INTLAND SOFTWARE

Intland Software GmbH
Wankel- Str. 3
D-70563 Stuttgart,
Germany

T +49 (711) 674006778
F +49 (711) 67400-686
info@intland.com
www.intland.com

codeBeamer

SCMLoop Installation manual

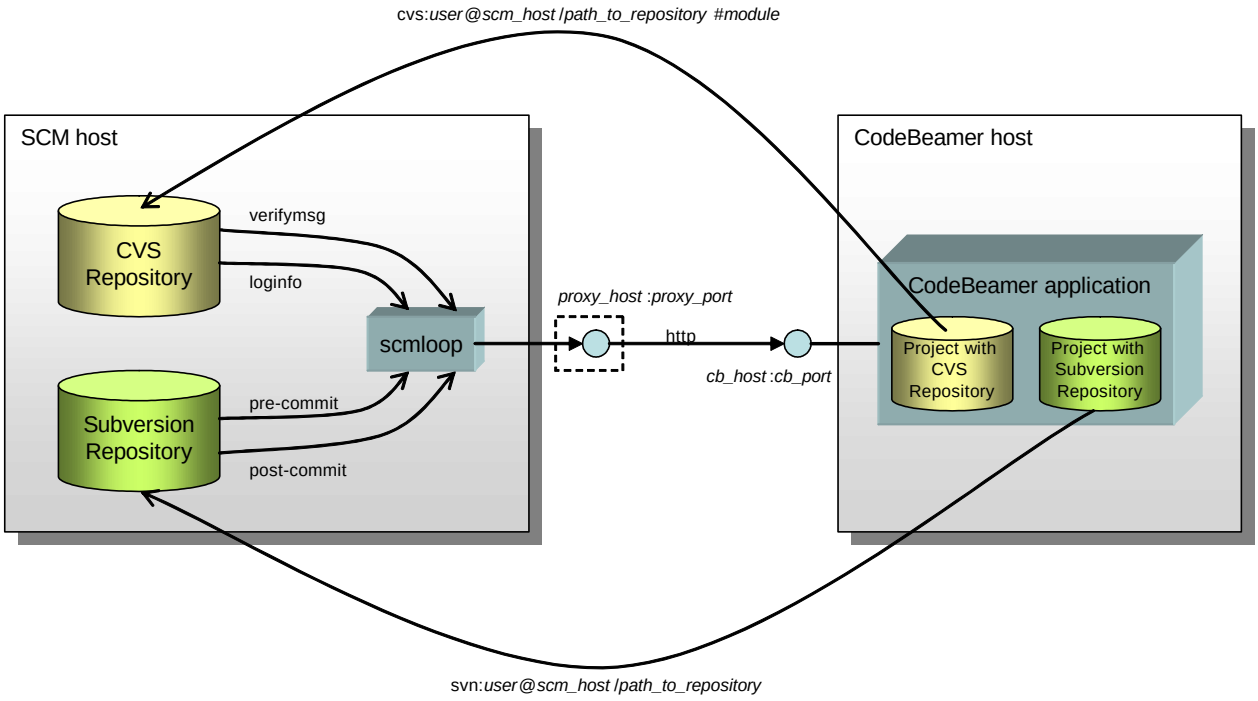
Release	4.2.4
Date	Juli 30. 2007

Content

1 Introduction

This Document describes how to install the IntLand CodeBeamer SCMLoop component.

SCMLoop is a notification mechanism, which allows to associate commits to a Source Code Management (SCM) repository with tasks in CodeBeamer.



Currently, SCMLoop is available for the following SCM systems:

- **CVS**
- **Subversion**

CodeBeamer has built in support for so called “Managed Repositories”. Managed repositories are Subversion repositories created and maintained by CodeBeamer itself.

An exclusive feature of Managed Repositories is the seamless integration of repository access control with CodeBeamer.

You only need to setup the extra SCMLoop component manually,

- if you want to connect already existing Subversion repositories to CodeBeamer,
- or the Subversion repositories reside on a different machine than CodeBeamer,
- or you are using CVS.

2 Installing SCMLoop

You need to install the SCMLoop component on every server that hosts SCM repositories you want to connect.

Unpack the SCMLoop distribution archive `scmloop.zip` to a publicly accessible directory/folder. Either on each server that hosts SCM repositories, or on a network folder that is accessible from each SCM host.

Please note, that also a Java Runtime Environment (JRE 1.4 or higher) is required.

If you don't want to install a standard JRE, you can simply copy the `jre` directory/folder from the CodeBeamer distribution to the same location where `scmloop` resides.

In order for SCMLoop to work, its configuration must be adjusted to the current environment.

On Windows systems you have to edit the script `scmloop/scmloop.bat` , on Unix systems `scmloop/scmloop` , and modify the values of the following variables:

- `SCM_DIR` is the path to the `scmloop[.bat]` script itself
- `CBHOST` is the hostname/IP- address of the CodeBeamer server
- `CBPORT` is the port number where CodeBeamer listens (default 8080)
- `JAVA` must point to the `java` executable of the JRE to use.

