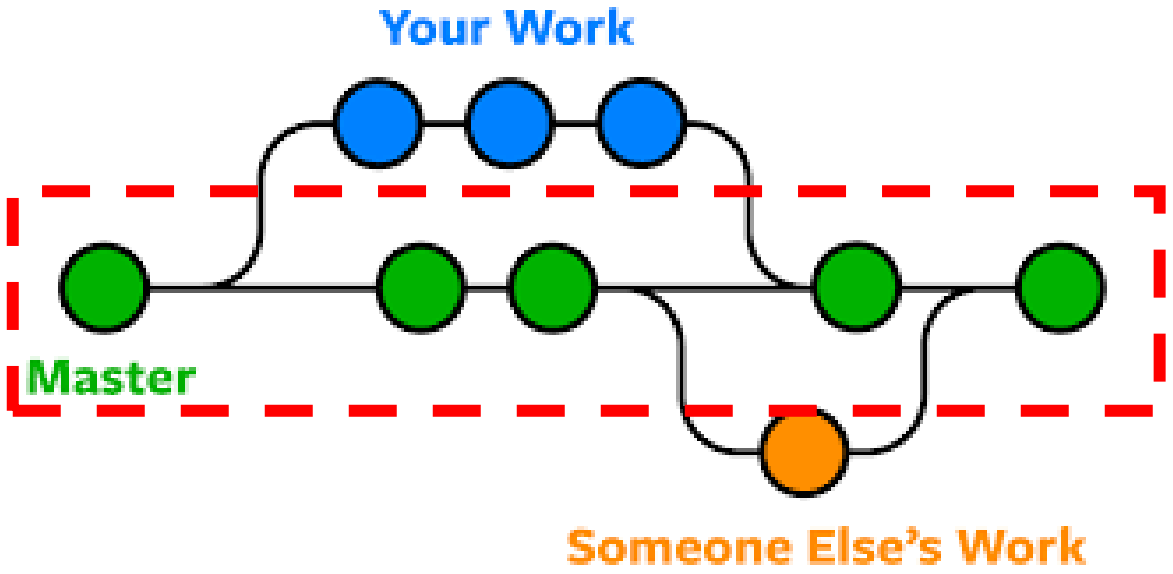
Git Fundamentals
Day 3:
Commits

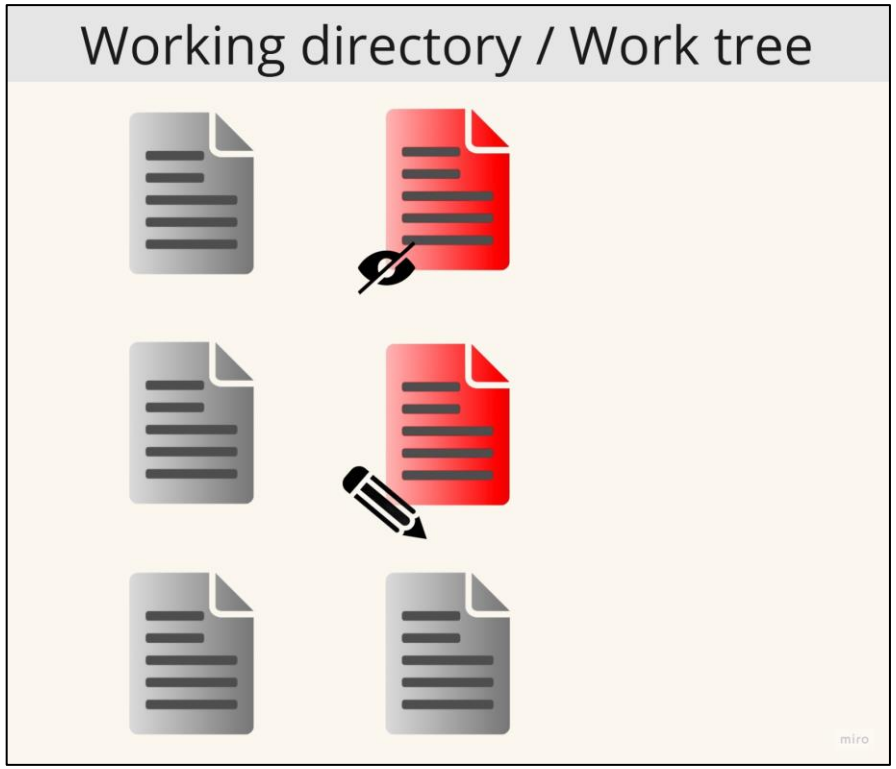
Today:

