Agility and MBT applied to software communication client project at Alcatel-Lucent Enterprise

Mathieu PAMARD – Ronan THEPAUT

ETSI UCAAT 2013 October 22-24 2013, France
Agenda

- Previous MBT experiences in our team
- A new target: OpenTouch™ Conversation, Windows PC edition
- OTC PC – why Model-Based-Testing?
- High-level modeling on video feature
- MBT application
- Tests generation
- Some figures
- Lessons learned
Alcatel-Lucent Enterprise R&D: previous MBT experiences in our team

- Previous experiences on several Alcatel-Lucent Enterprise projects
- Well targeted subjects identified on suitable criteria
- Modeling
  - IBM Rhapsody
- Tests generation
  - Conformiq© Designer

✅ Successful stories!

Decision to *industrialize the process* within Software Clients development projects following *Agility constraints*
Agenda

- Previous MBT experiences in our team
- A new target: OpenTouch™ Conversation, Windows PC edition
- OTC PC – why Model-Based-Testing?
- High-level modeling on video feature
- MBT application
- Tests generation
- Some figures
- Lessons learned
A new target:

OpenTouch™ Conversation, Windows PC edition

- Aka OTC PC
- New Alcatel-Lucent Enterprise communication client on PC
- Start of project August 2012
- Agile process, development in 2-weeks sprints
OTC PC – a component of OpenTouch solution

USER-CENTRIC

User-centric communication platform

MULTIPARTY

Native conferencing capabilities during the conversation, at any time

MULTIDEPVICE

Rapid session shift between devices while keeping the context

MULTIMEDIA

Audio, video, instant messaging, collaboration Escalation from voice to video, or from any media to any other during the conversation
Agenda

• Previous MBT experiences in our team
• A new target: OpenTouch™ Conversation, Windows PC edition
• OTC PC – why Model-Based-Testing?
• High-level modeling on video feature
• MBT application
• Tests generation
• Some figures
• Lessons learned
OTC PC – why Model-Based-Testing?

- **Definition**
  - 140 marketing requirements, 605 user stories, in 43 families
  - **User stories are interdependent**
  - **Scenario = concatenation** of user stories

✔ Easiness of modeling with scenario approach
OTC PC – why Model-Based-Testing?

• Definition
  - 140 marketing requirements, 605 user stories, in 43 families
  - User stories are interdependent
  - Scenario = concatenation of user stories

• Agility
  - SCRUM
  - User stories are delivered sprint after sprint
  - Some user stories evolve, some disappear, new ones appear

- Easiness of modeling with scenario approach
- Progressive and dynamic test plan
OTC PC – why Model-Based-Testing?

- **Definition**
  - 140 marketing requirements, 605 user stories, in 43 families
  - **User stories are interdependent**
  - Scenario = **concatenation** of user stories

- **Agility**
  - SCRUM
  - User stories are **delivered sprint after sprint**
  - Some user stories **evolve**, some disappear, new ones appear

- **Quality Assurance**
  - In sync with development
  - **Strong control** of coverage

- **Easiness of modeling** with scenario approach
- **Progressive and dynamic test plan**
- **Requirements traceability and control**
OTC PC – why Model-Based-Testing?

- **Definition**
  - 140 marketing requirements, 605 user stories, in 43 families
  - User stories are interdependent
  - Scenario = concatenation of user stories

- **Agility**
  - SCRUM
  - User stories are delivered sprint after sprint
  - Some user stories evolve, some disappear, new ones appear

- **Quality Assurance**
  - In sync with development
  - Strong control of coverage
  - Automated tests

- Easiness of modeling with scenario approach
- Progressive and dynamic test plan
- Requirements traceability and control
- Scripts automatic generation
Agenda

• Previous MBT experiences in our team
• A new target: OpenTouch™ Conversation, Windows PC edition
• OTC PC – why Model-Based-Testing?
• High-level modeling on video feature
• MBT application
• Tests generation
• Some figures
• Lessons learned
OTC PC – Process – Model for MBT ... but not only

Marketing Requirements analysis -> User stories (grouped in families) -> Families Modeling -> Sprint planning

Development -> MBT (QA)

High-level model

✓ is a graphics rendering of a user stories’ family
OTC PC – Process – Model for MBT ... but not only

- Marketing Requirements analysis
- User stories (grouped in families)
- Families Modeling

