

# Contents

Foreword .....	iii
Preface .....	v

---

## Part 1. CORBA Meets Java ..... 1



### Chapter 1. Client/Server, CORBA-Style ..... 3

Distributed Objects, CORBA-Style .....	4
What Is a Distributed CORBA Object? .....	4
Everything Is in IDL .....	5
CORBA Components: From System Objects to Business Objects .....	6
OMG's Object Management Architecture .....	7
The Object Request Broker (ORB) .....	7
The Anatomy of a CORBA 2.0 ORB .....	10
CORBA 2.0: The Intergalactic ORB .....	14
CORBA services .....	17
Object Services: Build-to-Order Middleware .....	19
CORBA facilities .....	20
CORBA Business Objects .....	21
Cooperating Business Objects .....	22
The Anatomy of a CORBA Business Object .....	23
The Anatomy of a Client/Server Business Object .....	25
CORBA Component Nirvana .....	25
3-Tier Client/Server, Object-Style .....	27
From Business Objects to CORBA Beans .....	28
CORBA 3.0: The Next Generation .....	30
Conclusion .....	32



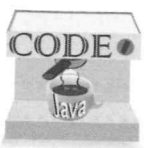
### Chapter 2. The Object Web: CORBA Meets Java ..... 33

CORBA Meets Java .....	34
Why the Shotguns? .....	34
The Evolution of the Web .....	36

CGI: The Protocol That Won't Go Away .....	36
CORBA/Java and the Web: Look Ma, No Cookies .....	38
How CORBA/Java Augment Today's Web .....	39
Life Without CGI .....	40
Scalable Servers .....	40
What CORBA Brings to Java .....	40
What Java Brings to CORBA .....	42
The CORBA/Java Object Web .....	43
CORBA and JavaBeans .....	43
The 3-Tier CORBA/Java Object Web .....	44
Meet the Players .....	46
The Next Client/Server Wave .....	48
Conclusion .....	50

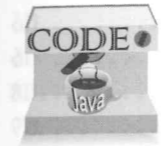
**Chapter 3. Meet the CORBA/Java ORBs .....** 51

The New CORBA Java/IDL Mapping .....	52
What These CORBA/Java Standards Cover .....	52
The Just-in-Time CORBA/Java ORB .....	53
The CORBA/Java ORBs .....	53
JavaSoft's Java IDL .....	54
Iona's OrbixWeb 3.0 .....	55
Borland/Visigenic's VisiBroker for Java 3.1 .....	56
Which Java ORB? .....	57
Conclusion .....	60

**Part 2. Core CORBA/Java .....** 61**Chapter 4. Your First CORBA Program .....** 63

The Static CORBA .....	64
CORBA Method Invocations: Static Versus Dynamic .....	64
CORBA Static Method Invocations: From IDL to Interface Stubs .....	67
Your First CORBA Program .....	68
The Count IDL .....	69
Mapping CORBA IDL to Java .....	70
The Server Side of Count .....	72
The Client Side of Count .....	75

Compile the Java Code Using Symantec Visual Café .....	78
Run the Client/Server Program .....	80
Looking at Some Test Results .....	81
Local Versus Remote Pings .....	81
Symantec JIT Versus Sun JDK .....	82
Conclusion .....	82

**Chapter 5. ORBlets Meet Applets .....** 83

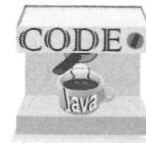
Applets 101 .....	84
Applet Life Cycle Management .....	84
The HTML Applet Tag .....	85
Your First CORBA-Enabled Applet .....	86
The Count IDL .....	86
So, What Exactly Does This Applet Do? .....	86
The Client Applet Code .....	88
Compile the Java Applet .....	91
Create a Web Page .....	92
Test the Applet .....	92
Run the Client/Server Program .....	92
Let's Look at Some Test Results .....	94
Local Versus Remote Pings .....	94
Applets Versus Applications .....	95
Conclusion .....	95

**Chapter 6. Java ORBs Meet C++ ORBs .....** 97

The C++ Count Program .....	98
The Count IDL .....	98
Mapping CORBA IDL to C++ .....	98
The Server Side of the C++ Count .....	100
The Client Side of C++ Count .....	104
Compile the C++ Code .....	106
Run the Client/Server Program .....	106
Looking At Some Test Results .....	107
Local Versus Remote Pings .....	107
C++ Client to Java Count Server .....	108
Java Client to C++ Count Server .....	109
Conclusion .....	111



