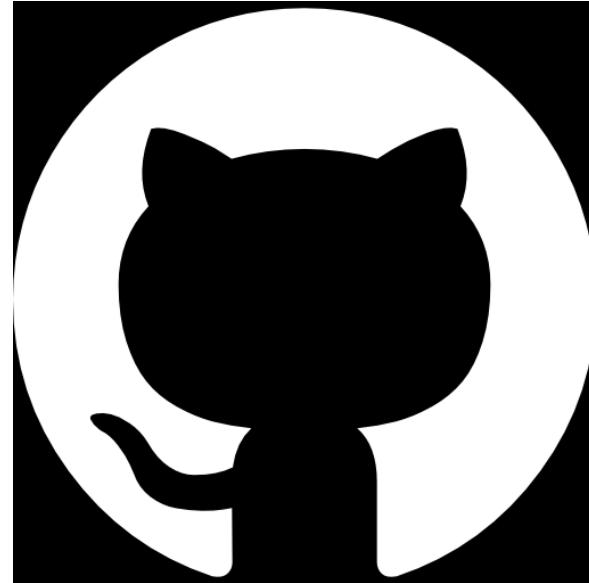


GitHub

- track changes on steroids



Brendan Palmer,
Statistics & Data Analysis Unit,
Clinical Research Facility - Cork

How is research presented?

Theses

Papers



Network Analysis of the Chronic Hepatitis C Virome Defines Hypervariable Region 1 Evolutionary Phenotypes in the Context of Humoral Immune Responses

Brendan A. O'Farrior,^a Daniel Schmidt-Martin,^a Zoya Dimitrova,^b Pavel Skums,^b Orla Crosbie,^c Elizabeth Kenny-Walsh,^c Liam J. Fanning^d

^aMolecular Virology Diagnostic & Research Laboratory, Department of Medicine, University College Cork, Cork, Ireland; ^bDivision of Viral Hepatitis, Centers for Disease Control and Prevention, Atlanta, Georgia, USA; ^cDepartment of Hepatology, Cork University Hospital, Cork, Ireland

ABSTRACT
Hypervariable region 1 (HVR1) of hepatitis C virus (HCV) comprises the first 27 N-terminal amino acid residues of E2. It is classically recognized as the major antigenic target of the humoral immune response. HVR1 undergoes rapid sequence evolution in chronically infected patients over a short, 10-week period. Organization of the sequence set into connected components that represented single nucleotide substitution events revealed a network dominated by highly connected, centrally positioned master sequences. HVR1 phenotypes were observed to be under strong purifying (stationary) and strong positive (antigenic drift) selection pressures, which were consistent with advancing patient age and disease course of HCV infection. It was found that the HVR1 network structure was dominated by stationary variants, which were composed from conservative single amino acid substitution events. We present evidence to suggest that neutralization antibody efficacy was diminished for stationary-virome HVR1 variants. Our results identify the HVR1 network structure during chronic hepatitis C as the predominant dominance of a single variant within a narrow sequence space.

IMPORTANCE
Hepatitis C virus (HCV) is often asymptomatic, and chronic infection is generally well established in advance of initial diagnosis and subsequent treatment. HVR1 can undergo rapid sequence evolution during acute infection, and the variant pool is typically seen to diverge away from ancestral sequences as infection progresses from the acute to the chronic phase. In this report, we describe HVR1 viromes in chronically infected patients that are defined by a dominant epitope located centrally within a narrow variant pool. Our findings suggest that weakened humoral immune activity, as a consequence of persistent chronic infection, allows for the acquisition and maintenance of host-specific adaptive mutations at HVR1 that reflect virus fitness.

Hepatitis C virus (HCV) infection is a global health burden and a major etiological agent of liver related diseases worldwide. In the United States alone, the estimated prevalence of HCV represents approximately 2% of the global adult (15 years of age and older) population (2). Following transmission, HCV infection is characterized by a long asymptomatic period of infection, with many infections initially passing undetected (3). It is estimated that up to 4 million Americans are living with the virus, the majority of whom became infected prior to the first clinical identification of the virus (4, 5). Consequently, the U.S. Centers for Disease Control and Prevention now recommend that Americans born from 1945 to 1965 be screened for the presence of the virus, as well as all individuals at risk of contracting HCV (6).

HCV is a single-stranded positive-sense RNA virus of considerable genomic heterogeneity. A recent reclassification defines the major genotypes 1 through 6 and 67 subtypes within genotypes 1 and 5 accounting for the majority of infections worldwide (6, 7). An error-prone RNA-dependent RNA polymerase, together with an inherent propensity of defining hypervariable regions (HVRs), is responsible for much of this variability. These HVRs are located within the envelope glycoprotein E2 (residues 456–656), the most variable region of the protein, near the C-terminal end of the E2 glycoprotein (8). Recent studies indicated that the central region of E2 (residues 456 to 656) is globular and surprisingly compact, whereas the first 80 amino acids (including

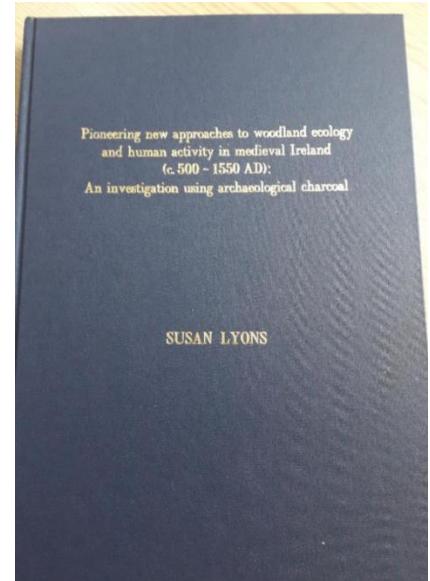
Received 25 November 2015; Accepted 22 December 2015
Accepted manuscript posted online 10 December 2015
Editorial decision received 10 December 2015
Editorial decision received 10 December 2015
Editor: M. S. Diamond
Copyright © 2016, the authors. Some rights reserved.
Published online in *Infect. Immun.* 84:1903–1910, 2016. doi:10.1128/IAI.01400-15
Copyright © 2016, American Society for Microbiology. All Rights Reserved.

3318 *Infect. Immun.* April 2016, Volume 80, Number 7

Books



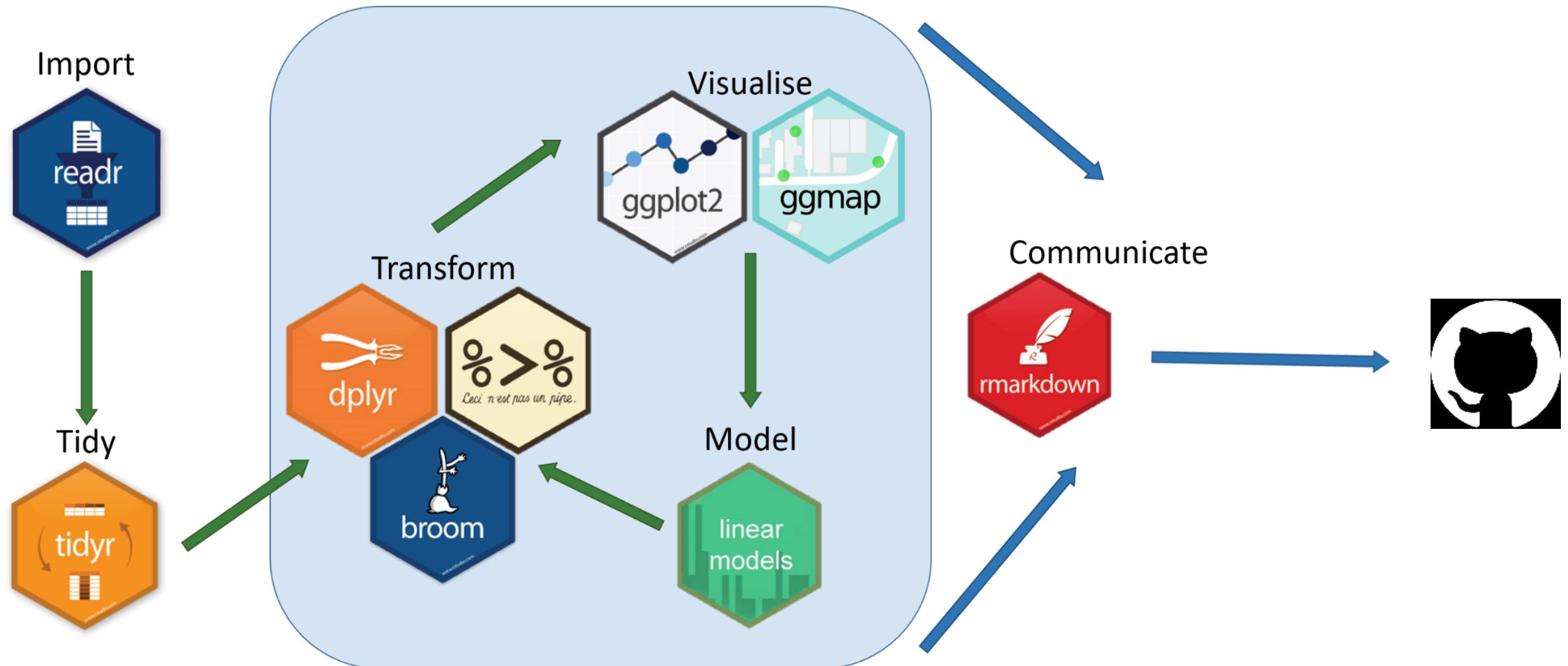
Talks



Posters



Putting the pieces together



Sounds complicated

- It isn't
- To get up and running takes only a few steps
- Any idiot could do it...



Brendan Palmer
bapalmer

Twitter: @B_A_Palmer

[Edit bio](#)

<https://crfcdau.github.io/about/>

Organizations

OPEN
SCIENCE
MOOC

Overview Repositories 7 Stars 0 Followers 7 Following 6

Pinned repositories

reproducible-workflows
1-day R workshop for experienced users. First session is a comprehensive introduction to the tidyverse. The second session covers script writing, file names, reporting using R Markdown and overall ...
R ★ 10 1

project-structure
Final half day session of a 3-day introduction to R/RStudio for beginners
R ★ 1

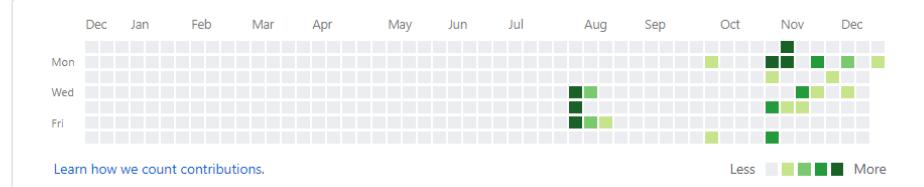
lunchtime_sessions
Short 1 hour introductions to R-related topics such as creating R projects, using GitHub through RStudio and more
HTML ★ 1

Module-5-Open-Research-Software-and-Open-Source
Forked from OpenScienceMOOC/Module-5-Open-Research-Software-and-Open-Source
Module 5: Open Research Software and Open Source
HTML

54 contributions in the last year

Contribution settings ▾

Less More



What is git, gist, GitHub?



git is a version control system

- originally designed to allow developers share code
- since repurposed to manage whole data projects
- gists are individual files or part of files that can be shared



GitHub is a hosting service for your Git projects

- remotely stored

Step 1: Install git after step 0, install R/RStudio

 **git** --distributed-is-the-new-centralized

[About](#)
[Documentation](#)
Downloads
 GUI Clients
 Logos
[Community](#)

The entire [Pro Git book](#) written by Scott Chacon and Ben Straub is available to read [online for free](#). Dead tree versions are available on [Amazon.com](#).

