

Index

4GL (Fourth-Generation Language), 315

A

Access control (security), 187, 199

Accessing information. *See* information access.

Accounting, 169–170, 180. *See also* systems management.

Accuracy of information. *See* information accuracy.

ACID (Atomicity, Consistency, Isolation, and Durability), 27, 315

Active Data Object (ADO), 26, 315

Active Server Page (ASP), 55, 75, 84, 87, 89–90, 315

Ad hoc database queries, and performance, 159–160

Administration, 169–170, 180. *See also* systems management.

ADO (Active Data Object), 26, 315

ADO.NET, 26

Agent objects, 43

Agile alliance manifesto, 208

American Standard Code for Information Interchange (ASCII), 315

Analysis

business process stage, 234–238

data safety, 138

error-handling, 138

integration design, 247–250

process analysis technique, 227–228

strategic business analysis, 263–264

uptime, 138

API (Application Programming Interface), 315

Application designers, as architects, 15–16

Application Programming Interface (API), 315

Application Service Provider (ASP), 315

Applications

changing. *See* applications, converting to services.

contribution to systems management, 184–185

integrating, 93–94. *See also* applications, converting to services.

Internet. *See* Internet applications.

software failure, 136–137

Applications, converting to services

batch processes, 291–294

business processes, 286–289

converting existing code, 282–284

core server model, 288

example, 278–279

interface size mismatch, 281–282

large message problem, 281–282

middle tier, 285–286

middleware between transaction servers, 289–291

presentation layer, creating, 280–286

reuse model, 288–289

screen scraping, 281

wrapping, 284–286

Applications, designing. *See also* architecture.

business process level, 219, 222

levels of design, 217–220, 222

problems

agile alliance manifesto, 208

architectural reuse, 214–215

business rules, 209–210

classification rule pattern, 209

computation rule pattern, 209

constraints rule pattern, 209

322 Index

- Applications, problems (*cont.*)
 - enumeration rule pattern, 209
 - existing systems, 210–211
 - iterative development, 207
 - list constraints rule pattern, 209
 - MDA (Model Driven Architecture), 207
 - RAD (Rapid Application Development), 207–208
 - reuse, 211–215
 - role of architecture, 216–217
 - rule patterns, 209
 - serendipitous reuse, 211–214
 - silos and monolithic development, 215
 - UML (Unified Modeling Language), 207
 - up front design *versus* as needed design, 206–209
 - waterfall method, 206–207
 - reconciling approaches, 220–221
 - task level, 219, 222
 - transaction level, 219–220, 222
- Architects, 15–16
- Architectural patterns, 231–234
- Architectural reuse, 214–215
- Architecture. *See also* applications, designing; middleware, vendor architectures.
 - choosing an architect, 15–16
 - common data format, 8
 - communication links, 8
 - components, 6–9. *See also* middleware.
 - definition, 5
 - naming and directory services, 8
 - programmatic interface, 8
 - protocols, 8
 - purpose, 5–6
 - scalability, 15
 - security, 8, 200–202
 - server process control, 8
 - SOA (Service-Oriented Architecture), 9
 - systems management, 8
 - traditional configuration
 - evolving, 10–12, 12–14
 - rewriting, 12–14
 - silos, 9–10
 - Surround Architecture, 10–12
 - Web services, 63–65
- Architecture, building
 - architectural tiers, 302
 - case studies
 - common mistakes, 306–307
 - integration infrastructure, 298–300
 - new applications, 303–306
 - service-oriented architecture, 300–303
 - common middleware API, 300
 - common services, 300
 - future directions, 307–310
 - key points
 - distributed systems implementation, 312–313
 - distributed systems principles, 312
 - guidelines, 311–312
 - middleware technology alternatives, 310–311
 - outsourcing, 308–310
 - standard middleware transport, 299
 - Architecture, choosing
 - business object dependency, 119
 - business process dependency, 119
 - configuration dependency, 118
 - dependencies, 118–119
 - hub, 114–116
 - loosely coupled *versus* tightly coupled, 118–120
 - message format dependency, 118–119
 - message semantic dependency, 119
 - middleware bus, 112–113
 - protocol dependency, 118
 - security dependency, 119
 - session state dependency, 119
 - Web services, 116–117
 - ARPANET, 17, 21
 - ASCII (American Standard Code for Information Interchange), 315
 - As-needed development, 206–209
 - ASP (Active Server Page), 55, 75, 84, 87, 89–90, 315
 - ASP (Application Service Provider), 315
 - Asymmetrical (public key) encryption, 196
 - Asynchronous (send-and-forget, deferrable) messages, 61, 100–103
 - Asynchronous threads, 83
 - Asynchronous (send-and-forget, deferrable) transactions, 100–103
 - Asynchronous Transfer Mode (ATM), 315
 - ATM (Asynchronous Transfer Mode), 315
 - ATM (Automatic Teller Machine), 315
 - ATMI (Tuxedo API set), 29, 315
 - Atomic property, 27
 - Atomicity, Consistency, Isolation, and Durability (ACID), 27, 315
 - Authentication
 - definition, 187
 - distributed systems, 191
 - Authorization. *See* access control.
 - Automatic Teller Machine (ATM), 315
 - Automation, 181–184. *See also* systems management.

B

B2B (Business-to-Business), 315
 Backup, performance, 161–162
 Backup, switching to. *See also* recovery; resiliency.
 batch *versus* online, 128
 client failure, 128–129
 failure detection, 125–126
 heartbeats, 125–126
 overview, 124
 reprocessing lost messages, 129–130
 restarting the system, 127–129
 saving work in progress, 126
 server failure, 128–129
 Backup servers, 124–130
 Bandwidth problems, 162–163
 Banking, business processes, 228
 Batch processes
 converting applications to services, 291–294
 job scheduling, 180
 performance, 155–156
 as a presentation layer, 295
 replacing with online, 293–294
 shortening, 292–294
 Benchmarks *versus* reality, 144–146
 Blocked threads, 83
 BNA (Burroughs Network Architecture), 19, 315
 Boiling lava security model, 192–194
 Books and publications
 Business Rules and Information Systems, 209
 Reengineering the Corporation, 223
 “The Structure of the “THE”-Multiprogramming System,” 82
 Bottleneck detection, 147–150
 Buffer management, 130–132
 Burroughs Network Architecture (BNA), 19, 315
 Business intelligence systems, performance, 159–161
 Business object dependency, 119
 Business process dependency, 119
 Business process design level, 219, 222
 Business process design
 all businesses, 228
 analysis stage, 234–238
 architectural patterns, 231–234
 banking business, 228
 clarification, 234–238
 converting applications to services, 286–289
 Copy out/Copy in applications, 231–234

 data distribution, 224–225
 definition, 225
 description, 225–228
 error handling, 236–237
 flexibility, 237–238
 flow, 226–227
 implementation diagram, 234
 information accuracy, 274
 information processing, 229–230
 integrating, 223–224
 levels of detail, 225–228, 235
 manufacturing business, 228
 middleware support for, 100–103
 migration, 237
 Multiple Centralized applications, 231–234
 Pass Through applications, 231–234
 plan objects, 229–230
 prescriptive, 227
 process analysis technique, 227–228
 resource objects, 229–230
 result objects, 229–230
 Single Centralized applications, 231–234
 tasks, 225–226
 timing, 237
 tracking objects, 229–230
 Business rules, 209–210
Business Rules and Information Systems, 209
 Business-to-Business (B2B), 315

