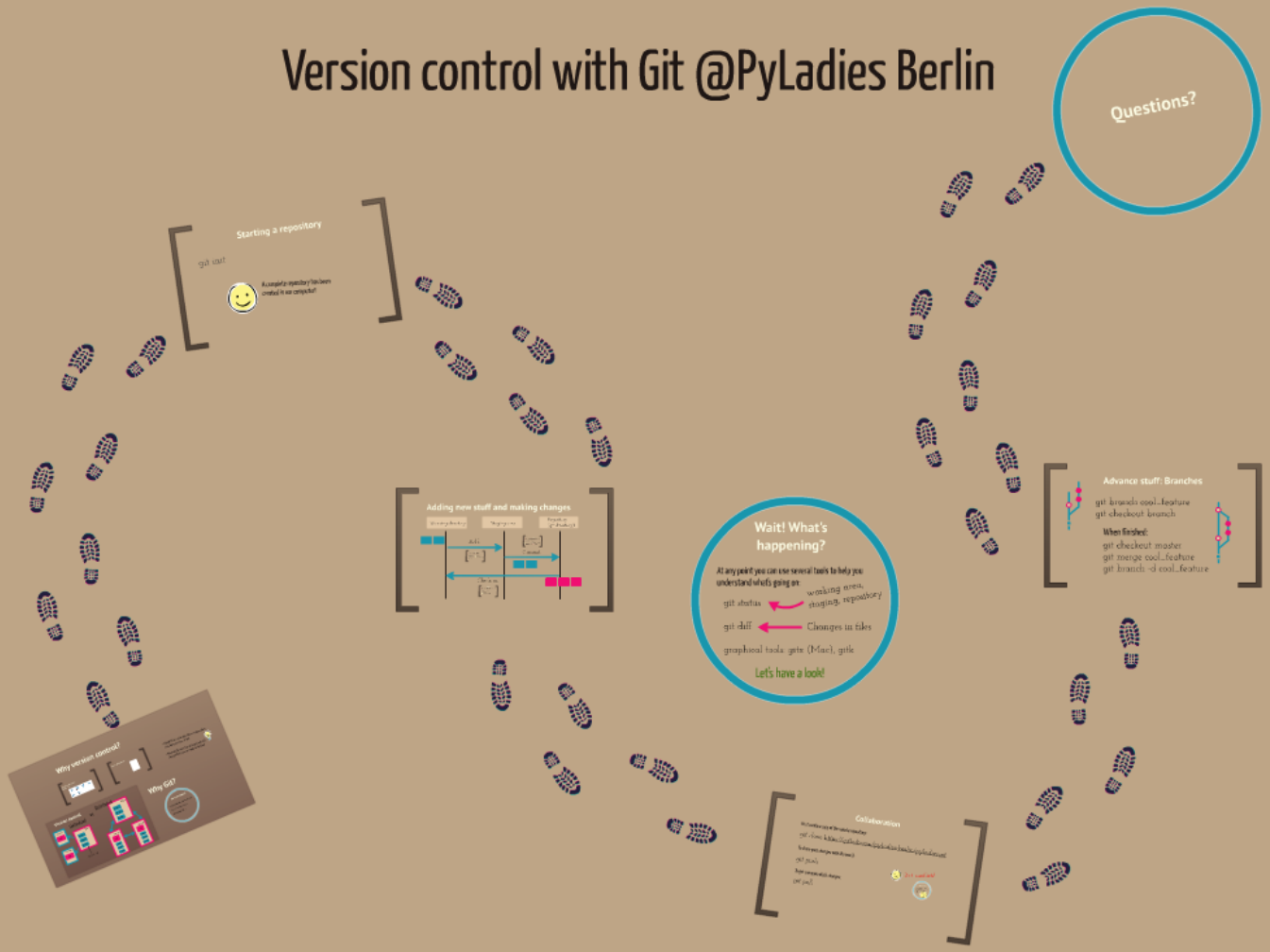
Version control with Git @PyLadies Berlin



Patricia Garcia
@patggs

Version control with Git @PyLadies Berlin

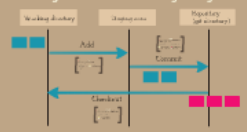
Questions?

