Index

A

- **access control** 295, 320
  - at method level 323
  - EJB endpoints 323
  - security granularity 320
  - security-role element 320
  - specifying 320
  - Web service endpoints 322
- **adventure builder application** 11–17
  - business collaborations 358
  - canonical data model 375
  - centralized manager 373–375
  - communication
    - architecture 349–351
    - synchronous to asynchronous 364–366
    - technologies 346
  - customer Web site 340
  - customers 340
  - designing 339
    - endpoints 351–358
  - endpoint types 352
  - external partners 340
  - functions 340–342
  - granularity 353
  - handling exception conditions 379–381
  - interface development 351
  - layered view 355
  - message exchanges 346
    - correlating 360
    - format 348
    - split and join 361–362
  - order fulfillment processing 344
  - order processing center
    - external communication 347
    - integration with partners 347
    - JMS 347
    - module communication 347
    - modules
      - Customer Relations Manager 343
      - Finance 343
      - Order Filler 344
      - Order Receiver 342
      - Workflow Manager 342
  - order processing center (OPC) 340
  - parameter passing 353–355
  - participants 340
  - processing flow 344
  - publishing 358
  - request delegation 357
  - synchronous communication 357
  - Web service clients 357
  - Web service interactions 352
- **anyType** 355
- **asynchronous communication** 24, 363
  - connectors 260
  - converting from synchronous 364–366
  - correlation identifier 95
  - delegating requests 92
  - EIS access 260
  - handling errors 379
  - integration 271
  - message-driven beans 258
- **attachments** 89, 106
- **authentication** 295, 299
  - basic 301
  - data origin authentication 329
  - digital certificate 295
  - EIS resource 304
    - application-managed 304
    - container-managed 304
  - form-based 301
  - mutual 299, 301
  - on EIS tier 303
  - on EJB tier 302–303
  - on Web tier 301–302
  - peer-based entity authentication 329
  - principal identity 295
  - RMI-IIOP 303
  - single sign-on 302
- **authorization** 304
  - consistency across components 324

declarative 305, 320
programmatic 306–307
Web service endpoints 320, 324

B

basic authentication 223, 301, 309
client programming model 315
Web services endpoint 311
binding 37
data
integration 272
dynamic 104
encoded 87
literal 87
parameters 74
static 104
XML documents 100
business information entities (BIES) 134
business logic
delegating requests to 63, 357
in processing layer 99
lengthy processing 107
with document-handling logic 157
business partner integration 14
aggregating data 16
Business Process Execution Language (BPEL) 288
business vocabulary 132

C

canonical data model 276, 375
DAO 280
data filtering 282
data translation 279
interfaces to 279
schema 278
XML 276
Canonical XML 127
centralized manager 373
adventure builder application 373–375
client
accessing a service 104
accessing WSDL 211
binding to service location 104
communication 200–201, 210, 357
DII 212, 215
dynamic proxy 212, 214
guidelines 217–218
HTTP protocol 201
JAX-RPC stub classes 200, 212
JMS 201
RMI/IIOP 201
configuring security 224
configuring stub 223
conversational state 207, 235
synchronizing 237
creating service delegate 211
discovering service implementation 104
discovering service interface 104
diversity 16
EJB component
conversational state 236
handling return values 228
graphical interface 207
interoperability 199–200
invoking a service 211
J2EE client 202–207
EJB components 206
generated stubs 213
JNDI 219
locating a service 220
packaging 240–243
Web component 227
J2ME client 202, 208–210, 214
bandwidth 210
connected mode 209
disconnected mode 209
JAX-RPC runtime 209
locating a service 220
MIDlet 244
packaging 244–245
J2SE client 202, 207–208
DII 222
locating a service 219–220
packaging 243–244
Swing components 238
JAX-RPC APIs 213
locating a service 219–223
getPort method 220
mapping types 225
network access 207
non-Java client 202

