# Introduction to Revision Control Systems

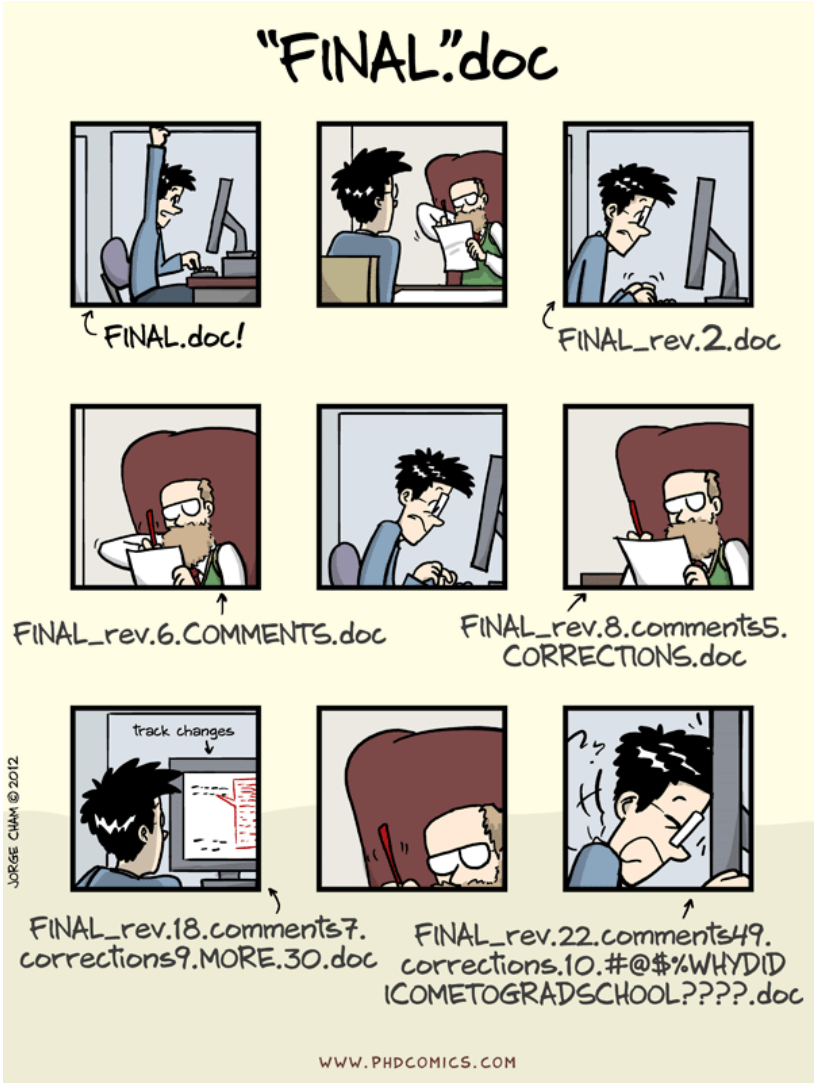
Dr. Gowtham, PhD

Director of Research Computing, IT
Research Associate Professor, College of Computing
Michigan Technological University