C

C2B (Consumer-to-Business), 316
 Call-back objects, 43
 Cellular MultiProcessing (CMP), 144, 316
 Central Processing Unit (CPU), 316
 Change management, 171–172. *See also* systems management.
 Changing applications. *See* applications, converting to services.
 Choreography, 97
 CICS (Transaction processing system), 26, 29, 117, 149, 285, 315
 CIM (Common Information Model), 179, 316
 Clarification, 234–238, 245–247
 Classification pattern, 209
 Client failure, 128–129
 Client/server protocols, 79
 Clustering, 130–132
 CMP (Cellular MultiProcessing), 144, 316
 COBOL (Common Business-Oriented Language), 316
 Collaboration, 105
 Columns, 22

324 Index

- COM+. *See also* .NET.
 benchmark study, 50–51
 deactivating objects, 48
 definition, 316
 history of, 40
 storing session data, 49
- COM (Component Object Model), 40, 316
- Committing transactions, 26–27
- Common Business-Oriented Language (COBOL), 316
- Common data format, 8
- Common Information Model (CIM), 179, 316
- Common middleware API, 300
- Common mistakes, 306–307
- Common Object Request Broker Architecture (CORBA), 39, 316
- Common services, 300
- Common-language-based interface, 83
- Communication links, 8, 77–79
- Component Object Model (COM), 40, 316
- Components
 architecture, 6–9. *See also* middleware.
 characteristics of, 40
 reusing, 40
- Computation pattern, 209
- COMS (Transaction Processing Environment), 26, 31, 46, 149, 316
- COMWare, 46
- Configuration changes, recovering from, 135
- Configuration dependency, 118
- Connection protocols, 78–79
- Connectionless protocols, 78–79
- Consistent property, 27
- Constraints pattern, 209
- Consumer-to-Business (C2B), 316
- Controlled duplication, 269–271
- Cookies, 55–56
- Copy out/Copy in applications, 231–234
- CORBA (Common Object Request Broker Architecture), 39, 316
- Core server model, 288
- CPU (Central Processing Unit), 316
- CRM (Customer Relationship Management/Manager), 316
- Customer relationship management systems, 266
- Customer Relationship Management/Manager (CRM), 316
- Customer view, 263
- D**
- DARPA (U.S. Department of Defense Advanced Research Projects Agency), 17
- Darwen, Hugh, 209
- Data consistency, 250
- Data distribution, 224–225
- Data duplication, 266, 269–271. *See also* backup; data replication; dual active strategy.
- Data Encryption Standard (DES), 196, 316
- Data integrity, 248
- Data link layer, 82–83
- Data marts, 258–262
- Data migration problem, 274
- Data presentation, 84
- Data replication, performance, 160–161
- Data safety analysis, 138
- Data tier, 109–111
- Databases
 ad hoc queries, performance, 159–160
 backing up, 134–135
 backup and recovery. *See* resiliency.
 columns, 22
 controlled duplication, 269–271
 customer relationship management systems, 266
 data duplication, 266, 269–271
 data migration problem, 274
 data tier, 109–111
 design, 242
 embedded calls *versus* front-end calls, 268–269
 freezing, 293
 heavily used records, 168
 hot spots, 168
 imposing consistency, 271–274
 Information Broker problem, 272
 information controller, 274–275
 integrating different types, 93–94
 integration example, 94
 locking, 168
 manual reprocessing, 136
 matching objects problem, 272
 primary keys, 22
 remote access, 25–26
 rollback facility, 136
 rows, 22
 scalability example, 15
 schemas, 22
 shared data, 267–269
 sources of inaccuracies, 265
 tables, 22
 transactions
 ACID (Atomicity, Consistency, Isolation, and Durability), 27
 across multiple databases, 26–29, 31–34
 committing, 26

- completing. *See* committing.
 - definition, 26
 - distributed processing, 26–29, 31–34
 - two-phase commit, 27
 - validating data entry, 265
- Date, Chris, 209
- DCA (Distributed Communications Architecture), 18–19, 316
- DCE (Distributed Computing Environment), 23, 25, 29, 316
- DCOM (Distributed Component Object Model), 40, 316
- Deactivating objects, 48–49
- DEC, 19, 23
- Deferrable (asynchronous, send-and-forget) messages, 100–103
- Deferrable (asynchronous, send-and-forget) response, 61
- Deferrable (asynchronous, send-and-forget) transactions, 100–103
- Deleting objects, 43–44
- Demilitarized Zone (DMZ), 191, 316
- Dependencies, architectural, 118–119
- DES (Data Encryption Standard), 196, 316
- DHCP (Dynamic Host Configuration Protocol), 316
- Dialogue recovery, 243–245
- Dijkstra, E W, 82
- Directory services, 85
- Disaster recovery, 132–134. *See also* resiliency.
- Discovery, 72–73
- Disk speed, transaction processing, 150
- Distributed applications
 - performance, 156–158
 - versus* silos, 173–177
- Distributed applications, building. *See also* performance; resiliency; scalability.
 - choosing an architecture
 - business object dependency, 119
 - business process dependency, 119
 - configuration dependency, 118
 - dependencies, 118–119
 - hub, 114–116
 - loosely coupled *versus* tightly coupled, 118–120
 - message format dependency, 118–119
 - message semantic dependency, 119
 - middleware bus, 112–113
 - protocol dependency, 118
 - security dependency, 119
 - session state dependency, 119
 - Web services, 116–117
 - tiers
 - data, 109–111
 - presentation, 106–108
 - processing, 108–109
 - versus* services, 111
 - uses for middleware
 - collaboration, 105
 - deferrable messages, 100–103
 - deferrable transactions, 100–103
 - information retrieval, 103–105
 - message queuing, 102
 - real-time messages, 100–103
 - real-time transactions, 100–103
 - supporting business processes, 100–103
 - transaction-to-transaction calls, 101–102
- Distributed Communications Architecture (DCA), 18–19, 316
- Distributed Component Object Model (DCOM), 40, 316
- Distributed Computing Environment (DCE), 23, 25, 29, 316
- Distributed interNet applications Architecture (DNA), 1, 316
- Distributed Management Task Force (DMTF), 179, 316
- Distributed Network Architecture (DNA), 19, 316
- Distributed Relational Database Architecture (DRDA), 26, 316
- Distributed systems
 - implementation, 312–313
 - principles, 312
 - security, 190–194
- Distributed Transaction Integration (DTI), 299–300, 316
- Distributed transaction processing, 26–29, 31–34
- Distributed Transaction Processing (DTP), 316
- DMTF (Distributed Management Task Force), 179, 316
- DMZ (Demilitarized Zone), 191, 316
- DNA (Distributed interNet applications Architecture), 1, 316
- DNA (Distributed Network Architecture), 19, 316
- DNS (Domain Name Service), 52, 78, 85, 316
- Document Type Definition. (DTD), 69, 316
- Domain Name Service (DNS), 52, 78, 85, 316
- Downtime, 134–135, 139
- DRDA (Distributed Relational Database Architecture), 26, 316
- DTD (Document Type Definition.), 69, 316
- DTI (Distributed Transaction Integration), 299–300, 316

