

CRCM

Centre de Recherche
en Cancérologie de Marseille

Gestion de version de codes ou de documents: gitlab au crcm et smartgit

Les technos gourmandes
Bernard Chetrit
30/01/2018

Gestion de version de codes ou de documents: gitlab au crcm et smartgit

CRCM

Centre de Recherche
en Cancérologie de Marseille

Cette oeuvre est mise à disposition selon les termes de la Licence Creative Commons Attribution - Pas d'utilisation Commerciale - Partage dans les Mêmes Conditions 4.0 International.

Vous êtes autorisé à :

- **Partager** — copier, distribuer et communiquer le matériel par tous moyens et sous tous formats
- **Adapter** — remixer, transformer et créer à partir du matériel

Selon les conditions suivantes :

- **Attribution** — Vous devez mentionner le nom de l'auteur de la manière suivante :« Bernard Chetrit, CRCM-CNRS, 2018 »
- **Pas d'Utilisation Commerciale** — Vous n'êtes pas autorisé à faire un usage commercial de cette oeuvre, tout ou partie du matériel la composant.
- **Partage dans les Mêmes Conditions** — Si vous modifiez, transformez ou adaptez cette oeuvre, vous n'avez le droit de distribuer votre création que sous une licence identique ou similaire à celle-ci.

Voir la version intégrale de la licence: <https://creativecommons.org/licenses/by-nc-sa/4.0/>

Gestion de version de codes ou de documents: gitlab au crcm et smartgit

CRCM

Centre de Recherche
en Cancérologie de Marseille

C'est quoi un logiciel de gestion de versions de codes?

c'est un outils qui permet de maintenir l'ensemble des versions d'un ou plusieurs fichiers.

Alors c'est juste pour la gestion des programmes des informaticiens ?

En fait non pas seulement, Gitlab peut permettre la gestion de plusieurs version de vos projets et donc d'avoir une sauvegarde et de l'archivage de votre travail.

Il va permettre de versionner votre code mais aussi vos publications, documents, résultats etc. (seuls les fichiers binaires et les fichiers trop volumineux sont interdits).

Vous allez aussi pouvoir travailler à plusieurs sur un même projet (grâce à la notion de branche)

GitLab en plus d'être une agréable interface pour explorer vos fichiers, les modifier et les comparer, vous fourni aussi un Wiki très pratique pour prendre des notes et faire de la documentation.

et c'est quoi Smartgit ?

Smartgit est le client git utilisable à la fois sur Windows, MacOSX et Linux qui permet d'avoir une interface local de gestion de vos dépôts gitlab.

Les différents systèmes de gestions de version

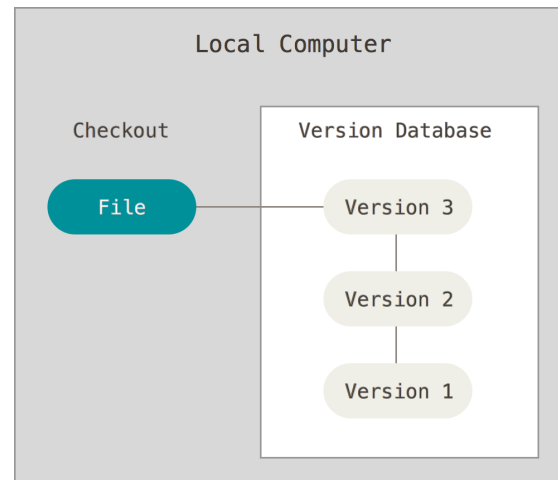
CRCM

Centre de Recherche
en Cancérologie de Marseille

Un système de gestion de version VCS « Version Control System » est un outils qui permet de suivre l'évolution d'un fichier ou d'un ensemble de fichiers à travers l'enregistrement des modifications (ajout, mise à jour, suppression)

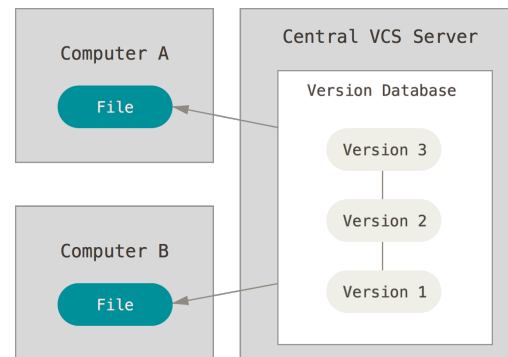
On distingue trois types de VCS:

Les systèmes de gestion de version locaux: GNU RCS et CVS

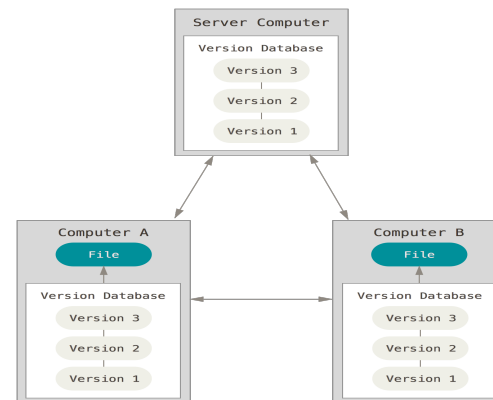


Les différents systèmes de gestions de version

Les systèmes de gestion de version centralisés (CVCS): Subversion (SVN)



Les systèmes de gestion de version distribués (DVCS): Mercurial, Git



Les différents systèmes de gestions de version

CRCM

Centre de Recherche
en Cancérologie de Marseille

Locaux:

- Pas de dépôts
- pas de travail collaboratifs
- Des fonctionnalité limités à la gestion de version

Centralisés:

- Architecture client serveur
- Un seul dépôt
- Le serveurs central contient toutes les données
- Toutes les requêtes se font entre le client et le serveur

Distribués

- Les données sont locales
- Opération rapides, connexions uniquement pendant les transferts
- Dépôts multiples

GIT Les bases: démarrage

CRCM

Centre de Recherche
en Cancérologie de Marseille

Installation en local

<https://git-scm.com/downloads>

créer un nouveau dépôt

créez un nouveau dossier, ouvrez le et exécutez la commande

git init

pour créer un nouveau dépôt.

cloner un dépôt

créez une copie de votre dépôt local en exécutant la commande

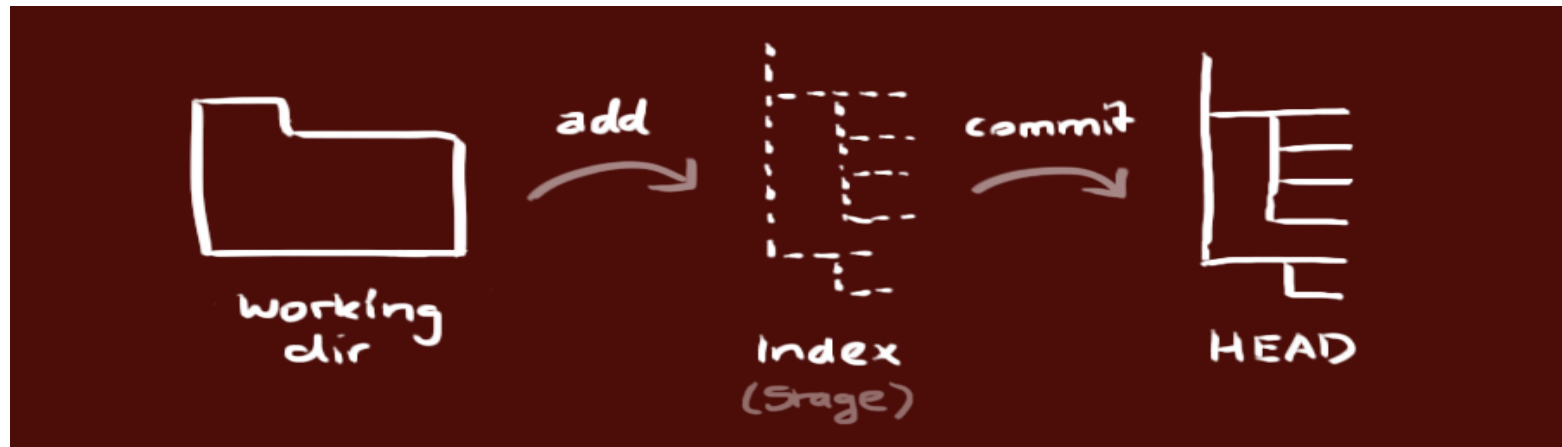
git clone /path/to/repository

si vous utilisez un serveur distant, cette commande sera

git clone username@host:/path/to/repository

Arbres

- votre dépôt local est composé de trois "arbres" gérés par git. le premier est votre espace de travail qui contient réellement vos fichiers. le second est un Index qui joue un rôle d'espace de transit pour vos fichiers et enfin HEAD qui pointe vers la dernière validation que vous avez faite.



