

Aide Mémoire GIT

INSTALLATION

Outil Git + interphases graphique de base git gui et gitk.

Linux: sudo apt-install git-all

Mac/Windows: <https://git-scm.com/download>

CONFIGURATION

Configurer les informations de l'utilisateur pour le dépôt courant ou pour tous avec l'option --global

git config --global user.name "[nom]"

Définit le nom de l'utilisateur

git config --global user.email [email]

Définit l'email de l'utilisateur

GESTION DES DÉPÔTS

Créer un dépôt local ou en utiliser un distant via son url.

git init [nom_dépôt]

Crée un dépôt local vide dans le dossier courant ou en créer un avec le nom spécifié.

git clone [url]

Duplique en local le dépôt git pointé par l'url.

git remote add [url]

Ajoute comme remote le dépôt pointé par l'url.

git branch -u [remote/branch] [branch]

Fait suivre une branche distante par une branche locale.

GÉRER LES MODIFICATIONS

Voir les changements, les sélectionner et les enregistrer.

git status

Liste tous changements apportés à l'espace de travail.

git checkout [fichier]

Supprime les modification courantes du fichier.

git add --patch [fichiers]

Ajoute les modifications d'un fichier a la zone d'index. Mode interactif avec --patch

git reset HEAD [fichiers]

Enlève les modifications de l'index mais les conserve dans le workspace.

git diff --staged [fichier]

Montre les modifications du fichier non indexé, pour un fichier indexé, ajouter l'option --staged.

git commit --amend -m "[message]"

Enregistre les modifications présentes dans la zone d'index dans l'historique du dépôt. --amend permet de modifier le commit précédent.

git rm --cached [fichier]

Supprime du suivi de git le fichier. Ce fichier est également supprimé du WD sauf avec --cached

git mv [src] [dest]

Déplace/Renomme un fichier dans le WD tout en conservant le suivi de son historique.

SYNCHRONISER LES CHANGEMENTS

Synchroniser le dépôt local et distant. sans paramètres, cela revient a origin master

git fetch [remote]

Met à jour les informations locales concernant le serveur distant.

git pull [remote] [branch]

Met à jour son dépôt local avec les commits présents sur le serveur distant.

git push [remote] [branch]

Envoie les commits en local sur le serveur distant.

GESTION DE L'HISTORIQUE

Voir et naviguer dans l'historique des commits

git log [-n] --graph

Montre l'historique des commits pour la branche courante. -n pour limiter aux n derniers commits.

git diff [tag|sha1] [tag|sha1] Montre les

différences de contenu entre deux commits.

Aide Mémoire GIT

git show [tag|sha1]

Montre les modifications enregistrés lors du commit spécifié.

git checkout [tag|sha1|branche]

Se déplacer dans l'historique sur le commit donné

git tag [nom_tag]

Créer un tag sur le commit courant

RÉÉCRIRE L'HISTORIQUE

Corriger des erreurs, regrouper ou modifier des commits.

git commit --amend

Ajoute au dernier commit le contenu de la zone d'index et change le message de commit.

git revert [commit]

Créer un nouveau commit qui est l'opposé du commit spécifié afin d'annuler ses effets.

git reset [commit]

Annule tous les commits après `[commit]`, en conservant les modifications localement.

git reset --hard [commit]

Supprime tout l'historique et les modifications effectuées après le commit spécifié.

git rebase -i HEAD~n

Réordonne, fusionne, supprime, modifie les n derniers commits locaux. A ne pas faire sur des commits poussés sur le remote.

TRAVAILLER EN BRANCHE

Créer une branche de travail

git branch -r -a

Liste toutes les branches locales dans le dépôt courant. -r pour les branches distantes. -a pour tout

git branch [nom_branche]

Crée une nouvelle branche.

git branch -d [nom_branche]

Supprime la branche locale.

git checkout -b [nom_branche]

Change de branche et se placer sur le dernier commit de celle-ci. -b pour en plus créer la branche.

git merge [autre_branche]

Combine dans la branche courante l'historique de la branche spécifiée via un commit de merge.

git rebase [autre_branche]

Déplace les commits de la branche courante sur la branche spécifiée.

git cherry-pick [-x] [commit]

Applique un commit à l'espace de travail. -x permet d'ajouter le message « cherry-picked from commit [sha1_commit] ».

