## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preface</td>
<td>xi</td>
</tr>
<tr>
<td>Contributors</td>
<td>xv</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td>xvi</td>
</tr>
<tr>
<td>Introduction</td>
<td>xviii</td>
</tr>
<tr>
<td>Notes for Educators: AMA Teaching Methods</td>
<td>xxix</td>
</tr>
</tbody>
</table>

### Chapter 1

**Collaborative Engineering**

1. Management Style as an Important Part of the Development Process: True Leaders Versus “Pure” Managers 2
2. Development Methodologies: Capability Maturity Model and More 4
4. Six Sigma 5
5. Distributed Collaborative Development 8
6. 24×7 Distributed Development Practices 9
7. Steps in the Process 10
8. Basic Steps of the Development Process with an Object-Oriented Approach 21
9. Learn by Example: Compare OOP and Procedural Programming 24
10. UML Notations 26
11. Example of Object-Oriented Analysis: Create an OMD for Document Services 26
12. Create the DocumentService Model 26
13. Architecture Steps: Find Playground-Tiers for Your Objects 28
14. From Single-User to Client-Server and Multi-Tier Architecture Models 33
15. Basic Design Steps and Rules 35
16. Instead of a Summary: How Direct Access to Products and Services Improves the Balance of Supply and Demand 40
## Chapter 2

### Software Architecture and Integration Technologies

- Software Architecture—The Center Place of Software Development... 42
- Architectural Elements, Styles, Views, and Languages... 43
- Programming Styles... 48
- Integration Technologies... 49
- Object Linking and Embedding (OLE) and ActiveX... 49
- CORBA and IDL... 51
- Microsoft's Distributed COM Architecture... 52
- Java Technology Architecture... 53
- Java Applet... 53
- The Java Bean Architecture... 53
- J2EE Architecture... 55
- Java Server Pages... 55
- The Enterprise JavaBeans Architecture... 56
- EJB Components and Containers... 56
- The Java RMI Architecture... 57
- The Bridge from Java to CORBA Technology... 57
- XML Web Services... 59
- An Example of an XML-Based Service API... 60
- Additional Benefits: Ability to Add or Change Services at Run-Time... 63
- How Can We Register a New Web Service?... 64
- Is There a Mechanism to Pack Both Data and Services into a Message?... 64
- How Do Software Vendors Support Web Services?... 64

---

## Chapter 3

### From a Specific Task to “Integration-Ready” Components

- User Requirements, Version 1: The “News Watch” Applet... 70
- User Requirements, Version 2: The Reusable NewsLine Component... 79
- User Requirements, Version 3: View Multiple Web Information Channels in the NewsLine Component... 88
- Integration-Ready Service Components and Extensible Service Containers... 100
- An XML-Based Configuration File... 103
- Start from the Parameters for a Single Component... 105
- How Would We Use XML-Based Parameters while Building Components?... 106
- Event-Handling Procedure... 114
- What Is the Dispatch() Method for and How Do We Define Its Function?... 115
- Provide the Possibility of Interactive Components (Event Handling)... 125
- The ControlComponent Class Description... 127
- Reuse, Not Abuse... 130
# Contents

## Chapter 4
**Integration with Voice**

- What Is the Base for Creating a Voice Component? 133
- How Are Voice Components Coded? 140

## Chapter 5
**An Introduction to Knowledge Technologies**

- Ontology 152
- DAML+OIL: A Semantic Markup Language for Web Resources 153
- Topic Maps 156
- Data-Mining Process and Methods 160
- Frames and Slots 161
- The CycL Language 163
- How to Begin with OpenCyc 173

## Chapter 6
**Write Once**

- Multiple Types of Data Storage 195
- Control Systems and Controllers 199
- Document-Handling Services 203

## Chapter 7
**The New Generation of Client–Server Software**

- What Are the Best Ways to Provide Client Functionality? 225
- Thin Clients 226
- Multiple Scenarios 235
- Synchronous or Asynchronous Client-Server Communication 243
- Example of XML Multiple-Screen Scenario 244
- Good Performance Follows Good Design 251
- Keep a Stable API while Changing Communication Mechanisms 252
- Add Open Office Features to Rich Clients 254
- How Much Abstraction Is Too Much? 255

## Chapter 8
**Wireless Technologies**

- Wireless TDMA, CDMA, GSM, and Other Standards 258
- 802.11 and WLANs 261
- Bluetooth Technology 262
Table of Contents

Chapter 9
Programming Wireless Application Protocol Applications 267

Rethink the Existing Web Page Paradigm in WAP Terms 268
What Is a Presentation Factory and How Do You Create One? 269
Programming WAP/WML Pages 270
Secure Transactions with WML Factory 271
Is the Data Size Too Big for a Device? Not a Problem! 279
WAP Push 281
WAP Devices and Web Services 284

Chapter 10
A Single JavaCard Identity Key for All Doors and Services 286

What Is a Smart Card? 286
What Is a JavaCard? 287
Why Do We Use Multiple Keys? 287
Can We Have a Single “Identity Key”? 288
How to Program JavaCards 288
What Are JavaCard Programming’s Limitations? 290
The javacard.framework Package for JavaCard Programming 290
Writing a Sample JavaCard Applet: VirtualCurrency 291