326 Index

- DTP (Distributed Transaction Processing), 316
- Dual active strategy
 buffer management, 130–132
 clustering, 130–132
 lock management, 130–132
 lock manager duplicate, 131–132
 log file, 131–132
 overview, 130
 two-database approach, 130–132
- Duplicating data. *See* data duplication; data replication; dual active strategy.
- Durable property, 27
- Dynamic Host Configuration Protocol (DHCP), 316
- E**
- EAE (Enterprise Application Environment), 261, 301, 303, 316
- EAI (Enterprise Application Integration), 70, 91–95, 96, 114, 316
- EBCDIC (Extended Binary Coded Decimal Interchange Code), 316
- E-business, 2–5, 74–75
- EDI (Electronic Data Interchange), 316
- EJB (Enterprise JavaBean). *See also* J2EE.
 benchmark study, 50–51
 deactivating objects, 49
 entity beans, 49, 56
 history of middleware, 40
 passivating objects, 49
 session beans, 49, 56
 stateful session beans, 49, 56
 stateless session beans, 49
- Electronic Data Interchange (EDI), 316
- Embedded calls *versus* front-end calls, 268–269
- Encina, 29
- Encryption, 196–197
- End-to-end data integrity, 248
- End-to-end service provision, 184. *See also* systems management.
- Enterprise Application Environment (EAE), 261, 301, 303, 316
- Enterprise Application Integration (EAI), 70, 91–95, 96, 114, 316
- Enterprise data consistency, 250
- Enterprise JavaBean (EJB). *See* EJB (Enterprise JavaBean).
- Entity beans, 49, 56
- Entrypoint objects, 43
- Enumeration pattern, 209
- Error handling, 139, 236–237. *See also* resiliency.
- Event-based push, information retrieval, 104–105
- Evolving existing architecture, 10–12, 12–14
- Extended Binary Coded Decimal Interchange Code (EBCDIC), 316
- eXtensible Markup Language (XML), 67–69, 320
- eXtreme Programming (XP), 208, 242, 320
- F**
- Failure detection, 125–126
- Fast-Food Fallacy, 281
- Fault tracking and resolution. *See also* systems management.
 definition, 169–170
 responsible group, 172
 software tools, 181
- File Transfer Protocol (FTP), 317
- Firewalls, 192
- Flexibility
 business processes, 237–238
 information retrieval, 104–105
 integration design, 250
- FML (Forms Markup Language), 317. *See also* Tuxedo.
- Front-end calls *versus* embedded calls, 268–269
- FTP (File Transfer Protocol), 317
- G**
- GPI (Generated Programming Interface), 81, 83–84, 317
- GUI (Graphical User Interface), 43, 48, 54, 57, 107, 302, 317
- H**
- Handles, 56
- Hardware architecture gap, 144–146
- Hashing, 196
- Heartbeats, 125–126
- Hewlett-Packard, 23
- High Volume TIP (HVTIP), 319
- HTML (HyperText Markup Language), 51, 54, 67–68, 317
- HTTP (HyperText Transfer Protocol), 54, 65–68, 93, 96, 162–163, 179, 317
- Hub architecture, 114–116
- HVTIP (High Volume TIP), 319
- HyperText Markup Language (HTML), 51, 54, 67–68, 317
- HyperText Transfer Protocol (HTTP), 54, 65–68, 93, 96, 162–163, 179, 317

- I**
- IATA (International Air Transport Association), 18, 317
 - IBM
 - DRDA (Distributed Relational Database Architecture), 26
 - OSF (Open Source Foundation), 23
 - SNA (Systems Network Architecture), 18–19
 - SNA LU6.2, 29
 - IDL (Interface Definition Language), 44, 317
 - IETF (Internet Engineering Task Force), 317
 - IIOF (Internet Inter-ORB Protocol), 317. *See also* CORBA.
 - IL (Intermediate Language), 87, 317
 - Implicit architecture, 91
 - Information access
 - basic process information, 261
 - customer view, 263
 - data marts, 258–262
 - marketing, 263–264
 - process improvement, 262–263
 - process management, 261–262
 - requirements, 264
 - strategic business analysis, 263–264
 - Information accuracy
 - business process problem, 274
 - controlled duplication, 269–271
 - customer relationship management systems, 266
 - data duplication, 266, 269–271
 - data migration problem, 274
 - database administration. *See* information controller.
 - embedded calls *versus* front-end calls, 268–269
 - imposing database consistency, 271–274
 - Information Broker problem, 272
 - information controller, 274–275
 - matching objects problem, 272
 - shared data, 267–269
 - sources of inaccuracies, 265
 - validating data entry, 265
 - Information Broker problem, 272
 - Information controller, 274–275
 - Information retrieval, 103–105
 - Information Technology (IT), 317
 - Integrating applications. *See* applications, converting to services.
 - Integration design. *See also* task-level design.
 - analysis, 247–250
 - clarification, 245–247
 - database design, 242
 - dialogue recovery, 243–245
 - ease of implementation, 250
 - end-to-end data integrity, 248
 - enterprise data consistency, 250
 - evaluating quality of, 251–254
 - flexibility, 250
 - information requirements, 241–242
 - IT strategy compliance, 250
 - level of detail, 241
 - locks, 244
 - long transactions, 243–245
 - loosely coupled applications, 251–253
 - performance, 248
 - process description, 245–250
 - pseudo locks, 244
 - scalability, 248
 - security, 248–249
 - system management, 249–250
 - task/message flow diagrams, 245–247
 - tiers, 253–254
 - tightly coupled applications, 251–253
 - UML sequence diagrams, 245–247
 - Integration infrastructure, 298–300
 - Integrity, 80–81
 - Interface Definition Language (IDL), 44, 317
 - Interface size mismatch, 281–282
 - Interface styles, 83
 - Interfaces
 - existing, changing, 284–286
 - Internet applications, 44
 - object *versus* procedure, 40
 - Intermediate Language (IL), 87, 317
 - International Air Transport Association (IATA), 18, 317
 - International Organization for Standardization (ISO), 19–21, 317
 - Internet applications. *See also* object-oriented middleware; TCM (Transactional Component Middleware).
 - cookies, 55–56
 - effects on IT environment, 51–52
 - entity beans, 49, 56
 - handles, 56
 - identifying the user, 52
 - monitor resolution, 52
 - security, 52
 - session beans, 49, 56
 - session management, 54–55
 - stateful session beans, 49, 56
 - stateless session beans, 49
 - versus* traditional applications, 52–53
 - typical hardware configuration, 53
 - user control, 52
 - Internet Engineering Task Force (IETF), 317

