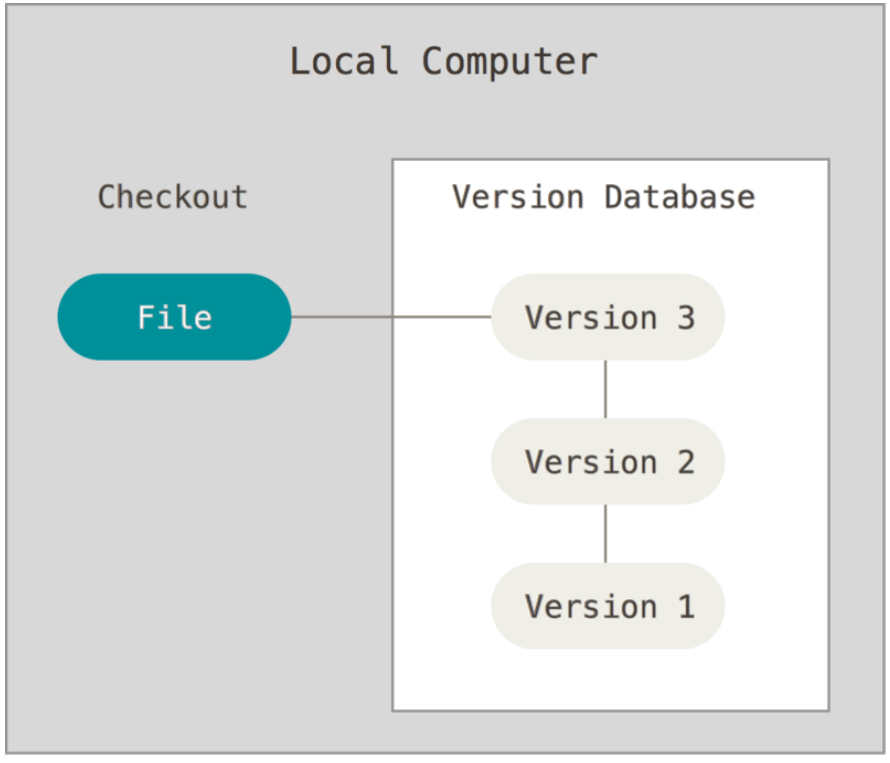
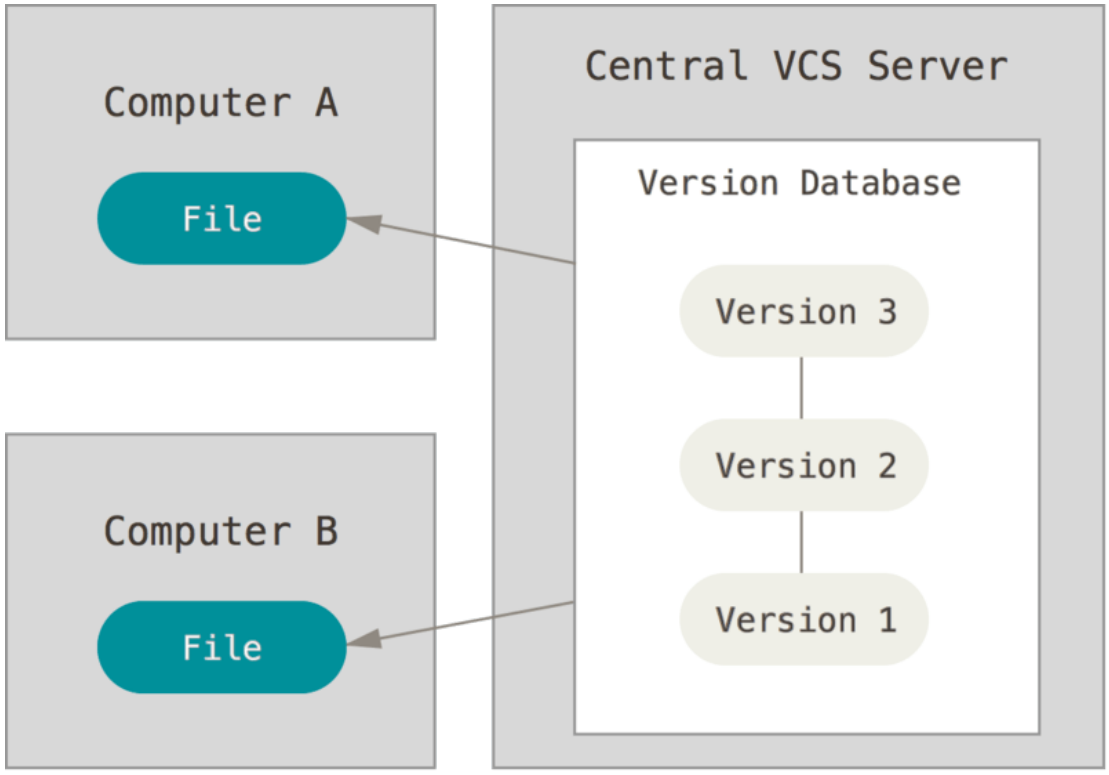
Version Control with Git

