SERENA®

DIMENSIONS® CM 14.2.0.2

CM Bridge Getting Started Guide

Serena Proprietary and Confidential Information

The content of this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Serena. Serena assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

Trademarks

Serena, TeamTrack, StarTool, PVCS, Comparex, Dimensions, Prototype Composer, Mariner, and ChangeMan are registered trademarks of Serena Software, Inc. The Serena logo and Version Manager are trademarks of Serena Software, Inc. All other products or company names are used for identification purposes only, and may be trademarks of their respective owners.

U.S. Government Rights

Any Software product acquired by Licensee under this Agreement for or on behalf of the U.S. Government, its agencies and instrumentalities is "commercial software" as defined by the FAR. Use, duplication, and disclosure by the U.S. Government is subject to the restrictions set forth in the license under which the Software was acquired. The manufacturer is Serena Software, Inc., 1850 Gateway Drive, 4th Floor, San Mateo California, 94404-4061.

Publication date: March 2016

Table of Contents

Welcome to Serena Dimensions CM

Serena® Dimensions® CM is a powerful process management and change control system. Dimensions CM helps you organize, manage, and protect your software development projects on every level—from storing and tracking changes to individual files, to managing and monitoring an entire development cycle.

Before you Begin

The Dimensions CM readme contains the following information:

- What's new
- Fixed issues
- Software compatibility requirements
- Installation notes
- Known issues

The readme is available online at:

http://help.serena.com/doc_center/doc_center.html#dcmDoc

Contacting Serena Technical Support

Serena provides technical support for all registered users of this product, including limited installation support for the first 30 days. If you need support after that time, contact Serena Support at the following web site and follow the instructions:

http://www.serena.com/support

Language-specific technical support is available during local business hours. For all other hours, technical support is provided in English.

You can use the Serena Support web page to:

- Report problems and ask questions.
- Obtain up-to-date technical support information, including that shared by our customers via the web, automatic email notification, newsgroups, and regional user groups.
- Access a knowledge base, which contains how-to information and allows you to search on keywords for technical bulletins.
- Download updates and fix releases for your Serena products.

Videos

Videos of Dimensions CM features can be viewed online at:

http://help.serena.com/doc_center/doc_center.html#dcmVid

License and Copyright Information for Third-Party Software

License and copyright information for third-party software included in this release can be found as part of the software download available at:

`http://www.serena.com/support`

Chapter 1

Introduction

What is Dimensions CM Bridge?

Dimensions CM Bridge is a connector between a Subversion enabled client and a CM server that enables:

- Your development tools to talk to a CM repository instead of Subversion.
- Development teams familiar with Subversion to continue using their existing IDEs and tools with the additional benefits of Dimensions CM.
- Dimensions CM to be used with IDEs that do not have a “native” integration with CM but do support Subversion.

Serena supports any Subversion client that uses the HTTPv1 protocol. Serena has certified CM Bridge with the following clients:

- Apple Xcode version 6.1 on Apple Macintosh OS X 10.8 and 10.9
- TortoiseSVN versions 1.7 and 1.8
- Subversion command line versions 1.7 and 1.8
- SoMachine Motion Logic Builder 4.0
- IntelliJ IDEA 14

NOTE CM Bridge is automatically installed with a Dimensions CM server.

Dimensions CM Repository URL

A Dimensions CM repository is not structured as a single file and folder tree like Subversion, instead each stream has its own files and folders. When working with a Dimensions CM stream from a Subversion client you must specify the Dimensions CM *product* and *stream* names in uppercase as part of the URL. For example:

- URL that points to the root of the QLARIUS product:

  ```
  http://<web server>:8080/cmbridge/QLARIUS/
  ```

- URL that points to the root folder of the JAVA_BRANCHA_STR stream in the QLARIUS product:

  ```
  http://<web server>:8080/cmbridge/QLARIUS/JAVA_BRANCHA_STR
  ```