- packaging 240
- return values 227, 229–230
    - JSP 229
- security 314–316
- service exceptions 232–235
- stand-alone client 204
- state 86
- system exceptions 230–232
- using DII 215
- using dynamic proxies 214
- using stub classes 213
- Web service scenarios 202–204
- WSDL-to-Java tools 226

Command pattern
- multiple document types 371

Commerce XML (CXML) 126

Common Business Language (CBL) 126

Common Client Interface (CCI) 261

Common Secure Interoperability (CSIv2) 303

communication technologies
- adventure builder application 346

confidentiality mechanisms 307
- transport layer 307

connector 250, 258
- EIS 269
    - custom systems 284
- integration layer 272
- JAX-RPC 269
- nonrelational data 275
- resource adapter 260
- Web service interface 269

container-based component management 18

container-managed persistence 253–254
- data mapping 274

container-managed relationships 254

context information 366
- adding to messages 367–371
- correlation identifier 366
- in message contents 368
- in service interface 367
- in SOAP header 368
- security 366

conversational state
- client management of 235
- synchronizing 237

correlation identifier 95, 98, 359, 364
- context information 366

customer relations management (CRM) 12

D

data
- aggregation 16
- integration 248, 250
    - filtering 282–283
    - mapping 274–275
    - technologies 253
    - transformation 275–282
        - canonical data model 276
        - transition façade 282

data access object 157

data access object (DAO) 272

data caching 72, 98

data mapping 274

data model 143

data origin authentication 329

database systems
- integrating 248, 250

demilitarized zone (DMZ) 348

deployment descriptors 297
- auth-constraint element 322
- ejb-jar.xml 114
- ejb-ref element 300
- external references 300
- handlers 115
- hybrid authentication 313
- identity 317
- jaxrpc-mapping-file element 115
- method-permission element 323
- port information 115
- port-component-name element 115
- resource-ref element 304
- security policy 314, 325
- security-constraint element 322
- security-role element 320
- security-role-ref element 307
- service-endpoint-interface element 115
- service-impl-bean element 115
- unchecked element 323
- Web services information 113
- web.xml 114
- wsdl-file element 115
- wsdl-port element 115

deserialization 87, 164

DII 210, 212, 215, 222
- Call interface 216
    - configuring 223

    - system exceptions 231
- **document**
  - handling and business logic 157
  - handling. See XML
- **Document Object Model (DOM)** 127, 164
- **document processing**
  - fragmenting 153
- **Document Type Definition (DTD)** 30, 123
- **Document Type Definition. See DTD**
- **document-centric processing model** 151
  - DOM
    - XPath 152
  - schema-derived classes 170
- **DOM** 43, 164, 167, 175
  - caching 194
  - limitations 168
  - serialization 168
  - validating document 141
  - when to use 169, 184
  - with XPath 175
- **dom4j** 168
  - when to use 185
- **DTD**
  - data types 125
  - namespaces 125
  - syntax limitations 125
  - validating document 139, 141
- **dynamic aggregation of data** 16
- **dynamic binding** 104
- **dynamic invocation interface (DII)** 45
  - See DII
- **dynamic proxy** 45, 50, 210, 212, 214, 222
  - getPort method 214
    - system exceptions 231

E

- **EAI** 248, 265
  - data mapping 274
- **EIS**
  - connector access 259, 269, 271
  - DAO 272
  - integration 270
    - custom systems 284
    - metadata 285
  - integration technologies 252–262
  - relational databases
    - accessing with JDBC 253
  - Web service interfaces 266
- **EJB service endpoint** 49, 51, 67
  - declarative transaction services 69
  - deployment descriptor 114
  - within EJB tier 68
- **EJB-JAR file** 114, 117
- **Electronic Business with XML (ebXML)** 126
- **electronic data interchange (EDI)** 14
- **endpoint interface** 50
- **endpoints** 37
  - conversational state 235
  - delegating requests 94, 357
  - deploying 113–119
  - designing 351–358
  - EJB 49, 62
  - granularity 353
    - stateless session beans 71
  - HTTP session access 70
  - idempotent 377
    - designing 378
  - JAX-RPC 49, 62
  - method-level access 69
  - packaging 113–119
  - transforming XML document 106
  - types 67
    - by tier 68
    - considerations when choosing 68–70, 352
    - for existing application 68
    - multi-threaded access 68
    - transaction considerations 69
  - validating XML document 90, 106
- **enterprise application integration (EAI)** 247
  - See EAI
- **enterprise beans**
  - integration 271
- **enterprise information systems (EIS)** 15, 247
  - See EIS
- **Enterprise JavaBeans container-managed persistence (CMP)** 253
- **enterprise resource planning (ERP)** 247
- **entity catalog** 179
- **entity resolution** 125, 178–182
  - entity catalog 179
  - validating document 143
- **entity resolver** 179
- **errors**

