

## UniDive Training School Course 3.1

# Git Infrastructure



Daniel Zeman, Agata Savary, Bruno Guillaume

[zeman@ufal.mff.cuni.cz](mailto:zeman@ufal.mff.cuni.cz)

<https://unidive.lisn.upsaclay.fr/>

# Outline

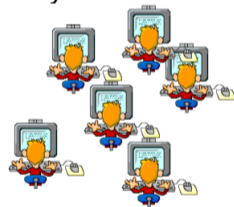
- 1 Git for beginners
- 2 UD GitHub repositories
- 3 PARSEME GitLab repositories
- 4 GitHub synchronization in Grew

# Installation

- Make sure Git is **installed on your system**
  - ▶ `https://shorturl.at/l5cf0`
  - ▶ (`https://github.com/UniDive/2024-UniDive-Chisinau-training-school/blob/main/Course-3-corpus-annotation-infrastructure/git-tutorial.pdf`)
- Two platforms: Microsoft Windows and Ubuntu. The latter should partly generalize to other Linux architectures and even to MacOS

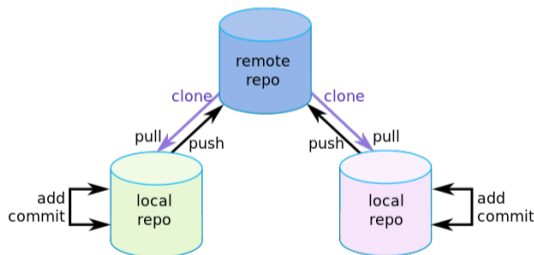
# Git in a nutshell

- Git is a piece of **software**, originating from the **Linux** development community
- It is a **version control** system: It records changes to a set of files over time so that you can **recall specific versions** later
- **Any types of files** can be versioned (e.g. images, `.pdf` files etc.)
- The power of versioning is optimal with **plain text** files, because it is easy to **visualize differences** between different versions in such files
- Examples of plain text files:
  - ▶ software source codes (in Python, C, Java, bash, PHP, etc.)
  - ▶ web pages in HTML or Markdown (e.g. Wiki pages),
  - ▶ XML files (see FLAT in Course 2)
  - ▶ language resources (e.g. in `.conllu` or `.cupt` formats – see below)
- Git is a de facto standard tool for **collaborative development** of software and data



# Git repository

- A Git **repository** (*repo* for short) is a directory containing:
  - ▶ user's files and directories to be versioned
  - ▶ a Git database (in a hidden folder called `.git`) containing all the history of previous versions
- There are usually **several copies** of the same repository: one central **remote repo** on the server, and several **local repos** on the personal machines of the collaborators



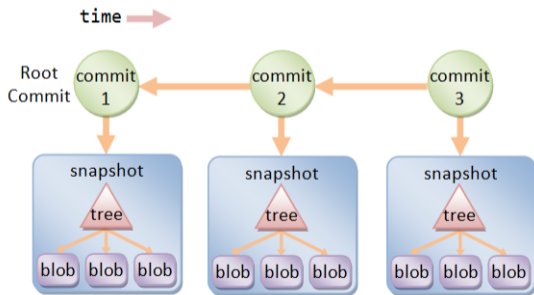
Source: [https://www3.ntu.edu.sg/home/ehchua/programming/howto/Git\\_HowTo.html](https://www3.ntu.edu.sg/home/ehchua/programming/howto/Git_HowTo.html)

# Tracked and untracked files

- A working directory of a Git repo can contain:
  - ▶ **tracked** files – versioned in the git database
  - ▶ **untracked** files – not versioned
- An **untracked** file can be turned into **tracked** and vice-versa

# Commit

- **Snapshot** = a picture of how your files look like at a given moment
- **Blob** = a picture of a single file
- **Commit** = taking a snapshot of all your **tracked** files
- The **history** of file versions in repository is stored by commits in the repo's **database** (stored in the `.git` directory)



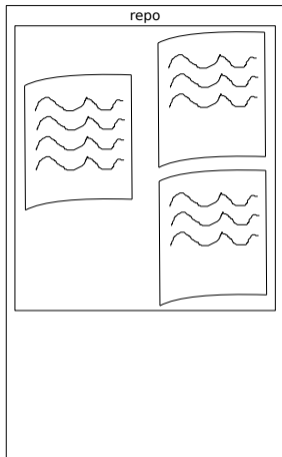
Source: [https://www3.ntu.edu.sg/home/ehchua/programming/howto/Git\\_HowTo.html](https://www3.ntu.edu.sg/home/ehchua/programming/howto/Git_HowTo.html)

# A commit

Authorship: Chantal Keller, Univ. Paris-Saclay

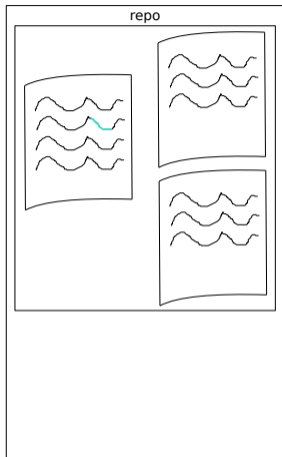


# A commit



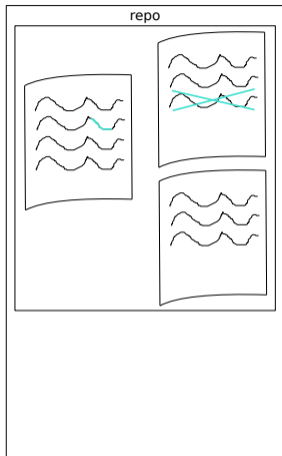
Authorship: Chantal Keller, Univ. Paris-Saclay

# A commit



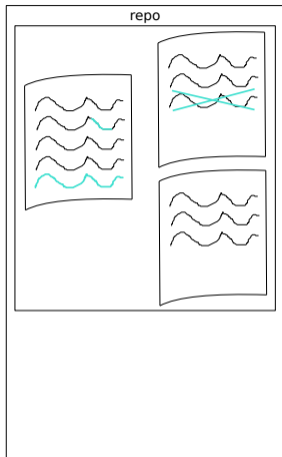
Authorship: Chantal Keller, Univ. Paris-Saclay

# A commit



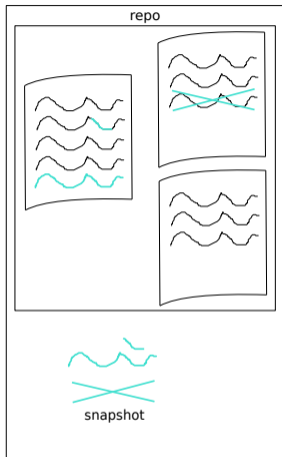
Authorship: Chantal Keller, Univ. Paris-Saclay

# A commit



Authorship: Chantal Keller, Univ. Paris-Saclay

# A commit

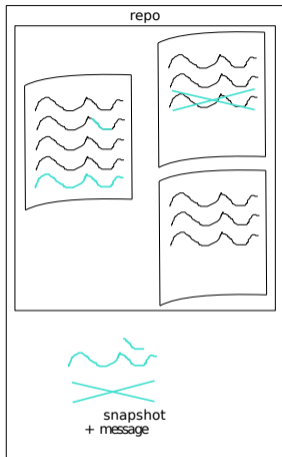


For simplicity, only the differences are shown in the snapshot.

In reality, a snapshot is taken of the whole file, not only of the changes.

