



# Créer des applications modernes sur AWS

Gérez moins. Créez rapidement.

Innovez davantage.



# Les applications modernes changent la façon dont vous fournissez de la valeur client

Des organisations du monde entier misent principalement sur l'innovation, la satisfaction du client et l'efficacité opérationnelle<sup>1</sup>. Pour atteindre ces objectifs, les entreprises doivent montrer l'exemple à travers leurs applications. Or, nombreuses sont les entreprises qui se compliquent la vie pour créer leurs applications et peinent à trouver un juste équilibre entre la gestion de la technologie et la publication de nouvelles fonctionnalités.

Si le cloud promet l'agilité, celle-ci n'est pas automatique. Alors que les organisations cherchent à accélérer l'innovation, à tirer davantage de leurs données et à créer de nouvelles expériences client, elles doivent moderniser la façon dont elles créent et exploitent les applications. Les applications modernes sont construites avec une combinaison de modèles d'architecture modulaire, de modèles opérationnels sans serveur et de processus de développement agiles.

Dans cet e-book, nous vous guiderons à travers les trois voies qui vous permettront d'établir les bases du développement d'applications modernes au sein de votre organisation. Nous découvrirons également comment le développement d'applications modernes avec AWS peut aider votre organisation à innover, réduire les coûts, accélérer la mise sur le marché et améliorer la fiabilité.

<sup>1</sup> Enquête IDC : Future Enterprise Resiliency & Spending – Wave 10, IDC, octobre 2021



# Les applications modernes renforcent la capacité des innovateurs numériques

## Innover, c'est être à l'écoute des clients

À l'occasion d'une récente commission consultative d'IDC réunissant des CIO du monde entier, il a été souligné l'importance cruciale de garder une perspective centrée sur le client afin de favoriser l'innovation numérique. Un innovateur numérique ne présente pas une solution à la recherche d'un problème. Le moteur de l'innovation, c'est de regarder le parcours du client tel qu'il le perçoit et de créer dans le but d'éliminer les points de friction qu'il rencontre.

Se concentrer réellement sur le client, cela implique de prendre des décisions métier en partant du point de vue du client. Cela se traduit également par une amélioration continue de vos produits et services afin de fournir à vos clients des résultats encore plus satisfaisants. De même, cela suppose d'être pleinement à l'écoute des priorités du client afin de pouvoir anticiper ses besoins dans le temps et ainsi d'entretenir votre capacité à inventer et à itérer en sa faveur. Cette approche est ce que l'on appelle le « cycle de l'innovation ».

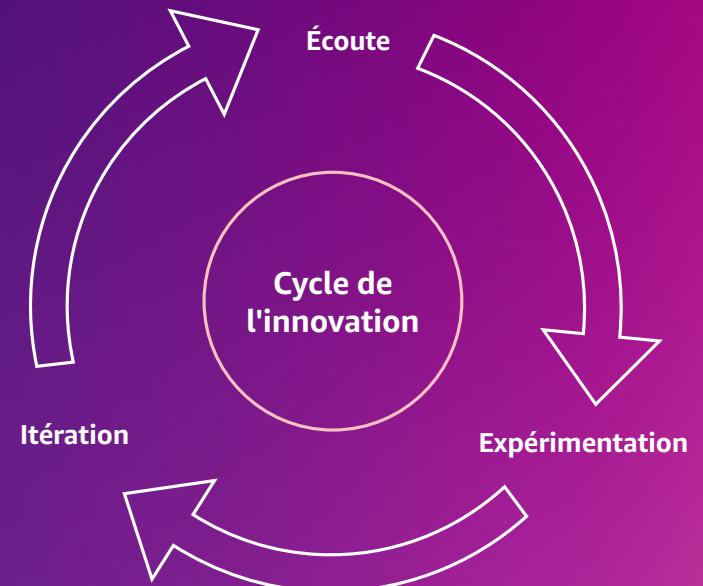
Le moteur de toute innovation est la demande exprimée par le client. L'innovation s'améliore ensuite grâce aux retours d'informations des clients, et se répète de manière constante (et rentable) jusqu'à ce que la demande change et que le cycle tout entier recommence. Plus vite vos équipes pourront faire tourner leur propre cycle d'innovation, plus vous serez à même de créer des applications modernes et plus vous vous démarquerez de la concurrence.

<sup>2</sup> « IDC Perspective: Reframing the Digital-First Customer Journey: Key Takeaways from IDC's Global CIO Advisory Board — April 2022 Edition »



En tirant parti des nouvelles architectures prêtes pour MACH (microservices, API, cloud et sans écran), les organisations peuvent rapidement introduire de nouvelles fonctionnalités qui s'intègrent parfaitement à leur pile technologique et répondre aux opportunités de croissance en tant qu'entreprises misant sur le numérique et centrées sur le client<sup>2</sup>. »

Marci Maddox, Directrice de recherche, Stratégies en expérience numérique, IDC



En créant des applications modernes sur AWS, vous **accéderez plus rapidement au marché**. En accélérant le cycle de création et de publication et en vous déchargeant des frais généraux opérationnels, vos développeurs pourront créer rapidement de nouvelles fonctionnalités. Vous **favoriserez l'innovation** en utilisant une architecture modulaire qui offrira aux équipes la liberté d'expérimenter des composants d'application individuels sans risque pour l'ensemble de l'application. En automatisant les procédures de test et la surveillance à chaque étape du cycle de vie du développement, vous **améliorerez la fiabilité**. Et vous améliorerez le **coût total de possession** (TCO) grâce à un modèle de tarification à l'utilisation qui réduit les coûts liés au surprovisionnement ou au paiement des ressources non utilisées.

Pour créer des applications modernes, vous devrez peut-être reconSIDéRer les bases sur lesquelles elles reposent. Même si la transformation numérique peut chambouler une organisation, le processus n'est pas nécessairement brutal. De nombreuses organisations se lancent bille en tête dans la création de nouvelles applications modernes dans le cloud, mais beaucoup d'autres optent pour une approche hybride, qui consiste à créer des applications qui s'exécuteront aussi bien sur site que dans le cloud. Pour autant, ces cheminementS ne sont pas incompatibles. De fait, le processus implique généralement des cheminementS parallèles, avec des équipes et des charges de travail qui suivent des parcours distincts, avançant de manière opportune, étape par étape.

**50 %**  
des dépenses informatiques seront directement affectées à la transformation numérique d'ici 2023

**67 %**  
des dirigeants pensent qu'ils doivent accélérer le rythme pour rester compétitifs

**90 %**  
des nouvelles applications devraient être natives cloud d'ici 2025

# Trois voies de modernisation qui génèrent une véritable valeur métier

À travers notre expérience de création d'applications pour Amazon.com et de prestation de services à des millions de clients AWS, nous avons identifié trois voies que nos clients peuvent emprunter pour concrétiser leur vision de la modernisation des applications et ainsi générer de la valeur pour leur entreprise.

## **1. Créer des applications avec Kubernetes sur AWS et étendre les applications Kubernetes existantes à AWS pour une distribution hybride.**

Les organisations qui exécutent Kubernetes sur site ou qui envisagent de créer des applications sur Kubernetes peuvent exécuter ces charges de travail sur AWS pour accroître la fiabilité, la sécurité et la capacité de mise à l'échelle. Cette approche offre un contrôle au niveau du serveur, des outils opérationnels courants et un accès intégré au riche portefeuille de services AWS.

## **2. Créer des applications sur des conteneurs gérés et une architecture sans serveur.**

Pour la création d'applications ou de fonctionnalités, nous recommandons aux organisations d'utiliser des conteneurs gérés, des technologies sans serveur et des bases de données sur mesure pour optimiser l'agilité et accélérer le développement tout en éliminant la gestion des serveurs.

