

Early Git + GitHub Usage

Special tips for R & Rmd

Jenny Bryan
RStudio (& UBC)

 @jennybc
 @JennyBryan

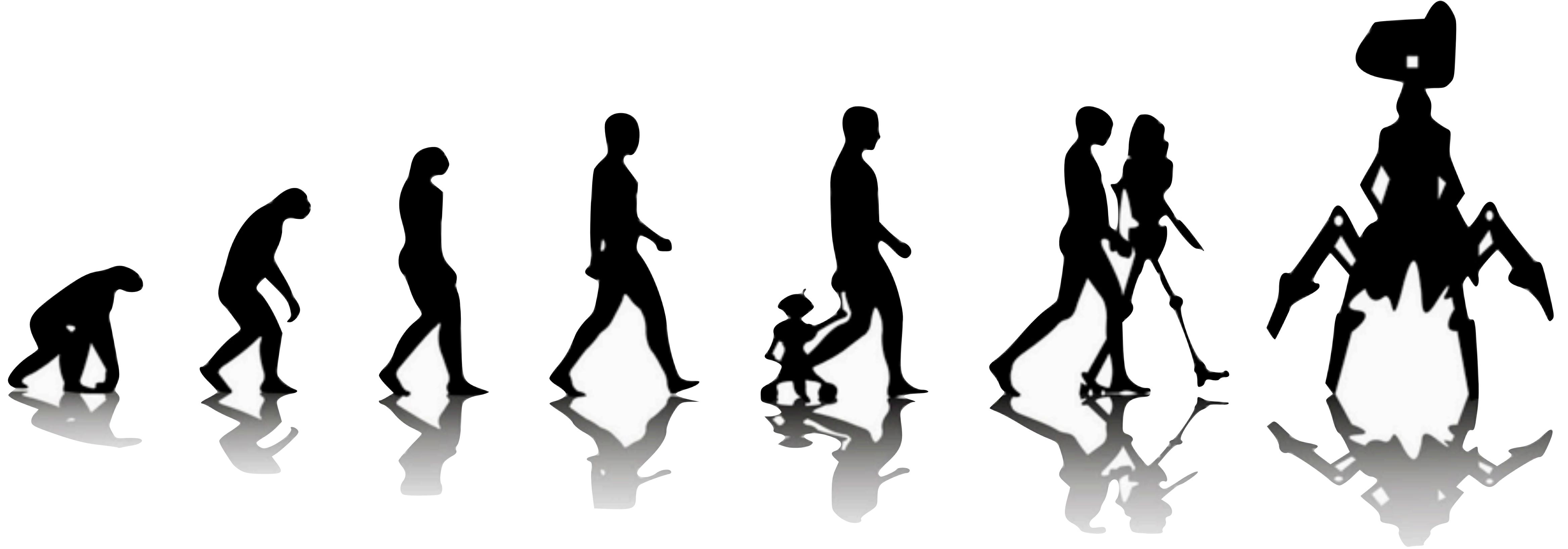
This work is licensed under a **Creative Commons**
Attribution-ShareAlike 4.0 International License.

To view a copy of this license, visit
<http://creativecommons.org/licenses/by-sa/4.0/>

