

ML

Fundamentals



Instituto
Balseiro

Instituto Balseiro
16/08/2022

Luis G. Moyano - Fundamentos de ML
Instituto Balseiro



Special session



git



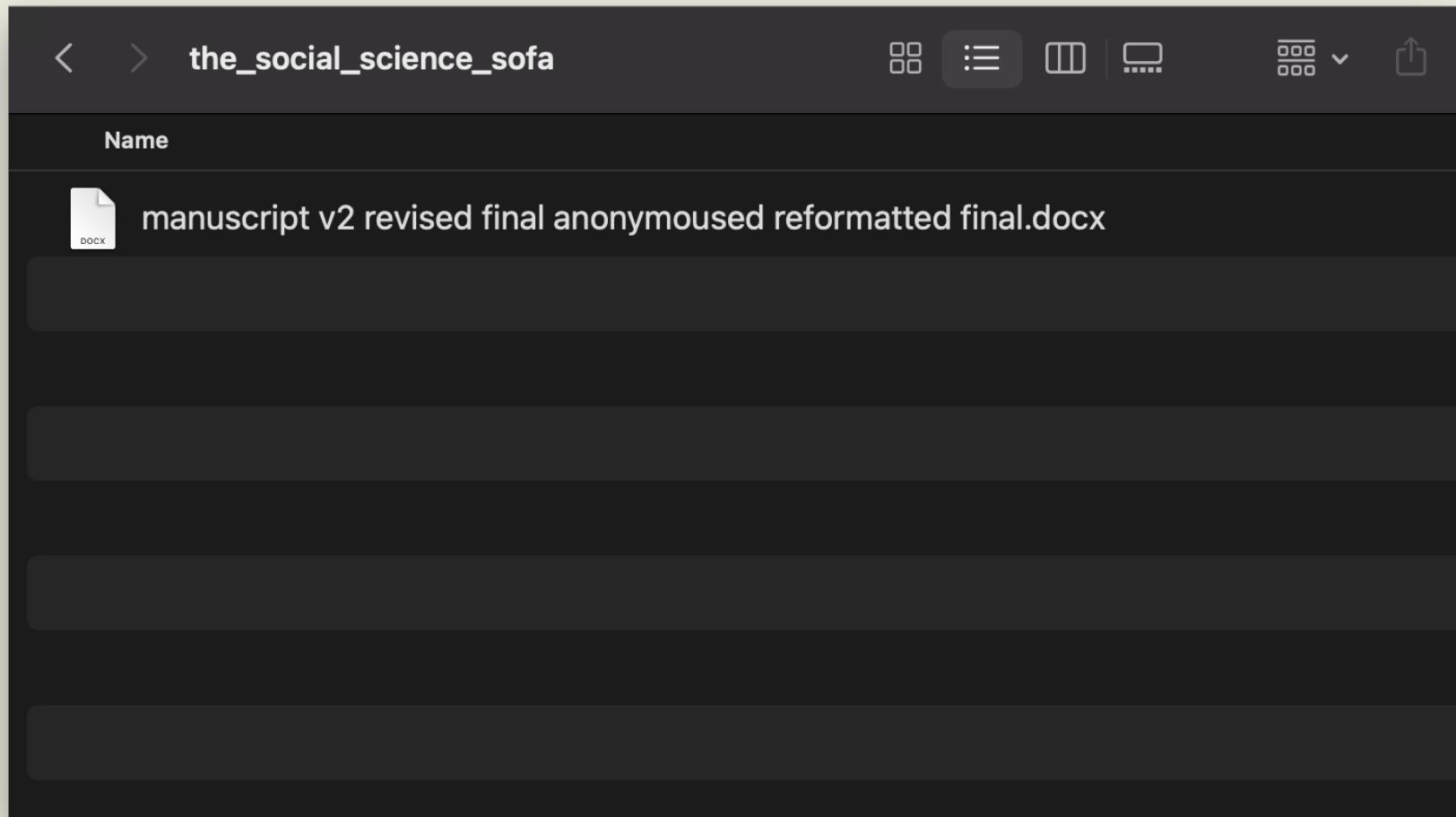
THIS IS GIT. IT TRACKS COLLABORATIVE WORK
ON PROJECTS THROUGH A BEAUTIFUL
DISTRIBUTED GRAPH THEORY TREE MODEL.

COOL. HOW DO WE USE IT?

NO IDEA. JUST MEMORIZIZE THESE SHELL
COMMANDS AND TYPE THEM TO SYNC UP.
IF YOU GET ERRORS, SAVE YOUR WORK
ELSEWHERE, DELETE THE PROJECT,
AND DOWNLOAD A FRESH COPY.



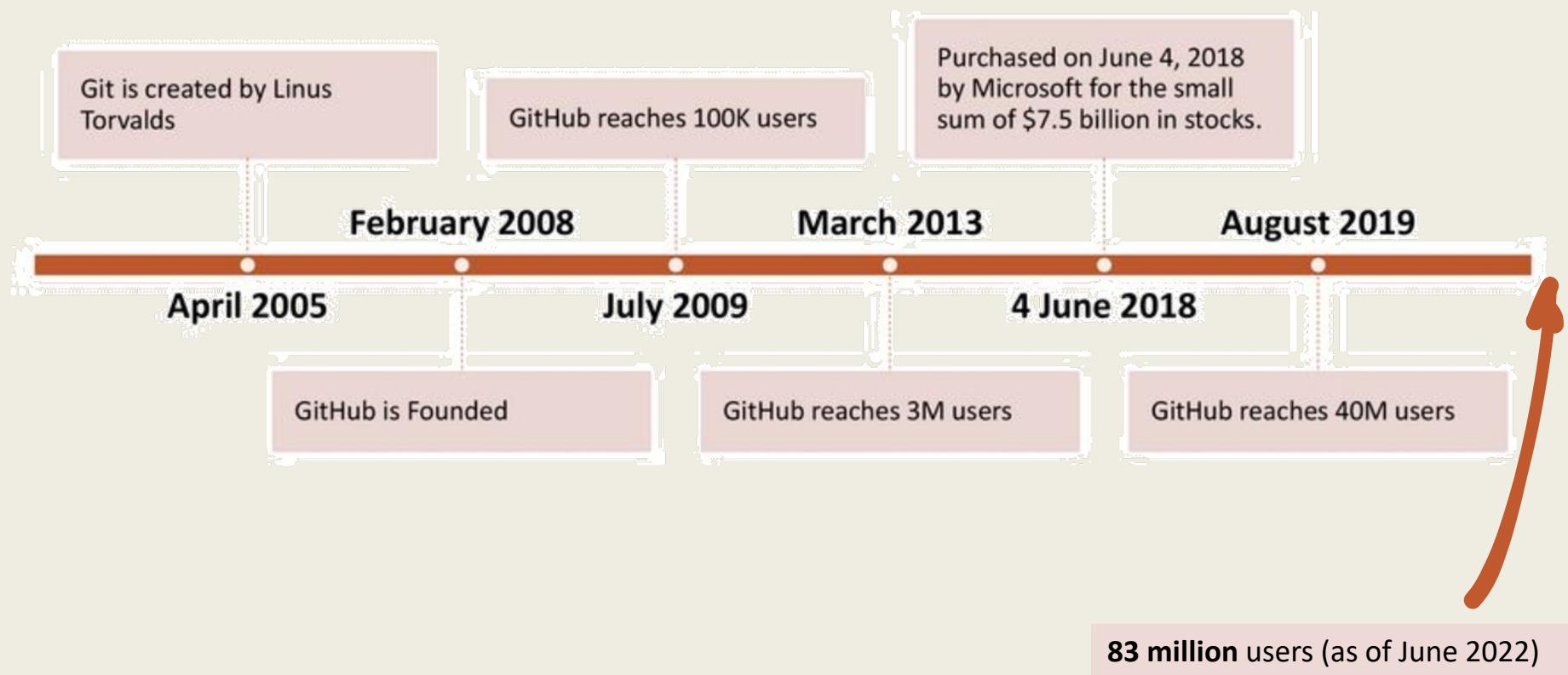
The filename nightmare



A penguin named Linus



Git & GitHub timeline



Why git, why GitHub?... why? why?!

