

Git 101

Petr Svarny, 2020

Versioning

"FINAL".doc



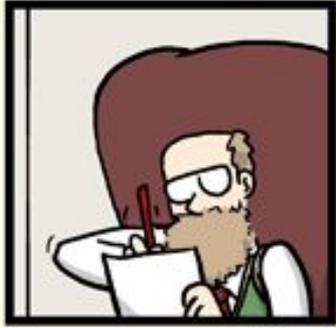
FINAL.doc!



FINAL_rev.2.doc

<http://phdcomics.com/comics/archive.php?comid=1531>

Versioning



FINAL_rev.6.COMMENTS.doc

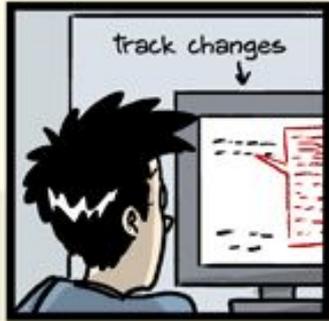


FINAL_rev.8.comments5.
CORRECTIONS.doc



Versioning

JORGE CHAM © 2012

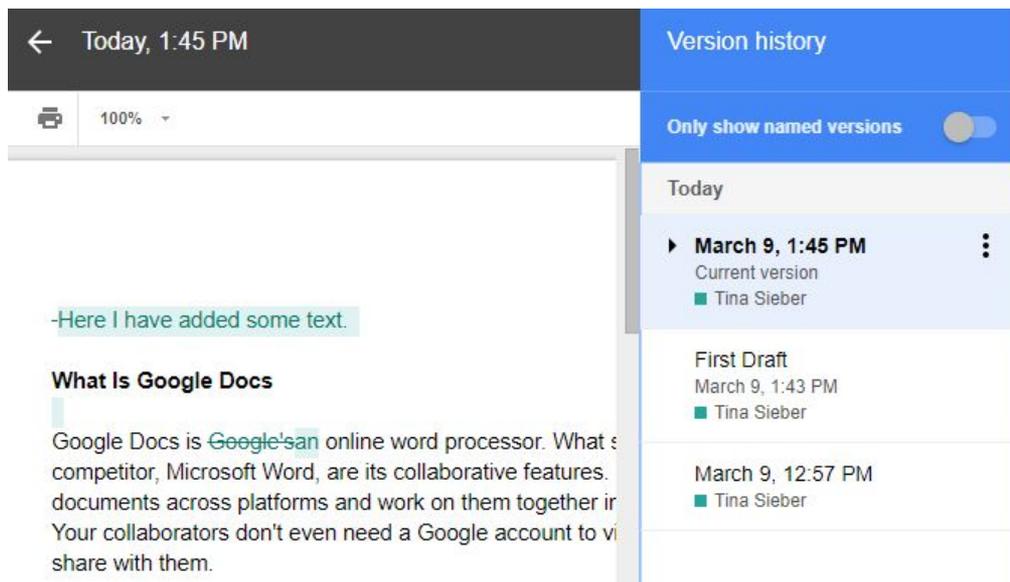


FINAL_rev.18.comments7.
corrections9.MORE.30.doc

FINAL_rev.22.comments49.
corrections.10.#@\$%WHYDID
ICOMETOGRADSCHOOL????.doc



Implicit versioning



The screenshot shows the Google Docs interface with the version history panel open on the right. The main document area on the left contains a heading "What Is Google Docs" and a paragraph of text. The version history panel on the right is titled "Version history" and has a toggle switch for "Only show named versions" which is turned on. The panel lists three versions under the heading "Today":

- March 9, 1:45 PM** (Current version) by Tina Sieber
- First Draft** (March 9, 1:43 PM) by Tina Sieber
- March 9, 12:57 PM** by Tina Sieber

The document content in the background includes a back arrow, the time "Today, 1:45 PM", a printer icon, and a zoom level of "100%". The text in the document is:

-Here I have added some text.

What Is Google Docs

Google Docs is Google's online word processor. What's its competitor, Microsoft Word, are its collaborative features. documents across platforms and work on them together in. Your collaborators don't even need a Google account to view share with them.

Git

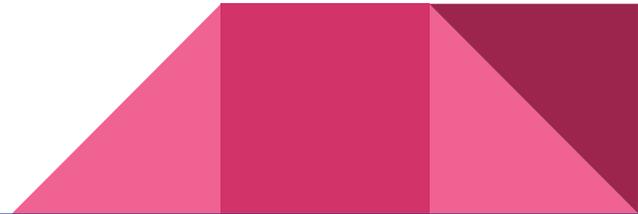
THIS IS GIT. IT TRACKS COLLABORATIVE WORK
ON PROJECTS THROUGH A BEAUTIFUL
DISTRIBUTED GRAPH THEORY TREE MODEL.

COOL. HOW DO WE USE IT?

NO IDEA. JUST MEMORIZE THESE SHELL
COMMANDS AND TYPE THEM TO SYNC UP.
IF YOU GET ERRORS, SAVE YOUR WORK
ELSEWHERE, DELETE THE PROJECT,
AND DOWNLOAD A FRESH COPY.

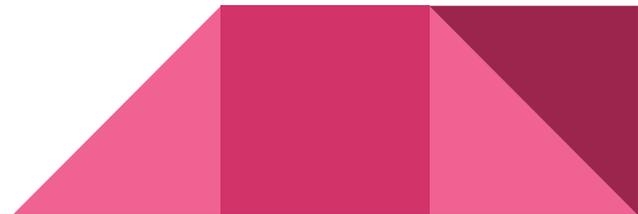


Why do you need to version?



