



Introduzione a **Git** e **GitHub**



Vittorio Romeo

<http://vittorioromeo.info>

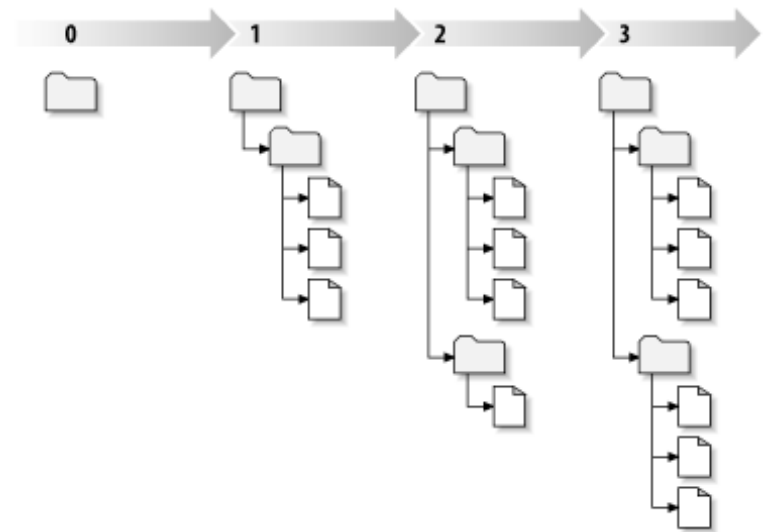
vittorio.romeo@outlook.com

Cosa è un VCS?



Cosa è un VCS?

- Un **VCS (Version Control System)** è un software dedicato alla gestione dei cambiamenti di **documenti, codici sorgenti, siti web** ed altre informazioni.



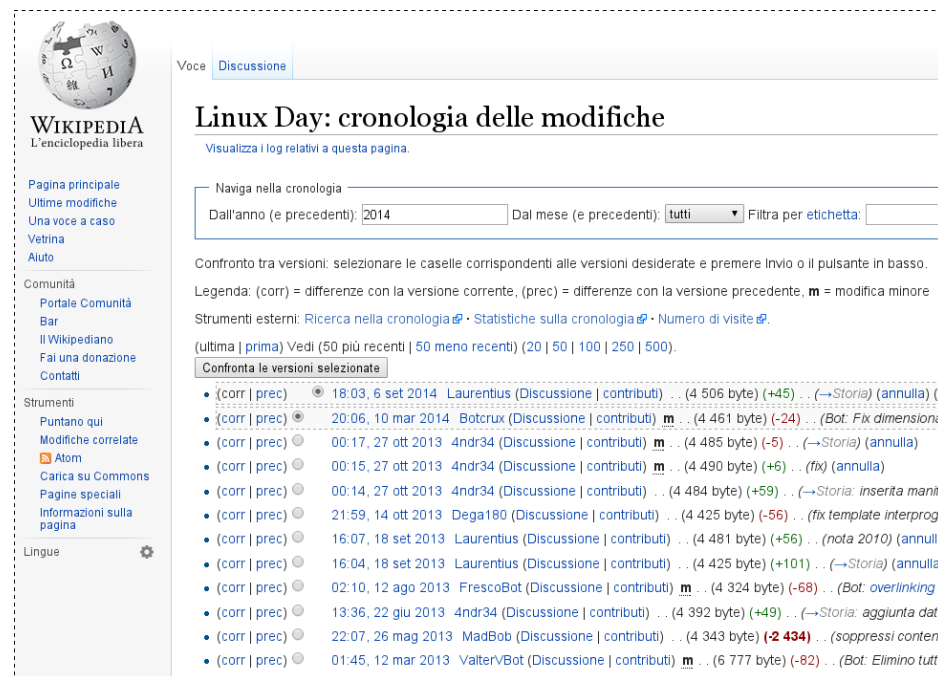
Cosa è un VCS?