Authorship: Chantal Keller, Univ. Paris-Saclay

# A commit

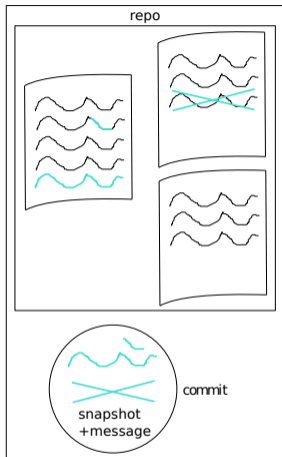


For simplicity, only the differences are shown in the snapshot.

In reality, a snapshot is taken of the whole file, not only of the changes.

Authorship: Chantal Keller, Univ. Paris-Saclay

# A commit



For simplicity, only the differences are shown in the snapshot.

In reality, a snapshot is taken of the whole files, not only of the changes.

Authorship: Chantal Keller, Univ. Paris-Saclay

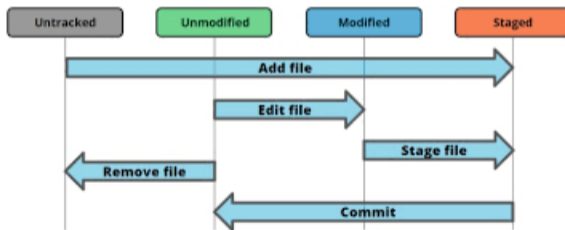
# Checksum

- Each commit has a unique identifier called a **checksum**
- **Checksum** = 40-character string composed of **hexadecimal** characters (0–9 and a–f) and calculated based on the contents of all tracked files
- Examples:
  - ▶ 17eac1dfa333bf9d757ed90c93a14e1fff120df1
  - ▶ df4ebd134564a52d98ead3da4e24cb689bef61d6
- The checksum is often shortened in git to the first few characters:
  - ▶ 17eac1d
  - ▶ df4ebd1



# States of a file

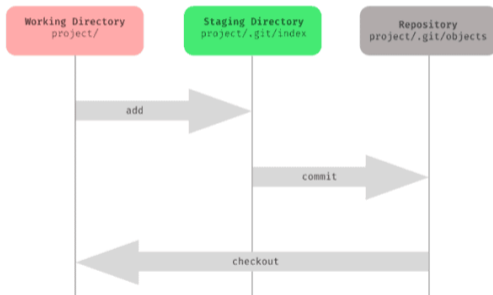
- Every file **tracked** by Git can be in one of the 3 states:
  - ▶ **modified** - you have **changed** the file but have **not committed** it to your database yet
  - ▶ **staged** - you have marked a modified file in its current version to **go into your next commit**
  - ▶ **committed** (or **unmodified**) - the current version of your file is safely stored in your local database



Source: [https://link.springer.com/chapter/10.1007/978-1-4842-6270-2\\_2](https://link.springer.com/chapter/10.1007/978-1-4842-6270-2_2)

# Areas for a file

- **working directory** – the actual location of your files
- **staging area** – a temporary area where changes are prepared for committing
- **Git database** – stores the history of commits

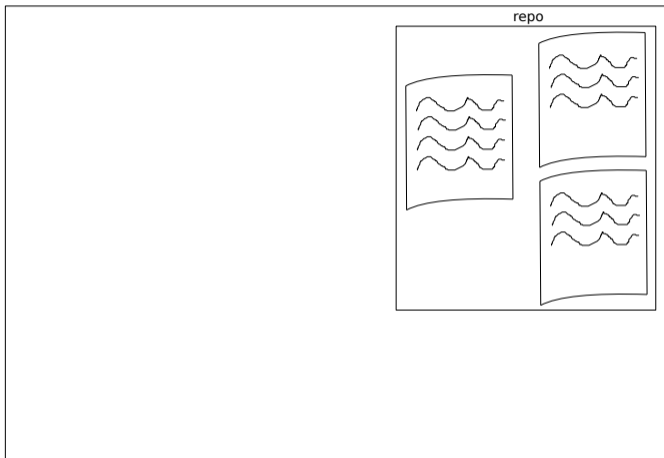


Source: <https://www.softwaretestinghelp.com/git-interview-questions/>

# Local and remote repository

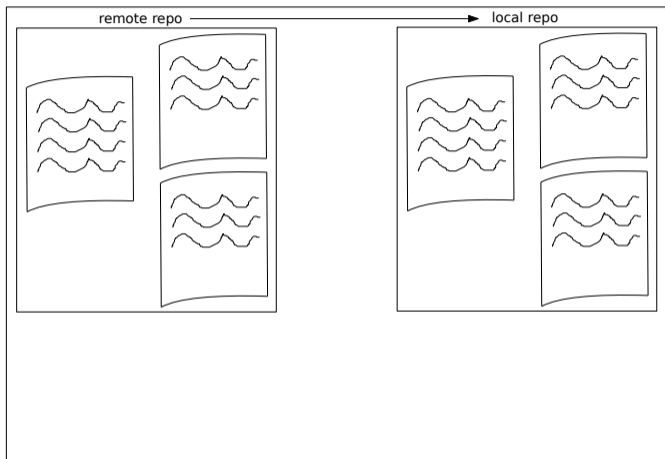
Authorship: Chantal Keller, Univ. Paris-Saclay

# Local and remote repository



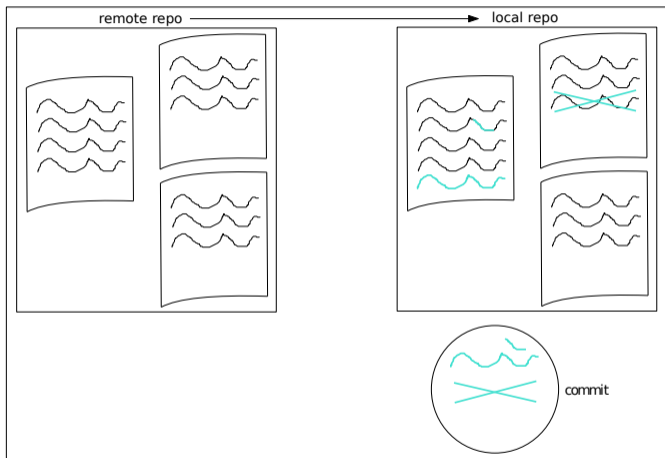
Authorship: Chantal Keller, Univ. Paris-Saclay

# Local and remote repository



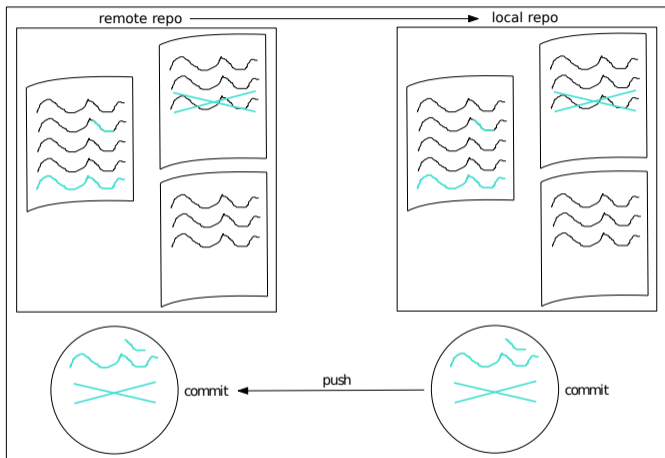
Authorship: Chantal Keller, Univ. Paris-Saclay

