

Initiation à Git

Initiation à Git

- ♦ **Présentation de git**
- ♦ **Les commandes git**
 - ♦ *Configuration initial :*
 - ♦ git config
 - ♦ *Travailler localement :*
 - ♦ Git init, git status, git add, git commit, git log, git commit, ...
 - ♦ *Travailler avec un serveur git :*
 - ♦ Aparté gitlab.liris.cnrs.fr
 - ♦ Git clone, git push, git pull et gestion des conflits
 - ♦ *Configuration local du repo*

Présentation de git

- ♦ **Outil très utile pour :**

- ♦ La gestion de version
- ♦ Permet “facilement” le travail collaboratif
- ♦ Sauvegarde de vos fichiers TEXTE vers un serveur distant

- ♦ **SUPPORTE MAL LES FICHIERS BINAIRES !**

- ♦ **Installation :**

- ♦ <https://git-scm.com/downloads>

Les commandes git

Configuration initiale

- **git config --global user.name "<your name>"**
- **git config --global user.email "<your@mail.com>"**
 - Définit quelques paramètres par défaut utiles pour plus tard

Les commandes git

Travailler localement

- ♦ **git init**

- ♦ Initialisation du dossier versionné

- ♦ **git status**

- ♦ Obtenir des informations concernant l'état du dossier

Les commandes git

Travailler localement

- **git add <filename(s)>**
 - Ajout d'un(des) fichier(s) au « staging »
- **git commit**
 - Appliquer les changements sur le(s) fichier(s) en « staging »
- **git commit <filename(s)>**
- **git commit <filename(s)> -m “Votre message”**
- **EVITER ! git commit -am “Votre message”**

Les commandes git

Travailler localement

- **git diff**

- Affiche toutes les différences avec le précédent commit

- **git diff <filename(s)>**

- Affiche toutes les différences avec le précédent commit
- UNIQUEMENT pour le(s) fichier(s)

Les commandes git

Travailler localement

- **git log**

- Consulter l'historique des commits

- **gitk --all &**

- Consulter l'historique des commits via une UI

Les commandes git

Travailler localement

- **git checkout <commit_ref>**
 - Revenir à un commit en particulier
- **git show <commit_ref:path_to_file>**
 - Afficher l'état d'un fichier à un commit donné

Les commandes git

Travailler localement

- **git checkout master**
 - Revenir à l'état courant
- **git checkout -- <filename>**
 - Annuler les modifications pas encore commitées

Les commandes git

Travailler localement

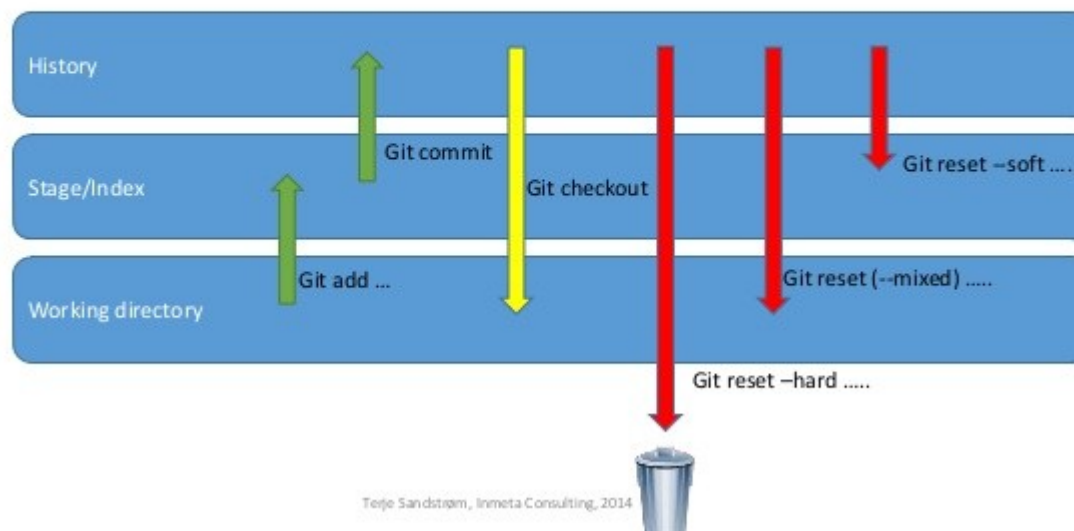
- **git rm --cache <filename(s)>**
 - Annuler un 'git add', i.e. retire le(s) fichier(s) retire du « staging »
 - NE SUPPRIME PAS LE FICHIER ;-)
- **git rm <filename(s)>**
 - Supprime COMPLÈTEMENT le(s) fichier(s)
 - À UTILISER AVEC PRÉCAUTIONS
- **git revert <commit_ref>**
 - Créer le commit inverse du dernier commit
 - ATTENTION ! Change l'historique dans git log !

Les commandes git

Travailler localement

- **git reset [--soft, --mixed, --hard] <commit_ref>**
 - Modifie (voire efface) les *n* derniers commits
 - A UTILISER AVEC PRÉCAUTION

Git tree movements visualized



Les commandes git

Travailler localement

- ♦ **Les branches git**

- ♦ Pouvoir développer, tester, etc., sans tout casser
- ♦ Ex : branche 'dev' et 'prod'

