A model-based testing solution based on a symbolic technology

Julien Deltour
PragmaDev
PragmaDev

- French SME,
- Created in 2001 by 2 experts in modelling tools and languages.
- Dedicated to the development of a modelling and testing tool for the development of **Event driven software**.

**Aero/Defence**
- AIRBUS
- ESA
- DGA
- Euronews

**Automotive**
- RENAULT

**Telecoms**
- Alcatel-Lucent
- Aeroflex
- ERCOM
- WIPRO
- NewLogic
- LG Electronics
- THOMSON images & beyond
- KETI

**Semi-conductor**
- ST Ericsson
- TOSHIBA
- MITSUMI

700 active university licenses around the world
Several Collaborative Projects with big accounts

- **Alcatel-Lucent**
  - Focus on Model Checking
  - Started in 2005, finished in 2009

- **Thales**
  - Focus on property verification
  - Started in 2012, finished in 2014

- **CEA- LIST**
  - Focus on Model Based Testing
  - Started in 2013
Requirements for a good modelling language

- The abstract model must be platform independent, as its name states.
- The abstract model must be translatable to an execution platform.
- For that purpose, the abstract model is based on a virtual machine offering:
  - Some **basic services**.
  - An execution **semantic**.

**SDL** international standard is the best candidate to model event driven systems.

Key features for Model Based Testing capabilities
Verify the model

Since the model is executable, it is possible to simulate it in order to verify it is correct.
Requirements for a good testing language

- Relies on the same basic services as SDL:
  - Messages
  - Procedures
  - Timers
  - Parallel execution

- TTCN-3 international standard:
  - Data types definitions or ASN.1,
  - Templates definitions,
  - Test cases,
  - Verdict,
  - Execution control.
Same level of abstraction

- Specification
- Design
- Implementation
- Simulation
- Execution
- Validation testing
- Integration testing
- Unit testing
Model analysis technologies

• Partnership with specialized labs:
  • Exhaustive simulation,
  • **Symbolic resolution**.
• Properties:
  • **Model coverage**,  
  • Static or dynamic property: 
    • Property verification,  
    • Test objectives.
Reference testing

Reference Model

Simulation Execution

Traces

Requirements

Conformance Tests

Test objectives

Coverage

Result of PragmaList
CEA – A major European RTO

» 16 000 people
» 10 centers in France
» Budget: 4.3€ billions
» 1 600 patents
» 4 000 publications/year
» 150 startup created since 1984
INSTITUTES

**leti**
Laboratoire d'**Electronique et des Technologies de l'Information**
1967 - Grenoble
1800 pers.

**list**
Laboratoire d’**Intégration des Systèmes et des Technologies**
2003 - Paris Sud
700 pers.

**liten**
Laboratoire d’**Innovation pour les Technologies des Énergies nouvelles et les Nanomatériaux**
2005 - Grenoble / Chambéry
1100 pers.
CEA LIST R&D PROGRAMMES

ADVANCED MANUFACTURING
Systems for industry
- Robotics
- Virtual reality
- Non destructive testing
- Vision

EMBEDDED SYSTEMS
- Software engineering
- Safety & security
- Computing architectures
- Communication and interfaces

AMBIENT INTELLIGENCE
Sensing systems and big data
- Sensors, instrumentation
- Metrology
- Big data and multimedia

SYSTEMS OF SYSTEMS
Diversity principle

Model:
- Several execution semantics: Synchronous / Asynchronous
  State machine / Dataflow
- Several communication semantics: Rendez vous / FIFO / …

Coverage criteria:
- states / transitions
- MC/DC

Structural constraints:
- nb of tests,
- size of a test

DIVERSITY - xLIA

Test cases

Coverage information
Diversity kernel

Symbolic simulation of the model:

- Defines **symbolic behaviours**, i.e. **equivalence classes** of numerical behaviours of the system.
- Represented as a tree.
- Each path = a distinct symbolic behaviour.
- Random choice of a numerical behaviour for each equivalence class → **Test Case**

Numerical Test Case
Diversity outputs

Generate a set of scenarios (i.e. test cases) \textit{wrt} a specific objective.

This set is reduced with regard to redundancy.

Moreover, during the analysis phase, the tool can detect:

- \textit{inconsistancies} among data types,
- \textit{dead locks},
- \textit{dead parts} of the model,
- ...
The project in four steps.

- **Step 1 : SDL to xLIA translation rules :**
  - Write the translation rules to convert SDL to xLIA.

- **Step 2 : SDL to xLIA translator :**
  - Write the xLIA generator from an SDL model.

- **Step 3 : Diversity adaptation to support SDL semantic :**
  - Work on SDL communication semantic,
  - Work on SDL timer semantic.

- **Step 4 : TTCN-3 formats output generation :**
  - TTCN-3 test cases formatting to be supported by RTDS.

xLIA is the CEA List Diversity file format to describe the model.
Architecture

Model
SDL
Observer

Property

Test case
TTCN

PragmaDev RTDS

translate to file

xLIA
Resolution Exploration
Resulting scenarios

CEA List Diversity
Four types of targets

- **Code coverage:**
  - To generate the minimum number of test cases that cover all transitions.

- **Transition:**
  - To generate a test case that covers a specific transition in the SDL model.

- **Property:**
  - To generate the test cases verifying a static property (process state, variable value, …).

- **Observer:**
  - To generate the test cases verifying a dynamic property (succession of action or temporal rules). A dynamic property is defined as a state machine called observer.
Demonstration

An Access Control System:
- 2 state machines
- A card input with a 0..65535 integer as parameter
- A key input with a 0..11 integer as parameter
Test cases are automatically generated

A Test manager helps to select the test cases

Coverage information shows full coverage
CEA List - Diversity

- Exploration time is always the same (10 seconds) whatever are the message parameter ranges.

Verimag - IF toolbox

- Exhaustive exploration
- Exploration time depends on message parameter range.

<table>
<thead>
<tr>
<th>Card range</th>
<th>Digit range</th>
<th>0..1</th>
<th>0..2</th>
<th>0..3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0..1</td>
<td>13</td>
<td>126</td>
<td>721</td>
<td></td>
</tr>
<tr>
<td>0..2</td>
<td>38</td>
<td>316</td>
<td>2169</td>
<td></td>
</tr>
<tr>
<td>0..3</td>
<td>64</td>
<td>650</td>
<td>28234</td>
<td></td>
</tr>
</tbody>
</table>
On-going use cases

- SNCF: Radio Block Center (RBC)
- Alstom Belgium: Radio Gateway
- Alstom France: Passenger exchange
- Airbus: Air Traffic Control (ATC)
- Other: Secure transactions
Model Based Testing solution

- Integrated tool chain
- Non dedicated model
- Efficient symbolic kernel
  - Test automation
  - Reduce the number of test cases
  - Early in the development process