Contents

Foreword by Peter Freeman xv
Foreword by Bran Selic xvii
Preface xix

PART I UML NOTATION, DESIGN CONCEPTS, TECHNOLOGY, LIFE CYCLES, AND METHODS .......... 1
1 Introduction ....................................................... 3
  1.1 Object-Oriented Methods and the Unified Modeling Language ...................... 4
  1.2 Method and Notation ........................................... 5
  1.3 Concurrent Applications ...................................... 6
  1.4 Real-Time Systems and Applications ................................ 8
  1.5 Distributed Systems and Applications ................................ 10
  1.6 Summary .................................................... 11

2 Overview of UML Notation ................................. 13
  2.1 UML Diagrams .............................................. 13
  2.2 Use Case Diagrams ......................................... 14
  2.3 UML Notation for Classes and Objects ................................ 14
  2.4 Class Diagrams ............................................. 15
  2.5 Interaction Diagrams ....................................... 17
  2.6 Statechart Diagrams ....................................... 19
  2.7 Packages .................................................. 20
  2.8 Concurrent Collaboration Diagrams ................................ 21
### CONTENTS

2.9 Deployment Diagrams .............................................. 23  
2.10 UML Extension Mechanisms .............................. 24  
2.11 The UML as a Standard ........................................ 25  
2.12 Summary .......................................................... 26  

3 Software Design and Architecture Concepts ................. 27  
3.1 Object-Oriented Concepts ........................................ 27  
3.2 Information Hiding ............................................... 30  
3.3 Inheritance ......................................................... 36  
3.4 Active and Passive Objects ...................................... 37  
3.5 Concurrent Processing ............................................ 38  
3.6 Cooperation between Concurrent Tasks ...................... 40  
3.7 Information Hiding Applied to Access Synchronization ... 49  
3.8 Monitors ............................................................. 51  
3.9 Design Patterns .................................................... 53  
3.10 Software Architecture and Component-Based Systems .... 55  
3.11 Summary ............................................................ 56  

4 Concurrent and Distributed System Technology .......... 57  
4.1 Environments for Concurrent Processing .................. 57  
4.2 Runtime Support for Multiprogramming and Multiprocessing Environments .............................. 60  
4.3 Task Scheduling ...................................................... 63  
4.4 Operating System Input/Output Considerations .......... 65  
4.5 Client/Server and Distributed System Technology ...... 68  
4.6 World Wide Web Technology ..................................... 73  
4.7 Distributed Operating System Services ...................... 75  
4.8 Middleware .......................................................... 78  
4.9 Common Object Request Broker Architecture (CORBA) .. 81  
4.10 Other Component Technologies ............................... 85  
4.11 Transaction Processing Systems ............................. 86  
4.12 Summary ............................................................ 88  

5 Software Life Cycles and Methods .............................. 91  
5.1 Software Life Cycle Approaches .............................. 91  
5.2 Design Verification and Validation ......................... 98  
5.3 Software Testing ................................................... 99  
5.4 Evolution of Software Design Methods ..................... 101
### 5.5 Evolution of Object-Oriented Analysis and Design Methods