Part 3. The Dynamic CORBA . . . . . 113



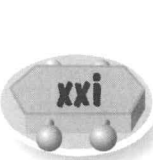
Chapter 7. The Portable Count . . . . . 115

- The CORBA Naming Service. . . . . 116
  - Object Naming in a Nutshell . . . . . 116
  - What's in a CORBA Object Name? . . . . . 116
  - How Does It Work? . . . . . 118
  - Naming Service: The CORBA IDL . . . . . 119
  - Naming Service: The Java Interfaces . . . . . 121
  - Client/Server Naming Scenarios . . . . . 123
  - Scenario 1: Creating the Namespace . . . . . 124
  - Scenario 2: Finding Objects . . . . . 127
- The Portable Count. . . . . 128
  - The Portable Count Client. . . . . 128
  - The Portable Count Server . . . . . 130
- Compile The Client/Server Program . . . . . 133
- Looking at Some Test Results. . . . . 134
- Conclusion . . . . . 134



Chapter 8. The Dynamic Count . . . . . 135

- CORBA 101: Dynamic Invocations . . . . . 136
  - Dynamic Invocations: The Big Picture . . . . . 136
  - The Dynamic Invocation Interfaces . . . . . 138
  - Dynamic Invocation Scenarios . . . . . 140
  - Dynamic Invocation: The Do-It-Yourself Scenario . . . . . 141
  - Dynamic Invocation: The ORB-Can-Help Scenario . . . . . 142
  - Dynamic Invocation: The Yet-Another-Way Scenario . . . . . 144
- The Dynamic Count. . . . . 145
  - The Count IDL . . . . . 146
  - The Client Side of Count . . . . . 146
  - Compile the Java Code Using The Symantec Compiler . . . . . 149
  - Run the Client/Server Program . . . . . 150
- Looking At Some Test Results . . . . . 150
- When to Use Dynamic. . . . . 151
- Conclusion . . . . . 152



Chapter 9. MultiCount: The Jazzed-Up Count . . . . . 153

- The Design of MultiCount . . . . . 154
  - Meet the Players . . . . . 155
  - The MultiCount CORBA Interfaces . . . . . 156
  - A MultiCount Callback Scenario . . . . . 157
- The MultiConsole Applet . . . . . 159
  - What the Applet Looks Like. . . . . 160
  - The MultiConsole Applet HTML . . . . . 160
  - The MultiConsoleApplet Class . . . . . 161
- The Client . . . . . 165
  - Client Classes and Threads . . . . . 165
  - A Multithreaded Client/Server Scenario . . . . . 167
  - The Client Code . . . . . 168
- The Coordinator. . . . . 175
  - The Coordinator Classes . . . . . 176
  - The Coordinator Code. . . . . 176
- Compile the Client and Server Programs . . . . . 179
- Run the Client/Server Programs . . . . . 180
- Conclusion . . . . . 181

Part 4. CORBA and Its Competitors . . . . . 183



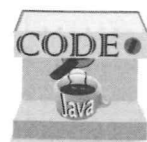
Chapter 10. Sockets Versus CORBA/Java ORBs . . . . . 185

- Berkeley Sockets 101 . . . . . 186
  - What Is a Socket? . . . . . 186
  - How Do You Like Your Socket? . . . . . 186
  - Socket Ports . . . . . 189
  - Java's InetAddress Class. . . . . 189
  - The Anatomy of a Sockets Exchange . . . . . 190
- Java Sockets 101 . . . . . 192
  - The Java Datagram Sockets Classes. . . . . 192
  - A Java Datagram Scenario. . . . . 195
  - The Java Stream Sockets Classes . . . . . 197
- Java Streams 101. . . . . 201
  - The Java Output Stream Classes. . . . . 203
  - The Java Input Stream Classes. . . . . 207
  - A Java Buffered-Stream Socket Scenario . . . . . 211