What's GIT? It's 'the' distributed version control system

- Performance
- Security
 - Integrity focused, SHA1-based
- Flexibility
 - Supports nonlinear workflow
- Open-source ☺

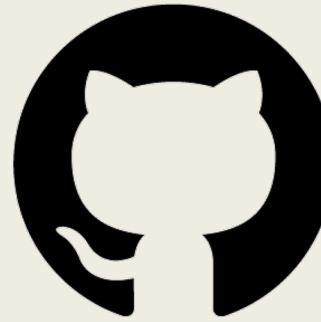
What's GitHub? It's 'the' git-based hosting service

- Popularity & Community
- Noob-friendly
- Not open source ☹ (but mostly free as in beer)
 - (GitLab is! :D)
- Many (if not most) ML projects are hosted here





git



GitHub

Why GitHub? Team Enterprise Explore Marketplace Pricing Search Sign in Sign up

amueller / introduction_to_ml_with_python Watch 335 Star 4.5k Fork 2.9k

Code Issues 16 Pull requests 0 Actions Projects 0 Security Insights

Join GitHub today

GitHub is home to over 50 million developers working together to host and review code, manage projects, and build software together.

Dismiss

Sign up

Notebooks and code for the book "Introduction to Machine Learning with Python"

90 commits 1 branch 0 packages 0 releases 11 contributors

Branch: master New pull request Find file Clone or download

amueller Switch branches or tags im jorijnsmit/master ... Latest commit 465dc4b on Mar 4

data	add adult dataset, fix path.	4 years ago
images	also add api table image	3 years ago
mlearn	don't use joblib from externals	8 months ago
.gitignore	add gitignore	4 years ago
01-introduction.ipynb	update notebooks for new print / sklearn 0.20 / new spacy etc	2 years ago
02-supervised-learning.ipynb	explicitly use liblinear solver for l1 penalty in logistic regression	8 months ago

Meet the Data

```
In [10]: from sklearn.datasets import load_iris
iris_dataset = load_iris()

In [11]: print("Keys of iris_dataset:\n", iris_dataset.keys())

In [12]: Keys of iris_dataset:
dict_keys(['data', 'target', 'target_names', 'DESCR', 'feature_names', 'filename'])

In [12]: print(iris_dataset['DESCR'][:193] + "\n...")

.. _iris_dataset:
-----
Iris plants dataset
-----
**Data Set Characteristics:**

:Number of Instances: 150 (50 in each of three classes)
:Number of Attributes: 4 numeric, pre
...

In [13]: print("Target names:", iris_dataset['target_names'])
Target names: ['setosa' 'versicolor' 'virginica']

In [14]: print("Feature names:\n", iris_dataset['feature_names'])

Feature names:
['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)', 'petal width (cm)']

In [15]: print("Type of data:", type(iris_dataset['data']))
Type of data: <class 'numpy.ndarray'>

In [16]: print("Shape of data:", iris_dataset['data'].shape)
Shape of data: (150, 4)

In [17]: print("First five rows of data:\n", iris_dataset['data'][:5])

First five rows of data:
[5.1 3.5 1.4 0.2]
[4.9 3. 1.4 0.2]
[4.7 3.2 1.3 0.2]
[4.6 3.1 1.5 0.2]
```

https://github.com/amueller/introduction_to_ml_with_python

<https://github.com/ageron/handson-ml2>

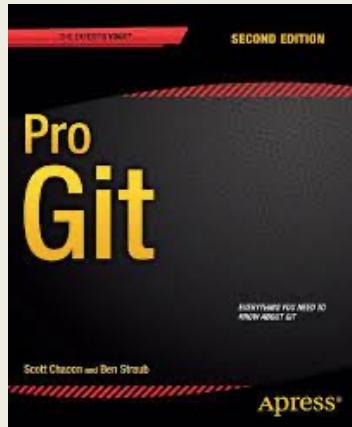


Git es un sistema de control de version distribuido



Reference Manual

The official and comprehensive **man pages** that are included in the Git package itself.



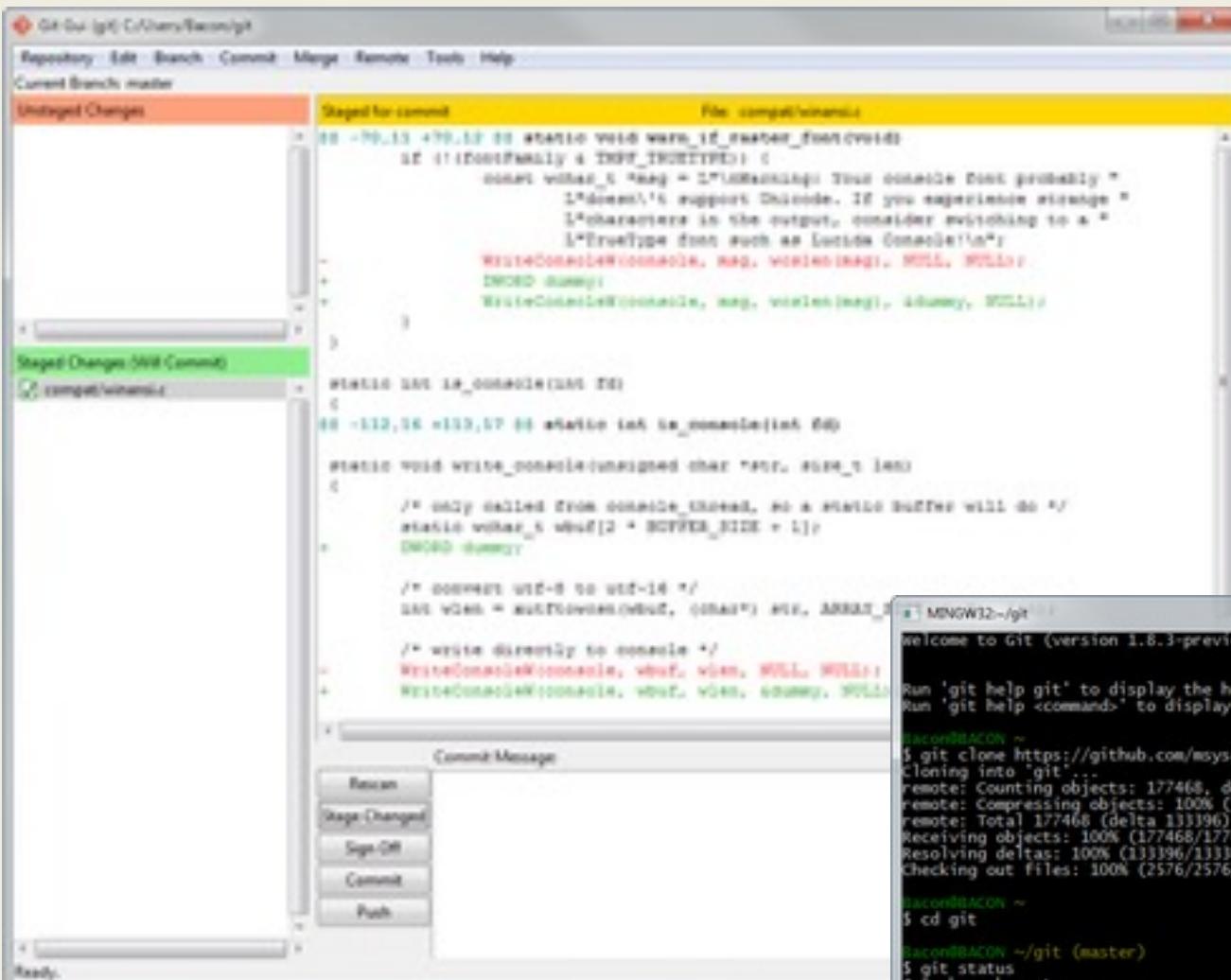
Pro Git

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

In course GitHub repo

Instalación en Windows

<https://gitforwindows.org/>



```
MINGW32--/git
welcome to Git (version 1.8.3-preview20130601)

Run 'git help git' to display the help index.
Run 'git help <command>' to display help for specific commands.

BaconBACON ~
$ git clone https://github.com/msysgit/git.git
Cloning into 'git'...
remote: Counting objects: 177468, done.
remote: Compressing objects: 100% (52057/52057), done.
remote: Total 177468 (delta 133396), reused 166093 (delta 123576).
Receiving objects: 100% (177468/177468), 42.16 MiB | 1.84 MiB/s, done.
Resolving deltas: 100% (133396/133396), done.
Checking out files: 100% (2576/2576), done.

BaconBACON ~
$ cd git

BaconBACON ~/git (master)
$ git status
# on branch master
nothing to commit, working directory clean

BaconBACON ~/git (master)
$
```

Rest: <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

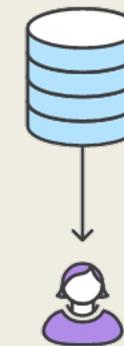
Some git scenarios

- **Local:** Single user, local.



