Version control with git

Alberto Tonda, Ph.D., Senior permanent researcher (DR)

UMR 518 Mathématiques et Informatique Appliquées - PS, INRAE, U. Paris-Saclay
UAR 3611 Institut des Systèmes Complexes Paris Ile-de-France, CNRS

alberto.tonda@inrae.fr

Another example of title for a slide

- This is some text
  - And some smaller text

INRAᴇ

TITLE OF THE PRESENTATION HERE (MODIFY IN VIEW -> MASK / AFFICHAGE -> MASQUE DE DIAPOSITIVES)
Alberto TONDA, Team EKINOCS, UMR 518 MIA-PS, INRAE, Université Paris-Saclay

alberto.tonda@inrae.fr

2

What is GIT?

- Version control software
  - Similar to Dropbox, Google Drive, and many others
  - Differences: open source, independent from environment
  - **GitHub.com** exists (Microsoft), local servers (Forge INRAE)
  - Author: Linus Torvalds (of Linux fame)
- "Git" is British slang for "stupid idiot", there is no official meaning for the acronym
- Computer Scientists think they are *hilarious*

TITLE OF THE PRESENTATION HERE (MODIFY IN VIEW -> MASK / AFFICHAGE -> MASQUE DE DIAPOSITIVES)
Alberto TONDA, Team EKINOCS, UMR 518 MIA-PS, INRAE, Université Paris-Saclay
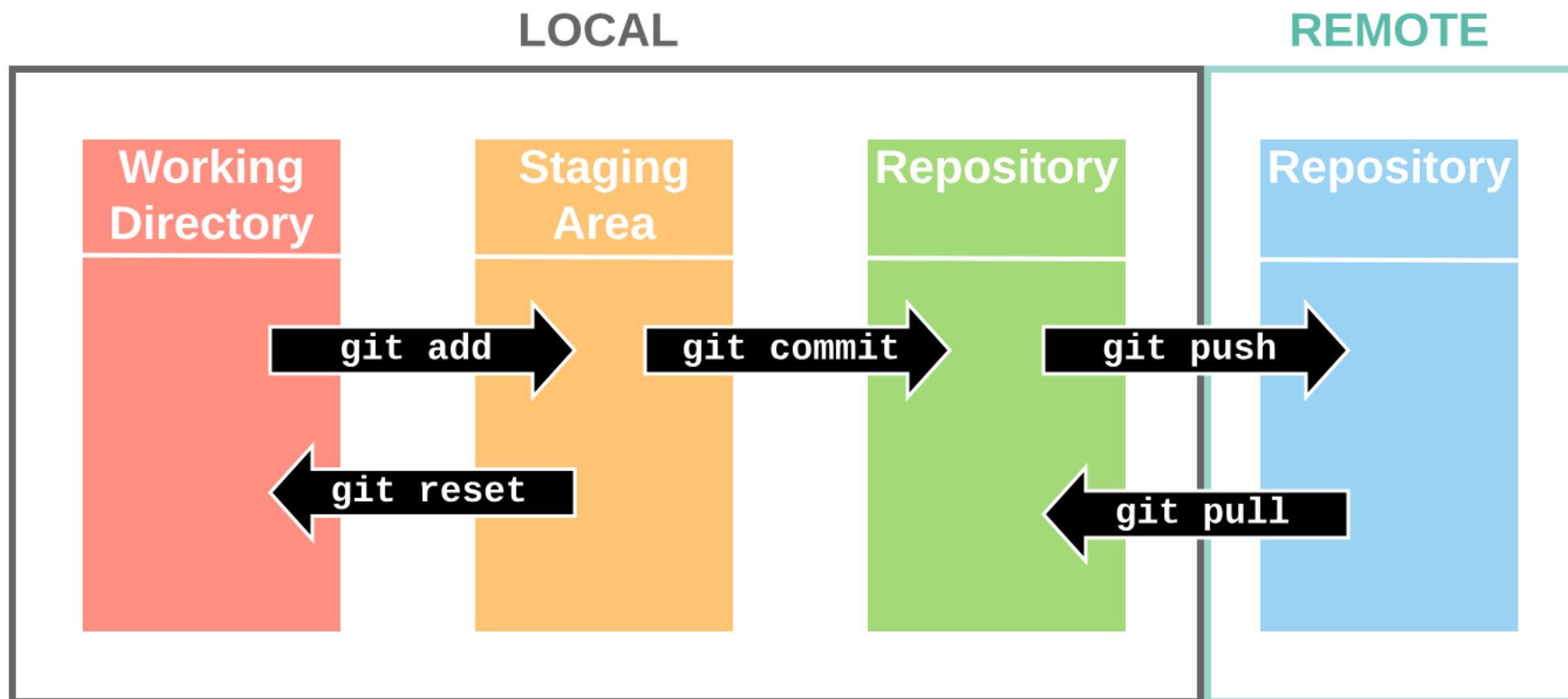
alberto.tonda@inrae.fr   3

Why should we care?