g@mtu.edu

2022-04-05

# The Why Evolution of most of our documents




https://phdcomics.com/comics/archive.php?comicid=1531

# The Why My PhD dissertation had 48 different folders

sgowtham@feynman:~/research/Dissertation/20080122.color

File Edit View Search Terminal Help

```
[sgowtham@feynman Dissertation]$ ls
20070924.0  20071025.0    20071123.0  20071203.0    20071216.0  20080114.0
20070924.1  20071026.000  20071124.0  20071204.0    20071217.0  20080114.bw
20070925.0  20071030.0    20071125.0  20071205.000  20071218.0  20080114.color
20070927.0  20071030.1    20071126.0  20071211.0    20071219.0  20080121.bw
20070928.0  20071119.0    20071127.0  20071211.1    20071220.0  20080121.color
20071002.0  20071120.0    20071128.0  20071212.0    20071220.1  20080122.bw
20071022.0  20071121.0    20071129    20071213.0    20080107.0  20080122.color
20071023.0  20071122.0    20071129.0  20071214.0    20080109.0
[sgowtham@feynman Dissertation]$ cd 20080122.color
[sgowtham@feynman 20080122.color]$ ls
Abstract.tex           Chapter6.tex              MTUPhDThesis.sty
Abstract.txt           Chapter7.bib              MTUPhDThesis.sty.0
Acknowledgements.tex   Chapter7.tex              MyThesis.bib
Appendix.tex           Dedication.tex            MyThesis.dvi
Beowulf_Cluster.bib    Future_Work.bib           MyThesis.pdf
Beowulf_Cluster.tex    Future_Work.tex           MyThesis.tex
Bibliography.tex       Graphs                    Nano_Bio_Physics.bib
Chapter1.bib           Images                    Nano_Bio_Physics.tex
Chapter1.tex           Index.tex                 nextpage.sty
Chapter2.bib           Introduction.bib          PublishedPapers
Chapter2.tex           Introduction.tex          README.PLEASE
Chapter3.bib           ListOfFigures.tex         TableOfContents.tex
Chapter3.tex           ListOfPublications.bib    Theoretical_Details.bib
Chapter4.bib           ListOfPublications.tex    Theoretical_Details.tex
Chapter4.tex           ListOfTables.tex          TOC.pdf
Chapter5.bib           Makefile                  TOC.tex
Chapter5.tex           Metal_Oxide_Clusters.bib
Chapter6.bib           Metal_Oxide_Clusters.tex
[sgowtham@feynman 20080122.color]$
```

# The How My introduction to formal revision control system

* Physics major (or the `X` in `X + CS`)
* Had no idea that formal revision control systems even existed
* The letters, `c` and `v`, are next to each other in a keyboard
* A typo when attempting to open a file using `vi FILENAME`

  `ci FILENAME`
* Seemingly steep learning curve and too close to graduation
* Didn't make the time to learn it and paid a pretty high price

`ci` was the command to check in the document in RCS/CVS

# The How Working of a revision control system

* Start with a base version of the document
* Record the changes made each step of the way (i.e., a `commit`)
* `commit`ing can be treated as a higher form of saving the file
* `commit`s can be rewinded and played back - unlimited undo and redo

Image courtesy: Software Carpentry
A `commit` is analogous to a checkpoint in a journey or an anchor in rock climbing

## Centralized

RCS, CVS and SVN

## Distributed

Bazaar, Darcs, Git and Mercurial

# The Setup GitHub.com

* Visit GitHub's website

  https://github.com

* Open an account (it's free) using Michigan Tech email address

  Check Michigan Tech email and complete the verification process

* Update the profile

  https://github.com/settings/profile

* Add the SSH keys from the local computer

  https://github.com/settings/keys

# The Setup Configuring idenity in a local computer

## Git configuration

```
# Replace NAME with our real or preferred name, and
# EMAIL with our email address used in GitHub.com.
#
# Note that name and email address will be associated
# with every single commit we make to the repository
# and will show up in the log as such.

git config --global user.name "NAME"
git config --global user.email "EMAIL"
git config --global core.editor vim
git config --global core.pager less
git config --global color.ui true
git config --list
```

# The Repository Creating in GitHub.com

* Visit GitHub's website

  https://github.com/new

* Select the owner from the drop down list

* Provide a repository name (needs to be unique)

  `MTUCoCRCS`

* Add a description (optional but helpful)

  `A test repository for 2022 MTU Computing Showcase`

* Select `Public` (or `Private`)

* Select other options as necessary

# The Repository Cloning it in a local computer

* Create a meaningfully named workspace

```
mkdir -p ${HOME}/git_work
cd ${HOME}/git_work
```

* Clone the `MTUCoCRCS` repository

```
git clone git@github.com:USERNAME/MTUCoCRCS.git
```

* Verify that the local copy is up to date

```
cd ${HOME}/git_work/MTUCoCRCS
pwd
ls
git pull
git status
```

Replace `USERNAME` with our GitHub.com username

# The Repository Adding a file

* Change into the workspace

```
cd ${HOME}/git_work/MTUCoCRCS
pwd && ls && git pull && git status
```

* Create a file

```
touch HelloWorld.txt
git status
```

* Add the file to the repository and commit

```
git add HelloWorld.txt
git status
git commit -m "Added HelloWorld.txt (base version)"
git status
git push origin main
```

# The Repository Updating a file

* Change into the workspace

```
cd ${HOME}/git_work/MTUCoCRCS
pwd && ls && git pull && git status
```