- backup
- trial and error
- flexibility

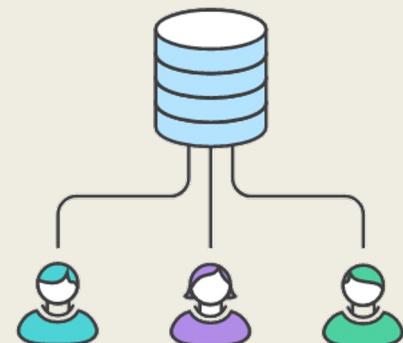
- **Remote:** Single user, local and remote.



- remote backup
- work with 3rd party code
- sharing purposes

- **Distributed:** More than user, each working local and also collaborating on remote.

- collaboration
- E.g. writing a paper



Commit!



Commit often! and in logical units!



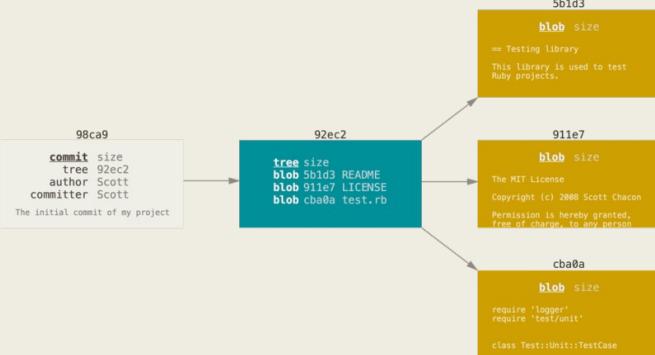
GitHub:

The screenshot shows the GitHub commit history for a repository. It is organized into three main sections corresponding to the dates of the commits:

- Commits on Aug 14, 2022:**
 - recover guia_01-git.github.tex - moyano committed 14 minutes ago
- Commits on Aug 13, 2022:**
 - finish merge - moyano committed 13 hours ago
 - commit aux files - moyano committed 13 hours ago
- Commits on Aug 12, 2022:**
 - nuevos problemas en guia 02 - OneZan committed 2 days ago
 - add paper.cls - moyano committed 2 days ago
 - update schedule - moyano committed 2 days ago
 - update first two practices - moyano committed 2 days ago

Shell:

```
(base) workmac:guias-profes moyano$ git log --pretty=oneline
80bb9ded1bb5054c619da3644f179126c3f87b90 (HEAD -> main, origin/main) recover guia_01-git.github.tex
ba84e19cf46c8a9d4f4f23d3dfe56bcf6d34f10f finish merge
6572c328ceebadb68c2af3691a183953f4f91b5d commit aux files
6baa57a0a955f755e38b351a1ef688a00a7ab33e nuevos problemas en guia 02
b8054a15ba29a022bf6d0a10eef92a62e7050560 add paper.cls
c287286c84b6e33d1a9c30c9bea0d04a93003751 update schedule
df3ad07f563f5ec01bfe2d0931a59b8b166b8764 update first two practices
28393621d484762d0e46087d91c2b59d3d1e7fcf add guias folder with practice template
```



Commit tree

- Snapshot of chosen files
- References
- Plus metadata
 - author, date, time
 - message
 - SHA-1 hashes
 - own
 - parent's
 - blob's
 - etc.

Commit!



Commit often! and in logical units



GitHub

Commits on Aug 14, 2022

- recover guia_01-git.github.tex - moyano committed 14 minutes ago

Commits on Aug 13, 2022

- finish merge - moyano committed 13 hours ago
- commit aux files - moyano committed 13 hours ago

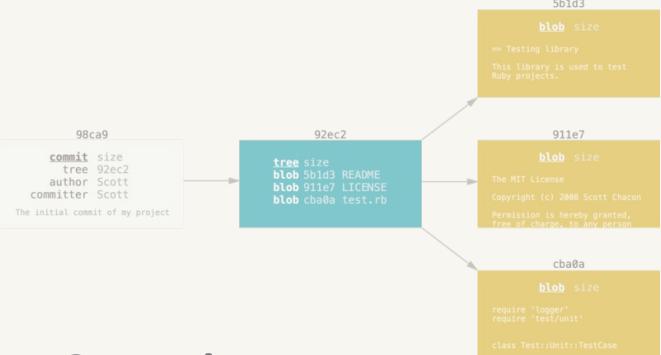
Commits on Aug 12, 2022

- nuevos problemas en guia 02 - OneZan committed 2 days ago
- add paper.cls - moyano committed 2 days ago
- update schedule - moyano committed 2 days ago
- update first two practices - moyano committed 2 days ago

Shell:

```
(base) work  
80bb9ded1bb  
ba84e19cf46  
6572c328ce6  
6baa57a0a95
```

```
b8054a15ba29a022bf6d0a10eeff92a62e7050560 add paper.cls  
c287286c84b6e33d1a9c30c9bea0d04a93003751 update schedule  
df3ad07f563f5ec01bfe2d0931a59b8b166b8764 update first two practices  
28393621d484762d0e46087d91c2b59d3d1e7fcf add guias folder with practice template
```

