Contents

Chapter 1 Introduction 1
Who Should Read This Book? 2
What Software Testing Is—and Isn’t 3
What Is Different about Testing Object-Oriented Software? 5
Overview of Our Testing Approach 6
Test Early 7
Test Often 7
Test Enough 8
The Testing Perspective 8
Organization of This Book 8
Conventions Used in This Book 9
A Continuing Example—Brickles 10
Basic Brickles Components 11
Brickles Physics 11
Game Environment 14
## Chapter 2 The Testing Perspective  15

- Testing Perspective  15
- Object-Oriented Concepts  17
  - Object  18
  - Message  20
  - Interface  21
  - Class  22
  - Inheritance  31
  - Polymorphism  32
- Development Products  39
  - Analysis Models  40
  - Design Models  56
  - Source Code  59
- Summary  62

## Chapter 3 Planning for Testing  65

- A Development Process Overview  66
- A Testing-Process Overview  69
- Risk Analysis—A Tool for Testing  74
  - Risks  74
  - Risk Analysis  74
- A Testing Process  78
  - Planning Issues  78
  - Dimensions of Software Testing  78
  - Who Performs Testing?  80
  - Which Pieces Are Tested?  81
  - When Is Testing Performed?  82
  - How Is Testing Performed?  83
  - How Much Testing Is Adequate?  84
- Roles in the Testing Process  86
- A Detailed Set of Test Activities  87
- Planning Activities  91
  - Scheduling Testing Activities  91
  - Estimation  91
  - A Process for Testing Brickles  93
  - Document Templates  94
  - Test Metrics  103
- Summary  107
Chapter 4 Testing Analysis and Design Models 109

An Overview 110
Place in the Development Process 115
The Basics of Guided Inspection 116
   Evaluation Criteria 118
Organization of the Guided Inspection Activity 120
   Basic Roles 120
   Individual Inspection 121
Preparing for the Inspection 121
   Specifying the Inspection 121
Realistic Models 121
   Selecting Test Cases for the Inspection 123
Creating Test Cases 127
Completing Checklists 128
The Interactive Inspection Session 128
Testing Specific Types of Models 131
   Requirements Model 131
   Analysis Models 138
   Design Models 141
   Testing Again 151
Testing Models for Additional Qualities 151
Summary 154
   Model-Testing Checklist 155
Addendum: A Process Definition for Guided Inspection 157
   Steps in the Process 157
   Detailed Step Descriptions 157
   Roles in the Process 161

Chapter 5 Class Testing Basics 163

Class Testing 164
   Ways to Test a Class 164
   Dimensions of Class Testing 166
Constructing Test Cases 168
   Adequacy of Test Suites for a Class 179
Constructing a Test Driver 183
   Test Driver Requirements 186
   Tester Class Design 188
Summary 210
Chapter 6 Testing Interactions 213
Object Interactions 214
   Identifying Interactions 215
   Specifying Interactions 220
Testing Object Interactions 222
   Testing Collection Classes 222
   Testing Collaborator Classes 223
   The Interaction between Testing and Design Approach 224
Sampling Test Cases 225
Testing Off-the-Shelf Components 237
   Case Study in Component Acceptance Testing 238
Protocol Testing 241
Test Patterns 242
   Listener Test Pattern 242
   Specific Example 244
Testing Exceptions 245
   Testing Interactions at the System Level 247
Summary 248

Chapter 7 Testing Class Hierarchies 249
Inheritance in Object-Oriented Development 250
Subclass Test Requirements 250
   Refinement Possibilities 251
   Hierarchical, Incremental Testing 253
Organizing Testing Software 262
Testing Abstract Classes 263
Summary 266

Chapter 8 Testing Distributed Objects 269
Basic Concepts 270
Computational Models 271
   Concurrent 271
   Parallel 271
   Networked 271
   Distributed 272
<table>
<thead>
<tr>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Differences 272</td>
</tr>
<tr>
<td>Nondeterminism 272</td>
</tr>
<tr>
<td>Additional Infrastructure 273</td>
</tr>
<tr>
<td>Partial Failures 273</td>
</tr>
<tr>
<td>Time-Outs 274</td>
</tr>
<tr>
<td>Dynamic Nature of the Structure 274</td>
</tr>
<tr>
<td>Threads 274</td>
</tr>
<tr>
<td>Synchronization 274</td>
</tr>
<tr>
<td>Path Testing in Distributed Systems 275</td>
</tr>
<tr>
<td>Thread Models 278</td>
</tr>
<tr>
<td>Life-Cycle Testing 280</td>
</tr>
<tr>
<td>Models of Distribution 281</td>
</tr>
<tr>
<td>Basic Client/Server Model 281</td>
</tr>
<tr>
<td>Standard Models of Distribution 282</td>
</tr>
<tr>
<td>Comparisons and Implications 284</td>
</tr>
<tr>
<td>A Generic Distributed-Component Model 284</td>
</tr>
<tr>
<td>Basic Architecture 285</td>
</tr>
<tr>
<td>Local and Remote Interfaces 287</td>
</tr>
<tr>
<td>Specifying Distributed Objects 287</td>
</tr>
<tr>
<td>Interface Definition Language 287</td>
</tr>
<tr>
<td>Traditional Pre-Postconditions and Invariants 288</td>
</tr>
<tr>
<td>Temporal Logic 288</td>
</tr>
<tr>
<td>Temporal Test Patterns 291</td>
</tr>
<tr>
<td>Eventually(a) 291</td>
</tr>
<tr>
<td>Until(a,b) 292</td>
</tr>
<tr>
<td>Always 293</td>
</tr>
<tr>
<td>A Test Environment 293</td>
</tr>
<tr>
<td>Class Testing 293</td>
</tr>
<tr>
<td>Interaction Testing 295</td>
</tr>
<tr>
<td>Test Cases 295</td>
</tr>
<tr>
<td>Model-Specific tests 296</td>
</tr>
<tr>
<td>Testing Every Assumption 297</td>
</tr>
<tr>
<td>Infrastructure Tests 300</td>
</tr>
<tr>
<td>Logic-Specific Test Cases 301</td>
</tr>
<tr>
<td>The Ultimate Distributed System—The Internet 303</td>
</tr>
<tr>
<td>Web Servers 304</td>
</tr>
<tr>
<td>Life-Cycle Testing of Internet Applications 305</td>
</tr>
<tr>
<td>What Haven’t We Said? 306</td>
</tr>
<tr>
<td>Summary 306</td>
</tr>
</tbody>
</table>
Chapter 9 Testing Systems  309

Defining the System Test Plan  311
  Features Tested and Not Tested  311
  Test Suspension Criteria and Resumption
  Requirements  311
Complementary Strategies for Selecting Test Cases  313
  Use Profile  313
  ODC  314
Use Cases as Sources of Test Cases  315
  Constructing Use Profiles  316
  Using Scenarios to Construct Test Cases  317
  The Expected Results Section of a Test Case  319
  Brickles  320
Testing Incremental Projects  323
  Legacy Projects  323
Testing Multiple Representations  324
  What Needs to Be Tested?  326
  Testing against Functional Requirements  326
  Testing for Qualitative System Attributes  326
  Testing the System Deployment  327
  Testing after Deployment  328
  Testing Environment Interactions  328
  Test System Security  330
Types of Testing  331
  Stress Testing  331
  Life-Cycle Testing  331
  Performance Testing  333
Testing Different Types of Systems  334
  Reactive Systems  334
  Embedded Systems  335
  Multitiered Systems  336
  Distributed Systems  338
Measuring Test Coverage  338
  What Is to Be Covered?  338
  When Is Coverage Measured?  339
  When Is Coverage Used?  339
  ODC—Defect Impacts  339
  More Examples  341
Summary  341
Chapter 10 Components, Frameworks, and Product Lines 343

Component Models 344
  Enterprise JavaBeans Component Model 345
  Testing Components versus Objects 346
  Component Test Processes 348
  Test Cases Based on Interfaces 349
  Case Study—A GameBoard Component 351

Frameworks 359
  Basic Issues 359
  Framework Testing Processes 360
  Inspecting a Framework 360
  Structuring Test Cases to Support a Framework 361

Product Lines 362
  Testing at the Organizational Management Level 362
  Testing at the Technical Management Level 363
  Testing at the Software Engineering Level 363
  Testing in a Product Line Project 363
  Future 364

Summary 364

Chapter 11 Conclusion 367

Organization and Process 367
  Standards 368
  Software Infrastructure 370
  Techniques 370
  Risks 371
  Brickles 371
  Finally 373

Bibliography 375