

TACT factory

mobile agency



Fournisseur **souple** et **réactif**
d'applications mobiles innovantes
avec une **méthodologie** projet
rigoureuse.

Présentation

Mickael Gaillard (Architecte logiciel)

mickael.gaillard@tactfactory.com

Yoan Pintas (Lead Developer)

yoan.pintas@tactfactory.com

Erwan LeHuitouze (Lead Developer)

erwan.lehuitouze@tactfactory.com



Planning session 1

Démarrer avec Git :

- Présentation de Git
- Dépôt local vs distant
- Snapshots vs différences
- Installation

Configurer Git :

- Préférences Git
- Définir son nom et son email
- l'éditeur et l'outil de différentiel

Fondamentaux Git :

- Initialiser un dépôt
- Architecture d'un dépôt
- Ajouter des fichiers
- Cloner un dépôt existant
- Ignorer des fichiers
- Supprimer, déplacer, renommer
- Consulter l'historique
- Annuler des modifications

Pré-requis

Unix/Linux (shell base, hidden file, vfs)

Concept de différentiel (diff)

Concept de snapshot

ssh

Session utilisateur

Fichier « ini » (ou à plat)

regex

Présentation de Git

SCM : Source Control Manager ou Gestion de Version

...alternative à la méthode « la-rache » !

Historique SCM :

CSSC (IBM- 1972) > RCS (1982) > CVS (1986) > Subversion (2000) > GIT (2005)

Historique Git :

- Noyau Linux => BitKeeper (propriétaire)
- en 2005, retro-ingénierie => abandon de la version freeware...
- Création de GIT par Linus Torvalds (In 4 days)

Numérotation

Exemples : 0.1.0 – 1.1.9.3216

Pas de règle absolue :

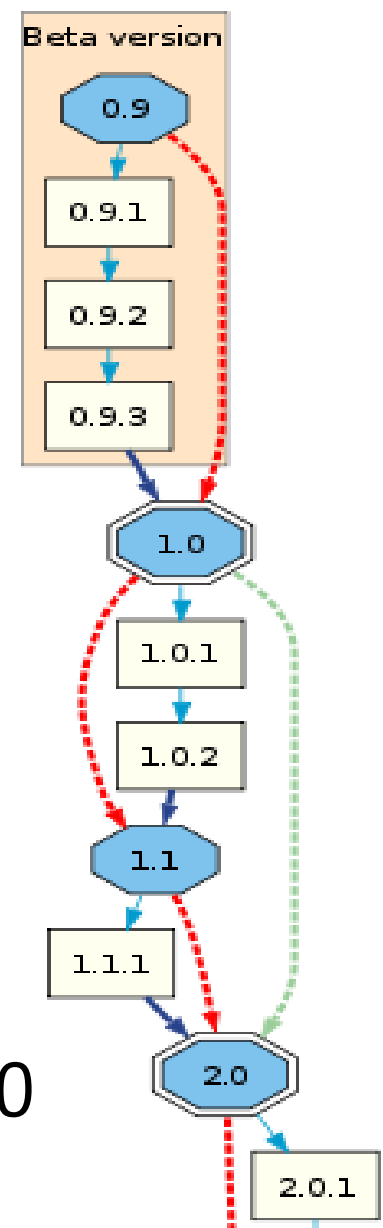
Gestion des numéros de version en fonction de vos besoins

Documentation :

http://en.wikipedia.org/wiki/Software_versioning

http://wiki.eclipse.org/Version_Numbering

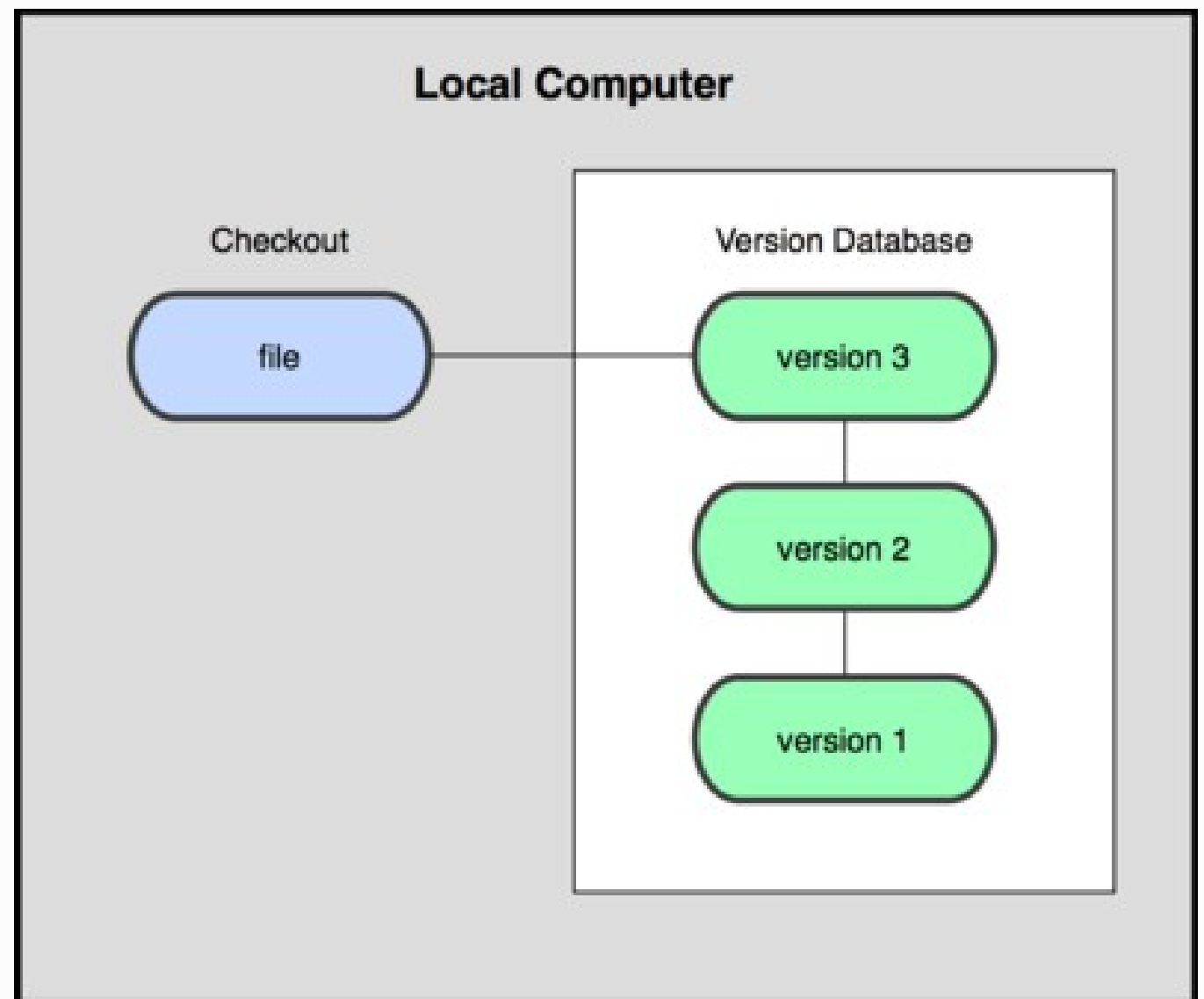
http://fr.wikipedia.org/wiki/Noyau_Linux#Mode_de_num.C3.A9rotation