Downloads

 Mac OS X  Windows
 Linux/Unix

Older releases are available and the Git source repository is on GitHub.

GUI Clients

Git comes with built-in GUI tools (`git-gui`, `gitk`), but there are several third-party tools for users looking for a platform-specific experience.

[View GUI Clients →](#)

Git via Git

If you already have Git installed, you can get the latest development version via Git itself:

`git clone https://github.com/git/git`

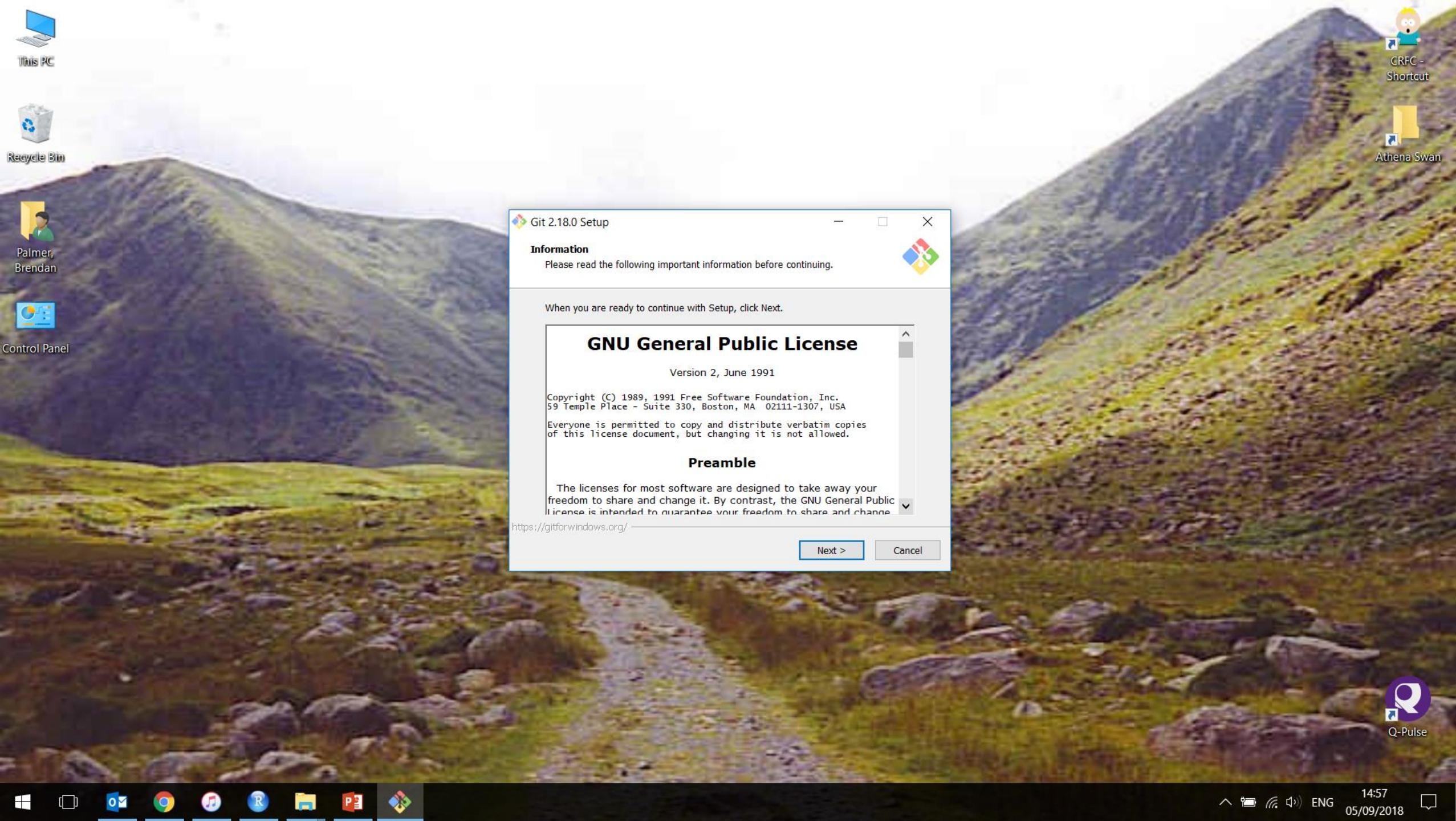


Latest source Release
2.18.0
Release Notes (2018-06-21)
[Download 2.18.0 for Windows](#)

Logos

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

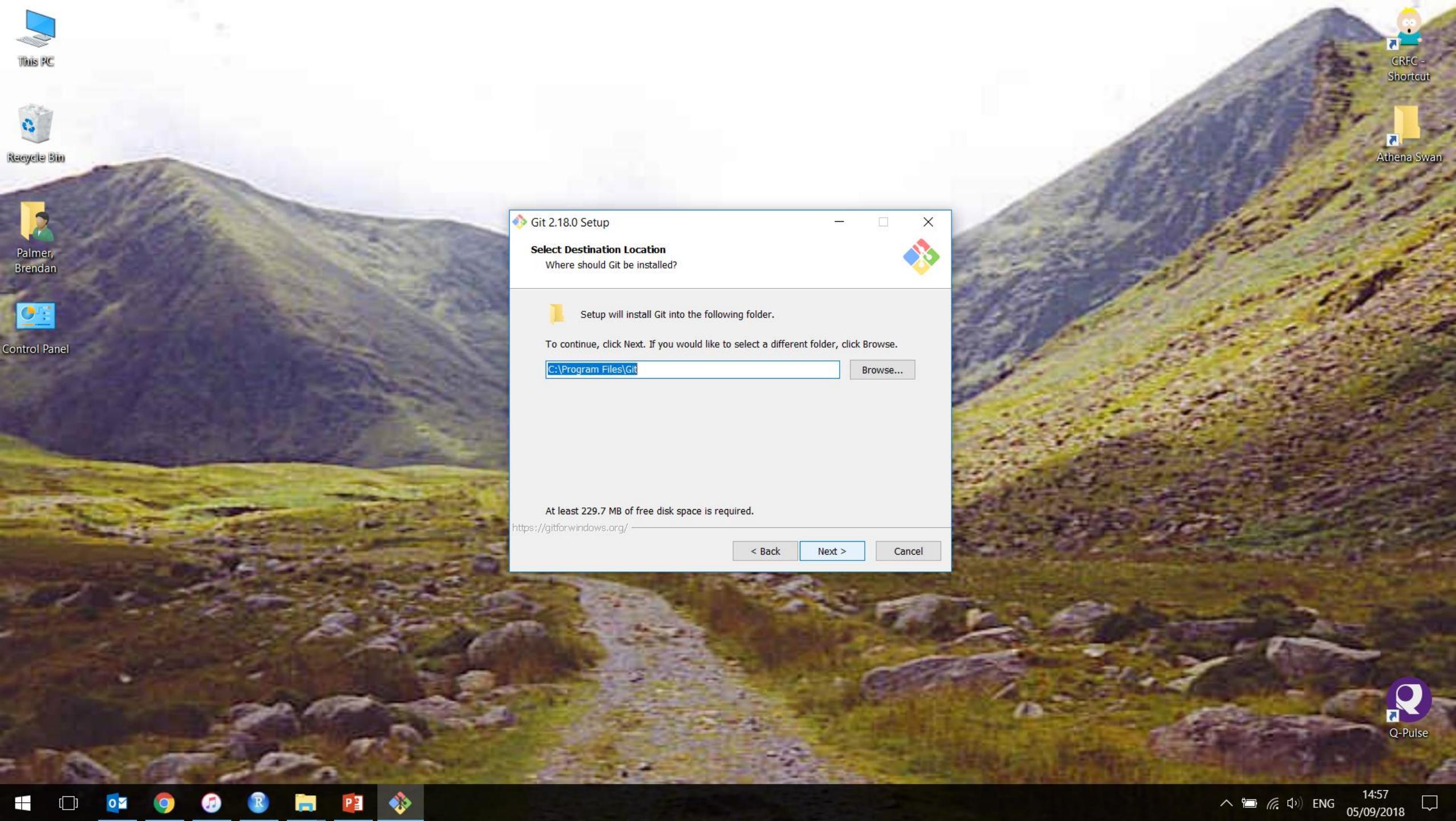
[View Logos →](#)



