

NEUF BONNES PRATIQUES **POUR LA GESTION DES ARTEFACTS**

Les artefacts désignent les composants et les produits critiques du développement applicatif. Dans le paysage actuel du développement logiciel, en perpétuelle évolution et dynamique, la gestion efficace des artefacts apparaît comme un facteur de réussite stratégique. Mises en œuvres à bon escient, des pratiques efficaces de gestion des artefacts permettent de réaliser des gains de productivité importants, de rationaliser vos processus et d'assurer la livraison correcte de vos projets logiciels.

Cet e-book vous présente neuf bonnes pratiques en matière de gestion des artefacts. Nous explorerons chaque bonne pratique en détail et vous proposerons pour chacune des conseils faciles à traduire en actions. Les recommandations qui suivent vous apporteront les connaissances et les outils nécessaires à la gestion efficace de vos artefacts tout au long de leur cycle de vie.



NEUF BONNES PRATIQUES

- 1** Stockez vos packages et vos artefacts dans des outils conçus pour héberger chaque type de fichiers 03
- 2** Séparez les artefacts et les dépendances en créant une structure de dépôts locaux, distants et virtuels 04
- 3** Créez des proxys de dépôts publics pour disposer d'artefacts en cache local 05
- 4** Prévoyez des dépôts pour chaque phase de développement 06
- 5** Affectez à chaque équipe un ensemble dédié de dépôts locaux, distants et virtuels 07
- 6** Promouvez vos artefacts — d'un environnement à l'autre au sein du SDLC — sans les recréer systématiquement 08
- 7** Stockez et gérez les métadonnées associées à chaque artefact au même endroit que celui-ci 09
- 8** Créez des politiques d'identification et d'autorisation d'accès à un dépôt donné assorties d'actions précises 10
- 9** Définissez des routines de nettoyage pour améliorer l'hygiène et les performances des dépôts 11

1. STOCKEZ VOS PACKAGES ET VOS ARTEFACTS DANS UN OUTIL CONÇU POUR HÉBERGER CHAQUE TYPE DE FICHIER

Structure, contenu, configuration : tous les packages sont différents. S'il reste possible de stocker plusieurs types de fichiers au sein d'un même dépôt générique, les entreprises qui font ce choix limitent la capacité d'automatisation de leurs pipelines logiciels et leur capacité de capture des données dans le cadre du processus de développement. C'est pourquoi on recommande généralement de stocker les packages et les artefacts dans des dépôts conçus pour héberger chaque type de packages spécifique.

EN QUOI EST-CE IMPORTANT ?

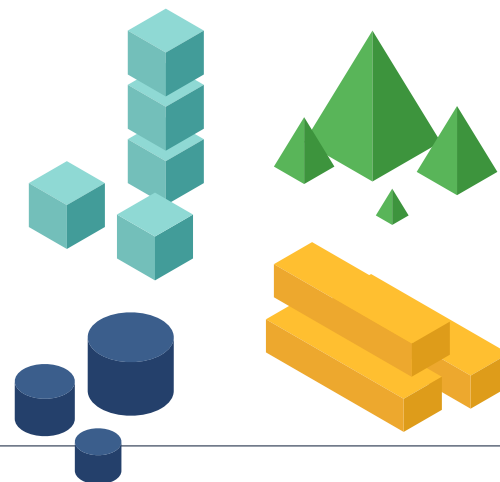
Chaque gestionnaire de package présente des commandes, des caractéristiques et des données de sortie spécifiques. Pour favoriser l'automatisation et leur intégration à vos pipelines logiciels, les dépôts dans lesquels vous stockez vos artefacts doivent être capables d'intégrer vos différents outils usuels dans vos processus de build et de communiquer avec eux.

Le stockage des packages et des artefacts au sein d'un outil conçu spécifiquement pour un type de fichier donné facilite également la capture des métadonnées.

Quand les packages et les artefacts sont stockés dans un outil spécifique, ce dernier peut capturer et enregistrer des informations relatives aux fichiers, qu'il s'agisse de leur type, de leur date de création, de leur auteur ou de toute autre caractéristique.

