

Chapitre 1 – Introduction

- **A- Pourquoi une application de Versioning ?**

- **Contexte**

Si vous avez déjà travaillé sur un projet informatique sans outils de versioning, que ce soit un petit projet personnel ou un plus gros projet professionnel, vous avez certainement déjà rencontré un de ces problèmes :

- « Qui a modifié le fichier X, il marchait bien avant et maintenant il provoque des bugs ! » ;
- « Arnaud, tu peux m'aider en travaillant sur le fichier X pendant que je travaille sur le fichier Y ? Attention à ne pas toucher au fichier Y car si on travaille dessus en même temps je risque d'écraser tes modifications ! » ;
- « Qui a ajouté cette ligne de code dans ce fichier ? Elle ne sert à rien ! » ;
- « À quoi servent ces nouveaux fichiers et qui les a ajoutés au code du projet ? » ;
- « Quelles modifications avons-nous faites pour résoudre le bug de la page qui se ferme toute seule ? »

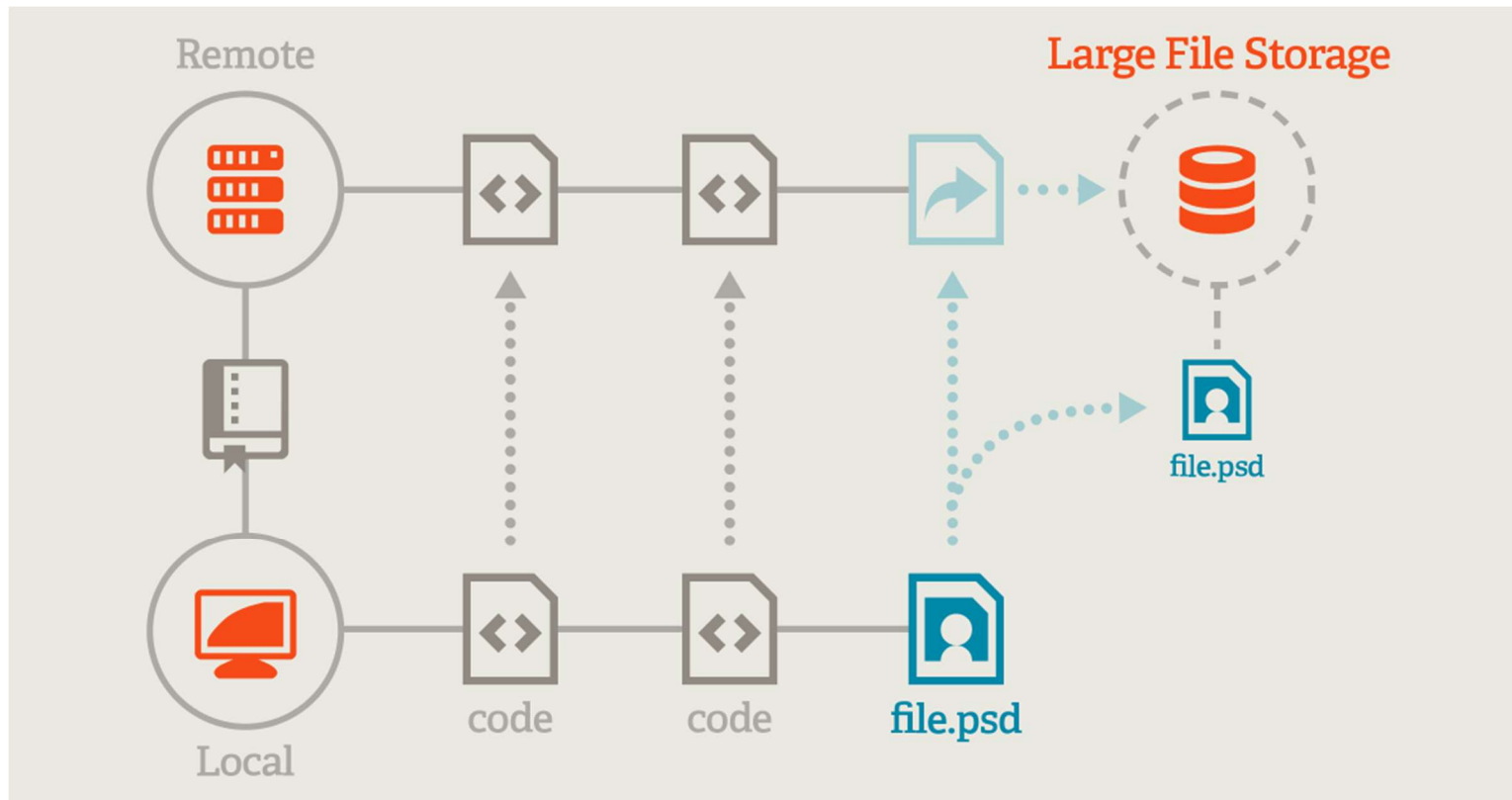


Dans ce cours, comme vous vous en doutez, je vous présenterai **Git** (prononcez « guite ») qui est un des plus puissants logiciels de ce genre. Qui utilise Git ? Ho de petites entreprises comme :



- Qu'est-ce que c'est ?

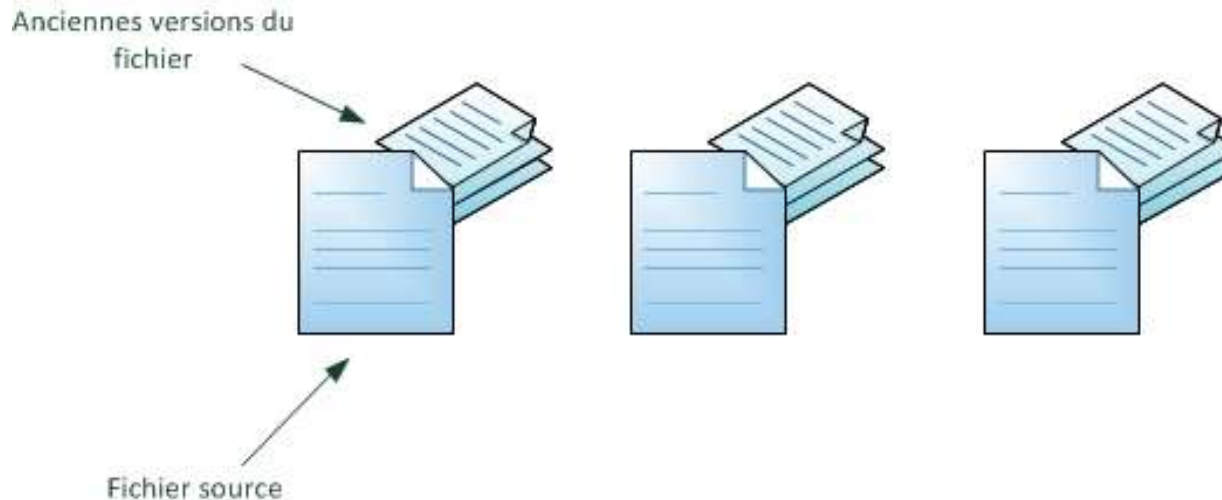
Ces outils suivent l'évolution de vos fichiers source et gardent les anciennes versions de chacun d'eux.