14:57

ENG

05/09/2018



This PC



Recycle Bin



Palmer,
Brendan



Control Panel



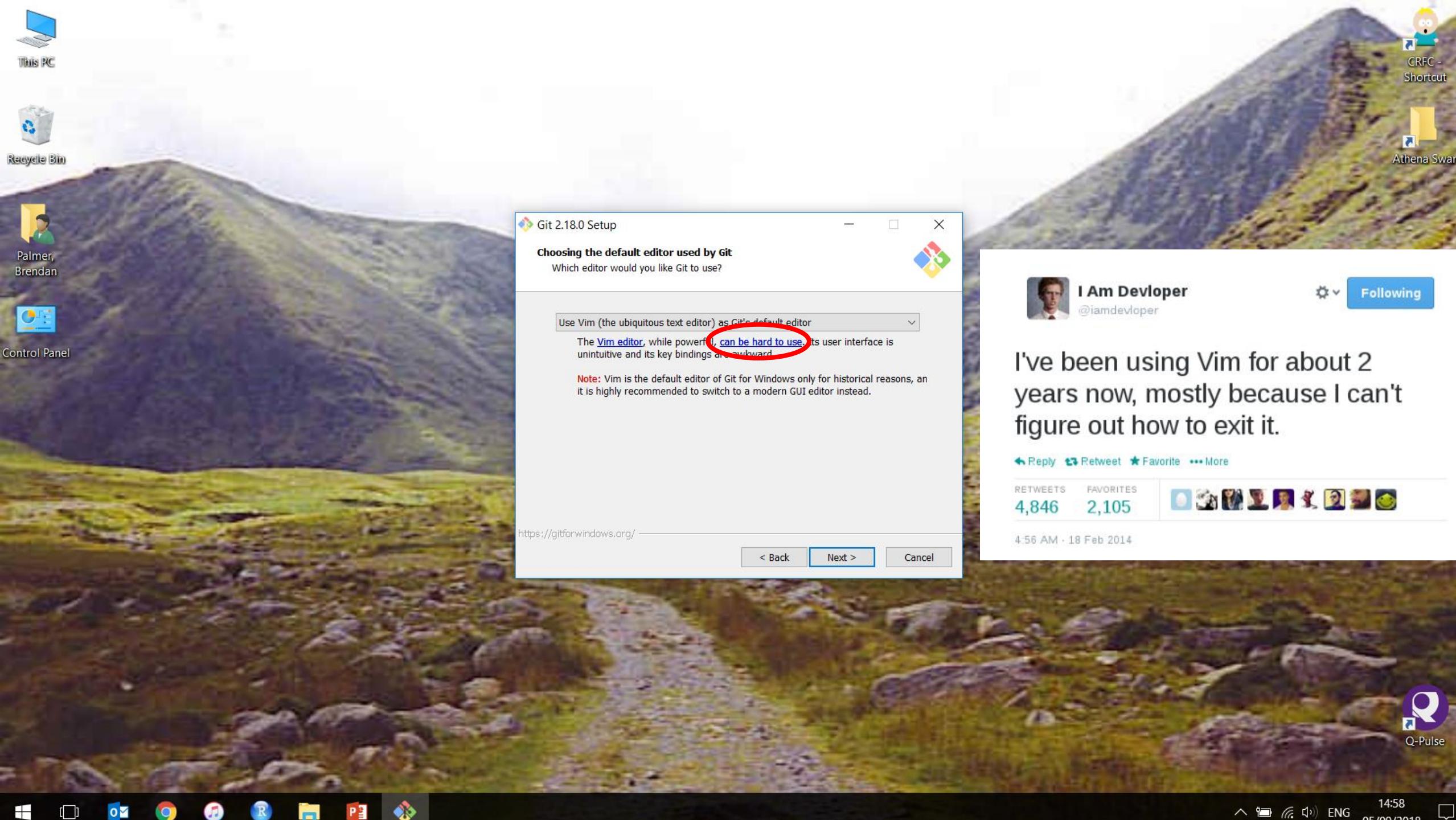
CRFC -
Shortcut

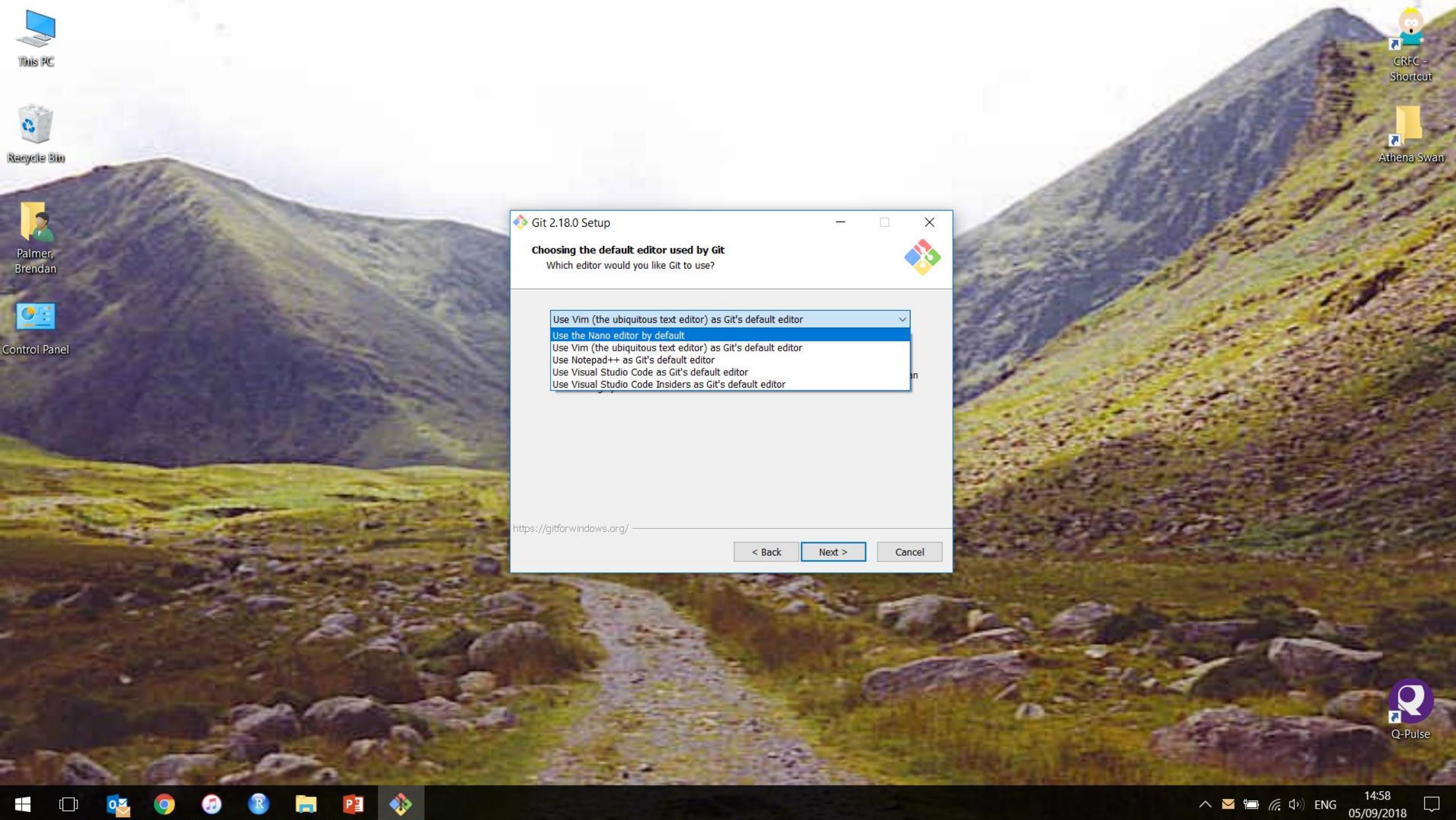


Athena Swan

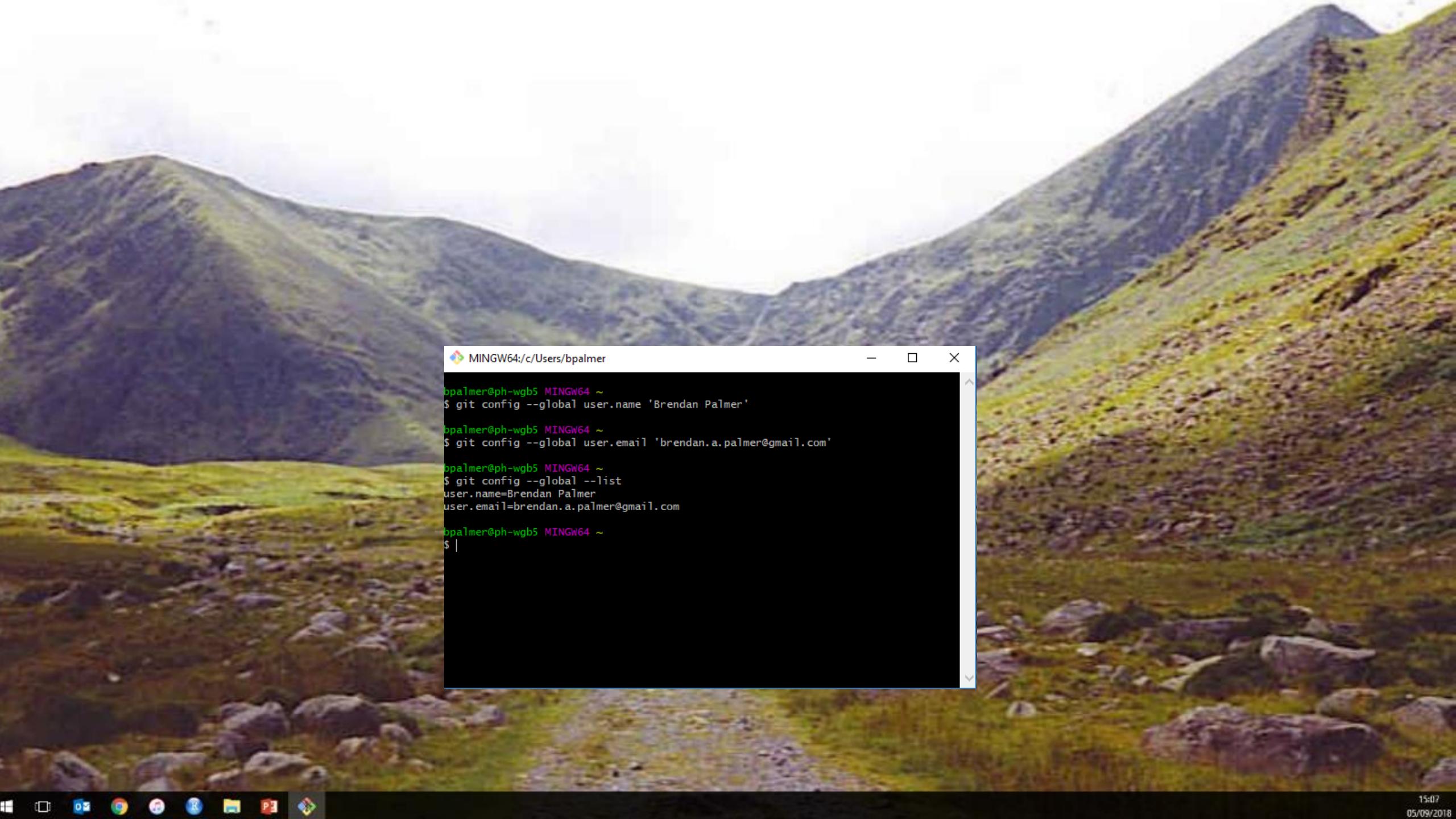


Q-Pulse





Q-Pulse

The background of the image is a photograph of a mountainous landscape with green slopes and rocky ridges under a clear sky.

```
MINGW64:/c/Users/bpalmer
bpalmer@ph-wgb5 MINGW64 ~
$ git config --global user.name 'Brendan Palmer'
bpalmer@ph-wgb5 MINGW64 ~
$ git config --global user.email 'brendan.a.palmer@gmail.com'
bpalmer@ph-wgb5 MINGW64 ~
$ git config --global --list
user.name=Brendan Palmer
user.email=brendan.a.palmer@gmail.com
bpalmer@ph-wgb5 MINGW64 ~
$ |
```

What's there to gain

- A record of your research
- Avoid that chaotic state where you have multiple copies of the same file, but without a stupid and annoying file naming convention.