- Git is great for <u>collaborative development of code</u>
    - Other uses: personal storage, share code, **website**
    - Optimized for text files (and maybe a few images)
    - Git is fantastically integrated with a lot of other tools (Overleaf)
    - E.g. you can create your own Python package with Git!
    - Also, keep track of *just a few files* (different from DBX!)

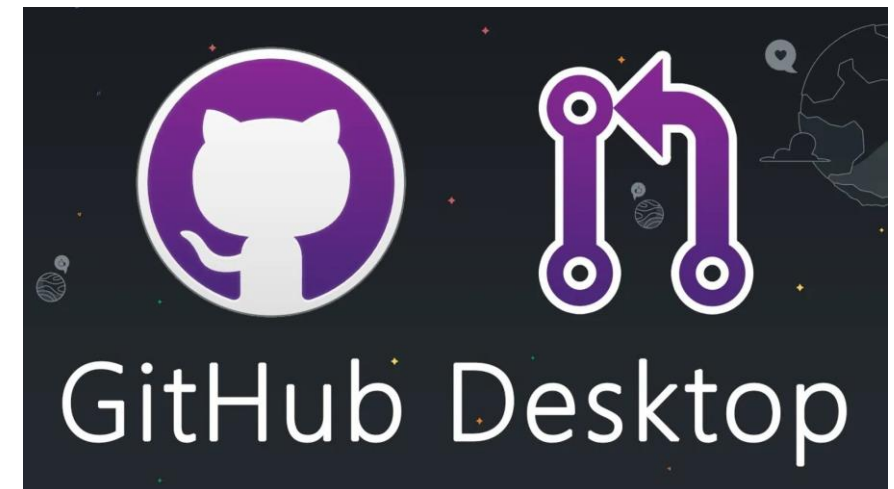- Show here example of git-based, GitHub hosted site

INRA℮

TITLE OF THE PRESENTATION HERE (MODIFY IN VIEW -> MASK / AFFICHAGE -> MASQUE DE DIAPOSITIVES)
Alberto TONDA, Team EKINOCS, UMR 518 MIA-PS, INRAE, Université Paris-Saclay

alberto.tonda@inrae.fr   4

How does it work?

INRAe
TITLE OF THE PRESENTATION HERE (MODIFY IN VIEW -> MASK / AFFICHAGE -> MASQUE DE DIAPOSITIVES)
Alberto TONDA, Team EKINOCS, UMR 518 MIA-PS, INRAE, Université Paris-Saclay

alberto.tonda@inrae.fr   5

GitHub Desktop

- Git was originally designed as a **command-line** tool
  - But now we have Graphical User Interface (GUI) tools
  - In fact, in CS <u>almost everything is command line hidden by GUI</u>

- Follow these steps
  1. Create an account on GitHub
  2. Create a new repository on GitHub
  3. Download GitHub desktop
  4. Generate public/private key pair
  5. Create local copy of your repository

TITLE OF THE PRESENTATION HERE (MODIFY IN VIEW -> MASK / AFFICHAGE -> MASQUE DE DIAPOSITIVES)
Alberto TONDA, Team EKINOCS, UMR 518 MIA-PS, INRAE, Université Paris-Saclay

alberto.tonda@inrae.fr   6

More git fun!

- Create two new files
  - "added.txt" and "ignored.txt"
  - GitHub Desktop automatically adds local files to **commits**
  - You can manually tell it to ignore files
  - Add "added.txt" and ignore "ignored.txt"
  - **commit** the changes, then **push** them
  - Check the repository page on GitHub.com ; did it change?
- Commit messages
  - Useful to keep track of what happened, **don't leave them empty**

INRAe

TITLE OF THE PRESENTATION HERE (MODIFY IN VIEW -> MASK / AFFICHAGE -> MASQUE DE DIAPOSITIVES)
Alberto TONDA, Team EKINOCS, UMR 518 MIA-PS, INRAE, Université Paris-Saclay

alberto.tonda@inrae.fr   7

Advanced use: git merge

- What happens if you modify local files with git...
  - ...and someone else in the meantime commits/push?
  - The remote repository and your local files <u>differ greatly</u>!
  - It becomes necessary to reconciliate the two versions

- Let's try it out *live*!

