctionne Kubernetes ?



Principale abstraction : le cluster

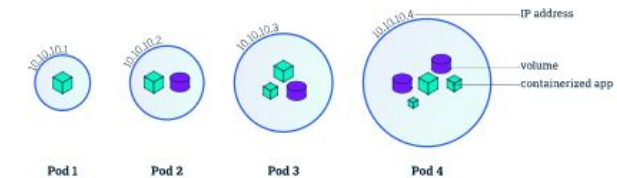
1. Un cluster doit avoir un master
2. Un cluster contient des noeuds Kubernetes (machines physiques ou virtuelles)
3. Un noeud exécute des pods
4. Un pod représente une seule instance d'une application / processus
5. Un pod se constitue d'un ou plusieurs conteneurs
6. Un conteneur est lancé et répliqué dans le pod

5. Comment fonctionne Kubernetes ?

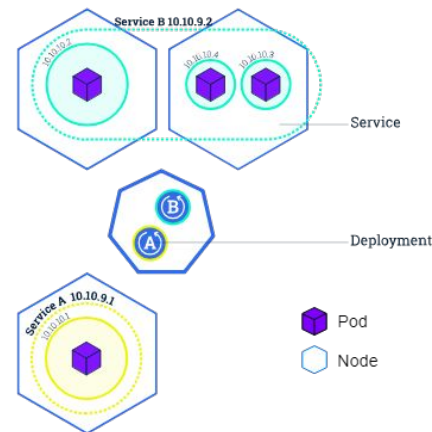
Autres abstractions

1. Contrôleur
 - a. Gère la façon dont les pods sont déployés, créés ou détruits
2. Service
 - a. Assure la persistance des applications même si les pods sont détruits
 - b. Décrit la façon dont un groupe de pods peut être accédé via le réseau
3. Scheduler
 - a. Répartit les workloads entre les noeuds pour assurer l'équilibre entre les ressources
 - b. Garantit que les déploiements correspondent aux besoins des applications
4. Controller manager
 - a. Assure que l'état du système correspond à l'état désiré défini dans les paramètres de configuration

