Enterprise Application Development

Tutorial

Introduction to Subversion (SVN)

SLIIT-2015

Background Details

1. Discuss the role of a code repository in software development projects

   The term codebase, or code base, is used in software development to mean the whole collection of source code used to build a particular application or component. Typically, the codebase includes only human-written source code files, not source code files generated by other tools or binary library files. However, it generally does include configuration and property files. The codebase for a project is typically stored in a source control repository. A source code repository is a place where large amounts of source code are kept, either publicly or privately. They are often used by multi-developer projects to handle various versions and handle conflicts arising from developers submitting conflicting modifications in an organized fashion. Subversion, Git and Mercurial are popular tools used to handle this workflow, and are common in open source projects.

   [Source: WIKIPEDIA: http://en.wikipedia.org/wiki/Codebase]

2. Name few open-source/Proprietary code repositories and discuss the differences of those code repositories

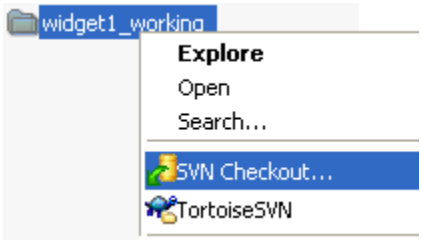
Software	Maintainer	Repository Model	Concurrency Control	License	Platform Supported	Cost
CVS	The CVS Team	Client-Server	Merge	GPL	Unix-like, Windows, Mac OS X	Free
Subversion (SVN)	Apache Software Foundation	Client-Server	Merge or Lock	Apache/BSD style	Unix-like, Windows, Mac OS X	Free
Star Team	Borland	Client-Server	Merge or Lock	Proprietary	Windows and Cross-platform via Java based client	$7500 per concurrent, $2500 per fixed user
Team Foundation Server	Microsoft	Client-Server	Merge or Lock	Proprietary	Windows Server 2003 or later, Windows 7 or 8	Free for up to 5 users, else licensed through MSDN subscription or direct buy
Visual Source Safe	Microsoft	Shared Folder	Merge or Lock	Proprietary	Windows	~$500 per license

[Source: WIKIPEDIA: http://en.wikipedia.org/wiki/Comparison_of_revision_control_software]

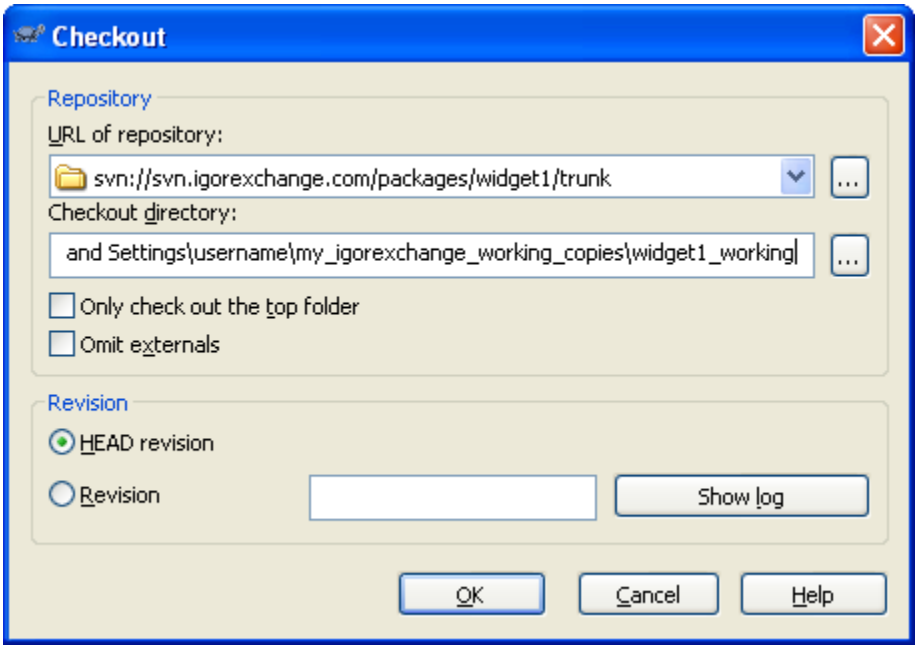
Checking out a working copy

Whenever you want to modify code for a project, you first need to create a working copy of the project files on your personal computer, and then you can modify the files in your working copy and later send those changes to the repository on the server.

1. Create a directory your computer to hold the working copy you are about to check out.
2. Right click on the above directory and choose SVN Checkout.



3. Since right now you just want the contents of the /trunk directory in your working copy, change the "URL of repository" setting to **svn://<REPOSITORY_URL>/trunk** and click the OK button.

