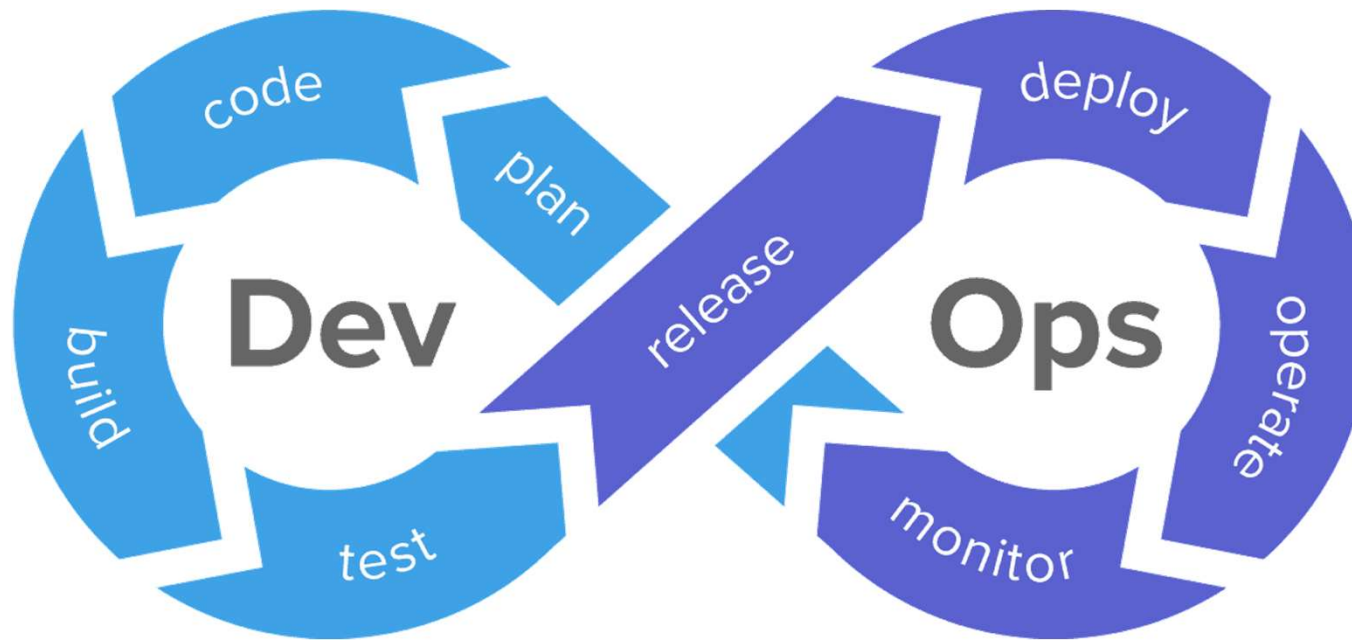


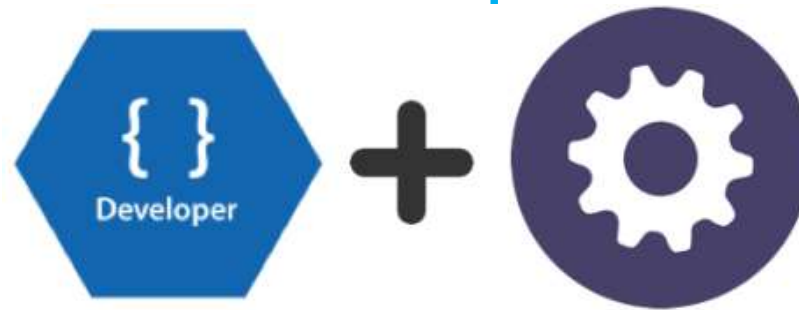
# DevOps



**Dev + Ops = DevOps !**

# Qu'est ce que c'est ?

**Méthodologie pour une évolution rapide et continue des services informatiques**

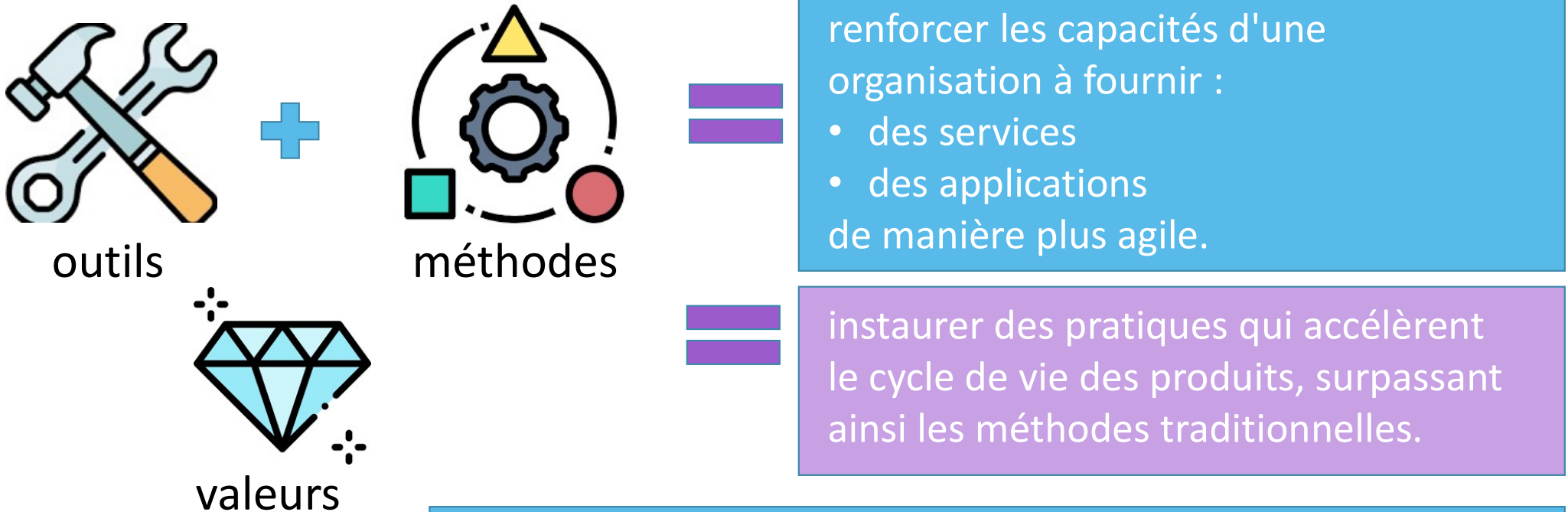


**Fusion des termes "développement" et "opérations".**

Cette approche méthodologique vise à intégrer le développement, le déploiement, l'intégration et l'assurance qualité dans un processus continu et unifié.

