



The Ultimate Guide

J2EE™ WEB SERVICES

XML * SOAP * WSDL * UDDI * WS-I
JAX-RPC * JAXR * SAAJ * JAXP

Richard Monson-Haefel

力

Contents

Preface xxxiii

Are Web Services Important? xxxiii

What Do I Need to Know to Read This Book? xxxiv

What Does This Book Cover? xxxiv

How Is This Book Organized? xxxv

What Doesn't This Book Cover? xxxvii

Non-Web Service Aspects of the J2EE Platform xxxvii

Vendor-Specific Configuration and Administration xxxvii

Other Web Service "Standards" xxxviii

Acknowledgments xxxix

Chapter 1 An Overview of J2EE 1.4 Web Services 1

1.1 The J2EE Platform 2

1.2 The Technologies of Web Services 3

1.2.1 WS-I Basic Profile 1.0 5

1.2.2 XML 6

1.2.3 SOAP 7

1.2.4 WSDL 9

1.2.5 UDDI 9

1.3 The J2EE Web Service APIs 10

1.3.1 JAX-RPC 10

1.3.2 SAAJ 11

1.3.3 JAXR 11

1.3.4 JAXP 12

1.4 Wrapping Up 12

Part I XML 15

Chapter 2 XML Basics 17

2.1 XML Primer 17

2.1.1 XML Document Instance 20

2.1.2 Anatomy of an XML Document 21

2.1.2.1 XML Declaration 21

2.1.2.2 Elements 21

2.1.2.3 Attributes 22

2.1.2.4 Comments 23

2.1.2.5 CDATA Section 24

2.1.3 Processing XML Documents 25

2.2 XML Namespaces 26

2.2.1 An Example of Using Namespaces 26

2.2.2 Default Namespaces, Prefixes, and Qualified Names 29

2.3 Wrapping Up 32

Chapter 3 The W3C XML Schema Language 33

3.1 XML Schema Basics 33

3.1.1 Why XML Schema Is Preferred to DTDs in Web Services 33

3.1.2 The XML Schema Document 35

3.1.3 Simple Types 36

3.1.4 Complex Types 39

3.1.4.1 Sequences of Elements 41

3.1.4.2 Attributes 43

3.1.4.3	Occurrence Constraints	45
3.1.4.4	The all Element	48
3.1.5	<i>Declaring Global Elements in a Schema</i>	49
3.1.6	<i>Qualified and Unqualified Elements</i>	52
3.1.7	<i>Assigning and Locating Schemas</i>	55
3.2	Advanced XML Schema	57
3.2.1	<i>Inheritance of Complex Types</i>	58
3.2.1.1	Extension	58
3.2.1.2	Restriction	60
3.2.1.3	Polymorphism and Abstract Base Types	61
3.2.1.4	Abstract and Final Complex Types	63
3.2.2	<i>Inheritance of Simple Types</i>	64
3.2.2.1	The pattern Facet	66
3.2.2.2	The enumeration Facet	68
3.2.3	<i>List and Union Types</i>	68
3.2.3.1	List Types	69
3.2.3.2	Union Types	69
3.2.4	<i>Anonymous Types</i>	70
3.2.5	<i>Importing and Including Schemas</i>	72
3.2.5.1	Importing	72
3.2.5.2	Including	75
3.3	Wrapping Up	77

Part II SOAP and WSDL 79

Chapter 4 SOAP 81

4.1	The Basic Structure of SOAP	84
4.2	SOAP Namespaces	88
4.3	SOAP Headers	95
4.3.1	<i>The actor Attribute</i>	98
4.3.2	<i>The mustUnderstand Attribute</i>	102
4.3.3	<i>The WS-I Conformance Header Block</i>	105
4.3.4	<i>Final Words about Headers</i>	106

