Recherche Reproductible

Module 1: Introduction à Git

Practical Git

by Guille Polito
@GuillePolito

Practical 

What are we going to do today?

0. Install Git?
1. Create a Repository/Forge
2. Create a Clone
3. Commit (+ status, + add)
4. Fetch, Pull et Push

0. Install Git

- **OSX:** if you have XCode you probably have it already. Otherwise, install homebrew, then

```
$ brew install git
```

- **Linux:** `$ sudo apt-get install git`

- **Windows:** Download Git for Windows (sorry…)

Verify it! `$ git –version`

1. Create a Repository

1. Create a login in GitHub or Gitlab
    1. https://gitlab.cristal.univ-lille.fr
2. Create a repository (name it e.g., git-tutorial)
3. Get the HTTPS url
4. Clone

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

2. Commit

```
[ create some file ]
$ git status

$ git add [file]
$ git status

$ git commit -m “message”
$ git status
```

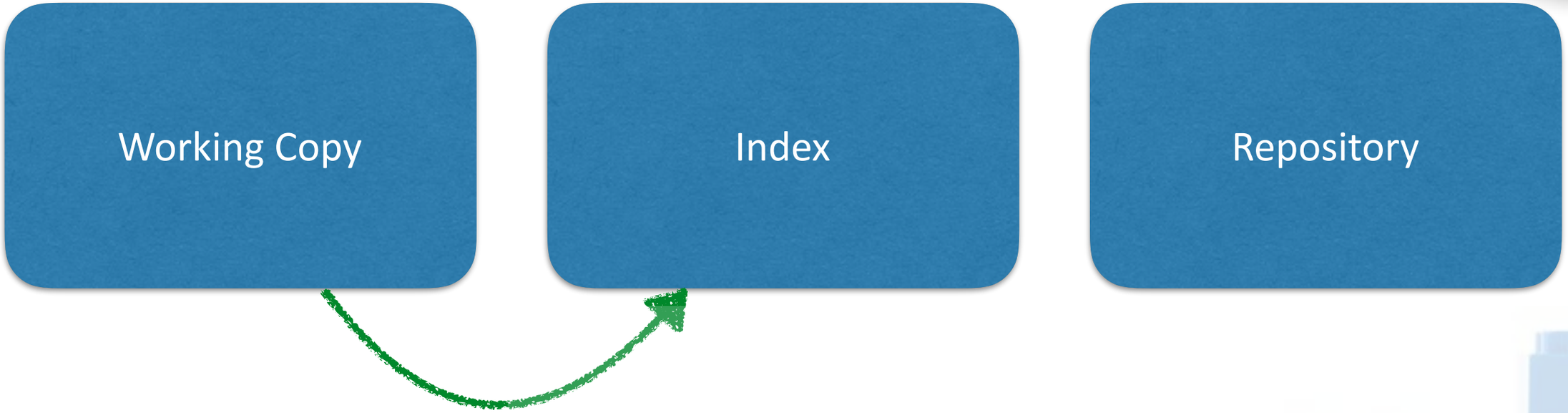
git as a 2-stage Database

Remote repositories

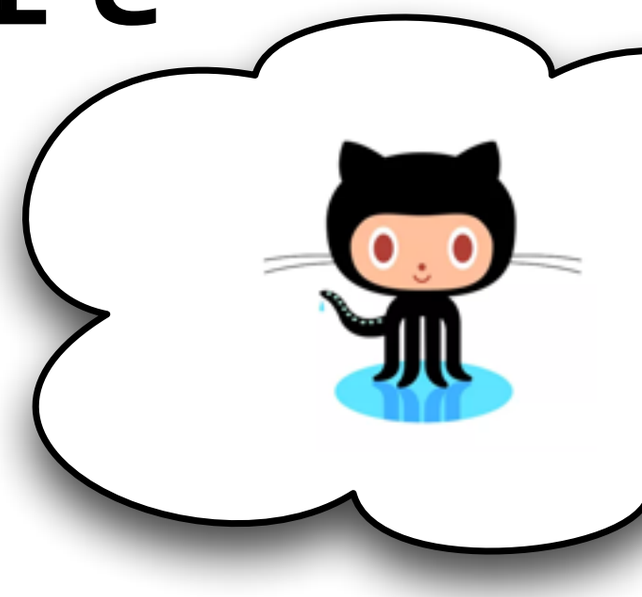
Working Copy	Index	Repository
What you see	A “hidden” staging area (sort of)	Where commits are

