

Introduction à Git

Introduction

Qu'est-ce qu'un gestionnaire de versions ?

Introduction

Qu'est-ce qu'un gestionnaire de versions (VCS)
?

- Conserver des versions de fichiers et leurs arborescences
- Identifier une arborescence de versions de fichiers
- Fournir les outils de gestion associés

Introduction

Qu'est-ce que Git ?

Introduction

Qu'est-ce que Git ?

Git est un gestionnaire de versions décentralisé

Introduction

Un gestionnaire de versions décentralisé (DCVS) permet :

- Calculer les différences entre deux versions (diff / patch)
- Un gestionnaire d'historiques des diff

Git

Créé par Linux Torvalds pour versionné le noyau Linux

Devenu le VCS le plus populaire

Utilisé par toutes les plateformes (GitHub, Bitbucket, ...)

Fonctionnalités

Présentation

Il existe 3 versions d'un même fichier :

- locale
- commit
- remote

Présentation

Git stocke les versions dans le répertoire .git

Un fichier modifié sera commité dans .git et
envoyer sur un serveur distant

Il existe donc deux étapes pour partager du
code

init

La commande `init` permet de rendre un répertoire versionnable.

git init

Crée le répertoire `.git` dans lequel les fichiers sont versionnés.

clone

La commande clone permet de récupérer l'ensemble d'un dépôt distant

```
git clone <ip> <répertoire>
```

Toutes les modifications sur <ip> sont placées dans le répertoire <répertoire>

add

Lors de la création d'un nouveau fichier, il faut l'ajouter dans git.

git add "mon fichier"

Dès lors le fichier est dans l'état "stage" et peut être commité.

commit

Une fois le fichier référencé dans Git, un commit permet de stocker l'ensemble des modifications et créer une version unique.

git commit -m "Mon message de commit"

push

Afin d'envoyer un ensemble de commit vers un serveur distant, il faut push les commits.

git push origin master

fetch

Récupérer des modifications distants vers .git

git fetch origin master

Les commits distants sont récupérés mais ne sont pas appliqués sur le code.

merge

Un merge effectue une mise à jour OU une fusion.

git merge origin/master

Applique l'ensemble des commits à votre code

pull

Afin de faciliter la récupération de commit, la commande pull effectue un fetch / merge

