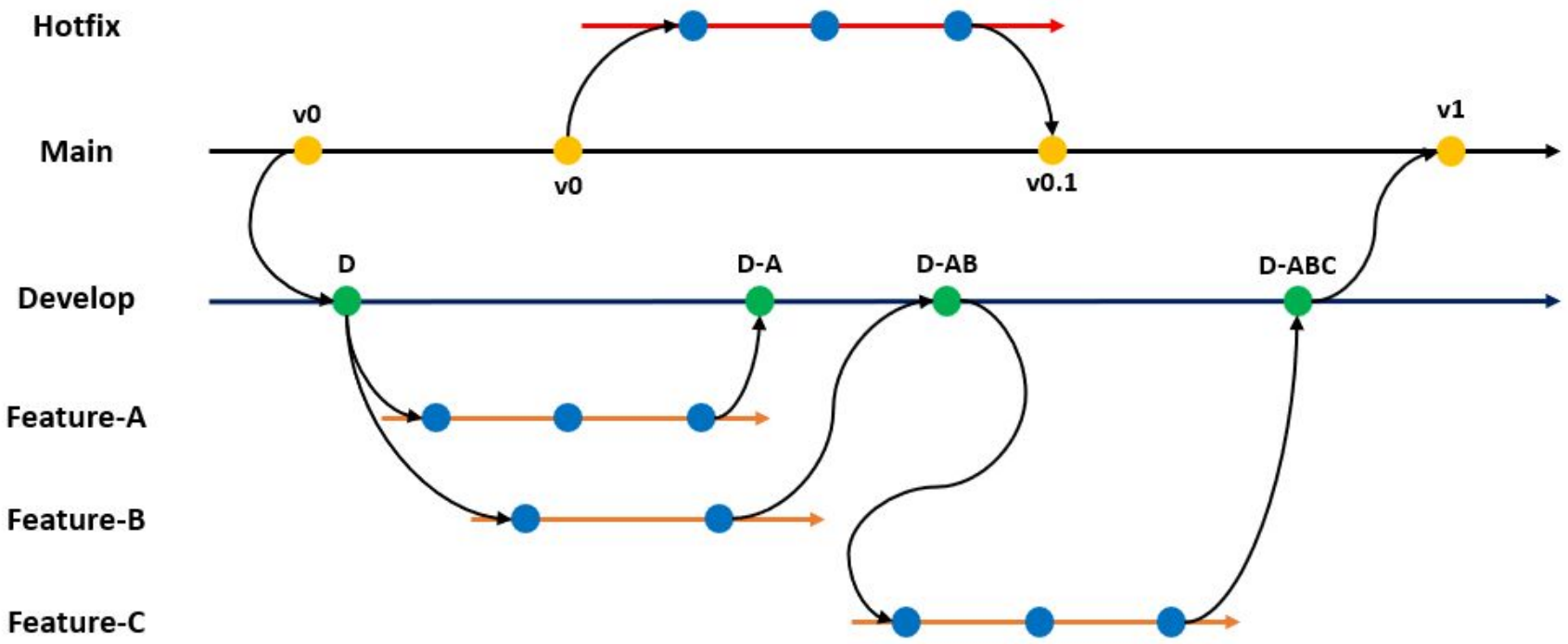
Outillage GIT