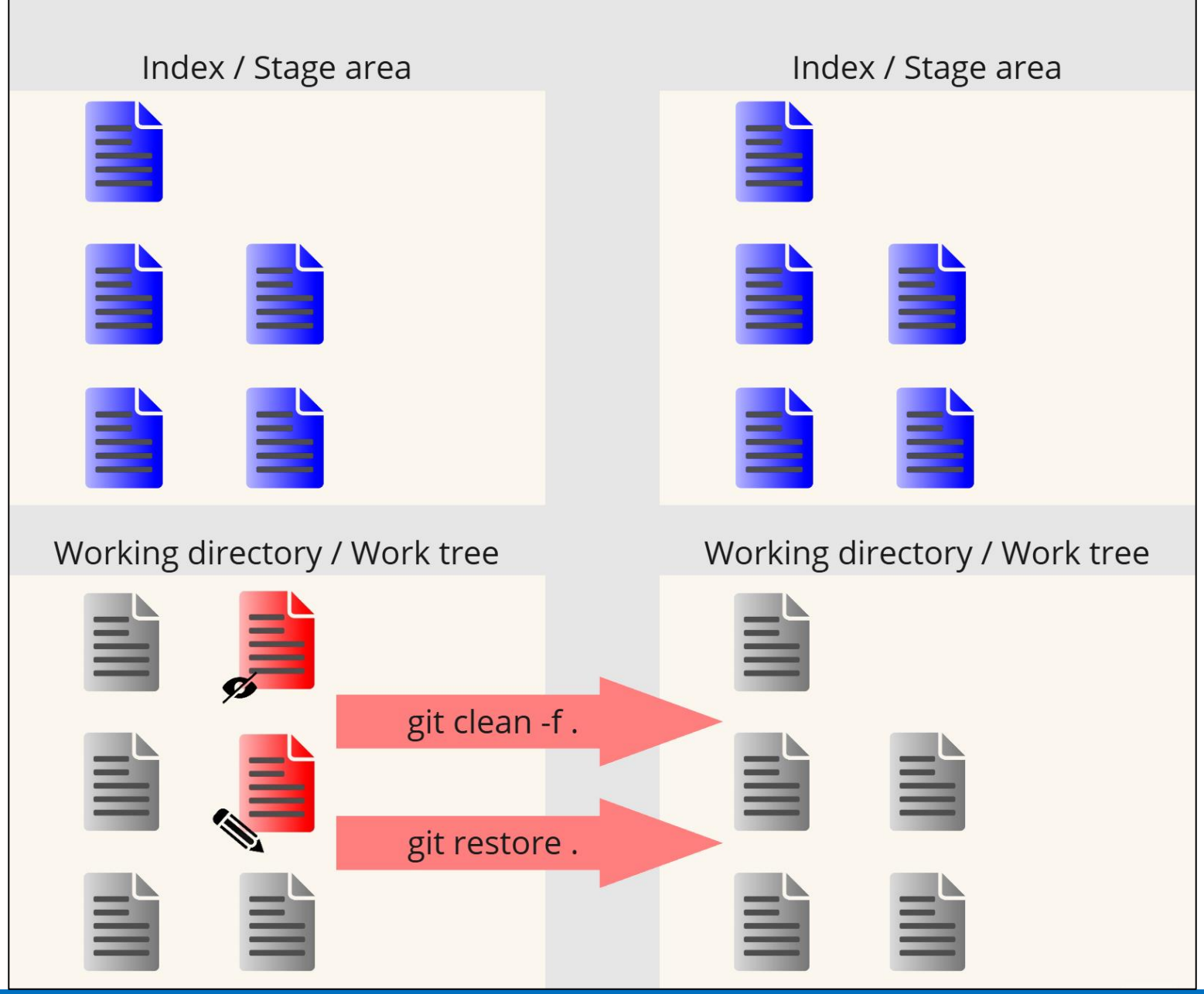
- ✓ Day 2 Recap
- ✓ Commit guidelines
- ✓ Navigating and comparing commits
- ✓ Ignoring files



Recap: Work tree

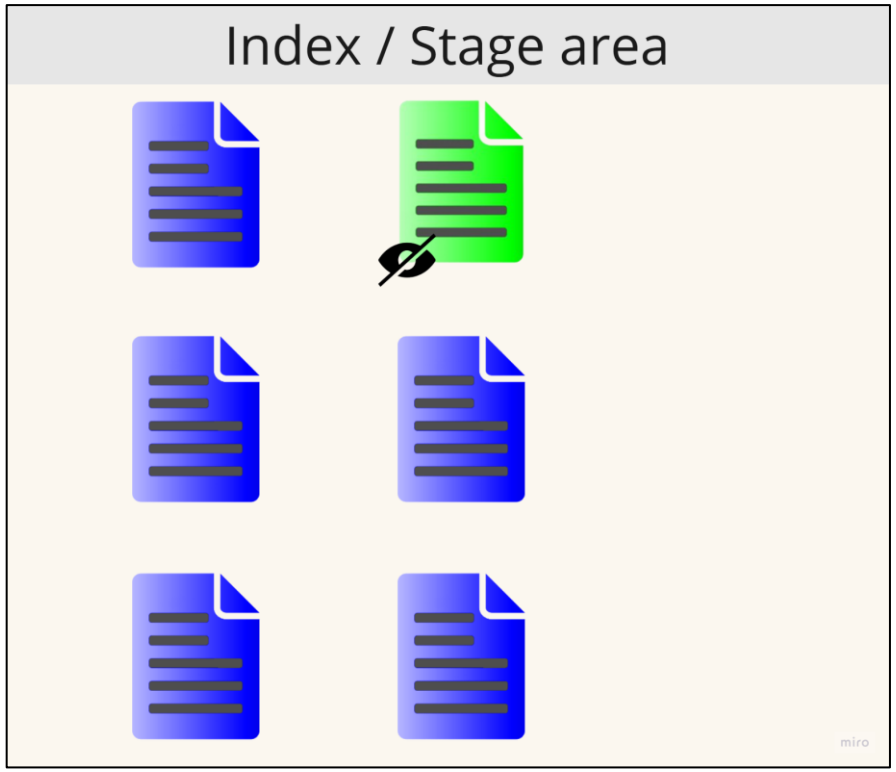
- ✓ Create new files with echo or touch
- ✓ Edit existent files (we use code)
- ✓ Restore work directory to a previous commit