Pods overview

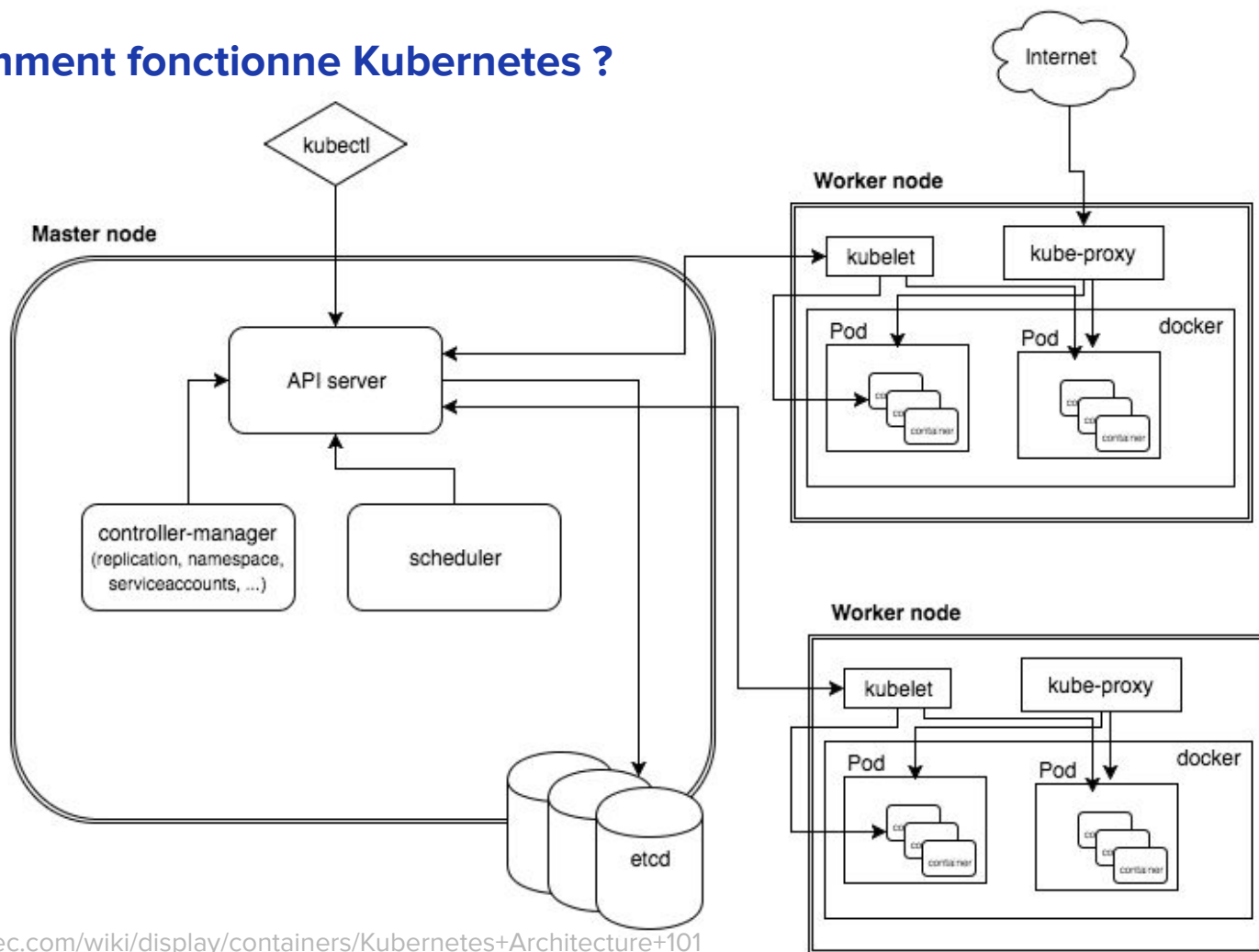


<https://kubernetes.io/docs/tutorials/kubernetes-basics/explore/explore-intro/>



<https://kubernetes.io/docs/tutorials/kubernetes-basics/expose/expose-intro/>

5. Comment fonctionne Kubernetes ?



6. Quels sont les composants Kubernetes ?

ClusterIp

Valeur par défaut

Service est uniquement accessible à partir du cluster Kubernetes

LoadBalancer

Service accessible en externe via la fonctionnalité d'équilibrage de charge d'un fournisseur de cloud

Le fournisseur de cloud créera un équilibreur de charge, qui acheminera ensuite automatiquement les demandes vers le service Kubernetes

NodePort

Service accessible sur un port statique sur chaque nœud du cluster. Cela signifie que le service peut gérer les demandes provenant de l'extérieur du cluster

ETCD

Base de données "clé-valeur" consistante et hautement disponible utilisée comme mémoire de sauvegarde pour toutes les données du cluster

6. Quels sont les composants Kubernetes ?

Kubelet

Ensemble de PodSpecs qui sont fournis par divers mécanismes (principalement par l'apiserver) et s'assure que les conteneurs décrits dans ces PodSpecs fonctionnent et sont en bonne santé

Kube-scheduler

Fonction spécifique à la charge de travail, riche en politiques et tenant compte de la topologie, qui a un impact significatif sur la disponibilité, la performance et la capacité

Kube-proxy

Maintient les règles réseau sur les nœuds

Contrôleur

Boucle de contrôle qui surveille l'état partagé de du cluster par l'intermédiaire de l'apiserver et effectue des changements en essayant de faire passer l'état actuel à l'état souhaité

7. Kubernetes Service NodePort : exemple avec un fichier YAML

```
kind: Service
apiVersion: v1
```

```
metadata:
```

```
  name: hostname-service
```

```
spec:
```

```
  type: NodePort
```

```
  selector:
```

```
    app: echo-hostname
```

```
  ports:
```

```
    - nodePort: 30163
```

```
      port: 8080
```

```
      targetPort: 80
```

Make the service available
to network requests from
external clients

Forward requests to pods
with label of this value

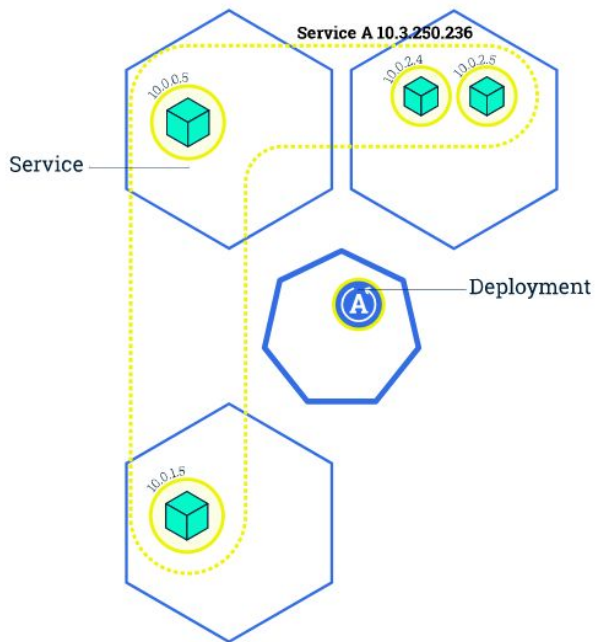
nodePort
access service via this external port number

