

Introduction

CVSSource is a CVS 'source' (i.e. a new protocol) implementation for Cocoon, giving you read/write access to files stored in a CVS repository.

NOTE

This component is made available outside of the main Cocoon CVS repository because it relies on the CVS client package of [JCVS](#)¹ which is LGPL licensed and as such cannot be added to Apache's CVS repository.

Project resources

- SVN: <http://svn.cocoondev.org/repos/cvssource/>

Using the CVSSource

NOTE

The CVSSource requires a Cocoon 2.0 CVS snapshot from 10 january 2003 or later (yet untested on Cocoon 2.1).

Building the library

There is currently no binary release of this component. To build the library, checkout the CVS repository (see "Project resources" above) and launch "build.bat" or "build.sh" depending on your environment. This creates a jar file in the build directory, which you can add to the WEB-INF/lib directory of your Cocoon webapp.

Configuring cocoon.xconf

The following cocoon.xconf snippet shows how to access Cocoon's document files on the Apache public CVS repository :

```
<protocol name="cocodoco-cvs" class="com.anwrt.cocoon.cvssource.source.CVSSourceFactory">
  <hostname>cvs.apache.org</hostname>
  <repository>/home/cvspublic</repository>
  <module>xml-cocoon2/src/documentation</module>
  <username>anoncvs</username>
  <password>anoncvs</password>
  <!-- optionnal hierarchy refresh period (in seconds, defaults to 600) -->
  <refresh-period>3600</refresh-period>

  <!-- optionnal commit message expression (sitemap syntax) -->
  <commit-message-expr>{request-attr:cvs-message}</commit-message-expr>

  <!-- Optionnal per-user authentication configuration.
  If this element is present, use its children to override default values.
  If this element isn't present, per-user authentication is disabled -->
```

1. <http://www.jcvs.org/>

```
<per-user-auth>
  <!-- username expression. Default is "{request-attr:cvs-user}" -->
  <username-expr>{session-attr:cvs-user}</username-expr>
  <!-- password expression. Default is "{request-attr:cvs-password}" -->
  <password-expr>{session-attr:cvs-passwd}</password-expr>
</per-user-auth>
</protocol>
```

You should add it to the `<source-handler>` section of `cocoon.xconf`.

Running the samples

Having added the above to your `cocoon.xconf`, you can copy the `src/samples/cvssource` directory to the mount directory of your Cocoon webapp.

The demo is then available at `http://localhost:8080/cocoon/mount/cvssource/` and allows you to browse the Cocoon documentation source files on the Apache CVS repository. This demo isn't really fast because of the slow response time of `cvs.apache.org`. Using a CVS server on the same LAN is far more fast.

Future directions

This component is far from being finished, and here are my current plans about it :

- Provide a multi-repository naming scheme : the current implementation requires one protocol definition per CVS repository. There are some applications where a multi-repository protocol may be useful (e.g. a `viewcvs` replacement).
- Improve performances. Directory browsing could be faster by using other CVS calls than the one used to retrieve several file properties at once. Having some CVS connection pools would also be a great improvement.
- Get rid of the JCVS client. The main reason for its use is because it's the only Java CVS client I found which is not intimately tied to a checkout on the local filesystem (NetBeans and Eclipse clients require it). But using it required some ugly hacks and "empirical programming" (try some method calls and look what's sent on the wire until it matches what you expect).

Acknowledgements

My work on this component is sponsored by my employer, [Anyware Technologies](http://www.anyware-tech.com/)³.

Many, many thanks to [Steven Noels](http://blogs.cocoondev.org/stevven/)⁴ for setting up `Cocoondev.org`.

Thanks in advance for *your* feedback and patches !

3. <http://www.anyware-tech.com/>

4. <http://blogs.cocoondev.org/stevven/>