Augmenting Test Suites Automatically

Konstantin Rubinov¹

Jochen Wuttke²

¹ University of Lugano – Switzerland
² University of Washington, Seattle – USA

Problem & Status
Test cases of different kind and granularity are important to test SW systems thoroughly.
Generating complex test cases is expensive.
Modern applications are provided with many unit test cases that contain useful information.
Complex test cases are difficult to build.

Solution
Reuse information from existing test cases to automatically generate complex test cases with longer and more complex execution scenarios that promote unit interactions.

From Unit to Integration Test Cases

Step 1: Identify class dependencies for the system under test based on the interfaces of class constructors.

Step 2: Extract instantiation and execution sequences from existing test cases using data-flow analysis.

Step 3: Manipulate and combine instantiation and execution sequences using integration strategies.

Prototype

Eclipse Plug-in based on the Eclipse JDT program manipulation services works with Java and JUnit test cases.

Sequence manipulations:
(1) Extend object instantiation and execution sequences by appending compatible execution sequences.
(2) Extend execution sequence by merging several execution sequences for dependent objects.
(3) Replicate and append fragments of execution sequences iteratively.

JUnit

Experimental results

TestabilityExplorer
82 test cases

JGraphT
64 test cases

More than 60% of test cases execute immediately.

Improvement in statement coverage over the developers’ test suites.

Complex interaction patterns not exercised by the developers’ test suites.

Contact: konstantin.rubinov@usi.ch