Cette pratique est d'autant plus importante qu'elle vous

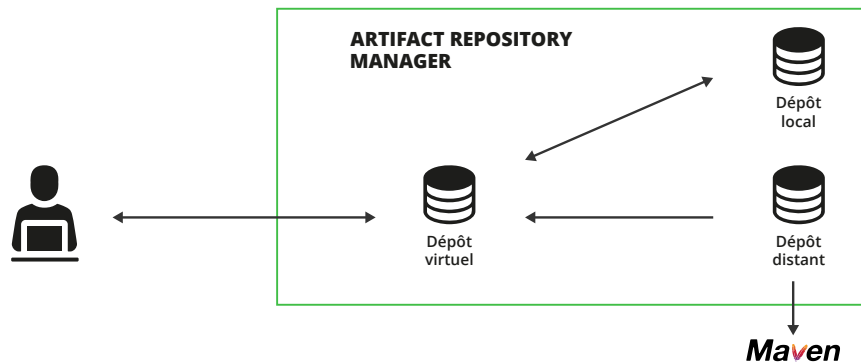
aide à mieux vous organiser. Lorsque vous disposez d'un outil conçu spécifiquement pour un type de fichier donné, vous pouvez l'utiliser pour réorganiser les packages et les artefacts en dossiers, selon des balises



EN SAVOIR PLUS

[Consolidation des outils de gestion des packages](#)

2. SÉPAREZ LES ARTEFACTS ET LES DÉPENDANCES EN CRÉANT UNE STRUCTURE DE DÉPÔTS LOCAUX, DISTANTS ET VIRTUELS



EN QUOI EST-CE IMPORTANT ?

Pour des raisons structurelles, de sécurité et de conformité, les entreprises doivent disposer d'un moyen simple pour stocker et organiser leurs artefacts de manière à pouvoir identifier clairement les éléments propriétaires et ceux issus de sources publiques (les dépendances, par ex.). Pour bien distinguer les artefacts propriétaires des autres artefacts, l'idéal consiste à utiliser deux dépôts distincts pour ces deux classes d'artefacts.

Les dépôts locaux et distants permettent aux entreprises de séparer leurs artefacts propriétaires des autres types d'artefacts. Les dépôts virtuels facilitent l'administration et les projets de gouvernance lorsque des changements s'imposent.

Les applications logicielles modernes se composent d'un grand nombre de packages et de dépendances. Il peut s'agir de composants créés en interne ou de projets open source utilisés pour accélérer le processus de publication. Par exemple, un projet peut nécessiter une fonctionnalité d'identification : plutôt que de coder et de tester l'intégralité d'un référentiel d'identification en partant de zéro, vous pouvez avoir recours à des frameworks d'identification OSS comme log4j, plus sophistiqués et plus matures.

Les bonnes pratiques recommandent d'utiliser une structure de dépôts incluant à la fois des dépôts locaux, distants et virtuels.

Les dépôts **locaux** accueillent la totalité de vos artefacts propriétaires, développés sur mesure en fonction de vos besoins opérationnels.

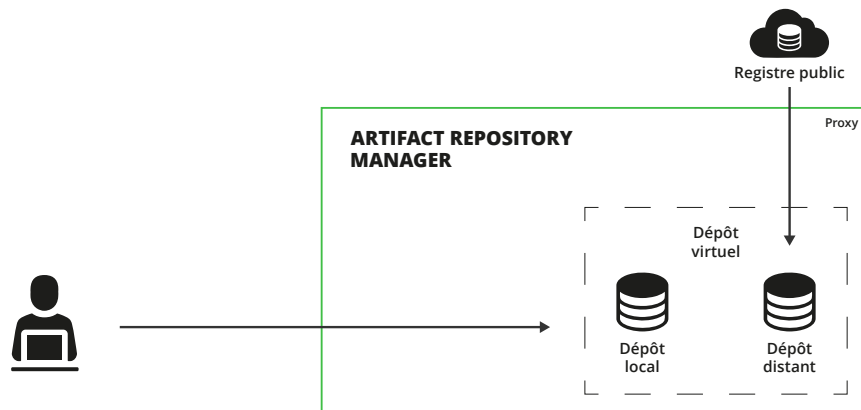
Les dépôts **distants**, eux, peuvent être utilisés pour mettre en cache des packages provenant de dépôts OSS publics comme Maven Central, Docker Hub, npm, etc.