* Update the file

```
echo "Hello, World" > HelloWorld.txt
git status
```

* Add the updated file to the repository and commit

```
git add HelloWorld.txt
git status
git commit -m "Updated HelloWorld.txt"
git status
git push origin main
```

# The Repository

## Adding/Updating a file - schematic representation

* `git add FILENAME` puts the file in the staging area
* `git commit` locally saves the staged content as a new commit
* `git push` sends the new commmit to GitHub.com

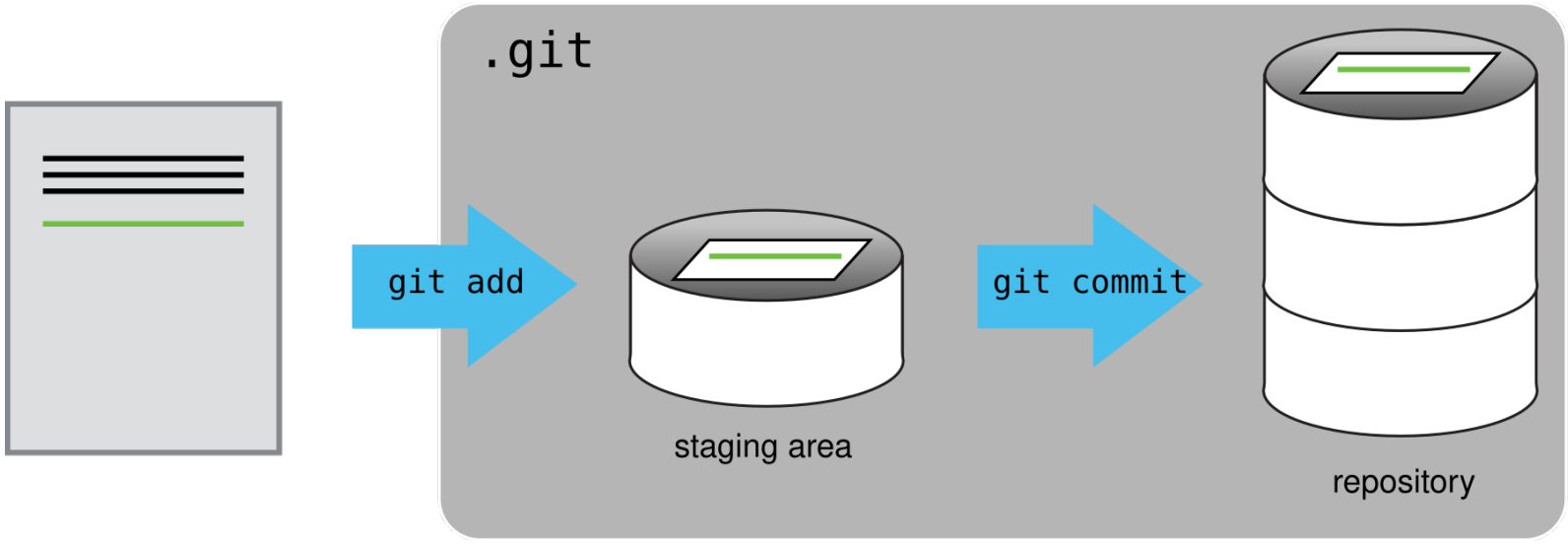



Image courtesy: Software Carpentry

# The Repository Deleting a file

* Change into the workspace

```
cd ${HOME}/git_work/MTUCoCRCS
pwd && ls && git pull && git status
```

* Delete the file from the repository and commit

```
git rm HelloWorld.txt
git status
git commit -m "Deleted HelloWorld.txt"
git status
git push origin main
```

Deleting a file from the repository doesn't necessarily delete it from all previous commits.