Make sure, that all SCM users are allowed to access and execute the `scmloop[.bat]` script.

After you have successfully installed and configured SCMLoop, you can now start to instrument your SCM repositories.

2.1 Special SCMLoop settings

Under rare conditions, you may have to adjust other special SCMLoop settings.

2.1.1 Accessing CodeBeamer via a Proxy

If `scmloop` cannot access CodeBeamer directly, but only via an intermediate proxy, you have to set additional script variables:

```
PROXY_OPTIONS= -Dhttp.proxyHost= <hostname> -Dhttp.proxyPort= <portno>
```

If the proxy server also requires authentication, you have to provide the proxy authentication information via the options `–proxyuser` and `–proxypassword` in the `SCM_OPTIONS` variable:

```
SCM_OPTIONS= … -proxyuser <username> -proxypassword <password> …
```

2.1.2 Allowing the SCM to commit changes even if CodeBeamer is not running

After a SCM repository has been instrumented for `scmloop` (see 3), it will not longer be possible to commit changes to the repository while CodeBeamer is not running or reachable.

This mode enables CodeBeamer to control, whether commits without a valid tracker reference are allowed or not.

If for some reason, this constraint is not adequate for your installation, you can tell `scmloop` via the "`-catch`" option in the `SCM_OPTIONS` variable, to allow commits without CodeBeamer confirmation:

```
SCM_OPTIONS= … -catch …
```

Commits executed while CodeBeamer is unreachable will not get lost, because CodeBeamer regularly synchronizes with all SCM repositories and will pick up these commits during this process.

2.1.3 Host name problems when SCM system and CodeBeamer run on different hosts

Some common installation problem is with host names. By default, the scmloop script sends the official TCP/IP host name as the SCM system host name. When in CodeBeamer an alias host name is configured for the SCM system, CodeBeamer can't find the appropriate project for the commit because CodeBeamer compares host names and not IP addresses. The "`-host`" option can be used in `scmloop` script to send a host name instead of the official TCP/IP host name, that should match the SCM system host name configured in CodeBeamer.

```
SCM_OPTIONS= … -host <hostname> …
```

Please consider the following example: your CVS or Subversion server has the TCP/IP name "**pluto**" and it has also an alias "**cvs-host**" and this name is configured as SCM host in a CodeBeamer project.
In this case as default, the scmloop script sends "**pluto**" as the host name of the SCM server and CodeBeamer won't find the project because this name doesn't match "**cvs-host**".
But when you add "`-host cvs-host`" to the scmloop `SCM_OPTIONS`, CodeBeamer will find the appropriate project.

3 Instrumenting SCM repositories

Instrumenting a SCM repository means to add appropriate SCMLoop callbacks/hooks to every SCM repository you want to connect to Codebeamer.

The SCMLoop distribution contains pre-instrumented examples for all files named in this chapter.
When copying the example files, please take care to adjust the path to the `scmloop` directory accordingly.

3.1 Instrumenting a CVS repository

Setting up a CVS repository and access to it, is not part of this document. For more information about CVS see http://www.cvshome.org .

We assume that the repository already exists and is accessible.

1. Check out the CVS repository administrative files (CVSROOT) to some working directory.
2. Edit the file **verifymsg** in the **CVSROOT** folder in the working directory and add the following line:

   ALL [cmd /c][1] *<path_to_scmloop>* [2]**scmloop -mode verifymsg**

3. Edit the file **loginfo** in the **CVSROOT** folder in the working directory and add the following line:

   ALL [cmd /c][1] *<path_to_scmloop>* [2]**scmloop -mode loginfo %{sVv}**

4. On some CVS servers you might have to add "`LogHistory=all `" to **CVSROOT/config**
5. Commit your changes to the repository.

---

[1] Only on Windows systems.
[2] Replace with the absolute path to the directory, where the `scmloop[.bat]` script resides (see chapter 2).

3.2 Instrumenting a Subversion repository

We assume that the repository already exists and is accessible.

1. Check out the **hooks** directory of the Subversion repository to some working directory.
2. Edit the file **pre-commit[.bat]** in the **hooks** folder in the working directory:
    a. On Windows, add the following line to **pre-commit.bat :**

    ```
    <path_to_scmloop> scmloop -mode svn_pre_commit %*
    ```

    b. On Unix, add the following line to **pre-commit :**

    ```
    exec <path_to_scmloop> scmloop -mode svn_pre_commit "$@"
    ```

3. Edit the file **post-commit[.bat]** in the **hooks** folder in the working directory:
    a. On Windows, add the following line to **post-commit.bat :**

    ```
    <path_to_scmloop> scmloop -mode svn_post_commit %*
    ```

    b. On Unix, add the following line to **post-commit :**

    ```
    exec <path_to_scmloop> scmloop -mode svn_post_commit "$@"
    ```

4. Commit your changes to the repository.