103

### 5.6 Survey of Concurrent and Real-Time Design Methods

105

### 5.7 Summary

106

### PART II COMET: CONCURRENT OBJECT MODELING AND ARCHITECTURAL DESIGN WITH UML

107

#### 6 Overview of COMET

109

##### 6.1 COMET Object-Oriented Software Life Cycle

109

##### 6.2 Comparison of the COMET Life Cycle with Other Software Processes

112

##### 6.3 Requirements, Analysis, and Design Models

113

##### 6.4 The COMET in a Nutshell

115

##### 6.5 Summary

118

#### 7 Use Case Modeling

119

##### 7.1 Use Cases

119

##### 7.2 Actors

120

##### 7.3 Actors, Roles, and Users

123

##### 7.4 Identifying Use Cases

123

##### 7.5 Documenting Use Cases in the Use Case Model

124

##### 7.6 Examples of Use Cases

125

##### 7.7 Use Case Relationships

130

##### 7.8 Use Case Packages

134

##### 7.9 Summary

135

#### 8 Static Modeling

137

##### 8.1 Associations between Classes

137

##### 8.2 Composition and Aggregation Hierarchies

145

##### 8.3 Generalization/Specialization Hierarchy

147

##### 8.4 Constraints

149

##### 8.5 Static Modeling and the UML

149

##### 8.6 Static Modeling of the System Context

152

##### 8.7 Static Modeling of Entity Classes

155

##### 8.8 Summary

157

#### 9 Object and Class Structuring

159

##### 9.1 Object Structuring Criteria

160

##### 9.2 Categorization of Application Classes

160
12 Software Architecture Design ........................................ 253
  12.1 Software Architectural Styles .................................. 253
  12.2 System Decomposition Issues ................................. 257
  12.3 Guidelines for Determining Subsystems ...................... 259
  12.4 Consolidated Collaboration Diagrams ......................... 260
  12.5 Subsystem Software Architecture .............................. 261
  12.6 Separation of Concerns in Subsystem Design ................ 261
  12.7 Subsystem Structuring Criteria ............................... 265
  12.8 Examples of Subsystem Decomposition ....................... 269
  12.9 Static Modeling at the Design Level ......................... 270
  12.10 Summary ..................................................... 274

13 Architectural Design of Distributed Applications .............. 275
  13.1 Configurable Architectures and Software Components ......... 276
  13.2 Steps in Designing Distributed Applications ................. 276
  13.3 System Decomposition ........................................ 277
  13.4 Designing Subsystem Interfaces ............................... 283
  13.5 Transaction Management ....................................... 292
  13.6 Design of Server Subsystems .................................. 295
  13.7 Distribution of Data ........................................... 300
  13.8 System Configuration .......................................... 301
  13.9 Summary ..................................................... 303

14 Task Structuring .................................................... 305
  14.1 Concurrent Task Structuring Issues ......................... 306
  14.2 Task Structuring Categories .................................. 307
  14.3 I/O Task Structuring Criteria ................................ 308
  14.4 Internal Task Structuring Criteria ........................... 317
  14.5 Task Priority Criteria ........................................ 324
  14.6 Task Clustering Criteria ...................................... 325
  14.7 Design Restructuring by Using Task Inversion ............... 337
  14.8 Developing the Task Architecture .............................. 341
  14.9 Task Communication and Synchronization ..................... 345
  14.10 Task Behavior Specifications ................................. 354
  14.11 Summary ..................................................... 359
15 Class Design ....................................................... 361
  15.1 Designing Information Hiding Classes .................. 361
  15.2 Designing Class Operations ............................ 362
  15.3 Data Abstraction Classes ............................... 367
  15.4 Device Interface Classes ................................ 369
  15.5 State-Dependent Classes ............................... 375
  15.6 Algorithm Hiding Classes .............................. 378
  15.7 User Interface Classes .................................. 378
  15.8 Business Logic Classes .................................. 381
  15.9 Database Wrapper Classes .............................. 383
  15.10 Software Decision Classes ............................ 384
  15.11 Inheritance in Design .................................. 386
  15.12 Examples of Inheritance ............................... 387
  15.13 Class Interface Specifications ....................... 393
  15.14 Summary .................................................. 396

16 Detailed Software Design ........................................ 397
  16.1 Design of Composite Tasks ............................. 397
  16.2 Synchronization of Access to Classes .................. 405
  16.3 Designing Connectors for Inter-Task Communication .... 414
  16.4 Task Event Sequencing Logic ......................... 420
  16.5 Summary .................................................. 421

17 Performance Analysis of Concurrent Real-Time Software Designs ................. 423
  17.1 Real-Time Scheduling Theory ......................... 423
  17.2 Advanced Real-Time Scheduling Theory .................. 431
  17.3 Performance Analysis Using Event Sequence Analysis .... 436
  17.4 Performance Analysis Using Real-Time Scheduling Theory and Event Sequence Analysis .... 437
  17.5 Example of Performance Analysis Using Event Sequence Analysis .... 438
  17.6 Example of Performance Analysis Using Real-Time Scheduling Theory .................. 442
  17.7 Example of Performance Analysis Using Real-Time Scheduling Theory and Event Sequence Analysis .... 444
  17.8 Design Restructuring .................................. 452
  17.9 Estimation and Measurement of Performance Parameters .... 453
  17.10 Summary .................................................. 454
## Part III Case Studies in Concurrent, Distributed, and Real-Time Application Design

