Summary

Test Coverage shows how much test code covers over the source code by percentile and code lines. This document provides a guide on analyzing Test Coverage, and reporting the result.

Description

Test Coverage is a task of measuring test code over developed source code. It’s beneficial to understand following ideas.

Instrument : refers to tasks such as running target test code, performing JUnit Test(EMMA), and injecting test code info to target code while compiling(Cobertura.)

Test Coverage report : Test Coverage Result Reporting (XML, HTML, and etc)

XML and HTML are basically supported. EMMA supports TXT, too.

Test Coverage Tool

There are lots of tools to perform Test Coverage. Each maintains own logic to measure coverage, and reporting corresponds to the logic.Therefore, choose right tool for the needs.

This IDE selects EMMA and EclEmma due to license agreements.

However, Cobertura is also recommended to be used as a tool not to fix source code.

Cobertura Features

Though it isn’t included in the IDE, it was one of the considerations. Cobertura has following features.

Home : http://cobertura.sourceforge.net/ [http://cobertura.sourceforge.net/]

License : Apache Software License (Cobertura ant task), GPL 2.0

Free tool based on jcoverage. Calculates percentage of codes accessed by test code.

Executable from ant, Maven, and command line.

Compile target code, then instrument Java bytecode.

Reports in HTML, XML.

Shows Line / branch percentages of class, package, all classes.

Notifies McCabe's cyclomatic code complexity on class, package, all classes

[http://www.arisa.se/compendium/node96.html].

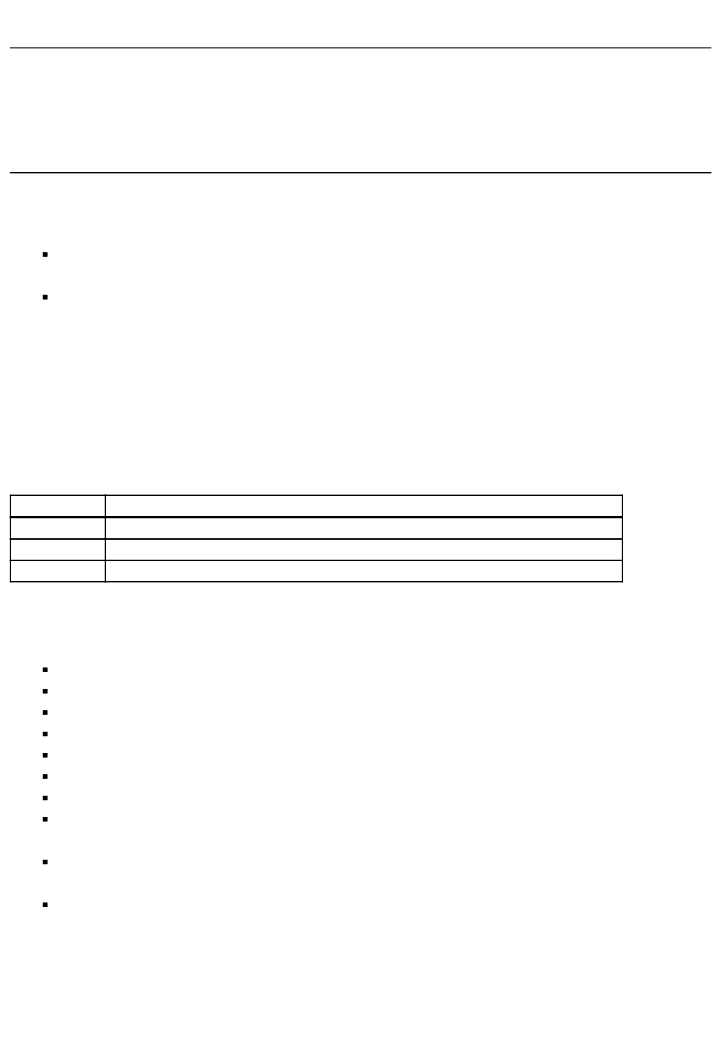
Provides sorting (ascending, descending) on Class Names, Covered line percentile, Covered branch percentile from HTML result.

Save the report in UTF-8.

Test Coverage - EMMA

EMMA is an Open Source Project licensed under CPL (Common Public License) v1.0.

It is executable from Ant and Maven, even in Command-line.



|  |  |
| --- | --- |
|  | Tool |
| Commercial | C lover [http://www.ce nqua.com /clove r] |
| Open Source | C obertura [http://cobe rtura.source forge .ne t], EMMA [http://e m m a.source forge .ne t/] |
| Eclipse Plug-in | EclEmma [http://www.e cle m m a.org/], C overlipse [http://cove rlipse .source forge .ne t/] |

EMMA features

Home page : http://emma.sourceforge.net/ [http://emma.sourceforge.net/]

Perform Coverage instrument before or while loading class.

