



Model Based testing of VDM models

Uwe Schulze

`uschulze@informatik.uni-bremen.de`

Universität Bremen

11th Overture Workshop on VDM

AUG 28, 2013

Overview

Motivation

Architecture

Signal Mapping

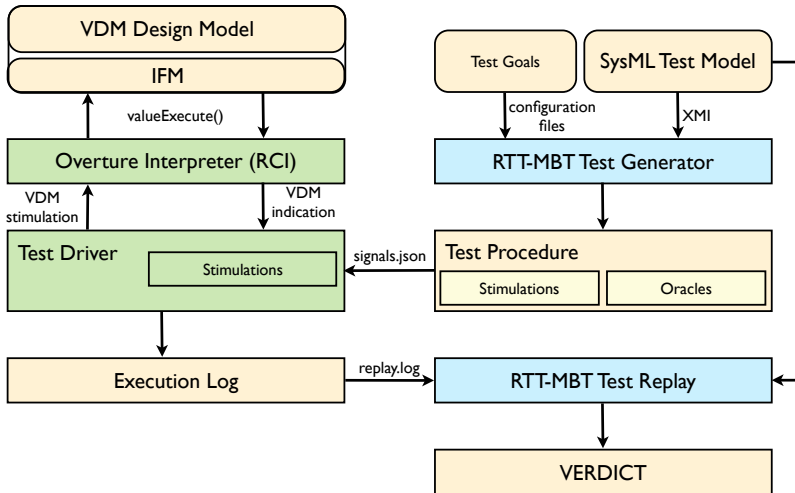
Test Execution

Test Evaluation

Conclusion and future work

Motivation

- ▶ Finding failures early saves resources
- ▶ Problems when defining test procedures during system design
 - ▶ Abstract and informal test procedure definitions cannot be evaluated/executed
 - ▶ Test procedures cannot be reused in later test integration levels
- ▶ Benefits of testing a design models
 - ▶ Uncover testability problems of the design
 - ▶ Signal/interface tracing from specification to design model
 - ▶ Define Requirements tracing
 - ▶ Validate Test procedure functionality, and strength



Architecture

- ▶ RTT-MBT test generation from UML/SysML models
 - ▶ Developed for test generation from deterministic models
 - ▶ Used in software integration and HIL test projects
 - ▶ Test procedure contains stimulations and checkers
- ▶ Model-in-the-loop test of VDM model
- ▶ Test driver uses stimulations from the RTT-MBT test generation
- ▶ Off-line test evaluation by interpretation of the test model
- ▶ Signal mapping in interface module inside the VDM model
- ▶ Timing requirements can only be tested with the usual restrictions from software tests

Signal Mapping

- ▶ General problem in all test integration levels
- ▶ Signals must be traceable from high level specifications to low level specifications and implementation
- ▶ Can require protocols and complex data transformation
- ▶ Can effect test procedure design
- ▶ Should be separated from the application and test logic

Interface Modules

- ▶ Separates signal mapping from application logic
- ▶ Central place for signal mapping (supports verification)
- ▶ Can be used to adjust test procedures to different test integration levels
- ▶ Implemented as part of the VDM model that can be attached for testing
- ▶ Generic interface to test driver
- ▶ IFM provides copy of test model signals as attribute.
- ▶ Refinement of the VDM model can require adjustment of the interface module
- ▶ Test model and test goals / generated test procedures can be re-used

Test Execution

```
void testExecution() {
    long timestamp = 0;
    while (timestamp > 0) {

        // assign stimulations to TE2SUT signals in the IFM
        List<String> stimulations = getStimulations(timestamp);
        for (int idx = 0; idx < stimulations.size(); idx++) {
            // stimulations are of the form "setTE2SUTsignalName(value)"
            interpreter.execute("ifm." + stimulations[idx]);
        }
        logSutInputs(stimulations);

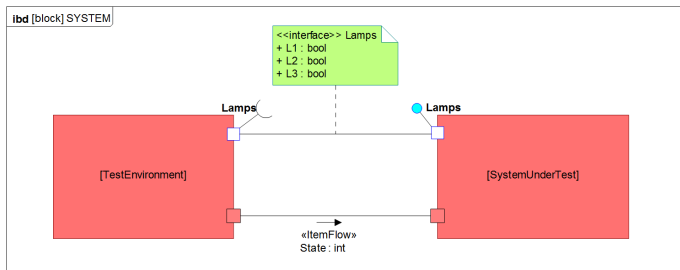
        // perform stimulation with new input signals
        interpreter.execute("ifm.performStimulation()");

        // retrieve and log changed states of the SUT
        Value v = interpreter.valueExecute("ifm.getChangedOutputs()");
        logSutOutputs(v);

        // calculate next timestamp (and sleep until then)
        timestamp = getNextTimestamp();
    }
}
```


Test Execution

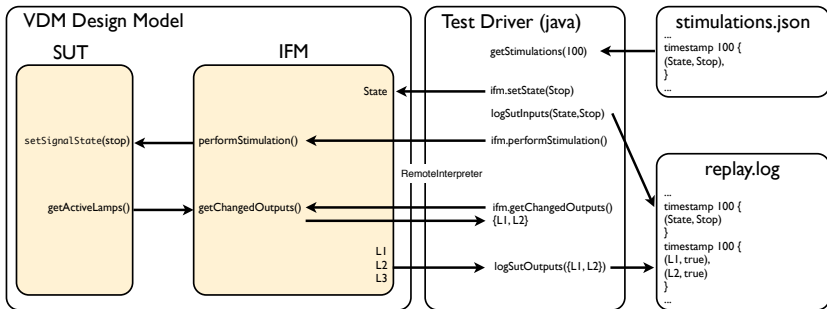
SysML test model



VDM model SUT

- ▶ `LampId = <L1> | <L2> | <L3>;`
- ▶ `Signal = set of LampId;`
- ▶ `public setSignalState : Signal ==> ()`
- ▶ `public getActiveLamps : () ==> Signal`

Test Driver



Test Evaluation

- ▶ Test case coverage and verdict is calculated by interpreting the test model
- ▶ Off-line evaluation - no stop on fail
- ▶ Benefits of off-line evaluation
 - ▶ no delay for verdict calculation during the test
 - ▶ no communication delays during runtime between tools

Replay

- ▶ Uses the same stimulations as the test to interpret the test model
- ▶ Compares the expected outputs from the test model interpretation against the test log
- ▶ Enables tracing of internal model states (model must be deterministic)
- ▶ Result is a list of covered model test cases with verdict PASS or FAIL

Conclusion

- ▶ Generic framework and test driver for model-in-the-loop tests of VDM models
- ▶ Supports requirements tracing
- ▶ Supports development, execution and evaluation of test procedures at design time of a system
- ▶ The test model and generated test procedures can be re-used for
 - ▶ different abstraction levels of the design model
 - ▶ different test integration levels

Future Work

- ▶ Integration into RTT-MBT plug-in / COMPASS tool
- ▶ Extend test driver for output triggered stimulations
- ▶ Tool support for IFM / signal mapping definition
- ▶ Test the approach with timed VDM models
- ▶ Test the UML-VDM mapper

Discussion

Thank you for your attention!

Example