Starting a repository

git init

A complete repository has been created in user computer!

Adding new stuff and making changes



Wait! What's happening?

At any point you can use several tools to help you understand what's going on:

git status ← working area, staging, repository

git diff ← Changes in files

graphical tools: gitx (Mac), gitk

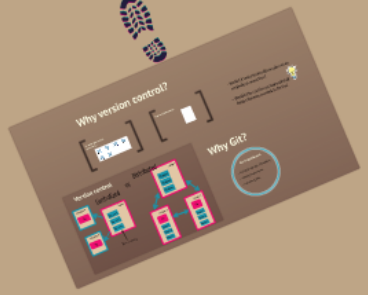
Let's have a look!

Advance stuff: Branches

git branch cool_feature
git checkout branch

When finished:

git checkout master
git merge cool_feature
git branch -d cool_feature



Collaboration

git clone https://github.com/pyladies-berlin/pyladies.git

Share your changes with the world:

git push

Get everyone else's changes:

git pull

but, conflicts!

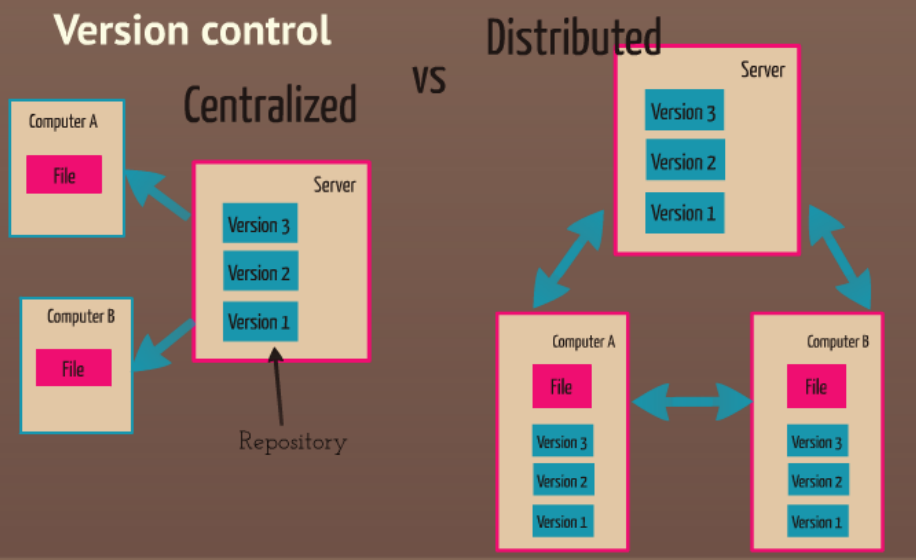
Patricia Garcia
@patggs

Why version control?

An early approach to versioning...