4.4	The SOAP Body	106
4.5	SOAP Messaging Modes	107
4.5.1	<i>Document/Literal</i>	107
4.5.2	<i>RPC/Literal</i>	108
4.5.3	<i>Messaging Modes versus Messaging Exchange Patterns</i>	110
4.5.4	<i>Other Messaging Modes</i>	111
4.6	SOAP Faults	111
4.6.1	<i>The faultcode Element</i>	113
4.6.1.1	The Client Fault	114
4.6.1.2	The Server Fault	114
4.6.1.3	The VersionMismatch Fault	115
4.6.1.4	The MustUnderstand Fault	116
4.6.1.5	Non-standard SOAP Fault Codes	116
4.6.2	<i>The faultstring Element</i>	117
4.6.3	<i>The faultactor Element</i>	118
4.6.4	<i>The detail Element</i>	118
4.6.4.1	Processing Header Faults: Omitting the detail Element	119
4.6.5	<i>Final Words about Faults</i>	120
4.7	SOAP over HTTP	120
4.7.1	<i>Transmitting SOAP with HTTP POST Messages</i>	121
4.7.2	<i>HTTP Response Codes</i>	123
4.7.2.1	Success Codes	123
4.7.2.2	Error Codes	123
4.7.3	<i>Final Words about HTTP</i>	124
4.8	Wrapping Up	124
Chapter 5	WSDL	127
5.1	The Basic Structure of WSDL	129
5.2	WSDL Declarations: The definitions, types, and import Elements	132
5.2.1	<i>The XML Declaration</i>	132
5.2.2	<i>The definitions Element</i>	133

5.2.3	<i>The types Element</i>	134
5.2.4	<i>The import Element</i>	136
5.3	The WSDL Abstract Interface: The message, portType, and operation Elements	137
5.3.1	<i>The message Element</i>	137
5.3.1.1	The message Element for RPC-Style Web Services	138
5.3.1.2	The message Element for Document-Style Web Services	139
5.3.1.3	Declaring Fault Messages	140
5.3.2	<i>The portType Element</i>	142
5.3.3	<i>The operation Element</i>	143
5.3.3.1	Parameter Order within an Operation	144
5.3.3.2	Operation Overloading	145
5.4	WSDL Messaging Exchange Patterns	145
5.4.1	<i>Request/Response Messaging</i>	145
5.4.2	<i>One-Way Messaging</i>	147
5.4.3	<i>Notification and Solicit/Response Messaging</i>	147
5.5	WSDL Implementation: The binding Element	148
5.5.1	<i>SOAP Binding</i>	151
5.5.1.1	The soapbind:binding Element	151
5.5.1.2	The soapbind:operation Element	152
5.5.1.3	The soapbind:body Element	155
5.5.1.4	The soapbind:fault Element	156
5.5.1.5	The soapbind:header Element	157
5.5.1.6	The soapbind:headerfault Element	159
5.6	WSDL Implementation: The service and port Elements	160
5.6.1	<i>The soapbind:address Element</i>	161
5.7	WS-I Conformance Claims	161
5.8	Wrapping Up	162