git

add



git commit



Working Copy

Index

Repository

2. Publish your code

```
$ git push
```

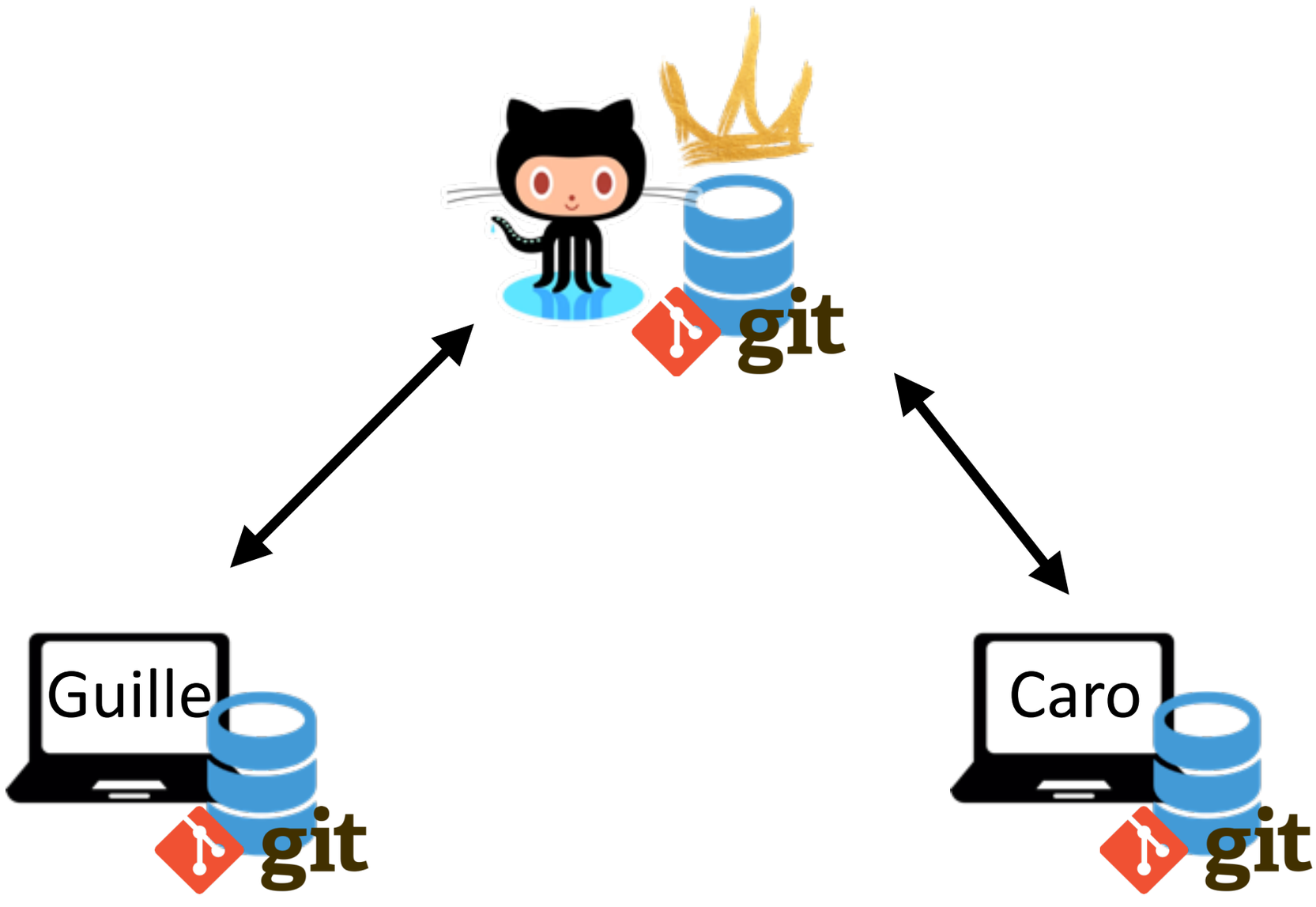
git push

Working Copy

Index

Repository

3. Collaboration



Now in pairs

1. Give permissions to your repository to your partner

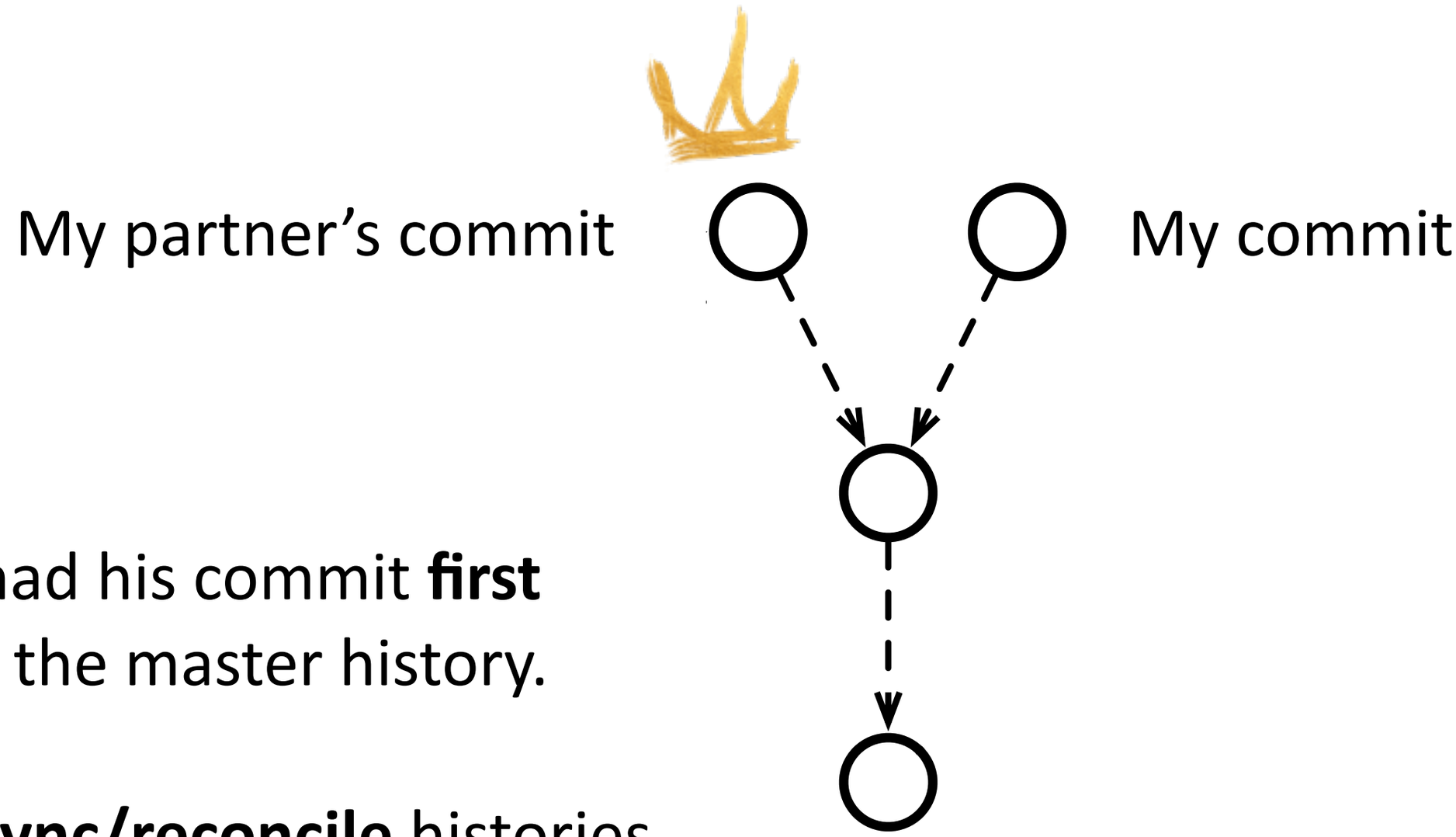
2. Clone your partner’s repository

3. Start doing commit & push together

    - Did your **push** got **rejected**? You branched history!

    - Git requires that all history is synchronized

Why my push got rejected



My partner had his commit **first** integrated in the master history.

