Common Pitfalls of Test Automation (and how to avoid them)

Robert Crews
Common Pitfalls of Test Automation
(And How to Avoid Them)

Robert Crews
CorTechs, Inc.
Dec. 8, 2000
Common Pitfalls of Test Automation (And How to Avoid Them)

- Introduction to Available COTS Test Automation Products
- Common Myths and Errors of Test Automation
- Resolutions for Common Errors
- Questions & Answers
Introduction to Available COTS Test Automation Products

- Caliber-RBT
- Caliber-RM
- TestDirector
- WinRunner
- LoadRunner
- TestQuest
Requirements Management Tools

- **Caliber-Requirements Based Testing (RBT)**
  - A functional test case design tool that can be used to generate test definitions for any type of application, written in any language, running on any platform.

- **Caliber-Requirements Management (RM)**
  - A collaborative, Internet-based requirements management system that facilitates more effective requirement definition and management throughout the development cycle while also providing a centralized requirement repository and automatic change notification.

- **TBI (Technology Builders, Inc.)**
Test Planning and Automation Tools

- **TestDirector**
  - An integrated management tool for organizing and managing the testing process by combining planning, execution, defect tracking, with an open architecture and a central repository.

- **WinRunner**
  - An enterprise functional testing tool that verifies applications work as expected by capturing and replaying user interactions automatically.
Test Planning and Automation Tools (continued)

- **LoadRunner**
  - A load testing tool that predicts system behavior and performance. It exercises an entire enterprise infrastructure by emulating thousands of users to isolate problems, optimize performance and accelerate deployment.

- **Mercury Interactive, Inc.**
Non-Intrusive Test Automation Tool

- TestQuest
  - A non-intrusive automated test tool for Function, Regression, and System testing. Used extensively by many large organizations to test embedded systems. TestQuest is designed to fully automate the manual test cases of Software/System Quality Assurance organizations.
Common Myths and Errors of Test Automation

- Absence of a Test Plan
- Testing to Code (and Not to Requirements)
- Failure to customize testing tools
- Must Automate ALL Test Cases
- Automated Test Script Overly Complex
- Incorrect Use of Data Driven Tests
Resolutions for Common Errors

- Create a solid test plan.
  - What is a solid test plan?
  - What are the benefits of a test plan?
- Test to requirements.
- Brainstorm, plan, and customize!
- Do not have to automate all tests.
- Avoid overly complex automated tests.
- Use data driven tests more efficiently.
What is a Solid Test Plan?

- A documented test strategy which indicates:
  - Processes being tested
  - Requirements being tested for those processes
  - Actions required to test
  - Data to be tested
  - Expected Results
Benefits of a Solid Test Plan

- Eases verification of requirements coverage.
- Helps define priorities.
- Assists in determining which test cases to automate and which to leave manual.
- Eases delegation of testing tasks.
Test to the Requirements - Not to the Code !!!

- Know the requirements.
- Ensure coverage is complete.
- Consider using requirement management tool (i.e., Caliber-RM).
- Goal of testing is to find defects – not to ensure code works.
Brainstorm, Plan, Customize, and Standardize!

- Collect all available information (documentation, schematics, interviews, etc.) and determine customization needed.
- Determine types of customization:
  - User Defined Fields
  - Common Testing Routines and Functions
- Develop standards:
  - Test Names
  - Execution names
  - Variables and constants
Must Automate ALL Test Cases – Probably Not!!!

- A solid test will often consist of both automated and manual test cases.
- Define Conditions in Which Test Cases Should be Automated.
- Define Conditions in Which Test Cases Should be Manual.
Which Test Cases Not to Automate?

- Usability testing
  - "How easy is the application to use?"
- One-time testing
- "ASAP" testing
  - "We need to test NOW!"
- Ad hoc/random testing
  - based on intuition and knowledge of application
- Tests without predictable results
Which Test Cases to Automate?

☑️ Tests that need to be run for every build of the application (*sanity level*)

☑️ Tests that use multiple data values for the same actions (*data driven tests*)

☑️ Tests that require detailed information from application internals (e.g., SQL, GUI attributes)

☑️ Stress/load testing
Avoid Creating Overly Complex Test Scripts

- Use modular development for test scripts.
- Test scripts should **NOT** be as complex as the applications they are testing.
- Level of re-use should equal or exceed level of expected maintenance effort.
Use Data Driven Tests More Efficiently

- Use data file for expected results (as well as input data).
- Do not attempt to address ALL data derivations with one data driven test.
### Example

