

The cover features a diagonal gradient bar transitioning from light blue at the top to yellow at the bottom. The title 'TRAC2SVN' is centered in a bold, blue, sans-serif font with a slight 3D effect.

# TRAC2SVN

## Admin Guide

*The content of this publication is the property of Clearvision-CM  
Reproduction of this content is strictly prohibited © 2007 - 2009*

Copyright © Clearvision CM 2007 - 2009

# Contents

<b><u>1 Prerequisites</u></b> .....	<b>1/18</b>
<u>1.1 Linux prerequisites</u> .....	1/18
<u>1.2 Important Information for Windows Vista Users</u> .....	1/18
<u>1.3 Configuration planning</u> .....	1/18
<u>1.3.1 Fully Flexible Configurations</u> .....	2/18
<u>1.3.2 Alternative Configurations</u> .....	2/18
<u>1.4 License</u> .....	2/18
<b><u>2 Setup Trac</u></b> .....	<b>4/18</b>
<u>2.1 Install XML-RPC Plugin</u> .....	4/18
<u>2.1.1 Easy Install</u> .....	4/18
<u>2.1.2 Manual Install</u> .....	4/18
<u>2.2 Adjust Permissions for XML-RPC Plugin</u> .....	4/18
<u>2.3 Test XML-RPC Plugin</u> .....	5/18
<b><u>3 Trac2SVN Installation</u></b> .....	<b>6/18</b>
<u>3.1 Trac2SVN Server</u> .....	6/18
<u>3.1.1 Configuring the Server to run on startup</u> .....	7/18
<u>3.2 Trac2SVN Client</u> .....	7/18
<u>3.2.1 Configuring User Ids for each Trac2SVN Client</u> .....	7/18
<u>3.2.2 Configuring the Client to run automatically</u> .....	8/18
<u>3.3 Setting permissions on Linux</u> .....	9/18
<u>3.4 Subversion Hook Setup</u> .....	9/18
<u>3.5 Portable Hooks</u> .....	10/18
<b><u>4 Customization</u></b> .....	<b>11/18</b>
<u>4.1 Configuration Options</u> .....	11/18
<u>4.1.1 Trac2SVN Server Configuration Options</u> .....	11/18
<u>4.1.2 Hook Configuration Options</u> .....	14/18
<u>4.1.3 Client Configuration Options</u> .....	15/18
<b><u>5 Troubleshooting</u></b> .....	<b>16/18</b>
<u>5.1 Installation Support</u> .....	16/18
<b><u>6 Appendix A: Portable Hooks Examples</u></b> .....	<b>17/18</b>
<u>6.1 Sun Solaris 10</u> .....	17/18
<u>6.1.1 Set path to pick up GNU tools</u> .....	17/18
<u>6.1.2 Python</u> .....	17/18
<u>6.1.3 Twisted 8.2.0</u> .....	17/18
<u>6.1.4 PySVN 1.6.2</u> .....	17/18

# 1 Prerequisites

- Trac2SVN is independent of the Subversion client used.
- Trac2SVN is built, tested and certified against specific versions of Subversion.
- Trac2SVN uses the Trac XML-RPC API to communicate with Trac, which needs to be installed as a plugin - please see below for details.

## 1.1 Linux prerequisites

The **client** executable uses GTK+ 2.0 or later. If GTK+ is not available on your system, it can be installed on Centos or Redhat Enterprise Linux 5 as follows:-

```
yum install gtk+
```

On systems without GTK+, the Trac2SVN client will exit with an error message reporting a missing `libgmodule.so` library.

## 1.2 Important Information for Windows Vista Users

On Windows Vista, there are two steps which you need to carry out before Trac2SVN functions correctly.

1. If using the installer, make sure you run it as an Administrator (Right click on the installer and select 'Run as Administrator')
2. After installing Trac2SVN, you will need to run both Server and Client as an Administrator, which can be done using either of the following two options:-
  - ◆ Either right-click on the executable and select 'Run as Administrator' (you will have to repeat this every time you load the executable)
  - ◆ Or right-click on the executable and select 'Properties'. Click on the 'Compatibility' tab and at the bottom, tick 'Run this program as an administrator' (this will mean every time the executable is run, it will be run as an Administrator)

## 1.3 Configuration planning

You will need to choose a machine and available port for the Trac2SVN Server process. Note that the client machines will require access to this port on the server, which may require firewall configuration.

As per the IANA recommendations, we recommend that you use one of the Dynamic and/or Private Ports from 49152 through 65535. The default port number for Trac2SVN is 54327.

