Basics of version control

Data plumbers' corner, session 3

Arianna Masciolini
Språkbanken Text, University of Gothenburg

8 November 2024

Today's session

1. Variations on the theme of Alice (bash exercises)
2. Intro to version control and Git
3. Hands-on Git tutorial

What you will need

- a Unix-like shell
- the basic Git CLI program
- a GitHub account

Alice in Unixland

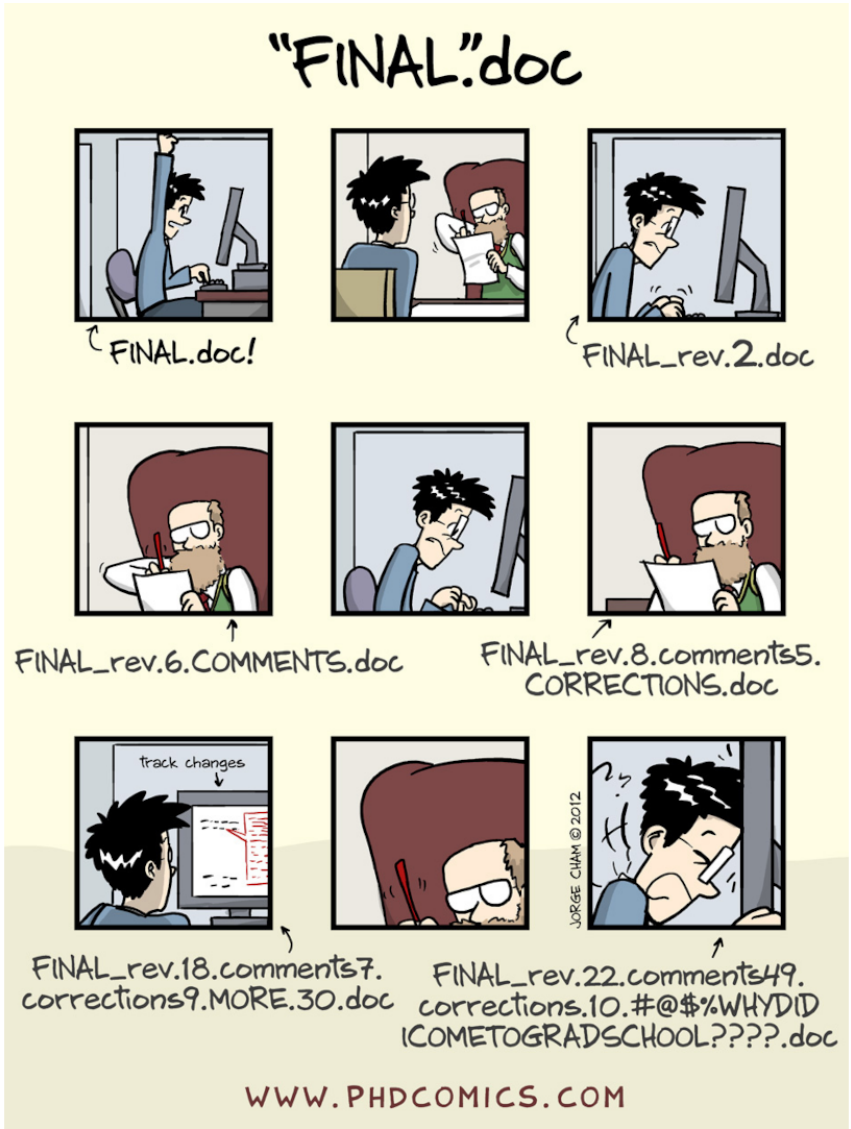
```
cat alice.txt | tr " " "\n" | sort
| uniq -c | sort -nr | head -n10
```

Exercises

Modify the text processing command so to:

- show the *20* most frequent words
- show the 20 *least* frequent words
- show all words and their counts *in alphabetical order*
- write the results to a new text file
- show the 10 most frequent words, *excluding punctuation*
- show all words *whose frequency is exactly 1*
- ...

Manual versioning



Version Control Systems

1. need to save disk space: **patch-based VCS** (e.g. RCS)
2. need for collaborative development: **centralized VCSs** (e.g. SVN)
3. need to work offline and prevent the server from being the "single point of failure": **distributed VCSs** (e.g. Git, the current *de facto* standard)

Behind Git

Linus Torvalds, the creator of both Linux and Git

recommended: youtube.com/watch?v=4XpnKHJAok8

Git: basic concepts

image from xkcd.com

- (named) filesystem snapshots rather than patches
- version history as a directed graph
- distributed, supports easy branching
- several interfaces and “hubs”

Hands-on Git tutorial

1. local usage (simple versioning)
2. setting up a remote GitHub repository (versioning + backup)
3. collaborative usage with GitHub