Please email bmarohn@nmt.edu if you have any questions.

Table of Contents

Subversion Introduction

Version control is based around the idea that every change is important. As files evolve from version to version, this control system makes it possible to return any file to any previous version. For example, if you accidently save over a file that is under version control, or break a piece of code moments before a demonstration, it is easy to revert it back to any previous saved version.

For this project, version control has the added benefit of allowing multiple people to work on the same set of files at the same time. Every time someone makes a change, it is easy to obtain the latest copy, and to know which version is really the latest one.

To use the subversion (SVN), a program such as TortoiseSVN must be installed on your computer. Subversion works by first “checking out” the repository to a working copy, making changes, and “committing” those changes. Other users can then “update” their working copies to reflect those changes.

Checking out files is exactly the same as copying the latest files to a folder on your hard drive. To check out a copy of the repository on Windows or Mac, right click in an explorer / finder window and select SVN→Checkout. The checkout directory is where you want the files to be placed, and the checkout URL is:
https://subversion.assembla.com/svn/nmt-2011-satellite-instrumentation/
This will prompt you for your Assembla username and password. Every file in the latest version is then available, and can be opened, edited, or run just like normal. You can make changes, delete files, move files, add files, or anything else you could do to a normal file.

Before starting to work on and change a working copy, it is important to import any changes other people have made. Otherwise, you may be working on an old version of a file. To do this, right click inside the working copy and select SVN→Update, which will automatically update all files in the current folder and subfolders to the latest version.

When you add a file, you have to tell subversion that the file is important and should be placed under version control, which can be done via SVN→Add. Similarly, when you want to delete a file, you have to right click on it and choose SVN→Delete. Even after a file is deleted, it can always be restored later (as long as it was stored in the repository at some point).

When you are done making changes to your working copy, you need to commit those changes so that others can see them. This can be done by right clicking inside a version controlled folder and selecting SVN→Commit. This brings up a text prompt, where you need to enter some description of what changes have been made. For example, the summary message I used to commit this document was “Added README documentation to the root folder”. This allows other people to know what has changed when they update to the new version. The window also lists the changes that will be saved after the commit is complete; this is a good way to make sure that all of the changes you have made will be saved.

Folder Structure

Subversion recommends a particular folder layout that helps make it easier to keep track of files. Although this is not enforced in any way, it will make it simpler to know which files are stable. Essentially, any files considered to be “stable” or “working” should be checked into the trunk. For this project, this is nearly every file. For those files that you may want to separate from the aggregated production files, like a personal to do list, you may want to create your own folder under “branches”. Here is an example:

```
  [Base Folder]
  |            |
trunk       branches
                 |
             bmarohn
```

In my “branch”, I have some pseudo code that I am working on, but it is not yet a useful part of the project. It is still helpful to me to have it under version control, but it doesn’t need to be in the trunk.

SVN Setup

The URL to get to the project space is here shown below. In order to obtain access, you should ask either Dr. Jorgensen or Byron for an invitation to the workspace.

http://www.assembla.com/spaces/nmt-2011-satellite-instrumentation/

Invitations expire after 1 or 2 days, so it is not possible to send out a bunch of invitations ahead of time. It would be best to ask for an invitation and set up an account before you need access. By default, Assembla will email you when a change is made, but the emails can be turned off if they become annoying.

You can download individual files from the web interface using the “Source/SVN” tab. To upload files or download multiple files simultaneously, you need a subversion client.

Subversion clients exist for Windows, Linux, and Mac.

Windows Setup

Download and install **TortoiseSVN**, a free SVN client. Default settings are fine.

http://tortoisesvn.net/downloads.html

Tortoise works as a shell extension, which means you don’t run it like a normal program. It adds additional options in your right click menu which lets you access it directly from where you are working.

Mac Setup

I have no experience using a Mac, but I believe the program **scplugin** does essentially the same thing as TortoiseSVN.

http://scplugin.tigris.org/servlets/ProjectProcess?pageID=4adHJG

Linux Setup

Under Linux, I am not familiar with any graphical tools. You can, however, use the command line, which you are probably already familiar with.

To install subversion, run one of the following commands (Depending on your distribution):

```
yum install subversion
apt-get install subversion
```

Checkout

```
svn checkout https://subversion.assembla.com/svn/nmt-2011-satellite-instrumentation/ foldername
```

Commit (from a working copy)

```
svn commit
```

Update

```
svn update
```