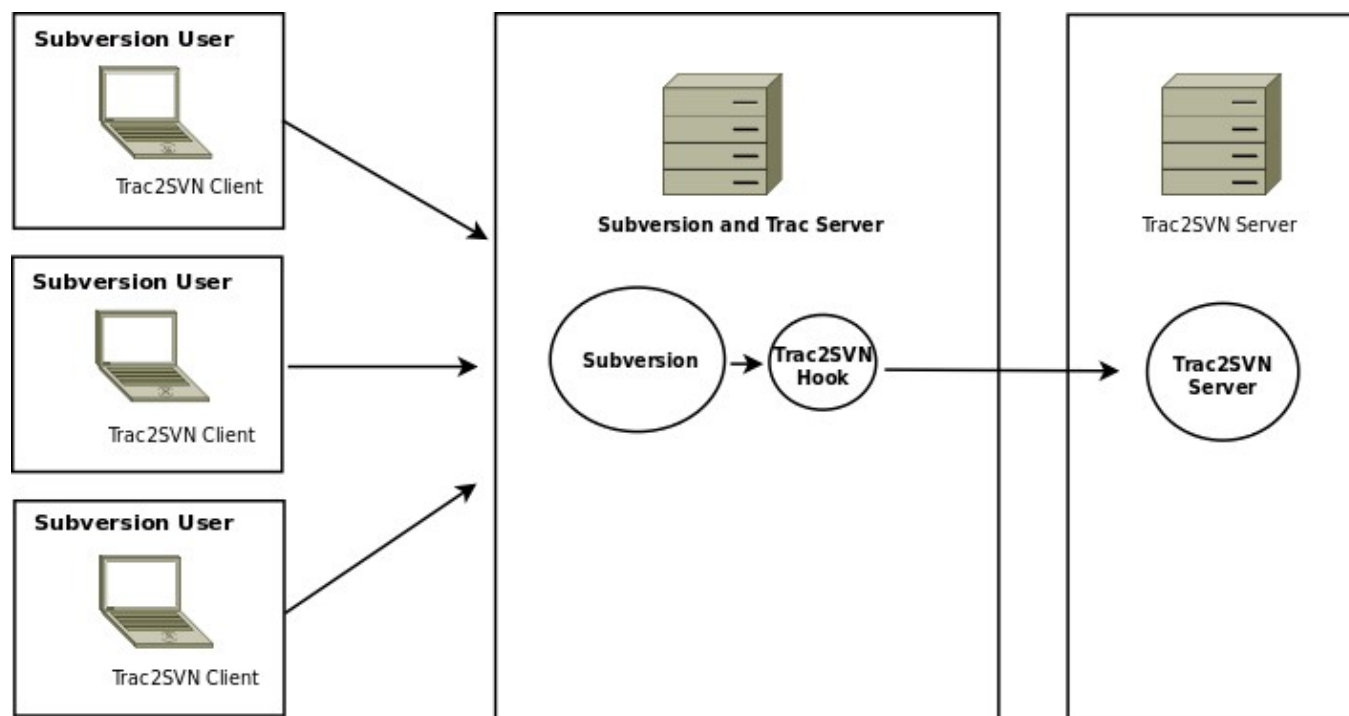
Trac2SVN is designed with maximum flexibility in mind to cater for various network infrastructure requirements. Therefore, the physical location of all of the components is flexible. In particular, the location of the Trac2SVN Server is flexible to help manage network traffic and data latencies. It may be installed on your Subversion server machine, your Trac server machine or on its own machine as shown below.

However, when choosing a suitable configuration, please bear in mind that the current Trac Subversion plugin, which allows you to browse the Subversion repository from Trac, requires the Subversion repository to live on the same host as the Trac server (this is Trac itself, not the Trac2SVN server). If you follow this convention, the browsable links created by Trac2SVN in Trac tickets will allow you to navigate directly to Subversion changesets associated with Trac tickets.

### 1.3.1 Fully Flexible Configurations

Each part of the system can live on its host within the network. The communication between the different parts happens through network channels, which allows you to freely configure the system as you desire.

However, in order to get the most out of the integration between Trac and Subversion, you are strongly advised to keep Subversion repositories and the Trac server on the same host. Note that this is a limitation of the Trac Subversion plugin, and not a limitation of Trac2SVN. The next release of Trac is expected to address this limitation.



The location of the Trac2SVN server is fully flexible and the server can live anywhere within the network. However, you must install Trac2SVN hooks in your Subversion repository for the integration to function.