Chapter 6 The UDDI Data Structures 165

6.1 The businessEntity Structure 168

6.1.1 *The businessEntity Element and the businessKey Attribute* 171

6.1.2 *The discoveryURL Element* 171

6.1.3 *The name Element* 172

6.1.4 *The description Element* 172

6.1.5 *The contacts Element* 173

6.1.6 *The businessServices Element* 174

6.1.7 *The identifierBag Element* 174

6.1.7.1 *The D-U-N-S Identification System* 176

6.1.7.2 *The Thomas Register Identification System* 176

6.1.8 *The categoryBag Element* 176

6.1.8.1 *The NAICS Category System* 176

6.1.8.2 *The UNSPSC Category System* 177

6.1.8.3 *The ISO 3166 Geographic Locator System* 178

6.1.8.4 *Other Categorizations* 178

6.2 The businessService and bindingTemplate Structures 180

6.2.1 *The businessService Structure* 181

6.2.1.1 *The categoryBag Element* 182

6.2.1.2 *The bindingTemplates Element* 182

6.2.2 *The bindingTemplate Structure* 182

6.2.2.1 *The accessPoint Element* 183

6.2.2.2 *The hostingRedirector Element* 184

6.2.2.3 *The tModelInstanceDetails and
tModelInstanceInfo Elements* 184

6.3 The tModel Structure 185

6.3.1 *tModels for WSDL Documents* 187

6.3.2 *tModels as Taxonomy Identifiers* 189

6.3.3 *The uddi-org:type tModel* 190

6.3.4 *Checked and Unchecked tModels* 193

6.3.5	<i>The tModel XML Schema</i>	194
6.3.5.1	The name and description Elements	194
6.3.5.2	The overviewDoc Element	195
6.3.5.3	The identifierBag and categoryBag Elements	197
6.4	The publisherAssertion Structure	198
6.5	UUID Keys	199
6.6	WS-I Conformance Claims	200
6.7	Wrapping Up	200
Chapter 7	The UDDI Inquiry API	201
7.1	General Information about UDDI SOAP Messaging	201
7.2	The Inquiry Operations	204
7.2.1	<i>Find Operations</i>	204
7.2.1.1	Using Search Elements	205
7.2.1.2	Operation Definitions and Payloads	214
7.2.2	<i>Get Operations</i>	224
7.2.2.1	The get_businessDetail Operation	225
7.2.2.2	The get_businessDetailExt Operation	227
7.2.2.3	The get_serviceDetail Operation	228
7.2.2.4	The get_bindingDetail Operation	229
7.2.2.5	The get_tModelDetail Operation	231
7.3	Wrapping Up	232
Chapter 8	The UDDI Publishing API	233
8.1	Operation Definitions and Payloads	234
8.1.1	<i>Authorization Operations</i>	235
8.1.1.1	The get_authToken Operation	235
8.1.1.2	The discard_authToken Operation	237
8.1.2	<i>Save Operations</i>	238
8.1.2.1	The save_business Operation	239
8.1.2.2	The save_service Operation	241

8.1.2.3	The <code>save_binding</code> Operation	242
8.1.2.4	The <code>save_tModel</code> Operation	243
8.1.2.5	The <code>add_publisherAssertions</code> Operation	245
8.1.2.6	The <code>set_publisherAssertions</code> Operation	246
8.1.3	<i>Delete Operations</i>	247
8.1.3.1	The <code>delete_business</code> Operation	249
8.1.3.2	The <code>delete_service</code> Operation	250
8.1.3.3	The <code>delete_binding</code> Operation	251
8.1.3.4	The <code>delete_tModel</code> Operation	252
8.1.3.5	The <code>delete_publisherAssertions</code> Operation	253
8.1.4	<i>Get Operations</i>	254
8.1.4.1	The <code>get_assertionStatusReport</code> Operation	255
8.1.4.2	The <code>get_publisherAssertions</code> Operation	259
8.1.4.3	The <code>get_registeredInfo</code> Operation	261
8.2	Fault Messages	264
8.3	Wrapping Up	268

Part IV JAX-RPC 271

Chapter 9 JAX-RPC Overview 273

9.1	The Server-Side Programming Models	274
9.1.1	<i>JAX-RPC Service Endpoint</i>	274
9.1.2	<i>Enterprise JavaBeans Endpoints</i>	275
9.2	The Client-Side Programming Models	277
9.2.1	<i>Generated Stubs</i>	277
9.2.2	<i>Dynamic Proxies</i>	280
9.2.3	<i>DII</i>	280
9.3	Other JAX-RPC Topics Covered	281
9.4	SAAJ	282
9.5	Wrapping Up	282

