Git collaboration best practices

Quentin D. Read

5/15/2020

Goals for today

- Discuss the rationale behind using version control and a few possible use cases
- Share experiences, good and bad
- Explore using Git integration in RStudio
- Show some examples from Quentin's work and from the SESYNC-CI GitHub

Why use it? or better, why not use it?

- Everyone uses version control – this is a way to formalize it

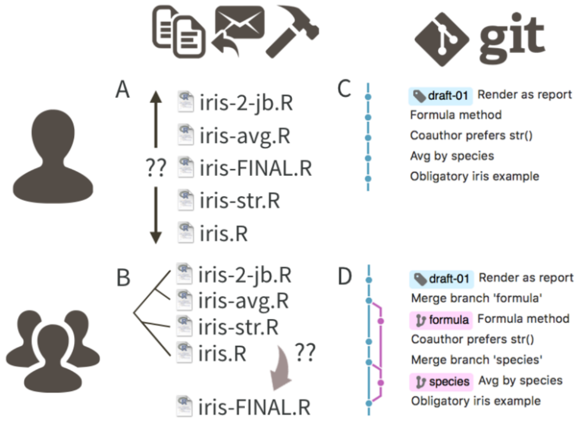


Figure 1: A: Solo work with DIY version control via filename. B: Collaborative work with DIY version control. C: Solo work with Git. D: Collaborative work with Git.

Why use it? or better, why not use it?

- Project management
- It is your project's web presence, and part of your personal web presence
- In some fields it is almost an expectation that you will have an active GitHub site
- Part of a suite of open science tools

Arguments against

- It does take some buy-in from everyone participating
- (perceived to have) a steep learning curve

Stories?

- Does anyone have any good stories or horror stories?

Use cases

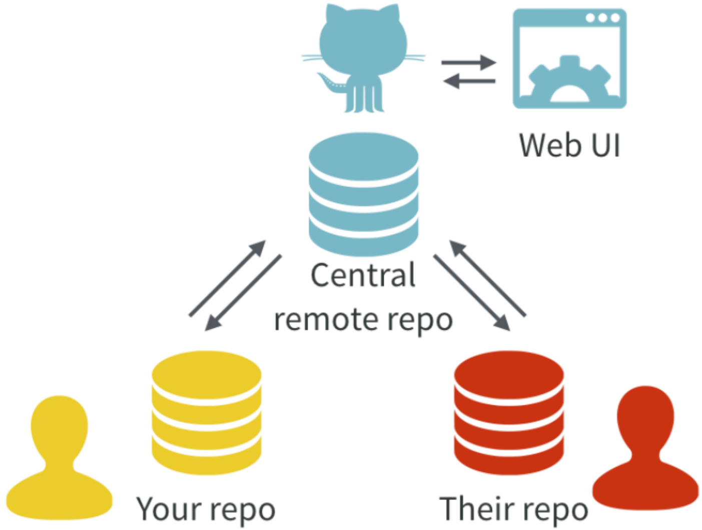
During

- use it solo to organize your work
- multiple team members can contribute to a repository simultaneously

After

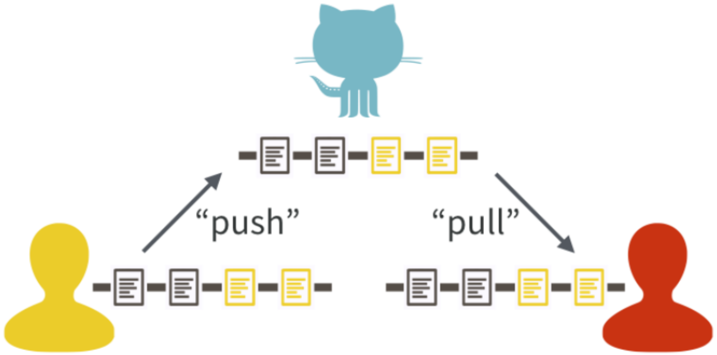
- publish a repository to the web (for open/reproducible science)
  - *Note*: this may require ongoing maintenance!

How it works

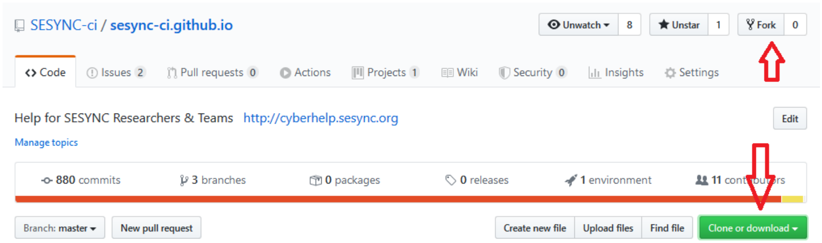


- git vs GitHub vs GitLab?
- What's a repo(sitory)?