See exceptions
evolution 286–288
exceptions
for non-Java platforms 80
handling 80, 376–381
in asynchronous communication 379
in SOAP messages 80
inheritance 83
Java-specific 82
mapping of 233
RemoteException 81
service
client handling of 232–235
WSDL fault elements 232
service-specific 81
stack trace 84
system 81
client handling of 230–232
wrapping 83
Extensible HyperText Markup Language (XHTML) 126
eXtensible Markup Language (XML)
See XML
eXtensible Stylesheet Language (XSL) 127
eXtensible Stylesheet Language Transformations (XSLT) 127, 164
external entity references 179

F

flexible mapping design 155, 160
Form-based authentication 301
fragmenting
incoming XML document 153

G

global transaction 9
granularity 353

H

handlers 85–86
client state 86
context information 368
MessageContext interface 369
interoperability 89
message-level security 335
port-specific state 86
HTML 3
HTTP 200
basic authentication 301
peer-based entity authentication 329
Transport Layer Security (TLS) 328
HTTP session access 70
HTTPS 296
message-level security 337
mutual authentication 301
hybrid authentication 312
HyperText Markup Language (HTML) 122

I

idempotent endpoints 377
designing 378
identity
getCallerPrincipal 319
getUserPrincipal 319
invocation identity 316
propagating 316–319
protection domains 318
integration
application 248–250
business process 249
container-managed persistence 253–254
data 248, 250
data mapping 274–275
filtering 282–283
transformation 275–282
canonical data model 276
transition façade 282
design strategy 264
comparison 273
J2EE technologies 265
document exchange 249
EAI 248
EIS 15, 270
enterprise beans 271
evolution 286–288
example scenarios 249–252
design strategy 266
integration architecture 272

integration layer 263
EIS 271
Web services 266
J2EE Connector architecture 258
JDBC 253
JDO 253
JMS 258, 271
legacy systems 15, 252
message-driven beans 258
messaging technologies 255–259
metadata 285
one-off solution 263, 283
registry 286
requirements 250
screen scraping 284
versioning 286–288
WSDL 268
XML documents 249

integration technologies 252–262

integrity mechanisms 307
transport layer 307

interaction layer 64, 155
adventure builder application 355
binding 100
centralized manager 373
consolidating interactions 373
design 66
receiving requests 90–92
response generation 98
separation of layers 100
XML document
transformation 357
validation 357

interactions
consolidating 373

Internet Inter-ORB Protocol (IIOP) 33

interoperability 4–6, 9, 86–89
attachments 89
deserialization 87
encoded binding 87
handlers 89
Java-MIME mappings 89
literal binding 87
messaging styles 87
serialization 87
service design 87
SOAPBuilders 87
WS-I protocols 86

J

J2EE 1.4 SDK 88

J2EE connector 250
See also connector

J2EE Connector architecture 258–259
Common Client Interface (CCI) 261
system-level contracts 261–262