### 1.3.2 Alternative Configurations

#### 1.3.2.1 Multiple Subversion Repositories with Multiple Trac Servers

If you regularly switch between different Subversion repositories, which link to different Trac2SVN servers, you can run multiple clients on your machine, i.e. one for each Trac2SVN server. Whenever you commit a change, the Subversion hook of the repository will contact the appropriate Trac2SVN server, which will then present you with a pop-up window using the client connected to that server.

In order to run multiple clients, prepare a configuration file for each client and run the client using the `-f` option to point to the correct configuration file.

## 1.4 License

Trac2SVN requires a licence. Trac2SVN licences are:-

- Time limited
- Tied to a particular machine with a particular IP address
- Tied to a particular port on that machine

- Restricted to a maximum number of users (i.e. client connections)

To request a license you can use the `get_license_info` executable and run it on the machine you want to use as the **Trac2SVN server**. The `get_license_info` tool can be found within the tools directory of installation. The tool will retrieve system information from the machine and present you with a license request to return to Clearvision. Alternatively, you can fill in the details below and send them to [sales@clearvision-cm.com](mailto:sales@clearvision-cm.com):-

```
License Request for Clearvision Trac2SVN:
.... Evaluation: yes/no
.... IP Address: 192.168.64.1
..... Port: 54322
... MAC Address: 00-50-56-C0-00-05
..... Users: 50
..... Company: You Company name here
· Email Address: yourco@yourco.com
... Expiry Date: 2010-03-31
..... Product: TRAC2SVN
```

The `get_license_info` executable is run from a command prompt, eg on Windows:

```
> get_license_info.exe

Please either cut and paste the form and email the request to
sales@clearvision-cm.com or visit our website and fill in the
form online at: http://www.clearvision-cm.com/products/

Note: ALL fields are required

-----
Server License Request:

Company Name=_____
Contact Email Address=_____
Required Expiry Date=_____ (eg 2008-12-31)
Server port number=_____ (eg 54321 for CQ2SUN, 54322 for Jira2SUN)
Number of Users=_____

PRODUCT= CQ2SUN or JIRA2SUN
HOSTNAME=tushar-win

Please select the main Ethernet interface from the following set
(ie the one which matches the IP address for the Server machine)
(use ipconfig/all to see more information):

:      IP=192.168.47.1      MAC=00-50-56-C0-00-08
:      IP=192.168.174.1    MAC=00-50-56-C0-00-01
:      IP=192.168.1.46     MAC=00-1A-A0-96-C2-D3
:      IP=2001             MAC=00-00-76-1F-AE-A9
```

Cut and paste the highlighted section, check all the fields, fill in the empty ones (Company Name, etc) and send to [sales@clearvision-cm.com](mailto:sales@clearvision-cm.com).

## 2 Setup Trac

### 2.1 Install XML-RPC Plugin

Trac2SVN requires the Trac XML-RPC plugin. The installation is very straightforward, especially when using `easy_install`. Please see the XML-RPC plugin web site <http://trac-hacks.org/wiki/XmlRpcPlugin> for details on the installation.

As a quick guide, you can install the XML-RPC plugin as follows:-

#### 2.1.1 Easy Install

Using Easy Install with Trac 0.11:-

```
easy_install -Z -U http://trac-hacks.org/svn/xmlrpcplugin/trunk
```

Using Easy Install with Trac 0.10:-

```
easy_install -Z -U http://trac-hacks.org/svn/xmlrpcplugin/0.10
```

#### 2.1.2 Manual Install

- Download source, unpack, change into source directory, then:-

```
python setup.py bdist_egg
cp dist/*.egg <path_to_trac>/env/plugins
```

- Then edit `trac.ini` and add:-

```
[components]
tracrpc.* = enabled
```

### 2.2 Adjust Permissions for XML-RPC Plugin

Once the XML-RPC plugin has been installed, you need to give all authenticated users access to the XML-RPC API. This can be done as follows:-

- Log in to Trac as an administrator
- Select the Admin tab
- In the Grant Permission box on the right, type `authenticated` in the Subject field and select `XML_RPC` from the Action pull-down list.
- Press Add

**Grant Permission:**

Subject:

Action:

Grant permission for an action to a subject, which can be either a user or a group.

If you prefer, you can limit access to the XML-RPC API on a per-user basis. However, as a minimum requirement every Trac2SVN user must have access to the XML-RPC API.