Chapter 10	JAX-RPC Service Endpoints	283
10.1	A Simple JSE Example	283
10.2	The JSE Runtime Environment	284
10.2.1	<i>Servlets: The Foundation of JSE</i>	286
10.2.2	<i>JNDI Environment Naming Context</i>	287
10.2.3	<i>The ServletEndpointContext and ServiceLifecycle Interfaces</i>	293
10.2.3.1	Using the Life-Cycle Methods <code>init()</code> and <code>destroy()</code>	293
10.2.3.2	Using the ServletEndpointContext Interface	296
10.3	Multi-threading and JSEs	302
10.4	Wrapping Up	303
Chapter 11	JAX-RPC EJB Endpoints	305
11.1	An Enterprise JavaBeans Primer	305
11.1.1	<i>Transactions in a Nutshell</i>	305
11.1.2	<i>Understanding EJB</i>	308
11.1.2.1	What Is EJB?	309
11.1.2.2	Why Use EJB?	309
11.1.2.3	RPC Components: Stateless, Stateful, and Entity Beans	309
11.1.2.4	Asynchronous Components: The Message-Driven Bean	311
11.1.2.5	The EJB Container System	311
11.1.3	<i>Where to Go from Here</i>	314
11.2	Enterprise JavaBeans Web Services	315
11.2.1	<i>A Simple Example</i>	315
11.2.1.1	The Endpoint Interface	315
11.2.1.2	The Bean Class	317
11.2.1.3	Deployment Descriptors and Packaging	319

11.2.2	<i>The EJB Runtime Environment</i>	320
11.2.2.1	<i>The JNDI Environment Naming Context</i>	320
11.2.2.2	<i>The SessionBean Interface</i>	321
11.2.2.3	<i>The SessionContext</i>	323
11.3	<i>Wrapping Up</i>	326
Chapter 12	JAX-RPC Client APIs	327
12.1	Generated Stubs	327
12.1.1	<i>The Endpoint Interface</i>	328
12.1.2	<i>The Generated Stub</i>	330
12.1.3	<i>Service Interfaces</i>	331
12.1.4	<i>Using a Generated Stub in J2EE</i>	332
12.2	Dynamic Proxies	338
12.2.1	<i>Using a Dynamic Proxy</i>	339
12.2.2	<i>Under the Covers</i>	344
12.3	DII	344
12.3.1	<i>Using DII with a WSDL Document</i>	344
12.3.2	<i>Using DII without a WSDL Document</i>	349
12.3.3	<i>Using One-Way Messaging with DII</i>	349
12.3.4	<i>JAX-RPC Standard Properties and Constants</i>	349
12.4	Wrapping Up	352
Chapter 13	SAAJ	353
13.1	A Simple SAAJ Example	355
13.2	Creating a SOAP Message	356
13.2.1	<i>The MessageFactory Class</i>	357
13.2.1.1	<i>The newInstance() Method</i>	357
13.2.1.2	<i>The createMessage() Method</i>	357
13.2.1.3	<i>The createMessage Method with Parameters</i>	358
13.2.2	<i>SaaJOutputter Classes</i>	360

