



FLORENT SOULIÈRE

CONSULTANT DEVOPS

florent.souliere@smile.fr





SOMMAIRE DEVOPS



CONCEPTS



ATTENTES



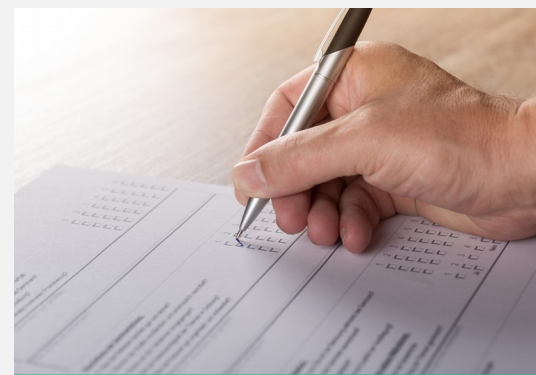
MIX MÉTHODOLOGIQUE



OUTILLAGE



UTILISATEURS DE DEVOPS



RETOURS D'EXPÉRIENCE



CONCLUSION





Concepts



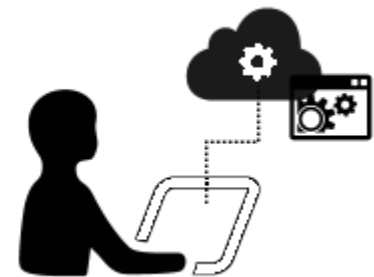
CONCEPTS

- **Dev** : les équipes de développeurs
- **Ops** : les équipes d'exploitation
- **DevOps ≠ méthodologie !**
- Le DevOps s'apparente plus à une **démarche**, toujours en construction depuis 2010
- Ses racines : le Lean, appliqué à l'IT



CARACTÉRISTIQUES

- **Décloisonner** les équipes Dev et Ops
 - Collaboration renforcée
 - Partage des responsabilités
- Une **démarche itérative** et **mutualisée** : amélioration continue et application inter-projet
- Étend **l'agilité** des développements jusqu'au déploiement



CARACTÉRISTIQUES

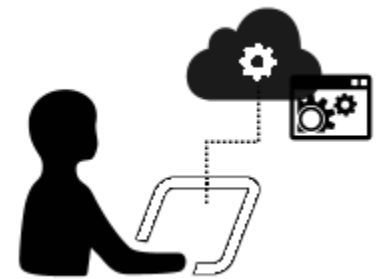
- **Décloisonner** les équipes Dev et Ops
 - Collaboration renforcée
 - Partage des responsabilités
- Une **démarche itérative** et **mutualisée** : amélioration continue et application inter-projet
- Étend **l'agilité** des développements jusqu'au déploiement



Agile



DevOps



CARACTÉRISTIQUES

- Une chaîne d'**outillage** adaptée pour livrer « plus petit, plus vite »
 - Automatisation
 - Communication entre les outils
- **Processus associés**
 - **Intégration Continue**
 - **Livraison Continue**



CARACTÉRISTIQUES

- Une version n'est considérée comme achevée que lorsque la version suivante la remplace en production
- → importance des **Mesures** mises à dispositions des équipes informatiques
 - Fréquence des déploiements
 - Vitesse des déploiements
 - Taux de succès des déploiements
 - Vitesse de restauration de la situation initiale en cas de problème bloquant
 - ... ainsi que les mesures applicatives



VALEURS

- C.A.M.P. :
 - Culture
 - Automatisation
 - Mesures
 - Partage





Attentes



ATTENTES

- Pour le Métier
 - **Mettre en production plus rapidement** de nouvelles fonctionnalités tout en préservant un niveau de qualité élevé
 - Permettre d'**expérimenter** des innovations dans les services rendus aux utilisateurs
- Pour la DSI
 - **Fiabiliser** le processus global de production de nouvelles versions d'une application
 - Améliorer la **performance opérationnelle**
 - Réduire les temps de mise en production et les risques associés



IMPACTS

- Sur les utilisateurs
 - Des services informatiques plus **réactifs**
 - Une **implication** réelle dans les évolutions apportées aux services
- Sur les équipes (Métier, Dev, Ops...)
 - Amélioration de la **collaboration**
 - Augmentation de la **motivation**
 - Possibilités d'**expérimenter**/innover





Mix méthodologique



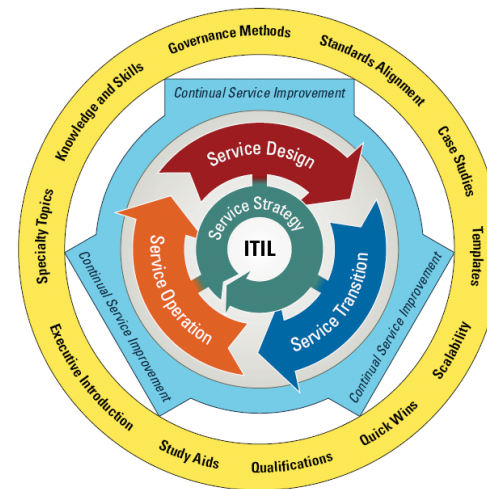
CADRE MÉTHODOLOGIQUE

- Une méthode **agile** de développement est un prérequis à la mise en place d'une réelle démarche DevOps
 - ...sans quoi l'efficacité de la démarche est fortement réduite
- Les méthodologies les plus courantes étant :
 - Scrum
 - Kanban



DEVOPS & LES RÉFÉRENTIELS

- ITIL
 - En lui-même, ITIL peut amener à créer des **silos** entre les équipes
 - ITIL et DevOps n'ont pas les mêmes échelles de temps
 - Pour autant, ITIL peut être vu comme le « quoi », et DevOps le « comment »
→ complémentarité possible entre les 2 concepts
 - Alors ITIL, compatible ou pas avec DevOps ? La question reste ouverte !



