

Client Installation

General

All users should obtain version 1.4 or later of Subversion and the corresponding clients. The primary developers (cf developers of ports or other forked versions) is at <http://tigris.org/>.

Windows

The favoured subversion tool for Subversion users on Windows seems to be TortoiseSVN, which is pretty good in our experience. You can download it from [here](#)². Usually it will arrive on your desktop (e.g. if you use FireFox). You will probably have to re-start your machine after installation :-)

Once installed, TortoiseSVN will be visible in the Windows Explorer (not the Internet Explorer!) on the right-click menu, and also on the Properties right-click menu choice on a file or directory - this is how you see properties in subversion.

Before doing anything, you need to verify and possibly modify the TortoiseSVN config file, which will have been created in the Windows directory

C:\Documents and Settings\`<your_user_name>`\Application Data\Subversion, or you can get to it by right-clicking on any directory in the Windows Explorer and choosing TortoiseSVN>Settings->Edit config file. See [below](#)³ for continuing the setup process.

Unix

All unices

Subversion 1.2 is available as a standard part of most linux releases. For example, in Debian (which we use on the openEHR.org server), the apt-get command and X-windows synaptic application can be used to obtain Subversion 1.2 for command-line use.

Linux

For GUI use, there are links to Debian, Red Hat and other Linux downloads [here](#)⁴. However, we strongly suggest you look for these downloads in existing application management tools such as apt-get etc, before doing uncontrolled downloads.

FreeBSD

Ports exist of all the unix-based Subversion tools - see the [FreeBSD](#)⁵ 'devel' ports page.

2. <http://tortoisesvn.sourceforge.net/downloads>

3. `#allusers`

4. <http://rapidsvn.tigris.org/>

5. <http://www.freebsd.org/ports/devel.html>

Mac OS X

Binaries for Mac OS X can be found [here](#)⁶.

Subclipse for Eclipse

Subclipse is an Eclipse plug-in that incorporates Subversion capabilities into Eclipse, with a similar approach to using CVS within Eclipse. Note that Subclipse at the moment only supports the Windows environment.

Firstly ensure that Eclipse is installed on your system, versions 3.0 or 3.1 or later are ideal.

Then within Eclipse itself:

- go to the install new plug-in features section, accessed by Help, Software Updates, Find and Install...
- select "Search for new features to install" and a list of available sites to update from should appear.
- select "New Remote Site" and enter an appropriate name to identify the site (such as "Subclipse") and for the URL enter: <http://subclipse.tigris.org/update>
- once added, make sure it is selected to search and click "Finish"

Eclipse will now access the site and should from that, follow the remaining instructions that appear to install the necessary plug-ins.

Using Subclipse, as mentioned before, is markedly similar to using the CVS capabilities that exist in Eclipse. Accessing a repository is a case of opening the appropriate Perspective (in this case "SVN Repository Exploring") and then checking out the repository contents into a new Project or an existing one.

Most other functions will bear resemblance to the CVS, or TortoiseSVN.

[thanks to David Nguyen, CS dept, UCL for this section]

All Users

Configuration File

Various settings need to be set to make subversion work properly - mainly to do with what is called "auto-properties". Properties need to be set up in advance so that when you create new files, they have the correct MIME types, ignore status and so on. An example config file is [here](#)⁸. Here are the modifications you should make:

- replace the global-ignores setting in the [miscellany] section with something like the following (make sure the line is uncommented):
 - `global-ignores = *.o *.lo *.la ##*.rej *.rej .*~ *~ .*#* .DS_Store EIFGEN *.epr *.bpr *.wb *.backup.* *.log *.lck`
- uncomment the use-commit-times setting in [miscenllany] section, i.e. make sure it is "yes"
- make sure the line enable auto-props = yes is uncommented