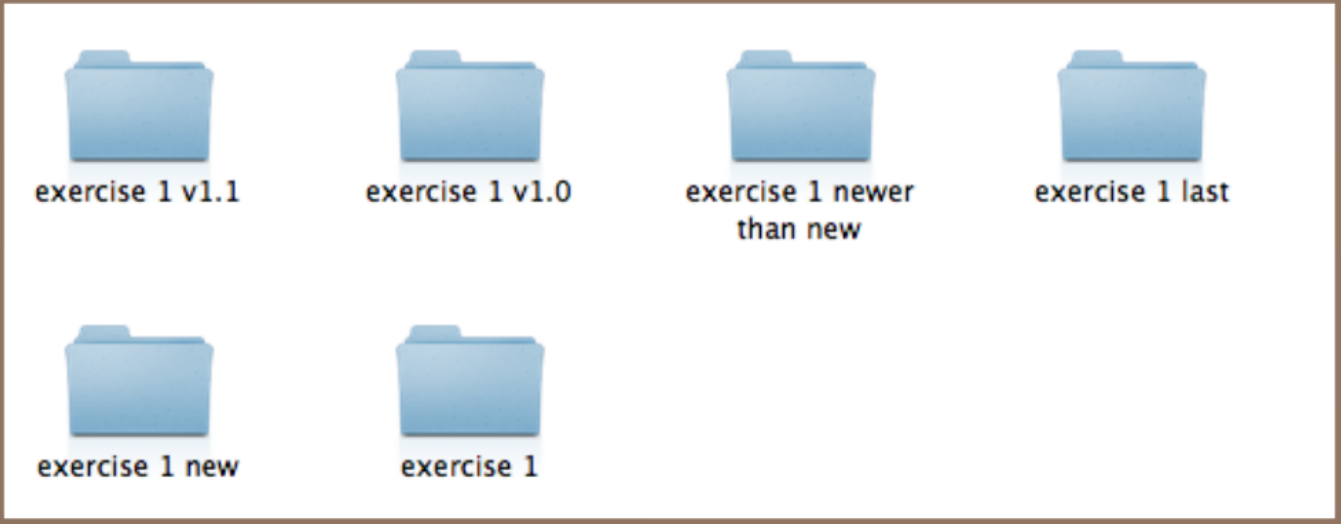
And to collaboration...

- Wouldn't it be nice to store files in a place where everybody can access them?
- Wouldn't it be nice if we can keep track of all changes that were ever made to the files?



Why Git?

An early approach to versioning...



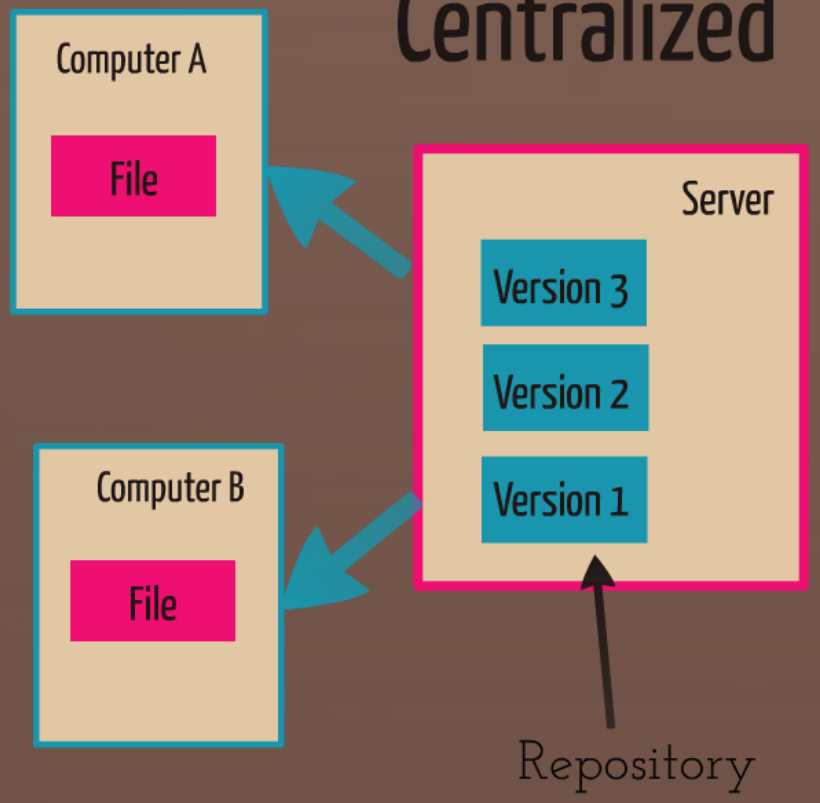
And to collaboration...