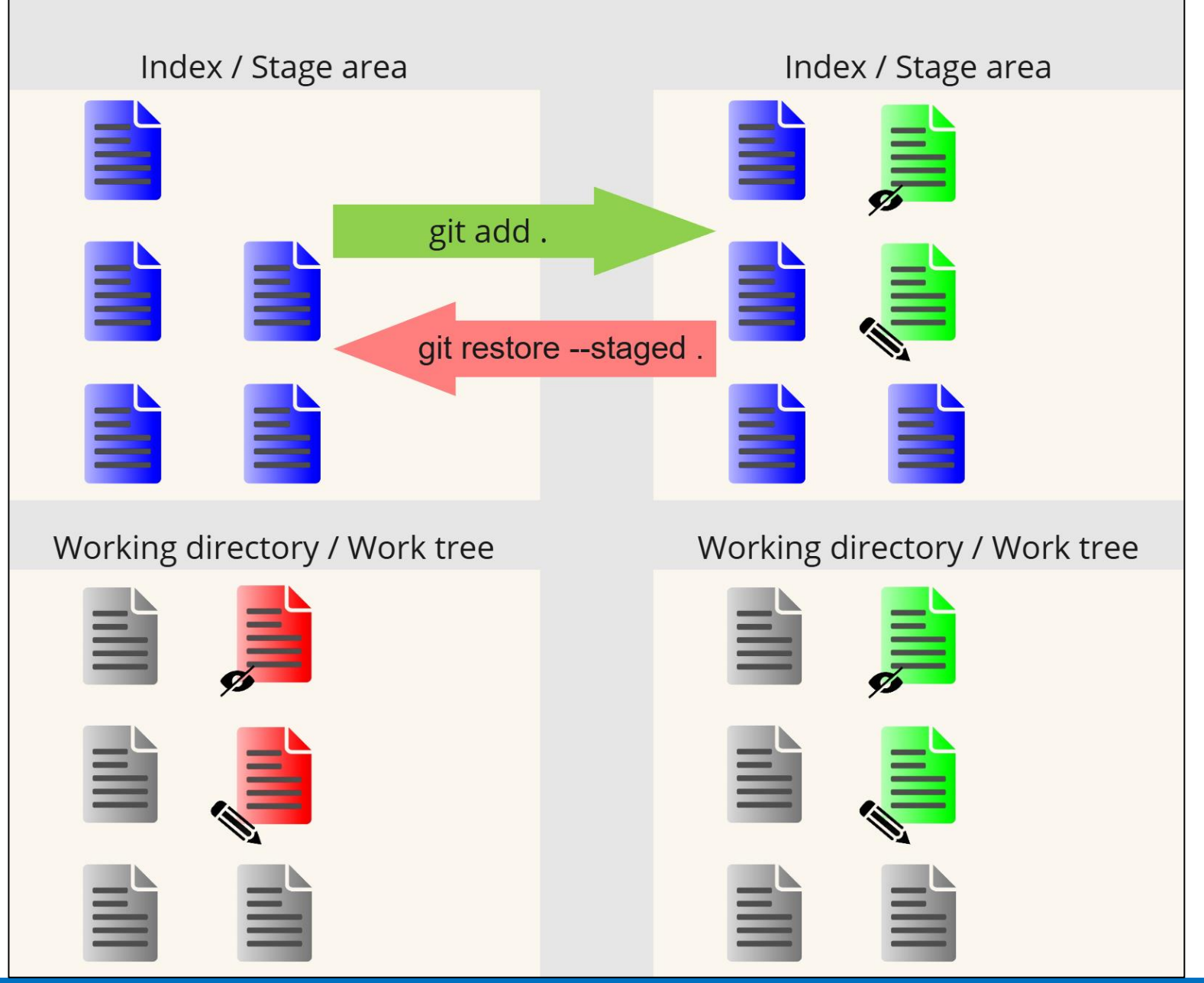


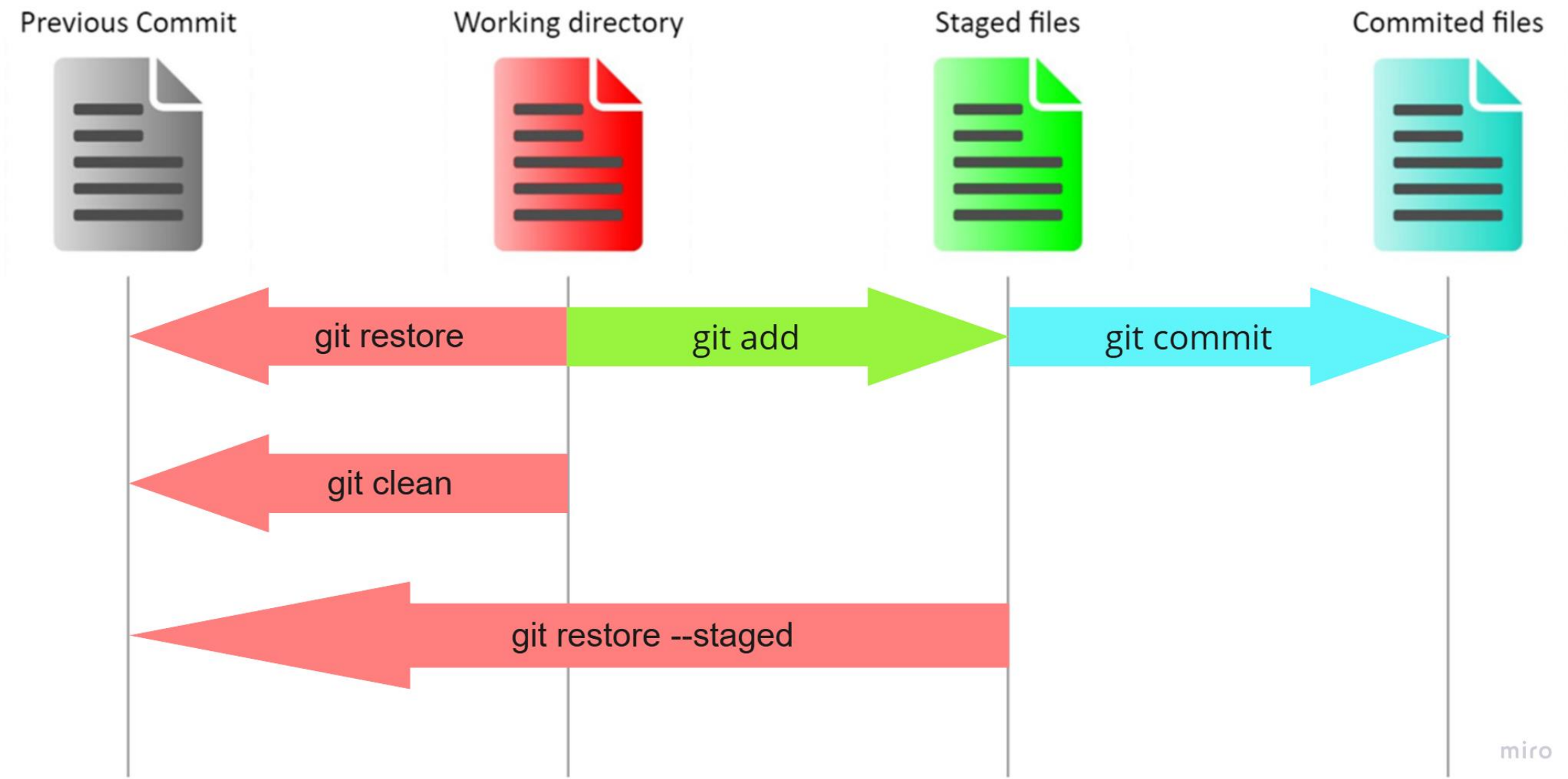

Index / Stage area
Index / Stage area
Working directory / Work tree
Working directory / Work tree
git clean -f .
git restore .

Recap: Index

- ✓ Preview of a selection of changes we want to keep
- ✓ Add or discard changes
- ✓ Discard = restore to previous commit state

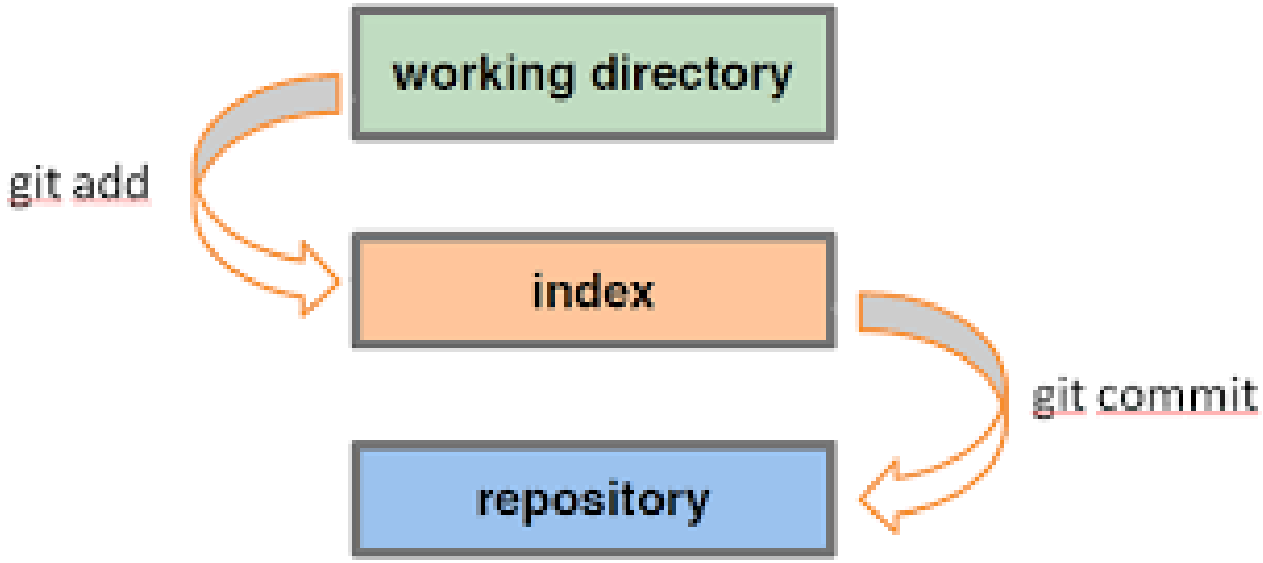



Index / Stage area
Index / Stage area
git add .
git restore --staged .
Working directory / Work tree
Working directory / Work tree


Previous Commit
Working directory
Staged files
Commited files
git restore
git add
git commit
git clean
git restore --staged
miro

Recap: Commits

- Snapshots of changes in a repository
- They include:
  - ✓ Username and email
  - ✓ Date and Commit ID
  - ✓ Message

