

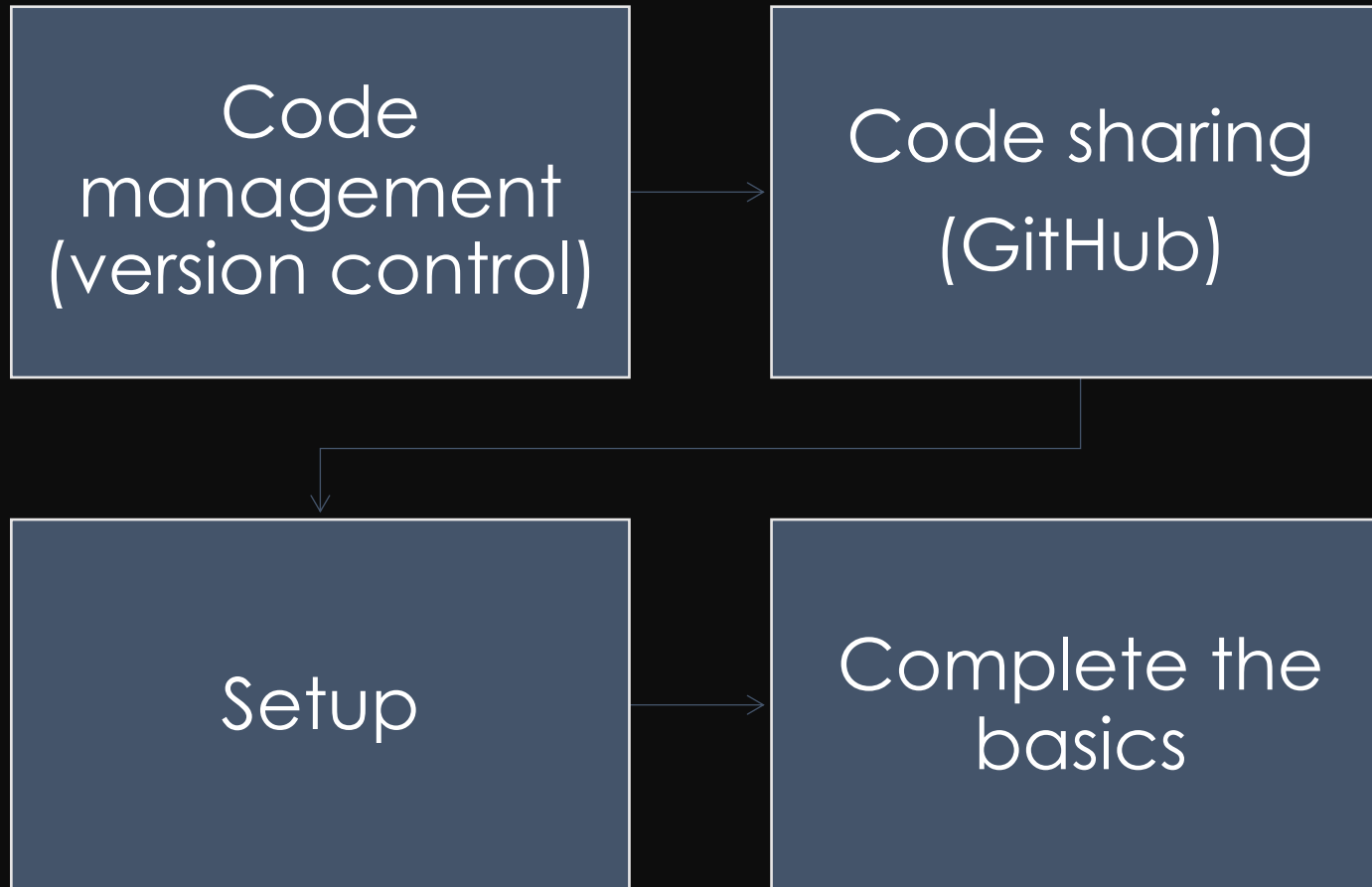
# Git and GitHub

Raul Rodriguez Cruces



# Objectives

---

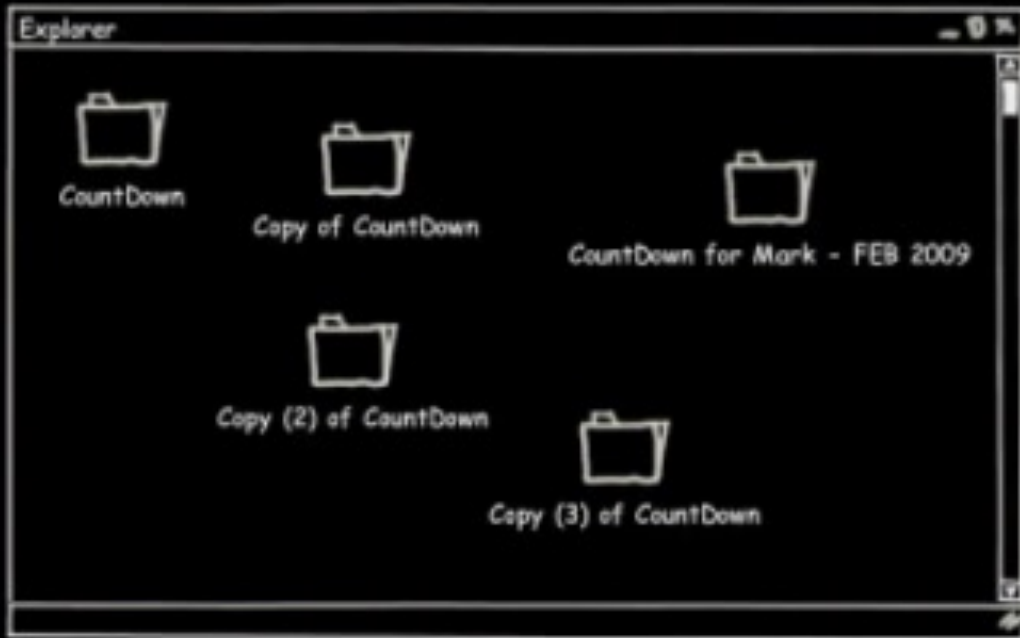


```
mirror_mod = modifier_ob.  
set mirror object to mirror.  
mirror_mod.mirror_object =  
operation == "MIRROR_X":  
mirror_mod.use_x = True  
mirror_mod.use_y = False  
mirror_mod.use_z = False  
operation == "MIRROR_Y":  
mirror_mod.use_x = False  
mirror_mod.use_y = True  
mirror_mod.use_z = False  
operation == "MIRROR_Z":  
mirror_mod.use_x = False  
mirror_mod.use_y = False  
mirror_mod.use_z = True  
  
#selection at the end -add  
mirror_ob.select= 1  
modifier_ob.select=1  