328 Index

- Internet Inter-ORB Protocol (IIOP), 317.
See also CORBA.
 - Internet Protocol (IP), 21, 52, 54, 78–79, 82, 85, 127, 158, 175–176, 182–184, 317
 - Internet Service Provider (ISP), 317
 - Interoperability, 92
 - IP (Internet Protocol), 21, 52, 54, 78–79, 82, 85, 127, 158, 175–176, 182–184, 317
 - IPSec (Secure Internet Protocol), 196, 317
 - ISO (International Organization for Standardization), 19–21, 317
 - Isolation property, 27
 - ISP (Internet Service Provider), 317
 - IT (Information Technology), 317
 - IT architecture. *See* architecture.
 - IT strategy compliance, 250
 - Iterative development, 207
- J**
- J2EE (Java 2 Enterprise Edition), 87–89, 317
 - Java, Internet applications, 40
 - Java Data Object (JDO), 26, 317. *See also* J2EE.
 - Java DataBase Connectivity (JDBC), 26, 317.
See also J2EE.
 - Java Message Service (JMS), 317. *See also* J2EE.
 - Java Naming and Directory Interface (JNDI), 317. *See also* J2EE.
 - Java Virtual Machine. (JVM), 40, 87, 146, 151–152, 317
 - JavaServer Page (JSP), 317. *See also* J2EE.
 - JCL (Job Control Language), 317
 - JDBC (Java DataBase Connectivity), 26, 317.
See also J2EE.
 - JDO (Java Data Object), 26, 317. *See also* J2EE.
 - JIT (Just-in-Time), 87, 317
 - JMS (Java Message Service), 317. *See also* J2EE.
 - JNDI (Java Naming and Directory Interface), 317. *See also* J2EE.
 - Job Control Language (JCL), 317
 - JSP (JavaServer Page), 317. *See also* J2EE.
 - Just-in-Time (JIT), 87, 317
 - JVM (Java Virtual Machine.), 40, 87, 146, 151–152, 317
- K**
- KDC (Key Distribution Center), 317. *See also* Kerberos.
 - Kerberos, 197
 - Key Distribution Center (KDC), 317. *See also* Kerberos.
- L**
- LAN (Local Area Network), 317
 - Language transparency, 22–23
 - Large message problem, 281–282
 - Layering, 82–83. *See also* tiers.
 - Levels of design, 217–220, 222
 - LINC *See* EAE (Enterprise Application Environment)
 - List constraints pattern, 209
 - Load balancing, 158–159
 - Local Area Network (LAN), 317
 - Location security, 192
 - Location transparency, 22–23
 - Lock management, 130–132
 - Lock manager duplicate, 131–132
 - Locks, 244
 - Log files, 131–132, 162
 - Long transactions, 243–245
 - Loosely coupled applications, 251–253, 307–310
 - Loosely coupled architecture, 118–120
- M**
- Management Information Base (MIB), 179, 318. *See also* SNMP.
 - Manuals. *See* books and publications.
 - Manufacturing, business processes, 228
 - Marketing, information access, 263–264
 - Marketing middleware vendor architecture, 90–91
 - Matching objects problem, 272
 - MCP (Unisys operating system), 284, 315, 317
 - MDA (Model Driven Architecture), 207, 317
 - Memory, transaction processing, 149
 - Message semantic dependency, 119
 - Message-based interface, 83
 - Messages
 - delivery integrity, 22–23
 - format dependency, 118–119
 - format integrity, 22–23
 - queuing, 29–34, 102, 153
 - Messaging, 67, 70
 - MIB (Management Information Base), 179, 318. *See also* SNMP.
 - Microsoft Message Queuing (MSMQ), 30, 318
 - Microsoft Transaction Server (MTS), 46, 318
 - Middle tier, 285–286
 - Middleware
 - blocked threads, 23
 - calling remote objects. *See* object-oriented middleware.
 - characteristics of, 21–23

- client/server technology, 22
 - definition, 7
 - distributed transaction processing, 26–29, 31–34
 - ease-of-use, 21–23
 - interoperability, 92, 95–97
 - language transparency, 22–23
 - linking multiple technologies, 92, 95–97
 - location transparency, 22–23
 - message delivery integrity, 22–23
 - message format integrity, 22–23
 - message queuing, 29–34
 - multithreading, 22–23
 - processes, 22
 - remote database access, 25–26
 - RPC (Remote Procedure Call), 23–25
 - standard middleware transport, 299
 - technology alternatives, 310–311
 - threads, 22–23
 - between transaction servers, changing, 289–291
 - transactional. *See* RPC (Remote Procedure Call).
- Middleware, elements
- asynchronous threads, 83
 - blocked threads, 83
 - client/server protocols, 79
 - common-language-based interface, 83
 - communication link, 77–79
 - connection protocols, 78–79
 - connectionless protocols, 78–79
 - data link layer, 82–83
 - data presentation, 84
 - directory services, 85
 - event-based threads, 83
 - integrity, 80–81
 - interface styles, 83
 - layering, 82–83. *See also* tiers.
 - message-based interface, 83
 - naming, 85
 - network layer, 82–83
 - object management, 85
 - operation-call-based interface, 83
 - peer-to-peer protocols, 79
 - physical layer, 82–83
 - process control, 83–84, 84–85
 - programmatic interface, 81, 83–84
 - protocols, 78–79, 79–81
 - push protocols, 79–80
 - resource management, 85
 - security, 85
 - server control, 84–85
 - SOAP protocol, 80
 - synchronous threads, 83
 - system management, 85–86
 - thread control, 83–84, 84–85
 - transport layer, 82–83
 - two-phase commit, 81
 - unblocked threads, 83
- Middleware, history of
- ARPANET, 17, 21
 - BNA (Burroughs Network Architecture), 19
 - DARPA (U.S. Department of Defense Advanced Research Projects Agency), 17
 - DCA (Distributed Communications Architecture), 18–19
 - DNA (Distributed Network Architecture), 19
 - EJB (Enterprise JavaBean), 40, 49–51, 56
 - first use of the term, 17
 - IATA (International Air Transport Association), 18
 - Internet applications, 49–56
 - ISO (International Organization for Standardization), 19–21
 - object-orientation, 40–45
 - open systems movement, 19–21
 - OSI (Open Systems Interconnection), 19–21
 - SITA (Société Internationale de Télécommunications Aeronautiques), 18
 - SNA (Systems Network Architecture), 18–19
 - SQL (System Query Language), 20
 - SWIFT (Society for Worldwide Interbank Financial Telecommunication), 18
 - TCP/IP (Transmission Control Protocol/Internet Protocol), 20–21
 - Web services, 59–75
- Middleware, uses for
- collaboration, 105
 - deferrable messages, 100–103
 - deferrable transactions, 100–103
 - information retrieval, 103–105
 - message queuing, 102
 - real-time messages, 100–103
 - real-time transactions, 100–103
 - supporting business processes, 100–103
 - transaction-to-transaction calls, 101–102
- Middleware, vendor architectures. *See also* architecture.
- implicit, 91
 - J2EE, 87–89
 - marketing, 90–91
 - mixing and matching, 90

