



ATELIER 1

GROUPE 3

SELMI SOUHEIL
MARIEM BEN SALEM
RAFIK BOUCHAOUA
KAMEL GARMASSI
ANIS SAADOULI
ALA FRAOUA
AZIZ MASSAOUDI

QU'EST CE QU'UN SYSTEME DE CONTROLE DE VERSION?



Un système de contrôle de version (VCS) est un outil logiciel qui permet de suivre et de gérer les modifications apportées à un ensemble de fichiers au fil du temps. Il s'agit d'un outil essentiel pour les développeurs de logiciels



quels sont les types des systèmes de contrôle de version?

Les deux principaux types de systèmes de contrôle de version (VCS) sont :

Systèmes de contrôle de version centralisés (CVCS)

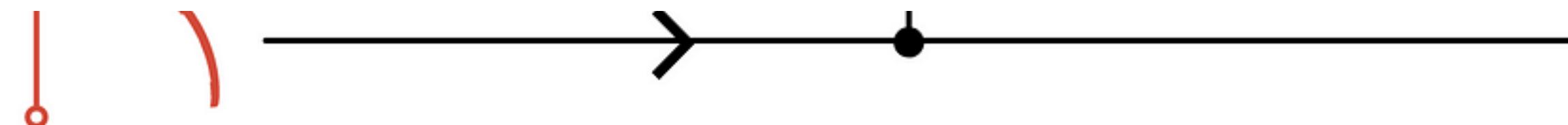
- Privilégiez-les si vous avez besoin d'une structure de gestion claire, d'un contrôle centralisé et que vous travaillez avec des équipes hiérarchisées.

Systèmes de contrôle de version distribués (DVCS)

- Optez pour les DVCS si vous recherchez une flexibilité pour le travail hors ligne, une collaboration plus décentralisée et des workflows parallèles.



Fonctionnalité	Systèmes de contrôle de version centralisés (CVCS)	Systèmes de contrôle de version distribués (DVCS)
Stockage du référentiel	Centralisé sur un serveur	Distribué sur chaque ordinateur de développeur
Accès aux fichiers	Nécessite une connexion au serveur central	Accès possible hors ligne
Collaboration	Structure hiérarchique, workflow plus structuré	Plus flexible pour la collaboration décentralisée
Gestion des conflits	Gérée par le serveur central	Nécessite une résolution manuelle des conflits de fusion
Scalabilité	Mieux adapté aux grands projets avec des équipes hiérarchisées	Peut devenir complexe pour les très grands projets
Exemples	Subversion, CVS, Perforce	Git, Mercurial, Bazaar



qui utilise le système de contrôle de version?



les systèmes de contrôle de version sont des outils essentiels pour toute personne ou équipe qui travaille avec des fichiers et a besoin de les gérer, de les suivre et de collaborer efficacement.

Example :

Développeurs logiciels

Équipes DevOps

Concepteurs et graphistes

c'est quoi git?quels sont les avantages et les inconvénients lors de l'utilisation de l'outil GIT?

Git est un outil puissant et flexible pour le développement logiciel qui offre de nombreux avantages en matière de suivi des modifications, de collaboration et de gestion de versions.

Git ne stocke pas les fichiers dans un référentiel centralisé. Chaque développeur possède une copie complète du référentiel sur son ordinateur local, ce qui le rend plus robuste et flexible.