Why do you need to version?

- Sharing
- Annotated list of changes
- Back up (also from mistakes)
- Review and history

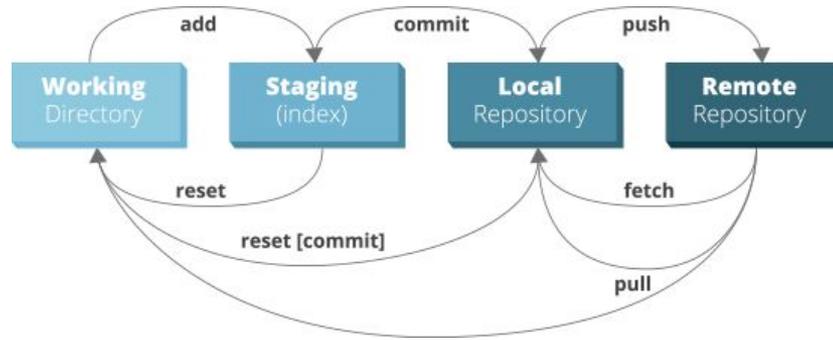


How does Git work?



- History
 - List of changes
 - Change unit = commit or change set
- + means added line, - deleted line
 - If something changed, you will see previous state as deleted and new state as added
- Commit reference the previous state (parent commit)
- Tracking changes works well only for plain text files (i.e. not Word or images)

Git cheat sheet



[GitHub-cheat-sheet-graphic-v1.jpg](#)

Basic practice and insight

- A [visual representation](#) of git working for learning purposes
- [SW Carpentry](#) tutorial
- [Git Bash](#) for Windows, [Git](#) for Linux
- [Git](#) homepage



Configure git

- When using Git for the first time on computer we need to configure
 - Name and email address
 - Preferred text editor ([more information](#))
 - Use these settings globally

```
$ git config --global user.name "name surname"
```

```
$ git config --global user.email "name@example.com"
```

Exercise

- Configure Git on your computer
 - All settings are global
 - Name
 - Email
 - Text editor



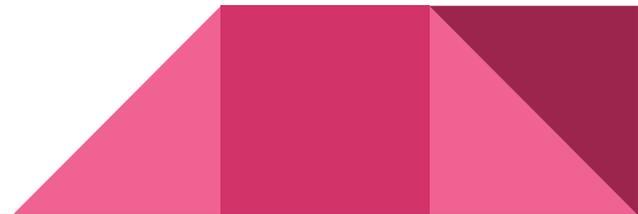
How does Git work

0. Tell git that you want to use version control in this folder, a.k.a. "make git repository"
 - `git init`
1. Make changes in files.
 - As usual, git is just watching.

```
very_good_cake.txt
1 Perfect cake
2 =====
3 Batter 1
4 -----
5 flour 2 cups
6 baking powder 1 pc
7 sugar 1 cup
8 unsweetened cocoa powder 3 tbsp
9 eggs 2
10 milk 1 cup
11 cooking oil 3/4 cup
```

How does Git work

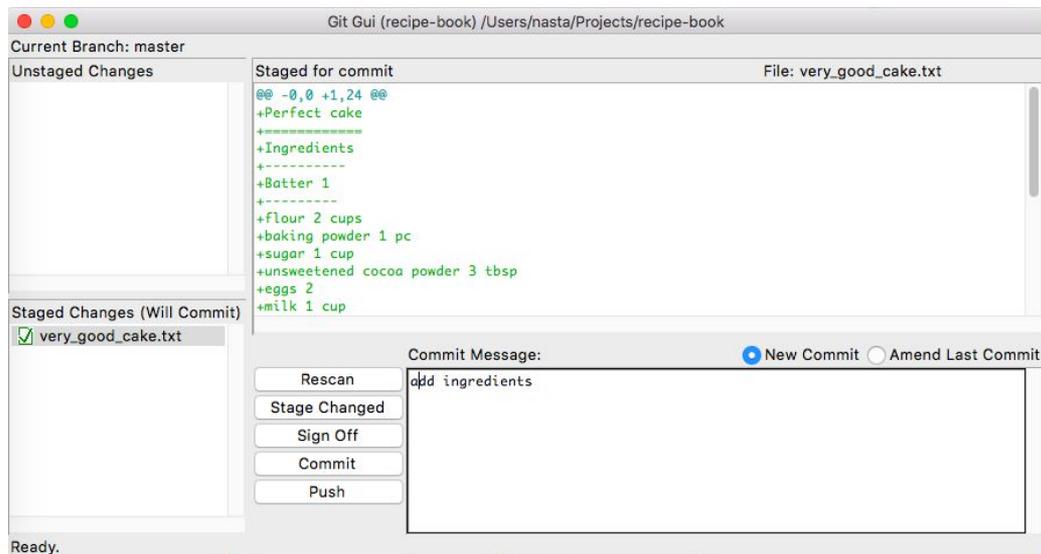
2. Prepare changes that will be saved as one commit.
 - `git add`



How does Git work

3. Save set of changes as commit.

- `git commit`
- I can return to this state whenever I want.
- Describe change!