J2EE platform 1
asynchronous communication 24
authentication mechanisms 309
benefits 19–20
choice in servers, tools, and components 20
enhanced application development productivity 19
simplified security model 20
clients 17
designing 205–207
communication
Internet protocols 23
Remote Method Invocation (RMI) protocols 24
Compatibility Test Suite 18
component technologies
client 22
Enterprise JavaBeans (EJB) 22
Web 22
component-based development 18
container services
deployment service 23
naming service 23
security service 23
transaction service 23
container-based component management 18
deployment descriptor 113
EJB service endpoint 49
exceptions
handling mechanisms 84
integration technologies 252–262
design strategy 265
JAX-RPC service endpoint 49
multi-tier model 17
packaging structure 113, 117–119
parameter types
supported 73
parameters
MIME-encoded content 75

- SDK 19
- security
  - client programming model 315
- security model 297–308
- service endpoints 49
- specifications 18
- using Web services 50
- Web services 2
  - client 50
  - endpoints 49
  - implementation 59
  - port component 50
  - portability 50
  - standards 24, 54
  - technologies 17–24, 49, 54
- WS-I Basic Profile 51
- WS-I standards 19
- XML data-binding facility 52

J2EE specifications xxi

Java 2 Platform, Micro Edition (J2ME) 208

Java API for XML Messaging (JAXM) 112
- See JAXM

Java API for XML Registries (JAXR) 47

Java API for XML-based RPC (JAX-RPC) 44

Java APIs for XML Processing (JAXP) 41

Java Application Descriptor (JAD) file 245

Java Architecture for XML Binding (JAXB) 52

Java Authentication and Authorization Service (JAAS) 315

Java Business Integration (JBI) SPI 288

Java Community Process (JCP) 1

Java Data Objects (JDO) 253
- See JDO

Java Message Service (JMS) 201, 255, 257

Java Message Service API 24

Java Remote Method Protocol (JRMP) 33

Java Standard Tag Libraries (JSTL) 227

Java Web Start 208

JavaMail API 24

JavaServer Pages (JSP) 227

javax.xml.soap.SOAPElement API 110

JAXB 76, 164, 169, 175, 279
- binding compiler 53
- binding framework 53
- binding language 53
- content tree 169
- schema-derived classes 170–171
- when to use 185
- XML data binding 52

JAXM
- when to use 112

JAXP 164
- DOM model 41
- SAX model 41
- support for XML schemas 41
- XSLT 41

JAXR
- access business registries 48
- client registry access 215
- pluggable provider 48
- registry-specific provider 48

JAX-RPC
- APIs 213
- client-side runtime 59, 203
  - packaging 240
- dynamic invocation interface 45
- dynamic proxy 45
- exception mapping 233
- generated stub classes 44
- getPort method 219
- handlers
  - extracting context information 368
  - message-level security 335
- locating a service 219
- mapping file 117, 241
- mappings 44, 65
- modes of operation 46
- non-blocking RPC invocation mode 46
- non-Java clients of 46
- one-way RPC mode 46
- parameter mapping 225
- passing XML documents 137
- runtime 45
- secure interactions 308–326
- security
  - guidelines 324–325
- service endpoint 49, 67
  - deployment descriptor 114
  - transaction logic 69
  - within Web tier 68
- SOAP 44
- SOAP message handlers 46
- stub classes 213
  - system exceptions 231
- supported types 225
- synchronous request-response mode 46

use by client application 44
value types
nested 74
value types as parameters 74
JDBC 253
client API 253
data mapping 275
integration 275
relational data 275
system-level contract 253
JDO 255
data mapping 274
JDOM 168
when to use 185
JMS
See Java Message Service
integration 271
queue 95, 107
receiving XML document 137
sending XML document 137
topic 95, 107
JNDI 219
JTA 69

K

Kerberos service ticket 300

L

legacy system integration 6, 15
legacy systems
integrating 252, 283
screen scraping 284
transforming data 275
Lightweight Directory Access Protocol (LDAP) 285

