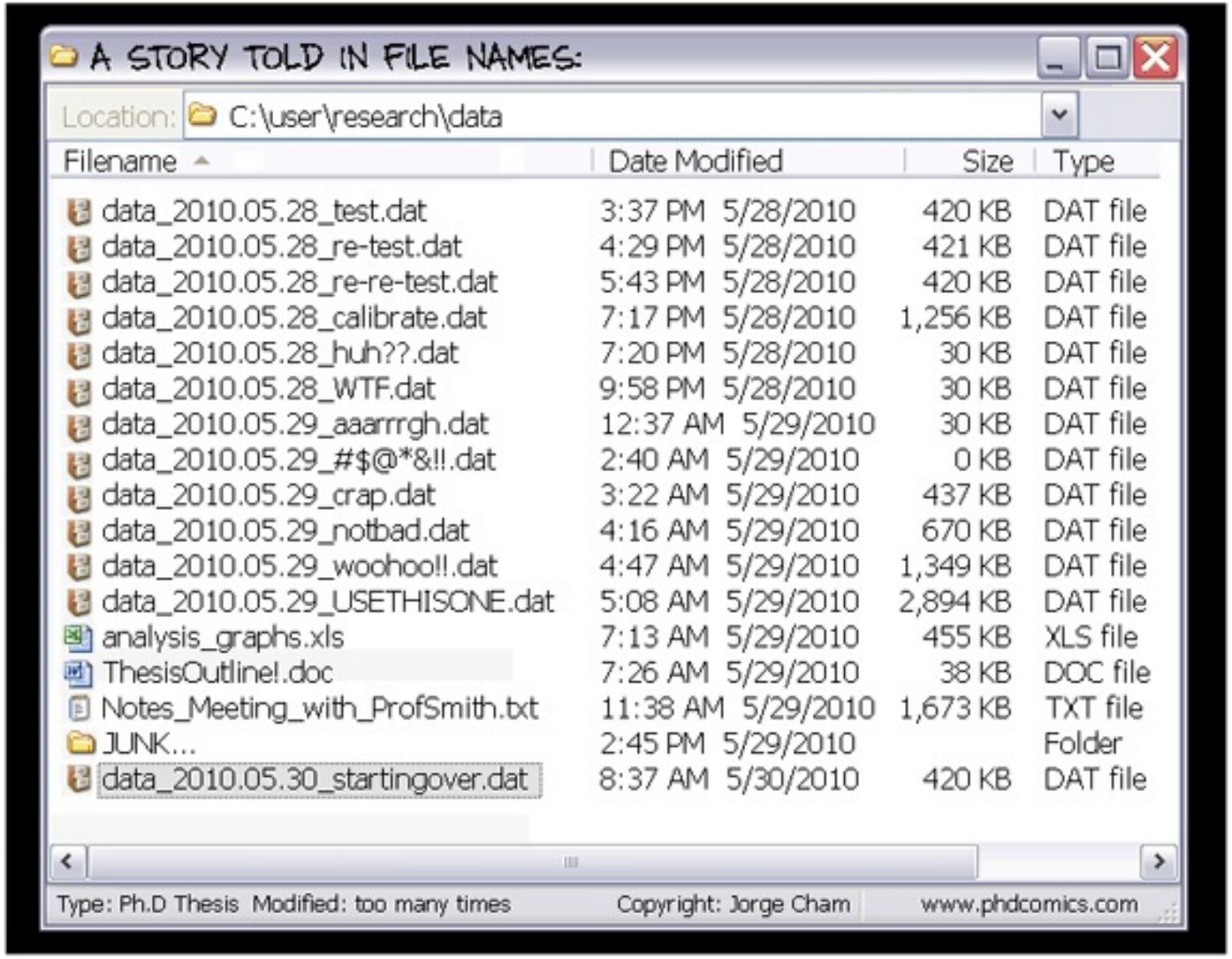
Version Control with Git



Eloise A Marais

Group Meeting

2 December 2021

Warning

I’m mostly a self-taught Git user (with some initial guidance from Bob Yantosca)!

I get by with a handful of commands and functions and Google searchers.

Benefits of Good Version Control?

- Easier to collaborate with others using the same code
- Easier to apply bug fixes
- Easier to add code patches to add or update a feature in your code
- Allows you to work in a separate code development space without impacting the original code that works
- Allows you to revert your own code developments if you're using a complex code like GEOS-Chem if you discover that your updates don't improve the model
- Others are: traceability, history, identity (credit for authorship), better management, efficient
- Nothing is permanent. You can revert any changes applied using git, so don't fear experimenting.

