

TECHNOLOGY TRANSFER PRESENTS

RANDY RICE

**SOFTWARE
TESTING**

NOVEMBER 17-18, 2008

**ADVANCED
SOFTWARE
TESTING**

NOVEMBER 19-21, 2008

RESIDENZA DI RIPETTA - VIA DI RIPETTA, 231
ROME (ITALY)



info@technologytransfer.it
www.technologytransfer.it

ABOUT THIS SEMINAR

This is a quick start course in software Testing for people just getting into the field, or for people who just need a refresher course or validation for their current Testing techniques. This is a practical hands-on seminar to cover the critical path of Testing. Your instructor will be a certified instructor in the QA and Testing field. You will learn the terminology, process, and challenges of Testing in the real world. As a result of attending this seminar, you should have a good working knowledge of software Testing and what it takes to design and conduct an effective Test of software, regardless of the technology.

Software Testing will help you become more comfortable and confident in Testing software applications at just about any level of detail: unit, integration, system, and user acceptance. You will emerge from this one-day session knowing how to develop Test cases and Test plans. You will also leave with a knowledge of how tools can help you perform Testing.

Sometimes people feel intimidated by the technical aspects of software Testing and lack the confidence they need to be credible Test leaders in their organization. Learn the issues and processes for effectively Testing software by attending this hands-on course.

WHAT YOU WILL LEARN

- Learn how to find costly and embarrassing problems before your customers find them
- Understand the key issues in Testing software applications
- Learn how to design Tests that adequately cover requirements and Business events
- Get the most out of your existing investment in Testing and how to leverage that investment
- Advance your career by reinforcing your Testing expertise

WHO SHOULD ATTEND

- Test Analysts
- Testers
- Web Developers
- Mainframe and Client/Server Developers making the transition to Web development

Please, bring your laptop to the seminar.

1. Surviving The Top 10 Challenges of Software Testing

- The Top 10 Challenges
- Solutions to Each Challenge

2. Terminology

- The Deming Workbench Model
- Software Life Cycle
- Test Terminology

3. The Economics of Testing

- Where Defects Originate
- Where Testing Resources are Used
- The Relative Cost of Fixing Defects

4. Basic Test Planning

- Basic Testing Principles
- The Basic Testing Process
- What is a Test Strategy?
- The Components and Format of a Test Strategy
- Identifying Critical Success Factors
- Defining Test Objectives
- The Components and Format of a System-level Test Plan
- Major Elements of a Test Plan
- How Much Time Should be Spent on Test Planning?
- Planning Time Guidelines
- Tips for Test Planning

5. The Test Planning Process

- Key Test Planning Components
- Major Test Planning Tasks
- Developing a Test Strategy
- Critical Success Factors
- Defining Test Objectives
- Identifying Needed Test Resources
- Planning the Test Environment
- Defining the Test Procedures
- Identifying the Functions To Be Tested
- Identifying the Interfaces With Other Systems or Components
- Writing Test Scripts
- Defining Test Cases
- Designing Test Data
- Building a Test Matrix
- Determining Test Schedules
- Assembling the Information
- Finalizing the Test Plan

6. Test Case Development

- Key Test Planning Components
 - Test Scripts
 - Test Cases
 - How to Document Test Cases
- Types of Test Cases
 - Functional
 - Boundary
 - Equivalence
 - Requirements-based
 - Business-oriented
 - Behavioral
 - Structural
 - Logic-based
 - Behavioral
- Ways to Design Functional Test Cases
 - Boundary Value Analysis
 - Equivalence Classes
 - Requirements-based Cases
 - Error Guessing
 - Business-oriented Cases
 - Transaction Threads Based on Business Scenarios
 - * Test/Cycle Matrix
 - * What is a Test Cycle?
 - * Why Use Test Cycles?
 - Use Cases
 - * Use Case Components
 - * Structural Test Cases - Logic-based
 - * Structural Test Coverage Levels
 - * Structural Test Cases - Behavioral
 - * Regression Test Cases

