A

Access control, 11
Acquisitions, growth of, 20
Acrobat Reader, 5 App. C
Advisor, 514
Amazon model, 23
Amazon Web Services Toolkit, 483
ANT
  customizing, 465–466
  installation and configuration, 11 App. C
  resources, 4 App. C
  running, 467–468
Apache Axis, 2, 3, 4 App. C, 8, 9–10 App.
  C, 11, 20–21 App. C, 42, 493,
  500–510
Apache Crimon, 75
Apache HTTP Web Server, 25
Apache Tomcat, 377, 379–381, 381
Apache Xerces, 4 App. C, 75
Apache Xindice, 381
APPC adapter, 225
Application layer, 296
Application partitioning, 184
Application Platform Layer, 158, 159, 164,
  444, 445
Application Server Analyzer, 166
Application Servers, 114, 156
Application Service Provider, 24, 52
Application-to-application pattern,
  309–312, 342
Application tooling, 446
Architecture. See also Reference
  architecture
  advantages of, 150
  definition of, 149–150
  infrastructure tools, 165–166
  principles, 161–164
  product architecture, 167
reference architecture (See Reference
  architecture)
  role of, 149
security infrastructure, 166
Web Services Deployment
  Platform, 165
Web Services Development
  Platform, 165
Web Services Open Standards, 166
ASP, 24
ASP.NET server, 527
Asynchronous mainframe web services
  pattern, 266–268
Asynchronous messaging, 229
Asynchronous Web Services, 193
Auditing, 111
Authentication, 7, 111, 365, 376
Authentication design, 269
Authority, 111
Authorization, 7, 365
Automation, growth of, 24
Availability, 1, 160, 366
Availability patterns, 172
Avinon, 515, 519
Axis. See Apache Axis

B

Back-end implementation, 110
Back-end resources management, 282
Banking intermediaries, new,
  23, 28
Banking services, core, 27–28
B2B Commerce, 30
B2B financial exchanges, 29
B2Bi, 4
BEA, 515, 520
Best practices, 47
Bind, 61
bindStudio Collaborative Process
  Modeling, 520
bindStudio Model-to-ebXML
  Generation, 521
bindStudio Model-to-WSDL
  Generation, 521
bindsystems, 515, 520–521
BizTalk server, 528
Bluetooth™, 26
Bottomline Technologies, 29
Bouncy Castle JCE Provider,
  4 App. C
Bouncy cryptography provider, 27
bowstreet, 515, 521–522
BPEL4WS, 8, 517
BPWS4J, 524
Broadband Internet, growth of, 26
Broker Integration Patterns, 335–341
build.properties file, 466
build.xml file, writing the, 221, 356
Business case, establishing
  advantages, 47
  application of, 32
  benefits of implementing Web
    Services, 43
  candidate characteristics, 33
  comparison of implementation
    costs, 46
  criteria for pilot project, 41–42
  disadvantages, 47
  membership award example,
    33–37
  payment services example, 37–40
  prerequisites, 41
  return on investment model, 43–46
  risk analysis, 42–43
Business messages, 104
Business process automation, 20
Business Process Execution Language for
  Web Services, 8, 292
Business processes, 104
Business requirements, 437
Business service interface, 105
Business Tier, 155, 231, 232, 260–261,
  296, 444
Business-to-business integration, 4, 18–19,
  20, 21, 53, 288
Business Web Factory, 5, 521
Buying tools versus building
  tools, 409

C
C#, 5, 15, 113, 123, 527
C# Station, 515
Cache, 113
Candidate characteristics, 33
Cape Clear 4 Manager, 522
Cape Clear 4 Server, 522
Cape Clear 4 Studio, 522
Cape Clear, 515, 522–523
Capital markets, growth and development
  of, 29–30
Case studies
  Dollar Rent-A-Car Case Study, 32
  Foreign Exchange (FX) Spot Rate Quote
    Service Case Study (See Foreign
    Exchange (FX) Spot Rate Quote
    Service Case Study)
  Hewitt Associates Case Study, 32
  Trans-Canada Pipeline Case Study, 31
CCI, 300
CDLC. See Connected, Limited Device
  Configuration
CE/FACT, 119
Certificate management, 166
Certification Path (CertPath), 406
CICS, 227
CICS-proprietary EXCI/MRO
  protocol, 230
CICS APPC calls, 227
CICS EJB server, 265
CICS Resource Adapter, 305
CICS screens, 225
CICS SOAP technology, 499
CICS Transaction Gateway (CTG)
  3270 Web Bridge, 250–253
  CICS Web Server Plug-in, 249–250
  client daemon, 230
  components, 229–230
  configuration tool, 230
  CTG Java Class Library, 230
  deployment configurations, 230, 246
  direct connection, 247–249
  distributed platform configurations,
    231–232
  gateway daemon, 229
  interoperability and integration, and
    mainframe, 260
  remote CTG, 245–246
  same platform configuration, 230–231
sample CTG ECI call, 233–239
sample CTG EPI call, 239–245
terminal servlet, 230
Web Support (CWS), 247–253
CICS Web Server Plug-in, 249–250
CICS Web support, 260
CIF, 278
CLASSPATH set-up, 15–16 App. C
Client daemon, 230
Client proxy for FX Web Services Call, sample, 302–304
Client requester, 472–473
Client responsible trust domain, 375
Client-server interaction, 99
Client-side security, 111
Client Tier, 156, 160, 161, 296, 444
Closed Process Integration Pattern, 329–332, 342, 343
Cobol, 6
COBOL-to-Java migration, 281
Code analysis, 459–462
Code conversion methodology, 281–282
Code customization and deployment, 464–480
Collaboration Protocol Agreement (CPAs), 105
Collaboration Protocol Profiles (CPPs), 105
COMMAREA, 225, 232
Commercial implementation of resource adapter, 304
Common client interface, 298, 300–304
Compiling
client requester, 472–473
message provider, 470–471
source codes, 221–222, 356–357
stub/skeleton into Java classes, 507–508
Confidentiality, 7, 111, 366
Configuration file for jSAML demo, 464
Config.xml Contents, 140
Confirmation of receipt of message using ebXML with JAXM, 359
Confirming SOAP message sent, 224
Connected, Limited Device Configuration, 27
Connection management, 299
Construction Phase, 153
Consumer-Service Provider Relationship, 10
Coordination of enabling technologies, 122–124
CORBA, 168, 169, 253
Core J2EE Patterns, 232
Core security services, 111
Creating a New Web Service with JWSDP, 137–146
Credit card processing, 28
Cross-enterprise integration framework
application layer, 296
benefits of, 297
data integration, 295
data transport integration, 294
lower layer, 296
middleware integration, 295
process integration, 295
role and application of, 293–294, 296–297
security integration, 294
upper layer, 296
virtual layer, 296
CTG ECI call, 233–239
CTG EPI call, 239–245
CTG Java Class Library, 230
CRM, 279
Customer relationship banking, 28
Customer relationship management, 20, 278–280

