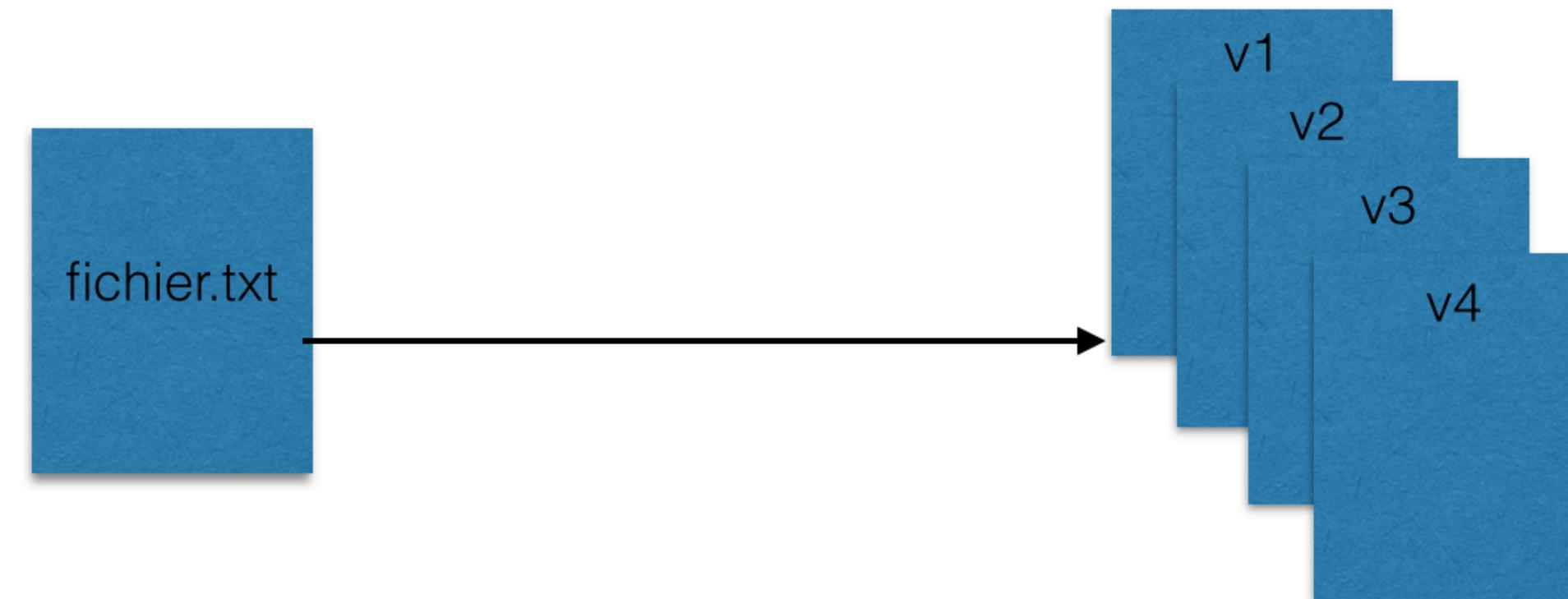


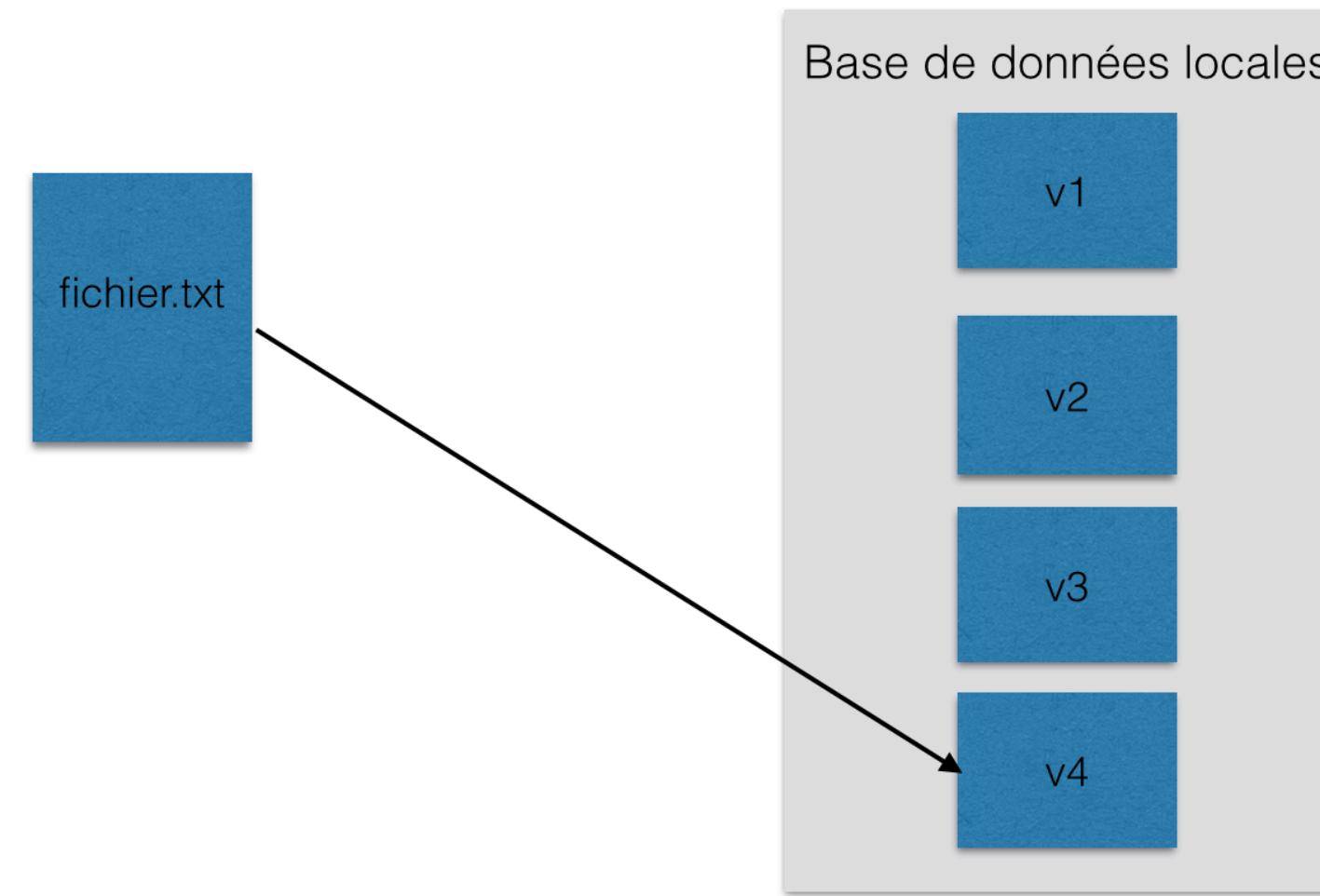
Cours 1 : Introduction

Une petite histoire des gestionnaires de version.