Dépôt local vs distant (local)

Dépôt local : (RCS)

- **File (system)** : les fichiers disponibles dans votre navigateur de fichier
- **Version database** : est l'historique de votre projet (généralement accessible depuis l'outil de versioning)



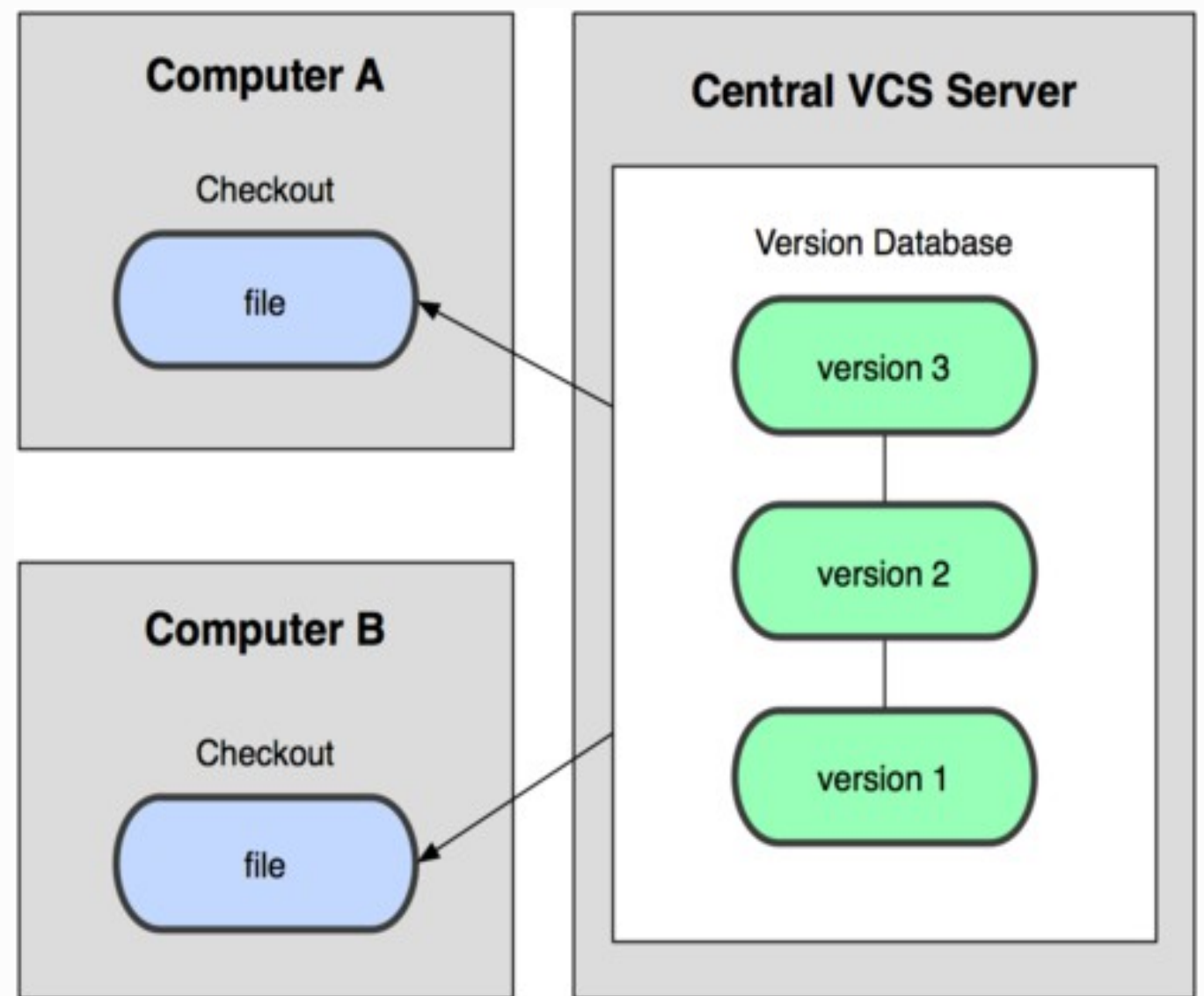
Dépôt local vs distant (centralisé)

Dépôt centralisé : (cvs - svn)

L'inconvénient du dépôt local est de faire fonctionner cette solution dans un contexte collaboratif (en équipe) !!