The Java Datagram Socket Count .....	215
The Datagram Socket Count Client .....	215
The Datagram Socket Count Server .....	217
The Buffered Socket Count .....	219
The Buffered Socket Count Client .....	219
The Buffered Socket Count Server .....	221
The Data Stream Socket Count .....	223
The Data Stream Socket Count Client .....	223
The Data Stream Socket Count Server .....	225
The Buffered Data Stream Count .....	227
Compile the Socket Counts .....	227
Run the Client/Server Programs .....	228
Looking at Some Test Results .....	228
Sockets Versus CORBA IIOP .....	229
Conclusion .....	230

**Chapter 11. HTTP/CGI Versus CORBA/Java ORBs .....** 231

HTTP 101 .....	232
So What Exactly Is HTTP? .....	232
HTTP Data Representations .....	233
So What Does an HTTP Request Look Like? .....	233
So What Does an HTTP Response Look Like? .....	234
CGI 101 .....	236
HTML Forms .....	238
3-Tier Client/Server, CGI-Style .....	238
CGI and State .....	241
The HTTP/CGI Count .....	243
The CGI Count Client Applet .....	244
The CGI Count Server .....	248
Compile the Client/Server Program .....	249
Create a Web Page .....	250
Run the Client/Server Programs .....	250
Looking At Some Test Results .....	251
Conclusion .....	253

**Chapter 12. Servlets Versus CORBA/Java ORBs .....** 255

Servlets 101 .....	256
The Life Cycle of a Servlet .....	256
The Core Servlet Interfaces .....	256

The HTTP Servlet Interfaces .....	262
Servlet Scenario 1: A Simple HTTP Request-Response .....	266
Servlet Scenario 2: The Client/Server Count .....	269
The Servlet Count .....	271
The Servlet Count Client .....	271
The Servlet Count Server .....	274
Compile The Client/Server Program .....	276
Create a Web Page .....	276
Run The Client/Server Program .....	277
Looking at Some Test Results .....	278
Servlets Versus CORBA/IIOP .....	278
Conclusion .....	280

**Chapter 13. RMI Versus CORBA/Java ORBs .....** 281

RMI 101 .....	281
How RMI Extends Local Java Calls .....	282
RMI's Garbage Collection of Remote Objects .....	282
The RMI Development Process .....	283
RMI Interfaces and Classes .....	286
The RMI Core .....	286
The Java RMI Naming Service .....	291
RMI's Dynamic Stub Loading Classes .....	294
The RMI Marshaling Objects .....	299
RMI-Over-IIOP .....	300
The JDK 1.2 RMI .....	300
RMI Scenario .....	301
The RMI Count .....	303
The RMI Count Client .....	303
The RMI Count Server .....	305
Compile the Client/Server Program .....	308
Run the Client/Server Programs .....	308
Looking At Some Test Results .....	309
RMI-over-IIOP Versus RMI-over-RMP .....	309
RMI Versus CORBA/IIOP .....	310
Conclusion .....	311

**Chapter 14. Caffeine: The "Pure" CORBA/Java ORB .....** 313

Caffeine 101 .....	314
Java2IIOP—Look Ma, No IDL .....	314

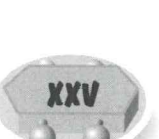


- Java2idl: CORBA IDL Without Pain ..... 315
- The Caffeine Development Process ..... 317
- The Caffeine URL Name Service..... 319
- A URL-Naming Scenario..... 321
- The Caffeinated Count..... 323
  - The Caffeinated Count Client..... 323
  - The Caffeinated Count Server ..... 325
- Compile the Client/Server Program ..... 327
- Run the Client/Server Programs ..... 328
- Looking At Some Test Results ..... 328
- Conclusion ..... 329



**Chapter 15. DCOM Versus CORBA/Java ORBs ..... 331**

- DCOM 101 ..... 332
  - Looking at DCOM Through CORBA Eyes ..... 332
  - DCOM Style Interfaces ..... 333
  - So, What's a DCOM Object? ..... 335
  - What's a DCOM Server? ..... 337
  - DCOM Local/Remote Transparency..... 338
  - The Ubiquitous IUnknown Interface..... 339
  - Interface Negotiations Using QueryInterface..... 340
  - IClassFactory2: Object Creation and Licensing ..... 342
  - A DCOM Object Creation Scenario ..... 344
  - DCOM Style Inheritance: Aggregation and Containment..... 346
  - The DCOM IDL..... 346
  - DCOM's Dynamic Invocation Facilities ..... 347
  - What's a Dispinterface? ..... 349
  - The IDispatch Interface ..... 349
  - The DCOM ODL ..... 353
  - Building and Registering Type Libraries..... 355
  - Finding and Loading a Type Library ..... 355
- DCOM and Java ..... 356
  - The DCOM for Java Development Process..... 356
  - DCOM Programming: Java Vs. C++ ..... 359
  - DCOM/Java Mappings ..... 361
- The DCOM Java Count..... 362
  - The DCOM Count Client..... 363
  - The DCOM Count IDL ..... 364
  - Run MIDL and JavaTLB ..... 365
  - The DCOM Count Server..... 366