port
port number exposed internally in cluster

targetPort
port that containers are listening on

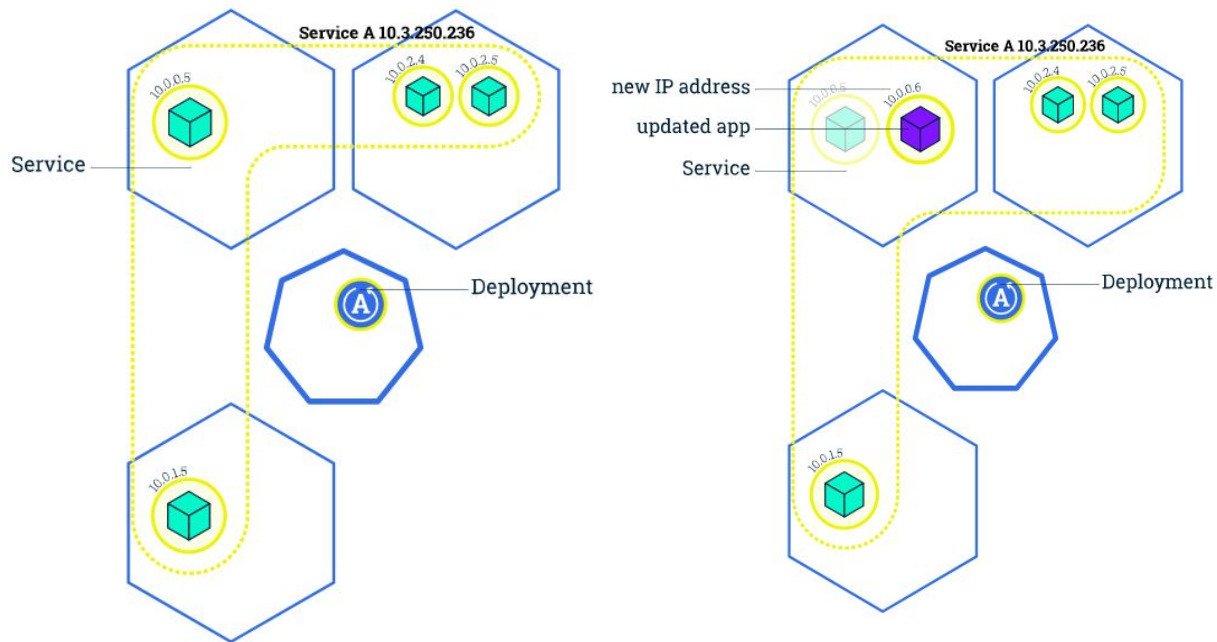
Comment fonctionne Kubernetes ?

Rolling update



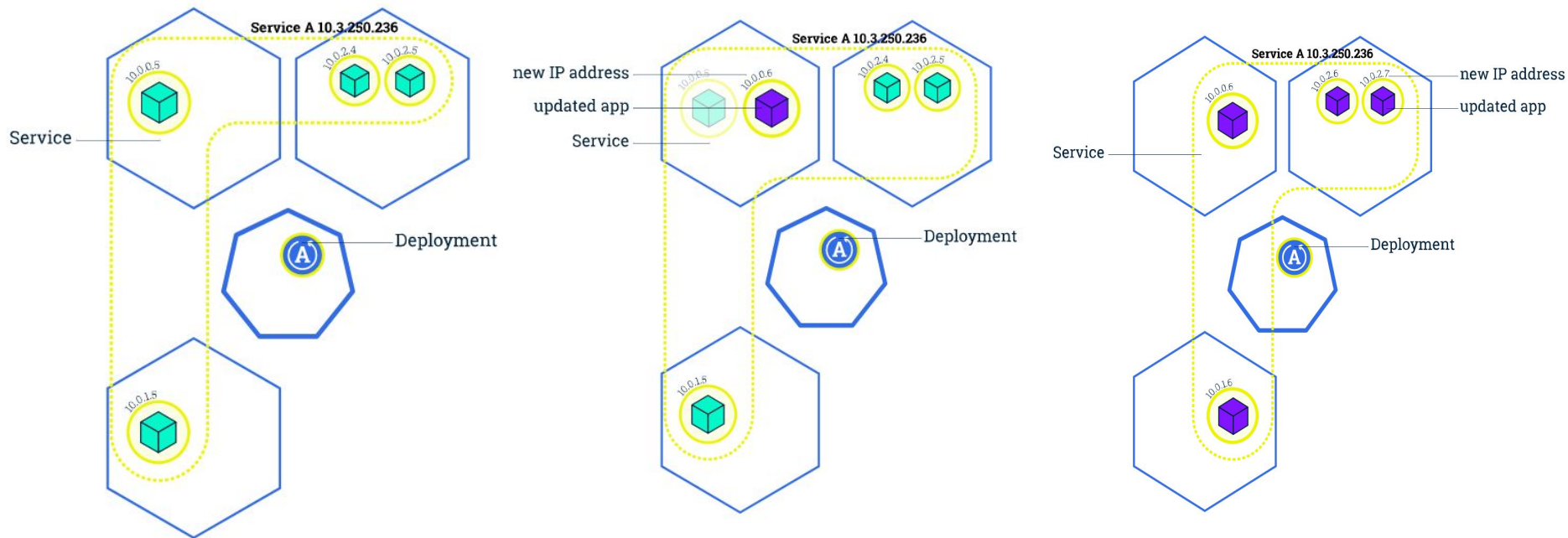
Comment fonctionne Kubernetes ?

