CHAPTER 1 Windows Client for SVN (user manual)

Chapter overview

1.1 Introduction

Subversion (SVN) is an open source system for revision control (SVN is also the name of its command line interface). It's designed to keep track of all work and all changes in a set of files, typically the implementation of a software project, and allows several (potentially widely separated) developers to collaborate.
Subversion is an alternative (a better one) to the older Concurrent Versions System (CVS).

1.2 How SVN works

SVN utilizes a client-server architecture: a server stores the current version(s) of the project and its history, and clients connect to the server in order to check-out a complete copy of the project, work on this copy and then later check-in their changes. Typically, client and server connect over a LAN or over the Internet, but client and server may both run on the same machine if SVN has the task of keeping track of the version history of a project with only local developers.
The server software normally runs on Unix , while SVN clients may run on any major operating-system platform.

1.3 SVN - Server Side

The Server Side of the SVN it's usually the "IT guy"'s business. It implies putting a repository on track so that the authorized users can work with it and setting up accounts. In order to create a new user account on the SVN server the new user must provide the SVN administrator with an encrypted password in **MD5** format. The password can be generated using the **htpasswd** tool from apache, which is usually installed together with the **httpd** service from linux distribution CD.

Example to create a new user password: **htpasswd** -sm <output_file> <username>

The <output_file> should be encrypted via PGP and sent to the "IT guy" to register it on the SVN source server. After the user:password are registered, then the new user can access the repository.

Example Repository Path: https://86.34.201.199:1234/svn/TOT/projectx (for current project)
Example Repository Path: https://86.34.201.199:1234/svn/TOT/pending (for new pending docs)
Example Repository Path: https://86.34.201.199:1234/svn/TOT/projectx/docs/manuals/production

1.4 SVN - Client Side

The Client Side of the SVN is the SVN from a user's point of view. It actually implies working with the repository set up on the previous section.

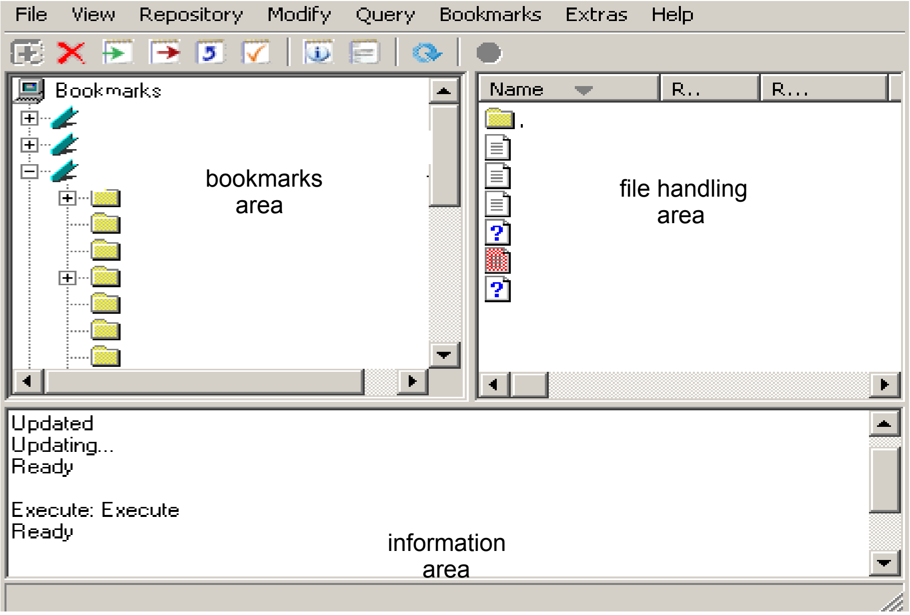
1.4.1 RapidSVN windows client

The first thing you need to do is to download the RapidSVN client for windows. You can use this link

```
http://rapidsvn.tigris.org/
```

to get a copy of RapidSVN for windows. This is a GUI front-end for SVN (as discussed earlier SVN has a command line interface which is not user-friendly at all. RapidSVN comes into play to help a user to concentrate on the changes he's supposed to do, and not on how to do it). Just install the kit and run the installed program. Focus your attention on the menu bar (you will be using that alot):

FIGURE 1.1 RapidSVN



Updated
Updating...
Ready

Execute: Execute
Ready

The main window of the RapidSVN can be divided into 3 areas:

1. The bookmarks area presents the structure of the repository using a tree view control. From this window you can navigate through the file structure to get where you need to make changes.

