Index

: (colon), in JMX object names, 435
, (comma), in JMX object names, 435
* (asterisk)
  in JMX object names, 435
  wildcard, 258, 259
= (equal sign), in JMX object names, 435
? (question mark)
  in JMX object names, 435
  wildcard, 258, 259

A
Active attribute
  definition, 265
  RelationService class, 313
  TimerMBean class, 297
adapter tools, 69–70
adapters
  deployment patterns, 340
  description, 69–70
  JCA resource adapters, 443
  JSR, 59–60
  registering MBeans, 410–411
adding
  custom descriptors, 214–215
  notification listeners, 331
  addListener() method, 331
  addNotification() method, 297
  addNotificationListener() method, 79–82, 261–262
  addRelation() method, 314
Admin Runtime Module, 57
Admin View Module, 57
AdminClient interface, 321–322
AdminClientFactory() method, 321–322
AdventNet ManageEngine, 521
AdventNet Middleware Manager, 522
AdventNet Web NMS, 523
AdventNetAgent toolkit, 521
agent level compliance, 62
agents
  definition, 339
  description, 10
  domains, 11
  implementations, 519–522
  MBeanServer, 68
  vs. MBeanServer, 339
  MLet, 69
  monitoring service, 69
  relation service, 69
  required services, 69
  timer service, 69
AgentX, 11
aggregate data types, 127–129
AIC (Application Information and Control), 15
AllNotificationIDs attribute, 297
AllRelationIDs attribute, 313
AllRelationTypeNames attribute, 313
and() method, 256
anySubString() method, 256–257
Apache server
  MBeans, 122–124
  model MBeans, 168, 191
  monitoring
    with MBean Refresher, 269–270
Apache server continued
  status, 268–270
  for traffic spikes, 281–282
  optimizing, 275–277
ApacheMBean class
  activating, 122–124
  Apache management interface, 100, 102–108
  creating and configuring monitors, 270–271
  description, 269–270
  listening for notifications, 271–273
  using GaugeMonitor, 286–287
ApacheMBeanException
  activating Apache MBean, 123
  ApacheMBean interface, 269–270
  defining the Apache management interface, 100
  model MBean example, 153–157
apachembeans.xml, 223–233
ApacheMonitor class
  attributes, 273
  conditions, 273
  creating and configuring monitors, 270–271
  listening for notifications, 272–273
  MLeT examples, 303–306
  notification types, 274
  using, 275–277
  using GaugeMonitor, 286–287
ApacheServer class, 139, 141–157
ApacheServerManager class, 138–140, 141–157
API-based reconfiguration, 19
AppClientModule class, 442–444
Application Information and Control (AIC), 15
Application Response Management (ARM), 15–16
application servers. See J2EE.
applications
  candidates for JMX management, 52–54
  centralized, 50
  complex, 52–53
  corporate, 53
  cross-platform, 49
  distributed, 50, 406
  e-business, 51
  high-volume, 53
  Internet, 50–51
  J2EE application component, 441–444
  J2EE application server, 438–441
  J2EEApplication object, 441–442
  management
    automation, 35–37
availability management, 38–39
configuration, 33–35
distribution, 29–31
event, 35–37
for hardware resources, 31–32
health status, 38–39
inventory, 31–32
monitoring, 38–39
operations, 35
performance, 38–39
for software resources, 31–32
topology, 32–33
as MBeans, 344–364
MBeans on behalf of, 364–392
mission-critical, 53
multiple management, 51–52
self-management, 415–418
service-oriented, 51
Sun ONE Application Server, 525–526
unattended upgrades and installations, 30
apply() method, 255, 258–260
architecture
  agent domains, 11
  agents, 10
cascaded, 11
communication model, 8
functional model, 8
hierarchical, 11
information model, 8
managed resources, 9–10
management system, 11–12
MLMs, 11
organizational model, 8
overview, 64–68
parts of, 8–9
RFCs, 10–11
standards bodies, 10
subagents, 10–11
arithmetic operations, 256
ARM (Application Response Management), 15–16
array aggregate type, 129
asterisk (*)
  in JMX object names, 435
  wildcard, 258, 259
asynchronous messaging support. See JMS.
attr() method
  AttributeValueExp class, 256
Query interface, 256
Attribute class, 123–124, 159
attribute descriptors
  caching, 168–169
defining a static attribute, 171–172
get and set requests, 170–171
ModelMBeanAttributeInfo descriptors, 170
  persistence, 168
protocol mapping, 171
AttributeChangeNotification class, 120–122. See also Notification class.
AttributeList class
  applications, as MBeans, 360–361
  applications, MBeans on behalf of, 379–382
connector client proxy, 320–321
  extending Container management interface, 113–114
  introspection algorithm, 95
listening for notifications, 277
MBean access methods, 251–252
monitoring queue requests, 115–119
monitoring todd, 81
overriding RequiredModelMBean, 215–221
  using GaugeMonitor, 286–287
AttributeNotFoundException
  applications, as MBeans, 359–361
  applications, MBeans on behalf of, 378–380
connector client proxy, 320–321
dynamic MBeans, 115–119
overriding RequiredModelMBean, 215–221
attributes
  Active
    definition, 265
    RelationService class, 313
    TimerMBean class, 297
AllNotificationIDs, 297
AllRelationIDs, 313
AllRelationTypeNames, 313
DeployedObjects, 440
DeploymentDescriptor, 441
DerivedGauge, 278–281, 283–284
DerivedGaugeTimeStamp, 273, 278–281, 283–284
DifferenceMode, 278–281, 283–284
ejbs, 444
Empty, 297
eventProvider, 435
GaugeMonitor
  DerivedGauge, 278–281
DerivedGaugeTimeStamp, 278–281
DifferenceMode, 278–281
Modulus, 278–281
Notify, 278–281
Offset, 278–281
Threshold, 279–281
getMBeansFromURL() method, 308
getResource() method, 308
getResourceAsStream() method, 308
HighThreshold, 283–284
impact, 112
ImplementationName, 254
ImplementationVendor, 254
ImplementationVersion, 254
JavaMailResource, 446
javaVendor, 445
javaVersion, 445
JavaVMs, 442
javaVMS, 440
JCAResource, 445–446
JDBCResource, 445–446
JMSResource, 445–446
JNDIResource, 446
JTAResource, 446
LibraryDirectory, 307–308
LowThreshold, 283–284
MBeanServerId, 254
MLetBean class, 307–308
Modules, 442
Modulus, 278–281
NbNotifications, 297
node, 445
Notify, 278–281
NotifyDiffer, 273
NotifyHigh, 283–284
NotifyLow, 283–284
objectName, 435
Offset, 278–281
PurgeFlag, 313
RelationService class, 313
resourceAdapters, 443
Resources, 440
RMI_IOOPResource, 446
SendPastNotifications, 297
Server, 439
servers, 438
ServerVendor, 440
ServerVersion, 440
attributes continued
  servlets, 443
  setting thresholds, 283–284
  SpecificationName, 254
  SpecificationVendor, 254
  SpecificationVersion, 254
  stateManageable, 435
  static, defining, 171–172
  statisticsProvider, 435
  StringToCompare, 273
  Threshold, 279–281
  TimerMBean class, 297
  URLResource, 446
  URLs, 307
  values, returning, 93
AttributeValueExp class, 255–258. See also
  ValueExp.
  auditing, J2EE, 462
  automation applications, 35–37
  availability management applications, 38–39

B
  base managed-object, J2EE, 434–438
  BEA, 58
  BEA Systems WebLogic Server 7.0, 524
  best practices
    application self-management, 415–418
    application-specific managers, 415–418
    federation options, 422–428
    local monitoring and recovery, 417–418
    logging, 420, 422
    MBean granularity, 411–415
    multi-agent-view topologies, 427–428
    notifications, 420, 422
    resource schema, 418–420
    single-agent-view topologies, 422–427
  between() method, 256
  boolean value, 265
  Borland, 58
  browser control, 82–83
  byte data type, 126

C
  caching, 168–169
  cascaded architecture, 11
  cascaded operations, 35
  Catalog class
    applications, as MBeans, 344–364
  applications, MBeans on behalf of, 384
  catalog application example, 412–413
  publish-only MBeans, 395–406
  CatalogManager class, 365–392
  CatalogManagerDD class, 365–392
  CatalogManagerMBean class, 365–368
  Category interface, 344–364
  centralized applications, 50
  char data type, 126
  CIM (Common Information Model), 10, 14
  CIM Object Manager (CIMOM), 14
  CIM/WBEM, 14, 39
  CIMOM (CIM Object Manager), 14
  Class descriptor, 174
classattr() method, 256
classes. See also
  exceptions; interfaces.
  ApacheMBean
    activating, 122–124
    Apache management interface, 100, 102–108
    creating and configuring monitors, 270–271
    description, 269–270
    listening for notifications, 271–273
    using GaugeMonitor, 286–287
  ApacheMonitor
    creating and configuring monitors, 270–271
    listening for notifications, 272–273
    MLet examples, 303–306
    using, 275–277
    using GaugeMonitor, 286–287
  ApacheServer, 139, 141–157
  ApacheServerManager, 138–140, 141–157
  AppClientModule, 442–444
  Attribute, 123–124, 159
  AttributeChangeNotification, 120–122. See also
  Notification class.
  AttributeList
    applications, as MBeans, 360–361
    applications, MBeans on behalf of, 379–382
    connector client proxy, 320–321
    extending Container management interface,
      113–114
    introspection algorithm, 95
    listening for notifications, 277
    MBean access methods, 251–252
    monitoring queue requests, 115–119
    monitoring todd, 81
    overriding RequiredModelMBean, 215–221
    using GaugeMonitor, 286–287
AttributeValueExp, 255–258. See also ValueExp.