## **3. Adopter un modèle DevOps moderne.**

Pour opérer un changement culturel qui permette de fournir des applications modernes à grande échelle, les organisations peuvent tirer parti des services et des outils DevOps pour créer et fournir rapidement des produits tout en conservant une exigence élevée en matière de sécurité et de gouvernance.

Nous explorerons chaque voie plus en détail et nous vous montrerons comment chacune d'entre elles peut contribuer à accroître l'agilité, à réduire les coûts et à créer de meilleures applications qui contribuent au succès de l'entreprise. Quel que soit le point de départ de la modernisation de vos applications, le résultat doit être le même : des applications sûres, fiables, évolutives et rapidement disponibles pour vos clients et partenaires dès le début de vos efforts communs.

# Les applications modernes dans la pratique

## Les trois voies du développement d'applications modernes

Le développement d'applications modernes est une approche puissante pour concevoir, créer et gérer des logiciels dans le cloud. Cette approche éprouvée accroît l'agilité de vos équipes de développement ainsi que la fiabilité et la sécurité de vos applications, vous permettant de créer et de publier de meilleurs produits plus rapidement. Nous avons collaboré avec des organisations très variées. Aussi bien celles dont les équipes opérationnelles expérimentées et étoffées aspirent à conserver une maîtrise maximale avec des outils d'orchestration portables, que celles dont les équipes limitées en ressources préfèrent se décharger des tâches opérationnelles pour se concentrer sur la création et l'itération d'applications.

Ces expériences nous ont permis d'identifier les trois piliers du développement d'applications modernes qui vous aideront dans votre parcours de modernisation :

- 1** Crée des applications Kubernetes et étendre celles qui existent déjà
- 2** S'appuyer sur des conteneurs gérés et une architecture sans serveur
- 3** Adopter un modèle DevOps moderne

# Créer des applications Kubernetes et étendre celles qui existent déjà

Aujourd'hui, les clients sont de plus en plus nombreux à se réinventer, créant des applications qui permettent à leurs organisations de tirer pleinement avantage du cloud. En réalité, il n'existe pas une seule voie de modernisation ; les applications peuvent coexister dans différents états (dont certains sont évoqués dans cet eBook). Des organisations du monde entier continuent d'adopter et d'intensifier leur utilisation de conteneurs, qu'elles considèrent comme un moyen léger et portable d'exécuter et de déployer un grand nombre de nouvelles applications. Pour beaucoup, les conteneurs sont devenus le meilleur moyen de prendre en charge les architectures applicatives complexes qui demandent de la flexibilité, un déploiement rapide et une innovation constante dans un environnement qui peut s'exécuter quasiment partout.

Avant de passer aux applications conteneurisées, les organisations doivent réfléchir à où et comment allouer leurs ressources opérationnelles. L'élément central de cette réflexion consiste à bien choisir la plateforme d'orchestration de conteneurs la mieux adaptée à la gestion des déploiements d'applications complexes et à l'automatisation de l'infrastructure. Les organisations qui ont une préférence pour les logiciels open source (OSS) ont adopté en masse Kubernetes en raison de son écosystème et de sa communauté dynamiques, de ses API open source cohérentes et de sa grande flexibilité.

En tant que projet open source, Kubernetes forme la base idéale pour exécuter des applications conteneurisées à grande échelle dans le cloud ou dans une configuration cloud hybride, offrant un moyen consistant et standardisé de mettre à l'échelle et de gérer les applications. Les organisations qui prévoient d'utiliser Kubernetes, qui l'utilisent déjà ou qui envisagent de créer des applications sur Kubernetes, peuvent exécuter leurs applications sur Amazon Elastic Kubernetes Service (EKS) pour accroître la fiabilité, la sécurité et la capacité de mise à l'échelle, tout en conservant la maîtrise au niveau du serveur, les outils opérationnels courants et un accès intégré à l'ensemble du portefeuille de services AWS.

**Pour les déploiements sur site ou hybrides, Amazon EKS Anywhere apporte l'expérience Amazon EKS dans votre centre de données, ce qui vous permet de conserver les applications sensibles à la latence ou les applications réglementées dans un cloud privé sur site tout en tirant parti d'un ensemble d'outils courant dans les environnements.**

Kubernetes est une technologie d'orchestration de conteneurs puissante et extensible qui vous permet de déployer et de gérer des applications conteneurisées à grande échelle. Du fait de la nature extensible de Kubernetes (et des diverses possibilités de conception qu'il offre), la création d'un cluster EKS sur mesure est un processus long et fastidieux. Elle consiste notamment à intégrer un large éventail d'outils open source et de services AWS qui peuvent nécessiter une connaissance technique approfondie d'AWS et de Kubernetes. Pour simplifier ce processus, EKS Blueprints a été conçu dans l'optique de faciliter l'intégration du paysage d'outils Kubernetes. L'allocation d'Amazon EKS s'en trouve simplifiée pour répondre à vos besoins spécifiques en matière d'applications en utilisant des outils bien connus.

# Pourquoi utiliser Kubernetes aujourd'hui

Kubernetes étant un projet open source, vous pouvez l'utiliser pour exécuter vos applications conteneurisées n'importe où sans avoir à changer vos outils opérationnels. Kubernetes est tenu à jour par une grande communauté de volontaires et s'améliore constamment. Par ailleurs, beaucoup d'autres projets et d'éditeurs open source créent et tiennent à jour des logiciels compatibles Kubernetes dont vous pouvez vous servir pour améliorer et étendre votre architecture.



## Exécutez des applications à grande échelle

Parce que Kubernetes vous permet de définir des applications conteneurisées complexes et de les exécuter à grande échelle sur un cluster de serveurs.



## Déplacez des applications sans interruption

Avec Kubernetes, il est possible de déplacer des applications conteneurisées vers des déploiements de production sur le cloud en utilisant les mêmes outils opérationnels et sans interrompre les machines de développement locales.



## Exécutez vos applications n'importe où

Exécutez des clusters Kubernetes hautement disponibles et évolutifs sur AWS tout en maintenant une compatibilité totale avec vos déploiements Kubernetes s'exécutant sur site.