- Compile The Client/Server Program..... 367
- Run the Client/Server Programs ..... 368
- Looking At Some Test Results ..... 369
- DCOM Versus CORBA/IIOP ..... 369
- Conclusion ..... 371



**Chapter 16. And the Winner Is... ..... 373**

- The Report Card ..... 374

**Part 5. The Existential CORBA ..... 381**



**Chapter 17. How Do I Find My ORB?..... 383**

- The CORBA 2.0 Initialization Interface ..... 383
- An Initialization Scenario ..... 384
- How You Find Your Other Objects ..... 385
  - CORBA Trader 101 ..... 386
- Conclusion ..... 388



**Chapter 18. Who Activates My Objects?..... 389**

- The Server Side of CORBA ..... 390
  - The CORBA::BOA Interface ..... 390
  - BOA and Other Object Adapters ..... 391
  - BOA Shared Server ..... 392
  - BOA Unshared Server ..... 394
  - BOA Server-per-Method ..... 394
  - BOA Persistent Server..... 395
  - An Object Activation Scenario..... 396
- The VisiBroker Activation Daemon..... 398
- CORBA 3.0's POA..... 400
  - Is It Goodbye BOA? ..... 400
  - POA: The Big Picture ..... 401
  - Persistent Object References ..... 402
  - Servant Managers ..... 403
  - POA Policies ..... 404
  - Playing the Policy Game..... 406
  - POA: Creating Object References..... 407



- Object Activation, POA-Style . . . . . 408
- Finding the Target Object . . . . . 409
- POA/Java . . . . . 409
- The POA Interfaces . . . . . 410
- POA Scenario 1 . . . . . 410
- POA Scenario 2 . . . . . 413
- Conclusion . . . . . 415

**Chapter 19. Metadata: Who Am I? . . . . . 417**

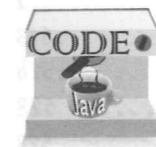
- The CORBA IDL: A Closer Look . . . . . 418
  - What Does an IDL Contract Cover? . . . . . 419
  - The Structure of the CORBA IDL . . . . . 419
  - An IDL Example . . . . . 421
  - Type Codes: CORBA's Self-Describing Data . . . . . 423
- The CORBA 2.0 Interface Repository . . . . . 424
  - What's an Interface Repository? . . . . . 424
  - Why Is an Interface Repository Needed Anyway? . . . . . 424
  - Interface Repository Classes: The Containment Hierarchy . . . . . 425
  - The Interface Repository Class Hierarchy . . . . . 426
  - Federated Interface Repositories . . . . . 430
  - What Does a Global Repository ID Look Like? . . . . . 431
  - CORBA IDL Without Pain . . . . . 432
- Conclusion . . . . . 432



**Chapter 20. The CORBA IDL-to-Java Mapping . . . . . 433**

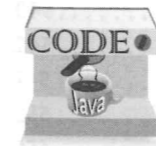
- Summary of IDL-to-Java Mappings . . . . . 434
- General Constructs . . . . . 436
  - CORBA Modules . . . . . 436
  - CORBA Exceptions . . . . . 437
  - CORBA Parameters and Holder Classes . . . . . 438
  - CORBA Helper Classes . . . . . 441
  - CORBA Attributes . . . . . 443
- CORBA Basic Types . . . . . 443
  - CORBA Constants . . . . . 443
  - The CORBA Primitive Types . . . . . 445
- CORBA Constructed Types . . . . . 446
  - CORBA Interface . . . . . 446
  - CORBA Sequence . . . . . 447
  - CORBA Array . . . . . 448
  - CORBA Structs . . . . . 448

