

DevOps



Qu'est-ce que DevOps ?

- Le nom « **DevOps** » vient de la contraction du mot « **development** » (développement) et « **operation** » (exploitation).
- DevOps est une culture qui vise à améliorer **la communication entre les développeurs et l'équipe d'exploitation**.
- C'est aussi un ensemble de bonnes pratiques pour automatiser les différentes phases du projet (test, monitoring, déploiement, ..)
- => Réduction du Time To Market (TTM)

Le Problème : Le "Mur de la Confusion"

- **Constat** : Une rupture totale de communication entre **Dev** (Développement) et **Ops** (Opérations).
- **Conséquence** : Les obstacles techniques ne sont découverts qu'au moment de la livraison du code.
- **Impact métier** : Retards de mise sur le marché, instabilité logicielle et frustration des équipes.

La Solution DevOps

- DevOps est **une culture et ensemble de pratiques** qui vise à unifier le développement logiciel (Dev) et les opérations informatiques (Ops).
- Collaboration entre les équipes Dev et Ops
- Automatisation des processus
- Livraison continue et rapide
- Amélioration de la qualité logicielle
- Feedback constant et itératif

DevOps = **culture + pratiques + outils**

Pourquoi adopter DevOps ?

⚡ Rapidité

Déploiements plus fréquents et rapides