7. Test Execution Strategies

- Automated Testing
 - Risks of Not Automating Testing
 - Risks of Automating Testing
 - Where Do Tools Fit In?
 - The Major Issues
 - Top 10 Test Tools
 - Critical Success Factors
- Test Execution - Manual Methods
 - Test Folders
 - Test/Function Matrix
- Building the Test Environment
- How to Create and Maintain Test Data
- The Process for Configuration/Release Management
- Pitfalls to Avoid

8. Regression Testing

- What is Regression Testing?
- No Regression Testing: Hidden Defects
- Regression Testing: No Hidden Defects
- Regression Testing - The Process
- Regression Testing - What's Needed?
- Regression Testing Issues
- Regression Testing - How Much is Enough?
- Tips for Performing Regression Testing

9. Test Evaluation and Reporting

- Prerequisites for Test Evaluation
- Test Evaluation and Reporting Process
- Test Reporting Attributes
- Types of Test Reporting
 - Defect Reports
 - Status Reports
 - Final Report
- System Test Evaluation - Defect Reporting
 - Paper-based
 - Defect Tracking Tools
 - The Role of the Defect Administrator
 - The Defect Life Cycle
 - Defect Tracking - Things You Need to Know
 - Sample Defect Categories
 - Sample Defect Priorities
- Status Reporting
- Final Reporting
- Test Summary Report
- How Can This Data be Used?

ABOUT THIS SEMINAR

This course is designed for software Testers that want to go deeper than the basic concepts. Attendees will learn how to create an effective Test strategy, how to design creative Test cases, how to optimize Test cases to get the most Testing from the fewest number of cases and how to measure and report the results of Testing. Attendees will leave this course with a solid foundation for Testing in situations which are very diverse and dynamic.

This course is centered around a common Case Study which builds throughout the course.

Advanced Software Testing will help you reach the next level in your Testing skills. You will emerge from this three-day session knowing how to plan and conduct Tests in diverse and complex environments.

WHAT YOU WILL LEARN

- Describe the major software development lifecycles and how Testing fits into those methodologies
- Develop a Test strategy
- Write a high-level Test plan
- Develop Test scripts and Test cases using a wide variety of techniques
- Create decision tables
- Get the most Testing from the least number of Test cases
- Track and control Test scripts and cases
- Assess risk from the project, technical and Business perspectives
- Describe which tools are the best to use in a particular Test
- Describe effective Test tools available
- Write meaningful Test reports
- Measure your Testing efforts
- Use the results from Testing to improve the Testing process and other processes

WHO SHOULD ATTEND

- Test Analysts
- Testers
- Test Engineers
- Software Developers
- Project Managers
- Test lab Managers
- QA Analysts

Please, bring your laptop to the seminar.

1. Test Planning Overview: Strategies and Tactics

- Basic Testing Principles
- Why Is It Important That Testing Uses A Process?
- Process Benefits
- The V Diagram
- What is a Test Strategy?
- The Components and Format of a Test Strategy
- Identifying Critical Success Factors
- Sample Test Strategy
- Defining Test Objectives
- Sample Test Objectives
- Major Elements of a Test Plan
- Planning Time Guidelines
- Tips for Test Planning

2. Advanced Test Design Strategies

- How Much Effort Should Be Dedicated to Test Design and Types?
- Tests by Type of Project
- Test Case Economy
- How to Match the Test to the Project and Test Stage
- Rules for Test Applicability
- Producer Testing
- Customer Testing
- Automated Test Strategies
- Test Cases
 - Test Case Types
 - How to Document Test Cases
 - Organizing the Test Cases
 - The Decomposition of Test Cases from Test Objectives and Functions
 - A Test Case Design and Tracking Tool
 - Sample Test Cases
 - Test Case Best Practices
 - How to Be Creative in Test Case Design
- Types of Test Cases
 - Functional Test Cases
 - Structural Test Cases
 - Regression Test Cases
 - Security Test Cases
 - Performance Test Cases
 - Interoperability Test Cases
 - Compatibility Test Cases
 - Portability Test Cases
 - Usability Test Cases
 - Scalability Test Cases
 - End-to-end Testing
 - Batch Test Cases

- Conversion Test Cases
- Link Test Cases
- Browser Test Cases
- Exploratory Testing
- Parallel Test Cases
- Vendor Test Cases
- Test Cases by Phase
 - * Unit Test Cases
 - * Integration Test Cases
 - * System Test Cases
 - * Acceptance Test Cases

3. Advanced Test Case Design Techniques

- Orthogonal Arrays
 - What's the Need?
 - What is Orthogonality?
 - The Fault Model Behind Orthogonal Array Testing
 - Double-mode Defects
 - What Research Indicates
 - The Likelihood of Triple-mode or Higher Defects
 - What Common Sense Indicates
 - The Value of Designing Tests With Orthogonal Arrays
 - Terminology
 - Orthogonal Notation
 - Orthogonal Array - Example
 - Facts About Orthogonal Arrays
 - The Good News
 - A Process for Applying Orthogonal Arrays
 - An Example
 - Exercise - Build an Orthogonal Array for Case Study Exercise
 - Sample Orthogonal Arrays
 - The Allpairs Algorithm
 - Tools for Pairwise Testing - Allpairs
 - A Word of Warning About Pairwise Testing
 - Review - How Do Orthogonal Arrays and Pairwise Testing Help?
 - Resources for Orthogonal Arrays and Pairwise Testing
- Decision Tables
 - Decision Table Example
 - Decision Table with Three States
 - How Decision Tables Help in Test Case Design
 - How Decision Tables are Derived
 - Reducing Decision Tables Example

- Why Reduce Tables?
- Expanding Decision Tables
- Why Expand Tables?
- How do Organizations Use Decision Tables?
- Cause-Effect Graphing
 - The Cause-Effect Graphing Process
 - Cause-Effect Graphing Symbols
 - Cause-Effect Graphing Example
 - Helpful Tips for Decision Tables
- Test Cases from Use Cases
 - Use Cases
 - Use Case Components
 - The Use Case to System Test Case Process
 - More About Use Cases as Test Cases
- Functional Test Cases - Requirements-based Cases
 - The Process for Decomposing Requirements into Test Cases
 - Benefits of the Process
 - Overview of the Process
 - Requirement Specification to System Test Case Process Approach
 - Process Tailoring
 - Exercise
 - Conclusions About the Requirements to Test Case Process
 - How to Maximize Requirements Coverage
 - Example: Maximizing Requirements Coverage
 - Can the Number of Test Cases be Reduced?
 - A Redundant Test Case
 - Change the Outcome with One Simple Change
- Tests Based on Business Scenarios
 - Tests Based on Business Scenarios - Example
 - Test/Cycle Matrix
 - Sample Test/Cycle Matrix
 - What is a Test Cycle?
 - Why Use Test Cycles?
 - Traditional Test Cycle
- Functional Test Cases - Behavioral

4. Risk Assessment

- What is Risk?
- The Nature of Risk
- The Three Views of Risk
- The Elements of Risk

- Risk Assessment
- Why is Risk Assessment Important?
- Computer System Risks
- Three Views of Project Risk
- Assessing Technical Risks
 - Likelihood of Failure
 - Impact of Failure
 - A Sample Risk Chart
 - Practical Application
 - Ways to Apply the Results
 - Example: Applying the Results
- Assessing Business Risk
- Sample Business Risk Chart
- How Can This Information Be Used?
- When is Testing Complete?
- A Problem in Risk-Based Testing - Cases that Span Risk Levels
- When is Risk Assessment Performed?
- Who Performs Risk Assessment?
- Who Owns Risk Assessment?
- Additional Resources

5. Test Tool Overview

- Working Definition of a Test Tool
- Risks of Not Automating Testing
- Risks of Automating Testing
- The Role of Test Tools
- Manual Testing
- Automated Testing
- The Major Issues
- Top 10 Test Tools
- Interactive Test/Debug
- Capture/Playback
- Version Control
- Stress and Load Testing
- Defect Tracking
- Memory Testing
- Test Management
- Coverage Analyzers and Thread Testers
- Checklists
- Critical Success Factors
- Closing Thought

6. Performance Testing

- Key Concerns
- The Challenge
- Hitting the Performance Wall
- Performance Testing
- Terminology
- Prerequisites
- Type of Technology

- An E-Commerce Schematic
- Client/Server Load Testing
- Manual vs. Automated Tools
 - Which Tools are Applicable?
 - How Can Tools Help?
 - Common Problems and Pitfalls
- Memory Leaks
 - Anatomy of a Boundary Violation
 - Anatomy of a Memory Leak
 - How to Test for Memory Leaks
 - Bounds Checkers
 - Example: NuMega Bounds Checker Interface

7. Advanced Test Evaluation and Analysis

- Test Evaluation Workbench
- An Evaluation Toolkit
- A Testing Dashboard
- Statistical Methods
 - Example
 - Sample Control Chart
 - Example - Standard Deviation
 - Example - Computing the Control Limits
 - What Does This Tell Us
 - How to Go Forward
 - Common Causes vs. Special Causes
- How to Measure Test Coverage
 - Requirements Coverage
 - Adding Traceability
 - Requirements and Test Case Coverage
 - Tools for Test Case Design from Requirements
 - Requirements and Test Case Coverage
 - Business Process Coverage - The Test/Iteration Matrix
- How to Capture Test Results
- How to Manage the Level of Measurement Intrusiveness
- Comparison Tools and Techniques
- How to Build a Robust Testing Baseline
- Regression Testing - Comparing to a Baseline
- How to Create and Maintain Baseline Test Data
- Test Baseline Data Cycle
- Regression Testing - Maintaining the Baseline
- How to Analyze and Make Sense of Test Results

- Comparison to the Baseline: Correctness
- Tracking to Current Project Goals
- Comparison to Past Projects
- Comparison to Estimates
- How to Present Complex Information in Understandable Ways
- Continually Improving the Process
- The Goal/Question/Metric Paradigm
- Identifying Needs and Goals
- Answering the Right Questions
- Critical Success Factors
- Exercise - Develop Your Own Action Plan for Improvement

8. Metrics and Measurements

- Terminology
- Valuable Test Measurements & Metrics
- Benefits of Testing Metrics and Measurements
- What Needs to be in Place to Capture Measurements and Metrics
- Tools and Techniques for Measuring Testing Activities
- Why Track Defects?
- Tracking and Understanding Defect Trends
- The Role of the Defect Administrator

9. Test Evaluation and Reporting

- Prerequisites for Test Evaluation
- Test Evaluation and Reporting Process
- What Test Reporting Should Be
- Types of Test Reporting
- System Test Evaluation - Defect Reporting
- The Defect Life Cycle
- Defect Tracking - Things You Need to Know
- Sample Defect Categories
- Sample Defect Priorities
- Status Reporting
- Final Reporting
- Test Summary Report
- How Can This Data be Used?
- Final Thought

