Test coverage

Alexandre Bergel
abergel@dcc.uchile.cl
23/04/2013

Test coverage

Code coverage is a *measure* used in software testing

It describes the *degree* to which source code is tested

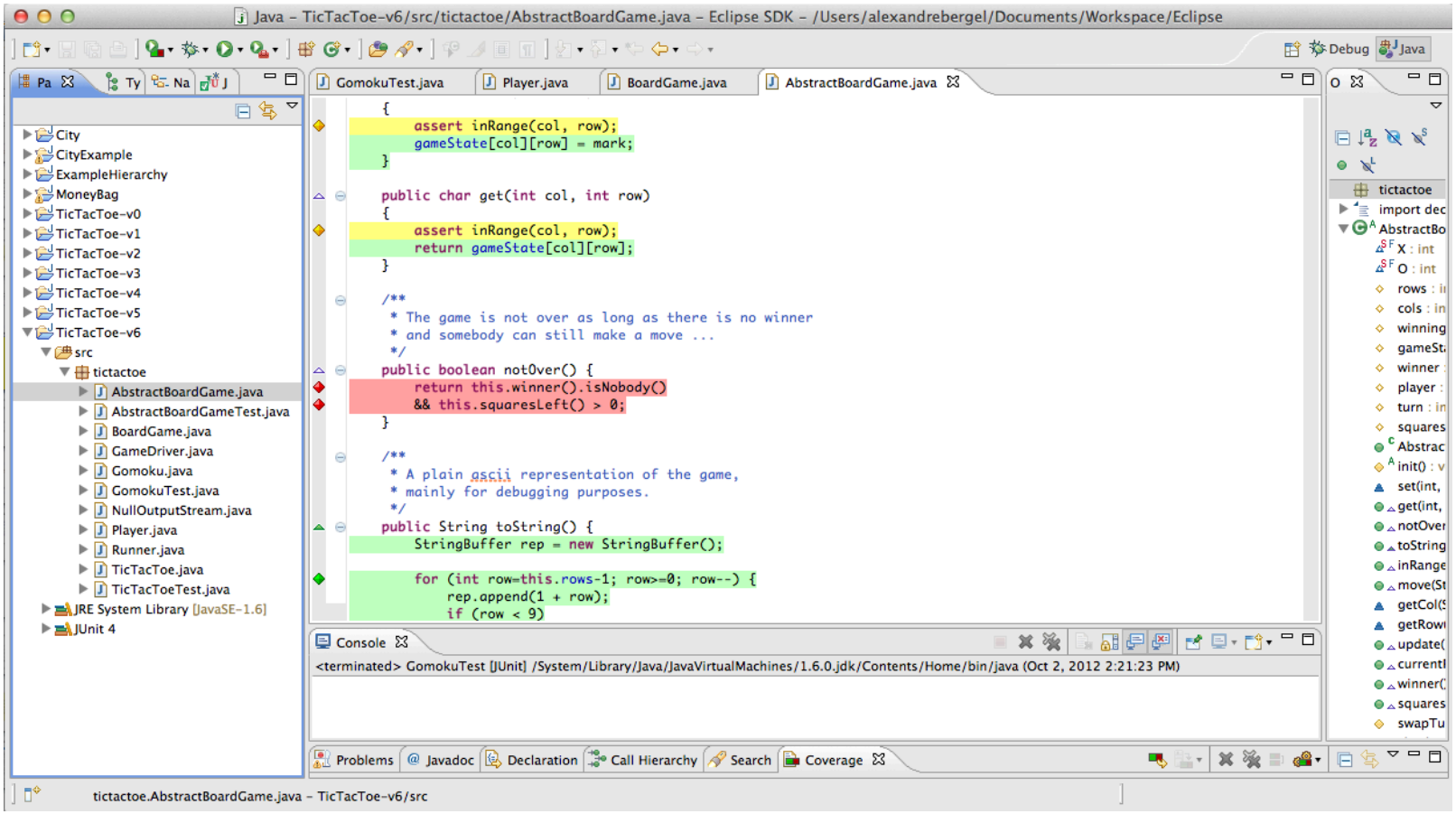
Higher a coverage, *better tested* is your software