Catalog applications, as MBeans, 344–364
applications, MBeans on behalf of, 384
catalog application example, 412–413
publish-only MBeans, 395–406
CatalogManager, 365–392
CatalogManagerDD, 365–392
CatalogManagerMBean, 65, 365, 370–371
clonedMBean, 243–244
Connection, 79, 193–210
ConnectionQueue, 115–119
Container, 113–114
CounterMonitor, 278–282. See also CounterMonitorMBean; Monitor class; NotificationBroadcasterSupport.
DefaultLoaderRepository, 246
DescriptorSupport. See also Descriptor interface.
adding custom descriptors, 214–215
Apache server example, 168, 170–172
applications, MBeans on behalf of, 389–392
attribute descriptors, 169–170
defining a static attribute, 171–172
implementing Descriptor interface, 165–167
model MBean example, 141–157
notification example, 178–179
publish-only MBeans, 403–406
EJBModule, 442–444
exercise, todd (time of day daemon), 70–71
ExtendedContainer, 113–114
GaugeMonitor. See also GaugeMonitorMBean;
Monitor class; NotificationBroadcasterSupport.
attributes, 283
creating relation types, 310–312
description, 282–285
instance of, 81–82
notification types, 284–285
using, 285–287
Item, 344–364
J2EEDeployedObject, 441
Listener, 75–76, 194–213
loading, eliminating, 66–67
MBeanAttributeInfo. See also MBeanFeatureInfo;
ModelMBeanAttributeInfo.
applications, as MBeans, 362–364
applications, MBeans on behalf of, 381–382
extending MBeanFeatureInfo, 110
in MBean Management interface, 93
model MBeanInfo example, 162
monitoring queue requests, 116–119
MBeanConstructorInfo. See also MBeanFeatureInfo; ModelMBeanConstructorInfo.
applications, as MBeans, 363–364
applications, MBeans on behalf of, 382
extending MBeanFeatureInfo, 110–111
in MBean Management interface, 93
MBeanInfo class, 109
model MBeanInfo example, 162
monitoring queue requests, 116–119
MBeanFeatureInfo, 93, 109–111. See also
MBeanAttributeInfo; MBeanConstructorInfo;
MBeanNotificationInfo; MBeanOperationInfo;
MBeanParameterInfo.
MBeanInfo. See also ModelMBeanInfoSupport.
applications, as MBeans, 362–364
applications, MBeans on behalf of, 365–392
code sample, 109
connector client proxy, 319–321
extending Container management interface, 113–114
MBean access methods, 251–252
monitoring queue requests, 116–119
XMLServiceMBean API, 190–192
MBeanNotificationInfo. See also MBeanFeatureInfo; ModelMBeanNotificationInfo.
applications, as MBeans, 363–364
applications, MBeans on behalf of, 382, 392
extending MBeanFeatureInfo, 111
in MBean Management interface, 93
MBeanInfo class, 109
model MBeanInfo example, 162
monitoring queue requests, 116–119
MBeanOperationInfo. See also MBeanFeatureInfo; ModelMBeanOperationInfo.
applications, as MBeans, 363–364
applications, MBeans on behalf of, 382
extending MBeanFeatureInfo, 111–112
in MBean Management interface, 93
MBeanInfo class, 109
model MBeanInfo example, 162
monitoring queue requests, 116–119
static model MBeans, 202–210
MBeanParameterInfo. See also MBeanFeatureInfo.
classes continued
  applications, as MBeans, 363–364
  applications, MBeans on behalf of, 382, 391–392
  extending MBeanFeatureInfo, 110–111
  in MBean Management interface, 93
  model MBean example, 147–157
  monitoring queue requests, 118
  OpenMBeanConstructorInfo interface, 132
  OpenMBeanOperationInfo interface, 133
  operation descriptor example, 175–176
  static model MBeans, 202–210
  MBeanServerDelegate, 233–254. See also NotificationBroadcaster.
  MBeanServerFactory
    description, 238–239
    finding an MBeanServer, 191–192
    incorporating an MBeanServer, 78–79
    listening for notifications, 123–124, 272–273, 276–277
    MLet examples, 304–306
    model MBean example, 138–139
    populating MBeanInfo, 208–210
    publish-only MBeans, 395–406
    registering MBeans, 248–249
    self-managing service example, 211–213
    static model MBeans, 198–210
    StockQuote service, 489–491
    using timer services, 290–292
    WebServiceManagerMBean implementation, 511–515
    WebServiceRuntimeManagerMBean implementation, 507–510
  MBeanServerNotification, 120–122. See also Notification class.
  MLet, 303–308. See also MBeanRegistration.
  ModelMBeanAttributeInfo. See also
    DescriptorAccess; MBeanAttributeInfo
    accessing default and protocolMap fields, 170–171
    applications, MBeans on behalf of, 389–392
    attribute descriptors, 169–170
    defining a static attribute, 171–172
    model MBean example, 141–157
    model MBeanInfo example, 162
    publish-only MBeans, 403–406
  ModelMBeanConstructorInfo. See also
    DescriptorAccess; MBeanConstructorInfo
    applications, MBeans on behalf of, 391–392
    constructor descriptor example, 173
  model MBean example, 147–157
  publish-only MBeans, 405–406
  static model MBeans, 202–210
  ModelMBeanInfo. See also ModelMBeanInfoSupport.
  applications, MBeans on behalf of, 388–392
  example, 161–162
  model MBean example, 141–157
  overriding RequiredModelMBean, 215–221
  populating, 208–210
  publish-only MBeans, 397–406
  self-managing service example, 210–213
  static model MBeans, 194–210
  ModelMBeanInfoSupport. See also MBeanInfo;
    ModelMBeanInfo
    applications, MBeans on behalf of, 392
    model MBean example, 151–157
    publish-only MBeans, 405–406
    static model MBeans, 202–210
  ModelMBeanNotificationInfo. See also
    DescriptorAccess; MBeanNotificationInfo
    model MBeanInfo example, 162
    notification example, 178–179
    publish-only MBeans, 405–406
    static model MBeans, 207–210
  ModelMBeanOperationInfo. See also
    DescriptorAccess; MBeanOperationInfo
    applications, MBeans on behalf of, 391–392
    model MBeanInfo example, 162
    operation descriptor example, 175–176
    publish-only MBeans, 405–406
    static model MBeans, 202–210
  Monitor, 264–265. See also CounterMonitor;
    GaugeMonitor; MBeanRegistration;
    MonitorMBean; NotificationBroadcasterSupport; StringMonitor.
  MonitorNotification. See also Notification class.
    CounterMonitor, 279
    field identifiers, 274
    GaugeMonitor, 284
    handleNotification() method, 80
    JMX notifications, 120–122
    using ApacheMonitor, 275–277
  Notification, 120–121. See also
    AttributeChangeNotification; MBeanServerNotification;
    MonitorNotification; RelationNotification; TimerNotification.
    activating Apache MBeans, 123
    JMX notifications, 120–122
JSR 77 sample code, 464
listening for notifications, 123–124
TimerNotification, 294–296
using timer services, 290–292
NotificationBroadcasterSupport, 120, 122–124. See also CounterMonitor;
GaugeMonitor; Monitor; NotificationBroadcaster; StringMonitor.
NotificationFilterSupport, 464. See also NotificationFilter.
ObjectName
connector client proxy, 319–321
constructors, 242
creating and configuring monitors, 270–271
description, 242–243
incorporating MBeanServer, 78–79
instantiating NotificationService, 310
JSR 77 sample code, 463–465
listening for notifications, 123–124
MBean access methods, 251–252
MBeanServer creation, 246–247
MBeanServer registration, 245
model MBean example, 139, 192
overriding RequiredModelMBean, 220–221
populating ModelMBeanInfo, 209–210
self-managing service example, 211–213
static model MBeans, 199–210
StockQuote service, 489–491
ObjectInstance
description, 243–244
MBean access methods, 251–252
MBeanServer creation, 246–247
MBeanServer registration, 245
model MBean example, 139, 192
overriding RequiredModelMBean, 220–221
populating ModelMBeanInfo, 209–210
self-managing service example, 211–213
static model MBeans, 199–210
StockQuote service, 489–491
WebServiceManagerMBean implementation, 511–515
WebServiceRuntimeManagerMBean implementation, 507–510
XMLServiceMBean API, 190–192
Ping, 496–497, 500
ProcessManagerMBean, 332–334
Query, 255
RelationNotification, 120–122. See also NotificationFilter.
RelationService. See also MBeanRegistration;
NotificationListener.
creating relation types, 310–312
external relation type, 313
external relations, 314
instantiating, 310
internal relation type, 313
internal relations, 314
methods, 314–316
operations, 313
query operations, 316
relation operations, 314–316
RelationSupport, 316. See also MBeanRegistration.
RequiredModelMBean, 215–221. See also ModelMBean.
ResourceAdapterModule, 442–444
RestartingService, 290–292
RestartingServiceMBean, 290
Role, 311
RoleInfo, 310–312, 314
RoleList, 311–312, 314–316
RoleResult, 315–316
scheduling behaviors, 298–299
SecureContainer, 113–114
Server
incorporating an MBeanServer, 78–79
self-managing service example, 210–213
ServerMBean interface, 75–76
static model MBeans, 199–210
Session, 77, 195–210
SessionPool
creating relation types, 310–312
self-managing service example, 211–213
ServerMBean interface, 75–76
static model MBeans, 194–210
SimpleSecureContainer, 113–114
SimpleSwitch, 247–249
SimpleSwitchMBean, 247–249
SimpleSecureContainer, 113–114
SimpleSwitch, 247–249
SimpleSwitchMBean, 247–249
classes continued