# Local and remote repository



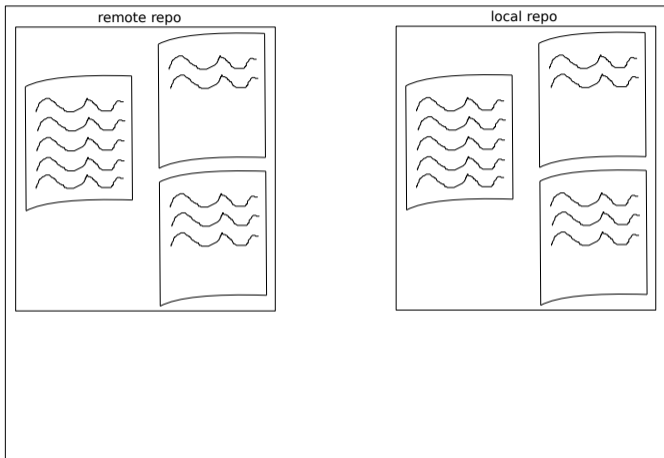
Authorship: Chantal Keller, Univ. Paris-Saclay

# Local and remote repository



Authorship: Chantal Keller, Univ. Paris-Saclay

# Local and remote repository



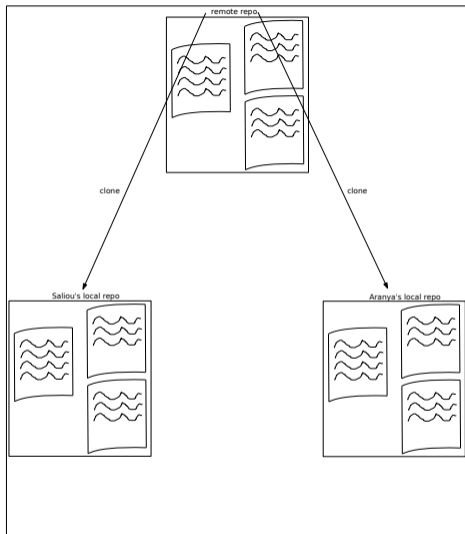
Authorship: Chantal Keller, Univ. Paris-Saclay



# Collaborative work

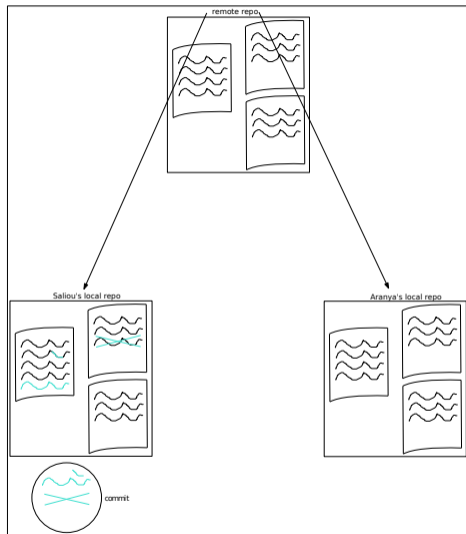
Authorship: Chantal Keller, Univ. Paris-Saclay

# Collaborative work



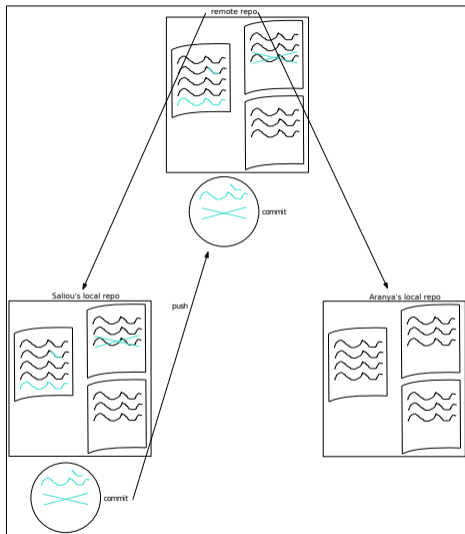
Authorship: Chantal Keller, Univ. Paris-Saclay

# Collaborative work



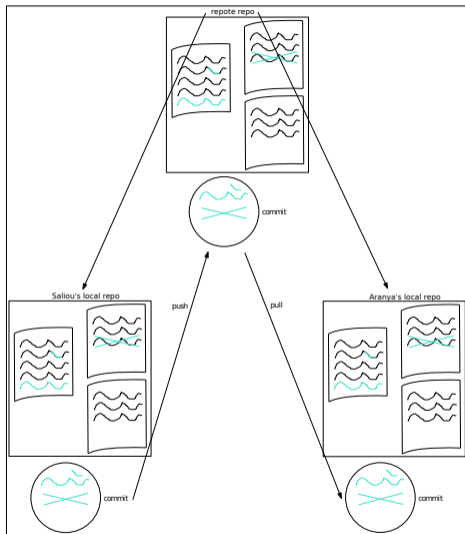
Authorship: Chantal Keller, Univ. Paris-Saclay

# Collaborative work



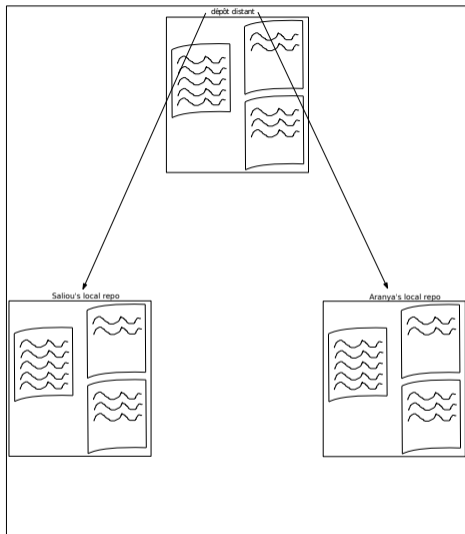
Authorship: Chantal Keller, Univ. Paris-Saclay

