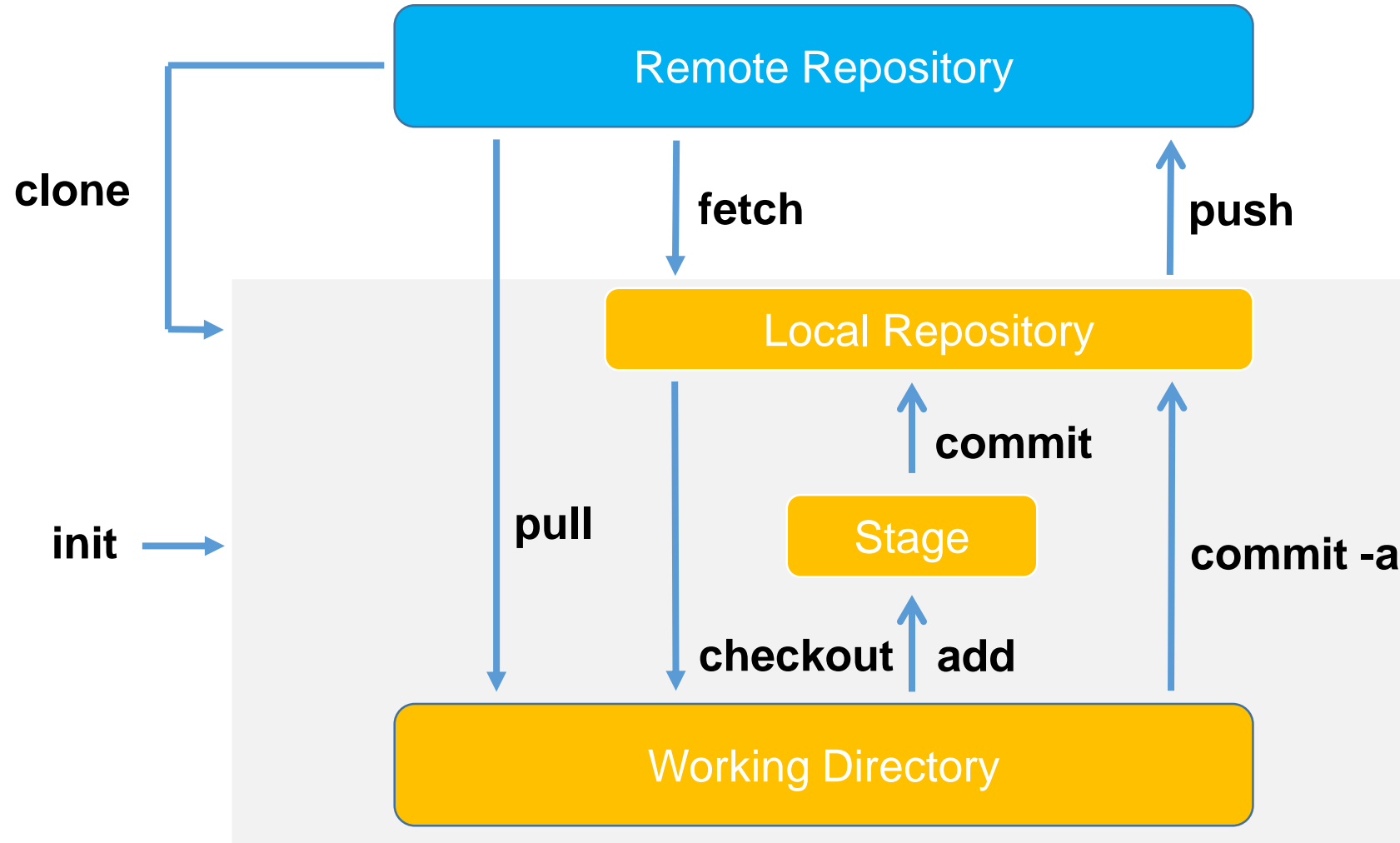


Understanding Git



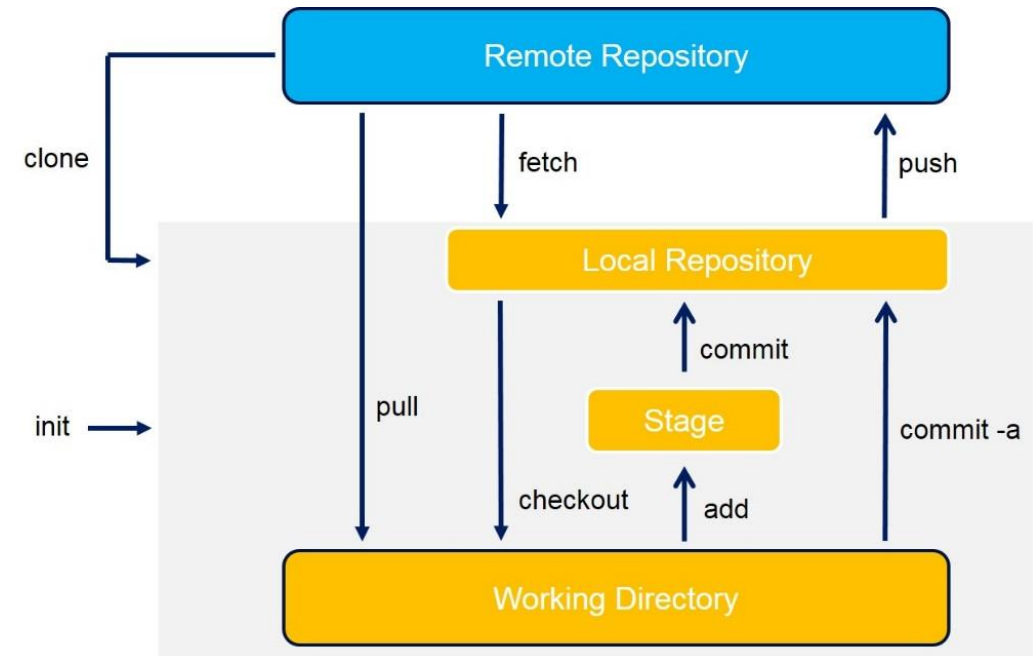
Understanding Git

Obtain a repository

- **git init**
Create an empty Git repository or reinitialize an existing one.
