

DevOps - Rappels



Objectifs

- Rappels sur le DevOps
- Intégration continue
- Déploiement continue

Historique

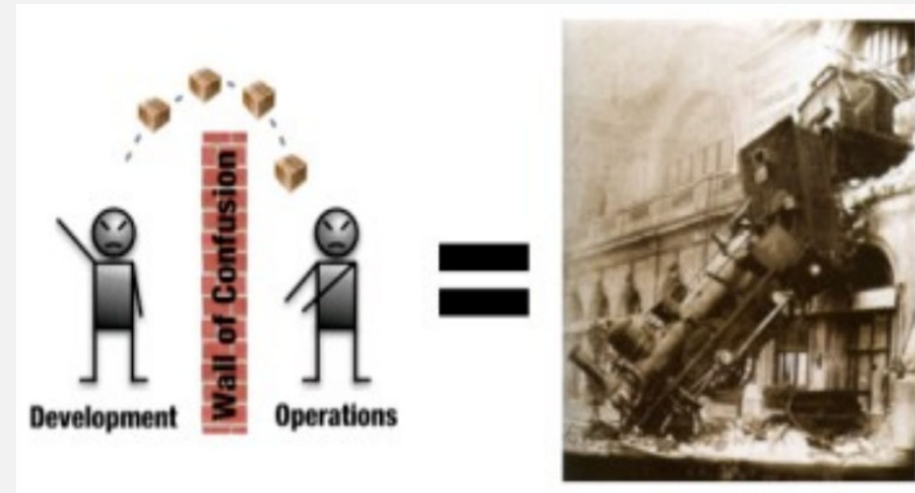
- Premières applications :
 - Taille limitée
 - Autonomes : n'avaient que peu de besoins d'interagir avec d'autres applications
 - L'équipe qui développait l'application devait aussi la maintenir

Historique

- Des systèmes de plus en plus gros :
 - Apparition des progiciels de gestion intégré (PGI) ou Enterprise Resources Planning (ERP)
 - Applications ont besoins de communiquer avec d'autres applications / systèmes
- ⇒ Besoin d'être efficace et séparation des tâches plus développement et plus infrastructure

Historique

- Trois enjeux en gestion de projets :
 - Coût
 - Objectifs
 - Délais
- Opérateurs : Stabilité
 - Augmenter la qualité
 - Au détriment du coût et du temps
- Développeurs : Livrer
 - Ajouter / modifier des fonctionnalités vite et à faible coût
 - Au détriment de la qualité



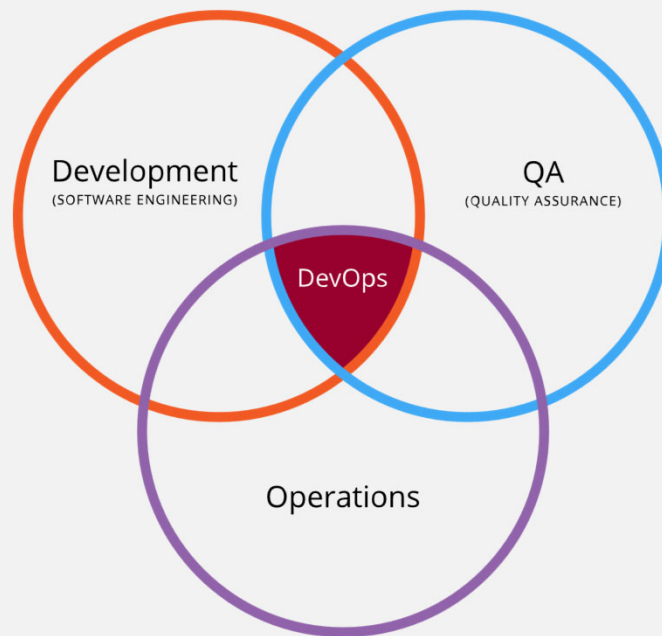
<https://fr.slideshare.net/rtang03/dev-ops-storyboardv1>

