

Index

A

- A/V formats, 438
- abstract leases, 76
- access limitations, 175
- ACID properties, 353
- action listeners, 192
- activation (extension to RMI), 389
- activation server (phoenix), 390–391
- activation, use of
 - configuration, 407–409
 - LookupDiscoveryService interface, 419–430
 - maintaining state information, 401–407
 - renewing leases, 410–419
 - server, 393–398
 - service, 391–393
 - service, running, 398–401
 - services, lazy and nonlazy, 401
- active references, 25
- additive permissions, 174
- Address class, 56
- administration, zero, 21
- alive, staying, 47
- AllPermission, 172, 173
- anonymous classes, 191
- Ant
 - build.xml file, 27–29
 - description, 27
 - directory structure, 27–28
 - parameters, 27
 - project files, 30–32
 - targets, 28
- antBuildFiles directory, 30
- Apache Axis server, 468
- Apache license, open-source, 1, 5
- API (application programming interface), 2
- applications, partitioning, 16–18
- Arnold, Ken, 91
- attribute matching, 55
- attributes registered, 65
- audio stream, push/pull, 439
- audio system example
 - audio-visual content parameters, 438
 - basic client, 462–464
 - content interfaces, 441
 - content sinks, 447–450
 - description, 437, 438
 - file source servers, 457–461
 - HTTP sinks, 446
 - HTTP source, 443–445
 - linkages between layers, 442
 - Ogg Vorbis interface, 445
 - playlists, 461
 - sink implementation, 450–453
 - sink servers, 453–456
 - source and sink interfaces, 439–441
 - source servers, 456–457
 - transport interfaces, 442
- authenticating servers, 338–343
- authentication
 - of clients, 303, 304, 344–349
 - required by clients, 343–344
 - required by servers, 349–350

author's Web site, 249
authorization, 303, 350
AWT events, 216

B

banking system, 4
books
 Design Patterns, 215
 The Jini Specifications, Second Edition, 91
BrevilleExtraLiftToaster example, 68–69, 70
bridge, Web Service-Jini
 discovery and invocation, 465–467
 examples, 469–471, 472–474
 security policy, 472
 structure, 471
broadcast discovery, 43–51
broadcast range, 49
build directory, 27, 28
build.xml file, 27–29

C

cache, monitoring, 246–248
CD databases, 456
certificates, signed, 178
changes in services, leasing. *See* service changes, leasing
changes in services, monitoring, 230–235
class definitions
 loading, 69
 uploading, 177
class files variables, 30
classes
 accessibility from CLASS-PATH, 21
 anonymous, 191
 changes in Jini 2.0, 22

ConfigUtils, 274
convenience, 56
DiscoveryEvent, 45–47
DiscoveryListener, 44
file delivery to client, 23
FileClassifierImpl, 128, 133, 153
FileClassifierLandlord, 208–211
FileClassifierLeasedResource, 206–207
HttpmdUtil, 323
in .jar files, 23
Jeri, 136–140
JoinManager, 167–170
JRMP, 140–141
LandlordLease, 79
leased FileClassifier, 202–205
LeaseRenewalManager, 75
LeaseRenewalService, 410
lookup services, 50–51
LookupDiscovery, 43, 161
LookupDiscoveryManager, 164–166
LookupLocatorDiscovery, 163–164
MIMEType, 91–93
multiple class files, 192
non-RMI proxy implementation, 147
problem domain, 185–187
Reaper (cleanup), 207–208
ServiceItem, 60
ServiceMatches, 69
ServiceRegistrar, 50–51, 59, 67–69, 179
ServiceRegistration, 60, 64
ServiceType, 57
UIDescriptor, 376–379