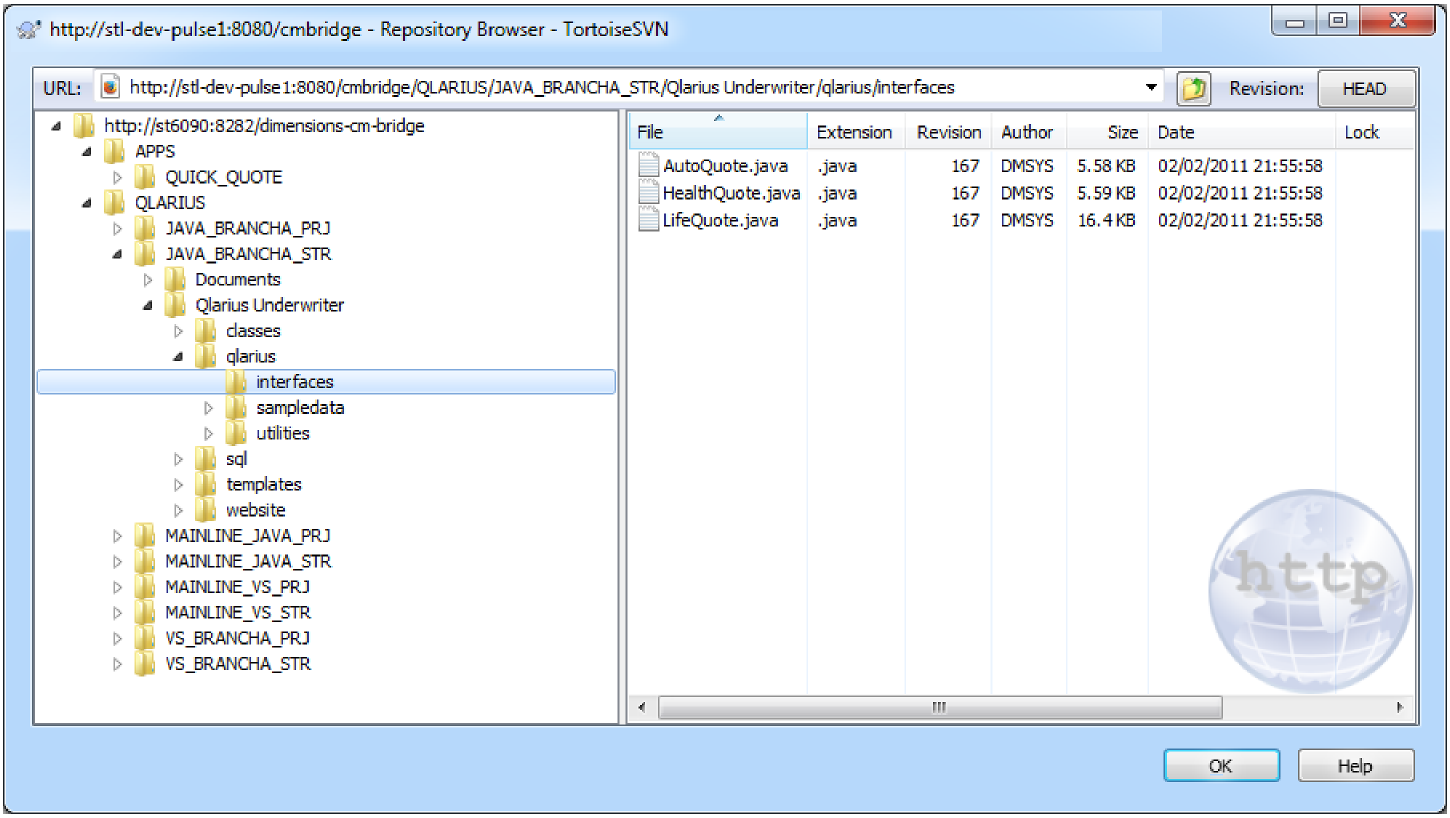
- URL that points to a folder and file in the JAVA_BRANCHA_STR stream:

  ```
  http://<web server>:8080/cmbridge/QLARIUS/JAVA_BRANCHA_STR/Qlarius
      Underwriter/qlarius/interfaces/AutoQuote.java
  ```

  TIP Products and streams are treated as directories in a repository.