Solution :

Déporter l'historique (gérer par l'outil de versioning) sur un serveur distant !



Dépôt local vs distant (distribué)

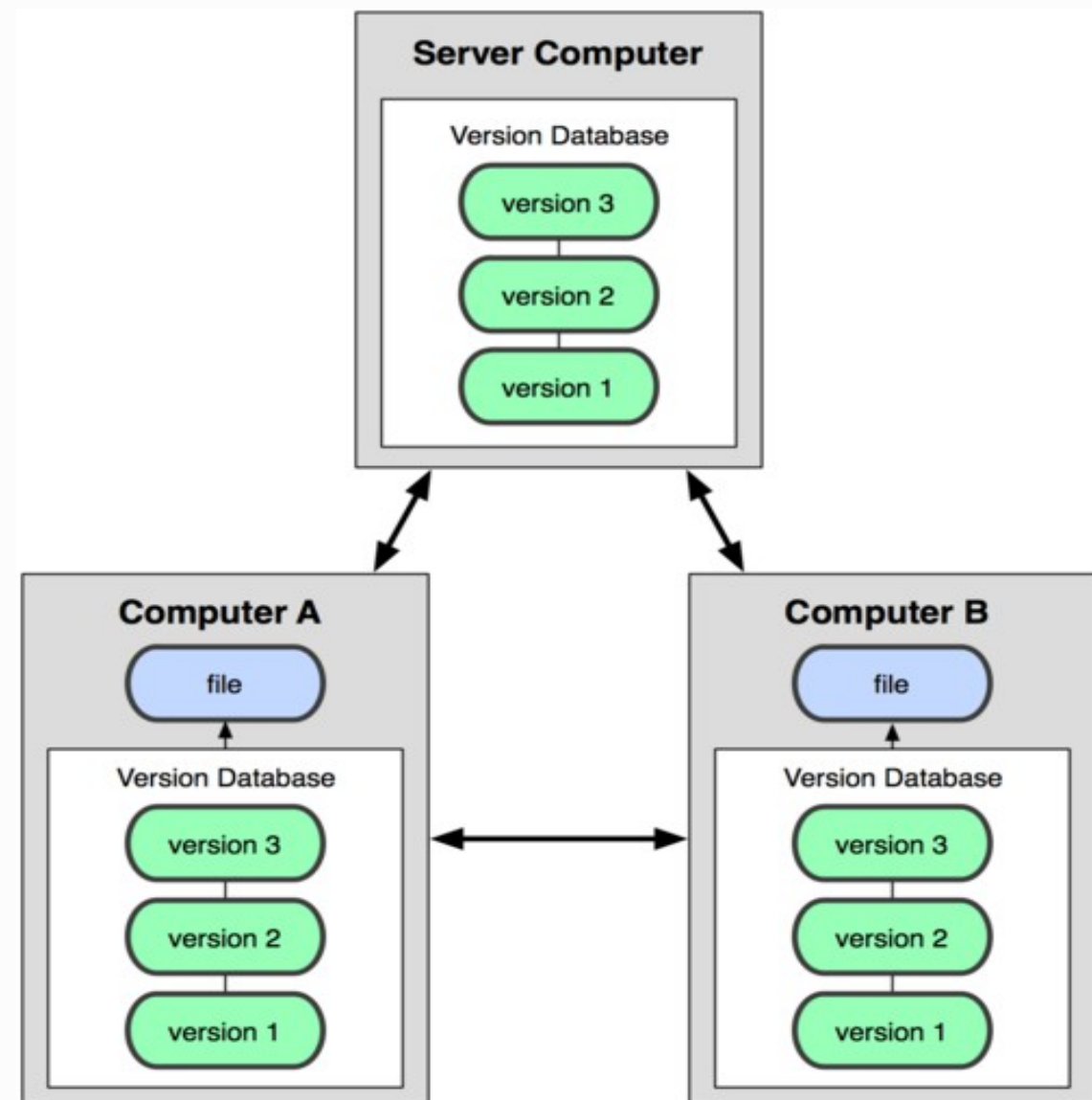
Dépôt distribué : (Git)

L'inconvénient du modèle centraliser est lié à l'impossibilité d'utiliser l'outil de versioning, si vous n'avez pas de réseau !!

Solution :

Synchroniser l'historique sur chaque machine !

L'avantage est de réaliser les traitements en local (gain de performance).

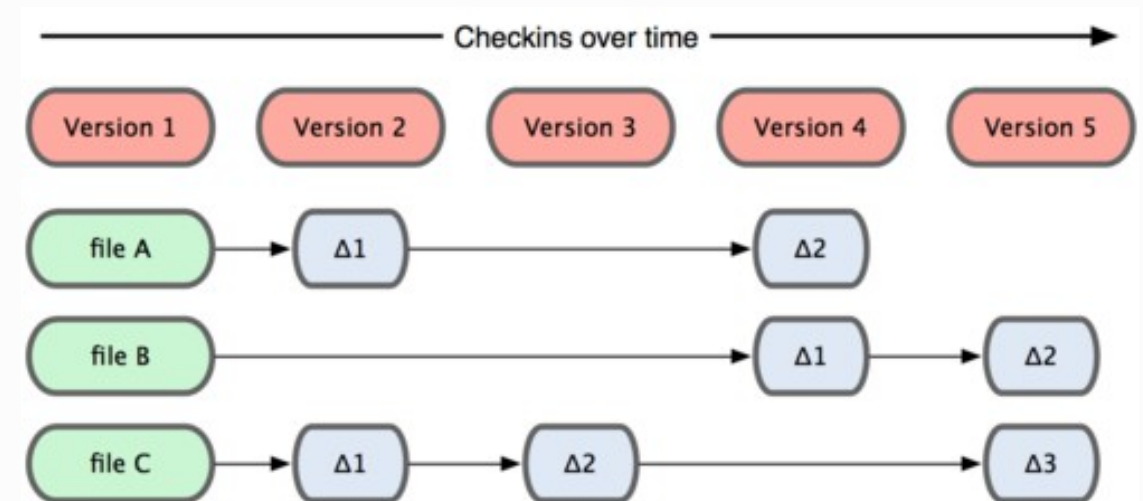


Snapshots vs différences

Mode Différentiel :

L'historique conserve les **modifications** des fichiers dans le temps.