## 2.3 Test XML-RPC Plugin

If you wish, you can test that the XML-RPC plugin is operational with the following Python code:-

```
import xmlrpclib
username="user"
password="password"
server_address="trac.company.com/myproject"
proxy = xmlrpclib.ServerProxy("http://%s:%s@s" % (username, password,
server_address))
print proxy.system.getAPIVersion()
```

The result of this should be a list with three elements indicating the Trac API version, e.g. [1, 0, 6].


---

## 3 Trac2SVN Installation

Trac2SVN has three separate parts: a Client, a Server and a set of Subversion hooks, which must be installed in your Subversion repository's `hooks` directory. There is also a tool to get license information from your Trac2SVN Server in the `tools` directory.

Start by unpacking the zip file into the `C:\Program Files` location on all of the machines that you will be installing on (any location can be used, however all the configuration files and instructions below use `C:\Program Files` as an example). The following directory structure will be created:-

- `client` directory contains the executable which must be started on every client.
- `server` directory contains the Trac2SVN server executable
- `subversion_hooks` contains the set of files which reside in the `hooks` directory of every subversion repository.
- `tools` contains the license request tool
- `docs` contains PDF documentation for Trac2SVN (including this guide)

 **NOTE** Please take care when editing any configuration files. We recommend the use of text editors such as Vim. Wordpad should also work, but avoid using Notepad or MS Word.

### 3.1 Trac2SVN Server

On your Trac2SVN Server machine:

- Unpack the zip file into `C:\Program Files`.
- Edit `Clearvision\trac2svn\server\trac2svn_server.cfg` using a programmers editor (or Wordpad *not* Notepad on Windows) and configure the following parameters with values according to your setup. For the `trac_server` please configure the full path to your project on the Trac server. Both `http` and `https` protocols are supported for this setting. If no protocol is specified, Trac2SVN assumes `http` as the protocol. The `trac_query` setting selects the tickets that will be shown to a user committing a change. The keyword string `CV_CURRENT_USER` is replaced by the Trac user id for the user committing the change at run time.

```
[Trac]¶
trac_query=owner=CV_CURRENT_USER&status!=closed¶
trac_server=http://trac.mycompany.com:8080/myproject¶
¶
[License]¶
company_name=Your Company Plc¶
licence_email=admin@yourcompany.com¶
expiry_date=2008-12-31¶
users=999¶
license_key=PLEASE_ADD_YOUR_LICENCE_key_HERE¶
license_certificate=PLEASE_ADD_YOUR_LICENCE_certificate_HERE¶
¶
[Server]¶
server_ip=192.168.1.2¶
server_port=54327¶
```

- open a command window, go to the server directory, then start the server (make it executable first on Linux), eg:

```
cd c:\Program Files\Clearvision\trac2svn\server¶
trac2svn_server.exe¶
```

- check the `TRAC2SVN.log` file in the server directory for any errors (an empty file indicates no errors).

### 3.1.1 Configuring the Server to run on startup

On Windows, we recommend that you add the Trac2SVN Server as a 'Scheduled Task', 'When my computer starts'. An alternative would be to add an item in the `autoexec.nt` file so that the server is started on boot and is available for all users. Alternatively, you may want to start the Server manually, e.g. whenever Subversion is started.

There is a `-f` command argument that you can use to choose a particular config file like this:-

```
trac2svn_server.exe -f c:\my_trac2svn_server.cfg
```

You may also want to refer to the `tools/run_server.sh` sample script for Unix platforms.

## 3.2 Trac2SVN Client

- Either unpack the whole zip file as above or copy just the `c:\Program Files\Clearvision\trac2svn\client` folder. (*An alternative location can be used, e.g. on Linux you may use `/opt/Clearvision/trac2svn`, however all the configuration files and instructions use `C:\Program Files` as an example*).
- edit the file `C:\Program Files\Clearvision\trac2svn\client\trac2svn.cfg` and set the information pointing to the Trac2SVN server related to the Subversion instance you like to use for this client.

```
[Server]¶
server_ip=192.168.1.2¶
server_port=54327¶
```

### 3.2.1 Configuring User Ids for each Trac2SVN Client

Trac2SVN uses two user ids: your Trac user name and your Subversion user name.

On each client machine, you must configure the Trac user id and password and optionally a subversion user id. By default, Trac2SVN uses the current login name as its id for Subversion. If you want to use an alternative id, configure this on the client machine with the `svn_uid` option in `trac2svn_client.cfg` as follows:

```
[Trac]¶
trac_uid=joe_bloggs¶
trac_password=my_secret_password¶
¶
[Options]¶
svn_uid=jbloggs¶
```

Note: The default server configuration disables the ability to use `svn_uid` on the clients. If you want to enable this functionality, please refer to the Customization section.

