jcov

JCov

The JCov open source project is used to gather quality metrics associated with the production of test suites. JCov is being opened in order to facilitate the practice of verifying test execution of regression tests in OpenJDK development.

The main motivation behind JCov is *transparency of test coverage metrics*. The advantage to promoting standard coverage based on JCov is that OpenJDK developers will be able to use a code coverage tool that stays in 'lock step' with Java language and VM developments.

JCov is a pure java implementation of a code coverage tool which provides a means to measure and analyze dynamic code coverage of Java programs. JCov provides functionality to collect method, linear block and branch coverage, as well as showing uncovered execution paths. It is also able to show a program's source code annotated with coverage information.

From a testing perspective, JCov is most useful to determine execution paths (in a Java application) that a test suite is (or is not) executing.

JCov supports applications on JDK 1.0 and higher (including JDK 8), CDC/CLDC 1.0 and higher, and JavaCard 3.0 and higher.

The JCov open source project is created in order to develop a community that will improve it, further its development, and use it to develop test suites. We encourage you to browse, download, contribute, and get involved.

News and Announcements

The JCov project opened in the CodeTools Project of OpenJDK (March 21, 2014)

Quick Links

Downloads (Provided by Adopt OpenJDK)

- Tip (latest <dev>)
- Released

Usage and development

- User's Guide - html | pdf
- FAQ
- Open issues

- Build instructions
- Repository and code structure
- Developer's guide

Related projects

- JT Harness
- Jtreg
- SigTest

Blogs

- Java Compatibility Tools
- Jaroslav Tulach

Related Projects and Links

This section contains links to technologies and blogs that are related to the JCov project.

Development Tools

AsmTools	Assemblers and Disassemblers for producing Java .class files
SigTest	A static analyzer used to to describe platform compatibility and API Changes

JT Harness Testing Harness and Extensions

JT Harness	The JT harness is a flexible test harness that can be used to control and execute test suites.
ME Framework	The ME Framework is an open source set of JT harness plugins that supports the Java ME platform. Test suite architects use the JT harness and the ME Framework to construct test suites for Java ME technologies.
jtreg	The jtreg test harness is used by the JDK test framework. This framework is intended primarily for regression tests. It can also be used for unit tests, functional tests, and even simple product tests.

Blogs

Java Compatibility Tools Blog	This blog provides topics of general interest for people writing tools to test Java platform compatibility and conformance.
Jaroslav Tulach	The NetBeans architect blogs about SigTest. He also maintains an excellent Wiki about API design and evolution.