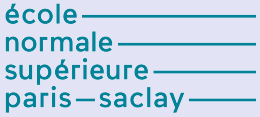
A short intro to git

Thomas Moreau ENS Paris-Saclay - CMLA



The goal of git...

Keeping track of a repository history:

- Versioning?? Why bother?
- Making archive of the full repo:
  Create a `zip` archive at fixed interval. This can become big.
- **Git!**

Sharing code with co-workers:

- Email changed files
- Dropbox-like synchronization
- **Git!**

The goal of git...

Keeping track of a repository history:

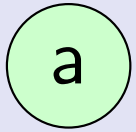
- Versioning?? Why bother?
- Making archive of the full repo:
  Create a `zip` archive at fixed interval. This can become big.
- **Git!**

Sharing code with co-workers:

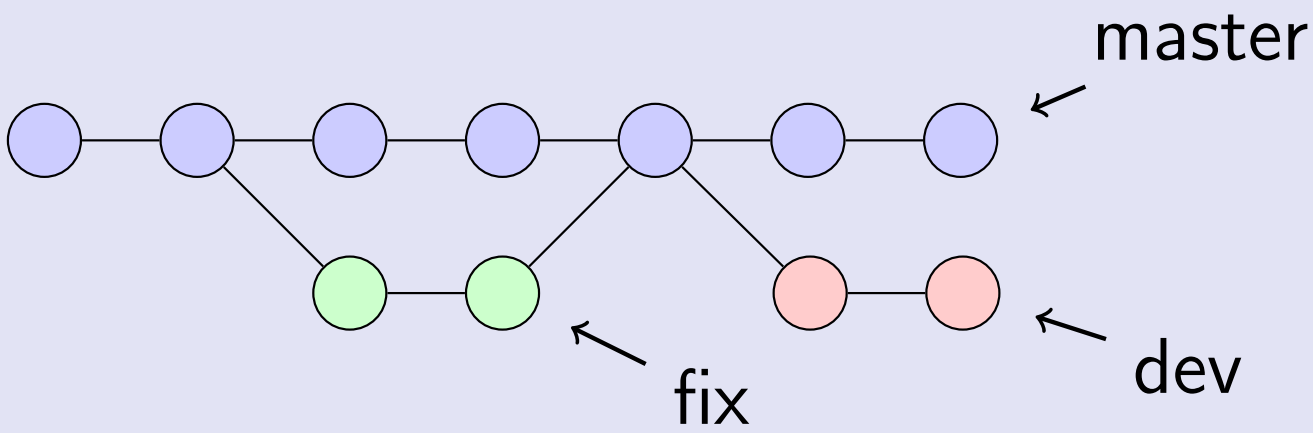
- Email changed files
- Dropbox-like synchronization
- **Git!**

Git is a tool to keep track of history and share code.

How does it work?

- Base object in git is the `commit` 

- Identifier commit has a hash `a : "baef63472"`.
- Git is base on a history tree notion.
  Each commit is linked to its parent.



3 branches: {master; fix; dev}

What is a commit?

Keep track of code changes with its parent by comparing lines:

```
103  136  \begin{itemize}
104       - \item Should we add login/password for identification ? Probably yes
105       -\item How big should the databases be ? Probably no more than 20-30 subjects
106       -\item How should the databases be stored ? ....
     137  + \item Should we add login/password for identification? Probably yes
     138  +  \item How big should the databases be? Probably no more than 20-30 subjects
     139  +  \item How should the databases be stored? ....
107  140  \end{itemize}
```

This is called a `diff`. The diff between the current state and the last commit is given by

```
$ git diff
```

Git diff

The `git diff` command permits to access the differences between 2 commits.

- `(dev)$ git diff 1ab12f3e c837f6b5e`
- `(dev)$ git diff master dev`
- `(dev)$ git diff master`

Creating a new commit

`git status`: show the current state of a repo.

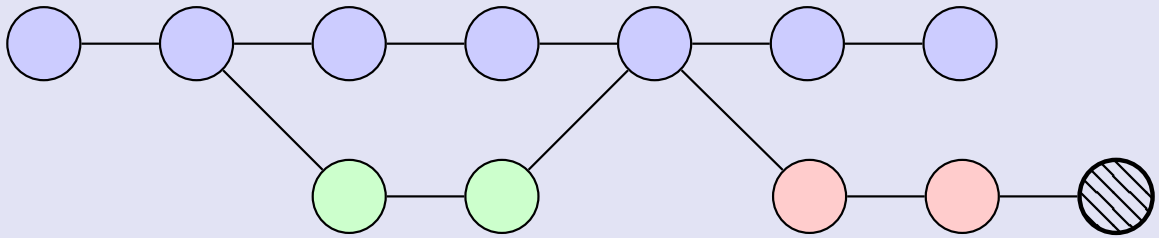
Modified file; Ignored files; Staged files; conflicts....