330 Index

- Middleware, vendor architectures (*cont.*)
 - needs assessment, 90
 - .NET, 87–88
 - positioning, 90
 - using, 89
 - vendor platform architectures, 87–88
 - vendor-distributed architectures, 88–89
- Middleware bus architecture, 112–113
- Migration, business processes, 237
- Model Driven Architecture (MDA), 207, 317
- Monitor and Control, 177
- Monitor resolution, 52
- Morgan, Tony, 209
- MQSeries:
- MSMQ (MicroSoft Message Queuing), 30, 318
- MTS (Microsoft Transaction Server), 46, 318
- Multiple Centralized applications, 231–234
- Multithreading, 22–23
- MVS (IBM operating system), 285, 318

- N**
- Naming, 85
- Naming and directory services, 8
- .NET (Operating system platform or framework), 87–88, 318
- .NET Enterprise Services. *See* COM+.
- Network File System (NFS), 197, 318
- Network layer, 82–83
- NFS (Network File System), 197, 318

- O**
- Object interfaces, transaction processing, 151
- Object Linking and Embedding (OLE), 318
- Object Linking and Embedding DataBase (OLE DB), 26, 318
- Object management, 85
- Object Management Group (OMG), 39, 46, 207, 210, 318
- Object Request Broker (ORB), 39, 318
- Object Transaction Manager (OTM), 46, 318
- Object-Oriented (OO), 318
- Object-oriented middleware. *See also* Internet applications; TCM (Transactional Component Middleware).
 - accessing remote objects, 41–42, 44–45
 - advantages of, 44–45
 - agent objects, 43
 - call-back objects, 43
 - COM+, 40
 - COM (Component Object Model), 40
 - components, 40
 - DCOM (Distributed COM), 40
 - deleting objects, 43–44
 - EJB (Enterprise JavaBean), 40
 - entrypoint objects, 43
 - getting object references, 42
 - IDL (Interface Definition Language) file, 44
 - interfaces, 44
 - Java, 40
 - object interfaces *versus* procedure interfaces, 40
 - RPC (Remote Procedure Call), 44
 - patterns, 42–44
 - proxy objects, 43
 - server objects, 42
- Objects
 - agent, 43
 - call-back, 43
 - deleting, 43–44
 - entrypoint, 43
 - proxy, 43
 - references, getting, 42
 - remote, accessing, 41–42, 44–45
 - server, 42
- ODBC (Open DataBase Connectivity), 26, 318
- OLE (Object Linking and Embedding), 318
- OLE DB (Object Linking and Embedding DataBase), 26, 318
- OLTP (On-Line Transaction Processing), 318
- OMG (Object Management Group), 39, 46, 207, 210, 318
- ONC (Open Network Computing), 318
- Onion security model, 190–192
- On-Line Transaction Processing (OLTP), 318
- OO (Object-Oriented), 318
- Open DataBase Connectivity (ODBC), 26, 318
- Open Group DTP API set (TX), 319
- Open Group DTP API set (TxRPC), 38, 319
- Open Group DTP API set (XATMI), 320
- Open Group DTP protocol for distributed database coordination (XA), 320
- Open Network Computing (ONC), 318
- Open Software Foundation (OSF), 23, 318
- Open Systems Interconnection (OSI), 19–21, 318
- Open Systems Interconnection-Transaction Processing (OSI-TP), 285, 306, 318
- Open systems movement, 19–21

- Operating system platform or framework (.NET), 87–88, 318
 - Operation-call-based interface, 83
 - Operations. *See also* systems management.
 - definition, 169–170
 - responsible group, 172
 - software tools, 177–180
 - Oracle Generic Connectivity, 26
 - Oracle Transparent Gateway, 26
 - ORB (Object Request Broker), 39, 318
 - Order Entry system example, 2–5
 - OS2200 (Unisys operating system), 284, 318
 - OSF (Open Software Foundation), 23, 318
 - OSI (Open Systems Interconnection), 19–21, 318
 - OSI-TP (Open Systems Interconnection-Transaction Processing), 285, 306, 318
 - OTM (Object Transaction Manager), 46, 318
 - Outsourcing, 308–310
 - OWL (Web Ontology Language), 71, 318
- P**
- Papers. *See* books and publications.
 - Pass Through applications, 231–234
 - Passenger Name Record (PNR), 318
 - Passivating objects, 49
 - Passwords, 193
 - Patterns, 42–44
 - PCI (Peripheral Component Interconnect), 318
 - Peer-to-peer protocols, 79
 - Performance. *See also* distributed applications, building.
 - backup and recovery, 161–162
 - bandwidth problems, 162–163
 - batch, 155–156
 - benchmarks *versus* reality, 144–146
 - business intelligence systems, 159–161
 - designing for, 165–166
 - distributed applications, 156–158
 - hardware architecture gap, 144–146
 - improvement strategies, 146
 - integration design, 248
 - load balancing, 158–159
 - locking, 168
 - log file size issues, 162
 - predictability, 163
 - RTT (Round Trip Time), 163
 - SOAP, 162–164
 - transaction processing
 - bottleneck detection, 147–150
 - disk speed, 150
 - memory, 149
 - message queuing, 153
 - object interfaces, 151
 - queuing, 148
 - realtime transactions, remote database access, 154
 - requirements history, 154–155
 - transactional component containers, 151–152
 - two-phase commit, 152–153
 - transmission time problems, 163
 - Web services, 162–164
 - Performance management. *See also* systems management.
 - definition, 169–170
 - responsible group, 172
 - software tools, 180
 - Peripheral Component Interconnect (PCI), 318
 - Pet Store benchmark, 50–51
 - Physical layer, 82–83
 - PIM (Platform-Independent Model), 318. *See also* MDA.
 - Plan objects, business processes, 229–230
 - Platform-Specific Model (PSM), 318. *See also* MDA.
 - PNR (Passenger Name Record), 318
 - Popper, Karl, 217
 - Positioning middleware vendor architecture, 90
 - Predictability, and performance, 163
 - Prescriptive business processes, 227
 - Presentation layer, 280–286
 - Presentation tier, 106–108
 - Preventive maintenance, 135
 - Primary keys, 22
 - Private key (symmetrical) encryption, 196
 - Process analysis, business processes, 227–228
 - Process and Manage, 177
 - Process control, 83–84, 84–85
 - Processes, 22
 - Processing tier, 108–109
 - Programmatic interface, 8, 81, 83–84
 - Programs. *See* applications.
 - Protection, 187, 189–190
 - Protocol dependency, 118
 - Protocols
 - architecture, 8
 - client/server, 79
 - connection, 78–79
 - connectionless, 78–79
 - description, 78–79, 79–81
 - peer-to-peer, 79
 - push, 79–80
 - SOAP, 80