INRA℮

TITLE OF THE PRESENTATION HERE (MODIFY IN VIEW -> MASK / AFFICHAGE -> MASQUE DE DIAPOSITIVES)
Alberto TONDA, Team EKINOCS, UMR 518 MIA-PS, INRAE, Université Paris-Saclay

alberto.tonda@inrae.fr   8
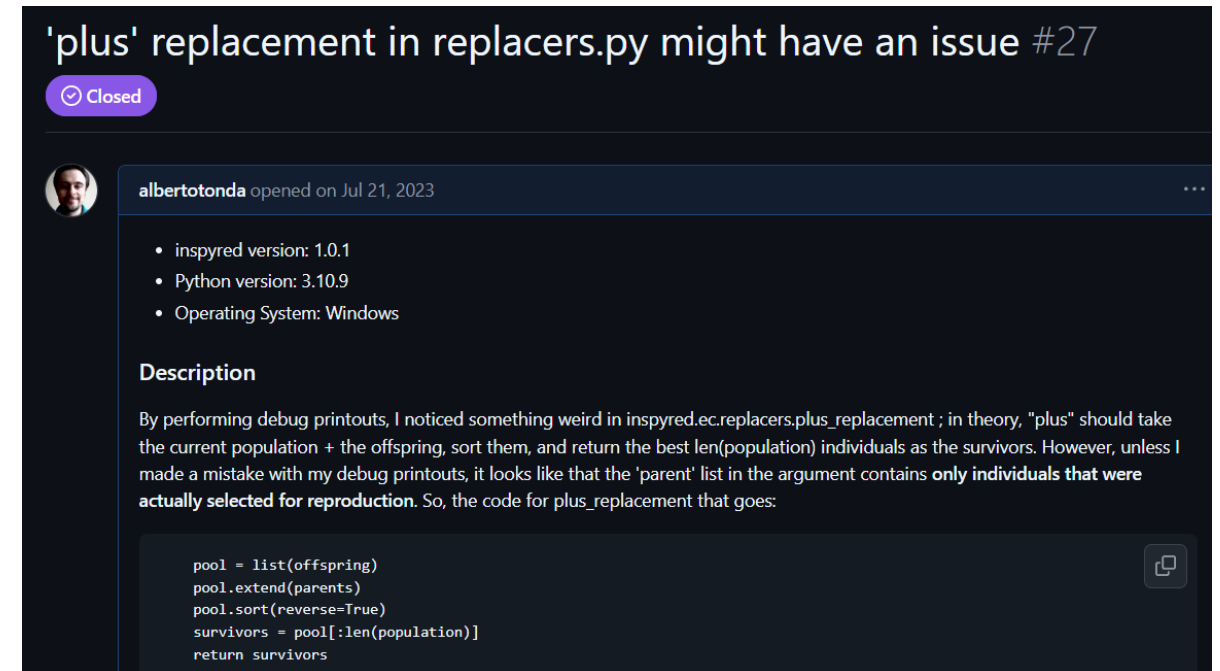
Advanced use: branches

- You can create different "branches" on the same repository
  - Work independently on separate parts without merging every time
  - At the end of development, big merge between branches
  - `main` is the default branch

- Let's try to create two different branches, then
  - Merge the first branch with `main`
  - Then merge the second branch also with `main`

Advanced use: recover previous versions

INRAE
TITLE OF THE PRESENTATION HERE (MODIFY IN VIEW -> MASK / AFFICHAGE -> MASQUE DE DIAPOSITIVES)
Alberto TONDA, Team EKINOCS, UMR 518 MIA-PS, INRAE, Université Paris-Saclay

alberto.tonda@inrae.fr   10

Advanced use: issues

- This is more of a GitHub use than git
    - If you are using a package and you identify a problem
    - You can **open an issue** on their repository page