We have to **sync/reconcile** histories

git pull

fetch

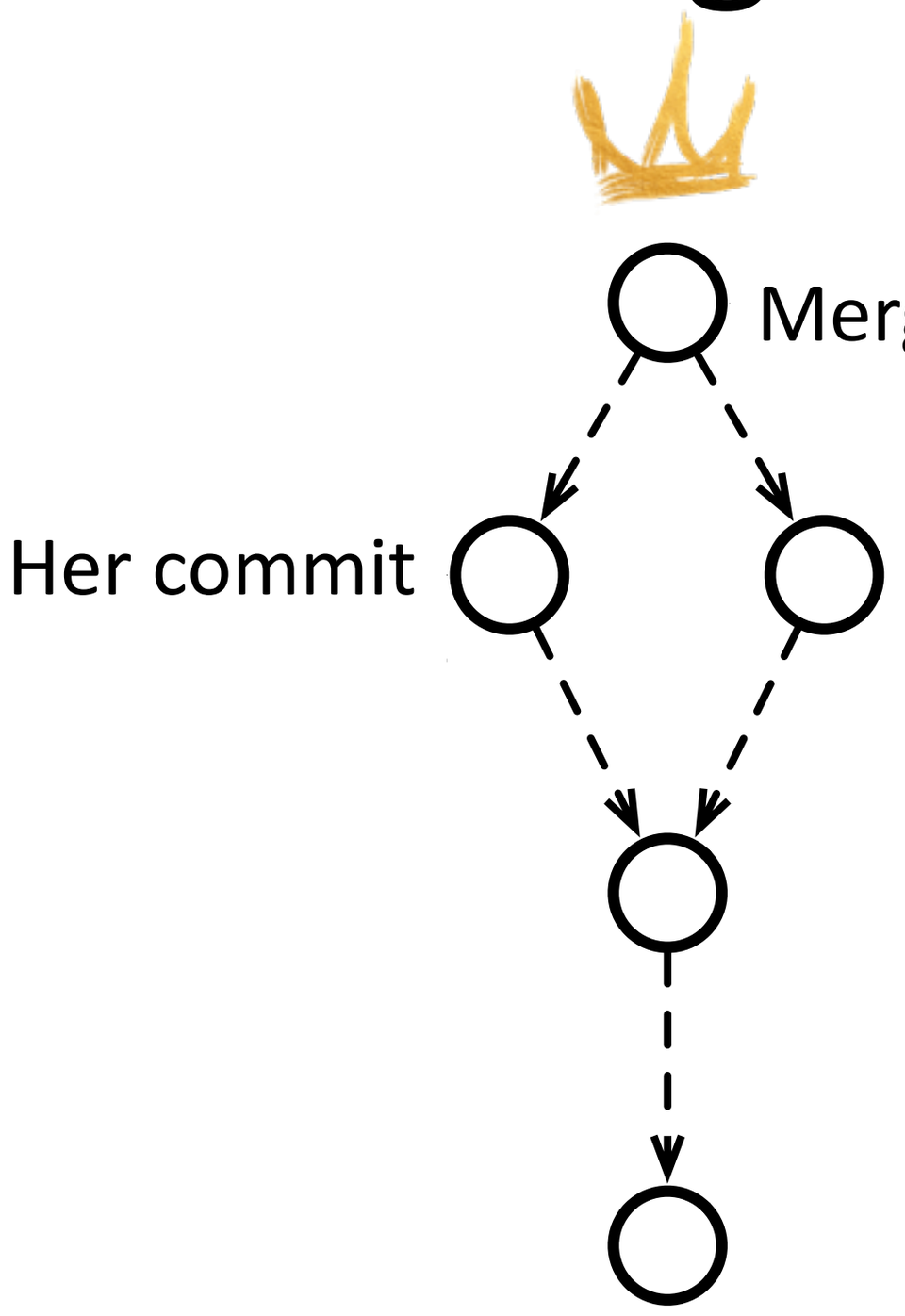
merge

Working Copy

Index

Repository

A first glimpse of a merge




```
[ edit the file ]
$ git status

$ git add [file]
$ git status

$ git commit -m “message”
$ git status
```

Conflicts can happen!

```
Some text without conflict.
<<<<<<< yours
I modified this line.
=======
I modified it too!
>>>>>>> theirs
More text without conflict.
```

Resolution 1: Keep mine

```
Some text without conflict.
<<<<<<< yours
I modified this line.
=======
I modified it too!
>>>>>>> theirs
More text without conflict.
```

```
Some text without conflict.
I modified this line.
More text without conflict.
```

Resolution 2: Keep theirs

```
Some text without conflict.
<<<<<<< yours
I modified this line.
=======
I modified it too!
>>>>>>> theirs
More text without conflict.
```

```
Some text without conflict.
I modified it too!
More text without conflict.
```

Resolution 3: Mix and Match

```
Some text without conflict.
<<<<<<< yours
I modified this line.
=======
I modified it too!
>>>>>>> theirs
More text without conflict.
```

```
Some text without conflict.
We both modified this line.
More text without conflict.
```

Summary

Remote repositories

Working Copy	Index	Repository
What you see	A “hidden” staging area (sort of)	Where commits are

- Add+commit save changes into the database
- Push sends our changes to the remote repository
- Pull brings changes from the remote repository
- Merge is automatic unless there are conflicts