332 Index

- Providers
 - description, 60–62
 - relationship with requesters, 72–73
 - rules for, 72–73
 - Proxy objects, 43
 - Pseudo locks, 244
 - PSM (Platform-Specific Model), 318. *See also* MDA.
 - Public key (asymmetrical) encryption, 196
 - Publications. *See* books and publications.
 - Pull-based, information retrieval, 104–105
 - Push protocols, 79–80
- Q**
- Queuing, transaction processing, 148
- R**
- RAD (Rapid Application Development), 207–208, 318
 - Rapid Application Development (RAD), 207–208, 318
 - Rational Unified Process (RUP), 220, 319
 - RDBMS (Relational DataBase Management System), 318
 - Real-time (synchronous) messages, 100–103
 - Real-time (synchronous) response, 61
 - Real-time (synchronous) transactions, 100–103
 - Recovery. *See also* backup; resiliency.
 - dual active strategy
 - buffer management, 130–132
 - clustering, 130–132
 - lock management, 130–132
 - lock manager duplicate, 131–132
 - log file, 131–132
 - overview, 130
 - two-database approach, 130–132
 - performance, 161–162
 - switching to backup
 - batch *versus* online, 128
 - client and server failure, 128–129
 - client failure, 128
 - failure detection, 125–126
 - heartbeats, 125–126
 - overview, 124
 - reprocessing lost messages, 129–130
 - restarting the system, 127–129
 - saving work in progress, 126
 - server and client failure, 128–129
 - server failure, 128
 - Reengineering the Corporation*, 223
 - Registries, 72–73
 - Relational DataBase Management System (RDBMS), 318
 - Remote database access, 25–26
 - Remote Method Invocation (RMI), 318. *See also* J2EE.
 - Remote Procedure Call (RPC). *See* RPC (Remote Procedure Call).
 - Reprocessing lost messages, 129–130
 - Request For Comment (RFC), 318
 - Requesters
 - description, 60–62
 - relationship with providers, 72–73
 - rules for, 72–73
 - Request-response messages. *See* real-time messages.
 - Resiliency. *See also* distributed applications, building.
 - application software failure, 136–137
 - backup servers, 124–130
 - benefits of, 139
 - configuration changes, 135
 - data safety analysis, 138
 - databases
 - backing up, 134–135
 - duplicate. *See* dual active strategy.
 - manual reprocessing, 136
 - rollback facility, 136
 - disaster recovery, 132–134
 - downtime, cost of, 139
 - downtime, planned, 134–135
 - error handling, 139
 - error-handling analysis, 138
 - preventive maintenance, 135
 - recovery, dual active strategy
 - buffer management, 130–132
 - clustering, 130–132
 - lock management, 130–132
 - lock manager duplicate, 131–132
 - log file, 131–132
 - overview, 130
 - two-database approach, 130–132
 - recovery, switching to backup
 - batch *versus* online, 128
 - client and server failure, 128–129
 - client failure, 128
 - failure detection, 125–126
 - heartbeats, 125–126
 - overview, 124
 - reprocessing lost messages, 129–130
 - restarting the system, 127–129
 - saving work in progress, 126
 - server and client failure, 128–129
 - server failure, 128
 - strategy, 137–139
 - system software failure, 134

- techniques, 132–134
 - uptime analysis, 138
 - Resource management, 85
 - Resource Manager (RM), 318. *See also* Open Group DTP.
 - Resource objects, 229–230
 - Restarting the system. *See* resiliency.
 - Restore. *See* resiliency.
 - Result objects, 229–230
 - Reuse, 211–215
 - Reuse model, 288–289
 - Rewriting existing architecture, 12–14
 - RFC (Request For Comment), 318
 - RM (Resource Manager), 318. *See also* Open Group DTP.
 - RMI (Remote Method Invocation), 318. *See also* J2EE.
 - Roles, and security
 - assigning, 189
 - distributed systems, 191–192
 - needs assessment, 188–189
 - Round Trip Time (RTT), 163, 319
 - Rows, 22
 - RPC (Remote Procedure Call)
 - alternatives to, 28–29
 - marshalling, 23–25, 30
 - mechanisms for, 23
 - multithreading, 23–25
 - versus* object interfaces, 44
 - serialization, 23
 - skeletons, 23
 - stubs, 23
 - uses for, 23
 - RSA (RSA Security Inc.), 319
 - RSA Security Inc. (RSA), 319
 - RTT (Round Trip Time), 163, 319
 - Rule patterns, 209
 - business rules, 209–210
 - classification rule pattern, 209
 - computation rule pattern, 209
 - constraints rule pattern, 209
 - enumeration rule pattern, 209
 - list constraints rule pattern, 209
 - rule patterns, 209
 - waterfall method, 206–207
 - Rules Engine, 178
 - RUP (Rational Unified Process), 220, 319
- S**
- SAA (Systems Application Architecture), 319
 - Scalability. *See also* distributed applications, building.
 - architecture, 15
 - backup and recovery, 161–162
 - bandwidth problems, 162–163
 - batch, 155–156
 - benchmarks *versus* reality, 144–146
 - business intelligence systems, 159–161
 - designing for, 165–166
 - distributed applications, 156–158
 - hardware architecture gap, 144–146
 - improvement strategies, 146
 - integration design, 248
 - load balancing, 158–159
 - locking, 168
 - log file size issues, 162
 - predictability, 163
 - RTT (Round Trip Time), 163
 - SOAP, 162–164
 - transaction processing
 - bottleneck detection, 147–150
 - disk speed, 150
 - memory, 149
 - message queuing, 153
 - object interfaces, 151
 - queuing, 148
 - realtime transactions, remote database access, 154
 - requirements history, 154–155
 - transactional component containers, 151–152
 - two-phase commit, 152–153
 - transmission time problems, 163
 - Web services, 162–164
 - Schemas, 22
 - Scope principle, systems management, 182–183
 - Screen scraping, 281
 - Secure Internet Protocol (IPSec), 196, 317
 - Secure Sockets Layer (SSL), 195, 319
 - Security. *See also* systems management.
 - access control, definition, 187
 - access control, XML messages, 199
 - architecture, 8, 200–202
 - asymmetrical (public key) encryption, 196
 - authentication, 187, 191
 - boiling lava model, 192–194
 - dependency, 119
 - distributed systems, 190–194
 - encryption, 196–197
 - firewalls, 192
 - hashing, 196
 - integration design, 248–249
 - Internet applications, 52
 - Kerberos, 197
 - location, 192
 - management, 187
 - middleware element, 85