D

Data encryption, 367
Data Exchange Pattern, 326–329, 342, 343
Data integration, 295
Data integrity, 7, 111, 366
Data migration, 281
Data privacy, 7, 366
Data sources, integration of different, 308
Data transformation, 291
Data transport integration, 294
Data transport security, 368, 372
Database Marketing, 28
Databases, 157
DB2 (UDB) with XML Extender, 523
DCOM, 168, 169
DEBA, 208
DefaultContent.htm to Redirect to New Service URL, 425
Defining the service description, 123
Delegate Pattern, 260
Demo program files, installation of, 465
Denial of service, 366, 382
Deploying codes, 479–480
tools, 116
Web Services, 142–144, 145
Deployment architecture, 200, 454
Deployment descriptor, 196
Design patterns
Application-to-application pattern, 309
Asynchronous Mainframe Web Services Pattern, 266
availability patterns, 172
considerations and guidelines, 207–208
Closed process integration pattern, 329
Data exchange pattern, 326
Federated replication pattern, 319
High Availability of Service Registry, 196–198
HTTP Load Balancer (SOAP Server Farm), 185–187
Hub-spoke replication pattern, 316
JMS Bridge, 178–182
manageability patterns, 172
Messaging transport pattern, 416
Multiple Servlet Engines, 182–185
Multi-step application integration pattern, 322
Open process integration pattern, 332
Publish, Unpublish, and Discover Web Services, 200–202
Registry Content Management, 204–207
Registry pattern (service versioning pattern) alias “service versioning”, 173
reliability patterns, 172
Reverse auction-broker integration pattern, 338
scalability patterns, 172
security patterns, 172
Service consolidation-broker integration pattern, 335
Single signon pattern, 410
SOAP cache, 172–178
SOAP Logger, 192–196
Standard build pattern, 312
State Management, 188–192
Synchronous Mainframe Web Services Pattern, 263
UDDI Deployment, 198–200
Version Management of Deployment and Service Registry, 202–204
Developer resources directory, 108
Developer tool log files, 420
Development life cycle, 153–154, 282
Development platform management, 282
Development platforms, 514
Development technologies, 119–121
Development tool model, 64
Digital certificates, 366
Digital signature client WSDD File, sample, 388
Digital signatures, 7, 366, 367, 407–409
Direct connection, 247–249
Directory/policy server, 157
Directory server, 166
Directory structure, 13–14 App. C
DISA, 119, 120
Discover/find, 60
Disruptive technologies, 495–497
Distributed platform configurations, 231–232
DMSConnect, 526
Document-based Web Services, 193
Document Object Model, 75, 81–83
Document type definitions (DTDs), 70–71
Dollar Rent-A-Car Case Study, 32
DOM, 81–83, 176
DOM/SAX versus JAXP, use of, 447
Domain model, 401
Dot-com, demise of, 24, 29
Download centers for software tools, 11
DSIG, 111
DTD, XML to, 133–136