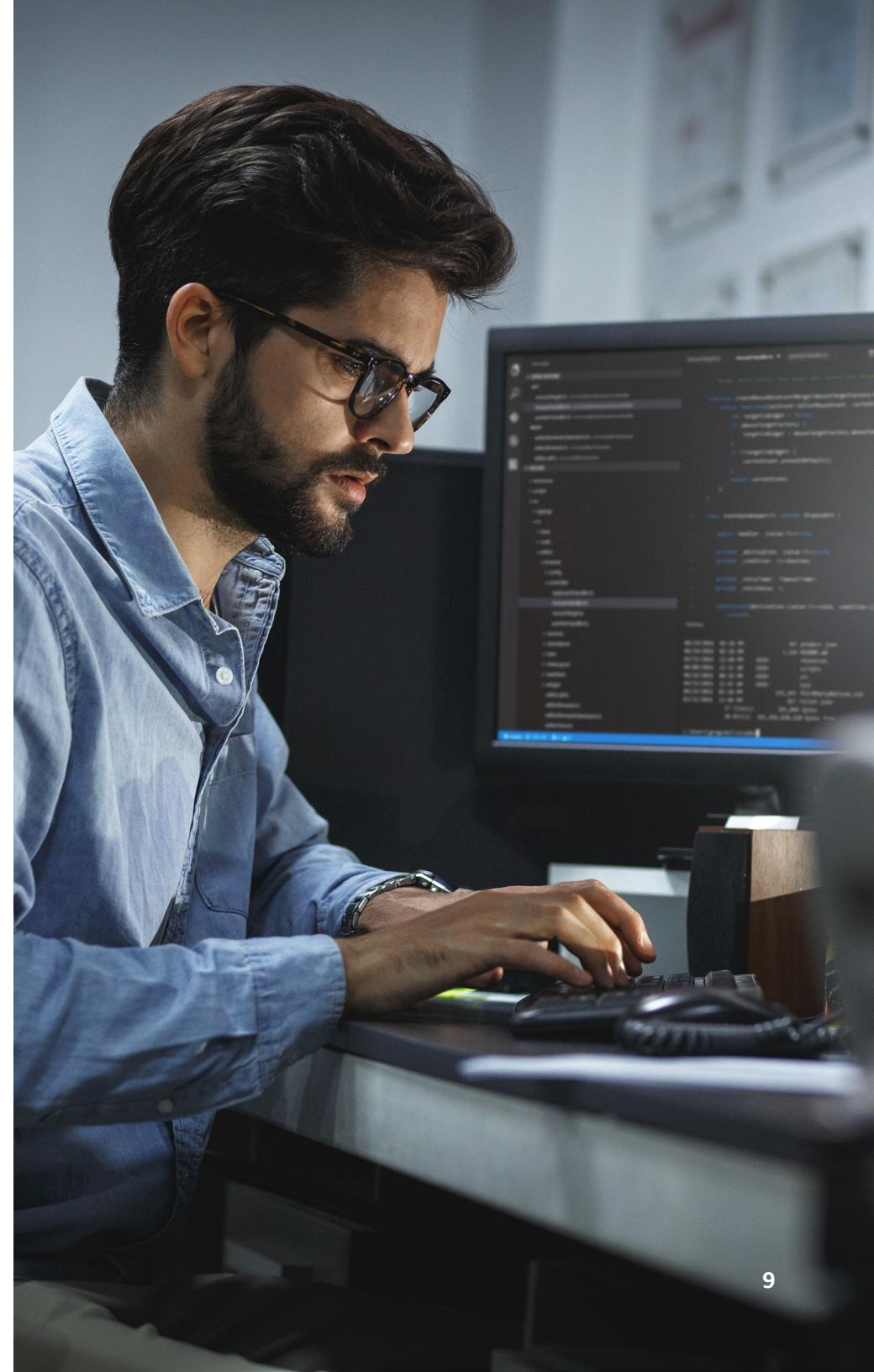
## Ajoutez de nouvelles fonctionnalités

Kubernetes étant un projet open source, il est facile d'ajouter de nouvelles fonctionnalités. Les développeurs et les entreprises de sa grande communauté créent des extensions, des intégrations et des plug-ins qui offrent des possibilités supplémentaires aux utilisateurs de Kubernetes.

Les organisations qui possèdent des compétences internes sur le plan des opérations et de Kubernetes peuvent déployer Amazon EKS ou Kubernetes sur Amazon Elastic Compute Cloud (EC2) afin de profiter d'une infrastructure évolutive, sécurisée et hautement disponible. Grâce à cette approche, vous bénéficiez d'un contrôle complet sur vos instances de calcul, les processus de déploiement, la maintenance et la mise à l'échelle. En s'appuyant sur l'infrastructure mondiale la plus étendue, les régions et les zones de disponibilité AWS offrent une fiabilité, une disponibilité et une tolérance aux pannes maximales.

Pour les organisations qui souhaitent utiliser Amazon EKS sans la complexité associée à l'exécution et à la mise à l'échelle de l'infrastructure sous-jacente, AWS Fargate est un environnement sans serveur qui n'exige pas de posséder et d'exécuter une infrastructure de calcul ni d'en gérer le cycle de vie. Avec AWS Fargate, vous pouvez :

- **éliminer les frais généraux opérationnels liés à la mise à l'échelle, à la correction, à la sécurisation et à la gestion de serveurs ;**
- **améliorer la sécurité grâce à l'isolation de charge de travail inhérente à sa conception (les pods Amazon EKS s'exécutent dans leur propre environnement d'exécution dédié) ;**
- **payer uniquement en fonction de votre utilisation (Fargate se met à l'échelle pour se rapprocher au mieux de vos besoins en ressources).**



## TÉMOIGNAGES CLIENTS

# New Relic

**New Relic** est un acteur phare dans le domaine de l'observabilité, offrant aux ingénieurs une plateforme SaaS (Software-as-a-Service) qui analyse, dépanne et optimise l'ensemble de leur pile logicielle. En 2020, New Relic s'est engagé sur la voie de la transformation de son modèle économique, passant d'un modèle de tarification basé sur l'hôte à un modèle basé sur la consommation. Par ailleurs, l'entreprise a mis en place un niveau de service gratuit pour tous les clients et s'est préparée à une augmentation significative de la demande sur sa plateforme. New Relic a décidé de migrer l'intégralité de sa plateforme sur AWS pour s'adapter à sa croissance attendue. En seulement huit mois, l'entreprise a tiré parti de ses propres outils et processus, mais aussi des capacités et des offres AWS, pour migrer plus de 20 000 serveurs et refactoriser l'intégralité de sa plateforme de services. La plateforme a été refactorisée avec **Amazon Elastic Kubernetes Service** (Amazon EKS). Pendant ce programme de migration et de refactorisation, New Relic a pu constater des améliorations notables du point de vue de l'efficacité de l'ingénierie et de la résilience de la plateforme, avec des perspectives de capacité de mise à l'échelle à long terme pour soutenir ses objectifs de croissance ambitieux.

[En savoir plus ici ›](#)



**« Grâce à AWS, nous pouvons nous concentrer sur nos compétences de base, ce qui nous permet de sortir des produits plus aboutis plus vite et de manière plus fréquente. »**

Andrew Harnett,  
Directeur principal des services gérés, New Relic



# S'appuyer sur des conteneurs gérés et des services sans serveur

Comme nous l'avons vu précédemment, les conteneurs offrent un moyen léger et portable d'exécuter et de déployer des applications. La conteneurisation des applications existantes est souvent la première étape du parcours de modernisation d'une organisation. De nombreuses entreprises déplacent leurs applications dans des conteneurs pour permettre à leurs équipes de consacrer leur temps et leurs ressources à la création d'applications et non à la gestion des tâches opérationnelles. Et de plus en plus, elles créent leurs charges de travail sur des services gérés AWS comme Amazon Elastic Kubernetes Service (Amazon EKS) ou Amazon Elastic Container Service (Amazon ECS) avec AWS Fargate. Un service de conteneurs géré permet de réduire la charge opérationnelle tout en améliorant l'évolutivité, la fiabilité, la sécurité et la disponibilité. Avec les services de conteneurs gérés sur AWS, vous n'avez plus à vous soucier de la gestion des conteneurs. Au lieu de cela, vous pouvez affecter vos ressources à l'amélioration des compétences de vos équipes de façon à développer des applications modernes avec l'informatique sans serveur tout en accélérant l'innovation des applications.



## TÉMOIGNAGES CLIENTS

# Groupe Volkswagen

Le constructeur automobile mondial **Groupe Volkswagen** (Volkswagen) crée des applications de réalité virtuelle (RV) et de réalité augmentée (RA) afin de gagner du temps, de réduire les coûts et d'améliorer les performances pour des cas d'utilisation qui recouvrent l'ensemble du cycle de vie des produits, depuis les revues de conception et des simulations de formation jusqu'au guidage à distance dans l'usine. Dans l'optique de mettre à l'échelle ses applications de RA/RV, Volkswagen avait besoin d'un moyen plus simple, plus rapide et plus efficace de préparer du contenu 3D et de le diffuser dans son organisation. La société a développé une architecture cloud qui lui permet d'accomplir deux choses :

- 1 automatiser les pipelines de préparation de données 3D de Volkswagen pour une optimisation plus simple et plus rapide des modèles 3D ;
- 2 tester la capacité de rendu et de diffusion à distance de graphismes 3D sur des casques de RA/RV, en remplacement des casques connectés à des stations de travail coûteuses.