**Evolution naturelle de la livraison continue et des méthodologies Agile.**

# Comment faire ?



**Objectif ultime** : améliorer la qualité des services offerts aux clients tout en renforçant la compétitivité de l'entreprise.

# Une démarche itérative

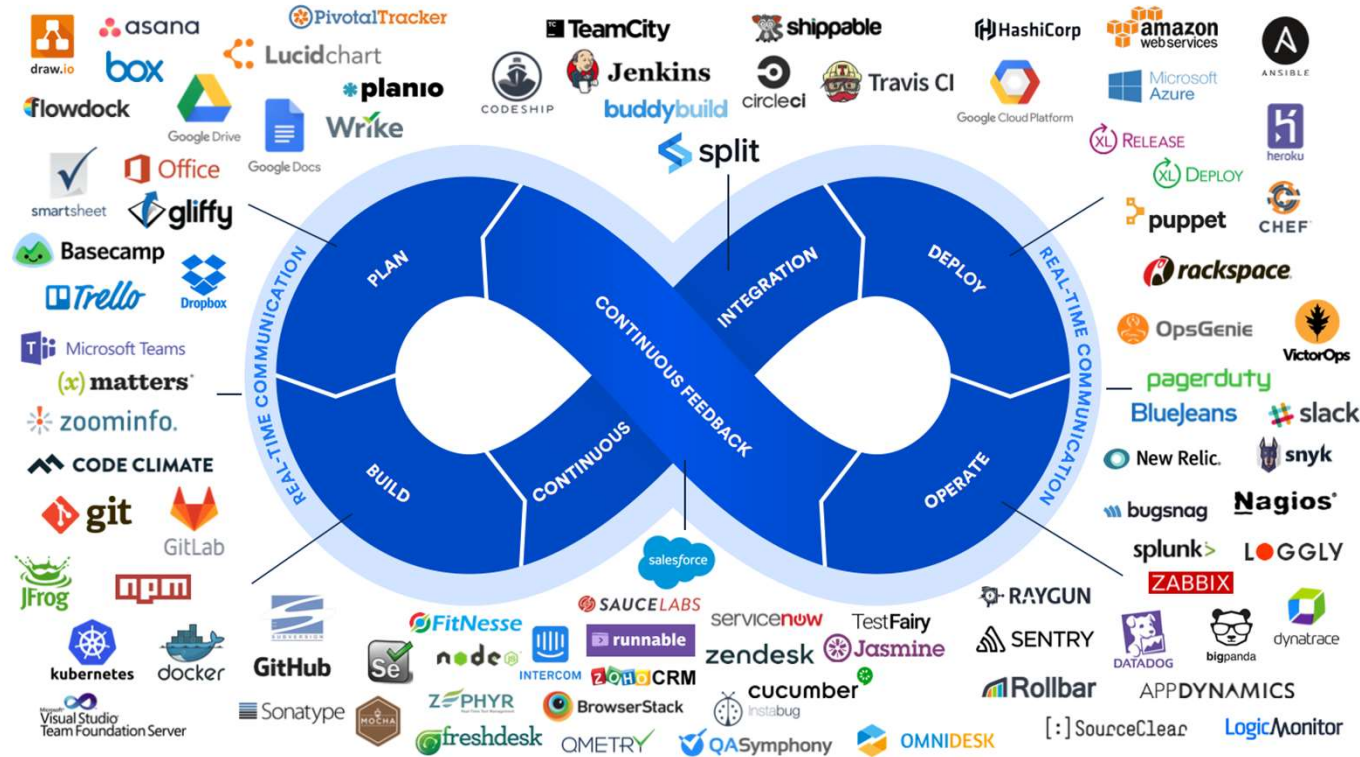
**Adopter une démarche DevOps = construire une démarche**

Établir des objectifs,  
des moyens, des  
manières de faire.

appliquer, déployer,  
tester, utiliser et  
contrôler les impacts  
positifs ou négatifs.

revoir les objectifs,  
moyens et manière de  
faire...

La démarche est donc itérative et évolutive...comme le ou les résultats produits.



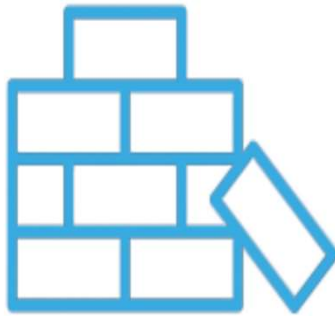
# Comment faire ?

pour une collaboration rapprochée et une automatisation efficace des processus de développement et de production

## Pas DevOps



Equipe Dev



Equipe prod

2 équipes isolées

## DevOps



1 équipe unique composée principalement d'ingénieurs qui participent à toutes les étapes du cycle de vie du développement, y compris les tests.

## Comment faire ?

Il s'agit d'une collaboration étroite entre les équipes d'assurance qualité, de sécurité et de développement logiciel

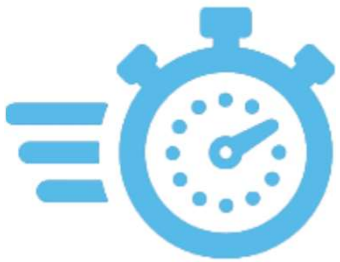
Aujourd'hui, ces équipes s'appuient sur des processus d'automatisation, utilisant des ensembles d'outils spécifiques.

Ces éléments sont essentiels pour faire évoluer rapidement les applications tout en maintenant leur fiabilité.

En conséquence, les ingénieurs sont capables d'accomplir leurs tâches de manière autonome.



# Avantages de DevOps



## Rapidité

- Accélérer le rythme des innovations en faveur de sa clientèle.
- Meilleure adaptation sur le marché (efficacité et croissance)
- les équipes de développements seront en mesure d'atteindre les objectifs. (la livraison continue, les [microservices](#) , une mise en place plus rapide des mises à jour..)

- Augmenter la fréquence des publications, tout en favorisant l'innovation et l'optimisation des produits.
- Multiplier les fonctionnalités et s'attarder sur la correction des bugs
- Gain en compétitivité, réponse favorable aux besoins des clients.
- Livraison et IC permettent l'automatisation du flux de publication d'une application ou d'un logiciel, mais également de sa création et de son déploiement.