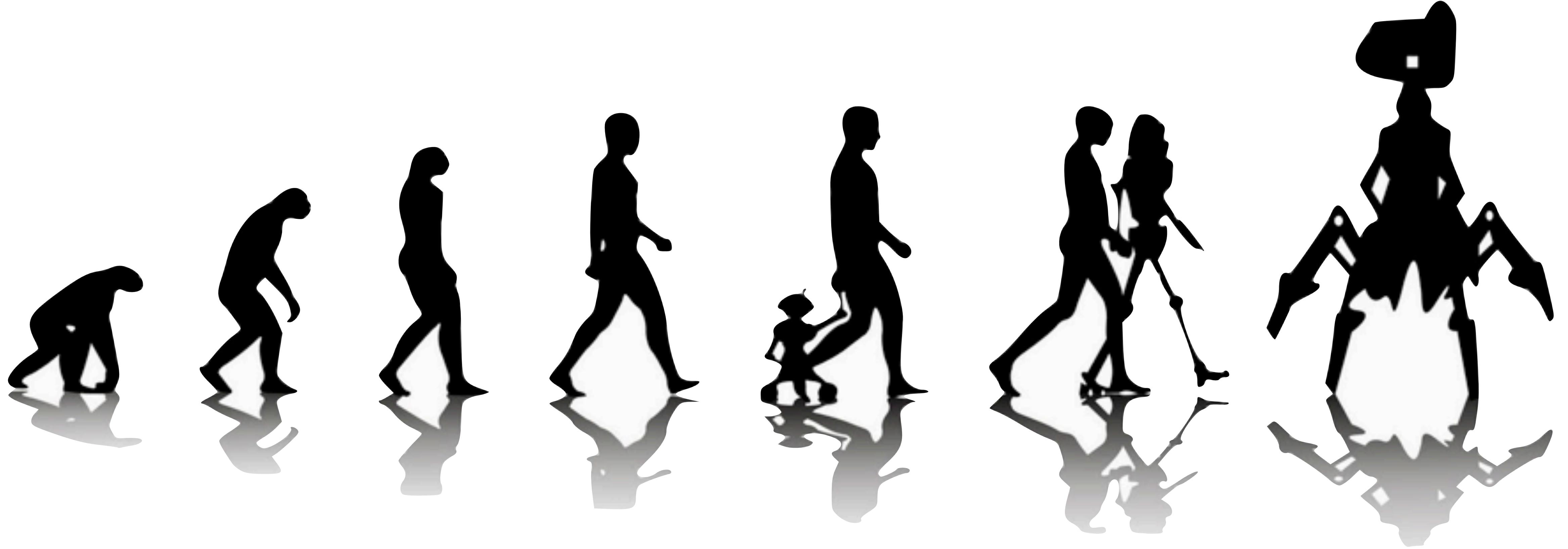
rstd.io/raukr

*Deep
Thoughts*

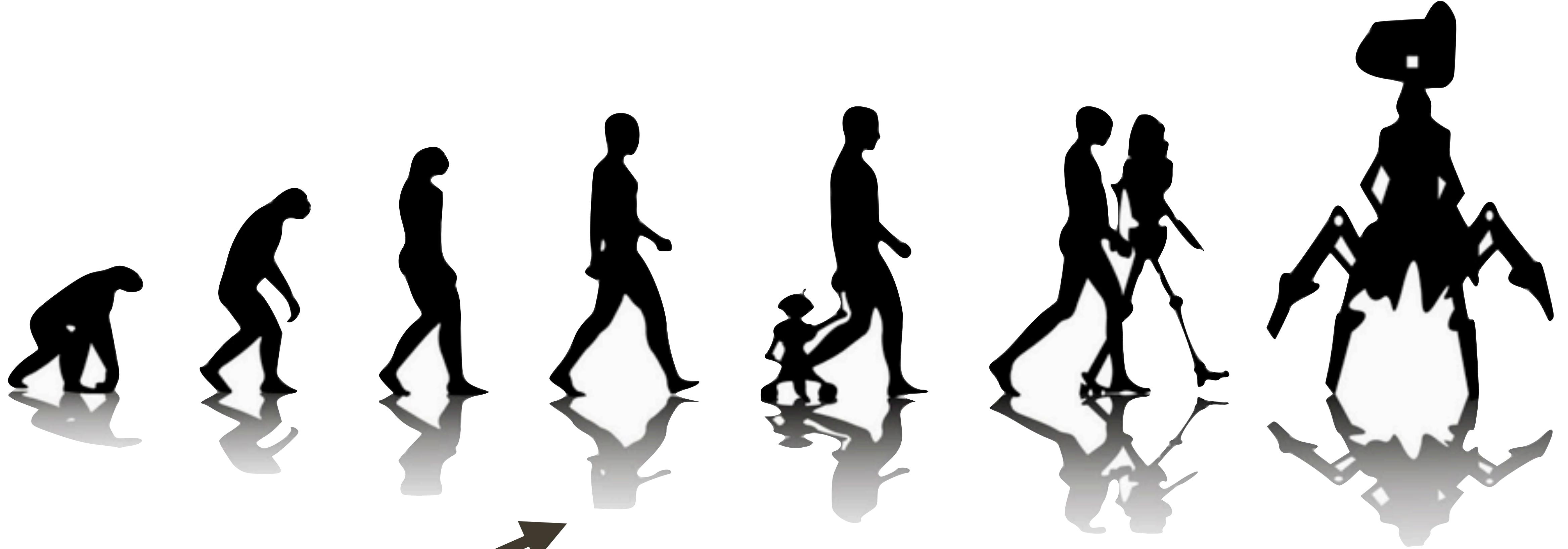




use version control

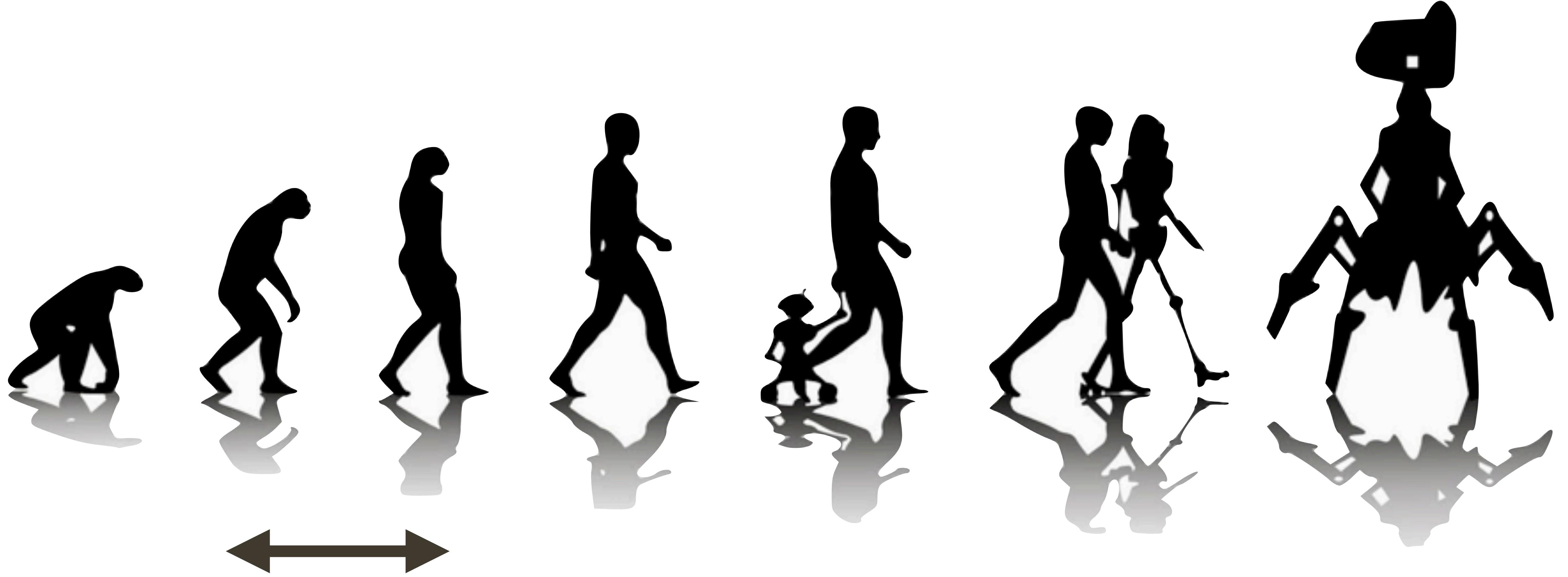


we teach Git + GitHub



"commit"

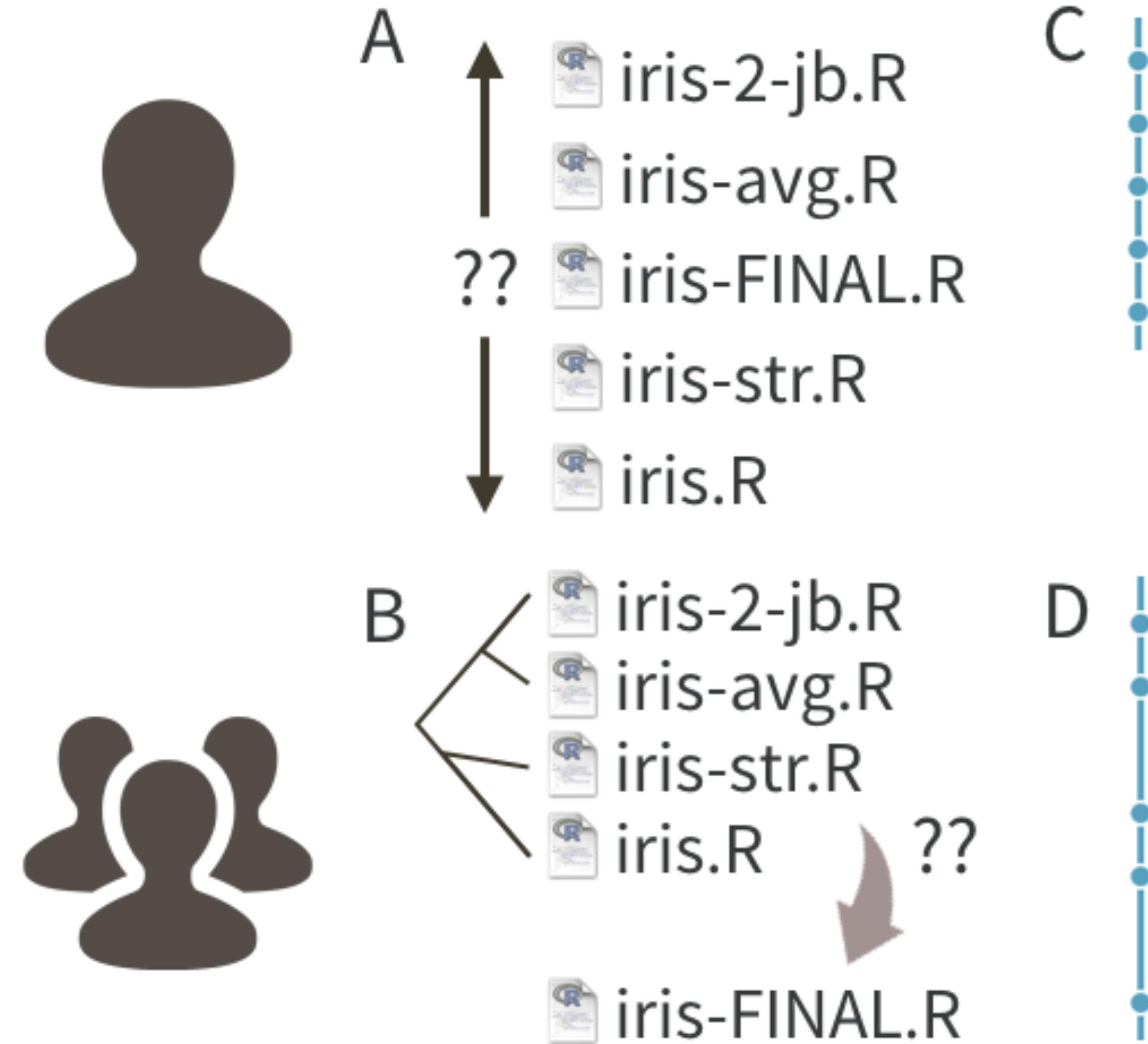
a file or project state that is **meaningful to you**
for inspection, comparison, restoration