context.scene.objects.active  
("Selected" + str(modifier_ob.  
mirror_ob.select = 0  
= bpy.context.selected_object  
data.objects[one.name].select  
  
print("please select exactly  
  
-- OPERATOR CLASSES -----  
  
types.Operator):  
on X mirror to the selected  
object.mirror_mirror_x"  
mirror X"  
  
context):  
context.active_object is not
```

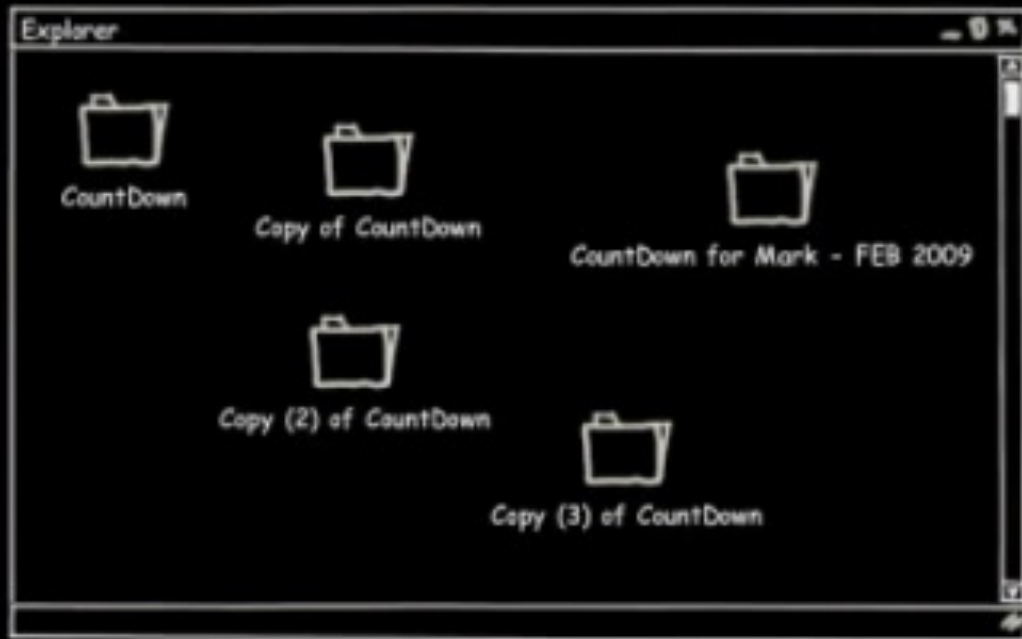
# Code management

# What is versioning?

- Keep track of the changes of your code/files
- Documenting those changes



# Is there an easier way?



- Keep track of your changes
- Troubleshooting
- Backup new features
- Simplify code sharing