Avantages de l'utilisation de Git :

Flexibilité et robustesse

Facilité d'utilisation

Gestion des branches et des workflows

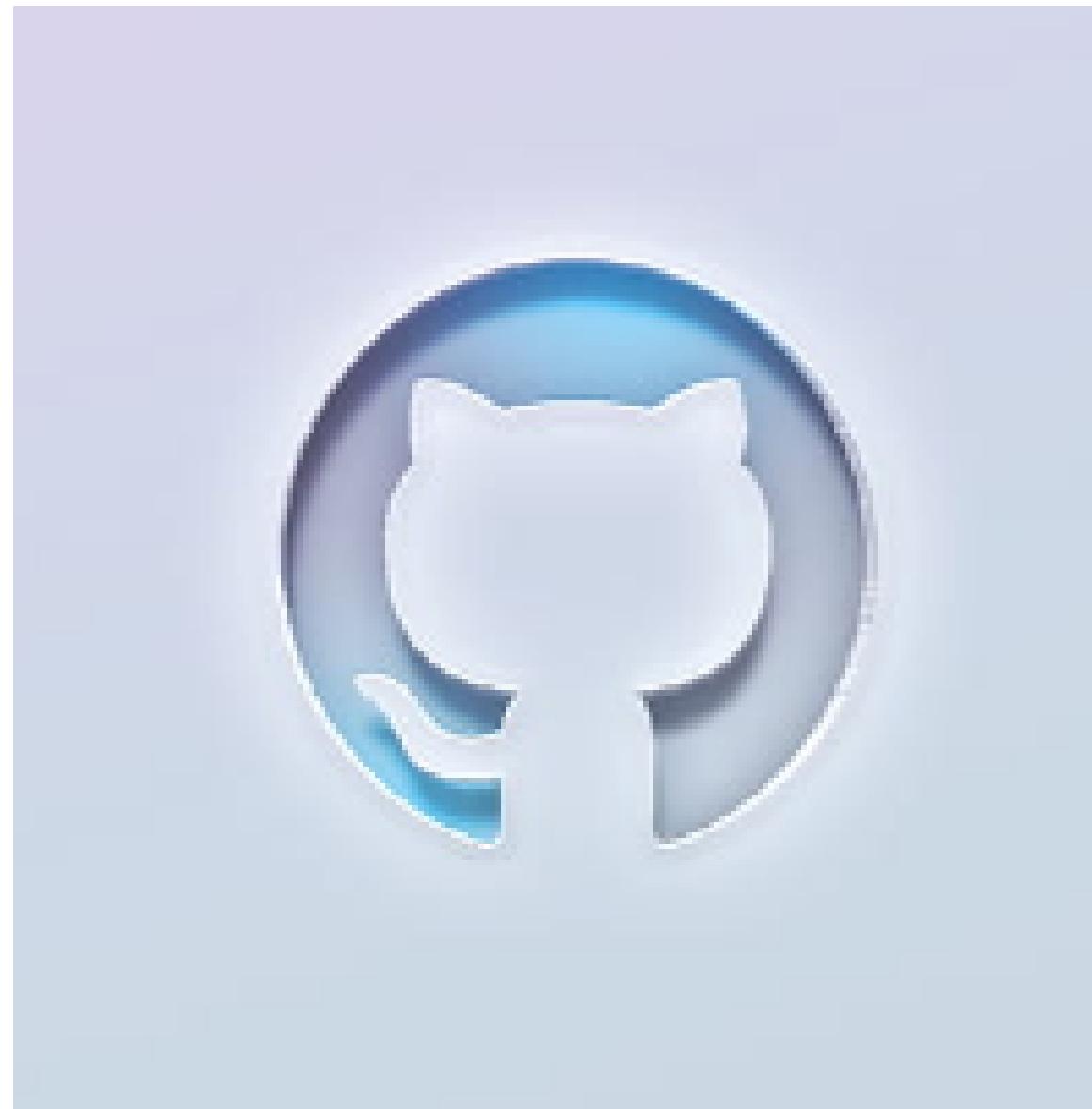
Inconvénients de l'utilisation de Git :