Δ

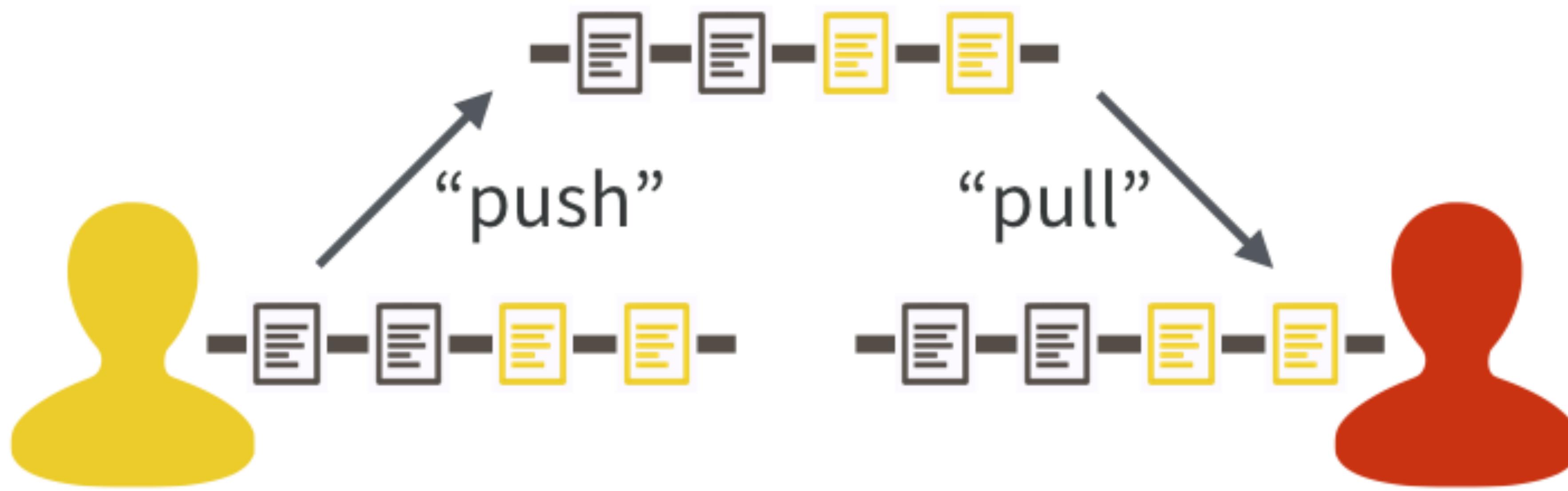
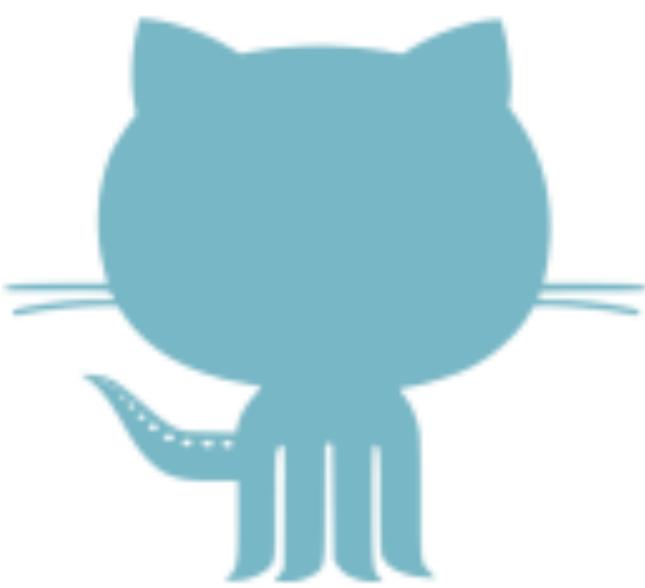
"diff"

What changed here?
Why?



draft-01 Render as report
Formula method
Coauthor prefers str()
Avg by species
Obligatory iris example

draft-01 Render as report
Merge branch 'formula'
 formula Formula method
Coauthor prefers str()
Merge branch 'species'
 species Avg by species
Obligatory iris example

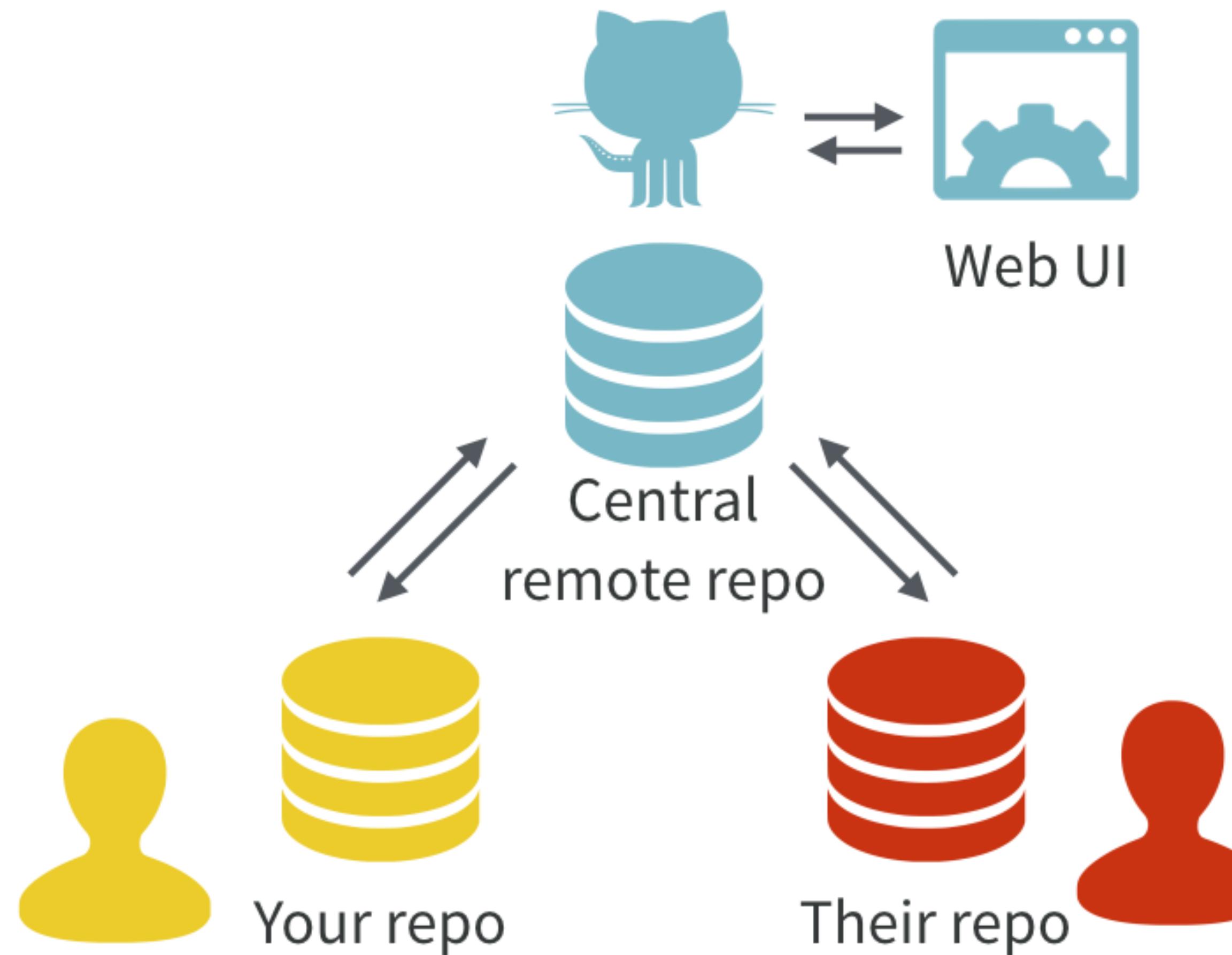


collaboration

Excuse me, do you have a moment to talk about version control?