# Collaborative work



Authorship: Chantal Keller, Univ. Paris-Saclay

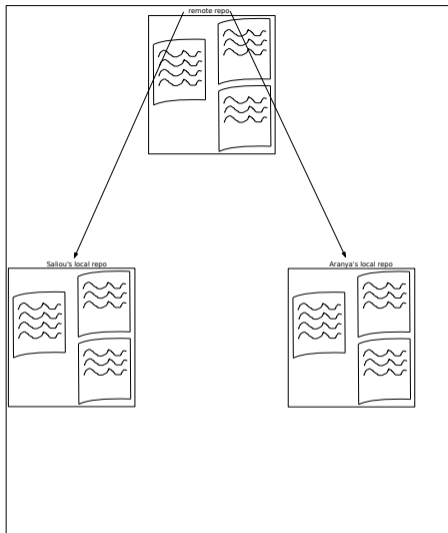
# Collaborative work



Authorship: Chantal Keller, Univ. Paris-Saclay

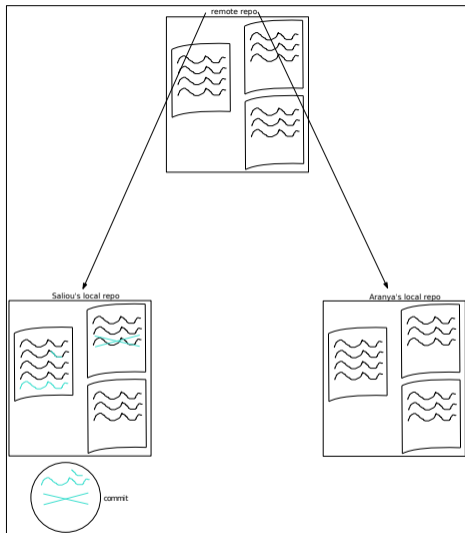
# Conflicts

# Conflicts

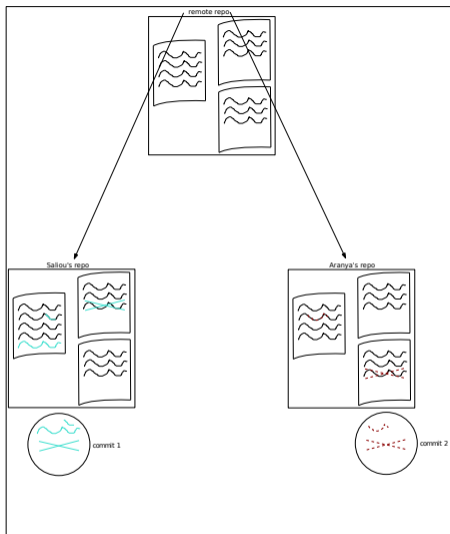




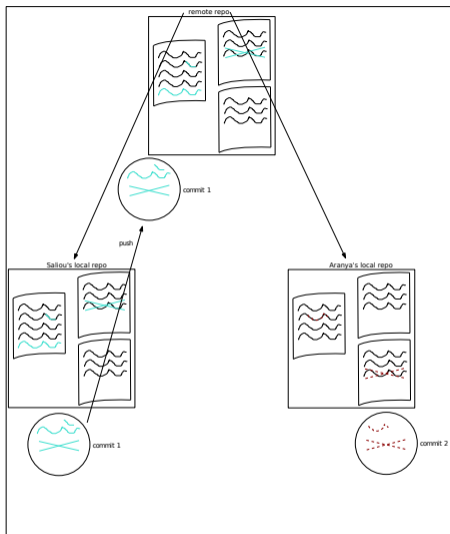
# Conflicts



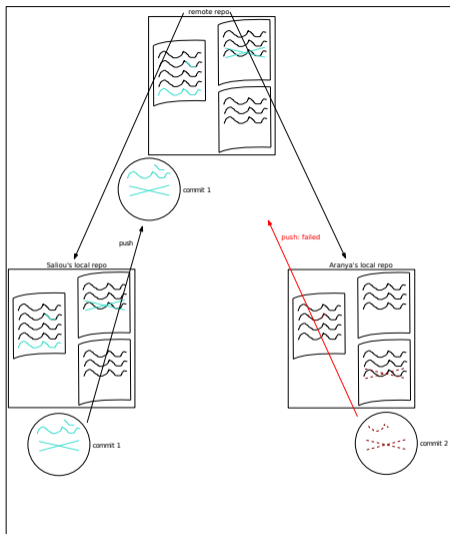
# Conflicts



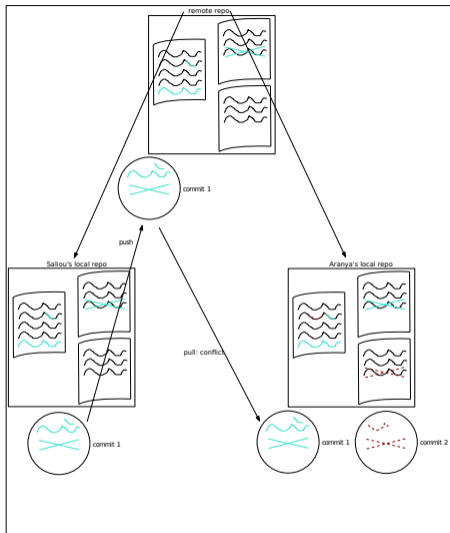
# Conflicts



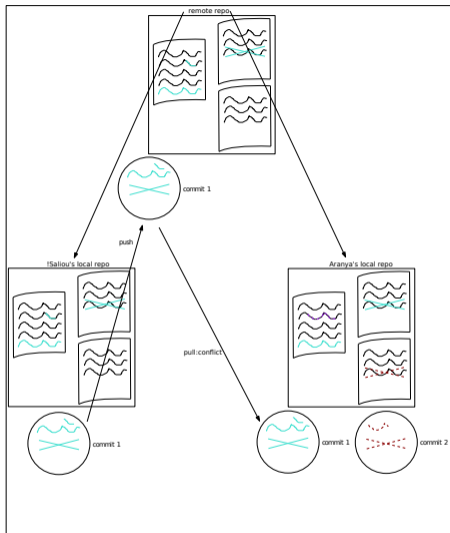
# Conflicts



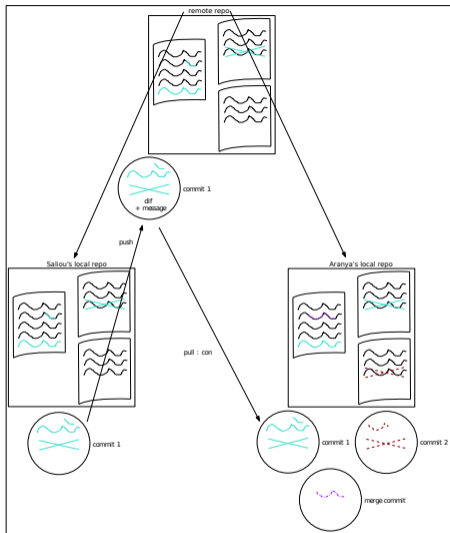
# Conflicts



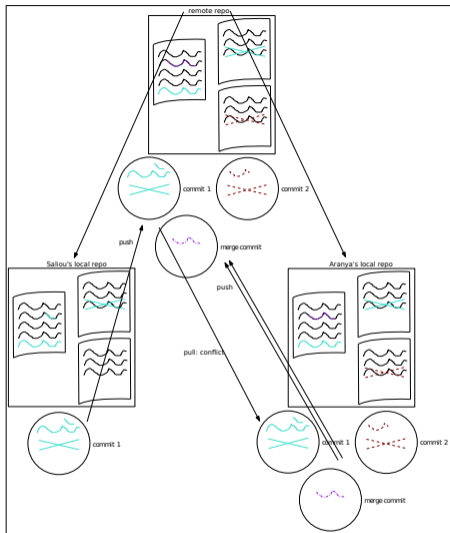
# Conflicts



# Conflicts

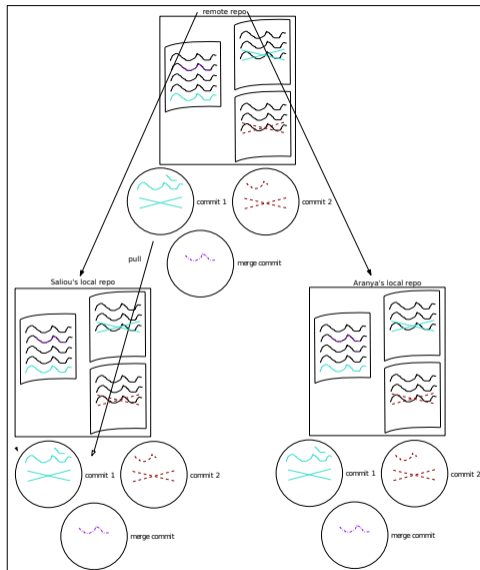


# Conflicts

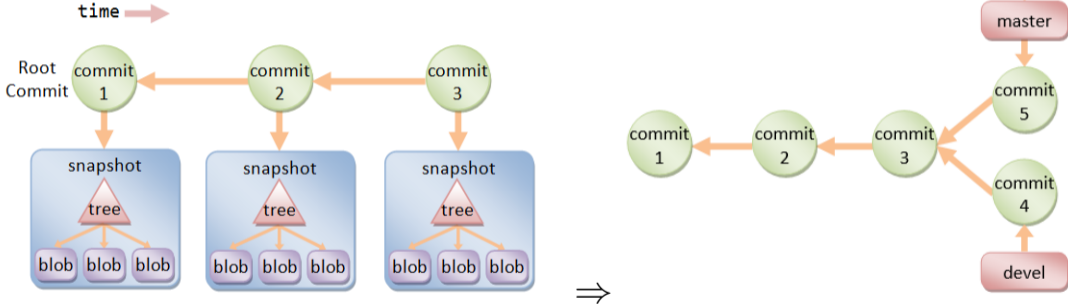




# Conflicts



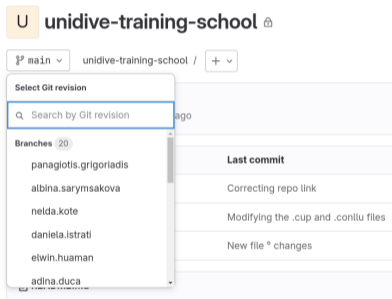
# Working with branches



Source: [https://www3.ntu.edu.sg/home/ehchua/programming/howto/Git\\_HowTo.html](https://www3.ntu.edu.sg/home/ehchua/programming/howto/Git_HowTo.html)