- **git clone <repository>**
Clone a repository into a new directory.

Example:

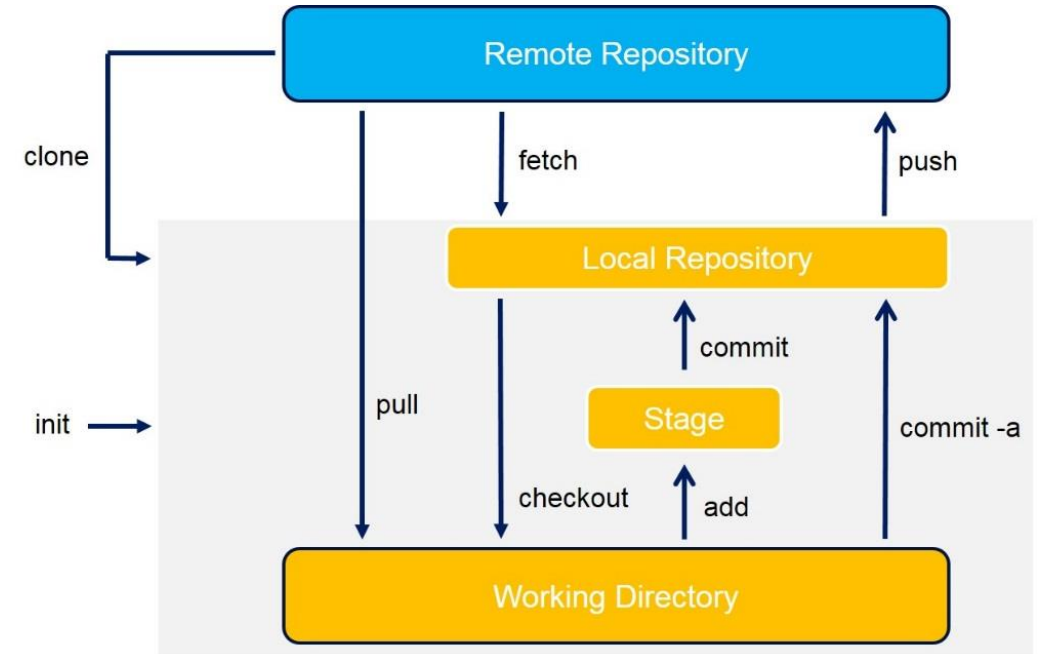
```
git clone https://github.com/gtrensch/SoftwareDevInScience2021.git
```