Courbe d'apprentissage

Gestion des conflits de fusion

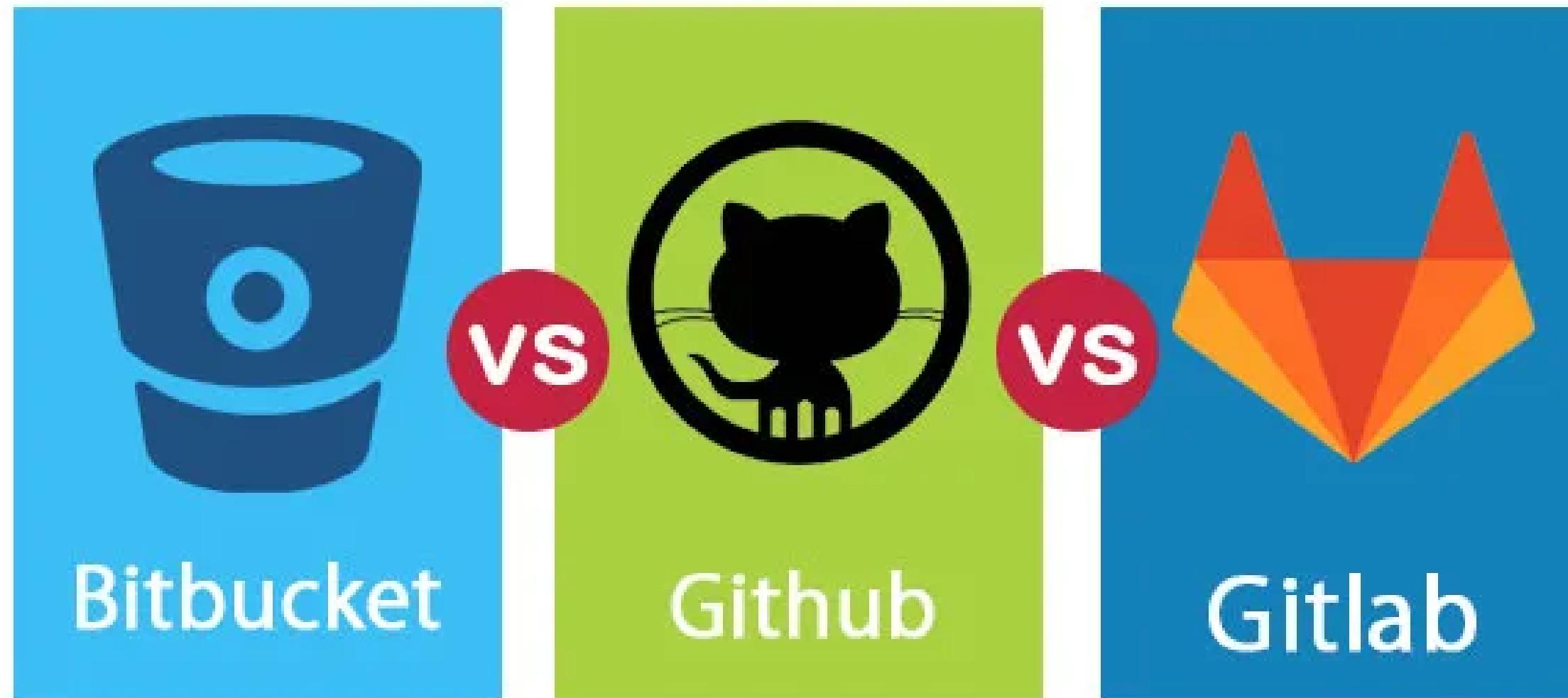
Dépendance aux outils en ligne

Write to Anis Saadouli

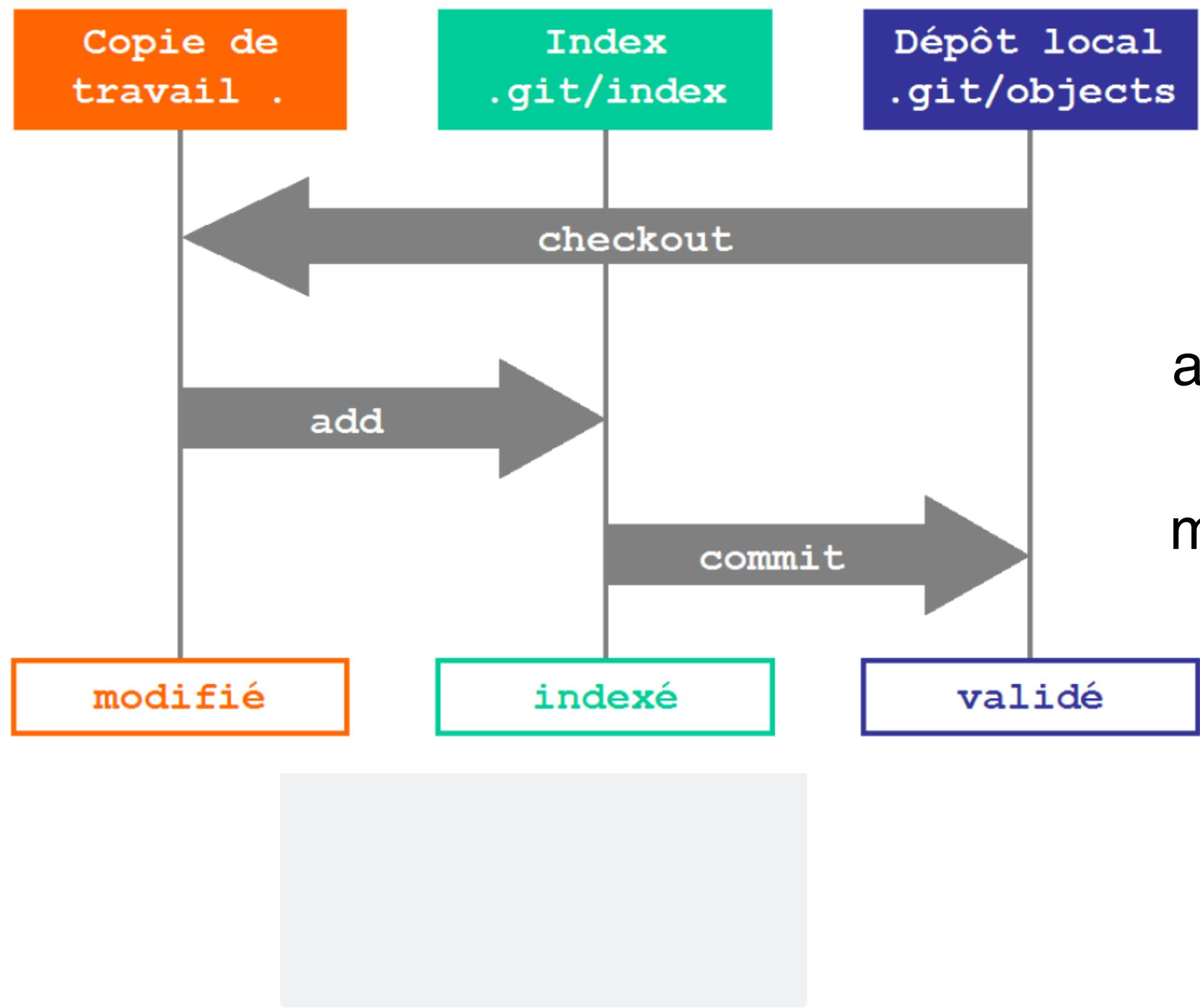


quelle est la différence entre github, gitlab et bittbucket?

GitHub est orienté vers l'open source, GitLab propose un cycle de vie complet du logiciel avec intégration continue, tandis que Bitbucket, intégré à d'autres outils Atlassian, supporte Git et Mercurial.



Les trois états d'un fichier



Comment assurer le travail collaboratif avec GIT?

Pour garantir un travail collaboratif efficace avec Git, utilisez des branches distinctes, des pull requests pour la révision du code et maintenez une communication régulière entre les membres de l'équipe.

Merci de votre
attention!