<https://doi.org/10.7287/peerj.preprints.3159v2>

<https://doi.org/10.1080/00031305.2017.1399928>



happygitwithr.com

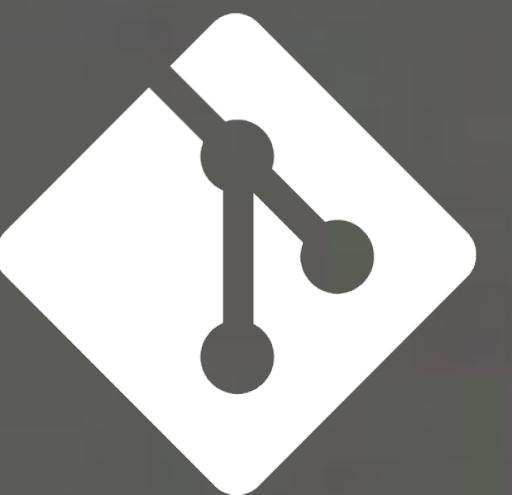


Why version control?

- experiment without fear
- explore cause and effect
- embrace incrementalism
- collaborate
- expose your work

A close-up photograph of a man with light brown hair and glasses, wearing a dark suit jacket over a white shirt. He is looking upwards and to the right with a thoughtful expression, his right index finger pointing towards the top of the frame.

how
git
feels



get off the beach!



agony : flow

agony: flow



agony reduction

Use a Git client, if you like

No one is giving out Hard-core Git Nerd Badges

I like RStudio + (GitKraken | SourceTree)

<https://happygitwithr.com/git-client.html>

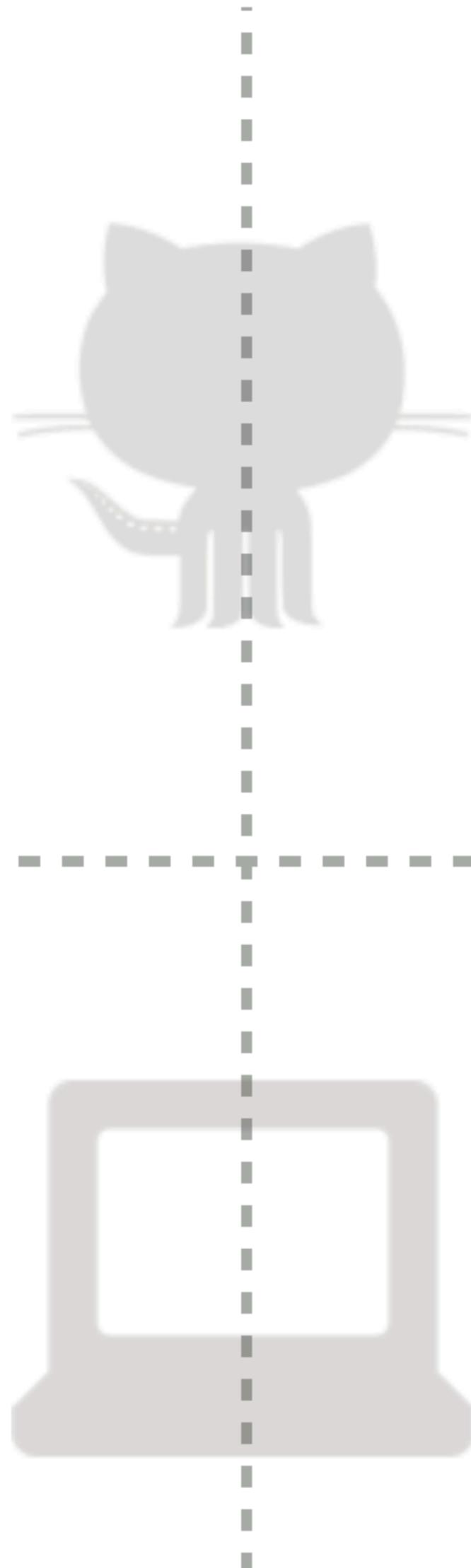
Them



You



not your
problem



Project initiation: the remote case

Make Your remote copy of Their remote repo = "fork"

Make a local Project from a remote repo, Yours or Theirs = "clone"

Make Your remote repo from a local Project ... a bit fiddly

Project initiation strategies, the remote case

Make Your remote copy of Their remote repo = "fork"

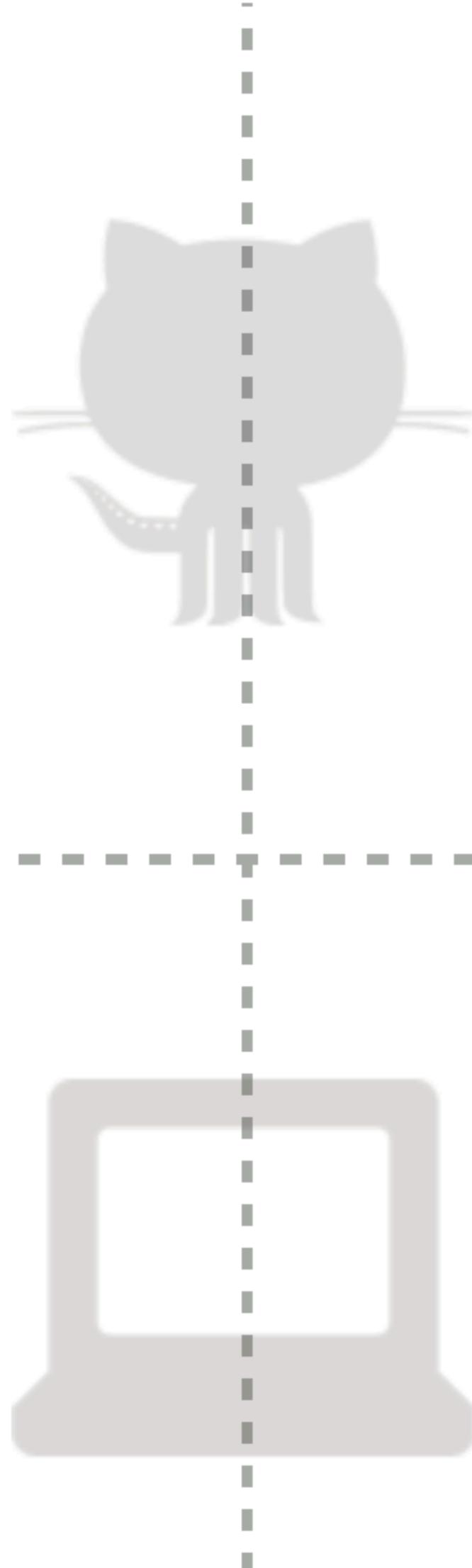
Make a local Project from a remote repo, Yours or ~~Theirs~~ = "clone"