Understanding Git

Get changes from a remote repository

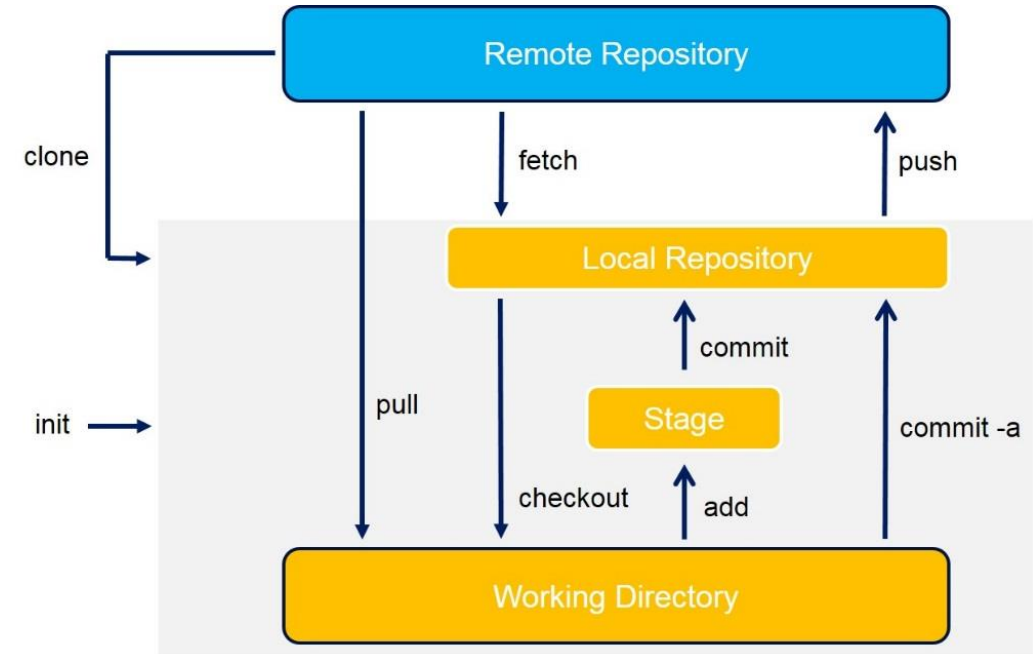
- **git fetch**
Download objects and refs from another repository.
- **git pull <repository>**
Fetch from and integrate with another repository or a local branch.
(shorthand for *git fetch* followed by *git merge FETCH_HEAD*)