GIT Les bases: ajout de fichiers

CRCM

Centre de Recherche
en Cancérologie de Marseille

ajouter & valider

Vous pouvez proposer un changement (l'ajouter à l'**Index**) en exécutant les commandes

```
git add <filename>
```

```
git add *
```

C'est la première étape dans un workflow git basique. Pour valider ces changements, utilisez

```
git commit -m "Message de validation"
```

Le fichier est donc ajouté au **HEAD**, mais pas encore dans votre dépôt distant.

GIT Les bases: du local vers le serveur

CRCM

Centre de Recherche
en Cancérologie de Marseille

envoyer des changements

Vos changements sont maintenant dans le **HEAD** de la copie de votre dépôt local. Pour les envoyer à votre dépôt distant, exécutez la commande

```
git push origin master
```

Remplacez *master* par la branche dans laquelle vous souhaitez envoyer vos changements.

Si vous n'avez pas cloné votre dépôt existant et voulez le connecter à votre dépôt sur un serveur distant, vous devez l'ajouter avec

```
git remote add origin <server>
```

Maintenant, vous pouvez envoyer vos changements vers le serveur distant sélectionné

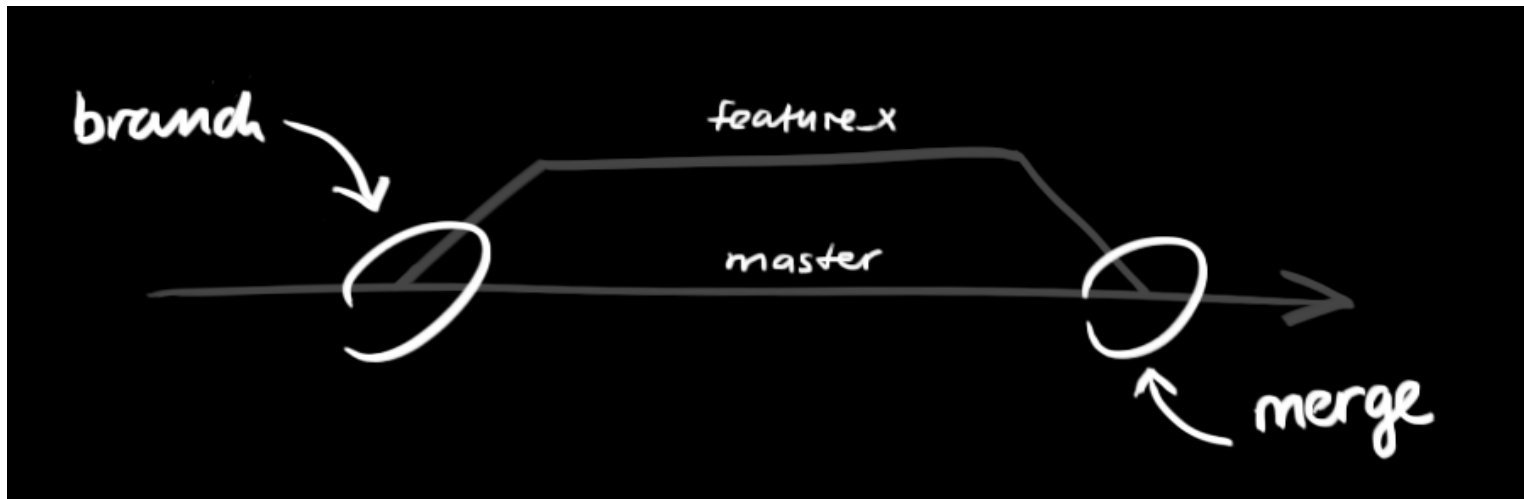
GIT Les bases: La notion de branches

CRCM

Centre de Recherche
en Cancérologie de Marseille

Branches

- Les branches sont utilisées pour développer des fonctionnalités isolées des autres. La branche *master* est la branche par défaut (version stable) quand vous créez un dépôt. Utilisez les autres branches pour le développement et fusionnez ensuite à la branche principale quand vous avez fini.



