Middleware for Communications

Edited by

Qusay H. Mahmoud
University of Guelph, Canada

John Wiley & Sons, Ltd
Contents

Preface xix

List of Contributors xxi

Introduction xxvii

1 Message-Oriented Middleware 1
Edward Curry

1.1 Introduction 1
1.1.1 Interaction Models 1
1.1.2 Synchronous Communication 2
1.1.3 Asynchronous Communication 2
1.1.4 Introduction to the Remote Procedure Call (RPC) 2
1.1.5 Introduction to Message-Oriented Middleware (MOM) 4
1.1.6 When to use MOM or RPC 6

1.2 Message Queues 7

1.3 Messaging Models 8
1.3.1 Point-to-Point 8
1.3.2 Publish/Subscribe 9
1.3.3 Comparison of Messaging Models 11

1.4 Common MOM Services 12
1.4.1 Message Filtering 12
1.4.2 Transactions 12
1.4.3 Guaranteed Message Delivery 15
1.4.4 Message Formats 15
1.4.5 Load Balancing 15
1.4.6 Clustering 16

1.5 Java Message Service 16
1.5.1 Programming using the JMS API 17

1.6 Service-Oriented Architectures 22
1.6.1 XML 22
1.6.2 Web Services 23
1.6.3 MOM  
1.6.4 Developing Service-Oriented Architectures  
1.7 Summary  
Bibliography

2 Adaptive and Reflective Middleware  
Edward Curry

2.1 Introduction  
2.1.1 Adaptive Middleware  
2.1.2 Reflective Middleware  
2.1.3 Are Adaptive and Reflective Techniques the Same?  
2.1.4 Triggers of Adaptive and Reflective Behavior  
2.2 Implementation Techniques  
2.2.1 Meta-Level Programming  
2.2.2 Software Components and Frameworks  
2.2.3 Generative Programming  
2.3 Overview of Current Research  
2.3.1 Reflective and Adaptive Middleware Workshops  
2.3.2 Nonfunctional Properties  
2.3.3 Distribution Mechanism  
2.4 Future Research Directions  
2.4.1 Advances in Programming Techniques  
2.4.2 Open Research Issues  
2.4.3 Autonomic Computing  
2.5 Summary  
Bibliography

3 Transaction Middleware  
Stefan Tai, Thomas Mikalsen, Isabelle Rouvellou

3.1 Introduction  
3.2 Transaction Processing Fundamentals  
3.2.1 ACID Transactions  
3.2.2 Distributed Transactions  
3.2.3 Common Extensions  
3.2.4 Programming Models for Transactions  
3.3 Distributed Object Transactions  
3.3.1 Transaction Model  
3.3.2 Transaction APIs  
3.3.3 Container-Managed Transactions  
3.4 Messaging Transactions  
3.4.1 Messaging Models  
3.4.2 Programming Models  
3.4.3 Queued Transaction Processing  
3.5 Web Transactions  
3.5.1 Web Services Coordination and Transactions  
3.5.2 Programming model
3.5.3 Web Services Messaging 70
3.6 Advanced Transactions 70
3.6.1 Long Running Unit of Work (LRUOW) 71
3.6.2 Conditional Messaging and D-Spheres 72
3.6.3 Transactional Attitudes (TxA) 74
3.7 Conclusion 77
Bibliography 78