Understanding Git

Push changes to a remote repository

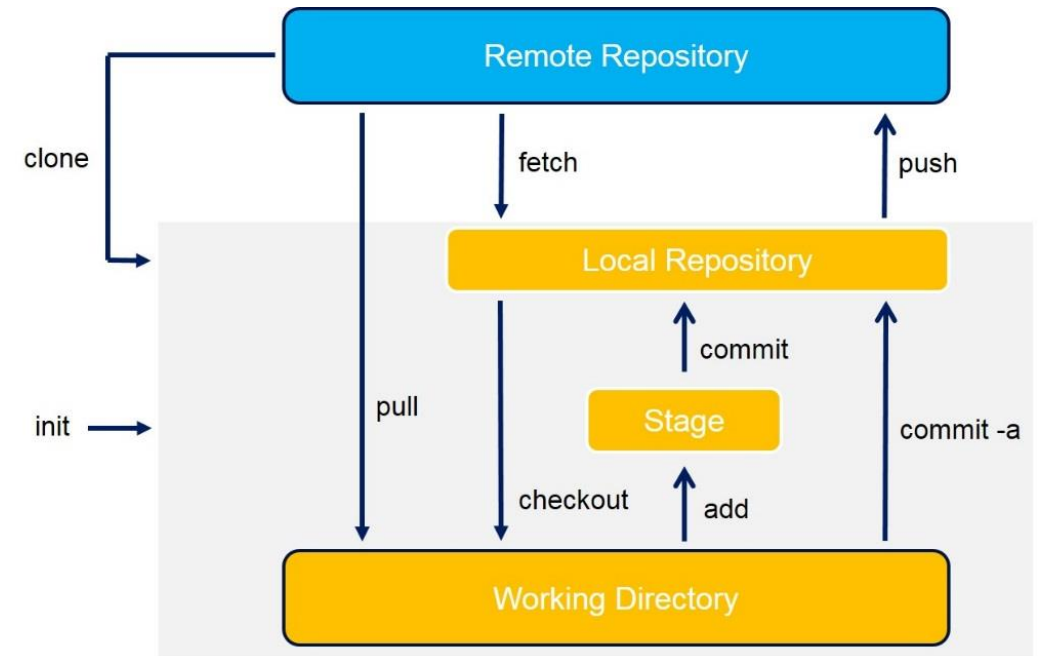
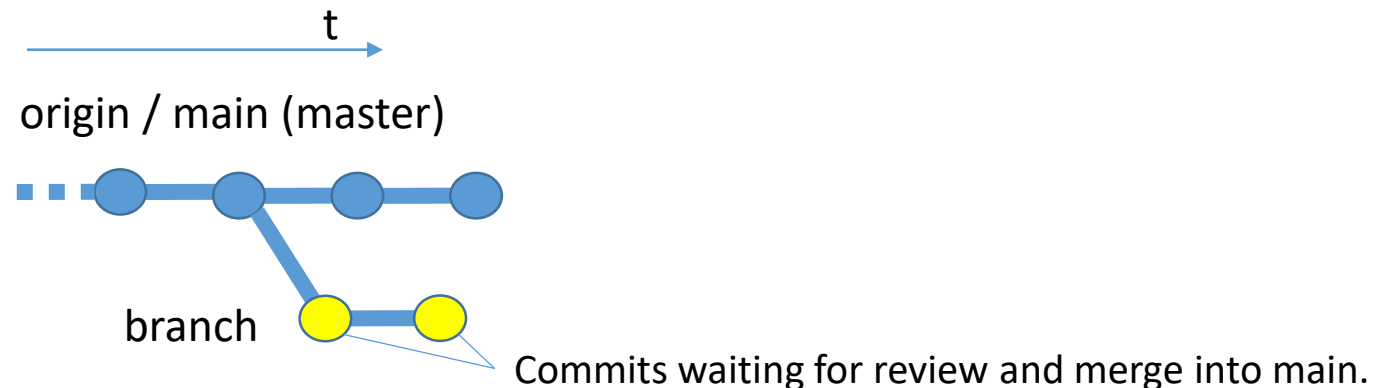
- **git add <file_name(s)>**
Add file contents to the index (stage).
- **git commit -m <message>**
Record changes to the repository.
- **git push <repository>**
Update remote refs along with associated objects.



Understanding Git

Branching

- **git checkout -b <new_branch>**
- **git branch -c <new_branch>**
Both commands creates a new branch.
- **git branch -d <old_branch>**
Deletes a branch.