M

markup language 122
XML 29
markup vocabulary 123
marshalling 126, 165
Mathematical Markup Language (MathML) 126
message exchanges
adventure builder application 346
business collaborations 358
confidentiality 307
context information
adding 367–371
correlating messages 359
format 348
handling duplicate messages 377
HTTP 329
integrity 307
joining messages 361–362
message-level security 330
secure channels 296
splitting messages 361–362
Message Oriented Middleware (MOM) 255
message router 256
message signatures 307
MessageContext interface 369
message-driven beans 258, 364
message-level security 296, 326–337
context information 366
data origin authentication 329
emerging standards 331
applying 332, 334–337
HTTPS 337
portability 336
security granularity 330
using JAX-RPC handlers 336
using SAAJ 336
with transport layer security 331
messaging styles 256
messaging systems
integration 255, 257
JMS 257
message-driven beans 258
metadata
directory service 286
integration 285
LDAP 285
See context information
mutual authentication 299, 309
client programming model 316
Web services endpoint 311

N

namespaces 127
North American Industry Classification System (NAICS) 103

O

Oasis Consortium 182
object-centric processing model 152–153
    domain-specific objects 153
    schema-derived classes 170
object-oriented design
    schema design 135
object-relational (O/R) mapping 253, 255
overloaded methods
    as SOAP messages 78
    example representation in WSDL 77
    exposing to clients 77
    in WSDL description 77
    invoking 79
    representation of 77
    WSDL method names 79

P

packaging 117–119
parameters
    binding standard types 74
    Java objects 73
    Java types 73
    Java-MIME mappings 76, 89, 107
    JAX-RPC value types 74, 353–355
    mapping to XML 72
    MIME-encoded content 75
    nonstandard type mappings 76
    passing 353–355
    SOAP document fragments 76
    XML documents 76, 353–355
peer-based entity authentication 329
permissions
    method-level access 69
point-to-point messaging 256
polling 98
port
    deployment descriptor 115
    in deployment descriptor 114
    state 86
port component 50
port type
    binding 37
port types 37
portability 1
    dynamic proxy 215
    port component definition 50
ports 37
    port instance 220
    service 37
preprocessing
    of requests 62, 90
principal identity 295
principal mapping 304
processing layer 64, 156
    adventure builder application 355
    as Web service client 100
    business logic 99
    delegating requests to 92
    design 99–101
    flexible mapping 156
    separation of layers 100
protection domains 300
    authentication boundary 300
    identity propagation 318
publish and subscribe messaging 256
publishing 101–105, 358
    registry 101
    XML schema 102

R

registry 5, 62, 101
    access to 48
    concepts 102–103
    corporate 102
    data repository 102
    implementation scenarios 103
    integration 286
    public 102
    security considerations 102
    See also publishing
    taxonomy 103
    using DII 215
    XML schema 102

- **relational database management systems (RDBMS)** 253
- **remote access**
  - impact 70
- **RemoteException** 81
- **request**
  - delegating 92
- **requests**
  - delegating
    - asynchronous 94
    - synchronous 94
    - validation 95
  - mapping from documents to objects 90
  - receiving 90, 375
- **resource adapter** 260
- **Resource Description Framework (RDF)** 126
- **resources**
  - protected 306
- **responses**
  - generation 98
  - to client 63
- **return values**
  - client handling 227–230
  - See parameters

S

- **SAAJ** 112
  - message exchanges 49
  - message-level security 336
  - SOAP messages 48
  - when to use 112
- **sample application**
  - obtaining xix
- **SAX** 42, 164–165, 175
  - caching 191
  - custom serialization handler 166
  - document parsing 166
  - document processing 165
  - entity resolver 191
  - event handlers 165
  - stream processing 165–167
  - validating document 141
  - when to use 167, 184
- **Scalable Vector Graphics (SVG)** 126
- **schema** 123, 126
  - canonical data model 278
  - custom 126
  - data transformation 276
  - derived classes
    - aggregation 171
    - in document-centric processing model 170
    - in object-centric processing model 170
  - design 131, 135
    - patterns 135
  - generic 150
  - use of term 126
- **screen scraping** 284
- **Secure Socket Layer (SSL)** 201
- **Secure Sockets Layer (SSL)** 298
  - Web services
    - securing interactions 309
- **security** 9
  - access control 295
  - attributes 305
  - authentication 295, 299
    - basic 301
    - form-based 301
    - JAX-RPC guidelines 324–325
    - mutual 299, 301
    - on EIS tier 303
    - on EJB tier 302–303
    - on Web tier 301–302
    - principal identity 295
    - RMI-IIOP 303
    - single sign-on 302
    - WAR file 325
  - authorization
    - declarative 305
    - JAX-RPC guidelines 324–325
    - programmatic 306
    - Web services 320–324
  - chain of calls 294
    - authentication 299
    - identity propagation 316
  - concepts 292
  - confidentiality 307
  - credential 300
  - declarative 297, 324
  - digital certificate 295
  - integrity 307
  - invocation identity 316
  - issues 10
  - J2EE platform 297–308

    - authentication 309
  - JAX-RPC guidelines 324–325
  - mechanisms
    - authorization 304
    - See also
      - authorization
  - message exchanges
    - HTTPS 296
    - secure channels 296
  - message-level security 296, 326–337
  - method-level access 69
  - operational environment 297
  - principal mapping 304
  - programmatic 298, 321, 324
  - protection domains 300
  - requirements 293–295
  - roles 305
  - security domains 294, 298
  - service endpoint access 69
  - use cases 292
  - Web services 291
    - basic authentication 311
    - certificates 309
    - endpoint type 310
    - hybrid authentication 312
    - identity propagation 316–319
    - interactions 308–326
    - mutual authentication 311
    - policy 313
    - tier mechanisms 310
- **serialization** 87, 126, 165
  - Java serialization 126
- **service**
  - ports 37
- **service endpoint interface** 50
  - design 66
  - publishing 62
  - See service interface
- **service implementation bean** 50
- **service interface**
  - data caching 72
  - definition 66
    - comparing approaches 67
    - interoperability 67
    - Java-to-WSDL 66
    - service evolution 67
    - WSDL-to-Java 66
  - endpoint types 67
  - exceptions 82
    - Java-specific 82
  - granularity 70
  - optimal granularity 72
  - overloaded methods 77
  - parameter types 72
  - return value types 72
- **service-oriented architecture** 4–5
- **Simple API for XML Parsing (SAX)** 164
- **Simple Object Access Protocol (SOAP)** 3, 33
- **SOAP**
  - anyType 110
  - body 33, 35
  - document fragment 109
  - elements 34
  - envelope 33
  - handlers 62, 85–86
    - extracting context information 367–368
    - interoperability 89
  - header 33–34
  - message
    - header
      - context information 367–368
    - parameter mapping 72
  - messages
    - encoding 34
    - exceptions 80
  - messages with attachments 48
- **SOAP messages**
  - body 327
  - document attachments 137
  - envelope 327
  - header 327
    - security information 327
  - message-level security 296, 326
    - duration of protection 328
  - XML format 34
- **SOAP with Attachments API for Java (SAAJ)** 48
- **SOAPBuilders Interoperability group** 87
- **Standard Generalized Markup Language (SGML)** 122
- **stateless session beans**
  - exposing as endpoints 71
- **static binding** 104
- **stub classes** 44, 50, 210, 212–213
- **Swing components** 238
- **Synchronized Multimedia Integration Language (SMIL)** 126

synchronous communication 363
delegating requests 92, 357

T

transactions
declarative 69
EJB service endpoint 69
JAX-RPC service endpoint 69
message-driven beans 259
transformation
of data 275
transition façade 282
of requests 90
of response 211
schemas 276
transition façade 282
Transport Layer Security (TLS) 309
types
client mapping of 226

U

UDDI
types of pages 36
XML schema 36
UDI
APIs 36
UML modelling 133
Universal Business Language (UBL) 126, 134
Universal Description, Discovery, and Integration (UDDI) 35
unmarshalling 126, 164
URI reference 32

V

validation
of requests 90
of XML document 90, 163
versioning 286–288
Virtual Private Network (VPN) 297