Make Your remote repo from a local Project ... a bit fiddly

Them

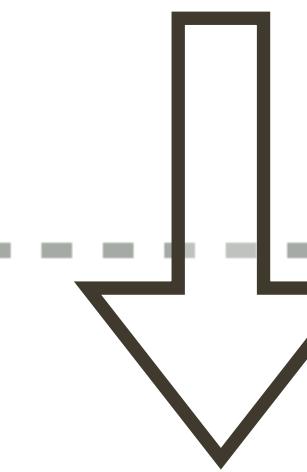


You



not your
problem

"clone"



Them



You



push ↑ ↓ pull



not your
problem



daily work, your stuff

Them



You



not your
problem



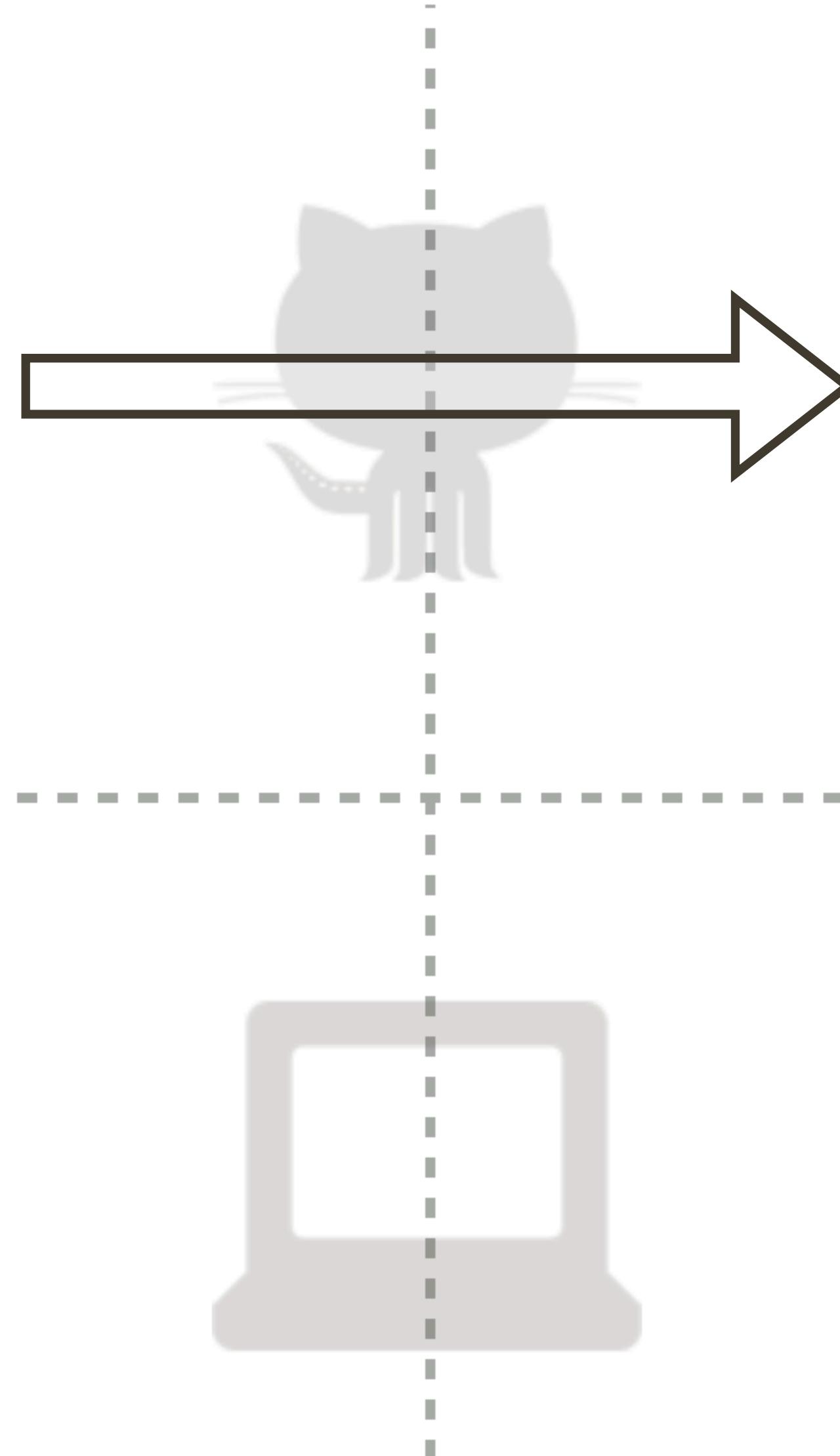
"clone"