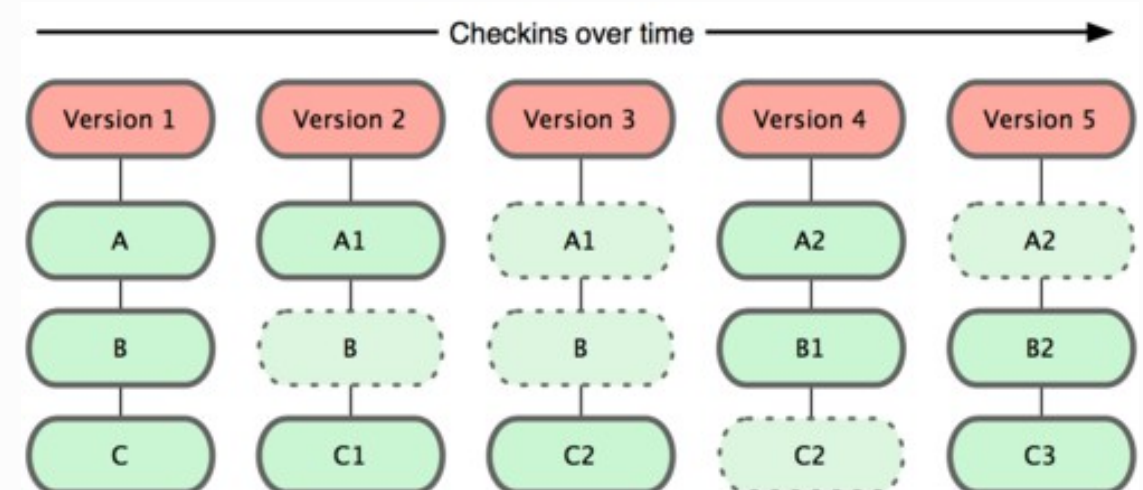
Il faut re-jouer l'ensemble des versions antérieures pour atteindre une version donnée. (cas de svn...)



Mode Snapshots :

L'historique conserve une **image complète** de l'espace de travail.

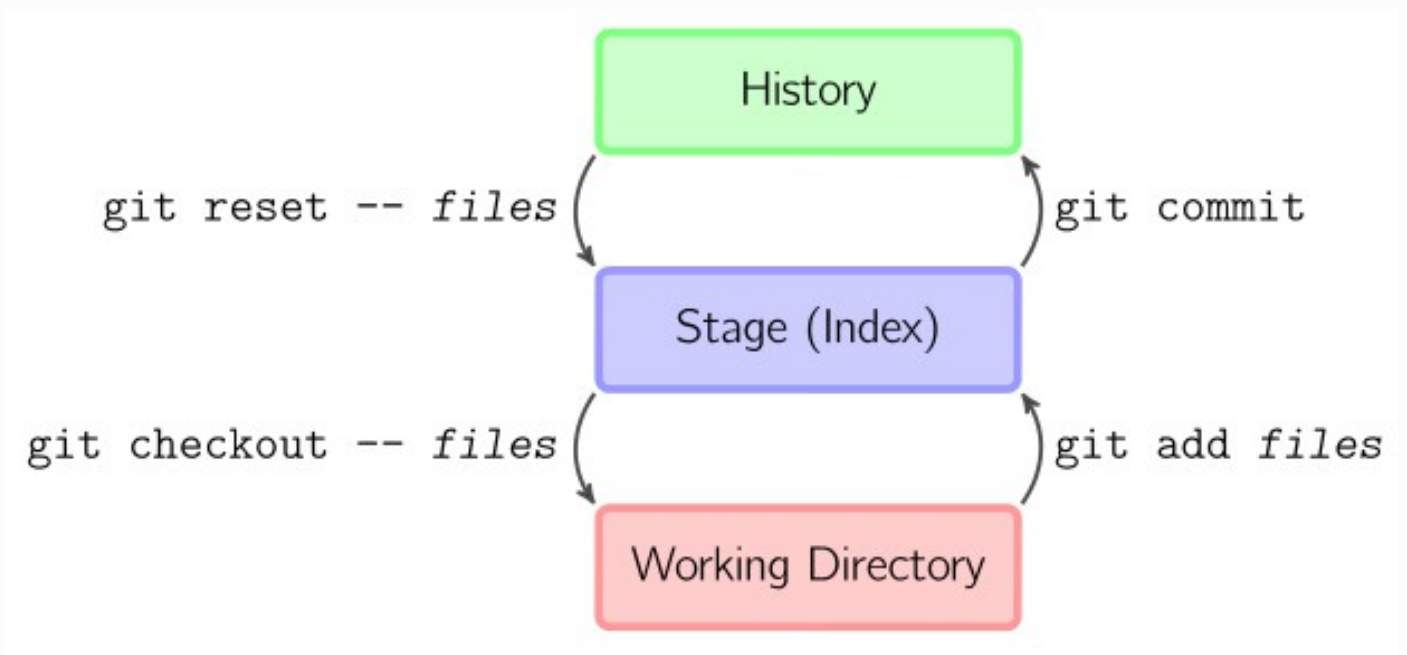
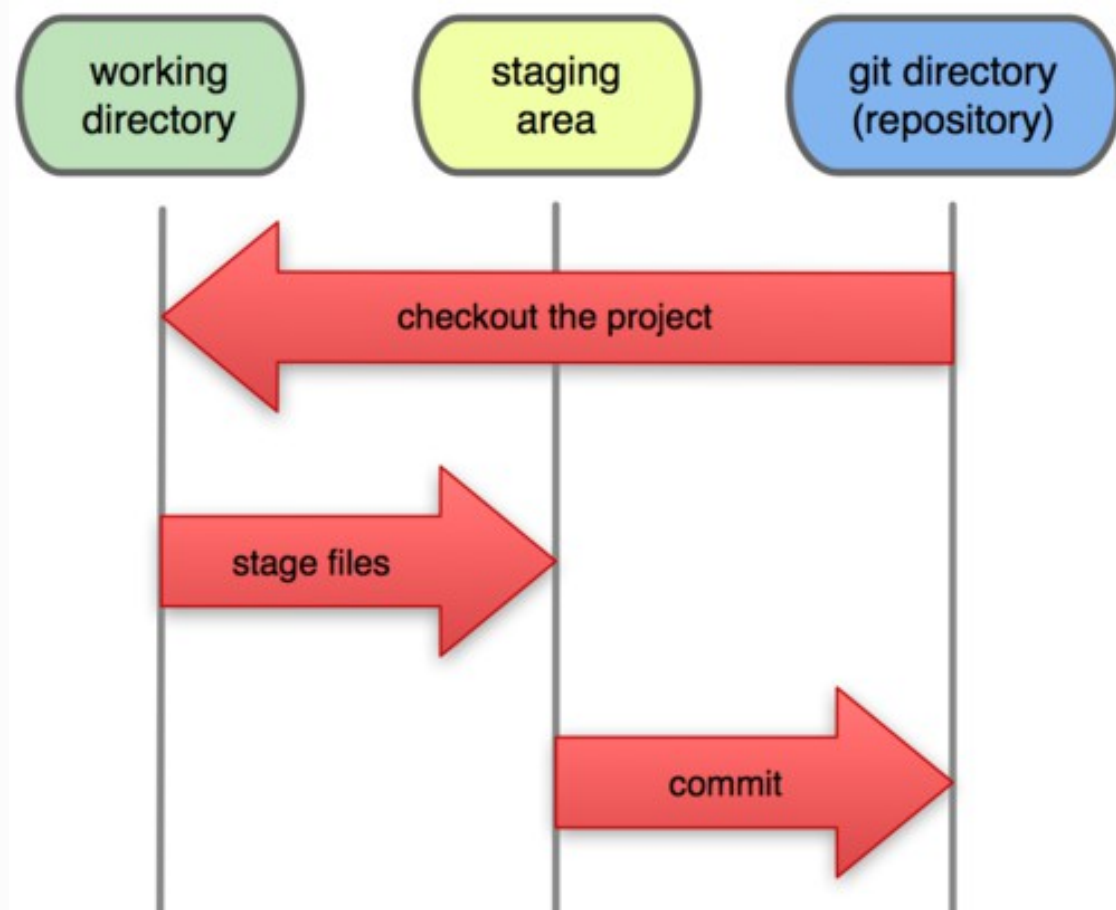
Optimisé par des références. (exclusivité de Git)