Enfin, les dépôts **virtuels** facilitent l'administration de la gestion des dépôts. Comme les dépôts virtuels agrègent les dépôts locaux et distants, vous pouvez les utiliser comme un endpoint unique afin de créer de nouveaux artefacts propriétaires en résolvant les dépendances de packages OSS et en publiant vos artefacts vers Artifactory.



Bonnes pratiques de taxonomie et de nommage des dépôts d'artefacts

3. CRÉEZ DES PROXYS DE DÉPÔTS PUBLICS POUR DISPOSER D'ARTEFACTS EN CACHE LOCAL



Aujourd'hui, les packages open source constituent en pratique les fondations de tous les logiciels du marché. Pour construire leurs applications logicielles, les développeurs et les systèmes de build s'appuient souvent sur ces bibliothèques open source et tierces, généralement hébergées sur des systèmes distants sur Internet. Si cette pratique est généralisée, les entreprises doivent néanmoins relever certains défis liés à l'usage d'artefacts hébergés par des sources publiques.

EN QUOI EST-CE IMPORTANT ?

Les entreprises doivent maîtriser la stabilité, la disponibilité, la sécurité et la traçabilité des artefacts extraits de dépôts publics. →

Des problèmes peuvent survenir en cas de suppression des artefacts/dépendances des dépôts publics, ou d'encombrement du réseau, source d'instabilité. En outre, la traçabilité peut s'avérer difficile si les dépendances sont directement téléchargées depuis Internet. La mise en cache locale d'artefacts hébergés dans des dépôts distants offre un accès stable et homogène aux packages sans nécessiter de téléchargements répétés.

BONNE PRATIQUE DE SÉCURITÉ

Interdisez aux développeurs les téléchargements de packages directement depuis Internet

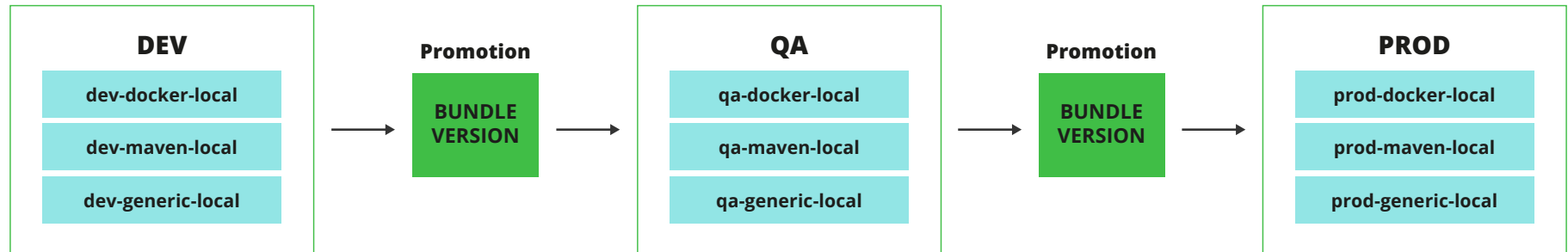
Les entreprises qui souhaitent adopter les bonnes pratiques de sécurité s'appuieront sur un gestionnaire de dépôts d'artefacts pour servir d'intermédiaire entre les développeurs et Internet en créant des proxys de dépôts publics. Même si aucun package malveillant n'est incorporé au build, un développeur pourrait télécharger un malware sur son poste de travail et exposer la totalité du réseau et des systèmes. L'utilisation d'Artifactory pour accéder à l'ensemble des packages constitue un premier niveau de défense et de contrôle.



