	Environmental Analysis Teaching and Research Laboratory	Date: X/XX/XXXX	Number: 06 v.01
	Standard Operating Procedure	Title: Rstudio Projects and Github	
	Approved By: TBD	Revision Date: August 12, 2016	

1. Scope and Application

1.1 The scope of this SOP is train researchers...

1.2 The applications of this SOP are for...

2. Summary of Method

2.1 This SOP does this...

2.2 The use of Github is not intuitive. So, you will find many on-line resources to help.

Contents

3. Acknowledgements

4. Definitions

4.1 R is a powerful, open source program but combined with RStudio and Github the program becomes an archetype of a program that enables 1) collaboration, 2) transparency, and 3) accessibility.

4.2 RStudio is use interface for R. Although R by itself is an amazing example of crowd sourcing, where a wide range of staticians and programmers have created a free programming environments with a robust range of statistical packages, the RStudio interface provides a user with the tools to track and publish their analysis process in an effecient and transparent way.

4.3 Local Install versus Server — R and RStudio can be installed on a local computer/laptop from the CRAN download mirror sites. However, we also have access to the R and RStudio Server installed on the Pomona College mainframe, where you can access it via a web browser. Wow, this is conveient!

4.4 GitHub is a web-based Git repository hosting service.

4.5 Version Control is a method to track changes in software, and often in the context of collaborative projects. The final component of R and RStudio is its capacity to create projects (RStudio's terminology) and repositories (Github's terminalogy) that can be shared among collaborators. In particular, the collaboration allows for contributions to be tracked via version control tools. There are a number of ways that we can access these tools, but we'll try to limit the methods so keep the process relatively "simple".

However, becoming facile in using these program is like learning how to walk. We need to approach this process in descrete steps. Some will require weeks of mistakes to learn, others will quickly learn to run with the programs. How quickly you can feel comfortable with these programs will depend on many factors, but will be greatly improved by the time you invest!

4.1. Rational

Almost Universal Compatibility No downloading programs on personal computers.

Accessibility Accessible from any computer via webserver

Collaboration

Version Control

4.6 I recommend reading the following sites to better understand Git and Github

- Github for Beginners

- Github Fw

- Understanding the GitHub Flow

- http://blog.osteele.com/posts/2008/05/my-git-workflow/

5. Interferences

5.1 R has an updated version about every six months. When performing advanced analyses, there are times that new versions will no longer run a code. Thus, older versions of R must be maintained. This is a unheard of issue for new users.

5.2 RStudio Server needs a functional network connection. If the network is down, then Rstudio Server is inaccessible. This can be a source of frustration.

6. Health and Safety

6.1 Describe the risk...

Safety and Personnnel Protective Equipment

7. Personnel & Training Responsibilities

7.1 Researchers training is required before this the procedures in this method can be used...

7.2 Researchers using this SOP should be trained for the following SOPs:

- SOP01 Laboratory Safety

- SOP02 Field Safety

8. Required Materials and Apparati

8.1 Laptop or destop computer

8.2 Access to Pomona College's SSO and server

8.3 Github account

8.4 Patience

9. Estimated Time

9.1 This set up procedure requires 30 minutes.

10. Maintaing XX Data

11. Procedure

11.1 Create Github Account: Go to Github.com and create an account. I suggest you use your Claremont e-mail adddress because this can come in handy later, should you want to create private repositories, which are free for college students! Your email address can be used to prove your student status.

11.2 Read the Readme file – THIS DOCUMENT WILL REPLACE THAT README FILE.

11.1.Creating a Connection between Github and RStudio

11.3 To use Rstudio's version control, you need to create a SSH Key that is used to create secure connections between RStudio and Github.

11.4 To create an SSH Key, go to RStudio/Tools/Global Options and select "Create an SSH key" NEED MORE INFO HERE, SCREEN SHOT? Copy this SSH key.

11.5 Add SSH Key to Github

1. Go to Profile?/Settings

2. Select the left menu on the button "SSH and GPG key"

3. Click on new SSH key, type in name of key (e.g. 'myRstudio SSH key')

4. Paste in the RSA key into the window below.

5. Hit green "add SSH key" button.

11.6 Now Github and R can communicate.

11.7 To obtain access to respository in Rstudio, you will need to "clone" the site as a new project in Rstudio. To accomplish this...

11.2.Creating a New Repository

11.3.Creating a New Project

12. Working on Projects

12.1.Rstudio Panels

File Structure and Rnw/Tex files

Console and Compile PDF windows

Git Panel

12.2.Best Practices

Pull When you open RStudio, the first thing you should do is "Pull" from the repository to ensure your files are up-to-date. When you "Pull", you will get one of three results:

Already up-to-date. This means that your files have not been updated on the Github site since you last pulled the files. If you suspect someone has worked on the files, but are not getting those changes, it means that your collaborator has failed to "Push" these changes onto the repsository. If this is the case, you might need to go to the troubleshooting question to address this problem.

Successful Updates If your files are successfully updated from the repository, you will see:

summary of updates

master 6038765 Started to explain set up procedures

5 files changed, 434 insertions(+), 8 deletions(-)

create mode 100644 06_Rstudio_Github/Rstudio-and-Github-concordance.tex

create mode 100644 06_Rstudio_Github/Rstudio-and-Github.log

create mode 100644 06_Rstudio_Github/Rstudio-and-Github.pdf

create mode 100644 06_Rstudio_Github/Rstudio-and-Github.tex

Unsuccesful "Pull"

Commit

Push As you might guess, when you "Push" you can also have several outcomes:

Everything up-to-date is certainly simple.