Understanding Git

Useful Git commands

- **git status**

Show the working tree status.

- **git reset**

Reset current HEAD to the specified state.

- **git diff**

Show changes between commits, commit and working tree, etc.

- **git merge**

Join two or more development histories together.

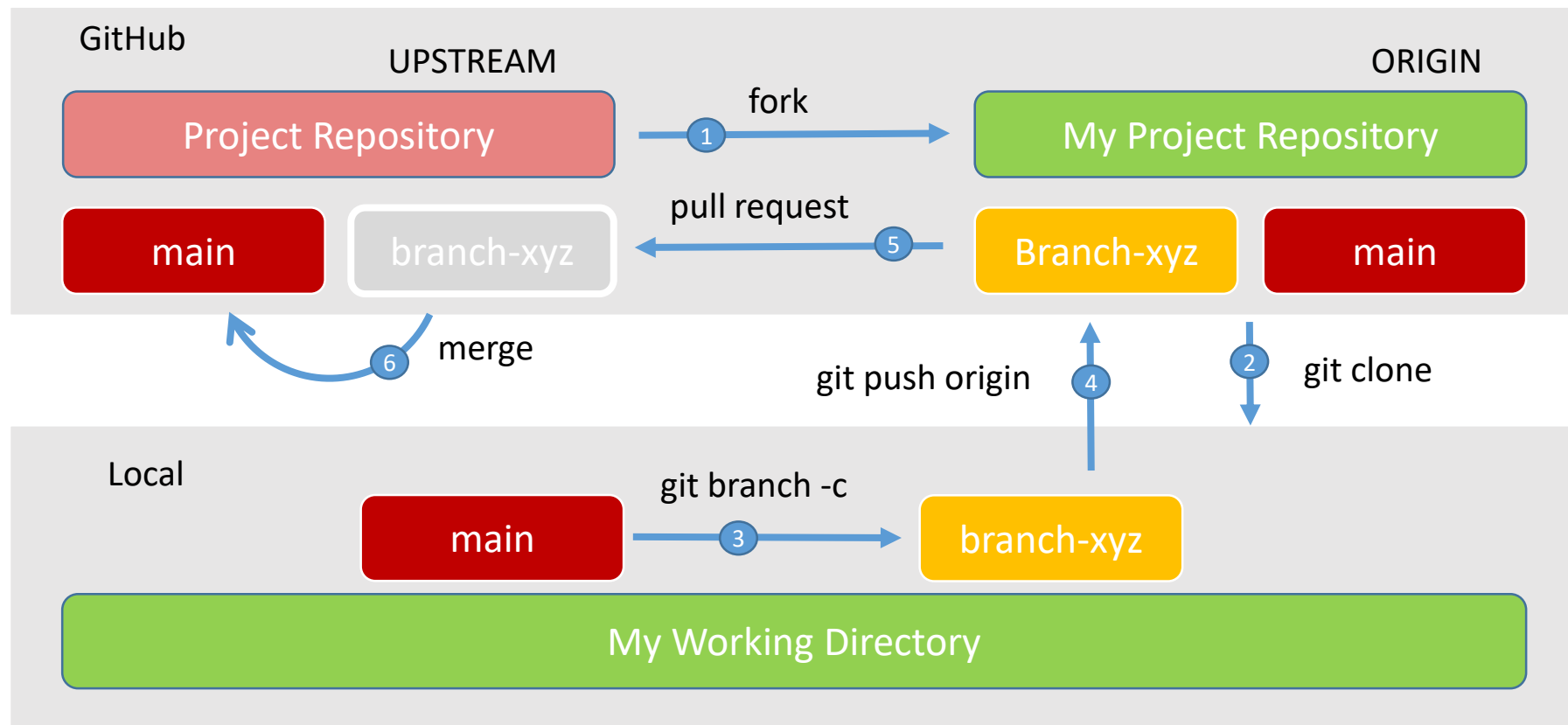
- **git remote**

Manage set of tracked repositories.