SimpleType, 126
spma, 332–335
StockQuoteManager, 488–491, 496–497
StockQuoteManagerMBean, 487–491
StockQuoteService, 494, 515–516
StringMonitor, 270–271, 275–277. See also
  Monitor class; NotificationBroadcaster-Support; StringMonitorMBean.
StringValueExp, 255–257. See also ValueExp.
TackCatalog, 344–392
TackCatalogDD, 355–356
TackCatalogMD, 373–378
TackCatalogMBean, 344–392
TackCatalogMM, 383–392
TackCatalogPO, 394–406
TackCatalogSS, 365–392
TaskItem, 344–364
Timer. See also MBeanRegistration; TimerMBean.
  creating relation types, 310–312
description, 296–300
  notification operations, 297–299
  using timer services, 290–292
timer notification operations, 297–299
TimerFilter, 295–296
TimerMBean attributes, 296–297
TimerMBeanNotification, 120–122
  294–296. See also Notification class.
UDDIRegistryClient, 478, 498–502, 503–505
UDDIRegistryClientManager, 503–505
UDDIRegistryClientManagerMBean, 478, 497–505
UDDIRegistryController, 475–477
UDDIRegistryData, 475–477
UDDIRegistryManagerMBean, 474–475
UDDIRegistryRuntimeManagerMBean, 476–477
WebModule, 442–444
WebServiceClientManager, 496–497
WebServiceClientManagerMBean, 495–497
WebServiceManager, 489–490, 511–512
WebServiceManagerMBean, 485, 511–515
WebServiceRuntimeManager, 507–510
WebServiceRuntimeManagerMBean, 481, 506–510
XMLModelMBean, 215–221
XMLServer, 220–221
XMLService, 192, 208–213
XMLServiceMBean, 190–192
c1oneMBean class, 243–244
CMIP (Common Management Information Protocol), 10, 14
colon (:), in JMX object names, 435
comma (,), in JMX object names, 435
Common Information Model (CIM), 10, 14
Common Management Information Protocol (CMIP), 10, 14
communication model, 8
Compiere Open Source ERP & CRM, 526
complex applications, 52–53
complex operations, 35
compliance
distributed services level, 63
instrumentation level, 62
JMX, 60–64
management level, 63
component model, 341–342
composite aggregate type, 127–128
CompositeData instance, 128
Computer Associates, 58
conceptual stack, 470–473
crcrete monitors. See also CounterMonitor;
  GaugeMonitor.
ApacheMonitor
  attributes, 273
  conditions, 273
  notification types, 274
  canonical usage pattern
  configuring the monitor, 270–271
  creating the monitor, 270–271
  listening for notifications, 271–273
  modes of operation
difference mode, 281
  raw mode, 280
StringMonitor, 273–277
configuration
  API-based, 19
dynamic reconfiguration, 19
  file-based reconfiguration, 19
J2EE management functions, 462
management applications, 33–35
resources, displaying, 33–35
static reconfiguration, 19
configuration applications, 33–35
configuration data, 22–23, 33–35
configuration points, 19
configure discipline, 19–20
configure stage, 17
configuring
  JMX security policies, 335–336
  model MBeans, 137–151
  monitors, 270–271
Connection class, 79, 195–210
ConnectionQueue class, 115–119
connector client proxy, 321–323
connectors
  connector client proxy, 318–321
  connector MBean, 318
description, 316–317
  implementation, 317
  standardization, 317
devaluating, 21, 22
deleting. See removing.
deploy discipline. See distributed discipline.
deploy discipline, 338–340
deployment patterns
  adapters, 340
  bootstrapping service, 342–343
definition, 337–338
deployment descriptor, 441
  deployment descriptor attribute, 441
deregistration, 249–251
Description interface. See also DescriptorSupport.
description layer, 471, 472
defining a static attribute, 171–172
definition, 337–338
  discovery, 340
  driver model, 342–343
  federating agents, 342
  JMX daemons vs. native daemons, 339
  remoteness, 340
  deployment descriptor attribute, 441
deregistration, 249–251
  DerivedGauge attribute, 278–281, 283–284
  DerivedGaugeTimeString attribute, 273, 278–281, 283–284
description layer, 471, 472
Creating
  external relations, 314
  MBeans, 246–249
  MBeanServer, 329
  model MBeans
    for Apache, 191
    from managed resources, 210–213
    XML to MBeanInfo service, 208–210
    ModelMBeanInfo instance, 189
  monitors, 270–271
cross-platform applications, 49
currencyTimeLimit descriptor, 167, 170, 175
D
daemon model, 338–340
data types. See also specific types.
  aggregate
    array, 129
    composite, 127–128
table, 128–129
  primitives, 126
  supported, 66–67
database access, 445
dB2 data storage option, 158
default descriptor, 170
defaultLoaderRepository class, 246
definition, 337–338
discovery, 340
driver model, 342–343
driving agents, 342
desktop model, 338–340
deploy discipline. See distributed discipline.
deployment patterns
  adapters, 340
  bootstrapping service, 342–343
definition, 337–338
deployment descriptor, 441
deregistration, 249–251
  DerivedGauge attribute, 278–281, 283–284
  DerivedGaugeTimeString attribute, 273, 278–281, 283–284
description layer, 471, 472
Description interface. See also DescriptorSupport.
    adding custom descriptors, 214–215
    Apache server example, 168, 170–172
    applications, MBeans on behalf of, 389–392
    attribute descriptors, 169–170
  constructor descriptor example, 173
defining a static attribute, 171–172
definition, 337–338
differentiation, 164–167
deployment descriptor, 441
deregistration, 249–251
  DerivedGauge attribute, 278–281, 283–284
  DerivedGaugeTimeString attribute, 273, 278–281, 283–284
  derivedGauge attribute, 276
  derivedGaugeTimeString attribute, 276
  deployment descriptor attribute, 441
deregistration, 249–251
  DerivedGauge attribute, 278–281, 283–284
  DerivedGaugeTimeString attribute, 273, 278–281, 283–284
description layer, 471, 472
Description interface. See also DescriptorSupport.
    adding custom descriptors, 214–215
    Apache server example, 168, 170–172
    applications, MBeans on behalf of, 389–392
    attribute descriptors, 169–170
  constructor descriptor example, 173
defining a static attribute, 171–172
definition, 337–338
Descriptor interface continued
  publish-only MBeans, 397–406
descriptor access interface. See also
  ModelMBeanAttributeInfo; ModelMBeanConstructorInfo; ModelMBeanNotificationInfo; ModelMBeanOperationInfo.
  constructor descriptor example, 173
description, 162
  purpose of, 164
  static model MBeans, 200–210
  descriptors. See also entries for specific descriptors.
  attribute
caching, 168–169
defining a static attribute, 171–172
get and set requests, 170–171
  ModelMBeanAttributeInfo descriptors, 170
  persistence, 168
  protocol mapping, 171
  constructor
default, 173
  public, 173
  signatures, 173
  model MBeans
  example for an Apache server, 168
  persistence policy, 168
  valid descriptors, 167
  notification
description, 176
  example, 178–179
  modelMBeanNotificationInfo descriptors, 177–178
  operation
Integer: start() method, 175–176
  modelMBeanConstructorInfo descriptors, 174
stop() method, 175–176
DescriptorSupport class. See also Descriptor interface.
adding custom descriptors, 214–215
  Apache server example, 168, 170–172
  applications, MBeans on behalf of, 389–392
  attribute descriptors, 169–170
defining a static attribute, 171–172
implementing Descriptor interface, 165–167
model MBean example, 141–157
notification example, 178–179
publish-only MBeans, 403–406
DescriptorType descriptor, 174
descriptorType descriptor, 167, 170, 174, 177
DynamicMBean interface. See also ModelMBean.

...applications, as MBeans, 355

extending Container management interface, 113–114

monitoring queue requests, 115–119

overriding RequiredModelMBean, 216–221

E

e-business applications, 51

e-mail support, 446

EAServer 4.0, 525

EJB (Enterprise Java Beans), 432

EJBModule class, 442–444

ejbs attribute, 444

EJBStats interface, 454–455

Empty attribute, 297

encapsulating metadata, 93–94

Enterprise Java Beans (EJB), 432

EntityBeansStats interface, 455

eq() method, 256

equal sign (=), in JMX object names, 435

ERP & CRM, 526

error messages, logging vs. notification, 420, 422.

See also notifications.

event applications, 35–37

event generation, J2EE, 449–451

Event Generator pattern, 26–27

eventProvider attribute, 435

events. See notifications.

exceptions. See also classes; interfaces.

ApacheMBeanException
activating Apache MBean, 123
ApacheMBean interface, 269–270
defining the Apache management interface, 100
model MBean example, 153–157

AttributeNotFoundException
applications, as MBeans, 359–361
applications, MBeans on behalf of, 378–380
connector client proxy, 320–321
dynamic MBeans, 115–119
overriding RequiredModelMBean, 215–221
during (de)registration, 249
InstanceNotFoundException, 215–221,
319–321

IntrospectionException, 319–321

InvalidAttributeValueException, 360–361,
379–380

InventoryUnderflowException
Catalog interface, 345
category application example, 413
DynamicMBean interface, 357
MBeans on behalf of applications, 367
publish-only MBeans, 397–398
TackCatalog class, 351, 385
TackCatalogDD class, 375

ItemExistsException
Catalog interface, 345
category application example, 413
DynamicMBean interface, 357
MBeans on behalf of applications, 367
publish-only MBeans, 396
TackCatalog class, 385
TackCatalogDD class, 375
TaskCatalog class, 350

MalformedObjectNameException, 100

MBeanException, 248. See also

MBeanRegistrationException.

applications, as MBeans, 359–361
applications, MBeans on behalf of, 379–380
DynamicMBean interface, 117
overriding RequiredModelMBean, 215–221
registering MBeans, 248–249

MBeanRegistrationException, 250. See also

MBeanException.

NoSuchCategoryException
Catalog interface, 345
category application example, 412–413
DynamicMBean interface, 358
MBeans on behalf of applications, 368–369
publish-only MBeans, 399
TackCatalog class, 352, 386
TackCatalogDD class, 376

NoSuchItemException
Catalog interface, 345
category application example, 413
DynamicMBean interface, 357
MBeans on behalf of applications, 369
publish-only MBeans, 399–401
TackCatalog class, 352–353, 386–387
TackCatalogDD class, 377

NotCompliantMBeanException, 94

ReflectionException, 248

applications, as MBeans, 359–362
applications, MBeans on behalf of, 379–381
connector client proxy, 319–320
overriding RequiredModelMBean, 215–221
registering MBeans, 248–249

RuntimeOperationException, 249
exceptions continued
- RuntimeOperationsException
  - overriding RequiredModelMBean, 215–221
  - registering MBeans, 249
- execute stage, 17
- executing an operation on specified MBean, 93
- execution environment owner, 480–482
- execution environment user, 482–483
- expectant customers, 53–54
- export descriptor, 167
- expressions, 253–258
- ExtendedContainer class, 113–114
- external relations, creating, 314

F
- facades for distributed applications, 406
- failure detection and recovery, 482
- federating agents, deployment patterns, 342
- federation options, best practices, 422–428
- file-based reconfiguration, 19
- finalSubString() method, 256–257
- findAssociateMBeans() method, 316
- finding
  - MBeans, 254–255
  - relationships of type, 316
- findMBeanServer() method, 329
- findReferencingRelations() method, 316
- findRelationshipOfType() method, 316
- float data type, 126
- functional description, 468–469
- functional model, 8

G
- GaugeMonitor
  - attributes, 283–284
  - constraints on attributes, 284
  - creating, 286–287
  - difference mode, 285
  - gauge definition, 282
  - HighThreshold attribute, 283–284
  - LowThreshold attribute, 283–284
  - notification types, 284–285
  - raw mode, 285
- setThresholds() method, 283–284
- setupRPSMonitor() method, 286–287
- using, 285–287

GaugeMonitor class. See also GaugeMonitorMBean;
  - Monitor class; NotificationBroadcaster-Support;
  - attributes, 283
  - creating relation types, 310–312
  - description, 282–285
  - instance of, 81–82
  - notification types, 284–285
  - using, 285–287
- GaugeMonitorMBean interface, 264, 282. See also
  - GaugeMonitor; MonitorMBean.
- generic management, 57
- geq() method, 256
- get requests, 168–172
- getActiveCount() method, 455
- getAdminClient() method, 321–322
- getAllRoles() method, 315
- getAttribute() method, 93, 251
- getAttributes() method, 93, 251
- getCloseCount() method, 458–459
- getCommittedCount() method, 455
- getConnectFactory() method, 457
- getConnectionPools() method, 458
- getConnections() method, 457–459
- getConnectPools() method, 457
- getConsumers() method, 460
- getCreateCount() method, 455, 458–459
- getDefaultDomain() method, 252
- getDestination() method, 460
- getDurableSubscriptionCount() method, 460
- getExpiredMessageCount() method, 460–461
- getFreePoolSize() method, 458–459
- getHeapSize() method, 454
- getJdbcDataSource() method, 458
- getManagedConnectFactory() method, 457
- getMBeanCount() method, 252
- getMBeanInfo() method, 93–94, 251, 331
- getMBeansFromURL() method, 308
- getMessageCount() method, 252
- getMessageCount() method, 460–461
- getMessageWaitTime() method, 460–461
- getMethod descriptor, 170
- getMethodReadyCount() method, 456
- getObjectName() method, 252
- getOrigin() method, 460
- getPassiveCount() method, 456
- getPendingMessageCount() method, 460–461
getPooledCount() method, 455
getPoolSize() method, 458–459
getProducers() method, 460
getReadyCount() method, 455
getReferencedMBeans() method, 316
getRelationTypeName() method, 315
getRemoveCount() method, 455
getResource() method, 308
getResourceAsStream() method, 308
getRoleCardinality() method, 316
getRoleInfo() method, 314
getRoles() method, 315
getRolesUnresolved() method, 316
getRolledbackCount() method, 455
getSentMailCount() method, 456
getServiceTime() method, 456
getSessions() method, 459
getStatisticNames() method, 454
getStatistics() method, 454
getUpTime() method, 454
getUseTime() method, 457–458
getWaitingThreadCount() method, 458–459
getWaitTime() method, 457–458
granularity, best practices, 411–415
GranularityPeriod, 265
group operations, 35
Groupe Bull (Evidian), 58
grow() method, 205–208
gt() method
 Query interface, 258
 QueryExp interface, 256
guidelines for design, 134

H
handleNotification() method, 79–82
handlers, 483
hardware resources, managing. See inventory applications.
hash tables, 127–128
hawkEye, 323
health status applications, 38–39
Hewlett Packard, 524–525
hierarchical architecture, 11
high-volume applications, 53
HighThreshold attribute, 283–284
homogenizing technologies, 12