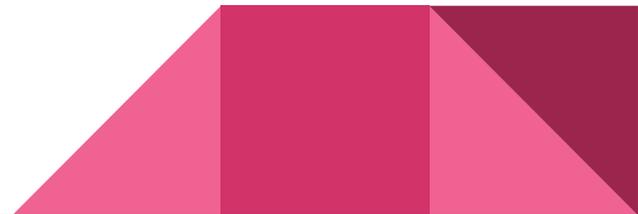


Git GUI and gutk

- Graphical user interface for git
 - `sudo apt-get install git-gui` (Ubuntu)
 - `brew install git` (MacOS)
 - If you do not have brew install it using tutorial [here](#)
 - Software to view branches
 - `sudo apt-get install gitk` (Ubuntu)
 - `brew install gitk` (MacOS)
 - GitBash for Win has also Git GUI
- 

Github/Gitlab

- Github/Gitlab are servers to store your versioned software.
- GUI for git.
- Project management tools.
- Create an [account](#).



List of changes

Branch: master ▾

Commits on Aug 11, 2018

add ingredients

 anastazie committed 5 minutes ago



faa45c6



create cake recipe file

 anastazie committed 7 minutes ago



1e352d3



git log

```
commit faa45c602cd188ca8a7d5ccb02249086428afa2e (
Author: Anastazie Sedlakova <nasta@sedlakovi.org>
Date: Sat Aug 11 17:56:01 2018 +0200
```

```
    add ingredients
```

```
commit 1e352d3e66698a0febed73fa7ed63158b61cf707
Author: Anastazie Sedlakova <nasta@sedlakovi.org>
Date: Sat Aug 11 17:54:24 2018 +0200
```

```
    create cake recipe file
```

Create informative commit messages (see [here](#))

	COMMENT	DATE
○	CREATED MAIN LOOP & TIMING CONTROL	14 HOURS AGO
○	ENABLED CONFIG FILE PARSING	9 HOURS AGO
○	MISC BUGFIXES	5 HOURS AGO
○	CODE ADDITIONS/EDITS	4 HOURS AGO
○	MORE CODE	4 HOURS AGO
○	HERE HAVE CODE	4 HOURS AGO
○	AAAAA	3 HOURS AGO
○	ADKFJSLKDFJSDKLFJ	3 HOURS AGO
○	MY HANDS ARE TYPING WORDS	2 HOURS AGO
○	HAAAAAAAAAANDS	2 HOURS AGO

AS A PROJECT DRAGS ON, MY GIT COMMIT MESSAGES GET LESS AND LESS INFORMATIVE.

Sharing changes

- By e-mail...
 - This is the way Linux is developed
 - Uncomfortable
- ... Or have changes on server
 - You don't want to host your own server
 - Services



[GitHub.com](https://github.com) (biggest one)



[GitLab.com](https://gitlab.com)

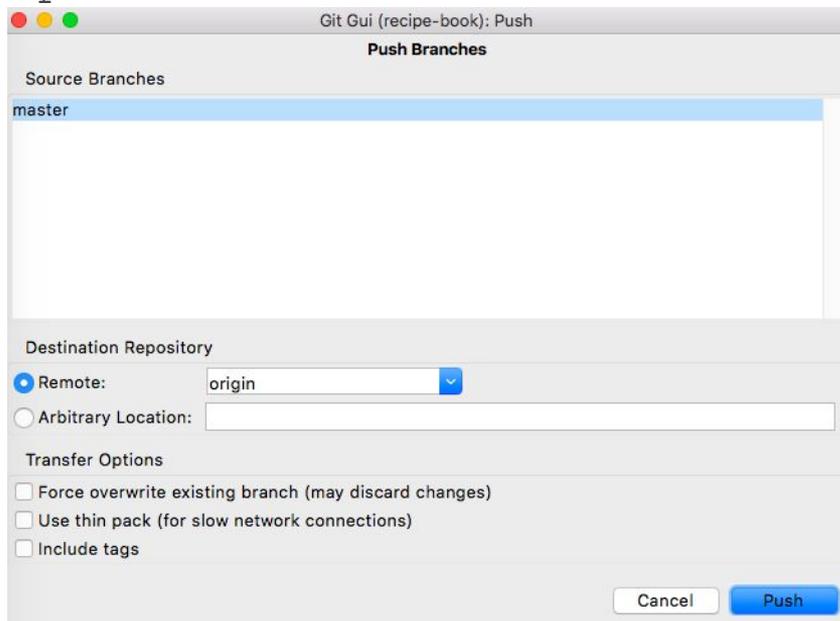


[Bitbucket.org](https://bitbucket.org)