- In the example below there are two products, *APPS* and *QLARIUS*, and the streams are listed as child folders of the products. There are multiple folders and files under JAVA_BRANCHA_STR.

NOTE You can use a web browser, SVN-based clients, or the SVN command line to view the contents of a repository.



http://stl-dev-pulse1:8080/cmbridge - Repository Browser - TortoiseSVN

URL: http://stl-dev-pulse1:8080/cmbridge/QLARIUS/JAVA_BRANCHA_STR/Qlarius Underwriter/qlarius/interfaces

Revision: HEAD

- http://st6090:8282/dimensions-cm-bridge
  - APPS
    - QUICK_QUOTE
  - QLARIUS
    - JAVA_BRANCHA_PRJ
    - JAVA_BRANCHA_STR
      - Documents
      - Qlarius Underwriter
        - classes
        - qlarius
          - interfaces
          - sampledata
          - utilities
        - sql
        - templates
        - website
    - MAINLINE_JAVA_PRJ
    - MAINLINE_JAVA_STR
    - MAINLINE_VS_PRJ
    - MAINLINE_VS_STR
    - VS_BRANCHA_PRJ
    - VS_BRANCHA_STR

File	Extension	Revision	Author	Size	Date	Lock
AutoQuote.java	.java	167	DMSYS	5.58 KB	02/02/2011 21:55:58	
HealthQuote.java	.java	167	DMSYS	5.59 KB	02/02/2011 21:55:58	
LifeQuote.java	.java	167	DMSYS	16.4 KB	02/02/2011 21:55:58	

Support for Subversion Properties

CM Bridge supports the storage and retrieval of Subversion properties. The following standard Subversion operations are supported:

- `propset`
- `propget`
- `propedit`

For details about using SVN externals with CM Bridge see page 22.

Limitations and Known Issues

- Operations are only supported with Dimensions CM streams (not projects).
- The log output is empty for a file or folder that has not changed since a stream was created.
- When files are checked-out or exported, item header substitution is not performed.
- Dimensions CM directory items are not supported.
- All Subversion commands are supported except:
  - `merge`
  - `mergeinfo`
  - `copy (`you can only copy streams and projects in a single product).
  - `switch`
- SVN tagging is not supported.
- SVN branches are only supported at the product level.

Troubleshooting

When investigating issues with CM Bridge first look in the log file:

```
C:\ProgramData\Serena\Bridge\_logs
```

This log may include an error message or stack trace that can help Serena Support to identify the issue.

TIP

- If you are having an issue in a particular client, try reproducing the problem from a Subversion command line as this can display a more detailed error message.
- Include logs, screenshots, error message text, and the steps you performed to reproduce a problem when you are reporting an issue to Serena Support.

Chapter 2

Apple Xcode

Introduction

The examples below describe some of the ways that you can use CM Bridge with Apple Xcode.

Watch a video about using Apple Xcode with Dimensions CM (Internet access is required):

http://help.serena.com/doc_center/demos/CM_141_xcode.html

Add an Xcode Project to Dimensions CM

You can add an existing Apple Xcode project to a stream in a Dimensions CM repository.

1 From a Terminal window change directory to the folder where the Xcode project is located.

2 Use the Subversion command line to import the folder into Dimensions CM. For example:

```
svn import Example http://<web server>:8080/cmbridge/QLARIUS/XCODE/
Example -m "Initial commit" --username <user ID> --password
<password>
```

Check Out an Xcode Project from Dimensions CM

You can check out an Xcode project from Dimensions CM.

1 Open Xcode.

2 Do one of the following:

- From the Welcome screen select **Check out an existing project**.
- From the Source control menu select **Check Out**.

The Check Out dialog box appears.

