

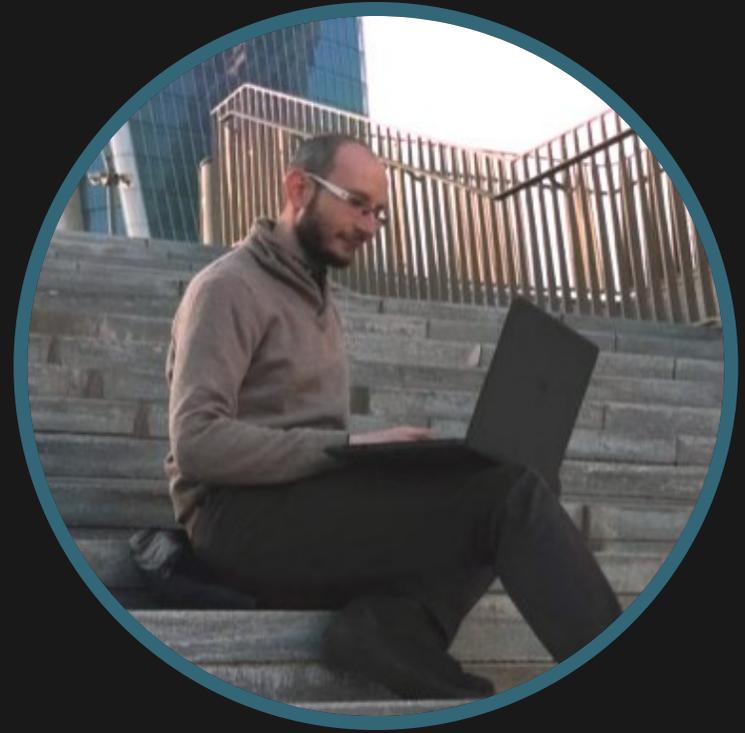


GIT COME NON L'AVETE MAI VISTO

Relatore: [Giovanni Sardisco](#) (aka gioisco)

GIOVANNI SARDISCO

(aka gioisco)



 Sviluppatore, principalmente **Java** e **Bash**

 Aiuto gli sviluppatori a destreggiarsi con **codice** e **command line**

 Mi piace farlo

COS'È GIT?

Un sistema di controllo di versione distribuito

PERCHÈ GIT?

Distribuito, affidabile e veloce



PERCHÈ È NATO?

Linux - 17 settembre 1991

Passarsi le patch via email...

BitKeeper - 2002–2005

Git - 2005

CHE CARATTERISTICHE DOVEVA AVERE

- **Velocità estrema:** doveva gestire decine di migliaia di file e commit senza rallentare.
- **Integrità dei dati:** ogni singolo pezzo di informazione doveva essere verificabile e immutabile.
- **Distribuzione reale:** ogni clone doveva essere un repository completo, indipendente.
- **Supporto per flussi di lavoro non lineari:** branching e merging dovevano essere operazioni naturali, non eccezioni

git init

```
1 git init my-repo
2 git config --local user.email "just.gioisco@gmail.com"
3 git config --local user.name "gioisco"
4
5 echo "Hello Git" > README.md
6 git add README.md
7 git commit -m "Initial commit"
```