- UnicastRemoteObject, 83, 118
 - unknown, 435
 - See also* LookupLocator class; ServiceStarter class
 - classifier, file, 222–230
 - classifying files, 87
 - CLASSPATH environment variable, 22, 189
 - clean directory, 28
 - client ignorance, 19
 - clients
 - audio system example, 462–464
 - authentication of, 344–349
 - confidentiality, 330
 - description, 94
 - exception handling, 104
 - lookup, 12–13
 - multicast, 99–104
 - proxy verification, 326
 - requiring authentication, 343–344
 - security requirements, 177–178
 - structure, 14–15
 - transactions example, 366–371
 - unicast, 94–99
 - client-side constraints, 312
 - ClockDevice package, 261–264
 - ClockFrame service, 252–259
 - clocks example. *See* flashing clocks example
 - code sources, 180
 - codebase, specifying, 271–274
 - command-line options, 265
 - Comment class, 56
 - commit protocol, two-phase, 353
 - competitors, 4
 - compile target, 28
 - compiler, rmic, 118, 119
 - complete service, uploading, 105–115
 - components, 9
 - ComputerTimer driver, 260
 - ComputerTimer service, 251
 - confidentiality
 - clients, 330
 - description, 303
 - invocation constraints, 304
 - SSL server, 331–332
 - configuration
 - codebase, 271–274
 - command-line options, 265
 - file classifier server, 315
 - generic servers, 274–279
 - runtime configuration, 265
 - runtime information, 274–279
 - service ID, 268–271
 - specifying, 267
 - zero configuration, 23
 - ConfigurationFile language, 266
 - ConfigUtils class, 274
 - constraints
 - alternative, 344
 - client-side, 312
 - mandatory and preferred, 304
 - on invocation, 303
 - on methods, 305
 - convenience classes, 56
 - cookies, 77, 81
- D**
- debugging, 25
 - deploy directory, 28
 - Design Patterns* book, 215
 - Deutsch, Peter, 3
 - device, finding on network, 196
 - diagrams
 - landlord classes, 78, 79, 80
 - UML, 39, 40, 45
 - DialogFactory, 375

digital signatures, creating, 181

directories

- antBuildFiles, 30

- build, 27, 28

- clean, 28

- deploy, 28

- dist, 27, 28

- httpd.classes, 27

- lib, 23

- resources, 27

- src, 27, 30

directory structure, Ant, 27–28

discovery between middleware

- services, 466

DiscoveryEvent class, 45–47

DiscoveryGroupManagement interface, 162

DiscoveryListener class, 44

DiscoveryLocatorManagement interface, 162

DiscoveryManagement interface, 162

dist directory, 27, 28

distance service, 196

Djava.rmi.server.logCalls flag, 25

Djava.security.debug flag, 25

djinns, 171

Dnet.jini.discovery.debug flag, 25

E

editors example, 53–55

Eight Fallacies, 3, 4

encumbered formats, 438

Entry class

- attribute matching, 55

- attributes registered, 65

- convenience classes, 56

- description, 54

- restrictions, 55

- uses of, 57

entry points, 373

environment variable CLASS-
PATH, 22, 189

ERI. *See* Jeri (Jini Extensible Re-
mote Invocation)