- ♦ **git branch <branch_name>**

- ♦ Créé la branche <branch_name>

- ♦ **git checkout <branch_name>**

- ♦ Permet de se "rendre" sur la branche choisie

- ♦ **git checkout -b <branch_name>**

- ♦ Créé la branche et s'y "rend" directement

Les commandes git

Travailler localement

- **git merge <branch_name>**

- Fusionne la branche courante avec la branche <branch_name>

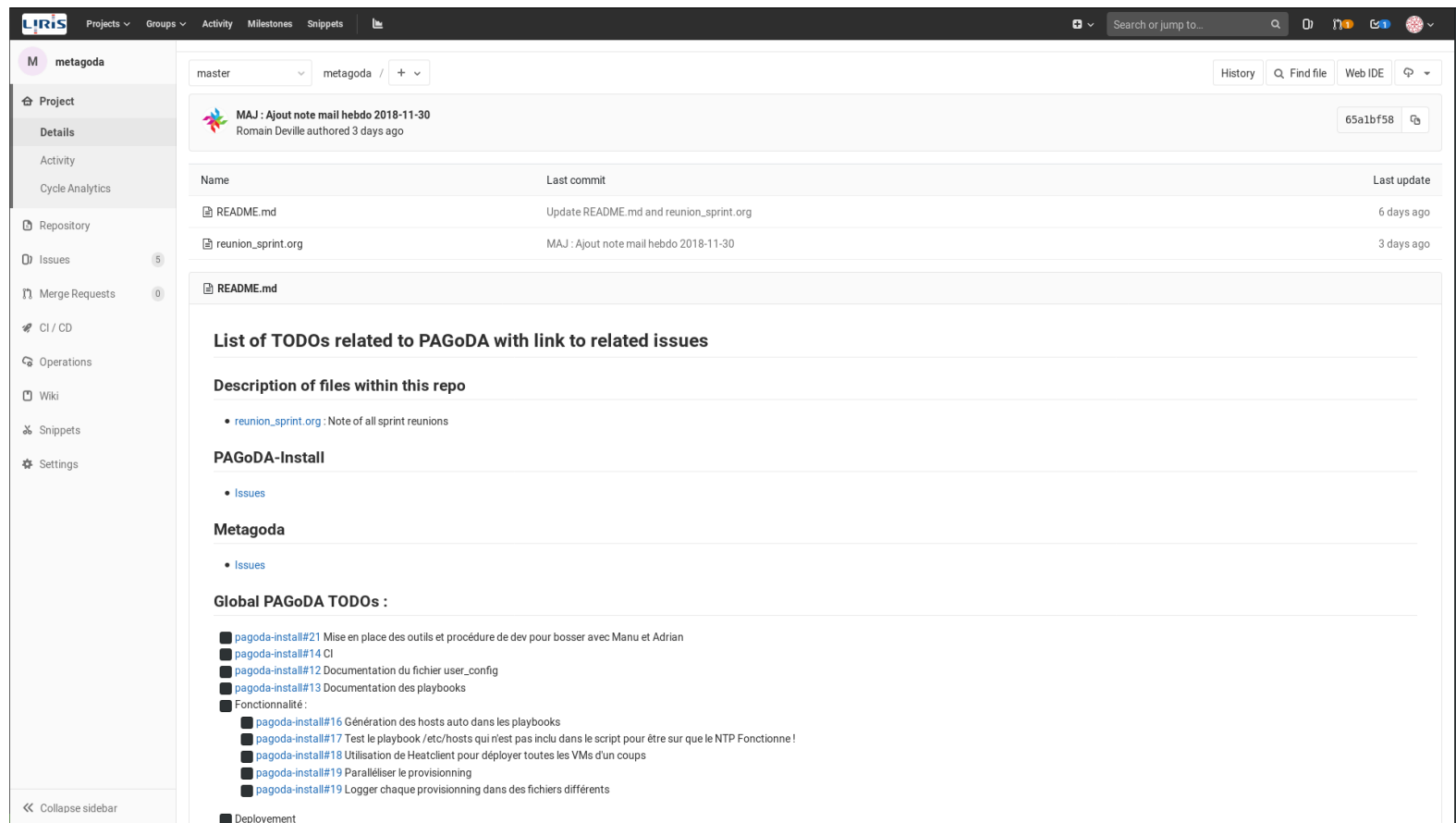
- **git branch -d <branch_name>**

- Supprime la branche <branch_name>

Les commandes git

Travailler avec un serveur git

- Apparté : <https://gitlab.liris.cnrs.fr>



Les commandes git

Travailler avec un serveur git

- **git remote add <remote_name> <url>**
 - Ajoute un lien (https ou ssh) vers le repo distant (ou remote)
- **git remote remove <remote_name>**
 - Retire le lien vers le repo distant
- **git remote rename <old_name> <new_name>**
 - Change le nom du lien vers le repo distant

Les commandes git

Travailler avec un serveur git

- **git push <remote_name> <branch_name>**
 - Envoi la branche du repo local vers la remote
- **git push -u <remote_name> <branch_name>**
 - Définit une remote et une branche à utiliser par défaut

Les commandes git

Travailler avec un serveur git

- **git pull <remote_name>**

- Récupère l'état du repo distant
- TOUJOURS FAIRE UN PULL AVANT DE TRAVAILLER !!!

- **git pull --all**

- Récupère toutes les branches de toutes les remotes
- A UTILISER AVEC PRÉCAUTIONS

Les commandes git

Travailler avec un serveur git

♦ **Gestion des conflits**

- ♦ En travaillant avec un repo distant et/ou à plusieurs on peut rencontrer des conflits
- ♦ Gestion manuel
- ♦ Il faut parcourir les fichiers conflictuels indiqués par git.

Les commandes git

Configuration local du repo

- **git config user.name "<your name>"**
- **git config user.email "<your@mail.com>"**
 - Définit les paramètres par défaut POUR LE REPO LOCAL
- **Config dans le fichier .gitconfig**

Les commandes git

Configuration local du repo

- **Ignorer des fichiers**

- Utile pour ne pas versionner des fichiers de configuration propre aux IDE par exemple

- **Les fichiers sont à lister dans le fichier .gitignore**

- <https://github.com/github/gitignore>