🕒 Fiabilité

Qualité constante des mises à jour

📈 Scalabilité

Gestion efficace de l'infrastructure

🤝 Collaboration

Meilleure communication entre équipes

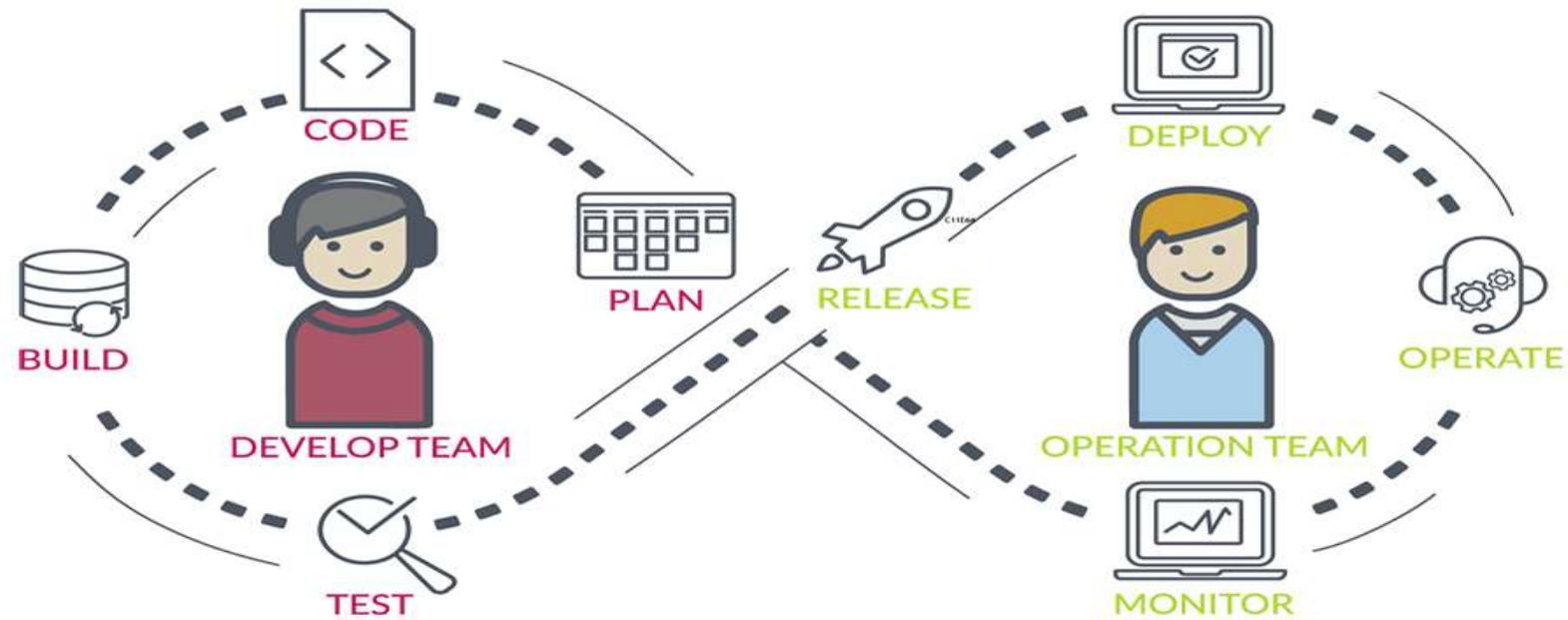
🔒 Sécurité

Intégration de la sécurité dès le début

💡 Innovation

Plus de temps pour l'innovation

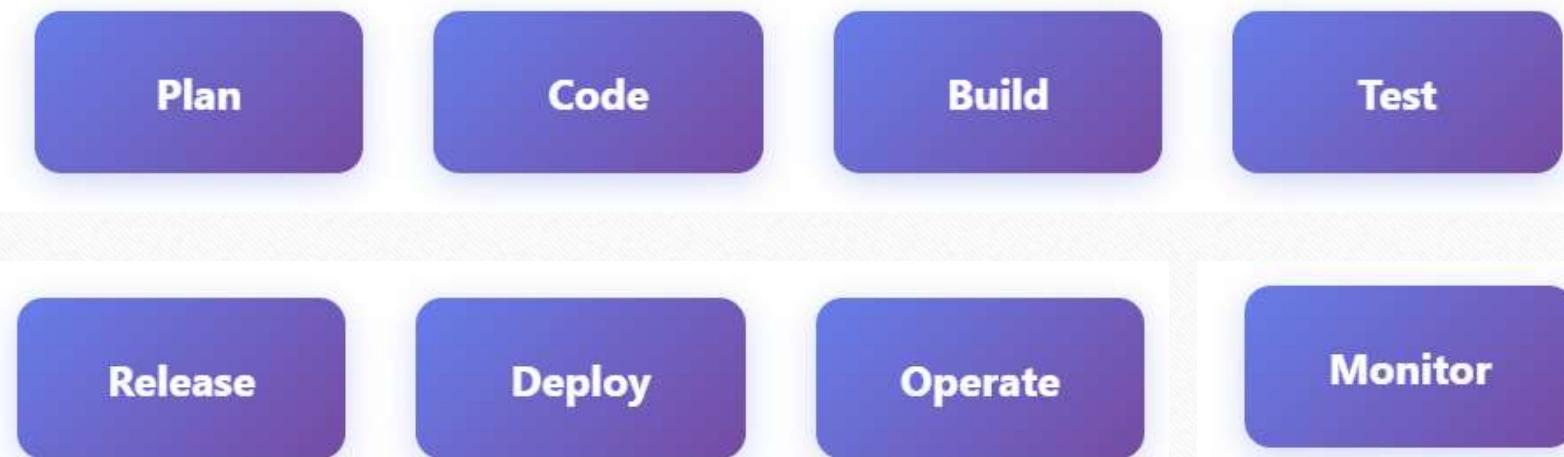
Processus Devops



Ce graphique illustre le cycle de vie DevOps, représenté par le symbole de l'infini pour souligner **la collaboration continue entre le développement et les opérations** :

- **L'Équipe de Développement (Develop Team)** : Gère la phase de création initiale, incluant la planification (PLAN), l'écriture du code (CODE), la construction de l'application (BUILD) et les tests (TEST).
- **L'Équipe des Opérations (Operation Team)** : S'occupe de la mise en production et de la maintenance, incluant le déploiement (DEPLOY), l'exploitation (OPERATE) et la surveillance des performances (MONITOR).
- **Le Point de Jonction (RELEASE)** : La phase de publication (RELEASE) agit comme le pont central où le logiciel passe du développement vers les opérations.
- **Flux Continu** : Les flèches en pointillés indiquent un mouvement cyclique et répétitif, visant à améliorer la vitesse et l'efficacité des mises à jour.