# The Repository Tags

* Specific points in the history of a repository deemed important
* Change into the workspace

```
cd ${HOME}/git_work/MTUCoCRCS
pwd && ls && git pull && git status
```

* Create a tag and commit

```
git tag
git tag -a v1.0 -m "Our first tag/release"
git push origin v1.0
```

# The Repository History

* Change into the workspace

```
cd ${HOME}/git_work/MTUCoCRCS
pwd && ls && git pull && git status
```

* Check the log

```
git log
```

* `gource` is a free tool that can visualize the log

  https://gource.io/

# The Repository Advanced topics

* Branching and merging
  * `main` branch is usually write-protected
  * May need testing and approval prior to merging
  * Tests can be automated
* (Humane and compassionate) Conflict resolution
  * Common occurence in collaborative projects
* Pull requests
  * Common practice in community/open source projects

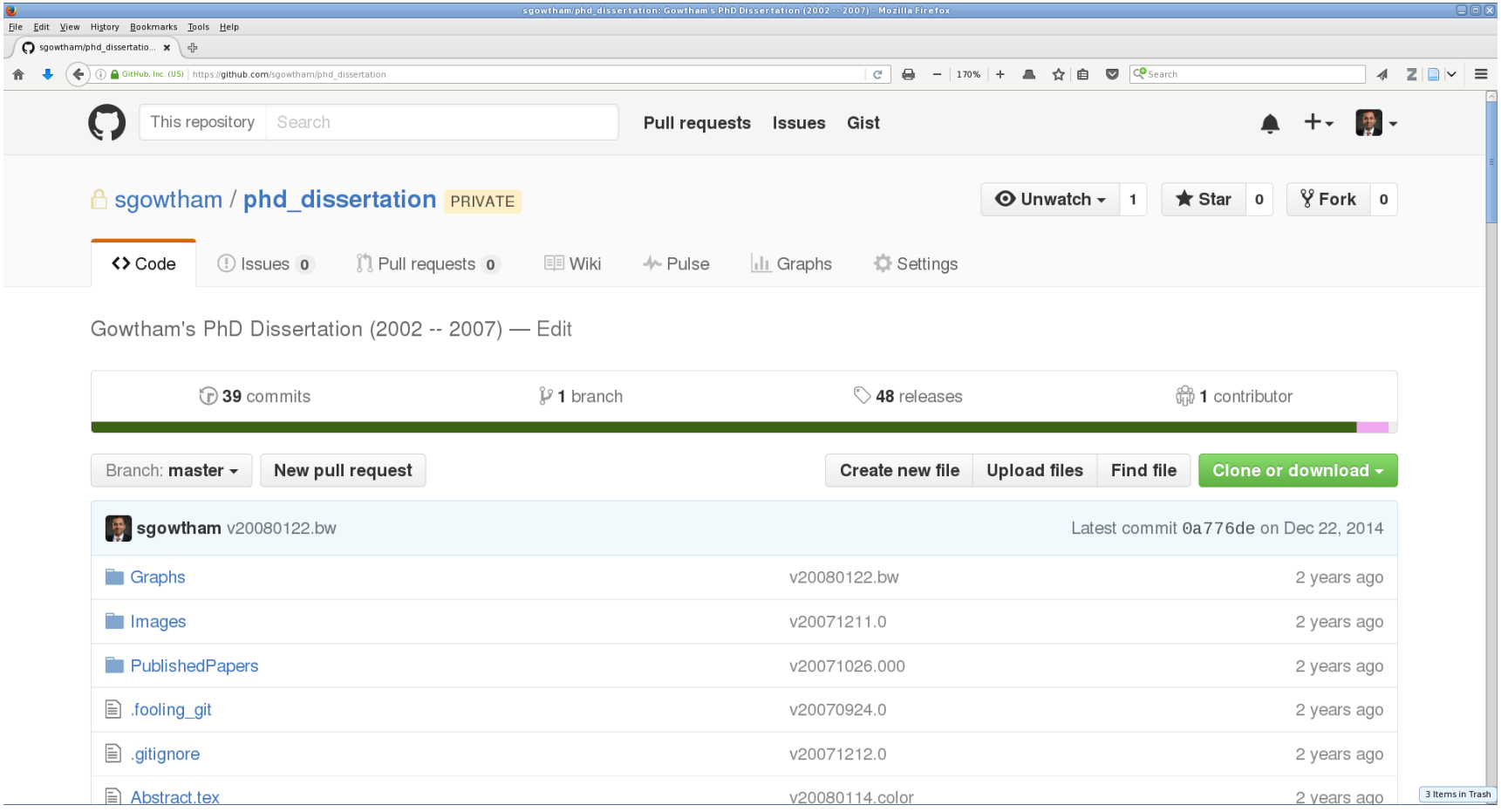
# Potential Applications

* Document preparation
  * Proposals for funding
  * Manuscripts for publications
  * Presentations for conferences and workshops
* Event planning (love requires revision too!)
  * Invitations
  * Guest list and seating arrangements
  * Thank You notes
* Systems administration