# Is there an easier way?

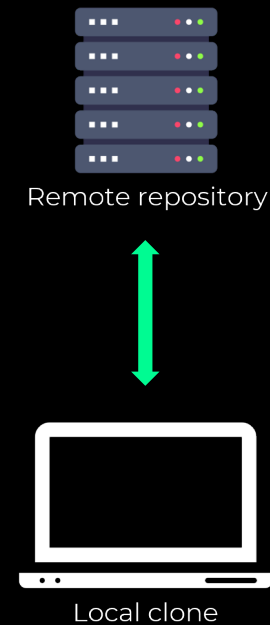


- Keep track of your changes (version control)
- Troubleshooting
- Backup new features
- Simplify code sharing



# What is git?

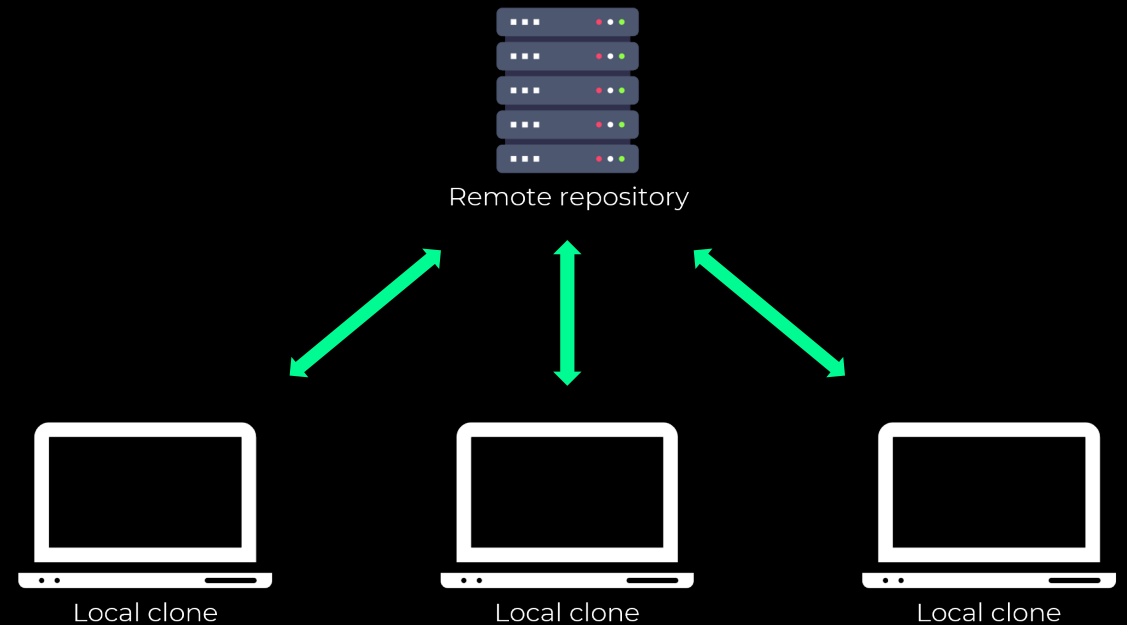
- An open source, distributed **version control system**
- Created by Linus Torvalds (*as same as Linux*)
- Who made what changes and when
- Revert back at any time
- Local and remote repositories
- Coordinates works between multiple developers





# What is git?

- An open source, distributed **version control system**
- Created by Linus Torvalds (*as same as Linux*)
- Who made what changes and when
- Revert back at any time
- Local and remote repositories
- Coordinates works between multiple developers

