



Python Biella Group

# Git Base

## **Docenti**

Andrea Guzzo

<https://andreaguzzo.com>

---

Anno 2020



# Agenda

Di cosa parliamo oggi



- 01** Introduzione VCS
- 02** GIT
- 03** Tutorial Pratico



# Python Biella Group

## JOIN US!

- **GitHub:** <https://github.com/PythonGroupBiella>
- **Telegram:** [https://t.me/joinchat/AAAAAFGSWcxhSln\\_SRhseQ](https://t.me/joinchat/AAAAAFGSWcxhSln_SRhseQ)

Tutto questo è stato reso possibile grazie a:

- Tutta la community di P.B.G = VOI!



# Perchè

## Ma questo mi serve davvero?

- Condividere codice, materiale digitale e lavorare in team
- Partecipare al mondo dell'open source
- Essenziale nel mondo del lavoro
- Collaborare
- Sicurezza e resilienza
- Tantissimo altro...

Strumento essenziale!





# Prima di iniziare...

## Visual Studio Code: Estensioni

- Git Lens
- Git History
- Git Graph

## Getting Started

- Installare Git sulla vostra macchina  
<https://git-scm.com/downloads>

## Cheatsheet

<https://i.redd.it/8341g68g1v7y.png>

## Tutorial

<https://learngitbranching.js.org/>





# Cos'è GIT?

Sistema di Controllo di Versione Distribuito (DVCS)



## VCS

- Quali cambiamenti sono stati fatti?
- Chi ha fatto quei cambiamenti?
- Quando sono stati fatti?
- Perché c'è stato bisogno di quei cambiamenti?

## Concetti principali

- **Repository** = metadata con la “storia” delle versioni
- **Commit** = snap del codice
- **Working copy** = copia di lavoro dell'utente

All'interno del controllo di versione risiedono tutti i files che consentono di riprodurre una build + codice sorgente