Documents > MSc_Bioinformatics > MSc_Project > R_scripts

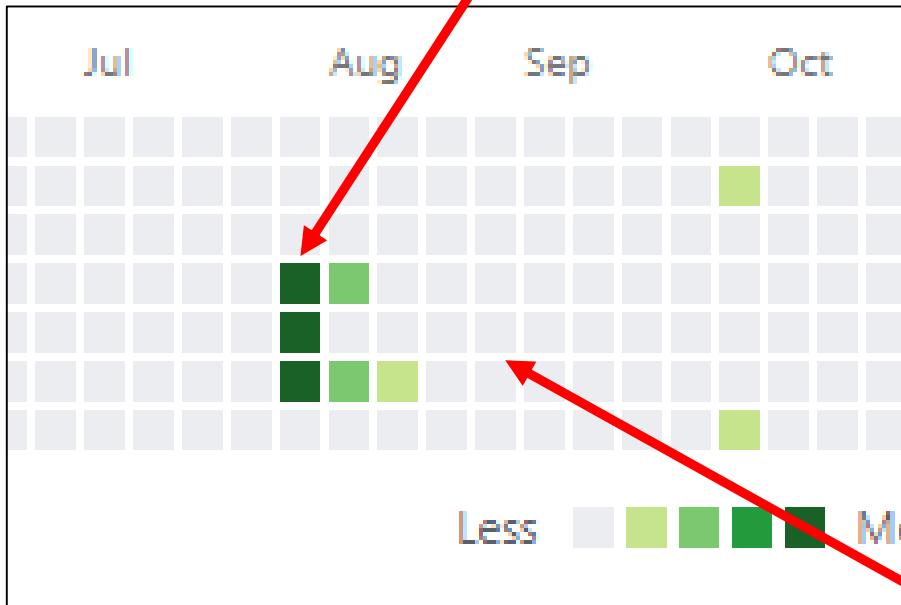
Name
cubefts_default_scripts
.Rhistory
R_Are_colours
R_Binomial_prob_calcs
R_Binomial_prob_calcs_weighted_by_frequency
R_Codon_counting_by_frequency_and_sample
R_Component_heatmap_modified_for_epi
R_ENC_graphs
R_Enumerating_codons_by_column
R_Epistatic_change
R_Epistatic_change_match_discovery
R_Epistatic_change_match_discovery_fig_2_point_1
R_Epistatic_change_v2
R_epistatic_codon_change_tracking
R_Epistatic_connection_network
R_Heatmap_for_epistatic_syn
R_Heatmap1_for_epi_site_co-change
R_Heatmap2_for_epi_fdr_adjusted_p-value
R_Heatmap2_for_epi_p-value
R_Initial_codon_graphs

| | Final Final version

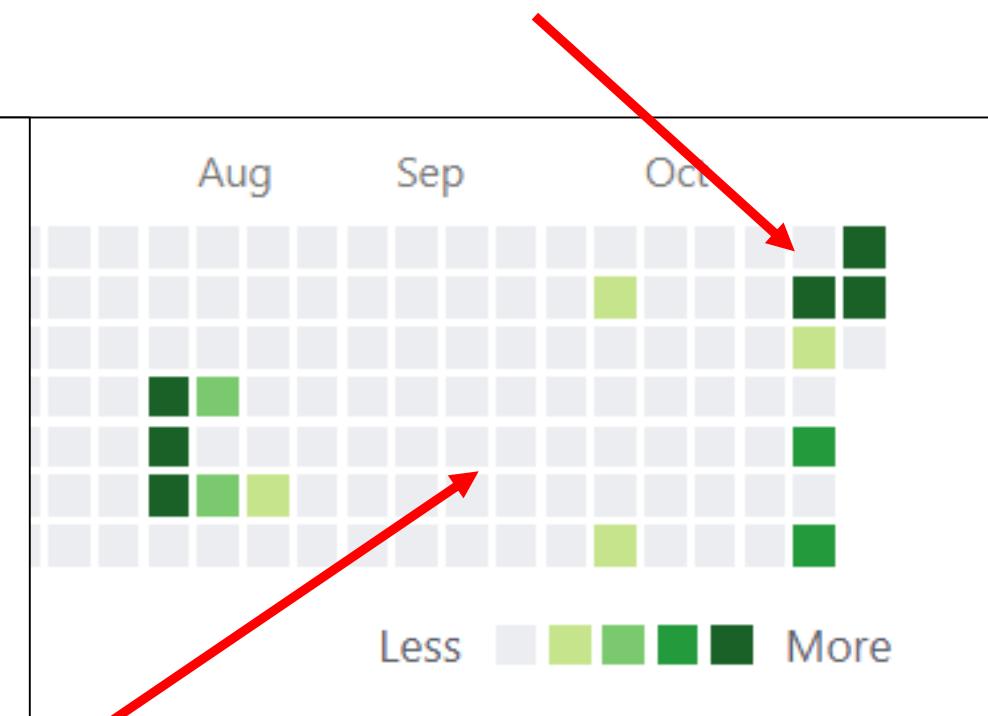
File Home Share View

← → ⌂ ⌃ This PC > B_Palmer_Medicine_Files > 4a Project > Pyrosequencing_analysis > Pyrosequencing_Paper > Draft_Paper_incl_Figs > Submission > JVI_Resubmission > JVI_resubmission_files > Final Final version Final Final version

Discovered git

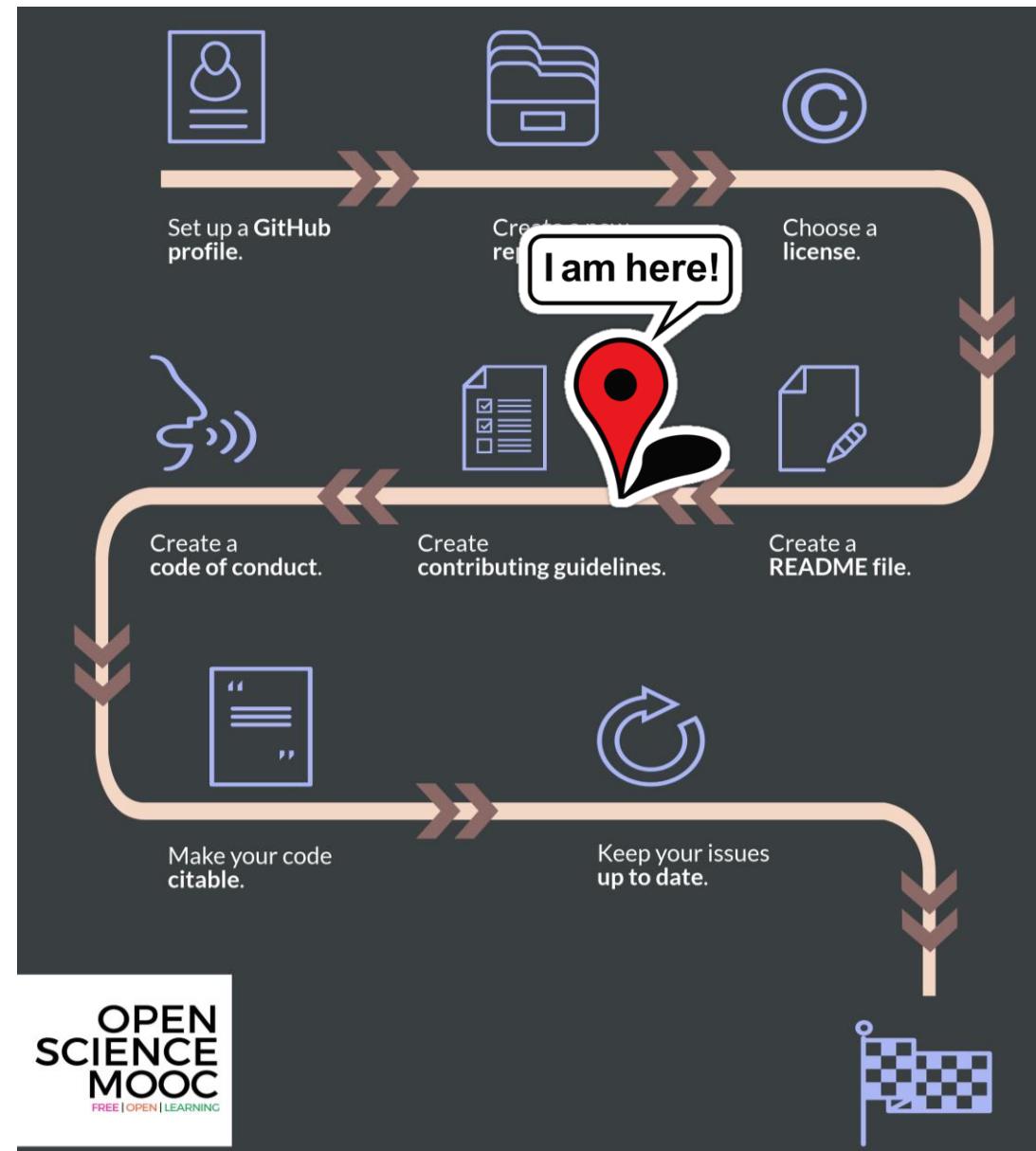


Discovered Open Science MOOC



Got confused!

Getting started with GitHub



Step 2: Go to GitHub and create a repository

The screenshot shows a GitHub user profile interface. At the top, there are navigation links: 'Overview', 'Repositories 4' (which is highlighted with an orange underline), 'Stars 0', 'Followers 2', and 'Following 6'. Below these are search and filter controls: a search bar containing 'Find a repository...', a 'Type: All' dropdown, a 'Language: All' dropdown, and a green 'New' button with a plus icon. The main content area displays a single repository card for 'reproducible-workflows'. The card includes the repository name in blue, a blue circular icon with 'R', a star icon with '2', a license icon with 'MIT License', and the text 'Updated 3 hours ago'. A horizontal green progress bar is visible on the right side of the card.

Overview **Repositories 4** Stars 0 Followers 2 Following 6

Find a repository... Type: All ▾ Language: All ▾ New

[reproducible-workflows](#)

● R ★ 2 MIT License Updated 3 hours ago

Step 3: Give it a useful name

Owner



bapalmer ▾

Repository name

/

Great repository names are short and memorable. Need inspiration? How about [probable-octo-potato](#).

Description (optional)



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.



Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Step 4: Populate the README file

The screenshot shows a text editor window with a title bar 'README.md' and a small edit icon. The main content area contains the following text:

lunchtime_sessions

Short 1 hour introductions to R-related topics such as creating R projects, using GitHub through RStudio and more

Session 1:

R-projects

The foundation to reproducible research starts not with your PI, your lab colleagues or your collaborators. It starts with you! How do you organise your research projects? How do you keep track of the data as you collect it. How readily can you go back to a set of analyses you've performed in the past and re-run to include new variables you may have at your disposal or, perhaps more importantly, confirm that critical calculation you performed 6 months ago?

The answer to many, if not all of these questions, can be found through the use of R/Rstudio and implementation of a defined research project structure made possible with R-projects.

Session 2:

Reproducible reports with R Markdown and knitr

Rmarkdown allows you to move directly from your analyses to the reporting stage thereby eliminating unseen copy/paste errors that inevitably occur when you consider the 1000 copy/paste actions that are likely to be performed every day of every week of every year.

Step 5: Add a licence

bapalmer / reproducible-workflows

Watch ▾ 2

Star 8

Fork 0

Code

Issues 0

Pull requests 0

Projects 0

Wiki

Insights

Settings

Branch: master ▾

reproducible-workflows / LICENSE

Find file Copy path



bapalmer/reproducible-workflows is licensed under the

MIT License

A short and simple permissive license with conditions only requiring preservation of copyright and license notices. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

Permissions

- ✓ Commercial use
- ✓ Modification
- ✓ Distribution
- ✓ Private use

Limitations

- ✗ Liability
- ✗ Warranty

Conditions

- ⓘ License and copyright notice

This is not legal advice. [Learn more about repository licenses](#).

Step 6: Interface git with RStudio

- The key here is to select the correct shell
 - Windows users have four options
 - Only one of these is correct

-/Projects/16.08.08.frank/analysis - RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Console Terminal ×

~/Projects/16.08.08.frank/analysis/ ↵

```
R version 3.4.4 (2018-03-15) -- "Someone to Lean On"
Copyright (c) 2018 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
R is a collaborative project with many contributors.
Type 'contributors()' for more information and 'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
```

> |

Tools

- Install Packages...
- Check for Package Updates...
- Version Control
- Shell...
- Terminal
- Addins
- Keyboard Shortcuts Help Alt+Shift+K
- Modify Keyboard Shortcuts...
- Project Options...
- Global Options... (selected)

Environment History Connections

Import Dataset

Global Environment -

Name	Type	Length	Size	Value
Environment is empty				

Files Plots Packages Help Viewer

New Folder Delete Rename More

Home > Projects > 16.08.08.frank > analysis > scripts

Name	Size	Modified
..	0 B	Aug 10, 2018, 9:16 AM
_Rhistory	8.5 KB	Aug 22, 2018, 4:50 PM
data.R	7.9 KB	Sep 4, 2018, 1:21 PM
data_RespPCT.R	13.3 KB	Aug 27, 2018, 11:30 AM
functions.R	0 B	Aug 10, 2018, 9:16 AM
project.R	3.4 KB	Aug 10, 2018, 9:16 AM
randomization.R	742 B	Sep 2, 2018, 12:22 PM
script_1_initial_data_prep.R	4.2 KB	Aug 30, 2018, 4:48 PM
script_2_data_overview.R	8.8 KB	Aug 27, 2018, 11:57 AM
table1.R	948 B	Aug 10, 2018, 9:16 AM
variables.R		

15:16
05/09/2018

-/Projects/16.08.08.frank/analysis - RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Console Terminal x

-/Projects/16.08.08.frank/analysis/

R version 3.4.4 (2018-03-15) -- "Someone to Lean On"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |

Environment History Connections

Import Dataset

Global Environment

Name Type Length Size Value

Environment is empty

Power

analysis — ~/Projects/16.00.00.frank

Grid C

OK Cancel Apply

Options

General Code Appearance Pane Layout Packages R Markdown Sweave Spelling Git/SVN Publishing Terminal

New terminals open with: Command Prompt (32-bit)
Git Bash
Command Prompt (32-bit)
Command Prompt (64-bit)
Windows PowerShell (32-bit)
Windows PowerShell (64-bit)

Connection

Connect with WebSocket

Miscellaneous

Close terminal when shell exits

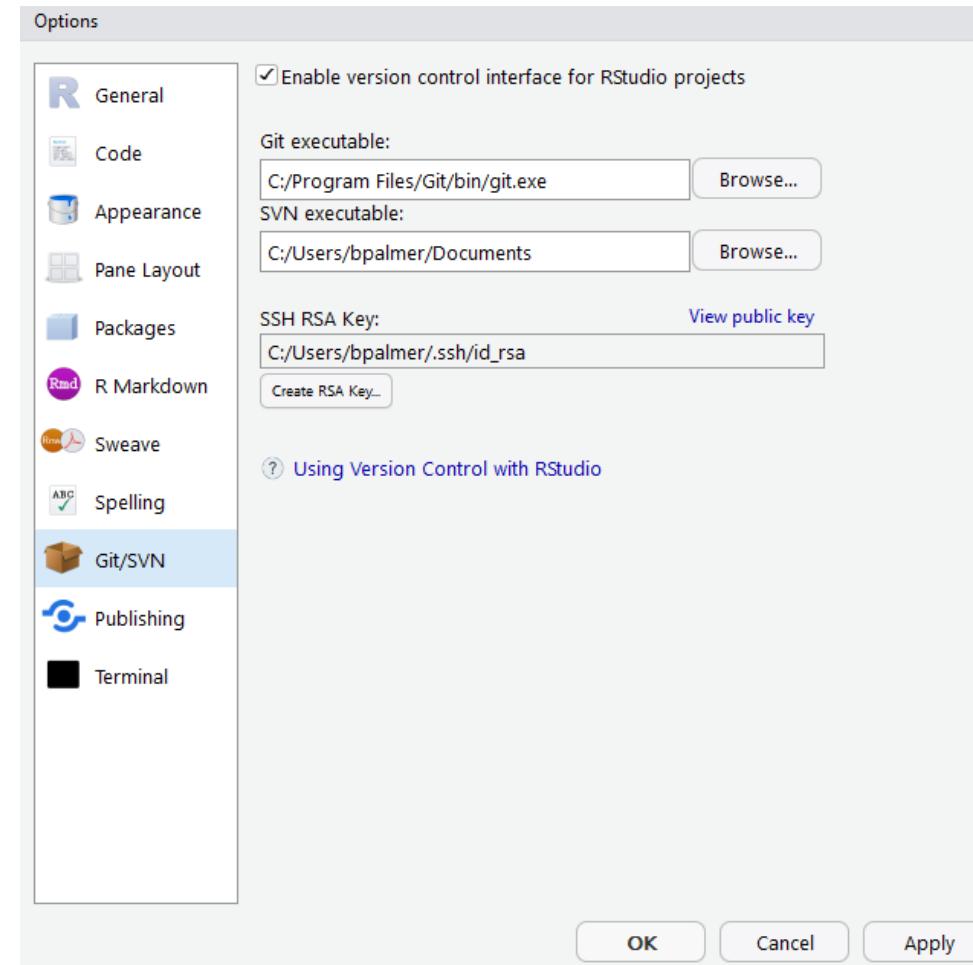
Using the RStudio terminal

Size Modified

0 B Aug 10, 2018, 9:16 AM
8.5 KB Aug 22, 2018, 4:50 PM
7.9 KB Sep 4, 2018, 1:21 PM
13.3 KB Aug 27, 2018, 11:30 AM
0 B Aug 10, 2018, 9:16 AM
3.4 KB Aug 10, 2018, 9:16 AM
742 B Sep 2, 2018, 12:22 PM
4.2 KB Aug 30, 2018, 4:48 PM
8.8 KB Aug 27, 2018, 11:57 AM
948 B Aug 10, 2018, 9:16 AM

15:17 05/09/2018

Step 6: Interface git with RStudio



Step 6: Interface git with RStudio

- Setup a version control R project to interface with the newly created repository

Console Terminal x

~/

```
R version 3.5.1 (2018-07-02) -- "Feather Spray"
Copyright (c) 2018 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
```

Environment History Connections

Import Dataset

Global Environment

Name	Type	Length	Size	Value
Environment is empty				

New Project

Create Project

- New Directory**
Start a project in a brand new working directory
- Existing Directory**
Associate a project with an existing working directory
- Version Control**
Checkout a project from a version control repository

Cancel

Viewer

Console Terminal

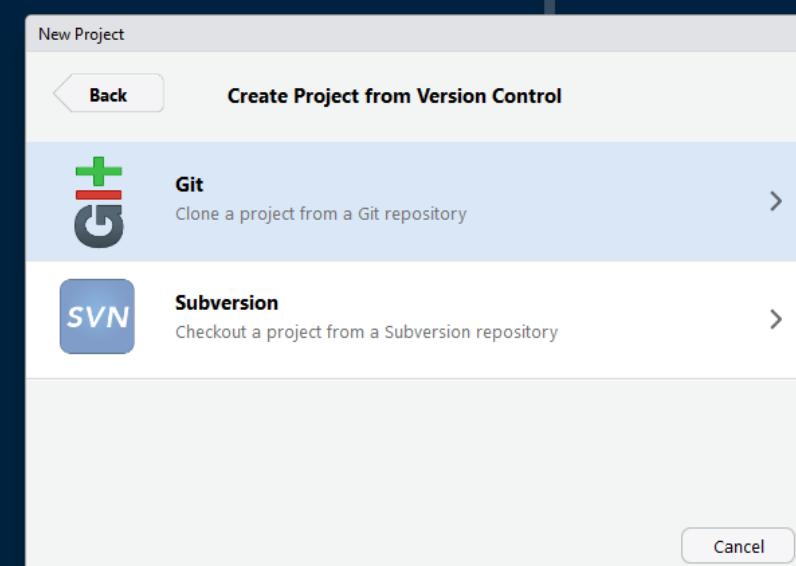
```
R version 3.5.1 (2018-07-02) -- "Feather Spray"  
Copyright (C) 2018 The R Foundation for Statistical Computing  
Platform: x86_64-w64-mingw32/x64 (64-bit)
```

```
R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.
```

```
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.
```

```
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.
```