# Simple Git exercise

- Follow the link: <https://gitlab.com/parseme/unidive-training-school>
- Examine the files



- Examine the branches
- Follow the steps in README

## Further reading

- Git reference and manual <https://git-scm.com/book/en/v2>

# Git vs. GitHub vs. GitLab

like two different cars with the same engine (except that you can use Git ('the engine') without GitHub and GitLab, just on your local system or with your own server)



# Git vs. GitHub vs. GitLab

like two different cars with the same engine (except that you can use Git ('the engine') without GitHub and GitLab, just on your local system or with your own server)



@thebikephoto



# UD GitHub Repositories

The screenshot shows the main page of the **UniversalDependencies** repository on GitHub. The repository has 169 followers and a website link <http://universaldependencies.org/>. It features several pinned repositories: **docs** (HTML, 266 stars, 242 forks) and **tools** (Python, 199 stars, 42 forks). A search for **UD\_** in the repository list shows 391 results, with the top results being **UD\_Kurmanji-MG** (Northern Kurdish data, updated 7 minutes ago), **UD\_English-EWT** (English data, updated 1 hour ago), **UD\_Greek-Lesbian** (updated yesterday), **UD\_Amharic-ADT** (updated yesterday), and **UD\_Georgian-GEOWIKI** (Public).

The screenshot shows the **UD\_English-EWT** repository page. The **dev** branch is selected and circled in red. The page shows the repository's commit history and file list. The commit history includes a commit by **nschneid** titled "optatives: verb feats (#538)" from 1 hour ago. The file list includes **not-to-release**, **.gitignore**, **CONTRIBUTING.md**, **LICENSE.txt**, **README.md**, **en\_ewt-ud-dev.conllu**, **en\_ewt-ud-test.conllu**, **en\_ewt-ud-train.conllu**, and **stats.xml**. The **README** file is expanded, showing the repository's description and a **Summary** section.

**Summary**

A Gold Standard Universal Dependencies Corpus for English, built over the source material of the English Web Treebank LDC2012T13 (<https://catalog.ldc.upenn.edu/LDC2012T13>).



# UD GitHub Branches

- UD GitHub: `master` and `dev` branches
  - ▶ `master` is for users, not for data maintainers
    - ★ It always matches the last official release (treebank data-wise, not 100% file-wise)
    - ★ `dev` merged into `master` automatically at release time
    - ★ You do not have permission to push to `master`
    - ★ Do not create Pull Requests (GitHub may offer them when you cannot push directly)
    - ★ Do not contaminate your history from `master` (do not pull the `eval.log` file)
- Local copies of remote branches: `orig/dev`
- Local copy of the `dev` branch

# Git Workflow (Local)

- Create repository or **clone** a repository from GitHub / GitLab

# Git Workflow (Local)

- Create repository or **clone** a repository from GitHub / GitLab
- Add files to repository: `git add`
  - ▶ Just copying a file to the folder is not enough!

# Git Workflow (Local)

- Create repository or **clone** a repository from GitHub / GitLab
- Add files to repository: `git add`
  - ▶ Just copying a file to the folder is not enough!
- 'Commit' new revision: `git commit`
  - ▶ `git commit -a -m 'Added CoNLL-U file to the repo.'`

# Git Workflow (Local)

- Create repository or **clone** a repository from GitHub / GitLab
- Add files to repository: `git add`
  - ▶ Just copying a file to the folder is not enough!
- 'Commit' new revision: `git commit`
  - ▶ `git commit -a -m 'Added CoNLL-U file to the repo.'`
- Edit and save a file

# Git Workflow (Local)

- Create repository or **clone** a repository from GitHub / GitLab
- Add files to repository: `git add`
  - ▶ Just copying a file to the folder is not enough!
- 'Commit' new revision: `git commit`
  - ▶ `git commit -a -m 'Added CoNLL-U file to the repo.'`
- Edit and save a file
- Check `git status`: Working copy has new changes

# Git Workflow (Local)

- Create repository or **clone** a repository from GitHub / GitLab
- Add files to repository: `git add`
  - ▶ Just copying a file to the folder is not enough!
- 'Commit' new revision: `git commit`
  - ▶ `git commit -a -m 'Added CoNLL-U file to the repo.'`
- Edit and save a file
- Check `git status`: Working copy has new changes
- Check the changes using `git diff`

# Git Workflow (Local)

- Create repository or **clone** a repository from GitHub / GitLab
- Add files to repository: `git add`
  - ▶ Just copying a file to the folder is not enough!
- 'Commit' new revision: `git commit`
  - ▶ `git commit -a -m 'Added CoNLL-U file to the repo.'`
- Edit and save a file
- Check `git status`: Working copy has new changes
- Check the changes using `git diff`
- `git commit -a`
  - ▶ Equivalent to `git add ; git commit`



# Git Workflow (with Remote Repo)

- Pull (download) the new changes from the remote repo: `git pull` (possibly with `--no-edit`)

# Git Workflow (with Remote Repo)

- Pull (download) the new changes from the remote repo: `git pull` (possibly with `--no-edit`)
- Local editing cycle (see previous slide)

## Git Workflow (with Remote Repo)

- Pull (download) the new changes from the remote repo: `git pull` (possibly with `--no-edit`)
- Local editing cycle (see previous slide)
- After commit, push (upload) your changes to the remote repo: `git push`
- Still possible that there are new changes: pull again
- If unlucky, there are edit conflicts **on the same lines**, which you must resolve, then commit and push again

# Select Files for Version Control (git add)

The screenshot shows a Windows File Explorer window with the address bar set to `C:\Users\Dan\Documents\Lingvistika\Projekty\universal-dependencies\UD_Amharic-Inku [dev .0 t0]`. The left sidebar shows a tree view of folders, with `UD_Amharic-Inku` selected. The main pane displays a list of files:

Název	Datum změny	Typ	Velikost
am_att-ud-test.conllu	30.05.2022 23:11	Soubor CONLLU	1 066 kB
CONTRIBUTING.md			1 kB
LICENSE.txt			1 kB
README.md			1 kB

A context menu is open over the `am_att-ud-test.conllu` file, showing options such as `Otevřít`, `Send a copy...`, `Move to Dropbox`, `Back up to Dropbox...`, `Sdílet na Skypu`, `Zkontrolovat pomocí ESET NOD32 Antivirus`, `Další možnosti`, `Sdílet`, `Otevřít v programu`, `Udělit přístup pro`, `TortoiseGit`, `TortoiseSVN`, `Obnovit předchozí verze`, `Odeslat`, and `Vymout`. A secondary menu is also visible, containing `Diff later`, `Add...`, `Add to ignore list`, `Settings`, `Help`, and `About`.