errors

- garbage collection-related, 24

- HTTP server-related, 23

- Java-related, 21

- Jini-related, 22

- localhost, 23

- lookup-related, 24

- network configuration, 23

- preferred value, 24

- proxy verification, 328–330

- RMI stubs-related, 24

events

- file classifier, 222–230

- listener lists, 218–221

- listener sources, 221

- listeners, leasing, 230

- models, 215

- registration, 217

examples

- BrevilleExtraLiftToaster,
68–69, 70

- file editors, 53–55

- FooLandlord, 83–86

- InvalidLookupLocator, 36–39

- matching, 196–202

- MulticastRegister, 45–49

- registering FileClassifierImpl,
168–170

- security policy file, 176, 183

- server side-specific action
listener, 192

- unicast server, 61–64

- UnicastRegister, 39–43
- Web Service-Jini bridge, 469–471, 472–474
- See also* audio system example; FileClassifier example; flashing clocks example; transactions example
- exceptions
 - clients, 104
 - security, 174
- explicit proxy, 134, 218, 231
- explicit references, 124
- exporter for Jeri, 120
- exporting for Java, 118
- exporting in Jini 2.0, 119
- exporting services, 23
- Extensible Remote Invocation.
 - See* Jeri (Jini Extensible Remote Invocation)
- F**
- factories
 - current set, 375
 - FileClassifier example, 380–382
 - implementation, 190
 - interfaces from, 374
 - landlord lease, 77, 79
 - marshalling, 375
- failure modes of methods, 91
- failure, transaction, 357
- federations, 2
- fiddler service, 421–422
- file classifier, 222–230
- file classifier server, 312–318
- file editors example, 53–55
- file source servers, 457–461
- FileClassifier example
 - back-end service, 133
 - class definition, loading, 114
 - client, 132
 - client implementation, 112–114
 - code, 89, 93
 - factory, 380–382
 - interface, 93, 132, 379
 - location of class file, 114
 - mixed RMI and non-RMI proxies, 131, 151–159
 - non-RMI proxy, 142–151
 - non-RMI proxy classes, 147
 - non-RMI proxy, running, 148
 - overall implementation, 105
 - RemoteFileClassifier interface, 132, 134
 - RMI proxy, 132–142
 - running, 114–115
 - server implementation, 106–112, 124, 154–156
 - service specification, 90
- FileClassifierImpl class, 128, 133, 153
- FileClassifierLandlord class, 208–211
- FileClassifierLeasedResource class, 206–207
- files
 - classifying, 87
 - signing, 181
- flags for debugging, 25
- flashing clocks example
 - ClockDevice package, 261–264
 - ClockFrame service, 252–259
 - ComputerTimer driver, 260
 - ComputerTimer service, 251
 - description, 7–9, 249
 - downloading compiled classes, 7
 - multiple clocks, 260, 264
 - runtime behavior, 264

- TickerTimer driver, 259
- TickerTimer service, 250
- Timer service, 249
- UPnP solution, 249
- FooLandlord example, 83–86
- formats, A/V, 438
- FrameFactory, 375
- fully qualified path name, 21

G

- Gamma, Eric, 215
- garbage collection, Jeri, 124–125
- garbage collection-related errors, 24
- getGroups() method, 51
- getHost() method, 39
- getLocator() method, 51
- getPort() method, 39
- getRegistrar() method, 39–41
- getServiceID() method, 51
- globally unique identifier (GUID), 60
- Gosling, James, 3
- groups of services, 43
- GUID (globally unique identifier), 60

H

- handback data, 217
- home audio system example. *See* audio system example
- hostile lookup services, 171, 172–173, 178–181
- HTTP server errors, 23
- httpd.classes directory, 27
- HTTPMD (HTTP + Message Digest) protocol, 322–325
- HttpmdUtil class, 323

I

- icons, 387
- identity management
 - authenticating servers, 338–343
 - clients requiring authentication, 343–344
 - clients, authentication of, 344–349
 - constraints, alternative, 344
 - description, 334
 - JAAS (Java Authentication and Authorization Service), 335–336
 - keystores, 336–337
 - servers requiring authentication, 349–350
- ignorance, client, 19
- images, 387
- implementation, 10
- inexact service matching, 193–195
- inheritance in project files, 30
- installation process, 6
- integrity of transport
 - client, 318
 - description, 303, 318
 - invocation constraints, 304
 - SSL or TLS server, 320–322
 - TCP server, 319
- interfaces
 - as entry points, 373
 - configuration, 265
 - DiscoveryGroupManagement, 162
 - DiscoveryLocatorManagement, 162
 - DiscoveryManagement, 162
 - factory objects, 374