- Wouldn't it be nice to store files in a place where everybody can access them?

- Wouldn't it be nice if we can keep track of all changes that were ever made to the files?

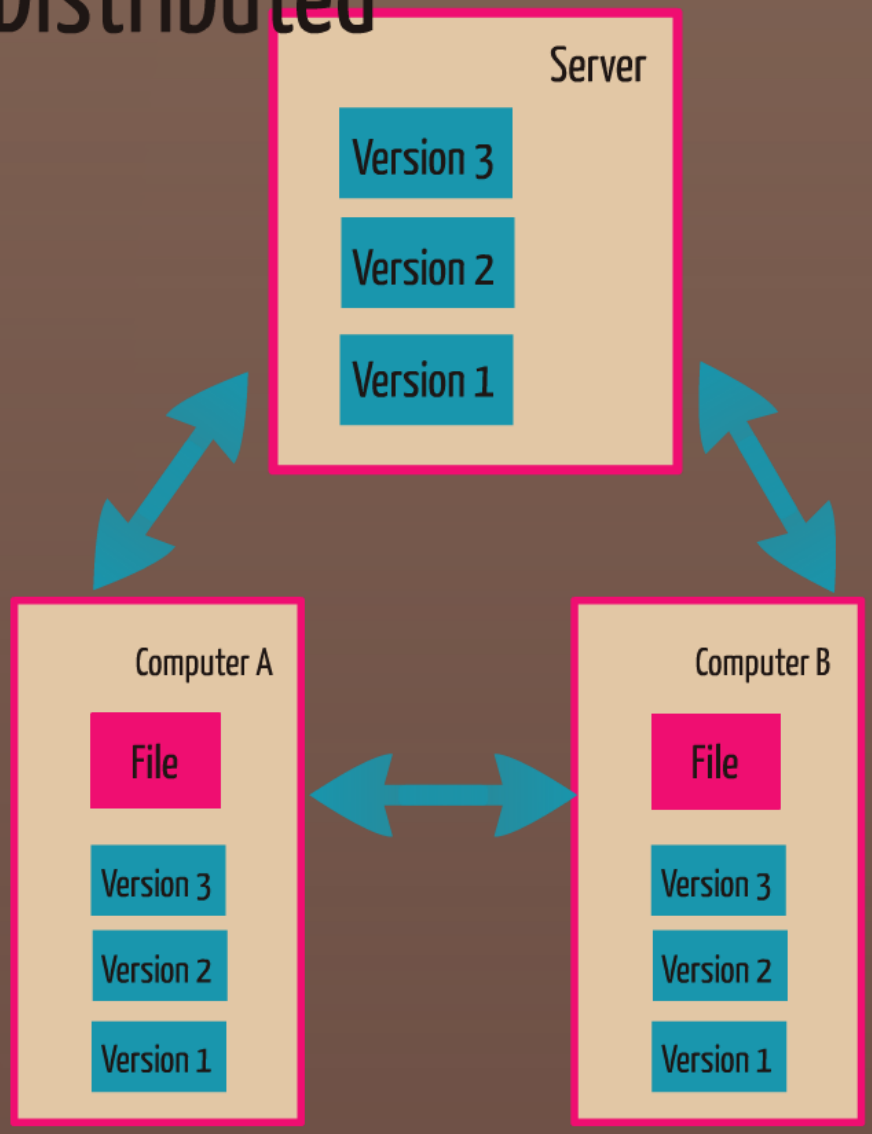
Version control

Centralized



vs

Distributed