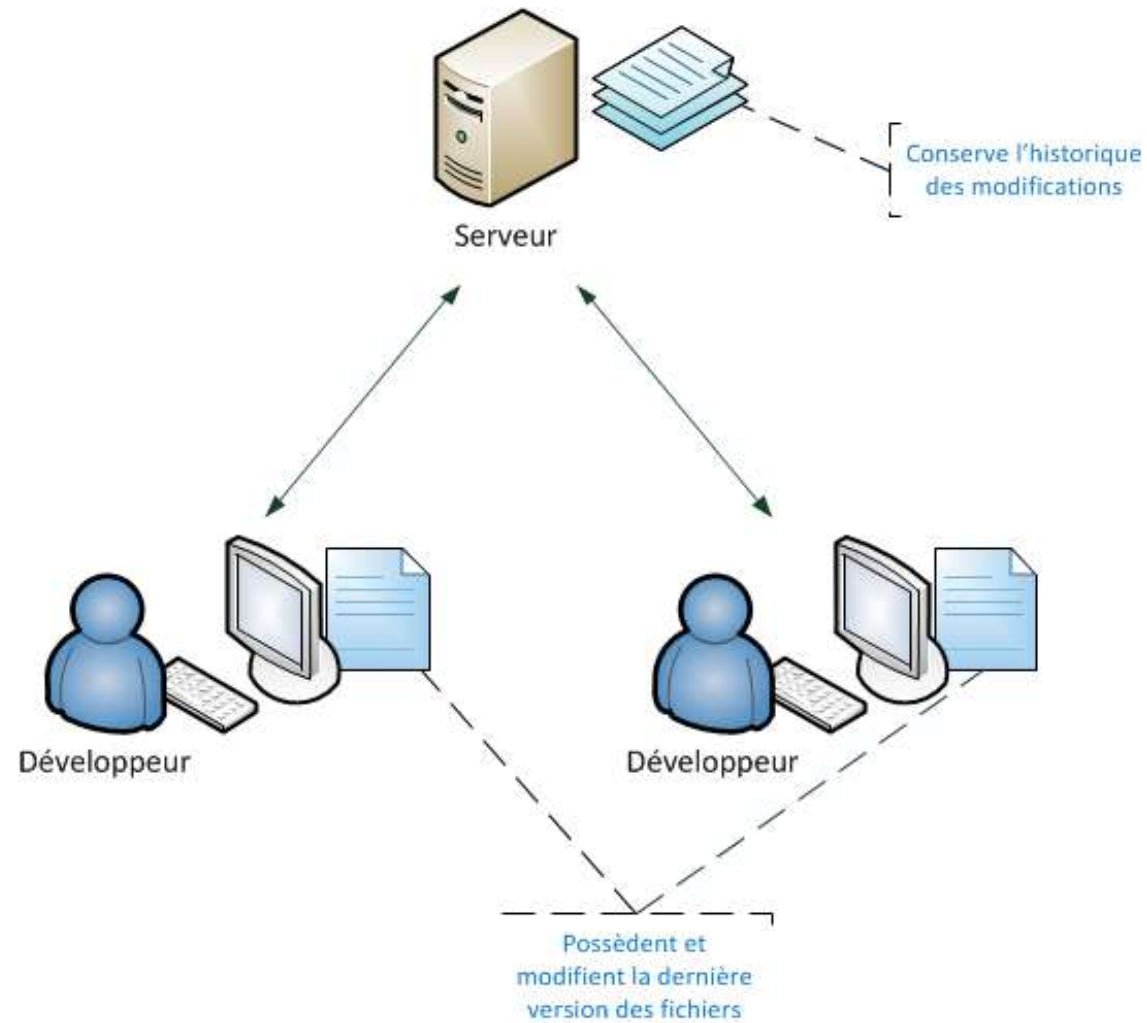
Ces logiciels ont donc par conséquent deux utilités principales :

- suivre l'évolution d'un code source, pour retenir les modifications effectuées sur chaque fichier et être ainsi capable de revenir en arrière en cas de problème ;
- travailler à plusieurs, sans risquer de se marcher sur les pieds. Si deux personnes modifient un même fichier en même temps, leurs modifications doivent pouvoir être fusionnées sans perte d'information.