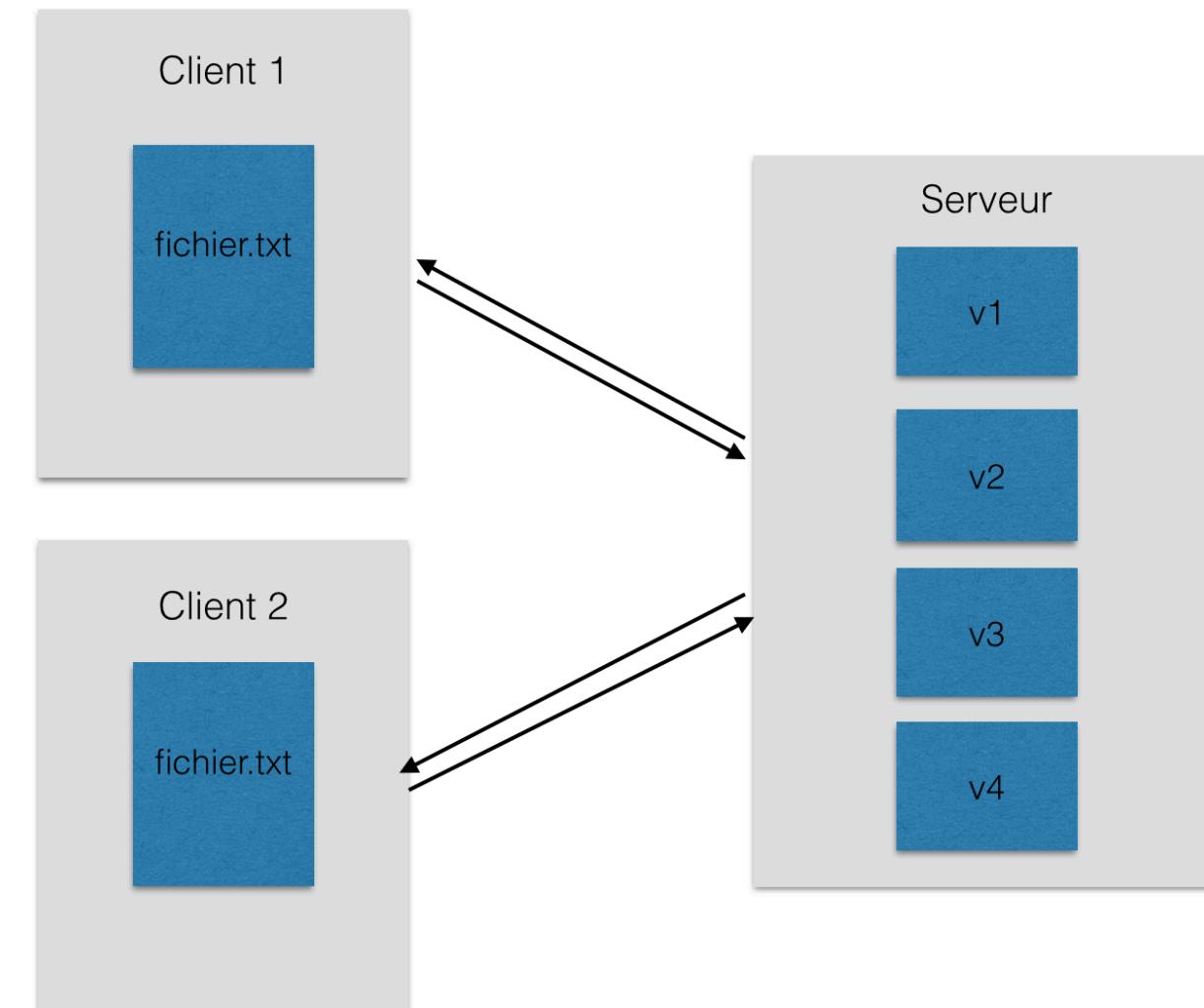
Gestionnaire de version local par duplication.



