Using SVN

SVN Repository Location for Marama

https://subversion.sfac.auckland.ac.nz/svn/prj_marama/

SVN Server Setup (for Administrators)

SVN Folder Structure

• /archives
  - Intended as a backup location for older resources located in Marama CVS, which is now outdated.

• /code
  - Contains source codes. Each folder represents a DS Tools project, and contains subfolders for every Eclipse project. Each project contains trunk, branches, and tags folders. The trunk contains the latest merged version the developers are working on. The branches folder contains number of subfolders where each contains a copy of the trunk where an individual developer is working on to fix a particular problem or to implement a new feature. The tags folder contains snapshot of the trunk or branches where the project is in a state ready to be released.

• /documentation
  - Contains documentations for various DS Tools projects.

• /management
  - Intended for contractual documents and overall long term plans for Marama project.

Introduction to SVN

Subversion (SVN - official site) is an open source version control system. In addition to the command line client, there is a Windows GUI based client called Tortoise SVN (official site), which features a Windows Explorer embedded interface.
Introduction to Subclipse

Subclipse (official site) is an Eclipse plugin for SVN. This is used only within Java based projects under Eclipse environment. The University of Auckland Windows systems have Tortoise SVN installed by default, so you would only need to download Subclipse if you are working on an Eclipse based project. Subclipse interface is very similar to Eclipse CVS interface.

How-to Guides

How to create a copy of SVN locally

Create a folder on your local drive.

Right click on the folder to bring up the context menu. Click on SVN Checkout...

Enter the SVN repository location and click OK (this example uses Marama project SVN repository).

Structuring your code repository

The minimal layout recommended for a project repository looks similar to this with branches and tags added:
Each project contains trunk, branches, and tags folders. The trunk contains the latest merged version the developers are working on. The branches folder contains number of subfolders where each contains a copy of the trunk where an individual developer is working on to fix a particular problem or to implement a new feature. The tags folder contains snapshot of the trunk or branches where the project is in a state ready to be released.

On larger projects, branches may contain more branches at levels below. The paths listed under tags are snapshots of the trunk on the same repository level, and they are commonly named after the release dates or version related to snapshot. The paths under tags are generally for read-only access once created.

**Branching**

Everyone who gets to modify a certain set of codes, is encouraged to create their own working branch and use simple yet informative description of changes when committing/merging their work. Branches are usually created when a developer starts working on a new fix or an update.

To create a branch, all you need to do is right click on the trunk, and click TortoiseSVN -> Branch/tag.

Then enter appropriate branch name and click OK and it is done.
Tagging

The tags directory contains snapshots of the trunk at particular milestones, usually major releases or perhaps prior to a major departure in development. Tags are created the same way as creating a branch. SVN will recognise tags and will warn users from making changes into tags. Tags should be used as read only release points, and should not need to be updated once created.

Merging

Merging a branch into the main trunk.

- First, make sure you have committed all your changes.
- Next switch back to the main trunk.
- Then right click the folder and select TortoiseSVN->Merge.
- In the from url, enter the url of the branch you were working on.
- Click merge.
- The changes you have made will only be on your local machine, you will need to commit them to the trunk.

Moving a branch

Creating a new SVN location from Subclipse

Loading a project into Eclipse from SVN

Best Practices

- No binaries for Eclipse projects
  - When committing an Eclipse workspace with source codes, it is recommended to exclude the /bin folder. The sources are recompiled live within Eclipse environment, therefore any binary information will be redundant.
• No spaces in folder/file names
  ○ Some systems does not handle spaces in the folder or file names. It is recommended to use Pascal casing (two or more words with initial capitals, run together, for example PascalCasing) for multiple worded names.
• Branch/tag naming conventions
  ○ For branches and tags, a word or two representing the branch or the tag followed by a date in YYYYMMDD format should be used. There needs to be a little more thoughts and discussions put into this for branch and tag keywords.

The following pages describe SVN best practices. Summaries of the pages will be added here soon.

http://ariejan.net/2006/11/24/svn-how-to-structure-your-repository/