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Sections</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>Elevator Control System Case Study</td>
<td>18.1 Problem Description, 18.2 Use Case Model, 18.3 Static Model of the Problem Domain, 18.4 Object Structuring, 18.5 Dynamic Model, 18.6 Statechart Model, 18.7 Consolidation of Collaboration Diagrams, 18.8 Subsystem Structuring, 18.9 Structuring System into Tasks, 18.10 Design of Distributed Elevator Control System, 18.11 Design of Information Hiding Classes, 18.12 Developing Detailed Software Design, 18.13 Target System Configuration, 18.14 Performance Analysis of Non-Distributed Elevator Control System, 18.15 Performance Analysis of Distributed Elevator Control System</td>
</tr>
<tr>
<td>20</td>
<td>Cruise Control and Monitoring System Case Study</td>
<td>20.1 Problem Description</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>20.2</td>
<td>Use Case Model</td>
<td>597</td>
</tr>
<tr>
<td>20.3</td>
<td>Use Case Descriptions</td>
<td>600</td>
</tr>
<tr>
<td>20.4</td>
<td>Problem Domain Static Modeling</td>
<td>605</td>
</tr>
<tr>
<td>20.5</td>
<td>Dynamic Modeling</td>
<td>608</td>
</tr>
<tr>
<td>20.6</td>
<td>Subsystem Structuring</td>
<td>623</td>
</tr>
<tr>
<td>20.7</td>
<td>Refined Static Modeling</td>
<td>632</td>
</tr>
<tr>
<td>20.8</td>
<td>Structuring the System into Tasks</td>
<td>635</td>
</tr>
<tr>
<td>20.9</td>
<td>Information Hiding Class Design</td>
<td>657</td>
</tr>
<tr>
<td>20.10</td>
<td>Developing Detailed Software Design</td>
<td>665</td>
</tr>
<tr>
<td>20.11</td>
<td>Software Architecture of Distributed Automobile System</td>
<td>671</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>Distributed Factory Automation System Case Study</td>
<td>673</td>
</tr>
<tr>
<td>21.1</td>
<td>Problem Description</td>
<td>673</td>
</tr>
<tr>
<td>21.2</td>
<td>Use Case Model</td>
<td>675</td>
</tr>
<tr>
<td>21.3</td>
<td>Conceptual Static Model of the Problem Domain</td>
<td>678</td>
</tr>
<tr>
<td>21.4</td>
<td>Object Structuring</td>
<td>680</td>
</tr>
<tr>
<td>21.5</td>
<td>Dynamic Model</td>
<td>682</td>
</tr>
<tr>
<td>21.6</td>
<td>Subsystem Structuring</td>
<td>698</td>
</tr>
<tr>
<td>21.7</td>
<td>Distributed Software Architecture</td>
<td>702</td>
</tr>
<tr>
<td>21.8</td>
<td>System Configuration</td>
<td>710</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>Electronic Commerce System Case Study</td>
<td>713</td>
</tr>
<tr>
<td>22.1</td>
<td>Electronic Commerce Problem</td>
<td>713</td>
</tr>
<tr>
<td>22.2</td>
<td>Use Case Model</td>
<td>714</td>
</tr>
<tr>
<td>22.3</td>
<td>Agent Support for Electronic Commerce System</td>
<td>715</td>
</tr>
<tr>
<td>22.4</td>
<td>Object Broker Support for Electronic Commerce System</td>
<td>717</td>
</tr>
<tr>
<td>22.5</td>
<td>Static Modeling of the Problem Domain</td>
<td>718</td>
</tr>
<tr>
<td>22.6</td>
<td>Collaboration Model</td>
<td>719</td>
</tr>
<tr>
<td>22.7</td>
<td>Distributed Software Architecture</td>
<td>728</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appendix A</td>
<td>Conventions and Alternative Notations.</td>
<td>737</td>
</tr>
<tr>
<td>A.1</td>
<td>Conventions Used in This Book</td>
<td>737</td>
</tr>
<tr>
<td>A.2</td>
<td>Alternative Notation for Stereotypes</td>
<td>739</td>
</tr>
<tr>
<td>A.3</td>
<td>Alternative Notation for Active Objects</td>
<td>740</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glossary</td>
<td></td>
<td>743</td>
</tr>
<tr>
<td>Bibliography</td>
<td></td>
<td>757</td>
</tr>
<tr>
<td>Index</td>
<td></td>
<td>767</td>
</tr>
</tbody>
</table>