3 Do one of the following:

- Select a previously used repository.
- Enter a URL that identifies a Dimensions CM server and product in this format:

```
http://<web server>:8080/cmbridge/<CM product>
```

For example:

```
http://dev-pulse:8080/cmbridge/QLARIUS/
```

4 Click **Next**.

5 From the repository **Type** list select Subversion.

6 If prompted enter a CM user ID and password and click **Next**.

7 Open a CM product.

8 Select the project or stream to be checked out and click **Next**.

9 Select the local directory where the project or stream will be checked out.

10 Click **Check Out.** The project or stream is checked out from Dimensions CM to the local directory.

Commit Changes in an Xcode Project to Dimensions CM

You can commit changes that you make in an Xcode project to a stream or project in Dimensions CM. You can also relate commits to Dimensions CM change requests.

1 From the **Source Control** menu select **Commit**.

2 In the **Messages** box:

- Enter a commit comment for your changes.
- Optionally enter the IDs of one or more Dimensions CM change requests as part of the comment in this format: [CR1,CR2]. For example:

```
[QLARIUS_CR_21,QLARIUS_CR_36] Your comment
```

3 Click **Commit**.

Update an Xcode Project

You can update the local version of an Xcode project with changes committed by other developers in your team. From the **Source Control** menu select **Update**.

Chapter 3

TortoiseSVN

Introduction

CM Bridge supports the majority of TortoiseSVN operations. The following examples describe some of the ways that you can use CM Bridge with TortoiseSVN.

Watch a video about using TortoiseSVN with Dimensions CM (Internet access is required):

http://help.serena.com/doc_center/demos/CM_141_tortoisesvn.html

Browse Files and Folders

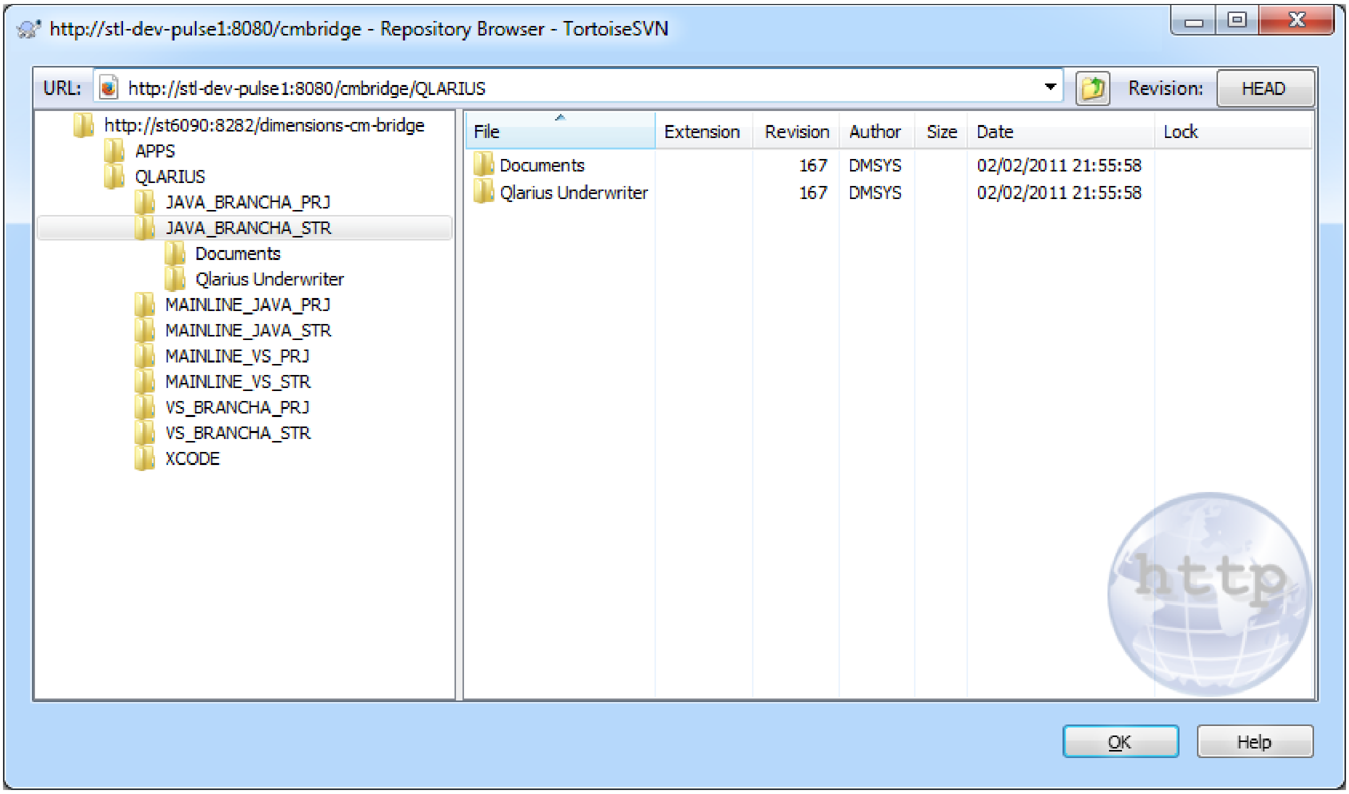
You can use TortoiseSVN to browse the folder structure of a stream and view the contents of files.

1 From Windows Explorer open the TortoiseSVN repository browser.

2 Enter a URL to a Dimensions CM stream, for example:

```
http://<web server>:8080/cmbridge/QLARIUS/JAVA_BRANCHA_STR
```

3 If prompted enter a CM user ID and password. The structure of the repository is displayed in the repository browser and the stream that you specified is expanded:



URL: http://stl-dev-pulse1:8080/cmbridge/QLARIUS

Revision: HEAD

File	Extension	Revision	Author	Size	Date	Lock
Documents		167	DMSYS		02/02/2011 21:55:58	
Qlarius Underwriter		167	DMSYS		02/02/2011 21:55:58	

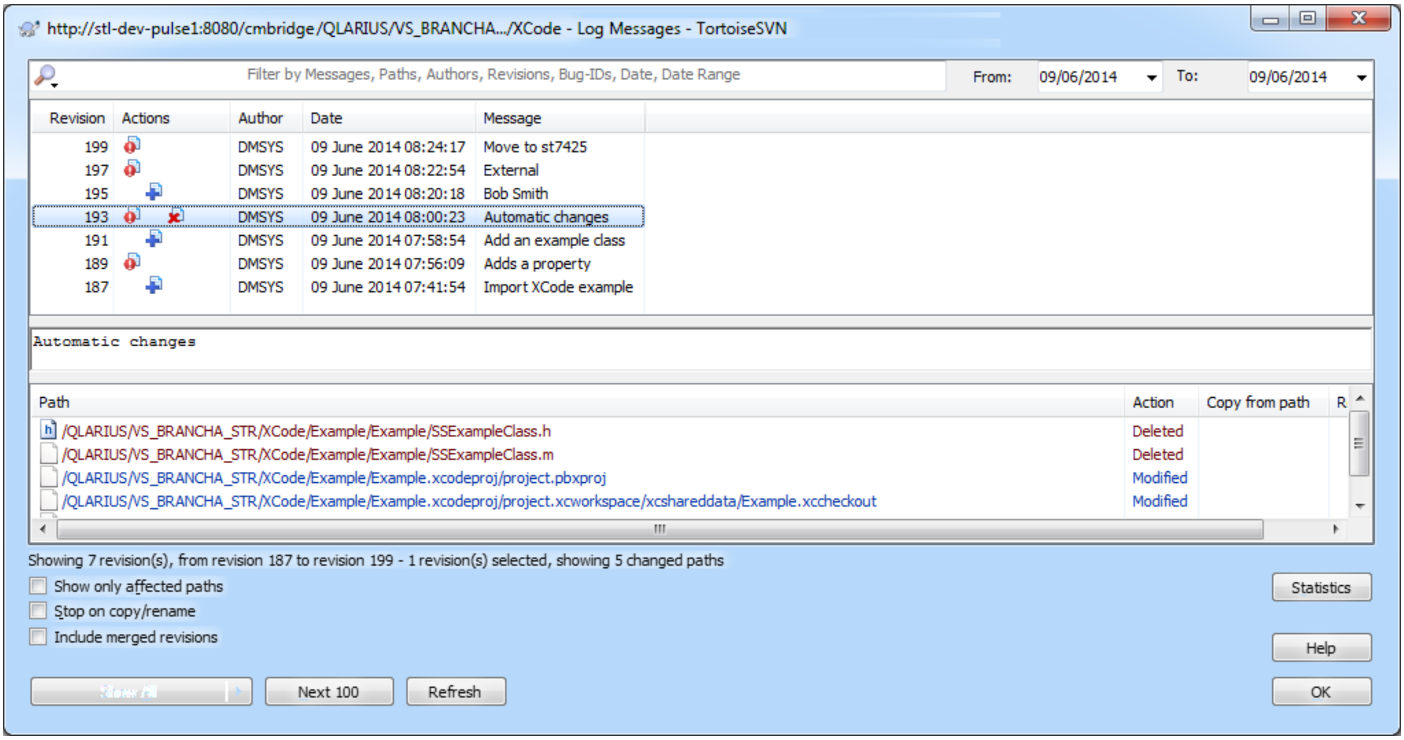
4 To browse a folder's contents, select it.

5 To browse a file, right-click it and select **Open**. The file is downloaded from Dimensions CM and opened in the default application for the file type.

View Log Messages

You can use TortoiseSVN to view the history of any object and see what changes were delivered to it.

Right-click a folder and select **Show Log**. The details of all the changesets delivered to that folder are displayed including the files modified in each commit.



http://stl-dev-pulse1:8080/cmbridge/QLARIUS/VS_BRANCHA.../XCode - Log Messages - TortoiseSVN

Filter by Messages, Paths, Authors, Revisions, Bug-IDs, Date, Date Range

From: 09/06/2014 To: 09/06/2014

Revision	Actions	Author	Date	Message
199		DMSYS	09 June 2014 08:24:17	Move to st7425
197		DMSYS	09 June 2014 08:22:54	External
195		DMSYS	09 June 2014 08:20:18	Bob Smith
193		DMSYS	09 June 2014 08:00:23	Automatic changes
191		DMSYS	09 June 2014 07:58:54	Add an example class
189		DMSYS	09 June 2014 07:56:09	Adds a property
187		DMSYS	09 June 2014 07:41:54	Import XCode example

```
Automatic changes
```

Path	Action	Copy from path	R
/QLARIUS/VS_BRANCHA_STR/XCode/Example/Example/SSExampleClass.h	Deleted		
/QLARIUS/VS_BRANCHA_STR/XCode/Example/Example/SSExampleClass.m	Deleted		
/QLARIUS/VS_BRANCHA_STR/XCode/Example/Example.xcodeproj/project.pbxproj	Modified		
/QLARIUS/VS_BRANCHA_STR/XCode/Example/Example.xcodeproj/project.xcworkspace/xcshareddata/Example.xccheckout	Modified		

Showing 7 revision(s), from revision 187 to revision 199 - 1 revision(s) selected, showing 5 changed paths

☐ Show only affected paths
☐ Stop on copy/rename
☐ Include merged revisions

Check Out a Folder

You can use TortoiseSVN to get the contents of a stream to an area on disk.

1 From Windows Explorer open the TortoiseSVN repository browser.

2 Enter a URL to a Dimensions CM stream, for example:

```
http://<web server>:8080/cmbridge/QLARIUS/JAVA_BRANCHA_STR
```

3 If prompted enter a CM user ID and password. The structure of your repository is displayed and the stream you specified is expanded.

4 Right-click the stream and select **Checkout**.

5 Enter the location where you want to check out the files

6 Click **OK**. The files and folders are copied from the Dimensions CM stream to the disk.

Commit Changes to Dimensions CM

You can use TortoiseSVN to commit changes that you have made on disk back to a Dimensions CM repository. You can also relate commits to Dimensions CM change requests.

1 If you have added a new file, to register the addition right-click it and from the **TortoiseSVN** menu select **Add**.

2 To commit all your changes, navigate to the stream's parent folder, right-click, and select **SVN Commit**.

3 In the **Commit** box:

- Enter a commit comment for your changes.
- Optionally enter the IDs of one or more Dimensions CM change requests as part of the comment in this format: [CR1,CR2]. For example:

```
[QLARIUS_CR_21,QLARIUS_CR_36] Your comment
```

4 Click **OK**.

Chapter 4

Subversion Command Line

Introduction

CM Bridge supports the majority of Subversion commands. The following examples describe some of the ways that you can use CM Bridge with the Subversion command line.

Watch a video about using the Subversion command line with Dimensions CM:

http://help.serena.com/doc_center/demos/CM_141_svn.html

List Files and Folders

Use the Subversion *ls* command to browse the folder structure of a stream and view its details, for example:

```
svn ls “http://<web server>:8080/cmbridge/QLARIUS/JAVA_BRANCHA_STR”
    --username <user ID> --password <password>
```

Use the *ls* verbose option to list additional details about the files and folders and list the subfolders, for example:

```
svn ls -v “http://<web server>:8080/cmbridge/QLARIUS/JAVA_BRANCHA_STR/
    Qlarius Underwriter/qlarius/interfaces” --username <user ID> --
    password <password>
```

The verbose directory listing contains additional information about each file or folder, such as:

- The repository version number when it was last modified.
- The user who last modified it.
- The date and time of the last modification.

List Previous Revisions of Files and Folders

Use the Subversion *log* command to browse the history of a stream's folder structure in a repository and view details of previous revisions of files and folders, for example:

```
svn log "http://<server>:8080/cmbridge/QLARIUS/JAVA_BRANCHA_STR" --
    username <user ID> --password <password>
```

Output:

```
------------------------------------------------------------------------
    -
r199 | EMMA  | 2014-06-09 08:24:17 +0100 (Mon, 09 Jun 2014) | 1 line
Delivered some changes for the CM Bridge preview
------------------------------------------------------------------------
    -
r197 | DMSYS | 2014-06-09 08:22:54 +0100 (Mon, 09 Jun 2014) | 1 line
External test
------------------------------------------------------------------------
    -
r195 | USER1 | 2014-06-09 08:20:18 +0100 (Mon, 09 Jun 2014) | 1 line
Initial delivery
```

In the example above there have been three commits creating repository versions r199, r197, and r195.

Use the *ls* command -r option to list the contents of a specific version of a stream, for example:

```
svn ls –v –R –r 195 "http://<web server>:8080/cmbridge/QLARIUS/
    JAVA_BRANCHA_STR" --username <user ID> --password <password>
```

This command lists the contents of the stream at revision 195.

Check Out a Stream

Use the Subversion *checkout* command to check out an entire stream to a local disk, for example:

```
svn checkout "http://<web server>:8080/cmbridge/QLARIUS/
    JAVA_BRANCHA_STR" --username <user ID> --password <password>
```

Commit Changes to Dimensions CM

Use the Subversion *add* command to register files that have been added and the *commit* command to deliver changes made on disk back to a Dimensions CM stream. You can also relate commits to Dimensions CM change requests.