Processus local

Working :

Local Operations



Installation

Client : <http://git-scm.com/book/fr/D%C3%A9marrage-rapide-Installation-de-Git>

- Linux (ubuntu/debian) : apt-get install git-core gitg
- Windows : msysgit
- Mac : git-osx-installer ou par MacPorts

Authentification :

- par Mots de passe
- par Clés SSH

Génération de la clé : ssh-keygen -t rsa -b 4096

Note : il est impératif que la paire de clés soit dans le dossier **.ssh** à la racine de votre home.

Préférences Git

Niveau de configuration :

- Repository : au dépôt (comportement par défaut) (./git/config)
- Global : à la session utilisateur (~/.gitconfig)
- System : pour l'ensemble de la machine (/etc/gitconfig)

Définir des préférences :

`git config <niveau> <name> <value>`

besoin d'aide ? `man git` ou `man git-config`

Définir son nom et son email

Pourquoi s'identifier :

Pour vos commits et ceci afin de tracer les auteurs, Vous devez commencer par définir vos paramètres personnels

Comment :

```
git config --global user.name "Mickael Gaillard"
```

```
git config --global user.email "mickael.gaillard@tactfactory.com"
```

L'éditeur et l'outil de différentiel

L'éditeur : Pour les messages de commit

git config --global core.editor <editor>

Linux : nano ou gedit

Windows : 'C:/Program Files/Notepad++/notepad++.exe'

L'outil de différentiel : En cas de conflit lors d'un merge

git config --global merge.tool <diff_editor>

Exemple : kdiff3, tkdiff, meld, xxdiff, emerge, vimdiff, gvimdiff, ecmerge, et opendiff

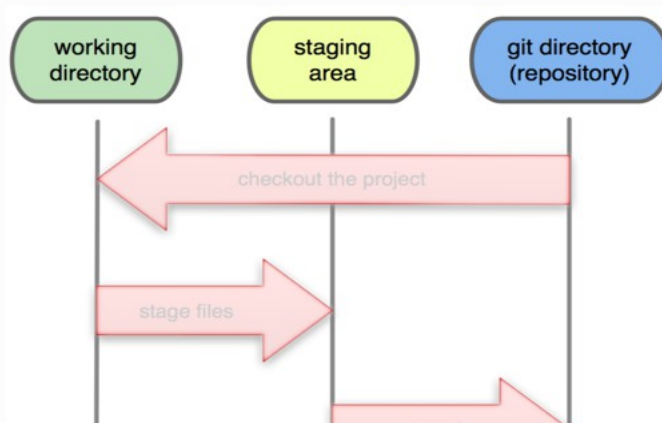
Initialiser un dépôt

Détecter un dépôt :

Recherche d'un sous-dossier « `.<database_folder>` »