Pour atteindre ces objectifs, Volkswagen a fait appel à Amazon Web Services (AWS) pour migrer des charges de travail spécifiques de rendu de RV et d'optimisation de données 3D dans le cloud. La société utilise également une application personnalisée, appelée Innoactive Portal, qui a été développée avec la société de logiciels de RV **Innoactive** pour déployer et gérer des applications de RV. Grâce à plusieurs services AWS, notamment **Amazon Elastic Container Service** (Amazon ECS), service d'orchestration de conteneurs entièrement géré destiné à faciliter le déploiement, la gestion et la mise à l'échelle des applications conteneurisées, Volkswagen améliore l'efficacité de son pipeline de préparation de données 3D et en accroît la puissance et la vitesse de rendu dans le cloud.

[Lire le témoignage complet ici ›](#)



**VOLKSWAGEN**  
GROUP

**« Pour nous, utiliser AWS s'avère simple et efficace du fait des nombreux services natifs qui vont dans le sens de nos objectifs et de nos projets d'architecture. »**

Jan-Paul Brückmann

Propriétaire de produit, Volkswagen Digital Reality Hub



# Innover avec le modèle sans serveur

La modernisation du développement d'applications consiste à adopter des services, des pratiques et des stratégies qui permettent aux développeurs de créer des applications plus agiles. Et grâce à la rapidité et à la fiabilité d'une infrastructure moderne, les développeurs peuvent fournir des applications sécurisées qui, d'un simple prototype, vont être automatiquement mises à l'échelle pour des millions d'utilisateurs, avec la possibilité d'innover et de répondre plus rapidement au changement.

De nombreuses applications modernes sont créées en privilégiant le sans serveur, une stratégie qui donne la priorité à l'adoption de services sans serveur pour que les clients puissent gagner en agilité dans toute la pile d'applications. Avec les technologies sans serveur, vous n'avez plus à gérer de serveurs physiques et vous bénéficiez d'une scalabilité automatique, d'une haute disponibilité intégrée et d'un modèle de facturation en fonction de la valeur. Plutôt que de vous soucier de la gestion et de l'exploitation des serveurs ou des environnements d'exécution, vous pouvez vous concentrer sur l'innovation des produits tout en écourtant le délai de mise sur le marché.

En outre, des outils et des services Web et mobiles frontaux peuvent être créés sur AWS. La fiabilité de cette infrastructure aide les marques à fournir des applications sécurisées, hautement disponibles et évolutives automatiquement dans le monde entier.

## Facteurs clés à prendre en compte pour la création d'applications modernes évolutives

### Les microservices comme modèle architectural

Bien que votre application monolithique puisse être facile à gérer aujourd'hui, des difficultés surgissent souvent au fur et à mesure de votre croissance, notamment la manière de répartir la propriété de l'application entre vos équipes. Vous pouvez créer une solide culture de propriété, mais vous avez encore du mal à évoluer si l'architecture de votre application comprend des dépendances matérielles qui empêchent les équipes de s'approprier le produit final. C'est pourquoi nous vous recommandons de créer des architectures de microservices pour les applications qui se développent et évoluent rapidement. Les microservices sont l'expression architecturale d'une culture de propriété : ils divisent proprement les applications complexes afin qu'une seule équipe puisse les posséder et les faire fonctionner indépendamment.

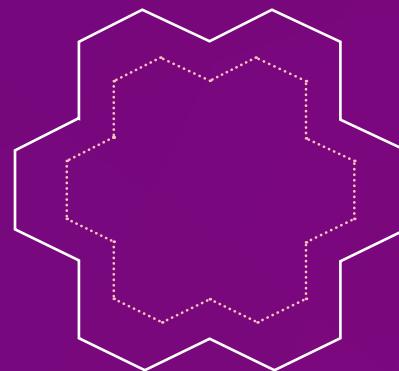
Avec un monolithe, de nombreux développeurs transmettent les changements via un pipeline de diffusion partagé, ce qui provoque des frictions à de nombreux points du cycle de vie. Au début, pendant le développement, les ingénieurs doivent coordonner leurs modifications pour s'assurer qu'ils ne brisent pas le code de quelqu'un d'autre. Pour mettre à niveau une bibliothèque partagée afin de profiter d'une nouvelle fonction, vous devez convaincre tous les autres de le faire en même temps - une conversation difficile s'il en est ! Et si vous souhaitez appliquer rapidement un correctif important pour votre fonction, vous devez toujours le fusionner avec les modifications en cours.

Après le développement, vous êtes également confronté à une surcharge lorsque vous faites passer les modifications dans le pipeline de diffusion. Même pour modifier une simple ligne de code, les ingénieurs doivent coordonner leurs modifications à l'avance, fusionner leur code, résoudre les conflits relatifs aux versions, recréer l'application tout entière, exécuter toute une série de tests et redéployer l'application.

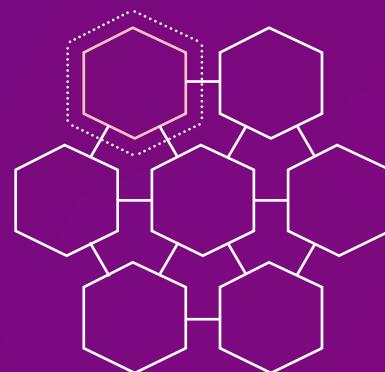
Avec l'architecture de microservices, une application est constituée de composants indépendants exécutant chaque processus applicatif en tant que service. Les services sont conçus à des fins de fonctionnalités métier, chacun exécutant une fonction unique. En raison de son indépendance d'exécution et de sa gestion par une équipe de développeurs unique, chaque service peut être mis à jour, déployé et mis à l'échelle afin de répondre à la demande de fonctions spécifiques d'une application. Par exemple, le panier d'une boutique en ligne conçu sous forme de microservice peut être utilisé par plusieurs clients à la fois, car il se met à l'échelle de manière indépendante en fonction de la demande. Les microservices s'échangent les données via des interfaces bien définies en utilisant des API, des événements ou des flux légers. Nos clients s'appuient de plus en plus sur des architectures axées sur les événements (des architectures dans lesquelles les actions sont déclenchées en réponse aux modifications de données) afin d'améliorer la capacité de mise à l'échelle et la résilience de leurs applications, tout en réduisant les coûts.

## Tous les aspects ou un seul aspect : deux types d'application distincts

### Applications monolithiques



### Microservices



Gestion de tous les aspects

Application unique

Déploiement intégral de l'application

Base de données unique

Organisation autour de couches technologiques

État dans chaque instance d'exécution

Une pile technologique pour l'ensemble de l'application

Gestion d'un seul aspect

Services de fonctions minimaux

Déploiement séparé et interaction entre les compartiments

Un magasin de données propre à chaque microservice

Organisation autour des fonctionnalités métier

Externalisation des états

Choix de la technologie spécifique à chaque microservice

## TÉMOIGNAGES CLIENTS

# Centrica

Centrica est une multinationale britannique de l'énergie et des services qui cherchait à réduire ses coûts et à augmenter son agilité en modifiant la manière dont son application était architecturée. Elle a choisi de se refactoriser en optant pour une architecture de microservices et d'adopter une stratégie sans serveur pour atteindre ces objectifs. Fortes de ce succès, d'autres équipes ont également adopté cette approche : le sans serveur est aujourd'hui couramment utilisé dans toute l'entreprise. Grâce au sans serveur, Centrica est désormais en mesure de voir les problèmes des clients et de les résoudre en temps réel, ce qui était impossible auparavant.