E

E-Marketplace UDDI, 97
EAI adapter for mainframe, 225
EAI and Web Services, 7
EAI patterns, 315–326
EAI products, 6
eBay model, 23
EBCDIC conversion, 269
ebMS, 171
ebPML, 515
ebxml
adoption of, 482
application, 103–104
architecture, 104–105, 170–171
business messages, 104
business processes, 51, 104, 306
business service interface, 105
Collaboration Protocol Agreement (CPAs), 105
Collaboration Protocol Profiles (CPPs), 105
confirmation of receipt of message using ebxml with JAXM, 359
demonstration, 106–107
development technology, 119
ebxml Message Service, 105
ebxml Registry Information Model, 105
ebxml Registry Service Specification, 105
evelope and structure, 105
implications, 107
Message Provider Sent Log, 360
message provider using JAXM to send XML message, demonstration of, 345–360
Message Service, 51, 105
Message Specification, 55
ebMS, 171
registry, 107, 113
Registry Information Model, 105
Registry Service Specification, 105
Service Registry, 55, 108, 109, 110
service registry/repository, 105, 108–111
SOAP, and, 120–121
trading partner agreement, 105
transport and routing layer, 105
use of, 54
user interface to send message using ebxml with JAXM, 359
 ebxml registry, 122
 ebxml service registry, 122
Web Services, and, 103–108
ECI, 230
Eclipse, 25, 514
Economic landscape, transformation of the, 19
Edge products, 167
Edge server, 113
EDI, 127
Elaboration phase, 153
Electric Server Page, 528
Electric XML 4.0, 528
Electronic banking, 28
End-to-end security framework, 7
Enterprise Application Integration (EAI)
advantages, 288
disadvantages, 288
risk factors, 288
use of, 288
Web Services, and, 290–293
Enterprise Distributed Object Computing (EDOC), 292
Enterprise Java Bean (EJB), 261
CORBA server, 253
design features, 253
EJB container, 253
Jar files, deployed, 253
Java Connector for CICS, 254–257
Java Virtual Machine, 253
object store (DFHEJOS), 253
request models, 253
request processor, 253
request receiver (DFHIRR), 253
request stream directory (DFHEJDIR), 253
support, 253–254
TCP/IP Listener, 253
Enterprise Resource Planning (ERP), 288
Entitlement, 7, 111, 365
Environment set-up for directories, 464–465
Environment variables setting, 14–15
App. C
Eontec, 28
EPI, 230
Escrow-based payment services, growth of, 29
ESI, 230
Exchanges gateway, 157
Exercises
Creating a New Web Service with JWSDP, 137–146
Writing the First JAXM Program, 136–137
XML to DTD, 133–136
eXtend Composer, 529
eXtensible Markup Language. See XML; XML (eXtensible Markup Language)

F

Factory Integrations, 522
Factory Options, 522
Federated Replication Pattern, 319–322, 342, 343
Financial intermediaries, emergence of, 20
Fine-grained data access rights, 7
Flamenco Networks, 518
Flexibility, 161
Foreign Exchange (FX) Spot Rate Quote Service Case Study
ANT scripts, customizing, 465–466
ANT scripts, running, 467–468
Application Platform Layer, 444, 445
application tooling, 446
assumptions, 436
build.properties file, 466
business requirements, 437
Business Tier, 444
code analysis, 459–462
code customization and deployment, 464–480
compile FX quote provider, 467–468
compile the client requester, 472–473
compile the message provider, 470–471
configuration file for jSAML demo, 464
demo program files, installation of, 465
deploying codes, 479–480
deployment architecture, future, 454
design pattern applied, 447
DOM/SAX versus JAXP, use of, 447
environment set-up for directories, 464–465
front page to accessing remote FX quote web service, 458
Hardware Platform Layer, 444, 445
high-level design, 441–443
install the client requester, 473–474
install the message provider, 471–472
integration of components, 463–464
integration testing, 479
Integration Tier, 444
interaction between components, 442
interoperability with external systems, 447
invoke the client, 474
Java classes used to support SAML, 459
logical architecture, 443, 444
Lower Platform Layer, 444
main menu of the Demo system, 456
modifying the remote FX quote engine, 475–476
objective, 435
physical architecture, 453
platform layers, 444
Presentation Tier, 444
private label customization, 447
program changes, management of, 479
quality of service matrix, 444, 445
quote response, 459
quote service design, 450–452
quote web service page, 458
request for quote, 437
request for quote use case, 440–441
Resource Tier, 444
SAML assertion request to remote FX quote Web Service, 457
Secure Message Service, 462–463
security design considerations, 446
service-level requirements, 437
set up FX quote service, 469–470
Single Sign-on design, 336, 437, 439, 448–449, 455, 457
system security, 437
technology applied, 447
unit testing, 479
Upper Platform Layer, 444, 445
use case scenario, 436
use cases, 438–440
user experience, 455–459
verify the deployed services, 474
verifying set-up and remote services, 476–478
Virtual Platform Layer, 444, 445
Web Services relationship, 442
XKMS Trust Verifiers, 463
XML messaging standards, 446
Forte Transaction Adapter for a remote CICS transaction, 258–260
Front Controller Pattern, 260
FXProviderIF.java Program, 138
FXProviderImpl.java Program, 138–139

G
GAIA, 528
Gateway daemon, 229
GLUE 3.2/4.0, 528
GoXML Integration Workbench, 534
GoXML Messaging, 534
GoXML Registry, 534
GoXML Repository, 534
GoXML Transform, 534
Green pages, 95, 96
Grid computing, growth and development of, 25, 497
GSM technology, 26

H
Hacker attacks, 381–382
Hardware encryption, 166
Hardware level, 197
Hardware Platform Layer, 158, 159, 444, 445
Hardware requirements, 3 App. C
HD Courier, 526
Hewitt Associates Case Study, 32
High Availability of Service Registry, 196–198
High-level design, 441–443
High-speed computing, growth and development of, 25
Host gateway, 157
Hosts, 377
HTTP binding, 98
HTTP Load Balancer (SOAP Server Farm), 185–187
HTTPS, 7, 110
Hub-Spoke Replication Pattern, 316–319, 342