○

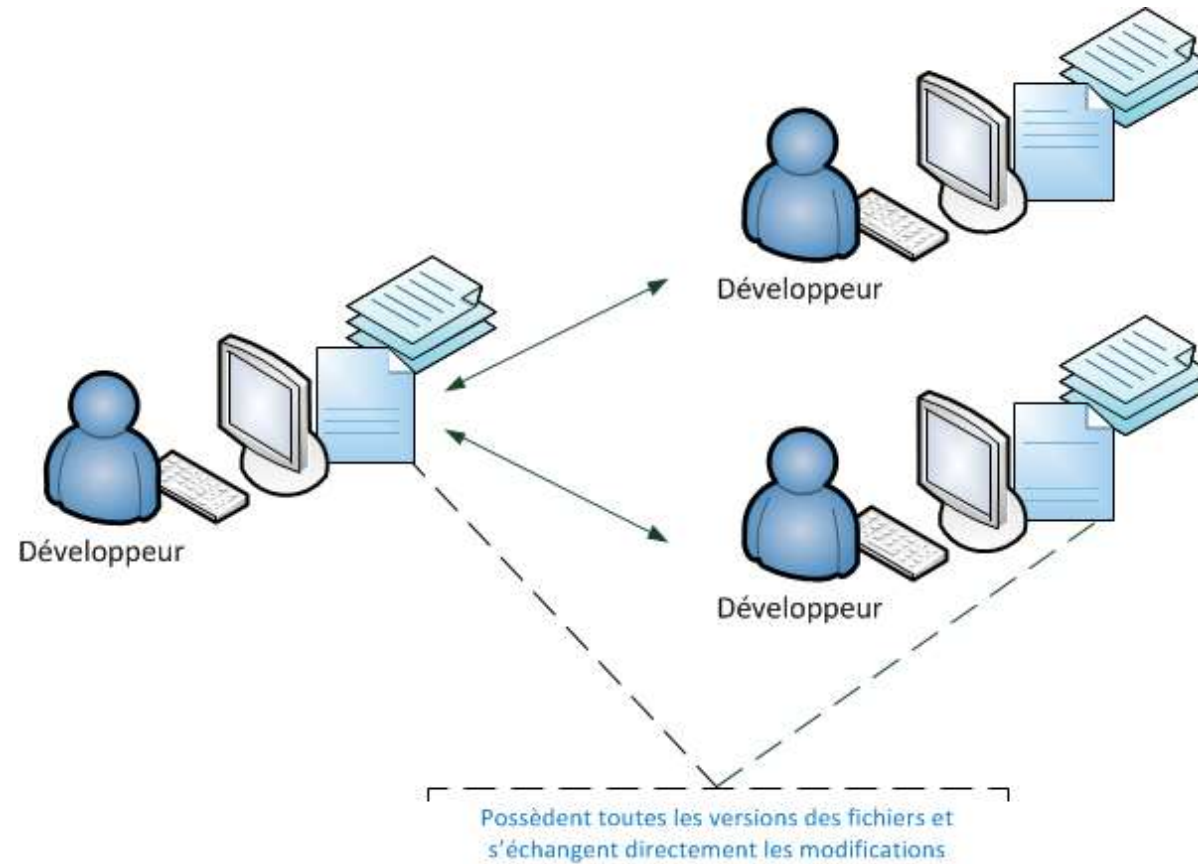


Voici, schématiquement, comment fonctionne un logiciel de gestion de versions centralisé :



Fonctionnement d'un logiciel de gestion de versions centralisé.

Voici le fonctionnement d'un logiciel de gestion de versions distribué :