[Regarder la vidéo ›](#)



## TÉMOIGNAGES CLIENTS

# Liberty Mutual

Avec un chiffre d'affaires annuel de 40 milliards de dollars, Liberty Mutual se classe au sixième rang mondial des compagnies d'assurance IARD et fait figure de leader du secteur sur le plan de l'innovation technologique. Pour atteindre son objectif de devenir une entreprise numérique mondiale, l'assureur s'est concentré sur trois axes de transformation principaux : le caractère central du client, l'agilité et le développement natif cloud. À cette fin, la compagnie a pris une décision stratégique : adopter une approche sans serveur (une évolution visant à lui donner un avantage concurrentiel sur le marché mondial). Liberty Mutual a fait appel à AWS pour migrer ses systèmes sur site vers le cloud, se moderniser et favoriser la transformation à l'échelle de l'entreprise. En utilisant l'architecture sans serveur et en laissant le soin à AWS de s'occuper des tâches de gestion d'infrastructure comme l'allocation de capacité et l'application de correctifs, Liberty Mutual a réduit sa charge opérationnelle et a nettement diminué ses coûts. La compagnie a également eu recours à des solutions sans serveur pour pouvoir créer rapidement des applications plus agiles et de qualité. En éliminant les frais généraux opérationnels, l'architecture sans serveur facilite l'expérimentation, ce qui permet aux équipes de Liberty Mutual de publier rapidement, d'obtenir des commentaires et d'itérer pour accélérer la mise sur le marché.

[En savoir plus ici ›](#)



**« Notre collaboration, c'est bien plus qu'une simple relation client-fournisseur. Nous avons vraiment l'impression qu'AWS fait partie de notre équipe. »**

Dave Anderson,  
Directeur technologique, Liberty Mutual



# Un modèle sans serveur, autant que possible

Au fur et à mesure de l'évolution de vos modèles architecturaux et de vos processus de livraison des logiciels, vous souhaiterez probablement opter pour un modèle opérationnel qui vous permet de supprimer toute activité ne faisant pas partie des compétences de base de votre entreprise. Pour gagner en agilité et permettre une innovation rapide, nous vous recommandons de créer une architecture de microservices, d'exploiter et de déployer des logiciels en utilisant l'automatisation pour certains aspects comme la surveillance, l'allocation, la gestion des coûts, le déploiement, la sécurité et la gouvernance des applications. Le choix d'une stratégie privilégiant les technologies sans serveur dans la mesure du possible vous permet d'optimiser les avantages opérationnels d'AWS.

Avec un modèle opérationnel sans serveur, vous pouvez créer et exécuter des applications et des services sans louer ni gérer de serveurs. Cela élimine la nécessité de gérer des serveurs et fournit une mise à l'échelle flexible. Cette démarche vous permet également de payer uniquement en fonction de la valeur et de bénéficier d'une haute disponibilité automatisée. Avec ce modèle, vous créez et gérez les volets de votre application qui fournissent aux clients de la valeur, sans avoir à vous préoccuper des détails sous-jacents. De la création d'applications entièrement nouvelles à la migration d'applications existantes, le recours aux services sans serveur pour le calcul, les données et l'intégration vous permet de tirer pleinement parti de l'agilité du cloud.

## Comment nous définissons l'approche « sans serveur » chez AWS

Pour nous, le « sans serveur » est une approche caractérisée par l'élimination des tâches complexes indifférenciées que constituent les opérations de serveurs. Il s'agit d'une distinction importante, car cette approche vous permet de vous concentrer sur la création des applications, et non sur la gestion et la mise à l'échelle de l'infrastructure sous-jacente. Les quatre principes du modèle opérationnel sans serveur sont :

- 1 Aucune gestion de serveur** : il n'est pas nécessaire d'allouer ou de maintenir des serveurs. Il n'y a ni logiciels ni exécutions à installer, à maintenir ou à administrer.
- 2 Mise à l'échelle flexible** : votre application peut être mise à l'échelle automatiquement ou en ajustant sa capacité en basculant les unités de consommation (par exemple, le débit, la mémoire) plutôt que les unités de serveurs individuels.
- 3 Facturation en fonction de la valeur** : au lieu de payer pour des unités de serveur, payez pour ce qui compte à vos yeux : un débit ou une durée d'exécution cohérents.
- 4 Haute disponibilité automatisée** : le modèle opérationnel sans serveur offre une disponibilité et une tolérance aux pannes intégrées. Vous n'avez pas besoin de concevoir une architecture spécifique pour ces fonctionnalités, car les services exécutant l'application les fournissent par défaut.



Le modèle opérationnel sans serveur est idéal pour les entreprises à forte croissance désireuses d'innover rapidement. Il accélère le cycle d'innovation de votre entreprise en permettant à vos équipes d'être encore plus productives et de se concentrer davantage sur ces activités qui permettent de vous démarquer de vos concurrents.

# S'appuyer sur des conteneurs gérés et des services sans serveur

## Exploitation d'AWS Lambda et des services de conteneurs gérés sur AWS

Avec l'essor des conteneurs et de l'informatique sans serveur, les instances ne constituent plus votre seule option de cloud computing. Le choix du calcul optimal pour votre application moderne commence par l'examen de plusieurs questions. L'infrastructure autogérée améliore-t-elle les résultats de votre entreprise ? Avez-vous l'expertise requise ? Cet effort supplémentaire créera-t-il finalement de la valeur ?

Les clients choisissent de plus en plus de se décharger de la gestion des serveurs en adoptant des services de conteneurs comme Amazon ECS et Amazon EKS ou des services de calcul sans serveur orientés événements comme AWS Lambda.

En réalité, la plupart des clients utilisent une combinaison de services. Environ 80 % des clients de conteneurs AWS ont également adopté AWS Lambda<sup>3</sup>. L'utilisation des deux options présente des avantages, notamment les services entièrement gérés et profondément intégrés à l'infrastructure AWS, la prise en charge d'un large éventail de cas d'utilisation, l'absence de complexité et un vaste écosystème de partenaires.



<sup>3</sup> State of Serverless de Datalog, 2020

## Alors comment faire son choix ?

Les clients choisissent AWS Lambda lorsqu'ils ont des équipes qui se concentrent principalement sur l'écriture du code et qu'ils n'ont pas de limitations sur les instances ou les plateformes de conteneurs existantes. AWS Lambda offre une abstraction maximale de l'infrastructure, ce qui permet aux clients de publier leurs logiciels le plus rapidement possible. C'est ce qui explique pourquoi les nouvelles applications sont parfaitement adaptées à AWS Lambda.

Les clients choisissent souvent les conteneurs lorsqu'ils ont déjà investi dans des conteneurs, qu'ils ont des préférences pour Kubernetes et le modèle open source ou qu'ils ont des exigences spécifiques en matière de gestion ou de configuration d'infrastructure. Les conteneurs constituent le moyen le plus populaire pour empaqueter du code et représentent un excellent choix pour moderniser des applications héritées.