High-level model

- is a graphics rendering of a user stories’ family
- gives the context (before/after) of each user story
OTC PC – Process – Model for MBT ... but not only

Marketing
Requirements
analysis

User stories
(grouped in
families)

Families
Modeling

Sprint
planning

Development

MBT (QA)

Test Plan:
Manual / auto
Full / partial

High-level model:
✓ is a **graphics rendering** of a user stories’ **family**
✓ gives the **context** (before/after) of each user story
✓ helps to **identify scenario**
OTC PC – Process – Model for MBT … but not only

Marketing Requirements analysis → User stories (grouped in families) → Families Modeling

 Corrections / Updates

Sprint planning → Development → MBT (QA) → Generate

Test Plan: Manual / auto Full / partial

High-level model
✓ is a graphics rendering of a user stories’ family
✓ gives the context (before/after) of each user story
✓ helps to identify scenario
✓ allows to detect missing User Stories
High-level modeling: a first step to Model-Based-Testing

- **Idle**
  - US 732 – Video call presented on PC
  - US 734 – Call rejected

- **Ringing**
  - US 173 – Call diverted to voicemail
  - US 179 – Call answered in IM

- **Answered audio/video**
  - US 164 – Call answered on PC

- **Diverted**
  - US 330 – add video
  - US 370 – remove video
  - US 504 – pause video
  - US 505 – play video

- **IM session established**
Agenda

• Previous MBT experiences in our team
• A new target: OpenTouch™ Conversation, Windows PC edition
• OTC PC – why Model-Based-Testing?
• High-level modeling on video feature
• MBT application
• Tests generation
• Some figures
• Lessons learned
Model-Based-Testing - application to video feature (1)

- Start - Idle
  - Distant makes video call
- Incoming video call
  - Take call in IM
  - Take call in audio
  - Audio call established
    - Add video
    - Remove video
  - Take call in video
  - Video call established
    - Divert to voicemail
    - Call diverted
    - End call / session
- End - Idle

COPYRIGHT © 2013 ALCATEL-LUCENT ENTERPRISE. ALL RIGHTS RESERVED.
Model-Based-Testing - application to video feature (1)

- **Start - Idle**
  - Distant makes video call
    - US 732

- **Incoming video call**
  - Take call in IM
    - US 179
  - Take call in audio
    - US 164
  - Add video
    - US 330
  - Remove video
    - US 370
  - Take call in video
    - US 164
  - Divert to voicemail
    - US 173

- **IM session established**
  - Take call in IM
    - US 179
  - Take call in audio
    - US 164

- **Audio call established**
  - Add video
    - US 330
  - Remove video
    - US 370

- **Video call established**
  - Take call in video
    - US 164
  - Divert to voicemail
    - US 173

- **Call diverted**
  - End call / session

- **End - Idle**

Requirements:
- US 732 – Video call presented
- US 179 – Call answered in IM
- US 164 – Call answered on PC
- US 330 – Add video
- US 370 – Remove video
- US 173 – Call diverted to VM
- US 734 – Call rejected
Model-Based-Testing - application to video feature (1)

Requirements:
- US 732 – Video call presented
- US 179 – Call answered in IM
- US 164 – Call answered on PC
- US 330 – Add video
- US 370 – Remove video

Sprint 7: IM + add/remove video not delivered

25 tests generated
Model-Based-Testing - application to video feature (2)

Sprint 12: play/pause video added

**Incoming video call**
- Distant makes video call
  - US 732

**Start - Idle**
- Take call in IM
  - US 179
- Take call in audio
  - US 164

**Video call established**
- Take call in video
  - US 164
- Add video
  - US 330
- Remove video
  - US 370
- Video played
  - Play video
    - US 505
  - Pause video
    - US 504
- Video paused

**Audio call established**
- Take call in IM
  - US 179

**IM session established**
- Take call in audio
  - US 164
- Add video
  - US 330
- Remove video
  - US 370

**End - Idle**
- End call / session

**Requirements:**
- US 732 – Video call presented ✔
- US 179 – Call answered in IM ✗
- US 164 – Call answered on PC ✔
- US 330 – Add video ✗
- US 370 – Remove video ✗
- US 505 – Play video ✔
- US 504 – Pause video ✔
- US 173 – Call diverted to VM ✔
- US 734 – Call rejected ✔

30 tests generated (+ 5 tests)
Sprint 15: reject call removed from scope

Requirements:
- US 732 – Video call presented
- US 179 – Call answered in IM
- US 164 – Call answered on PC
- US 330 – Add video
- US 370 – Remove video
- US 504 – Pause video
- US 505 – Play video
- US 173 – Call diverted to VM
- US 734 – Call rejected

28 tests generated
(- 2 tests)
Sprint 19: IM + add video finally delivered!

Model-Based-Testing - application to video feature (4)

Requirements:
- US 732 – Video call presented
- US 179 – Call answered in IM
- US 164 – Call answered on PC
- US 330 – Add video
- US 370 – Remove video
- US 505 – Play video
- US 504 – Pause video
- US 173 – Call diverted to VM
- US 734 – Call rejected

36 tests generated (+8 tests)
Agenda

• Previous MBT experiences in our team
• A new target: OpenTouch™ Conversation, Windows PC edition
• OTC PC – why Model-Based-Testing?
• High-level modeling on video feature
• MBT application
• Tests generation
• Some figures
• Lessons learned
Tests generation – Automatic and manual tests

- High level scripts automatic generation

```javascript
source(findFile("scripts","QI_Video.js"));

function main()
{
beginSetup();
in_initConfiguration( "OTUser1", "OTUser2NotFavorite", "no", "", "no", "");
// requirement: Preconditions/User/User is/OTUser1/ (covered)
// requirement: Preconditions/Distant/User is/OTUser2NotFavorite/ (covered)
// requirement: Preconditions/User/Active Call/No active video call/ (covered)
// requirement: Preconditions/Distant/Forward/No forward/ (covered)
// requirement: Preconditions/Distant/Active call/No active call/ (covered)
// requirement: Preconditions/User/Forward/No forward/ (covered)
in_CorrespondantMakeAvideoCall( "OTUser2NotFavorite", "OTUser1");
// requirement: Transitions/OTCStartedUsersLoggedin - CorrespondantMakeACall -> IncomingCallToastDisplayed/ (covered)
// requirement: User Stories/Epic 38 video/732 - FC ringing - toast for video/ (covered)
// requirement: User Actions/Idle/Incoming video call/ (covered)
out_ToastDisplayed( "OTUser1", "OTUser2NotFavorite", "video");
out_CnvCardCreatedInWall( "OTUser1", "OTUser2NotFavorite");
in_UserAnswerInIMfromToast( "OTUser1", "OTUser2NotFavorite");
// requirement: User Stories/Epic 39 video advanced functionalities/179 - incoming video - answer in IM/ (covered)
// requirement: User Actions/Incoming video call/From toast/answered in IM/ (covered)
out_IMSessionOpened( "OTUser1", "OTUser2NotFavorite");
in_CloseIMSession( "OTUser1", "OTUser2NotFavorite");
// requirement: User Actions/IM established/Clos e IM/ (covered)
out_IMSessionWindowClosed( "OTUser1", "OTUser2NotFavorite");
out_CnvCardNoMoreActiveInWall( "OTUser1", "OTUser2NotFavorite");
endSetup();
}
```
Tests generation – **Automatic and manual tests**

- High level scripts automatic generation

```javascript
source(findFile("scripts","QI_Video.js"));

function main(){
  beginSetup();
  in_initConfiguration("OTUser1", "OTUser2NotFavorite", "no", "", "no", "");
  // requirement: Preconditions/User/User is/OTUser1// (covered)
  // requirement: Preconditions/Distant/User is/OTUser2NotFavorite// (covered)
  // requirement: Preconditions/User/Active Call/No active video call// (covered)
  // requirement: Preconditions/Distant/Forward/No forward// (covered)
  // requirement: Preconditions/Distant/Active call/No active call// (covered)
  // requirement: Preconditions/User/Forward/No forward// (covered)
  in_CorrespondantMakeAVideoCall("OTUser2NotFavorite", "OTUser1");
  // requirement: Transitions/OTCStartedUsersLoggedin - CorrespondantMakeACall _> IncomingCallToastDisplayed// (covered)

  // requirement: User Stories/Epic 38 video/732 - PC ringing - toast for video
  out_ToastDisplayed("OTUser1", "OTUser2NotFavorite", "video");
  out_ConvCardCreatedInWall("OTUser1", "OTUser2NotFavorite");

  // requirement: User Stories/Epic 39 video adv/179 incoming video answer in IM.
  in_IMSessionOpened("OTUser1", "OTUser2NotFavorite");
  in_CloseIMSession("OTUser1", "OTUser2NotFavorite");
  // requirement: User Actions/IM established/Close IM// (covered)
  out_IMSessionWindowClosed("OTUser1", "OTUser2NotFavorite");
  out_ConvCardNoMoreActiveInWall("OTUser1", "OTUser2NotFavorite");
  endSetup();
}
```
Agenda