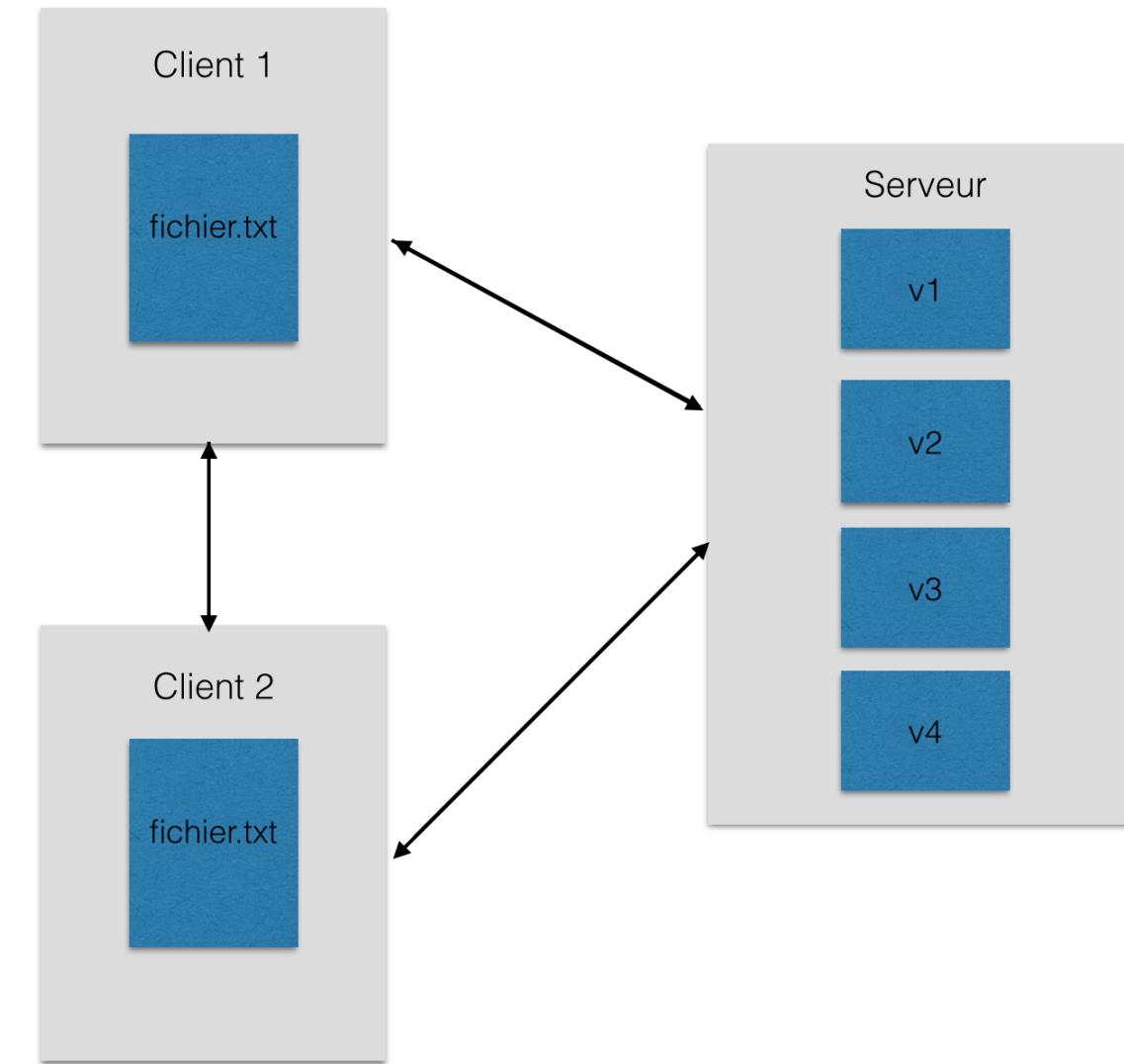
Gestionnaire de version avec base de données locale.



Gestionnaire de version centralisé.



Gestionnaire de version décentralisé .



Histoire de git.

