Happy Scientist Seminar Series

Introduction to Collaborating with GitHub

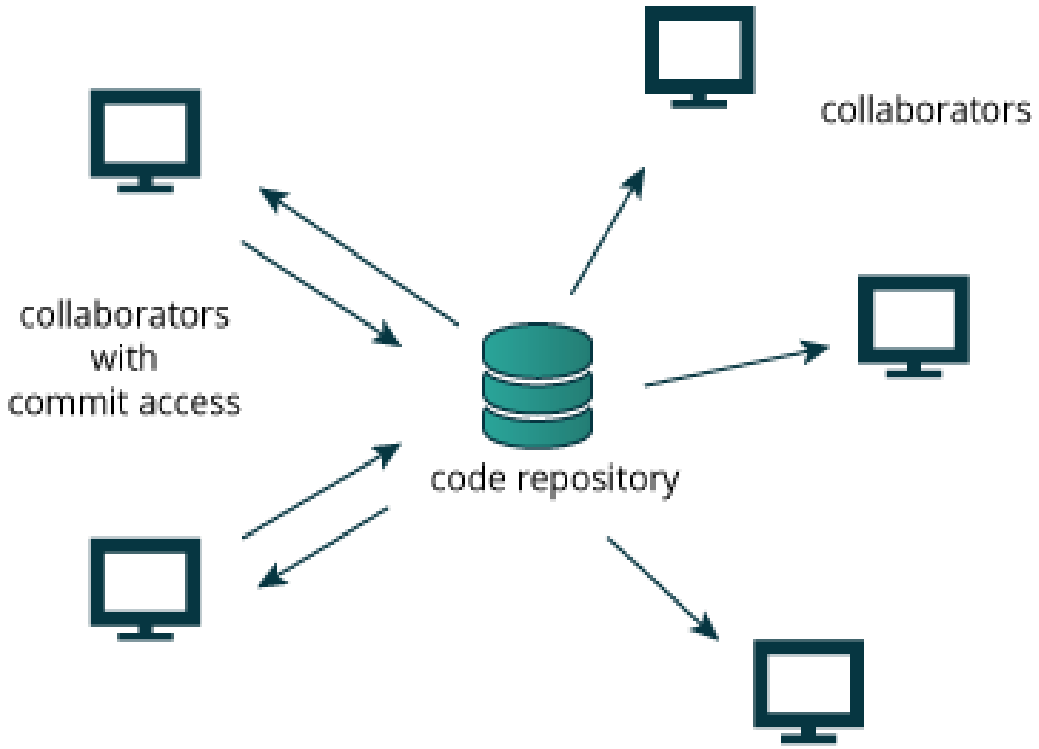
What is Git?

Git is software.