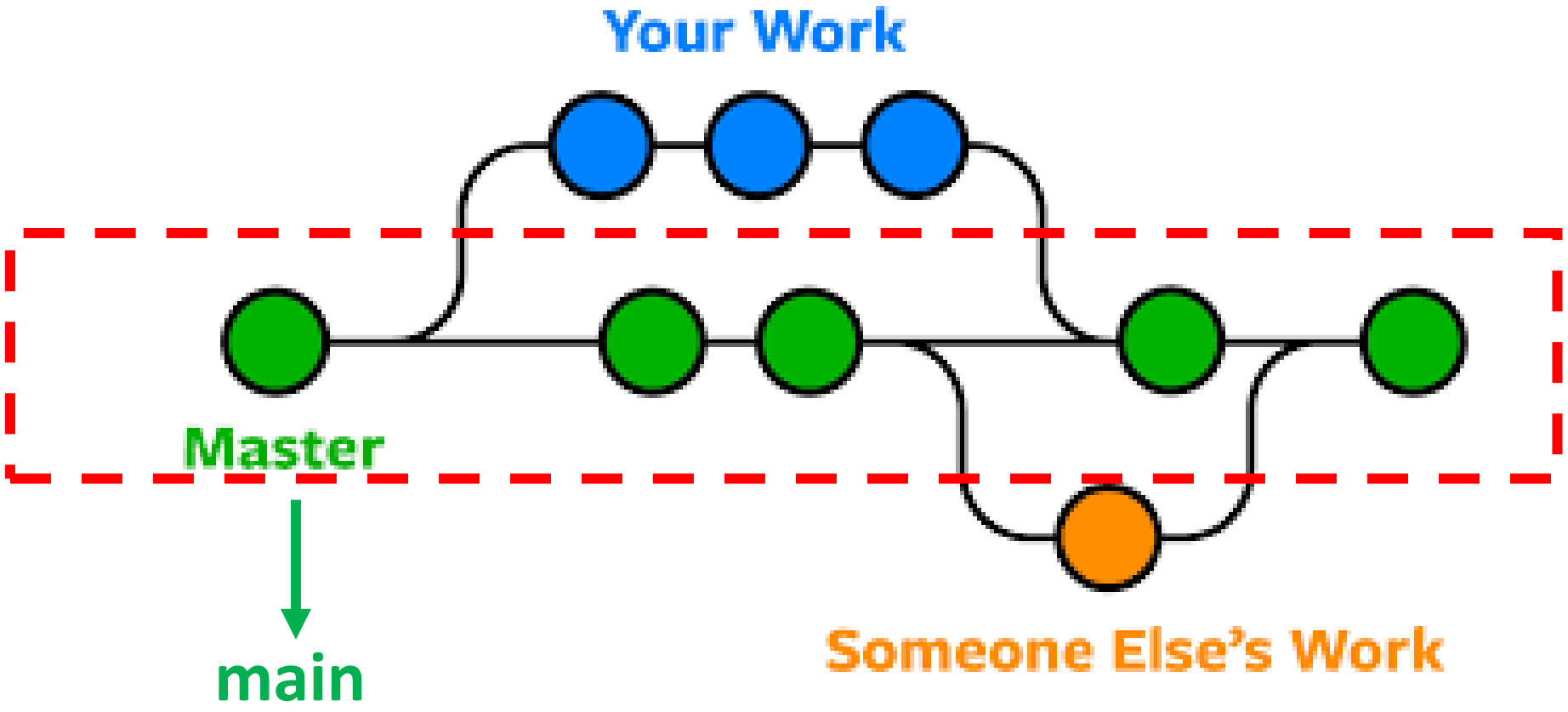


Commits: Guidelines

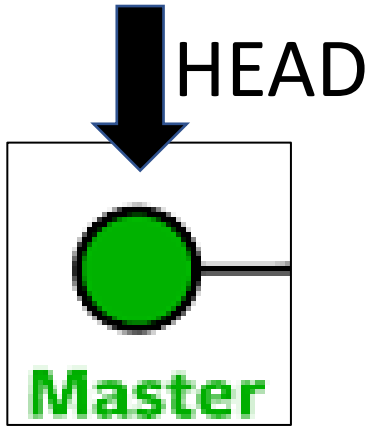
Here are some IMPORTANT guidelines for a good commit message:

- Write your commit message subject in the imperative: "Fix bug" and not "Fixed bug" or "Fixes bug."
- Do not end the subject line with a period
- Separate subject from body with a blank line
- Capitalize the subject line and each paragraph
- Wrap lines at 72 characters
- Use the body to explain what and why you have done something

Note: Single timeline (for now)