2. The file handling area reveals the content of the directories clicked in the bookmarks area (try to click or double click on one directory in the bookmarks area and see what happends in the file handling area). if the directory you clicked is a leaf in the tree structure on the left, then on the roght you will see the files it contains. Now you can operate on those files as you will see in the next section.

3. The information area provides you with information about the commands that RapidSVN performs. Here you will see when an Update is finished (the updated files and their paths are also displayed), the status of your Commit or if a file was added or not to the repository.

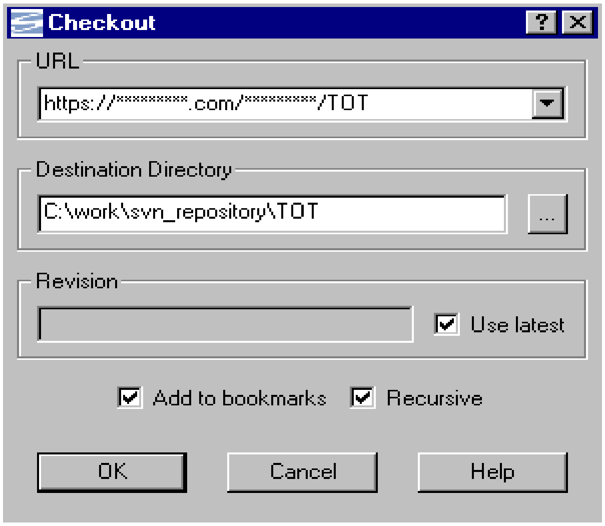
1.4.1.1 Getting your hands on the repository

To start making changes on the project you are working on you must first get a copy of the repository on your local machine. This is done in 2 steps:

1. Create a directory on your local hard drive where you will store the repository (e.g. C:\work\svn_repository).

2. Get the copy of the repository from the server. For this you need to click Repository on the menu and then choose Checkout. This dialog box will appear:

FIGURE 1.2 Checkout dialog box



URL: https://**********.com/**********/TOT

Destination Directory: C:\work\svn_repository\TOT

Revision:

☑ Use latest

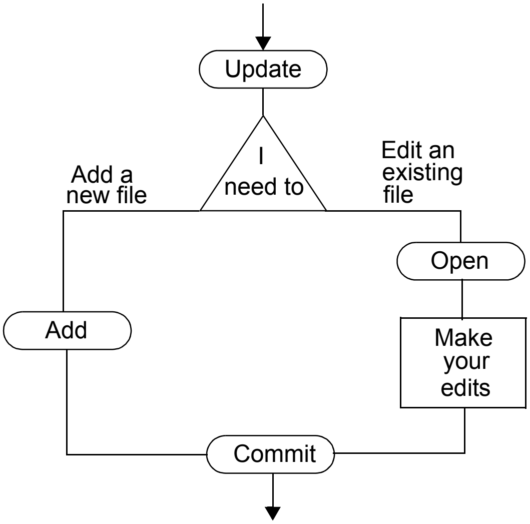
☑ Add to bookmarks ☑ Recursive

First make sure that all checkboxes are checked. Then look at the Destination Directory edit box. Choose the local path to the directory where the repository will be stored (on my drive I've created an additional directory TOT - Top Of the Tree. It's best that this directory's name be the same with the directory name on the server - the admin will have this name provided to you).
Look at the URL edit box. You will notice that I've used a secure protocol to access the repository. This is mandatory only when you wannt to access the repository from a computer outside the network at your firm (this is ofcourse for information protection). Just type the address that the admin gave's you, press OK and authenticate when asked. Depending on how big and how fragmented (how many small files) is the repository, the checkout process may take longer than anyone can expect. But is important to know that **you perform checkout only once (or every time the admin tells you)**. By performing checkout a copy of the repository located on the server will be placed on your local machine at the local path you gave.

