Software and System Engineering for Cyber-Physical Systems: technical challenges and collaboration opportunities

Model-Based Testing for Internet of Things and Cyber-Physical Systems

Fabien PEUREUX

IRIT, Toulouse, 03/02/2016
Brief introduction about the organisations

- Département d’Informatique des Systèmes Complexes (Université de Franche-Comté – Model-Based Testing)
  http://www.femto-st.fr/
  contact: fabien.peureux@femto-st.fr

- Easy Global Market
  (SME – standardization & solutions for IoT interoperability)
  http://www.eglobalmark.com/
  contact: franck.le-gall@eglobalmark.com

- Smartesting Solutions & Services
  (SME – Model-Based Testing solution provider)
  http://www.smartesting.com/
  contact: bruno.legeard@smartesting.com
Presentation outline

- Model-Based Testing overview
- CertifyIt MBT process & tooling
- Ongoing work within IoT
- Ongoing work within CPS
Functional needs
Business needs
Requirements

Test design and implementation

Test Repository (Excel, HP/ALM...)

Functional tests

Manual execution & scripts for automation
Model-Based Testing positioning

- Functional needs
- Business needs
- Requirements

Test Repository (Excel, HP/ALM...)

Certify'lt®

Functional tests

Manual execution

& scripts for automation
The CertifyIt Solution

Functional needs
Business needs
Requirements

Modeling for test generation
Functional flows
Business rules/data + Strategy

Automatic test generation

Test Repository
(Excel, HP/ALM...)

Functional tests

Manual Execution
& scripts for automation

Business needs

The CertifyIt Solution
The CertifyIt Solution

Functional tests
Manual execution & scripts for automation

Modeling for test generation
Functional flows
Business rules/data + Strategy

Test Repository (Excel, HP/ALM...)

Business needs
Requirements

Automatic test generation

Traceability

The CertifyIt Solution
Typical MBT process tooled with CertifyIt

- **Requirement Management**
  - Requirements
  - Test objectives

- **Test Design**
  - Test models
  - generated tests

- **Test Management & Execution**
  - Requirement links
  - Test scripts
  - Adaptation layer
  - Keywords library

- **Iterative Process**

Keywords library
Test model using CertifyIt

Business Process Model (BPMN)

Business Entities and Logical Test Data (UML)

Business Rules and Behavioral Model (UML/OCL)

Modeling notations
Test case derivation principles
A **test** is a **sequence of operations**, that brings the System Under Test from its **initial state** to a state where the expected behavior can be **activated**, then activate it.

**Test context creation**

- **Initial State**
  - Operation1\((x, 1)\)
  - State 1
    - Operation3\((val1, y)\)
    - State n
      - OperationX\((val4)\)
  - State n+1
Test cases results in CertifyIt
Test execution results
• FIWARE is a future internet platform for IoT
• Generic Enabler (GE) = API
• Fiware GE’s are general purpose
• Open source implementation
• RESTful systems

• Building confidence in FIWARE leads to testing challenges
- Smarttesting MBT
- Generation Tool: **CertifyIt**
- MBT Process:
  1. Define the test objectives
  2. Generate abstract test cases
  3. Test repository publication
  4. Tests execution
FIWARE IoT Case-Study: EspR4FastData

- Real-time data handling GE
- Filter, merge and aggregate real-time data from different sources
- Complex Event Processing (CEP)

<table>
<thead>
<tr>
<th>@REQ</th>
<th>Requirement description</th>
<th>@AIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESET_GE</td>
<td>This operation completely destroy the CEP threads, the CEP data, and all the application level data. It can be used whether the application is running or not. It leaves the application clean and ready to start. EXAMPLE: <a href="http://localhost/EspR4FastData-3.3.3/admin/espr4fastdata">http://localhost/EspR4FastData-3.3.3/admin/espr4fastdata</a> (queried with HTTP Delete method)</td>
<td>Success</td>
</tr>
<tr>
<td>REGISTER_CONTEXT</td>
<td>This operation registers a NGSI context within EspR4FastData. This makes an input event type and related entities to be known from an EspR4FastData perspective. An event is characterized by the involved entities (eg: Van1, Van2, Plane), its type, and properties that related to the type. For an event to be processed, it is mandatory to be &quot;known&quot; by the application. Otherwise it would be rejected.</td>
<td>Success and is Empty</td>
</tr>
</tbody>
</table>

NGSI-9/10 compliant:
- Context registration
- Context query
- Context publish/subscribe
- …
EspR4FastData MBT modeling

UML class diagram + OCL

+ UML object diagram
EspR4FastData MBT Test Generation

CertifyIt
EspR4FastData Test Concretization & Execution

• HTML export: Abstract Test Cases

<table>
<thead>
<tr>
<th>Steps</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>resetCE</td>
</tr>
</tbody>
</table>

This operation completely destroy the CEP threads, the CEP data, and all the application level data. EXAMPLE: http://localhost/Espr4FastData-3.3.3/admin/espr4fastdata (queried with HTTP Delete method)

- @synthesis@ Check that the error code is URL_ERROR
- @/synthesis@
- Check if an error code URL_ERROR is returned

<table>
<thead>
<tr>
<th>Requirements, aims and custom tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>REQ resetEspR4FastData</td>
</tr>
<tr>
<td>AIM resetEspR4FastData/Invalid url</td>
</tr>
</tbody>
</table>

• SoapUI Export ➔ Executable script

- Adapt with mapping file (abstract to concrete)
- Import to SoapUI and launch tests
H2020 U-Test project (2015-18):
Testing Cyber-Physical Systems under Uncertainty

http://www.u-test.eu/

MBT: to provide a framework to manage uncertainty models based on UML using the Eclipse Modeling Framework:

- **ModelBus** (provided by FOKUS), a model-driven tool integration framework, which allows definition of a seamlessly integrated tool environment for system engineering processes.
- **Fokus!MBT** (provided by FOKUS), supporting Modelling and Model-Based Testing approaches from UML MBT models
- **CertifyIt** (provided by EGM), supporting Model-Based Testing approach from UML/OCL MBT models
U-Test Challenges regarding MBT

- To adapt / extend current MBT process to address CPS domain and technologies
- Integration of the uncertainty taxonomy into modeling artefacts
- Traceability of the uncertainty attributes from the test model to the generated test cases
- Improvements and integration of test generation algorithms dedicated to CPS and uncertainty
Expected MBT process within U-Test project

Functional needs
Business needs
Requirements

Requirements with Uncertainties

Modeling for test generation with uncertainties

Automated test generation

Test Repository (Excel, HP/ALM…)

Test Cases and Test Scripts

Manual Execution & scripts for automation

Traceability
## Summary

### MBT for IoT:
- Based on UML/OCL models
- First encouraging results with FIWARE specifications
- Experiments starting with OneM2M specifications
- Next results expected within H2020 ARMOUR project

### MBT for CPS:
- Starting work within H2020 U-Test project
- Based on UML/OCL models including uncertainties
- Test generation driven by uncertainty coverage criteria
Summary

- **MBT for IoT:**
  - Thèse CIFRE – EGM/Femto-st (2015-18)

- **MBT for CPS (uncertainty)**
  - ARMOU (1/02/2016-18) – EGM/Smartesting (Large-Scale Experiments of IoT Security Trust)

- **U-Test (2015-18) – EGM**
  - [http://www.u-test.eu/](http://www.u-test.eu/)
Thank you for your attention

Questions/Comments?

“Testing is always model-based!”
Robert Binder