13.2.3 The SOAPMessage Class 360

13.2.3.1 The writeTo() Method 361

13.2.3.2 The getSOAPBody() and getSOAPHeader() Methods 361

13.2.3.3 The getProperty() and setProperty() Methods 362

13.3 Working with SOAP Documents 363

13.3.1 The SOAPPart and SOAPEnvelope Types 363

13.3.2 The SOAPFactory Class and Name Types 364

13.3.3 The SOAPElement Type 367

13.3.4 The Node Type 370

13.3.5 The SOAPHeader Type 372

13.3.6 The SOAPHeaderElement Type 375

13.3.7 The SOAPBody Type 376

13.3.8 The SOAPBodyElement Type 377

13.3.9 The Text Type 378

13.3.10 The SOAPConstants Class 378

13.3.11 The SOAPException Class 379

13.3.12 The SOAPFactory and SOAPElement Types 380

13.4 Working with SOAP Faults 382

13.4.1 The SOAPFault Type 382

13.4.2 The Detail Type 387

13.4.3 The SOAPFaultElement Type 387

13.4.4 The DetailEntry Type 388

13.5 Sending SOAP Messages with SAAJ 388

13.6 SAAJ 1.2 and DOM 2 390

13.7 Wrapping Up 393

Chapter 14 Message Handlers 395

14.1 A Simple Example 396

14.1.1 Defining a Message-Handler Class 396

14.1.2 Defining a WSDL Document 400

14.1.3 Generating the Service and Endpoint Interfaces 402

14.1.4	<i>Configuring Message Handlers</i>	403
14.1.5	<i>Using Message Handlers in a J2EE Component</i>	404
14.2	Handler Chains and Order of Processing	406
14.2.1	<i>Return Values and Order of Processing</i>	411
14.2.2	<i>Exceptions and Order of Processing</i>	414
14.2.2.1	<i>The SOAPFaultException Type</i>	414
14.2.2.2	<i>The JAXRPCException Type</i>	415
14.3	The Handler Runtime Environment	416
14.3.1	<i>Statelessness and Multi-threading</i>	416
14.3.2	<i>JNDI Environment Naming Context</i>	417
14.3.3	<i>A Message Handler's Life Cycle</i>	419
14.3.4	<i>The MessageContext Type</i>	422
14.4	Wrapping Up	425
Chapter 15	Mapping Java to WSDL and XML	427
15.1	Mapping WSDL to Java	428
15.1.1	<i>WSDL-to-Endpoint Interfaces</i>	429
15.1.2	<i>Declaring Multiple Parts</i>	430
15.1.3	<i>Defining Multiple Operations</i>	432
15.1.4	<i>One-Way Messaging</i>	433
15.2	Mapping XML Schema to Java	434
15.2.1	<i>XML Schema Built-in Simple Types</i>	435
15.2.2	<i>XML Schema Complex Types</i>	436
15.2.3	<i>Arrays</i>	440
15.2.4	<i>Enumerations</i>	441
15.2.5	<i>SOAPElement: Supporting Non-standard Types in Document/Literal Encoding</i>	443
15.2.6	<i>SOAPElement: The xsd:any Element</i>	450
15.2.7	<i>Nilable Elements</i>	451
15.3	Holders	453
15.3.1	<i>Pass-by-Copy: IN Parameters</i>	453
15.3.2	<i>Pass-by-Reference: INOUT and OUT Parameters</i>	454
15.3.3	<i>Holders: Supporting INOUT and OUT Parameters in JAX-RPC</i>	456