6. <http://metissian.com/projects/macosex/subversion/>

8. <http://www.openehr.org/downloads/developer/config>

- replace the [auto-props] section with something like the following:

```
*.pl = svn:mime-type=text/plain;svn:eol-style=native
*.c = svn:mime-type=text/plain;svn:eol-style=native;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.cpp = svn:mime-type=text/plain;svn:eol-style=native;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.cs = svn:mime-type=text/plain;svn:eol-style=native;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.vb = svn:mime-type=text/plain;svn:eol-style=native;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.h = svn:mime-type=text/plain;svn:eol-style=native;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.e = svn:mime-type=text/plain;svn:eol-style=native;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.ace = svn:mime-type=text/plain;svn:eol-style=native;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.java =
svn:mime-type=text/plain;svn:eol-style=native;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.py = svn:mime-type=text/plain;svn:eol-style=native;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.html = svn:mime-type=text/html;svn:keywords="LastChangedDate LastChangedRevision
URL"
*.htm = svn:mime-type=text/html;svn:keywords="LastChangedDate LastChangedRevision
URL"
*.dadl = svn:mime-type=text/plain;svn:keywords="LastChangedDate LastChangedRevision
URL"
*.sh =
svn:mime-type=text/plain;svn:eol-style=native;svn:executable;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.y = svn:mime-type=text/plain;svn:eol-style=native;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.l = svn:mime-type=text/plain;svn:eol-style=native;svn:keywords="LastChangedDate
LastChangedRevision URL"
*.pdf = svn:mime-type=application/pdf
*.fm = svn:mime-type=application/x-maker
*.book = svn:mime-type=application/x-maker
*.mif = svn:mime-type=application/x-mif
*.dsp = svn:eol-style=CRLF
*.dsw = svn:eol-style=CRLF
*.txt = svn:mime-type=text/plain;svn:eol-style=native
*.png = svn:mime-type=image/png
*.jpg = svn:mime-type=image/jpeg
*.gif = svn:mime-type=image/gif
*.rtf = svn:mime-type=application/rtf
*.adl = svn:mime-type=text/plain;svn:keywords="LastChangedDate LastChangedRevision
URL"
*.adls = svn:mime-type=text/plain;svn:keywords="LastChangedDate LastChangedRevision
URL"
Makefile = svn:eol-style=native
Makefile.* = svn:eol-style=native
```

You need to make sure these settings correspond to the kinds of files and directories you create, especially programming language files. Note the `svn:keywords` settings - these properties set on a file are what actually causes keyword substitution to occur.

If you discover that you need to set or change certain properties on numerous files, it will be easier to do it via the command line - see [expert users](#)⁹ below.

Create a Project Area

Now create an 'openehr' project area, if one does not already exist on your file-system. In this documentation, this area is always referred to using \$OPENEHR.

Expert Users

9. #expertusers

Fixing SVN properties

Sometimes your files will end up with incorrect properties set on them. The most common symptom of this is HTML, text and other files that would normally be served by a web server may be served in raw form (unreadable) because their MIME type property is not set. This happens in SVN because of checking in files before the correct MIME types are defined in the config file, as above. The command to change existing files is as follows:

```
svn propset svn:mime-type text/html *.htm
```

Other properties could be set using this same kind of command with appropriate arguments. This can be run in a normal Windows command shell window or unix shell session. Commands like this can be run over a whole directory tree using the method below.

Bulk file changes

If you want to make changes in ways not supported by the client end GUI on your platform, you will need to use subversion from the command line. For Windows users, we recommend you obtain the Windows port of the standard subversion 1.2 release via cygwin - along with other unix-like tools to make your Windows command-line experience sane! The Cygwin installer is available [here](#)¹⁰. We recommend installing direct from the internet, and with the default settings the installer offers you. The following examples are the same on Windows/cygwin/subversion and subversion on unix.

Here are a couple of typical command lines you might use to set properties on whole directories of files:

- set svn keywords properties on every .htm file from here down (ignoring .svn directories):

```
$ find . \( -name '.svn' -prune \) -o -name '*.htm' -exec svn propset svn:keywords  
" LastChangedDate URL " {} \;
```

- set property svn:ignore on all directories, so that .adl2 files are ignored:

```
$ find . -type d -a \( ! -wholename '*\.svn*' \) -exec svn propset svn:ignore  
'*.adl2' {} \;
```

Note that these are examples only; note also that you do only need to set properties once on anything, and commit the changes - properties settings are stored in the subversion server, so any new checkout will cause them to be automatically set in the new work-area.

10. <http://cygwin.com/>