

Ross Collard's Integration, Interface & Interoperability Testing (3i Testing)

Duration

1 or 2 days

Instructor

Roland Stens

Class Limit

20 students

Prerequisite

A working knowledge of system testing & quality assurance fundamentals is assumed. No specific technical background is assumed, though examples will be given in various areas where integration is important (e.g., TCP/IP, XML, Corba).

Price

On-site:

Please contact SPC for pricing

Public Training:

\$995 (2 days)

*Discount available for early registration

Materials Provided

- Student manual containing the course slides
- Student handouts with class exercises and class studies

Integration is an essential characteristic of any network, database or assembled set of interacting systems. The human brain is a collection of many small, relatively simple units (neurons), which produces complicated results through the interaction of its parts. An individual neuron cannot do much; it cannot even survive on its own but it is connected with thousands of other neurons. It is their integration that gives neurons their collective power.

Integration testing is the process of ensuring that components fit together and work together correctly, after they have been shown to work individually by unit testing. For example, the client-based queries in a system need to work correctly with the remote server data bases. The art of integrating together components, applications or systems is a complex one. There are many possibilities for incompatibility and interference among the components, leading to the need to test their integration. The term "integration" means different things to different people, such as:

- Transaction flows among systems.
- Database integration.
- Connectivity of networks.
- Compilation and linkage of re-usable software components.
- Interfaces between applications which pass data back and forth between themselves, and reside on different platforms.
- Interaction among different applications which co-exist and share the same resources, e.g., reside on the same personal computer or use the same database.

The testing of all these types of integration will be discussed in this class.

Objective

At the end of this seminar, participants will be able to:

- Use the major integration test techniques.
- Examine the causes of interface errors.
- Test an interface.
- Test systems for interoperability and connectivity
- Test integrated third-party software and hardware.
- Test systems for compatibility across different configurations and platforms.
- Automate integration testing, and the advantages of limitations of automation.

For more information on this or other SPC Springboard courses, please visit www.spcspringboard.com or e-mail SPC at info@spc.ca

Software Productivity Center
Suite 460—1122 Mainland Street
Vancouver, BC V8M 4T8
www.spc.ca

Toll Free:
1.877.548.1948
Vancouver:
604.662.8181

Fax:
604.689.0141
Toronto:
416.885.0512



Instructor

Ross Collard is president of Collard & Company, and specializes in software testing and quality assurance. His consulting assignments have included strategic planning for technology, managing large software development projects, and development of software engineering practices. His clients have included Amazon.com, American Express, Boeing, General Electric, Hewlett Packard, IBM and NASA. Ross has an MS in Computer Science from the California Institute of Technology and an MBA from Stanford University.

Roland Stens is an independent consultant based in Vancouver who specializes in QA and testing - particularly performance and robustness testing. His broad background in programming, system analysis, database and network management, testing and project/test management enables him to relate to all the issues that play during the software development lifecycle.

Intended Audience

System testers and quality assurance professionals, project leaders, system designers, software engineers, and users who are involved in integration testing.



Outline

Introduction and Overview

- The Nature of Integration Projects
- Definition of Integration Testing
- Types of Integration Tests
- The Need for Integration Tests
- Advantages of Integration Testing
- Disadvantages and Limitations
- Levels of Integration Testing
- Keys for Effective Integration Testing
- Who Performs the Integration Testing?
- Teamwork in Integration Testing
- Forming the Integration Test Team
- Entry and Exit Criteria for Integration Testing
- The Reverse Big-Bang Approach
- Sources of Integration-Level Errors

The Nature of Integration Testing

- Testing the Interaction of Two Components
- Dependency Tables
- Unnoticed Connections
- Integration Test Cycles
- The Flow of Integration Test Cycles
- Finger Pointing in Integration Testing
- Subsystem Acceptance Criteria
- Integration Testing Strategy: A Case Study
- End-to-End Testing
- Pairwise Testing
- Cross-Functional Coverage (Feature Interaction Testing)
- Interoperability Testing
- The Scope of System Integration Activities
- Integration Test Alternatives
- Definition of a Thread

- Multi-threaded Testing
- Definition of a Build
- The Building Process
- Frequent Re-Builds
- Testing Incremental Builds

Integration Techniques

- Determining the Order of Integration
- Stringing Components
- Top-Down Integration
- Inheritance Integration
- Bottom-Up Integration
- Sandwich Integration
- Upstream / Downstream
- Step-by-Step
- Integration Test Plan Outline

Testing Interfaces

- Interfaces, Protocols and Sockets
- Interface Agreements
- Natural vs. Simulated Interfaces
- Building Interface Simulators
- Mapping and Matching the Interface
- TCP/IP Example
- Open Vs. Proprietary Interfaces
- Environment-Specific Interfaces
- Testing Multiple Interfaces
- Synchronization of Independent Programs and Processes
- Managing Shared Databases and Tables

Specific Integration Environments

- Integrating Third-Party Products
- Hardware/Software Integration
- Hardware Testing
- API Testing
- Middleware Testing
- Testing Device Drivers
- Client/Server Integration Testing
- Network Integration
- Integration with Legacy Systems
- The Influence of System Architecture

Architecture and Integration

- Architectural Mismatches
- Best-of-Breed vs. Integrated Solutions
- "Disintegration" Testing

Issues and Success Factors

- Estimating the Integration Test Effort
- Meeting Aggressive Deadlines with Smart Integration
- Integration Test Issues
- Integration Success Factors
- Integration Testing Success Factors



TRAINING