- La funzionalità base di ogni VCS è il **salvataggio** in modo **incrementale** di dati.
- Il VCS offre all'utente la possibilità di **recuperare** versioni precedenti dei dati.



Perché usare un VCS?

- Backup di dati
- Collaborazione
- Testing/sviluppo
- Risoluzione di bug
- Backtracking
- ...



The screenshot shows the Wikipedia page for "Linux Day: cronologia delle modifiche". It displays a list of revisions with columns for date, time, user, and byte changes. The page includes navigation links like "Voce", "Discussione", and "Visualizza i log relativi a questa pagina". It also features a search bar for the history and a legend for the revision markers.

WIKIPEDIA
L'enciclopedia libera

Pagina principale
Ultime modifiche
Una voce a caso
Vetrina
Aiuto

Comunità
Portale Comunità
Bar
Il Wikipediano
Fai una donazione
Contatti

Strumenti
Puntano qui
Modifiche correlate
Atom
Carica su Commons
Pagine speciali
Informazioni sulla pagina

Lingue

Voce **Discussione**

Linux Day: cronologia delle modifiche

Visualizza i log relativi a questa pagina.

Naviga nella cronologia

Dall'anno (e precedenti): 2014 Dal mese (e precedenti): tutti Filtra per etichetta:

Confronto tra versioni: selezionare le caselle corrispondenti alle versioni desiderate e premere Invio o il pulsante in basso.

Legenda: (corr) = differenze con la versione corrente, (prec) = differenze con la versione precedente, m = modifica minore

Strumenti esterni: Ricerca nella cronologia · Statistiche sulla cronologia · Numero di visite

(ultima | prima) Vedi (50 più recenti | 50 meno recenti) (20 | 50 | 100 | 250 | 500).

Confronta le versioni selezionate

Modifica	18:03, 6 set 2014	Laurentius (Discussione contributi)	(4 506 byte) (+45)	(→Storia) (annulla)
(corr prec)	20:06, 10 mar 2014	Botcrux (Discussione contributi)	m (4 461 byte) (-24)	(Bot: Fix dimensiona
(corr prec)	00:17, 27 ott 2013	4ndr34 (Discussione contributi)	m (4 485 byte) (-5)	(→Storia) (annulla)
(corr prec)	00:15, 27 ott 2013	4ndr34 (Discussione contributi)	m (4 490 byte) (+6)	(fix) (annulla)
(corr prec)	00:14, 27 ott 2013	4ndr34 (Discussione contributi)	(4 484 byte) (+59)	(→Storia: inserita manife
(corr prec)	21:59, 14 ott 2013	Dega180 (Discussione contributi)	(4 425 byte) (-56)	(fix template interproge
(corr prec)	16:07, 18 set 2013	Laurentius (Discussione contributi)	(4 481 byte) (+56)	(nota 2010) (annulla
(corr prec)	16:04, 18 set 2013	Laurentius (Discussione contributi)	(4 425 byte) (+101)	(→Storia) (annulla)
(corr prec)	02:10, 12 ago 2013	FrescoBot (Discussione contributi)	m (4 324 byte) (-68)	(Bot: overlinking q
(corr prec)	13:36, 22 giu 2013	4ndr34 (Discussione contributi)	(4 392 byte) (+49)	(→Storia: aggiunta data
(corr prec)	22:07, 26 mag 2013	MadBob (Discussione contributi)	(4 343 byte) (-2 434)	(soppressi contenu
(corr prec)	01:45, 12 mar 2013	ValterVBot (Discussione contributi)	m (6 777 byte) (-82)	(Bot: Elimino tutti



Cosa è **Git**?

- **Git** è un **VCS** sviluppato da **Linus Torvalds**.
- E' nato dal bisogno di un nuovo **VCS** per lo sviluppo del kernel **Linux**.
- **Git** è un **VCS** “distribuito” o “peer to peer”.



Perché Git?

- **Non-lineare**
- **Distribuito**
- **Efficiente**
- **Scalabile**
- **Intelligente**
- **Semplice**
- **FOSS**
- **Popolare**



Cosa è GitHub?