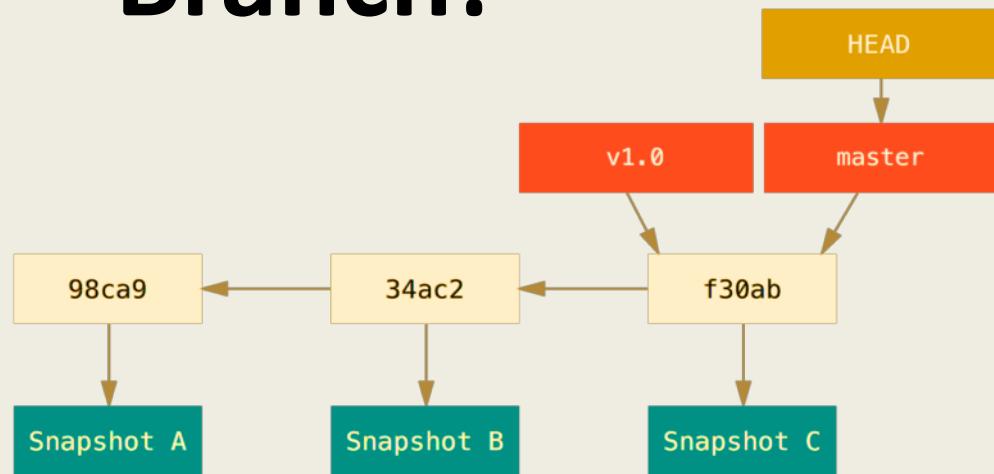


Commit tree

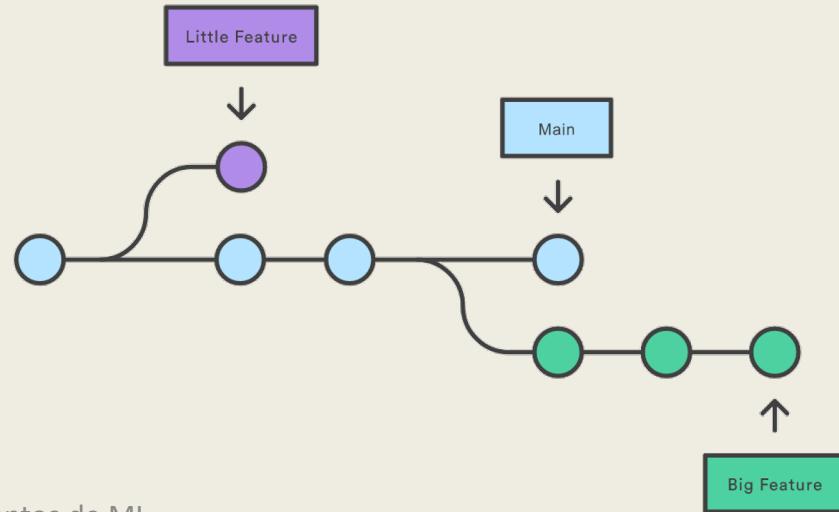
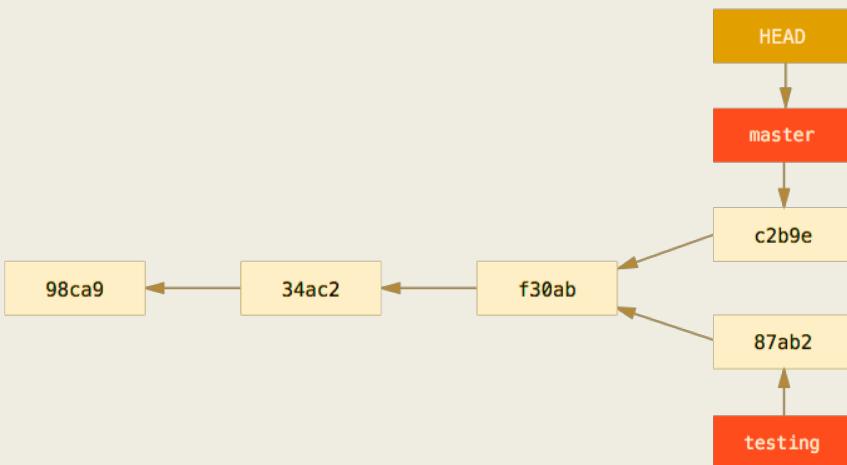
- Snapshot of chosen files



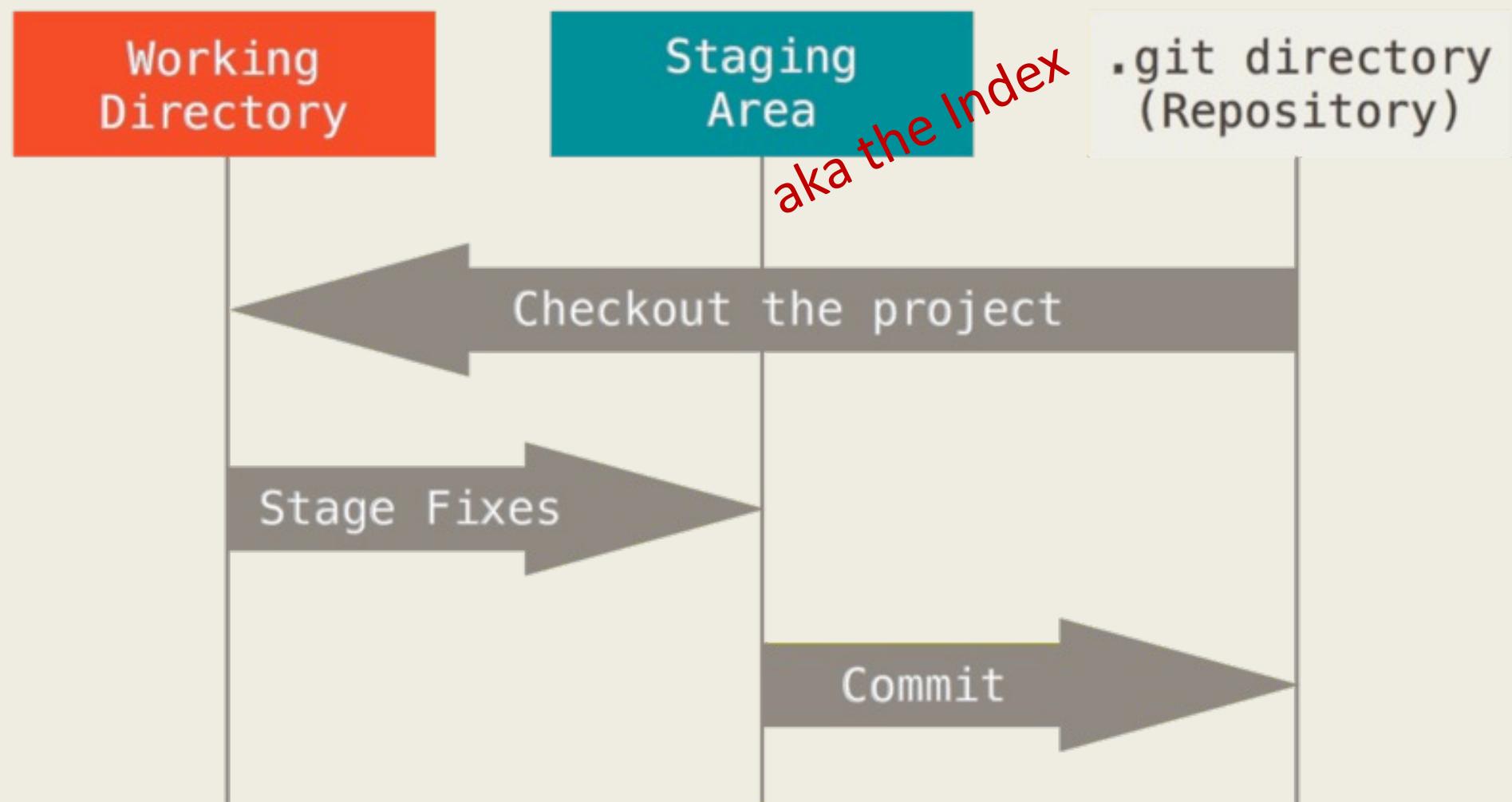
Branch!



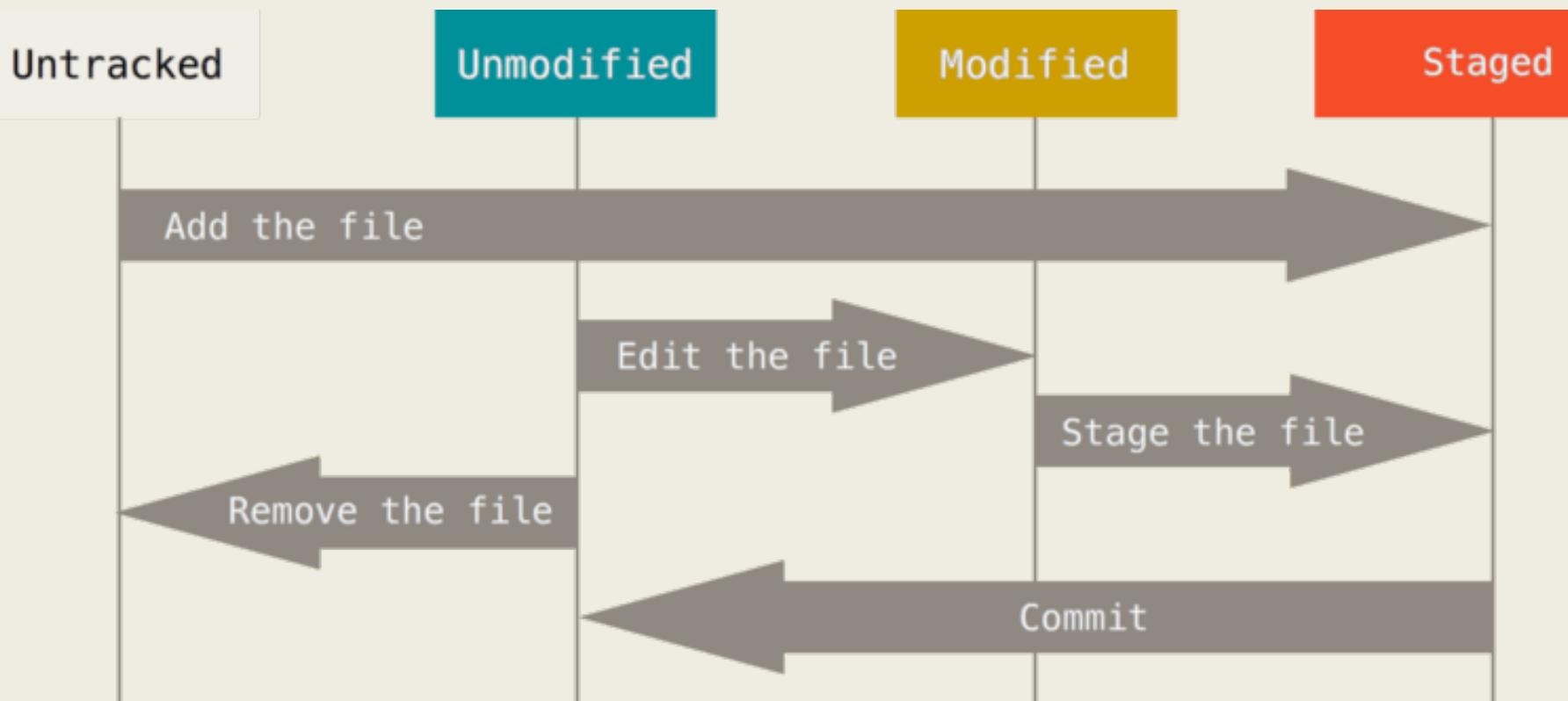
- Just a reference (think of it as a little flag) to a selected commit (& parents)
- Used to "branch" a commit 'time line'
- Gets carried up when committing
- As many as we want (really cheap!)
- HEAD is a special reference to the checkout commit