You can now avoid including the password in the config file if you desire (e.g. for security reasons). Simply comment out the `trac_password` configuration variable (by adding a semi-colon at the start of the line as below), and you will then be prompted to enter your password when the client starts.

```
[Trac]¶
trac_uid=joe_bloggs¶
;trac_password=¶
```

## 3.2.2 Configuring the Client to run automatically

You will probably want to have the Trac2SVN Client running whenever a user is logged on to his/her machine so that Trac2SVN is available whenever they check in to Subversion.

### 3.2.2.1 On Windows

#### 3.2.2.1.1 Logon

We recommend that you add a shortcut to the Trac2SVN Client as an item in the `StartMenu/Programs/Startup` folder for each Trac2SVN User. This setup ensures that whenever a user logs on to Windows, they automatically register with Trac2SVN.

Depending on your requirements, you may be able to use the same configuration file for all users. You can point to a shared configuration file using the `-f` option. However, if your Subversion user ids do not match your login user id, then users will require individual config files with their individual `svn_ui` settings. You can add the `-f` setting to the shortcut definition so that the correct config file is used when the shortcut is invoked.

```
trac2svn_client.exe -f c:\Program
Files\Clearvision\trac2svn\client\trac2svn_client.cfg
```

An alternative may be to automatically start the client is to run it as a 'logon script' in the Profile tab of the Computer Management/Users settings.

Another alternative is to use a Windows 'Scheduled Task' and run the client 'When I Log On'.

#### 3.2.2.1.2 Logoff

You may be able to assign a logoff script as [documented here](#) and write a simple batch script (using the DOS [taskkill](#) command) to kill the process on shutdown, for example:

```
taskkill /f /t /im trac2svn_client.exe
```

### 3.2.2.2 On Linux

As a Gnome user, you can make the Trac2SVN client a Startup Application that runs when you log in to Gnome. The mechanism for configuring this varies from distribution to distribution.

- On Centos or Fedora (using Gnome as the Session Manager) go to System->Preferences->More Preferences->Sessions, select the Startup Options tab and press Add to add Trac2SVN to your startup applications. Specify the full path to trac2svn\_client and use -f to point to a config file (see example below).
- On Ubuntu Linux (using Gnome as the Session Manager) to System->Preferences->Startup Applications and press Add to add Trac2SVN. Specify the full path to trac2svn\_client and use -f to point to a config file (see example below).

```
/opt/Clearvision/trac2svn/client/trac2svn_client -f
~/trac2svn_client.cfg
```

You may also want to refer to the `tools/run_client.sh` sample script for Unix platforms.

### 3.3 Setting permissions on Linux

On Linux, you will have to set permissions appropriately. Log files are written to the current working directory (ie wherever you run the process from), so this directory will require write-access. All of the executables, ie the client, server and all of the hook files will require execute permission. As an example, we've provided the `tools/post_install.sh` script which will need to be customized, eg:

```
# This is an example post-installation script which you
# will need to customise according to your installation.
#
# Run this before copying the hook files to your Subversion repository.
# Also please make sure that the files and directories are readable!
#
TRAC2SVN_HOME="/opt" # change this appropriately
#
chmod +x $TRAC2SVN_HOME/Clearvision/trac2svn/client/trac2svn_client
chmod +x $TRAC2SVN_HOME/Clearvision/trac2svn/server/trac2svn_server
chmod +x $TRAC2SVN_HOME/Clearvision/trac2svn/subversion_hooks/*
#
# if you run the client and server in-place, and don't set the
# log_file config variable, you will need to make the
# directories writable:
chmod +w $TRAC2SVN_HOME/Clearvision/trac2svn/client
chmod +w $TRAC2SVN_HOME/Clearvision/trac2svn/server
```

### 3.4 Subversion Hook Setup

Trac2SVN requires two Subversion hooks to be installed in your Subversion repository's hooks area. The hooks and related programs are delivered in the `C:\Program Files\Clearvision\Trac2SVN\subversion_hooks` directory. These need to be copied to your hooks directory, and will require a small amount of customization for your local setup. If you have local (non Trac2SVN) Subversion hooks to run, you may integrate these with Trac2SVN (see the `bat/shell` files for more info).

On your *Subversion* Server machine:


- unpack the zip file into `C:\Program Files`.