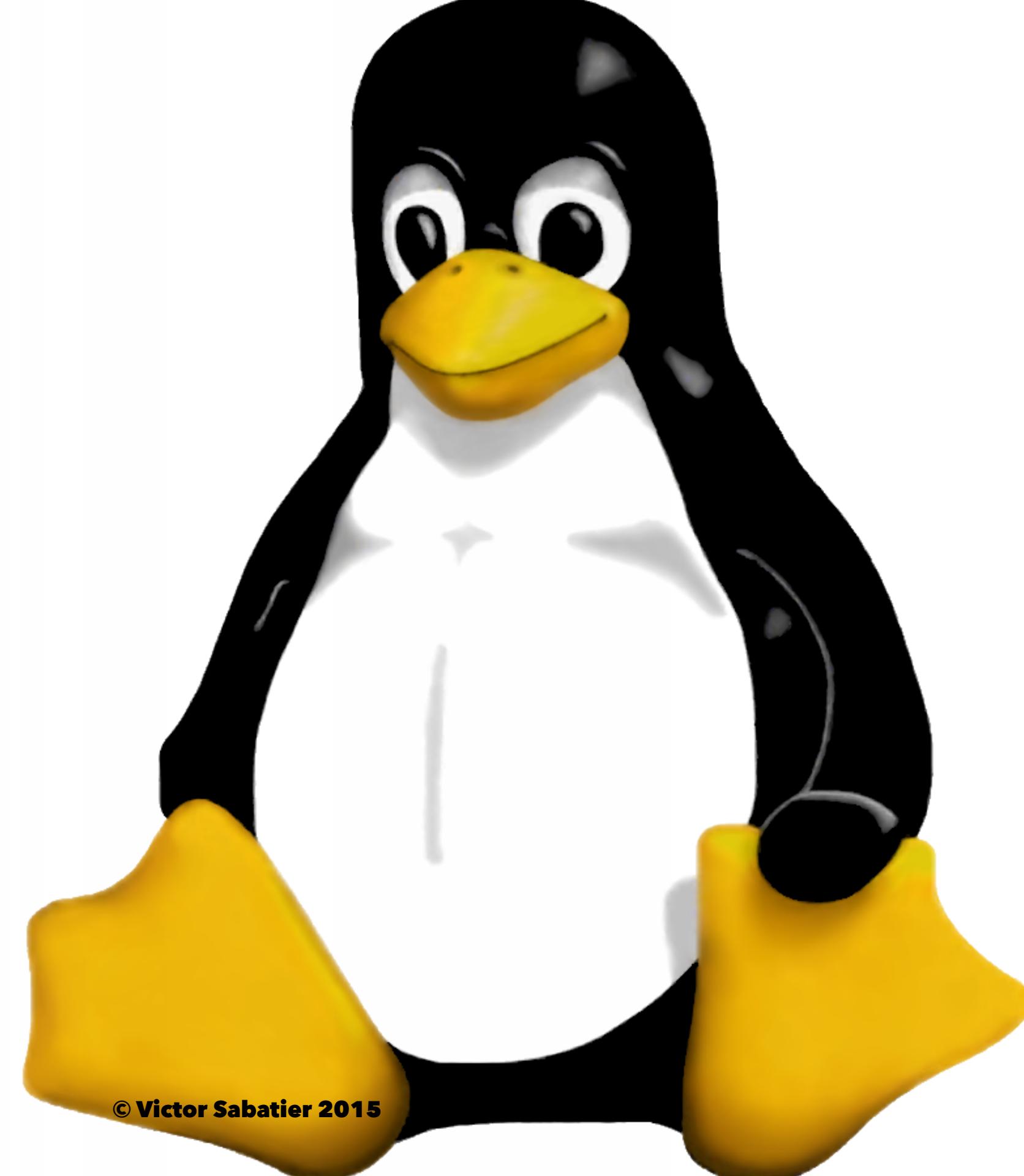
- Jusqu'en 2002 : utilisation de patchs.
- De 2002 à 2005 : BitKeeper propriétaire mais gratuit.
- Développé en quelques jours par Linus pour le noyau Linux¹.

¹ L'histoire de git est étroitement liée avec celle du noyau linux.

Les avantages de git

- La pluspart des opérations sont locales.
- Git gère l'intégrité.
- Chaque ordinateur est une sauvegarde complète du repository.
- Très peu d'opérations destructrices.