Local git workflow



File life cycle



Config

Configura la información del usuario para todos los repositorios locales

```
$ git config --global user.name "[name]"
```

Establece el nombre que estará asociado a tus commits

```
$ git config --global user.email "[email address]"
```

Establece el e-mail que estará asociado a sus commits

Ignore

- To ensure that certain files (not already tracked by Git) remain untracked.
- There is a practical side, and a security side.
- How? put *globs* in hidden `.gitignore` file in project folder*
 - globs: **shell pattern matching**, as in `.*` or `*.txt`

```
# ignore all .a files
*.a

# but do track lib.a, even though you're ignoring .a files above
!lib.a

# only ignore the TODO file in the current directory, not subdir/TODO
/TODO

# ignore all files in any directory named build
build/

# ignore doc/notes.txt, but not doc/server/arch.txt
doc/*.txt

# ignore all .pdf files in the doc/ directory and any of its subdirectories
doc/**/*.pdf
```

For more see: [Ignoring files](#) and [in the official git docs](#) and [man gitignore](#)

Luis G. Moyano - Fundamentos de ML

Instituto Balseiro 2022 2020

* It's possible to have multiple `.gitignore` files

Create repositories

Inicializa un nuevo repositorio u obtiene uno de una URL existente

```
$ git init [project-name]
```

Crea un nuevo repositorio local con el nombre especificado

```
$ git clone [url]
```

Descarga un proyecto y toda su historial de versiones

In most cases, the cloned remote repo remains with the default name 'origin'

Make changes

Revisa cambios y crea un commit

```
$ git status
```

Enumera todos los archivos nuevos o modificados de los cuales se van a guardar cambios

```
$ git diff
```

Muestra las diferencias entre archivos que no se han enviado aún al área de espera

```
$ git add [file]
```

Guarda el estado del archivo en preparación para realizar un commit

```
$ git commit -m "[descriptive message]"
```

Registra los cambios del archivo permanentemente en el historial de versiones

diffs

```
Changes from main to working tree
1 file changed, 1 insertion(+), 1 deletion(-)
guias-profes/guia_01-git_github.aux | 2 +-  
  
modified   guias-profes/guia_01-git_github.aux  
@@ -5,7 +5,7 @@  
  \nameuse{es@quoting}  
  \babel@aux{spanish}{}  
  \writefile{toc}{\contentsline {section}{\numberline {1}Questions}{1}}  
- \writefile{toc}{\contentsline {section}{\numberline {2}Problems}{1}}  
+ \writefile{toc}{\contentsline {section}{\numberline {2}Exercises}{1}}  
  \writefile{toc}{\contentsline {subsection}{\numberline {2.1}Local}{1}}  
  \writefile{toc}{\contentsline {subsection}{\numberline {2.2}Remote}{2}}  
  \writefile{toc}{\contentsline {subsection}{\numberline {2.3}Distributed}{2}}
```



Undoing things - locally

- *Change* last commit: `--amend`

Only amend commits that are still local and have not been pushed somewhere.

```
$ git commit -m 'Initial commit'  
$ git add forgotten_file  
$ git commit --amend
```

- *Unstage* with `reset` (or `restore`)

```
$ git add *  
$ git status  
On branch master  
Changes to be committed:  
(use "git reset HEAD <file>..." to unstage)
```

```
renamed: README.md -> README  
modified: CONTRIBUTING.md
```

```
$ git reset HEAD CONTRIBUTING.md  
Unstaged changes after reset:  
M      CONTRIBUTING.md
```

```
$ git restore --staged CONTRIBUTING.md
```

- *Unmodifying* a modified file with `checkout` (or `restore`)

Reverts to last committed form

Luis G. Moyano - Fundamentos de ML -
Instituto Balseiro 2022

```
$ git checkout -- CONTRIBUTING.md
```

```
$ git restore CONTRIBUTING.md
```



Branch and merge

Nombra una serie de commits y combina esfuerzos ya completados

```
$ git branch
```

Enumera todas las ramas en el repositorio actual

```
$ git branch [branch-name]
```

Crea una nueva rama

```
$ git checkout [branch-name]
```

Cambia a la rama especificada y actualiza el directorio activo

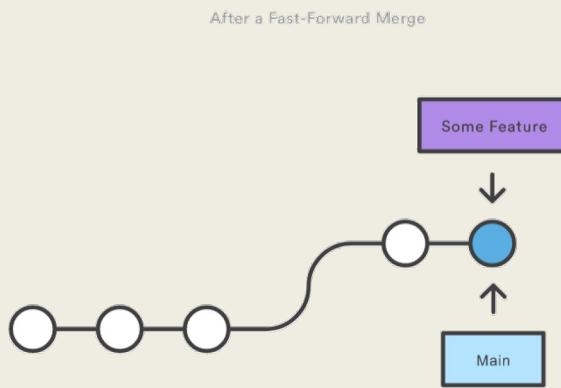
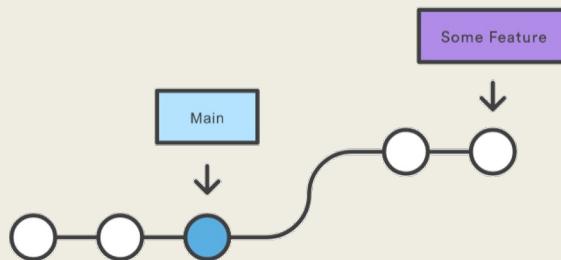
```
$ git merge [branch-name]
```

