P17 System Testing
Monday, May 24, 2007

Module 1: Definition and Purpose of Testing
“Discovery consists of looking at the same thing as everyone else does and thinking something different.”

Albert Szent-Györgyi, 1937 Nobel Prize winner in physiology and medicine
Module Objectives

After completing this module, you will be able to:

- Define the purpose of testing
- Describe the core testing activities
- Define testware and provide examples
- Define static and dynamic testing
- Explain how testware relates to the core testing activities
What is Testing Purpose?

The purpose for testing is to execute or evaluate programs or systems that do the following:

- **Measure** the results against the requirements
- **Document** the difference between the expected and actual results
- **Assist in resolving** those differences by providing the proper debug aids
Testing Purpose Examples

- Uncovering defects and finding important problems
- Assessing quality and risk
- Certifying to standards
- Fulfilling process mandates
- Blocking premature releases
- Minimizing safety-related lawsuit risks
- Minimizing technical support costs
- Maximizing efficiency
- Verifying correctness
- Assessing conformance to specifications or regulations
# Testing Focus by Perspective

Based on the role of the person who has a stake in the testing process the focus or purpose of the testing process can vary.

<table>
<thead>
<tr>
<th>Perspective</th>
<th>Focus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developer</td>
<td>Testing validates that a program or system conforms to the requirements</td>
</tr>
<tr>
<td>Project management</td>
<td>Testing measures whether the deliverable is of high quality</td>
</tr>
<tr>
<td>Tester</td>
<td>Testing finds the meaningful errors in the timeframe allocated and verifies that fixes provided for the errors bring the program or system into conformance with the requirements</td>
</tr>
<tr>
<td>Corporation or business development</td>
<td>Testing can improve quality and customer satisfaction, reduce costs related to customer service calls and rewrites of software, and improve the profit margin for the business</td>
</tr>
</tbody>
</table>
Testing Dimensions of Quality

- **Reliability**: Does the application behave consistently and predictably?
- **Performance**: Test performance under expected and worst-case load.
- **Functionality**: Test the accurate workings of each usage scenario.
- **Supportability**: Test the ability to maintain and support application under production use.
- **Usability**: Test application from the perspective of convenience of its users.
## Core Testing Activities

<table>
<thead>
<tr>
<th>Core Activity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analyze and model</td>
<td>The Tester analyzes the requirements and develops use cases that will verify that the implementation meets the requirements.</td>
</tr>
<tr>
<td>Design</td>
<td>Tester refines the use cases into designs based on the architecture and environments using a set of methodologies to help scope the effort.</td>
</tr>
<tr>
<td>Implement</td>
<td>The Tester implements the design.</td>
</tr>
<tr>
<td>Execute</td>
<td>The test executes the test cases, analyzes the results, reports problems found and drives them to closure, evaluates the software and documents the results.</td>
</tr>
</tbody>
</table>
## Types of Testing

<table>
<thead>
<tr>
<th>Techniques</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static Testing</td>
<td>Looks at the source code without executing it</td>
</tr>
<tr>
<td>Dynamic Testing</td>
<td>Testing that requires execution of the code</td>
</tr>
<tr>
<td>White-box Testing</td>
<td>Focuses on identifying tests through a deep understanding of the software internals - you read the code to identify test cases.</td>
</tr>
<tr>
<td>Black-box Testing</td>
<td>Focuses on testing from a user perspective. -you do not need to understand the internals of the software, but must have a thorough understanding of the requirements and use cases so you can identify test cases.</td>
</tr>
</tbody>
</table>
Value of Static Testing

- Perform earlier than dynamic testing
- You can look at the entire design/program and you can consider all use cases at once
- Dynamic only test anticipated variations/interests - Inspection lets you identify missing assumptions as well
What is Testware?

Testware is the *deliverable* from the testing process. Rational Unified Process (RUP) refers to these deliverables as test artifacts.

It is important to *maintain* testware as it has a life beyond its initial use. Testware is similar to software in that you can reuse it without incurring the cost of development with each use.

As with software, develop testware under *version control* so that it can be maintained and not lost.
# Testware Examples

<table>
<thead>
<tr>
<th>Testware</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Plans</td>
<td>Documents that are produced from the activity of test planning.</td>
</tr>
<tr>
<td>Use Cases</td>
<td>The results of analyzing requirements</td>
</tr>
<tr>
<td>Test Design Specifications</td>
<td>Document the design of the test</td>
</tr>
<tr>
<td>Test Cases, Test Scripts and Programs</td>
<td>The results of implementing the test design</td>
</tr>
<tr>
<td>Test Procedures</td>
<td>Describe how to execute a test case</td>
</tr>
<tr>
<td>Test Data Files</td>
<td>Data results from executing the test cases</td>
</tr>
<tr>
<td>Defect Report</td>
<td>Document problems found during execution</td>
</tr>
<tr>
<td>Test Status</td>
<td>Document the progress of the testing effort</td>
</tr>
<tr>
<td>Final Test Report</td>
<td>Document the final overall results and experience gained from this test effort</td>
</tr>
</tbody>
</table>
Design Reusable Testware

- Modularize your test scripts
- Documented standards for test product usage
- Minimize test scripts with detailed test data
Testware and Core Testing Activities

- Requirements specification
  - Analyze

- Use cases for testing
  - Design

- Test design specifications
  - Implement

- Test cases, test scripts, programs, procedures
  - Execute

- Defect reports

IBM Rational RequisitePro

IBM Rational Software Modeler

IBM Rational ClearQuest TestManager

IBM Rational Manual Tester, Functional Tester & Performance Tester

IBM Rational ClearQuest
Module summary

- The purpose of testing is to evaluate products to measure the results against the requirements, document the differences, and help to resolve the differences.

- Core testing activities include analyze and model, design, implement and execute.

- Types of testing techniques include: Static, Dynamic, White-box and Black-box Testing

- The deliverables from the testing process are called testware.

- Testware includes test plans; test cases, test scripts, and programs; test procedures; test data files; defect reports; and test reports.

- It is important to design testware so that it can be reused