What is version control?

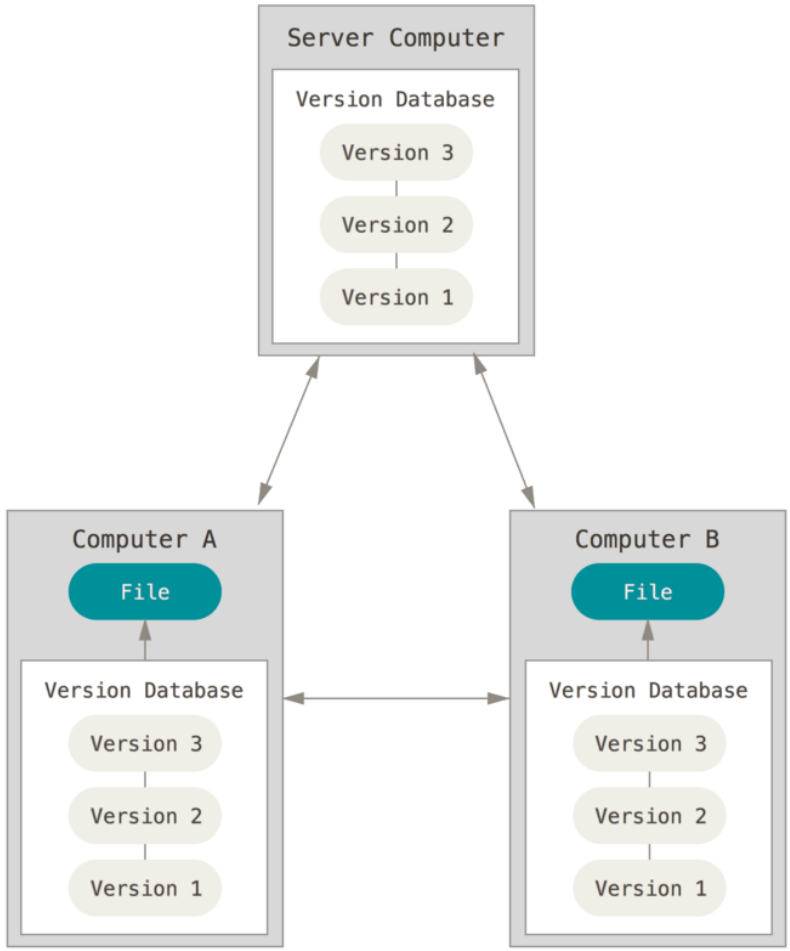


Centralized version control systems



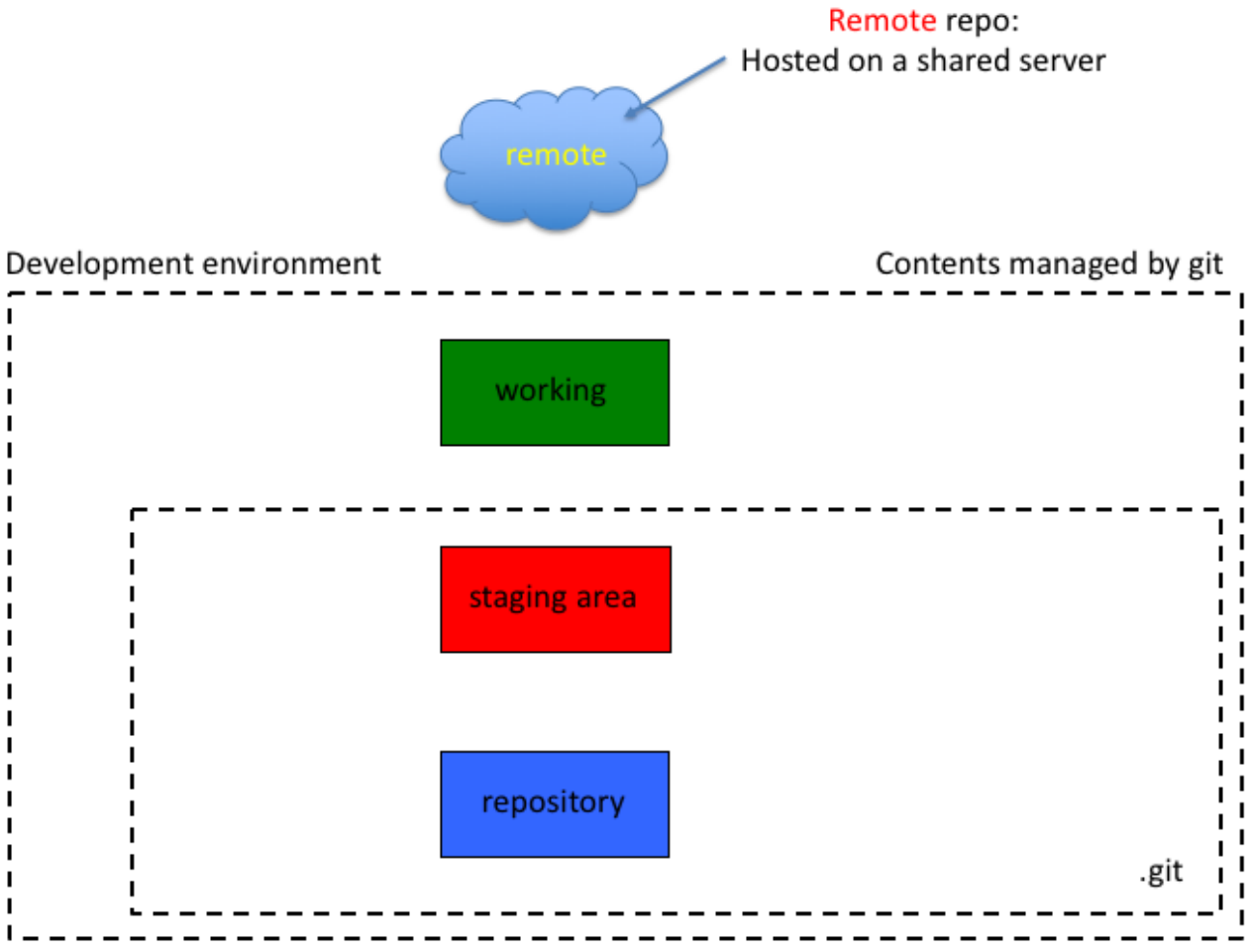
Example: SCCS, RCS, CVS, SVN

Distributed version control systems

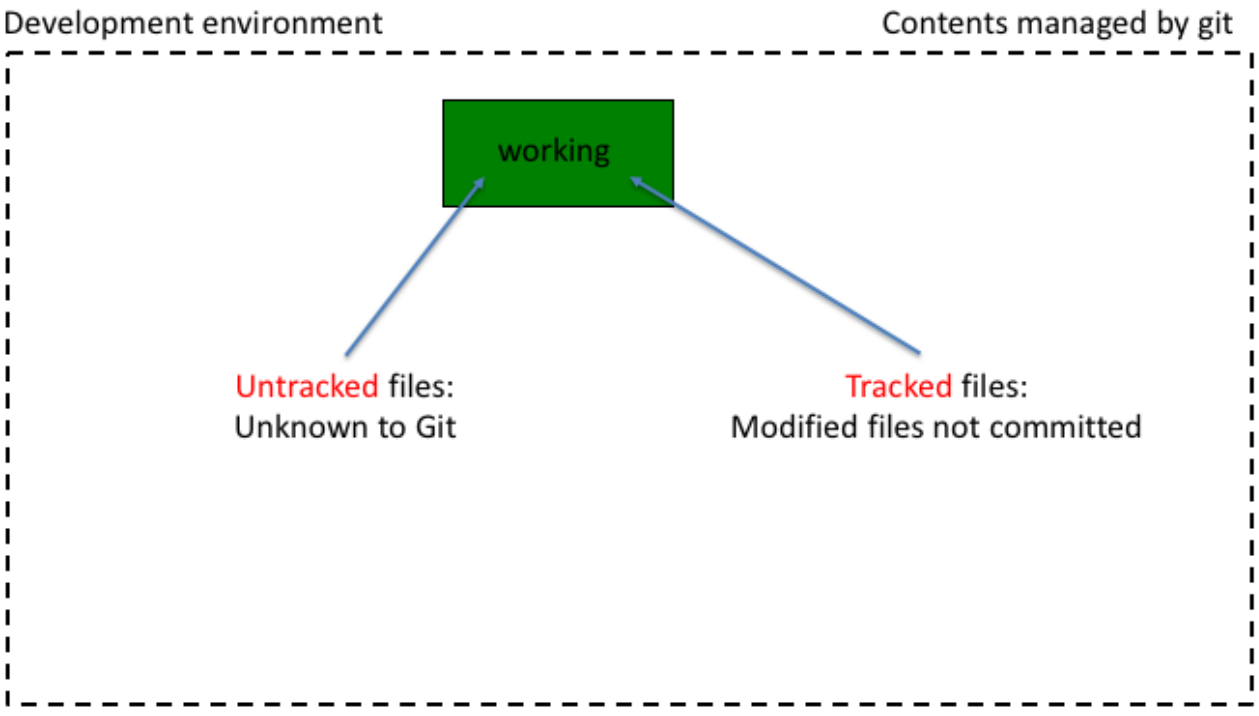


Example: Git, Mercurial

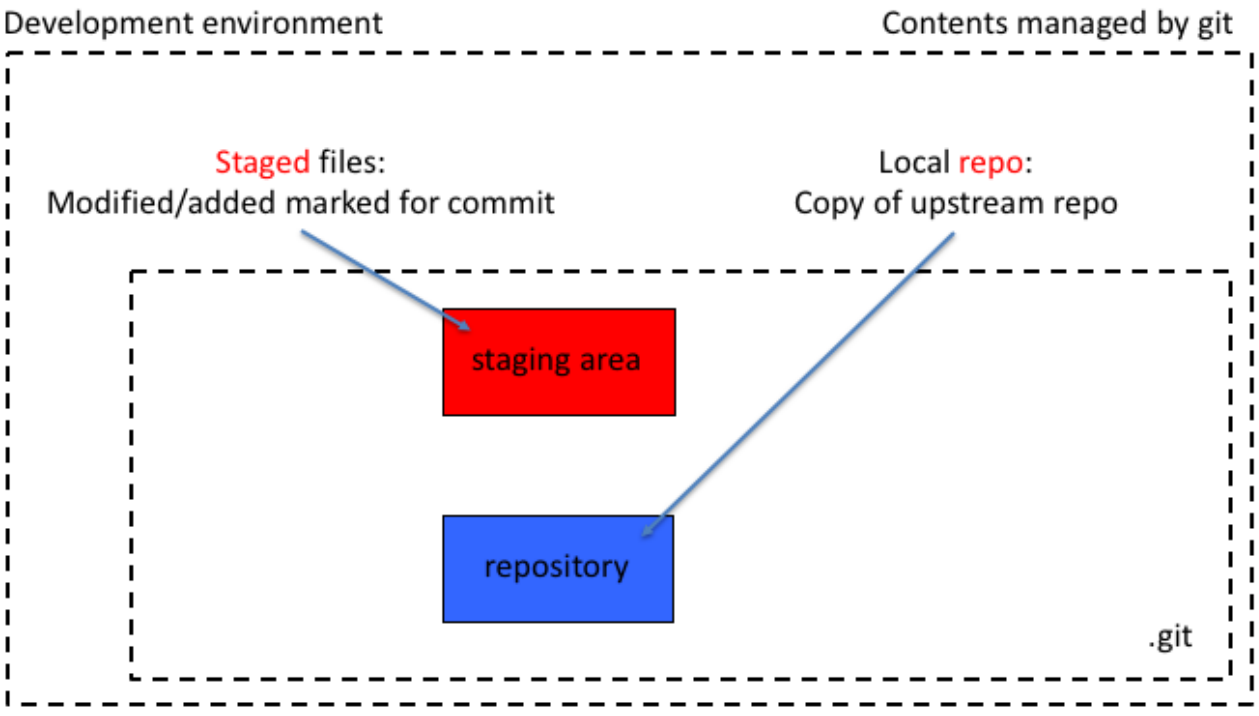
Git Concepts



Git Concepts



Git Concepts



Getting started

- Configuration

  $ git config –global user.name "Your Name"

  $ git config –global user.email you@yourdomain.example.com