Roam around in GEOS-Chem GitHub page:

GEOS-Chem Git Hub is divided into a CodeBase: https://github.com/geoschem/geos-chem
And SuperProjects:

- GCClassic: https://github.com/geoschem/GCClassic
- GCHP: https://github.com/geoschem/GCHP
- HEMCO: https://github.com/geoschem/HEMCO

Some relevant features associated with the GEOS-Chem GitHub page:

- Root page: https://github.com/geoschem/geos-chem
- Code branches: https://github.com/geoschem/geos-chem/branches
- Pull requests: https://github.com/geoschem/geos-chem/pulls
- Milestones: https://github.com/geoschem/geos-chem/milestones
- Link to source code in the Green ”Code” button drop down menu (use to clone the source code)
- Issues reporting page: https://github.com/geoschem/geos-chem/issues
- Discussions page: https://github.com/geoschem/geos-chem/discussions
- Releases of GCClassic (version of the model we use) with links to Zenodo page with unique DOI: https://github.com/geoschem/GCClassic [under Releases]

With very simple python code, we're going to:

- Clone a repository
- Create and work on code changes in a separate branch
- Stage and commit changes. Merge changes to the main/master branch
- Report a source code issue using the Issues tab in GitHub
- Push changes to the remote repository
- Pull changes from the remote repository
- Create a patch
- Add a patch to existing code

Clone remote repository and address coding bug

Clone a remote repository with very simple python scripts (source code). This creates a local repository copy. To do this, enter the following at the command line to create a new directory called “get_test”:
[ ]$ git clone https://github.com/eamarais/git_test

Alternately, you can name the directory something else using this command:
[ ]$ git clone https://github.com/eamarais/git_test git_test_scripts

Move to the new directory:
[ ]$ cd git_test

Open the file with your preferred text editor (emacs in example below):
[ ]$ emacs hello_world.py &

Open git GUI and gik:
[ ]$ git gui &
[ ]$ gitk &

We’re going to address the typo by creating a new branch, editing the code in that branch, merging the changes to the main branch, reporting the issue on GitHub, pushing the change to the remote repository, updating your local code to the latest version on the remote repository.

Create a patch of your code changes

Open the file with your preferred text editor (emacs in example below):
[ ]$ emacs sys_exit.py &

If you run this python script using “python sys_exit.py”, you will get the following error, as the sys package hasn’t been imported:

```
Traceback (most recent call last):
  File "sys_exit.py", line 14, in <module>
    sys.exit(1)
NameError: name 'sys' is not defined
```

Create a branch with a clear name, reopen the file under this new branch, address the error by adding “import sys” below or above “import os”. Rerun the script to confirm it works. Stage and commit the changes, move back to your main/master branch, merge with main branch.

Find the commit number you’d like to apply using either “git log” at the command line or open “gitk &”. The commit number will be something like “670adf54835cf4477e653d8793900361e08428bd”.

Create a patch using “git format-patch” at the command line to output the patch to a dedicated file:
[ ]$ git format-patch *branch_name* -1 *commit_number* --stdout > fix_sys_err.patch

Open the fix_sys_err.patch file to see what a simple patch looks like.

Apply a patch to your code

Apply patch bugfix.patch to make_array.py.

Open and inspect bugfix.patch code:
[ ]$ emacs bugfix.patch &

Create new branch:
[ ]$ git branch apply_patch
[ ]$ git checkout apply_patch
(or do both steps at once: git checkout -b apply_patch)

Add patch to new branch:
[ ]$ git am bugfix.patch

Check added:
[ ]$ git log
(or open git gui to check added)

Merge apply_patch branch with main branch:
[ ]$ git checkout main
[ ]$ git merge apply_patch
(check merge with gitk)

Ways to merge the patch to avoid conflicts: https://newbedev.com/when-applying-a-patch-is-there-any-way-to-resolve-conflicts

If Time Permits …

Follow Optional Steps 2, 5, 6 on the GEOS-Chem wiki:
http://wiki.seas.harvard.edu/geos-chem/index.php/Downloading_GEOS-Chem_source_code_(13.0.0_and_later_versions)

Other Git-Related Information

- Git format-patch command: https://git-scm.com/docs/git-format-patch
- Semantic Versioning (used by GEOS-Chem and HEMCO): https://semver.org/
- Other features of GitHub: https://www.hongkiat.com/blog/github-overlooked-features/
- Linking GitHub and Zenodo accounts: https://docs.github.com/en/repositories/archiving-a-github-repository/referencing-and-citing-content
- GitLab: https://about.gitlab.com/devops-tools/github-vs-gitlab/
- GitHub Pages to host project page or professional website: https://pages.github.com/, https://gist.github.com/TylerFisher/6127328
- Got Tower: https://www.git-tower.com/ (alternate GUI software to collaborate)

Other Training Resources

- YouTube video by YouTube video by Bob Yantosca from GEOS-Chem Support Team: https://www.youtube.com/watch?v=1fhI-HObyV4
- Tutorials, news, tips: https://www.atlassian.com/git
- UCL Git user manual: http://github-pages.ucl.ac.uk/rsd-engineeringcourse/ch02git/
- Turing guide: https://the-turing-way.netlify.app/reproducible-research/vcs.html
- Software carpentry courses: https://software-carpentry.org/