GIT Les bases: La notion de branches

CRCM

Centre de Recherche
en Cancérologie de Marseille

Branches

créer une nouvelle branche nommée "feature_x" et passer dessus pour l'utiliser

```
git checkout -b feature_x
```

retourner sur la branche principale

```
git checkout master
```

et supprimer la branche

```
git branch -d feature_x
```

une branche n'est *pas disponible pour les autres* tant que vous ne l'aurez pas envoyée vers votre dépôt distant

```
git push origin <branch>
```

GIT Les bases: du serveur vers le local

CRCM

Centre de Recherche
en Cancérologie de Marseille

mettre à jour & fusionner

pour mettre à jour votre dépôt local avec les dernières validations, exécutez la commande

`git pull`

dans votre espace de travail pour *récupérer* et *fusionner* les changements distants.

pour fusionner une autre branche avec la branche active (par exemple master), utilisez

`git merge <branch>`

dans les deux cas, git tente d'auto-fusionner les changements. Malheureusement, ça n'est pas toujours possible et il en résulte des *conflicts*. Vous devez alors régler ces *conflicts* manuellement en éditant les fichiers indiqués par git. Après l'avoir fait, vous devez les marquer comme

fusionnés avec

`git add <filename>`

après avoir fusionné les changements, vous pouvez en avoir un aperçu en utilisant

`git diff <source_branch> <target_branch>`

Tags

il est recommandé de créer des tags pour les releases de programmes. c'est un concept connu, qui existe aussi dans SVN. Vous pouvez créer un tag nommé *1.0.0* en exécutant la commande

```
git tag 1.0.0 1b2e1d63ff
```

le *1b2e1d63ff* désigne les 10 premiers caractères de l'identifiant du changement que vous voulez référencer avec ce tag. Vous pouvez obtenir cet identifiant avec

```
git log
```

vous pouvez utiliser moins de caractères pour cet identifiant, il doit juste rester unique.

Le Serveur GIT du CRCM GITLAB

CRCM

Centre de Recherche
en Cancérologie de Marseille

GIT au CRCM sur un serveur Gitlab

<https://gitcrcm.marseille.inserm.fr>

disc

Disc's Gitlab Server



CRCM's Users gitlab service

Provided by CRCM's DISC platform

CRCM	Standard
CRCM Username	
<input type="text" value="bernard2"/>	
Password	
<input type="password" value="....."/>	
<input type="checkbox"/> Remember me	
<input type="button" value="Sign in"/>	

Le Serveur GIT du CRCM GITLAB

CRCM

Centre de Recherche

Welcome to GitLab

Code, test, and deploy together



You can create a group for several dependent projects.
Groups are the best way to manage projects and members.

New group



You don't have access to any projects right now

Le Serveur GIT du CRCM GITLAB

CRCM

Centre de Recherche
en Cancérologie de Marseille

Group path

Description

Group avatar

The maximum file size allowed is 200KB.

Visibility Level


- Private**
The group and its projects can only be viewed by members.
- Internal**
The group and any internal projects can be viewed by any logged in user.
- Public**
The group and any public projects can be viewed without any authentication.

- A group is a collection of several projects
- Members of a group may only view projects they have permission to access
- Group project URLs are prefixed with the group namespace
- Existing projects may be moved into a group

Le Serveur GIT du CRCM GITLAB

CRCM

Centre de Recherche
en Cancérologie de Marseille

Group Issues 0 Merge Requests 0 **Members** 

Users were successfully added.

Members



Add new member to **test-git-group**

Search for members by name, username, or email, or invite new ones using their email address. [Read more](#) about role permissions

On this date, the member(s) will automatically lose access to this group and all of its projects.