EN SAVOIR PLUS

[Gestion des risques et des vulnérabilités open source](#)

4. PRÉVOYEZ DES DÉPÔTS POUR CHAQUE PHASE DE DÉVELOPPEMENT

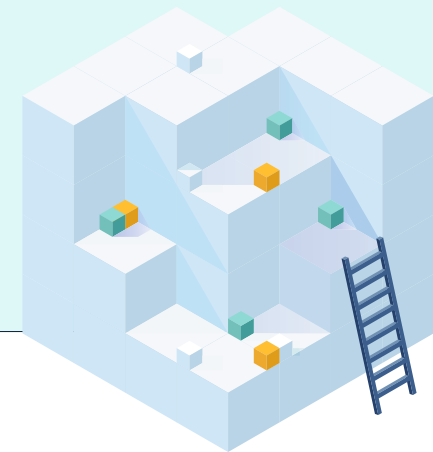


Les entreprises doivent aligner la structure de leurs dépôts sur les différentes étapes de leur SDLC. Les versions logicielles se composent d'une multitude de builds/packages/artefacts. En prévision de leur publication, les entreprises font progresser les logiciels dans les différentes phases du SDLC en réalisant différents tests à chaque étape.

EN QUOI EST-CE IMPORTANT ?

En attribuant des dépôts à chaque phase du SDLC, les entreprises peuvent déplacer l'intégralité des composants d'une version logicielle vers les dépôts adaptés. Ainsi, elles contrôlent la promotion du logiciel (en général à l'issue d'une série donnée de validations) ainsi que les profils autorisés à accéder aux composants en fonction de l'étape spécifique du SDLC, sans jamais nécessiter la moindre reconstruction, même partielle, du logiciel. Cette approche renforce l'intégrité et la sécurité des données, tout en permettant aux entreprises de suivre de façon transparente l'état de leurs versions.

 **EN SAVOIR PLUS**
[Repenser le SDLC](#)



5. AFFECTEZ À CHAQUE ÉQUIPE UN ENSEMBLE DÉDIÉ DE DÉPÔTS LOCAUX, DISTANTS ET VIRTUELS

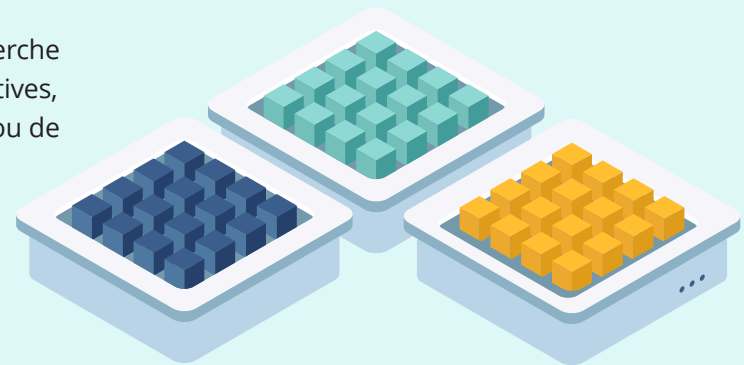
Fournissez à chaque équipe ou chaque « projet » un ensemble dédié de dépôts à utiliser pour ce développement spécifique. En outre, la solution de gestion des artefacts que vous utilisez doit disposer d'une entité de gestion permettant de regrouper les ressources telles que les dépôts, les builds, les bundles de version et les pipelines. Cette fonctionnalité facilite la délégation des ressources à une équipe spécifique.

EN QUOI EST-CE IMPORTANT ?

On considère cette approche comme une bonne pratique à plusieurs titres : tout d'abord, si les entreprises ont d'instinct tendance à conserver tous leurs artefacts au sein d'un même système, il est plus intéressant pour certaines équipes d'avoir accès uniquement à un sous-ensemble prédéfini d'artefacts, à des fins de simplicité et de sécurité.