• Previous MBT experiences in our team
• A new target: OpenTouch™ Conversation, Windows PC edition
• OTC PC – why Model-Based-Testing?
• High-level modeling on video feature
• MBT application
• Tests generation
• Some figures
• Lessons learned
Some figures

- **Modeling**:  
  ✓ Project total size: 43 families, 605 user stories  
  ✓ Candidates for MBT: 22 families, 250 user stories  
  ✓ Yet modeled: 14 families, 133 user stories

- **Regression automatic tests campaign**:  
  ✓ 10 sub-campaigns  
  ✓ 364 tests operational and running

- **Bug reports**:  
  ✓ 107 bug reports thanks to automatic tests  
  ✓ ~60% found during modeling / test automation development phase  
  ✓ ~40% are regressions found with automatic tests campaign
Agenda

• Previous MBT experiences in our team
• A new target: OpenTouch™ Conversation, Windows PC edition
• OTC PC – why Model-Based-Testing?
• High-level modeling on video feature
• MBT application
• Tests generation
• Some figures
• Lessons learned
Lessons learned

• Model brought at the **project level**
  ✓ **Overall view** of user stories
  ✓ **Rigor** in user stories writing
  ✓ Help for planning, development, QA

• MBT in agile process
  ✓ **Confidence** in tests coverage
  ✓ Big **focus on automation**
  ✓ **Rigorous** implementation of libraries
Back-up slides
Tests generation – **Automatic and manual tests**

- High level library skeleton automatic generation

```java
/**
 * DESCRIPTION_HERE
 *
 * @param who can take values "OTUser2NotFavorite", "OTUser2Favorite", "Anonymous". DESCRIPTION_HERE
 * @param callee can take values "OTUser1". DESCRIPTION_HERE
 *
 * @return nothing
 * @throws nothing
 */

function in_CorrespondentMakeAVideoCall(who, callee)
{
    try
    {
        
    } catch(e)
    {
        test.Fail(calleeFunction(arguments), e);
    }
}

/**
 * DESCRIPTION_HERE
 *
 * @param who can take values "OTUser1". DESCRIPTION_HERE
 * @param Distant can take values "OTUser2NotFavorite", "OTUser2Favorite", "Anonymous". DESCRIPTION_HERE
 *
```
Tests generation – **Automatic and manual tests**

- Manual tests Excel file generation – steps of description / expected result

<table>
<thead>
<tr>
<th>Test Name</th>
<th>Description</th>
<th>Step Name</th>
<th>Step description</th>
<th>Expected Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>005_IncomingVideoCall</td>
<td>Covered user stories: Epic 30 video/732 - PC ringing - toast for video</td>
<td>Step 1</td>
<td>Incoming video call from OTUser 7 in favorite list</td>
<td>Incoming video Call toast Displayed - Conv Card corresponding to the conversation created in the wall</td>
</tr>
<tr>
<td></td>
<td>Epic 39 video advanced functionalities/179 - incoming video - answer in IM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 2</td>
<td>User answer in IM through the toast of incoming video call</td>
<td>IM Session Opened -</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 3</td>
<td>User close the IM session</td>
<td>IM session closed</td>
</tr>
</tbody>
</table>
Problematics

• Manual tests - one generation?
  ➢ Brand new campaign at each new generation
  ➢ Redundant work for tester
  ➢ Only one generation, with tests partly postponed?

• Automatic (and manual) campaign status
  ➢ One test covers several user stories
  ➢ Failed test, how to easily identify the impacted user story?
MBT in agility process – conclusion

• Agile process, a new way of working
  ➢ User stories can evolve
  ➢ New user stories can be added
  ➢ Decisions to scope-out some user stories

☑ MBT, for a fast update of test plan
  ✓ Statecharts modification
  ✓ Generation triggers re-calibration
  ✓ New test plan generation

• Traceability : management strong request !

☑ User stories as requirements in statecharts
  ✓ Possibility to use them as triggers
  ✓ Each test includes the user stories it covers