How does Git work

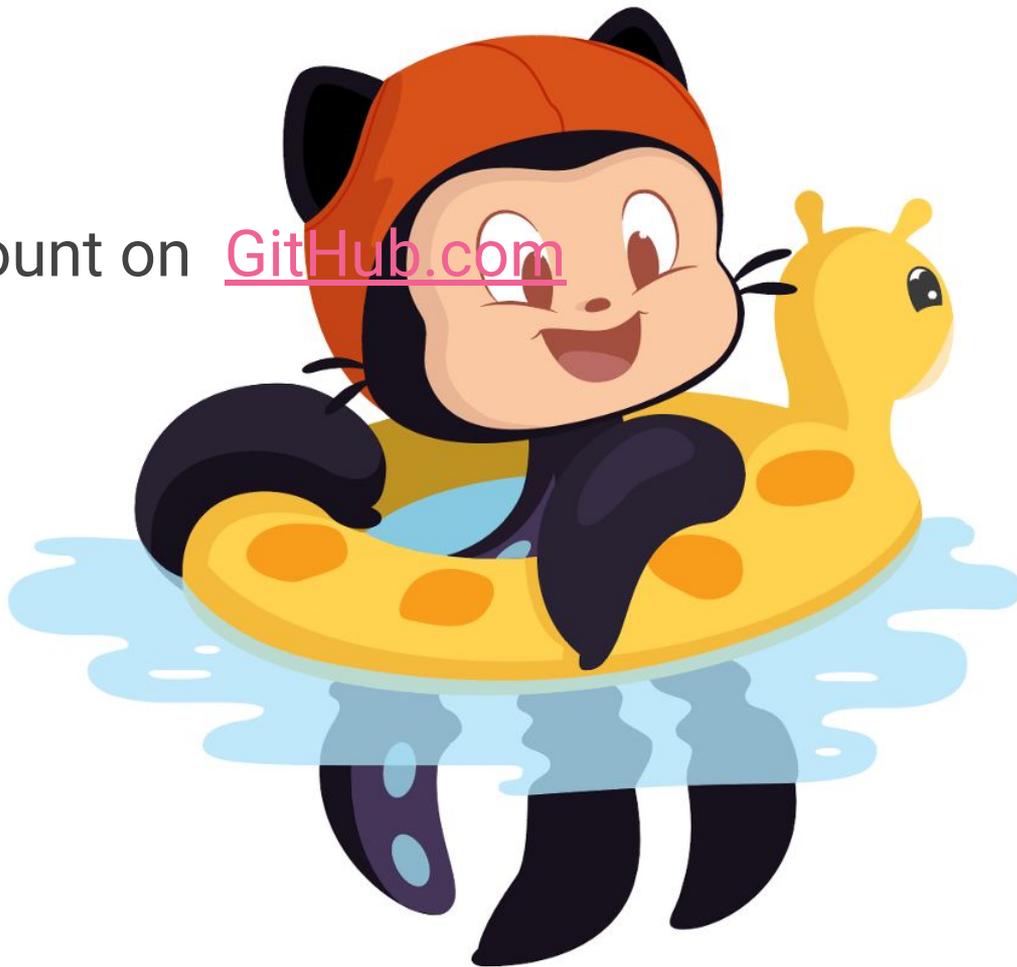
4. Push changes to remote server

- `git push`



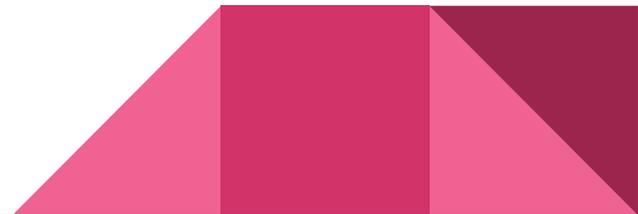
Exercise

- Create account on [GitHub.com](https://github.com)



Sharing changes using git

1. Creating project (called repository) at github.com
2. Linking local repository to the remote server
 - `git remote add origin git@github.com:uživatel/projekt.git`
3. Downloading changes
 - `git pull`
4. Sending my changes
 - only the ones I've already committed
 - `git push`