Combina el historial de la rama especificada con la rama actual

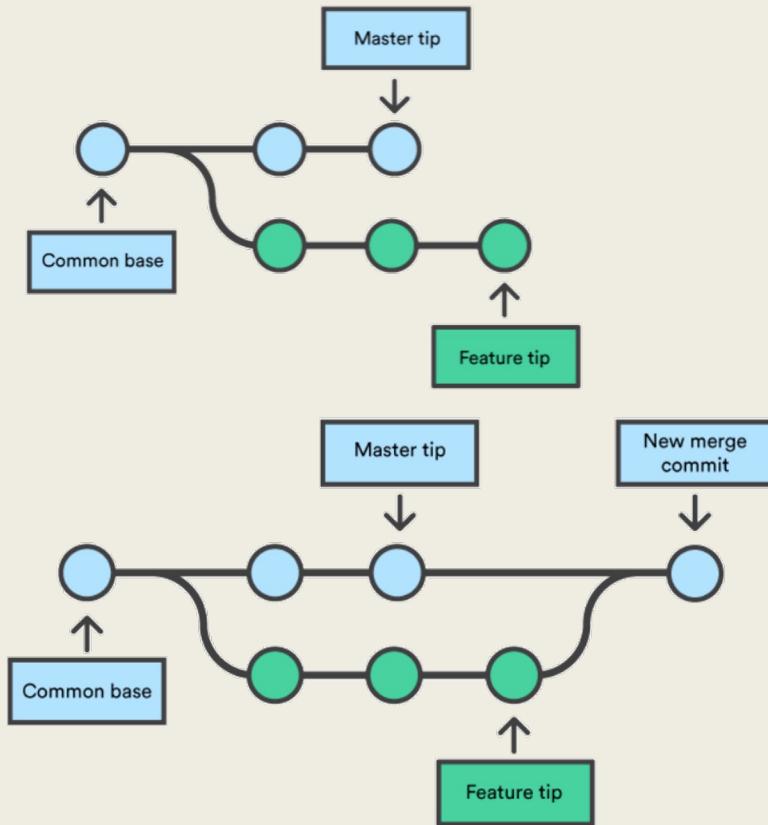
```
$ git branch -d [branch-name]
```

Borra la rama especificada

Merge strategies



Fast-forward merge



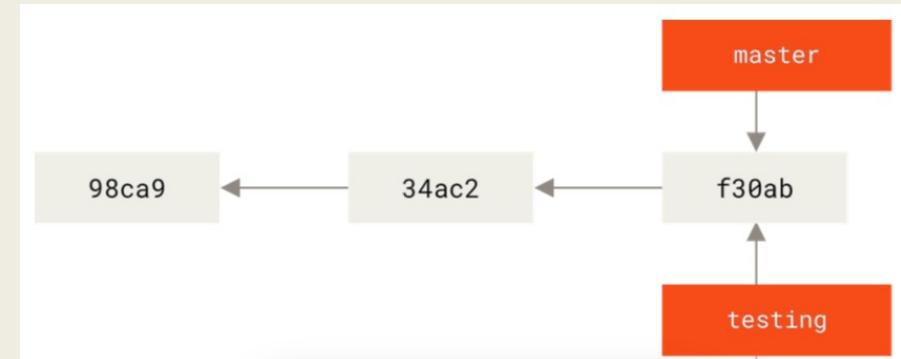
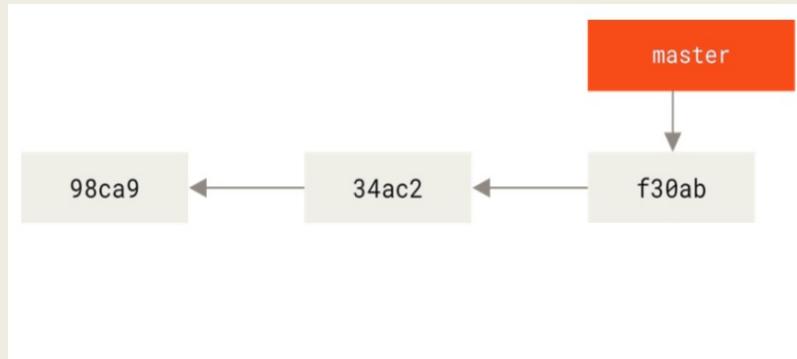
Recursive (3-way merge)

There are more types of merge strategies :

- Ours
- Octopus
- Resolve
- Subtree
- ...

Basic branching: HEAD

HEAD is an alias for the tip of the currently checked out branch



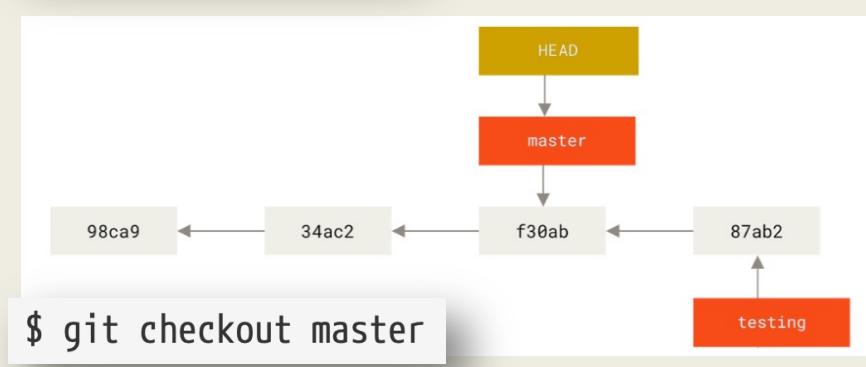
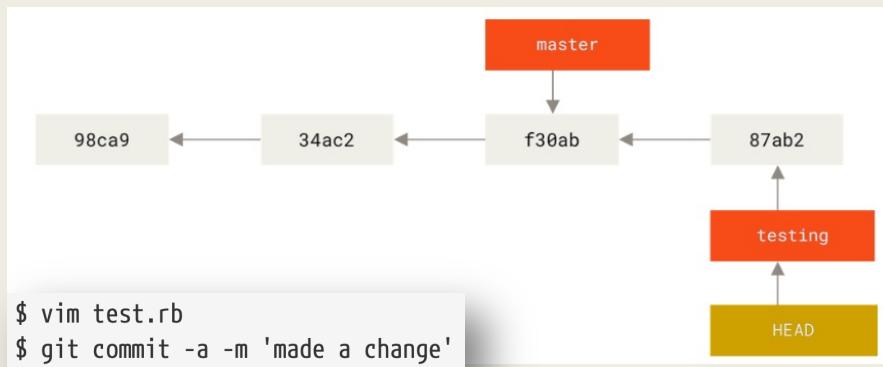
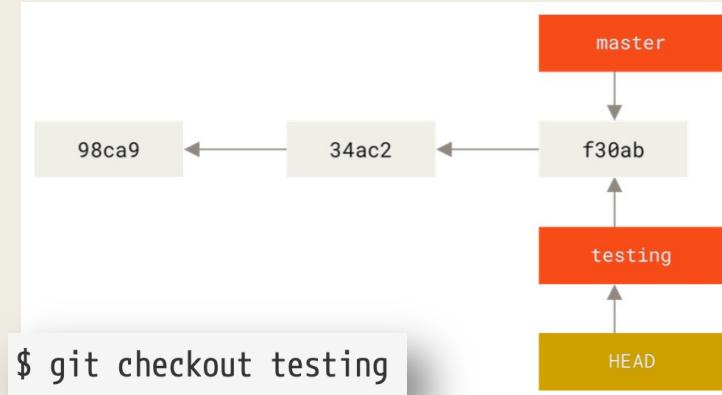
```
$ git branch testing
```