I
IBM developerWorks LIVE! Conference, 517
IBM products, 523–524
IBM Web Services, 121, 514, 515
IBM Web Services Architecture, 64–66, 68
IBM's XML Security Suite, 407
IBM z/OS mainframe, 226–227
IDE/Development Environment, 114
Identity, 7, 63
IDL. See Interface Definition Language
abilities, 58
iMode, 26
Inception phase, 153
Information model, 96, 109
InJoin's BATCH, 275
InJoin's TRANS, 275
Installation of
client requester, 473–474
message provider, 471–472
Integration. See Interoperability and integration, mainframe
Integration design patterns
application-to-application pattern, 309–312, 342
Broker Integration Patterns, 335–341
Closed Process Integration Pattern, 329–332, 342, 343
Data Exchange Pattern, 326–329, 342, 343
EAI patterns, 315–326
Federated Replication Pattern, 319–322, 342, 343
Hub-Spoke Replication Pattern, 316–319, 342
Multi-Step Application Integration Pattern, 322–325, 342, 343
Open Process Integration Pattern, 332–335, 342, 343
Process Integration Patterns, 329–335
Reverse Auction-Broker Integration Pattern, 338–341
Service Consolidation-Broker Integration Pattern, 335–338, 342, 343
standard build pattern, 312–315, 342
Integration of components, 463–464
Integration points, 261–262
Integration technology for Legacy mainframe, 229
Integration testing, 479
Integration Tier, 155, 160, 161, 231, 260–261, 296, 444
Interbind (IBX), 527, 528
Interface Definition Language, 291
Interfaces, synchronizing all, 280
Intermediaries, new, 23–24
Internal EAI UDDI, 97
Internal service directory, 108
Internet, role of the, 20
Internet layer, 57
Internet or network layer security, 368
Interoperability and integration, mainframe
alternative methods, 270–284 (See also specific methods)
arboricultural perspectives, 260–262
architecture implications, 282
asynchronous mainframe web services pattern, 266–268
asynchronous messaging, 229
authentication design, 269
business tier, 260–261
CICS Transaction Gateway (CTG) (See CICS Transaction Gateway (CTG))
CICS Web Support, and, 260
design considerations, 283
development life cycle, integration with, 282
EBCDIC conversion, 269
external systems, and, 226–227
functional perspectives, 262–263
IBM z/OS mainframe, 226–227
integration points, 261–262
integration tier, 260–261
Java technology, and, 253–257, 261 (See also Enterprise Java Bean)
Legacy systems, 3, 6–7, 21, 229
Linux on mainframe, 228–229
resource tier, 260–261
risk factors, 283
security considerations, 269
SOAP Proxy on mainframe, 257–260, 261
synchronous mainframe web services pattern, 263–266
synchronous messaging, 229
technology update, current, 226–227
Invoking a public Web Service with JAVA™ Client, 47–48
with SOAP-Lite Client, 48–49
IONA, 515, 524–525
IP port, 80, 362
IPSec, 369, 370
J
jar files, deployed, 253
Java™, 18, 76, 123
Java 2 Platform Micro Edition (J2ME™), 27
Java 2 Standard Edition (J2SE)
installation and configuration, 6 App. C
verification of set-up, 16–17 App. C
Java Authentication and Authorization Service (JAAS), 405, 409
Java classes
on Legacy system, 273
used to support SAML, 459
Java Connector Architecture (JCA)
CICS, for, 254–257
CICS Resource Adapter, 305
client proxy for FX Web Services Call, sample, 302–304
commercial implementation of resource adapter, 304
common client interface, 298, 300–304
connection management, 299
program except, ČCI, 301–302
resource adapter architecture, 298–299
role and application of, 297–298
SAP R/3 Resource Adapter, 305
transaction management, 299
Java Cryptographic Architecture (JCA), 406
Java Cryptographic Extension (JCE), 406, 409
Java for Document Object Model, 75
Java GSS, 405
Java Message Service, 57, 178, 295, 307
Java Secure Socket Extension (JSSE), 405, 409
Java Specification Requests (JSRs), 488–490
Java™ Technology and Web Services, 514
Java Virtual Machine (JVM), 6, 253, 408
Java Web Services Developer Pack™ (JWSDP), 2, 3, 4 App. C, 7–9 App. C,
8, 11, 18–19 App. C, 117, 118, 277, 482
config.xml Contents, 140
creating new Web Service,
142–146
demonstration of creating a new
web service, 137–146
deploying Web Services,
142–144, 145
FXProviderIF.java Program, 138
FXProviderImpl.java Program, 138–139
jaxrpc-ri.xml Contents, 141
publishing to registry, 145
verification of Web Services
deployment, 144
Web Services Description Language,
138–142
web.xml Contents, 140
wscompile in build.xml, use of, 142
WSDL generated from JWSDP, 145
Java™ XML Pack, 76
JAX, 110, 113, 277
JAX Packs, 76–77, 118
JAX-RPC, 77, 93
JAXB, 77, 89–91, 118
JAXM
application, 83
architecture, 83
confirmation of receipt of message using
ebXML with JAXM, 359
demonstration, 84–89
ebXML message provider using JAXM to
send XML message, demonstration of, 345–360
message for the service request,
426–429
Program, writing the, 136–137
Secure Message Provider Class,
43–432
SOAP local message provider with JAXM
to send XML message, using,
209–224
user interface to send message using
ebXML with JAXM, 359
JAXP, 76, 77, 118
JAXR, 77, 92, 109, 118, 201
jaxrpc-ri.xml Contents, 141
JBroker, 531
JDBC, 281
JDOM, 75, 176
J2EE Application Server, 108, 121,
152–153
J2EE Application Server Platform, 167
J2EE Architecture, 276
J2EE RMI/IIOP, 158
J2ME™, 27
J2ME™ Wireless Toolkit, 486
JMS, 57
JMS binding, 120
JMS Bridge, 113, 178–182, 307
JMS-enabled middleware, 113
JMS Messaging Bridge, 520
JNDI, 207
JNDI Look-up vs. UDDI, 207
JAXM Toolkit, 5 App. C, 11
demonstration of, 422–425
installation and configuration,
12–13 App. C
verification of set-up, 22–25 App. C
J2SE SDK, 4 App. C
JUnit, 4 App. C
JXTA, 27, 496
K
Kerberos ticket, 362, 406
Key management, 367, 375–376
Key registration, 7
Key store, 436
Kildara, 515, 525–526
Killer application, Web Services as, 4
kSOAP, 27
kSOAP jar files, 27
kSOAP reference implementation, 486
KVM, 27
L
LDAP, 376
LDAP/390, 226
Legacy application migration, 282
Legacy systems, 6–7
LegacyJ, 272–273
LegacyJ PerCOBOL, 273
Liberty, 7. See Project Liberty
Liberty Alliance, 7, 121. See Project Liberty
Liberty-compliant Identity Server, 367
Linux, 25
Linux 390/zOS, 225
Linux on mainframe, 228–229
Locate a key demonstration, 391–392
Locate client example, 395
Log Files, 420
Logger design, 195
Logical architecture, 443, 444
Logistics intermediaries, new, 24
Lotus Domino, 524
Lower layer, 296
Lower Platform Layer, 158, 159, 163, 444

