Introduction

Matthias Rasking

- Senior Principal, Accenture GmbH, Germany
- EALA Platform Lead for Accenture Application Test Services
- Owner Accenture Delivery Methods for Testing and ADM for Testing Estimator
- Deputy Chair for Model Development and Model Maintenance for the TMMi Foundation, TMMi Lead Assessor

Vinit Shah

- Sr. Automation Engineer, Accenture with hands on experience in a number of tools, including Tosca.
- Managing automation teams at offshore in India and implementing new testing processes.
- Core competence is in software testing of Client-Server based applications using Manual and Automated Testing Techniques, currently supporting a German insurance client of Accenture on-site.

Copyright © 2014 Accenture All Rights Reserved.
Agenda

- Introduction
- Software testing approaches
- Introduction to Model Based Testing (MBT)
- MBT: Advantages and Disadvantages
Test Automation evolution (Tricentis)

1. Generation
- Record/Replay
- FRAGILE

2. Generation
- Test Automation Frameworks
- COSTLY

3. Generation
- Model-Based
- Robust
- Cost Effective
- Scalable
What does MBT promise?

<table>
<thead>
<tr>
<th>Quality</th>
<th>Spec Defect Detection Rate – Test Design</th>
<th>SW Defect Detection Rate – Test Execution</th>
<th>Test Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Your System</td>
<td>![arrow up]</td>
<td>![arrow up]</td>
<td>![arrow up]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Effort</th>
<th>Test Design effort</th>
<th>Test Execution Effort</th>
<th>Effort to train testers</th>
<th>Maintenance effort</th>
</tr>
</thead>
<tbody>
<tr>
<td>Your System</td>
<td>![arrow down]</td>
<td>![arrow down]</td>
<td>Same as for traditional Testing</td>
<td>![arrow down]</td>
</tr>
</tbody>
</table>
Model Based Testing (MBT)

**Definition**

Model-based testing is *software testing* in which *test cases* are derived in whole or in part from a *model* that describes some (usually functional) aspects of the *system under test* (SUT).

**Types of MBT**

- Graphical (UML, BPMN…)
- Textual (TTCN, TDL, ExTRA…)
- Technical (System model)
Model Based Testing - Graphical

Requirements Documents → Develop Abstract Model

Model(s) for System Behavior

Test Generation Tool Generates Test Plans and Keywords for Test Automation

Test Plans in HP QC / ALM

Define Test Objectives and Acceptance Criteria

Functional Analyst confirms
- Scope coverage
- Critical path
- Risk-based priorities

Modeling Engineer confirms
- Proper level of abstraction
- Conflicts in models
- Optimized test design

Simplified Business Process Model (BPMN2.0) or Use Case description with decision tables
Model Based Testing - Textual

System Under Test

Modeler

Test Models

Add input and verification data
Automated or Manual

Test Plans

- Model Based Testing
- Textual

- Formalistic Description
  - Domain Specific Language
    - Enables generator to create tests
    - Requires Functional Analysts to adhere to pseudo-code
  - Test Engineer
    - Adds test data to functional model
    - Implements the generated keywords in the automation tool

- Machine-generated tests from DSL

  ```
  fachentitaet Mitarbeiter (beschreibung "Stammdaten eines Mitarbeiters")
  attribute
  id ganzeZahl (primaeer pflicht beschreibung "Interne Datensatznummer (Primärschlüssel)")
  anrede SAL (beschreibung "Anrede des Mitarbeiters")

  maske NachrichtVerfassen
  (titel "Nachricht verfassen")
  abschnitt IhreNachricht
  (titel "Ihre Nachricht")
  daten Empfaenger $Kunde.name
  (beschreibung "Empfänger" status anzeigend)
  daten Betreff $Nachricht.nachrichtenBetreff
  (beschreibung "Betreff")
  ```

Copyright © 2014 Accenture All Rights Reserved.
Model Based Testing - Technical

System Under Test

Tosca Wizard

Add input and verification data Automated or Manual

Test Plans

Modules

Application Scan Result

Execution Lists

Automated Scans

Functional Analyst confirms
  • Application areas to be included
  • Inherent value of test cases

Tosca Engineer confirms
  • Coverage of UI and non-UI controls
  • Module completeness

Functional Test Analysts use modules to define execution lists
Requirement: “The login screen asks for username and password. Successful login requires, that the password matches the one stored in the database for the account identified by the given username.”
### MBT – Comparison with Traditional Approach

#### Test Development Phase

<table>
<thead>
<tr>
<th>Traditional testing Approach</th>
<th>MBT Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓ Define the Business Requirements  ✓ Define the Test strategy</td>
<td>✓ Make a model of the application  ✓ Get the test plans and scripts automatically generated by the tool</td>
</tr>
<tr>
<td>✓ Prepare test plans  ✓ Prepare test Scripts</td>
<td></td>
</tr>
</tbody>
</table>

#### Test Maintenance Phase

<table>
<thead>
<tr>
<th>Traditional testing Approach</th>
<th>MBT Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓ Identify Changes</td>
<td>✓ Identify the impact of the test plans</td>
</tr>
<tr>
<td>✓ Identify the impact in the model  ✓ Generate test scripts</td>
<td>✓ Update the model of the application</td>
</tr>
<tr>
<td>✓ Review and modify test scripts</td>
<td>✓ Review and modify test conditions</td>
</tr>
</tbody>
</table>

Copyright © 2014 Accenture All Rights Reserved.
Options for Modeling

Different options exist to create test cases from requirements via models, both include a manual component.

Formal Specification includes requirements and functional design documentation consisting of Use Cases, class diagrams, sequence diagrams, flow charts, decision trees etc.

Copyright © 2014 Accenture All Rights Reserved.
MBT: Advantages and Disadvantages

Advantages

• Higher degree of automation (test case generation)
• Allows more exhaustive testing
• Fastest way to get use of automated test
• Focuses on requirement coverage, not how many test cases you executed
• Good for correctness/functional testing
• Model can be easily adapted to changes

Disadvantages

• Requires a formal specification/model
• Test case explosion problem
• Test case generation has to be controlled appropriately to generate a test case of manageable size
• Small changes to the model can result in a totally different test suite
• Time to analyze failed tests (model, SUT, adaptor code)
MBT: Advantages and Disadvantages

Experiences using MBT

© 2012 Robert V. Binder
"2011 Model-Based Testing User Survey: Results and Analysis"
MBT: Case Study results

- Structured way for test design
- Usage of UML => early verification of SE Models
- Standardized test cases => easy automation
- Integrates in Standard Tool-Chain

- Complexity of the Models
- Dependency on Tools

- Lack of UML knowledge in testing community:
  - Lack Adoption
  - Maintenance on models
- Many changes the last few months due to GTS transformation
- Lack of adoption
- Initial investment

- More engineering skills for testers
- Improved test automation coverage
- Image of testing as a profession