1.4.1.2 The work flow

Now that you have the repository on your computer you can start working on it. Let's examine the next figure in which are synthesized the most important tasks one can perform with the RapidSVN

FIGURE 1.3 The work flow

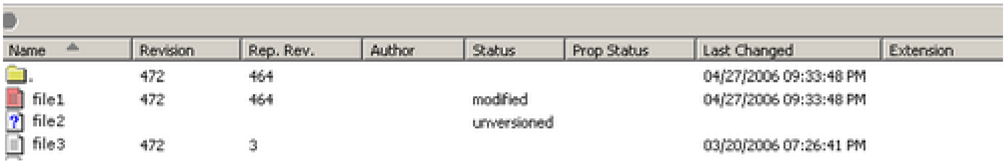


It’s good practice to Update before making any changes. You can Update the whole repository (not recommended, it will take long) or you can update just the directory you intend to work in. To update a directory, from the Bookmarks window navigate to the directory, right click on it and choose Update. This will update the version of the directory you already have on your local drive (if someone made changes in that directory they will be updated). This command is the reason why you don’t have to Checkout the repository every time you wannt to keep up to date.

The flow chart above try to guess what one may wannt to do: add a new file or editing an existing one.

To add a file to the repository, the file must first be placed on your local drive somewhere inside TOT directory so that RapidSVN can see it. This is how the program sees your file:

FIGURE 1.4 File states in RapidSVN



Name	Revision	Rep. Rev.	Author	Status	Prop Status	Last Changed	Extension
.	472	464				04/27/2006 09:33:48 PM	
file1	472	464		modified		04/27/2006 09:33:48 PM	
file2				unversioned			
file3	472	3				03/20/2006 07:26:41 PM	

First study the layout of the window to become familiar with it. It displays useful information about the files on the reposittory. The figure above shows 3 possible file states. Notice that file1 has a red icon on its left. That means that file1 was edited and saved by you and RapidSVN warns you to Commit the file on the repository (check out its status name : modified). File2 has a question mark on its left meaning it exists only on your HDD and not on the repository. This is an unversioned file. File3 is a file in a “normal” state : it exists on both repository and your HDD and waits to be modified.

There are 2 more file states which are not shown in the figure : added file - a small icon with an A on it, meaning the file was added to the repository and must therefore be Commited and deleted file - a n icon with a small x on it, meaning that the file was deleted from the repository and therefore must be Commited.

OK, let’s get back to Figure 1.3 on page 5. I was about to show how one can add a file to the repository. Now you know that the file must first exist somewhere in the TOT direcotry on your HDD. If you search for it with the RapidSVN you will find it as an unversioned file. Right click on it and choose Add. After this the RapidSVN will update the state of the file to an added file (the icon with an A on it). Accordingly to Figure 1.3 on page 5 the next operation you must perform is the Commit. The Commit operation upload all your changes from your local drive to the server (right click on the file you want commited and choose Commit : a new window will appear. Write in a nutshell all the changes made).

If you just want to edit an existing file, search it with the RapidSVN (you will find it as a normal file), right clik on it and choose Open (remeber that it’s better to Update first). This will automatically open the file with the appropriate editor. After you make some changes and save, Rapid SVN will update the state of the file to an edited file (red icon). Choose View -> Refresh View if the file isn’t red yet. Now you can Commit the file with your changes to the repository.

1.4.1.3 Other features

Update/Open/Edit/Commit/Delete are the basic operations you can perform with RapidSVN. You can also find detailed information about a file by right clicking on it and choosing Log or Info.

Within RapidSVN you can move a file from a location to another : right click on the file, choose Move, paste in the new path (on your local drive ofcourse) and press OK. After that you must Commit the file.