Chapter 11
The J2ME Family 306

The MIDP 308
How Do We Display Screens and Handle Events? 308
Multimedia on Wireless 309
What Are MIDlets? 310
Can We Store Data on Mobile Devices? 310
MIDlet Security 311
Can We Push from a Server to the MIDP Device? 311
Wireless Messaging with Short Message Service and Other Protocols: An Important Component in Our Application 312
Wireless Messaging Client Application 314

Chapter 12
Speech Technologies on the Way to a Natural User Interface 338

What Is a Natural User Interface? 338
Speaking with Style 339
Java Speech API Markup Language 341
# Table of Contents

## Chapter 13

**Integration with Knowledge**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge Integration Participants, Processes, and Products</td>
<td>388</td>
</tr>
<tr>
<td>Connect Software and Knowledge Technologies</td>
<td>390</td>
</tr>
<tr>
<td>What Are the Main Goals of the Knowledge Connector Package?</td>
<td>392</td>
</tr>
<tr>
<td>Object Model Diagram</td>
<td>393</td>
</tr>
<tr>
<td>Formatting and Presentation Layers</td>
<td>393</td>
</tr>
<tr>
<td>The Magic of Service Invocation</td>
<td>394</td>
</tr>
<tr>
<td>What Does It Mean to Play a Scenario?</td>
<td>415</td>
</tr>
<tr>
<td>Do You Want to Be a Scenario Writer?</td>
<td>416</td>
</tr>
<tr>
<td>Application Scenario Language</td>
<td>416</td>
</tr>
<tr>
<td>Installing and Running the Package</td>
<td>438</td>
</tr>
</tbody>
</table>

## Chapter 14

**Distributed Life in the JXTA and Jini Communities**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distributed Processing and the Flat World of XML</td>
<td>446</td>
</tr>
<tr>
<td>What Is JXTA?</td>
<td>449</td>
</tr>
<tr>
<td>Jini</td>
<td>463</td>
</tr>
<tr>
<td>JXTA and Jini: Just Neighbors or Collaborators?</td>
<td>466</td>
</tr>
</tbody>
</table>

## Appendix 1

**Java and C#: A Saga of Siblings**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Java Virtual Machine and Common Language Run-Time</td>
<td>468</td>
</tr>
<tr>
<td>From Basics to the Next Level on the Java/C# Programming Trail</td>
<td>484</td>
</tr>
</tbody>
</table>

## Appendix 2

**XML and Web Services**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>XML Extends the Web and Builds a Playground for Its Children</td>
<td>539</td>
</tr>
<tr>
<td>XML Describes Business Rules and Data Structures; XSLT and X Path</td>
<td>539</td>
</tr>
<tr>
<td>Describe Their Transformations</td>
<td>539</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
</tr>
<tr>
<td>---------------------------</td>
<td></td>
</tr>
<tr>
<td>XML Provides Direct Hooks to Services on the Web with SOAP, WSDL, and UDDI</td>
<td>540</td>
</tr>
<tr>
<td>Interactive Web with XForms</td>
<td>540</td>
</tr>
<tr>
<td>XML in Voice Applications</td>
<td>540</td>
</tr>
<tr>
<td>XML Drives Semantic Web and Knowledge Technologies</td>
<td>541</td>
</tr>
<tr>
<td>XML Web Services</td>
<td>541</td>
</tr>
<tr>
<td>Web Services at Work</td>
<td>541</td>
</tr>
<tr>
<td>Encode Service Requests with SOAP</td>
<td>542</td>
</tr>
<tr>
<td>Describe Web Services with WSDL</td>
<td>542</td>
</tr>
<tr>
<td>Publish and Discover Web Services with UDDI</td>
<td>544</td>
</tr>
<tr>
<td>Business Process Execution Language for Web Services</td>
<td>544</td>
</tr>
</tbody>
</table>

**Appendix 3**

**Source Examples**

<table>
<thead>
<tr>
<th>Source Examples</th>
<th>551</th>
</tr>
</thead>
<tbody>
<tr>
<td>Getting into Collaborative Services</td>
<td>551</td>
</tr>
<tr>
<td>The MIME Multipart/Related Content-Type; RFC 1867</td>
<td>554</td>
</tr>
<tr>
<td>Working with Geographical Information Systems</td>
<td>570</td>
</tr>
<tr>
<td>Reading AutoCAD Vector Graphics</td>
<td>577</td>
</tr>
<tr>
<td>Instant Screen Share</td>
<td>581</td>
</tr>
<tr>
<td>Instant Voice Share with JMF</td>
<td>581</td>
</tr>
<tr>
<td>Java Messaging Services (JMS): A New Way to Think about Applications</td>
<td>587</td>
</tr>
<tr>
<td>Create Speech Recognition and TTS Applications in C#</td>
<td>589</td>
</tr>
<tr>
<td>Fight Email Spam and Increase Email Server Efficiency</td>
<td>590</td>
</tr>
</tbody>
</table>

**Index**

| Index | 599 |