<https://fr.wikipedia.org/wiki/Devops>

DevOps

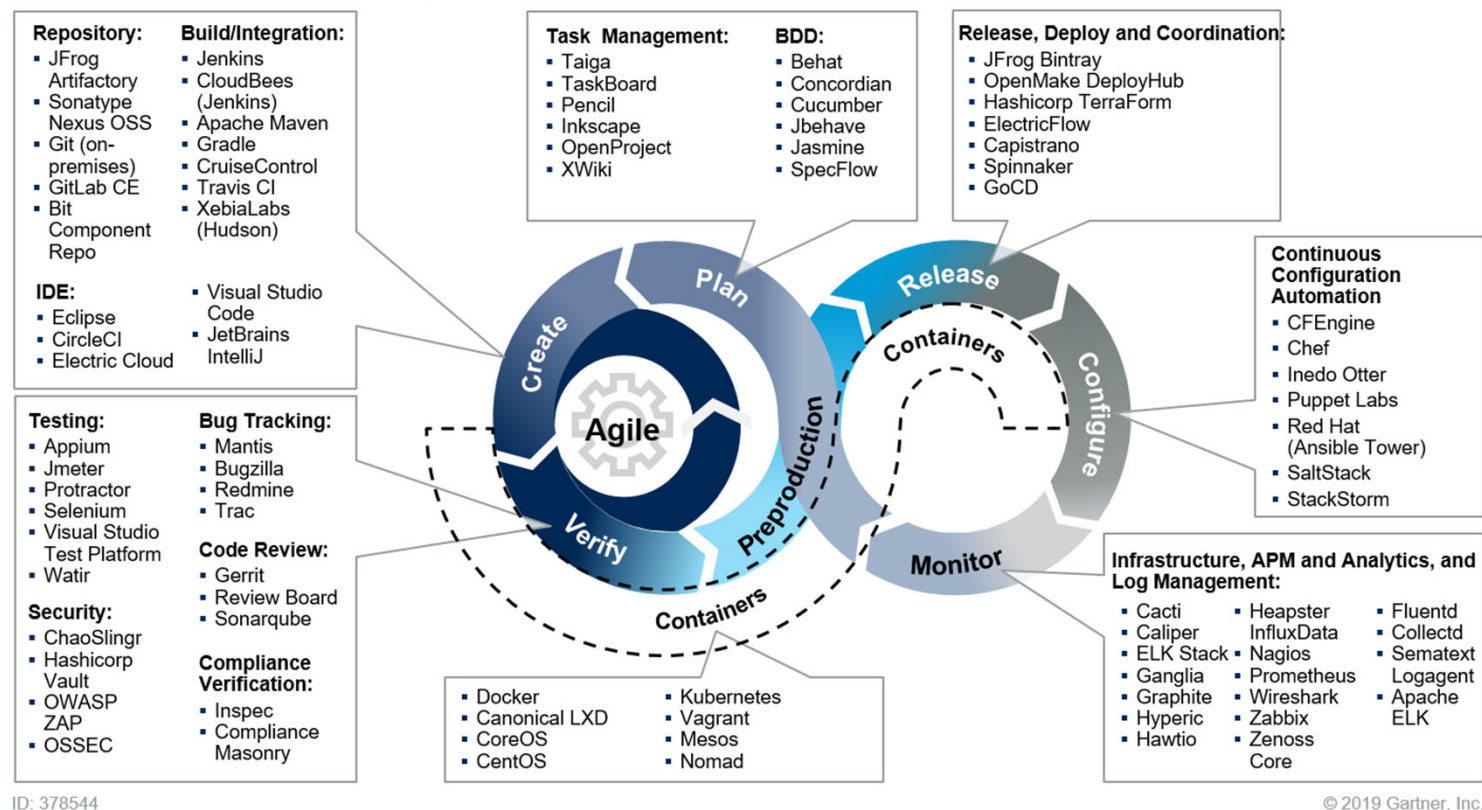
- Contraction de « développement » et d'« opération »
- Idée, c'est de faire collaborer les deux mondes, de rapprocher les équipes
⇒ ce n'est donc pas un outil !
- Pratiques DevOps
 - Utilisation des méthodes agiles pour le développement mais aussi pour l'exploitation
 - Automatiser le plus possible :
 - Création des artefacts
 - Tests (qualité ↗, coût des erreurs ↘)
 - Infrastructure
 - Installation
 - Boucle de rétroaction :
 - Livraison en continue : fréquence ↗, maîtrise du processus de déploiement ↗
 - Livrer de petits incréments : difficulté ↘
 - Livrer souvent : pratique ↗
 - Avoir une rétroaction rapide : ajustement ↗, qualité ↗