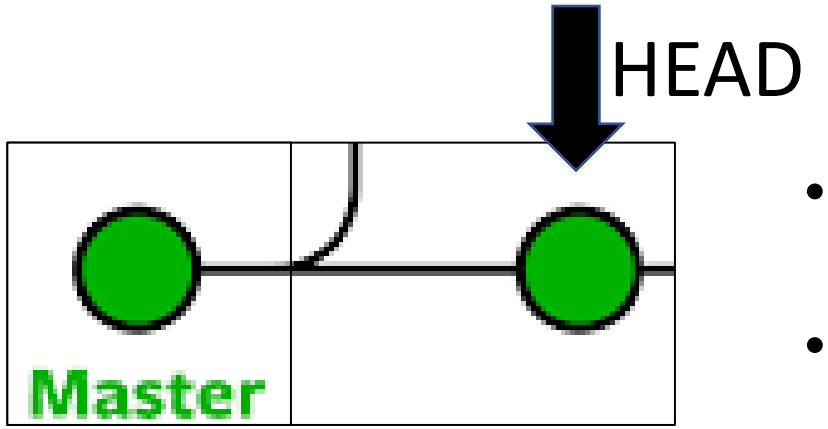


Time traveling: HEAD



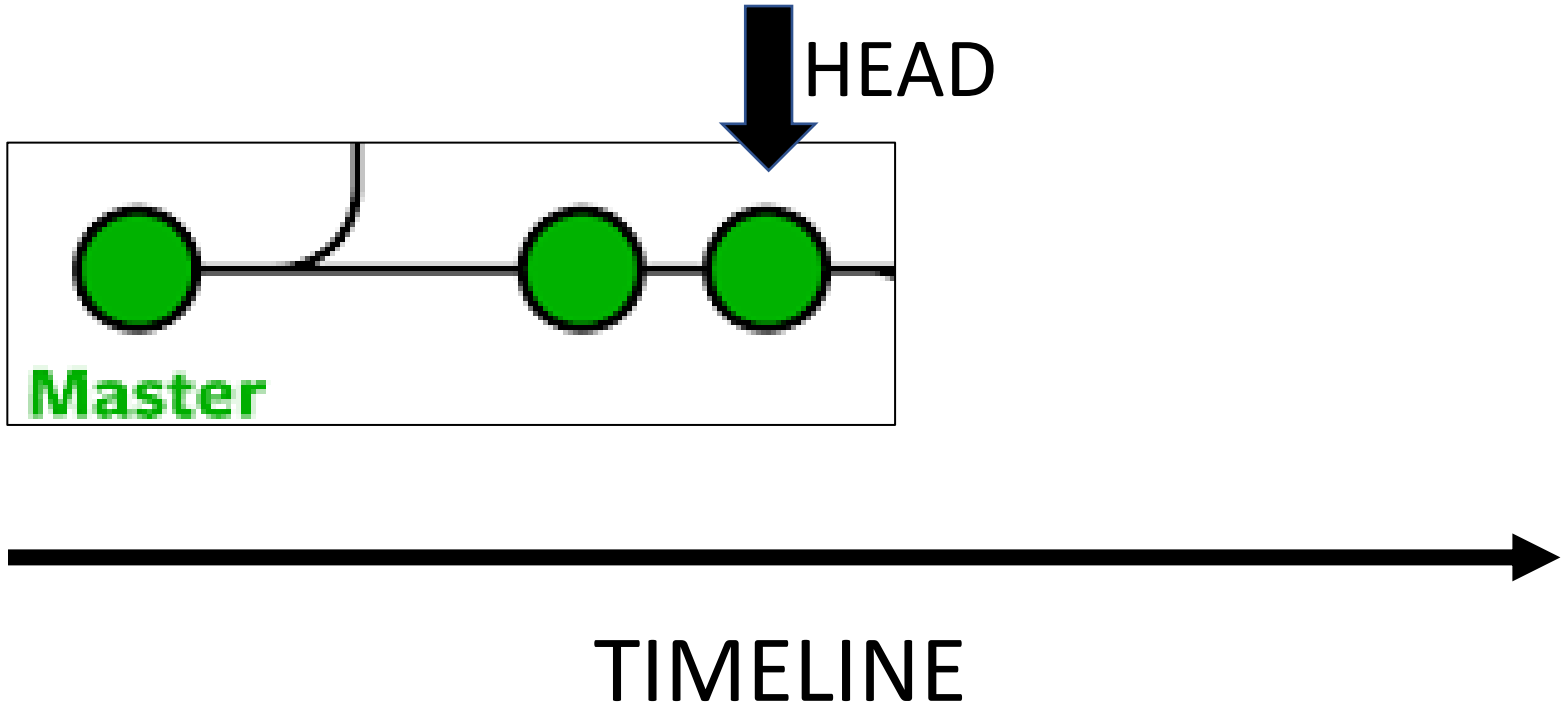
- A reference to a commit
- Commit we are pointing to