You can make a directory on a location : right clock, Make Directory, paste in a name, press OK and then Commit the new directory.

1.4.2 Turtoise windows client

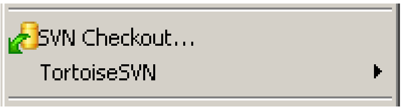
You can find this SVN windows client at :

```
http://tortoisesvn.tigris.org/download.html
```

After downloading install the program and reboot your computer.

The main difference between RapidSVN and Turtoise is the GUI (Graphic User Interface). If RapidSVN has a GUI of its own, Turtoise is integrated in Windows Explorer. So if you want to “start” Turtoise, the usual (Start -> Programs -> TurtoiseSVN ->...) won’t do you no good. Run Explorer first : Start -> Run -> type “explorer” and press OK. Now if you right click on a random directory you will notice this portion in the window that will appear :

FIGURE 1.5 Turtoise options



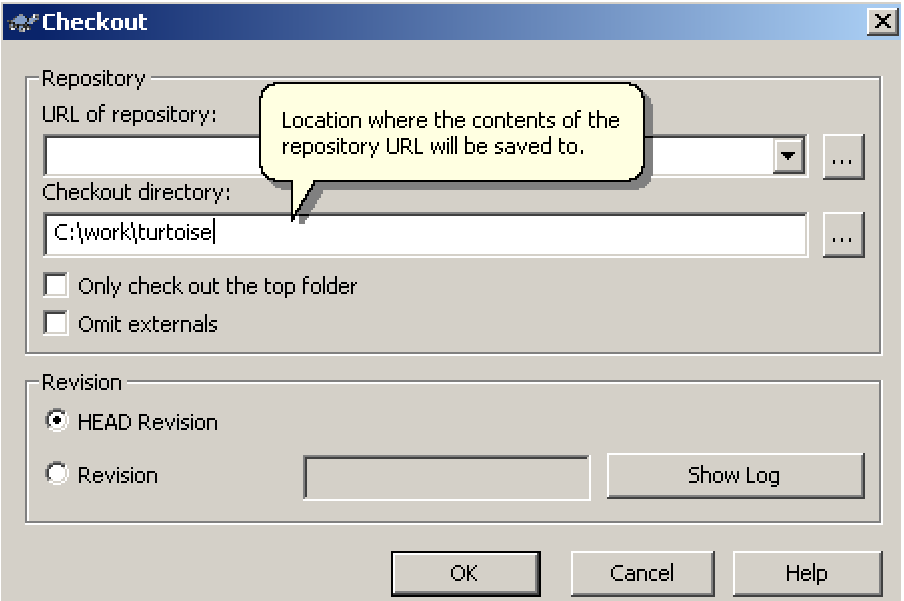
1.4.2.1 Getting your hands on the repository

Ofcourse if you want to make changes to the repository you must first have a copy to your local machine. Follow these steps:

1. Create a directory on your local hard drive where you will store the repository (e.g. C:\work\tutoise).

2. Get the copy of the repository from the server. For this you need to right click on the directory created at step1 and choose SVN Checkout (see Figure 1.5 on page 6). This dialog box will appear:

FIGURE 1.6 Turtoise Checkout



Location where the contents of the repository URL will be saved to.

Checkout directory: C:\work\turtoise

□ Only check out the top folder

□ Omit externals

● HEAD Revision

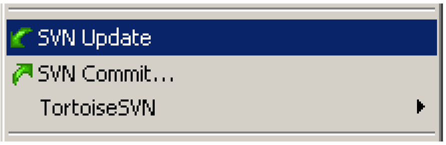
○ Revision

The URL of repository is the address that the admin will give you. Press OK, authenticate when asked (check Save authentication so you don't authenticate every time you perform an action) and wait untill the repository is entirely copied on your local hard drive.
The observations in 1.4.1.1 Getting your hands on the repository are also valid here.