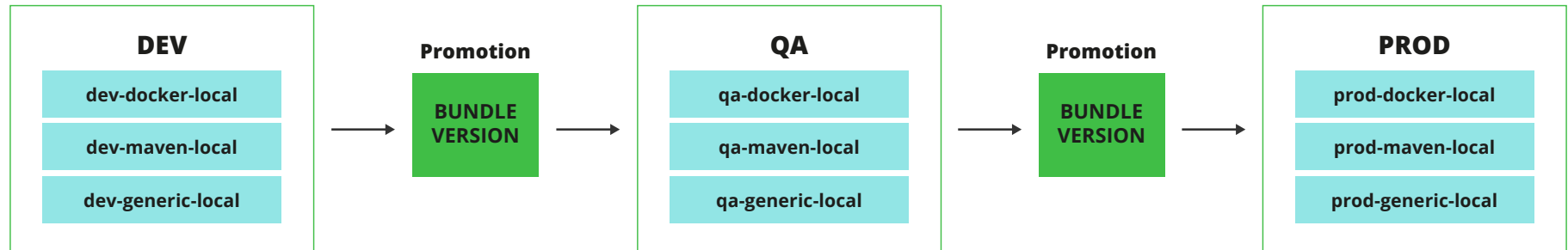
Par ailleurs, affecter un jeu de dépôts distinct à chaque équipe facilite la recherche et l'accès aux composants indispensables à l'exécution de leurs tâches respectives, favorise le partage de composants entre les équipes et permet de dupliquer ou de fédérer les dépôts en cas de nécessité.

Manipuler des dépôts de grande envergure devient rapidement complexe. En fournissant aux équipes des dépôts dédiés, vous délestez les administrateurs centraux d'une partie des tâches de gestion et de maintenance. Par exemple, un administrateur central ne sait pas ce qu'il peut supprimer ou mettre à jour sans impacter le travail d'une équipe donnée. Enfin, cette décision est intéressante du point de vue financier : elle permet d'allouer des budgets précis et de suivre plus efficacement les coûts liés à chaque équipe.



 **EN SAVOIR PLUS**
Dépôts fédérés JFrog

6. PROMOUVEZ VOS ARTEFACTS — D'UN ENVIRONNEMENT À L'AUTRE AU SEIN DU SDLC — SANS LES RECRÉER SYSTÉMATIQUEMENT



Le cycle de vie du développement logiciel (SDLC) se décompose en plusieurs étapes, qui vont de la planification et de la conception initiales au déploiement et à la maintenance, en passant par le codage et les tests. Le transfert des artefacts d'un environnement à l'autre est un aspect essentiel du cycle de vie applicatif. Pour que ce processus soit aussi efficace et efficient que possible, on conseille de promouvoir les artefacts plutôt que de les recréer à chaque phase du SDLC.

EN QUOI EST-CE IMPORTANT ?

La promotion des artefacts tout au long du cycle de vie applicatif garantit l'homogénéité, la stabilité et la traçabilité des composants tout en économisant des ressources et en évitant les pertes de temps. En promouvant les artefacts au lieu de les recréer, les développeurs savent que le code qu'ils testent sera identique à la version déployée, tout en assurant un suivi rigoureux du degré de maturité du logiciel. Les analyses de sécurité sont exécutées dans le cadre du processus de promotion. Les refus et les autorisations de promotion s'appuient sur les résultats de ces analyses. Vous garantisiez ainsi la sécurité optimale de vos builds avant de passer à l'étape suivante du SDLC. Cette approche réduit le risque d'erreur, améliore l'assurance qualité, augmente la productivité et réduit les délais de lancement : un atout précieux pour toutes les équipes de développement.



EN SAVOIR PLUS

[Comment fonctionne la promotion des builds ?](#)

7. STOCKEZ ET GÉREZ LES MÉTADONNÉES ASSOCIÉES À CHAQUE ARTEFACT AU MÊME ENDROIT QUE CELUI-CI

Pour rappel, on appelle métadonnées les attributs complémentaires associés à un artefact. Ces attributs offrent aux administrateurs le moyen d'organiser leurs artefacts plus efficacement. Intéressons-nous aux différents types de métadonnées :