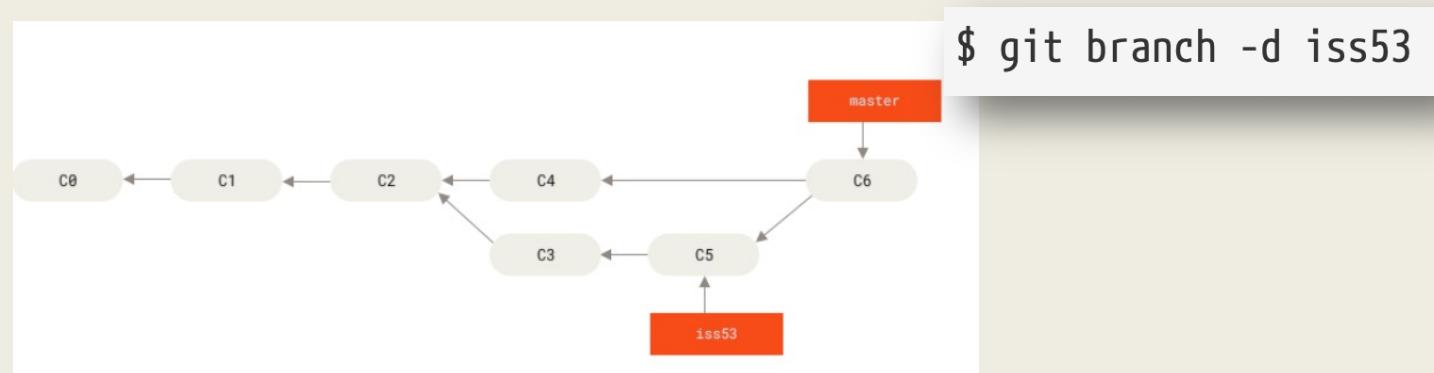
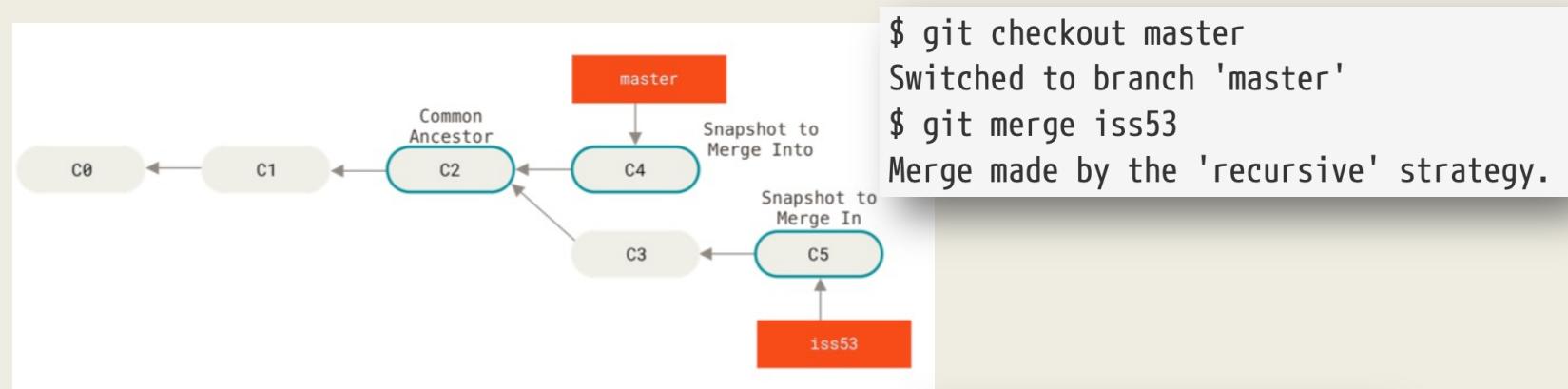
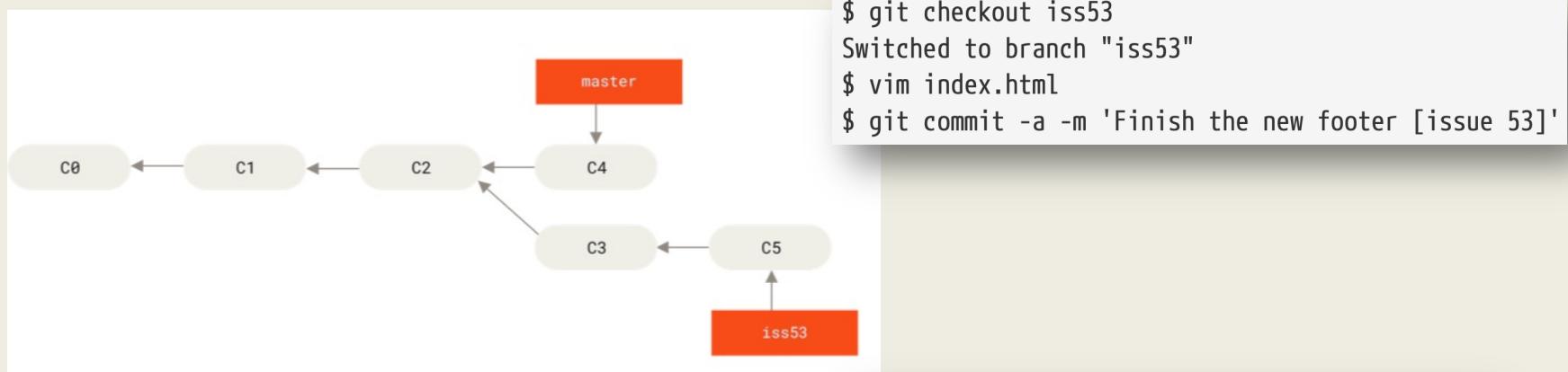
```
$ git log --oneline --decorate  
f30ab (HEAD -> master, testing) Add feature #32
```

```
$ git checkout testing
```

Basic branching: diverging branches



Basic branching: 3-way merge



Basic branching: conflict

```
$ git merge iss53
Auto-merging index.html
CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit the result.
```

```
$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")

Unmerged paths:
  (use "git add <file>..." to mark resolution)

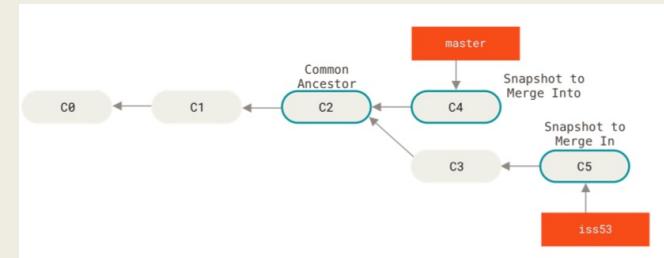
    both modified:   index.html
```

```
no changes added to commit (use "git add" and/or "git commit -a")
```

```
$ git status
On branch master
All conflicts fixed but you are still merging.
  (use "git commit" to conclude merge)
```

Changes to be committed:

```
modified:   index.html
```



```
<<<<< HEAD:index.html
<div id="footer">contact : email.support@github.com</div>
=====
<div id="footer">
  please contact us at support@github.com
</div>
>>>>> iss53:index.html
<div id="footer">
  please contact us at email.support@github.com
</div>
```

GitHub – main page

Screenshot of a GitHub repository page for [lgmoyano/redescomplejasyfisicasocial](#).

The page includes the following sections:

- Header:** Search bar, Pull requests, Issues, Marketplace, Explore, Notifications, and Profile icon.
- Repository Information:** Public repository, Forked from [luisgmo/redescomplejasyfisicasocial](#), 2 watchers, 0 forks, 0 stars.
- Navigation:** Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, Settings.
- Code Overview:** master branch, 1 branch, 0 tags, Go to file, Add file, Code tab.
- Commits:** A list of 27 commits by user [lgmoyano](#).

Commit	Message	Date
Set theme jekyll-theme-cayman	488e047 on Mar 30, 2017	27 commits
REPASO-DE-C	add igraph again	6 years ago
community-detection-examples	added comments	6 years ago
figs	fig punishment	6 years ago
first-examples	changed dir name and added some BA files	6 years ago
.gitignore	add fig	6 years ago
README.md	Update README.md	6 years ago
_config.yml	Set theme jekyll-theme-cayman	6 years ago
- About:** Prácticas para clase de Redes Complejas y Física Social.
- Readme:** README.md
- Releases:** No releases published. Create a new release.
- Packages:** No packages published. Publish your first package.

GitHub – main page

Screenshot of a GitHub repository page for [lgmoyano/redescomplejasyfisicasocial](#).

The top navigation bar includes: Search or jump to..., Pull requests, Issues, Marketplace, Explore, and a user icon.

The repository details show it is Public, created by [lgmoyano](#), and has 2 watchers, 0 forks, and 0 stars.

The main content area shows the repository structure and commit history:

- Branch: master (selected)
- Tags: 1 branch, 0 tags

Commit	Message	Date
Igmoyano Set theme jekyll-theme-cayman		✓ 488e047 on Mar 30, 2017
REPASO-DE-C	add igraph again	6 years ago
community-detection-examples	added comments	6 years ago
figs	fig punishment	6 years ago
first-examples	changed dir name and added some BA files	6 years ago
.gitignore	add fig	6 years ago
README.md	Update README.md	6 years ago
_config.yml	Set theme jekyll-theme-cayman	6 years ago

On the right side, there are sections for About, Releases, and Packages.

About: Prácticas para clase de Redes Complejas y Física Social

Releases: No releases published. Create a new release

Packages: No packages published. Publish your first package

GitHub – main page: Code

The screenshot shows a GitHub repository page for [lgmoyano / redescomplejasyfisicasocial](#). The page has a dark theme. At the top, there's a navigation bar with links for Pull requests, Issues, Marketplace, and Explore. On the right, there are icons for notifications, creating a new repository, and user profile.

The main content area shows the repository details: it's public, has 1 branch, and 0 tags. The commit history lists several changes made by the user [lgmoyano](#), including setting the theme to `jekyll-theme-cayman` and adding `igraph`.