- FileClassifier, 93
- LeasedResource, 81–82
- LeaseRenewalService, 412–419
- LookupDiscoveryService, 419–430
- MutableFileClassifier, 222, 401
- NameEntry, 187–189, 191, 193
- Ogg Vorbis, 445
- ProxyAccessor, 223
- ProxyPreparer, 307–312
- remote, 121
- RemoteEventListener, 218–221
- RemoteFileClassifier, 132, 134
- ServiceDescriptor, 285–287
- ServiceDiscoveryManager, 237
- ServiceItemFilter, 238, 241–242
- source and sink, 439–441
- introspection library, 434
- InvalidLookupLocator example, 36–39
- invocation constraints, 303
- J**
- JAAS (Java Authentication and Authorization Service), 335–336
- .jar files, 23
- Java Media Framework (JMF), 377
- Java Remote Method Protocol (JRMP), 117, 137, 140–141
- Java security model, 181
- Java virtual machine (JVM), 31, 60
- java.lang.Object method, 50
- Java-related errors, 21
- JComponentFactory, 375
- jdb debugger, 25
- JDialogFactory, 375
- JDK 1.2 security model, 171
- Jeri (Jini Extensible Remote Invocation)
 - classes, 136–140
 - configuration, 122–124
 - exporter class, 119
 - garbage collection, 124–125
 - Java Remote Method Protocol (JRMP), 117, 137, 140–141
 - proxy accessor, 125
 - remote interfaces, 121
 - RMI, 117–119
 - standard exporter, 120
- JFrameFactory, 375, 376
- Jini
 - meaning, 2
 - success stories, 3
 - uses, 2
- Jini 2.0
 - configuration mechanism, 122
 - ConfigUtils class, 274
 - constraints, 304
 - dynamic policy use, 350
 - explicit proxy, 134, 218, 231
 - explicit references, 124
 - exporting and unexporting, 119
 - HttpmdUtil class, 323
 - .jar packages repackaged, 23
 - Java security mechanism, 303
 - Jeri instead of RMI, 24
 - logging API, 25
 - logging package, 281
 - LookupDiscovery constructors, 44
 - main classes changed, 22

- ProxyAccessor interface, 223
 - runtime configuration, 265
- Jini 2.1 files for compilation, 38
- Jini Extensible Remote Invocation. *See* Jeri (Jini Extensible Remote Invocation)
- jini.home parameter, 27
- jini-core.jar file, 38
- Jini-related errors, 22
- JMF (Java Media Framework), 377
- JoinManager class, 167–170
- JRMP (Java Remote Method Protocol), 117, 137, 140–141
- jsk-lib.jar, 38
- jsk-platform.jar, 38
- JTextArea, 60
- JVM (Java virtual machine), 31, 60
- JWindowFactory, 375
- K**
- keepalive means, 47
- KeyEvent events, 215
- keystores, 181, 336–337
- keytool, 181
- Konno, Satoshi, 252
- L**
- landlord interface, 82
- landlord packages, 77–81
- LandlordLease class, 79
- LaunchAll program, 36
- lazy services, 401
- LeasedResource interface, 81
- LeasePeriodPolicy interface, 82
- LeaseRenewalService interface, 412–419
- leases
 - abstract, 76
 - cancelling, 74
 - cookies, 77, 81
 - description, 73
 - event listeners, 230
 - expiring, 25, 74
 - granting and handling, 75
 - interfaces, 81–82
 - landlord class diagrams, 78, 79, 80
 - landlord packages, 77–81
 - policies, 82
 - requesting and receiving, 73
 - typical lease time, 73
 - See also* renewing leases
- levels, severity, 281
- lib subdirectory, 23
- licensing model, 1, 5
- listeners
 - action listeners, 192
 - DiscoveryListener class, 44
 - MulticastRegister, 45
 - WindowListener, 191
 - See also* events
- listing services, 431–433
- localhost parameter, 27, 274
- localhost, error in use of, 23
- Location class, 56
- locators, lookup, 161–166, 175
- logging, 305–306
- logging API, 25
- logging package
 - description, 281
 - LookupDiscovery class, 281–283
- lookup locators, 161–166, 175
- lookup service-related errors, 24
- lookup services
 - broadcast discovery, 43–51
 - definition, 9