1. Métadonnées générales

- Nom de l'artefact : nom de l'artefact
- Version de l'artefact : numéro de version de l'artefact
- Type d'artefact : type de format de l'artefact (par ex. JAR, WAR, image Docker)
- Taille de l'artefact : taille du fichier d'artefact en octets
- Checksum de l'artefact : valeur checksum (par ex., MD5, SHA1, SHA256) du fichier d'artefact

2. Métadonnées liées aux builds

- Nom de la build : nom de la build associée à l'artefact
- Numéro de la build : numéro ou identifiant de la build
- Horodatage de la build : horodatage de création de la build

3. Métadonnées de dépendances :

informations relatives aux dépendances de l'artefact, y compris leur nom et leur version

4. Métadonnées personnalisées :

il est possible de définir des métadonnées supplémentaires spécifiques au projet et de les associer aux artefacts

EN QUOI EST-CE IMPORTANT ?

Les métadonnées jouent un rôle crucial dans le développement logiciel. Elles permettent aux entreprises de comprendre l'historique d'un artefact ou d'une build et de suivre son parcours individuel (en termes de validation par exemple) à mesure que le logiciel progresse vers la publication. Les métadonnées sont également déterminantes du point de vue de l'analyse des modifications éventuelles de certaines propriétés d'un package. En effet, ces modifications révèlent parfois des failles de sécurité. La capture détaillée des métadonnées est essentielle pour une bonne traçabilité logicielle, et peut même servir à déclencher des workflows, des webhooks ou des plugins.



EN SAVOIR PLUS

[Collecte et gestion des métadonnées de vos fichiers binaires à l'aide des informations de build](#)

8. CRÉEZ DES POLITIQUES D'IDENTIFICATION ET D'AUTORISATION D'ACCÈS À UN DÉPÔT DONNÉ ASSORTIES D'ACTIONS PRÉCISES

Les entreprises doivent mettre en place des règles et des politiques claires en vue de contrôler qui parmi les individus et les systèmes peut accéder aux artefacts logiciels et à quel moment. Votre solution de gestion des artefacts doit être en mesure d'appliquer ces politiques en utilisant un contrôle d'accès basé sur les rôles (RBAC) ou des intégrations à d'autres solutions de gestion des identités.



EN QUOI EST-CE IMPORTANT ?

Les artefacts logiciels sont l'essence du développement logiciel. Comme les packages renferment du code et des informations propriétaires, une grande partie de la gestion des artefacts consiste à s'assurer qu'aucun profil non autorisé n'est en mesure d'accéder aux ressources.



EN SAVOIR PLUS

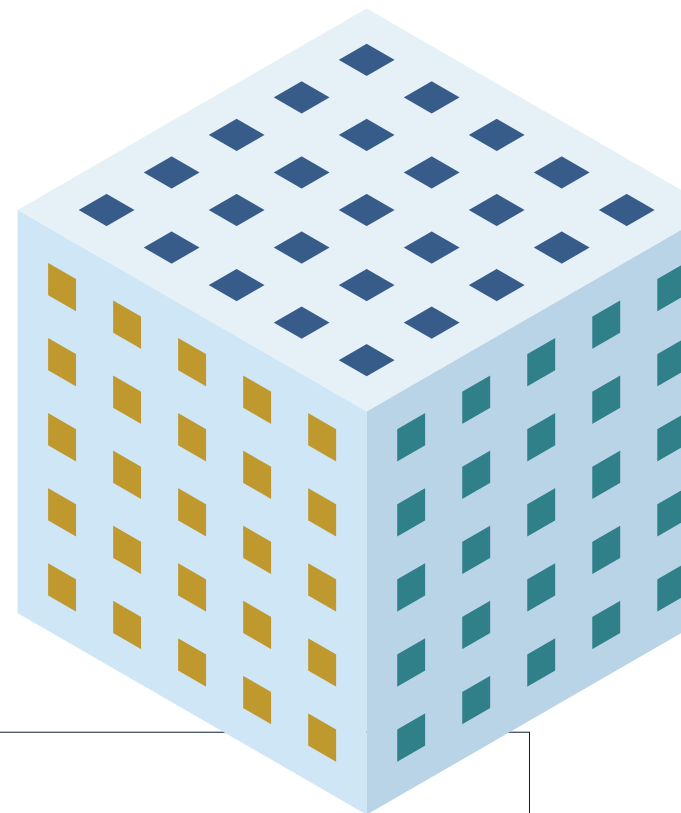
[Gérez les rôles et les membres d'un projet avec JFrog](#)