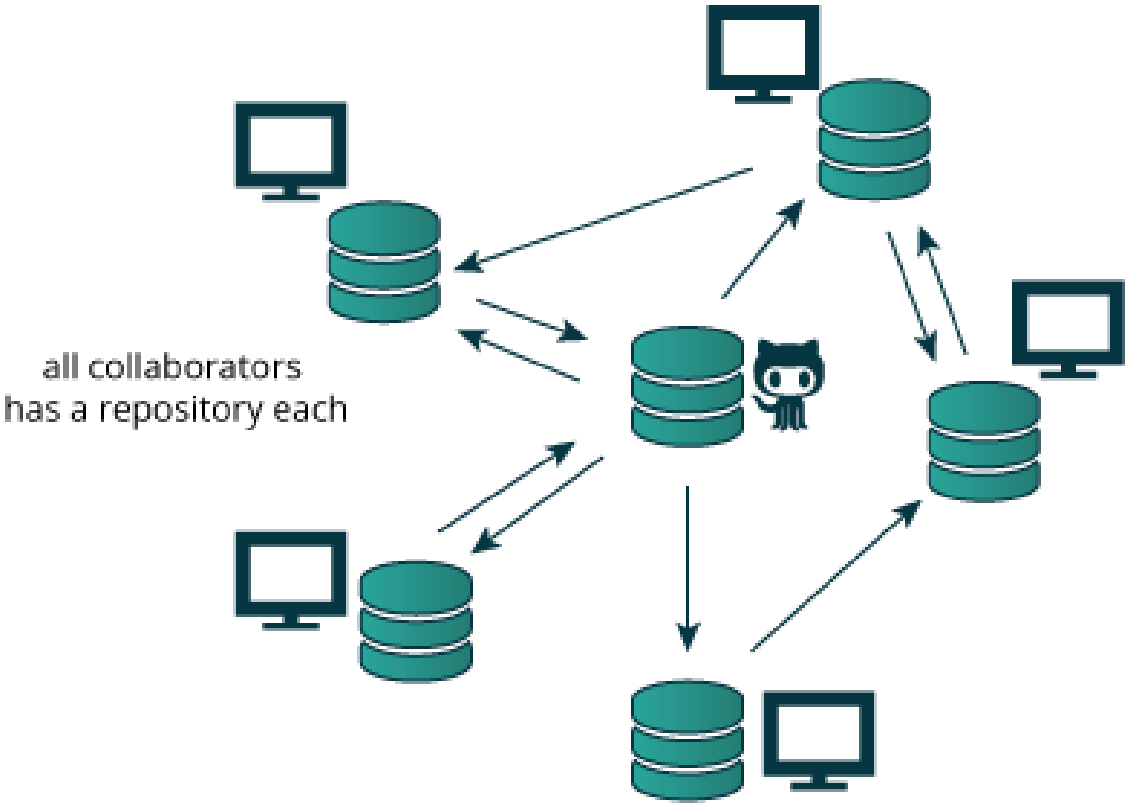
Git is a version control system that is used for software development and other version control tasks. As a distributed revision control system it is aimed at speed, data integrity, and support for distributed, non-linear workflows.

As with most other distributed version control systems, and unlike most client–server systems, every Git directory on every computer is a full-fledged repository with complete history and full version-tracking capabilities, independent of network access or a central server.

Centralized Management



Decentralized Management



What is GitHub?

GitHub is a service.

GitHub · Preferred description

GitHub GitHub is a web-based Git repository hosting service. It offers all of the distributed revision control and source code management (SCM) functionality of Git as well as adding its own features. Unlike Git, which is strictly a command-line tool, GitHub provides a Web-based graphical interface and desktop as well as mobile integration.

Why Use GitHub?

- Revision Tracking – Can track changes and revert to previous versions
- Collaboration – Multiple people can work on the same repository simultaneously
- Distribution – A repository can be made public allowing others to download your files
- Backup – Not the intent of GitHub but if something happens to your computer you can recover the repository from GitHub or another computer that has the repository

GitHub Terms

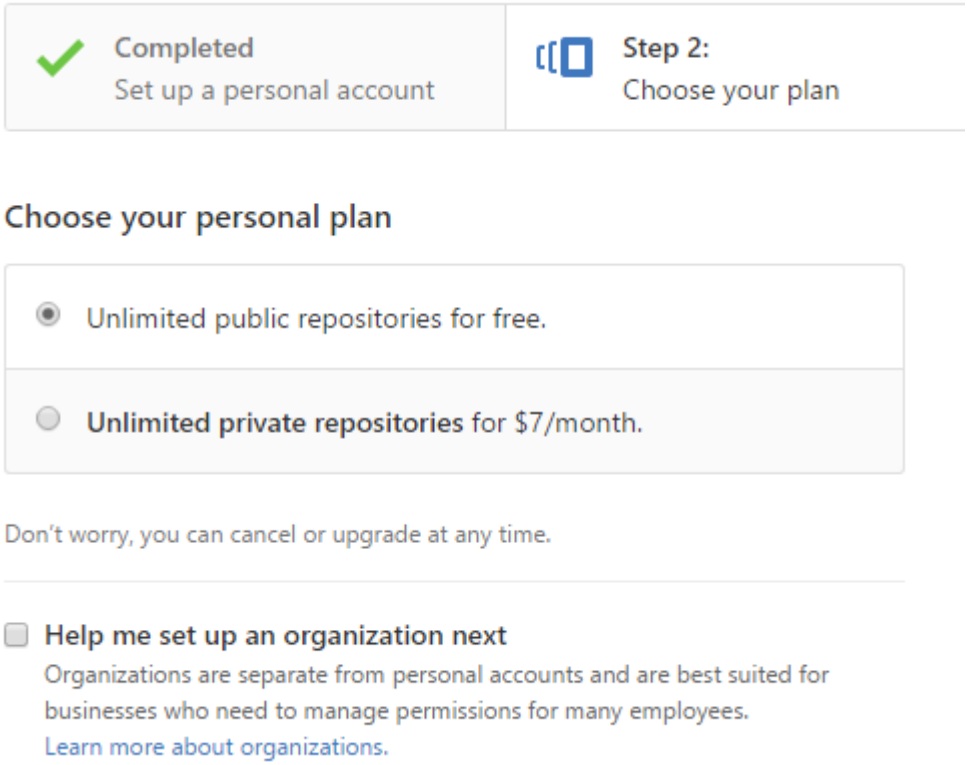
- Organization – A type of account that has several people as members
- Repository – A set of related files that are having their revisions tracked
- Public Repository – A repository can be read by anyone
- Private Repository – A repository that can only be read by a specified group of people
- Team – A group of people in an organization that have been grouped together for a specific purpose
- Project – A job several people can be working on. A repository may have more than one project. Quite often people refer to a repository as a project. This is even done on GitHub's website.