EclEmma

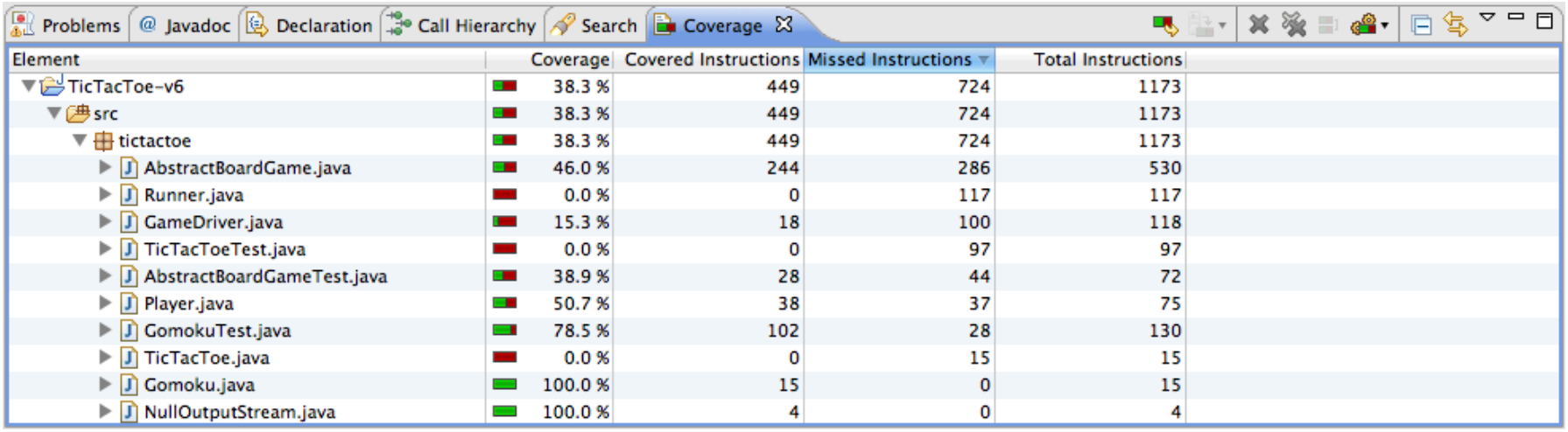
EclEmma is a free Java code coverage tool for Eclipse

Feel free to opt for a different tool coverage if you fan of other programming environment

www.eclemma.org



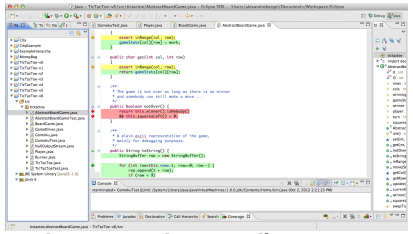
www.eclemma.org



Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
TicTacToe-v6	38.3 %	449	724	1173
src	38.3 %	449	724	1173
tictactoe	38.3 %	449	724	1173
AbstractBoardGame.java	46.0 %	244	286	530
Runner.java	0.0 %	0	117	117
GameDriver.java	15.3 %	18	100	118
TicTacToeTest.java	0.0 %	0	97	97
AbstractBoardGameTest.java	38.9 %	28	44	72
Player.java	50.7 %	38	37	75
GomokuTest.java	78.5 %	102	28	130
TicTacToe.java	0.0 %	0	15	15
Gomoku.java	100.0 %	15	0	15
NullOutputStream.java	100.0 %	4	0	4

Coloring source code

Source lines containing executable code get the following color code:



green for fully covered lines,

yellow for partly covered lines (some instructions or branches missed) and

red for lines that have not been executed at all.

The colors for the diamonds have a similar semantic than the line highlighting colors:

green for fully covered branches,

yellow for partly covered branches and

red when no branches in the particular line have been executed.