15.3.4 Mapping Holder Types from WSDL 457

15.3.4.1 Examples 458

15.3.4.2 Rules for Mapping parts to Method Parameters 461

15.3.4.3 Standard Holder Types 462

15.3.4.4 Generated Holder Types 463

15.4 Faults and Java Exceptions 467

15.4.1 WSDL Faults and Application Exceptions 467

15.5 Wrapping Up 473

Part V JAXR 475

Chapter 16 Getting Started with JAXR 477

16.1 Using a UDDI Test Registry 478

16.2 Connecting to a UDDI Registry 479

16.2.1 Obtaining a ConnectionFactory 481

16.2.2 Configuring the ConnectionFactory 481

16.2.3 Connecting to the UDDI Registry 484

16.2.4 Authenticating to a UDDI Registry 484

16.2.5 Obtaining a JAXR Connection in J2EE 486

16.3 Using the RegistryService and BusinessLifeCycleManager 487

16.3.1 The RegistryService Interface 489

16.3.2 Using the BusinessLifeCycleManager 491

16.3.2.1 The LifeCycleManager Interface 491

16.3.2.2 The BusinessLifeCycleManager Interface 494

16.4 The BulkResponse Type 496

16.4.1 The BulkResponse Interface 496

16.4.2 Handling the BulkResponse in Examples 497

16.5 Exceptions 498

16.6 Wrapping Up 498

Chapter 17 The JAXR Business Objects 499

- 17.1 The RegistryObject Interface 501**
 - 17.1.1 The UUID Key 502**
 - 17.1.1.1 Understanding UUIDs 503
 - 17.1.1.2 The Key Interface 504
- 17.2 The Organization Information Object 504**
 - 17.2.1 businessEntity 508**
 - 17.2.2 ExternalLinks 509**
 - 17.2.2.1 The ExternalLink Information Object 511
 - 17.2.3 Name and Description 512**
 - 17.2.3.1 The InternationalString Interface 512
 - 17.2.3.2 The LocalizedString Interface 513
 - 17.2.3.3. Working with InternationalString Objects 514
 - 17.2.4 User 516**
 - 17.2.4.1 Creating a New User Object 520
 - 17.2.4.2 Setting a Person's Name 521
 - 17.2.4.3 Adding Descriptions 521
 - 17.2.4.4 Adding E-Mail Addresses 522
 - 17.2.4.5 Adding Phone Numbers 523
 - 17.2.4.6 Adding a Postal Address 524
 - 17.2.5 Classification 527**
 - 17.2.5.1 The Standard UDDI Classifications 527
 - 17.2.5.2 The Classification Interface 534
 - 17.2.5.3 The ClassificationScheme Interface 535
 - 17.2.5.4 Reading Classification and ClassificationScheme Objects 537
 - 17.2.6 External Identifiers 538**
 - 17.2.6.1 Managing External Identifiers 541
 - 17.2.6.2 The ExternalIdentifier Interface 542
 - 17.2.7 Services 543**
- 17.3 Wrapping Up 544**

Chapter 18 The JAXR Technical Objects 545

- 18.1 The Service and ServiceBinding Information Objects 545**
 - 18.1.1 Creating a New Service Object 548*
 - 18.1.2 Creating a New ServiceBinding Object 549*
 - 18.1.3 The Service Interface 550*
 - 18.1.4 The ServiceBinding Interface 551*
- 18.2 The Concept Information Object 553**
 - 18.2.1 Creating a New Concept 556*
 - 18.2.1.1 Create a Concept Instance 556*
 - 18.2.1.2 Set the Concept Name 556*
 - 18.2.1.3 Set the Overview Document 557*
 - 18.2.1.4 Assign the wsdlSpec Classification 558*
 - 18.2.1.5 Assign the WS-I Conformance Claim 558*
 - 18.2.1.6 Save the Concept Object 559*
 - 18.2.2 The Concept Interface 559*
 - 18.2.2.1 Concept Methods and Taxonomy Browsing 560*
 - 18.2.2.2 RegistryObject Methods and WSDL Concepts 563*
- 18.3 The SpecificationLink Information Object 564**
 - 18.3.1 Using SpecificationLink Objects 565*
 - 18.3.1.1 Create a Service and a ServiceBinding 567*
 - 18.3.1.2 Create a SpecificationLink 568*
 - 18.3.2 The SpecificationLink Interface 569*
- 18.4 The Association Information Object 570**
 - 18.4.1 Creating an Association 570*
 - 18.4.1.1 Find the Concept Object Representing the Association Type 572*
 - 18.4.1.2 Create an Association Instance 574*
 - 18.4.1.3 Save the Association 574*
- 18.5 Predefined Enumerations 575**
 - 18.5.1 The AssociationType Enumeration 575*
 - 18.5.2 The URLType Enumeration 576*
 - 18.5.3 The ExtensibleObject and Slot Interfaces 577*
- 18.6 Wrapping Up 578**

Chapter 19 The JAXR Inquiry and Publishing APIs 581

19.1 Mapping JAXR to the UDDI Inquiry API 581

19.1.1 Using Search Criteria 586

- 19.1.1.1 The namePattern Criteria 586
- 19.1.1.2 The classifications Criteria 587
- 19.1.1.3 The externalIdentifiers Criteria 588
- 19.1.1.4 The specifications Criteria 589
- 19.1.1.5 The externalLinks Criteria 590
- 19.1.1.6 The findQualifiers Criteria 591