Existing members

Members with access to **test-git-group** **2**

	Ghislain Bidaut @bidaut Joined less than a minute ago	<input type="text" value="Developer"/>	<input type="text" value="Expiration date"/>	<input type="button" value="Remove"/>
	bernard2 @bernard2 It's you Joined 3 minutes ago			Owner

Le Serveur GIT du CRCM GITLAB

CRCM

Centre de Recherche
en Cancérologie de Marseille

New project

Create or Import your project from popular Git services

Project path


https://gitcram.marseille.inserm.fr/ test-git-group


Project name

test-git-proj1


Want to house several dependent projects under the same namespace? [Create a group](#)


Import project from

 GitHub


 Bitbucket


 GitLab.com

 Google Code

 Fogbugz

 Gitea




 git Repo by URL

 GitLab export

Project description (optional)

Description format

Visibility Level

-  Private
Project access must be granted explicitly to each user.
-  Internal
The project can be cloned by any logged in user.
-  Public
The project can be cloned without any authentication.

Le Serveur GIT du CRCM

GITLAB

CRCM

Centre de Recherche
en Cancérologie de Marseille

Git global setup

```
git config --global user.name "bernard2"  
git config --global user.email "bernard2@crcm.mrs"
```

Create a new repository

```
git clone https://bernard2@gitcrcm.marseille.inserm.fr/test-git-group/test-git-proj1.git  
cd test-git-proj1  
touch README.md  
git add README.md  
git commit -m "add README"  
git push -u origin master
```

Existing folder

```
cd existing_folder  
git init  
git remote add origin https://bernard2@gitcrcm.marseille.inserm.fr/test-git-group/test-git-proj1.git  
git add .  
git commit  
git push -u origin master
```

Existing Git repository

Le Serveur GIT du CRCM GITLAB

CRCM

Centre de Recherche
en Cancérologie de Marseille

Project Repository Issues 0 Merge Requests 0 Pipelines **Wiki** Settings

Home · Create Page

New Page

Clone repository

Format Markdown

Content

Write Preview

B *I* “ ” </> ☰ ☷ ☑ ✕

tes-git-proj1
=====

****Projet test pour démo****

1. blablabla
- 1| blablabla

Markdown is supported

Attach a file

To link to a (new) page, simply type `[Link Title](page-slug)`. More examples are in the documentation.

Commit message

More Pages

Le Serveur GIT du CRCM GITLAB

CRCM

Centre de Recherche
en Cancérologie de Marseille

The screenshot shows a GitLab repository page for 'test-git-proj1'. At the top, there is a profile picture with the letter 'T' and the repository name 'test-git-proj1'. Below this, there are statistics for 'Star' (0) and 'Fork' (0), along with the HTTPS URL 'https://bernard2@gitcrcm.marse...'. A dropdown menu is open, showing options: 'New issue', 'New merge request', 'New file', 'New branch', and 'New tag'. Below the menu, there are links for 'Files (123 KB)', 'Commit (1)', 'Branch (1)', 'Tags (0)', 'Readme', and 'Apache License 2.0'. A commit history table is visible, showing a commit '6e8c2dff' by 'bernard2' with the message 'Add license' from 7 minutes ago. Below the table, there is a file browser showing 'README.md'.

Files (123 KB) Commit (1) Branch (1) Tags (0) Readme Apache License 2.0

6e8c2dff Add license · 7 minutes ago by **bernard2**

master test-git-proj1 / +

Name	Last commit > 6e8c2dff 7 minutes ago - Add license	History	Last Update
LICENSE	Add license		7 minutes ago
README.md	Update README.md		9 minutes ago

README.md

Le Serveur GIT du CRCM GITLAB

CRCM

Centre de Recherche
en Cancérologie de Marseille

test-git-group / test-git-proj1 ▾

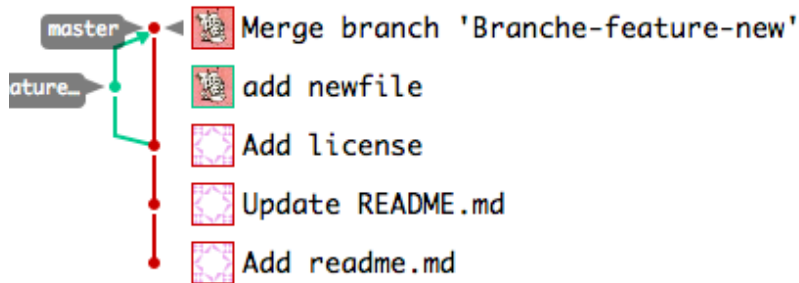
This project Search 🔍 🔔

Project **Repository** Issues 0 Merge Requests 0 Pipelines Wiki Settings

Files Commits Branches Tags Contributors **Graph** Compare Charts

▾ You can move around the graph by using the arrow keys.