Pulling and pushing



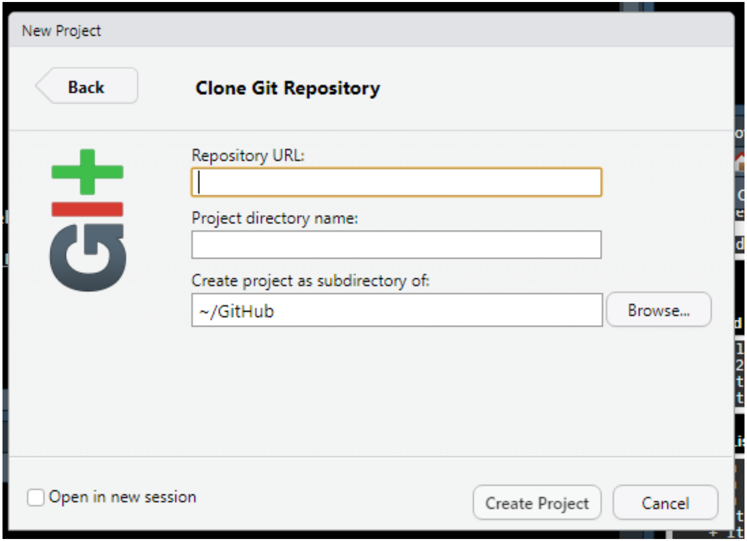
Forking and cloning



Help for SESYNC Researchers & Teams http://cyberhelp.sesync.org

Git in RStudio: creating a project

- RStudio uses "projects" to organize your work
  - projects correspond to repos


New Project

Back

Clone Git Repository

Repository URL:

Project directory name:

Create project as subdirectory of:
~/GitHub

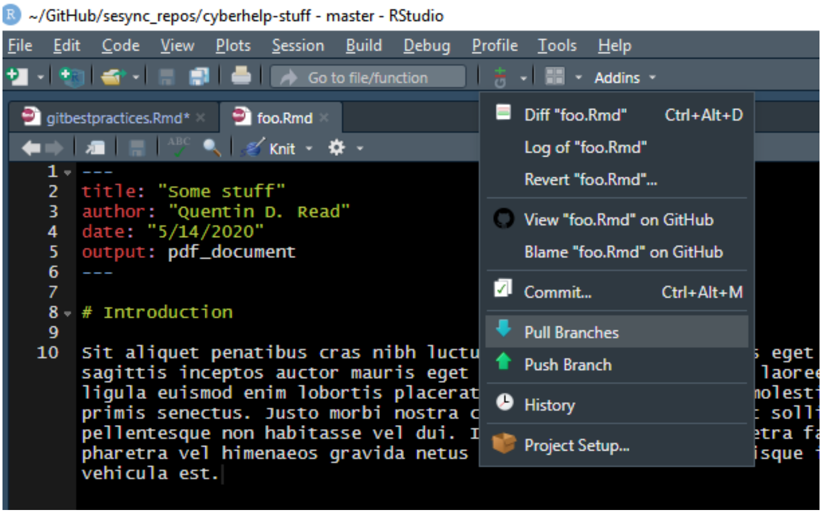
Browse...

Open in new session

Create Project

Cancel

Git in RStudio: pull before new work!

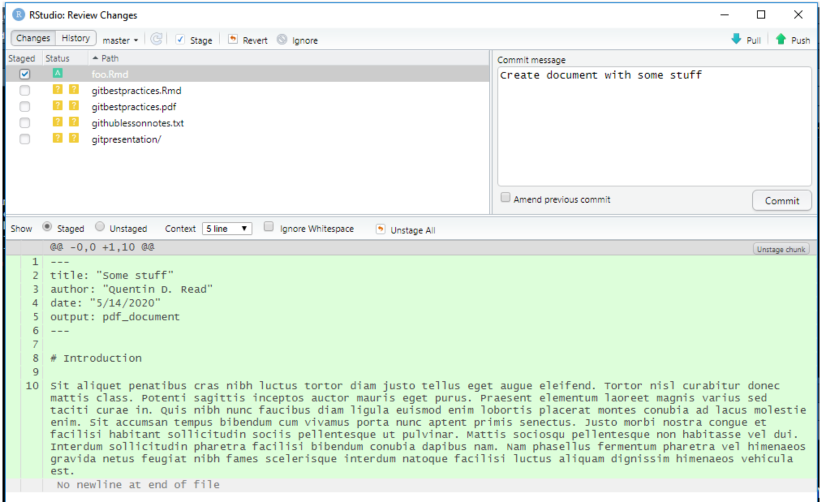


```
---
title: "Some stuff"
author: "Quentin D. Read"
date: "5/14/2020"
output: pdf_document
---

# Introduction

Sit aliquet penatibus cras nibh luctu                                      s eget
sagittis inceptos auctor mauris eget                                        laoree
ligula euismod enim lobortis placerat                                      molesti
primis senectus. Justo morbi nostra c                                      t solli
pellentesque non habitasse vel dui. I                                      etra fa
pharetra vel himenaeos gravida netus                                       isque 
vehicula est.
```