- PMBOK, CMMI, Prince2, etc.
 - À partir du moment où une méthode (ou outil méthodologique) permet d'appuyer de l'agilité dans les développements, elle peut cohabiter avec DevOps... et même le soutenir





Outillage







OUTILLAGE



Collaboration

- Jira 
- Redmine 

Gestion de code source

- GitLab 
- Subversion 
- Mercurial 
- Visual Studio Team Services 


Vérification de code source

- SonarQube 
- Jacoco 





Plateforme d'Intégration Continue

- Jenkins 
- GitlabCI 





Tests automatisés

- Selenium 
- Protractor 

Automatisation des déploiements

- Ansible 
- Puppet 
- Chef 
- Rundeck 
- GoCD 

Plateformes d'exécution

- Docker 
- Rancher 
- Kubernetes 
- OpenShift 

Monitoring/Logs

- ELK 
- Prometheus 



Utilisateurs de DevOps



UTILISATEURS DE DEVOPS

- La motivation principale de DevOps étant la réduction du « time to market », elle s'adresse naturellement aux **organisations à but lucratif**
- Pour autant, l'efficacité opérationnelle est recherchée par tous types de structures, et en particulier les **organismes publics**
- Quelle que soit la taille de la structure, **les équipes DevOps d'un même projet devraient se limiter à une dizaine de personnes, tous profils confondus**
 - C'est aussi pourquoi sa mise en œuvre sera moins lourde dans une petite structure (ex : startup) que dans une multinationale





Retours d'expérience



BONNES PRATIQUES

- **Identifier les réelles motivations** d'une mise en place de DevOps
- **Expérimenter** sur de petits projets avant de généraliser
- Ne pas s'obliger à changer l'organisation de ses équipes
- Chercher à travailler avec des **applications** les **moins monolithiques** possibles
 - Approche microservices
 - Projets maîtrisables par des petites équipes autonomes
- Le contexte opérationnel le plus favorable à DevOps est celui d'un projet en **phase d'évolution/maintenance**
- **Améliorer continuellement** la démarche par étapes successives



APPROCHE TYPIQUE DE DEVOPS

- **Contexte**
 - **Un grand groupe d'assurance** est organisé en équipes de développement et équipes de production séparées
 - Des **initiatives isolées** d'amélioration de l'outillage ont été détectées, mais ne sont pas uniformisées ni généralisées
 - L'entreprise souhaite développer et généraliser les initiatives isolées dans une **démarche DevOps globale**
- **Mode opératoire**
 - **Audit** des projets et des processus actuels
 - **Identification** des différents points d'amélioration possibles à l'aide d'ateliers spécifiques
 - **Définition** d'une stratégie DevOps
 - Évolutions à apporter dans l'organisation
 - Modification de processus
 - Outillage
 - Accompagnement au changement



DEVOPS & MODE PROJET

- **Contexte**
 - **UMS** créée à l'initiative du Ministère de l'Enseignement Supérieur et de la Recherche
 - L'équipe pluridisciplinaire profite d'un projet de refonte complète de ses applicatifs pour entamer une **démarche DevOps globale**
- **Mode opératoire**
 - La partie « développement » du projet est menée avec la méthode agile **SCRUM**
 - Une démarche DevOps vient inclure SCRUM dans un processus étendu de gestion des développements, de l'émission de la demande à l'audit en production
- **Retours d'expérience**
 - Les sprints sont une unité de travail encore trop conséquente pour optimiser les effets de DevOps
 - Pour maximiser les effets de la démarche DevOps, il est décidé que **le mode « projet » devra durer le moins de temps possible**
 - ...pour permettre de se passer des sprints et fluidifier encore les mises en production de nouvelles versions



LES LOCOMOTIVES DE DEVOPS

- Avant même l'existence du nom, ce sont les « purs acteurs Internet » qui ont fondé les bases de DevOps : Google, Amazon, Netflix, Facebook, Twitter...
- Des contributions fortes
 - Google : Kubernetes
 - Amazon : AWS
 - Netflix : Chaos Monkey
 - ...





Conclusion



CONCLUSION

- Une règle d'or dans la mise en place d'une démarche DevOps : toujours privilégier une **approche agile**, en amélioration continue par **petites étapes maîtrisées**
- S'agissant d'une **évolution culturelle de l'IT**, sa mise en œuvre varie d'une structure à une autre : il n'y a pas (encore) de règles
- Mots-clés
 - **Expérimentation**
 - **Bon sens**
 - **Collaboration**
 - **Small is beautiful**