DevOps

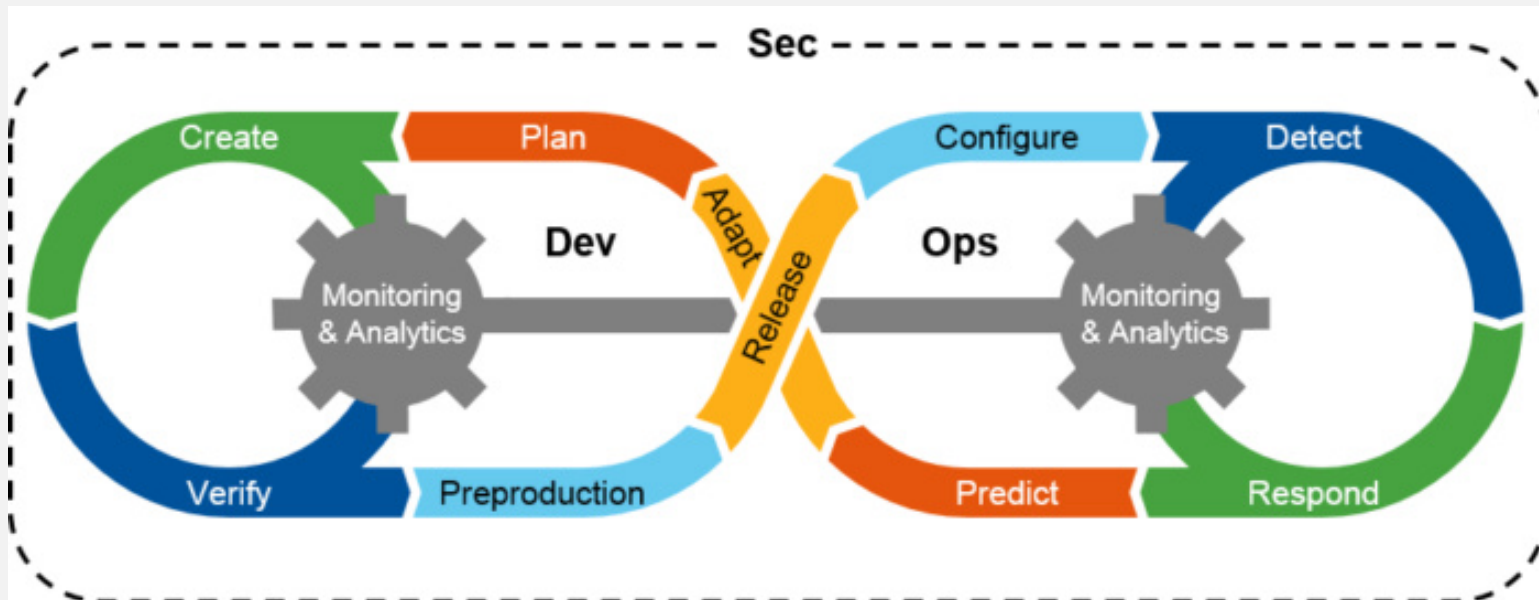


DevOps

Build an Open-Source DevOps Toolchain



DevSecOps

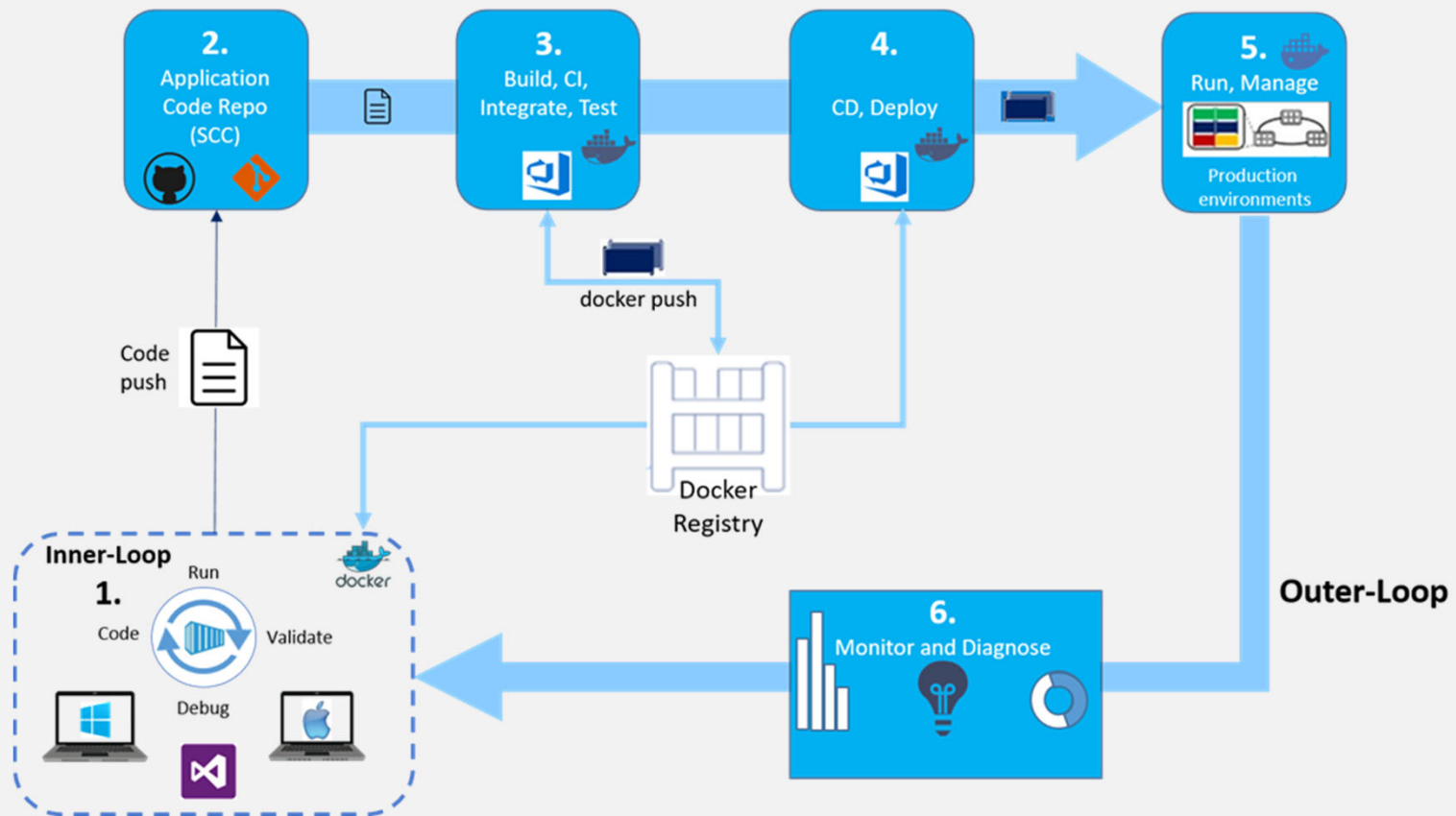


Intégration continue / déploiement continu

- L'intégration continue / Continuous integration (CI) :
 - Générer des artefacts à partir d'un gestionnaire de source / source control management (SCM)
 - Exécuter les tests
- Le déploiement continu / continuous delivery (CD) :
 - Prendre les artefacts
 - Les déploiements dans les environnements ciblés
 - Souvent, il y a un flux de travail pour migrer sur plusieurs environnements avec des validations à chaque niveau

On parle souvent de CI/CD

Exemple



Azure DevOps

- Démo à partir de 17:47

Références

- <https://fr.wikipedia.org/wiki/Devops>
- <https://docs.microsoft.com/en-us/dotnet/architecture/containerized-lifecycle/>
- <https://channel9.msdn.com/Events/Microsoft-Azure/Azure-DevOps-Launch-2018/A101/player>
- <https://channel9.msdn.com/Events/Microsoft-Azure/Azure-DevOps-Launch-2018/A102/player>