

Paper Title: Wodel-Test: a model-based framework for language-independent mutation testing

Paper doi: <https://doi.org/10.1007/s10270-020-00827-0>

Team members: Diaconu Ana-Maria, Duma Amalia-Diana, Drăghiciu Diana

Tool Description

Wodel-Test is a tool used for Java Mutation testing. This Mutation Tool parses the program representing it as a model conformant to the language meta-model, applies the mutation operators, and evaluates the test suite on the generated mutants, offering a **rich collection of metrics** about the MT process.

Installation

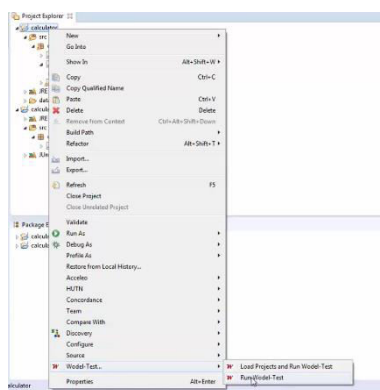
This tool was quite difficult to install since it is only available as an eclipse plugin, and we have encountered some errors during installation. After a few hours of research, we have finally managed to overcome those errors.

Project

The project the tool was used on is a calculator engine which can add, subtract, multiply and divide. The source code is from: <https://github.com/DimaSanKiev/Calculator>

Use of application

You can select the mutation operators you want to apply. There is a wide range of such operators: logic, relational, conditional, shift, assignment, arithmetic operator replacement and others.



After configuring everything it is quite simple to test. Simply right click on the project and run Wodel-Test command as shown in the image.

After generating the mutants models a score is obtained showing how many tests passed and how many failed (with a message of the reason it failed as can be seen on the right side of the table).

Filter	Results	Package/class/mutant	#Executed tests	#Failed tests	#Passed tests	Applied mutations/Failed test message
	Equivalent					
		testDivide	1	1	1	
		testMultiplyByZero	1	1	1	
		testAdd	1	1	1	
		testTestStringInput	1	1	1	
		testMultiply	1	1	1	expected<1.0> but was<100.0>
		testEqual	1	1	1	
		testMultiplyByZero	1	1	1	
		/calculator/asmdb/Output1/src/calculator/CalculatorEngine.java	6	3	5	modify information mutation: += replaced by /=
		/calculator/asmdb/Output2/src/calculator/CalculatorEngine.java	6	2	6	modify information mutation: += replaced by -=
		/calculator/asmdb/Output3/src/calculator/CalculatorEngine.java	6	3	5	modify information mutation: += replaced by -=
		/calculator/asmdb/Output4/src/calculator/CalculatorEngine.java	6	1	7	modify information mutation: += replaced by +=
		/calculator/asmdb/Output5/src/calculator/CalculatorEngine.java	6	2	6	modify information mutation: += replaced by +=
		/calculator/asmdb/Output6/src/calculator/CalculatorEngine.java	6	1	7	modify information mutation: += replaced by +=
		/calculator/asmdb/Output7/src/calculator/CalculatorEngine.java	6	1	7	modify information mutation: += replaced by +=
		/calculator/asmdb/Output8/src/calculator/CalculatorEngine.java	6	1	7	modify information mutation: += replaced by +=
		/calculator/asmdb/Output9/src/calculator/CalculatorEngine.java	6	3	5	modify information mutation: += replaced by +=
		/calculator/asmdb/Output10/src/calculator/CalculatorEngine.java	6	3	5	modify information mutation: += replaced by +=

Running time (s): 126

Mutation operators applied/not applied (9/34)

Mutants killed/equivalent/live (9/0/9)

Mutants killed/live (9/9)

Tests failed/passed (5/3)

% Mutation operators applied 26.92% % Mutation operators not applied 73.07%

% Mutants killed 50% % Mutants live 50%

Mutation score 50%

% Tests failed 62.5% % Tests passed 37.5%

A nice feature is that you can filter these tests and select to only see the ones that failed or passed.

Filter	Results	Test suite/Test case	#Killed mutants/Message	Applied mutations
Failed		/calculator-test/src/calculator/CalculatorEngineTest.java	10	
		testSubtract	2	
		testDivide	4	
		testMultiplyByZero	2	
		/calculator/asmdb/Output0/src/calculator/CalculatorEngine.java	2	expected<-10.0> but was<0.0>
		/calculator/asmdb/Output1/src/calculator/CalculatorEngine.java	2	expected<-10.0> but was<0.0>
		testAdd	3	
		testMultiply	7	

Running time (s): 126

Mutation operators applied/not applied (9/34)

Mutants killed/equivalent/live (9/0/9)

Mutants killed/live (9/9)

Tests failed/passed (5/3)

% Mutation operators applied 26.92% % Mutation operators not applied 73.07%

% Mutants killed 50% % Mutants live 50%

Mutation score 50%

% Tests failed 62.5% % Tests passed 37.5%