- CORBA Enums . . . . . 450
- CORBA Union . . . . . 451
- CORBA Typedef . . . . . 454
- CORBA Any . . . . . 455
- CORBA/Java Server-Side Mappings . . . . . 458
  - Inheritance: The ImplBase Approach . . . . . 459
  - Delegation: The Tie Approach . . . . . 460
- Mapping CORBA Pseudo-Objects To Java . . . . . 464
  - CORBA::ORB Mapping . . . . . 465
  - CORBA::Object Mapping . . . . . 470
  - CORBA::NamedValue Mapping . . . . . 471
  - CORBA::NVList Mapping . . . . . 472
  - CORBA::Request Mapping . . . . . 473
  - CORBA::ServerRequest and the DSI Mappings . . . . . 474
  - CORBA::TypeCode . . . . . 478
  - CORBA::TCKind . . . . . 480
- Conclusion . . . . . 483



**Chapter 21. The CORBA Java-to-IDL Mapping . . . . . 485**

- RMI Meets CORBA . . . . . 486
  - Why RMI to IDL? . . . . . 486
  - Why RMI Over IIOP? . . . . . 487
  - CORBA's New Objects-By-Value . . . . . 488
  - The Value Type . . . . . 489
  - Passing Objects By Value . . . . . 490
  - Boxed Values . . . . . 492
  - Mapping Value Types to Java . . . . . 493
  - An Example . . . . . 493
- The Java-To-IDL Mapping . . . . . 494
  - RMI Meets CORBA . . . . . 494
  - The Java-to-IDL Mapping at a Glance . . . . . 494
  - RMI/IDL Remote Interfaces . . . . . 497
  - RMI/IDL Value Types . . . . . 498
- Conclusion . . . . . 499



**Chapter 22. The Introspective CORBA/Java Object . . . . . 501**

- CORBA Introspection . . . . . 502
- Java Reflection . . . . . 503
- The Ask Me Program . . . . . 504

The AskMe Code .....	504
Compile AskMe .....	513
Run AskMe .....	513
The Walk IR Program .....	514
The WalkIR Code .....	515
Compile WalkIR .....	516
Run WalkIR .....	516
Conclusion .....	518

**Part 6. JDBC 2-Tier Versus 3-Tier ..... 519****Chapter 23. JDBC Primer ..... 521**

The Origins of JDBC .....	522
The X/Open SAG CLI .....	522
The Microsoft ODBC CLI .....	523
CLI Versus Embedded SQL .....	526
The JDBC Architecture .....	528
JDBC Drivers .....	528
The JDBC URL Naming Conventions .....	529
Registering Drivers .....	530
JDBC Security .....	530
Mapping SQL Data Types to Java .....	530
JDBC Transactions .....	532
JDBC Stored Procedures .....	532
JDBC Interfaces: An Overview .....	532
The JDBC Core .....	533
The JDBC Java Language Extensions .....	534
The JDBC Java Utilities .....	536
JDBC Metadata .....	536
JDBC Interfaces: In-Depth .....	537
JDBC Core: The Driver Interfaces .....	537
JDBC Core: The Connection Interface .....	540
JDBC Core: The Statement Interfaces .....	542
JDBC Core: The ResultSet Interfaces .....	547
JDBC Java Language Extensions: The Exception Classes .....	552
JDBC Date and Time Classes .....	555
JDBC DatabaseMetaData Interface .....	558

JDBC Scenarios .....	567
JDBC Scenario 1: Invoking a SQL Query .....	567
JDBC Scenario 2: Invoking a Prepared Command .....	569
JDBC Scenario 3: Invoking a Stored Procedure .....	571
Conclusion .....	573

**Chapter 24. The JDBC Debit-Credit Benchmark. .... 575**

Designing the Benchmark .....	576
What to Look for in an OLTP Benchmark .....	576
The Debit-Credit Benchmark .....	577
The Debit-Credit Database .....	578
The Debit-Credit Transaction .....	578
The Debit-Credit Constraints .....	579
Price/Performance of Debit-Credit .....	580
The JDBC Debit-Credit Benchmark Facility .....	580
Build Your Java Debit-Credit Benchmark in One Afternoon .....	581
2-Tier Versus 3-Tier .....	582
JDBC 2-Tier .....	583
JDBC 2-Tier Plus .....	584
JDBC 3-Tier .....	586
Let's Create a Bank .....	587
The CreateBank Class .....	587
The BankAdmin Class .....	588
Compile the Create Bank Program .....	596
Run the Create Bank Program .....	597
Debit-Credit: To Cheat or Not To Cheat? .....	597
Conclusion .....	598