- **GitHub** è la più grande e famosa piattaforma Git.
- GitHub è **gratuito**, ricco di **features**, **sociale**, **semplice** da usare, **portabile**, **sicuro**.
- <https://github.com>

GitHub

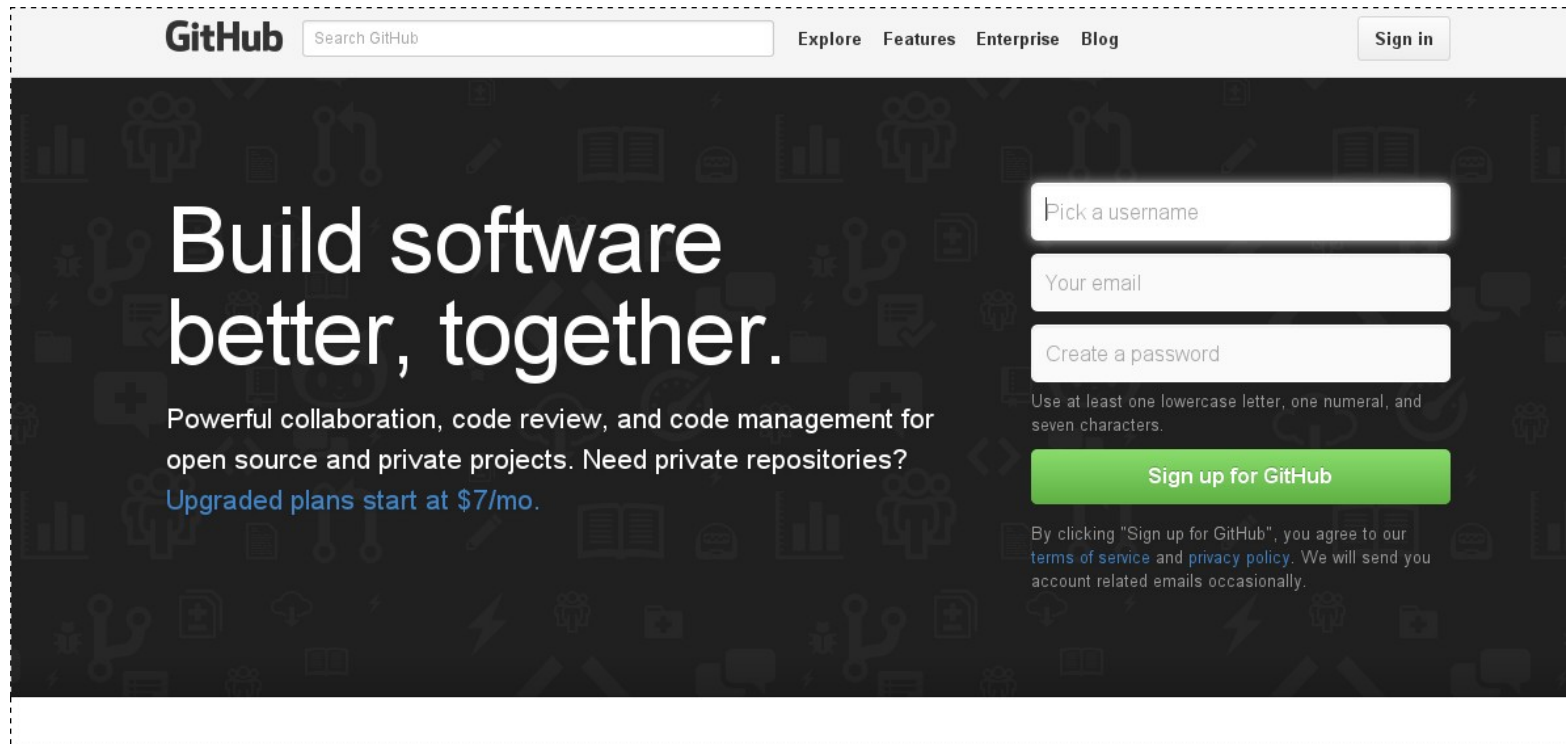


Dimostrazione **live**



Creare/clonare un repository

- Lavoreremo sin da subito in **remoto**, usando **GitHub**.