Exemple : `.git`, `.svn`

Initialiser le dépôt : `git init`



```
micky@uranium:~$ mkdir learning_git
micky@uranium:~$ cd learning_git/
micky@uranium:~/learning_git$ git init
Initialized empty Git repository in /home/micky/learning_git/.git/
micky@uranium:~/learning_git(master #)$ ls -lFa
total 32
drwxr-xr-x  3 micky micky 4096 avril  3 16:07 ./
drwxr-xr-x 148 micky micky 20480 avril  3 16:06 ../
drwxr-xr-x  7 micky micky 4096 avril  3 16:07 .git/
micky@uranium:~/learning_git(master #)$
```


Architecture d'un dépôt

Branches (deprecated)

Config

Description (for gitweb)

HEAD (ref of current branch)

Hook (command trigger)

Info (extend description)

Objects (data stored with SHA-1)

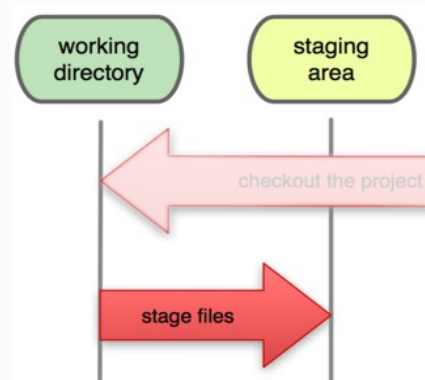
Refs (heads, remotes and tags)

```
micky@uranium:~/learning_git(master #)$ tree .git
.git
|-- branches
|-- config
|-- description
|-- HEAD
|-- hooks
|   |-- applypatch-msg.sample
|   |-- commit-msg.sample
|   |-- post-update.sample
|   |-- pre-applypatch.sample
|   |-- pre-commit.sample
|   |-- prepare-commit-msg.sample
|   |-- pre-push.sample
|   |-- pre-rebase.sample
|   |-- update.sample
|-- info
|   |-- exclude
|-- objects
|   |-- info
|   |-- pack
|-- refs
|   |-- heads
|   |-- tags
9 directories, 13 files
```

Ajouter des fichiers (status - add)

Ajouter un nouveau fichier à la zone de staging (index) :
git add <file>

Connaître l'état de la zone de staging :
git status

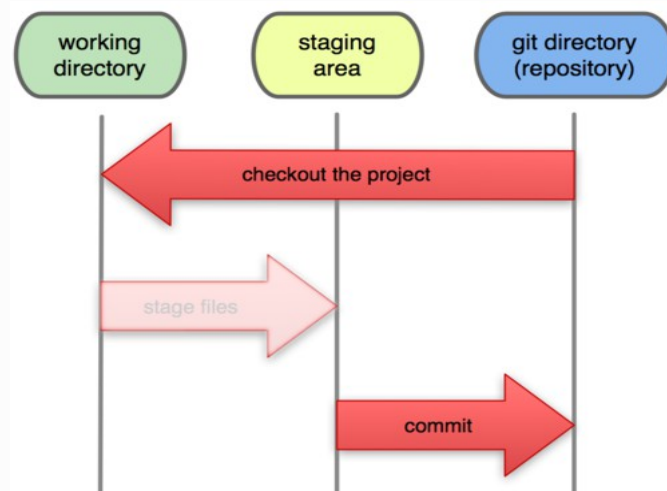


```
micky@uranium:~/learning_git(master #)$ touch Readme.txt
micky@uranium:~/learning_git(master #)$ ls -lFa
total 32
drwxr-xr-x  3 micky micky 4096 avril  3 16:17 ./
drwxr-xr-x 148 micky micky 20480 avril  3 16:06 ../
drwxr-xr-x  7 micky micky 4096 avril  3 16:07 .git/
-rw-r--r--  1 micky micky   0 avril  3 16:17 Readme.txt
micky@uranium:~/learning_git(master #)$ git status
# On branch master
#
# Initial commit
#
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#       Readme.txt
nothing added to commit but untracked files present (use "git add" to track)
micky@uranium:~/learning_git(master #)$ git add Readme.txt
micky@uranium:~/learning_git(master #)$ git status
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   (use "git rm --cached <file>..." to unstage)
#
#       new file:   Readme.txt
#
```

Ajouter des fichiers (commit - log)

Historiser les modifications dans le dépôt :
`git commit [-m "message"]`

Connaître l'état du dépôt :
`git log`



```
micky@uranium:~/learning_git(master #)$ git status
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   (use "git rm --cached <file>..." to unstage)
#
#       new file:   README.txt
#
micky@uranium:~/learning_git(master #)$ git commit -m "Initial commit"
[master (root-commit) 5c26b0a] Initial commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.txt
micky@uranium:~/learning_git(master)$ git status
# On branch master
nothing to commit, working directory clean
micky@uranium:~/learning_git(master)$ git log
commit 5c26b0a495b2da9fb10e2ae6ddcd888036b9d726
Author: Mickael Gaillard <mickael.gaillard@tactfactory.com>
Date:   Wed Apr 3 16:22:19 2013 +0200

    Initial commit
```

Ajouter des fichiers (remote - push)

Ajout d'un dépôt distant : `git remote add <server_name> <URL>`

Synchronisation des dépôts : `git push <server_name> <branch>`

The screenshot shows a GitHub repository page for 'TACTfactory / learning_git'. The page includes navigation tabs for Code, Network, Pull Requests (0), Issues (0), Wiki, Graphs, and Settings. Below these, there's a section for cloning the repository with options for ZIP, HTTP, SSH, and Git Read-Only. The SSH URL is 'git@github.com:TACTfactory/learning_git.git'. A dropdown menu shows 'branch: master'. The 'Files' tab is active, showing an 'Initial commit' by 'Theosakamg' 23 minutes ago, with a file named 'Readme.txt'.