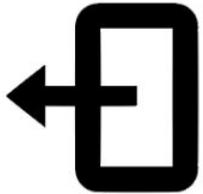
In case of fire



1. `git commit`



2. `git push`



3. leave building

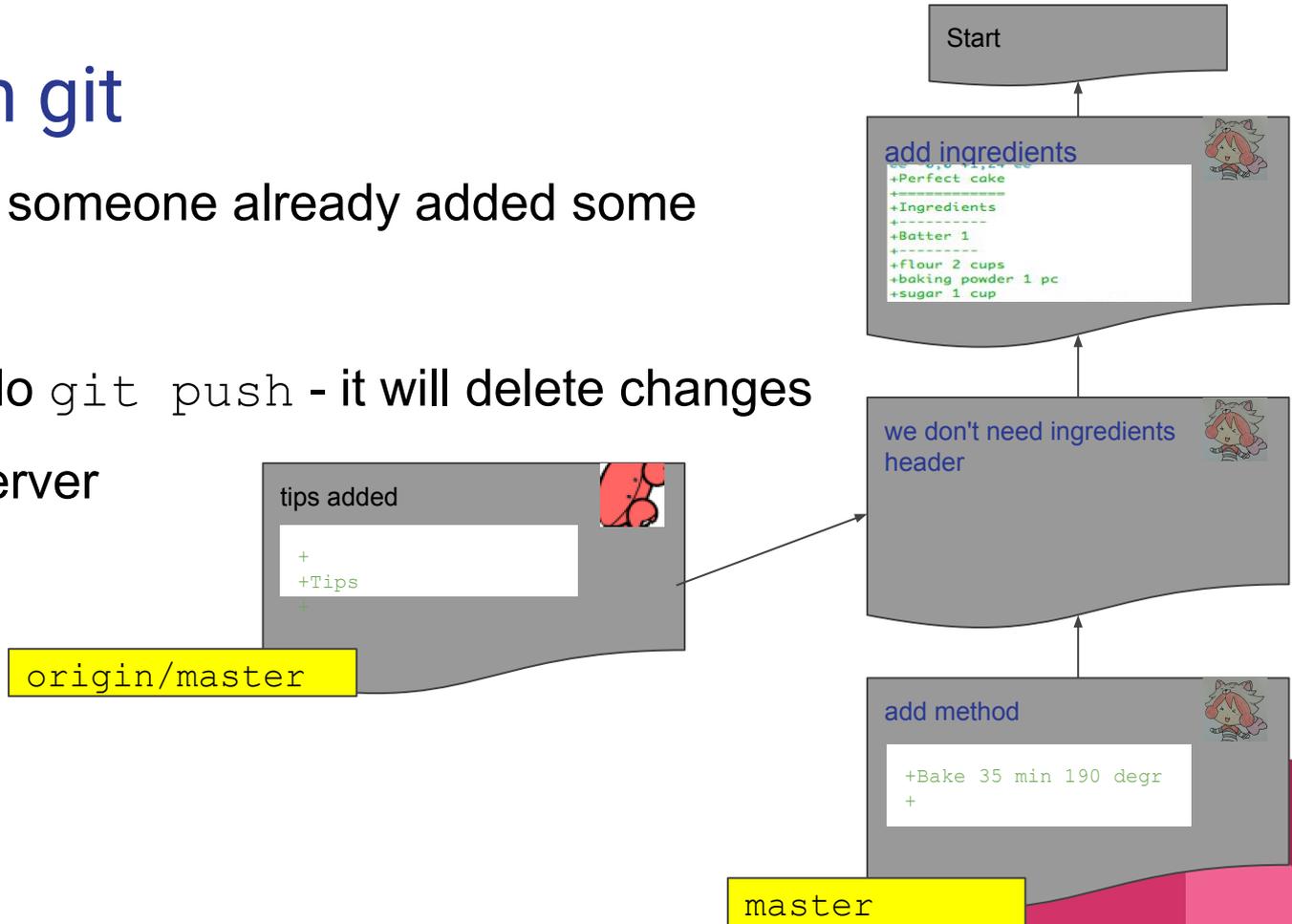
Branches in git

- I added one commit and did not yet push it to server yet
- Naming convention
 - Remote server - `origin`
 - Main branch - `master`



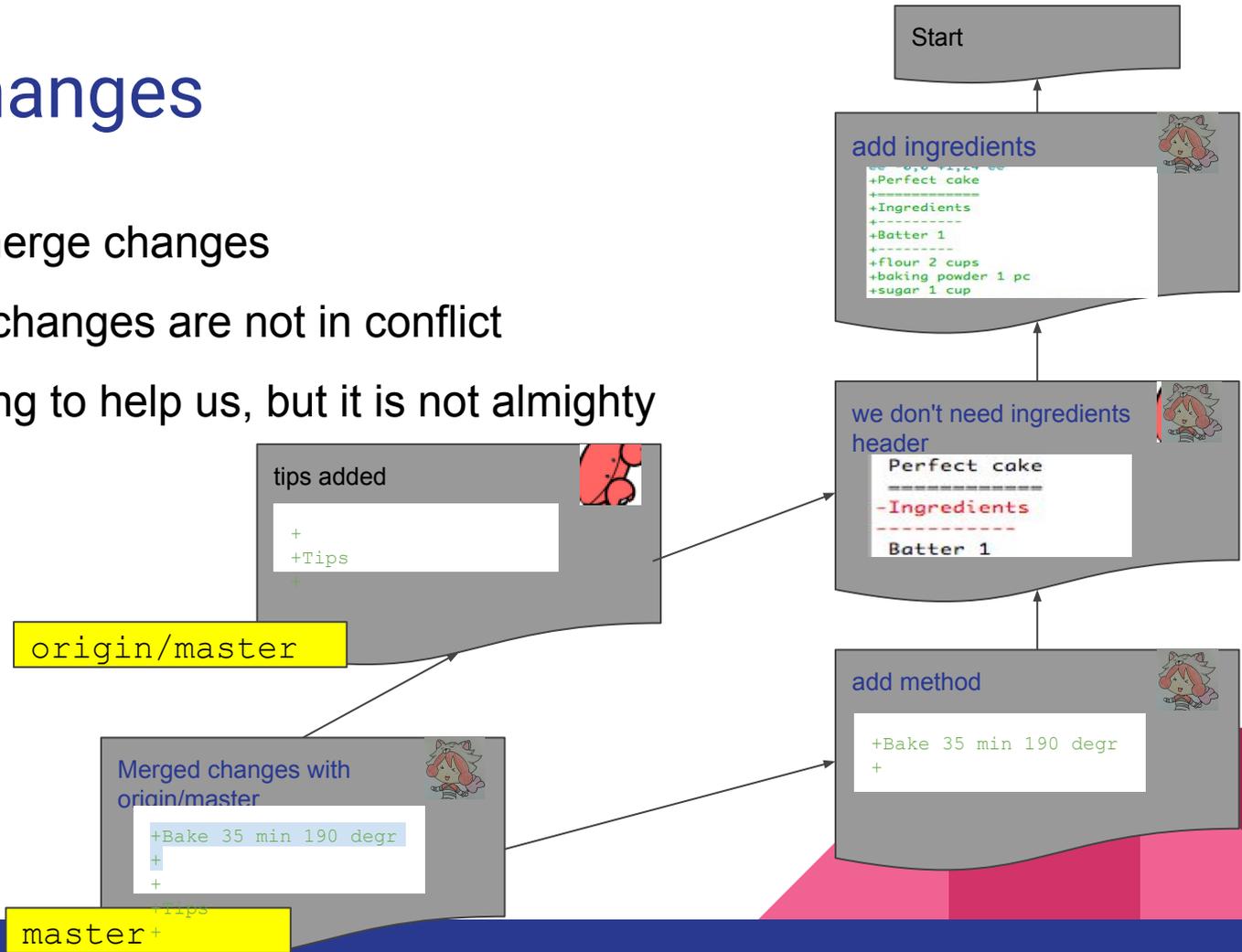
Branches in git

- In meantime someone already added some changes
- Now I can't do `git push` - it will delete changes on remote server