334 Index

- Security (*cont.*)
 - needs assessment, 188–190
 - onion model, 190–192
 - passwords, 193
 - people and roles, 188–189, 191–192
 - private key encryption, 196
 - protection, 187, 189–190
 - signed security tokens, 195–196
 - SSL (Secure Sockets Layer), 195
 - stateless Web services, 194–195
 - statements of intent, 189
 - symmetrical (private key) encryption, 196
 - tokens, 195–199
 - user codes, 193
 - Web services systems, 194–199
 - XACML (XML Access Control Markup Language), 199
- Send-and-forget (asynchronous, deferrable)
 - messages, 61, 100–103
- Send-and-forget (asynchronous, deferrable)
 - transactions, 100–103
- Serendipitous reuse, 211–214
- Server control, 84–85
- Server objects, 42
- Server process control, 8
- Service-oriented architecture, 300–303
- Service-Oriented Architecture (SOA), 9, 319
- Services *versus* tiers, 111
- Session beans, 49, 56
- Session management, 49, 54–55
- Session state dependency, 119
- SGML (Standard Generalized Markup Language), 67–69, 319
- Shared data, 267–269
- Signed security tokens, 195–196
- Silos
 - description, 9–10
 - versus* distributed environments, 173–177
 - evolving, 10–12, 12–14
 - and monolithic development, 215
 - rewriting, 12–14
 - Surround Architecture, 10–12
- Simple Mail Transfer Protocol (SMTP), 319
- Simple Network Management Protocol (SNMP), 178
- Simple Object Access Protocol (SOAP). *See* SOAP (Simple Object Access Protocol).
- Single Centralized applications, 231–234
- Single Point Operations (SPO), 179, 319
- SITA (Société Internationale de Télécommunications Aeronautiques), 18, 319
- SMTP (Simple Mail Transfer Protocol), 319
- SNA (Systems Network Architecture), 18–19, 319
- SNA LU6.2, 29
- SNMP (Simple Network Management Protocol), 178
- SOA (Service-Oriented Architecture), 9, 319
- SOAP (Simple Object Access Protocol)
 - message elements, 70
 - messaging, 67
 - middleware protocol, 80
 - performance, 162–164
- Société Internationale de Télécommunications Aeronautiques (SITA), 18, 319
- Society for Worldwide Interbank Financial Telecommunication (SWIFT), 18, 319
- Sperry, 18–19
- SPO (Single Point Operations), 179, 319
- SQL (System Query Language), 20, 319
- SSL (Secure Sockets Layer), 195, 319
- Standard Generalized Markup Language (SGML), 67–69, 319
- Standard middleware transport, 299
- Stateful session beans, 49, 56
- Stateless session beans, 49
- Stateless Web services, 194–195
- Statements of intent, 189
- Surround Architecture, 10–12
- SWIFT (Society for Worldwide Interbank Financial Telecommunication), 18, 319
- Symmetrical (private key) encryption, 196
- Synchronous messages. *See* real-time messages.
- Synchronous (real-time) messages, 61, 100–103
- Synchronous threads, 83
- Synchronous (real-time) transactions, 100–103
- System Query Language (SQL), 20, 319
- System software
 - failure, 134
 - locking, 168
- Systems Application Architecture (SAA), 319
- Systems management. *See also* security.
 - accounting, 169–170, 180
 - administration, 169–170, 180
 - application contribution to systems management, 184–185
 - architecture, 8
 - automation, 181–184
 - batch job scheduling, 180
 - change management, 171–172
 - CIM (Common Information Model), 179

- DMTF (Distributed Management Task Force), 179
 - end-to-end service provision, 184
 - evolutionary approach, 185
 - fault tracking and resolution
 - definition, 169–170
 - responsible group, 172
 - software tools, 181
 - functional categories, 169–170
 - functional relationships, 170–171
 - integration design, 249–250
 - MIB (Management Information Base), 179
 - middleware element, 85–86
 - model diagram, 178
 - Monitor and Control, 177
 - operations
 - definition, 169–170
 - responsible group, 172
 - software tools, 177–180
 - performance management
 - definition, 169–170
 - responsible group, 172
 - software tools, 180
 - primary function, 171
 - principles, 182–185
 - Process and Manage, 177
 - Rules Engine, 178
 - scope principle, 182–183
 - silos *versus* distributed environments, 173–177
 - SNMP (Simple Network Management Protocol), 178
 - SPO (Single Point Operations), 179
 - WBEM (Web-Based Enterprise Management), 179
 - Systems Network Architecture (SNA), 18–19, 319
- T**
- Tables, 22
 - Task-level design. *See also* integration design.
 - business processes, 225–226
 - definition, 219
 - related responsibilities, 222
 - Task/message flow diagrams, 245–247
 - TCM (Transactional Component Middleware)
 - See also* COM+
 - See also* EJB (Enterprise JavaBean)
 - See also* Internet applications
 - See also* object-oriented middleware advantages, 46–48
 - benchmark study, 50–51
 - COM+, 48–49
 - containers, 46
 - deactivating objects, 49
 - definition, 46
 - EJB (Enterprise JavaBean), 49–50, 50–51
 - .NET, 46, 50–51
 - object wrappers, 48
 - passivating objects, 49
 - TCP (Transmission Control Protocol), 20–22, 30, 54, 68, 77, 78–80, 82, 115, 117, 127, 163, 319
 - TCP/IP (Transmission Control Protocol/Internet Protocol), 20–21
 - Technologists, as architects, 15–16
 - TGT (Ticket Granting Ticket), 197, 319
 - “The Structure of the “THE”-Multiprogramming System,” 82
 - Thread control, 83–84, 84–85
 - Threads
 - asynchronous, 83
 - blocked, 23, 83
 - event-based, 83
 - multithreading, 22–23
 - synchronous, 83
 - thread control, 83–84, 84–85
 - unblocked, 83
 - Ticket Granting Ticket (TGT), 197, 319
 - Tiers. *See also* layering.
 - architectural, 302
 - data, 109–111
 - integration design, 253–254
 - middle tier, 285–286
 - presentation, 106–108
 - processing, 108–109
 - versus* services, 111
 - Tightly coupled applications
 - future directions, 307–310
 - integration design, 251–253
 - Tightly coupled architecture, 118–120
 - Time-based push, information retrieval, 104–105
 - Timeliness, information retrieval, 103
 - Timing, business processes, 237
 - TIP (Transaction processing system), 26, 149, 319
 - TLS (Transport Layer Security), 195, 319
 - TM (Transaction Manager), 319. *See also* Open Group DTP.
 - Tokens, 195–199
 - TPPC (Transaction Processing Performance Council), 319
 - tpmC (Transactions Per Minute, Type C), 319
 - Tracking objects, 229–230
 - Transaction design level, 219–220, 222