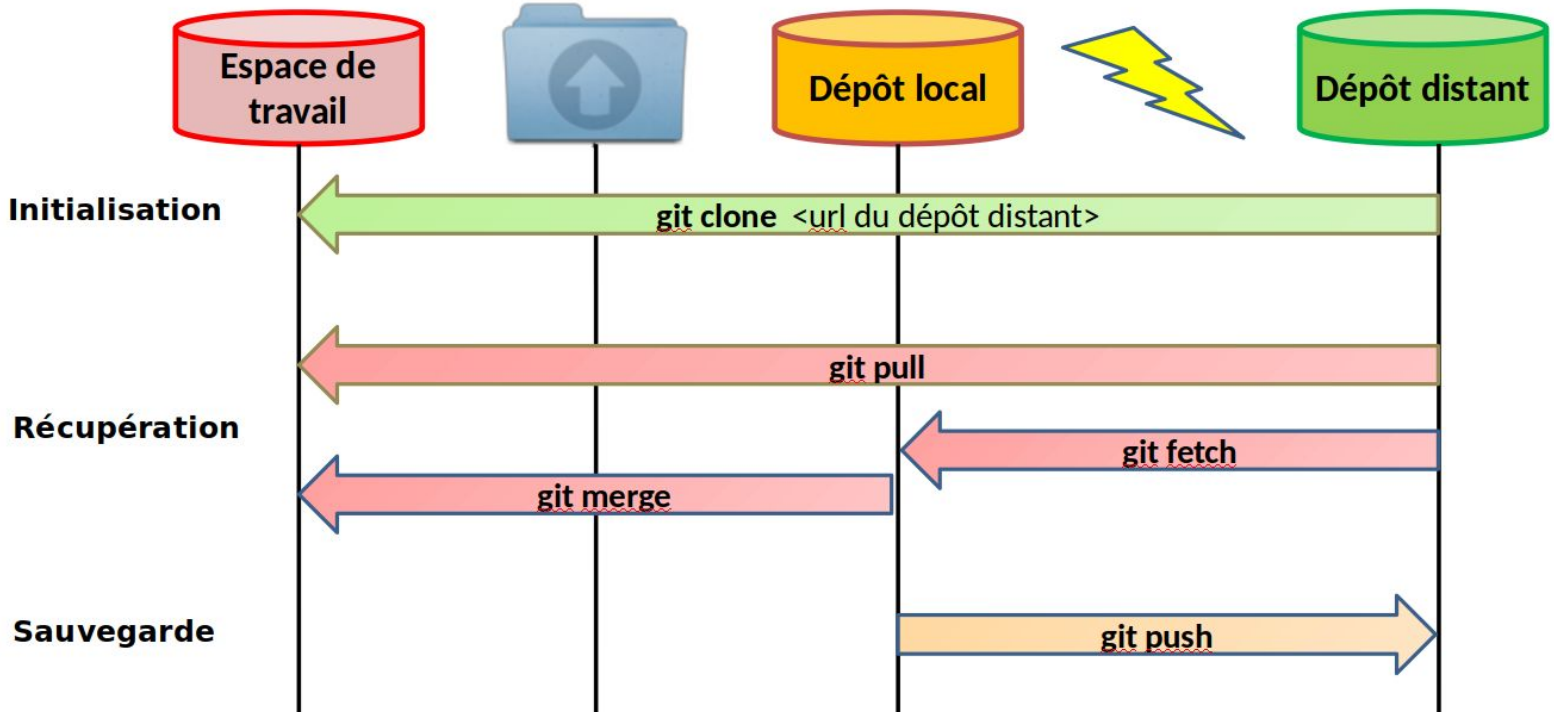
ou comment versionner et
travailler ensemble.

