

CSCI 379: Software Design
Bucknell University
Computer Science Department
<http://www.eg.bucknell.edu/~cs379>

Using Subversion with Xcode

Table of Contents

- 1. Introduction**
- 2. One Time Setup**
- 3. Setup for Each Project**
- 4. Working With Subversion**
 - 4.1. Determining File Status**
 - 4.2. Adding Files**
 - 4.3. Committing Changes**

1. Introduction

Version control is an important part of any software project. As of this writing, Xcode has built-in support for Subversion, CVS and Perforce. We will use Subversion.

2. One Time Setup

The first thing you need to do is tell Xcode where the Subversion repository lives. This section will guide you through that process.

1. Tell Xcode about your Subversion repository and where it can be found.
 - In Xcode, select **SCM** → **Configure SCM Repositories...** Click on the plus sign to add a new repository. In the sheet that appears, give the repository a name (“Software Design Repository”, “My Repository”, etc.) and select Subversion from the popup menu.
 - In the URL field, enter the string `https://svn.eg.bucknell.edu/csci379/f09/yourname`, where `yourname` is your user ID. You will get a dialog asking whether you want to accept an SSL Certificate. Click on the button **Accept Certificate**. In the User and Password fields, enter your user name and password. If you have done everything correctly, you should see an indicator that reads **Authenticated** with a green light next to it.
 - Press **Apply** and **OK**.
2. Adjust Xcode Build settings. We would like to avoid having the build product (the compiled binary) be under control of the versioning system. It tends to be large and

there's no reason to keep each binary version. You can always recreate it if necessary. In this step, you will tell Xcode to put the builds in separate directory.

- Select **Xcode** → **Preferences...** and click on the **Building** icon at the top of the preferences window. At the top of the building preference pane you will see an option that lets you choose the directory in which Xcode will place build products. Click on the “customized location” radio button and select a directory that all of your projects can share. It should not be within any other project directory. (Use the **New Folder** button to create a new directory if necessary.) Click **Apply** and **OK**.

3. Setup for Each Project

If you are starting a new project create it now. You can also perform these steps on an existing project.

1. Tell Xcode that this project will be using source code management.
 - Select **SCM** → **Configure SCM For This Project...** At the bottom of the window, select your SCM Repository in the popup menu. Close the window.
2. Perform an initial import of the project into the repository.
 - Select **SCM** → **Repositories** and in the leftmost column select the repository that will hold your project. Click in the **Import** button and select your project directory. Provide a message describing the project and click the **Import** button. This is the initial import.

3. Check out the project from the repository. This step doesn't seem like it should be necessary, but in my experience, nothing worked without it. If someone comes up with a way around this, please let me know.
 - Close the project window and delete the directory that contains the project.
 - Check out the project from the repository by highlighting the project in the repository and clicking on the Checkout button in the Repositories window. Click the **Open** button to open the project.

4. Working With Subversion

4.1. Determining File Status

If you click on SCM in the Groups & Files list, Subversion will show you which have been modified (M), which have been added (A) and not yet under version control, and files it doesn't know about (?).

Try making a change to a file (add a space to the end of a line), save, and observe that the file has its status changed to modified.

4.2. Adding Files

If you add new files to your project and look in the SCM group, they will appear with a question mark next to them indicating that SCM doesn't know about them. Highlight the file names, control click on SCM and select **Add to Repository**. You can also select **SCM** → **Add to Repository** to accomplish the same thing. The status of each file that you add will change to A (added).

4.3. Committing Changes

Commit your changes to the repository when you reach a significant milestone such as getting something to work. Select **SCM** → **Commit Entire Project...** to commit it to the repository. If you have done everything properly, nothing should appear in the SCM group.

Note: You may see a `pbxuser` file in the SCM group when you think that you haven't changed anything. Xcode stores preferences in this file and you don't need to do much to have it appear as changed. Don't be concerned about it.

It's a good idea to check the status of your files before closing Xcode to make sure that everything is in the repository.