I
IBM, 523–524
IBM/Tivoli, JMX Expert Group, 58
identification data, 22
IDL (Interface Definition Language), 164
IIOP, 59
impact attribute, 112
implementation name, getting, 254
implementation vendor name, getting, 254
implementation version, getting, 254
ImplementationName attribute, 254
ImplementationVendor attribute, 254
ImplementationVersion attribute, 254
in() method, 256
information model, 8
initialSubString() method, 256–257
InputStream instance, returning, 308
install discipline, 18–19
install stage, 17
installation, unattended, 30
InstanceNotFoundException, 215–221, 319–321
instantiater() method, 245–246
instantiation, 245–246
instrumentation level compliance, 62
instrumentation patterns
 applications as MBeans, 344–364
 facades for distributed applications, 406
 MBeans on behalf of applications, 364–392
 publish-only MBeans, 393–406
integer data type, 126
Integer start() method, 175–176
Interface Definition Language (IDL), 164
interfaces. See also classes; exceptions.
Category, 344–364
CounterMonitorMBean, 264. See also CounterMonitor; MonitorMBean.
Descriptor. See also DescriptorSupport. adding custom descriptors, 214–215
Apache server example, 168, 170–172
applications, MBeans on behalf of, 389–392
attribute descriptors, 169–170
constructor descriptor example, 173
defining a static attribute, 171–172
description, 164–167
model MBean example, 141–157
model MBeanInfo example, 161–162
interfaces continued

notification example, 178–179
overriding RequiredModelMBean, 215–221
publish-only MBeans, 397–406
static model MBeans, 200–210
DescriptorAccess. See also ModelMBean-
AttributeInfo; ModelMBeanConstructorInfo;
ModelMBeanNotificationInfo; ModelMBean-
OperationInfo.

congstructor descriptor example, 173
description, 162
static model MBeans, 200–210
DynamicMBean. See also ModelMBean,
applications, as MBeans, 355
extending Container management interface,
113–114
monitoring queue requests, 115–119
overriding RequiredModelMBean, 216–221
GaugeMonitorMBean, 264, 282. See also
GaugeMonitor; MonitorMBean.
MBeanRegistration, 249–251. See also MLet
class; Monitor; RelationService class;
RelationSupport class; Timer class.
MBeanServer
creating, 246–247
instantiating, 245
instantiating RelationService, 310
listening for notifications, 123–124, 272–273
MBeanRegistration interface, 249–251
methods, 252
MLet examples, 304–306
populating ModelMBeanInfo, 209–210
publish-only MBeans, 394–406
query methods, 260–261
registering MBeans, 245, 248–249
self-managing service example, 210–213
spma, 333
static model MBeans, 194–210
StockQuote service, 489–491
using ApacheMonitor, 276–277
using timer services, 290–292
WebServiceManagerMBean implementation,
511–515
WebServiceRuntimeManagerMBean implementa-
tion, 507–510
MLetBean, 307–308
MLetMBean, 307. See also MLet class.

ModelMBean. See also DynamicMBean;
ModelMBeanNotificationBroadcaster;
PersistentMBean; RequiredModelMBean,
aplications, MBeans on behalf of, 388–392
notification example, 178–179
publish-only MBeans, 394–406
self-managing service example, 211–213
XMLServiceMBean API, 190–192
ModelMBeanNotificationBroadcaster, 159. See
also ModelMBean; NotificationBroadcaster
interface.

MonitorMBean, 265–266. See also
CounterMonitorMBean; GaugeMonitorMBean;
Monitor; StringMonitorMBean.
NotificationBroadcaster, 119–120, 120–122,
159. See also MBeanServerDelegate;
ModelMBeanNotificationBroadcaster;
NotificationBroadcasterSupport.

NotificationFilter. See also
NotificationFilterSupport.
connector client proxy, 320–321
GaugeMonitor instance, 80
JMX notifications, 120–122, 261–262
TimerNotification, 294–296

NotificationListerner. See also
RelationService class.
active MBeans, 119
connector client proxy, 320–321
JMX notifications, 120–122
JSR 77 sample code, 464
listing for notifications, 123–124, 272–273
MLet examples, 303–306
monitoring todd, 80
notifications, 261–262
ServerMBean interface, 75–76
using ApacheMonitor, 275–277
using timer service, 290–292
using timer services, 290–292
PersistentMBean, 158. See also ModelMBean.
QueryExp
MBeanServer query methods, 260–261
methods, 255
pattern matching, 258–260
query expressions, 255–258
query scope, 258–260
RelationType, 314
ServerMBean, 75–76, 96
StringMonitorMBean, 264, 273. See also MonitorMBean, StringMonitor.
TimerMBean, 296–299. See also Timer class.
ValueExp, 255–257. See also AttributeValueExp; StringValueExp.
internal instrumentation, 24–25
International Standards Organization (ISO), 14
Internet applications, 50–51
IntrospectionException, 319–321
InvalidAttributeValueException, 360–361, 379–380
inventory applications, 31–32
InventoryUnderflowException
Catalog interface, 345
category application example, 413
DynamicMBean interface, 357
MBeans on behalf of applications, 367
publish-only MBeans, 397–398
TackCatalog class, 351, 385
TackCatalogDD class, 375
invoke() method, 93, 251
invokeOperation() method, 331
IONA, 58, 524
iPortal, 524
isInstanceOf() method, 252
ISO (International Standards Organization), 14
isolation, 56
isRegistered() method, 252
isTransactional() method, 459
Item class, 344–364
ItemExistsException
Catalog interface, 345
category application example, 413
DynamicMBean interface, 357
MBeans on behalf of applications, 367
publish-only MBeans, 396
TackCatalog class, 385
TackCatalogDD class, 375
TaskCatalog class, 350
Iterable descriptor, 175
iterable descriptor, 170
J
J2EE application component, 441–444
J2EE application server, 438–441
J2EE Connector Architecture (JCA)
performance measurement, 457–458
resource adapters, 443
resource manager, 443
J2EE (Java 2 Enterprise Edition)
application-specific extensions, 461
containers, 433
definition, 431
goals, 462–463
management, 432–434
management tool access, 434
MEJB, 434
specification, 431–432. See also JSR 77.
J2EE management functions
auditing, 462
configuration, 462
event generation, 449–451
load balancing, 462
performance monitoring, 451–453
security, 461–462
shortcomings, 461–462
state management, 447–449
Stats interface details, 454–461
J2EE management models. See also j2eeManagedObject model.
base managed-object, 434–438
domain part, 435
J2EE application component, 441–444
J2EE application server, 438–441
J2EE runtime components, 444–446
management domain, 438
properties part, 435
restricted characters, 435
J2EE runtime components, 444–446
J2EEApplication object, 441–442
J2EEDeployedObject class, 441
J2EEDeployedObject object, 439, 441–444
J2EEDomain object, 438
j2eeManagedObject model. See also J2EE management models.
: (colon), in JMX object names, 435
, (comma), in JMX object names, 435
* (asterisk), in JMX object names, 435
= (equal sign), in JMX object names, 435
? (question mark), in JMX object names, 435
eventProvider attribute, 435
j2eeType property, 436
name property, 436
objectName attribute, 435
j2eeManagedObject model continued
parent properties, 436
stateManageable attribute, 435
statisticsProvider attribute, 435
J2EEModule object, 442–444
J2EEResource model, 444–446
J2EEServer object, 438–441
j2eeType property, 436
Java Database Connectivity (JDBC), 432
Java Dynamic Management Kit (JDMK), 58
Java Management API (JMAPI), 57–58
Java Management Extensions (JMX). See JMX (Java Management Extensions).
Java Management Extensions (JMX) v1.0 Specification, 60
Java Message Service (JMS), 432, 445
Java Naming and Directory Interface (JNDI), 446
Java Specification Requests (JSRs), 14, 59–60
Java Transaction API (JTA), 446
JavaMail1Resource attribute, 446
JavaMail1Stats interface, 456
JavaServer Pages (JSP), 432
javaVendor attribute, 445
javaVersion attribute, 445
JavaVMs attribute, 442
javaVMS attribute, 440
JBoss, 58, 525
JCA (J2EE Connector architecture) performance measurement, 457–458
resource adapters, 443
resource manager, 443
JCAConnectionPool1Stats interface, 458
JCAConnectionStats interface, 457
JCAResource attribute, 445–446
JCASTats interface, 457
JDBC (Java Database Connectivity), 432
JDBCConnectionPool1Stats interface, 459
JDBCConnectionStats interface, 458
JDBCResource attribute, 445–446
JDBCSSStats interface, 458
JDMK 4.2, 520
JDMK (Java Dynamic Management Kit), 15, 58
JMAPI (Java Management API), 57–58
JMImplementation domain, 253–254
JMS (Java Message Service), 432, 445
JMSConnectionStats interface, 459
JMSConsumerStats interface, 460
JMSEndpointStats interface, 461
JMSProducerStats interface, 460
JMSResource attribute, 445–446
JMSConnectionStats interface, 460
JMSStats interface, 459
JMX 1.0 reference implementation, 519–520
JMX adapters. See adapters.
JMX agents. See agents.
JMX connectors. See connectors.
JMX daemons vs. native daemons, 339
JMX-enabled products
BEA Systems WebLogic Server 7.0, 524
Compiere Open Source ERP & CRM, 526
Hewlett Packard Core Services Framework, 525
Hewlett Packard OpenView, 524
IBM Web Services Tool Kit 3.1, 523
IBM WebSphere 5.0, 523
IBM WebSphere Business Components Composer, 524
IBM WebSphere Voice Server, 524
IONA iPortal, 524
IONA Technologies PLC: Orbix..., 524
JBoss, 525
Pramati Server, 525
Sonic Software Sonic XQ, 525
Sun Microsystems: Sun ONE Application Server, 525–526
Sun Microsystems: Sun ONE Portal Server, 526
Sybase EAServer 4.0, 525
Sybase EAServer 4.0, 525
Sun Microsystems: Sun ONE Portal Server, 526
Sybase EAServer 4.0, 525
JMX Expert Group, 58
JMX (Java Management Extensions). See also MBeans.
agent level compliance, 62
APIs, 55
application-specific management systems, 52
candidates for management, 52–54
centralized applications, 50
complex applications, 52–53
compliance, 60–64
components, 64–68
corporate applications, 53
cross-platform applications, 49
description, 14–15
distributed applications, 50
distributed services level compliance, 63
diversity, 50–51
dynamic management, 56
Index

- e-business applications, 51
- exercise, todd (time of day daemon)
  - browser control, 82–83
  - classes, 70–71
  - management, 71–73
  - MBeans, 74–78
  - MBeanServer, 78–79
  - monitoring, 79–82
- expectant customers, 53–54
- generic management, 57
- goals of, 54–57
- high-volume applications, 53
- history of, 57–64
- instrumentation level compliance, 62
- Internet applications, 50–51
- isolation, 56
- JDMK, 58
- JMAPI, 57–58
- JSRs, 59–60
- mailing list, 58
- management container, 56
- management level compliance, 63
- manager-level API, 61–62
- MBeanServer, 56
- mediation, 56
- mediation layer, 56
- mission-critical applications, 53
- multiple management applications, 51–52
- need for, 48–52
- overview, 64–68
- reference implementation, 63–64
- service-oriented applications, 51
- simple API, 56
- specification, 60–64
- Web site, 58

**JMX (Java Management Extensions) v1.0 Specification**, 60

JMX-managed resources. See resources.

JMX managers. See managers.

JMX monitor service. See monitoring service.

JMX notification model. See notification model.