336 Index

- Transaction Manager (TM), 319. *See also* Open Group DTP.
 - Transaction processing performance
 - bottleneck detection, 147–150
 - disk speed, 150
 - memory, 149
 - message queuing, 153
 - object interfaces, 151
 - queuing, 148
 - realtime transactions, remote database access, 154
 - requirements history, 154–155
 - transactional component containers, 151–152
 - two-phase commit, 152–153
 - realtime transactions, remote database access, 154
 - Transaction Processing Environment (COMS), 26, 31, 46, 149, 316
 - Transaction Processing Performance Council (TPPC), 319
 - Transaction processing system (CICS), 26, 29, 117, 149, 285, 315
 - Transaction processing system (TIP), 26, 149, 319
 - Transactional component containers, 151–152
 - Transactional Component Middleware (TCM). *See* TCM (Transactional Component Middleware).
 - Transactions
 - ACID (Atomicity, Consistency, Isolation, and Durability), 27
 - committing, 26
 - definition, 26
 - distributed processing, 26–29, 31–34
 - two-phase commit, 27
 - Transactions Per Minute, Type C (tpmC), 319
 - Transaction-to-transaction calls, 101–102
 - Transmission Control Protocol (TCP), 319
 - Transmission Control Protocol/Internet Protocol (TCP/IP), 20–21
 - Transmission time problems, 163
 - Transport layer, 82–83
 - Transport Layer Security (TLS), 195, 319
 - Tuxedo, 29
 - Tuxedo API set (ATMI), 29, 315
 - Two-database recovery approach, 130–132
 - Two-phase transaction commit, 27, 81, 152–153
 - TX (Open Group DTP API set), 319
 - TxRPC (Open Group DTP API set), 38, 319
- ## U
- UDDI (Universal Description, Discovery, and Integration), 70, 72–73, 319
 - UDP (User Datagram Protocol), 78, 197, 319
 - UML (Unified Modeling Language), 207, 320
 - UML sequence diagrams, 245–247
 - Unblocked threads, 83
 - Unified Modeling Language (UML), 207, 320
 - Uniform Resource Identifier (URI), 320
 - Uniform Resource Locator (URL), 320
 - Unisys operating system (MCP), 284, 315, 317
 - Unisys operating system (OS2200), 284, 318
 - Unisys Standard Airline System (USAS), 304–306, 320
 - Universal Description, Discovery, and Integration (UDDI), 70, 72–73, 319
 - Up front development, 206–209
 - Uptime analysis, 138
 - URI (Uniform Resource Identifier), 68–69, 320
 - URL (Uniform Resource Locator), 68, 320
 - U.S. Department of Defense Advanced Research Projects Agency (DARPA), 17
 - Usability, information retrieval, 103–104
 - USAS (Unisys Standard Airline System), 304–306, 320
 - User codes, 193
 - User Datagram Protocol (UDP), 78, 197, 319
- ## V
- Validating data entry, 265
 - VB (Visual Basic), 320
 - Vendor platform architectures, 87–88
 - Vendor-distributed architectures, 88–89
 - Visual Basic (VB), 320
- ## W
- W3C (World Wide Web Consortium), 320
 - WAN (Wide Area Network), 320
 - WAP (Wireless Application Protocol), 320
 - Waterfall development, 206–207
 - WBEM (Web-Based Enterprise Management), 179, 320
 - Web applications. *See* Internet applications.
 - Web Ontology Language (OWL), 71, 318
 - Web Service Coordination Framework (WS-CF), 320. *See also* WS-CAF.
 - Web services
 - concepts, 59–65
 - deferrable response, 61
 - definition, 65
 - discovery, 72–73

- e-business (B2B) approach, 74–75
 - future directions, 307–310
 - intra-organizational approach, 74
 - messaging, 67, 70
 - performance, 162–164
 - providers
 - description, 60–62
 - relationship with requesters, 72–73
 - rules for, 72–73
 - real time response, 61
 - registries, 72–73
 - requesters
 - description, 60–62
 - relationship with providers, 72–73
 - rules for, 72–73
 - SOAP, 67, 70
 - technology overview, 65–67, 70–71
 - typical architecture, 63–65
 - UDDI (Universal Discovery, Description, and Integration), 70, 72–73
 - using, 71
 - WSDL (Web Services Definition Language), 65, 70
 - XML (eXtensible Markup Language), 67–69
 - Web services architecture, 116–117
 - Web Services Composite Application Framework (WS-CAF), 320
 - Web Services Context (WS-CTX), 320. *See also* WS-CAF.
 - Web Services Definition (WSD), 320
 - Web Services Definition Language (WSDL), 65, 70, 320
 - Web services systems, 194–199
 - Web Transaction Management (WS-TXM), 320. *See also* WS-CAF.
 - Web-Based Enterprise Management (WBEM), 179, 320
 - Weinberg, Gerald, 281
 - Wide Area Network (WAN), 320
 - Wireless Application Protocol (WAP), 320
 - World Wide Web (WWW), 320
 - World Wide Web Consortium (W3C), 71–72, 320
 - Wrappers, object, 48
 - Wrapping interfaces, 284–286
 - WS-CAF (Web Services Composite Application Framework), 320
 - WS-CF (Web Service Coordination Framework), 320. *See also* WS-CAF.
 - WS-CTX (Web Services Context), 320. *See also* WS-CAF.
 - WSD (Web Services Definition), 320
 - WSDL (Web Services Definition Language), 65, 70, 320
 - WS-TXM (Web Transaction Management), 320. *See also* WS-CAF.
 - WWW (World Wide Web), 320
- X**
- XA (Open Group DTP protocol for distributed database coordination), 320
 - XACML (XML Access Control Markup Language), 199, 320
 - XATMI (Open Group DTP API set), 320
 - XATMI protocol, 29
 - XML (eXtensible Markup Language), 67–69, 320
 - XML Access Control Markup Language (XACML), 199, 320
 - X/Open consortium, 28
 - X/Open DTP model, 28
 - XP (eXtreme Programming), 208, 242, 320