M
Mainframe Batch Manager, 275
Mainframe interoperability. See Interoperability and integration, Mainframe
Mainframe Transaction Processing Architecture, 274
Maintainability, 43
Man-in-the-middle attack, 382
Manageability, 160
Manageability patterns, 172
Managed trust domain, 375
Manifest, 105
Market Data Server, 157
Megamergers, 20
Membership award example, 33–37
Mergers and consolidations, 28, 29
Message-Oriented Middleware, 7, 291
Message provider, 470–471
Message security, 111, 372
Message to request password authentication, 402
Messaging, 291
Messaging servers, 156
Messaging Transport Pattern, 416–419, 426–432
Methodology and development life cycle, 153–154
Microsoft Global XML Web Services Architecture, 66–67, 68
Microsoft PASSPORT, 7, 8, 366, 406
Microsoft products, 527–528
Microsoft SOAP Toolkit, 528
Microsoft Web Services, 514
Microsoft Windows, 113
Middleware, 113, 115, 166, 290, 295
MIDP™, 26, 27, 486
MIME, 120
MIME MIME structure, 105
mindElectric, 515, 528
Mobile Information Device Profile, 26
MOM. See Message-Oriented Middleware
MSXML, 67, 75
Multi-Step Application Integration Pattern, 322–325, 342, 343
Multiphase customer relationship management, 278–280
Multiple Servlet Engines, 182–185
Multisourcing, role of, 24
MVS, 226
MyRemote Servlets, deployment of, 358
MyRemote source codes, compiling, 356–357
MySimple Servlets, deployment of, 221–221
MySimple Source Codes, compiling, 221–222

N
NAICS (North American Industry Classification System), 95, 109
.NET, 15, 116, 121
Netbeans, 25
NetScenario Business Server, 519
NetScenario Business Service Templates, 519
NetScenario Studio, 519
Network connectivity security, 368
Network/data transport security, 111
Network identity management, 363
Network level, 198
Next-generation SOAP engine, 498–499
NMAP, 377
Nonrepudiation, 7, 111, 366
Novell, 515, 531

O
OASIS, 26, 108, 119, 120, 406, 486, 491–493, 515, 516
Object store (DFHEJOS), 253
Open Applications Group, 482
Open Process Integration Pattern, 332–335, 342, 343
Open Source Web Services Software, 493–495
Open Standards, 25–26, 54, 58, 121, 411
Open Travel Alliance, 482
Operating system security, 368
Orbix2A XMLBus, 525
OS/390 Unix Services, 226
OSF Apache, 120
Out-sourcing, 52
Out-tasking, 52

P

Partner catalog UDDI, 97
PASSPORT, 7, 8, 366, 406
Password authenticated, 403
Payment services example, 37–40
PDA, 26
PDP, 401
Peer-to-peer computing, 27
Penetration testing, 368
PEP, 401
Performance, 160
Performance tuning, 166, 207
Perl, 5, 6 App. C
Personal Digital Assistants, use of, 26
Physical architecture, 453
Pilot, identification and selection of a, 130
Pilot project, criteria for, 41–42
Pitfalls, 47
PKI, 384
Platform, 63
Platform security, 368
Policy server, 166
Port 8080, 362
Portal Servers, 156
Portal UDDI, 97
Portals, Web Services
Presentation Tier, 156, 160, 161, 231, 232, 296, 444
Private label customization, 447
Private labeling, 288
Private service registry, 108, 156
Private UDDI service registry, 30
Process Integration, 329–335
Process integration, 295
Process management and methodology, 115
Product architecture, 167
Product ROI, 43, 45
Production Phase, 153
Production platform management, 282
Program changes, management of, 479
Program except, CCI, 301–302
Programming model, 64
Progress eXcelon, 515
Project Liberty, 7, 8, 363, 404–406
Proprietary interfaces, 289
Publish, 61, 124, 145
Publish, Unpublish, and Discover Web Services, 200–202