Overlaid on the bottom right is a terminal window titled 'micky@uranium: ~/learning_git'. It shows the following commands and output:

```
micky@uranium:~/learning_git$ git remote add origin git@github.com:TACTfactory/learning_git.git
micky@uranium:~/learning_git(master)$ git push origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 225 bytes, done.
Total 3 (delta 0), reused 0 (delta 0)
To git@github.com:TACTfactory/learning_git.git
 * [new branch]      master -> master
micky@uranium:~/learning_git(master)$
```

Cloner un dépôt (clone - fetch)

Récupérer un dépôt :
git clone <URL>

Synchronisation générale
git fetch --all

```
micky@uranium:~$ git clone git://github.com/TACTfactory/learning_git.git
Cloning into 'learning_git'...
remote: Counting objects: 3, done.
remote: Total 3 (delta 0), reused 3 (delta 0)
Receiving objects: 100% (3/3), done.
micky@uranium:~$ cd learning_git/
micky@uranium:~/learning_git(master)$ ls -lFa
total 32
drwxr-xr-x  3 micky micky  4096 avril  3 16:51 ./
drwxr-xr-x 148 micky micky 20480 avril  3 16:51 ../
drwxr-xr-x  8 micky micky  4096 avril  3 16:51 .git/
-rw-r--r--  1 micky micky    0 avril  3 16:51 Readme.txt
micky@uranium:~/learning_git(master)$ git status
# On branch master
nothing to commit, working directory clean
micky@uranium:~/learning_git(master)$ git log
commit 5c26b0a495b2da9fb10e2ae6ddcd888036b9d726
Author: Mickael Gaillard <mickael.gaillard@tactfactory.com>
Date:   Wed Apr 3 16:22:19 2013 +0200

    Initial commit
micky@uranium:~/learning_git(master)$ git fetch --all
Fetching origin
```

Ignorer des fichiers

Exclure des fichiers du
depots : **.gitignore**

Avec les **règles de construction** suivante

: un commentaire

regex : règle d'exclusion

/ : un dossier complet

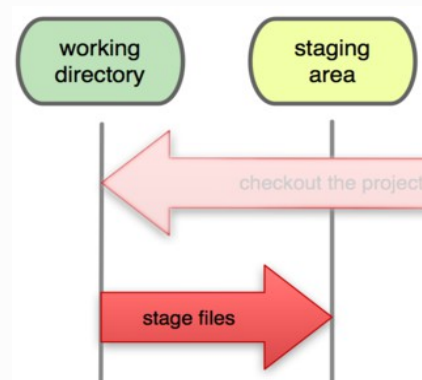
! : inclusion

```
micky@uranium:~/learning_git(master)$ ls -lFa
total 36
drwxr-xr-x  3 micky micky 4096 avril  3 17:10 ./
drwxr-xr-x 148 micky micky 20480 avril  3 17:06 ../
drwxr-xr-x  8 micky micky 4096 avril  3 17:09 .git/
-rw-r--r--  1 micky micky  108 avril  3 17:07 MyApp.java
-rw-r--r--  1 micky micky   0 avril  3 16:51 Readme.txt
micky@uranium:~/learning_git(master)$ javac -g MyApp.java
micky@uranium:~/learning_git(master)$ git status
# On branch master
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#       MyApp.class
#       MyApp.java
nothing added to commit but untracked files present (use "git add" to track)
micky@uranium:~/learning_git(master)$ echo "*.class" > .gitignore
micky@uranium:~/learning_git(master)$ git status
# On branch master
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#       .gitignore
#       MyApp.java
nothing added to commit but untracked files present (use "git add" to track)
micky@uranium:~/learning_git(master)$ git add .
micky@uranium:~/learning_git(master +)$ git commit -m "Add my first class and
[master f47781a] Add my first class and git ignore *.class binary
2 files changed, 10 insertions(+)
create mode 100644 .gitignore
create mode 100644 MyApp.java
```

Supprimer, déplacer, renommer

Supprimer un fichier à la zone de staging :
`git rm <file>`

Déplacer/Renommer un fichier de la zone de staging :
`git mv <source> <destination>`



```
micky@uranium:~/learning_git(master)$ ls -lFa
total 44
drwxr-xr-x  3 micky micky 4096 avril  3 17:20 ./
drwxr-xr-x 148 micky micky 20480 avril  3 17:06 ../
drwxr-xr-x  8 micky micky 4096 avril  3 17:19 .git/
-rw-r--r--  1 micky micky   8 avril  3 17:10 .gitignore
-rw-r--r--  1 micky micky  420 avril  3 17:10 MyApp.class
-rw-r--r--  1 micky micky  108 avril  3 17:18 MyApp.java
-rw-r--r--  1 micky micky   0 avril  3 17:19 Readme.txt
micky@uranium:~/learning_git(master)$ git mv Readme.txt Readme.md
micky@uranium:~/learning_git(master +)$ git rm MyApp.java
rm 'MyApp.java'
micky@uranium:~/learning_git(master +)$ git status
# On branch master
# Your branch is ahead of 'origin/master' by 1 commit.
#   (use "git push" to publish your local commits)
#
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       deleted:    MyApp.java
#       renamed:    Readme.txt -> Readme.md
#
micky@uranium:~/learning_git(master +)$ ls -lFa
total 40
drwxr-xr-x  3 micky micky 4096 avril  3 17:21 ./
drwxr-xr-x 148 micky micky 20480 avril  3 17:06 ../
drwxr-xr-x  8 micky micky 4096 avril  3 17:21 .git/
-rw-r--r--  1 micky micky   8 avril  3 17:10 .gitignore
-rw-r--r--  1 micky micky  420 avril  3 17:10 MyApp.class
-rw-r--r--  1 micky micky   0 avril  3 17:19 Readme.md
micky@uranium:~/learning_git(master +)$ git checkout HEAD MyApp.java
micky@uranium:~/learning_git(master +)$ git status
# On branch master
# Your branch is ahead of 'origin/master' by 1 commit.
#   (use "git push" to publish your local commits)
#
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       renamed:    Readme.txt -> Readme.md
#
```