🔍 Begin with the selected commit



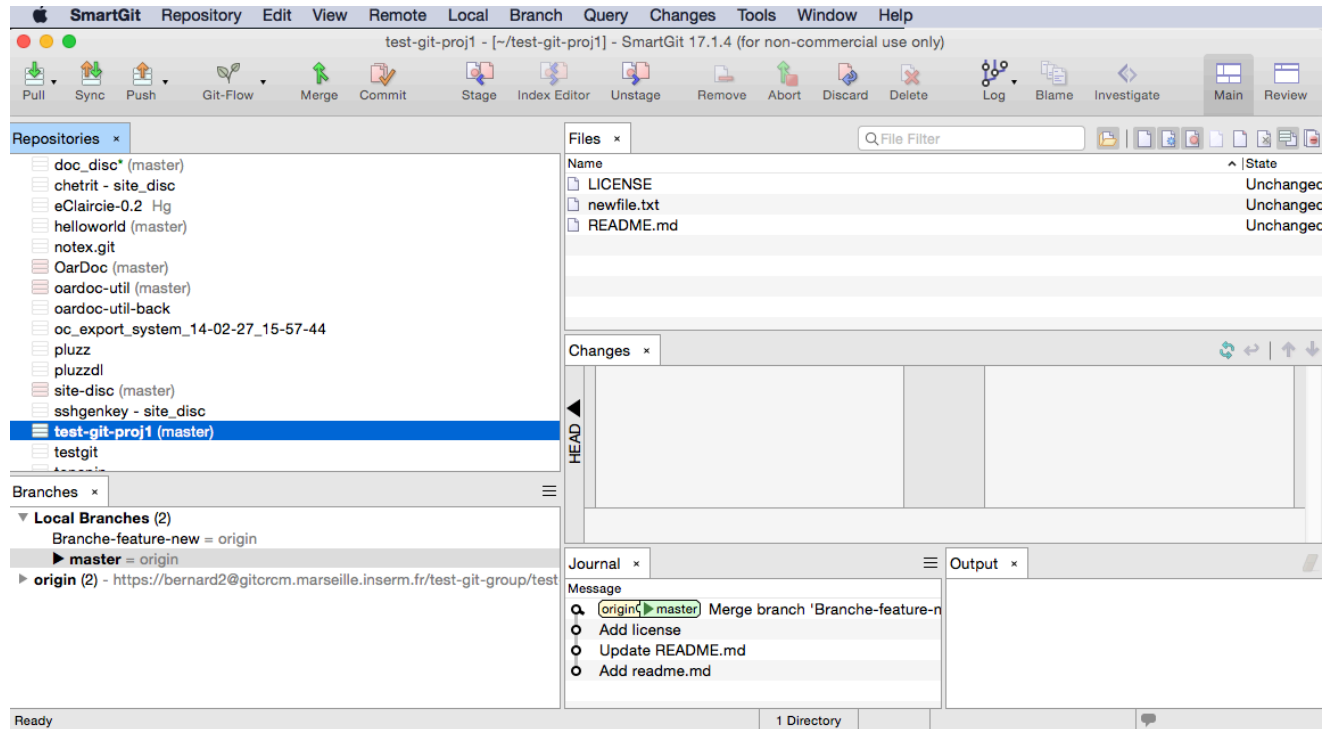
Le client SMARTGIT

CRCM

Centre de Recherche
en Cancérologie de Marseille

Télécharger smartgit

<http://www.syntevo.com/smartgit/>



Gestion de version de codes ou de documents: gitlab au crcm et smartgit

CRCM

Centre de Recherche
en Cancérologie de Marseille

Références et Documentations

- © Doc GIT: <https://git-scm.com/book/en/v2>
- © Gitlab User Doc: <https://docs.gitlab.com/ee/user/index.html>
- © Doc smartGIT: <http://www.syntevo.com/doc/display/SG171/Manual>

Outils pédagogiques Git

- © Concepts de base de Git: <https://onlywei.github.io/explain-git-with-d3/>
- © Apprendre les branches par l'exemple: <https://learngitbranching.js.org/>
- © Tutoriel visuel Git: <https://agripongit.vincenttunru.com/>