## Livraison rapide



## Avantages de DevOps (suite)



### Fiabilité

- Proposer une expérience positive à ses clients,
  - fiabilité des mises à jour, tout en veillant à n'importe quel changement sur l'infrastructure.
- Livraison et IC sont indispensables pour vérifier que n'importe quelle modification soit sûre et fonctionnelle.
  - Mesure des performances en temps réel, avec des méthodes de journalisation et de supervision.



### Evolutivité

- Impact sur la gestion et les opérations des processus de configuration de logicielles et d'infrastructures à grande échelle seront directement impactées
- Gestion plus efficace et surtout moins risquée grâce à la cohérence et l'automatisation (exemple sur l'infrastructure)

## Avantages de DevOps (fin)



### Collaboration

- Equipes plus efficaces en se basant sur des principes de prise de responsabilité.
- meilleure combinaison des flux de travail.
- gain de temps et perte d'efficacités limitées (délais de transfert d'une équipe à l'autre, écriture du code avec une prise en compte immédiate de l'environnement dans lequel il sera exécuté.)

- Contrôle total, qui permet de préserver la conformité
- Contrôles plus rigoureux
- Définir soi même la conformité à suivre à grande échelle.



### Sécurité

# Les méthodes DevOps



- basée sur des cycles plus courts
- très bonne réactivité dans l'évolution des besoins.
- les exigences sont définies sous la forme de user stories grâce aux retours clients de façon continue, mais également avec l'organisation de réunions quotidiennes.

- développée par Toyota.
- obtenir une meilleure efficacité dans les usines de montage.
- sur un suivi des travaux en cours en se basant sur l'usage d'un tableau de Kanban.



**KANBAN**



- chaque membre d'une équipe a des tâches particulières pour collaborer de manière plus efficace.
- accélération des projets de configuration, tout en gagnant en assurance qualité.

# Chaines d'outils DevOps

Il existe toute une gamme d'outils qui peuvent être utilisés pour faciliter :

- le déploiement continu,
- l'automatisation des processus,
- la gestion de la configuration,
- la surveillance
- ....

Le choix des outils dépend :

- des besoins spécifiques de l'organisation,
- de la taille du projet
- des compétences de l'équipe.

De plus, en raison de l'évolution rapide du domaine DevOps, de nouveaux outils sont constamment introduits, tandis que d'autres peuvent devenir obsolètes.

**Il est donc crucial de rester à jour avec les tendances et les évolutions du domaine.**

# Gestion de versions et de code source

Essentiels pour assurer la collaboration efficace des équipes de développement, le suivi des modifications du code source, la gestion des branches et la mise en œuvre de pipelines d'intégration continue et de déploiement continu



Contrairement à Git et Mercurial qui stockent les versions de manière distribuée sur chaque poste de travail, SVN utilise un dépôt central où toutes les versions sont stockées.

Ils permettent aux équipes de développer, de tester et de déployer des applications de manière agile et coordonnée, contribuant ainsi à l'amélioration de la qualité du logiciel et à la rapidité des livraisons.

# Outils de build et d'intégration continue



Jenkins



Travis CI



GitLab CI/CD



TeamCity

Ce sont tous des outils d'intégration continue et de déploiement continu utilisés en DevOps pour automatiser les processus de développement logiciel, accélérer les livraisons, et améliorer la qualité des applications.

## Gestion des configurations et des infrastructures



Ansible



Ce sont tous des outils utilisés en DevOps pour automatiser la gestion des configurations et des infrastructures.



## Gestion des configurations et des infrastructures

- 1. Automatisation des tâches répétitives** : Les tâches de configuration et de déploiement peuvent être fastidieuses et sujettes aux erreurs humaines lorsqu'elles sont effectuées manuellement. L'automatisation permet de scripter ces tâches pour les exécuter de manière cohérente et reproductible, réduisant ainsi les risques d'erreur et accélérant les déploiements.
- 2. Gestion des configurations** : Automatiser la gestion des configurations signifie définir l'état désiré des systèmes informatiques, y compris les logiciels installés, les paramètres de configuration, les autorisations d'accès, etc., et utiliser des outils pour garantir que les systèmes restent conformes à cet état.
- 3. Provisionnement d'infrastructure** : Automatiser le provisionnement d'infrastructure implique la création et la gestion automatisée des ressources matérielles et cloud nécessaires pour exécuter les applications, comme les serveurs virtuels, les conteneurs, les bases de données, les réseaux, etc.
- 4. Orchestration des déploiements** : L'automatisation permet également d'orchestrer les déploiements d'applications et de services sur l'infrastructure, en définissant des pipelines de déploiement automatisés qui gèrent les étapes telles que la compilation du code, les tests automatisés, le déploiement sur les serveurs, etc.
- 5. Suivi et mise à jour continus** : En automatisant la gestion des configurations et des infrastructures, les équipes DevOps peuvent mettre en place des processus de suivi et de mise à jour continus pour garantir que les systèmes restent à jour, sécurisés et conformes aux normes de l'entreprise.

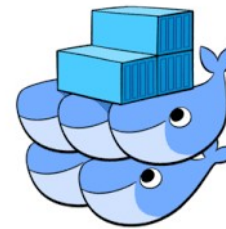
# Conteneurisation et orchestration



Docker est une plateforme de conteneurisation



Kubernetes est un système d'orchestration de conteneurs open-source



Docker Swarm est un outil d'orchestration intégré à Docker



Amazon ECS est un service d'orchestration de conteneurs proposé par AWS

Chacun de ces outils peut être utilisé dans une approche DevOps pour automatiser le déploiement et la gestion des applications conteneurisées, mais ils offrent des fonctionnalités différentes pour répondre aux besoins spécifiques des entreprises

# Conteneurisation et orchestration

## 1. Docker :

1. Docker est une plateforme de conteneurisation qui permet d'emballer des applications et leurs dépendances dans des conteneurs légers et portables.
2. Les conteneurs Docker offrent une isolation des ressources et une portabilité des applications, ce qui facilite le déploiement cohérent des applications sur différents environnements.
3. Docker fournit des outils pour créer, gérer et exécuter des conteneurs, ainsi qu'une bibliothèque de conteneurs prêts à l'emploi appelée Docker Hub.

## 2. Kubernetes :

1. Kubernetes est un système open-source d'orchestration de conteneurs, développé par Google, qui automatise le déploiement, la mise à l'échelle et la gestion des applications conteneurisées.
2. Kubernetes offre des fonctionnalités avancées telles que la répartition de la charge, la mise en réseau, la surveillance, la gestion des mises à jour et la haute disponibilité pour les applications conteneurisées.
3. Il permet également de définir des ressources de calcul et de stockage pour les conteneurs, et de les orchestrer de manière à maintenir l'état désiré de l'application.

## 3. Docker Swarm :

1. Docker Swarm est un outil d'orchestration de conteneurs intégré à Docker Engine, qui permet de gérer un cluster de nœuds Docker et de déployer des services conteneurisés.
2. Il offre une solution simple et native pour orchestrer des conteneurs Docker sur plusieurs hôtes, en fournissant des fonctionnalités telles que la mise à l'échelle automatique, l'équilibrage de charge et la gestion des services.

## 4. Amazon ECS (Elastic Container Service) :

1. Amazon ECS est un service de conteneurisation et d'orchestration de conteneurs proposé par Amazon Web Services (AWS).
2. Il permet de déployer et de gérer facilement des conteneurs Docker sur l'infrastructure cloud d'AWS, en fournissant des fonctionnalités telles que le dimensionnement automatique, la haute disponibilité, la surveillance et l'intégration avec d'autres services AWS.

# Gestion des déploiements



## Ansible

Ansible est un outil polyvalent de gestion de la configuration



Capistrano est spécifiquement conçu pour le déploiement d'applications web



DeployBot est un service cloud pour le déploiement continu



AWS CodeDeploy est un service d'AWS spécifiquement pour le déploiement d'applications sur sa plate-forme cloud.

La gestion des déploiements en DevOps joue un rôle crucial dans le processus de développement logiciel en automatisant et en rationalisant le déploiement d'applications et de services sur des infrastructures informatiques.

# Gestion des déploiements

## 1. Ansible :

1. Ansible est une plateforme open-source de gestion de la configuration et d'automatisation des déploiements.
2. Il utilise des playbooks YAML pour décrire les tâches à exécuter sur les serveurs cibles, permettant ainsi de déployer et de configurer des applications de manière cohérente et reproductible.
3. Ansible peut être utilisé pour le déploiement sur site ou dans le cloud, et il prend en charge une large gamme de systèmes d'exploitation et de technologies.

## 2. Capistrano :

1. Capistrano est un outil open-source spécifiquement conçu pour le déploiement d'applications web, principalement en Ruby on Rails.
2. Il permet de déployer des applications sur des serveurs distants via SSH, en automatisant des tâches telles que la copie de fichiers, l'exécution de commandes, et la gestion des versions.
3. Capistrano prend en charge des fonctionnalités telles que le rollback automatique en cas d'échec du déploiement et la gestion des dépendances.

## 3. DeployBot :

1. DeployBot est un service cloud conçu pour simplifier et automatiser le déploiement continu d'applications web.
2. Il offre une interface conviviale pour configurer des pipelines de déploiement, intégrant des outils de gestion de versions comme Git et des services de déploiement comme FTP, SSH, et les services cloud.
3. DeployBot prend en charge des fonctionnalités telles que la gestion des branches, les notifications, et les intégrations avec d'autres services DevOps.

## 4. AWS CodeDeploy :

1. AWS CodeDeploy est un service proposé par Amazon Web Services (AWS) pour automatiser le déploiement d'applications sur des instances EC2 ou des services gérés par AWS, comme ECS et Lambda.
2. Il prend en charge le déploiement de code sur une large gamme de plates-formes, y compris les applications web, les applications mobiles, et les microservices.
3. AWS CodeDeploy permet de définir des stratégies de déploiement flexibles, telles que le déploiement en place, le déploiement blue-green, et le déploiement canary.

# Avantages dans la Gestion des déploiements

1. **Automatisation des déploiements** : La gestion des déploiements vise à automatiser autant que possible le processus de déploiement d'applications, de la compilation du code à la mise en production. Cela permet d'accélérer les cycles de livraison et de réduire les risques d'erreur humaine.
2. **Fiabilité accrue** : En automatisant les déploiements, les équipes DevOps peuvent garantir une cohérence et une fiabilité accrues dans le processus de déploiement, réduisant ainsi les erreurs et les temps d'arrêt imprévus.
3. **Reproductibilité** : La gestion des déploiements permet de définir et de documenter les processus de déploiement de manière à ce qu'ils puissent être répétés de manière cohérente dans différents environnements, garantissant ainsi une reproductibilité des déploiements.
4. **Rapidité et agilité** : En automatisant les déploiements, les équipes DevOps peuvent accélérer le cycle de développement en mettant en œuvre des déploiements continus ou même des déploiements en continu, permettant aux nouvelles fonctionnalités d'être mises à disposition des utilisateurs plus rapidement.
5. **Déploiements sans interruption de service** : La gestion des déploiements permet de mettre en place des stratégies de déploiement telles que le déploiement progressif (ou "blue-green deployment") ou le déploiement canary, qui permettent de minimiser les interruptions de service lors des mises à jour logicielles.
6. **Rollbacks rapides** : En cas de problème lors d'un déploiement, la gestion des déploiements permet de réaliser rapidement des rollbacks vers des versions précédentes stables, réduisant ainsi l'impact des erreurs sur les utilisateurs finaux.
7. **Suivi et surveillance** : Les outils de gestion des déploiements en DevOps offrent souvent des fonctionnalités de suivi et de surveillance en temps réel, permettant aux équipes de suivre les performances des déploiements et de détecter rapidement les problèmes potentiels.

# Surveillance et logging



ELK Stack (Elasticsearch, Logstash, Kibana)

suite d'outils pour la gestion des logs et la surveillance



**Grafana**

plateforme de visualisation des données souvent utilisée avec d'autres outils de surveillance



**Prometheus**

système de surveillance spécialisé dans les métriques

**splunk**>

plateforme complète de gestion des données offrant des fonctionnalités avancées de recherche et d'analyse

La supervision correspond à la dernière phase. Elle est également indispensable dans le but de procéder à l'identification des problèmes pouvant avoir une influence néfaste sur la version logicielle en production. Il faut alors collecter les informations en vue de prendre des mesures correctives.



# Surveillance et logging

## 1. ELK Stack :

1. ELK est un acronyme pour Elasticsearch, Logstash et Kibana, trois outils souvent utilisés ensemble pour la gestion des logs et la surveillance.
2. Elasticsearch est une base de données de recherche et d'analyse distribuée, utilisée pour stocker et interroger les logs.
3. Logstash est un outil de collecte, d'analyse et de transformation des logs, permettant de les envoyer vers Elasticsearch.
4. Kibana est une interface de visualisation des données, offrant des tableaux de bord interactifs et des graphiques pour analyser les logs stockés dans Elasticsearch.