Installation de Git



Linux

Le plus simple est de l'installer² via votre gestionnaire de paquets.

```
1 | apt-get install git  
2 | yum install git
```

² Il est très probable que git soit par défaut sur votre distribution.

Mac

Soit via votre gestionnaire de paquet :

```
1 | brew install git
```

Soit via <http://sourceforge.net/projects/git-osx-installer/>

Windows

Installer msysGit³ : <http://msysgit.github.io>



³ Vous devez utiliser Git avec le shell bash fourni par msysGit. Retrouvez le raccourci sur votre bureau.

Ligne de commande

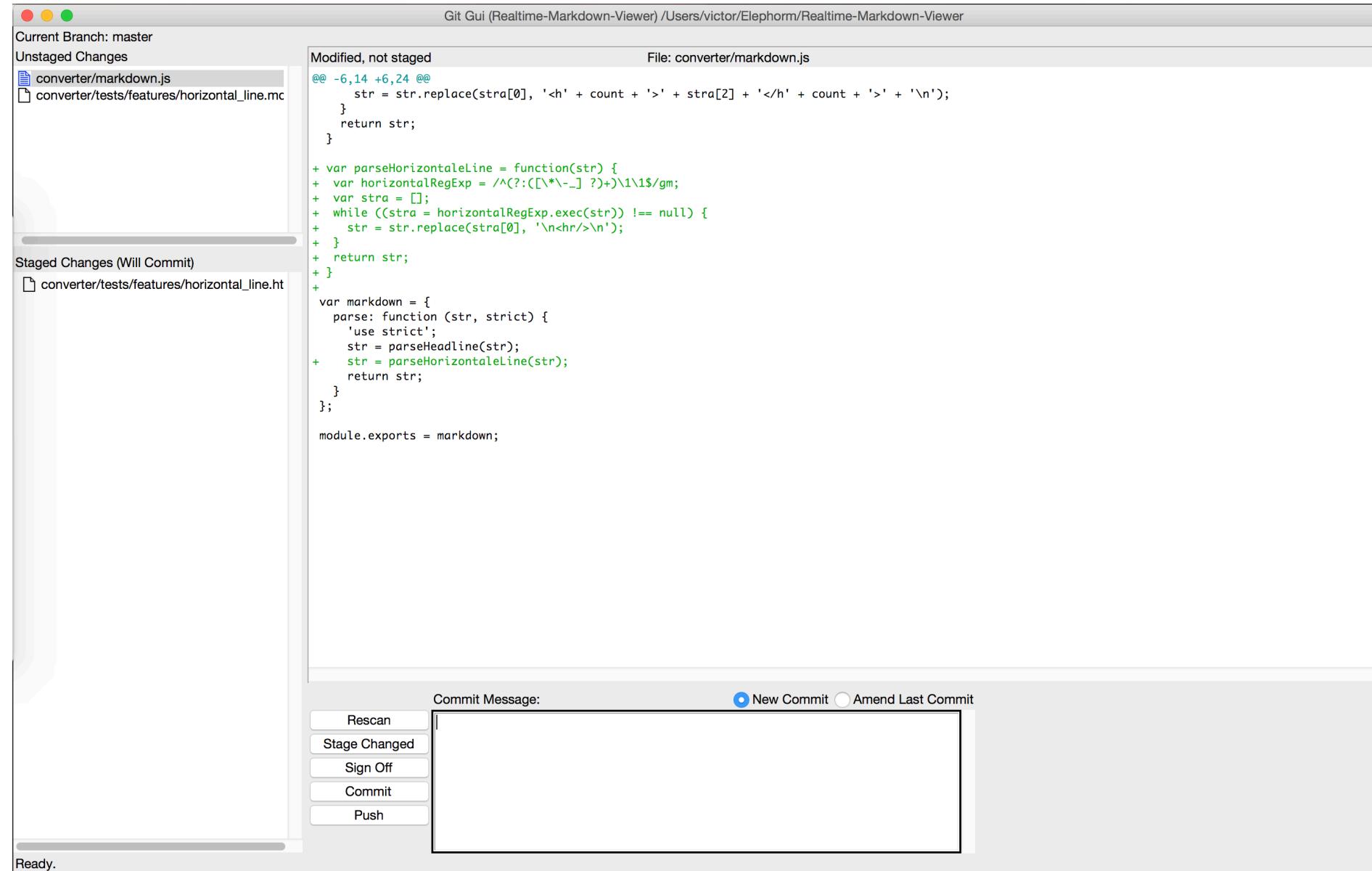
Pour vérifier que git est installé, ouvrez votre terminal⁴:



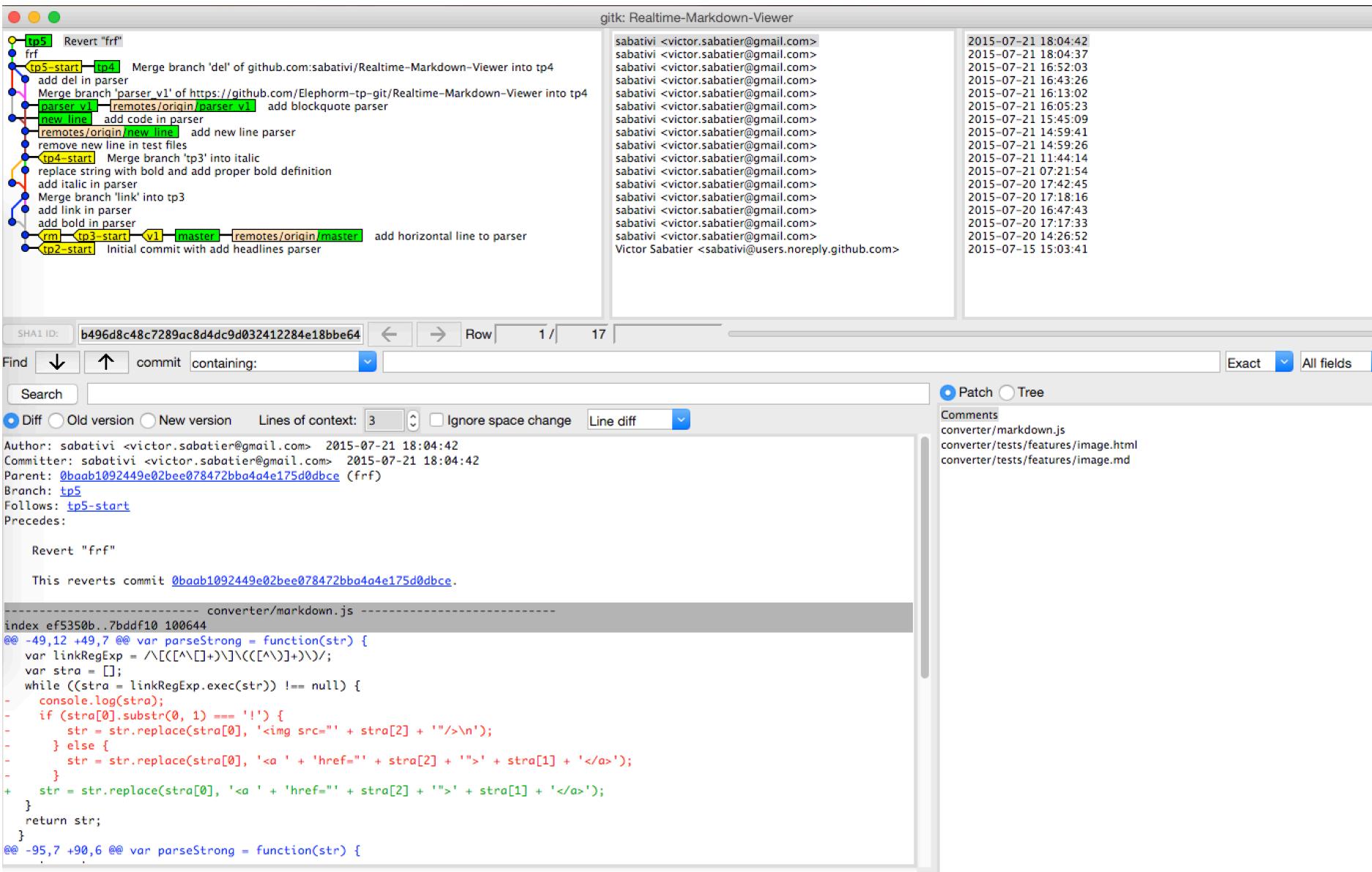
```
Realtime-Markdown-Viewer — victor@MacBook-Pro-de-victor: ~/Elephorm/Realtime-Markdow...
→ Realtime-Markdown-Viewer git:(master) git --version
git version 2.3.1
→ Realtime-Markdown-Viewer git:(master) which git
/usr/local/bin/git
```