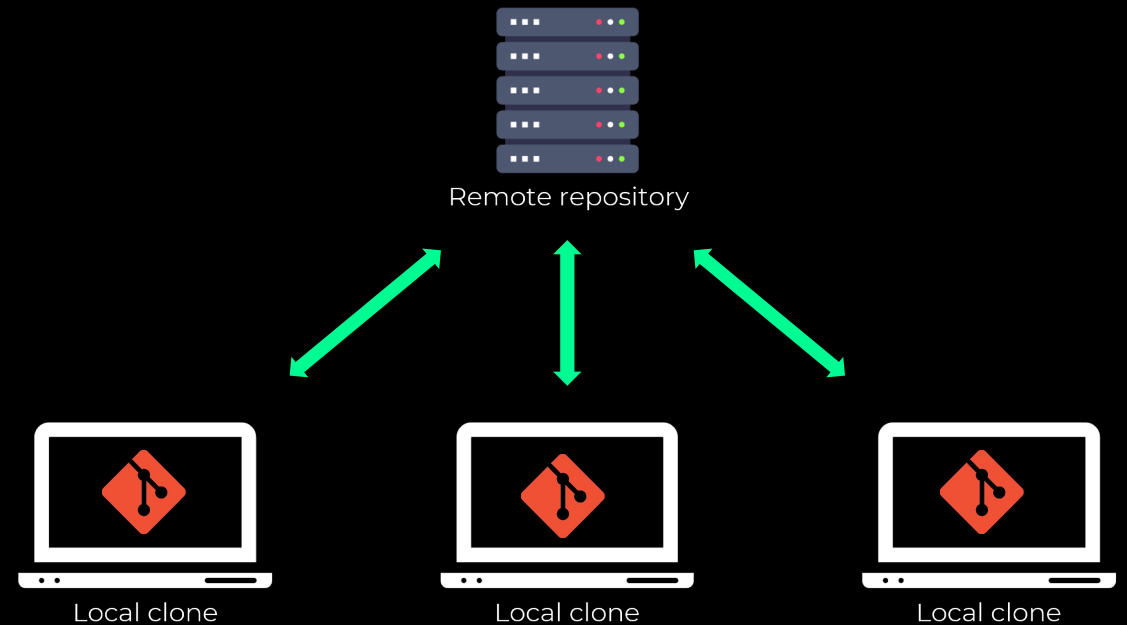






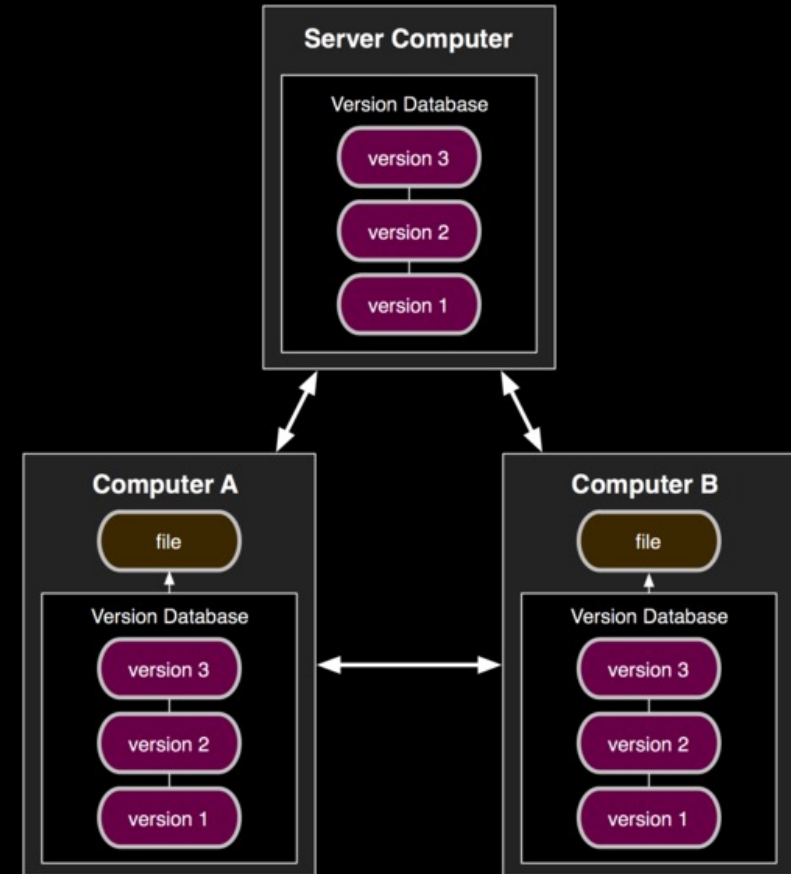
# What is git?

- An open source, distributed **version control system**
- Created by Linus Torvalds (*as same as Linux*)
- Who made what changes and when
- Revert back at any time
- Local and remote repositories
- Coordinates works between multiple developers



# Distributed version control

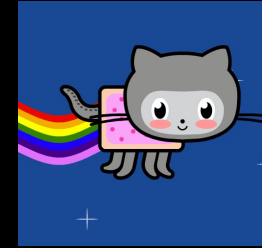
- Retains file history
- Your local repo is a complete copy of everything on the remote server
- Most of the changes will be local
- When you are done, you **push** the changes back to the server
- Other users can **pull** your changes to their local computer