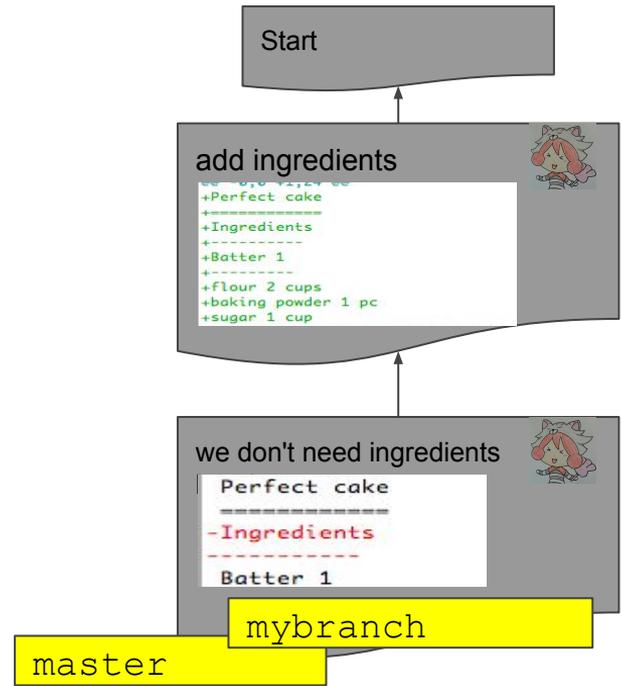
Merging changes

- We need to merge changes
 - Check if changes are not in conflict
 - Git is trying to help us, but it is not almighty



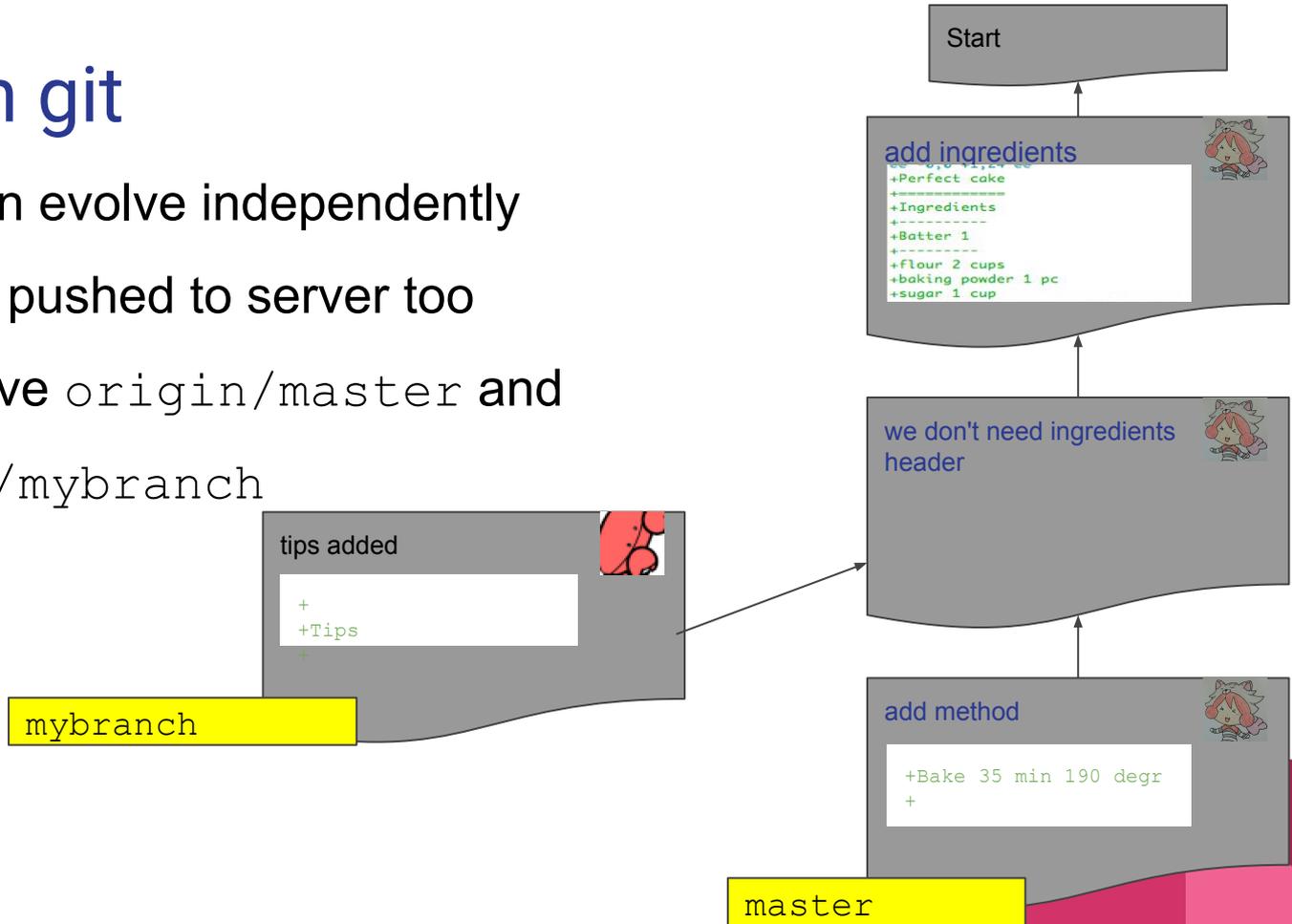
Branches in git

- Created a named branch using
`git checkout -b mybranch`
- Now the commit belongs
to two branches:
`git checkout master`
and `git checkout mybranch`
take me to the same commit.