The command `git add` permits to prepare a commit:

- `(dev)$ git add $FILE1 $FILE2 ...`
- `(dev)$ git add -u`: Add all files modified

The command `git commit` permits to create a commit.

Composed of a message; put at the end of the current branch!



Remotes

Communication with the server:

- *HTTPS*: remote appears as
  `https://reine.cmla.ens-cachan.fr/user/repo`
- *SSH*: remote appears as
  `git@reine.cmla.ens-cachan.fr:user/repo`

To see the available remotes: `git remote -v`

Ssh disgression

- Secured communication protocole based on keys:
  Need to generate a key to communicate!

- Default use the port 22, CMLA use port 2333:
  Need configuration to access outside!

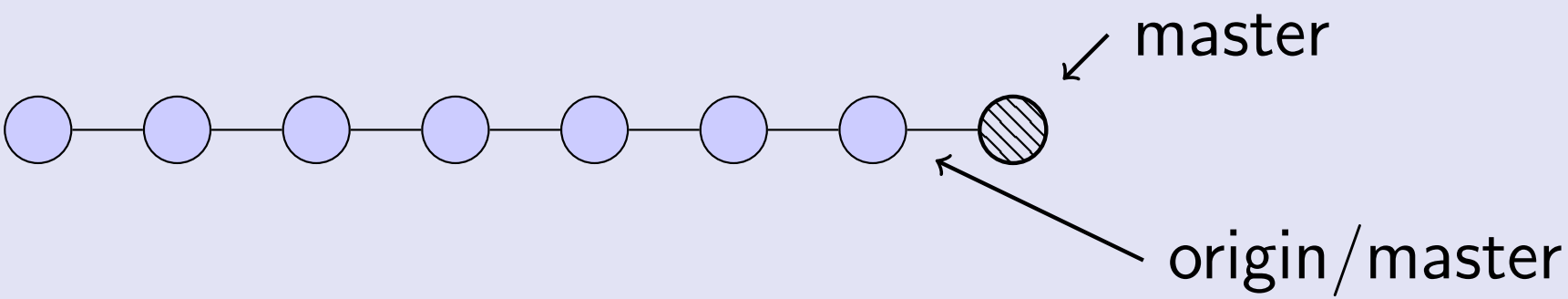
See https://reine.cmla.ens-cachan.fr/mldma/presentation_git/wikis/git_setup

Interacting with a server (remote)

Git is a distributed system. Each `remote` store the history tree.

There is a need for synchronization steps!

```
(dev)$ git push
```

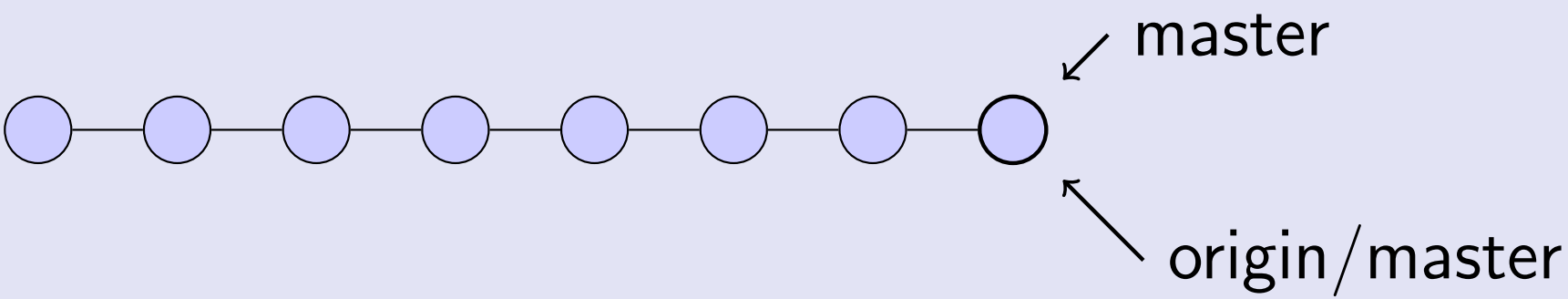


Interacting with a server (remote)

Git is a distributed system. Each `remote` store the history tree.

