



git

SAS

Using git Version Control

Overview

We use “git” to keep versions of images in a central “cloud” location.

Developers new to the project “clone” this image in its entirety.

It is possible to get just the recent images related to the project.

You make your local copy and do some work.

1. New files
2. Modify existing files

Several modifications, until you are ready to “add” those changes back into the mainstream.

Then you “commit” to the files you added, adding a brief message about what was done. This updates your local copy.

Then you “push” your changes to the remote location to be available for all.

So:

1. Clone - make a copy of a repository, that is a repository in its own right.
2. Work - Do work in the directory with an eye to permanent changes.
3. Add - Add the new/updated to the local list of things to make permanent in the local repository.
4. Commit - Commit those changes locally.
5. Push - Push the changes out to the rest of the world. It's on you to make sure you don't break anything.
6. Pull - Get other people's changes down to the local level.

With git, the main development “thread” is called the “master branch”¹ and is usually the stable path for the code. When new ideas are agreed upon, you create a “branch”, do the work/add/commit (locally) – until you are happy. You may push those changes up to the remote repository's “branch” of the same name – issue a pull request; have your peers work on the code with changes etc. At some point, the results of the branch are “merged” back into the “main branch” and a “release” is made. This “release” is “tagged” with a revision number, and life moves on.

¹The current repository is called “master”, the new move is to call the main branch, well “main”.

Branches may be branched even deeper. This may allow two competing ideas to duke it out – survivor is merged upward.

Rinse and Repeat.

0.1 Bits of Arcana

0.2 Issues

We maintain “private” repositories:

1. Respect the privilege of everyone’s privacy
2. Yours is respected to the extent possible (hackers are good)
3. We will all agree to make the repository public, or not.

Here we try to distill the capabilities of a program to meet a complex issue of managing changes for projects in a way as to be 100% safe in retaining code, documentation, and data.

1 Get Started

It is critical to have an agreed-upon way to put information into a version control system. This respects other people's ability to find things.

To join the SAS Spectrography GitHub project:

1. Go to <https://github.com/>
2. Sign-up (please fill in the damned profile and provide a recognizable icon).
3. Send login name to project leader

The project leader will add you to the list of people that use the repository.

2 GIT Version Control

The “git” package was written in over several weeks, yes weeks, because Linux Torvalds was disappointed with existing “version control” systems. It went through a brief period of improvement by the team that manages the roughly 1e6 lines of the Linux Kernel. It was released, and in minor ways, improved over the years.

Git forms the foundation for almost all serious code development. It is not limited to code.

The basic idea, is each file is copied to a “main” repository “somewhere”. The files are saved with a name that “is” the SHA1 hash code for their contents – offering a one in 1e48-ish chance of two different files having the same name. A little side-table tracks the file's real name.

In a repository, such as the one on GitHub, has copies of each revision of the file – going back.

You can make a complete copy of that repository onto your machine. That repository is just as capable of being the main repo – main by virtue of being agreed-upon by social convention.

A truly independent mirroring of important details that makes it almost impossible² for a “bad actor” to create Chaos™.

²Barring stupidity that is in abundant supply.

A Examples

A.1 Use the Browser to Add New Files

Go to the GitHub repository and log in.

To create a new directory, the fast way, is to create a new file `MyNewDirName/README.md` – to start a README file (good idea anyway) and put in some descriptive information about this thing.

Then simply browse down into the new directory, drag and drop the new files “up there”.

B Under the Hood

Action Items: