Model-based testing for component-based applications
Presented by Wenbin Li, Easy Global Market, France
Outline

- Easy Global Market
- H2020 PHANTOM
- MBT in PHANTOM
- Testing Automation
Easy Global Market

**R&D SUPPORT**
Involved in +30 EU projects since 2001

**ADVANCED TESTING**
State of the Art, advanced testing methods (eg MBT, TTCN3)

**PLatforms INTEGRATION**
Mastering Future Internet (FIWARE) and IoT platforms (eg OneM2M)

**INTEROPerability EXPERTS**
World specialists in technical and semantics interoperability programs
H2020 PHANTOM
Multi-Objective Programming for Parallel Computing System
H2020 PHANTOM: Use Case Applications

Airflow Simulation

Telecom Network Management

Surveillance Image Processing
H2020 PHANTOM: Component-based applications

Telecom Network Management

Surveillance Image Processing
H2020 PHANTOM: Component Network

Application and components

Inputs

C1

C2

C3

C4

Outputs

C5

User Conference on
Advanced Automated Testing
Model Based Testing in PHANTOM

MBT: testing based on or involving models

- Early Validation
  - Model validation
  - Performance Estimation

- Test Execution
  - Functional Testing
  - Non-functional Testing

Legends

1. Early Validation
   1. Model Validation
   2. Performance Estimation

II. Test Execution
   3. Functional Testing
   4. Non-Functional Testing

PHANTOM stages

Development and Preparation Stage ➔ Execution Stage
H2020 PHANTOM: Model Validation
H2020 PHANTOM: Model Validation

@machine:

stateMachine or > HPCApplicationMachine
{
    @private:
    port input RequestExecute(CommandLine, Experiment, integer, VariatingParameter);
    port output ResponseExecutionResult(CommandLineResult);
    port output CheckResult(ExperimentResult);
    port input ResponseEndExecution(VariatingParameter);

    state(initial) State_Initial_Non_Functionnal
    {
        transition StartExperiment1 --> State_TimeStampModulation
        {
            TimeStamp = TimeStamp;
            vCurrentExperiment = EXPERIMENT_1;
        }
    }

    transition StartExperiment2 --> State_TimeStampModulation
    {
        TimeStamp = TimeStamp;
        vCurrentExperiment = EXPERIMENT_2;
    }

    transition StartExperiment3 --> State_TimeStampModulation
    {
        TimeStamp = TimeStamp;
        vCurrentExperiment = EXPERIMENT_3;
    }
}

Communicating Finite State Machine in xLia

DIVERSITY

Model Validation Result

REPORT
STOP CRITERIA: PROCESSOR
The CONTEXT count : 8532
The STEP count : 2190

The Max HEIGHT reaching : 19
The Max WIDTH reaching : 6011
The RUNEXIT count : 6858
TRANSITION COVERAGE: PROCESSOR
All the << 58 >> transitions are covered!
Number of nodes cut back: 8587

step: 0, context:100, height:100, width:100
step: 0, context:18532, height: 1, width: 1
-- 0ms @ Mon Jul 30 12:50:52 2018

REPORT
STOP CRITERIA: PROCESSOR
The CONTEXT count : 8532
The STEP count : 0

The Max HEIGHT reaching : 1
The Max WIDTH reaching : 1
EXECUTION CHAIN
TRACE GENERATOR
The TRACE count : 29
DONE!
H2020 PHANTOM: Performance Estimation
H2020 PHANTOM: Performance Estimation

Application 1

- C1
- C2

Application 2

- C1
- C2

Testing results

<table>
<thead>
<tr>
<th>Object</th>
<th>Execution Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>100 ms</td>
</tr>
<tr>
<td>C2</td>
<td>20 ms</td>
</tr>
<tr>
<td>Application 1</td>
<td>120 ms</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Estimation Methods for Composition Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sequence</td>
</tr>
<tr>
<td>Execution Time (ET)</td>
<td></td>
</tr>
<tr>
<td>RAM Usage (RU)</td>
<td></td>
</tr>
<tr>
<td>Reliability (RE)</td>
<td></td>
</tr>
<tr>
<td>Energy Consumption (EC)</td>
<td></td>
</tr>
</tbody>
</table>

- Execution time = 140ms
H2020 PHANTOM: Functional & Non-functional testing
Testing Automation

- Model validation
  - Require automation of model creation
- Performance Estimation (Automated Process)
- Functional Testing
  - Require automation of model creation
- Non-functional Testing
  - Require automation of model creation
Testing Automation: Model Creation

- for new applications that reuse tested components, e.g., airflow simulation use case
Testing Results

```
artemis@artemis-EGM:~/git/titan_phantom/MRT:Titan/intec_v2

MTC@artemis-EGM: Default with ld 1 (altstep R/TDS_fall) was deactivated.
MTC@artemis-EGM: Stop timer t_phantom: 100 s
MTC@artemis-EGM: Port ePortGeneric_Port was stopped.
MTC@artemis-EGM: Component type TTCN_Component was shut down inside testcase TestCase_71.
MTC@artemis-EGM: Waiting for PTCs to finish.
MTC@artemis-EGM: Setting final verdict of the test case.
MTC@artemis-EGM: Local verdict of MCT: pass
MTC@artemis-EGM: No PTCs were created.
MTC@artemis-EGM: Test case TestCase_71 finished. Verdict: pass
MTC@artemis-EGM: Execution of control part in module TTCN_TestSuite finished.
MTC@artemis-EGM: Test execution finished.
MTC@artemis-EGM: Execution of [EXECUTE] section finished.
MTC@artemis-EGM: Terminating MCT.
MTC@artemis-EGM: Verdict statistics: 0 none (0.00 %), 57 pass (80.28 %), 0 inconc (0.00 %), 14 fall (19.72 %), 0 error (0.00 %).
MTC@artemis-EGM: Test execution summary: 71 test cases were executed. Overall verdict: fall
MTC@artemis-EGM: Exit was requested from MCT. Terminating MCT.
MTC@artemis-EGM: MCT terminated.
MCT> exit
MTC@artemis-EGM: Shutting down session.
MTC@artemis-EGM: Exit was requested from MCT. Terminating MCT.
MTC@artemis-EGM: Shutdown complete.
```
Conclusion and Future Work

- **Early testing** in parallel with application development
- **Thorough test execution** for functional and non-functional aspects
- **End-to-end test automation** for component-based applications

- Offline testing solution,
  To combine testing automation with online testing
- Rely on previously modeled/tested components,
  To consider/tolerate non-tested components in automation
Thank you!

Dr. Wenbin Li
Research Engineer

wenbin.li@eglobalmark.com