4 Peer-to-Peer Middleware 81
Markus Oliver Junginger

4.1 Introduction 81
4.1.1 Peer-to-Peer and Grids 82
4.1.2 Lack of Peer-to-Peer Middleware 82
4.1.3 Group Communication 83
4.1.4 Challenges 83
4.1.5 Chapter Outline 83
4.2 JXTA 84
4.2.1 Overview 84
4.2.2 Resources and Advertisements 84
4.2.3 Peer Groups 85
4.2.4 Services and Modules 85
4.2.5 Protocols 86
4.2.6 Messages and Pipes 86
4.2.7 Security 87
4.2.8 Relay and Rendezvous Peers 87
4.2.9 Group Communication 88
4.2.10 Applications using JXTA 89
4.2.11 Challenges 89
4.2.12 Summary 90
4.3 P2P Messaging System 90
4.3.1 Self-Organizing Overlay Networks 90
4.3.2 Failure Tolerance 92
4.3.3 Implicit Dynamic Routing 93
4.3.4 Quality-of-Service 94
4.3.5 System Model 95
4.3.6 Network Abstraction Layer 96
4.3.7 Implementation 96
4.3.8 Challenges and Comparison with JXTA 96
4.3.9 Summary 97
4.4 Hybrid Middleware – a Conceptual Proposal 97
4.4.1 Service Providers 99
4.4.2 Conceptual Model and Services 99
4.4.3 Service Connectors 100
4.4.4 Peer Group Membership Service 101
4.4.5 Synchronization Service and P2P Synchronization 102
4.4.6 Summary 104
4.5 Conclusion 105
Bibliography 105
5 Grid Middleware

Gregor von Laszewski and Kaizar Amin

5.1 The Grid 110
5.2 Grid Architecture 112
5.3 Grid Middleware Software 114
5.4 Grid Middleware Challenges 115
5.5 Grid Middleware Standardization 115
5.6 Grid Middleware Services 115
  5.6.1 Elementary Grid Middleware Services 116
  5.6.2 Advanced Grid Management Services 117
5.7 Grid Middleware Toolkits 118
  5.7.1 Globus Toolkit 118
  5.7.2 Commodity Grid Kits 120
  5.7.3 Open Grid Services Architecture 120
5.8 Portal Middleware for Grids 123
5.9 Applications Using and Enhancing Grid Middleware 125
  5.9.1 Astrophysics 125
  5.9.2 Earthquake Engineering 126
  5.9.3 High-energy Physics Grids 126
5.10 Concluding Remarks 127
Acknowledgments 127
Bibliography 128

6 QoS-enabled Middleware

Nanbor Wang, Christopher D. Gill, Douglas C. Schmidt, Aniruddha Gokhale, Balachandran Natarajan, Joseph P. Loyall Richard E. Schantz, and Craig Rodrigues

6.1 Introduction 131
  6.1.1 Emerging Trends 131
  6.1.2 Key Technical Challenges and Solution Approaches 132
  6.1.3 Chapter Organization 135
6.2 The Evolution of Middleware 135
  6.2.1 Overview of Middleware 135
  6.2.2 Limitations of Conventional Middleware 137
6.3 Component Middleware: A Powerful Approach to Building DRE Applications 138
  6.3.1 Overview of Component Middleware and the CORBA Component Model 138
  6.3.2 Limitations with Conventional Component Middleware for Large-scale DRE Systems 142
6.4 QoS Provisioning and Enforcement with CIAO and QuO Qoskets 144
  6.4.1 Static QoS Provisioning via QoS-enabled Component Middleware and CIAO 145
  6.4.2 Dynamic QoS Provisioning via QuO Adaptive Middleware and Qoskets 150
  6.4.3 Integrated QoS provisioning via CIAO and Qoskets 154
6.5 Related Work 156  
6.6 Concluding Remarks 158  
Bibliography 159  

7 Model Driven Middleware 163  
Aniruddha Gokhale, Douglas C. Schmidt, Balachandran Natarajan, Jeff Gray, Nanbor Wang  

7.1 Introduction 163  
7.2 Overview of the OMG Model Driven Architecture (MDA) 169  
7.2.1 Capabilities of the MDA 169  
7.2.2 Benefits of the MDA 172  
7.3 Overview of Model Driven Middleware 172  
7.3.1 Limitations of Using Modeling and Middleware in Isolation 173  
7.3.2 Combining Model Driven Architecture and QoS-enabled Component Middleware 173  
7.4 Model Driven Middleware Case Study: Integrating MDA with QoS-enabled Middleware 178  
7.5 Related Work 181  
7.6 Concluding Remarks 183  
Bibliography 184  

