DevOps : le nouveau standard agile et fiable

En conséquence, afin de garantir l'agilité, fiabilité et sécurité aux clients, les entreprises doivent assurer une collaboration entre développeurs, ingénieurs de production et responsables métiers – bref, adopter une démarche DevOps.

DevOps : Phénomène passager ou innovation incontournable ?

Il y a encore quelques années, une démarche DevOps se résumait, grosso-modo, à donner aux développeurs l'accès direct à l'environnement de production. Cette première étape a permis d'extraire le développeur de son silo et l'associer à la mise en œuvre des opérations IT ; ce fut le début d'un consensus mutuel entre « Ops » et « Devs » sur la nécessité d'automatiser les tâches répétitives de mise en production.
Accélération numérique oblige, la démarche DevOps a profondément modifié l'organisation même du travail.

Dans sa définition la plus simple, DevOps est un ensemble de pratiques permettant de livrer des applications, des logiciels ou d'autres outils informatiques en production, de manière plus rapide et plus fiable. Les fondamentaux sont l'automatisation, mais également la collaboration et le partage. Là où auparavant les équipes se passaient la main, elles doivent maintenant travailler ensemble et se faciliter mutuellement le travail.

Une méthode moderne

DevOps est une approche et non pas une famille d'outils. Ce vocable n'est donc pas juste une mode ou le buzzword de ces 10 dernières années, malgré certains abus de langage et la généralisation des titres « ingénieur DevOps ». Il serait plus juste de parler du concept de Site Reliability Engineer (SRE) qui reflète bien mieux la nouvelle répartition des responsabilités.

Le rôle des Ops est de créer l'autoroute permettant le déploiement rapide et simple (donc automatisé) du code depuis le repository vers la production et d'identifier les solutions architecturales. À ce titre, l'évolution la plus symptomatique de l'environnement de travail des Ops a été le recul de l'importance de l'outil de service desk/ticketing et l'adoption des outils de gestion de version. Les Ops ont ainsi adopté des techniques de Dev.

La sécurité : un enjeu clé du DevOps

L'approche DevOps est devenue une nécessité dans un environnement économique en constante accélération et qui fait la part belle aux start-ups et aux géants de la tech. Pour rester dans la course, les entreprises doivent opter pour le cloud, le SaaS et l'ouverture de leurs systèmes

d'information via des Apps mobiles ou des APIs. Mais cette médaille a un revers.

Les organisations font face à des risques sécuritaires majeurs car leurs applications sont exposées sur le web. Elles sont composées majoritairement de briques logicielles open source qui érigent, à raison, la transparence comme argument principal de la sécurité et de la confiance.
Dès lors, la communauté informatique a dû remettre en question les dogmes mêmes sur lesquels elle avait bâti son fonctionnement. Elle a été obligée de renoncer à placer l'environnement de production dans une bulle de protection au nom de la sacro-sainte croyance : « moins on y touche, plus on garantit la sécurité ».

Gare à la passivité !

Voilà pourquoi l'impératif de réactivité, essentiel à la survie des entreprises, doit s'imposer dans l'agenda des DSI ! Par exemple, ne pas mettre à jour ses environnements de production régulièrement, voire de manière réactive et à la demande, c'est laisser la possibilité aux hackers, même novices, d'exploiter des failles, sachant qu'elles sont connues et documentées car déjà corrigées dans les versions récentes.

En 2017, avec un coût (amendes, remédiations, indemnités…) estimé à 1.4 Milliards de dollars, une entreprise comme Equifax a payé au prix fort cette passivité.
Les différentes études sur DevOps (State of DevOps par DORA notamment) montrent que, contrairement aux habitudes et croyances héritées du passé, ce sont les équipes qui déploient le plus fréquemment qui obtiennent la meilleure stabilité. Même les applications sont concernées, une fois finalisées : la découverte d'une vulnérabilité telle que « Zip Slip », par exemple, nécessite un examen immédiat du code applicatif, ainsi que la correction et le déploiement d'une nouvelle version pour lui éviter d'être prise pour cible.

La nécessité d'une collaboration multidisciplinaire

Pour aller plus loin dans la démarche DevOps, les entreprises ne doivent plus attendre que le code soit considéré comme « prêt pour la production » avant d'être déployé. En d'autres termes, elles doivent être capable de tester en production afin de valider des hypothèses métier et technique. Cette approche requiert des fonctions d'observabilité, d'aiguillage/filtrage d'accès aux fonctionnalités (feature toggle) et d'isolation du code.
Celles-ci sont déjà disponibles sur le marché mais pas assez répandues.

Grâce à ces techniques, on dissocie l'action de mise en production du code d'avec l'événement de mise à disposition d'une fonctionnalité. La granularité des déploiements est plus fine, avec des impacts moindres et plus facilement identifiables. Mais, au-delà des bénéfices techniques, ce niveau de maturité DevOps apporte véritablement un nouveau levier d'agilité. La notion de sprint disparaît. De plus, un réel partenariat entre le métier et les développeurs peut se mettre en place. Il est ainsi possible d'expérimenter rapidement, tester des implémentations ou dissocier des expériences en fonction des utilisateurs.

La maturité DevOps signée par des échanges fluides et efficaces

Depuis une dizaine d'année, les Dev sont décisionnaires en matière de choix d'outils et d'environnement informatique, auparavant du ressort des Ops, qu'il s'agisse de serveurs applicatifs, puis de bases de données et maintenant, avec l'essor des microservices et des conteneurs, d'architectures d'exploitation complète.
Paradoxalement, alors que dans l'idéal les équipes Ops auraient dû faire disparaître l'infrastructure et les problématiques de déploiement du quotidien des développeurs, ces derniers s'engouffrent massivement dans la vague Kubernetes.

Le processus de développement doit donc, plus que jamais, être un système qui favorise et unifie les échanges entre développeurs, métier, sécurité et Ops. Ce n'est plus ce quelque chose de mystérieux qu'on alimente avec des pizzas et des post-it. Les principes de visibilité et traçabilité du code, qu'il soit applicatif ou d'infrastructure, doivent également s'appliquer aux décisions architecturales et fonctionnelles car leur impact est omniprésent.
Au-delà de la technique, une communication fluide et transparente est le signe extérieur ultime de la maturité DevOps.

Bien plus qu'un mot à la mode, DevOps est devenu une véritable nécessité. Véritable levier d'agilité, cette approche représente une opportunité unique de faire bouger les lignes dans la relation entre métier et technique, au-delà des pratiques Agiles stéréotypées habituelles.

Néanmoins, DevOps ne peut pas faire l'impasse sur la sécurité qu'il contribue d'ailleurs à améliorer en favorisant son intégration tout au long du cycle de développement. La sécurité doit donc faire partie intégrante de la collaboration entre développeurs, opérationnels et experts de ce domaine.