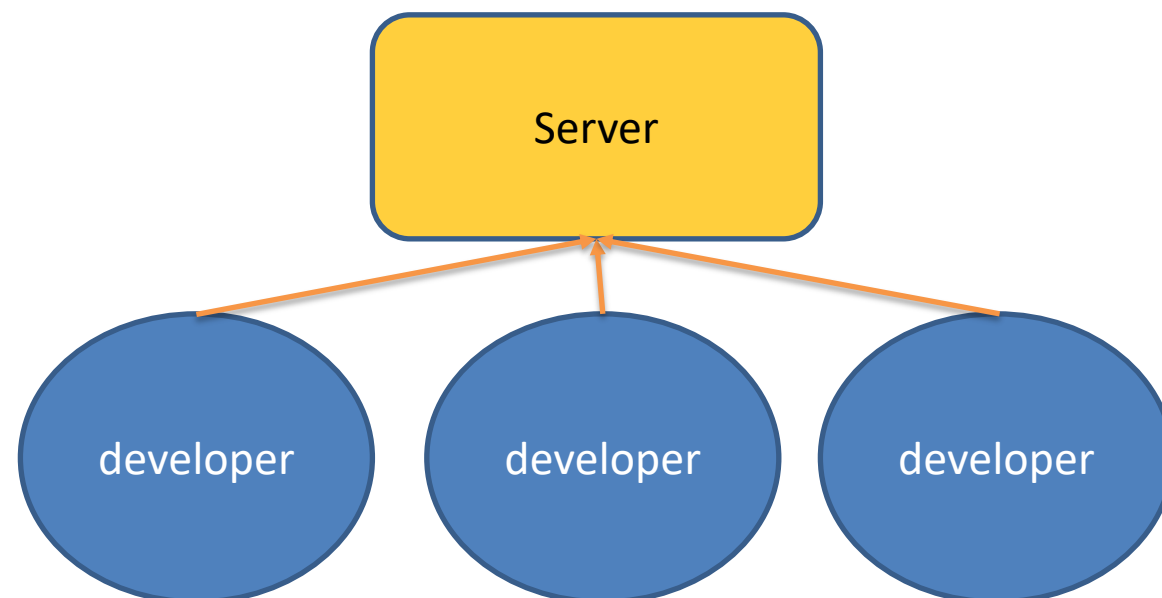
# Differenti VCS

Esistono molti tool che fungono da VCS: strumenti di controllo di versione



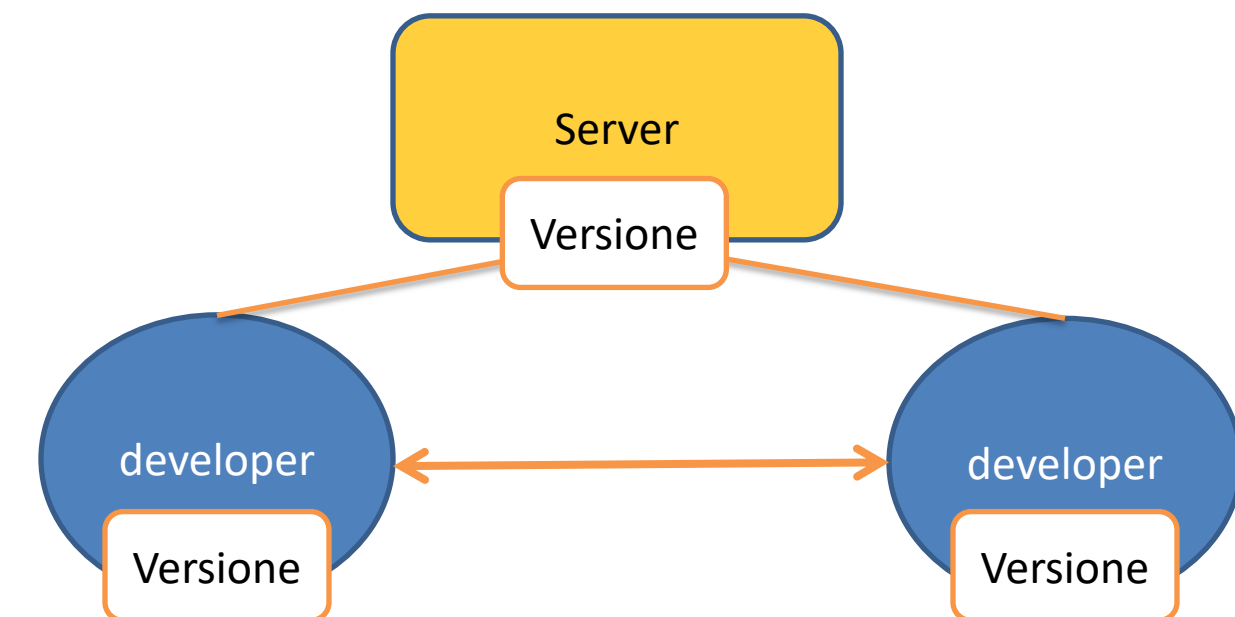
## Centralize VCS

- Ogni sviluppatore ha **solo la propria** working copy
- Quasi tutte le operazioni **richiedono** una connessione con il server (repository)
- **Modello ingessato** (collaborazione non facilitata)



## Distributed VCS

- Ogni sviluppatore ha la **copia completa** del repository con tutte le informazioni
- Le operazioni **non richiedono** sempre la connessione con il server
- **Modello flessibile**: modifiche condivise tra tutte le copie





# GIT

Distributed Version Control System (DVCS)



## Obiettivi

- Velocità
- Design semplice
- Tanto sviluppo in contemporanea
- Completamente distribuito
- Adatto per grandi progetti (velocità e dimensionamento dei dati)

## Caratteristiche

- Quasi tutte le operazioni sono locali
- Integrità dei dati
- Difficile perdere qualcosa
- Tanti modelli di sviluppo diversi
- Cross platform (universale)