git pull origin master

=

git fetch origin master

git merge origin/master

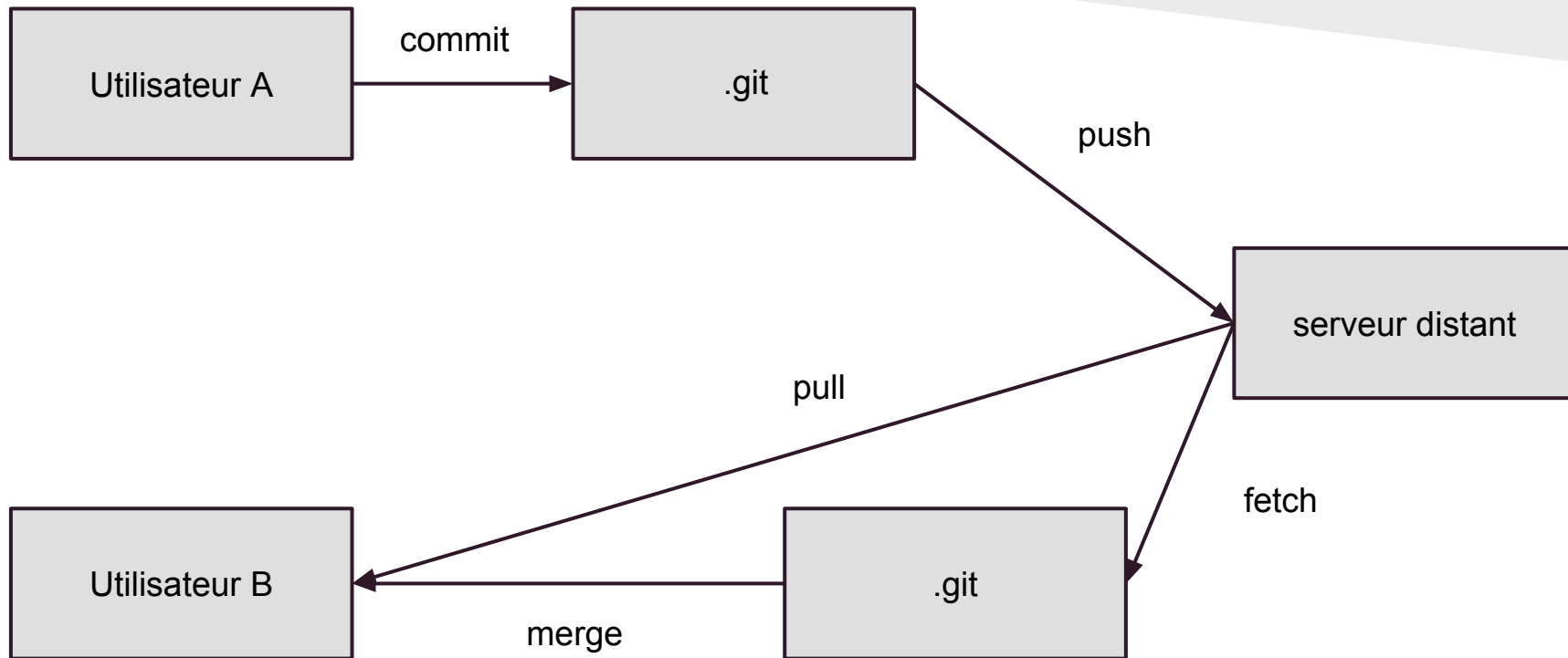
remote

Chaque serveur distant est identifié par un nom unique.

```
git remote add origin <ip>
```

Lors d'un push vous choisissez le serveur que vous désirez.

Vue globale



Branches

Une branche est une sous-partie du graphe des commits.

Un développement précis = une branche

Les branches peuvent être fusionnées entre elles

Branches

La branche par défaut s'appelle master.

git checkout <ma_branche> → Se déplacer vers une branche

git checkout -b <ma_branche> → Créer une nouvelle branche

Branches

Fusionner des branches se résument à donner le nom de la branche à merger.

git merge sprint25

Applique les commits de sprint25 dans la branche courante.

Workflow

Créer son environnement

```
echo "bonjour" > README.md
```

```
git init
```

```
git add README.md
```

```
git commit -m "Add REAMDE.md"
```

```
git remote add origin https://github.com/olivier-pitton/dant.git
```

```
git push origin master
```

Développement standard

```
git pull origin master
```

```
git add file1.java file2.java
```

```
git commit -m "Mes modifications de fichiers"
```

```
git fetch origin master
```

```
git merge origin/master
```

```
git push origin master
```

Fusion de branches

```
git checkout -b dant121
git add file1.java file2.java
git commit -m "Mes modifications de fichiers"
git fetch origin master
git checkout master
git merge origin/master
git merge dant121
git push origin master
```

