

01 - Introduction à Kubernetes

Kubernetes est une plateforme d'orchestration de conteneurs. Elle répond aux défis des applications modernes composées de multiples services.

Kubernetes offre une gestion efficace des déploiements, des mises à jour et de la scalabilité.

 **par Tancrede SUARD**

Code : KUB-A-1 (2024)



Pourquoi Kubernetes ?

1

Haute disponibilité

Fonctionnalités d'auto-récupération et de gestion des pannes.

2

Scalabilité

Ajustement automatique du nombre d'instances selon les besoins.

3

Mises à jour continues

Déploiement progressif de nouvelles versions avec peu ou pas d'indisponibilité.

Défis Modernes du Déploiement



Montée en charge

Adaptation aux variations de trafic sans intervention manuelle.



Gestion des versions

Déploiement progressif des nouvelles versions.



Résilience

Redémarrage automatique des conteneurs défectueux pour maintenir la disponibilité.

Rôle de Kubernetes dans l'Infrastructure

Modularité

Développement et maintenance indépendants des composants d'application.

Automatisation

Gestion automatique du placement des conteneurs et des ressources.

Flexibilité

Facilite la transition vers des architectures microservices.

Historique et Contexte de Kubernetes

1

Origines : Google Borg

Inspiré de l'orchestrateur interne de Google, introduit dans les années 2000.

2

Open-Source

Google rend Kubernetes open-source en 2015.

3

CNCF

Transfert à la Cloud Native Computing Foundation pour promouvoir l'interopérabilité.

Alternatives et Concurrency

Docker Swarm

Solution plus simple mais moins évolutive que Kubernetes.

Apache Mesos

Orchestration de conteneurs et d'autres ressources avec une approche différente.

Kubernetes

Solution la plus populaire grâce à sa flexibilité et son évolutivité.

Principes de Base de l'Architecture Kubernetes

1

Control Plane

Gère l'état du cluster et orchestre les tâches.

2

Worker Nodes

Exécutent les conteneurs et gèrent les pods.

3

Composants

API Server, Scheduler, Controller Manager, ETCD, Kubelet, kube-proxy.

Objets Principaux de Kubernetes

Pod	Unité de base contenant un ou plusieurs conteneurs
Service	Expose un groupe de pods
Namespace	Isolation logique des ressources
Deployment	Gère l'état souhaité des pods

Réseau et Communication dans Kubernetes

1

Modèle de réseau plat

Permet aux pods de communiquer librement.

2

Plugins réseau

Calico, Flannel, Weave pour gérer la communication entre pods.

3

Service Discovery

Identification des services par leurs noms dans le cluster.

Kubernetes et Docker



Kubernetes

Plateforme d'orchestration compatible avec divers runtimes de conteneurs.



Docker

Runtime de conteneurs le plus couramment utilisé avec Kubernetes.