Contents

Foreword by Ivar Jacobson xv
Foreword by John Vlissides xvii
Preface xix

CHAPTER 1 Design Concepts 1
Object Machinery 2
Roles 3
Object Role Stereotypes 4
Roles, Responsibilities, and Collaborations 5
Object Contracts 7

Conditions-of-Use and Aftereffect Guarantees 8
Domain Objects 8
Application-Specific Objects 10
Interfaces 12
Classes 13

Two Roles 13
Composition 16
Contents

Strategies for Assigning Responsibilities 125
  Recording Responsibilities 126
  Making Initial Assignments 128
  Getting Unstuck 138

Implementing Objects and Responsibilities 140
Testing Your Candidates' Quality 145
Summary 146
Further Reading 146

CHAPTER 5 Collaborations 149

What Is Object Collaboration? 150
  Preparing for Collaboration 150
  Recording Candidate Collaborations 151

The Design Story for the Speak for Me Software 152

Collaboration Options 153
  Who's In Control? 155
  How Much Should Objects Trust One Another? 155

Strategies for Identifying Collaborations 158
  Looking at an Individual Object's Role: Stereotypes Imply Collaborations 159
  Looking at Individual Responsibilities: They Imply Collaborations 166
  Designing the Details of a Complex Responsibility 168
  Designing Collaborations for a Specific Task 169
  Identifying Applicable Patterns 170
  Identifying How Architecture Influences Collaborations 172
  Solving Problems in Collaborations 173
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Control Style</td>
<td>195</td>
</tr>
<tr>
<td></td>
<td>What Is Control Style?</td>
<td>196</td>
</tr>
<tr>
<td></td>
<td>Control Style Options</td>
<td>197</td>
</tr>
<tr>
<td></td>
<td>Making Trade-offs</td>
<td>198</td>
</tr>
<tr>
<td></td>
<td>Centralizing Control</td>
<td>198</td>
</tr>
<tr>
<td></td>
<td>Delegating Control</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td>The Limits of Control Decisions</td>
<td>201</td>
</tr>
<tr>
<td></td>
<td>Developing Control Centers</td>
<td>205</td>
</tr>
<tr>
<td></td>
<td>A Case Study: Control Style for External User Events</td>
<td>206</td>
</tr>
<tr>
<td></td>
<td>Centralizing Control in the MessageBuilder</td>
<td>208</td>
</tr>
<tr>
<td></td>
<td>Refactoring Decision Making into State Methods within the MessageBuilder</td>
<td>220</td>
</tr>
<tr>
<td></td>
<td>Abstracting Away Decisions</td>
<td>221</td>
</tr>
<tr>
<td></td>
<td>Delegating More Responsibility</td>
<td>224</td>
</tr>
<tr>
<td></td>
<td>Designing the Control Style for the Guessing Neighborhood</td>
<td>225</td>
</tr>
<tr>
<td></td>
<td>Designing a Similar Control Center: Can We Be Consistent?</td>
<td>230</td>
</tr>
<tr>
<td></td>
<td>Summary</td>
<td>237</td>
</tr>
</tbody>
</table>
CHAPTER 7  Describing Collaborations  239
   Telling Collaboration Stories  240
   A Strategy for Developing a Collaboration Story  241
   Establishing Scope, Depth, and Tone  242
   Listing What You Will Cover  243
   Deciding on the Level of Detail  243
      Showing a Bird's-Eye View  244
      Showing Collaborators Only  245
      Showing a Sequence of Interactions Among Collaborators  250
      Showing an In-Depth View  251
      Showing a Focused Interaction  253
      Showing an Implementation View  254
      Showing How to Adapt a Collaboration  254
      Where UML Diagrams Fall Short  258
   Choosing the Appropriate Form  263
   Tell It, Draw It, Describe It: Guidelines  264
   Organizing Your Work  270
      Adding Emphasis  271
      Unfolding Your Story  271
      Understanding What's Fundamental  272
      Putting It All Together  273
   Preserving Stories  274
   Summary  275
   Further Reading  275

CHAPTER 8  Reliable Collaborations  277
   Understanding the Consequences of Failure  278
   Increasing Your System's Reliability  280
   Determining Where Collaborations Can Be Trusted  280
Contents

Trusted Versus Untrusted Collaborations 281
Implications of Trust 284

Identifying Collaborations to Be Made
Reliable 285
What Use Cases Tell Us 286
Distinguish Between Exceptions and Errors 287
Object Exceptions Versus Use Case Exceptions 288
Object Exception Basics 288
Exception- and Error-Handling Strategies 294
Determining Who Should Take Action 296

Designing a Solution 299
Brainstorm Exception Conditions 299
Limit Your Scope 300
Record Exception-Handling Policies 302

Documenting Your Exception-Handling Designs 303
Specifying Formal Contracts 307

Reviewing Your Design 311
Summary 312
Further Reading 313

CHAPTER 9 Flexibility 315
What Does It Mean to Be Flexible? 316
Degrees of Flexibility 317
The Consequences of a Flexible Solution 319
Nailing Down Flexibility Requirements 320
Recording Variations 324
Contents

Variations and Realizations 327
   Identifying the Impact of a Variation 328
   Exploring Strategies for Realizing Flexibility 329
   Using Templates and Hooks to Support Variations 330

The Role of Patterns in Flexible Designs 338
   Varying an Object's Behavior with the Strategy Pattern 338
   Hiding Interacting Objects with Mediator 339
   Making a Predefined Object or System Fit Using Adapter 340
   How Do Patterns Increase Flexibility? 340

How to Document a Flexible Design 342
   Consider Your Audience 344
   Describing How to Make a Variation 345

Changing a Working System's Design 350

Summary 352

Further Reading 353

CHAPTER 10 On Design 355

The Nature of Software Design 356

Tackling Core Design Problems 357

Frame the Problem 358

Dealing with Revealing Design Problems 361
   A Story About Managing Shared Information 362
   A Story About Connection Problem Complexity 363
   A Story About a Design Problem That Never Got Easier 364
   Can Revealing Problems Be Wicked, Too? 365
Strategies for Solving Revealing Problems 366

Redefining the Problem 368
Synthesizing a Solution 369

Working on the Rest 370
Designing Responsibly 371
Further Reading 374

Bibliography 375

Index 381