- copy **all** of the files from `Clearvision\trac2svn\subversion_hooks` into the hooks directory of your Subversion server.
- on Linux, make all of the hook scripts executable (i.e. everything except the config file)
- edit the config file (`trac2svn_hook.cfg`) that is *in your Subversion hooks directory* and configure the following parameters with values according to your setup, for example, if your Trac2SVN Server is at IP address 192.168.1.2 and is licensed for port 54327:-

```
[Server]¶
server_ip=192.168.1.2¶
server_port=54327¶
```

## 3.5 Portable Hooks

To use the portable hooks rather than executables, please replace `pre-commit` and `post-commit` with `portable-pre-commit` and `portable-post-commit`. Note that you **must rename** the `portable-xxx` files as otherwise they will be ignored by Subversion.

 Also note that, unlike the executables, the portable version requires a set of software tools to be installed on your system. Please refer to the Prerequisites section at the top of this document for details.

---

## 4 Customization

Trac2SVN can be customized according to your local requirements in a number of ways. This section describes all of the configuration options.

### 4.1 Configuration Options

There are a number of options to control the behaviour of Trac2SVN. The default options are configured to enable an **enforcement** policy. Using the options below, you can allow a more *permissive* policy for your system.

The options are set in the configuration files in a section called `[Options]`. There are options for the server, for the hooks and for the clients. The server configuration options control site-wide behaviour of clients, such as allowing multiple records to be selected, or allowing configuration of the subversion user id. The hook configuration options control commit behaviour for that repository such as permitting anonymous commits or allowing clients to not choose a change record for their changeset. The clients can set their user id for Subversion (if this is allowed by the server).

All of these options are detailed below.

#### 4.1.1 Trac2SVN Server Configuration Options

##### 4.1.1.1 Allow Multiple Trac Tickets per Changeset

The default configuration pops up a GUI which allows the User to select a **single** Trac ticket to associate with the Subversion changeset. This policy can be changed server-wide by setting the `allow_multiple_choices=1` option in `trac2svn_server.cfg`, for example:

```
[Options]¶
allow_multiple_choices=1¶
```

##### 4.1.1.2 Allow Subversion User Id Configuration on Clients

Normally a client will use the operating system's user id as its Subversion user id. Depending on your local policy and setup, you may need to allow clients to tell Trac2SVN that they have a different user id for Subversion.

To enable this configuration option on all clients, you can set the `allow_client_svn_uid_configuration=1` option in the *server* configuration file.

```
[Options]¶
allow_client_svn_uid_configuration=1¶
```

Then, a client can configure their `svn_uid` option (see below)

##### 4.1.1.3 Allow Issues to be Moved to the Next State

The default configuration pops up a GUI which allows the User to select a Trac ticket to associate with the Subversion changeset. Sometimes it is useful to let a user also move that ticket to its next state (for example saying that development is complete, and a fix is ready to be tested). The `allow_next_state` option

enables this functionality for all clients. Once enabled, the clients will be presented with an enhanced pop-up window that allows them to optionally progress the state of a record.

*PLEASE NOTE:* If you enable this feature, you must also specify a configuration that specifies for each state which state to move a ticket to. This can be done through the `trac_workflow_matrix` option in the Trac section.

The example below shows a workflow matrix for the default Trac 0.11 workflow. If you use customised workflows in Trac, you will need to adjust the setting accordingly. The `trac_workflow_matrix` needs to be defined such that for each state it defines which state a ticket should be moved to. The format is `{{trac_workflow_matrix=<old_state>:<new_state>[;<old_state>:<new_state>...]}`.

To enable this feature, set the `allow_next_state` option to 1 in the [Options] section of `trac2svn_server.cfg`, and define the workflow matrix in the `trac_workflow_matrix` option in the [Trac] section.

```
[Options]¶
allow_next_state=1¶
¶
[Trac]¶
; Format: trac_workflow_matrix=<old_state>:<new_state>[;<old_state>:<new_state>]¶
trac_workflow_matrix=new:assigned;assigned:accepted;accepted:closed;closed:reop
```

#### 4.1.1.4 Show Extra Field Information in Trac2SVN GUI

Sometimes the ticket id and ticket title are not enough to decide which ticket to attach a change to. If this is the case, you can show more information in the Trac2SVN GUI by adding extra fields and values to the information displayed in the GUI.


You can enable this feature using the `show_extra_fields` configuration option in the [Trac] section of the server configuration. The format is as follows:-

```
; Format: show_extra_fields=<field_id>[,<field_id>, ...]¶
show_extra_fields=status,type¶
```

#### 4.1.1.5 Enabling Trac2SVN Command Line Mode

Trac2SVN can scan for ticket ids in the Subversion log entered by the Subversion user. This is useful if you cannot or prefer not to use graphical applications such as the Trac2SVN client GUI.