GitHub (Agile Workflow)

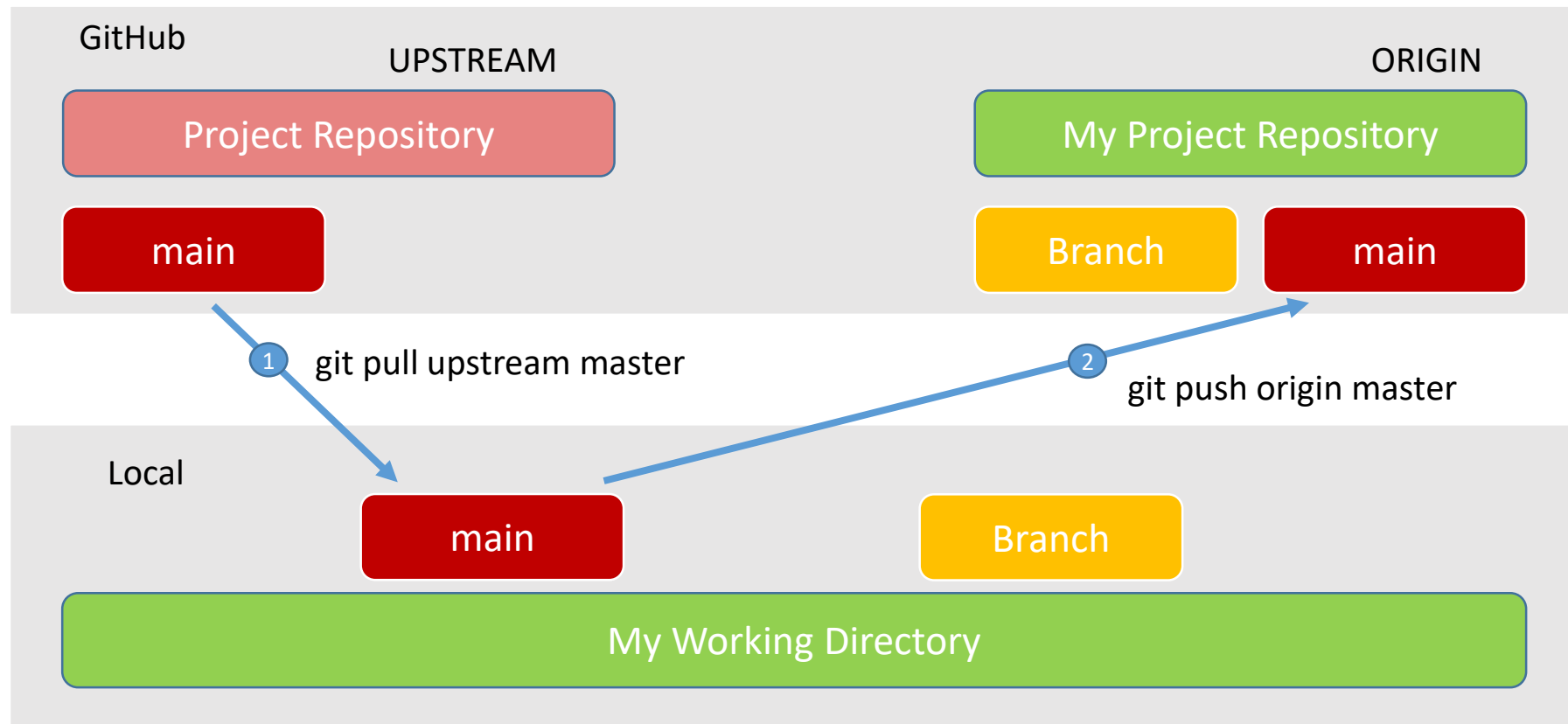
GitHub/Git Workflow: **Contribute to a project**

- Not rules. Guidelines and best practices developers should follow.