W

WAR file 117
security 325
Web resources
protecting 322
Web services
application functions 4
benefits 6–8
business collaborations 358
business logic 61, 64
business process workflow 58
capabilities 61
challenges 8–10
client access to 104
client communication 200–201
client invocation 59
client types 199
choosing 204
consolidating interactions
centralized manager 373
database transactions 58
definition 3
deploying a service 113–119
deployment descriptor 113
information 113
design
client interaction 57
delegating request 92
delegating requests to business logic 63
endpoints 62, 67, 352
exception handling 80
fault tolerant 378
granularity 70–72, 353
handling errors 63
interaction layer 66
interface 66, 351
overloaded methods 77
interoperability 87
parameter types 72
processing 57
processing layer 99–101
public interface 62
publishing interface 62
receiving requests 62, 89–92
request preprocessing 62
request processing 63
responses 63, 98–99

    - return value types 72
    - SOAP handlers 62
  - design steps 61–66
  - development challenges
    - business partner integration 14
    - ensuring availability 10
    - ensuring reliability 10
    - ensuring scalability 10
    - maintaining security 9
  - document exchange using JMS 139
  - example scenarios 58
  - exposing application interoperable interface 61
  - handling failures 376–381
  - handling XML documents 105–112
    - See XML
  - implementation in J2EE platform 59
  - implementation layers 64
  - informational 58
  - integration APIs 262
  - integration layer 266
  - interaction layer 64, 155
    - design 66
    - preprocessing 65
  - interface
    - publishing 62
    - stability 354
  - interoperability 4–6, 199–200
  - JAX-RPC runtime 59
  - layered view
    - advantages 65
    - adventure builder application 355
    - merging layers 65, 356
    - scenario 65
  - logic flow 61
  - packaging a service 113–119
  - processing 60
  - processing layer 64, 156
  - programming productivity 7
  - publishing 101–105, 358
  - rationale 4
  - sample application 11–17
  - security 291
    - authentication 308
    - authorization 320–324
    - basic authentication 311
    - certificates 309
    - client programming model 314–316
    - endpoint type 310
    - establishing identity 318
    - hybrid authentication 312
    - identity propagation 316–319
    - interactions 308–326
    - invocation identity 316
    - mutual authentication 311
    - policy 313
    - securing interactions 309
    - SSL channel 308
    - tier mechanisms 310
  - standards 1, 3, 27–29
    - areas covered 28
    - criteria 28
    - SOAP 33
    - XML
  - transport protocol 200
  - when to use 11
  - XML-based applications 122
    - designing 131–163
    - interface 128
- **Web Services Definition Language (WSDL)** 3
- **Web Services Description Language (WSDL)** 36
- **Web Services Interoperability Organization (WS-I)** 19, 40
- **web.xml** 114
- **webservices.xml** 115
- **workflow** 12
  - coordination 9
- **World Wide Web Consortium (W3C)** 3
- **WSDL** 67
  - anyType 355
  - binding 37
  - deserialization 87
  - endpoints 37
  - fault elements 232
  - grammar 37
  - integration 268
  - messaging styles 87
  - overloaded methods 79
  - port types 37
  - ports 37
  - security policy 313
  - serialization 87
  - service 37
  - specifying binding 88
- **WSDL document** 39, 50
  - binding element 39

- elements 39
- message element 39
- port element 39
- portType element 39
- service element 39
- types element 39

WS-I 51, 67
- Basic Profile 19, 40, 67, 86, 355
  - attachments 89
  - integration 271
  - J2EE platform support 51
  - security 308
  - Web service failures 376
- interoperability protocols 86