⁴ Si votre version est légèrement différente, ce n'est pas grave. Néanmoins assurez vous d'être au moins à la version 2.0.

git gui



gitk



Où héberger son code ?



Github : gratuit pour les projets open source.

Gitlab : open source.

Gitbucket : gratuit aussi pour les projets privés.

Configurer Git.

Trois niveaux de configuration :

Système : /etc/gitconfig

Global : ~/.gitconfig

Local : .git/config

```
[core]
repositoryformatversion = 0
filemode = true
bare = false
logallrefupdates = true
ignorecase = true
precomposeunicode = true
[remote "origin"]
url = https://github.com/Elephorm-tp-git/Realtime-Markdown-Viewer.git
fetch = +refs/heads/*:refs/remotes/origin/*
[branch "master"]
remote = origin
merge = refs/heads/master
```

Pour afficher toute votre config :

```
git config -l
```

Par exemple :

```
➔ Realtime-Markdown-Viewer git:(code) git config -l
user.name=sabativi
user.email=victor.sabatier@gmail.com
alias.amend=commit --amend --no-edit
alias.st=status
alias.who=shortlog -sne
alias.l=log --pretty=oneline --abbrev-commit --graph --decorate
alias/lg=log --pretty=oneline --abbrev-commit --graph --decorate --all
alias.changes=diff --name-status
alias.dic=diff --cached
alias.diffstat=diff --stat
```

Ajouter une valeur :

```
git config [--global|--system] KEY VALUE
```

Ou éditer directement le fichier :

```
git config [--global] -e
```

Configuration minimale

Vous devez avoir une adresse mail et un nom pour enregister votre première modification sur git :

```
git config --global user.name NAME  
git config --global user.email EMAIL
```

Quelques options⁵:

```
git config --global core.editor vim/emacs/notepad/...
git config --global color.ui true
git config --global help.autocorrect 1
git config --global core.autocrlf true // à faire sous windows
```

```
git config --global alias.st status
git config --global alias.co checkout
```

⁵ Vous pouvez télécharger un exemple de config plus complet dans le TP.

Consulter l'aide⁶

git help command

```
GIT-HELP(1)                               Git Manual                               GIT-HELP(1)

NAME
    git-help - Display help information about Git

SYNOPSIS
    git help [-a|--all] [-g|--guide]
                [-i|--info|-m|--man|-w|--web] [COMMAND|GUIDE]

DESCRIPTION
    With no options and no COMMAND or GUIDE given, the synopsis of the git command
    and a list of the most commonly used Git commands are printed on the standard
    output.

    If the option --all or -a is given, all available commands are printed on the
    standard output.

    If the option --guide or -g is given, a list of the useful Git guides is also
    printed on the standard output.
```

⁶ L'aide de git est complète.