METTRE DE CÔTÉ DES MODIFICATIONS

Mettre de côté des modifications temporairement.

git stash save "[message]"

Enregistre toutes les modifications courantes dans une pile temporairement.

git stash list

Liste toutes les modifications mises de côté.

git stash pop id

Appliquer les modifications à l'index id de la pile dans l'espace de travail. sinon id = 0

git stash drop id

Supprime les modifications à l'index id de la pile. sinon id = 0.

DEBUGGER SON CODE

Outils d'aide au debug.

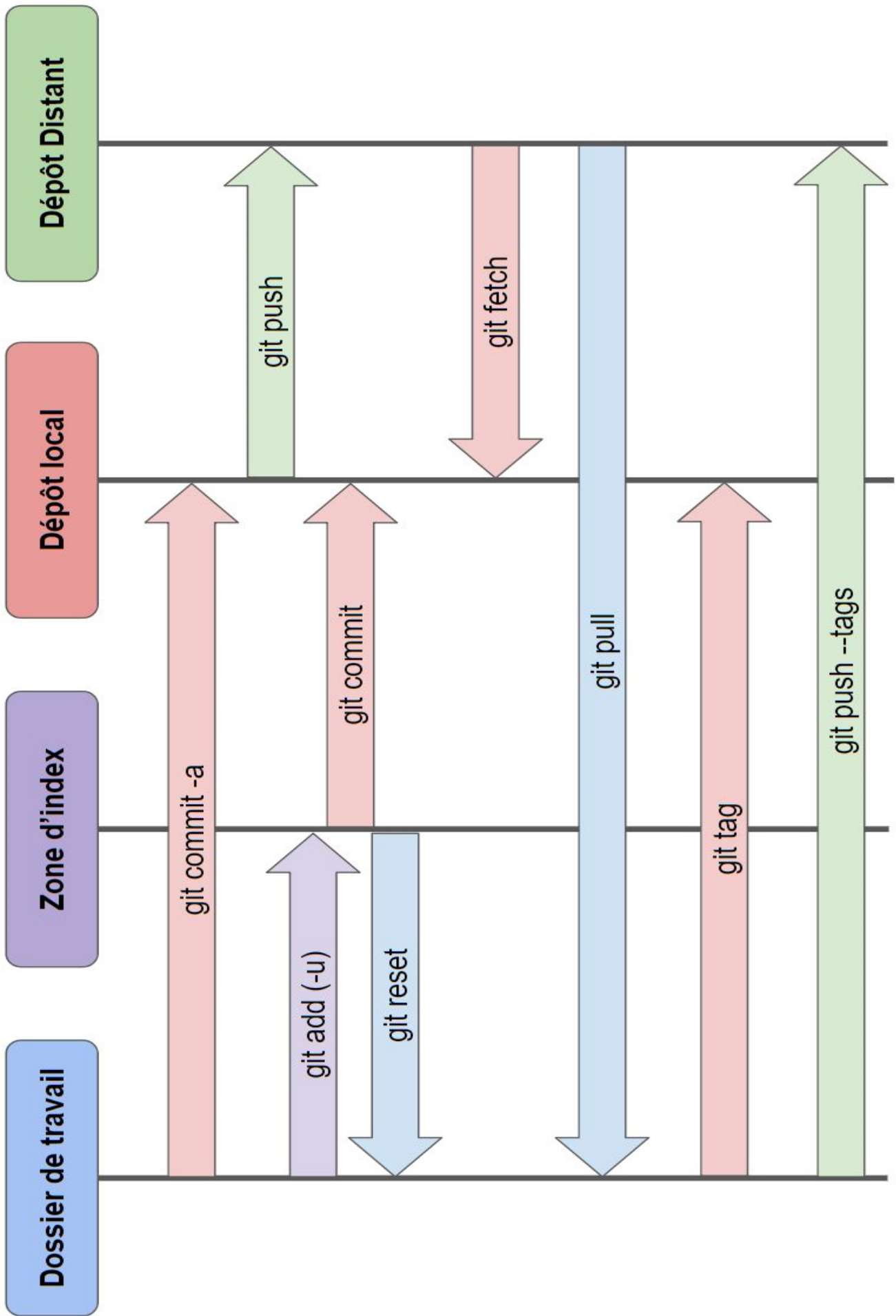
git blame -L [line_start],[line_stop] [file_name]

Affiche par qui et quand chaque ligne du fichier a été modifiée.

git bisept

Recherche par dichotomie l'origine d'un bug.

Aide Mémoire GIT



Aide Mémoire GIT

Notes: