

Using the Command Line Interface for the Documentation Generator

Joseph Wong

Last modified: August 24, 2006

Overview

The Quark Platform comes with a documentation generator for producing HTML documentation from CALDoc comments and metadata for the various entities in a CAL workspace. This generator can be accessed via a command line interface through the CALDocTool class.

Processing Details

The documentation generator produces one complete set of documentation each time it is run; it does not function incrementally. The tool also relies upon the compilation of the CAL workspace; hence all modules contained therein must be compilable without errors.

Synopsis

The CALDocTool can be invoked in this manner:

```
java -cp <classpath> org.openquark.cal.caldoc.CALDocTool <workspace name> [options]
```

Here, <classpath> represents the Java classpath that includes all the necessary binary directories and jars required for the Quark Platform and the CAL workspace declaration file specified by <workspace name>.

Alternatively, the tool can be invoked from an unpacked OpenQuark distribution using:

```
javacp org.openquark.cal.caldoc.CALDocTool <workspace name> [options]
```

Options

The documentation generator can be configured with the following set of command-line options.

-d *directory*

Specifies the output directory where the generator saves the generated HTML files. The directory can be absolute, or relative to the current working directory. The output directory is automatically created if it does not already exist.

Omitting this option causes the files to be saved to the current directory.

-public

Specifies that documentation should be generated for public entries only. This is the default behaviour of the documentation generator.

-private

Specifies that documentation should be generated for both public and private entries.

-modules *regex*

The documentation generator will only generate documentation for modules whose names match the specified regular expression.

For example, specifying the regular expression "`File.*`" would cause the generator to only generate documentation for modules whose names begin with the prefix `File`.

-excludeModules *regex*

The documentation generator will not generate documentation for modules whose names match the specified regular expression.

For example, specifying the regular expression "`.*_Tests`" would cause the generator to exclude from generation any module whose name ends with the suffix `_Tests`.

-entities *regex*

The documentation generator will exclude from the documentation any functions, data constructors, and class methods whose names do not match the specified regular expression.

-excludeEntities *regex*

The documentation generator will exclude from the documentation any functions, data constructors, and class methods whose names match the specified regular expression.

For example, specifying the regular expression

`"(*Example.*)|(*example.*)|(*test.*)|(*Test.*)"` would cause the generator to exclude entities whose names contain the string `Example`, `example`, `test`, or `Test`.

-metadata**-metadata override**

Specifies that CAL metadata should be included in the generated documentation.

If the “override” option is given, then the CALDoc for a CAL entity will not be included in the documentation if the corresponding metadata is present.

-author

Specifies that author information should be included in the generated documentation.

-version

Specifies that version information should be included in the generated documentation.

-qualifyPreludeNames

Specifies that the names of Prelude entities should be shown as fully qualified names outside the documentation of the Prelude module.

-use

Specifies that one usage page should be generated for each documented type and type class. The page describes which functions, data constructors, and class methods refer to the type/type class in their type signatures, as well as any associated class instances.

-locale *country_language_variant*

Specifies the locale that the documentation generator uses when generating documentation. The argument is the name of the locale, as described in *java.util.Locale* documentation, such as `en_US` (English, United States) or `en_US_WIN` (Windows variant).

If the **-metadata** option is specified, the locale specified by this option is used to access the localized version of the metadata.

-windowTitle *title*

Specifies the title to be placed in the HTML `<title>` tag. This appears in the window title and in any browser bookmarks (favorite places) that someone creates for this page. This title should not contain any HTML tags, as the browser will not properly interpret them.

-docTitle *title*

Specifies the title to be placed near the top of the overview file. The title may contain HTML tags and white space.

-header *header*

Specifies the header text to be placed at the top of each output file. The header will be placed just below the upper navigation bar. The text may contain HTML tags and white space.

-footer *footer*

Specifies the footer text to be placed at the bottom of each output file. The footer will be placed just above the lower navigation bar. The text may contain HTML tags and white space.

-bottom *text*

Specifies the fine-print text to be placed at the bottom of each output file. The text will be placed at the bottom of the page, below the lower navigation bar. The text may contain HTML tags and white space.

-verbose

Provides more detailed messages while the documentation generator is running.

-quiet

Shuts off non-error and non-warning messages, leaving only the warnings and errors appear, making them easier to view. Also suppresses the tool's version string.

-help

Displays the command line options.

Non-standard -X options

-XexcludeTestModules

Indicates to the documentation generator that *test modules* are to be excluded from the generated documentation. A test module is defined to be a module whose source file has a directory named `test` as part of its path.

Copyright (c) 2007 BUSINESS OBJECTS SOFTWARE LIMITED
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of Business Objects nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.