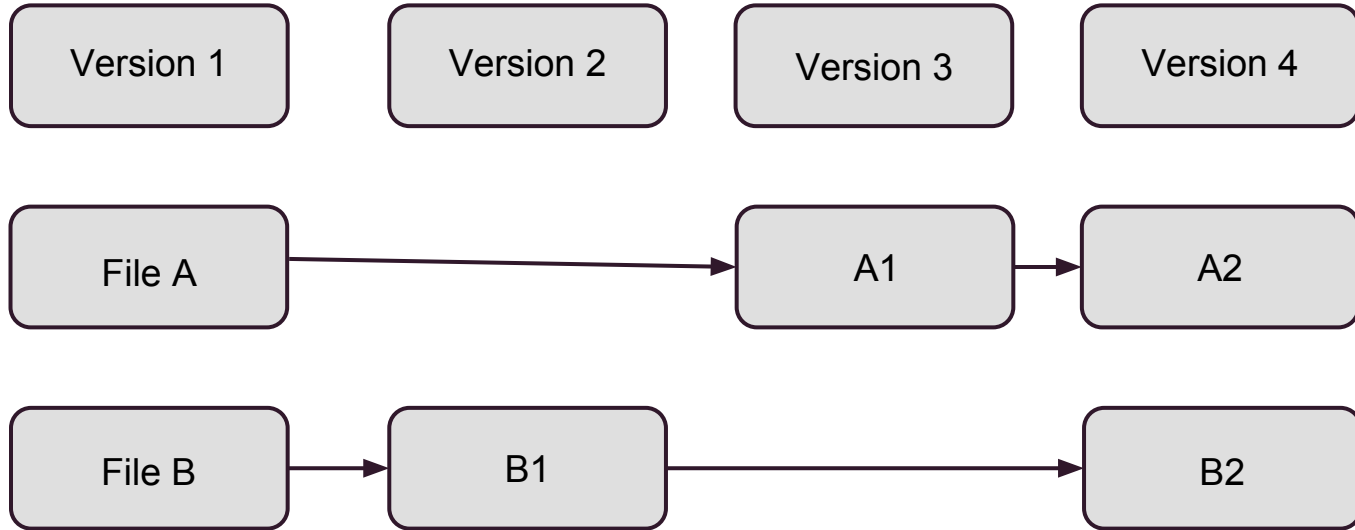
SVN vs Git

SVN vs Git

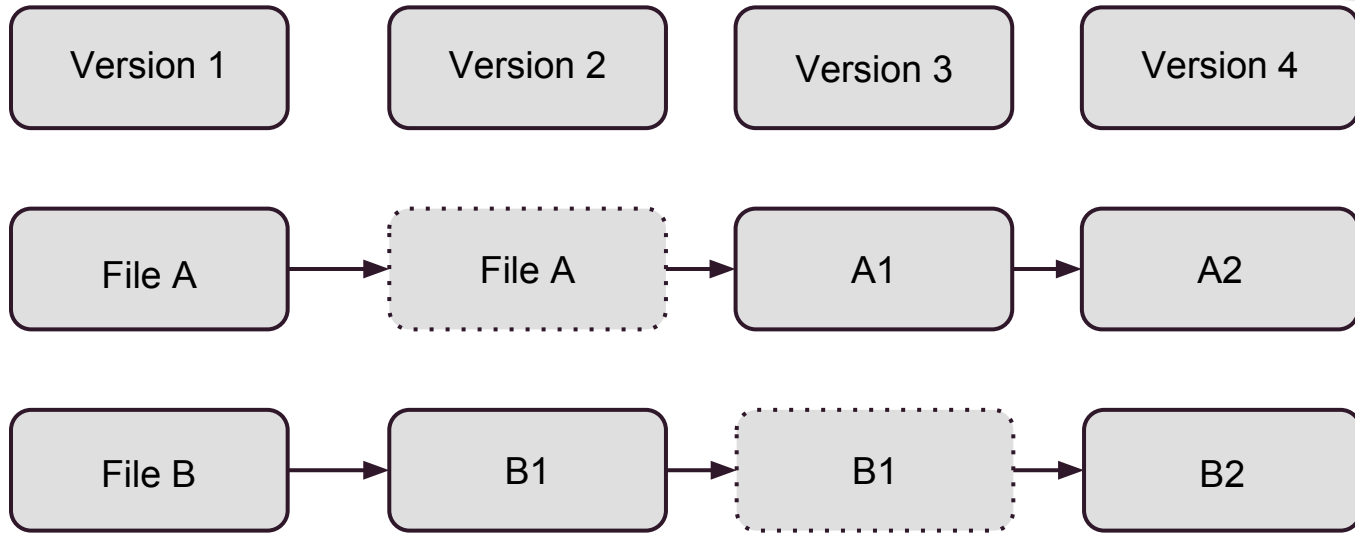
SVN est centralisé.

SVN conserve un état cohérent des versions
mais nécessite de réappliquer tous les patches
précédents !

SVN



Git



Conclusion

Git est utilisé dans toutes les entreprises

Facilite le développement logiciel
(contrairement à SVN)

Obligatoire à utiliser, via GitHub, pour le projet
!