# Select Files for Version Control (git add)

The screenshot shows a Windows File Explorer window with the address bar set to `C:\Users\Dan\Documents\Lingvistika\Projekty\universal-dependencies\UD_Amharic-Inku [ dev .0 t0 ]`. The left sidebar shows a tree view of folders, with `UD_Amharic-Inku` selected. The main pane displays a table of files:

Název	Datum změny	Typ	Velikost
am_att-ud-test.conllu	30.05.2022 23:11	Soubor CONLLU	1 066 kB
CONTRIBUTING.md	17.06.2018 23:35	Soubor MD	1 kB
LICENSE.txt	08.07.2016 17:31	Soubor TXT	1 kB
README.md	26.05.2021 18:33	Soubor MD	1 kB

The file `am_att-ud-test.conllu` is selected. The status bar at the bottom indicates `Počet položek: 4` and `Počet vybraných položek: 1; 1,04 MB`.

# Open a File to Edit

```
es_gsd-ud-test.conllu [C:\Users\Dan\Documents\Lingvistika\Projekty\universal-dependencies\UD_Spanish-GSD] - Notepad2
File Edit View Settings ?
25 22 Pedro pedro PROPN _ _ 19 nmod _ SpaceAfter=No
26 23 ; ; PUNCT _ _ 3 punct _ _
27
28 # sent_id = es-dev-003-s415
29 # text = La Provincia de Mamoré es una provincia del departamento del Beni en Bolivia.
30 1 La el DET _ _ Definite=Def|Gender=Fem|Number=Sing|PronType=Art 2 det _ _
31 2 Provincia provincia NOUN _ _ 7 nsubj _ _
32 3 de de ADP _ _ 4 case _ _
33 4 Mamoré mamoré PROPN _ _ 2 nmod _ _
34 5 es ser AUX _ _ Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin 7 cop _ _
35 6 una uno DET _ _ Definite=Ind|Gender=Fem|Number=Sing|PronType=Art 7 det _ _
36 7 provincia provincia NOUN _ _ Gender=Fem|Number=Sing 0 root _ _
37 8-9 del _ _ _ _ _ _ _ _
38 8 de de ADP _ _ 10 case _ _
39 9 el el DET _ _ Definite=Def|Gender=Masc|Number=Sing|PronType=Art 10 det _ _
40 10 departamento departamento NOUN _ _ Gender=Masc|Number=Sing 7 nmod _ _
41 11-12 del _ _ _ _ _ _ _ _
42 11 de de ADP _ _ 13 case _ _
43 12 el el DET _ _ Definite=Def|Gender=Masc|Number=Sing|PronType=Art 13 det _ _
44 13 Beni beni PROPN _ _ 10 nmod _ _
45 14 en en ADP _ _ 15 case _ _
46 15 Bolivia bolivia PROPN _ _ 10 nmod _ SpaceAfter=No
47 16 . . PUNCT _ _ 7 punct _ _
48
49 # sent_id = es-dev-003-s416
50 # text = Con la llegada de maquinaria (martillos hidráulicos y compresores), la perforación manual dejó de ser necesaria en las canteras y minas.
Ln 31: 13 546 Col 58 Sel 1 719 kB UTF-8 LF INS Default Text
```

# Edit and Save the File

```
* es_gsd-ud-test.conllu [C:\Users\Dan\Documents\Lingvistika\Projekty\universal-dependencies\UD_Spanish-GSD] - Notepad2
File Edit View Settings ?
25 22 Pedro pedro PROPN _ _ 19 nmod _ SpaceAfter=No
26 23 ; ; PUNCT _ _ 3 punct _ _
27
28 # sent_id = es-dev-003-s415
29 # text = La Provincia de Mamoré es una provincia del departamento del Beni en Bolivia.
30 1 La el DET _ _ Definite=Def|Gender=Fem|Number=Sing|PronType=Art 2 det _ _
31 2 Provincia provincia NOUN _ _ Gender=Fem|Number=Sing 7 nsubj _ _
32 3 de de ADP _ _ 4 case _ _
33 4 Mamoré mamoré PROPN _ _ 2 nmod _ _
34 5 es ser AUX _ _ Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin 7 cop _ _
35 6 una uno DET _ _ Definite=Ind|Gender=Fem|Number=Sing|PronType=Art 7 det _ _
36 7 provincia provincia NOUN _ _ Gender=Fem|Number=Sing 0 root _ _
37 8-9 del _ _ _ _ _ _ _ _
38 8 de de ADP _ _ 10 case _ _
39 9 el el DET _ _ Definite=Def|Gender=Masc|Number=Sing|PronType=Art 10 det _ _
40 10 departamento departamento NOUN _ _ Gender=Masc|Number=Sing 7 nmod _ _
41 11-12 del _ _ _ _ _ _ _ _
42 11 de de ADP _ _ 13 case _ _
43 12 el el DET _ _ Definite=Def|Gender=Masc|Number=Sing|PronType=Art 13 det _ _
44 13 Beni beni PROPN _ _ 10 nmod _ _
45 14 en en ADP _ _ 15 case _ _
46 15 Bolivia bolivia PROPN _ _ 10 nmod _ SpaceAfter=No
47 16 . . PUNCT _ _ 7 punct _ _
48
49 # sent_id = es-dev-003-s416
50 # text = Con la llegada de maquinaria (martillos hidráulicos y compresores), la perforación manual dejó de ser necesaria en las canteras y minas.
Ln 31: 13 546 Col 57 Sel 22 719 kB UTF-8 LF INS Default Text
```

# Working Folder Now Differs from Last Git Revision

The screenshot shows a Windows File Explorer window with the address bar displaying the path: C:\Users\Dan\Documents\Lingvistika\Projekty\universal-dependencies\UD\_Spanish-GSD [ dev 10 10 ]. The left pane shows a folder tree with 'UD\_Spanish-GSD' selected. The right pane shows a list of files with columns for Name, Date modified, Type, and Size.

Název	Datum změny	Typ	Velikost
.gitignore	06.02.2018 11:31	Textový dokument	0 kB
CONTRIBUTING.md	12.11.2018 15:12	Soubor MD	1 kB
es_gsd-ud-dev.conllu	18.06.2024 10:51	Soubor CONLLU	2 202 kB
es_gsd-ud-test.conllu	18.06.2024 12:09	Soubor CONLLU	720 kB
es_gsd-ud-train.conllu	18.06.2024 10:53	Soubor CONLLU	22 750 kB
LICENSE.txt	08.07.2016 17:31	Soubor TXT	1 kB
README.md	05.04.2024 18:04	Soubor MD	7 kB
stats.xml	18.06.2024 10:51	Soubor XML	12 kB



# Save ('git commit') New Revision to the History

Commit to: dev  new branch

Message:

Fixed gender of "provincia".