> |



Environment History Connections

Import Dataset

Global Environment

Name	Type	Length	Size	Value
Environment is empty				

Go to file/function

Addins +

Project: (None) ▾

Console Terminal

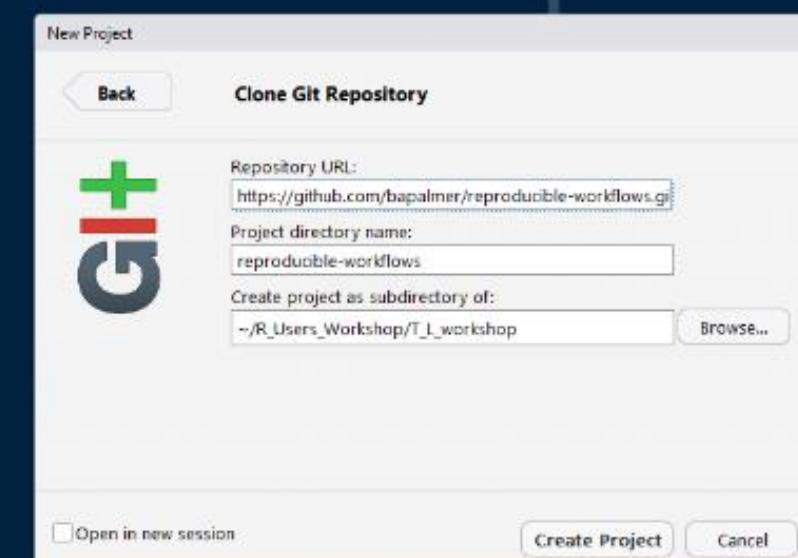
R version 3.5.1 (2018-07-02) -- "Feather Spray"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>



Environment History Connections

Import Dataset

Global Environment

Name	Type	Length	Size	Value
------	------	--------	------	-------

Environment is empty

Viewer

R ~R_Users_Workshop/T_L_workshop/reproducible-workflows - master - RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Console Terminal x

Terminal 1 : /c/Users/bpalmer/Documents/R_Users_Workshop/T_L_workshop/reproducible-workflows/reproducible-workflows

```
bpalmer@ph-wgb5 ~/Documents/R_Users_Workshop/T_L_workshop/reproducible-workflows (master)
$ git clone https://github.com/bpalmer/reproducible-workflows.git
Cloning into 'reproducible-workflows'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), done.

bpalmer@ph-wgb5 ~/Documents/R_Users_Workshop/T_L_workshop/reproducible-workflows (master)
$ cd reproducible-workflows

bpalmer@ph-wgb5 ~/Documents/R_Users_Workshop/T_L_workshop/reproducible-workflows/reproducible-workflows (master)
$ ls
LICENSE README.md

bpalmer@ph-wgb5 ~/Documents/R_Users_Workshop/T_L_workshop/reproducible-workflows/reproducible-workflows (master)
$ head README.md
# reproducible-workflows
bpalmer@ph-wgb5 ~/Documents/R_Users_Workshop/T_L_workshop/reproducible-workflows/reproducible-workflows (master)
$ git remote show origin
* remote origin
  Fetch URL: https://github.com/bpalmer/reproducible-workflows.git
  Push URL: https://github.com/bpalmer/reproducible-workflows.git
  HEAD branch: master
  Remote branch:
    master tracked
  Local branch configured for 'git pull':
    master merges with remote master
  Local ref configured for 'git push':
    master pushes to master (up to date)

bpalmer@ph-wgb5 ~/Documents/R_Users_Workshop/T_L_workshop/reproducible-workflows/reproducible-workflows (master)
$
```

Environment History Connections Git

Import Dataset

Global Environment

Name	Type	Length	Size	Value
Environment is empty				

Files Plots Packages Help Viewer

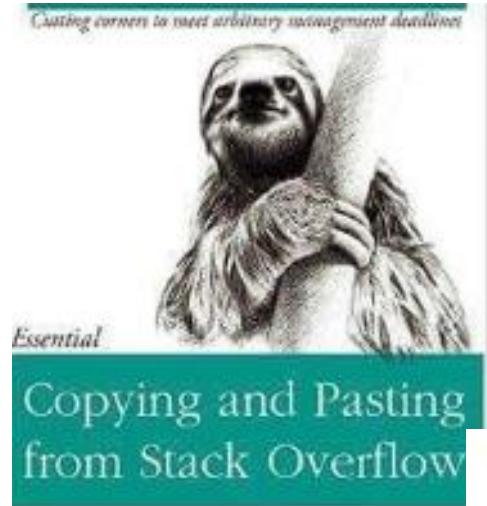
New Folder Delete Rename More

Home > R_Users_Workshop > T_L_workshop > reproducible-workflows

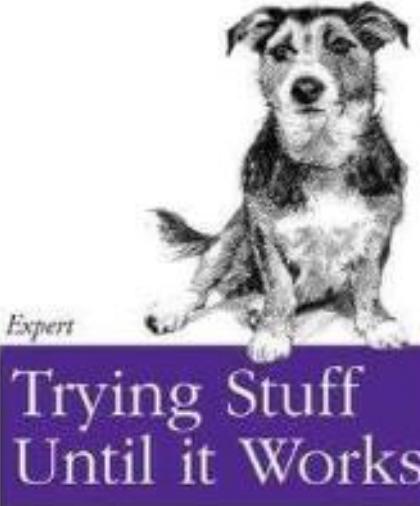
Name	Size	Modified
..		
.gitignore	44 B	Oct 1, 2018, 11:27 AM
LICENSE	1.1 KB	Oct 1, 2018, 11:27 AM
README.md	24 B	Oct 1, 2018, 11:27 AM
reproducible-workflows.Rproj	218 B	Oct 1, 2018, 11:27 AM
reproducible-workflows		

11:33
01/10/2018

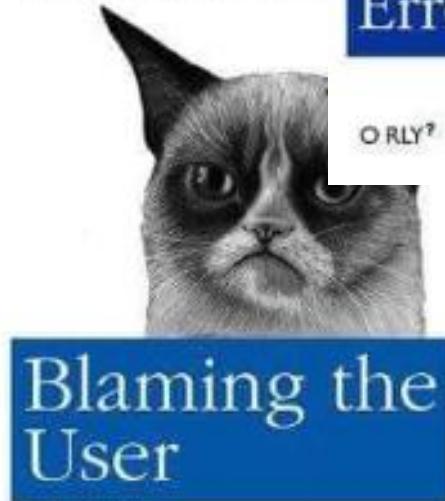
From the people who brought you.....



Software can be chaotic, but we make it work

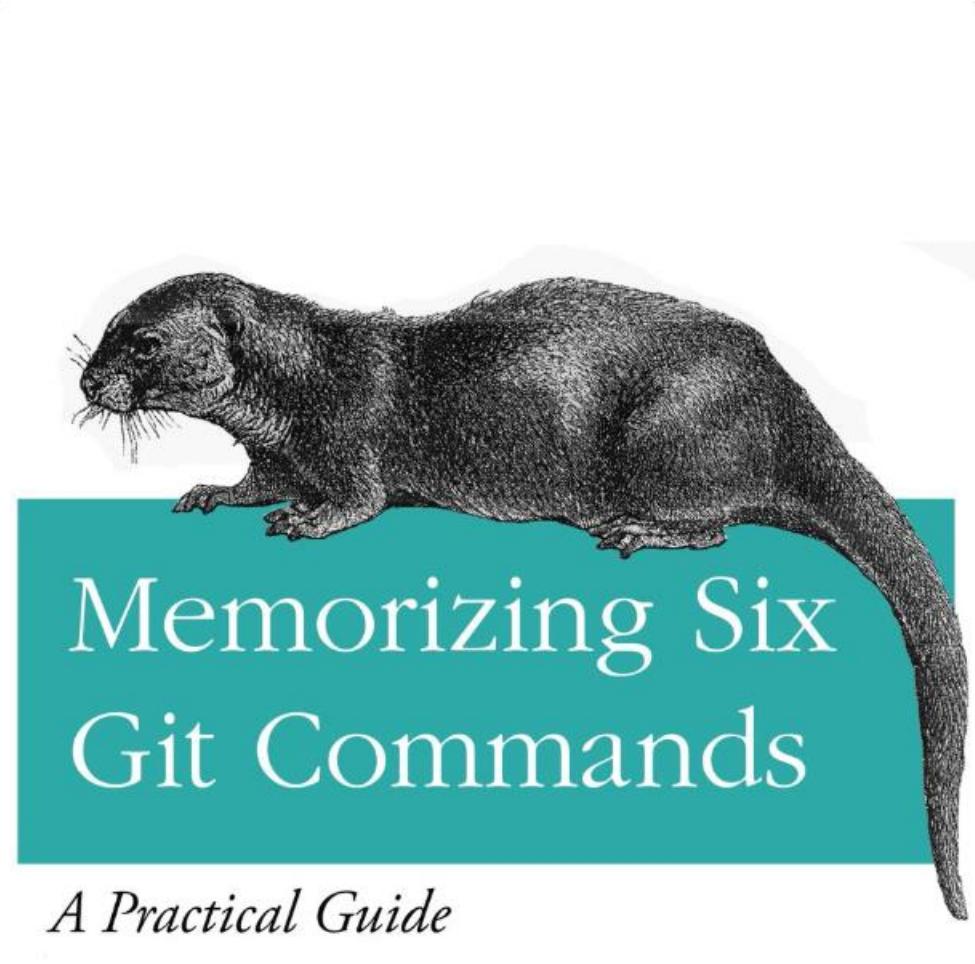


To actually learn any new programming concept



@ThePracticalDev

I know git, depending on what you mean by “know”



JD Long
@CMastication

Following

Um... So what are the other 2?



DEV Community 🧑‍💻🧑‍💻 @ThePracticalDev

I know Git, depending on what you mean by "know".