Git in RStudio: committing changes



Staged	Status	Path
✓	A	foo.Rmd
	? ?	gitbestpractices.Rmd
	? ?	gitbestpractices.pdf
	? ?	githublessonnotes.txt
	? ?	gitpresentation/

Commit message

```
Create document with some stuff
```

Amend previous commit

Show [✓ Staged] Unstaged

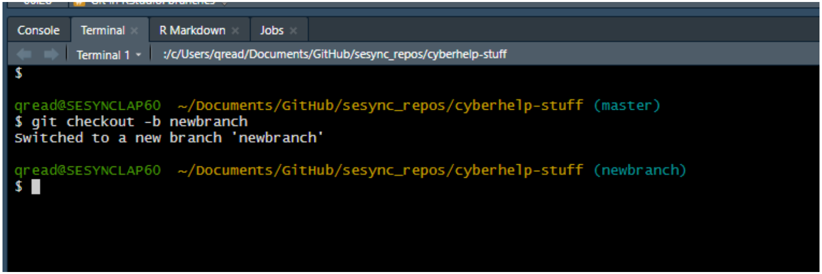
```
@@ -0,0 +1,10 @@
---
title: "Some stuff"
author: "Quentin D. Read"
date: "5/14/2020"
output: pdf_document
---

# Introduction

Sit aliquet penatibus cras nibh luctus tortor diam justo tellus eget augue eleifend. Tortor nisl curabitur donec
mattis class. Potenti sagittis inceptos auctor mauris eget purus. Praesent elementum laoreet magnis varius sed
taciti curae in. Quis nibh nunc faucibus diam ligula euismod enim lobortis placerat montes conubia ad lacus molestie
enim. Sit accumsan tempus bibendum cum vivamus porta nunc aptent primis senectus. Justo morbi nostra congue et
facilisi habitant sollicitudin sociis pellentesque ut pulvinar. Mattis sociosqu pellentesque non habitasse vel dui.
Interdum sollicitudin pharetra facilisi bibendum conubia dapibus nam. Nam phasellus fermentum pharetra vel himenaeos
gravida netus feugiat nibh fames scelerisque interdum natoque facilisi luctus aliquam dignissim himenaeos vehicula
est.
 No newline at end of file
```

Git in RStudio: branches

- Creating new branches must be done through the terminal or on the GitHub website


Console
Terminal
R Markdown
Jobs
Terminal 1
:/c/Users/qread/Documents/GitHub/sesync_repos/cyberhelp-stuff

```
$

qread@SESYNCLAP60  ~/Documents/GitHub/sesync_repos/cyberhelp-stuff (master)
$ git checkout -b newbranch
Switched to a new branch 'newbranch'

qread@SESYNCLAP60  ~/Documents/GitHub/sesync_repos/cyberhelp-stuff (newbranch)
$
```

Example: Q's personal GitHub

- https://github.com/qdread
  - **DISCLAIMER**: this does not entirely embody best practices!!!

Example: SESYNC Cyberinfrastructure GitHub

- https://github.com/sesync-ci/
  - Project containing all lessons in curriculum – good demonstration of project management
  - rslurm package – note the issues and pull requests

(Semi)-advanced topic: issues and pull requests

- Can be used either internally or externally
  - Internally to track tasks for project management
  - Externally for community members to contribute to your project

SESYNC resources

- basic git lesson
- all quick start, FAQ, and help pages tagged with git
- document with resources on learning GitHub Pages – this is still a draft in progress!

Outside resources

- Using zenodo to get a DOI for your published git repository
- Hadley Wickham's book on packages
- Excuse me, do you have a moment to talk about version control? Excellent paper by Jenny Bryan, which I shamelessly copied from for this presentation