

La collaboration est la clé du succès de DevOps

SAP est l'un des plus grands éditeurs de logiciels du monde et compte plus de 20 000 développeurs répartis aux quatre coins du globe. Dans l'environnement DevOps actuel, la direction de SAP sait parfaitement que la réussite de cette organisation tentaculaire dépend de la bonne collaboration entre des équipes éloignées géographiquement. Pour cela, il a fallu trouver une méthode efficace de communication et de collaboration, tout en favorisant l'efficacité et la créativité.

Pour Dominik Tornow, directeur de l'ingénierie chez SAP pour le groupe outils et processus des laboratoires, la centralisation des processus ne favorise pas l'innovation. Les développeurs, les architectes et les chefs de produit cherchent tous à proposer des fonctions d'exception, enthousiasmantes et en continu. Pour cela, ils doivent pouvoir collaborer à leurs propres conditions.

Oui, mais comment ?

DevOps a été si largement adopté que les organisations cherchent à optimiser l'efficacité en partant de ce modèle de développement logiciel. L'un des principaux objectifs du DevOps est d'optimiser l'efficacité tout au long du cycle de développement des systèmes. Et les études prouvent que ce modèle continue de se développer à grande vitesse.

L'efficacité DevOps n'est pas affaire de technologie, mais de collaboration

Dernièrement, dans une étude internationale auprès des responsables IT dans 10 grands secteurs économiques, TechTarget notait une augmentation annuelle de 17 % des organisations ayant adopté DevOps, contre 30 % seulement n'ayant encore aucune initiative DevOps. Pour les personnes interrogées, le premier obstacle DevOps consiste à « éliminer les goulets d'étranglement des processus pour accélérer

la publication des applications », suivi de près par « rationaliser la collaboration entre les équipes DevOps transverses ». S'ensuivent d'autres points épineux, dont l'infrastructure logicielle, l'intégration du Cloud et la sélection d'outils ; mais les deux premiers concernent bien les personnes, pas la technologie.

Dans la mise à jour 2018 d'une autre grande étude annuelle sur DevOps, le pourcentage de personnes interrogées travaillant dans un « service nommé DevOps » montre une brusque hausse de 20 %, la plus forte poussée ayant eu lieu au cours de l'année écoulée. Autre chiffre tiré de cette étude d'envergure mondiale, 36 % des personnes interrogées développent actuellement d'un quart à 100 % de leurs applications suivant le modèle DevOps.

Alanna Brown, à l'initiative en 2012 de l'étude mentionnée ci-dessus, a relancé l'idée que « les personnes, pas la technologie, » sont nécessaires pour accroître l'efficacité DevOps. « Pour nous, l'essentiel a toujours été que les équipes aient les moyens de donner toute la mesure de leur talent, en dépassant les différences culturelles entre les équipes de développement et de l'opérationnel et en améliorant l'IT au bénéfice de tous », écrivait-elle récemment dans un blog.

Et le rapport 2018 indique bien que les bonnes pratiques DevOps commencent par « l'amélioration de la collaboration ». Il pointe également que « le partage transversal entre les équipes est essentiel pour amplifier la réussite DevOps. Les pratiques qui ont l'impact le plus fort sur le parcours DevOps évolutif reposent sur le partage. Les organisations qui connaissent quelques petites réussites DevOps, mais n'arrivent pas à le généraliser, accusent du retard et ne gagnent pas en automatisation et en libre-service ».

Open source, DevOps et collaboration

L'adoption de DevOps s'est développée ces dernières années, tout comme l'open source. Partant de l'engouement pour l'open source dans le monde entier, une projection évoque le doublement du chiffre d'affaires en trois ans seulement, jusqu'à 26 milliards de dollars. La collaboration est intimement liée à l'open source, dans lequel les logiciels sont souvent développés suivant un processus public très collaboratif. De fait, les entreprises internationales de tous les secteurs ont rapidement intégré les bonnes pratiques de l'open source dans leurs méthodes de développement et de déploiement des logiciels.

De nombreuses organisations adoptent « l'innésource », qui est, au fond, de l'open source derrière un pare-feu. L'innésource est un mode de développement hautement collaboratif qui suit les bonnes pratiques des projets open source à grande échelle. De tels projets (Kubernetes, Microsoft Visual Studio Code, etc.) exigent une coordination à l'échelle de milliers de développeurs et autres contributeurs. Les expériences d'innésource ont permis d'établir une liste de bonnes pratiques visant à réunir l'innésource, l'open source et DevOps pour optimiser la collaboration et l'efficacité.