Creating an Account on GitHub

- Go to http://github.com
- Fill in the boxes and click Sign up for GitHub.

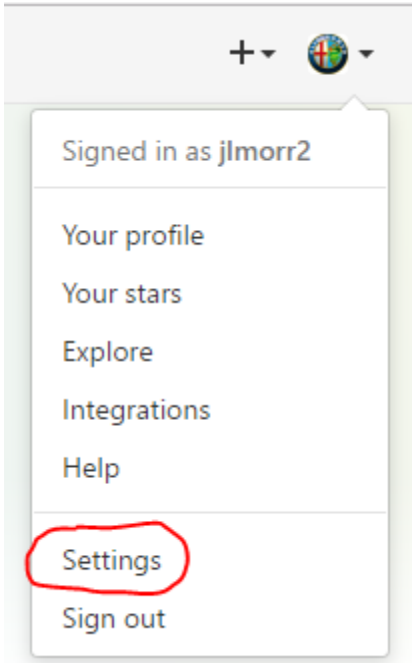
Creating an Account on GitHub

- Selecting account type



Creating an Account on GitHub

- Check your e-mail. You should have received an e-mail from GitHub.
- Click on the verify e-mail address in the e-mail.
- Update your account settings. This is done by selecting the strange icon in the upper right and selecting Settings.



Joining the USCbiostats Organization

- Send me an e-mail, jmorr@usc.edu asking to join. Include your GitHub account name.
- I will generate an invitation to join the organization in GitHub.
- You will receive the invitation from GitHub.
- Click on the Join button in the e-mail. You will be sent to the GitHub website. Log on if necessary. Click on the Join button.

Creating a Repository

- On the first page when you log on there is a button that says “Start a Project”. It should say “Create a Repository”. Project is sometimes used to mean repository but a repository can consist of several projects.
- You can create repositories in your own account. These will be public.

Creating a Repository in the USCbiostats Organization

Creating a repository in the USCbiostats organization requires special permissions. You will need to ask someone with these permissions to create the account. You will need to provide the following information.

- Name of the repository
- Team or people that will need access
- Is this a public or private repository

Tools for Accessing the Repository

	Mac	Windows	Linux
SmartGit	●	●	●
git-cola	●	●	●
GitEye	●	●	●
GitKraken	●	●	●
GitHub Desktop	●	●	
Tower	●	●	
SourceTree	●	●	
GitX-dev	●		
GitBox	●		
GitUp	●		
Fork	●		
Git Extension		●	
Aurees		●	
giggle			●
gitg			●

Green – Commercial product with free non-profit version
Red – Commercial product

Development Tools that can Access Git Repositories

Some development tools can be configured to access Git Repositories.

- R Studio
- Eclipse
- Atom
- Jupyter
- Visual Studio 2015

Show and Tell

- Tour of GitHub
- GitHub Desktop
- Source Tree
- R Studio