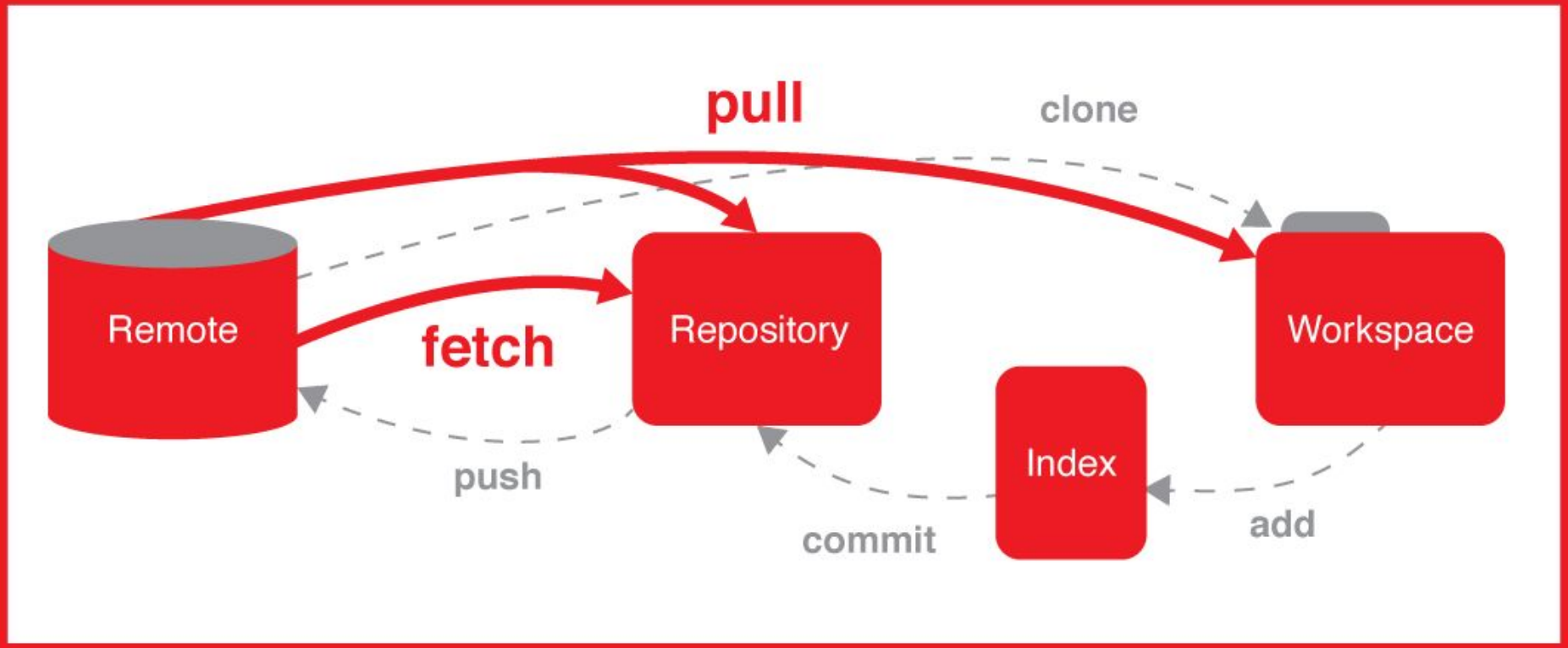
par Yann Guillerm yann.gv29@gmail.com

A quoi ça sert ?