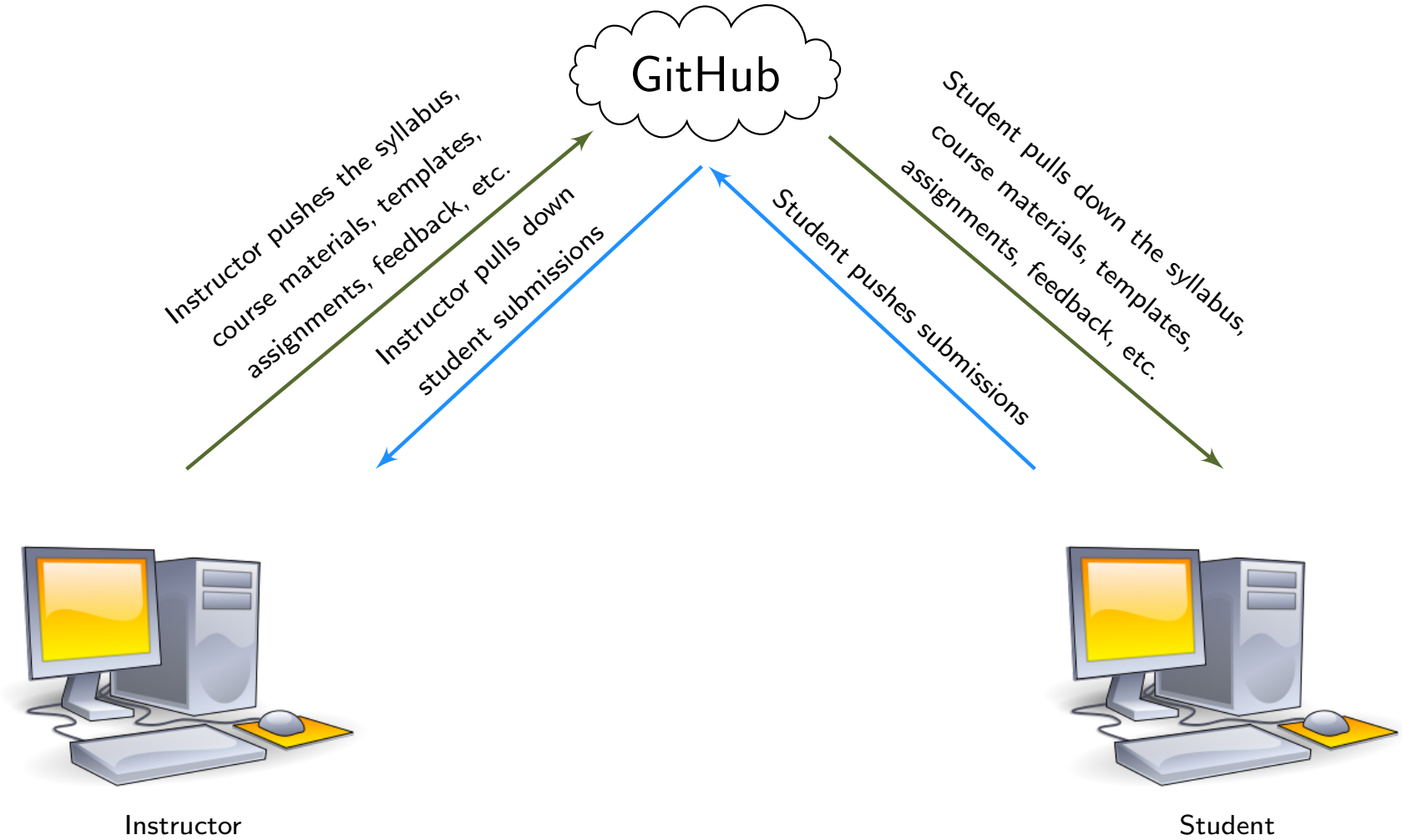


# Potential Applications

My PhD dissertation could have been a repository with 48 commits



# Potential Applications UN5390: Scientific Computing



# Got questions?

Please feel free to contact me, and I'll do my best to answer them in a timely fashion.

g@mtu.edu