1.4.2.2 The work flow

The work flow is the same with Turtoise. Only the way actions are performed is different (because of the GUI). Look again at Figure 1.3 on page 5.
It's good practice to Update before making any changes. You can Update the whole repository (not recommended, it will take long) or you can update just the directory you intend to work in. To update a directory navigate with Explorer untill you reach it and right click on it. You must be able to notice these options :

FIGURE 1.7 How to Update/Commit



Choose Update and only that directory will be updated.
The next figure presents the file states in Turtoise:

FIGURE 1.8 Turtoise file states

unversioned file must be added then commited

added file must be commited

file on the repository ready for editing

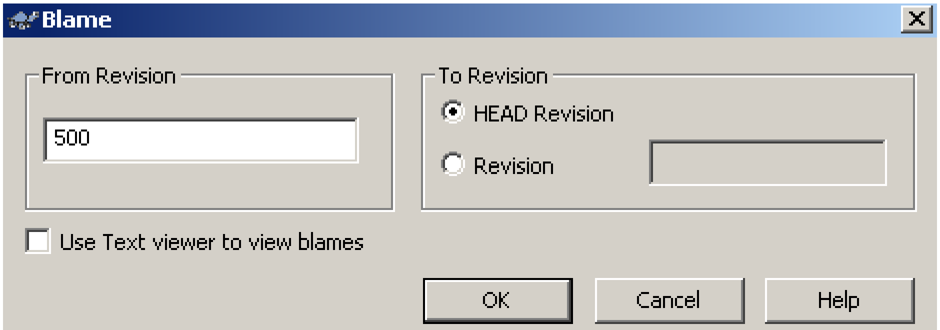
modified file must be commited

To add a file to the repository right click on it, choose Turtoise SVN and then Add. A blue + sign will appear on it. This means that the file must be Commited, so right click again and choose Commit. You will be prompted for a Log message. Every time you commit you must write a small log to let other people know about your changes. Press OK. You will be then informed about the status of the performed task.
If you want to delete a file right click on it, choose TurtoiseSVN and Delete. The file will disappear suddenly but that’s not it. You must also Commit this delete. Actually Turtoise will let you know that you’re not finish by changing the state of the directory (a red icon with an ! sign will appear on all the directories in the path of the deleted file). Right click on the directory where the file was deleted from and choose Commit. Write a log and press OK. After a refresh the ! sign should disappear.
If you want to edit an existing file first Update the directory of that file then double clik on the file. The appropriate editor will open the file. Make your edits, save often and close the file when you finish. Turtoise will change the state of the file : the ! sign will appear on the file and on all directories in the path of the file (press F5 to refresh). This way Turtoise reminds you to Commit the file. Better listen to it: right click on the file and choose Commit, enter a log when prompted and press OK. Simple, huh?

1.4.2.3 Other features

Turtoise has a nice diff tool wich can help you see the changes you’ve made to a file (this works only with text files). For example you change some lines of code in a file and then save it. If you are not sure of what you did, before Commit the file, right click on it and choose TurtoiseSVN -> Diff. A special viewer will open the file in a way that allows you to see your changes.
You can see also the changes that other people made to a text file. After you Update, right click on the file and choose TurtoiseSVN -> Blame. A dialog window will appear:

FIGURE 1.9 Blame



Blame

From Revision: 500

To Revision
● HEAD Revision
○ Revision

□ Use Text viewer to view blames

You have the option of choosing the starting and ending revision number(when someone Commits a change to the repository the revision number increases by 1. The revision number is not related to the file, but to the whole repository). If you want to see all the modifications made to a file, you normally enter 1 for From Revision and leave HEAD Revision (which is the last revision) checked. It's not recommended to do this because it will take very long untill all the changes are displayed. This feature is useful to see who and what did to a text file.
You can see some stats by doing this too : right click on a file and choose TurtoiseSVN -> Show Log. From the window that appears you can see the logs eneterd by those who modified the file (practically you can read the description of what was modified in the file). You can also press Statistics (bottom right corner) and see who was lasy and who was hard working.