*not as useful as you might think

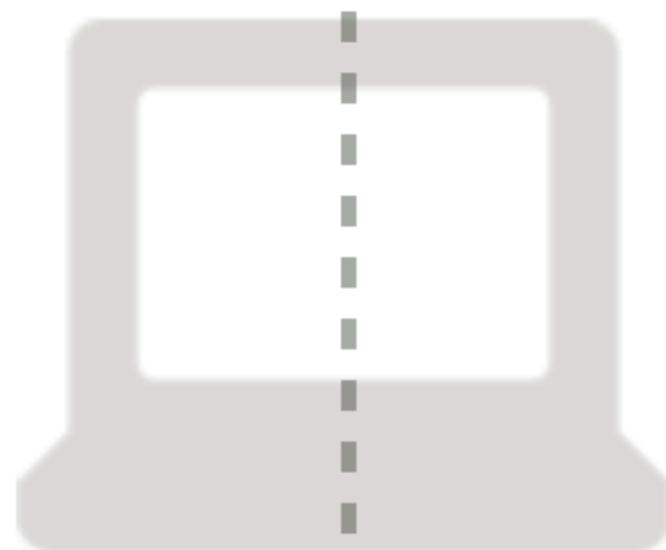
Them



You



not your
problem

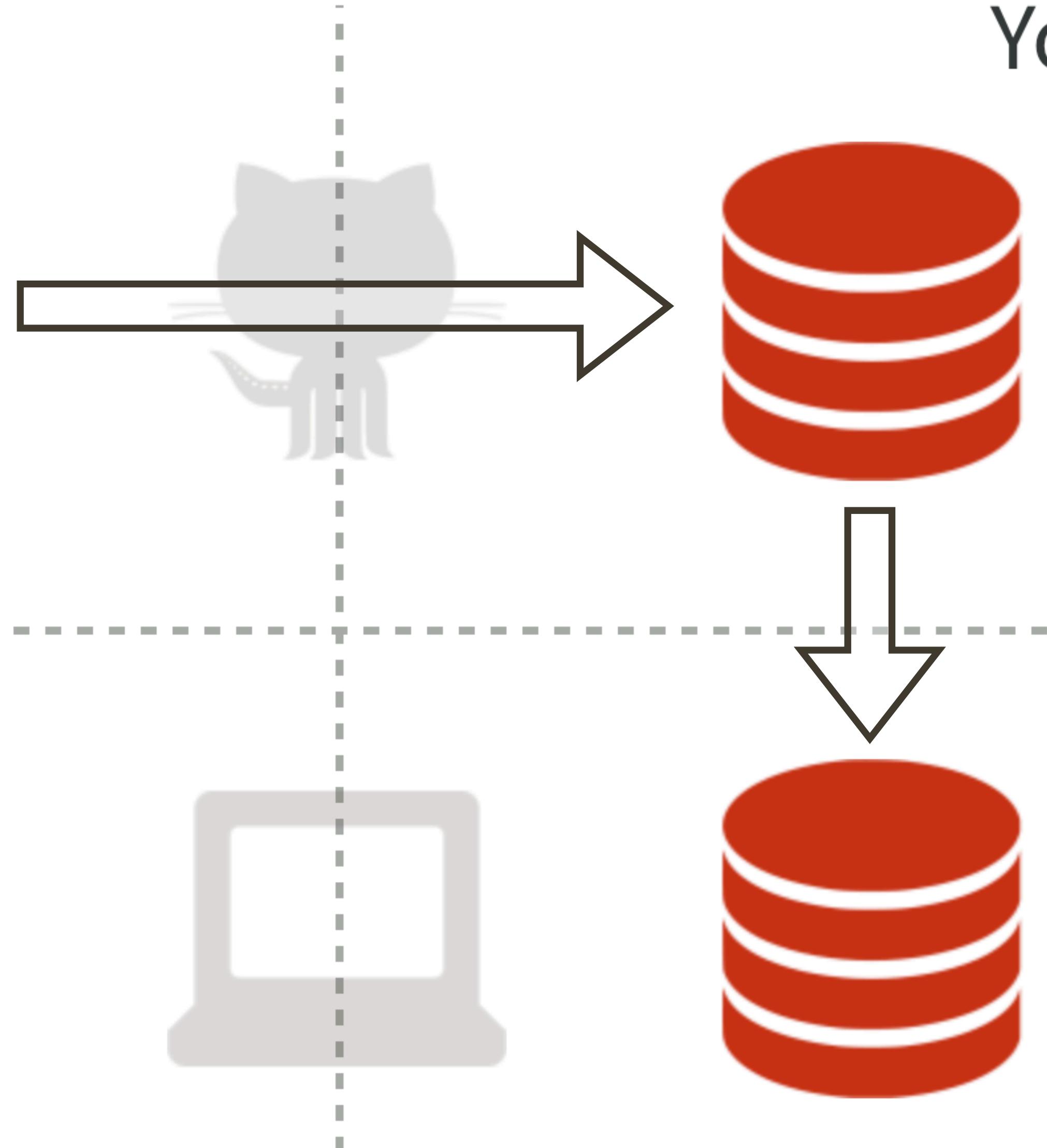


"fork"

Them



You



not your
problem

"fork and clone"

Them



You



pull request

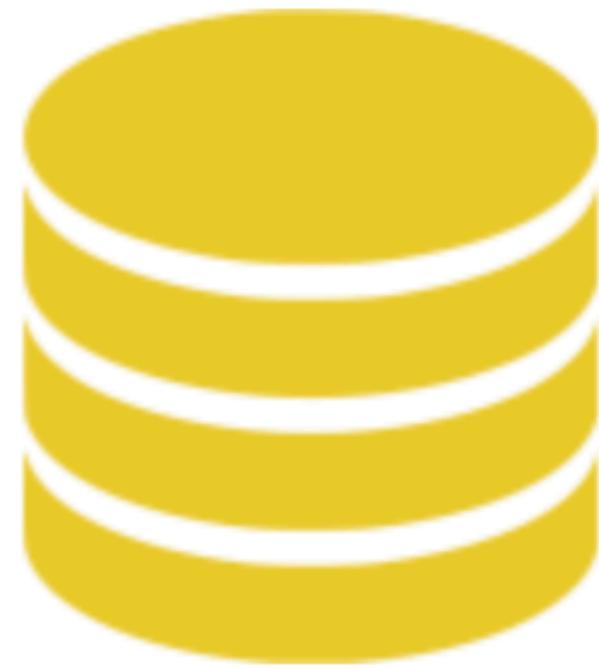
pull

push

not your
problem

contribute to other people's stuff

Them



You



push ↑
↓ pull



not your
problem



daily work, your stuff

"New project, GitHub first"

Why do I emphasize this?

My diagrams omit two big technical points:

- remotes
- branches

I want you to **get off the beach** before fannying around with remotes and branches.

"New project, GitHub first" workflow

<https://happygitwithr.com/new-github-first.html>

I suggest:

- repo / Project / folder name = "**packages-report**"

coordinated work through this:

<https://happygitwithr.com/new-github-first.html>

Create a new .R file.

Suggestion: explore your package library.

Doesn't matter much what it is.

Just make sure it does something.

What changed in Git pane?

Inspect the diff.

Stage.

Commit.

Push.

Verify the .R file is now on GitHub.

Wait ... is a .R file all I want to share?

what you
need to write



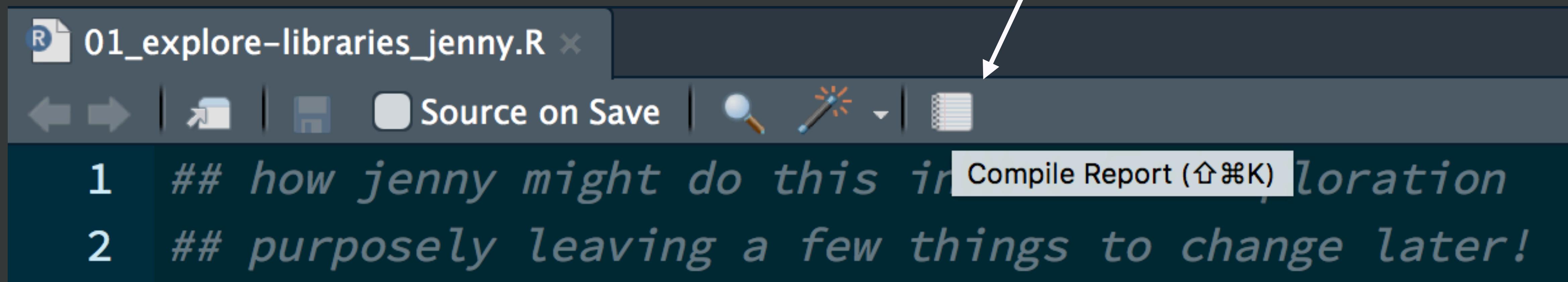
what people
like to read

foo.R
foo.Rmd

foo.md
foo.html



Compile Report



A screenshot of the RStudio interface. At the top, there's a dark header bar with a file icon and the text "01_explore-libraries_jenny.R". Below the header is a toolbar with several icons: back, forward, file, source on save (which is checked), search, and a compile report icon. A white arrow points from the word "Compile" in the title above to the "Compile Report" button in the toolbar. The main workspace shows two lines of R code:

```
1 ## how jenny might do this in exploration  
2 ## purposely leaving a few things to change later!
```

```
≈ rmarkdown::render("whatever.R")
```

Compile Report from R Script

Create a standalone report that contains the code and output from your R script.

For more information on compiling reports, see the documentation at [Compiling Reports from R Scripts](#)

Report output format:

HTML

Compile

Cancel

Sure, HTML is fine ... for now.

What changed in Git pane?

Inspect the diff. Or not.

Stage.

Commit.

Push.

Verify the .html file is now on GitHub.

Wait ... is .html immediately useful on GitHub?