The screenshot shows the GitHub homepage with a dark background and a pattern of small icons. The main heading is "Build software better, together." Below it, a subheading reads: "Powerful collaboration, code review, and code management for open source and private projects. Need private repositories? Upgraded plans start at \$7/mo." On the right side, there is a sign-up form with three input fields: "Pick a username", "Your email", and "Create a password". Below the password field, a note states: "Use at least one lowercase letter, one numeral, and seven characters." A green button labeled "Sign up for GitHub" is positioned below the form. At the bottom of the form, a disclaimer reads: "By clicking 'Sign up for GitHub', you agree to our terms of service and privacy policy. We will send you account related emails occasionally." The top of the page features the GitHub logo, a search bar, and navigation links for "Explore", "Features", "Enterprise", and "Blog". A "Sign in" button is located in the top right corner.



Creare/clonare un repository

- Per clonare in **locale** un repository:
 - `git clone <url>`
- Esempio:
 - `git clone`
<https://github.com/SuperV1234/linuxDay2014>



Aggiungere file

- Creazione ed aggiunta di file in locale:
 - touch <file1>; touch <file2>
 - git add <file1> <file2> <...>
 - git status -s
- Esempio:
 - touch test.txt
 - git add test.txt
 - git status -s



Commit dei cambiamenti

- Creazione di un **commit**:

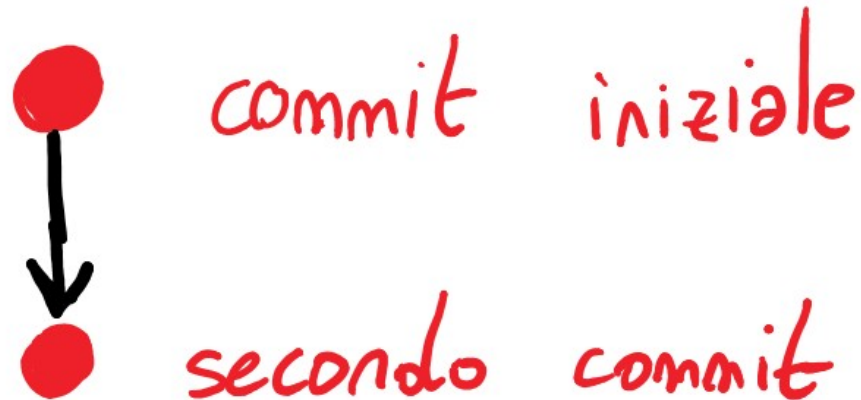
- `git commit <file1> <file2>`
`<...> -m <messaggio>`
- `git commit -a -m`
`<messaggio>`
- `git status`

- Esempio:

- `git commit test.txt -m`
`“primo commit”`
- `git status`



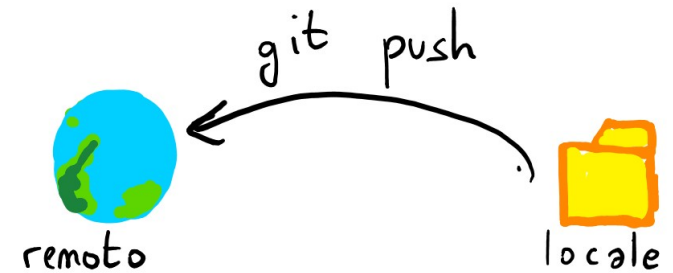
Struttura del repository



Push/pull (locale ↔ remoto)

- **Push** da locale a remoto:

- git push



- **Pull** da remoto a locale:

- git pull



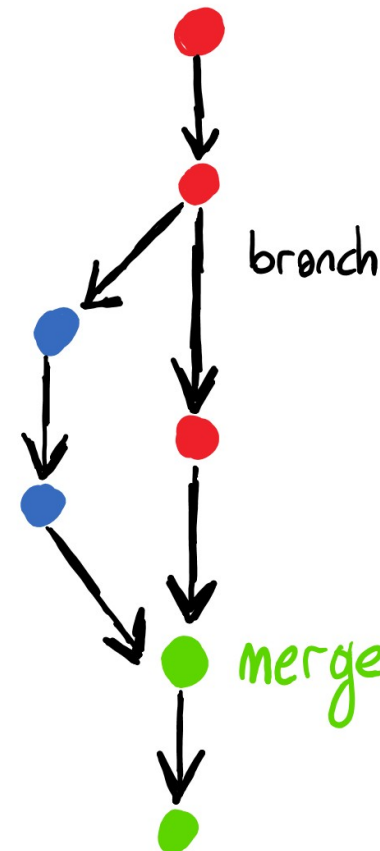
Commit precedenti

- Lista **commit** precedenti:
 - `git log`
 - `git log --pretty=oneline`
- **Checkout** di una versione precedente:
 - `git checkout <hash> .`
 - `git checkout <hash> -- <file>`
- **Differenze** con un commit precedente:
 - `git diff <hash> .`
 - `git diff <hash> -- <file>`



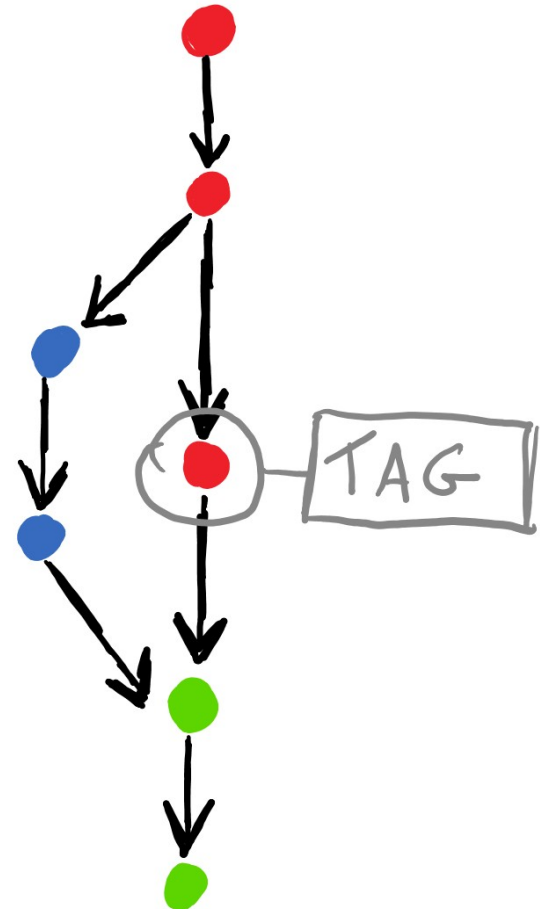
Branching

- **Visualizzare** branch attuale:
 - `git branch`
- **Creazione** di un **nuovo** branch:
 - `git checkout -b <branch>`
- **Rimozione** di un branch:
 - `git branch -d <branch>`
- **Push** di un branch:
 - `git push origin :<branch>`
- **Merging** di un branch:
 - `git checkout master`
 - `git merge`



Tagging

- **Creazione** di un tag:
 - `git tag -a <tag>`
- **Push** dei tag:
 - `git push -tags`
- **Lista** dei tag:
 - `git tag -l`
- **Checkout** di un tag:
 - `git checkout tags/<tag>`



GitHub: **features** aggiuntive

- **Stars e watchlist**
- **Fork e pull request**
- **Issues**
- **Gists**
- Grafici e visualizzazioni
- Bonus studenti





Grazie!



Vittorio Romeo

<http://vittorioromeo.info>
vittorio.romeo@outlook.com