



Contents at a Glance

<i>Introduction</i>	<i>xv</i>
Chapter 1: Software Testing: An Overview	1
Chapter 2: Current Testing Infrastructure vs. the Proposed Testing Methods	19
Chapter 3: .NET Namespaces and Classes for Software Testing	35
Chapter 4: .NET Reflection for Test Automation	63
Chapter 5: Spreadsheets and XML for Test Data Stores	99
Chapter 6: .NET CodeDom Namespace	149
Chapter 7: Generating Test Scripts	187
Chapter 8: Integration Testing	241
Chapter 9: Verification, Validation, and Presentation	277
Chapter 10: Finalizing the AutomatedTest Tool	317
Chapter 11: Updating the AutomatedTest Tool for Testing the Windows Registry	343
Chapter 12: Testing the AutomatedTest Tool	367
<i>Bibliography</i>	<i>389</i>
<i>Index</i>	<i>391</i>



Contents

<i>Introduction</i>	<i>xv</i>
Chapter 1 Software Testing: An Overview	1
Purpose of Software Testing	3
Expectations of Automated Software Testing	5
Automated Testing and XP Practice	6
Software Test Engineers	7
How to Automate Software Testing	7
Software Testing and Programming Languages	10
Using C# for Automation	15
Test Scripts	16
Summary	17
Chapter 2 Current Testing Infrastructure vs. the Proposed Testing Methods	19
Types of Software Testing	20
Automated Testing Tools in the Marketplace	22
DevPartner Studio from Compuware Corporation	22
Insure++ from Parasoft	23
Mercury Interactive	23
ObjectTester by ObjectSoftware, Inc.	24
Rational Software from IBM	25
Segue Software	26
TestWorks from Software Research, Inc.	26
Open Source Testing Tools	27
Comparing Testing Tools	28
The Proposed Software Testing Tool	29
Improvement of Unit Testing	29
Automated Generation of Testing Data	30
A Unique Approach for Integration Testing	30

	Ability to Be Updated with New Testing Capabilities	31
	Writing Test Script Based on Data	31
	Summary	33
Chapter 3	.NET Namespaces and Classes for Software Testing	35
	Discovering the Namespace of a Product	37
	Discovering the Namespace in Multiple Source Files	39
	Testing Classes and Namespaces	40
	Creating the AutomatedTest Tool Project	41
	C# Keywords: <i>using</i> and <i>namespace</i>	47
	Declaring <i>namespace</i> Directives with the Keyword <i>using</i>	48
	Primitive .NET Data Type and C# Representation	49
	Predefined .NET Namespaces for Automated Testing	50
	Retrieving Type Classes under Test	52
	Retrieving Types by Name	53
	Retrieving Types by Instance	58
	Retrieving Types from a Given Assembly	59
	Summary	61
Chapter 4	.NET Reflection for Test Automation	63
	The Basics of Reflection	64
	<i>System.Type</i> Class	65
	Discovering Type Information of a Variable	66
	Creating a Sample Class for Testing	67
	Gathering Test Information Using the <i>System.Type</i> Class	76
	Enumerating Method Parameters	80
	Testing Software with the .NET Reflection Namespace	82
	Loading an Assembly	83
	Loading Type Classes from an Assembly	89
	Dynamic Testing Invocation (Late Binding)	94
	Summary	97
Chapter 5	Spreadsheets and XML for Test Data Stores	99
	Working with MS Excel Objects in C#	100
	The Object Model of Excel	101
	The Excel Application Object	102
	Opening an MS Excel Application	102

Creating a Workbook Object	106
Workbook Properties	107
Workbook Methods	107
Workbook Events	110
Creating a Worksheet Object	111
Worksheet Properties	111
Worksheet Methods	112
Worksheet Events	114
Creating a Range Object	114
Range Properties	115
Range Methods	116
Building a Data Store for Automated Software Testing	118
Making a Utility Class	119
Collecting Test Information of Types	122
Creating an Excel Application	123
Testing against Expected Return Values	124
Implementing Data Stores	125
Completing the Method Inventory for the Types under Testing	132
Gathering Information for Test	136
Enabling XML Documentation for the Test Data Store	140
XML Programming	140
Testing with the Data Stored in the XML Document	146
Summary	147
Chapter 6	.NET CodeDom Namespace
Dynamic CodeDom Programming	150
The <i>System.CodeDom</i> Namespace	160
Types of <i>System.CodeDom</i> Namespace	162
The LastCodeDom Example	163
Summary	186
Chapter 7	Generating Test Scripts
Restarting the AutomatedTest Project	188
Starting the Test Script Generation	189
Using CodeDom to Write Test Scripts	191
Discovering Dependencies	197

	Programming the MS Excel Application	199
	Enumerating Type Information	203
	Enumerating Method Information	205
	Enumerating Parameter Information	216
	Closing the Test Script	222
	Executing Software Test Script	225
	Running the AutomatedTest	230
	Outcome of the AutomatedTest Project	231
	Summary	239
Chapter 8	Integration Testing	241
	Testing Parameters Passed by Objects	242
	Building a Higher-Level Module to Be Tested	244
	Building a Form for Manual Stubbing	250
	Code for Testing Parameters Passed by Objects	266
	Making Code Stubs with a Given Assembly	268
	Enumerating Assembly Information	269
	Finishing Testing Parameters Passed by Objects	272
	Summary	275
Chapter 9	Verification, Validation, and Presentation	277
	Automated Verification	278
	Verification by the Test Scripts	279
	Verification Decision	281
	Automated Validation	282
	Validation Scope of the Automated Test	283
	Creating Early Phase Test Scripts	284
	Presentation of the Test Results	306
	Passing a Test	307
	Failing a Test	309
	Summary	314
Chapter 10	Finalizing the AutomatedTest Tool	317
	Improving the Appearance of the AutomatedTest Project	318
	Automatically Generating .NET Project Components	320
	The <i>App.ico</i> and <i>AssemblyInfo.cs</i> Files	320
	The .NET *.csproj	324

Test Script Naming Convention	331	
Building Multiple Data Stores	334	
Auto-Execution of the Test Script Project	337	
Achieving Full Test Automation	340	
Summary	342	
Chapter 11	Updating the AutomatedTest Tool for Testing the Windows Registry	343
Windows Registry	344	
Accessing the Windows Registry	345	
RegEdit	345	
System Properties	347	
Command Prompt Window	349	
Windows Registry Programming	349	
Creating a Test Script to Test Software against the Windows Registry	354	
Adding CodeDom Methods to Update the AutomatedTest Tool	355	
Testing the AddAutoTestPath Project against the Windows Registry	361	
Summary	366	
Chapter 12	Testing the AutomatedTest Tool	367
Starting the Automated Test Tool	368	
Project Target Folder	370	
Result Target Folder	370	
.NET IDE Location	370	
Testing the <i>LowLevelObj.dll</i> Assembly	371	
Editing theData Store	372	
Reviewing Test Results	375	
Testing Parameters Passed by Objects	378	
Testing with Multiple Sets of Data Stores	381	
Testing Overloaded Methods	382	
Testing Parameters Passed by Arrays	384	
Summary	386	
<i>Bibliography</i>	389	
<i>Index</i>	391	