19.1.2 The findXXX() Methods 595

- 19.1.2.1 The findAssociations() Method 595
- 19.1.2.2 The findCallerAssociations() Method 596
- 19.1.2.3 The findOrganizations() Method 597
- 19.1.2.4 The findServices() Method 597
- 19.1.2.5 The findServiceBindings() Method 598
- 19.1.2.6 The findClassificationSchemes() Method 598
- 19.1.2.7 The findClassificationSchemeByName()
Method 598
- 19.1.2.8 The findConcepts() Method 599
- 19.1.2.9 The findConceptByPath() Method 599

19.2 Mapping JAXR to the UDDI Publishing API 599

19.3 Wrapping Up 601

Part VI JAXP 603

Chapter 20 SAX2 605

20.1 Parsing with SAX: XMLReaderFactory and XMLReader 606

20.2 The ContentHandler and DefaultHandler Interfaces 608

- 20.2.1 The startDocument() and endDocument() Methods 612
- 20.2.2 The startElement() and endElement() Methods 613
- 20.2.3 The startPrefixMapping() and endPrefixMapping()
Methods 617
- 20.2.4 The characters() Method 619

20.2.5 *Other ContentHandler Methods* 621

20.2.6 *Other SAX2 Listener Interfaces* 621

20.3 **Validating with W3C XML Schema** 623

20.4 **Wrapping Up** 624

Chapter 21 DOM 2 625

21.1 **Parsing with DOM: DocumentBuilderFactory and DocumentBuilder** 627

21.2 **Nodes** 628

21.2.1 *Type-Dependent Properties* 632

21.2.2 *XML Name Methods* 634

21.2.3 *Attribute Methods* 636

21.2.4 *The Owner Document* 638

21.2.5 *Navigating a Node Tree* 638

21.2.6 *Methods for Child Management* 641

21.2.7 *Other Methods* 641

21.3 **Building a DOM Document** 641

21.4 **Copying Nodes** 649

21.5 **Wrapping Up** 651

Part VII Deployment 653

Chapter 22 J2EE Deployment 655

22.1 **Overview of the J2EE Deployment Process** 655

22.2 **J2EE Web Services Deployment** 657

22.2.1 *Starting with a J2EE Endpoint* 657

22.2.2 *Starting with WSDL* 660

22.2.3 *JAX-RPC Mapping Files* 662

22.2.4 *Deployment Descriptors for J2EE Components* 663

22.3 **Deploying JSEs** 663

22.3.1 *Packaging JSEs in a WAR File* 663

22.3.2	<i>The web.xml File</i>	665
22.3.2.1	<i>Configuring the ServletContext</i>	667
22.3.2.2	<i>Configuring the JNDI ENC</i>	668
22.3.2.3	<i>Configuring Other Aspects of a JSE</i>	669
22.4	Deploying EJB Endpoints	672
22.4.1	<i>Declarative Transaction Attributes</i>	672
22.4.2	<i>Declarative Security Attributes</i>	674
22.5	Service References	677
22.5.1	<i>The service-ref-name Element</i>	679
22.5.2	<i>The service-interface Element</i>	680
22.5.3	<i>The wsdl-file and service-qname Elements</i>	681
22.5.4	<i>The jaxrpc-mapping-file Element</i>	683
22.5.5	<i>The port-component-ref Element</i>	683
22.5.6	<i>The Display Elements</i>	685
22.5.7	<i>The handler Element</i>	685
22.5.7.1	<i>The handler-name Element</i>	686
22.5.7.2	<i>The handler-class Element</i>	687
22.5.7.3	<i>The init-param Elements</i>	687
22.5.7.4	<i>The soap-header Elements</i>	688
22.5.7.5	<i>The soap-role Elements</i>	688
22.5.7.6	<i>The port-name Element</i>	690
22.5.7.7	<i>Proper Processing by Handlers</i>	690
22.5.7.8	<i>The Handler Display Elements</i>	691
22.6	Wrapping Up	692
Chapter 23	Web Service Descriptors	693
23.1	<i>The wsdl-file and wsdl-port Elements</i>	694
23.2	<i>The port-component-name Element</i>	696
23.3	<i>The service-endpoint-interface Element</i>	697
23.4	<i>The service-impl-bean Element</i>	698
23.5	<i>The jaxrpc-mapping-file Element</i>	700
23.6	<i>The handler Element</i>	701
23.6.1	<i>The handler-name Element</i>	702
23.6.2	<i>The handler-class Element</i>	702