## 2. Prometheus :

1. Prometheus est un système open-source de surveillance et d'alerting conçu pour les environnements cloud et conteneurisés.
2. Il collecte les métriques de performances à partir des cibles configurées, stocke ces métriques dans une base de données temporelle et fournit des requêtes et des graphiques pour l'analyse.
3. Prometheus prend en charge le scraping des métriques HTTP, le monitoring des services Kubernetes, et l'exportation de données via des agents appelés "exporters".

## 3. Grafana :

1. Grafana est une plateforme open-source de visualisation des données, souvent utilisée en conjonction avec des systèmes de surveillance comme Prometheus, Elasticsearch, InfluxDB, etc.
2. Il offre une interface graphique intuitive pour créer des tableaux de bord et des graphiques interactifs à partir de données collectées, ce qui permet une analyse en temps réel et une prise de décision basée sur les données.

## 4. Splunk :

1. Splunk est une plateforme logicielle de surveillance, d'analyse et de gestion des données, proposant des fonctionnalités avancées de recherche, d'indexation, et de reporting pour les logs et autres données.
2. Il permet de collecter, d'indexer et d'analyser des données provenant de diverses sources, y compris les logs, les métriques, les événements de sécurité, etc.
3. Splunk offre des fonctionnalités de recherche avancées, de corrélation d'événements, de génération de rapports personnalisés et de visualisation des données.

# Collaboration et communication



Mattermost



# Collaboration et communication

## 1. Microsoft Teams :

- Slack est une plateforme de messagerie instantanée en équipe qui permet la communication en temps réel.
- Il offre la possibilité de créer des canaux pour organiser les discussions par projet, équipe ou sujet, ce qui facilite la collaboration au sein d'équipes multifonctionnelles.
- Slack prend en charge une large gamme d'intégrations avec des outils de développement, de suivi de problèmes, d'intégration continue et d'autres outils DevOps, ce qui en fait une plateforme polyvalente pour la coordination des activités de développement et d'exploitation.

## 2. Microsoft Teams :

- Microsoft Teams est une plateforme de collaboration intégrée dans l'écosystème Microsoft 365.
- Il offre des fonctionnalités de messagerie instantanée, de visioconférence, de partage de fichiers et d'intégrations avec d'autres outils Microsoft, tels que SharePoint, OneDrive et Azure DevOps.
- Teams est particulièrement adapté aux organisations déjà intégrées à l'écosystème Microsoft, car il permet une collaboration transparente avec les outils et services Microsoft existants.

## 3. Mattermost :

- Mattermost est une alternative open-source à Slack et Microsoft Teams, offrant des fonctionnalités similaires de messagerie instantanée en équipe et de collaboration.
- Il est conçu pour les organisations souhaitant héberger leur propre solution de communication et de collaboration, offrant ainsi un contrôle total sur les données et la sécurité.
- Mattermost est hautement personnalisable et extensible, permettant aux organisations de l'adapter à leurs besoins spécifiques en matière de DevOps et de l'intégrer à leurs outils et workflows existants.

# Tests automatisés

La réalisation des tests continus est absolument indispensable pour s'assurer de la fiabilité du code. Ces derniers peuvent d'ailleurs être exécutés de façon automatisée ou manuelle



JUnit est principalement utilisé pour les tests unitaires en Java



Selenium est utilisé pour l'automatisation des tests fonctionnels des applications web



TestNG offre des fonctionnalités avancées pour les tests unitaires et fonctionnels, y compris la parallélisation et la gestion des suites de tests.

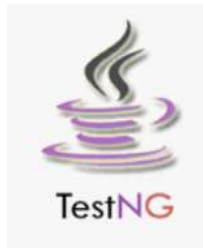


JMeter est utilisé pour les tests de performance des applications web

# Tests automatisés



framework de test unitaire pour Java pour tester des composants individuels ou des classes de code Java.



framework de test plus avancées que JUnit.

- divers types de tests : tests unitaires, les tests de groupe, les tests de paramètres, les tests parallèles, etc..

- offre des fonctionnalités telles que la dépendance des tests, la génération de rapports détaillés, la parallélisation des tests, la gestion des suites de tests, etc.



Application open-source principalement utilisée pour effectuer des tests de performance et de charge sur des applications web.

- simuler un grand nombre d'utilisateurs accédant simultanément à une application web afin de mesurer les performances, la stabilité et la fiabilité du système sous charge.
- Peut être utilisé pour tester d'autres protocoles, comme FTP, JDBC, LDAP, etc.



Il permet de simuler les actions d'un utilisateur sur un navigateur web, comme le clic sur des boutons, la saisie de texte, la navigation entre les pages, etc.