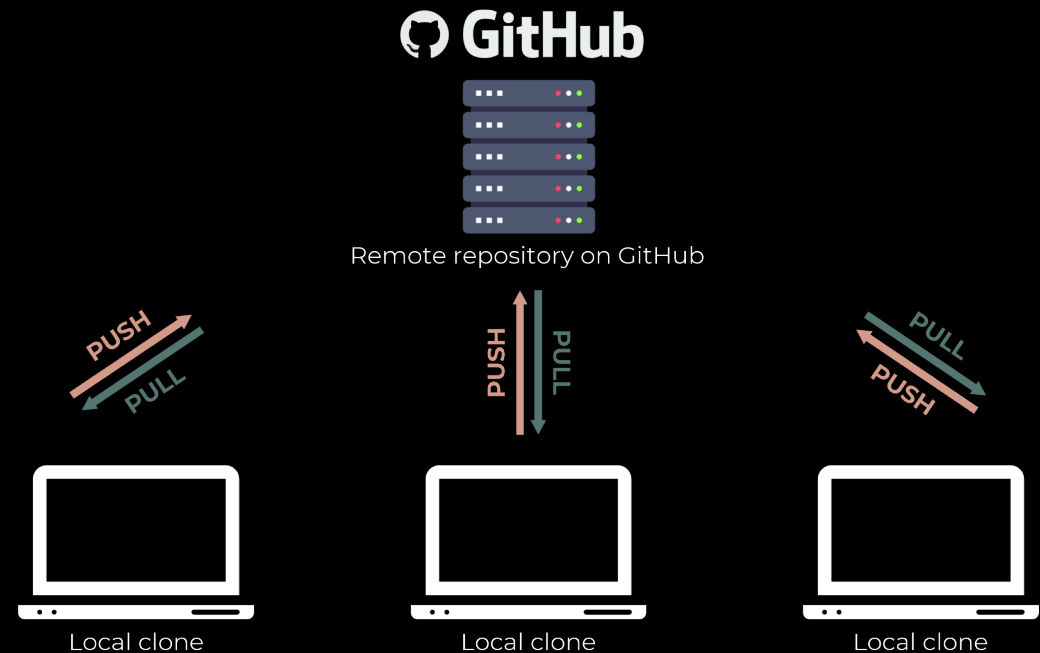


Code sharing

# What is GitHub?



- Core hosting platform for version control collaboration on Git repositories
- Allows to host a central repository in a remote server



# How do we use these tools?

- Command line interface (CLI)
- Graphical user interface (GUI)