Le Cycle DevOps

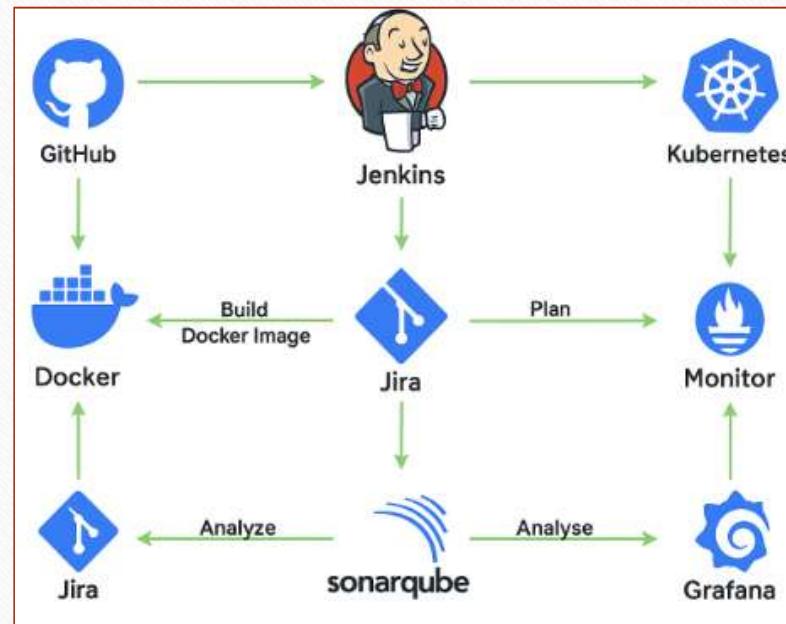


- **Plan** : Définition des objectifs, des besoins utilisateurs et de la feuille de route.
- **Code** : Développement du logiciel et conception du code source.
- **Build** : Compilation du code et création des artefacts (ex: conteneurs, packages).
- **Test** : Vérification de la qualité et détection des bugs via des tests automatisés.
- **Release** : Préparation et validation du package avant son déploiement.
- **Deploy** : Mise en production du logiciel dans l'environnement cible.
- **Operate** : Gestion de l'infrastructure et maintenance de l'application en direct.
- **Monitor** : Analyse des performances et collecte des retours pour alimenter le prochain cycle.

Principes Clés de DevOps

- **Automatisation** : Automatiser les tâches répétitives et les processus
 - Tests automatiques
 - Déploiements automatiques
 - Configuration automatique
- **CI/CD** : Intégration et déploiement continus
 - Intégration Continue (CI) : Le code est testé à chaque modification
 - Déploiement Continu (CD) : Le code validé est automatiquement mis en production
- **Infrastructure as Code** : Gérer l'infrastructure comme du code
- **Monitoring continu** : Surveillance et analyse permanentes
 - Surveillance & Feedback
 - Monitoring des performances
 - Logs et alertes
- **Microservices** : Architecture modulaire et scalable
- **Collaboration** : Culture de partage et de responsabilité, Dev et Ops travaillent ensemble,

Outils



Containerisation & Orchestration

Docker

Plateforme de containerisation leader

Kubernetes

Orchestration de containers

Docker Compose

Multi-containers en local

Helm

Gestionnaire de packages K8s

Podman

Alternative à Docker

OpenShift

Plateforme Kubernetes enterprise

Points Clés à Retenir

- DevOps est une **culture**, pas seulement des outils
- L'**automatisation** est au cœur de DevOps
- La **collaboration** entre Dev et Ops est essentielle
- Le **feedback rapide** améliore la qualité
- L'**amélioration continue** est un processus sans fin