

# Conteneurs : tout ce qu'un développeur d'applications professionnelles doit savoir

Même si elle apparaît comme un ancêtre à côté des normes informatiques du cloud, la technologie des conteneurs [s'est récemment révélée essentielle](#) à la transformation numérique dans de nombreuses organisations du monde entier. Pour comprendre ce phénomène, il faut tout d'abord s'intéresser à ce que sont réellement les conteneurs, puis comprendre pourquoi les développeurs d'applications professionnelles privilégient cette technique actuellement.

## Que sont les conteneurs ?

Le conteneur est une unité normalisée regroupant tous les composants et toutes les dépendances d'une application, y compris son code, son environnement d'exécution, ses fichiers binaires et ses bibliothèques. Il constitue dès lors un package logique et ordonné, qui est facile à gérer et à déplacer d'un environnement informatique à l'autre. Il contient tout ce dont l'application a besoin pour fonctionner, et lui permet d'avoir toujours la même apparence et le même comportement, qu'elle soit exécutée sur votre ordinateur portable, sur un serveur de test, dans un centre de données privé ou dans le cloud public.

Les conteneurs sont généralement considérés comme une alternative aux machines virtuelles (VM) qui permettent elles aussi d'isoler les ressources destinées aux applications. Tandis que les VM virtualisent le matériel sous-jacent, les conteneurs opèrent une virtualisation au niveau du système d'exploitation. Plusieurs conteneurs peuvent ainsi s'exécuter directement sur un noyau de système d'exploitation partagé. Ainsi, les conteneurs démarrent bien plus rapidement et monopolisent beaucoup moins de mémoire que les VM, qui doivent toujours démarrer un système d'exploitation complet.

## En quoi les conteneurs influencent-ils le développement des applications professionnelles ?

Pour les développeurs d'applications professionnelles, les conteneurs sont une révolution en termes de productivité, de flexibilité et de valeur commerciale. Grâce aux conteneurs, les développeurs peuvent créer des environnements prévisibles, isolés des autres applications, et travailler à partir d'hypothèses qui se vérifient en production. Un conteneur peut renfermer les dépendances logicielles requises par l'application, par exemple différentes versions des environnements d'exécution du langage de programmation et d'autres bibliothèques logicielles. L'application peut ainsi fonctionner de façon cohérente dans tous les contextes de déploiement. Résultat : les développeurs et les équipes informatiques passent moins de temps à corriger les bugs et à diagnostiquer les différences entre les environnements, et peuvent s'investir davantage dans la création des nouvelles fonctionnalités demandées par les utilisateurs.

Ces avantages ne sont qu'un début, et c'est pourquoi les développeurs ont largement œuvré pour l'adoption de la technologie des conteneurs. Ces efforts semblent porter leurs fruits puisqu'en 2015, deux tiers des organisations ont investi dans les conteneurs afin de les exploiter en développement et en production, selon une [enquête réalisée par NGINX](#).

## Où les applications en conteneurs peuvent-elles s'exécuter ?

L'un des grands avantages des applications basées sur les conteneurs est qu'elles fonctionnent pratiquement partout : sur les systèmes d'exploitation Linux, Windows et Mac, sur des machines virtuelles ou des systèmes nus, sur des machines de développeurs ou dans des centres de données sur site et, bien sûr, dans le cloud public. Cependant, pour tirer pleinement parti de cette technologie et éviter les problèmes de verrouillage liés aux fournisseurs, les développeurs ont besoin d'une solution orchestratrice qui soit aussi ouverte que les conteneurs eux-mêmes.

C'est l'idée du projet [Kubernetes](#), une technologie Open Source développée par Google pour automatiser le déploiement, l'évolutivité et la gestion des conteneurs. Kubernetes s'appuie sur la technologie que Google utilise pour déployer des milliards de conteneurs par semaine et permet aux développeurs de créer, de tester et d'exécuter des applications en conteneurs dans tous les contextes possibles. Pour s'épargner différentes tâches de routine, ils peuvent désormais choisir la commodité et l'efficacité d'un environnement d'orchestration entièrement géré tel que [Google Container Engine](#). Ils peuvent également opter pour un niveau de contrôle supérieur grâce à Kubernetes, qui leur permet de déployer eux-mêmes leurs applications, que ce soit dans une infrastructure sur site ou dans les services cloud publics de n'importe quel fournisseur.

## Qui utilise des conteneurs ?

De nombreuses applications parmi les plus récentes et les plus géniales exploitent la flexibilité de développement offerte par les conteneurs. À ce jour, le plus grand déploiement Google Container Engine a été réalisé par [Niantic pour le lancement de Pokémon GO](#) auprès d'un public international de joueurs passionnés. Grâce aux conteneurs, Niantic a réussi à se focaliser sur l'amélioration permanente du jeu, avec parfois plusieurs déploiements par jour. L'orchestration et l'évolution du cluster de conteneurs de Pokémon GO ont été entièrement gérées par Kubernetes et Google Container Engine. [CCP Games](#), un éditeur de jeux islandais, a lui aussi utilisé Kubernetes pour dépolluer des codes développés au cours de ces vingt dernières années et automatiser ses processus pour que son équipe puisse travailler plus efficacement.

Les petites équipes créatives des grandes entreprises devraient s'inspirer de la flexibilité de sociétés telles que Niantic et CCP Games. Grâce aux conteneurs, elles pourront se concentrer entièrement sur leurs applications et leurs utilisateurs, en laissant à d'autres le soin de gérer l'évolution et la maintenance de l'infrastructure.

**“Container Engine offre une infrastructure à la fois très performante et flexible qui nous permet de faire évoluer nos composants séparément, pour une efficacité optimale.”**

– George Yianni, inventeur de Hue et responsable des systèmes d'éclairage connectés pour la maison chez [Philips Lighting](#)

## Comment faire pour se lancer ?

Vous souhaitez essayer [Kubernetes](#) ou [Google Container Engine](#) ? [Inscrivez-vous](#) à un essai gratuit de Google Cloud Platform et suivez [ces étapes](#). Sinon, amusez-vous à [intégrer un peu de la magie de Minecraft](#).