EclEmma

Installation

Follow “Option 3” in http://www.eclemma.org/installation.html#updatesite

You should then see the icon 

Test coverage

Test coverage is often used as a *quantitative measure of quality*

Determine the % of the program code is covered by the test

A software that is *well tested* is commonly associated with a test *coverage of 70%- 80%*

Test coverage

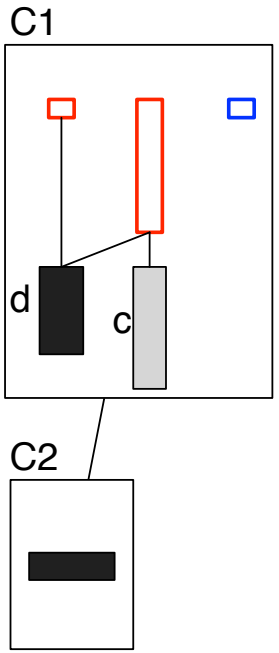
It has been recognized that *a coverage between 70 and 80%* results in a significant *decrease of bugs*

Audris Mockus, Nachiappan Nagappan, and Trung T. Dinh- Trong. Test coverage and post-verification defects: A multiple case study. In Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement, ESEM '09

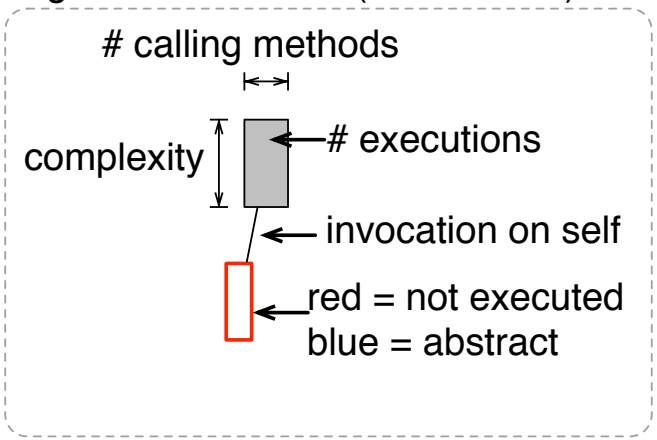
Paul Piwowarski, Mitsuru Ohba, and Joe Caruso. Coverage measurement experience during function test. In Proceedings of the 15th international conference on Software Engineering, ICSE '93

A coverage of over *80% is very hard to get*

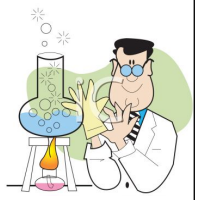
Test coverage with Hapao



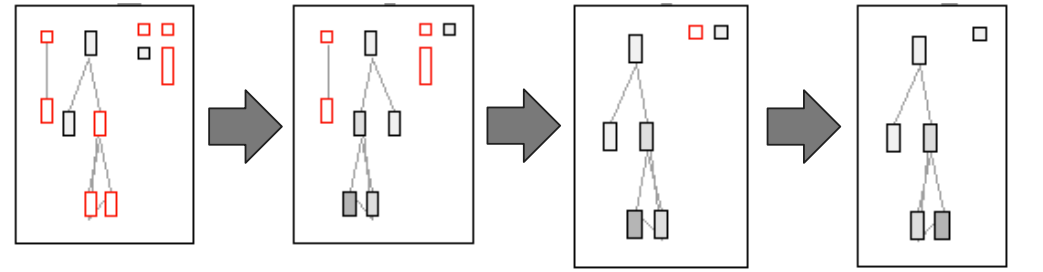
Legend for methods (inner boxes)