<b>AWS Lambda</b> Fonctions sans serveur
<b>AWS Fargate</b> Conteneurs sans serveur
<b>Amazon ECS/ Amazon EKS</b> Gestion de conteneurs en tant que service
<b>Amazon EC2</b> Infrastructure en tant que service

### Ce qu'AWS gère

Intégrations de sources de données  
Matériel physique, logiciels, mise en réseau, installations  
Provisionnement

Orchestration et mise en service des conteneurs  
Mise à l'échelle des clusters  
Matériel physique, système d'exploitation hôte/noyau, mise en réseau, installations

Plan de contrôle de l'orchestration des conteneurs  
Matériel physique, réseaux, installations

Matériel physique, réseaux, installations

### Ce que le client gère

Code d'application

Code d'application  
Intégrations de sources de données  
Configuration et mises à jour de la sécurité, configuration du réseau

Code d'application  
Intégrations de sources de données  
Configuration et mises à jour de la sécurité, configuration du réseau, pare-feu

Code d'application  
Intégrations de sources de données  
Mise à l'échelle  
Tâches de gestion

Tâches de gestion

Clusters de travail  
Tâches de gestion

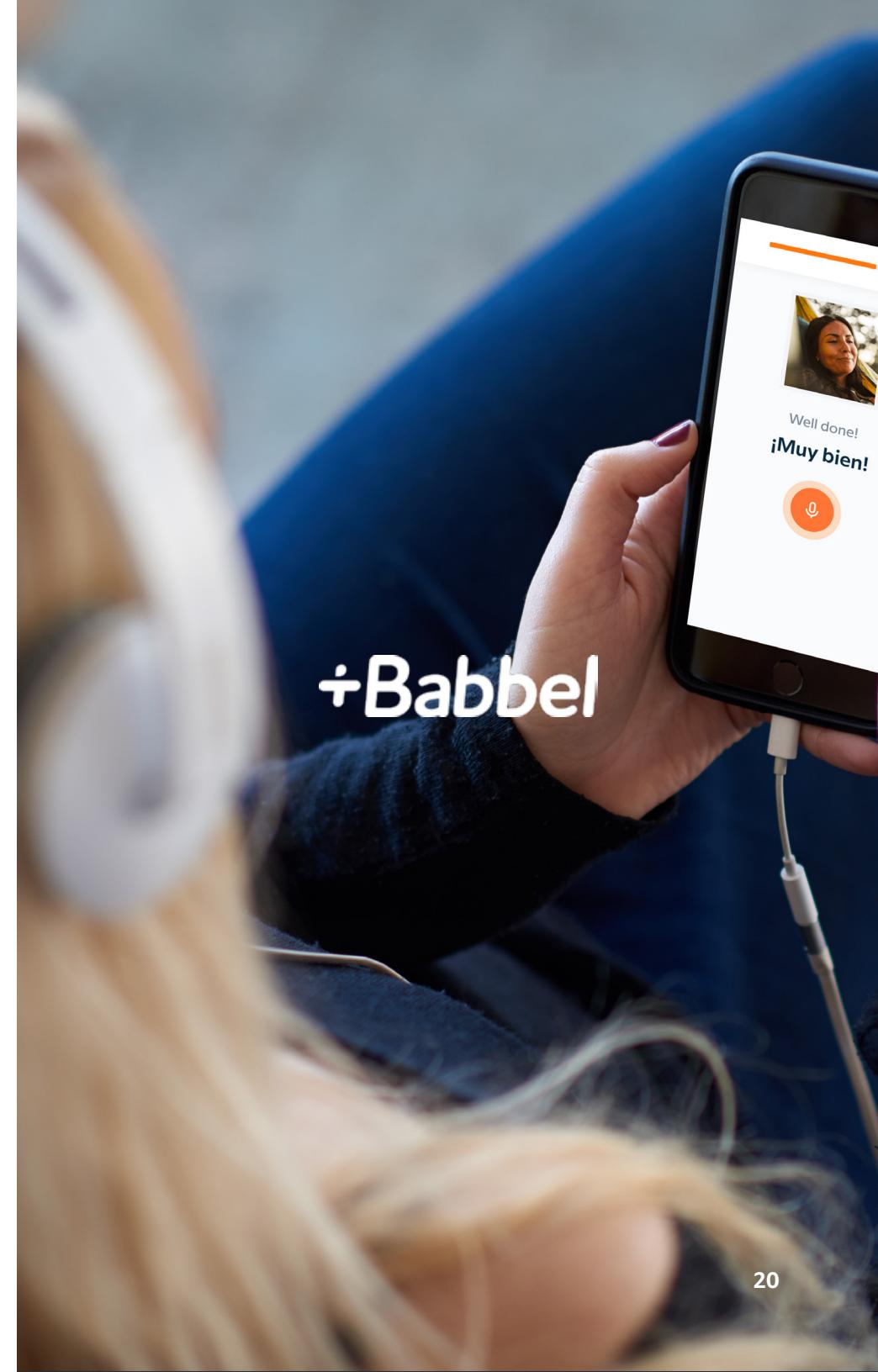
Configuration et mises à jour de la sécurité, configuration du réseau  
Mise en service, gestion de la mise à l'échelle et application de correctifs de serveurs

## TÉMOIGNAGES CLIENTS

# Babbel

Babbel est un écosystème complet d'offres d'apprentissage de langues, dont l'application d'apprentissage de langue la plus vendue au monde. Avec plus de 10 millions d'abonnements vendus et plus de 60 000 cours pour 14 langues, la société est devenue la destination mondiale numéro 1 pour les personnes qui souhaitent apprendre une langue. Babbel fait tourner sa plateforme sur AWS depuis son premier lancement en 2007 et fait souvent partie des premiers clients à adopter les nouvelles offres de services d'AWS. Lorsque Babbel a été confronté à une explosion du nombre de ses apprenants, son architecture initiale ne pouvait plus se mettre à l'échelle au rythme de la croissance de l'activité. En migrant vers Amazon ECS sur AWS Fargate et AWS Lambda, Babbel a gagné en vitesse de publication, a bénéficié d'une capacité de mise à l'échelle rapide et automatisée et a réduit ses coûts grâce à une utilisation plus efficace des charges de travail de calcul, le tout en suivant le rythme de la demande d'une clientèle de plus en plus importante.

[Lire le témoignage complet ›](#)





## Taco Bell

Taco Bell, l'une des marques de restauration rapide les plus emblématiques des États-Unis, compte plus de 7 000 restaurants dans ce pays. Pendant la pandémie de la COVID-19, Taco Bell a dû se réorienter rapidement pour répondre à la demande de livraison des consommateurs. La société exécute la quasi-totalité de son infrastructure sur AWS et utilise le modèle sans serveur sur AWS pour se concentrer moins sur la gestion de serveurs et plus sur la construction de la logique métier et les transformations de données afin de fournir des informations en temps réel sur les menus et les restaurants à ses partenaires de livraison.

[Regarder la vidéo ›](#)

**« Nous avons un menu qui est très complexe et qui doit être partagé sur plusieurs canaux numériques. Le sans serveur est parfaitement adapté à ce modèle. »**

Vadim Parizher,  
Vice-président du département Ingénierie et analytique, Taco Bell

## Coca Cola

Lorsque la COVID-19 est arrivée, les habitudes de consommation ont littéralement changé du jour au lendemain. Coca-Cola a réagi rapidement en proposant une expérience de distribution de boissons sans contact pour accompagner ses distributeurs de boissons Freestyle innovants. Coca-Cola a choisi d'utiliser AWS Lambda, ce qui a permis à son équipe de se concentrer sur l'application plutôt que sur la sécurité, la latence ou l'évolutivité. Avec AWS Lambda, tout ceci est intégré. La nouvelle application a été lancée en 100 jours seulement, et plus de 52 000 machines sont désormais équipées de la fonction sans contact.