- description, 35
 - discovery, 44
 - groups, 43
 - hostile, 171, 172–173, 178–181
 - locating, 35
 - multiple, 43
 - ServiceRegistrar class, 50–51
 - unicast discovery, 36–43
 - See also* reggie lookup service; services, registering
- LookupCache, 243–248
- LookupDiscovery class
 - description, 43
 - logging, 281–283
 - multicast search, 161
- LookupDiscoveryManager class, 164–166
- LookupDiscoveryService interface
 - fiddler service, 421–422
 - use of, 422–430
- LookupLocator class
 - description, 36
 - getHost() and getPort() methods, 39
 - getRegistrar() method, 39–41
- LookupLocatorDiscovery class, 163–164
- M**
- Magneti Marelli Motorsport project, 4
- mahalo transaction manager, 355
- matching attributes, 55
- matching example, 196–202
- matching services, 70
- methods, constraints on, 305
- middleware, 2, 90, 117
- MIME types, 87
- MIMETYPE class, 91–93
- monitoring cache, 246–248
- MouseEvent events, 215, 216
- MP3 format
 - encumbered, 438
 - interface for, 446
 - no player in JMF, 447
 - pulled, 439
- MPEG format
 - movie player in JMF, 447
 - MPEG-7, 439, 456
- multicast
 - clients, 99–104
 - description, 35
 - discovery, 43–51, 175
 - range, 49
- MulticastRegister example, 45–49
- multiple class files, 192
- MutableFileClassifier interface, 222, 401
- N**
- Name class, 56
- NameEntry interface, 187–189, 191, 193
- names of projects, 31
- Nedap N.V. project, 4
- network configuration error, 23
- network layer, dependence on, 23
- “network plug and play” definition of Jini, 21
- network plug and play, 1
- networked environments, 1
- networks, 9
- nonactivatable server, 297–300
- nonactivatable service, 286, 287–296
- norm service, 411–412
- O**
- Ogg Vorbis
 - format, 438
 - interface, 445

open-source Apache license, 1, 5
Orange project, 5
Orbitz project, 5

P

packages

- Java-related errors, 21
- Jini-related errors, 22
- landlord, 77–81

PanelFactory, 375

parameters

- Ant, 27
- RMI, 179
- set by preferences, 176

paranoia, 180

partitioning applications, 16–18

path name, fully qualified, 21

PayableFileClassifierImpl service,
358–363

permissions

- AllPermission, 172, 173
- granted to protection do-
mains, 182

phoenix activation server,
390–391

pictures, 387

playlists, 461

“plug and play” definition of Jini,
21

plug and play, network, 1

points of failure, transaction, 357

policies for leases, 82

policy files

- example, 176, 183
- security, 171, 183

preferences, 176

preferred constraints, 304

preferred value error, 24

print statements for debugging,
25

printer, finding on network, 196

problem domain, 185–187

project files, 30–32

project names, 31

PropertyChange events, 215

protection domains, 180

protocols

- choices, 122
- end of, 18–19
- HTTPMD (HTTP + Message
Digest), 322–325
- SSL (Secure Sockets Layer),
307, 320–322
- TCP (Transmission Control
Protocol), 306

proxies, service

- choices, 127–131
- description, 13–14, 104
- explicit, 134, 218, 231
- fat, 127
- FileClassifier example,
132–159
- mixed RMI and non-RMI, 131,
151–159
- non-RMI, 129
- ProxyPreparer interface,
307–312
- RMI, 128

proxy objects

- for lookup service, 50, 59
- maintaining state informa-
tion, 60
- RMI, 60

proxy verification

- clients, 326
- errors, 328–330
- HTTPMD, 322–325
- SSL trustor servers, 327–328
- verifiers, 325

ProxyAccessor interface, 223

ProxyPreparer interface, 307–312

push/pull audio stream, 439

R

- random ports used by RMI, 178
- range of broadcast, 49
- Reaper (cleanup) class, 207–208
- references
 - active, 25
 - strong and weak, 124
- reggie lookup service
 - description, 35
 - HTTPMD, 325
 - location of objects, 180
 - RMI, 50
 - ServiceStarter class, 300
- registrar, 6
- registration of events, 217
- registration. *See* services, registering
- remote events, 216
- remote interfaces, 121
- remote method invocation. *See* RMI
- remote procedure call (RPC)
 - communications, 3, 105, 151
- RemoteEventListener interface, 218–221
- RemoteFileClassifier interface, 132, 134
- renewing leases
 - LeaseRenewalManager class, 75
 - LeaseRenewalService class, 410
 - LeaseRenewalService interface, 412–419
 - norm service, 411–412
- Representational State Transfer (REST) community, 457
- resources directory, 27
- restrictions on Entry class, 55