OBJECT PROFILE



Test coverage with Hapao



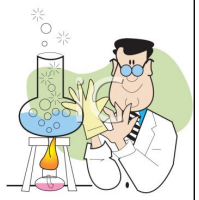
Version 2.2
27.27%

Version 2.3
54.54%

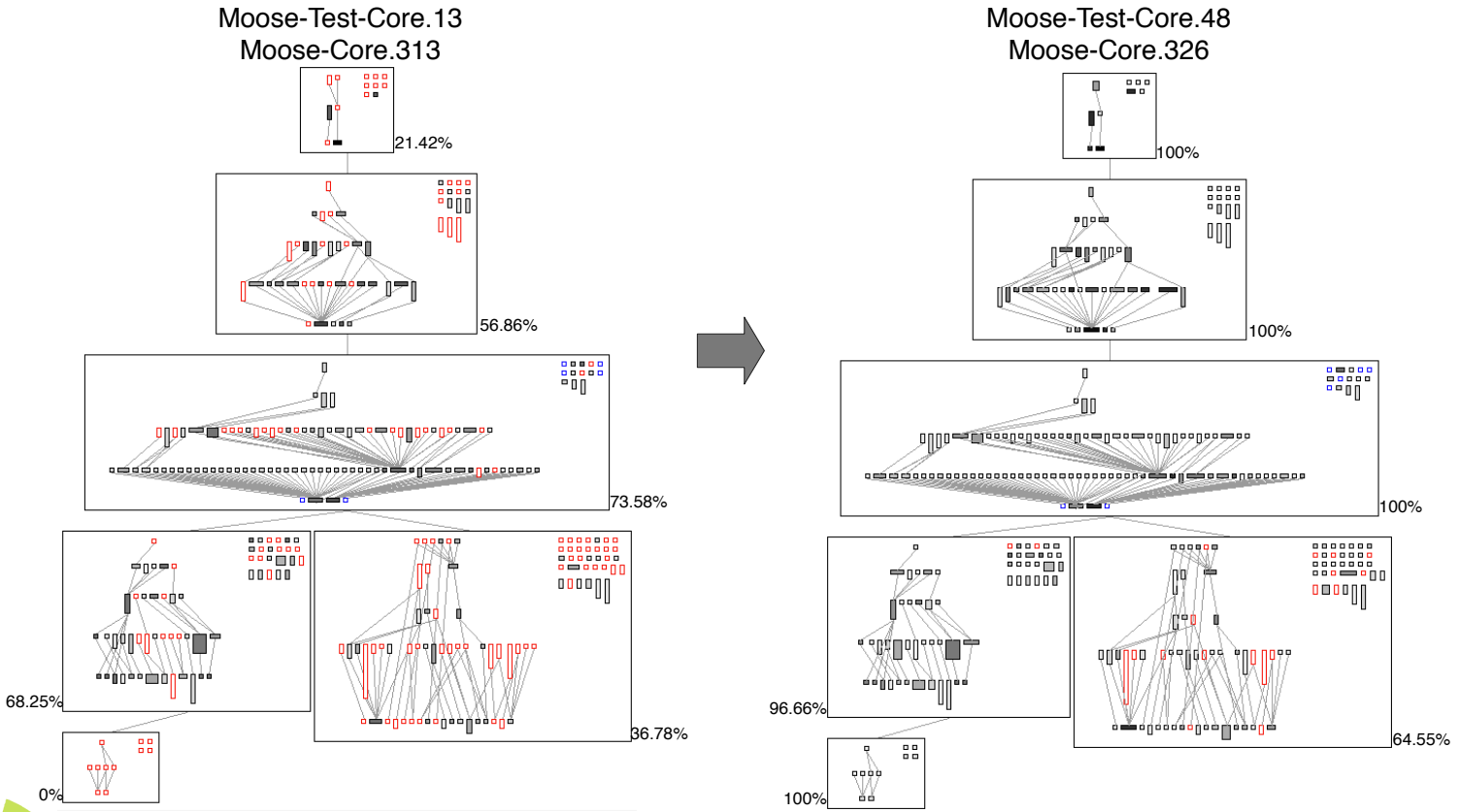
Version 2.4
87.71%

Version 2.5
100%

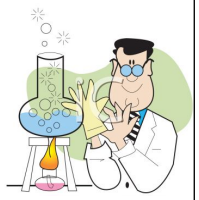
OBJECT
PROFILE



Test coverage with Hapao



OBJECT PROFILE



For your tarea 2 and 3

The quality of your test will be considered for your grade

You should reach a coverage of 70%

Do not write long method, this is harder to test

Branches, if-statement, loops are also harder to test