Subversion Version Control

Using Subversion

The use of a version control server, such as Subversion (SVN), allows multiple authors (e.g., you and your adviser) to share documents without having to worry about conflicting changes as well as having an archive of revisions and backups. This latter feature is especially useful for keeping track of your code!

There are several ways to interface with the Subversion server (https://or.nps.edu/svn/thchung), which is maintained by a system administrator (not me).

- Using command line (Mac, Linux)
- Using svnX (Mac)
- Using Tortoise SVN (Windows)

Using command line (Mac, Linux)

(Will outline this in future notes versions)

- Terminal help – Linux (Ubuntu) intro

Using svnX (Mac)

Download the latest version (svnX.1.3.3 as of this writing) from here

- Note the outdated older site

Install the application (e.g., copy "svnX.app" into your "Applications" directory)
Start the program

- The default windows showing "Working Copies" and "Repositories" should appear

- The repository represents the "server-side" where the main database and archive exists
- The working copy represents the "local copy" on your local machine(s) that you are currently modifying

Request repository information from TC, who will confirm creating of your account and appropriate credentials

- You will be given a URL of the form http://or.nps.edu/svn/thchung/Academic/ThesisAdvising/<student_name>/

Create and enter your repository information into the "Repositories" panel

- Press the little "+" sign to create a new repository
- You can choose any nickname for the "Name" field (shown as "My Repository")
- The "Path" field should be your repository address (in place of "blank_student")
- Enter your credentials as provided by TC
  - Note that your credentials are NOT your ERN account credentials
Double-clicking on your newly created repository (in the “Repositories” window) will open a new window.

This view (center panel) shows the log messages pertaining to one or more files in this repository. This view (lower panel) also shows the directory structure and files present in the repository.

Click the green "Checkout" button in the upper right to check out the repository (i.e., create a working copy) to your local machine.

- This will create a directory structure (e.g., "code," "proposal," "support," and "thesis" directories) in the directory you choose.
- Pick a directory that makes sense for you, as this will be the directory where you’ll be working with most (if not all) of your files, including proposals, briefs, code, and thesis document.
- You can check out a repository into multiple working copies, i.e., at different directory locations on your local machine.
A new working copy entry will be created in the "Working Copies" window

- Enter your credentials once again

Double-clicking your working copy entry will open up a new window

- The main panel reflects the change status (e.g., modified, added, missing) of any of the files in your working copy
  - For example, if we modify some elements of one of the files (e.g., thesis.tex), it should appear (annotated with "M" for "modified") in the refreshed working copy panel
As part of the standard workflow for using Subversion, you should always perform an "Update" prior to a "Commit"

* Click the great "Update" button (to ensure you have the latest version, e.g., in case collaborator(s) made any modifications)

Once you're sure you have updated your working copy to the latest version, you can go ahead and commit any/all changes

- Select one or more entries in your changed status panel that you wish to commit to the repository. This could be all of your changes within a directory or individual files (e.g., thesis.tex).
- Click the "Commit" button in the panel immediately above the status listing
- Enter a descriptive message associated with your change(s) to be logged in Subversion. Be sure to do this step (as it is incredibly helpful later on!)

* The changed status panel should clear those selected files or directories
If you create a new file or directory in your working copy, you have to let Subversion know to add these files/folders to the repository.

- For example, upon creating a new directory called "figs/" to contain all image files, you will see it appear in the changed status panel.

- To add it to the repository, click the "Add" button in the panel immediately above the changed status panel.
  - The dialog window will ask you to confirm the add, and for directories, enable you to recursively add all files and sub-directories contained within that directory.

- The "figs/" directory will reflect its changed status from "?" (unknown) to "A" (for added).

- You will still need to commit these changes (as done in the instructions above) to push these changes to the repository.

**Using Tortoise SVN (Windows)**

One of the most common clients for working with a Subversion on Windows is using TortoiseSVN, which is a MS shell extension (see this good overview) to interface with the SVN server and repository.

Here’s the link to the online manual, but this daily use guide might be more helpful.

Download the latest version (TortoiseSVN 1.7.6 as of this writing) from here.
Install the application (e.g., run the *.msi installer)

Upon completion, you should be able to see new TortoiseSVN-specific options appear under the right-mouse click menu.

A restart of Windows should be performed, just to ensure all the features are enabled in the context menus (in particular, the file annotations indicating status)

Request repository information from TC, who will confirm creating of your account and appropriate credentials

- You will be given a URL of the form http://or.nps.edu/svn/thchung/Academic/ThesisAdvising/<student_name>/

Create a directory where all your thesis files will be located

- For example, the path \my\pathname is used throughout this tutorial (but should likely be more information for your purposes)
Check out the repository by right-clicking from within your desired destination folder and selecting “SVN Checkout...”

- Enter in the repository information for the URL of the repository
- Enter your credentials to access the repository. It is recommended that you select the “Save authentication” button so you aren’t repeatedly asked to enter this information

This will create a number of directories (called your working copy) in your specified path, including \code, \support, \proposal, and \thesis folders
As part of the standard workflow for using Subversion, you should always perform an "Update" prior to a "Commit".

- This step ensures you have the latest version, e.g., in case one or more collaborators had made any modifications to any files since your last update.
Once you're sure you have updated your working copy to the latest version, you can go ahead and commit any/all changes.

You can either select one or more individual files or folders you want to commit their changes OR you can perform the commit on the parent folder which will commit all changes within its path.

- Sometimes you might not yet want to commit some of your changes (e.g., broken code) so you can commit specific files using the former method.

Enter a descriptive message associated with your change(s) to be logged in Subversion. **Be sure to do this step (as it is incredibly helpful later on)!**

- The file and/or folder annotations should change automatically to indicate a successful commit of additions or changes.
If you create a new file or directory in your working copy, you have to let Subversion know to add these files/folders to the repository by `svn add`

- For example, upon creating a new directory called "src/" to contain all your LaTeX source files, you can tell it isn't yet part of the repository by its icon.
- Using the right mouse context menu, you can add this folder to the repository.

- The "src/" directory will reflect its changed status by showing a refreshed icon.
- You will still need to commit these changes (as done in the instructions above) to push these changes to the repository.