Supports 4 Coverage types: class, method, line, basic block.

1)

Coverage statistics are provided separatedly for method, class, package, all classes.

Output Report type : Text, HTML, XML. HTML provides link to source code.

Highlight for Coverage items in HTML report.

Instrument individual .class file or whole .jar files.

Extremely fast. Memory footprint is also very small: several hundred bytes per Java class.

Eclipse support with EclEmma Plug-in.

Test Coverage Process using EMMA

Write TestCase

Evaluate Test Coverage using EclEmma in Eclipse

Generate build Script to auto-execute Test Coverage

Generate Report

EMMA Test Coverage HTML Report

Directory to generate report : In Ant, user-specified directory. Maven provides default location: target/site/emma/index.html, coverage.xml

Summary : Percentage for each class, method, block, line. Refer to EMMA HTML Report Sample

Environmental settings

Install EclEmma Eclipse Plugin

Install Ant, Maven Eclipse Plugin

Manual

How to use EclEmma

See EclEmma Home [http://www.eclemma.org/userdoc/index.html] for detail.

1. Install EclEmma : Add following URL to Help > Software Updates…

http://update.eclemma.org/

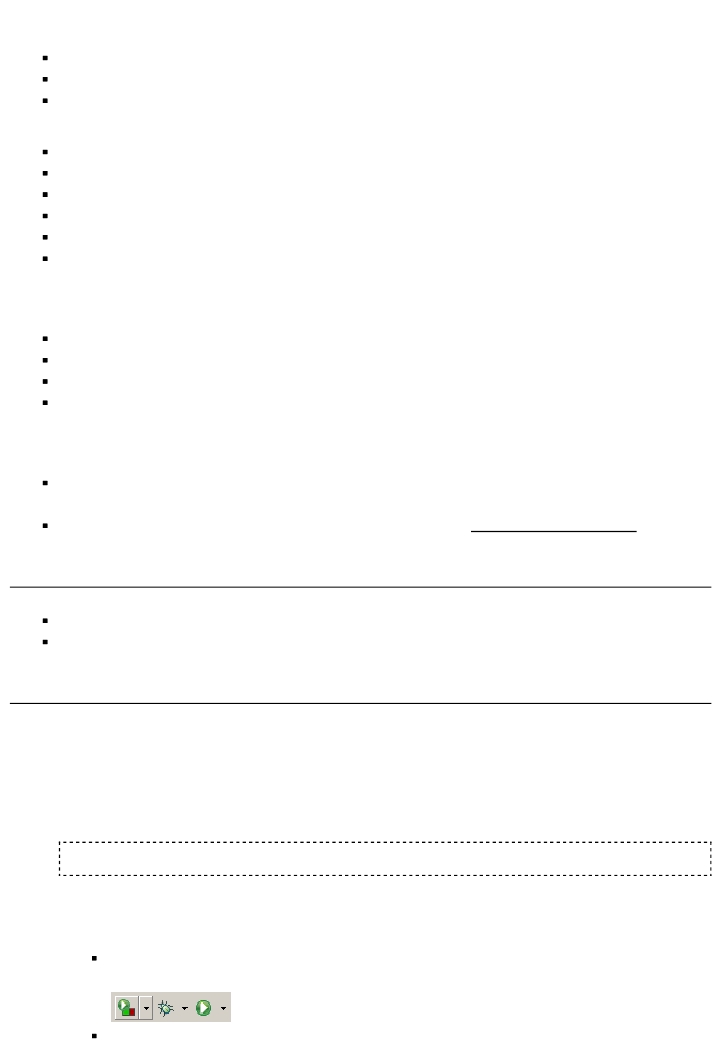
Click [Install…]

2. Run Test Coverage:

Using tool bar : Following icon will appear after installing EclEmma. Click Coverage As or select

Coverage Configuration menu to run.

Using Context Menu : Select project or test code, then right-click and select Coverage As > JUnit Test in the contextual menu to run.



[http://emma.sourceforge.net/faq.html#q.blockcoverage]

3. Check Coverage View: Coverage View will display result once Test Coverage instrument task is finished while running JUnit Test. Double click the item to open the source in editor view.

Run EMMA using Maven

1. Write pom.xml : Refer to EMMA configuration using Maven.

2. Run Maven : Refer to Running Maven in Eclipse

instrument : goal - emma:instrument

instrument + HTML Report : goal - emma:emma

3. Review report: You can find coverage.xml and index.html (emma:emma) report in target/site/emma directory.

Run EMMA using Ant

Using EMMA in Ant is a bit complicated. Refer to EMMA Site

[http://emma.sourceforge.net/userguide\_single/userguide.html#introANT] and Using EMMA With ANT

For JUnit Test Coverage Reporting

[http://wiki.metawerx.net/wiki/UsingEMMAWithANTForJUnitTestCoverageReporting] to run ANT.