INIZIO PROGETTO



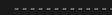
CREAZIONE BRANCH FEATURE

main

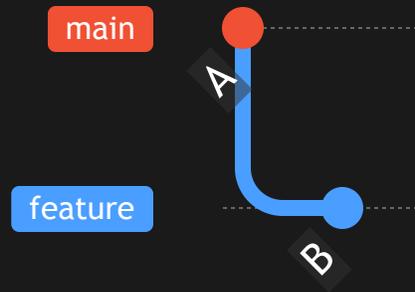


A

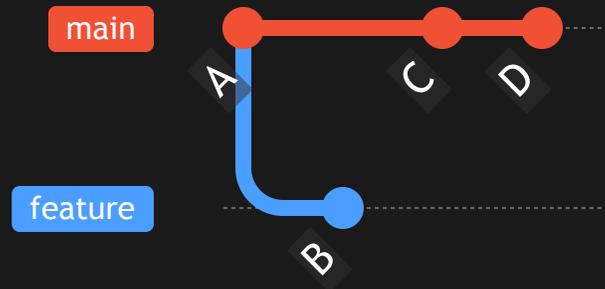
feature



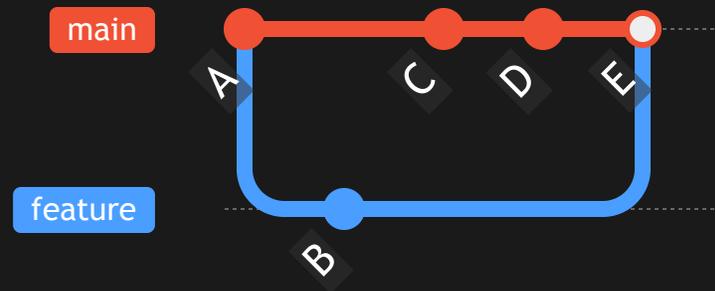
CREAZIONE BRANCH FEATURE



COMMIT PARALLELI SU MAIN



MERGE SU MAIN



IL REBASE

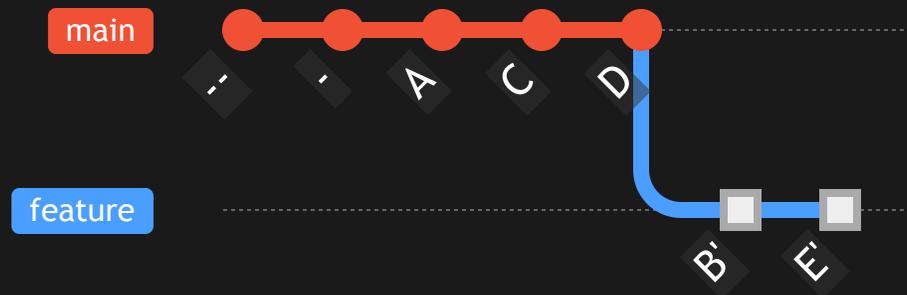
COMMIT SU FEATURE



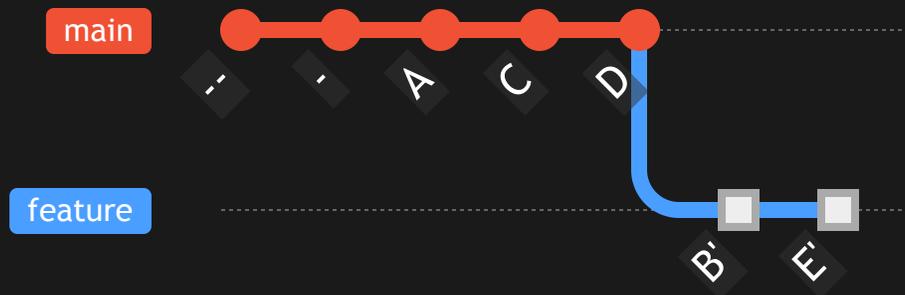
COMMIT PARALLELI SU MAIN



REBASE DI FEATURE SU MAIN



REBASE DI FEATURE SU MAIN



...ritornare al passato?

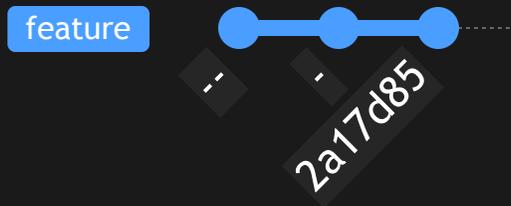
MACCHINA DEL TEMPO



ritorno al passato

```
1 git checkout feature
2 git status
3 git reset --hard 2a17d85
```

MACCHINA DEL TEMPO



MACCHINA DEL TEMPO



...ritorno al futuro?

MACCHINA DEL TEMPO



...ritorno al futuro?

```
1 git reset --hard d827b3f
```

MACCHINA DEL TEMPO



MACCHINA DEL TEMPO



...ritorno al passato meno drastico?

MACCHINA DEL TEMPO



...ritorno al passato meno drastico?

```
1 git reset --soft HEAD~
```

RISCRIVERE LA STORIA

dell'ultimo commit

```
1 git commit --amend
```

RISCRIVERE LA STORIA

cambiando l'ordine degli eventi

```
1 git rebase -i HEAD^3
```

MANTENENDO LA STORIA

andando avanti

```
1 git revert HEAD  
2  
3 git revert 2a17d85
```

AGGIUNGERE SOLO ALCUNE MODIFICHE

```
1 git add -p .
```

AGGIUNGERE SOLO ALCUNE MODIFICHE TRANNE...

```
1 git add . -p ':*bru'
```

git init --bare

```
1 git init --bare my-remote-repo
2
3 cd ../my-repo
4 git remote add origin ../my-remote-repo
5 git push
```

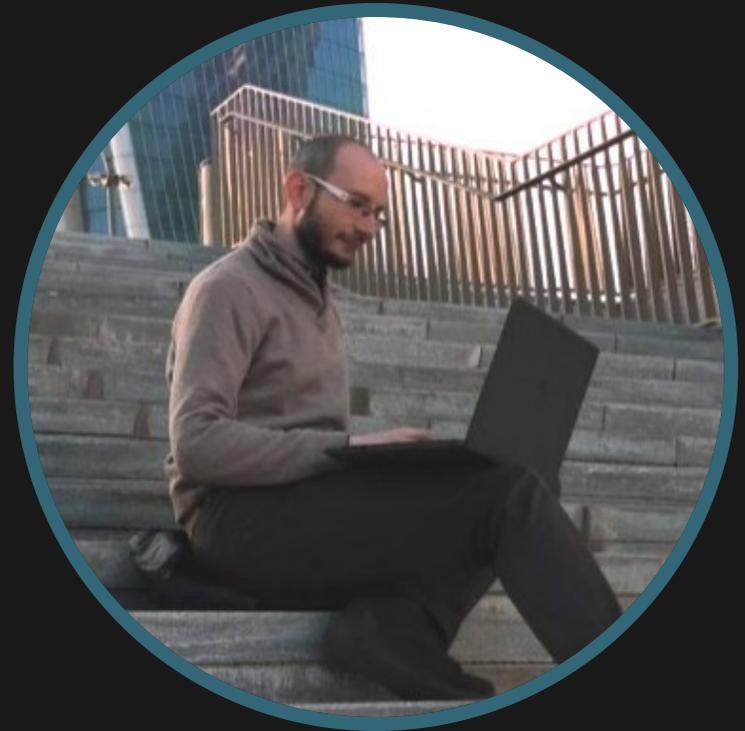
REVERT DI UN MERGE

```
1 git revert -m 1 hash-merge-commit
2
3 git checkout feature
4 git rebase main
5 git cherry-pick oldest_commit_feature^..latest_commit_feature
```

 Il mio articolo: [Storia del revert di un git merge](#)

GIOVANNI SARDISCO

(aka gioisco)



GRAZIE PER L'ATTENZIONE