JMX permissions. See permissions.

- jmx.monitor.error.attribute notification, 268
- jmx.monitor.error.mbean notification, 268
- jmx.monitor.error.runtime notification, 268
- jmx.monitor.error.threshold notification, 268
- jmx.monitor.error.type notification, 268
- jmx.monitor.gauge.high notification, 284–285
- jmx.monitor.gauge.low notification, 284–285
- jmx.monitor.string.differs notification, 274
- jmx.monitor.string.matches notification, 274

JMXperience Web site, 63

JNDI (Java Naming and Directory Interface), 446

- JNDIResource attribute, 446

JSP (JavaServer Pages), 432

JSR 77. See also J2EE.

  - application-specific extensions, 461
  - description, 433

J2EEServer object, 440–441

JMX requirements, 433–434

management functions

  - auditing, 462
  - configuration, 462
  - event generation, 449–451
  - load balancing, 462
  - performance monitoring, 451–453
  - security, 461–462
  - shortcomings, 461–462
  - state management, 447–449
  - Stats interface details, 454–461

management models. See also j2eeManagedObject model.

  - base managed-object, 434–438
  - domain part, 435

J2EE application component, 441–444

J2EE application server, 438–441

J2EE runtime components, 444–446

management domain, 438

  - properties part, 435
  - restricted characters, 435

  - object hierarchy, 438–439

  - property requirements, 435–438

  - sample code, 463–465

JSR adapters, 59–60

JSRs (Java Specification Requests), 59–60

  - manager-level API, 61

  - pertaining to JMX, 59

WBEM Services Specification, 14

JTA (Java Transaction API), 446

  - JTAResource attribute, 446

JTATstats interface, 455

JVM object, 444–445

JVMStats interface, 454

K

  - key properties, 241–242
L

lastReturnedValue descriptor, 167, 170, 175
lastUpdatedTimeStamp descriptor, 167, 170, 175
layers
description, 471, 472
discovery, 472, 473
mediation, 56
network, 470, 472
publication, 471–472, 473
service flow, 472
work flow, 473
XML-based messaging, 471
LDAP data storage option, 158
legalValues descriptor, 170
leq() method, 256
LibraryDirectory attribute, 307–308
lifecycle
gest practices
application self-management, 415–418
application-specific managers, 415–418
federation options, 422–428
local monitoring and recovery, 417–418
logging, 420, 422
MBean granularity, 411–415
multi-agent-view topologies, 427–428
notifications, 420, 422
resource schema, 418–420
single-agent-view topologies, 422–427
execution environment, 481–482
management systems, 16–17
registering MBeans by
application, 407–409
management adapter, 410–411
MBeanServer bootstrap, 411
third party, 409–410
resources, 16–17
Listener class, 75–76, 194–213
listening for notifications, 271–273
load balancing, J2EE, 462
load() method, 157–158
loading an .mlet file, 308
log descriptor, 167, 177
logFile descriptor, 167, 178
logging, best practices, 420, 422
logical methods, 256
long data type, 126
long value
milliseconds between observation, 265
time between observations, 265
Long value indicating time between observations, 265
looking up one or all of MBSevers, 329
LowThreshold attribute, 283–284
lt() method, 256
Lutris, 58

M

mailing list, 58
maintain stage, 17
MalformedObjectNameException, 100
Managed Object Format (MOF), 10, 164
managed resources. See also MBeans.
candidates for, 52–54
configuration data, 22–23
definition, 9–10
discovering, 32–33
displaying configuration, 33–35
Event Generator pattern, 26–27
external instrumentation, 24–25
identification data, 22
internal instrumentation, 24–25
lifecycle, 16–17
management data, 22–23
management instrumentation, 24–26
management operations, 23
management patterns, 26–29
ModelMBeanInfo, 161–164
Noninterruptible pattern, 27
Operational pattern, 28–29
Queryable pattern, 27–28
read-only management, 27–28
responsibilities, 22–26
statistical data, 23
status data, 23
unavailable, definition, 16–17
ManageEngine, 521
management
exercise, todd (time of day daemon), 71–73
instrumentation, 24–26
J2EE, 432–434
management applet. See MLet.
management applications
automation, 35–37
availability management, 38–39
configuration, 33–35
distribution, 29–31
event, 35–37
for hardware resources, 31–32
health status, 38–39
inventory, 31–32
monitoring, 38–39
operations, 35
performance, 38–39
for software resources, 31–32
topology, 32–33
unattended upgrades and installations, 30
Management Architecture, 13
management container, 36
management data
collecting. See monitoring.
description, 22–23
management disciplines
configure, 19–20
control, 21–22
distribute, 17–18
install, 18–19
monitor, 20–21
management domain, J2EE, 438
Management EJB (MEJB), 434
Management Information Bases (MIBs), 10, 13
management-level compliance, 63
management operations, 23
management patterns
Event Generator, 26–27
Noninterruptible, 27
Operational, 28–29
Queryable, 27–28
management systems. See also architecture; technologies.
definition, 3–4
history of, 5–8
lifecycle, 16–17
need for, 4–5
management technologies. See technologies.
management tool access, J2EE, 434
manager-level API, 61–62
managers, 522–523
Map object, 316
mapping MBeans to Java classes, 243–244
match() method, 258
matchSubString() method, 257
mathematical operations, 256
MBean access methods, 251–252
MBean lifecycle methods
createMBean(), 247
creation, 246–249
deregistration, 249–251
initialize(), 245–246
initialization, 245–246
MBeanRegistration interface, 249–251
registerMBean(), 246
registration, 246
MBean name of observed object, 265
MBeanAttributeInfo class. See also MBeanFeatureInfo; ModelMBeanAttributeInfo.
applications, as MBeans, 362–364
applications, MBeans on behalf of, 381–382
extending MBeanFeatureInfo, 110
in MBean Management interface, 93
model MBeanInfo example, 162
monitoring queue requests, 116–119
MBeanConstructorInfo class. See also MBeanFeatureInfo; ModelMBeanConstructorInfo.
applications, as MBeans, 363–364
applications, MBeans on behalf of, 382
extending MBeanFeatureInfo, 110–111
in MBean Management interface, 93
MBeanInfo class, 109
model MBeanInfo example, 162
monitoring queue requests, 116–119
MBeanException. See also
MBeanRegistrationException.
aplications, as MBeans, 359–361
applications, MBeans on behalf of, 379–380
DynamicMBean interface, 117
overriding RequiredModelMBean, 215–221
registering MBeans, 248–249
MBeanFeatureInfo class, 93, 109–111. See also
MBeanAttributeInfo; MBeanConstructorInfo;
MBeanNotificationInfo; MBeanOperationInfo; MBeanParameterInfo.
MBeanInfo class. See also ModelMBeanInfoSupport.
aplications, as MBeans, 362–364
applications, MBeans on behalf of, 365–392
code sample, 109
connector client proxy, 319–321
extending Container management interface,
113–114
MBean access methods, 251–252
monitoring queue requests, 116–119
XMLServiceMBean API, 190–192
MBeanInfo classes
MBeanAttributeInfo, 110
MBeanConstructorInfo, 110–111
MBeanNotificationInfo, 111
MBeanInfo classes continued
MBeanOperationInfo, 111–112
MBeanParameterInfo, 111
MBeanInfo structure, 108–112
MBeanNotificationInfo class, 111. See also MBean-FeatureInfo; ModelMBeanNotificationInfo.
applications, as MBeans, 363–364
applications, MBeans on behalf of, 382, 392
extending MBeanFeatureInfo, 111
in MBean Management interface, 93
MBeanInfo class, 109
model MBeanInfo example, 162
monitoring queue requests, 116–119
MBeanOperationInfo class, 111–112. See also MBeanFeatureInfo; ModelMBeanOperationInfo.
applications, as MBeans, 363–364
applications, MBeans on behalf of, 382
extending MBeanFeatureInfo, 111–112
in MBean Management interface, 93
MBeanInfo class, 109
model MBeanInfo example, 162
monitoring queue requests, 116–119
static model MBeans, 202–210
MBeanParameterInfo class, 111. See also MBeanFeatureInfo.
applications, as MBeans, 363–364
applications, MBeans on behalf of, 382
extending MBeanFeatureInfo, 110–111
in MBean Management interface, 93
model MBean example, 147–157
monitoring queue requests, 118
OpenMBeanConstructorInfo interface, 132
OpenMBeanOperationInfo interface, 133
operation descriptor example, 175–176
static model MBeans, 202–210
MBeanPermission, 331
MBeanRegistration interface, 249–251. See also MLet class; Monitor; RelationService class;
RelationSupport class; Timer class.
MBeanRegistrationException, 250. See also MBeanException.
MBeans. See also JMX; managed resources.
active
Apache MBean, 122–124
JMX notification model
Notification class, 120–121
NotificationBroadcaster, 120
NotificationFilter interface, 120
NotificationListener interface, 122
MBeans. See also JMX; managed resources.
JMX notifications, 120–122
listeners, 119–120
NotificationBroadcaster interface, 119
on behalf of applications, 364–392
construction of
active MBeans, 119–124
dynamic MBeans, 108–119
open MBeans, 124
standard MBeans, 96–108
creating, 246–249, 331
data types supported, 66–67
defining at development time, 65–66
defining at runtime, 66, 67–68
design guidelines, 133
dynamic, 66
exercise, todd (time of day daemon), 74–78
finding, 254–255
fundamentals, 91–95
identifying within a domain, 241–242
key properties, 241–242
mapping to Java classes, 243–244
model, 67–68
models. See model MBeans.
naming, 239–241
open, 66–67
passive, 96–119
propagation via proxies, 423–424
publish-only, 393–406
registered, identifying, 242–243
request propagation via broadcast, 424–426
request propagation via directory, 426–427
standard, 65–66
MBeanServer. See also patterns.
creating, 329
default domain, 238–239
definition, 56
description, 68
exercise, todd (time of day daemon), 78–79
finding MBeans, 254–255
getting information about, 252, 253–254
implementation name, getting, 254
implementation vendor name, getting, 254
implementation version, getting, 254
instantiating, 238–239
master vs. subordinate, 422–427
object names
domains, 240–247
key properties, 241–242
mapping to Java classes, 243–244
naming, 239–240
ObjectInstance class, 243–244
ObjectName class, 242–243
pattern matching, 258–260
server ID, getting, 254
specification name, getting, 254
specification vendor name, getting, 254
specification version, getting, 254
specifying patterns, 242–243
MBeanServer interface
creating, 246–247
instantiating, 245
instantiating RelationService, 310
listening for notifications, 123–124, 272–273
MBeanRegistration interface, 249–251
methods, 252
MLet examples, 304–306
populating MBeanInfo, 209–210
publish-only MBeans, 395–406
query methods, 260–261
registering MBeans, 245, 248–249
self-managing service example, 210–213
spma, 333
static model MBeans, 194–210
StockQuote service, 489–491
using ApacheMonitor, 276–277
using timer services, 290–292
WebServiceManagerMBean implementation, 511–515
WebServiceRuntimeManagerMBean implementation, 507–510
MBeanServerId attribute, 254
MBeanServerNotification class, 120–122. See also Notification class.
MBeanServerPermission permission, 328–330
MBeanTrustPermission permission, 328–332
mediation layer, 56
MEJB (Management EJB), 434
MessageDriven EJBs, 443
MessageDrivenBeanStats interface, 455
messageId descriptor, 177
messages. See also notifications.
logging vs. notification, 420, 422
methods
addListener(), 331
addNotification(), 297
addNotificationListener(), 79–82, 261–262
addRelation(), 314
AdminClientFactory(), 321–322
and(), 256
anySubString(), 256–257
apply(), 255, 258–260
attr() AttributeValueExp class, 256
AttributeValueExp interface, 256
Query interface, 258
between(), 256
classattr(), 256
ContainsKey(), 128
ContainsValue(), 128
createMBean(), 247
createMBeanServer(), 329
createRelation(), 314
createRelationType(), 310–312
div(), 256
eq(), 256
finalSubString(), 256–257