You can turn this option on by adding the `allow_change_ids_in_svn_log=1` option to the [Options] section in your server config file (the default is off).

 Please note that if you enable this option any log message starting with a number will cause the number at the beginning of the log message to be interpreted as a Trac ticket id.

```
[Options]¶
allow_change_ids_in_svn_log=1¶
```

Note that ticket ids must be provided at the beginning of the log message with no preceding white spaces and followed by a space character. Multiple issue id's can be provided, with issue id's separated by semicolons. The following examples are Subversion log messages that would result in Trac tickets being updated:-

```
17 Fixed problem¶
¶
10;12 Fixed two problems¶
```

Note that if this option is enabled and you would like to specify a log message that starts with a number but do not want the number to be interpreted as a ticket id, then you can prevent Trac2SVN from scanning the log message by starting your log message with a space (or any other character):-

```
 2 problems fixed¶
¶
Fixed 2 problems.¶
```

Both examples above work fine as Trac2SVN will not consider 2 as a Trac ticket id as it does not appear at the beginning of the log message. Note that if you specify a Trac ticket id at the beginning of the log message that is not part of the set of issues returned by the Trac filter, Trac2SVN will give an error.

Note that with the `allow_change_ids_in_svn_log` option it is possible to run Trac2SVN without a client. However, since Trac2SVN needs to be able to access the Trac server, you need to specify a default Trac username and password in the server config file. The `default_trac_uid` and `default_trac_password` is only used if the user is not logged in through a client.

```
default_trac_uid=myuser¶
default_trac_password=mypassword¶
```

Please note that once these options are set, any Trac2SVN user can make use of the default user name and password. If this is not desirable, then it may be preferable to require each user to log in to the Trac2SVN server through a Trac2SVN client even when using the command line mode. You can do this by not specifying the `default_trac_*` options in the config file.

Note, however, that if a user is logged in through a client, Trac2SVN will pop-up the client GUI if the information provided in the Subversion log is not sufficient.

#### 4.1.1.6 Configure Connection Timeout

When network problems occur, clients may not be able disconnect cleanly from the server, causing the server to believe that clients are still connected and wait for a response.

With the `connection_timeout` setting in the `Options` section, you can configure the time in seconds before the server considers a connection broken.

Setting the value too low can cause clients to be disconnected on slow systems. A value that is too large can cause commits too hang until the server recognises the broken client connection.

Note that connections are checked regularly and in most cases a reasonable large timeout such as 30 or 60 seconds should be suitable.

```
[Options]¶
connection_timeout=30¶
```

## 4.1.2 Hook Configuration Options

### 4.1.2.1 Allow Commit Without Change Id

A user who is logged in to the Trac2SVN system normally must choose a record to go with their changeset. If they cancel their choice, Trac2SVN will abort the Subversion commit.

If you want to allow them to optionally **not** select any of their current change records and 'cancel' their choice in the GUI (i.e. allow commits **without any change record**, you can set this option.

**⚠ WARNING:** Enabling this will mean that **some changes are NOT recorded in your change control system!**

```
[Options]¶
allow_commit_without_change_id=0¶
```

### 4.1.2.2 Allow Anonymous Commits

Default behaviour is to check that a user has logged on to the Trac2SVN Server, and to abort any commits from users who are not yet logged on (**a policy-'enforcing' mode**). You can disable this behaviour here (for example if some of your users do not use change records).

**⚠ WARNING:** this will mean that those users not running a Client will NOT be using this system to correlate Subversion with your change control system, and they effectively be bypassing Trac2SVN altogether!

```
[Options]¶
allow_anonymous_commits=1¶
```

### 4.1.2.3 Allow Commit On Error

Normally, if an error occurs in the system, Trac2SVN will try to abort the commit. If you want to let the commit proceed (for example as a temporary measure until you fix a system problem), you can set this option:

```
[Options]¶
allow_commit_on_error=0¶
```

Note that this is *not* guaranteed to work as it depends on the exact problem that is occurring, so, please use with caution.

### 4.1.2.4 Write Ticket Ids to Subversion Log

You can configure Trac2SVN to add ticket ids to the subversion log message. If you enable this option and commit a change, Trac2SVN will add the ids of all issues chosen by the user to the front of the log message.