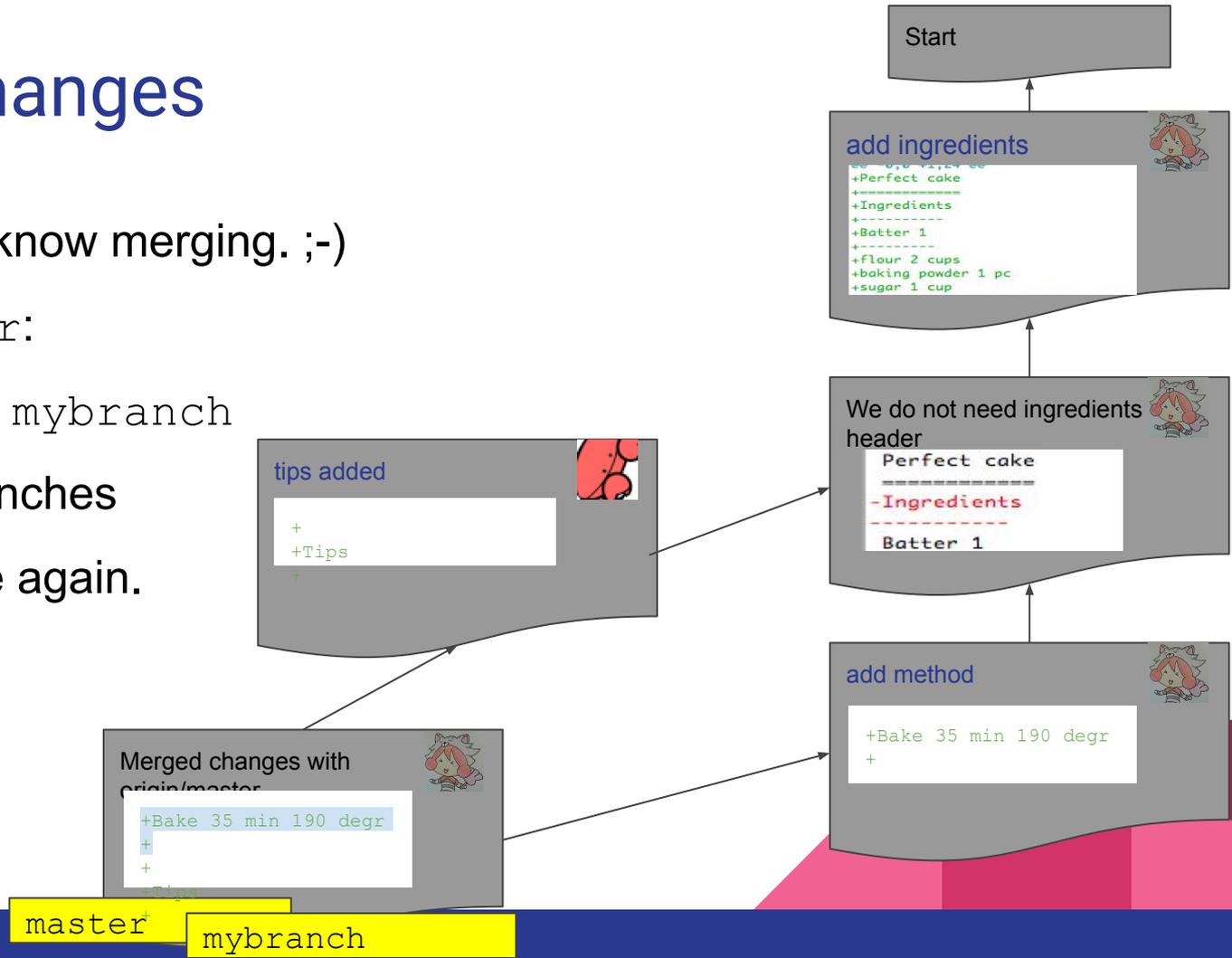
Branches in git

- Branches can evolve independently
- They can be pushed to server too
 - You'll have `origin/master` and `origin/mybranch`



Merging changes

- You already know merging. ;-)
- From master:
`git merge mybranch`
- Now, the branches are the same again.



Fork

- Your version of project of another person's project Github
- To fork, just use Fork button on Github
- Then download new project to your computer using

```
git clone https://github.com/YOUR-NAME/recipe-book
```

- New folder named `recipe-book` will be created in the current directory
- More information here:

<https://help.github.com/articles/cloning-a-repository/>

- Now you can make changes (commits)

```
git add file # Or using git gui
```

```
git commit
```

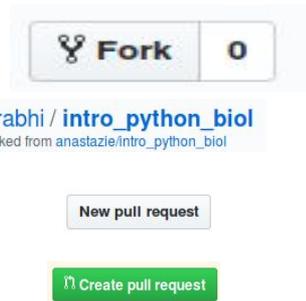
- Send your changes to Github

```
git push origin master
```