Q

QSAM, 281
Quality of Service, 1, 21, 58, 160–161, 444, 445
ilities, 58
Query, 60

R

RACF, 269
RDBMS, 369
Re-usability, 43, 161
Real-time information, growth of, 24
ReceivingServlet.java, 218–221, 354–356
Recompile, 272–273, 278
Reference architecture
Application Servers, 156
characteristics, 150
components, 155, 156
databases, 157
demonstration, 154–158
directory/policy server, 157
exchanges gateway, 157
framework, 151–153
host gateway, 157
Market Data Server, 157
Messaging Servers, 156
methodology and development life cycle, 153–154
Portal Servers, 156
Private Service Registry, 156
Quality of Service Analysis Matrix, 160–161
SOAP Server, 156
tiers versus platform layers architecture analysis, 158–159
Web Servers, 156
Refront, 275–277
Registration and authentication, 7
Registry Content Management, 204–207
Registry properties, sample, 201
Regression testing, 165
Rehost, 273–275, 278
Relativity’s RescueWare, 270–272
Release control, 165
Reliability, 160, 172
Request models, 253
Request processor, 253
Request receiver (DFHIRRS), 253
Request stream directory (DFHEJDIR), 253
Research and development costs, reduction of
Resource adapter architecture, 298–299
Resource Tier, 156, 160, 161, 231, 260–261, 296, 444
Retirement Phase, 153
Retrieval of cached data, 176–177
Return on Investment, 44
Return on investment model, 43–46
Reverse Auction-Broker Integration Pattern, 335–338
S
S/MIME, 119
Same platform configuration, 230–231
SAML, 7, 111, 112, 113, 121, 166, 376, 401–404, 406
SAML-based Single Sign-on environment, adding a new service provider in, 421–425
SAML Message to Request Password Authorization, sample, 402
SAML response-Password Authenticated, 403
SAP R/3 Resource Adapter, 305
SAX, 75, 78–81, 176
Scalability, 1, 160, 172
Screen scraping, 41
SDK, 113
Secure Message Service, 462–463
Secure Messaging Provider Class, 418
Security Assertion Markup Language (SAML), 410
Security design considerations, 446
Security file, sample, 384–386
Security infrastructure, 166
Security integration, 294
Security patterns, 172
SendingServlet.java, writing the, 210–218, 346–356
Sent log, verification of, 60
Server components, 99
Server Configuration File, 420
Server-side signing model, 408
Service broker, 59, 60
Service Consolidation-Broker Integration Pattern, 335–338, 342, 343
Service container, 63
Service creation and assembly, 63
Service definition, 291
Service delivery, 63
Service description, defining the, 123
Service Description Language, 51, 57
Service dimension, 58
Service discovery, 58, 369
Service integration, 63
Service-level requirements, 437
Service negotiation, 51, 58, 369–371
Service provider, 59, 60, 95
Service registries, 59, 60, 61, 94, 108–111, 115
Service registry/repository, 105, 108–111
Service requester, 59, 60, 94
Service Requester-Service Provider Relationship, 9–10
Service versioning, 173
Servlet engines, configuring, 184
Shallow model, 65
Index

Signing server, 407–409
Silverstream, 515, 529
Simple Object Access Protocol (SOAP) adaptations, 98
architecture, 167–168
CICS, on, 499
client proxy, 152
client-server interaction, 99
constraints, 168
debugger, 165
development technology, 119
ebXML, and, 120–121
enabling technology of, 4
HTTP, over, 51, 294
message security, 111
messaging capabilities, 2, 224
monitor, 498
protocol, 158
proxy from the Service Registry, 10
proxy on mainframe, 257–260, 261
resources, 516
server, 152, 156
server components, 99
server infrastructure tools, 2
TCP/IP, over, 294
technological history, 56
testing/performance testing, 165
SML Parsers, 75
SNA LU 6.2 verbs, 225, 231
SOAP, 1, 98–100. See also Web Services and SOAP
SOAP Cache
benefits of, 177
cost, 172–173
dexample, 178
force, 173
implementation, 177
problem, 173
resources, 518
retrieval of cached data, 176–177
risk factors, 177
sequence diagram, 176
solution, 174–176
transaction cache creation, 176
use cases, 175
SOAP Client and state management, 191
SOAP Client proxy, 123
SOAP-FTP, 294
SOAP/HTTP binding, 110
SOAP-JMS binding, 2, 295
SOAP-JMS Integration, 307
SOAP-Lite, 5, 6 App. C
client code, sample, 48
download links, 11
installation and configuration, 13 App. C
verification of set-up, 25 App. C
SOAP local message provider with JAXM, using, 209–224
SOAP Logger, 192–196
SOAP Messaging over JMS, 263
SOAP Messaging over MQ, 263
SOAP Proxy, 123
SOAP-RPC, 168
SOAP over CICS, 499
SOAP RPC layer, 98
SOAP-SEC, 8, 113, 166, 362
SOAP Server Farm. See HTTP Load Balancer (SOAP Server Farm)
SOAP-SMTP, 294
SOAP/UDDI, 120
SOAP/WRC, 517
SOAP4J, 524
Soft ROI, 43, 44–45
Software level, 198
Software requirements, 3–6 App. C
Software tools, 11
Solaris OE, 158
Solutions Sampler, 125–127
Source codes, 221–222, 356–357
Sphere™ source codes, 25
SQL mapping, 116
SQL Server, 528
Stack complexity scale, 58
Standard build pattern, 312–315, 342
State Management, 188–192
Stickiness, 4
Straight-Through Processing, 24, 29, 307
Stress/load testing, 165
Structural framework, 57–58
Stub/skeleton into Java classes, compile, 507–508
Sun Microsystems Java™ Technology and Web Services, 514
Sun ONE™, 121, 151

Index
Sun ONE™ Application Server, 276, 530
Sun ONE™ Architecture Framework, 63, 64, 68
Sun ONE™ Directory Server, 26
Sun ONE™ Directory Server 6.0, 531
Sun ONE™ Identity Server 6.0, 530
Sun ONE™ Integration Manager, 121
Sun ONE™ Integration Server, 530
Sun ONE™ Message Queue 3.0, 530
Sun ONE™ Smart Web Services, 121
Sun ONE™ Studio, 5 App. C, 121, 276, 488, 514, 530
Sun ONE™ Unified Development Server 5.0, 532
Sun ONE™ XML Adapter 1.0, 530
Sunk cost, 43
SunNetwork Conference, 517
Sun’s Mainframe Batch Manager (MBM), 275
Sun’s Mainframe Transaction Processing Software, 274–275
Sun’s N1, 495–496
N1, 495
SunTone, 155
SunTone Architecture Methodology, 155
Synchronization of all interfaces, 280
Synchronous mainframe web services pattern, 263–266
Synchronous messaging, 229
Synchronous Web Services, 189
Systems Management Tools, 166
Systinet, 515, 531–532

Throughput, 160
Tiered models, 390
Tiers versus platform layers architecture analysis, 158–159
Titan, 377, 407
tModel, 96, 109
tModellInstance objects, 96
Tool selection criteria, 118–119
Total cost of ownership, 20, 46
Traceability, 7, 365–366
Trading partner agreement, 105
Trans-Canada Pipeline Case Study, 31
Transaction cache creation, 176
Transaction integrity, 111
Transaction management, 299
Transaction routing, 51, 57, 369
Transaction security, 111, 376–377
Transcode, 270–272, 277
Transition Phase, 153
Transport and routing layer, 51, 57, 105, 368–369
Troubleshooting approach, Web Services, 26 App. C, 207
Trust, 373–374
Trust domains, 375–377
Tutorials, 514

U

UDDI, 1
UDDI4J, 524
UDDI registry, 118
UDDI service registry, 118
UN/CEFACT, 120
UN/SPSC (United Nation Standard Products and Services Classification), 95
Unified Messaging, 26
Unified Process-based methodology, 153
Uniform model, 65
Unit testing, 165, 479
Universal Discovery, Discovery, and Integration (UDDI) access design, 169
application, 94
architecture, 169–170
browser, 111
development, 198–200
differentiator, 170
e-Marketplace UDDI, 97
enabling technology of, 4
entities and relationships, 96–97
explorer, 524
Host Scan, 377–378
Host Security Healthcheck, 379–381
implementation, 98
implementation challenges, 94
information model, 96, 169
information types, 95–96
internal EAI UDDI, 97
operator node, 97
Partner catalog UDDI, 97
resources, 516
roles and operations, 94–95
service provider, 95
service requester, 94
technological history, 56
test bed UDDI, 97
types, 97
Web Services, 94–98
Universal Resource Identifier, 10
Upper Platform Layer, 158, 159, 163–164, 296, 444, 445
URL configuration file, sample entry in, 424
User interface to send message using ebXML with JAXM, 359

V

Vendor products, 113–116, 116, 515, 518–534
Verification of Web Service deployment, 144, 508–509
Verifying set-up and remote services, 476–478
Verisign’s Trust Services Integration Kit (XKMS), 395, 407, 436, 462
Version management, 165, 202–204
Virtual Platform Layer, 158, 159, 164, 296, 444
Virtual Private Network, 26, 111
Virtual Sequential Access Method, 227
Visual.NET studio, 527
Vitrius Web Services Engine, 525
VPN. See Virtual Private Network
VSAM, 6, 227

W

WAP, 26
War files, deployment as, 221–222, 357–358
WASP Developer, 532
WASP Server, 532
WASP UDDI, 532
W3C, 120, 490–491, 516
Wealth management, 29
Web Container, 420
Web Container/Application Server, 381
Web Container Log files, 420
Web servers, 156
Web Services
advantages of, 127, 290
application servers, 114
application tools, 115
definition of, 52–53
definition of, 52–53
demonstration, 128–130, 500–508
development platforms, 514
development technologies, 119–121
benefits of, 69
bind, 61
coordinated of enabling technologies, 122–124
defining the service description, 123
definition of, 52–53
definition of, 52–53
demonstration, 128–130, 500–508
development platforms, 514
development technologies, 119–121
disadvantages, 290
discover/find, 60
DOM, and, 81–83
EAI, and, 7
ebXML, and, 103–108
Enterprise Application Integration, and, 290–293
evolution of, 2
features of, 53–54
frontiers of, 8
growth and development of, 1–2, 18, 22–23, 46, 53
IBM Web Services Architecture, 64–66, 68
IDE/Development Environment, 114
implementation, 1–2, 123–124, 130–131
Internet layer, 57
invoke Web Services, 124
Java™, and (See Java™)
JAX-RPC, and, 93
JAXB, and, 89–91
JAXM, and, 83–89
JAXP, and, 77
JAXR, and, 91
limitations of, 127–128
management, 58, 114, 116
Microsoft Global XML Web Services
Architecture, 66–67, 68
middleware, 115
.NET to Java porting, 116
news and product updates, 514
Open Standards, 58
pilot, identification and selection
of a, 130
portals, 513
process management and methodology,
115
products, 113–116
publish, 61, 124
Quality of Service, 58
query, 60
risk factors, 290
role and application of, 124–125
SAX, and, 78–81
scenario, sample, 60
security, 7–8, 58, 111–112
sequence diagram, 62
service broker, 59, 60
Service Description Language, 57
service dimension, 58
service discovery, 58
service negotiation, 58
service provider, 59, 60
service registries, 59, 60, 61, 115
service requester, 59, 60
SOAP, and, 98–100
Solutions Sampler, 125–127
stack complexity scale, 58
standards bodies and communities,
515–516
structural framework, 57–58
success factors, 131
technical articles, 514
technological history, 54–56
technology stack, 57–59
tool selection criteria, 118–119
transaction routing, 57
Transport layer, 57
tutorials, 514
UDDI, and, 94–98
use case, 61–63
Web Services Appliances, 418
Web Services Choreography Interface
(WCSI), 8, 291–292
Web Services Conference, 517
Web Services Consumer, 441
Web Services Deployment Descriptor
(WSDD), 506
Web Services Deployment
Platform, 165
Web Services Description Language
(WSDL), 9–10, 51, 54, 69–75,
100–103, 111, 123–124, 138–142,
291, 369
Web Services design patterns, 148
Web Services DevCon, 517
Web Services Developer Pack 1.1., 530
Web Services Development
Platform, 165
Web Services Edge, 517
Web Services Enhancements (WSE) 1.0,
527
Web Services Gateway, 498, 524
Web Services Invocation
Framework, 524
Web Services Management, 497–498
Web Services Management/Network
Services, 165
Web Services Open Standards, 166
Web Services portals, 513
Web Services Security
Apache Tomcat/UDDI Host Security
Healthcheck, 379–381
authentication, 376
buying tools versus building
tools, 409
client responsible trust domain, 375
denial of service, 382
digital signatures, 407–409
framework, 371–373
hacker attacks, 381–382
hosts, 377
Internet or network layer security, 368
key management, 375–376
man-in-the-middle attack, 382
managed trust domain, 375
Messaging Transport Pattern, 416–419, 426–432
platform security, 368
resources, 516
Server-side signing model, 408
service discovery, 369
service negotiation, 369–371
signing server, 407–409
Single Sign-on Pattern, 410–416, 421–425
threat profiling, 374, 377–382
tools, 407
transaction routing, 369
transaction security, 376–377
transport layer security, 368–369
trust, 373–374
trust domains, 375–377
UDDI Host Scan, 377–378
UDDI Service Registry, 381
Web Container/Application Server, 381
Web Services Security Stack Process, 370
XML encryption and XML signature, 409
Web Services Security Forum, 516, 517
Web Services Security Roadmap, 7
Web Services Security Stack, 367–368, 370
Web Services Service Provider, 441
Web Services Service Registry, 441
Web Services Standards, convergence of, 484–486
Web Services Toolkit, 8, 523
Web Support (CWS), 247–253
Weblogic Integration, 520
Weblogic Server 7.0, 520
Weblogic Workshop, 520
WebSphere/390, 225
WebSphere Application Server™, 26, 523
WebSphere MQ, 227, 263, 523
WebSphere Studio, 523
WebSphere Studio Application Developer, 523
web.xml Contents, 140
WestBridge, 515
“White labeling,” 20, 24
White pages, 95, 96
Wholesale banking, 28
Windows, 113
Winzip, 5 App. C
Wireless technology, growth of, 26
Wireless web services, 26–27, 486–488
WML services, 26
Writing the First JAXM Program, 136–137
WS-Addressing, 527
WS-I test, 486, 516
WS-license, 362
WS-Policy, 367
WS-Referral, 527
WS-Routing, 527
WS-Secure, 367
WS Security specification, 7, 372
WSCl, 8
Wscompile in build.xml, use of, 142
WSDL, 96, 145, 503–505
WSDI, 4
WSDL-UDDI-SOAP, 5, 55
WSFL, 8
WSTK, 113
X
XACML, 7, 166
Xalan, 4 App. C
XAML, 113
Xerces. See Apache Xerces
Xjc, 89
XJS, 89
X-KISS, 399
XKMS, 7
XLANG, 8
XML Access Control Markup Language, 412
XML Business Integrator, 534
XML-ENC, 372
XML (eXtensible Markup Language) applications, 76
components of an XML document, 69–70
definition and validation of, 69
document type definitions (DTDs),
70–71
DTD, to, 133–136
encryption, 7, 367, 369, 371,
372, 417
encryption and XML
signature, 409
firewall products, 518
limitations, 76
messaging standards, 446
namespace, 71–72
parsing, 75, 77
proliferation of, 18
schema, 72–75, 279
security, 4 App. C
signature, 367, 371, 373,
386–387, 417
style sheet processor, 295
XML Namespace, 71–72
XML Global, 515, 532–533
XML Key Management Service, 462
XML Key Management Specification
(XKMS)
benefits, 400
design features, 390–391
implications, 399–400
locate a key demonstration, 391–392
locate client example, 395
overview, 389–390
result of validating key information from
XKMS client, 397–399
return the locate key result example,
392–393
return the validate key result example,
394–395
running sample XKMS client to locate
key information, 396
running sample XKMS client to validate
key information, 396
tiered models, 390
Trust Verifiers, 463
validate a key example, 393–394
XML Key registration Service
Specifications (X-KRSS), 391
XML message
confirmation of receipt of
message, 359
ebXML message provider using JAXM to
send XML message, demonstration
of, 345–360
SOAP local message provider with
JAXM, using, 209–224
user interface to send message using
ebXML with JAXM, 359
XML Message Server, 498
XML Processing Description Language
(XPDL), 292
XML Query Language (XQL), 295
XML-RPC, 6, 123, 207, 229
XML Security Suite (XSS), 407, 524
XML-to-Java binding schema
(XJS), 89
XML-to-Schema Compiler (xjc), 89
XML Trust Center, 516, 517
XML Trust Service, 7, 406
XML Web Services One, 517
XML Web Services technology, 21–22
XMLBus, 525
XMLSpy, 117, 135–136
XOption XML, 134
Xpath, 176
XSLT, 14
X.509v3, 362, 365, 383, 494
XWS, 30
Y
Yellow pages, 95, 96
Z
Zefer’s Web Services architecture,
151, 152
z/OS, 226–227