<p>PARTICIPATION FEE</p> <p>Software Testing € 1200</p> <p>Advanced Software Testing € 1500</p> <p>Special price for the delegates who attend both seminars € 2500</p> <p>The fee includes all seminar documentation, luncheon and coffee breaks.</p> <p>VENUE</p> <p>Residenza di Ripetta Via di Ripetta, 231 Rome (Italy)</p> <p>SEMINAR TIMETABLE</p> <p>9.30 am - 1.00 pm 2.00 pm - 5.00 pm</p>	<p>HOW TO REGISTER</p> <p>You must send the registration form with the receipt of the payment to: TECHNOLOGY TRANSFER S.r.l. Piazza Cavour, 3 - 00193 Rome (Italy) Fax +39-06-6871102</p> <p>within November 3, 2008</p> <p>PAYMENT</p> <p>Wire transfer to: Technology Transfer S.r.l. Banca Intesa Sanpaolo S.p.A. Agenzia 6787 di Roma Iban Code: IT 34 Y 03069 05039 048890270110</p>	<p>GENERAL CONDITIONS</p> <p>GROUP DISCOUNT</p> <p>If a company registers 5 participants to the same seminar, it will pay only for 4. Those who benefit of this discount are not entitled to other discounts for the same seminar.</p> <p>EARLY REGISTRATION</p> <p>The participants who will register 30 days before the seminar are entitled to a 5% discount.</p> <p>CANCELLATION POLICY</p> <p>A full refund is given for any cancellation received more than 15 days before the seminar starts. Cancellations less than 15 days prior the event are liable for 50% of the fee. Cancellations less than one week prior to the event date will be liable for the full fee.</p> <p>CANCELLATION LIABILITY</p> <p>In the case of cancellation of an event for any reason, Technology Transfer's liability is limited to the return of the registration fee only.</p>
--	--	--

RANDY RICE

SOFTWARE TESTING

Rome November 17-18, 2008
Residenza di Ripetta - Via di Ripetta, 231
Registration fee: € 1200

ADVANCED SOFTWARE TESTING

Rome November 19-21, 2008
Residenza di Ripetta - Via di Ripetta, 231
Registration fee: € 1500

BOTH SEMINARS

Special price for the delegates who attend both seminars: € 2500

first name

surname

job title

organisation

address

postcode

city

country

telephone

fax

e-mail



Stamp and signature

Send your registration form with the receipt of the payment to:
Technology Transfer S.r.l.
Piazza Cavour, 3 - 00193 Rome (Italy)
Tel. +39-06-6832227 - Fax +39-06-6871102
info@technologytransfer.it
www.technologytransfer.it

If anyone registered is unable to attend, or in case of cancellation of the seminar, the general conditions mentioned before are applicable.



Randy Rice is a leading author, speaker and consultant in the field of software Testing and software quality. Mr. Rice, a Certified Software Quality Analyst, Certified Software Tester, Certified Software Test Manager and an ASTQB Certified Tester – Foundation Level (CTFL), has worked with organizations worldwide to improve the quality of their information systems and optimize their Testing processes. Mr. Rice has over 29 years experience building and Testing mission-critical projects in a variety of environments. He is a member of the American Software Testing Qualifications Board and has been published by *the Journal of the Quality Assurance Institute*, *Crosstalk*, *Client/Server Computing*, *Powersoft Applications Developer* and *Enterprise Systems Journal*. He is a regular speaker at international conferences on software Testing in North America and Europe, and is also publisher of *The Software Quality Advisor*. He is co-author with William E. Perry of the books, “**Surviving the Top Ten Challenges of Software Testing**” and “**Testing Dirty Systems published**” by Dorset House Publishing Co. Mr. Rice served as chair of the Quality Assurance Institute’s International Software Testing Conference from 1995 – 2000 and was a founding member of the Certified Software Test Engineer (CSTE) certification program. As author and trainer of many software Testing training courses, he has had the privilege of training thousands of software testers throughout North America. He is a frequent speaker at major Testing conferences, including EuroStar, StarEast, StarWest, QAI’s International Software Testing Conference, Practical Software Quality Techniques, and the Software Technology Conference sponsored by the U.S. Dept. of Defense. He was a judge for the 2005 Codie awards for excellence in software Testing tools. He also serves on the board of directors for the American Software Testing Qualifications Board (ASTQB).