**Chapter 25. 2-Tier Debit-Credit with JDBC ..... 599**

The Debit-Credit 2-Tier Environment .....	600
The 2-Tier Client/Server Code Structure .....	602
Meet the Players .....	602
A Multithreaded Scenario .....	603
The Debit-Credit 2-Tier Code .....	606
Compile Debit-Credit 2-Tier .....	615
Run the Debit-Credit 2-Tier Benchmark .....	616
Conclusion .....	619



**Chapter 26. 3-Tier Debit-Credit with JDBC and CORBA . . . . . 621**

- The 3-Tier Debit-Credit Environment . . . . . 622
- The Debit-Credit 3-Tier Code Structure . . . . . 623
  - Meet the Client Players . . . . . 624
  - The Server: What the Client Sees . . . . . 625
  - Meet the New Server Players . . . . . 625
  - A 3-Tier Client/Server Scenario . . . . . 629
- The 3-Tier Debit-Credit Code . . . . . 631
  - The Debit-Credit 3-Tier Client Code . . . . . 631
  - The 3-Tier Debit-Credit Server Code . . . . . 638
- Compile the 3-Tier Debit-Credit . . . . . 645
- Run the Debit-Credit 3-Tier Benchmark . . . . . 646
- 2-Tier Versus 3-Tier . . . . . 648
- Conclusion . . . . . 650

**Part 7. From JavaBeans To Enterprise JavaBeans . . . . . 651**



**Chapter 27. The JavaBean Component Model . . . . . 655**

- The Bean Component Model . . . . . 656
  - Beans Versus Applets . . . . . 656
  - JavaBeans: The Black-Box View . . . . . 656
  - JavaBeans: The JDK as a Component Framework . . . . . 657
  - JavaBeans: Design-Time Versus Run-Time . . . . . 662
- The So-Called Design Patterns . . . . . 663
  - Simple Property Naming Patterns . . . . . 664
  - Indexed Property Naming Patterns . . . . . 665
  - Event Naming Patterns . . . . . 665
  - Method Naming Patterns . . . . . 666
- The JavaBeans Classes: An Overview . . . . . 666
  - JDK 1.1 JavaBeans Classes . . . . . 666
  - JDK 1.2 JavaBeans Enhancements . . . . . 668
- Conclusion . . . . . 672



**Chapter 28. The Smiley JavaBean . . . . . 673**

- The Minimalist Smiley Bean . . . . . 673
- The Minimalist Smiley Bean . . . . . 674

- SmileyPlace: A Rudimentary Bean Container . . . . . 675
- The Beans Class . . . . . 678
- First Guidelines for Writing Portable Beans . . . . . 679
- Conclusion . . . . . 681



**Chapter 29. The Event-Driven JavaBean . . . . . 683**

- The Delegation Event Model in a Nutshell . . . . . 684
  - Sources and Listeners . . . . . 684
  - The Big Bang Scenario . . . . . 686
  - The EventObject Class and EventListener Interface . . . . . 688
- Smiley Bean Does Events . . . . . 689
  - The ActionEvent and ActionListener Classes . . . . . 690
  - An Event for All Seasons . . . . . 692
  - The Event-Savvy Smiley . . . . . 692
  - SmileyPlace: A Hard-Wired Bean Container . . . . . 694
  - Anonymous Inner Class Adapters . . . . . 698
- Event Adapters . . . . . 698
  - Why Adapters? . . . . . 698
  - Event Adapters: A Taxonomy . . . . . 699
  - The Smiley Event Adapter . . . . . 702
  - The CORBA Count Event Adapter . . . . . 704
- More Guidelines for Writing Portable Beans . . . . . 708
- Conclusion . . . . . 709

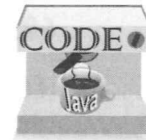


**Chapter 30. The Propertied JavaBean . . . . . 711**

- Standalone Properties . . . . . 712
  - Simple Properties . . . . . 712
  - Multi-Valued Indexed Properties . . . . . 713
- Active Properties . . . . . 714
  - Bound Properties . . . . . 714
  - The Bound Property Classes and Interfaces . . . . . 716
  - A Bound Property Scenario . . . . . 717
  - The Bound-Propertied Smiley . . . . . 719
  - The Event Adapters . . . . . 721
  - SmileyPlace: A Container for Bound Beans . . . . . 722
  - Constrained Properties . . . . . 725
  - Constrained Properties: Classes and Interfaces . . . . . 725
  - A Constrained Property Scenario . . . . . 728
  - The Constrained-Property Smiley . . . . . 730

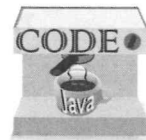


The Event Adapters . . . . . 733  
 SmileyPlace: A Container for Constrained Beans . . . . . 734  
 More Guidelines for Writing Portable Beans . . . . . 737  
 Conclusion . . . . . 738



**Chapter 31. The Persistent JavaBean. . . . . 739**

Bean Persistence . . . . . 740  
 The Bean-Related Persistence Interfaces . . . . . 740  
 The Container-Related Persistence Classes. . . . . 744  
 A Serialization Scenario . . . . . 749  
 A Deserialization Scenario . . . . . 750  
 The Persistent Smiley . . . . . 752  
 SmileyPlace: A Container for Persistent Beans . . . . . 752  
 The Versioning of a Bean's State . . . . . 756  
 JDK 1.2 Serialization Enhancements . . . . . 758  
 Beans In JARs . . . . . 758  
 A Manifest for JARs . . . . . 759  
 The JAR Utility . . . . . 761  
 The Structure of a Manifest . . . . . 764  
 Creating Your Own Manifest . . . . . 766  
 Signing a JAR . . . . . 768  
 JARs and Browsers . . . . . 771  
 Placing JARs in Your CLASSPATH . . . . . 772  
 JDK 1.2 JAR Enhancements . . . . . 772  
 The Java Security APIs . . . . . 772  
 The Security Classes and Interfaces . . . . . 772  
 A Security Signing Scenario . . . . . 779  
 A Security Signature Verification Scenario. . . . . 781  
 JDK 1.2 Security Enhancements . . . . . 782  
 The New JDK 1.2 Security Model . . . . . 782  
 The JDK 1.2 Security Enhancements: A Summary. . . . . 783  
 JDK 1.2 New Security Classes . . . . . 784  
 Guidelines for Writing Persistent Beans . . . . . 786  
 Conclusion . . . . . 787



**Chapter 32. The Introspective and Toolable JavaBean. . . . . 789**

Which Tool? . . . . . 790  
 The Introspective JavaBean . . . . . 791  
 The Core Reflection Classes . . . . . 791

JavaBeans Introspection: An Overview . . . . . 794  
 JavaBeans Introspection: The Metadata Descriptor Classes . . . . . 795  
 Introspecting with BeanInfo . . . . . 801  
 The Introspector and BeanInfo Interfaces . . . . . 803  
 Scenario Time . . . . . 806  
 Scenario 1: Discovering a Bean's Events . . . . . 806  
 Scenario 2: Reading a Property Value "On-the-Fly" . . . . . 807  
 The Introspective Smiley . . . . . 809  
 The Bean Analyst . . . . . 810  
 Tools Discover Smiley . . . . . 813  
 Property Editors and Customizers . . . . . 816  
 Built-In Property Editors . . . . . 816  
 The Property Editor Interfaces . . . . . 816  
 A Property Editing Scenario . . . . . 821  
 The Smile Editor . . . . . 823  
 Bean Customizers . . . . . 827  
 The Customizer Interface . . . . . 828  
 Guidelines for Writing Toolable Beans . . . . . 829  
 Conclusion . . . . . 830



**Chapter 33. CORBA Beans. . . . . 831**

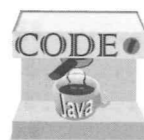
CORBA and JavaBeans . . . . . 832  
 What CORBA Can Do for JavaBeans . . . . . 832  
 What JavaBeans Can Do for CORBA . . . . . 833  
 The New CORBA Component Model . . . . . 834  
 The CORBA/JavaBeans Call for Action . . . . . 835  
 CORBA Components: JavaBeans++ . . . . . 837  
 CORBA Remote Events . . . . . 840  
 CORBA Containers . . . . . 842  
 Component Packaging. . . . . 844  
 How It All Comes Together . . . . . 845  
 Conclusion . . . . . 846