8 High-Performance Middleware-Based Systems 189  
Shikharesh Majumdar  

8.1 Introduction 189  
8.2 Performance of CORBA Middleware 191  
8.3 Impact of Client-Server Interaction Architectures 192  
8.3.1 Three Interaction Architectures 192  
8.3.2 Performance Comparison 194  
8.4 Middleware Performance Optimization 197  
8.4.1 Systems with Limited Heterogeneity 198  
8.4.2 Flyover 198  
8.4.3 Performance of Flyover 201  
8.5 Application Level Performance Optimizations 204  
8.5.1 Connection Setup Latency 204  
8.5.2 Parameter Passing 205  
8.5.3 Combination of Methods and Performance Recovery 206  
8.5.4 Method Placement and Object Packing 207  
8.5.5 Load Balancing 207  
8.6 Summary and Conclusions 208  
8.7 Acknowledgments 208  
Bibliography 208  

9 Concepts and Capabilities of Middleware Security 211  
Steven Demurjian, Keith Bessette, Thuong Doan, Charles Phillips  

9.1 Introduction 211
9.2 Security in CORBA, .NET, and J2EE 212
  9.2.1 CORBA Security Capabilities 213
  9.2.2 .NET Security Capabilities 217
  9.2.3 J2EE Security Capabilities 223
9.3 RBAC and MAC using CORBA and JINI 227
  9.3.1 Overview of the RBAC/MAC Security Model 228
  9.3.2 The Security Services of USR 229
  9.3.3 Prototyping/Administrative and Management Tools 231
9.4 Conclusion 232
Bibliography 234

10 Middleware for Scalable Data Dissemination 237
  Panos K. Chrysanthis, Vincenzo Liberatore, Kirk Pruhs

10.1 Introduction 237
10.2 Architecture Overview 238
10.3 Background and Historical Notes 242
  10.3.1 Multicast Data Dissemination 242
  10.3.2 Multicast 242
10.4 Middleware Components 243
  10.4.1 Transport Adaptation Layer 243
  10.4.2 Document Selection 244
  10.4.3 Multicast Push Scheduling 245
  10.4.4 Multicast Pull Scheduling 247
  10.4.5 Multicast Indexing 249
  10.4.6 Data Consistency and Currency 250
  10.4.7 Client Cache 251
10.5 Building Block Integration 252
  10.5.1 Integration 252
  10.5.2 Scheduling for Layered Multicast 253
10.6 Application: Real-Time Outbreak and Disease Surveillance 253
10.7 Conclusions 254
Bibliography 255

11 Principles of Mobile Computing Middleware 261
  Cecilia Mascolo, Licia Capra, Wolfgang Emmerich

11.1 Introduction 261
11.2 Mobile Distributed Systems 262
  11.2.1 Characterization of Distributed Systems 263
  11.2.2 Traditional Distributed Systems 264
  11.2.3 Mobile Nomadic Systems 264
  11.2.4 Mobile Ad Hoc Systems 265
11.3 Middleware Systems: A Reference Model 266
  11.3.1 Middleware for Fixed Distributed Systems 267
  11.3.2 Middleware for Mobile Nomadic and Ad hoc Systems 268
11.4 Fault Tolerance
   11.4.1 Connectivity
   11.4.2 Data-sharing
11.5 Heterogeneity
11.6 Openness
11.7 Scalability
   11.7.1 Discovery
   11.7.2 Quality of Service
11.8 Resource-sharing
   11.8.1 Transactions
   11.8.2 Security
11.9 Conclusions

Bibliography

12 Application of Middleware Technologies to Mobile Enterprise Information Services
   Guijun Wang, Alice Chen, Surya Sripada, Changzhou Wang

12.1 Introduction
12.2 Wireless Technologies
12.3 Middleware Technologies for Enterprise Application Integrations
12.4 An Integrated Architecture for Mobile Enterprise Information Services
   12.4.1 Enterprise Requirements
   12.4.2 Design Considerations and Our Approach
   12.4.3 An Integrated Mobile EIS Architecture
   12.4.4 Deployment and Operation
12.5 J2EE-Based Middleware in Mobile EIS
   12.5.1 J2EE Middleware Platform
   12.5.2 JMS
   12.5.3 JMS in Our Mobile EIS
12.6 Data Representation and Presentation in Mobile Enterprise Information Services
12.7 Challenges and Future Directions
12.8 Summary and Conclusions
12.9 Acknowledgment