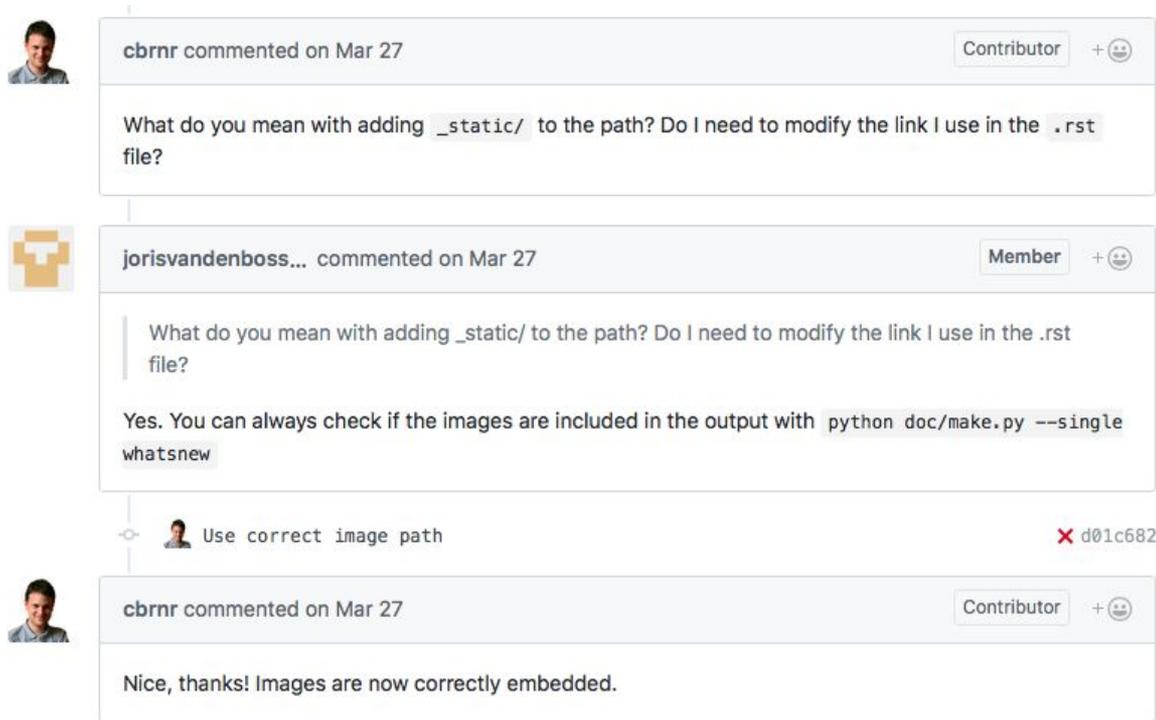
Review using GitHub/GitLab/Bitbucket/...

1. “Fork” project from other person
2. Make changes, git push to your project
3. Create pull request with commented changes



Review using GitHub/GitLab/Bitbucket/...

4. Discussion started



The screenshot shows a GitHub discussion thread with three messages. The first message is from user 'cbrnr' (Contributor) asking about adding `_static/` to a path. The second message is from 'jorisvandenboss...' (Member) providing a solution with the command `python doc/make.py --single whatsnew`. The third message is from 'cbrnr' (Contributor) thanking the user. A commit link `d01c682` is also visible.

 **cbrnr** commented on Mar 27 Contributor + 😊

What do you mean with adding `_static/` to the path? Do I need to modify the link I use in the `.rst` file?

 **jorisvandenboss...** commented on Mar 27 Member + 😊

What do you mean with adding `_static/` to the path? Do I need to modify the link I use in the `.rst` file?

Yes. You can always check if the images are included in the output with `python doc/make.py --single whatsnew`

 **cbrnr** commented on Mar 27 Contributor + 😊

Nice, thanks! Images are now correctly embedded.

[Use correct image path](#) ✖ d01c682

Review using GitHub/GitLab/Bitbucket/...

5. Person reviewing pull request is happy, code is merged to master

Merge pull request

Set `pd.options.display.max_columns=0` by default #17023

Merged jorismvandenboss... merged 21 commits into pandas-dev:master from cbrnr:nicer_display_defaults on Mar 28

Conversation 138

Commits 21

Checks 0

Files changed 10

+117 -53



jorismvandenboss... commented on Mar 28

Member



@cbrnr Thanks a lot for this (and for your patience getting this merged :))



1

Recapitulation

```
# make changes in files
```

```
git status
```

```
# download all changes from server
```

```
git pull origin master
```

```
# use git gui OR
```

```
git add filename # add file to commit
```

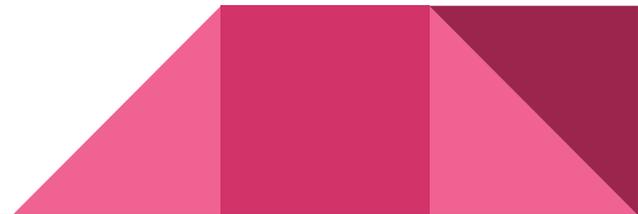
```
git commit -m 'fix name' # write commit  
message
```

```
git push origin master # push changes to
```

```
remote server
```

Exercise

- Fork repository [recipe-book](#)
- Clone repository to your computer
- Add your favourite recipe
- Make Pull Request with your recipe



Merge conflict!

Two people made different changes in the same place.

Person, who will be resolving conflicts need to decide whether:

- Add both versions
- Choose one
- Combine versions somehow



Creating profile page on github

1. Go to Github
2. Create a repository in the format <username>.github.io
3. Follow this [tutorial](#)