Successful "Push" "To git@github.com:marclos/SOPs.git

e2efe6f..0b18250 master -> master"

Unsuccessful "Push"

Basically git commit "records changes to the repository""" while git push "updates remote refs along with associated objects". So the first one is used in connection with your local repository, while the latter one is used to interact with a remote repository.

Here is a nice picture from Oliver Steele, that explains the git model and the commands:

13. Troubleshooting

13.1 At first when you're a newcomer to working on projects within Rstudio connected to a GitHub repository, you may forget the "best practices" of pulling, committing and pushing described above. You might have already gone through all the steps involved in setting up your GitHub account, linking your workspace in Rstudio with the correct project, and begun work on a specific file, but if you forget to update your workspace each time you come back to modify a file (especially in the case

Git Data Transport Commands

http://osteele.com
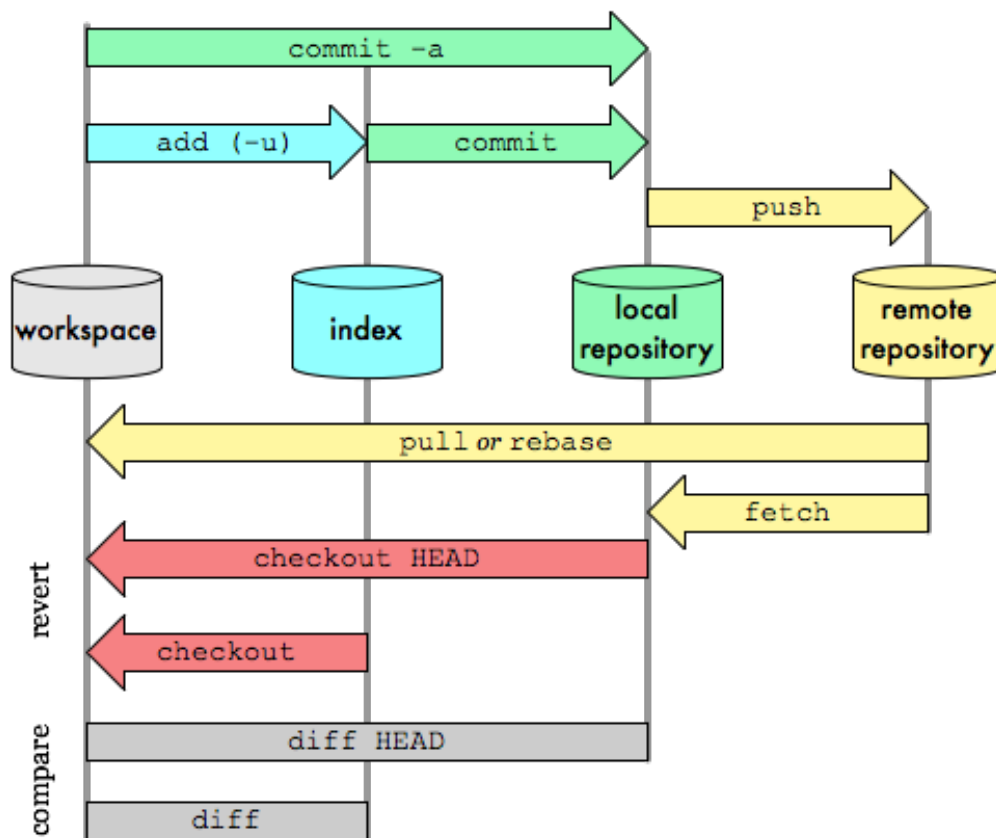
of coming back the next day to a file in Rstudio and forgetting to pull changes other collaborators may have made), you will run into problems committing, and pushing your changes. In this most common case, it's not likely that you'll notice any problems until you try to commit your changes. Thus, we'll begin this section on trouble shooting problems with Github and Rstudio at the committ level.

13.1."Commit" problems

13.2 In the scenario described above, you will get an error code when trying to committ your new changes to a file.

13.2."Pull" is rejected

13.3 Merging changes... how??

13.3.Dealing with non-fast-forward errors

13.4 Sometimes, Git can't make your change to a remote repository without losing commits. When this happens, your push is refused. If another person has pushed to the same branch as you, Git won't be able to push your changes.

13.5 You can fix this by fetching and merging the changes made on the remote branch with the changes that you have made locally:

- $ git fetch origin

- # Fetches updates made to an online repository

- $ git merge origin YOUR_BRANCH_NAME

- # Merges updates made online with your local work

13.6 Or, you can simply use git pull to perform both commands at once:

13.4."Push" fails

13.7 Below are several potential remedies:

merge asdfasdf

Deleting a Project in R Studio If you are willing to sacrifice the changes you made or have mailed them to a collaborator to deal with you can delete the entire project in Rstudio. To accomplish this delete all files in directory, clear workspace and console, don't save, then go to session tab: Terminate R, and hopefully that will do the trick. Then commence with starting a new project.

14. Collaboration and Version Control

14.1. Workflow tracking

14.1 Collaborators can create ways that each one is responsible for certain activities..

14.2 Branching...

15. References

15.1 APHA, AWWA. WEF. (2012) Standard Methods for examination of water and wastewater. 22nd American Public Health Association (Eds.). Washington. 1360 pp. (2014).