  $ git config –global core.editor "vim"

- Starting a new project

  $ git init

  $ git add .

  $ git commit -m"message" file1.py

- Editing files

  $ vim file2.py file1.py

  $ git status

  $ git diff file1.py

  $ git add -A # place file(s) in staging area

  $ git diff –cached

  $ git commit # record changes to repo

- Reviewing history

  $ git log

  $ git log –stat –summary

  $ gitk # History GUI

- Undoing changes

  $ vim file2.py

  $ git reset file2.py # restore file

  $ git checkout # restore file

- Removing files

  $ git rm file2.py

  $ git mv file1.py file3.py

  $ git add file3.py # always stage file before commit

  $ git commit

Exercise 1

1. create a new directory, open it and perform a “git init“ to create a new git repository.
2. run “git config –global [options]“ to set up user configurations
3. create a README file and run “git add“ and “git commit“ command
4. edit the file and run “git diff“
5. run “git commit -a“ command
6. create a Python file, calc.py, contains a function that computes the sum of 2 numbers. Commit file to your repo.
7. type “git log –oneline –graph –decorate“ What do you see?

Branching and collaborating

- Branching

  $ git branch

  $ git branch my_exp

  $ git checkout my_exp

  $ vim calc.py

  $ git commit -a

  $ git checkout master

  $ git merge my_exp

  $ git branch -d my_exp

  $ git branch -D my_exp

- Collaborating

  Bob begins with:

  $ cd /home/bob

  $ git clone /home/alice/a_project b_project

  (edit and commit files in b_project)

  Bob notifies Alice he's done.