<table>
<thead>
<tr>
<th>Test ID</th>
<th>User Name</th>
<th>Password</th>
<th>Expected Results</th>
<th>Requirements Tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Robert Crews</td>
<td>cortcst</td>
<td>Successful login</td>
<td>SRS 33-A1, 33-A2, 35-A1, 36-A1</td>
</tr>
<tr>
<td>3</td>
<td>Mary Johnson</td>
<td>“ ”</td>
<td>Receive message indicating “please enter password”</td>
<td>SRS 33-A1, 35-A4, 35-A5, 35-C1</td>
</tr>
<tr>
<td>4</td>
<td>Mickey Mouse</td>
<td>cortcst</td>
<td>Receive message indicating “invalid user”</td>
<td>SRS 33-A1, 36-A1, 37-C1</td>
</tr>
<tr>
<td>5</td>
<td>“ ”</td>
<td>cortcst</td>
<td>Receive message indicating “please enter user name”</td>
<td>SRS 33-A1, 36-A3, 36-A4</td>
</tr>
</tbody>
</table>
Questions & Answers
Thank You!!!

Visit the CorTechs booth for an informative demonstration of the premier automated testing and requirement management tools!!!
Name: Robert Crews
Company: CorTechs
Title: Common Pitfalls of Test Automation (And How to Avoid Them)

Description: Organizations often fall prey to the same myths and make the common errors when automating their testing efforts. This presentation will present those common errors and myths along with suggested solutions. These solutions include recommended COTS products as well as guidelines that the organization may follow.

Target Audience: Intermediate (QA Testers, Managers, and Team Leads)
Audience Will Learn: The audience will learn common myths and errors of automated testing to avoid. They will also learn suggested solutions. The importance of a solid test plan, especially when automating tests, will be presented. Available COTS products will be discussed. Emphasis will also be placed on suggested practices and guidelines, which will assist in avoiding automation errors.

Basis of Presentation: This presentation is based upon my own work experience in full life-cycle development (including automated and manual testing) as well as my certification as an instructor and specialist with a leading test automation tool.

Industry Experience: Robert Crews is a consultant with over ten years of experience in full life-cycle development involving design, development, and testing. He has worked on projects with well over 25,000 requirements and over 200,00 test cases (both manual and automated). He is currently a certified instructor of Mercury Interactive’s WinRunner (a test automation tool) and TestDirector (a test management tool) as well as Qronus’ TestRunner (a completely non-intrusive automation testing tool).

Outline:

I. Automated Data-Driven Test Demonstration

II. Introduction to Available COTS Test Automation Products
   A. Caliber-RM
   B. TestDirector
   C. WinRunner
   D. TestRunner

III. Common Myths and Errors of Test Automation
   A. Lack of a Test Plan
   B. Testing to Code (and Not to Requirements)
   C. Must Automate ALL Test Cases
   D. Automated Test Overly Complex
   E. Incorrect Use of Data Driven Tests

IV. Resolutions for Common Errors
   A. Lack of a Test Plan
      1. Importance of Test Plan
         a. Helps Define Priorities
         b. Helps Determine Which Test Cases to Automate
         c. Helps Determine Which Test Cases to Leave Manual
         d. Helps in Delegating Testing Tasks
2. Use Test Management and Requirement Management Tools
   a. Test Director
   b. Caliber-RM
   c. Caliber-RBT (SoftTest)

B. Testing to Code (and Not to Requirements)
   1. Caliber-RBT and Caliber-RM
   2. Definition of Requirement
   3. Know the Requirements
   4. Map Requirements to Test Cases

C. Must Automate ALL Test Cases
   1. Define Conditions in Which Test Cases Should be Automated
   2. Define Conditions in Which Test Cases Should be Manual
   3. Level of Re-use vs. Level of Maintenance

D. Automated Test Overly Complex
   1. Use Modular Development for Test Scripts
   2. Test Scripts Should NOT be as Complex as The Applications They are Testing
   3. Level of Re-use Versus Maintenance Effort

E. Incorrect Use of Data Driven Tests
   1. Do Not Attempt to Address ALL Data Derivations with One Data Driven Test
   2. Use Data File for Expected Results (As Well As Input Data)

V. Summary
Common Pitfalls of Test Automation
(and how to avoid them)

Robert Crews

Robert Crews, a Managing Partner with CorTechs, Inc., is a consultant with over eleven years of experience in full life-cycle development involving data modeling, design, development, and testing. He has worked on projects with well over 30,000 requirements and over 200,00 test cases (both manual and automated) in client/server, mainframe, and POS environments. He is currently a certified instructor of Mercury Interactive’s WinRunner (a test automation tool) and TestDirector (a test management tool) as well as TBI’s Caliber-RM (a requirements management and traceability tool). Mr. Crews has assisted numerous organizations in the development of efficient requirements, and test plans as well as the implementation of requirement management, test management and test automation tools.