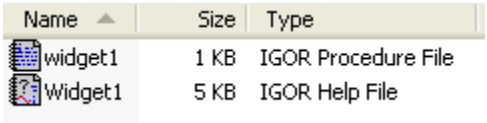


4. You should now have a green check mark icon next to your working directory. At this time, there will be nothing in this directory except for a hidden directory called .svn. You should never delete or modify the .svn directory or any files within the .svn directory, as they are used internally by TortoiseSVN.

Adding files to the repository

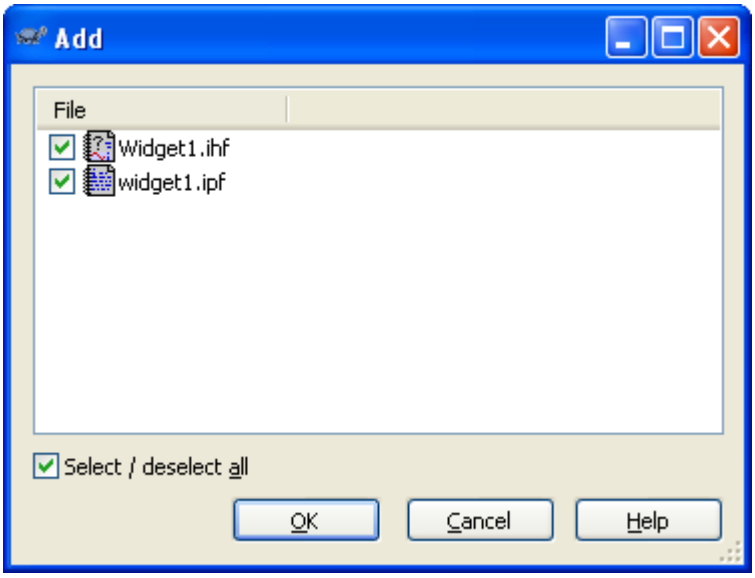
Now that you have checked out a copy of the /trunk directory from the repository into your working copy directory, it is time to add some files to the repository.

1. Using Windows Explorer, create the files necessary for your project into your working copy directory. The example working copy directory now looks like this:

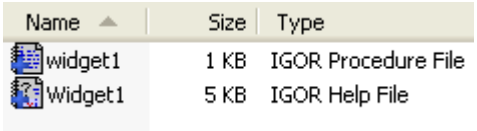


2. Right click on the name of the working directory and choose TortoiseSVN --> Add. In the dialog, check the box for each file you wish to add to the repository.

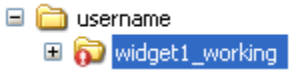
   Note: Certain types of files, like temporary files and non-essential files, should not be added to the repository. Just make sure not to check the box next to these types of files so that you don't add them to the repository. You can also tell TortoiseSVN to ignore certain files. See Ignoring files in the TortoiseSVN manual for information on how to do this. The dialog box should now look something like this:



3. Click the OK button. The icons representing the files you have just added are now be marked with a blue '+' sign:



And the working directory now is marked by a red exclamation point icon, like this:



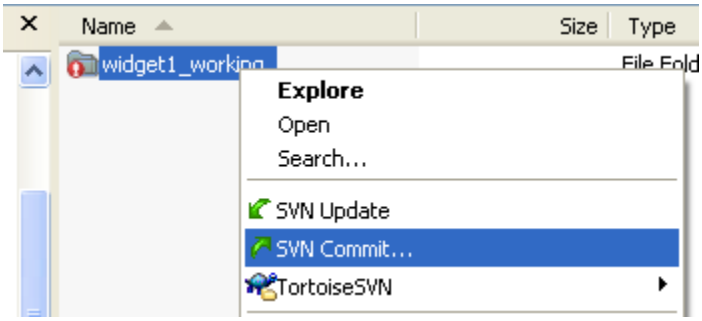
The red exclamation point indicates that the directory contains files that have been changed since the last time the working copy was updated.

4. At this point, you have told TortoiseSVN that you wish to add these files to the repository, but you have not yet actually added them to the repository. To do that, you have to commit the changes back to the repository. See the next topic for an explanation of how to do this.