23.6.3	<i>The init-param Elements</i>	702
23.6.4	<i>The soap-header Elements</i>	703
23.6.5	<i>The soap-role Elements</i>	704
23.6.6	<i>Proper Processing by Handlers</i>	705
23.6.7	<i>The Display Elements</i>	706
23.7	Wrapping Up	706
Chapter 24	JAX-RPC Mapping Files	709
24.1	Conditions for a Lightweight JAX-RPC Mapping File	709
24.2	A Lightweight Example	710
24.3	A Heavyweight Example	713
24.4	Anatomy of a Mapping File	716
24.4.1	<i>The java-wsdl-mapping Element</i>	717
24.4.2	<i>The package-mapping Element</i>	717
24.4.3	<i>The java-xml-type-mapping Element</i>	720
24.4.4	<i>The exception-mapping Element</i>	723
24.4.5	<i>The service-interface-mapping Element</i>	725
24.4.6	<i>The service-endpoint-interface-mapping Element</i>	728
24.5	Wrapping Up	735
Appendices	Introduction	737
Appendix A	XML DTDs	739
Appendix B	XML Schema Regular Expressions	745
B.1	Character Sets	746
B.2	Quantifiers	746
B.3	Other Meta-characters	747
B.4	Real-World Examples	748
B.4.1	<i>ISBN (International Standard Book Number)</i>	748
B.4.2	<i>United States Zip Code</i>	749

B.4.3 United States Phone Number	749
B.4.4 United Kingdom Postal Code	749

Appendix C Base64 Encoding 751

Appendix D SOAP RPC/Encoded 755

D.1 The soap:encodingStyle Attribute	756
D.2 The Operation Structs	756
D.3 Simple Types	759
D.4 Complex Types	760
D.5 Array Types	761
D.5.1 Array Size	765
D.5.2 Other Features of Arrays	766
D.6 References	767
D.7 Wrapping Up	770

Appendix E SOAP Messages with Attachments 771

E.1 Understanding MIME	772
E.2 Using MIME with SOAP	775
E.3 Wrapping Up	778

Appendix F SAAJ Attachments 781

F.1 The Java Activation Framework	782
F.1.1 DataHandler	783
F.1.2 DataContentHandler	785
F.1.3 DataSource	789
F.2 SAAJ and JAF: AttachmentPart	793
F.2.1 Data Objects	795
F.2.2 The createAttachmentPart() Method	795
F.2.3 The setContent() and setDataHandler() Methods	796

E2.4 The getContent() Method 798

E2.5 The MIME Header Methods 799

F.3 The SOAPPart 803

F.3.1 XSLT 805

F.3.2 TrAX 805

F.3.3 Using a StreamSource 805

F.3.4 DOMSource 806

F.3.5 SAXSource 807

F.4 The SOAPEnvelope 807

*F.4.1 The getHeader(), getBody(), addHeader(), and
addBody() Methods* 808

F.4.2 The createName() Method 808

F.5 Wrapping Up 809

Appendix G JAX-RPC and SwA 811

G.1 JAF Revisited: DataContentHandler and DataSource Types 811

G.2 A Simple Example 812

G.3 Mapping MIME Types to Java 816

G.3.1 The multipart/ MIME Type* 818

G.3.2 The text/xml and application/xml MIME Types 820

*G.3.3 MIME Attachments as Return Types, and INOUT and OUT
Parameters* 823

G.4 Using DataHandler and DataSource Types 825

G.5 Wrapping Up 828

Appendix H Using JAX-RPC DII without a WSDL Document 829

Bibliography 837

Index 843