RMI (remote method invocation)

- extending UnicastRemote-Object, 118
- Jeri, 117–119
- JRMP-style RMI, 137
- parameters, 179
- proxies, service, 128
- random ports, 178
- stubs-related errors, 24
- thin proxy to fat service
 - back-end, 105
- traditional, 118
- use of, 2

rmic compiler, 118, 119

rmid (replaced by phoenix), 390–391

RPC (remote procedure call)

- communications, 3, 105, 151

Rubean project, 4

runtime configuration, 265

S

Secure Socket layer (SSL). *See* SSL

security

- access limitations, 175
- client requirements, 177–178
- combinations, 332–334
- djinns, 171
- exceptions, 174
- file classifier server, 312–318
- invocation constraints, 303
- JDK 1.2 model, 171
- logging, 305–306
- methods, constraints on, 305
- paranoia, 180
- permissions, 172, 173, 182
- policy files, 171, 176, 183
- policy for Web Service-Jini
 - bridge, 472
- protection domains, 180

- ProxyPreparer interface,
 - 307–312
- service security requirements,
 - 175–176
- See also* identity management
- SecurityManager, 171
- semantic events, 215
- sequence diagrams, UML, 39, 40, 45
- server side-specific action listener, 192
- server, unicast, 61–64
- servers
 - Apache Axis, 468
 - authenticating, 338–343
 - file classifier, 312–318
 - file source servers, 457–461
 - generic, 274–279
 - nonactivatable, 297–300
 - proxy verification, 327–328
 - requiring authentication, 349–350
 - service providers, 104
 - sink servers, 453–456
 - source servers, 456–457
 - structure, 15–16
 - trusted, 327–328
- service changes, leasing
 - client, 211–213
 - description, 202
- FileClassifierLandlord class,
 - 208–211
- FileClassifierLeasedResource class, 206–207
- leased FileClassifier class,
 - 202–205
- Reaper (cleanup) class,
 - 207–208
- server, 205–206
- service ID, 65, 67, 268–271
- service matching, inexact, 193–195
- service proxies. *See* proxies, service
- ServiceDiscoveryManager class
 - cache, monitoring, 246–248
 - filtering services, 241–242
 - interface, 237
 - LookupCache, 243–248
 - services, cache of, 242–245
 - services, finding immediately, 238–241
- ServiceInfo class, 56
- ServiceItem class, 60
- ServiceItemFilter interface, 238, 241–242
- ServiceMatches class, 69
- ServiceRegistrar class
 - description, 50–51, 59, 179
 - finding services, 67–69
- ServiceRegistration class, 60, 64
- ServiceRegistration method, 50
- services
 - changes, monitoring, 230–235
 - ClockFrame, 252–259
 - ComputerTimer, 251
 - defining common knowledge, 87
 - definition, 9
 - description pseudocode, 287
 - distance service, 196
 - exporting, 23
 - fiddler, 421–422

- groups of, 43
- lister program, 431–433
- nonactivatable, 286, 287–296
- PayableFileClassifierImpl, 358–363
- signing, 182
- support services, 18
- supported, 1
- TickerTimer, 250
- Timer, 249
- transactions example, 363–366
- unknown, 433
- uploading completely, 105–115
 - See also* service changes, leasing; services, finding; services, registering
- services, cache of, 242–245
- services, finding
 - description, 67
 - immediately, 238–241
 - matching, 70
 - multiple matches, 69
 - ServiceRegistrar class, 67–69
 - specifying, 90
- services, registering
 - description, 10–11, 59
 - generating ServiceEvent type events, 231
 - join manager, 167–170
 - register(), 60
 - ServiceItem class, 60
 - ServiceRegistrar class, 50–51, 59, 179
 - ServiceRegistration class, 60, 64
- ServiceStarter class
 - description, 285
 - nonactivatable server, 297–300
 - nonactivatable service, 286, 287–296
 - reggie lookup service, 300
 - ServiceDescriptor interface, 285–287
- ServiceTemplate class
 - description, 70
 - specifying sought service, 67
- ServiceType class, 56, 57
- severity levels, 281
- signatures, creating, 181
- signed trust certificates, 178
- signing services, 182
- signing standard files, 181
- Simple Object Access Protocol (SOAP) invocation protocol, 465, 466
- sinks
 - content sinks, 447–450
 - HTTP sinks, 446
 - implementing, 450–453
 - servers, 453–456
 - sink interfaces, 439–441
- sleep() method, 47
- SOAP (Simple Object Access Protocol) invocation protocol, 465, 466
- Sommers, Frank, 4, 125
- source files, 27, 30
- sources of code, 180
- specification, 10
- specifying services, 90
- src directory, 27, 30
- src.files variable, 30
- SSL (Secure Sockets Layer)
 - configuration, 307
 - integrity checking, 320–322
 - server confidentiality, 331–332
 - trusted servers, 327–328