248 lines (201 sloc) | 723 KB

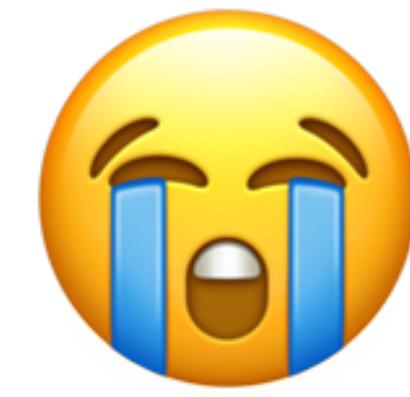
Raw

Blame

History



```
1 <!DOCTYPE html>
2
3 <html xmlns="http://www.w3.org/1999/xhtml">
4
5 <head>
6
7 <meta charset="utf-8" />
8 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
9 <meta name="generator" content="pandoc" />
10
11
12 <meta name="author" content="jenny" />
13
14
15 <title>01_explore-libraries_jenny.R</title>
16
17
18 <meta name="viewport" content="width=device-width, initial-scale=1" />
19
20 <script src="data:application/x-javascript;base64,LyohCiAqIEJvb3RzdHJhcCB2My4zLjUgKGh0dHA6Ly9nZXRib290c3RyYXAuY29tKQogKiBDb3B!
21 <script src="data:application/x-javascript;base64,LyoqCiogQHByZXNlcnZlIEhUTUw1IFNoaXYgMy43LjIgfCBAYWZhcmthcyBAamRhRvbIBAam9i
22 <script src="data:application/x-javascript;base64,LyohIFJlc3BvbQuanMgdjEuNC4y0iBtaW4vbWF4LXdpxHRoIG1lZGlhIHf1ZXJ5IHBvbHlmaWx:
23 <script src="data:application/x-javascript;base64,CgovKioKICogalF1ZXJ5IFBsdWdpbjogU3RpY2t5IFRhYnMKICoKICogQGF1dGhvcIBBaWRhbIB
24 <link href="data:text/css;charset=utf-8,%2Ehljs%2Dliteral%20%7B%0Acolor%3A%20%23990073%3B%0A%7D%0A%2Ehljs%2Dnumber%20%7B%0Aco
```



NO, raw .html is NOT immediately useful* on GitHub.

But Markdown = .md is useful.

Let's render .R to .md instead of .html!

* it CAN BE useful in actual web publishing workflows

foo.R



foo.html

```
#' ---  
#' title: "Untitled"  
#' output: html_document  
#' ---
```

foo.R



foo.html

```
#' ---  
#' title: "Untitled"  
#' output:  
#'   html_document:  
#'     keep_md: yes  
#' ---
```

foo.R



```
#' ---  
#' output: md_document  
#' ---
```

foo.R



foo.html

```
#' ---  
#' output: github_document  
#' ---
```

foo.R



foo.html

```
#' ---  
#' title: "Untitled"  
#' output: html_document  
#' ---
```

foo.R



foo.html

```
#' ---  
#' title: "Untitled"  
#' output:  
#'   html_document:  
#'   keep_md: yes  
#' ---
```

foo.R



```
#' ---  
#' output: md_document  
#' ---
```

foo.R



```
#' ---  
#' output: github_document  
#' ---
```

Add this YAML frontmatter (the "---" matter!)

Re-Compile Notebook

What changed?

This is what I mean by "explore cause and effect" and "experiment without fear".

```
#' ----  
#' output: github_document  
#' ----
```

What changed in Git pane?

Inspect the diff.

Stage.

Commit.

Push.

Verify the .md file is now on GitHub.

Revel in how nice the .md looks!

01_explore-libraries_jenny.R

jenny Sat Jan 27 22:46:07 2018



```
## how jenny might do this in a first exploration  
## purposely leaving a few things to change later!
```

Which libraries does R search for packages?

```
.libPaths()
```

```
## [1] "/Users/jenny/resources/R/library"  
## [2] "/Library/Frameworks/R.framework/Versions/3.4/Resources/library"
```

```
## let's confirm the second element is, in fact, the default library  
.Library
```

```
## [1] "/Library/Frameworks/R.framework/Resources/library"
```

This is what I mean by "expose your work".

Take away #1:

Consider putting rendered products on GitHub.

Just because someone can fork, clone, install all necessary packages, then run your code, it doesn't mean they want to or will.

Be kind. Be realistic.

Take away #2:

For consumption on GitHub, Markdown (.md) is vastly more useful than .html, .docx, .pdf, etc.

Binary formats like .docx and .pdf are also a reliable source of merge conflicts. Think carefully before you track them with Git.

Resources re: which files to commit & how to make your
repo browsable

Excuse Me, ... section re: "Which files to commit"

Make a GitHub repo browsable

<https://happygitwithr.com/workflows-browsability.html>

Keep developing this scripts.

After each meaningful change, re-render.

What changed? Look at the diffs.

Stage. Commit. Push. Check result on GitHub.

This is what I mean by "embrace incrementalism".

independent work on challenge

ideas:

- add a table
- add a figure
- add `devtools::session_info()` at the end
or `sessionInfo()` if no devtools

What now? Depends on what time it is!

Please open issues for questions that you 've raised and we discussed! Useful to me for planning Thursday's final Git/GitHub coverage.

What follows are slides I can imagine us referring to today or Thursday.

What now? Game time decision! Possibilities:

Equivalence between R and Rmd.

Work up to the fully developed README.Rmd in the example packages-report project.

GitHub Pages, the Simple Version.

Equivalence between .Rmd and .R

A

```
iris.Rmd x | ABC Knit | ⚙️
```

```
1 ---  
2 title: "Report from R/Rmd"  
3 author: "Jenny Bryan"  
4 date: "`r format(Sys.Date())`"  
5 output: github_document  
6 ---  
7  
8 The iris data is boring, but it won't distract  
9 from the Git content.  
10  
11 ``{r}  
12 aggregate(. ~ Species, data = iris, median)  
13 ````
```

B

```
iris.R x | Source on Save | ⚙️
```

```
1 #' ---  
2 #' title: "Report from R/Rmd"  
3 #' author: "Jenny Bryan"  
4 #' date: "`r format(Sys.Date())`"  
5 #' output: github_document  
6 #' ---  
7  
8 #' The iris data is boring, but it won't distract  
9 #' from the Git content.  
10  
11 aggregate(. ~ Species, data = iris, median)
```

C

Report from R/Rmd

Jenny Bryan
2017-06-29

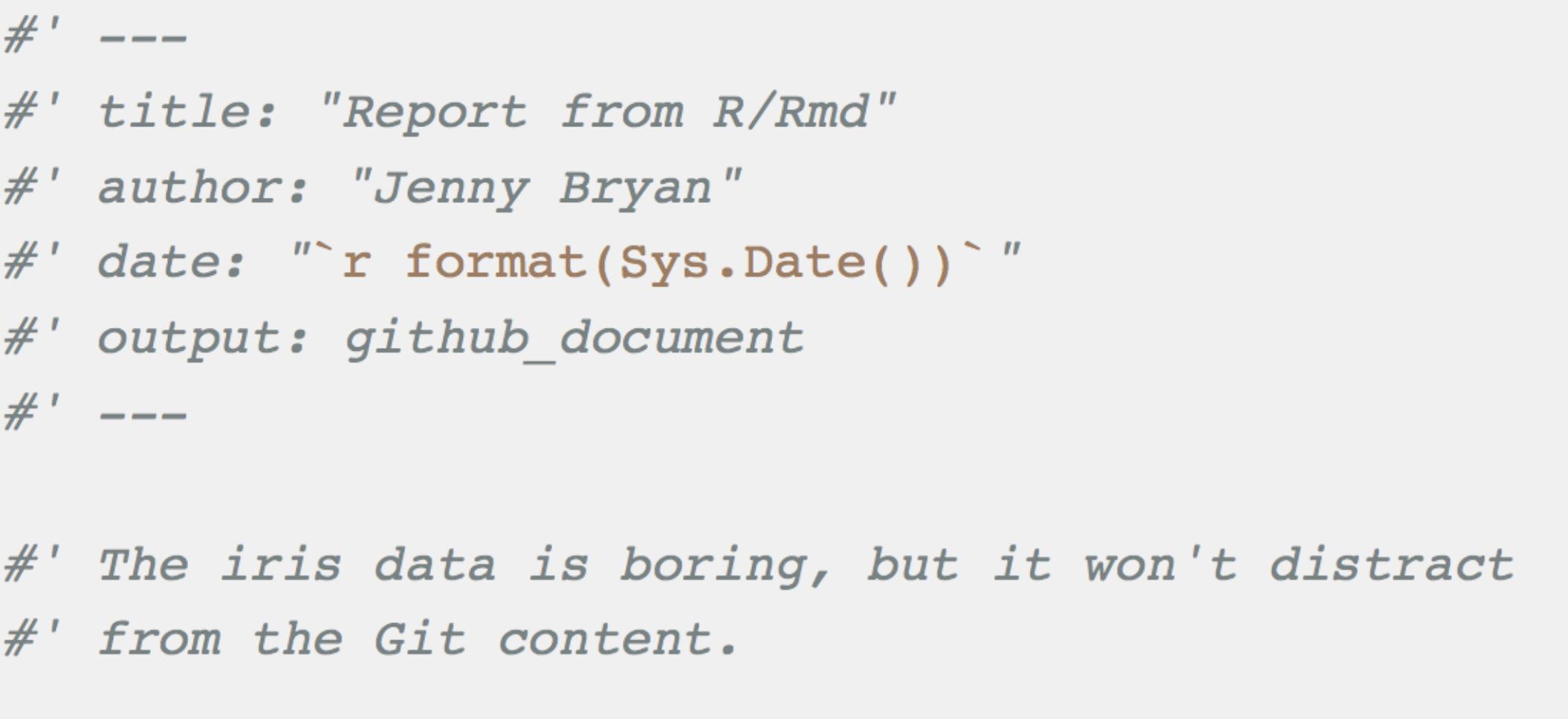
The iris data is boring, but it won't distract from the Git content.

```
aggregate(. ~ Species, data = iris, median)
```