Amend Last Commit 1/29

Set author date

Set author Dan Zeman <zeman@ufal.mff.cuni.cz> Add Signed-off-by

Changes made (F5: refresh, double-click on file for diff):

Check: **All** None Unversioned **Versioned** Added Deleted **Modified** Files Submodules

Path	Extension	Status	Lines ad...	Lines rem...
<input checked="" type="checkbox"/> es_gsd-ud-test.conllu	.conllu	Modified	1	1

1 files selected, 1 files total

Staging support (EXPERIMENTAL)

Show Unversioned Files

Do not autoselect submodules

Show Whole Project

Message only

Commit & Push | Cancel | Help

# View diff before Committing

The screenshot displays the TortoiseGit Merge interface for the file `es_gsd-ud-test.conllu`. It compares two versions: the 'Working Tree' (right pane) and a specific commit 'ca8d177f' (left pane). The diff highlights a change in the morphological analysis of the word 'provincia' at line 31. In the 'Working Tree' version, the analysis includes the gender and number information: `Gender=Fem|Number=Sing`. In the 'ca8d177f' version, this information is missing. The rest of the file content, including the sentence `La Provincia de Mamoré es una provincia del departamento del Beni en Bolivi`, is identical in both versions.

```
es_gsd-ud-test.conllu: ca8d177f
20 17 →en →en →ADP → 19 →case →
21 18 →la →el →DET → Definite=Def|Gender=Fem|Number=Sing|PronType=Art →19 →det →
22 19 →Plaza →plaza →PROPN → Gender=Fem|Number=Sing →16 →obl →
23 20 →de →de →ADP → 22 →case →
24 21 →San →san →PROPN → 22 →amod →
25 22 →Pedro →pedro →PROPN → 19 →nmod → SpaceAfter=No.
26 23 →; →; →PUNCT → 3 →punct →
27 |
28 # sent_id = es-dev-003-s415.
29 # text = La Provincia de Mamoré es una provincia del departamento del Beni en Bolivi
30 1 →La →el →DET → Definite=Def|Gender=Fem|Number=Sing|PronType=Art →2 →det →
31 2 →Provincia →provincia →NOUN → 7 →nsubj →
32 3 →de →de →ADP → 4 →case →
33 4 →Mamoré →mamoré →PROPN → 2 →nmod →
34 5 →es →ser →AUX → Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin →7 →cop
35 6 →una →uno →DET → Definite=Ind|Gender=Fem|Number=Sing|PronType=Art →7 →det →
36 7 →provincia →provincia →NOUN → Gender=Fem|Number=Sing →0 →root →
37 8 →del →de →ADP → 10 →case →
38 9 →de →de →ADP → 10 →case →
39 9 →el →el →DET → Definite=Def|Gender=Masc|Number=Sing|PronType=Art →10 →det →
40 10 →departamento →departamento →NOUN → Gender=Masc|Number=Sing →7 →nmod →
41 11 →del →de →ADP → 13 →case →
42 11 →de →de →ADP → 13 →case →
43 12 →el →el →DET → Definite=Def|Gender=Masc|Number=Sing|PronType=Art →13 →det →
44 13 →Beni →beni →PROPN → 10 →nmod →
45 14 →en →en →ADP → 15 →case →
46 15 →Bolivia →bolivia →PROPN → 10 →nmod → SpaceAfter=No.
47 16 →; →; →PUNCT → 7 →punct →
48 |
49 # sent_id = es-dev-003-s416.
50 # text = Con la llegada de maquinaria (martillos hidráulicos y compresores), la per
51 1 →Con →con →ADP → 3 →case →
52 2 →la →el →DET → Definite=Def|Gender=Fem|Number=Sing|PronType=Art →3 →det →
53 3 →llegada →llegada →NOUN → Gender=Fem|Number=Sing →19 →nmod →
54 4 →de →de →ADP → 5 →case →
55 5 →maquinaria →maquinaria →NOUN → Gender=Fem|Number=Sing →3 →nmod →
56 6 →( →( →PUNCT → 7 →punct → SpaceAfter=No.

es_gsd-ud-test.conllu: Working Tree
20 17 →en →en →ADP → 19 →case →
21 18 →la →el →DET → Definite=Def|Gender=Fem|Number=Sing|PronType=Art →19 →det →
22 19 →Plaza →plaza →PROPN → Gender=Fem|Number=Sing →16 →obl →
23 20 →de →de →ADP → 22 →case →
24 21 →San →san →PROPN → 22 →amod →
25 22 →Pedro →pedro →PROPN → 19 →nmod → SpaceAfter=No.
26 23 →; →; →PUNCT → 3 →punct →
27 |
28 # sent_id = es-dev-003-s415.
29 # text = La Provincia de Mamoré es una provincia del departamento del Beni en Bolivi
30 1 →La →el →DET → Definite=Def|Gender=Fem|Number=Sing|PronType=Art →2 →det →
31 2 →Provincia →provincia →NOUN → Gender=Fem|Number=Sing →7 →nsubj →
32 3 →de →de →ADP → 4 →case →
33 4 →Mamoré →mamoré →PROPN → 2 →nmod →
34 5 →es →ser →AUX → Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin →7 →cop
35 6 →una →uno →DET → Definite=Ind|Gender=Fem|Number=Sing|PronType=Art →7 →det →
36 7 →provincia →provincia →NOUN → Gender=Fem|Number=Sing →0 →root →
37 8 →del →de →ADP → 10 →case →
38 9 →de →de →ADP → 10 →case →
39 9 →el →el →DET → Definite=Def|Gender=Masc|Number=Sing|PronType=Art →10 →det →
40 10 →departamento →departamento →NOUN → Gender=Masc|Number=Sing →7 →nmod →
41 11 →del →de →ADP → 13 →case →
42 11 →de →de →ADP → 13 →case →
43 12 →el →el →DET → Definite=Def|Gender=Masc|Number=Sing|PronType=Art →13 →det →
44 13 →Beni →beni →PROPN → 10 →nmod →
45 14 →en →en →ADP → 15 →case →
46 15 →Bolivia →bolivia →PROPN → 10 →nmod → SpaceAfter=No.
47 16 →; →; →PUNCT → 7 →punct →
48 |
49 # sent_id = es-dev-003-s416.
50 # text = Con la llegada de maquinaria (martillos hidráulicos y compresores), la per
51 1 →Con →con →ADP → 3 →case →
52 2 →la →el →DET → Definite=Def|Gender=Fem|Number=Sing|PronType=Art →3 →det →
53 3 →llegada →llegada →NOUN → Gender=Fem|Number=Sing →19 →nmod →
54 4 →de →de →ADP → 5 →case →
55 5 →maquinaria →maquinaria →NOUN → Gender=Fem|Number=Sing →3 →nmod →
56 6 →( →( →PUNCT → 7 →punct → SpaceAfter=No.
```

For Help, press F1. Scroll horizontally with Ctrl-Scrollwheel

Left View: UTF-8 LF / -1

Right View: UTF-8 LF / +1

Conflicts: 0

CAP. NUM. SCRL...

# History of Commits (Revisions)

working copy

17010a96dc95e1ce1ce356664fd8091a5cef6a92 2024-06-17 8:54:32

EDIT and SAVE file(s)