9. DÉFINISSEZ DES ROUTINES DE NETTOYAGE POUR AMÉLIORER L'HYGIÈNE ET LES PERFORMANCES DES DÉPÔTS

Le nombre de fichiers binaires produits aujourd'hui dépasse l'entendement. Plus ces fichiers s'accumulent, plus ils requièrent un volume de stockage important. Le nettoyage des volumes de stockage couvre différentes tâches : il peut s'agir de supprimer directement les fichiers binaires à partir du gestionnaire de dépôts, mais dans bien des cas, le nettoyage consiste à archiver les fichiers binaires vers une solution de stockage moins coûteuse (en particulier dans les secteurs très réglementés : dans ce cas, on déplace les fichiers vers la solution d'archivage avant de les supprimer dans le gestionnaire de dépôts).

EN QUOI EST-CE IMPORTANT ?

Tout comme dans le monde physique, si l'on ne cesse d'accumuler des ressources sans les organiser ni les gérer, elles finissent par poser de graves problèmes. Les politiques de nettoyage applicables aux dépôts d'artefacts sont importantes parce qu'elles garantissent que les artefacts utilisés sont sûrs et à jour. La mise en place de ces politiques contribue à préserver l'intégrité du dépôt en évitant d'y laisser des artefacts obsolètes, vulnérables ou contaminés. En éliminant les artefacts qui ne sont plus utiles, les politiques de nettoyage réduisent le volume global du dépôt. Non seulement ce dernier est mieux organisé, plus efficace, mais les coûts liés à sa maintenance s'en trouvent également réduits.



EN SAVOIR PLUS

Fondamentaux des stratégies de nettoyage sur-mesure

CONCLUSION

En intégrant ces bonnes pratiques à vos processus de gestion des artefacts, vous mettez toutes les chances de votre côté. Vous bénéficiez de workflows rationalisés, d'une collaboration plus fluide et de résultats optimisés. Rappelez-vous : la gestion des artefacts n'est pas qu'une question d'organisation, mais aussi d'optimisation et de sécurisation de l'ensemble de votre cycle de développement.

Nous espérons que cet e-book vous aura fourni les connaissances et les outils nécessaires pour révolutionner vos pratiques de gestion des artefacts. Faites-les vôtres, adaptez-les à vos besoins spécifiques, et efforcez-vous de les améliorer constamment. Fort d'un socle robuste pour la gestion de vos artefacts, vous serez équipé pour relever les défis liés aux conditions actuelles du secteur et obtiendrez des résultats remarquables.

Pour plus de détails sur la gestion des artefacts, consultez notre livre blanc :

[Gestion des artefacts : les bonnes pratiques pour la réussite de votre entreprise](#)

À PROPOS DE JFROG

JFrog s'est donné pour mission de créer un monde où les logiciels sont distribués sans aucune friction, de l'équipe de développement aux terminaux utilisateur. Inspirée par la vision de « logiciels liquides », la plateforme de chaîne d'approvisionnement logicielle JFrog forme un système unique d'enregistrement au service des entreprises pour une construction, une gestion et une distribution rapide et sécurisée de leurs logiciels, tout en garantissant leur disponibilité, leur traçabilité et leur inviolabilité. Les fonctionnalités de sécurité intégrées permettent également d'identifier et de neutraliser les menaces et les vulnérabilités. Hybride, universelle et multicloud, la plateforme de JFrog est disponible à la fois sous forme de services autohébergés ou de services SaaS compatibles avec tous les grands fournisseurs de services cloud. Des millions d'utilisateurs et plus de 7 000 clients du monde entier, dont une majorité des entreprises classées au Fortune 100, font confiance à JFrog pour soutenir leur transformation numérique en toute sécurité.