MLet examples, 304–306
model MBean example, 138–139
populating MBeanInfo, 208–210
publish-only MBeans, 395–406
registering MBeans, 248–249
self-managing service example, 211–213
static model MBeans, 198–210
StockQuote service, 489–491
using timer services, 290–292
WebServiceManagerMBean implementation, 511–515
WebServiceRuntimeManagerMBean implementation, 507–510
MBeanServerId attribute, 254
MBeanServerNotification class, 120–122. See also Notification class.
MBeanServerPermission permission, 328–330
MBeanTrustPermission permission, 328–332
mediation layer, 56
MEJB (Management EJB), 434
MessageDriven EJBs, 443
MessageDrivenBeanStats interface, 455
messageId descriptor, 177
messages. See also notifications.
logging vs. notification, 420, 422
methods
addListener(), 331
addNotification(), 297
addNotificationListener(), 79–82, 261–262
addRelation(), 314
AdminClientFactory(), 321–322
and(), 256
anySubString(), 256–257
apply(), 255, 258–260
attr() AttributeValueExp class, 256
AttributeValueExp interface, 256
Query interface, 258
between(), 256
classattr(), 256
ContainsKey(), 128
ContainsValue(), 128
createMBean(), 247
createMBeanServer(), 329
createRelation(), 314
createRelationType(), 310–312
div(), 256
eq(), 256
finalSubString(), 256–257
methods continued
- findAssociateMBeans(), 316
- findMBeanServer(), 329
- findReferencingRelations(), 316
- findRelationshipOfType(), 316
- geq(), 256
- getActiveCount(), 455
- getAdminClient(), 321–322
- getAllRoles(), 315
- getAttribute(), 93, 251
- getAttributes(), 93, 251
- getCloseCount(), 458–459
- getCommittedCount(), 455
- getConnectFactory(), 457
- getConnectionPools(), 458
- getConnections(), 457–459
- getConnectionPools(), 457
- getConsumers(), 460
- getCreateCount(), 455, 458–459
- getDefaultDomain(), 252
- getDestination(), 460
- getDurableSubscriptionCount(), 460
- getExpiredMessageCount(), 458–459
- getFreePoolSize(), 458–459
- getHeapSize(), 454
- getJdbcDataSource(), 458
- getManagedConnectionFactory(), 457
- getMBeanCount(), 252
- getMBeanInfo(), 251, 331
- getMBeansFromURL(), 308
- getMessageCount(), 455
- getMessageCount(), 460–461
- getMessageWaitTime(), 460–461
- getMethodReadyCount(), 456
- getObjectId(), 252
- getOrigin(), 460
- getPassiveCount(), 456
- getPendingMessageCount(), 460–461
- getPooledCount(), 455
- getPoolSize(), 458–459
- getProducers(), 460
- getReadyCount(), 455
- getReferencedMBeans(), 316
- getRelationTypeName(), 315
- getRemoveCount(), 455
- getResource(), 308
- getResourceAsStream(), 308
- getRoleCardinality(), 316
- getRoleInfo(), 314
- getRoles(), 315
- getRolesUnresolved(), 316
- getRolledbackCount(), 455
- getSentMailCount(), 456
- getServiceTime(), 456
- getSessions(), 459
- getStatisticNames(), 454
- getStatistics(), 454
- getUptime(), 454
- getUseTime(), 457–458
- getWaitingThreadCount(), 458–459
- getWaitTime(), 457–458
- grow(), 205–208
- gt()  
  Query interface, 258
  QueryExp interface, 256
- handleNotification(), 79–82
- in(), 256
- initialSubString(), 256–257
- instantiate(), 245–246
- Integer start(), 175–176
- invoke(), 93, 251
- invokeOperation(), 331
- isInstanceOf(), 252
- isRegistered(), 252
- isTransactional(), 459
- leg(), 256
- load(), 157–158
- lt(), 256
- match(), 258
- matchSubString(), 257
- MBeanInfo getMBeanInfo(), 93–94
- minus(), 256
- newMBean(), 331
- newMBeanServer(), 329
- not(), 256
- or(), 256
- plus(), 256
- postDeregister(), 249–251
- postRegister(), 249–251
- preDeregister(), 249–251
- preRegister(), 249–251
- queryMBeans(), 260–261
- queryNames(), 260–261
- registerMBean(), 246, 331
- removeMBeanServer(), 329
- removeListener(), 331
removeNotification(), 297
removeNotificationListener(), 261–262
removeRelation(), 315
setAttribute(), 251, 331
setAttributes(), 93, 251
setManagedResource(), 159–161
setMBeanServer(), 255
setModelMBeanInfo(), 159
setRoles(), 315
setThresholds(), 283–284
setupRPSMonitor(), 286–287
shutdown(), 200
start(), 200, 265
store(), 157–158

value(), 130

void setAttribute(), 93

MIBs (Management Information Bases), 10, 13
Middleware Manager, 522
midlevel managers (MLMs), 11
minus() method, 256
mission-critical applications, 53
MLet API, 306–308
MLet class, 303–308. See also MBeanRegistration.
MLET file, 302–304
.mlet files
loading, 308
reading, 308
registering, 308
MLet service, 69
MLET tags, 300–302
MLetMBean interface, 307–308. See also MLet class.
MLMs (midlevel managers), 11
model MBean descriptors
example for an Apache server, 168
persistence policy, 168
valid descriptors, 167
model MBeans. See also dynamic MBeans; open MBeans.
adding custom descriptors, 214–215
behavior, 179–189
caching, 179–182
delegation, 182–184
logging, 186
persistence, 184–185
presentation descriptor field, 186–187
presentation string, 187–189
protocolMap descriptor, 188–189
values and validation, 182
visibility descriptor, 187–188
caveats, 222
class loader problems, 221
common mistakes, 221
common services, 136
compatibility, 136–137
compatible Java editions, 136–137
creating
for Apache, 191
by external force, 214
from managed resources, 210–213
XML to MBeanInfo service, 208–210
description, 67–68
dynamic model MBeans, 208–214
ease of support, 136
ease of use, 136
effects of dynamic model MBeans, 208–214
equivalent of model MBeans, 208–214
examples
associating with ApacheServer, 137–141
code sample, 137–157
configure model MBean, 141–151
creating ModelMBeanInfo instance, 141–151
exposing attributes, 137–157
exposing operations, 137–157
find MBeanServer, 137–141
getting data for ApacheServer, 137–141
instantiate ApacheServer, 137–141
instantiating model MBeans, 137–141
notification to MBean server, 137–141
notification to registered listeners, 137–141
sending notification upon error condition, 137–157
getting the attribute type wrong, 221
implementing time of day (todd) server example, 194–221
isolation, 136–137
overriding the RequiredModelMBean class, 215–221
reasons for, 136–137
remoteness, 222
static model MBeans, 194–208
styles, 194
transactionality, 222
trying to set descriptors used by model MBean code, 221
ModelMBean interface. See also DynamicMBean; ModelMBeanNotificationBroadcaster; PersistentMBean; RequiredModelMBean. applications, MBeans on behalf of, 388–392
dynamicMBean, 157
elements
configure model MBean, 141–151
create a ModelMBeanInfo instance, 141–151
ModelMBeanNotificationBroadcaster interface, 158–159
notification example, 178–179
persistentMBean interface, 158
publish-only MBeans, 394–406
self-managing service example, 211–213
XMLServiceMBean API, 190–192
ModelMBeanAttributeInfo class. See also
DescriptorAccess; MBeanAttributeInfo. accessing default and protocolMap fields, 170–171
applications, MBeans on behalf of, 389–392
attribute descriptors, 169–170
defining a static attribute, 171–172
model MBean example, 141–157
model MBeanInfo example, 162
publish-only MBeans, 403–406
ModelMBeanAttributeInfo descriptors, 170
ModelMBeanConstructorInfo class. See also
DescriptorAccess; MBeanConstructorInfo. applications, MBeans on behalf of, 391–392
constructor descriptor example, 173
model MBean example, 147–157
publish-only MBeans, 405–406
static model MBeans, 202–210
ModelMBeanInfo
creating, 189
DescriptorAccess interface, 164
setting, 208
ModelMBeanInfo class. See also
ModelMBeanInfoSupport.
applications, MBeans on behalf of, 388–392
example, 161–162
model MBean example, 141–157
overriding RequiredModelMBean, 215–221
populating, 208–210
publish-only MBeans, 397–406
self-managing service example, 210–213
static model MBeans, 194–210
ModelMBeanInfoSupport class. See also MBeanInfo;
ModelMBeanInfo.
MonitorNotification class. See also Notification class.
CounterMonitor, 279
field identifiers, 274
GaugeMonitor, 284
handleNotification() method, 80
JMX notifications, 120–122
using ApacheMonitor, 275–277
monitors. See concrete monitors.
Motorola, 58
multi-agent-view topologies, 427–428
multiple management applications, 51–52
MXJ4, 521–522