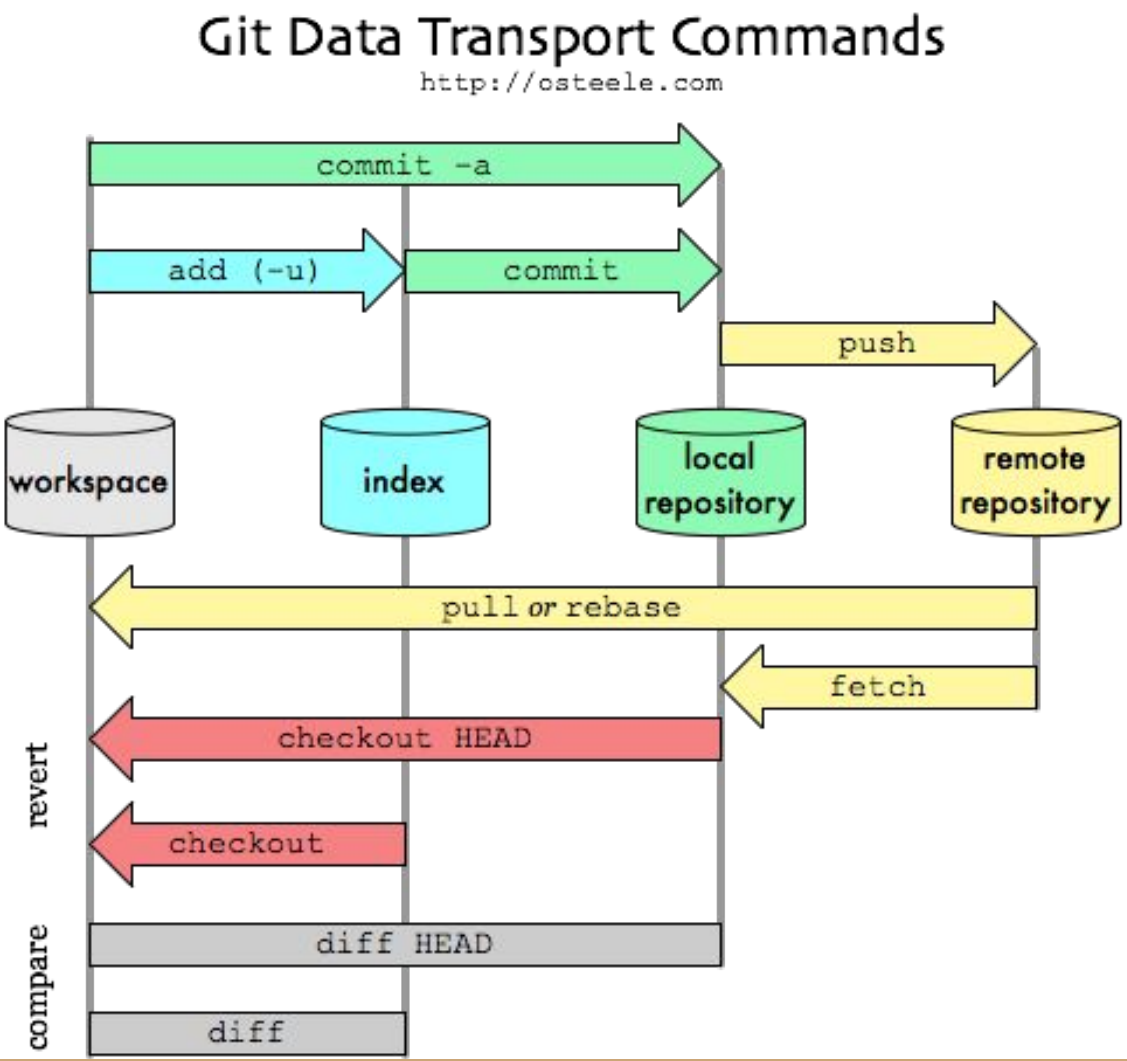


Ou ? ou ça ?


Espace de travail
Dépôt local
Dépôt distant
Initialisation
git clone <url du dépôt distant>
git pull
Récupération
git fetch
git merge
Sauvegarde
git push


pull
clone
Remote
fetch
Repository
Workspace
push
Index
commit
add

The Big Picture



les commandes de base

git config -global <user.name ou user.email ou ...> <votre nom ou votre email>
git init
git clone
git add *
git commit -m “....” ou -am “...” pour faire le add
git diff <nom du fichier> ou <brancheA> <BrancheB>
git tag

git push / git pull / git fetch

Les branches , le distant, le oupss i did it again

git branch <nomDeBranche> ou git checkout -b <nomDeBranche>
git checkout <nomDeBranche>

git remote add

git reset et git reset -hard <idCommit>

Do not forget and Do / Don’t

.gitignore

le répertoire .git

le tooling autour de git

on ne versionne pas :

- un fichier contenant un mot de passe ou une API_KEY, ou un Secret, ou une clé SSH, ….
- les fichiers générés
- les dépendances

les gros mots

versioning : le fait d'arrêter un état d’un livrable et de pouvoir y revenir.

CI : intégration continue

CI CD : intégration continue & déploiement continu

Tag : un nom qui a du sens fonctionnellement pour une version

git-flow : à la fois la représentation des branches et aussi une façon de travailler le branching.

références

a faire en TP : https://betterprogramming.pub/git-branching-strategy-for-better-team-collaboration-aacb5f235d05

cheat sheet : https://gist.github.com/aquelito/8596717

explication des commandes : https://www.commentcoder.com/commandes-git/

Tuto orléans : https://www.univ-orleans.fr/iut-orleans/informatique/intra/tuto/git-tutoriel/travailler-avec-git.html

encore des commandes

git stash

git stash pop

git stash drop

git log - - all - - graph