X

X.509 certificate 300

XDE
- when to use 159

Xerces 187

XML 29–33
- application
  - designing 131–163
  - interface 128
  - programming models 164
    - event-based 164
    - Java class-bound 164
    - SAX 165
    - template-based 164, 172
    - tree-traversal 164
    - XSLT 172
- canonical data model 276
- catalog specifications 182
- data mapping 148
  - document-oriented workflow 149
- data model 143
- data types 30, 125
- data-binding facility 52
- definition 122
- deserialization 126, 164
- document 123
  - anyType 110, 355
  - as attachments 136
  - as parameters 76, 107–110, 353–355
  - assembling 130
  - attachments 106
  - binding 100
  - business logic 130, 154
  - caching 193
  - disassembling 128
  - document-centric processing 149
  - dynamically generated 193
  - entity resolution 178–182
  - flexible mapping design 155
  - for integration 249
  - format 30
  - fragmenting 153, 155
  - generic schema 150
  - handling 105–112, 128–131, 157
    - multiple types 371
  - in adventure builder application 348
  - infoset 106
  - input processing 128
  - Java-MIME mappings 107
  - javax.xml.transform.Source objects 106–107, 355
  - JMS queue 136
  - JMS topic 136
  - logical form 124
  - logical fragments 111
  - manipulation logic 111
  - output processing 130
  - parsing 128, 164, 184
  - passing fragment 154
  - physical form 124
  - processing 183
    - encapsulating 161
    - recommendations 163, 175–176
  - processing logic 111
  - receiving 135
  - sending 135
  - serializing 130
  - SOAP attachment 137
  - SOAP document fragment 109
  - SOAP messages 136
  - String objects 107
  - structure 30, 52
  - transforming 90, 106, 128, 130, 211, 357
  - validity 106, 124, 128, 130, 139–141, 357
    - entity resolver 143
  - XDE 157
- document concepts 124–125
- document-centric processing model 151, 156
- domain 123

    - horizontal 126
    - vertical 126
    - vocabulary 132
- domain-specific objects 153
- DTD 30
- entity catalog 134, 179
- external entity references 134, 179
    - entity resolver 190
    - using proxy cache 190
- flexible mapping 156
- grammar 123
- guidelines for use 194–195
- integration technologies 262
- markup vocabulary 123
- marshalling 165
- namespaces 32, 125
    - URI reference 32
- object-centric processing model 152–153, 156
- parser 41
- parsing 10, 128
    - well-formedness 124
- performance 111
- processing
    - API choice 184–186
    - at system boundary 163
    - batch sequential 176
    - changes to 160
    - combining techniques 176–178
    - entity resolution 125
    - external entities 183
    - parsing 184
    - performance
        - caching 190–193
        - dynamically generated documents 193
        - external entities 189
        - improving 184–195
        - parsing 187
        - style sheets 188
        - validation 188
    - performance impact 183
    - pipes and filters 176
    - receiving documents 186
    - steps of 183
    - stream processing 176
    - transformation 177
- processing layer 156
- processing technologies 122, 156
- schema 32, 123, 126
    - custom 126
    - data mapping 148
    - decentralized mapping 148
    - definition 30
    - derived classes 152
    - design 131, 133, 135
    - entity catalog 179
    - entity resolution 179
    - generic 150
    - legacy adapter 148
    - map to data model 143
    - mapping strategy 143
    - meet-in-the-middle data mapping 145
    - out-to-in data mapping 144
    - publishing 102
- serialization 126, 165
- specification documents 123
- tag structure 29, 123
- transformations 155, 172
- unmarshalling 164
- validation 30
    - at system entry points 163
- vocabulary 126, 132
- XSD 30

XML componentization 156

XML data-binding 164, 169, 175
- content tree 169
- marshalling 169–170
- serialization 170
- unmarshalling 169–170
- when to use 171

XML document editor (XDE) design 157

XML Information Set 127

XML Schema Definition (XSD) 30, 123

XML Schema Definition language (XSD) 125

XML schema language (XSL) 277
- See XSL

XML validation 155

Xpath 127, 168, 172

XSD 125
- data types 125
- namespaces 125
- schema design 131
- validating document 139, 141

XSL 277
- style sheets 279
    - integration example 280–282

XSLT 41, 164, 172, 175
identity transformations 168
style sheets 172
templates 172
transformations 172, 229
generic schema 151
when to use 173, 185