[Lire le témoignage complet ›](#)

**« Une faible latence est essentielle pour l'expérience utilisateur, c'est pourquoi nous avons opté pour une solution sans serveur sur AWS. »**

Michael Connor,  
Architecte en chef, Coca-Cola Freestyle Equipment Innovation Center

# Adopter un modèle DevOps moderne

Le DevOps moderne est la combinaison de philosophies, de pratiques et d'outils culturels qui permettent à une organisation de développer des logiciels rapidement et de manière sécurisée, de les mettre en production et de gérer leur disponibilité et leurs performances cibles.

AWS a identifié un ensemble de pratiques communes et largement acceptées qui, lorsqu'elles sont adoptées, constituent un mécanisme permettant de mettre en place une organisation DevOps très performante. Cette approche repose sur une idée simple : l'amélioration continue. Elle s'applique à tous les aspects du cycle de vie DevOps, depuis la planification et l'écriture du code jusqu'au déploiement et à la surveillance. Cette approche, que nous appelons DevOps moderne, est centrée sur le rapprochement des développeurs et des opérations à travers le partage des tâches opérationnelles comme la conformité, l'observabilité, la résilience et l'infrastructure plus tôt dans le processus de développement. Cette approche est également améliorée avec l'intelligence artificielle et le Machine Learning (IA/ML).

## Agilité des développeurs : abstraction, automatisation et standardisation

Les architectures de microservices rendent les équipes agiles et leur permettent de se déplacer plus rapidement, ce qui signifie que vous créez plus de choses qui doivent être publiées. Une aubaine ! Cependant, vous ne pouvez pas fournir de nouvelles fonctions à vos clients rapidement si votre processus de publication ne suit pas le rythme de votre équipe. Les processus de développement traditionnels et les pipelines de publication sont principalement ralenti par les processus manuels et le code personnalisé. Le code personnalisé est en fin de compte un engagement à long terme, car il introduit la possibilité d'erreurs et de maintenance à long terme. De la modification du code au déploiement, en passant par l'élaboration des requêtes et les tests, les étapes manuelles sont le principal frein à la publication. La solution implique l'abstraction, l'automatisation et la standardisation.

Pour accélérer le processus de développement, abstrayez autant de code que possible, en particulier les lignes de code logique non métier, nécessaires pour développer et fournir des applications prêtes pour la production. Pour ce faire, vous pouvez utiliser des cadres et des outils qui réduisent la complexité de l'allocation et de la configuration des ressources. Cela offre aux développeurs la possibilité d'agir rapidement, tout en appliquant les bonnes pratiques en matière de sécurité, de confidentialité, de fiabilité, de performances, d'observabilité et d'extensibilité tout au long du processus de développement. Avec les frameworks de développement, vous avez ainsi la garantie que votre architecture soutiendra la croissance de votre entreprise à long terme.

En définissant votre processus de livraison de logiciels avec des modèles de bonnes pratiques, vous pouvez créer un standard de modélisation et d'allocation de toutes les ressources d'infrastructure dans un environnement cloud. Ces modèles d'« infrastructure en tant que code » aident les équipes à bien démarrer, car ils fournissent l'ensemble de la pile technologique dédiée à une application à travers du code, plutôt que du recours aux processus manuels.

Avec l'automatisation, vous pouvez créer des opérations réitérables qui accélèrent le cycle de vie de la distribution des logiciels. L'automatisation du pipeline de publication via l'intégration et la livraison continues (CI/CD) permet aux équipes de publier du code de qualité supérieure de manière plus rapide et plus fréquente. Les équipes qui pratiquent l'intégration et la livraison continues livrent davantage de code, le font plus rapidement et accélèrent la résolution de problèmes. En fait, selon le rapport 2020 State of DevOps de Puppet<sup>4</sup>, les équipes qui suivent les pratiques CI/CD affichent un taux de pannes 5 fois plus faible, un délai de validation/déploiement 440 fois plus rapide et un taux de déploiement 46 fois plus fréquent. Plus important encore, les équipes qui implémentent les pratiques CI/CD consacrent 44 % de temps en plus à la création de fonctionnalités et de code plutôt qu'à la gestion des processus et des outils.

Les pipelines CI/CD sont devenus le nouvel atelier pour la construction d'applications modernes. Chez Amazon, nous avons commencé à utiliser les pratiques CI/CD pour accélérer le rythme de publication, avec à la clé des résultats spectaculaires : des millions de déploiements par an et une croissance en hausse au fil des années. Pour aider les entreprises à tirer parti de notre expérience, nous avons mis au point une suite d'outils pour développeurs basée sur les outils que nous utilisons en interne, ce qui permet à nos clients de publier plus rapidement du code.

## Quelques définitions

**L'intégration continue (CI)** est une pratique de développement logiciel où les développeurs fusionnent régulièrement leurs modifications de code dans un référentiel central, à la suite de quoi des créations et des tests automatisés sont exécutés. L'intégration continue fait le plus souvent référence à l'étape de création ou d'intégration du processus de publication du logiciel et implique à la fois une composante d'automatisation (par exemple, un CI ou un service de construction) et une composante culturelle (par exemple, apprendre à intégrer fréquemment).

**La livraison continue (CD)** est une pratique de développement logiciel dans laquelle les modifications de code sont automatiquement préparées pour une mise en production. La livraison continue est le prolongement de l'intégration continue à travers le déploiement de toutes les modifications de code dans un environnement de test et/ou un environnement de production, consécutivement à la phase de création.

**Découvrez comment Amazon automatise ses déploiements de manière sécurisée et sans intervention >**

<sup>4</sup>Rapport 2020 State of Devops de Puppet, 2020



## lululemon

Grâce à AWS, lululemon athletica peut déployer des environnements de développement en quelques minutes, contre plusieurs jours auparavant, les automatiser et assurer une intégration et un déploiement continu. L'entreprise canadienne vend des vêtements de yoga et d'autres produits d'habillement dans plus de 350 points de vente à travers le monde. Lululemon exécute ses environnements de développement et de test, y compris son application mobile à venir, sur le Cloud AWS.

**« Un pipeline d'intégration et de déploiement continu doit être automatisé, facile à gérer et détectable. Et c'est exactement ce que nous obtenons avec AWS. Nous bénéficions ainsi d'un niveau de simplicité et de transparence tout à fait inimaginable dans notre environnement sur site précédent. »**

Sam Keen,  
Directeur de l'architecture produits, lululemon

L'entreprise a réduit ses délais de création de comptes de production de deux jours à quelques minutes grâce aux modèles AWS CloudFormation et à AWS CodePipeline. Fortes de cette agilité accrue, les équipes de développement de lululemon peuvent désormais mener des expérimentations pour créer des solutions optimisées, plutôt que de se limiter aux opérations pour lesquelles des ressources sont dédiées.

[Lire le témoignage complet ›](#)



## HyperTrack

HyperTrack est une plateforme cloud en libre-service pour le suivi de la localisation en direct via des applications. Lors de son lancement fin 2015, HyperTrack avait besoin de créer une plateforme capable de se mettre automatiquement à l'échelle pour faire face à sa croissance attendue, sans réduire le temps que ses développeurs consacraient à la création de nouvelles fonctionnalités. HyperTrack a choisi d'utiliser AWS Amplify pour permettre à un cadre de développement mobile et à une architecture sans serveur d'augmenter ou de diminuer automatiquement leur capacité sans l'intervention d'ingénieurs. Résultat : l'entreprise a réduit ses coûts de 30 % par rapport à l'architecture qu'elle utilisait avant de passer au sans serveur. Une grande partie de ces économies est liée au fait que l'entreprise n'a plus besoin de consacrer des ressources opérationnelles à la gestion de serveurs. Chaque semaine, HyperTrack économise 40 heures de travail tout en gérant des millions d'événements.