	Species	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
## 1	setosa	5.0	3.4	1.50	0.2
## 2	versicolor	5.9	2.8	4.35	1.3
## 3	virginica	6.5	3.0	5.55	2.0

From "Excuse Me,..." article

.R



The screenshot shows the RStudio interface with the file 'iris.R' open. The code in the editor is as follows:

```
1 #' ---
2 #' title: "Report from R/Rmd"
3 #' author: "Jenny Bryan"
4 #' date: "r format(Sys.Date())"
5 #' output: github_document
6 #' ---
7
8 #' The iris data is boring, but it won't distract
9 #' from the Git content.
10
11 aggregate(. ~ Species, data = iris, median)
```

R code is top-level
Use #' comment for prose
#+ for chunk header

.Rmd

Prose is top-level
Put R code in chunks

Nice defaults for global chunk options

```
knitr::opts_chunk$set(  
  collapse = TRUE,  
  comment = "#>",  
  out.width = "100%"  
)
```

foo.Rmd



foo.html

```
---
```

```
title: "Untitled"
```

```
output: html_document
```

```
---
```

foo.Rmd



foo.md



foo.html

```
---
```

```
title: "Untitled"
```

```
output:
```

```
    html_document:
```

```
        keep_md: yes
```

```
---
```

foo.Rmd



foo.md

```
---
```

```
output:
```

```
    md_document
```

```
---
```

foo.Rmd



foo.md

```
---
```

```
output:
```

```
    github_document
```

```
---
```

Simplest use of GitHub Pages = Project webpage

In your repo's Settings

The screenshot shows the GitHub Pages settings interface. At the top left, it says "GitHub Pages". Below that, a heading reads "GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository." Under the heading, there's a "Source" section with a message: "GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository." It includes a "Learn more." link. There are two buttons: "master branch" with a dropdown arrow and a "Save" button. Below the "Source" section is a "Theme Chooser" section with the text: "Select a theme to publish your site with a Jekyll theme using the master branch." It includes a "Choose a theme" button. A large black arrow points from the text "In your repo's Settings" towards the "Save" button.

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Source

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. [Learn more.](#)

master branch [Save](#)

Theme Chooser

Select a theme to publish your site with a Jekyll theme using the `master` branch. [Learn more.](#)

[Choose a theme](#)

<https://github.com/blog/2289-publishing-with-github-pages-now-as-easy-as-1-2-3>

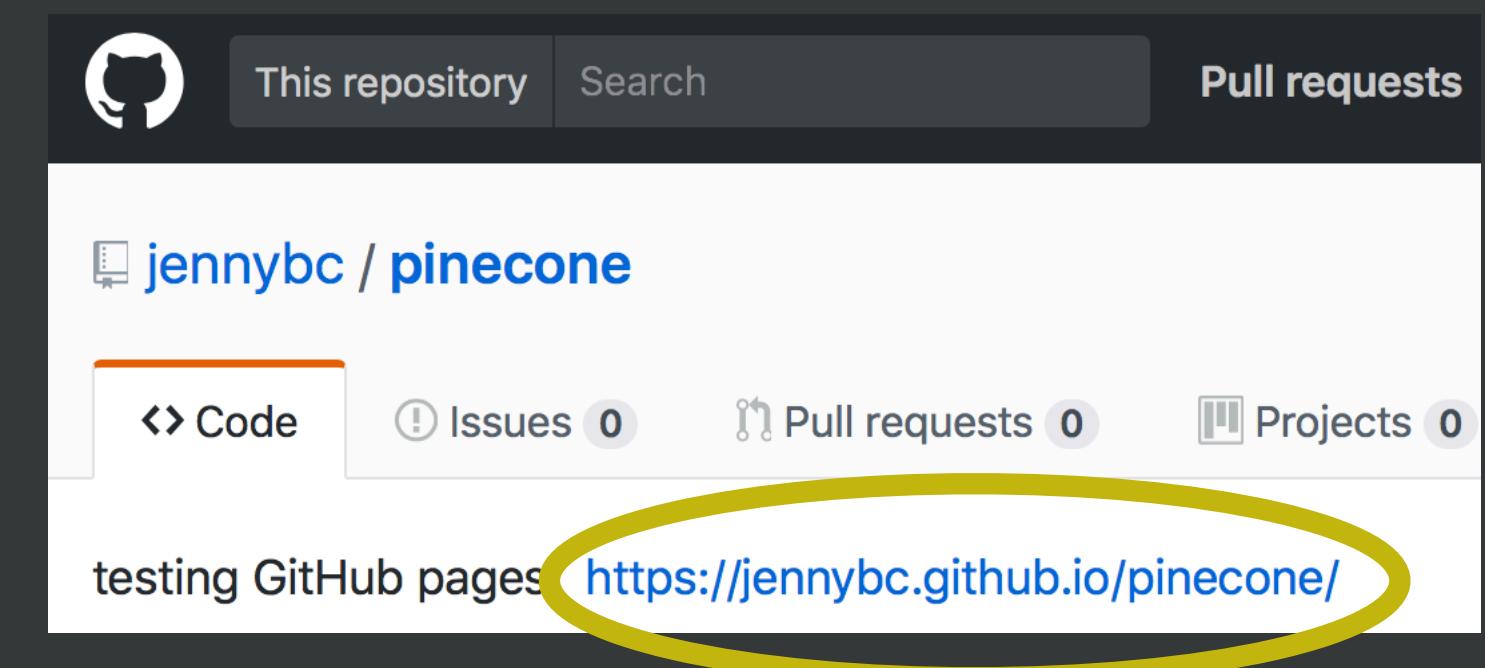
Hot tips for simple GitHub Pages

README.md becomes index.html, by default

Given that foo.md exists, these internal links work (trial & error):

👍 foo, [foo](foo.md), [foo](foo.html)

Record your site URL as your repo's website



wrap up here

when we return to Git/GitHub, we'll
wrap up loose ends:

- branches and remotes
- useful daily workflows