Fonctionnement d'un logiciel de gestion de versions distribué

- **B- Historique**

- **Quelques dates clés**

- 1982 : RCS



- 1986 : CVS



- 2000 : SVN ou subversion



- 2005 : GIT, Mercurial, Bazar, Fossil



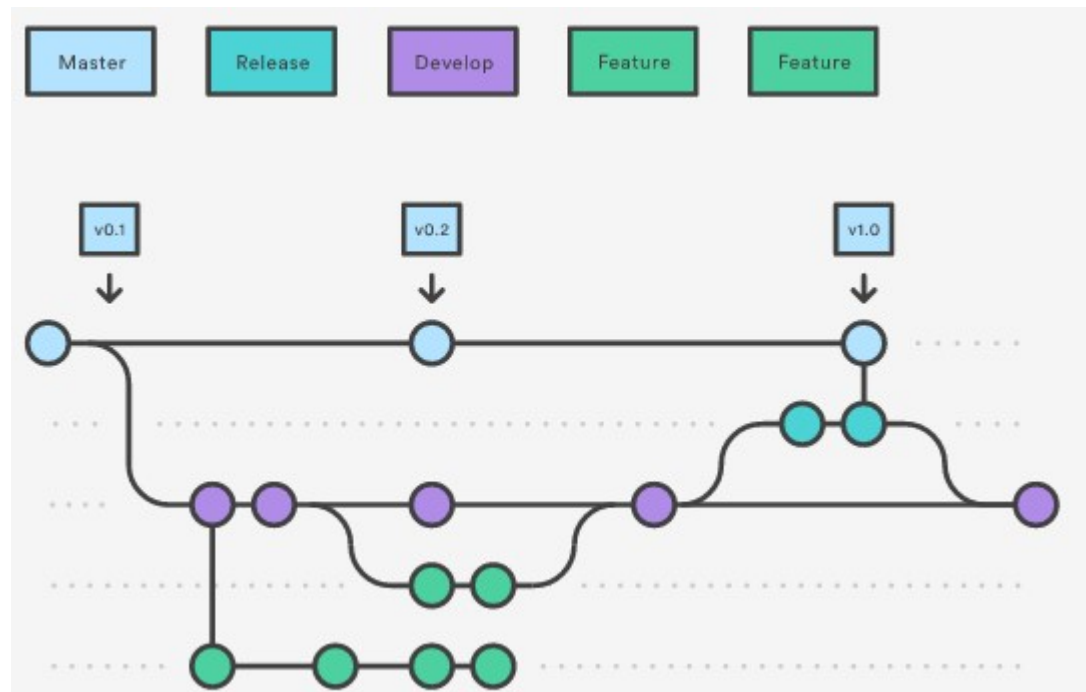
- Une histoire rapide de GIT

Comme de nombreuses choses extraordinaires de la vie, Git est né avec une dose de destruction créative et de controverse houleuse.



- Particularités techniques de GIT

Similaire en cela à BitKeeper, Git ne repose pas sur un serveur centralisé, mais il utilise un système de connexion pair à pair. Le code informatique développé est stocké non seulement sur l'ordinateur de chaque contributeur du projet, mais il peut également l'être sur un serveur dédié. C'est un outil de bas niveau, qui se veut simple et performant, dont la principale tâche est de gérer l'évolution du contenu d'une arborescence.



Représentation du travail collaboratif par branche

- Les différents logiciels de gestion de versions

Outil	Type	Description	Projets qui l'utilisent
CVS	Centralisé	C'est un des plus anciens logiciels de gestion de versions. Bien qu'il fonctionne et soit encore utilisé pour certains projets, il est préférable d'utiliser SVN (souvent présenté comme son successeur) qui corrige un certain nombre de ses défauts, comme son incapacité à suivre les fichiers renommés par exemple.	OpenBSD...
SVN (Subversion)	Centralisé	Probablement l'outil le plus utilisé à l'heure actuelle. Il est assez simple d'utilisation, bien qu'il nécessite comme tous les outils du même type un certain temps d'adaptation. Il a l'avantage d'être bien intégré à Windows avec le programme Tortoise SVN , là où beaucoup d'autres logiciels s'utilisent surtout en ligne de commande dans la console. Il y a un tutoriel SVN sur OpenClassrooms.	Apache, Redmine, Struts...
Mercurial	Distribué	Plus récent, il est complet et puissant. Il est apparu quelques jours après le début du développement de Git et est d'ailleurs comparable à ce dernier sur bien des aspects. Vous trouverez un tutoriel sur Mercurial sur OpenClassrooms.	Mozilla, Python, OpenOffice.org...
Bazaar	Distribué	Un autre outil, complet et récent, comme Mercurial. Il est sponsorisé par Canonical, l'entreprise qui édite Ubuntu. Il se focalise sur la facilité d'utilisation et la flexibilité.	Ubuntu, MySQL, Inkscape...
Git	Distribué	Très puissant et récent, il a été créé par Linus Torvalds, qui est entre autres l'homme à l'origine de Linux. Il se distingue par sa rapidité et sa gestion des branches qui permettent de développer en parallèle de nouvelles fonctionnalités.	Kernel de Linux, Debian, VLC, Android, Gnome, Qt...



- **Quizz 1 – 60 mn**

1-

2-

3-

4-

5-

6-

7-