Time traveling : HEAD

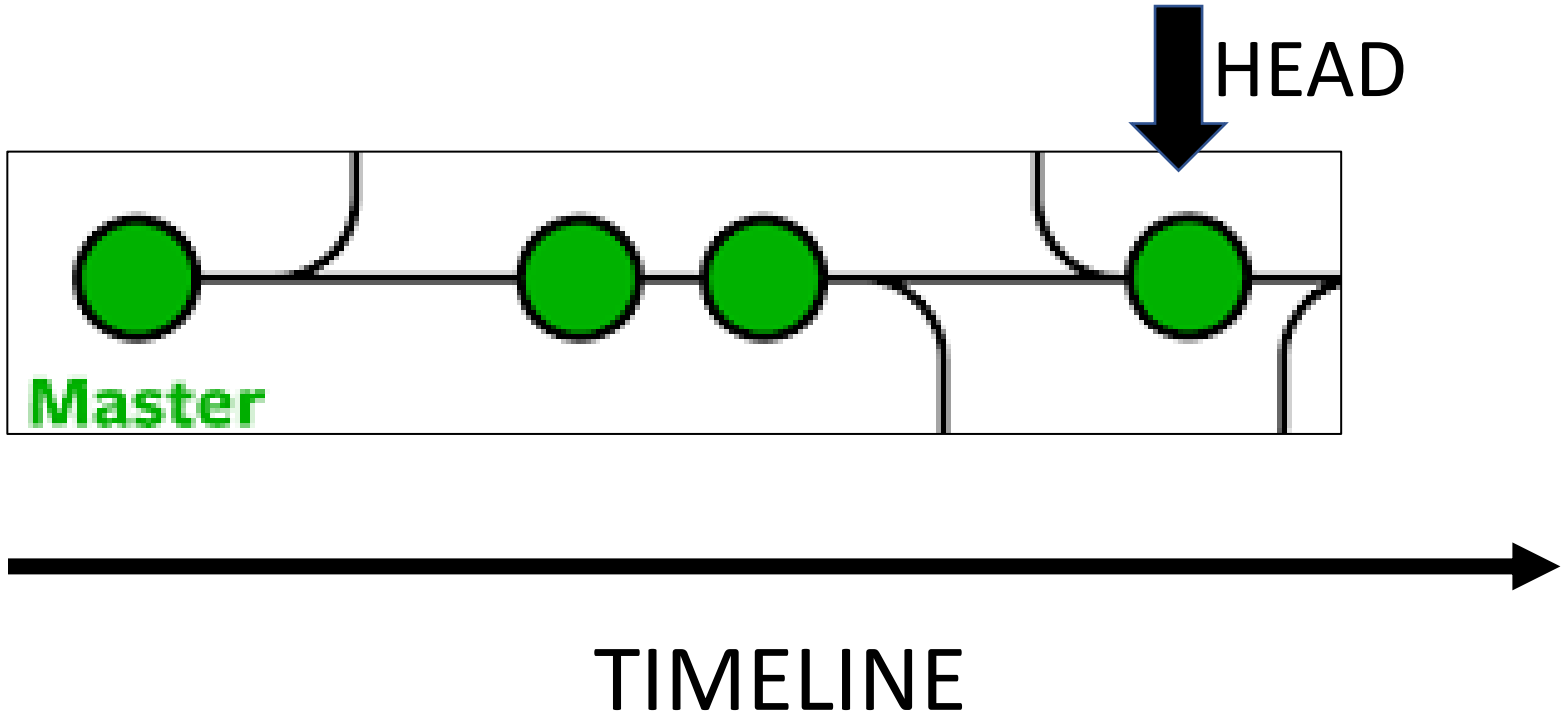


- Follows us as we commit
- Usually in the tip of the branch

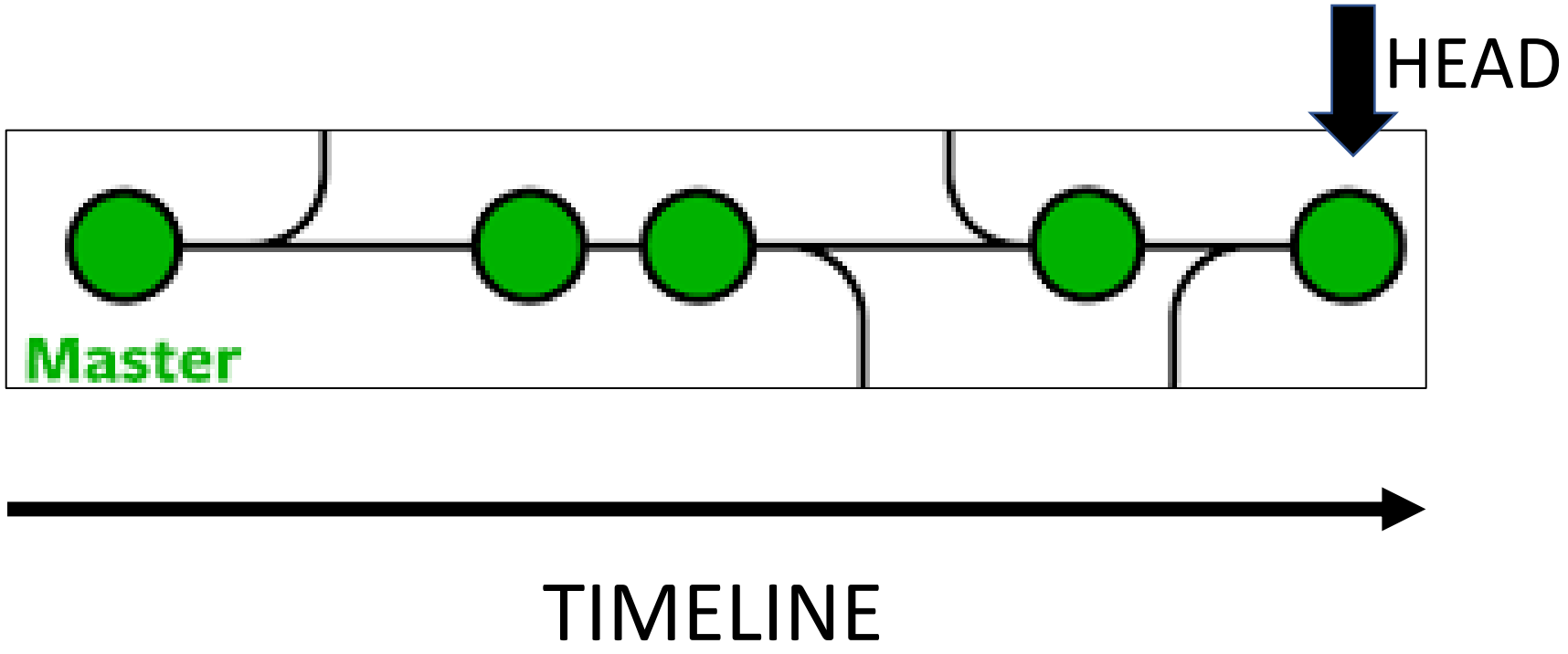
Time traveling : HEAD



Time traveling : HEAD

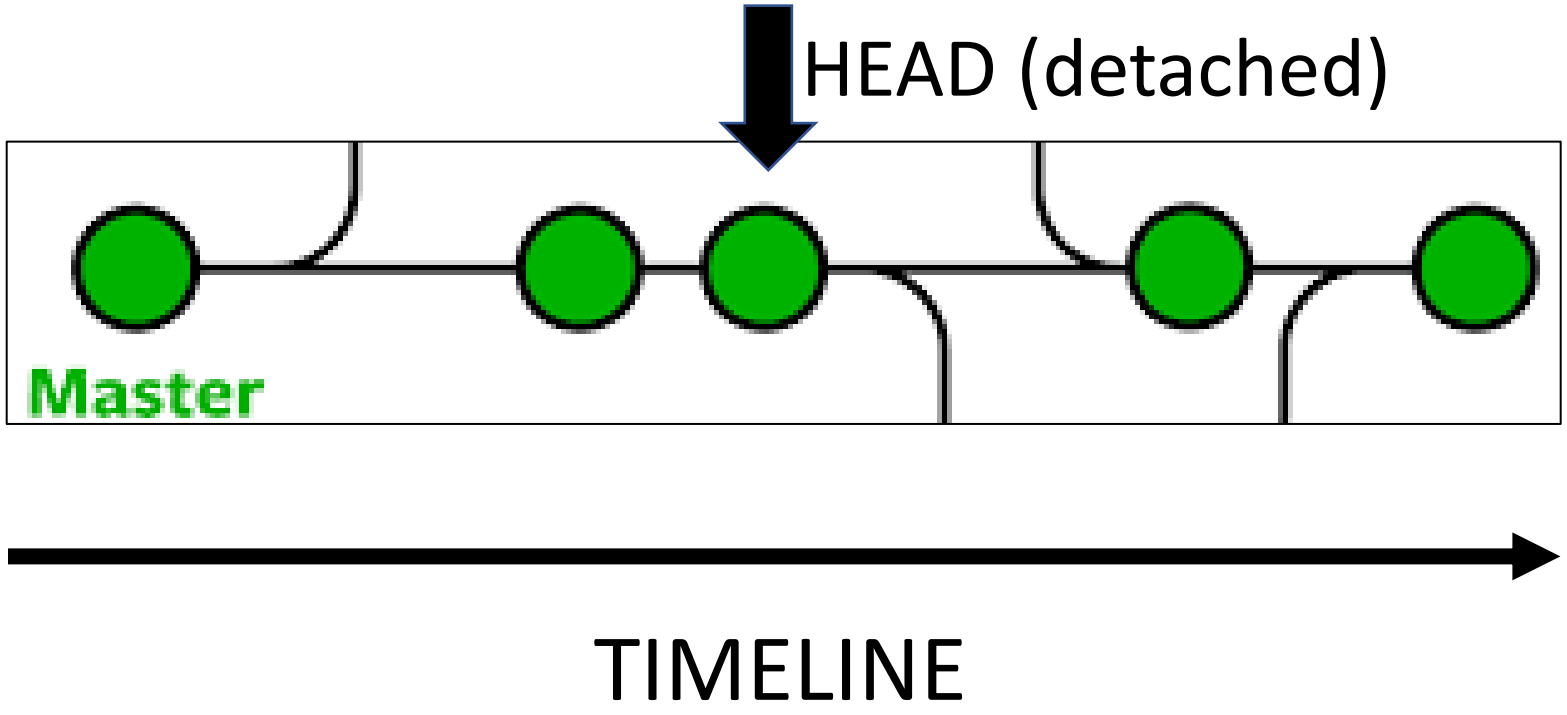


Time traveling : HEAD



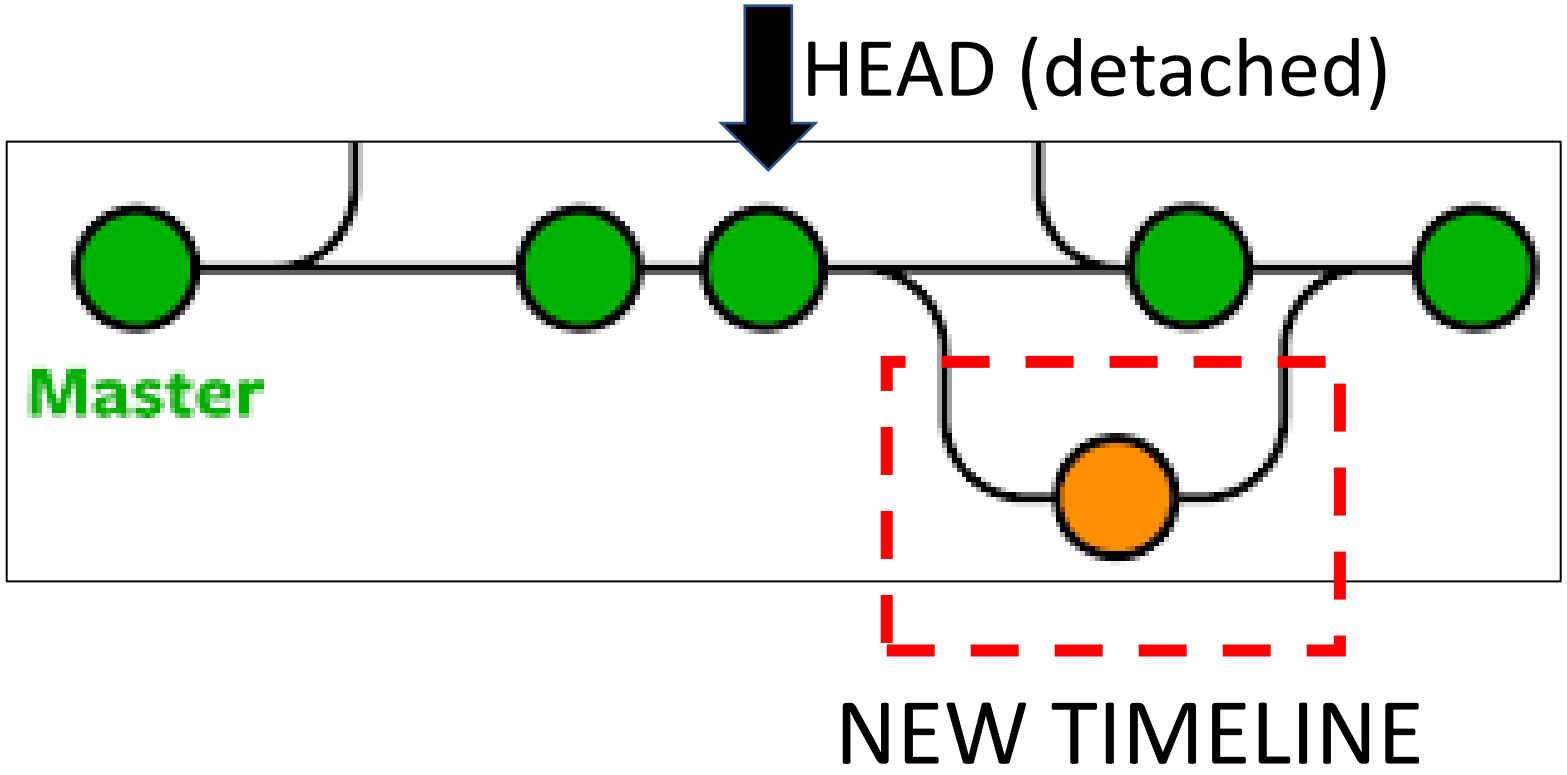
Time traveling : HEAD

- We can move to a previous commit (detached HEAD)



Time traveling : HEAD

- We can move to a previous commit (detached HEAD)



Navigate and compare

- git checkout to move HEAD
- git diff to compare

Ignore files

- .gitignore file
- GitHub provides us presets

TODO: Practice

- ✓ Define commits
- ✓ Navigate commits
- ✓ Compare commits
- ✓ Ignore files