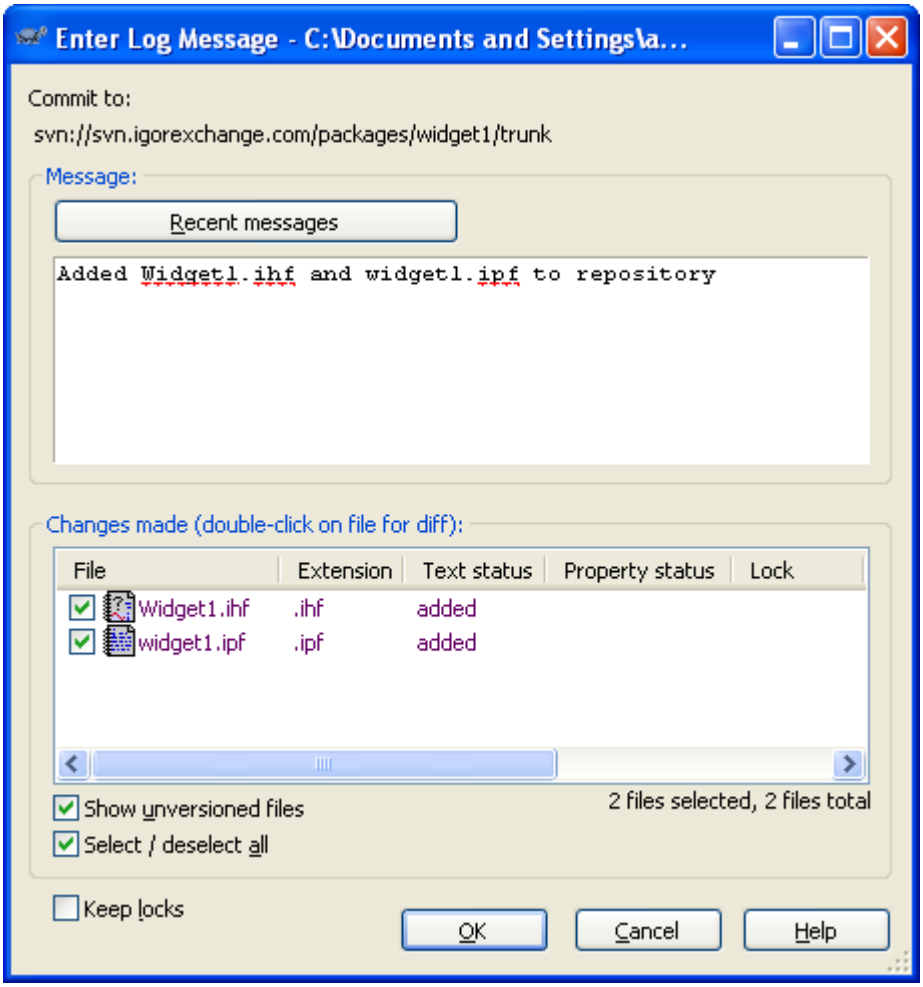
Committing changes to the repository

After you have added or removed files or made changes to your code (and tested those changes to make sure they work), you should commit those changes back to the repository so that others can access the changes.

1. From Windows Explorer, right click on the working directory and choose SVN Commit.



2. In the dialog, type in a commit message (log messages are always required). Make sure that the "Changes made" list has a check next to all of the files you want to add.



3. Click the OK button. If the commit worked, you will now see a dialog that says "Finished" and that indicates that the files were added to the repository.

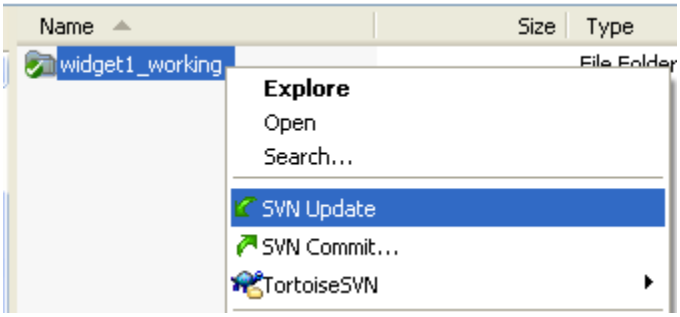
Notes:

- If you want to commit changes from only one file in your working copy (instead of the entire directory), you can right click on the file itself and choose SVN Commit.
- Instead of right clicking on a file and choosing TortoiseSVN --> Add, you can also add a file to the repository by checking the box next to the file in the "Changes made" section of any commit dialog.

Updating your working copy with changes from others

If your project has additional project maintainers other than yourself, these people are permitted to commit changes they have made to code for your project back to the repository. In order to be certain that the code you edit contains all the recent changes made by other project maintainers, you should update your working copy regularly.

1. Right click on the working directory of your project and choose SVN Update.



2. If, after updating your working copy, you receive any error messages or warning messages in the dialog, there may be a conflict between a file in your working copy and the version of the file in the repository (often by someone else). This can happen if you have edited a file since you last updated your working copy and during that time a change to that same file was committed back to the repository. If this happens, you will need to resolve the conflict in the two files. See Resolving conflicts in the TortoiseSVN online manual for more information on conflicts and how to resolve them.