# Gestion des tickets et des projets

Recenser les problèmes et d'appliquer un suivi dans leur gestion



## Cloud et services d'infrastructure



Microsoft Azure



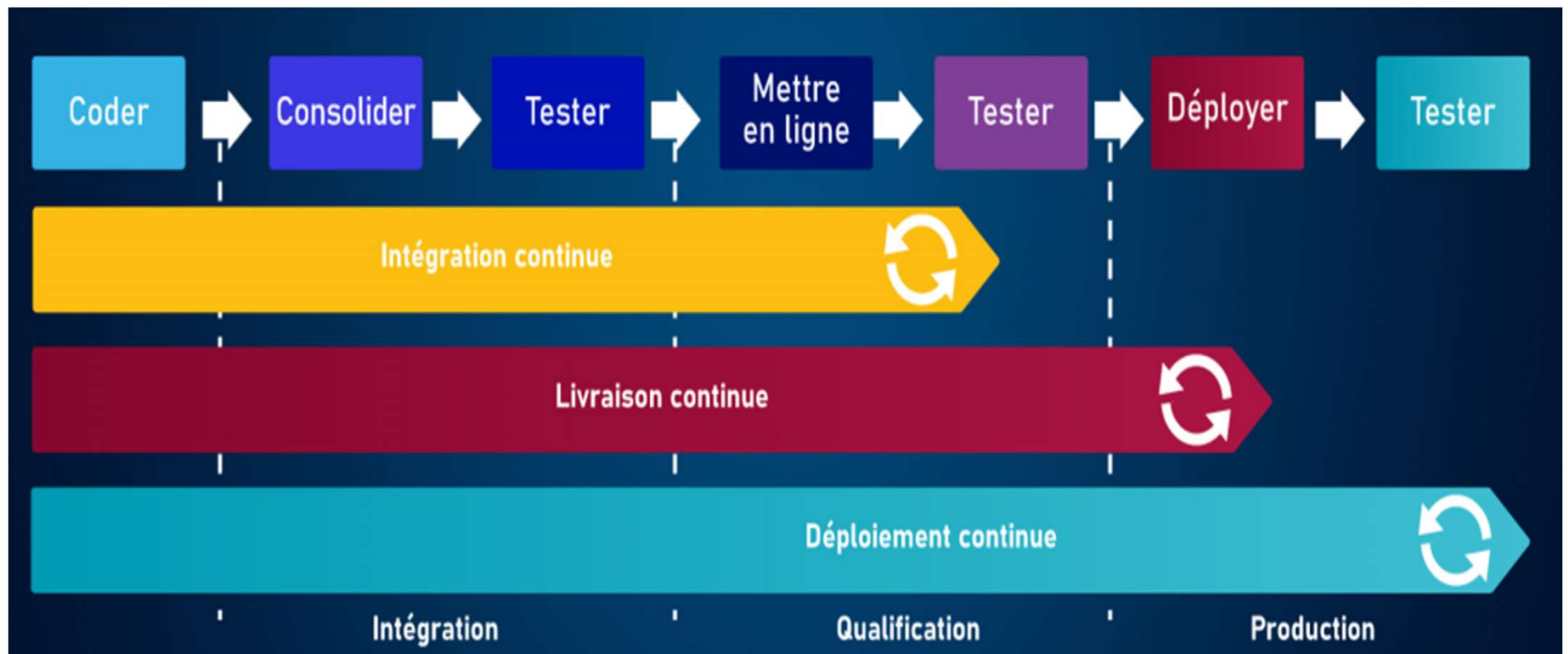
Google Cloud Platform

Impossible de ne pas parler des clouds providers comme Google Cloud Platform ou Azure , AWS.  
Ce sont des plate-formes Software as a Service ou SaaS, livrant une chaîne complète DevOps.

**incontournable pour la configuration et le déploiement des logiciels**

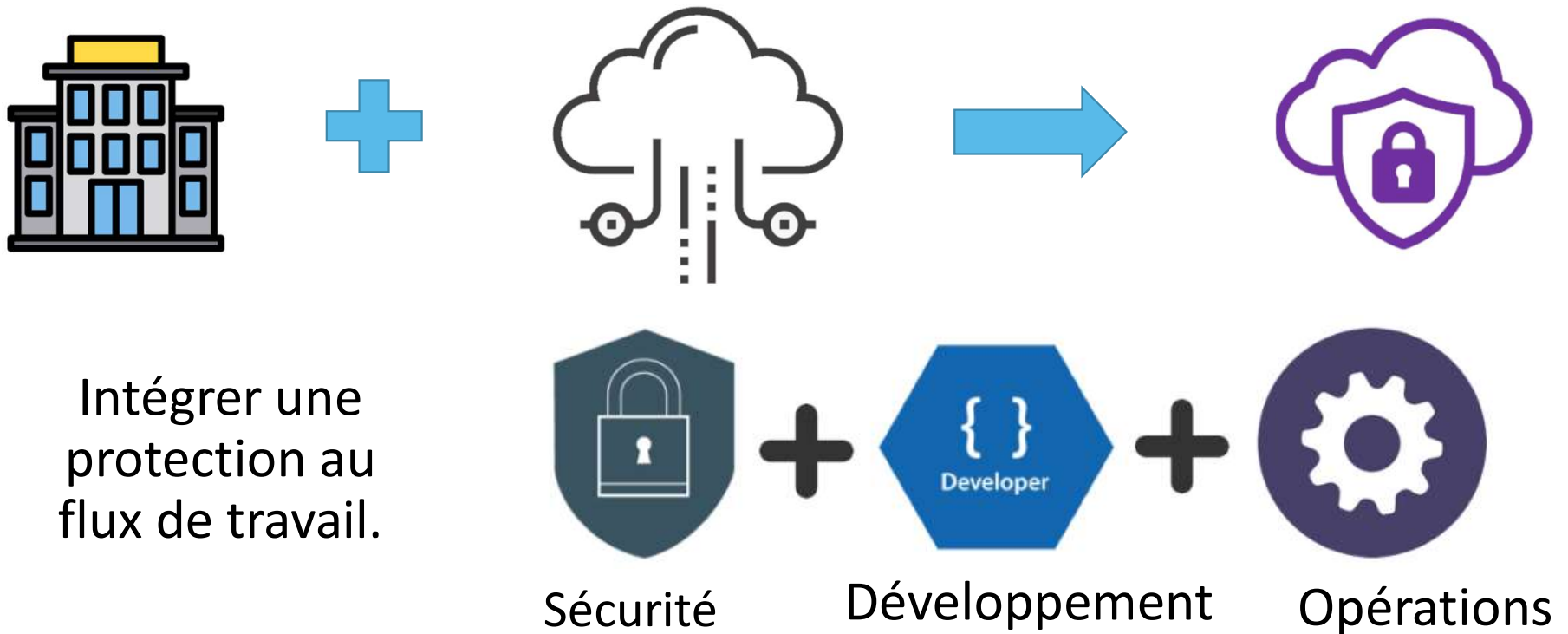


# Pratiques DevOps

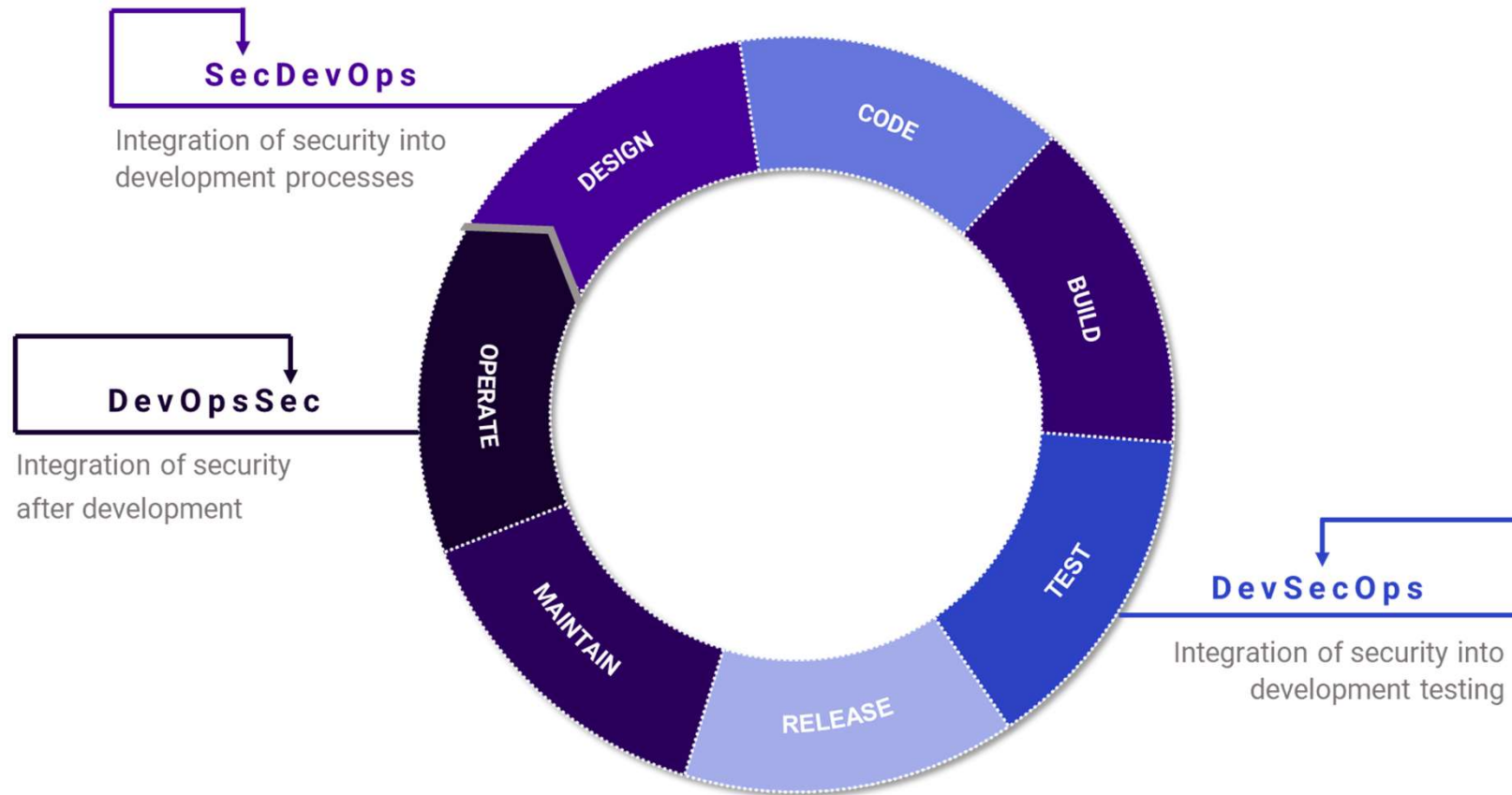


# Futur de DevOps : Sécurité

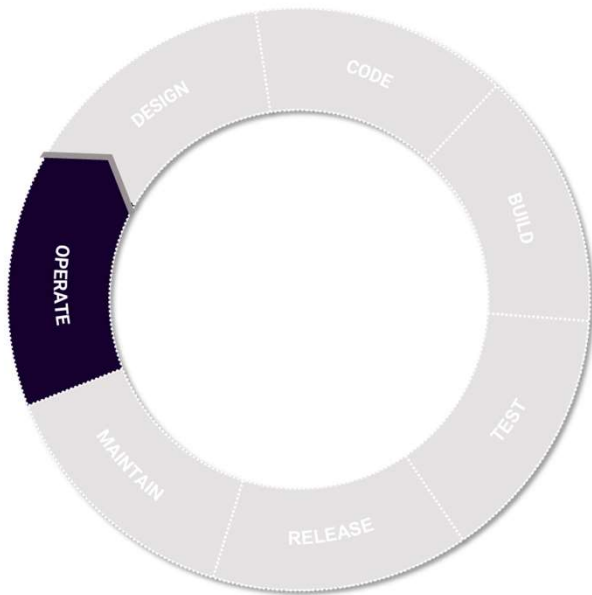
Efficacité du DevOps ne peut être remise en question =  
accélération des flux de développement + amélioration continue de la fiabilité des services.



# Sécurité et DevOps



# DevOpsSec



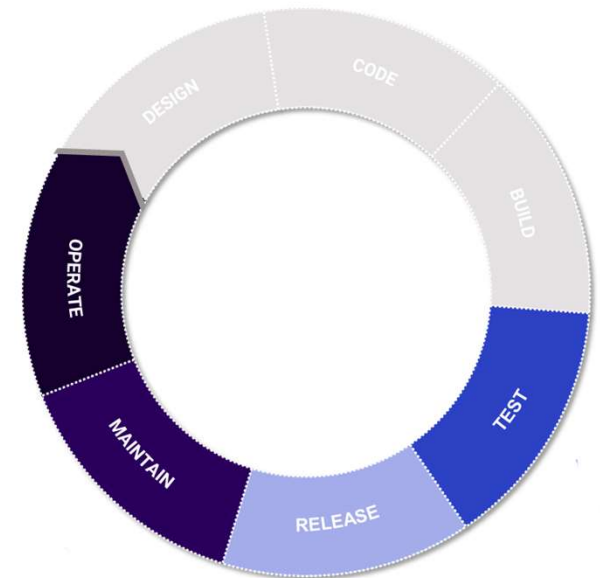
- Place la sécurité après son développement seulement une fois que le code est déjà en production.
- 90% des entreprises commencent les tests de sécurité après que le code soit en production = plus difficile et plus coûteux de corriger les vulnérabilités en production
- Les équipes de sécurité sont séparées des équipes de développement et ont peu d'interactions, sauf dans les situations critiques. Des analyses de vulnérabilité sont régulièrement programmées, car vous savez que la sécurité ne se limite pas à la conformité. Les résultats sont gérés dans des flux de travail distincts et sont probablement gérés manuellement.

C'est comme construire une maison, y faire emménager ses occupants, puis vérifier qu'elle a été construite correctement par la suite. Et quand vient le moment de la réparer, cela prend plus de temps et n'est jamais tout à fait correct...