L'innésource se révèle une arme redoutable contre les difficultés majeures liées à DevOps et pointées plus haut. L'innésource se concentre sur la collaboration du développement et des opérations, mais aussi sur la collaboration de toutes les équipes intégrées au processus logiciel.

« Lorsque vous l'avez adopté [l'innésource] et que vous voyez émerger de nouvelles équipes, vous êtes témoin non seulement de la contribution des personnes,

mais aussi du déblocage de goulets d'étranglement » explique Jeremy King, vice-président exécutif et directeur technique du secteur eCommerce international chez Walmart. Jeff Jagoda, ingénieur logiciels senior chez IBM, affirme : « Nous voyons l'innésource comme un moyen d'améliorer l'efficacité par la réutilisation du code. Mais, plus que cela, c'est un formidable moyen d'apprendre, d'échanger des idées et de faciliter l'innovation chez IBM. »

Des efforts conjugués pour dynamiser la collaboration dans DevOps

SAP exploite l'idée de la « sagesse des foules »

Certaines organisations ne progressent pas dans le franchissement des obstacles DevOps. Chez SAP, déjà mentionné plus haut, un groupe isolé dans cet immense complexe de développement interne a sollicité de la direction l'autorisation d'exploiter une plateforme de développement innovante, GitHub, pour optimiser la collaboration entre les membres de l'équipe disséminés aux quatre coins du globe.

Les premières réussites avec GitHub se sont répandues comme une traînée de poudre. Rapidement, d'autres développeurs travaillant sur d'autres projets ont partagé du code et des idées pour créer des flux de travail pratiques et agiles, raconte Dominik Tornow, directeur de l'ingénierie chez SAP pour le groupe outils et processus des laboratoires. Selon ses dires, les engagements, les commentaires et les problèmes partagés de façon collaborative sur la plateforme sont devenus les principaux ingrédients de l'efficacité et de l'innovation. Un jour, le groupe de Tornow a lancé sur la plateforme un projet ne contenant que quelques fichiers. Dans les 24 heures, une équipe travaillant 12 fuseaux horaires plus loin l'a contacté pour proposer sa contribution.

C.H. Robinson remplace les silos par la collaboration – et de meilleurs logiciels

Pour accompagner sa croissance rapide, C.H. Robinson, chef de file des services logistiques, devait s'assurer que ses 800 collaborateurs des équipes IT et de développement continuaient de fonctionner comme une équipe malgré leur dispersion géographique, sur de nombreux bureaux et sur trois continents. Il fallait pour cela que les flux de travail de l'équipe soient aussi collaboratifs et transparents que possible.

Par ailleurs, la relative autonomie laissée aux équipes de développement leur permettait de choisir leurs propres outils, ce qui ne facilitait pas la standardisation du code ni la visibilité et la documentation du projet. Les développeurs n'avaient donc aucun moyen pratique de travailler sur le code d'autres collègues, même s'ils pouvaient le localiser. En cherchant à favoriser l'autonomie des équipes de développement locales, l'entreprise avait fini par créer une organisation IT en silo. Pour débloquer cette situation, C.H. Robinson a adopté la plateforme de collaboration GitHub et un outil de gestion de projet agile. Pour Tyler Patterson, directeur technique ingénierie, il en résulte aujourd'hui une réorientation complète de l'entreprise « vers l'ouverture qui fait émerger les meilleures solutions. Nous prenons des mesures en interne pour rendre le code public. [Les outils] nous permettent de travailler de manière ouverte, de faire tomber les murs de l'IT, celui du code, celui du contrôle de la source... Nous sommes un service et une équipe ouverts et collaboratifs et nous produisons de meilleurs produits... Nous tirons pleinement parti du travail des autres développeurs, où qu'ils soient dans le monde. C'est cela qui fait la grandeur de notre univers IT. »

L'université du Minnesota exploite la puissance collaborative de ses 230 départements et service administratifs

La plateforme de collaboration adaptée s'est également avérée moteur d'efficacité et d'innovation à l'université du Minnesota, où les développeurs sont disséminés dans 230 départements de développement dans tout l'État. Comme pour SAP, le mouvement est parti d'un petit groupe de développeurs qui avaient besoin d'utiliser « un réseau social pour collaborer sur le code ». La plateforme de collaboration GitHub, d'abord adoptée par le groupe, a ensuite rapidement été mise à la disposition de la totalité des 60 000 étudiants de l'université.