**Chapter 34. Enterprise JavaBeans and CORBA. . . . . 849**

EJBs and CORBA Object Transaction Monitors . . . . . 850  
 So, What Exactly Is an OTM? . . . . . 851  
 CORBA OTMs . . . . . 852  
 Enterprise JavaBeans and CORBA OTMs . . . . . 854  
 The EJB Container Framework . . . . . 855

Session and Entity Beans . . . . .	857
The EJB Client/Server Development Process . . . . .	858
The Client's View of an EJB . . . . .	860
The Remote EJB Interfaces . . . . .	862
An EJB Client/Server Scenario . . . . .	865
The EJB/Container Protocol . . . . .	866
The Container/Bean Callbacks . . . . .	866
The Container/Bean Interfaces . . . . .	868
Support For Transactions . . . . .	872
The EJB Transaction Model . . . . .	872
EJB Transaction Usage . . . . .	873
Declarative Transaction Management . . . . .	874
EJB Packaging . . . . .	876
EJB-JAR and Manifest . . . . .	877
Deployment Descriptor . . . . .	877
Security Descriptor . . . . .	877
The EJB Deployment Classes . . . . .	877
EJB Design Guidelines . . . . .	882
Conclusion . . . . .	883

**Part 8. Grand Finale: Club Med with CORBA/JavaBeans . 885****Chapter 35. The Club Med 3-Tier Client/Server. . . . . 887**

The Club Med Client . . . . .	888
Ten Minutes in the Life of a Club Med Bean . . . . .	888
The Bargain Hunter Bean . . . . .	893
Club Med Server . . . . .	894
The 3-Tier Club Med . . . . .	895
The Club Med Server: What the Client Sees . . . . .	896
The Club Med IDL Contract . . . . .	896
Club Med Database . . . . .	898
The CreateClubMed Program . . . . .	899
Compile the CreateClubMed Program . . . . .	903
Run the CreateClubMed Program . . . . .	904
Conclusion . . . . .	904

**Chapter 36. The Club Med Client Beans . . . . . 905**

The Visual Café Club Med Bean . . . . .	907
Developing JavaBeans with Visual Café 2.1 . . . . .	907
The Visual Café ClubMedBean . . . . .	912
How Visual Café Does CORBA . . . . .	913
The JBuilder Club Med Bean . . . . .	917
JBuilder 101 . . . . .	917
Developing JavaBeans with JBuilder . . . . .	919
JBuilder's CORBA Support . . . . .	921
The JBuilder ClubMedBean . . . . .	922
The VisualAge Bargain Hunter Bean . . . . .	923
The ClubMedCorbaBean . . . . .	923
VisualAge 101 . . . . .	926
Creating the BargainHunterBean with VisualAge . . . . .	927
Conclusion . . . . .	932

**Chapter 37. The Club Med CORBA Server. . . . . 933**

Club Med Server: The Code Structure . . . . .	934
The 3-Tier Club Med . . . . .	934
The ClubMedDispenser . . . . .	935
Meet the Server Players . . . . .	936
A Club Med Client/Server Scenario . . . . .	937
The Club Med Server Code . . . . .	939
Compile the Club Med Server . . . . .	953
Running Club Med . . . . .	953
Conclusion . . . . .	954

**Chapter 38. CORBA/Java: The Good, the Bad, and the Ugly . . . 955**

So, What Did We Learn? . . . . .	956
CORBA ORBs: The Good, the Bad, and the Ugly . . . . .	956
CORBA/Java Clients: The Good, the Bad, and the Ugly . . . . .	959
CORBA/Java Servers: The Good, the Bad, and the Ugly . . . . .	961
The Object Web Vision and Shippable Places . . . . .	963
What Is a Shippable Place? . . . . .	963
The Future Web Client . . . . .	965
Portable Component Stores . . . . .	965
Jumping Beans . . . . .	967



## Client/Server Programming with Java and CORBA

---

Back to Reality .....	968
It's Time To Say Good-Bye .....	970
<b>Where to Go for More Information .....</b>	<b>973</b>
<b>Index .....</b>	<b>979</b>