GitHub Desktop



# Key concepts

## Workspace

```
$ git init  
$ git clone
```



Local clone

Local directory

## Staging area

```
$ git add
```

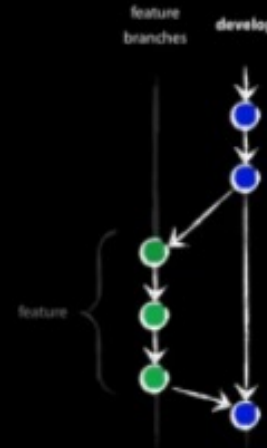


Local clone

Changes about  
to be stored

## Repository

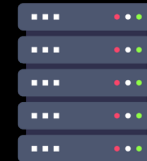
```
$ git commit
```



History and  
branches are  
store here

## Remote

```
$ git push  
$ git pull
```



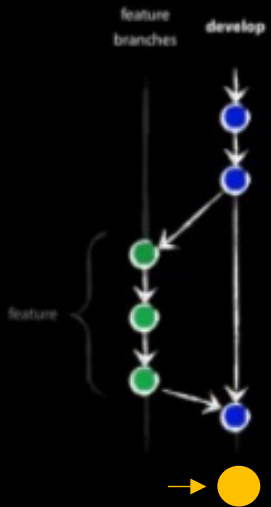
Remote repository

Same as  
repository but  
remote

# Key concepts

## Changes

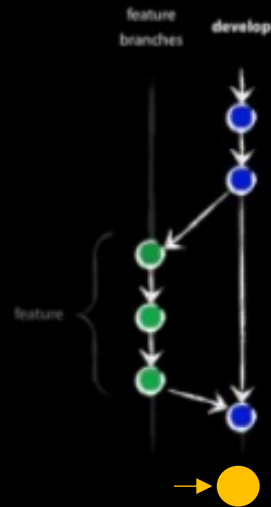
```
$ git status
```



Show changes

## Files

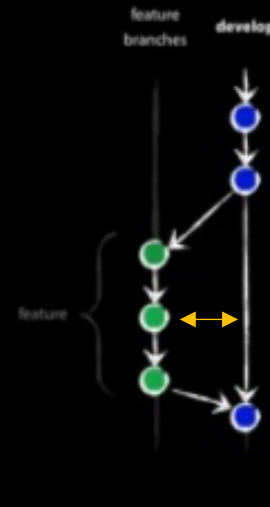
```
$ git diff
```



Differences in  
one or more files

## Branches

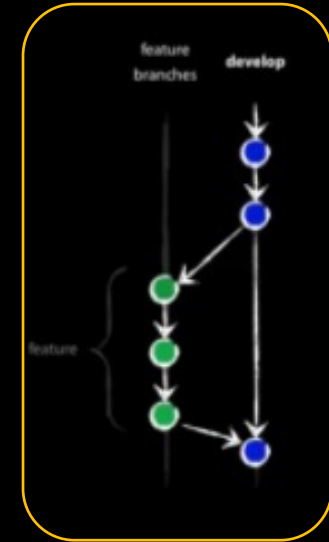
```
$ git checkout  
$ git branch
```



Shows and  
moves between  
branches

## History

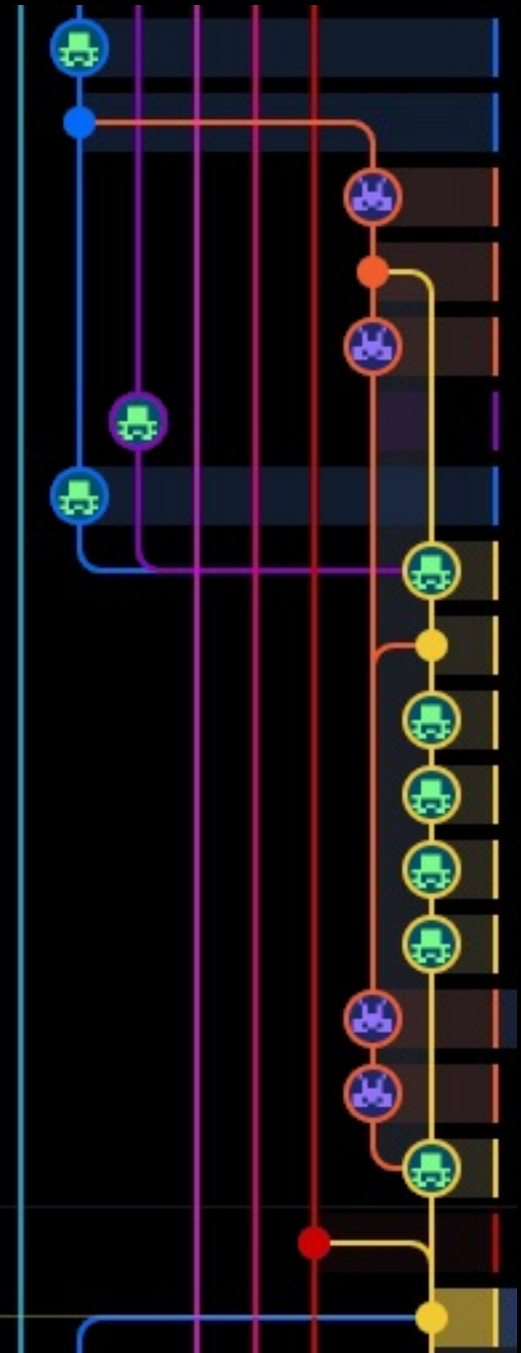
```
$ git log
```



Print the history  
of the repository

# Branch and Merge

- A **branch** is a split in the graph, and a **merge** is a merging of two split branches
- Conflicts
- Merge fails
- Fix merges

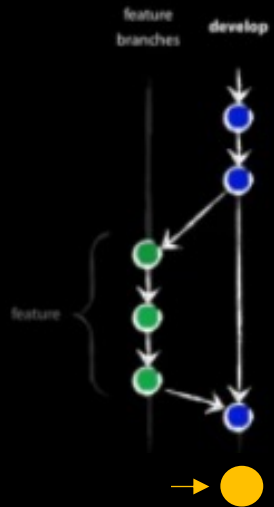




# Key concepts

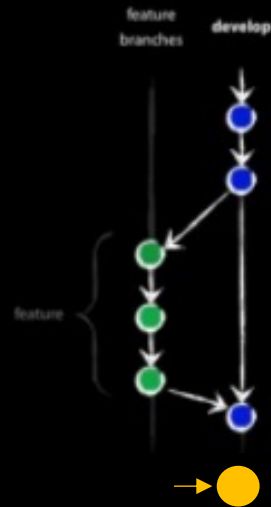
## Changes

```
$ git status
```



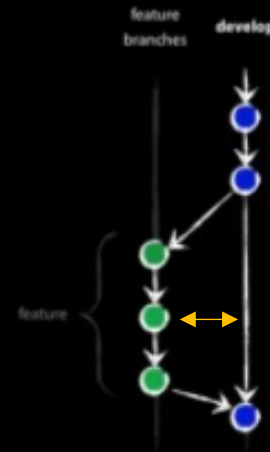
## Files

```
$ git diff
```



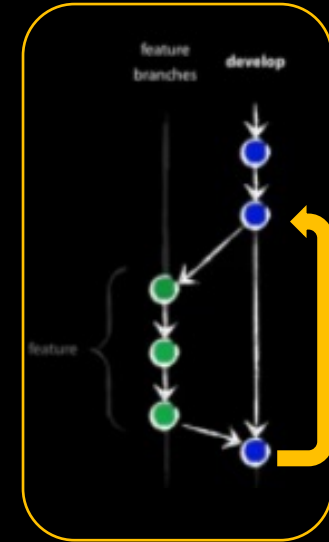
## Branches

```
$ git checkout  
$ git branch
```



## History

```
$ git log
```



## Re-write History

```
$ git rebase  
$ git reset
```



# \$ git status

## Something changed

On branch development

Your branch is up to date with 'origin/development'.

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: functions/02\_proc-dwi.sh

modified: functions/03\_SC.sh

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

## Nothing changed

Already up to date.

