

Teaching Survival Git

Enough Git to Survive as a Data Scientist

Rob Donald

Saturday 01 May 2021

Contents

Introduction	1
Moving from SVN to Git	1
Git setup in general	1
Lightbulb Moments	2
Switching back and forth between main and branches	2
Merge conflicts	2
Daily routine	2
How Do I Compare Two PDF Output Files	2
Getting Info	3
Order the branches by time and give me some context	3

Introduction

My notes on using git as a data scientist. In other words you *know and accept* you should be using git but man is it confusing and almost painful. “_(ツ)_/”

I’ve been a software developer and data scientist for more than 30 years and I’ve used all sorts of version control systems (started with Apollo DSEE, look that one up). I find git confusing.

Moving from SVN to Git

- <https://smartbear.com/blog/migrating-from-subversion-to-git-lessons-learned/>
- <https://dev.to/fpuffer/what-makes-companies-switch-from-svn-to-git-3687>
- <https://ohshitgit.com/>

Git setup in general

- <https://happygitwithr.com/>

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

Lightbulb Moments

Switching back and forth between main and branches

This is from the world famous Jenny Bryan ¹

- <https://happygitwithr.com/git-branches.html> Section 22.2

So you *can* do it and yes git **stash** is a way but she recommends the same idea I had read somewhere else about *always* doing a commit even if it is just with the comment ‘WIP xxx’. But she then goes on to explain how you can get rid of that temporary commit using terminal (bash) commands like:

```
$ git checkout issue-5
$ git reset HEAD^
```

Merge conflicts

- <https://happygitwithr.com/git-branches.html> Section 22.4

Daily routine

```
1 # fetch the changes from the remote
2 $ git fetch origin
3
4 # show commit logs of changes
5 $ git log master..origin/master
6
7 # show diffs of changes
8 $ git diff master..origin/master
9
10 # apply the changes by merge..
11 $ git merge origin/master
12
13 # .. or just pull the changes
14 $ git pull
```

How Do I Compare Two PDF Output Files

Well you would think this is easy eh? But the problem is the fact that it is a .pdf file i.e. **not** a text file. So what you need to do is get the older version’s commit SHA1 and then extract that particular commit to somewhere different. Note this command below is extracting the *whole folder structure* at the point in time from your SHA1.

From this link:

¹ If you don’t know who she is look at this link <https://www.rstudio.com/speakers/jenny-bryan/> and follow her on twitter: @JennyBryan.

- <https://stackoverflow.com/questions/11018411/how-do-i-export-a-specific-commit-with-git-archive>

```
$ git archive --format zip --output ./CompareResults/Cmp-01.zip 4081d45d
```

Using the above command you have the whole repo as at SHA1 4081d45d in the zip file Cmp-01.zip

You can of course just pull out a specific file using this command:

```
$ git archive --format zip --output ./CompareResults/SingleFile.zip 4081d45d Prospec_Stg_1_N69_ELvsDP.p
```

Getting Info

Sometimes you just wish you could ask a simple question “_(ツ)_/”.

Order the branches by time and give me some context

```
$ git branch -vv --sort=-committerdate
```