# History of Commits (Revisions)

f642a465f23760b262090a463ed4e54dccc582b5 2024-06-17 8:58:27



17010a96dc95e1ce1ce356664fd8091a5cef6a92 2024-06-17 8:54:32

`git commit -a`



# History of Commits (Revisions)



# History of Commits (Revisions)

9d9179665aaee45662259c70aefe3a098a16d5e6 2024-06-17 9:52:14

f642a465f23760b262090a463ed4e54dccc582b5 2024-06-17 8:58:27

17010a96dc95e1ce1ce356664fd8091a5cef6a92 2024-06-17 8:54:32



# History of Commits (Revisions)

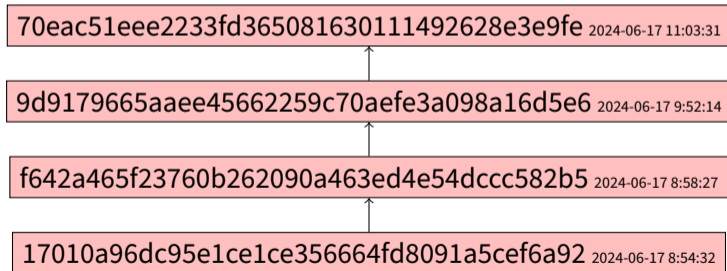


# History of Commits (Revisions)





# History of Commits (Revisions)



`git commit -a`

# History of Commits (Revisions)

working copy

17eac1dfa333bf9d757ed90c93a14e1fff120df1 2024-06-17 18:46:43



70eac51eee2233fd365081630111492628e3e9fe 2024-06-17 11:03:31

9d9179665aaee45662259c70aefe3a098a16d5e6 2024-06-17 9:52:14

f642a465f23760b262090a463ed4e54dccc582b5 2024-06-17 8:58:27

17010a96dc95e1ce1ce356664fd8091a5cef6a92 2024-06-17 8:54:32

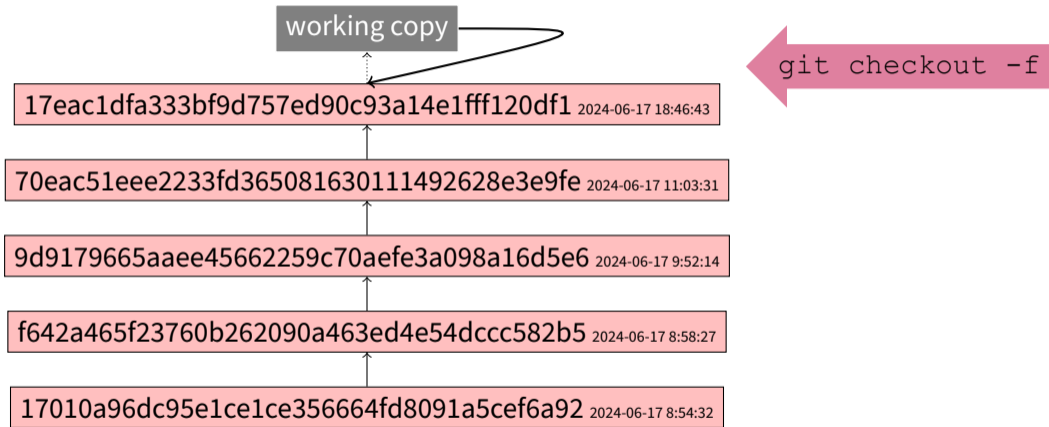
git commit -a

EDIT and SAVE file(s)

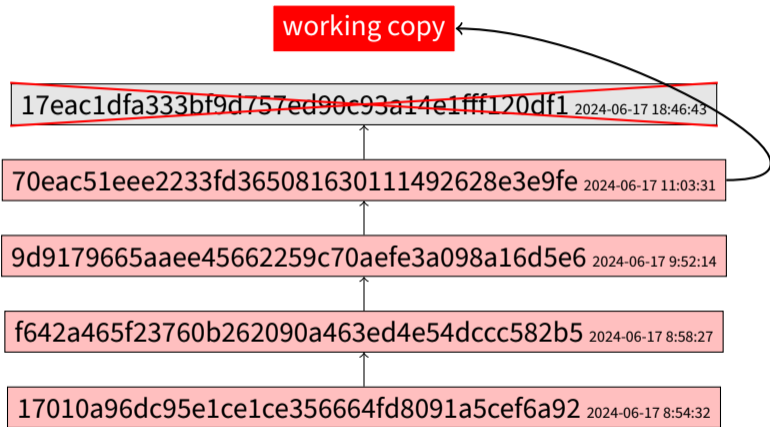
git commit -a

EDIT and SAVE file(s)

# Revert Changes in Working Copy



# Revert Changes Introduced in Commit X



# Revert to a Previous Commit in Terminal

- `git log` ... see commit numbers and messages
- `git checkout d95cb61bbd`
- `cp x.conllu x-old.conllu` ... temporarily copy as an unversioned file!
- `git checkout dev` ... back to the newest commit of the `dev` branch
- `mv x-old.conllu x.conllu` ... replace the new file with the old one
- `git commit -a -m "Reverted x.conllu to the version from last Tuesday (d95cb61bbd) "`

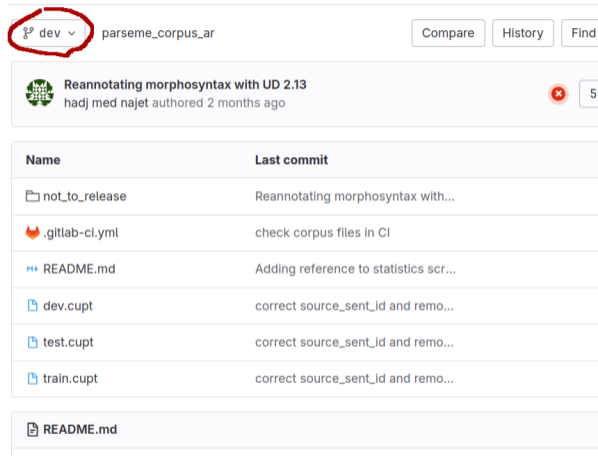
# More on Branches

- UD GitHub: `master` and `dev` branches
  - ▶ Recall: You only work with `dev`
- You may want to create your own additional branches
  - ▶ Big change, e.g. visit all instances of `PronType=Dem` and manually re-tag them as either `PRON` or `DET`
  - ▶ It takes time and you do not want to release it until it is done in the entire corpus: **create a branch** called `demonstratives`
  - ▶ In the meantime, fix a small bug in the `dev` branch and have it released in the next UD release
  - ▶ When the work in `demonstratives` is done, merge it back into `dev`


# PARSEME Gitlab repositories







Language	Working version	Latest release - version 1.3	Previous releases - version 1.2
 Arabic (AR)	[Repo] [Consistency check] [Grew-match]	Grew-match	NA
 Basque (EU)	[Repo] [Consistency check] [Grew-match]	Grew-match	Grew-match
 Bulgarian (BG)	[Repo] [Consistency check] [Grew-match]	Grew-match	NA

<https://gitlab.com/parseme/corpora/-/wikis/home>



dev parseme\_corpus\_ar Compare History Find

 Reannotating morphosyntax with UD 2.13  
hadj med najet authored 2 months ago

Name	Last commit
not_to_release	Reannotating morphosyntax with...
 .gitlab-ci.yml	check corpus files in CI
 README.md	Adding reference to statistics scr...
 dev.cupt	correct source_sent_id and remo...
 test.cupt	correct source_sent_id and remo...
 train.cupt	correct source_sent_id and remo...
 README.md	

[https://gitlab.com/parseme/parseme\\_corpus\\_ar](https://gitlab.com/parseme/parseme_corpus_ar)



- **Grew-match**

- ▶ Online interface for corpus exploration and visualization (see next courses)
- ▶ Available on versioned treebanks and on GitHub or GitLab *latest* data (open projects)
- ▶ The **Grew-match** backend is updated automatically after a push on the treebank repository
  - ★ **UD**: around 15 treebanks are available. Ex: UD\_English-EWT@dev (more on demand)
  - ★ **SUD**: all SUD native treebanks. Ex: SUD\_French-GSD@latest
  - ★ **Parseme**: all Parseme treebanks. Ex: PARSEME-FR@master





- **Arborator-Grew**

- ▶ Online interface for exploration annotation and maintenance (see next courses)
- ▶ Projects in **Arborator-Grew** can be synchronized with a GitHub repository
  - ★ You can **push** annotation done in **Arborator-Grew** to GitHub
  - ★ You can **pull** updates from GitHub in **Arborator-Grew** project