- standard files, signing, 181
- Starter Kit installation process, 6
- state information, maintaining, 60, 401–407
- Status class, 56
- staying alive, 47
- strong and weak references, 124
- stubs-related errors, 24
- subtractive permissions, 174
- success stories, 3
- support services, 18
- supported services, 1
- Swing
 - JTextArea, 60
 - staying alive, 47

T

- targets, Ant, 28
- taxi-driver, Jini, 217
- TCP (Transmission Control Protocol), 306
- The Jini Specifications, Second Edition* book, 91
- threads, daemon and user, 47
- TickerTimer driver, 259
- TickerTimer service, 250
- time to live (TTL), 43, 50
- Timer service, 249
- TLS (Transport Layer Security), 320–322
- toaster example, 68–69, 70
- transactions
 - mahalo transaction manager, 355
 - manager, 354
 - overview, 353
 - participants, 355
 - two-phase commit protocol, 353*See also* transactions example

- transactions example
 - AccountsImpl service, 363–366
 - client, 366–371
 - description, 355–358
 - PayableFileClassifierImpl service, 358–363
- transition match events, 231
- Transmission Control Protocol (TCP), 306
- transport layer as A/V parameter, 438
- Transport Layer Security (TLS), 320–322
- transport protocol choices, 122
- trust certificates, signed, 178
- trusted servers, 327–328
- TTL (time to live), 43, 50
- two-phase commit protocol, 353

U

- UDD, 466
- UDP (User Datagram Protocol), 43
- UIDescriptor class
 - attributes, 378
 - description, 376
 - role, 377
 - toolkit, 377
- UML sequence diagrams, 39, 40, 45
- unencumbered formats, 438
- unexporting, 119
- unicast
 - clients, 94–99
 - description, 35
 - discovery, 36–43, 175
 - server example, 61–64
- UnicastRegister example, 39–43
- UnicastRemoteObject class, 83

UnicastRemoteObject, extending, 118
universally unique identifier (UUID), 70, 77
unknown classes, 435
unknown services, 433
unlazy services, 401
uploading complete services, 105–115
UPnP solution to flashing clocks, 249
User Datagram Protocol (UDP), 43
user threads, 47
UUID (universally unique identifier), 70, 77

■ **V**
value error, 24
variables. *See* environment variables
Venners, Bill, 181
verifiers, 325
versions of Jini and Java, 22
versions of Jini. *See* Jini 2.0 and Jini 2.1

■ **W**
Waldo, Jim, 91
WAV format
 interface for, 446
 uncompressed, 438
weak and strong references, 124
Web Services
 integrating with Jini, 465–467
 simple, 467–469
 XML data types, 90

Web sites
 author's, 249
 banking system, 4
 failure modes of methods, 91
 JAAS (Java Authentication and Authorization Service), 335–336
 Java security model, 181
 Jeri, 125
 Jini, 5
 Magneti Marelli Motorsport project, 4
 Nedap N.V. project, 5
 Orange project, 5
 Orbitz project, 5
 success stories, 3
wildcarding, 55
WindowFactory, 375
WindowListener, 191

■ **Z**
zero administration, 21
zero configuration, 23