Bibliography

13 Middleware for Location-based Services: Design and Implementation Issues
   Peter Langendörfer, Oliver Maye, Zoya Dyka, Roland Sorge, Rita Winkler, Rolp Kraemer

13.1 Introduction
13.2 Related Work
13.3 Architecture
   13.3.1 Infrastructure
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.3.2</td>
<td>Platform Components</td>
<td>311</td>
</tr>
<tr>
<td>13.4</td>
<td>Concepts of Selected Components</td>
<td>314</td>
</tr>
<tr>
<td>13.4.1</td>
<td>Event Engine and Auras and Objects Engine</td>
<td>314</td>
</tr>
<tr>
<td>13.4.2</td>
<td>Profile Servers and Profile Database</td>
<td>319</td>
</tr>
<tr>
<td>13.5</td>
<td>Measurements</td>
<td>321</td>
</tr>
<tr>
<td>13.5.1</td>
<td>Testbed Settings</td>
<td>321</td>
</tr>
<tr>
<td>13.5.2</td>
<td>Results</td>
<td>322</td>
</tr>
<tr>
<td>13.5.3</td>
<td>Conclusions</td>
<td>324</td>
</tr>
<tr>
<td>13.6</td>
<td>Realization of PLASMA</td>
<td>325</td>
</tr>
<tr>
<td>13.6.1</td>
<td>Design Decisions and Lessons Learned</td>
<td>325</td>
</tr>
<tr>
<td>13.6.2</td>
<td>Implementation Notes</td>
<td>326</td>
</tr>
<tr>
<td>13.7</td>
<td>Conclusions</td>
<td>327</td>
</tr>
<tr>
<td>Bibliography</td>
<td></td>
<td>327</td>
</tr>
</tbody>
</table>

14 QoS-Enabled Middleware for MPEG Video Streaming

Karl R.P.H. Leung, Joseph Kee-Yin Ng, Calvin Kin-Cheung Hui

14.1 Introduction | 331
14.2 Related Works | 333
| 14.2.1 Overview of MPEG | 333
| 14.2.2 Quality of Services | 334
| 14.2.3 Video Distribution | 335
14.3 Requirements for QoS-enabled MPEG-Streaming Middleware | 336
| 14.3.1 Environment Analysis | 336
| 14.3.2 Requirements for the Middleware | 337
14.4 QoS Facilities | 337
| 14.4.1 QoS-GFS | 337
| 14.4.2 Transmission Scheme | 338
| 14.4.3 QoS Tuning Scheme | 341
14.5 Architecture | 341
| 14.5.1 Clientware | 341
| 14.5.2 Serverware | 345
14.6 Experiments | 347
| 14.6.1 System Setup | 347
| 14.6.2 Experiment Results | 348
| 14.6.3 QoS in QoS-Index | 353
| 14.6.4 QoS-Frame and QoS-Byte | 353
14.7 Discussions | 353
14.8 Acknowledgment | 354
14.9 Conclusion & Future Works | 355
Bibliography | 355

15 Middleware for Smart Cards

Harald Vogt, Michael Rohs, Roger Kilian-Kehr

15.1 Introduction | 359
15.2 ISO 7816 | 360
| 15.2.1 Communication between Card and Card Reader | 360
15.3 Data Structures on Smart Cards 361
15.3.1 Command Sets 362
15.4 JavaCards 362
15.4.1 Hardware Architecture 363
15.4.2 Runtime Environment 363
15.4.3 Developing JavaCard Applets 365
15.5 PC/SC: Data Communications 369
15.6 OpenCard Framework 371
15.6.1 Architectural Concepts 371
15.6.2 Configuration 373
15.6.3 Programming Model 373
15.6.4 Summary 374
15.7 JavaCard RMI 375
15.7.1 On-Card JCRMI 375
15.7.2 Off-Card JCRMI 378
15.7.3 Summary 382
15.8 PKCS #11 Security Tokens 382
15.9 Smart Cards as Distributed Objects 383
15.10 Smart Card Middleware for Mobile Environments 383
15.10.1 SIM Application Toolkit 383
15.10.2 J2ME Smart Card Middleware 385
15.11 JiniCard 386
15.12 Smart Cards on the Internet 387
15.12.1 A Browser Interface for Smart Cards 387
15.12.2 Smart Cards as Mobile Web Servers 387
15.12.3 Internet Smart Cards 388
15.13 Conclusion 389
Bibliography 389

