



Merge e conflitti

Corso di Laboratorio di Gestione progetto e organizzazione d'impresa

Prof. Dei Rossi, Leonardo Essam

L'unione di due mondi (1)

Una delle operazioni principali di Git è l'operazione di **merge**.

Il merge in Git è l'operazione dove un ramo X viene ricongiunto a un ramo Y , riprendendo l'esempio della Supercell ad esempio:

- Branch brawler-mrp e game-footbrawl;
- Branch master.

Supponiamo ora di voler unire i due branch al master.



L'unione di due mondi (2)

Dalla console di Git, ci si posiziona nel branch di destinazione:

```
git checkout master
```

Ora segue il merge dei branch:

```
git merge brawler-mrp  
git merge game-footbrawl
```

L'unione di due mondi (3)

Ora Git domanderà di scrivere il perché dell'operazione di *merge*, questo perché viene generato un nuovo *commit* e ciò che viene scritto sarà il messaggio del commit!

```
leode@ThinkBook-14:~/prova-5inb3$ git log
commit e53cb5113af98a79fc0f1c698cd2578f371a2096 (HEAD -> master)
Merge: a63bc77 1da9977
Author: Leonardo Essam <leode@casamika.local>
Date:   Sun Oct 5 15:58:29 2025 +0200

    Merge branch 'branch3'
```

I mondi che collidono (1)

«The shorthand is "the butterfly effect". A butterfly can flap its wings in Peking, and in Central Park, you get rain instead of sunshine.»

– Ian Malcom, Jurassic Park

I mondi che collidono (2)

Quando si sviluppa un progetto può succedere che due o più sviluppatori vadano a modificare un file che è comune a entrambe le parti del progetto su cui si sta lavorando.

Ora, ricordando sempre la lezione precedente, in un contesto senza l'uso di un sistema di controllo di versione, una situazione di questo tipo potrebbe creare non pochi grattacapi!

I mondi che collidono (3)

In un contesto dove invece si usa un sistema VCS *serio*, queste situazioni vengono gestite e rilevate automaticamente dal VCS stesso.

Provate a svolgere l'esercizio.

Riassumendo... (1)

- **Commit:** ?
- **Branch:** ?
- **Merge:** ?

Riassumendo... (2)

- **Commit:** operazione dove si va a creare un nuovo nodo nel *tree* per registrare nel VCS le nuove modifiche svolte;
- **Branch:** ?
- **Merge:** ?

Riassumendo... (3)

- **Commit:** operazione dove si va a creare un nuovo nodo nel *tree* per registrare nel VCS le nuove modifiche svolte;
- **Branch:** una "copia" logica del progetto che serve per poter lavorare senza interferire con il codice principale;
- **Merge:** ?

Riassumendo... (4)

- **Commit:** operazione dove si va a creare un nuovo nodo nel *tree* per registrare nel VCS le nuove modifiche svolte;
- **Branch:** una "copia" logica del progetto che serve per poter lavorare senza interferire con il codice principale;
- **Merge:** operazione dove si va ad unire due rami del progetto per allineare le modifiche.

Patti chiari, amicizia lunga: non voglio sentire *traduzioni* fantasiose (i.e. *committare*, *branchare* e *mergiare*); siete dei tecnici e da voi ci si aspetta un linguaggio tecnico!