Git is distributed

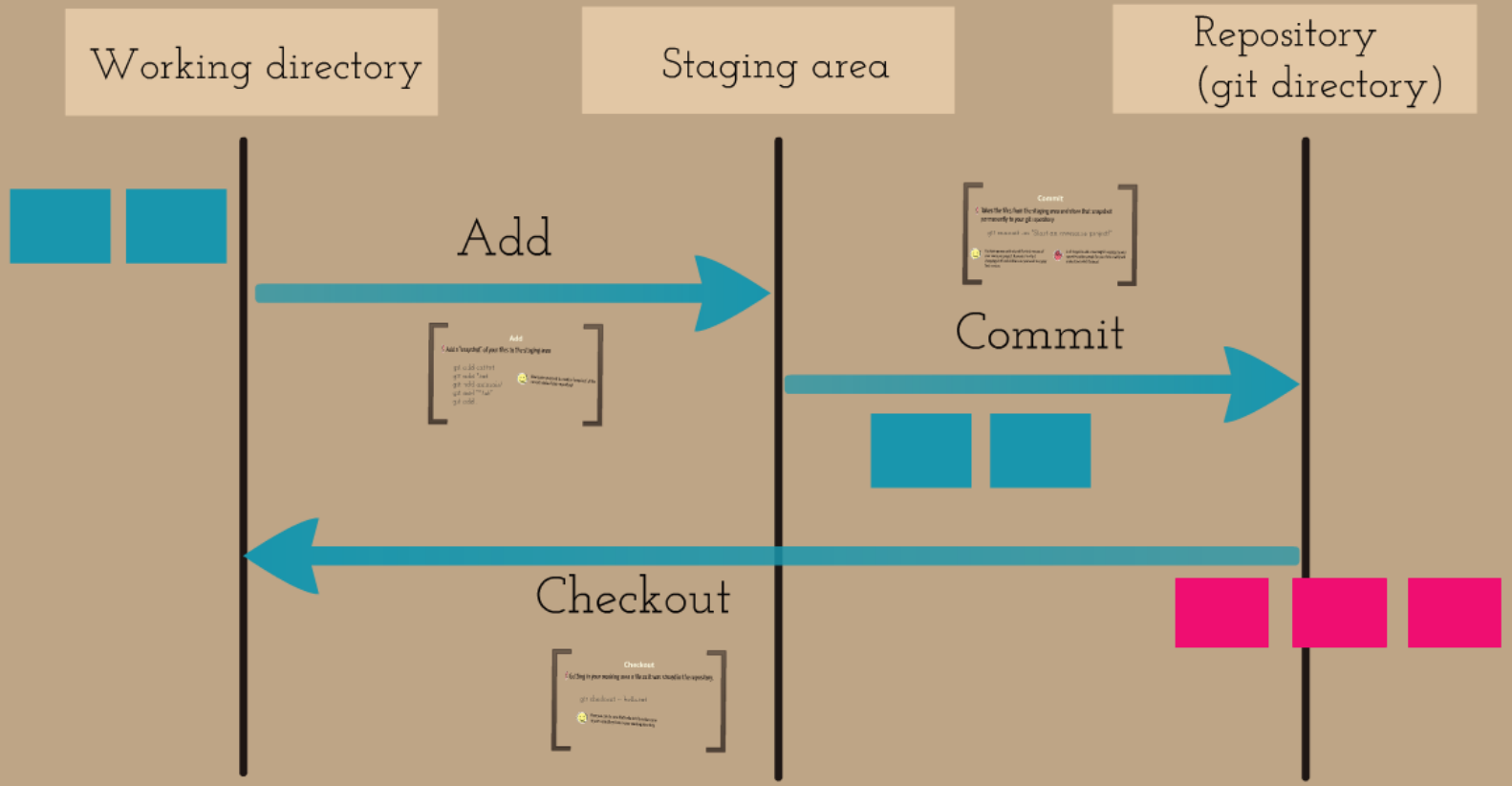
- Everybody has a full copy of the repository.
- Commiting changes is quicker.
- Allows working offline.

Starting a repository

git init

A complete repository has been created in our computer!