6:39 PM - 19 Aug 2018

1 Retweet 36 Likes



Q 5

1

36



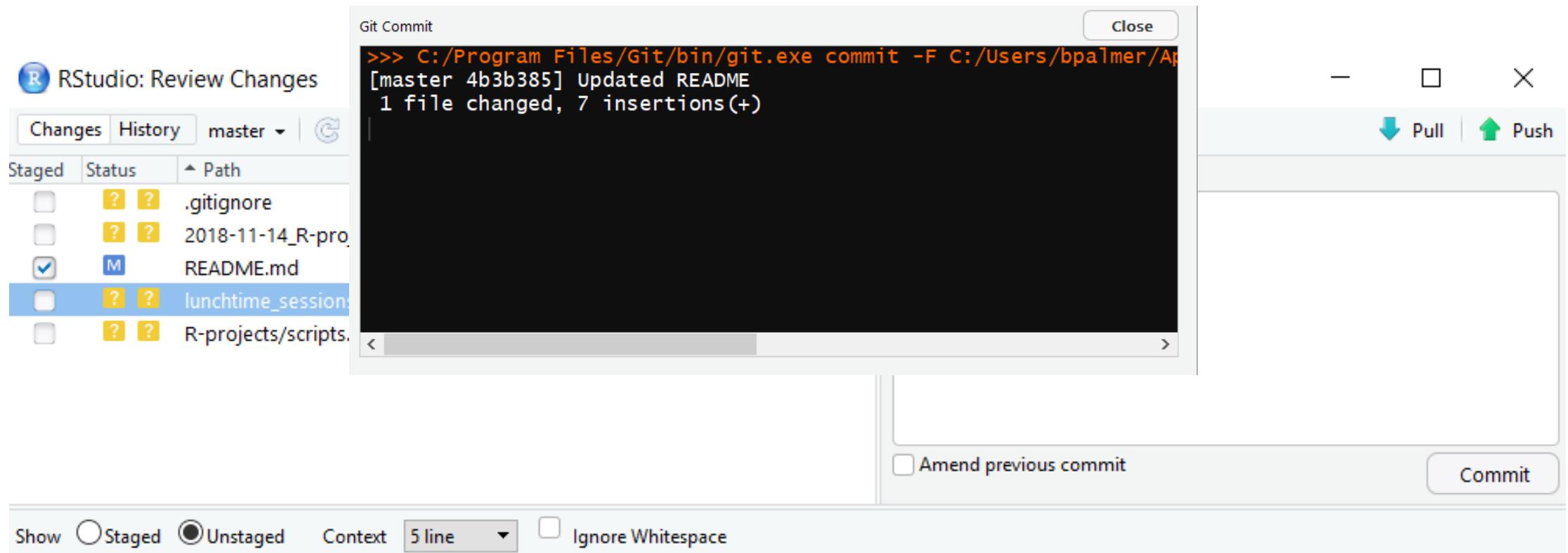
JD speaks the truth

4 main commands just to get started with.....

- **Add:** This is where you identify the files you wan to upload
- **Commit:** This is like “saving” your work by creating a new version or copy
- **Push:** This is how you send files from your local project to the online repository
- **Pull:** This is how you get files from your online repository to your local project

But I don't know anything about command line

- The good news here is that you don't need to.....



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 MEMORIZIZE 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.



Obviously there are others

Initialise

Checkout

Clone

Fork

Branch

Master

Index

Head

Merge

Stage

Issue

Fetch

Clean

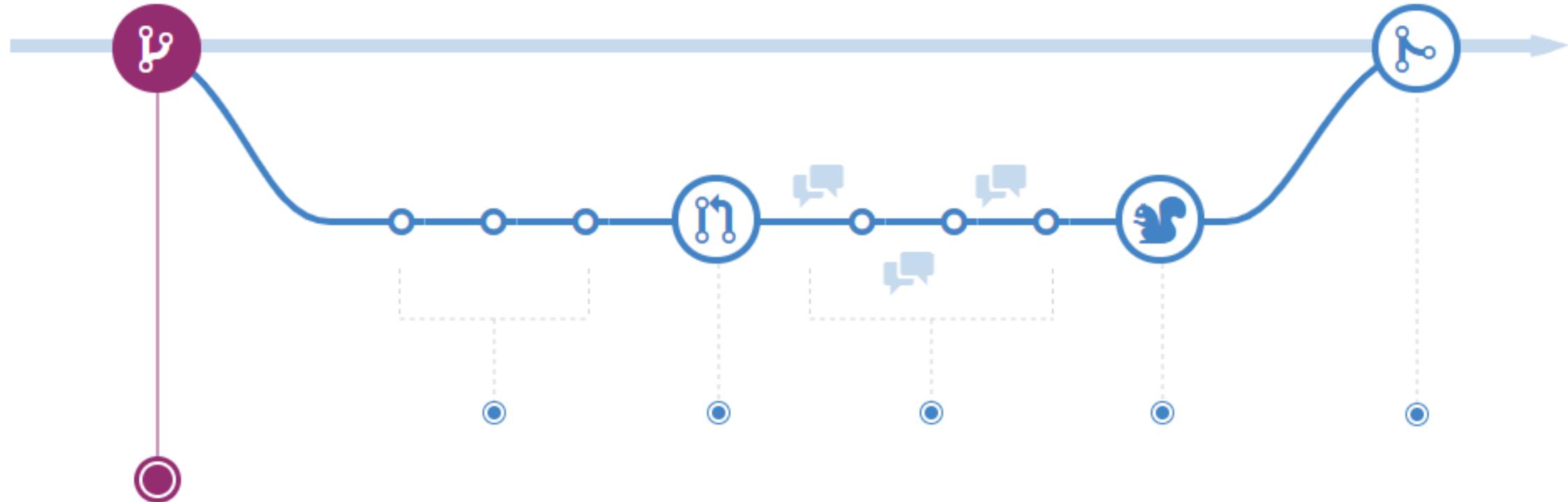
Check

Status

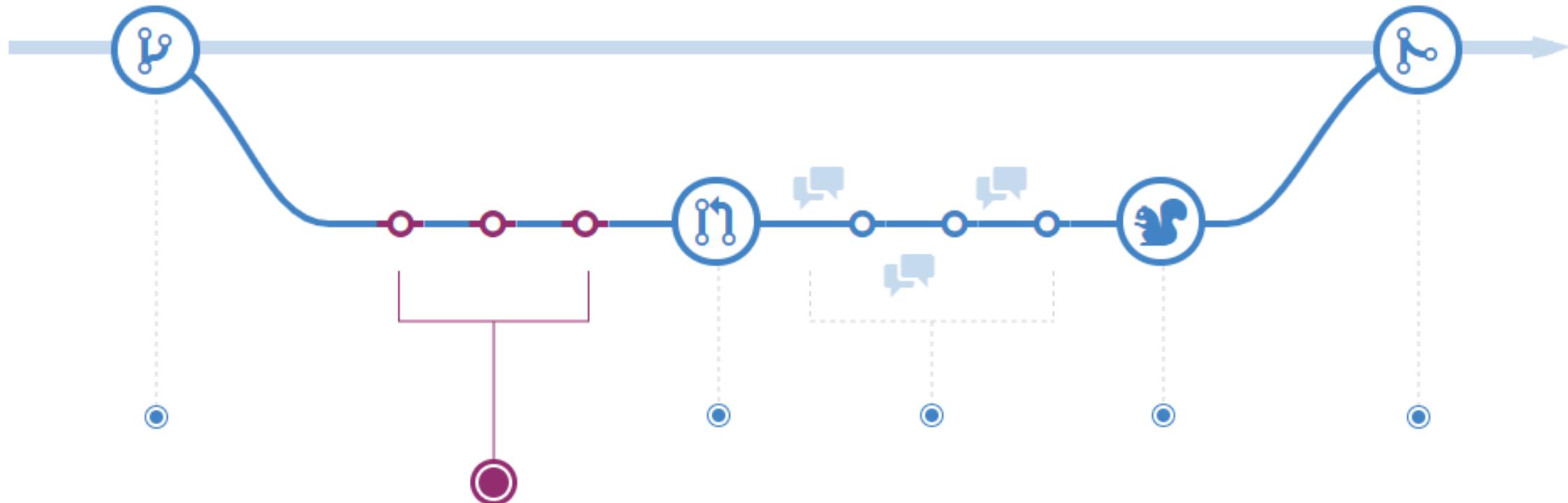
Commit message

Working directory

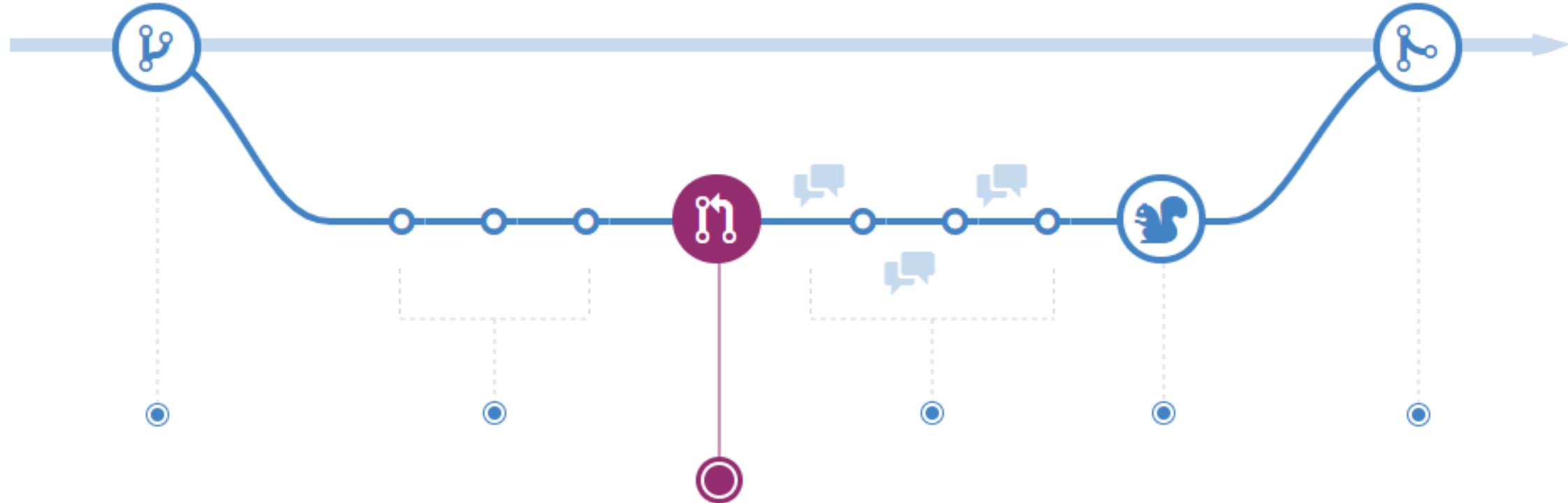
Understanding the GitHub flow:



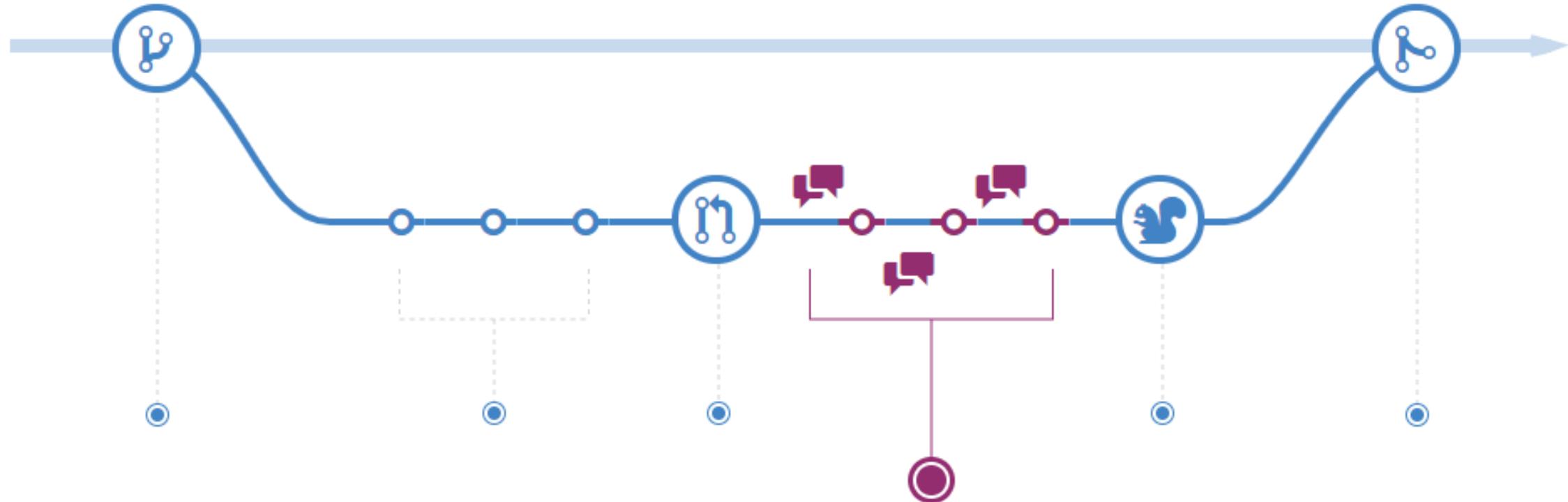
Understanding the GitHub flow: Commit



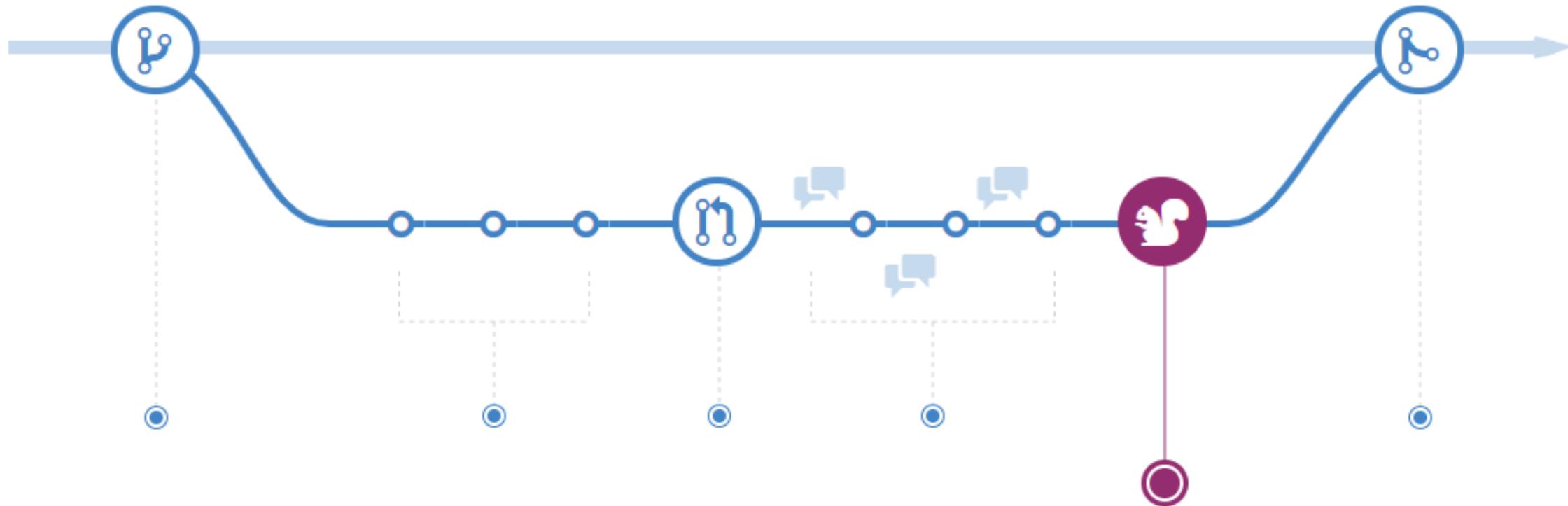
Understanding the GitHub flow: Pull request



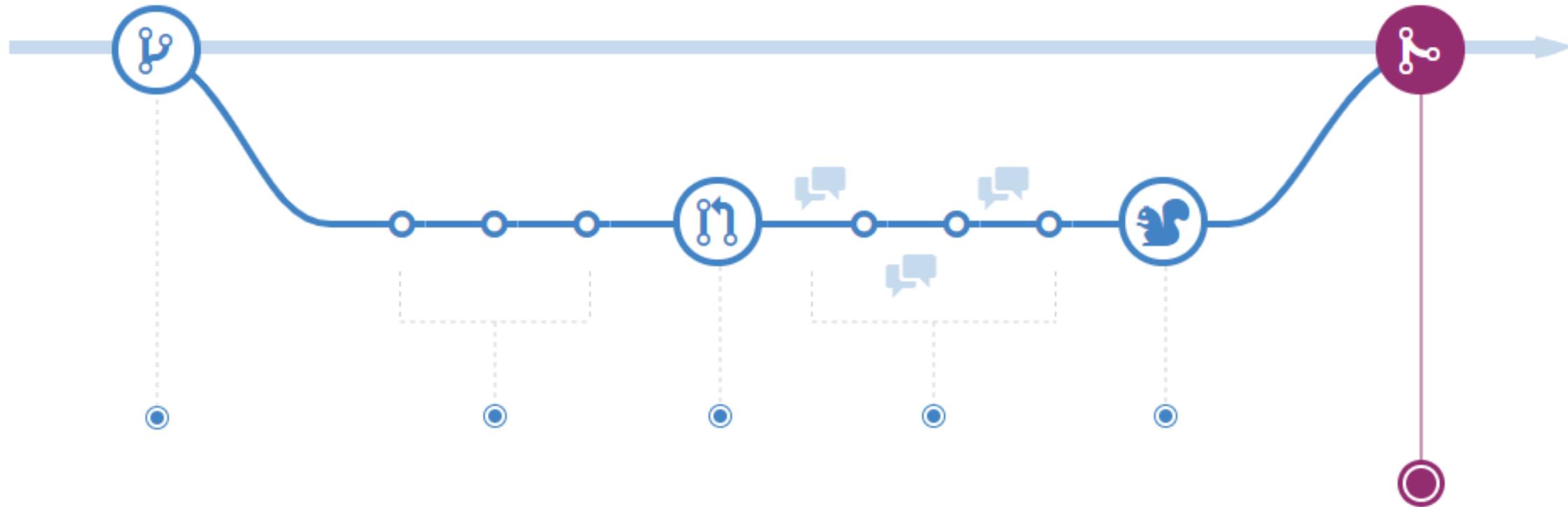
Understanding the GitHub flow: Review



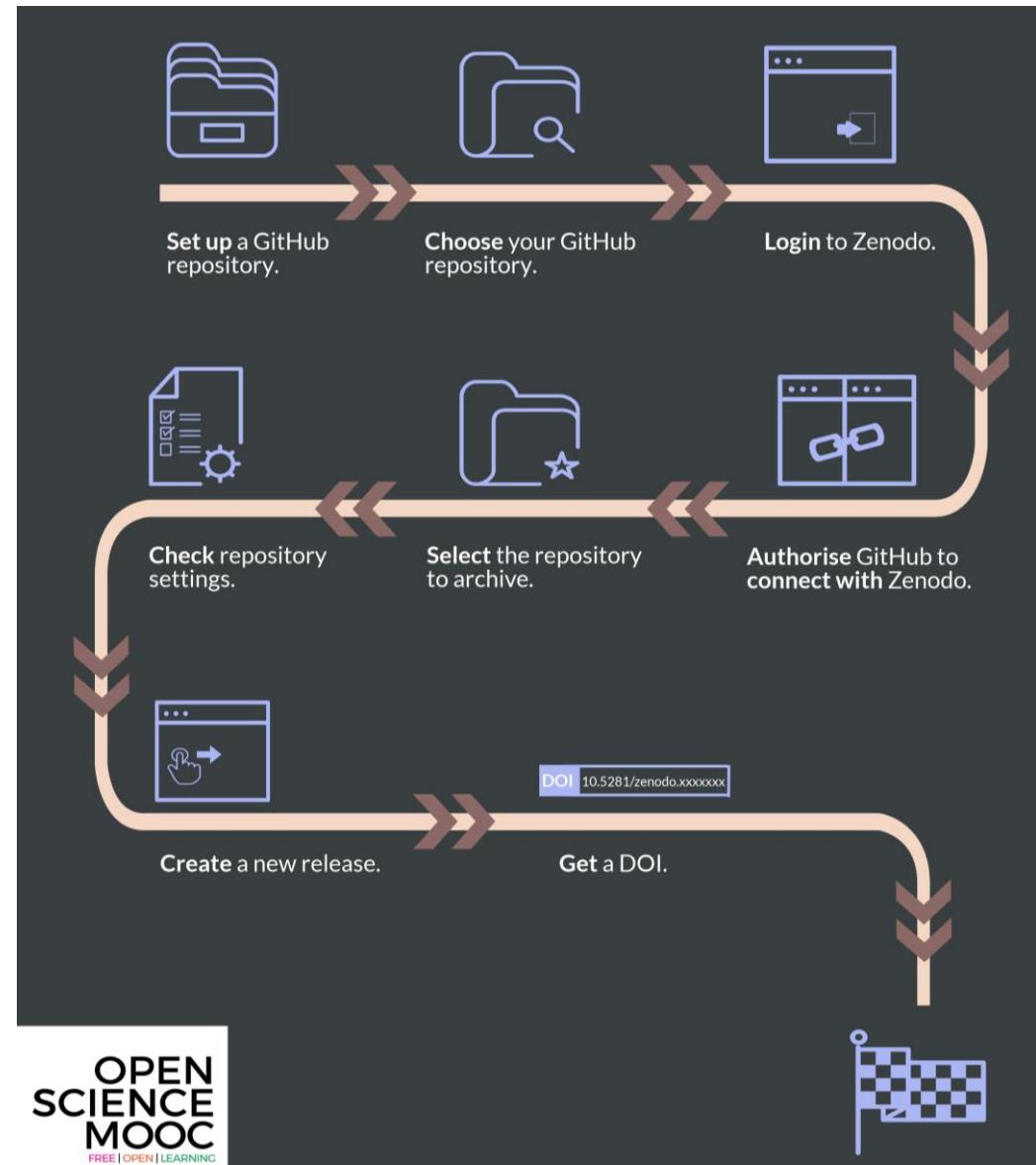
Understanding the GitHub flow: Deploy



Understanding the GitHub flow: Merge



Finishing up



Try it for yourself: Tasks 1-3

OPEN SCIENCE MOOC

FREE | OPEN | LEARNING