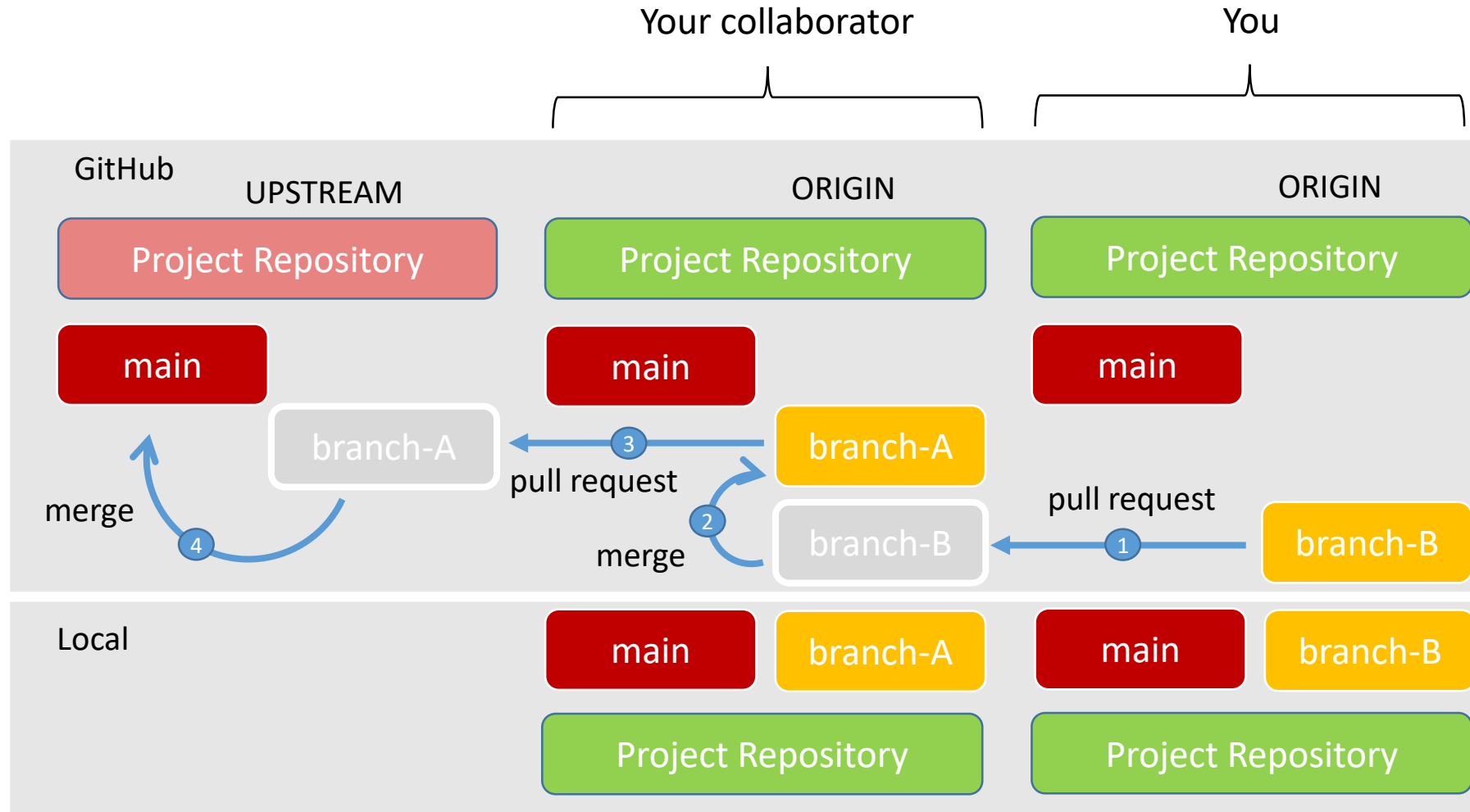
GitHub (Agile Workflow)

Actualize/sync your local repository



GitHub (Agile Workflow)

Collaborate



Feature Branch