N
name descriptor, 167, 170, 174–175, 177
name property, 436
naming MBeans, 239–241
NbNotifications attribute, 297
Netview, 13
network layer, 470, 472
network management technologies, 12, 13–14
networks, history of management systems, 7
newMBean() method, 331
newMBeanServer() method, 329
node attribute, 445
none, scheduling behavior, 298
nonfunctional description, 469
Noninterruptible pattern, 27
nonrelational database access, 445
NoSuchCategoryException
Catalog interface, 345
category application example, 412–413
DynamicMBean interface, 358
MBeans on behalf of applications, 368–369
publish-only MBeans, 399
TackCatalog class, 352, 386
TackCatalogDD class, 376
NoSuchItemException
Catalog interface, 345
category application example, 413
DynamicMBean interface, 358–359
MBeans on behalf of applications, 369
publish-only MBeans, 399–401
TackCatalog class, 352–353, 386–387
TackCatalogDD class, 377
not() method, 256
NotCompliantMBeanException, 94
notification bus, 237
Notification class. See also AttributeChangeNotification;
MBeanServerNotification;
MonitorNotification; RelationNotification;
TimerNotification.
activating Apache MBeans, 123
JMX notifications, 120–122
JSR 77 sample code, 464
listening for notifications, 123–124
TimerNotification, 294–296
using timer services, 290–292
notification descriptors
description, 176
eample, 178–179
modelMBeanNotificationInfo descriptors, 177–178
notification listeners, adding/removing, 331
notification model, 120–122
NotificationBroadcaster interface, 119–120, 120–122, 159. See also MBeanServerDelegate;
ModelMBeanNotificationBroadcaster;
NotificationBroadcasterSupport.
NotificationBroadcasterSupport class, 120, 122–124. See also CounterMonitor;
GaugeMonitor; Monitor; NotificationBroadcaster; StringMonitor.
NotificationFilter interface. See also NotificationFilterSupport.
connector client proxy, 320–321
GaugeMonitor instance, 80
JMX notifications, 120–122, 261–262
TimerNotification, 294–296
NotificationFilterSupport class, 464. See also NotificationFilter.
NotificationListener interface. See also RelationService class.
active MBeans, 119
connector client proxy, 320–321
JMX notifications, 120–122
JSR 77 sample code, 464
listening for notifications, 123–124, 272–273
MBean examples, 303–306
monitoring todd, 80
notifications, 261–262
ServerMBean interface, 75–76
using ApacheMonitor, 275–277
using timer service, 290–292
using timer services, 290–292
notifications
attribute value below low threshold, 284–285
attribute value exceeds high threshold, 284–285
best practices, 420, 422
definition, 79–82
description, 261–262
generating, 26–28
issuing, 253–254
misconfigured monitor, 268
object instance of attribute value is null, 268
observed attribute does not exist, 268
observed MBeans are not registered, 268
throwing an exception, 268
triggers, 36
Notify attribute, 278–281
NotifyDiffer attribute, 273
NotifyHigh attribute, 283–284
NotifyLow attribute, 283–284
NSF data storage option, 158
null data type, 126
O
object names
domains, 240–241
key properties, 241–242
mapping to Java classes, 243–244
of MBean attributes, 265
naming, 239–240
ObjectInstance class
description, 243–244
MBean access methods, 251–252
MBeanServer creation, 246–247
MBeanServer registration, 245
model MBean example, 139, 192
overriding RequiredModelMBean, 220–221
populating ModelMBeanInfo, 209–210
self-managing service example, 211–213
static model MBeans, 199–210
StockQuote service, 489–491
ObjectName aggregate type, 129
ObjectName attribute, 435
ObjectName class
connector client proxy, 319–321
constructors, 242
creating and configuring monitors, 270–271
description, 242–243
incorporating MBeanServer, 78–79
instantiating RelationService, 310
JSR 77 sample code, 463–465
listening for notifications, 123–124
MBean access methods, 251–252
MBeanRegistration interface, 249–251
MBeanServer instantiation, 245
MBeanServer query methods, 260–261
MLet examples, 304–306
model MBean example, 139, 141–157
monitoring todd, 80–82
overriding RequiredModelMBean, 220–221
populating MBeanInfo, 208–210
publish-only MBeans, 394–406
query expressions, 257–258
registering MBeans, 248–249
self-managing service example, 211–213
spma, 333
static model MBeans, 198–210
StockQuote service, 489–491
using ApacheMonitor, 276–277
using GaugeMonitor, 286–287
using timer services, 290–292
WebServiceManagerMBean implementation, 511–515
WebServiceRuntimeManagerMBean implementation, 507–510
XMLServiceMBean API, 190–192
ObservedAttribute, 265
ObservedObject, 265
Offset attribute, 278–281
one-shot notification, 298
open MBeans. See also dynamic MBeans; model MBeans.
basic data types
aggregate types, 126–129
ObjectName, 129
OpenType class, 125
primitive objects, 126
primitives, 126
definition, 124–125
description, 66–67
facilitating functionality
decomposability into Java primitives, 133
extended metadata, 133–134
restricted data types, 133–134
OpenMBeanInfo, 129–133
using, 133–134
OpenMBeanAttributeInfo interface, 131–132
OpenMBeanConstructorInfo interface, 132–133
OpenMBeanInfo interface, 130–131
OpenMBeanOperationInfo interface, 133
OpenMBeanParameterInfo interface, 131
OpenView, 13, 524
operate stage, 17
operation descriptors
  Integer start() method, 175–176
  modelMBeanConstructorInfo descriptors, 174
  stop() method, 175–176
Operational pattern, 28–29
operations, 35
operations applications, 35
or() method, 256
organizational model, 8
OSI/TMN, 14
P
parent properties, 436
pattern matching, 258–260
patterns
deployment
  adapters, 340
  bootstrap service, 342–343
  component model, 341–342
daemon model, 338–340
definition, 337–338
discovery, 340
driver model, 342–343
federating agents, 342
JMX daemons vs. native daemons, 339
remoteness, 340
instrumentation
  applications as MBeans, 344–364
  facades for distributed applications, 406
  MBeans on behalf of applications, 364–392
  publish-only MBeans, 393–406
management
  Event Generator, 26–27
  Noninterruptible, 27
  Operational, 28–29
  Queryable, 27–28
  specifying, 242–243
Pegasus, 14
performance
capturing time metrics, 15–16
management applications, 38–39
measurement, 457–458
monitoring, J2EE, 451–453
period, scheduling behavior, 298
permissions
  createMBeanServer() method, 329
findMBeanServer() method, 329
MBeanPermission, 329–332
newMBeanServer() method, 329
releaseMBeanServer() method, 329
persistence, 168
PersistentMBean interface, 157–158. See also
  ModelMBean.
persistLocation descriptor, 167
persistName descriptor, 167
persistPeriod descriptor, 167, 170
persistPolicy descriptor, 167, 170
Ping class, 496–497, 500
plus() method, 256
port, publishing, 491–495
portType, publishing, 491–495
postDeregister() method, 249–251
postRegister() method, 249–251
Powerware, 58
Pramati Server, 525
Pramati Technologies, 525
preDeregister() method, 249–251
preRegister() method, 249–251
presentation descriptor, 186–187
presentationString descriptor, 167, 170,
  174–175, 177
ProcessManagerMBean class, 332–334
properties part, J2EE, 435
proprietary technologies, 13
protocol map, 188
protocol mapping, 171
protocolMap descriptor, 170, 188–189
public constructor descriptors, 173
publication layer, 471–472, 473
publish-only MBeans, 393–406
publishing portType and port, 491–495
PurgeFlag attribute, 313
Q
queries. See MBeanServer queries.
Query class, 255
query service, 255
Queryable pattern, 27–28
QueryExp interface
  MBeanServer query methods, 260–261
  methods, 255
  pattern matching, 258–260
  query expressions, 255–258
  query scope, 258–260
queryMBeans() method, 260–261
queryNames() method, 260–261
question mark (?)  in JMX object names, 435
wildcard, 258, 259

R
read-only management, 27–28, 48
reading an .mlet file, 308
reconfiguration, 19
recovery, best practices, 417–418
reference implementation, 63–64
ReflectionException
applications, as MBeans, 359–362
applications, MBeans on behalf of, 379–381
connector client proxy, 319–320
overriding RequiredModelMBean, 215–221
registering MBeans, 248–249
registering
MBeans by
application, 407–409
management adapter, 410–411
MBeanServer bootstrap, 411
registerMBean() method, 331
third party, 409–410
.mlet files, 308
registerMBean() method, 246, 331
registration, 246
registry. See also MBeanServer.
availability, monitoring, 477–478
owner, 472–477
user, 477–478
regular expression methods, 256–257
relation types
returning, 315
roles
changing members, 315
describing a new, 310–312
number of MBeans, getting, 316
RelationService instances, getting, 315
unresolved, listing, 316
relational database access, 445
relational methods, 256
RelationNotification class, 120–122. See also Notification class.
relations service
access related sets, 308–309
example: the SessionPool relation, 309–312
RelationService class
attributes, 313
operations, 313–314
query operations, 316
relation operations, 314–316
relationship MBeans, 69
RelationService class. See also MBeanRegistration; NotificationListener.
addRelation() method, 314
attributes, 313
createRelation() method, 314
creating relation types, 310–312
external relation type, 313
external relations, 314
findAssociateMBeans() method, 316
findReferencingRelations() method, 316
findRelationshipOfType() method, 316
getAllRoles() method, 315
getReferencedMBeans() method, 316
getRelationTypeName() method, 315
getRoles() method, 315
getRolesUnresolved() method, 316
instantiating, 310
internal relation type, 313
internal relations, 314
methods, 314–316
operations, 313
query operations, 316
relation operations, 314–316
removeRelation() method, 315
setRoles() method, 315
RelationSupport class, 316. See also MBeanRegistration
RelationType interface, 314
releaseMBeanServer() method, 329
remoteness, deployment patterns, 340
removeListener() method, 331
removeNotification Listener() method, 261–262
removeNotification() method, 297
removeRelation() method, 315
removeRelationType method, 314
removing
MBeanServer, 329
notification listeners, 331
notifications, 297
relations, 315
Request for Comments (RFCs), 10–11
RequiredModelMBean class. See also ModelMBean.
instantiating and customizing, 67–68
overriding, 215–221
resource adapters, 443
resource manager, 443
resource schema, best practices, 418–420
ResourceAdapterModule class, 442–444
resourceAdapters attribute, 443
resources. See managed resources.
Resources attribute, 440
RestartingService class, 290–292
RestartingServiceMBean class, 290
RFCs (Request for Comments), 10–11
RMI-IIOP protocol, 446
RMI_IOOPResource attribute, 446
Role class, 311
role descriptor, 174
RoleInfo class, 310–312, 314
RoleList class, 311–312, 314–316
RoleResult class, 315–316
roles, relation type
changing members, 315
describing a new, 310–312
number of MBeans, getting, 316
RelationService instances, getting, 315
unresolved, listing, 316
runtime management, 16
RuntimeException, 249
RuntimeExceptionException
overriding RequiredModelMBean, 215–221
registering MBeans, 249