# DevSecOps

Le DevSecOps met l'accent sur l'intégration d'outils de test dans le processus de développement continu. Toutefois, cette focalisation sur les outils peut entraîner une perte de contexte projet (dilemme entre vitesse et sécurité). Avec 57 % des équipes de développement livrant du code chaque semaine ou encore plus fréquemment, ralentir le processus devient une décision difficile à prendre

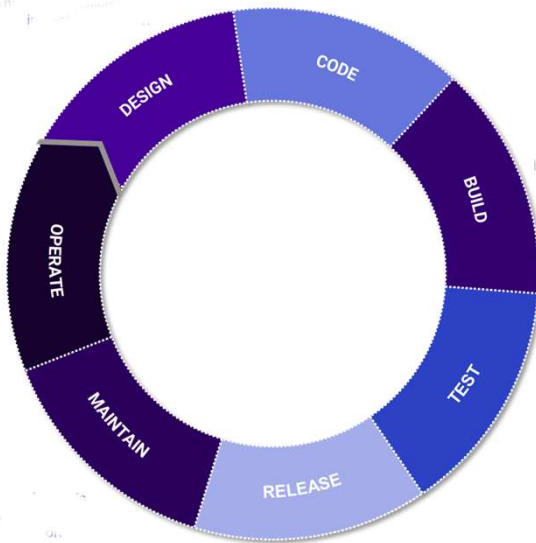
- Les équipes de sécurité et de développement ont une visibilité sur leurs processus respectifs.
- Les tests de sécurité ne se limitent pas à l'analyse statique ou dynamique, mais utilisent les tests interactifs ou au runtime (la sécurité des conteneurs, la sécurité des configurations dans le nuage).



Avec la construction d'une maison, DevSecOps revient à dresser une liste de choses à réparer au fur et à mesure de la construction, mais sans disposer de suffisamment d'informations pour savoir dans quel ordre les réparer, ou même s'il est nécessaire de les réparer toutes.

# SecDevOps

## Sécurité dans les processus de développement



La sécurité peut être intégrée à chaque étape, puis soutenue par les outils au lieu d'être freinée par eux.

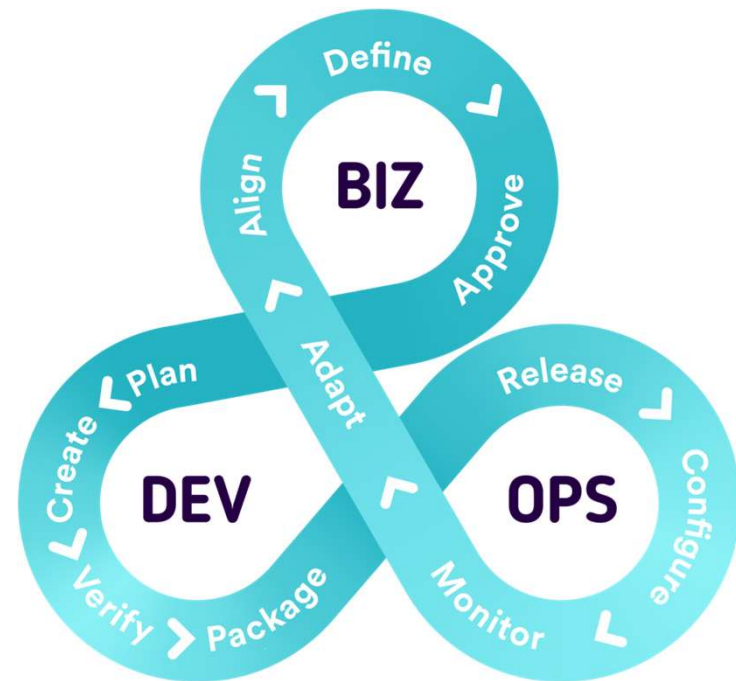
- Les équipes de sécurité et de développement ont une cadence régulière et intégrée (la sécurité peut même assister aux revues de code !).
- Les projets se voient attribuer des politiques spécifiques basées sur les détails du projet.
- Les résultats sont automatiquement classés par ordre de priorité et réintégrés dans le flux de travail du développement en tant qu'éléments de travail.

SecDevOps est la façon dont nous construisons les maisons : les processus sont définis à l'avance, pour s'assurer que les inspections sont menées au bon moment, et que les éléments critiques sont corrigés avant d'avancer, tandis que d'autres peuvent être correctement priorisés et corrigés plus tard ou autorisés à devenir une partie de la dette de sécurité.

# Futur de DevOps : BizDevOps

Supprimer les frontières entre les équipes business, les équipes de productions et les développeurs.

**Même objectif :**  
développement des  
produits avec une  
orientation utilisateur.





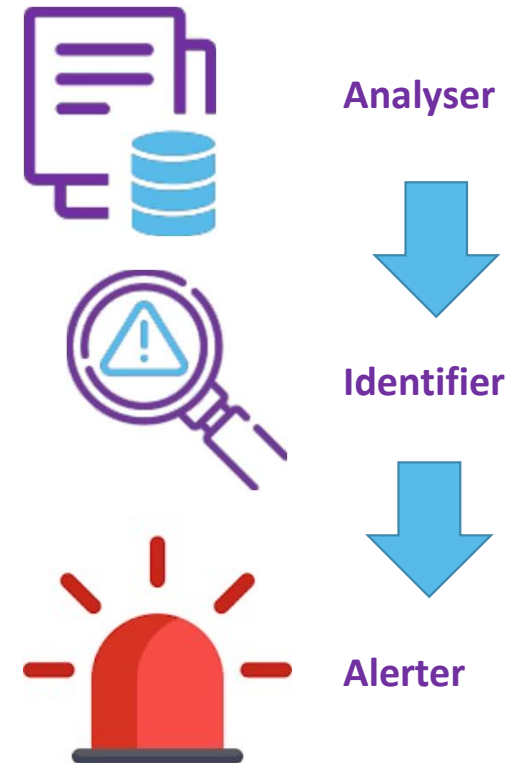
# DevOps, IA et Machine Learning

## 1 Pour exploiter les données de test

- Mettre en évidence des patterns,
- Identifier des problèmes éventuels et des bugs
- Alerter les équipes de développements afin de mener une enquête plus approfondie.

## 2 Pour examiner les données de sécurité

- déceler n'importe quelle menace potentielle.
- principe d'automatisation des alertes.



# Conclusion

Accélérer la production des produits et systèmes informatiques de manière fiable et efficace.



Optimisation des ressources via des outils et des méthodologies



favoriser la collaboration entre les équipes de développement et d'opérations



Amélioration de la qualité : systèmes plus robustes, évolutifs et fiables, en intégrant les retours d'expérience clients.