Rolling update



Comment fonctionne Kubernetes ?

Rolling update



Objectif du tutoriel

Contexte, description et résultats

Contexte

Projets agiles : la gestion des conteneurs facilite le cycle de développement, de test et de déploiement

Déplacer facilement des conteneurs d'un niveau à l'autre

Automatisation de nombreuses étapes de travail

Description

Plate-forme de conteneur

Plate-forme de microservices

Plate-forme cloud portable

Au lieu de laisser tourner des machines inutilisées, Kubernetes peut libérer ces ressources et les utiliser pour d'autres tâches ou simplement ne pas les utiliser afin d'économiser des coûts

Résultats

Création et déploiement agile d'application

Séparation des besoins entre Dev et Ops

Consistance entre les environnements de développement, tests et production

Micro-services faiblement couplés, distribués, élastiques

Configurations

Lignes de commandes

Installation de Docker pour Linux

1. Installer Docker : `apt install docker.io`
2. Vérifier la version de Docker installée : `docker --version`
3. Vérifier que la virtualisation est supportée : `grep -E --color 'vmx|svm' /proc/cpuinfo` Si la réponse n'est pas vide alors c'est supporté.

Installation de Minikube

1. Installation de Minikube : `curl -Lo minikube https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64 \ && chmod +x minikube`
2. Ajouter minikube en tant que variable d'environnement
 - `mkdir -p /usr/local/bin/`
 - `install minikube /usr/local/bin/`
3. Lancer le Minikube avec la machine virtuelle de votre choix : `minikube start --vm-driver=driver_name`
4. Vérification de la réussite de l'installation : `minikube status`

Lignes de commandes

Déployer une image

Lancer Minikube (permet de lancer une machine virtuelle qui héberge Kubernetes) : `minikube start --memory 8000 --cpus 2 --kubernetes-version v1.6.0`

Déploiement avec une image Docker

1. Pour lancer une image trouvée sur DockerHub : `kubectl run my-nginx --image=nginx --port=80`
2. Pour exposer l'image créée : `kubectl expose deployment my-nginx --port=80 --type=LoadBalancer` Le loadbalancer sert à répartir les tâches.
3. Vérification : connaître son port en tapant la commande `get service` et taper `localhost:port` sur votre navigateur.

Déploiement avec un fichier yaml

Pour lancer et exposer une image : `kubectl apply -f https://k8s.io/examples/controllers/nginx-deployment.yaml`

Lignes de commandes

Dashboard

La dashboard permet une meilleure gestion du déploiement pour des personnes moins familières avec les commandes de Kubernetes.

1. Pour mettre en place la dashboard : `sudo kubectl apply -f https://raw.githubusercontent.com/kubernetes/dashboard/v2.0.0-beta8/aio/deploy/recommended.yaml`
2. Pour pouvoir changer l'accessibilité de la dashboard : `sudo kubectl proxy`
3. Rentrer l'URL suivante dans un navigateur pour accéder à votre dashboard :
<http://localhost:8001/api/v1/namespaces/kubernetes-dashboard/services/https:kubernetes-dashboard:/proxy/#/deployment?namespace=default>
4. Trouver un token permettant de se connecter à la dashboard : `sudo kubectl get secret` (Celle-ci va retourner une liste de *secret*).
5. Mettre le nom du secret dans la commande suivante : `sudo kubectl describe secret default-name-secret` (Celle-ci va retourner une liste d'information dont le token, que l'on peut utiliser pour se connecter dans la dashboard).

Code source et script

<https://github.com/FannieLeFel/Tutoriel-Kubernetes>

Conclusion

Conclusion

Kubernetes, un outil...

- open source
- facilitant la création d'applications
- simple par son utilisation
- complexe par son nombre de fonctionnalités
- très utilisé et apprécié malgré des concurrents sérieux