S
scalars, 126
schedule notifications, 297
scheduling behaviors, 298–299
SCSL (Sun Community Source License), 63
SecureContainer class, 113–114
security, J2EE, 461–462
security, JMX
JMX permissions
MBeanPermission, 329–332
MBeanServerPermission, 328–330
MBeanTrustPermission, 329–330
MBeanTrustPermission, 328–332
permission-based security fundamentals
definition, 327
policy, 328
SecurityManager, 327–328
policy configuration, 335–336
running with security enabled, 334–335
security exposures, 326
SPMA, 332–334
unauthorized MBean access, 326
unauthorized MBeanServerFactory access, 326
untrusted MBean registration, 326
using JMX security, 332–336
SendPastNotifications attribute, 297
Server attribute, 439
Server class
incorporating an MBeanServer, 78–79
self-managing service example, 210–213
ServerMBean interface, 75–76
static model MBeans, 199–210
server ID, getting, 254
ServerInfo.xml, 211
ServerMBean interface, 75–76, 96
servers attribute, 438
ServerVendor attribute, 440
ServerVersion attribute, 440
service container, 479–480
service flow layer, 472
service implementation, 468–469
service-oriented applications, 51
service-oriented architectures (SOAs), 468–469
servlets attribute, 443
ServletStats interface, 456
Session class, 77, 195–210
session class with integrated model MBean,
210–211
SessionBean EJBs, 443
SessionBeanStats interface, 456
SessionInfo.xml, 221
SessionPool class
creating relation types, 310–312
self-managing service example, 211–213
ServerMBean interface, 75–78
static model MBeans, 194–210
SessionPool relation
relation type, 309
relation’s roles, 309
resource pooling, 309
sessionpool.mlet file, 302–304
set methods, 256
set requests, 168–172
setAttribute() method, 251, 331
setAttributes() method, 93, 251
setManagedResource() method, 159–161
setMBeanServer() method, 255
setMethod descriptor, 170
setModelMBeanInfo() method, 159
setRoles() method, 315
setThresholds() method, 283–284
setting
  attribute thresholds, 283–284
  MBean attribute values, 93, 331
  ModelMBeanInfo, 208
setupRPSMonitor() method, 286–287
severity descriptor, 177
short data type, 126
shut down() method, 200
shutting down Server model MBeans, 200
signatures, 173
simple operations, 35
SimpleSecureContainer class, 113–114
SimpleSwitch class, 247–249
SimpleSwitchMBean class, 247–249
SimpleType class, 126
single-agent-view topologies, 422–427
singleton operations, 35
SMI (System Management Interface), 13
SMS, 14
SMUX, 11
SNMP (Simple Network Management Protocol) description, 13
JSRs, 59
read-only management, 48
RFCs (Request for Comments), 11
standards, 10
SOAs (service-oriented architectures), 468–469
software resources, managing. See inventory applications.
Solstice, 14
Sonic Software Sonic XQ, 525
Sonic XQ, 525
specification name, getting, 254
specification vendor name, getting, 254
specification version, getting, 254
SpecificationName attribute, 254
specifications
  J2EE, 431–432. See also JSR 77.
  Java Management Extensions (JMX) v1.0 Specification, 60–64
  SpecificationVendor attribute, 254
  SpecificationVersion attribute, 254
  spma class, 332–335
  standard MBeans, 65–66
  standard technologies, 13–16
standards
  CIM, 10
  CMIP, 10
  ISO, 14
  MIBs, 10
  MOF, 10
  OSI/TMN, 14
  RFCs, 10–11
  SNMP, 10
standards bodies, 10
start() method, 200, 265
start stage, 17
starting Server model MBeans, 200
state management, J2EE, 447–449
stateful SessionBean EJBs, 443
StatefulSessionBean objects, 456
StatefulSessionBeanStats interface, 456
stateless SessionBean EJBs, 443
stateManageable attribute, 435
static attributes, defining, 171–172
static model MBeans
  approach for todd server
    availableSessions attribute, 205–208
    grow() method, 205–208
    instantiating the session, 198–199
    set ModelMBeanInfo() method, 205–208
    setting ModelMBeanInfo, 198–205
    size attribute, 205–208
    example of static approach for todd server, 194–208
SessionPool class without integrated MBeans, 197–198
static reconfiguration, 19
statistical data, 23
statistics. See Stats interface.
statisticsProvider attribute, 435
Stats interface
  EJBStats interface, 454–455
  EntityBeansStats interface, 455
  getActiveCount() method, 455
  getCloseCount() method, 458–459
  getCommittedCount() method, 453
  getConnectionFactory() method, 457
  getConnectionPools() method, 458
getConnections() method, 457–459
getConnectPools() method, 457
getConsumers() method, 460
getCreateCount() method, 455, 458–459
getDestination() method, 460
getDurableSubscriptionCount() method, 460
getExpiredMessageCount() method, 460–461
getFreePoolSize() method, 458–459
getHeapSize() method, 454
getJdbcDataSource() method, 458
getManagedConnectFactory() method, 457
getMessageCount() method, 460–461
getMessageWaitTime() method, 460–461
getMethodReadyCount() method, 456
getOrigin() method, 460
getPassiveCount() method, 456
getPendingMessageCount() method, 460–461
getPooledCount() method, 455
getPoolSize() method, 458–459
getProducers() method, 460
getReadyCount() method, 455
getRemoveCount() method, 455
getRolledbackCount() method, 455
getSentMailCount() method, 456
getServiceTime() method, 456
getSessions() method, 459
getStatisticNames() method, 454
getStatistics() method, 454
getUptime() method, 454
getUseTime() method, 457–458
getWaitingThreadCount() method, 458–459
getWaitTime() method, 457–458
isTransactional() method, 459
JavaMailStats interface, 456
JCAConnectionPoolStats interface, 458
JCAConnectionFactoryStats interface, 457
JDBCCo unreal PoolStats interface, 459
JDBCCollectionStats interface, 458
JDBCStats interface, 458
JMSConnectionStats interface, 459
JMSCConsumerStats interface, 460
JMSEndpointStats interface, 461
JMSProducerStats interface, 460
JMSSessionStats interface, 460
JMSStats interface, 459
JTAStats interface, 455
JVMStats interface, 454
MessageDrivenBeanStats interface, 455
ServletStats interface, 456
SessionBeanStats interface, 456
StatefulSessionBean objects, passivating to disk, 456
StatefulSessionBeanStats interface, 456
Stats interface details, 454–461
status data, 23
StockQuote service, 487–488, 515–516
StockQuoteManager class, 488–491, 496–497
StockQuoteManagerMBean class, 487–491
StockQuoteService class, 494, 515–516
stop() method, 175–176, 200, 265
stop stage, 17
stopping further observations, 265
stopping Server model MBeans, 200
store() method, 157–158
String name of the observed attribute, 265
StringMonitor
  ApacheMonitor, 275–277
  attributes, 273
  notification types, 274
  specific notification types, 274
StringMonitor class, 270–271, 275–277. See also Monitor class; NotificationBroadcasterSupport; StringMonitorMBean.
StringMonitorMBean interface, 264, 273. See also MonitorMBean; StringMonitor.
StringToCompare attribute, 273
StringValueExp class, 255–257. See also ValueExp.
StringValueExp interface, 255–256
subagents, 10–11
SubString() method, 257–258
Sun Community Source License (SCSL), 63
Sun Microsystems
  JDMK 4.2, 520
  JMX 1.0 reference implementation, 519–520
  JMX Expert Group, 58
  Sun ONE Application Server, 525–526
  Sun ONE Portal Server, 526
Sun ONE Application Server, 525–526
Sun ONE Portal Server, 526
Sybase EAServer 4.0, 525
System Management Interface (SMI), 13
T
  table aggregate type, 128–129
  TackCatalog class, 344–392
  TackCatalog0 class, 355–356
TackCatalogDD class, 373–378
TackCatalogMBean class, 344–392
TackCatalogMM class, 383–392
TackCatalogPO class, 394–406
TackCatalogSS class, 365–392
TackCategory class, 344–364, 367–392, 396–406
targetObject descriptor, 175
targetType descriptor, 175
TCK (Technology Compatibility Kit), 63
TCP/IP, history of management systems, 7

Technologies
AIC, 15
ARM, 15–16
choosing, 48–49
CIM, 14
CIM/WBEM, 14
CIMOM, 14
CMIP, 14
DMTF, 14
homogenizing, 12
for Java-based resources, 14–15
JDMK, 15
JMX, 14–15
Management Architecture, 13
Netview, 13
for network management, 12, 13–14
OpenView, 13
Pegasus, 14
proprietary, 13
SMI, 13
SMS, 14
SNMP, 13
Solstice, 14
standard, 13–16
TMX4J, 15
vendor-specific, 12, 13
WBEM, 14
WBEMsource, 14
Technologies PLC: Orbix..., 524
Technology Compatibility Kit (TCK), 63
Threshold attribute, 279–281
TIBCO, 58
time of day daemon (todd) exercise
browser control, 82–83
classes, 70–71
management, 71–73
MBeans, 74–78

MBeanServer, 78–79
monitoring, 79–82
Timer class. See also MBeanRegistration; TimerMBean.
creating relation types, 310–312
description, 296–300
notification operations, 297–299
timer notification operations
addNotification() method, 297
removeNotification() method, 297
scheduling behaviors, 298–299
TimerMBean attributes, 296–297
using timer services, 290–292
timer service
basics of use, 290–292
definition, 69
notification period, 292
notification queue
prioritized by date, 292
set of notifications, 292
at three points in time, 293–294
scheduling, 289
Timer class, 296–300
timer notifications, 294–296
TimerFilter class, 295–296
TimerMBean interface, 296–299. See also Timer class.
TimerNotification class, 120–122, 294–296. See also Notification class.
TimerNotification object, 297
times() method, 256
Tivoli, 520–521
TMN, 59
TMX4J, 15
TMX4J 1.0, 520–521
todd (time of day daemon) exercise
browser control, 82–83
classes, 70–71
management, 71–73
MBeans, 74–78
monitoring, 79–82
topology applications, 32–33
transactions, 446
triggers, 36

U
UDDIRegistryClient class, 478, 498–502, 503–505
UDDIRegistryClientManager class, 503–505
UDDIRegistryClientManagerMBean class, 478, 497–505
UDDIRegistryController class, 475–477
UDDIRegistryData class, 475–477
UDDIRegistryManagerMBean class, 474–475
UDDIRegistryRuntimeManagerMBean class, 476–477
unavailable resources, 16–17
uninstall stage, 17
unregisterMBean() method, 249–251, 331
upgrades, unattended, 30
URLResource attribute, 446
URLs
resource references, 446
to specified String resource, returning, 308
URLs attribute, 307
V
value descriptor, 170
value() method
Query interface, 258
ValueExp interface, 256
ValueExp interface, 255–257. See also
AttributeValueExp; StringValueExp.
vendor-specific technologies, 12, 13
version descriptor, 167
Vigor Soft hawkEye, 523
visibility descriptor, 167, 170, 174–175, 177, 187–188
void setAttribute() method, 93
W
WBEM (Web Based Enterprise Management), 10, 14
WBEMsource, 14
Web Based Enterprise Management (WBEM), 10, 14
Web Component Manager, 522
Web NMS, 523
Web service execution environment management
execution environment owner, 480–482
execution environment user, 482–483
failure detection and recovery, 482
handlers, 483
information tracked, 480–481
lifecycle, 481–482
overview, 479–480
service container, 479–480
Web service management
with execution environment, 483–486
execution environment MBean, 488–491
with its own MBean, 487–488
metrics, 483–486
monitoring for availability, 495–497
publishing portType and port, 491–495
StockQuote service WSDL document, code listing, 515–516
UDDIRegistryClient class, code listing, 497–502
UDDIRegistryClientManager class, code listing, 502–505
UDDIRegistryClientManagerMBean implementation, code listing, 497–503
UDDIRegistryClientManagerMBean interface, code listing, 502–503
with Web service, 486–495
Web service owner, 483–495
Web service user, 495–497
WebServiceManagerMBean implementation, code listing, 511–515
WebServiceRuntimeManagerMBean implementation, code listing, 506–510
Web service owner, 483–495
Web service registry management
registry availability, monitoring, 477–478
registry owner, 472–477
registry user, 477–478
Web service user, 495–497
Web services
conceptual stack, 470–473
description layer, 471, 472
discovery layer, 472, 473
functional description, 468–469
network layer, 470, 472
nonfunctional description, 469
overview, 467–468
publication layer, 471–472, 473
service flow layer, 472
service implementation, 468–469
SOAs, 468–469
work flow layer, 473
XML-based messaging layer, 471
Web Services Tool Kit 3.1, 523
WebLogic Server 7.0, 524
WebModule class, 442–444
WebServiceClientManager class, 496–497
WebServiceClientManagerMBean class, 495–497
WebServiceManager class, 489–490, 511–512
WebServiceManagerMBean class, 483, 511–515
WebServiceRuntimeManager class, 507–510
WebServiceRuntimeManagerMBean class, 481, 506–510
WebSphere 5.0, 523
WebSphere Business Components Composer, 524
WebSphere Voice Server, 524
work flow layer, 473

X
XML-based messaging layer, 471
XML file example, 223–233
XMLModelMBean class, 215–221
XMLServer class, 220–221

XMLService class, 192, 208–213
XMLService MBean
APIs, 190–192
CIMDTD, 190
creating model MBeans for Apache, 191
creating ModelMBeanInfo instance, 189
customizing data in separate file, 189
instantiating XMLService MBeans, 191–192
priming ModelMBeanInfo from XML files, 189–192
service MBean, 189–190
XMLServiceMBean class, 190–192