
Introduction à Git et GitHub

les versions control
comme git permettent le
“control Z sur les
stéroïdes” et la
collaboration accrue avec
les branches

Soutien aux équipes
distribuées : Git et
GitHub permettent aux
équipes de contribuer
depuis divers lieux sans
conflits majeurs.

Favorise
l'expérimentation : les
branches permettent de
développer des
fonctionnalités sans
perturber le code en
production.

Les Fondamentaux de Git



Architecture basée sur les snapshots : chaque commit capture l'état complet du projet.

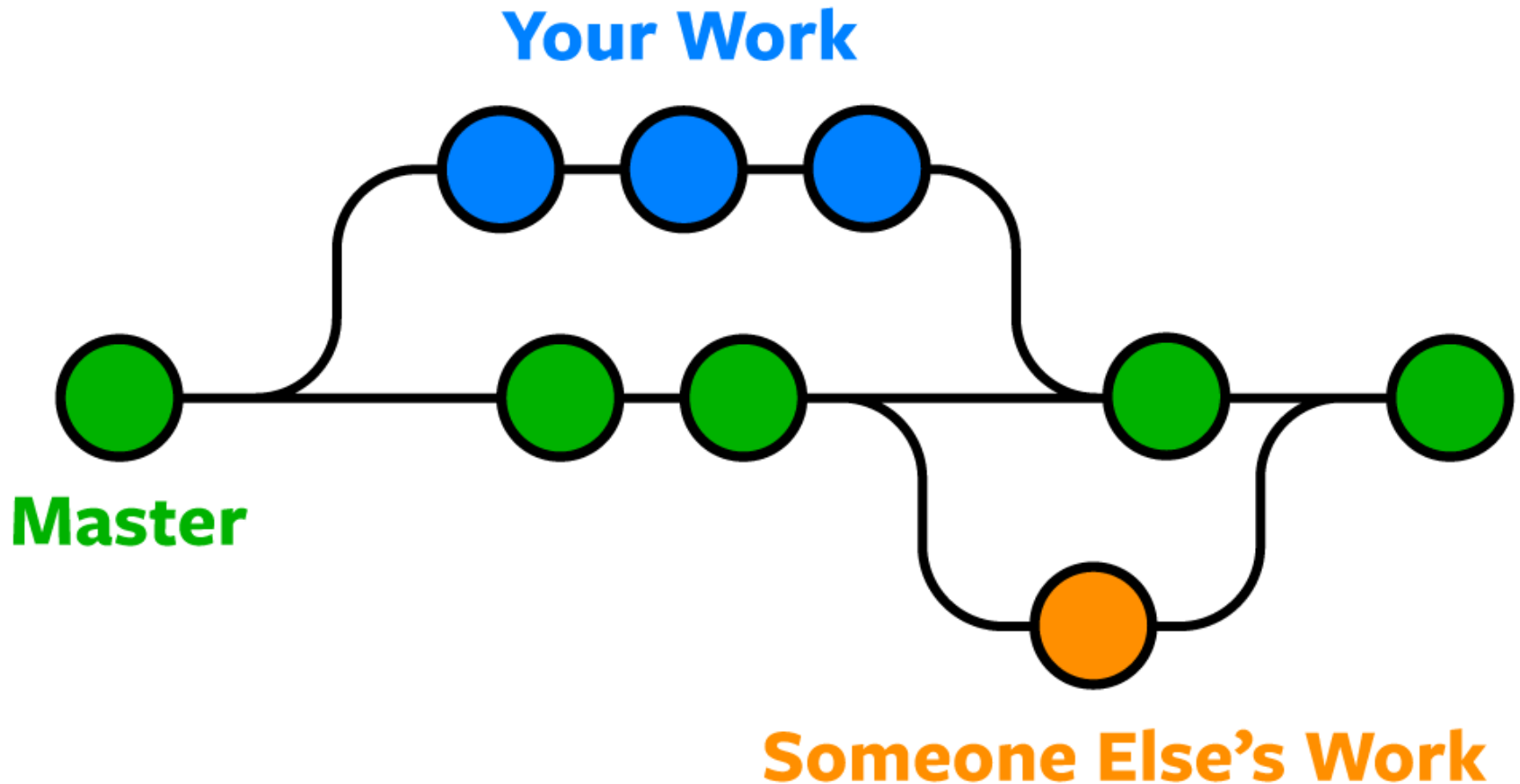


Efficacité des branches : branches légères et efficaces pour un développement parallèle.

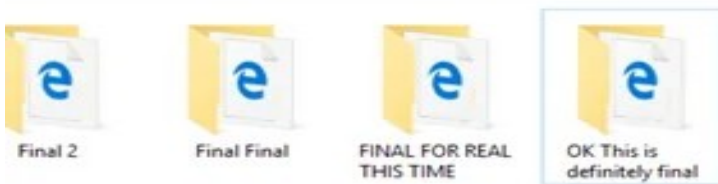


Adoption généralisée : plus de 87% du marché mondial utilise Git.