On branch master

Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

```
$ git log --graph --decorate --oneline
```

```
* a433270 Updated 'whatsNew' in v0.1.2
* 31e5c29 (origin/virtualization) Merge remote-tracking branch 'origin/master' into virtualization
|\
| * 5d6f612 (origin/recoverRtD) update docs additional tools
* | 1393bfa Merge branch 'virtualization' of https://github.com/MICA-MNI/micapipe into virtualization
|/
| * \ 238d165 Merge pull request #22 from PeerHerholz/virtualization
| | \
| | * | f9f8425 include nilearn in docker image
| | * | 4de8370 Merge branch 'master' of https://github.com/rcruces/micapipe into virtualization
| | |
| | | |
| | | * 1cba9f0 [ENH] reorder c69 surfaces, plot T1onDWI either Affine or SyN
| | | * 48538c3 [FIX] added missing semicolon to SC line 74
| | | * ae47902 [FIX] issue with transformations management in SC when only AFFINE was applied
| | | * 1aae113 [DOC] fixed typo
| | | * acad304 [FIX] removed unnecessary files
```

# \$ git diff <filename>

```
diff --git a/functions/03_SC.sh b/functions/03_SC.sh
index b9b8ee2..8194571 100755
--- a/functions/03_SC.sh
+++ b/functions/03_SC.sh
@@ -60,9 +60,9 @@ fod_wmN="${proc_dwi}/${idBIDS}_space-dwi_model-CSD_map-FOD_desc-wmNorm.mif"
     dwi_5tt="${proc_dwi}/${idBIDS}_space-dwi_desc-5tt.nii.gz"
     dwi_b0="${proc_dwi}/${idBIDS}_space-dwi_desc-b0.nii.gz"
     dwi_mask="${proc_dwi}/${idBIDS}_space-dwi_desc-brain_mask.nii.gz"
-    str_dwi_affine="${dir_warp}/${idBIDS}_space-dwi_from-dwi_to-nativepro_mode-image_desc-affine_"
+    str_dwi_affine="${dir_warp}/${idBIDS}_space-dwi_from-dwi${dwi_str}_to-nativepro_mode-image_desc-affine_"
     mat_dwi_affine="${str_dwi_affine}0GenericAffine.mat"
-    dwi_SyN_str="${dir_warp}/${idBIDS}_space-dwi_from-dwi_to-dwi_mode-image_desc-SyN_"
+    dwi_SyN_str="${dir_warp}/${idBIDS}_space-dwi_from-dwi${dwi_str}_to-dwi_mode-image_desc-SyN_"
     dwi_SyN_warp="${dwi_SyN_str}1Warp.nii.gz"
     dwi_SyN_affine="${dwi_SyN_str}0GenericAffine.mat"
     dti_FA="${proc_dwi}/${idBIDS}_space-dwi_model-DTI_map-FA.mif"
```

```
$ git branch
```

```
devTasks  
* development  
master  
virtualization
```

```
$ git checkout master
```

```
Switched to branch 'master'
```

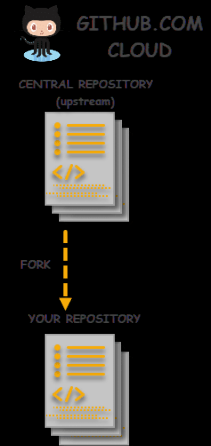
```
Your branch is behind 'origin/master' by 14 commits, and can be  
fast-forwarded.
```

```
(use "git pull" to update your local branch)
```

```
devTasks  
development  
* master  
virtualization
```

# Fork

- Make a copy of a repository from one account to another.
- Improve someone's code (then pull request)
- Reusing the code in a project



## `git reset --soft` and `--hard`

- `git reset --soft HEAD~` - move back one commit and keep your changes
- `git reset --hard HEAD~` - move back one commit and discard all changes

# Setup



# Set a GitHub account

The screenshot displays the GitHub profile of Raúl RC (@MICAMNI). The profile includes a custom avatar, a bio, and a list of repositories. The user has 25 followers and is following 24 users. The repositories section shows six public repositories, including `MRI_analytic_tools`, `R-plots`, `2020_NMA_surface-plot`, `micaopen`, `2019_uncertain_emotion_discrimination`, and `timeSeries`. The contribution graph shows 490 contributions in the last year, with a calendar view from March 2021 to March 2022. The user is affiliated with the Montreal Neurological Institute and has a website link to `https://rcruces.github.io`.

**Profile Information:**

- Name:** Raúl RC
- Username:** @MICAMNI
- Followers:** 25
- Following:** 24
- Location:** Montreal
- Website:** <https://rcruces.github.io>
- Twitter:** @rcruces2

**Repositories:**

- MRI\_analytic\_tools** (Public): Pipelines and tools for MRI analysis. Languages: HTML (7 stars, 8 forks).
- R-plots** (Public): Custom plots in R. Languages: R (3 stars, 4 forks).
- 2020\_NMA\_surface-plot** (Public): Jupyter Notebook (2 stars).
- micaopen** (Public): Forked from MICAMNI/micaopen. Languages: MATLAB (1 star, 1 fork).
- 2019\_uncertain\_emotion\_discrimination** (Public): Uncertain emotion detection. Languages: R (1 star).
- timeSeries** (Public): MATLAB (1 star).