There is a need for synchronization steps!

```
(dev)$ git push
```

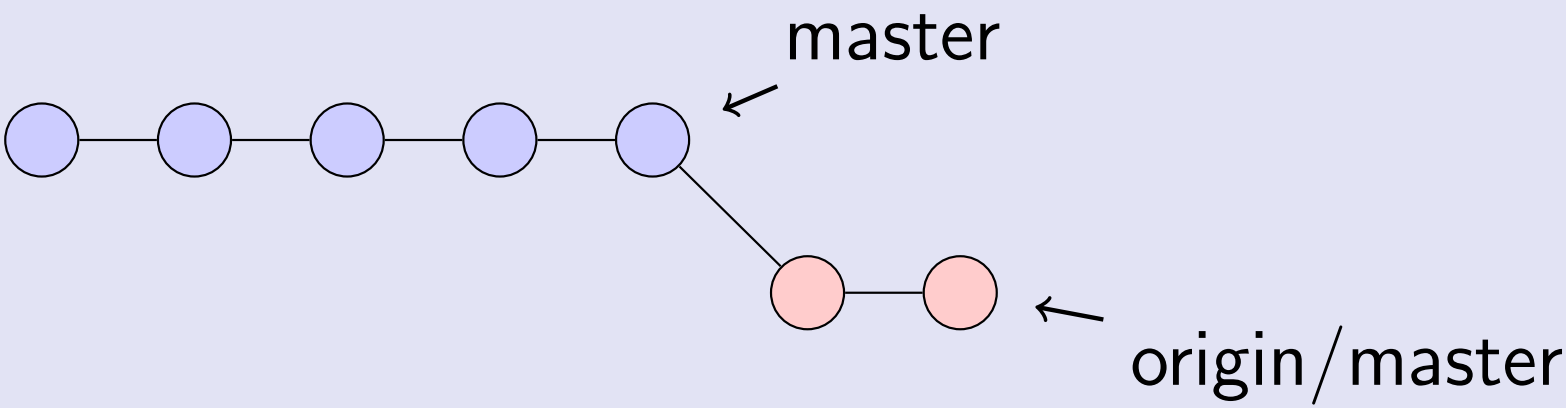


Interacting with a server (remote)

Git is a distributed system. Each `remote` store the history tree.

There is a need for synchronization steps!

```
(master)$ git pull
```

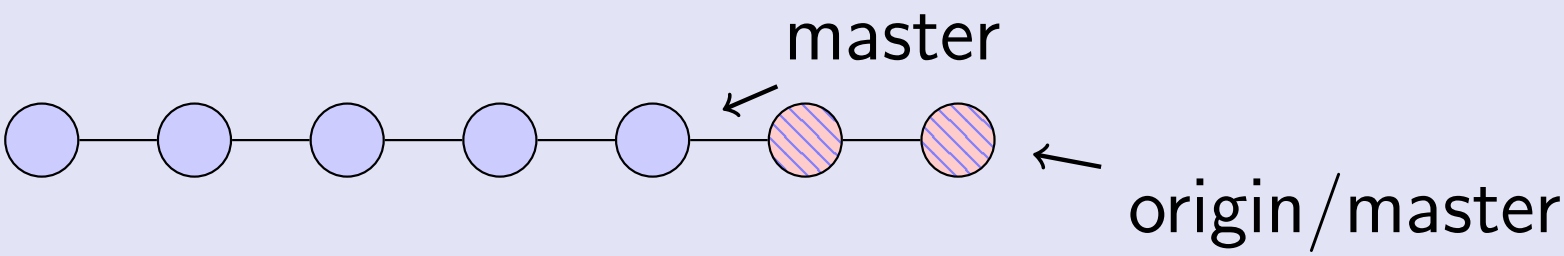


Interacting with a server (remote)

Git is a distributed system. Each `remote` store the history tree.

There is a need for synchronization steps!

```
(master)$ git pull
```

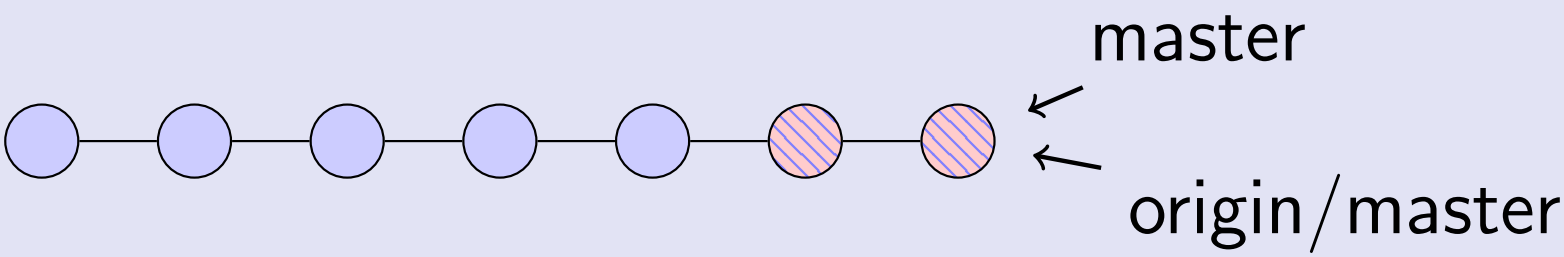


Interacting with a server (remote)

Git is a distributed system. Each `remote` store the history tree.

There is a need for synchronization steps!

```
(master)$ git pull
```



Extra concepts

- Rebase
- Merge
- Conflicts
- ...

That's all folks!

Questions?