  Meanwhile Alice:

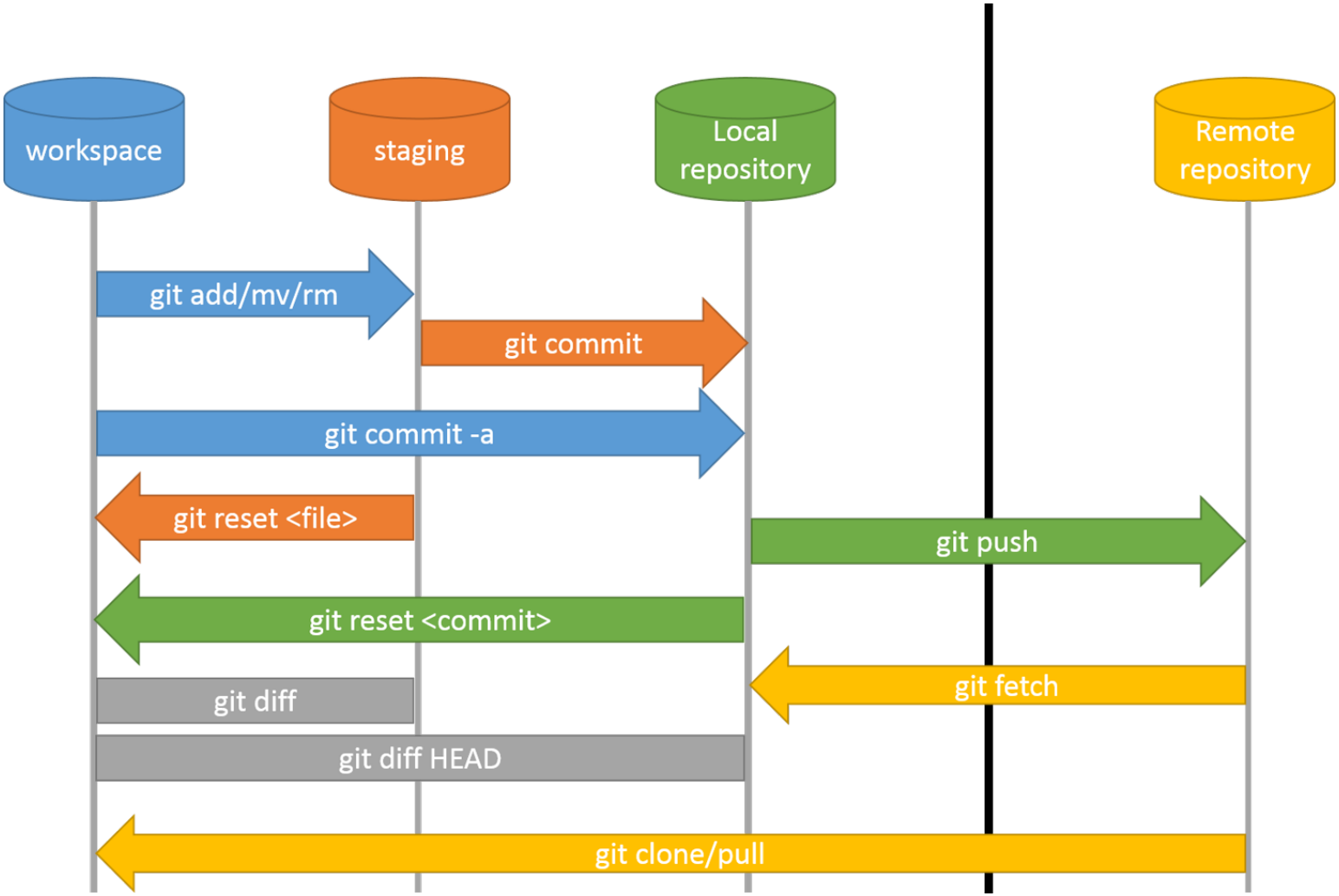
  $ cd /home/alice/a_project

  $ git pull /home/bob/b_project master

Exercise 2

- create a new branch “testing“.
- switch to that branch.
- check what branch you are in using “git branch“.
- create a python script that prints the first 10 integers, and commit it.
- look at the history of your repository.
- switch to the branch “master“, and look again at the history. What do you see?
- merge “testing“ into “master“, and look again at the history. What do you see?

Git workflow



Resources

- Git Community Book
- Pro Git
- A Visual Git Reference
- Git Cheat Sheet