**Contributions:**

- 490 contributions in the last year**
- Contribution settings:** 2022 (selected), 2021, 2020, 2019, 2018, 2017
- Calendar view:** Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec, Jan, Feb, Mar
- Legend:** Less (green), More (red)

**Achievements:**

- @MICAMNI**
- @sphinx-doc**



# Set your token

The screenshot shows the GitHub profile page for user Raúl RC (rcruces). The profile includes a bio, a repository grid, and a dropdown menu with the 'Settings' option highlighted. A yellow arrow points to the 'Settings' option.

**Profile Information:**

- Name:** Raúl RC
- Username:** rcruces
- Handle:** @MICA-MNI
- Followers:** 25
- Following:** 24

**Repositories:**

- MRI\_analytic\_tools** (Public)
  - HTML
  - 7 stars
  - 8 forks
- 2020\_NMA\_surface-plot** (Public)
  - Jupyter Notebook
  - 2 stars
- 2019\_uncertain\_emotion\_discrimination** (Public)
  - R
  - 1 star
- R-plots**
  - R
  - 3 stars
  - 4 forks
- micaopen** (Forked from MICA-MNI/micaopen)
  - MATLAB
  - 1 star
  - 1 fork
- timeSeries**
  - MATLAB
  - 1 star

**Dropdown Menu:**

- Signed in as rcruces
- Edit status
- Your profile
- Your repositories
- Your codespaces
- Your organizations
- Your projects
- Your stars
- Your gists
- Feature preview
- Help
- Settings**
- Sign out

# Set your token

Accessibility

Notifications

Access

Billing and plans

Emails

Password and authentication

SSH and GPG keys

Organizations

Moderation

Code, planning, and automation

Repositories

Packages

Pages

Saved replies

Security

Code security and analysis

Integrations

Applications

Scheduled reminders

Archives

Security log

Sponsorship log

Developer settings

raul rc

Your name may appear around GitHub where you contribute or are mentioned. You can remove it at any time.

Public email

Select a verified email to display

You have set your email address to private. To toggle email privacy, go to [email settings](#) and uncheck "Keep my email address private."

Bio

@MICA-MNI

You can @mention other users and organizations to link to them.

URL

<https://rcruces.github.io>

Twitter username

rcruces2

Company

Montreal Neurological Institute


You can @mention your company's GitHub organization to link it.

Location

Montreal

All of the fields on this page are optional and can be deleted at any time, and by filling them out, you're giving us consent to share this data wherever your user profile appears. Please see our [privacy statement](#) to learn more about how we use this information.

Update profile



Edit

Profile settings

# Set your token

Settings / Developer settings

GitHub Apps

OAuth Apps

**Personal access tokens**


## Personal access tokens

Generate new token Revoke all

Tokens you have generated that can be used to access the [GitHub API](#).

**mytoken** — *admin:enterprise, admin:gpg\_key, admin:org, admin:org\_hook, admin:public\_key, admin:repo\_hook, delete:packages, delete\_repo, gist, notifications, repo, user, workflow, write:discussion, write:packages* Last used within the last 2 weeks 

Delete

 This token has no expiration date.

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).



Store it in a save place!!!!

# Install Git on your local machine

## Installing

- Linux:
  - <http://git-scm.com/book/en/v2/Getting-Started-Installing-Git>
- Mac:
  - <http://git-scm.com/download/mac>
- Windows:
  - <https://gitforwindows.org>

## Setup git

- `git config --global user.name "Your name"`
- `git config --global user.email "me@example.com"`

# Resources

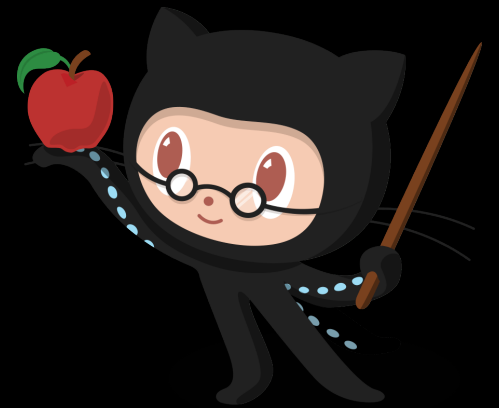


- Cheat sheet:
  - <https://training.github.com/downloads/github-git-cheat-sheet/>
- The simple guide slices
  - <https://rogerdudler.github.io/git-guide/>
- Git book
  - <https://git-scm.com/book/en/v2>
- Learn Git branching:
  - <https://learngitbranching.js.org>



# Git/GitHub basics

*hands on*



# Basics

- Clone a repository
- Create a repository
- Create a file
- Commit
- Show history
- Push / Pull
- Change
- Show history