Differenti servizi di hosting

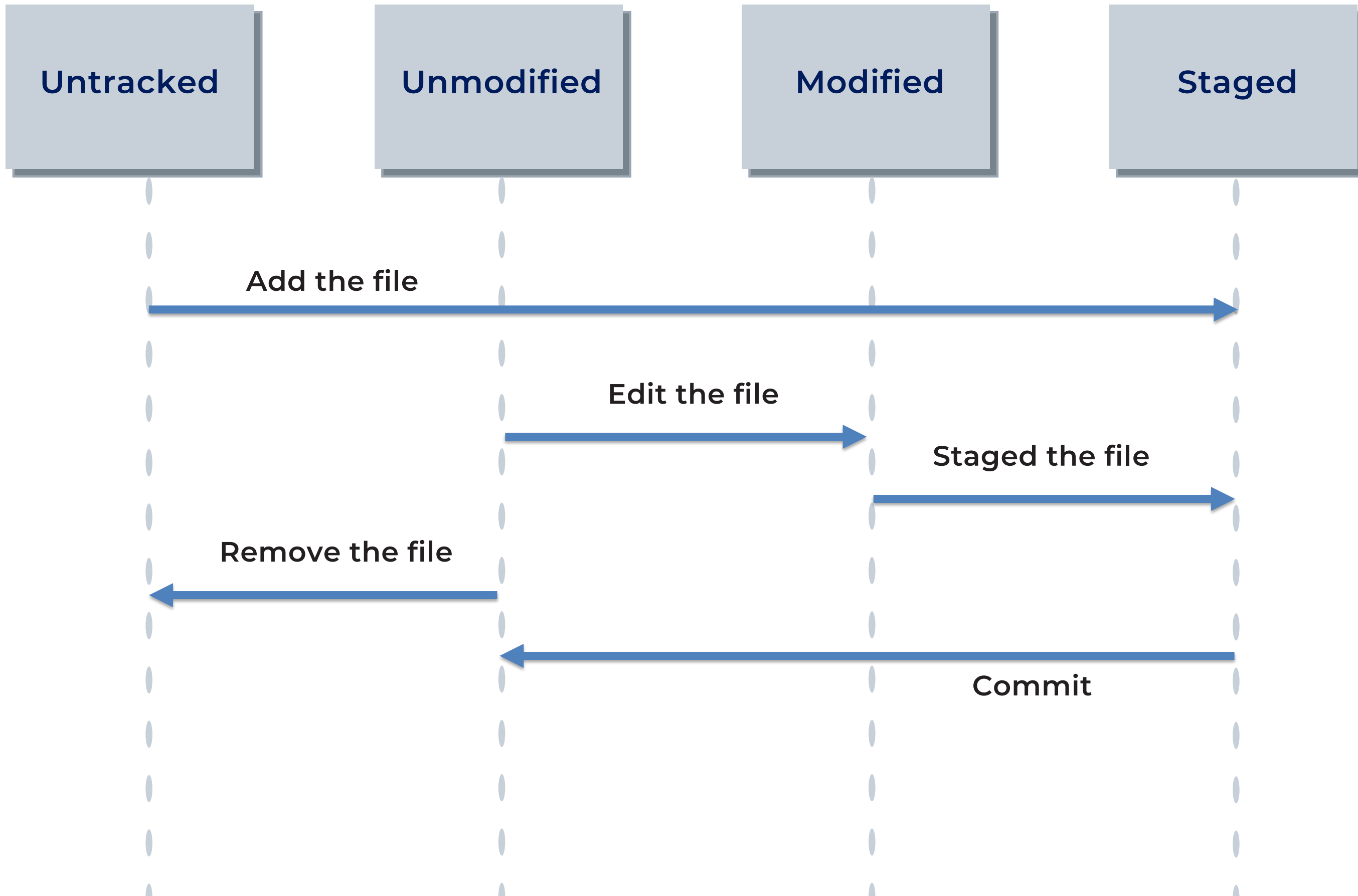
Github

Gitlab

Bitbucket



# Stati dei files





# Grazie a tutti!

- GitHub: <https://github.com/PythonGroupBiella/MaterialeLezioni>



study bunnies