Comment le système de branche fonctionne



Git et git hub en bref



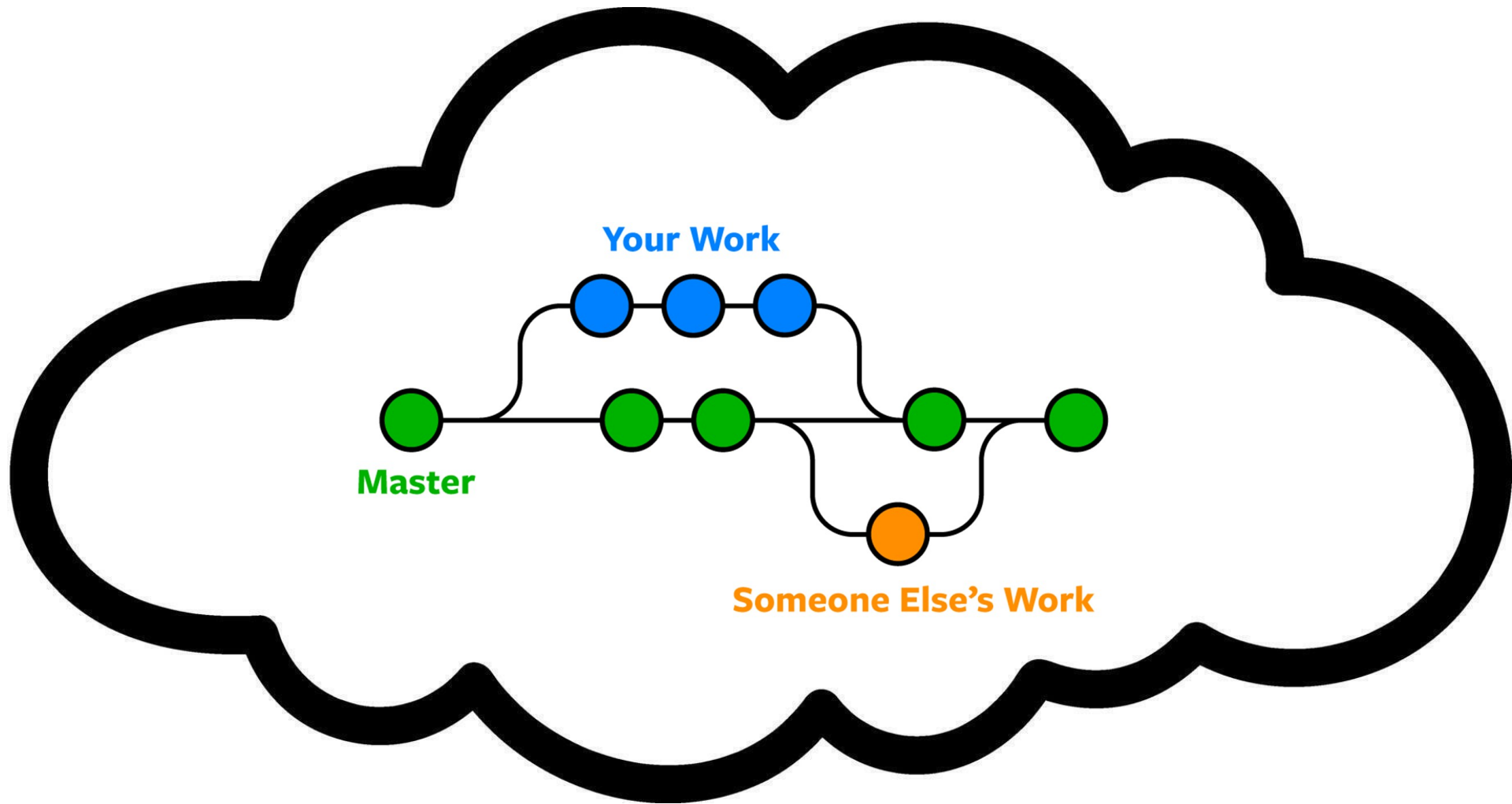
GitHub comme plateforme collaborative

Communauté mondiale : plus de 100 millions de développeurs collaborent via GitHub.

Outils de collaboration robustes : pull requests, gestion des bugs, revues de code.

Utilisation en entreprise : plus de 90% des entreprises du Fortune 100 utilisent GitHub.

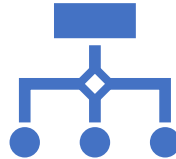
Git hub en bref



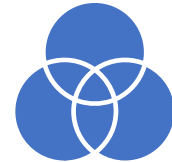
Bonnes pratiques Git & GitHub



Mettre des messages clairs à chaque commits pour pouvoir te retrouver dans tes snapshots



Utiliser des branches de fonctionnalité : permet un développement modulaire.



Ne jamais push directement dans main, pusher dans sa propre branche et utiliser les pull requests .

Conclusion & Questions



Git : contrôle de version basé sur des snapshots



GitHub : Manière de mettre dans le cloud un projet git



Bonne pratique : JAMAIS PUSH DIRECT DANS MAIN.