1. Write build.xml : Refer to EMMA configuration sample using Ant

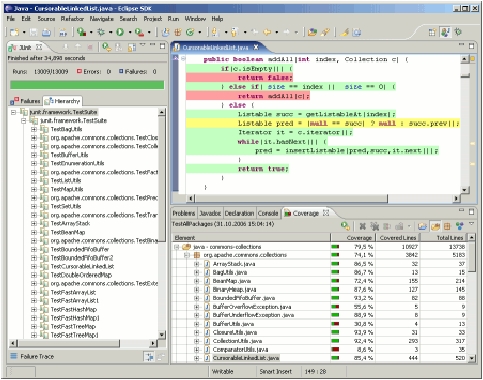
2. run target of Ant : Refer to Running Ant in Eclipse to execute specified target.

3. Check report : You can find report in specified location: outfile.

Sample

EMMA Configuration Sample using Maven

For Overall configuration, Refer to Configure build.xml to execute EMMA .



1. instrument : Analyze Test Coverage, report in XML result

<build>

<plugins>

<!-- test -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-surefire-plugin</artifactId>

<configuration>

<skipTests>false</skipTests>

<forkMode>always</forkMode>

<reportFormat>xml</reportFormat>

</configuration>

</plugin>

<!-- EMMA

-->

<plugin>

<groupId>org.codehaus.mojo</groupId>

<artifactId>emma-maven-plugin</artifactId>

<version>1.0-alpha-1</version>

</plugin>

</plugins>

</build>

2. Generate Test Coverage HTML Report

<reporting>

<plugins>

<!-- EMMA Coverage Reporting -->

<plugin>

<groupId>org.codehaus.mojo</groupId>

<artifactId>emma-maven-plugin</artifactId>

<inherited>true</inherited>

</plugin>

</plugins>

</reporting>

EMMA Configuration Sample using Ant

For Overall configuration, Refer to Configure build.xml to execute EMMA.

1. instrument : Analyze Test Coverage, report in XML result

<!-- EMMA 세 팅 생 략 -->

<target name="emma-jars" depends="build" description="Uses Emma to instrument the jar files">

<emma enabled="${emma.enabled}">

<instr mode="fullcopy"

outdir="${basedir}/build/temp"

merge="yes"

filter="egovframework.dev.tst.\*"

metadatafile="${artifactsDir}/test-coverage/coverage.em">



<instrpath>

<fileset dir="build/" includes="${ant.project.name}.jar" />

</instrpath>

</instr>

</emma>

</target>

<target name="test.with.emma" depends="emma-jars">

. . . 중략 . . .

<junit fork="yes"

printsummary="yes"

haltonfailure="no"

failureproperty="test.failed"

errorproperty="test.failed"

dir="${basedir}">

<classpath>

<path refid="master-classpath" />

<path refid="test-classpath" />

<path refid="emma.lib" />

<fileset dir="${basedir}/build/temp/lib"

includes="${ant.project.name}.jar" />

<pathelement location="${testbuild.dir}" />

</classpath>

<formatter type="xml" />

<batchtest fork="yes" todir="${artifactsDir}/test-results/xml">

<fileset dir="${testbuild.dir}">

<include name="\*\*/\*Test.class" />

</fileset>

</batchtest>

</junit>

</target>

2. Create Test Coverage Report

<!-- Test Code Coverage Report

-->

<target name="emmareport" depends="test.with.emma">

<move file="${basedir}/coverage.ec"

todir="${artifactsDir}/test-coverage" />

<emma description="now we can generate the emma report"

enabled="${emma.enabled}">

<report sourcepath="${src.dir}"

sort="+name,+method,+class"

metrics="method:70,line:80,class:100"

depth="method"

columns="name,class,method,block,line"

encoding="UTF-8">

<infileset dir="${artifactsDir}/test-coverage"

includes="\*.em, \*.ec" />



<!-- XML Report -->

<xml outfile="${artifactsDir}/test-coverage/coverage.xml" />

<!-- Text Report -->

<txt outfile="${artifactsDir}/test-coverage/coverage.txt" />

<!-- HTML Report -->

<html outfile="${artifactsDir}/test-coverage/coverage.html" />

</report>

</emma>

</target>

EMMA HTML Report Sample

EMMA created Test Coverage HTML Report Sample

References

EMMA Home : http://emma.sourceforge.net/ [http://emma.sourceforge.net/]

EclEmma Home : http://www.eclemma.org/ [http://www.eclemma.org/]

Maven EMMA plugin : http://emma.sourceforge.net/maven-emma-plugin/

[http://emma.sourceforge.net/maven-emma-plugin/]

1)

Base unit for EMMA coverage, somewhat like sequence of byte code command.

