

Introduzione

Durante lo sviluppo di progetti software, è indispensabile la **collaborazione**.

In altre parole, più persone devono poter *condividere* le modifiche al codice sorgente. Alcune delle operazioni sui file di testo sono:

- leggere il file in modo *concorrente* (da più utenti)
- scrivere modifiche sul file (gestire la scrittura simultanea)
- bloccare la modifica contemporanea dello stesso file (*lock*)
- scrivere commenti o descrizioni delle modifiche apportate
- gestire due salvataggi diversi con nomi diversi (*branch*)
- fusione di due modifiche (*merge* o *patch*)
- leggere la storia delle modifiche di un file (*history*)
- conservare la data delle modifiche
- conservare il nome dell'autore delle modifiche
- possibilità di annullare le modifiche (*revert* o *roll back*)

L'immagine a destra mostra un esempio di queste attività

https://en.wikipedia.org/wiki/File:Revision_controlled_project_visualization-2010-24-02.svg

Questo sistema di collaborazione e salvataggio dei dati può essere utilizzato anche per altri scopi, per esempio, anche quando invece di un gruppo di persone, c'è solo una sola persona che vuole scrivere un libro.

Anche i comuni software di videoscrittura (come MS Word oppure LibreOffice) permettono la collaborazione e la memorizzazione delle modifiche fatte da diversi autori, tuttavia i software usati dagli sviluppatori di codice sorgente sono più complessi e hanno più funzioni. Questi software sono detti VCS (**V**ersion **C**ontrol **S**ystem).

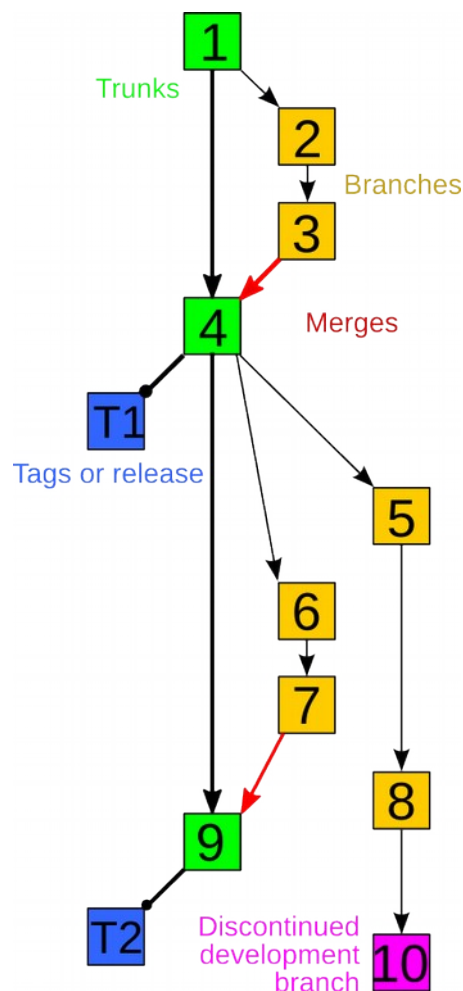
Alcuni esempi sono:

nome	tipo	protocolli	ultima versione
CVS	Concurrent		2009
Apache Subversion (SVN)	successore di CVS	http, webDAV	2017
Mercurial	Distributed	http, ssh	2017
Git	Distributed	https, ssh, git	2018
https://trac.edgewall.org/	è solo una interfaccia		

Oltre ad essere chiamati Version Control System, questi software possono essere anche chiamati **Source Control Management**.

Esistono due categorie di software:

- **Centralized** revision control (i dati si trovano su un unico server)
- **Distributed** revision control (si trovano copiati su più server, che devono essere sincronizzati)





Git gestisce le modifiche all'intero insieme di dati, invece che al singolo file.

Questo è meno intuitivo, ma facilita la gestione dei grandi progetti. Git possiede un proprio software client utilizzabile dal terminale, ma esistono anche interfacce web. Alcuni siti web che ospitano gratuitamente dei progetti di questo tipo sono:

- GitHub



- GitLab



Mediawiki è un altro software, molto noto, utile nel caso della scrittura di documenti con la collaborazione di più autori. Non è un software di versioning ma permette comunque di scrivere documenti con immagini e formattazione del testo. Una dimostrazione del suo utilizzo è il sito wikipedia.org

Software versioning

Spesso capita che quando viene pubblicato un software, e quando questo viene aggiornato, gli venga attribuito un nome e un numero (esempio: Debian GNU/Linux (v.10) Buster). Il processo di Software versioning (gestione delle versioni) è necessario quando si vuole associare un **nome** unico e un **numero** unico ad una certa fase dello sviluppo del software (questi numeri sono in ordine crescente).

Git è un VCS (Version Control System) per tenere traccia delle modifiche dei documenti elettronici che sono modificati da una squadra di programmatori.

Di solito, il programmatore, prima scarica (*clone*) una copia di tutti i file (e tutta la loro cronologia completa) che compongono il software. In altre parole, ottiene un *local repository* a partire dal un *remote repository*

Poi decide su quale ramo/versione lavorare (*checkout*), e si lavora su una copia locale (*working copy*).

La copia completa di un repository ad un certo istante è chiamata anche *snapshot* oppure *baseline*. Gli snapshot più significativi sono pubblicati con il nome di *release* oppure *milestone*.

Il programmatore può lavorare sul proprio computer, anche off-line.

Al termine delle modifiche l'utente "carica" la nuova versione, in modo da renderla disponibile anche agli altri utenti. Questa operazione viene detta "*commit*" oppure "*check in*"

Esercizio:

Sfogliare il menù di questo sito e spiegare le sue voci <https://trac.edgewall.org/>

Bibliografia

https://en.wikipedia.org/wiki/Revision_control

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

https://en.wikipedia.org/wiki/Apache_Subversion

<https://en.wikipedia.org/wiki/Mercurial>

<https://www.atlassian.com/git/tutorials/comparing-workflows>

<https://git-scm.com/docs/user-manual.html>

[https://en.wikipedia.org/wiki/Git_\(software\)](https://en.wikipedia.org/wiki/Git_(software))

<https://en.wikipedia.org/wiki/GitHub>

Fabio Proietti (c) 2017 Licenza: <http://creativecommons.org/licenses/by-sa/4.0/>

<https://en.wikipedia.org/wiki/GitLab>