Consulter l'historique

En console visualiser les commits :
`git log [--graph] [--stat]`

Usage avancé : Gistat & Git-Stat

Interface Graphique :

gitk, git-gui, gitg, gitextensions, git for windows,
TortoiseGit

```
micky@uranium:~/learning_git(master)$ git log
commit 3000ebca4b804be41999723174ddb00831b3dde6
Author: Mickael Gaillard <mickael.gaillard@tactfactory.com>
Date:   Wed Apr 3 17:24:17 2013 +0200

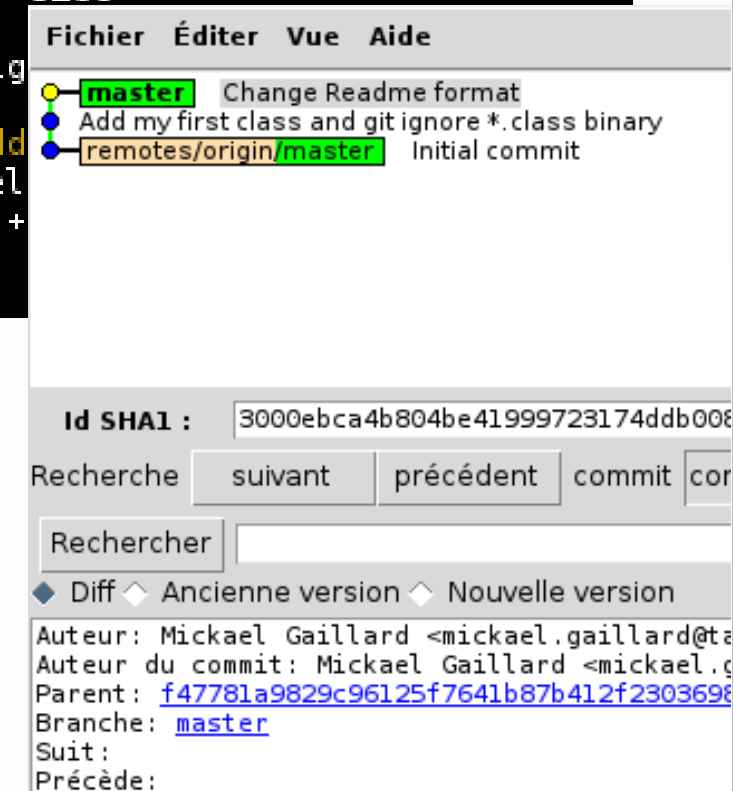
    Change Readme format

commit f47781a9829c96125f7641b87b412f23036980a0
Author: Mickael Gaillard <mickael.gaillard@tactfactory.com>
Date:   Wed Apr 3 17:12:12 2013 +0200

    Add my first class and git ignore *.class binary

commit 5c26b0a495b2da9fb10e2ae6dd
Author: Mickael Gaillard <mickael.gaillard@tactfactory.com>
Date:   Wed Apr 3 16:22:19 2013 +0200

    Initial commit
```



Annuler des modifications

Dans Git on ne fait qu'ajouter du contenu !!

Ajouter des fichiers au précédent commit

git commit --amend

Revenir au dernier commit (vider l'index)

git checkout -- <file_to_restore>

Revenir à un commit

git checkout <commit_hash>

Concepts Avancés

Tag (les releases officielles)

Branche (la révolution!)

Merge

Résolution de conflits (outil de différentiel)

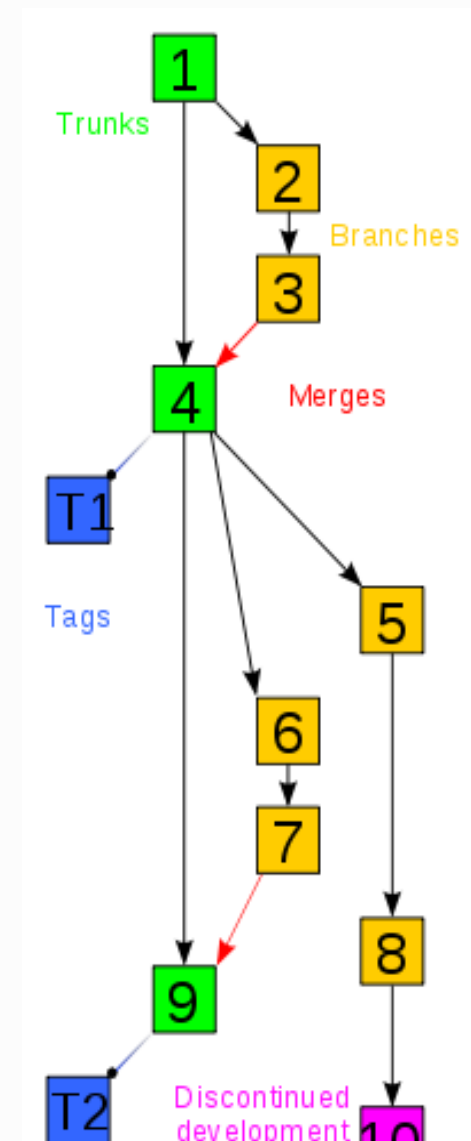
Workflow de Git

Référence :

<http://git-scm.com/book/fr/>

<http://git-scm.com/downloads/guis>

<https://git.wiki.kernel.org/index.php/InterfacesFrontendsAndTools>



Questions ?

Creative Commons 2013 TACTfactory.

Change log

- Mickael Gaillard 2013 : Initial document
- Yoan Pintas 2013 : Add branch
- Mickael Gaillard 2013 : Separate branch doc