[Lire le témoignage complet ›](#)

# Adopter un modèle DevOps moderne

## Créer une culture de propriété : gérer moins, innover plus avec le modèle DevOps moderne

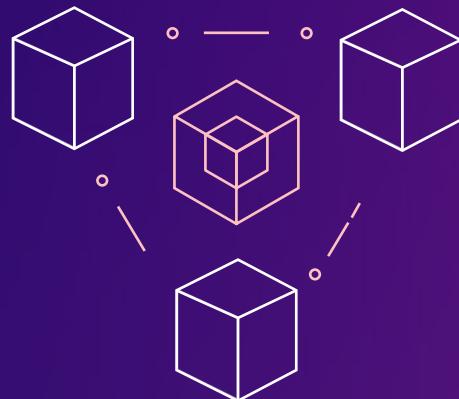
L'innovation provient finalement des gens. C'est donc en permettant à vos collaborateurs de fournir de meilleurs résultats aux clients que vous pouvez commencer à développer des applications modernes. Nous utilisons le concept de « produits, et non projets » pour décrire l'impact sur la structure des équipes. Pour faire simple, cela signifie que les équipes qui développent un produit sont chargées de l'exécuter et d'en assurer la maintenance. Conséquence : les équipes produits sont responsables du développement de l'ensemble du produit, et non simplement d'une partie.

Après plus de 10 années consacrées au développement et à l'exploitation d'Amazon.com, l'application Web hautement scalable, nous connaissons d'expérience l'importance de l'autonomisation des équipes. Lorsque nous octroyons à nos équipes la propriété de l'ensemble du cycle de vie des

applications, y compris la collecte des avis des clients, la planification des feuilles de route et le développement et l'exploitation de l'application en question, elles en deviennent les propriétaires et se sentent habilitées à développer et fournir aux clients de nouveaux produits, services et fonctions. L'autonomie favorise la motivation et la créativité, et développe une culture de prise de risque dans un environnement de confiance.

L'adoption de la culture de la propriété n'est certes pas technique en soi, mais elle demeure l'un des enjeux majeurs du développement des applications modernes. Permettre aux équipes de devenir propriétaires de produit implique de changer la mentalité de votre organisation, la structure de vos équipes et le travail dont elles sont responsables.

Pour la plupart des organisations, l'informatique se divise en deux camps. Elle est considérée comme une arme stratégique concurrentielle ou, plus couramment, comme un centre de coûts nécessaire pour soutenir la croissance de l'entreprise.



## Créer une culture d'innovation

- 1** Le client d'abord : toute innovation doit commencer par un besoin du client et aboutir à la satisfaction de ce dernier. Accordez la priorité absolue aux besoins des clients.
- 2** Embauchez des créateurs et laissez-les créer : levez tous les obstacles qui ralentissent le processus de création et de publication de produits et de fonctionnalités pour les clients. Plus vite vous itérez, plus votre cycle tourne rapidement.
- 3** Soutenez vos créateurs avec un système de pensée : ne faites pas de l'innovation un vœu pieu. Vivez et ressentez l'innovation à tous les niveaux de l'entreprise, des dirigeants au service commercial en passant par les équipes de support.

# Gérer moins, innover plus

Les applications modernes créent des facteurs de différenciation concurrentielle en favorisant l'innovation rapide. En adoptant des services, des pratiques et des stratégies qui mettent l'accent sur la rapidité et l'agilité, vous pouvez transférer des ressources utilisées dans le cadre des activités quotidiennes vers des activités différenciatrices à forte valeur ajoutée pour le client. Expérimitez davantage et transformez plus rapidement vos idées en publications. Encouragez un environnement dans lequel les créateurs consacrent plus de temps à la création et moins de temps à la gestion. Les applications modernes déterminent la manière dont les organisations, y compris Amazon, innovent de façon rapide et agile.

## Pourquoi construire des applications modernes sur AWS ?

<b>Une mise sur le marché plus rapide</b>	En accélérant le cycle de création et de publication et en se déchargeant des frais généraux opérationnels, les développeurs peuvent rapidement créer de nouvelles fonctionnalités. Les processus de test et de publication automatisés réduisent les taux d'erreur. Les produits sont ainsi prêts à être commercialisés plus rapidement.	<b>Voir la preuve :</b> <a href="#">Urbanbase lance ses services 20 fois plus vite avec AWS</a>
<b>Accroître l'innovation</b>	Avec une architecture modulaire, des modifications peuvent être rapidement apportées à n'importe quel composant individuel d'une application, avec un moindre risque pour l'application tout entière. Ainsi, les équipes peuvent expérimenter plus souvent de nouvelles idées.	<b>Voir la preuve :</b> <a href="#">iRobot utilise AWS Lambda et la plateforme AWS IoT pour gérer ses aspirateurs robotisés Roomba</a>
<b>Améliorer la fiabilité</b>	En automatisant les procédures de test et en surveillant chaque étape du cycle de vie du développement, les applications modernes sont fiables au moment de leur déploiement. Tous les problèmes peuvent être évalués et résolus en temps réel.	<b>Voir la preuve :</b> <a href="#">Siemens a réduit de 90 % les alertes du système de contrôle client et de 85 % les coûts d'infrastructure</a>
<b>Amélioration du CTP</b>	Avec un modèle de tarification en fonction de la valeur, les applications modernes réduisent les coûts de surprovisionnement ou des ressources inactives. Grâce au déchargement de la gestion des infrastructures, les coûts de maintenance sont également inférieurs.	<b>Voir la preuve :</b> <a href="#">Économiser jusqu'à 80 % sur la maintenance des applications avec AWS Lambda</a>

## Restructurer la plateforme en services de conteneurs gérés

80 % de toutes les applications conteneurisées exécutées dans le cloud le sont sur AWS. 84 % de toutes les charges de travail Kubernetes dans le cloud s'exécutent sur AWS<sup>5</sup>.

### Ressources

[Atelier Amazon ECS](#)

[Atelier Amazon EKS](#)

[Atelier AWS AppRunner](#)

### Formation recommandée (en classe)

[Exécuter des conteneurs sur Amazon Elastic Kubernetes Service \(Amazon EKS\)](#)

### Formation recommandée (en ligne)

[Amazon Elastic Container Service \(ECS\) Primer](#)

<sup>5</sup> Nucleas, 2019

## Créer des applications modernes avec les technologies et les outils du modèle sans serveur

Gagnez jusqu'à 80 % de temps sur la maintenance et environ 70 % sur le développement en adoptant une stratégie axée sur le sans serveur pour créer des applications modernes<sup>6</sup>.

### Ressources

[Innovator Island – Atelier de développement d'applications web sans serveur.](#)

[Didacticiel vidéo de création d'applications sans serveur.](#)

### Formation recommandée (en classe)

[Développement avancé sur AWS](#)

### Formation recommandée (en ligne)

[Architecting Serverless Solutions](#)

<sup>6</sup> Deloitte 2020

## Adopter un modèle DevOps moderne

60 % des équipes ayant des pratiques DevOps plus évoluées corrigent entièrement les vulnérabilités de sécurité en moins d'une journée<sup>7</sup>.

### Ressources

[La bibliothèque des créateurs Amazon Services DevOps AWS](#)

### Formation recommandée (en classe)

[DevOps Engineering on AWS](#)

### Formation recommandée (en ligne)

[Démarrer avec DevOps sur AWS](#)

<sup>7</sup> Puppet, 2020

**En savoir plus sur la création d'applications modernes sur AWS ›**

**Discuter avec un expert sur la mise en œuvre des bonnes pratiques de développement d'applications modernes ›**

**Contacter un partenaire AWS pour accélérer vos projets de modernisation ›**