Adding new stuff and making changes



Add

Add a "snapshot" of your files to the staging area

```
git add cat.txt
git add *.txt
git add animals/
git add "*.txt"
git add .
```

Now you're prepared to create a "snapshot" of the current state of your repository!

Commit

Takes the files from the staging area and store that snapshot permanently to your git repository

```
git commit -m "Start an awesome project!"
```

You have permanently stored the first version of your awesome project. Now you can start changing stuff and still be sure you won't lose your first version.

Don't forget to add a meaningful message to your commit so other people (or your future self) will understand what it's about

Checkout

! Getting in your working area a file as it was stored in the repository.

```
git checkout -- hello.txt
```

Now you can be sure that hello.txt file is the same in your repository than in your working directory.

Wait! What's happening?

At any point you can use several tools to help you understand what's going on:

git status ← working area, staging, repository

git diff ← Changes in files

graphical tools: gitx (Mac), gitk

Let's have a look!

Collaboration

First create a copy of the remote repository:

git clone https://github.com/pyladies-berlin/pyladies.git

To share your changes with the world:

git push

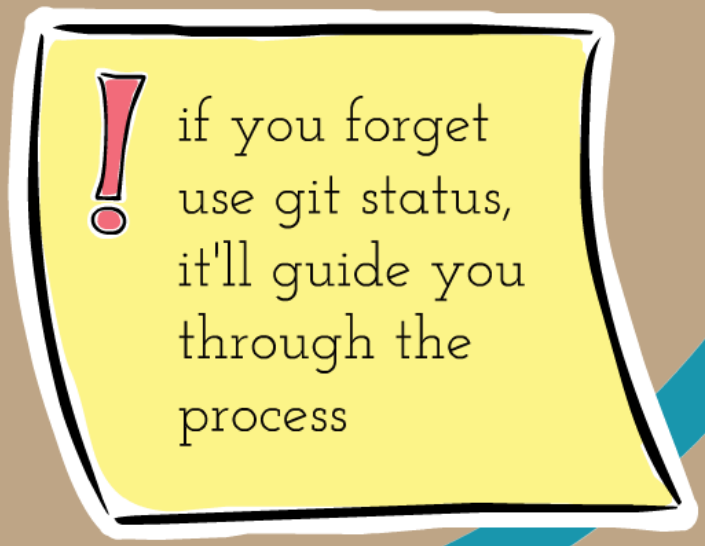
But.. conflicts!

To get everyone else's changes:

git pull

Don't panic!

- Edit the file with conflicts selecting the parts you want.
- Add the conflicting files for staging.
- Commit.
- Done!



Advance stuff: Branches

```
git branch cool_feature
git checkout branch
```

When finished:

```
git checkout master
git merge cool_feature
git branch -d cool_feature
```

Questions?

Version control with Git @PyLadies Berlin

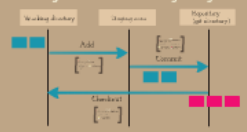
Questions?

Starting a repository

git init

A complete repository has been created in user computer!

Adding new stuff and making changes



Wait! What's happening?

At any point you can use several tools to help you understand what's going on:

git status ← working area, staging, repository

git diff ← Changes in files

graphical tools: gitx (Mac), gitk

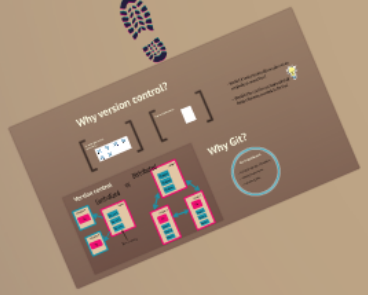
Let's have a look!

Advance stuff: Branches

git branch cool_feature
git checkout branch

When finished:

git checkout master
git merge cool_feature
git branch -d cool_feature



Collaboration

git clone https://github.com/pyladies-berlin/pyladies.git

Do some changes, add (the work)

git push

Forget everyone else's changes

git pull

but, conflicts!

Patricia Garcia
@patggs