

Slide 2 Logiciel de contrôle de version

Les systèmes de contrôle de version sont une catégorie d'outils de développement qui aident une équipe de développement à gérer les changements apportés au code source au fil du temps. Les logiciels de contrôle de version gardent une trace de chaque changement apporté au code dans un type spécial de base de données.

Il existe deux types principaux de logiciels de gestion de versions.

- **Les logiciels centralisés** : un serveur conserve les anciennes versions des fichiers et les développeurs s'y connectent pour prendre connaissance des fichiers qui ont été modifiés par d'autres personnes et pour y envoyer leurs modifications.
- **Les logiciels distribués** : il n'y a pas de serveur, chacun possède l'historique de l'évolution de chacun des fichiers. Les développeurs transmettent directement entre eux les modifications, à la façon du *peer-to-peer*.

Slide 10:

Une branche est essentiellement un ensemble unique de modifications de code avec un nom unique. Chaque référentiel (repository) peut avoir une ou plusieurs branches. La branche principale — celle où tous les changements finissent par être fusionnés de nouveau, et est appelée master. Il s'agit de la version officielle de travail de votre projet.

Le diagramme visualise un référentiel avec deux lignes de développement isolées, l'une pour une petite fonctionnalité, et l'autre pour une fonction plus longue. En les développant dans des branches, il est non seulement possible de travailler sur les deux en parallèle, mais il maintient également la branche principale libre de code douteux

Slide 14:

Et là intervient la notion de CI/CD. CI et CD sont deux acronymes fréquemment utilisés dans les pratiques modernes de développement et DevOps. CI signifie intégration continue, une pratique exemplaire fondamentale de DevOps où les développeurs fusionnent fréquemment les modifications de code en un référentiel central où les builds et les tests automatisés s'exécutent. Le CD peut signifier une livraison continue ou un déploiement continu.

Slide 18:

Git ou GitHub. C'est la même chose ? Sinon, sont-ils connectés d'une façon ou d'une autre ? Ou, comme Java et JavaScript, la connexion n'est-elle que superficielle ? La vérité est que Git et GitHub sont connectés beaucoup plus étroitement que Java et JavaScript - mais avec quelques différences clés les distinguent.

Conclusion

Git est très populaire, largement utilisé, et accepté comme un système de contrôle de version standard par la grande majorité au sein de la communauté du développeur. Il est beaucoup plus facile de tirer parti des bibliothèques tierces (third party libraries) et d'encourager d'autres développeurs à fourcher votre code open source à l'aide de Git. Le grand nombre d'utilisateurs de Git facilite la résolution des problèmes et demande de l'aide extérieure à l'aide de forums en ligne. (Large community).