Pour Peter Walz, ingénieur système à l'université, la plateforme de collaboration a permis d'accélérer les envois de code et les demandes d'extraction (respectivement « code push » et « pull request »),

ce qui a largement amélioré les flux de travail. « On voit facilement l'effet des modifications avant leur intégration », explique-t-il. Il a également noté une baisse de charge pour l'administration IT, car la plateforme permet aux utilisateurs de définir les autorisations et d'utiliser librement le support. Pour lui, dans l'ensemble, la plateforme a « simplifié la collaboration entre les départements ».

GitHub, boîte secrète de la collaboration

Les trois organisations ont toutes choisi la plateforme GitHub pour faire progresser la collaboration et l'efficacité. SAP, C.H. Robinson et l'université du Minnesota ont rejoint les 2,1 millions d'organisations et les 31 millions de développeurs qui utilisent GitHub dans le monde entier. Elles font désormais partie de la plus grande communauté open source du monde et partagent du code, collaborent en toute sécurité et développent plus rapidement des logiciels de meilleure qualité. Au-delà du fait d'être la plus grande communauté open source, GitHub est également simple d'utilisation. Amir Jaballah, responsable international de la plateforme de livraison continue de la Société Générale, constate : « J'ai rarement connu un tel enthousiasme pour le déploiement et l'adoption d'une nouvelle solution. »

GitHub est une plateforme très polyvalente qui peut être déployée localement, en mode SaaS ou dans une configuration hybride alliant les deux avec GitHub Connect. Ses outils d'audit et de supervision évolués favorisent l'entraide et la collaboration des équipes de développement dans le respect des exigences spécifiques de l'environnement interne sécurisé.

Qu'il soit utilisé localement sur les serveurs ou dans un Cloud privé, GitHub permet aux équipes DevOps de dynamiser globalement l'efficacité du développement avec des options de déploiement flexibles, des autorisations centralisées, des centaines d'intégrations et un support technique.

Voici les principaux éléments de la plateforme GitHub:



Collaboration

Les demandes d'extraction (« pull request ») et les problèmes (« issues »), la collaboration et la vérification du code sont intégrés à la procédure de développement. Les équipes internes et externes peuvent partager du travail, discuter des changements et recevoir des commentaires, le tout au même endroit. Les organisations peuvent donc partager l'expertise en interne et ne sont pas obligées de réinventer des solutions développées pour d'autres projets et déjà testées en conditions réelles.



Communauté

2,1 millions d'organisations et plus de 40 millions de développeurs font confiance à GitHub et l'ont adopté pour acheminer la puissance de la communauté open source jusqu'aux équipes. Ces dernières peuvent tirer parti de nouveaux projets, trouver des solutions et développer de meilleurs logiciels tout en bénéficiant du support de développeurs du monde entier.



Sécurité

Des alertes de sécurité déclenchées par IA envoient des notifications lorsque des dépendances vulnérables sont utilisées dans un projet. Depuis le lancement des alertes de sécurité en 2017, plus de cinq millions d'alertes ont été envoyées pour aider la communauté GitHub à protéger ses projets. L'analyse de jetons détecte et invalide les secrets poussés vers GitHub. Cette sécurité renforcée permet aux entreprises de livrer quand elles le souhaitent, de créer un environnement dans lequel elles peuvent réaliser des tests et des essais avec des applications existantes retouchées et d'optimiser les résultats. La protection des branches assure l'analyse des changements et leur vérification par CI, GPG et audit.



Intégration

GitHub prend en charge les outils personnalisés et des centaines d'applications et de services tiers. Associé à des services comme Jenkins, CircleCI, Travis CI ou BuildKite, GitHub permet d'automatiser les flux de travail pour votre environnement de production. Un nombre croissant d'outils et d'intégrations sont accessibles sur GitHub Marketplace. Parmi eux, Sentry permet la création de journaux d'erreurs et de rapports d'incidents interplateformes en temps réel ou issue.sh pour la gestion de projet agile dans GitHub, sans accès au code.

¹ "Informing the path to enterprise DevOps deployments," TechTarget, October 2018.

² "2018 State of DevOps Report," Puppet, 2018.

³ "2018 State of DevOps Report: Practical guidance for your DevOps evolution," Puppet Blog, September 2018.

⁴ Ibid(2).

⁵ "Projected revenue of open source services from 2017 to 2022 (in billions of U.S. dollars)," Statista, 2018.