A red box highlights the "Code" dropdown menu, which contains options for cloning the repository via HTTPS, SSH, or GitHub CLI. It also includes links to open the repository with GitHub Desktop or download it as a ZIP file.

On the right side, there are sections for "About", "Releases", and "Packages". The "About" section describes the repository as "Prácticas para clase de Redes Complejas y Física Social". The "Releases" section indicates no releases have been published, and the "Packages" section indicates no packages have been published.

GitHub - Commits

The screenshot shows a GitHub repository page for `lgmoyano / redescomplejasyfisicasocial`. The repository is public. The main navigation bar includes links for Pull requests, Issues, Marketplace, and Explore. On the right side, there are buttons for Pin, Unwatch (with 2 notifications), Fork (with 0 forks), and Star (with 0 stars). Below the navigation, there are tabs for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The Code tab is selected. A dropdown menu shows the current branch is master. The commit history for March 30, 2017, is displayed:

- Set theme jekyll-theme-cayman by Igmoyano committed on Mar 30, 2017 ✓ 488e047
- Set theme jekyll-theme-tactile by Igmoyano committed on Mar 30, 2017 ✓ 828a3c1
- Set theme jekyll-theme-minimal by Igmoyano committed on Mar 30, 2017 ✓ c2005cb
- test url by Igmoyano committed on Mar 30, 2017 ✓ 5c2d67e
- Update README.md by Igmoyano committed on Mar 30, 2017 ✓ c7d222c
- Update readme by Igmoyano committed on Mar 30, 2017 ✓ aed65f3
- Set theme jekyll-theme-slate by Igmoyano committed on Mar 30, 2017 ✓ 6d22afc



GitHub
Copilot

A screenshot of a dark-themed code editor interface. On the left is a vertical toolbar with icons for file operations, search, and other tools. The main workspace shows a single Python file named "main.py". The code in the file is:

```
1
2 | print("Hello, world!")
3
```

The number "2" is highlighted in red, indicating it's the current line of interest. Below the editor is a navigation bar with tabs: PROBLEMS, OUTPUT, TERMINAL, and DEBUG CONSOLE. The TERMINAL tab is active, showing the command-line prompt "rahulbanerjee@Rahuls-MBP co-pilot %". At the bottom, there are status indicators for Python version (Python 3.9.4 64-bit ('3.9')), code analysis (0△0), and various live preview and linting tools like Go Live, Prettier, and R. The status bar also shows the current cursor position (Ln 2, Col 1).

Config SSH keys to use GitHub

- GitHub improved security by dropping older, insecure key types on March 15, 2022.
- To access via SSH, one needs to authenticate with private SSH keys (access credentials for accessing the SSH protocol).
- **Steps:**
 - Generate a new private SSH key. Follow prompts.

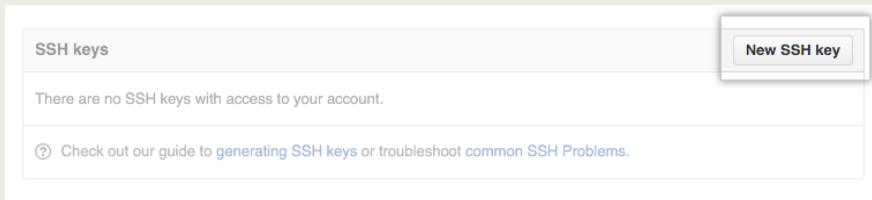
```
ssh-keygen -t rsa -b 4096 -C "your_email@example.com"
```

- Add it to the SSH agent.

```
$ eval "$(ssh-agent -s)"  
> Agent pid 59566
```

```
ssh-add -K /Users/you/.ssh/id_rsa
```

- Add the public SSH key to your account through GitHub's settings



Synchronize changes from 'origin'

Registrar un marcador para un repositorio e intercambiar historial de versiones

```
$ git fetch [bookmark]
```

Descarga todo el historial del marcador del repositorio

```
$ git merge [bookmark]/[branch]
```

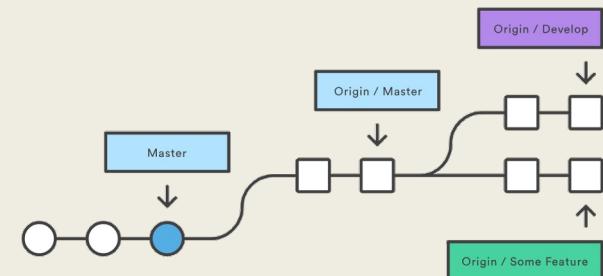
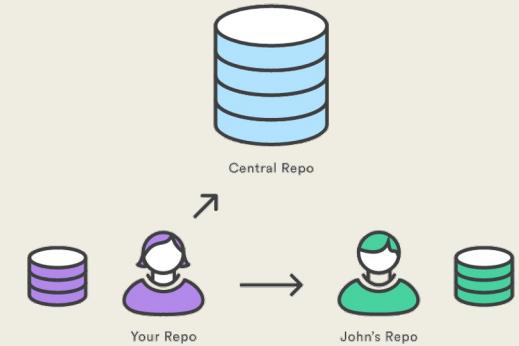
Combina la rama del marcador con la rama local actual

```
$ git push [alias] [branch]
```

Sube todos los commits de la rama local a GitHub

```
$ git pull
```

Descarga el historial del marcador e incorpora cambios



Basic remote workflow

Assuming one's got everithing runing smoothly

`git add <file1> <file2>`

`git commit -m 'my message'`

`git status`

(to check relative position wrt 'origin')

`git pull`

(to absorb changes if needed)

`git push`

(to 'upload' local changes)



Workflow Overview



The single thing **NOT** to do: **DO NOT rewrite remote history**

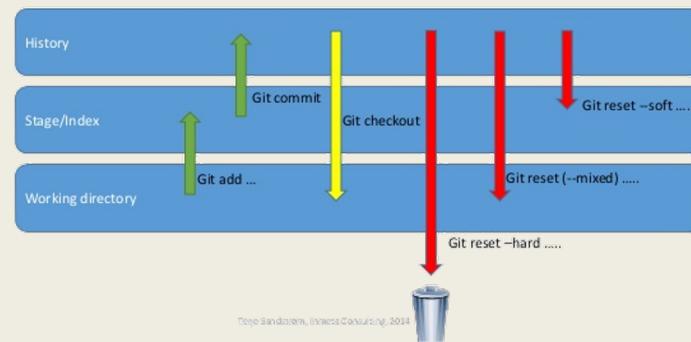
- Sometimes, you may need to change history. You may need to undo a commit.
- But don't rewrite ***remote*** history if working with collaborators. You could create big problems for them.
 - Don't do `git reset --hard` if changes already went to repo. Use `git revert` instead, the safest way to undo things.
 - Beware of `push --force`, you may delete other people's work
 - Don't *ammend* a commit in the remote casually



Many more useful commands to explore

- *"At some point in your Git journey, you may accidentally lose a commit."*
 - git reflog
 - git fsck --lost-found ([10.7 Git Internals - Maintenance and Data Recovery](#))
- reset –hard, --mixed, --soft
- stash
- tags and logs
- rebase, squash,
- cherrypick

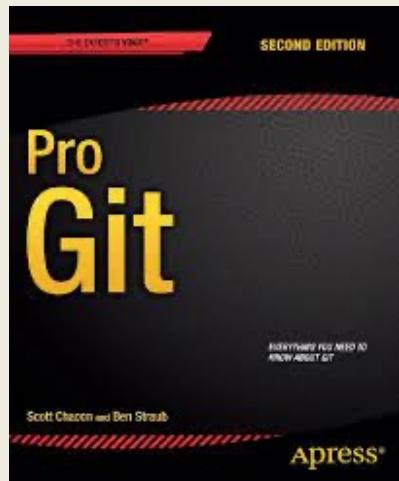
Git tree movements visualized



<https://davidzych.com/difference-between-git-reset-soft-mixed-and-hard/>

Git guide + Help

- Get hands dirty!
- Some questions
- Some problems, organized by scenarios
 - Third part in pairs
- Use Git Pro @biblio, as well as SO & the web



<https://git-scm.com/>



Summary: see [git - the simple guide](#)

<https://rogerdudler.github.io/git-guide/>