16 Application-Oriented Middleware for E-Commerce 393
Jesus Martínez, Luis R. López, Pedro Merino

16.1 Introduction 393
16.2 Previous Work on Networked Smart Card Applications 394
16.3 A Public Transport Ticketing System for e-Commerce 396
16.3.1 The System Architecture 396
16.3.2 The Electronic Ticket 399
16.3.3 Choosing a Smart Card for the System 400
16.4 Advanced Ticketing Management Using Middleware 402
16.4.1 Middleware Platform Security 403
16.4.2 The Smart Card Service 404
16.4.3 The Smart Card Object-Oriented Library 406
16.5 The Application Prototype 407
16.6 Summary and Conclusions 411
16.7 Acknowledgments 411
Bibliography 411
17 Real-time CORBA Middleware

Arvind S. Krishna, Douglas C. Schmidt, Raymond Klefstad, Angelo Corsaro

17.1 Introduction 413
17.2 DRE System Technology Challenges 415
  17.2.1 Challenges of Today’s DRE Systems 415
  17.2.2 Challenges of Future DRE Systems 416
  17.2.3 Limitations with Conventional DRE System Development 417
17.3 Overview of Real-time CORBA 418
  17.3.1 Overview of CORBA 418
  17.3.2 Overview of Real-time CORBA 1.0 419
  17.3.3 Overview of Real-time CORBA 2.0 421
17.4 TAO: C++-based Real-time CORBA Middleware 422
  17.4.1 Motivation 422
  17.4.2 TAO Architecture and Capabilities 423
  17.4.3 TAO Successes 425
17.5 ZEN: RTSJ-based Real-time CORBA Middleware 426
  17.5.1 Motivation 426
  17.5.2 ZEN Architecture and Capabilities 427
  17.5.3 ZEN Successes 430
17.6 Related Work 432
17.7 Concluding Remarks 433
Bibliography 434

18 Middleware Support for Fault Tolerance

Diana Szentiváni, Simin Nadjm-Tehrani

18.1 Introduction 439
18.2 Terminology 441
  18.2.1 Replication Styles 441
  18.2.2 Consensus 441
  18.2.3 Unreliable Failure Detectors 442
  18.2.4 Broadcast 442
18.3 Background 442
  18.3.1 Middleware Fault Tolerance 443
  18.3.2 CORBA and Fault Tolerance 443
18.4 Standard Support for FT in CORBA 444
  18.4.1 The FT-CORBA Standard 444
  18.4.2 Architecture Units 445
18.5 Adding Support for Full Availability 450
  18.5.1 Architecture Units 450
  18.5.2 Infrastructure Interactions 451
  18.5.3 Platform Implementation 453
18.6 Experiments with a Telecom Application 455
  18.6.1 The Service 455
  18.6.2 Experiment Setup 455
  18.6.3 Measuring Overheads 456
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>18.7  Trade-off Studies</td>
<td>456</td>
</tr>
<tr>
<td>18.7.1 Overheads</td>
<td>457</td>
</tr>
<tr>
<td>18.7.2 Failover Times</td>
<td>459</td>
</tr>
<tr>
<td>18.8  Conclusions</td>
<td>461</td>
</tr>
<tr>
<td>18.9  Acknowledgments</td>
<td>463</td>
</tr>
<tr>
<td>Bibliography</td>
<td>463</td>
</tr>
<tr>
<td><strong>Index</strong></td>
<td><strong>465</strong></td>
</tr>
</tbody>
</table>