1 Change directory to the stream's parent folder.

2 Enter the `commit` command. Optionally enter the IDs of one or more Dimensions CM change requests as part of a comment in this format: [CR1,CR2]. For example:

```
svn commit –m "[QLARIUS_CR_21,QLARIUS_CR_36] your comment " --username
    <user ID> --password <password>
```

View File Annotations

Use the Subversion *blame* command to view an annotated copy of a file and display the users who modified each line of code and when, for example:

```
svn blame "http://<web server>:8080/cmbridge/QLARIUS/JAVA_BRANCHA_STR/
    MyNewFile.txt" --username <user ID> --password <password>
```

The command lists which repository version last changed each line and the user who made the change, for example:

```
211      DMSYS This is the first line of the file
213      EMMA Emma added this line
213      EMMA and this line too
211      DMSYS this is the last line
```

Use SVN Properties

You can use SVN properties to create a link between folders so that the work area for one stream can use content from another stream.

1 Use the `checkout` command to get the content of the first stream:

```
svn checkout "http://<web server>:8080/cmbridge/QLARIUS/
    JAVA_BRANCHA_STR" --username <user ID> --password <password>
```

2 Use the `propset` command to set the `svn:externals` property on a folder in the first stream to point to a folder in the second stream:

```
svn propset svn:externals "<URL of shared folder > <folder name in
    work area>" "<folder location in work area>"
```

For example:

```
svn propset svn:externals "http://<web server>:8080/cmbridge/APPS/
    QUICK_QUOTE/Shared MyApps" "Qlarius_Underwriter"
```

This command links the folder *Shared* from the *APPS:QUICK_QUOTE* stream to the subfolder `MyApps` in the folder `Qlarius_Underwriter` in the stream that you have checked out on disk.

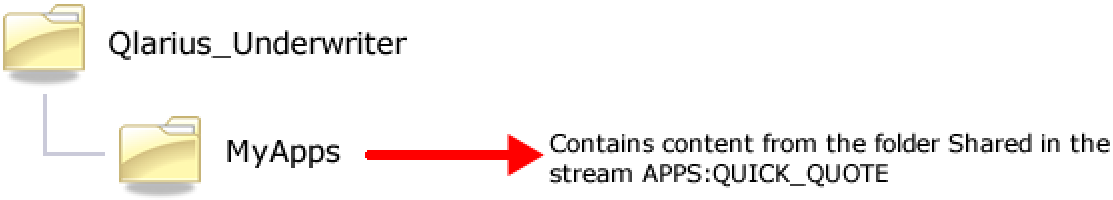
3 Use the `commit` command to deliver the property change:

```
svn commit -w “Linked the Shared folder”
```

4 Update your working copy or checkout the stream to an empty folder, for example:

```
svn checkout “http://<web server>:8080/cmbridge/QLARIUS/
    JAVA_BRANCHA_STR” --username <user ID> --password <password>
```

The folder `Qlarius_Underwriter` is checked out and the externals are fetched to the correct location on disk.


Qlarius_Underwriter
MyApps
Contains content from the folder Shared in the stream APPS:QUICK_QUOTE

You can also use SVN properties to set and modify the properties for any file or folder from the Subversion command line.

1 Use the `checkout` command to get the content of the stream.

2 Use the `propset` command to set a custom property on a file in the stream:

```
svn propset <property> "<value>" "<URL of file>"
```

For example, to set the custom property ‘complexity’ to ‘high’ on the file `MyNewFile.txt` in the stream JAVA_BRANCHA_STR:

```
svn propset complexity "high" http://<web server>:8080/cmbridge/
    QLARIUS/JAVA_BRANCHA_STR/MyNewFile.txt
```

3 Use the `commit` command to deliver the change:

```
svn commit –m “Set Complexity to High” --username <user ID> --
    password <password>
```

Using Git with a Dimensions CM Repository

Git users can connect to a Dimensions CM repository instead of Subversion. When you clone a Dimensions stream to a local Git repository it contains a link to your remote CM repository and you can run any Git `svn` command if your current directory is a local Git repository.

NOTE Remote branches and repositories are not supported.

To clone an existing Dimensions CM repository to a local machine:

```
git svn clone http://<CM server name>:8080/cmbridge/<CM product>/
    <CM stream or project> <Path of local folder>
```

For example:

```
git svn clone http://cmserver:8080/cmbridge/QLARIUS/GIT1 C:/dimWa1
```

Index