This makes it possible to see the association between a commit and a ticket from the log message.

```
[Options]¶
log_issue_ids_in_svn=1¶
```

## 4.1.3 Client Configuration Options

### 4.1.3.1 Changing the Subversion User Id

If the server is configured to allow it, you can configure the subversion user id that a client uses with the `svn_uid` option in the `trac2svn_client.cfg`:

```
[Options]¶
svn_uid=jbloggs¶
```

Note: The default server configuration *disables* the ability to use `svn_uid` on the clients. If you want to enable this functionality, please see the `allow_client_svn_uid_configuration` option above.

### 4.1.3.2 Prompting for a password


You can avoid including your Trac password in the config file (e.g. for security reasons). Simply comment out the `trac_password` configuration variable (by adding a semi-colon at the start of the line as below), and you will then be prompted in a window to enter your password when the client starts.


```
[Trac]¶
trac_uid=joe_bloggs¶
;trac_password=¶
```

---

## 5 Troubleshooting

To check that a Trac2SVN client or server is running on Windows, open the Windows Task Manager. You should see 2 entries for the server (on a server machine) and 2 for the client (on a client machine). To kill the process, right click on the process in the Processes tab and select 'End Process Tree' (to kill both processes at once). If the process is not running, check your log file for any errors. On Linux, use `ps -ef | grep trac2svn` - these processes can be safely killed.

 To investigate setup problems, it can be helpful to increase the logging level from the default of 'Warning' to 'INFO' (see `trac2svn.cfg`)

 Note that when an error occurs, the default behaviour is to *abort* the commit. Some possible reasons are:

1. When your User Id is not recognized as a Subversion User Id (you may want to set `svn_uid`) in your client's config file to override the default user id
2. When the user is not logged on to Trac2SVN (check in Task Manager that `trac2svn_client.exe` is running and that it is using the expected uid)
3. When there are no valid records assigned to the user (or there are no eligible records - run the query in the Trac web interface and check the results.).
4. When Subversion `svnserve.conf` file uses `anon-access = write`

If you have a local repository and are using the file protocol via the command line client, your commits will **not** use your windows logon id, they will be *anonymous*. So, to work with Trac2SVN you must use something like:

```
svn commit myfile.txt --username engineer
```

The config files use *unix* line-endings, so please be careful to use a well-behaved text editor such as [Cream/Vim](#) or Wordpad rather than Notepad or MS Word.

### 5.1 Installation Support

If you experience any issues during the installation of Trac2SVN please report your issues via the Clearvision Support Centre, which can be found at <http://www.clearvision-cm.com/support-home/support-center.html>.

---

## 6 Appendix A: Portable Hooks Examples

### 6.1 Sun Solaris 10

This appendix gives guidance on installing prerequisite tools for portable hooks on a Solaris 10 system. Please note that the instructions below are for *guidance only* and your system may require different steps.

#### 6.1.1 Set path to pick up GNU tools

```
export PATH=/usr/ccs/bin:/usr/sfw/bin:$PATH
```

#### 6.1.2 Python


Python 2.4 is available on the Solaris CD, and other versions (including python 2.5) are available from Sunfreeware.com. These use the standard Sun pkgadd. We would recommend using python 2.5 if possible.

#### 6.1.3 Twisted 8.2.0

 This package has long file names, therefore it **requires 'gtar'** to be unpacked.

```
python setup.py install
```

##### 6.1.3.1 Zope 3.3.1

Twisted requires the interfaces package in Zope.  This package has long file names, therefore it **requires 'gtar'** to be unpacked.

```
./configure --with-python=/opt/python/bin/python # use your python  
location here.¶  
make¶  
make install¶
```

#### 6.1.4 PySVN 1.6.2

Note that PySVN **requires** GNU's version of make, "gmake".

The version of gcc that we used had a problem with finding the right linker. GCC has [hard-coded search paths](#) (for as, ld, etc.). Our version was missing the path to the GNU linker, but had the Sun linker, so failed with an invalid (GNU-specific parameter, ie --rpath). To work around this problem, we moved /use/ccs temporarily (see below)

```
cd Source¶  
python setup.py configure¶  
mv /usr/ccs /usr/ccs.BAK... # see above¶  
gmake¶  
cp -r pysvn /opt/python/lib/python2.5/site-packages/... # use your python  
location here.¶
```

```
mv /usr/ccs.BAK /usr/ccs¶
```

---

```
Copyright (c) 2008 Clearvision CM All Rights Reserved¶  
Registered in England No. 5643578 VAT Registration No. UK881364801¶  
Registered Address Laurel Farm, Winters Hill, Durley Southampton, SO32  
2AH¶
```

<http://www.clearvision-cm.com>