INRAe

TITLE OF THE PRESENTATION HERE (MODIFY IN VIEW -> MASK / AFFICHAGE -> MASQUE DE DIAPOSITIVES)
Alberto TONDA, Team EKINOCS, UMR 518 MIA-PS, INRAE, Université Paris-Saclay

alberto.tonda@inrae.fr   11

Advanced use: pull requests

- Ask for permission: "Can I merge these changes?"
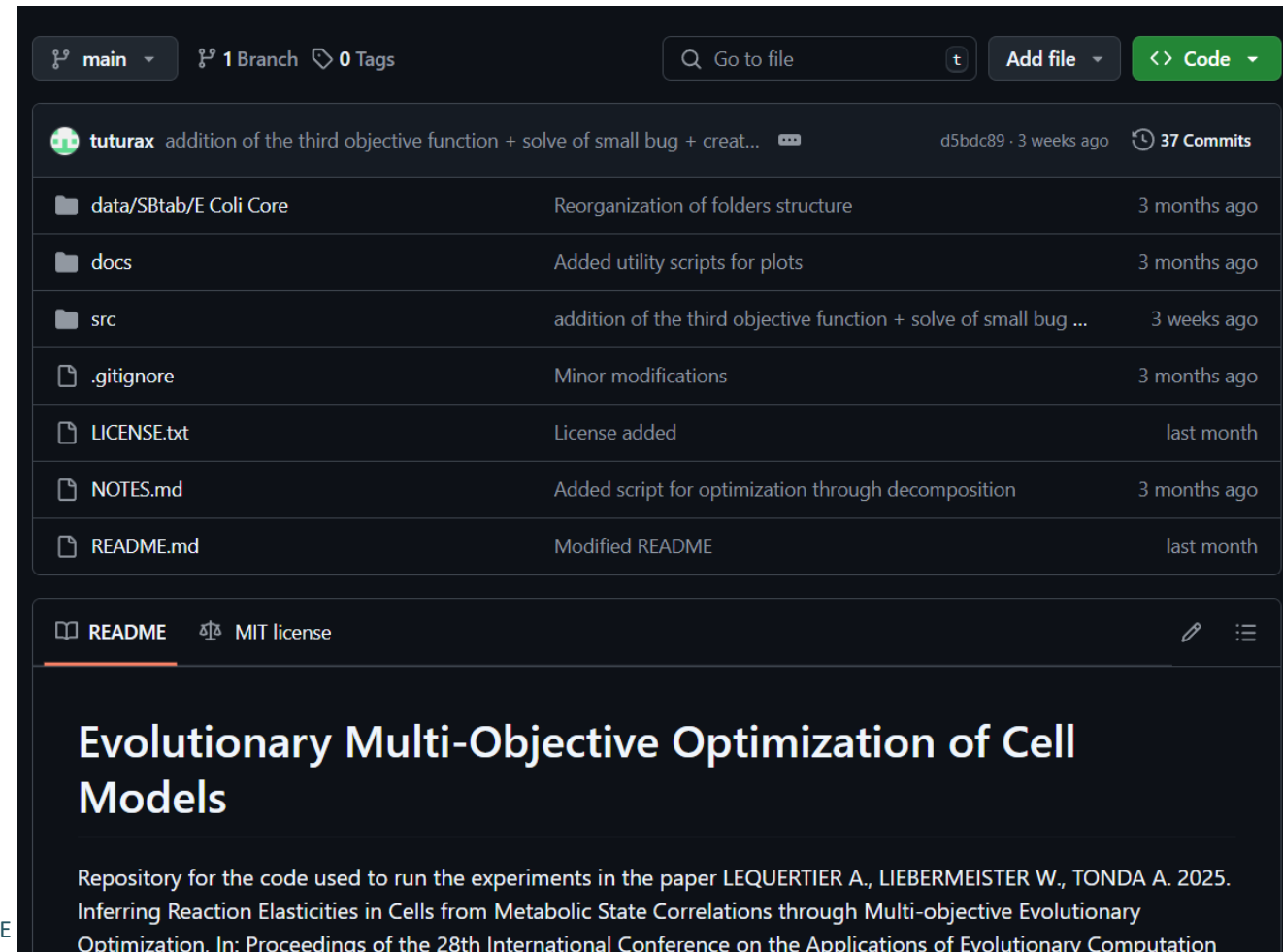 - Require review from repo owners
  - I used this feature **twice**

TITLE OF THE PRESENTATION HERE (MODIFY IN VIEW -> MASK / AFFICHAGE -> MASQUE DE DIAPOSITIVES)
Alberto TONDA, Team EKINOCS, UMR 518 MIA-PS, INRAE, Université Paris-Saclay

alberto.tonda@inrae.fr   12

Practical advice: structure your repository

- Create **subfolders** in your repository
  - **data/** for all data files
  - **docs/** for papers and stuff
  - **src/** for code
- Root usually contains
  - .gitignore
  - README.md (markdown)
  - LICENSE.txt (optional)
  - Other stuff

Questions?

alberto.tonda@inrae.fr