

Index

SYMBOLS

\$ (dollar sign) in \$CATALINA_HOME/conf directory, 58–59, 130

! (exclamation mark) for protecting manager URLs (!/), 134–135

/ (slash)

in context path for manager application, 134, 157

for protecting manager URLs (!/), 134–135

A

Access Log Valve, 163–166

ACLs (access control lists), 326

Active Server Pages (ASPs), 18–19

admin configurator application

admin role for, 54–55

advantages for configuration, 53–54

behind-the-scenes work of, 54

bootstrapping, 87

creating new users, 56, 58

enabling access to, 54–57

manager application versus, 88

security, 55, 322–323

Tomcat architectural model and, 57

typical screen, 53

URL for accessing, 55

admin role

admin configurator access and, 54–55

assigning to a user, 55

defining, 55

embedded mode in Tomcat and, 503–504

passwords, 55, 131, 322

administrator login for Tomcat on Windows, 34

admin.xml configuration file, 322–323

agent level (JMX), 427–430

Aggregate Report listener (JMeter), 403, 405

AJP (Apache JServ Protocol)

as Coyote Connector alternative, 47

implemented by JK and JK2 Connectors, 231, 234

overview, 238–239

for Web server Connectors, 231–232, 233–234

AJP 1.3 Connector

configuring for Apache, 243

configuring for IIS, 274

for load balancing with Apache, 256–257

performance tuning, 291

ports for clustering, 471–472

ajp13 worker

for load balancing with IIS, 291

overview, 240

for Tomcat with Apache, 246

for Tomcat with IIS, 278

Ajp13Connector class, 230

aliases, 50, 370–371. See also virtual hosting

all target (Ant), 569

Ant tool

as ASF project, 3

build process, 566–570

building a sample Web application, 566–570

build.xml file, 567

core tasks (table), 562–565

custom tasks in Modeler, 507–510

development-time directory structure, 566

downloading, 559

e-mail notification of build status, 570–572

further information, 565

Ant tool (continued)

- global properties, 561, 567
- installing, 560
- managing Web applications, 154–157
- overview, 560–562
- scripts for embedded Tomcat
 - features, 506–507
 - Jakarta Commons Modeler and, 507–510
 - minimal embedded server startup, 512–517, 521
 - operational flow, 510–511
- targets, 155–157, 567–569
- user-defined tasks, 565

Apache Crimson XML parser, 208

Apache JServ Connectors, 233–234

Apache JServ Protocol (AJP), 47, 231–232, 233–234, 238–239

Apache License, 5–6

Apache project, 2

Apache Server

- building with SSL support, 252
- combining Tomcat with, 418
- configuring for load balancing, 472–473
- configuring SSL, 251–254
- connecting Tomcat with
 - configuring the AJP 1.3 Connector, 243
 - configuring the `httpd.conf` file, 247
 - configuring the `jk2.properties` file, 247–248
 - configuring Tomcat workers, 243–247
 - files modified, 241
 - installing the `mod_jk2` module, 241–242
 - steps involved, 241
 - testing the setup, 248–250
- documentation, 238
- downloading binaries, 237
- JK2 Connector for, 237, 238–239
- `mod_ssl` setup, 254
- testing the load balancer, 265–270
- testing the SSL-enabled setup, 254–255
- Tomcat load balancing with, 255–265
- virtual hosting
 - approaches, 370
 - IP-based implementation, 371–372
 - name-based implementation, 373–374
 - with Tomcat 5, 381–386

Apache Software Foundation (ASF), 3, 12

Apache Web site, 3

APIs (application programming interfaces)

- defined, 7
- Java APIs, 7–8

J2EE APIs, 8–9

J2SE APIs, 9

Appender components (Log4J)

- behavior, 533
- configuring Log4J programmatically, 531
- overview, 527
- XML configuration file, 532

“Application already exists at path” error, 157

application server configuration (`server.xml` file)

- Connector component, 66–68
- Engine component, 68–70
- Host component, 72–76
- Logger component, 70–71
- Realm component, 72

application servers

- J2EE, 9–10
- standalone mode versus, 65–66
- Tomcat and, 10
- Tomcat configuration, 66–76

architecture. *See also* architecture (Tomcat)

- JMX, 424–430
- Model 1, 19, 20
- Model 2 (MVC), 25–26
- Web application, 24–25

architecture (Tomcat). *See also* specific components

- admin configurator and, 57
- Connectors, 47
- context, 49–50
- Engine, 48
- Host, 49
- key points, 50
- Loggers, 49
- other classes, 50
- overview, 45–46
- Realm, 48
- revised for Tomcat 4, 45
- Server, 47
- Services, 47–50
- Tomcat 5 and, 46
- top-level components, 46
- Valves, 48–49
- Web server Connector, 230–233

ASF (Apache Software Foundation), 3, 12

ASP.NET (Microsoft), 18

ASPs (Active Server Pages), 18–19

Assertion Results listener (JMeter), 403, 406

assertions (JMeter), 410–411

`<attribute>` subelement (Modeler), 509

`<auth-constraint>` element (`web.xml` file), 108, 167

authentication. See also specific Realms

- for admin configurator application, 323
- BASIC, 338
- bind mode versus comparison mode, 355
- browser-based, 338–339
- configuring for Web applications, 340–341
- DIGEST, 338–339
- embedded Tomcat, 520
- form-based, 339
- HTTP Authorization Manager (JMeter), 409
- HTTPS client certificate, 339–340
- for manager application, 158, 323
- multiple sign-on without the Single Sign-on Valve, 166–169
- Servlet engine security realms and, 48
- Single Sign-on Valve for, 163, 166, 169, 362
- standards-based mechanisms, 337
- tomcat-users.xml file and, 59, 77

B**backup servers for clustering, 460**

- balancer **filter**
 - advantages and disadvantages, 491–492
 - configuring, 493–494
 - cookies and, 496
 - hardware-assisted request distribution, 496–497
 - load balancing with, 491–492
 - mod_jk2 Apache module versus, 490
 - overview, 490–491
 - redirection by, 490–491, 496
 - rules, 494–495
 - standalone mode for, 492–493
 - testing, 495–496

base64 encoding, 338**BASIC authentication, 338****bin directory, 41****bind mode authentication, 355****bootstrap class loader, 195–197****browsers**

- browser-based authentication, 338–339
- cookie role in, 469
- HTTP Proxy Server (JMeter) configuration, 413

bugs

- digest utility, 346
- reload command (manager), 142
- remove command (manager), 146

build status (Ant), 570–572**build target (Ant), 155, 156, 157, 569****build.xml file (Ant), 154–157, 567****C****CA (certificate authority), 364–365****caching**

- Cache object, 443–444
 - of classes by class loaders, 199
 - JMX manageability, 443–444
- callisto.dom **virtual host example**
- directory structure, 376, 377, 385, 387–388
 - httpd.conf file, 383, 384
 - IP addresses, 375
 - JSP file, 380–381
 - server.xml file, 380, 389
 - Tomcat 5 configuration, 379–381
 - workers2.properties file, 384, 385–386, 390

“Cannot resolve user database reference” error, 158**Catalina class loader, 204****Catalina Servlet engine, 48–49, 85****catalina-ant.jar file, 154, 156****CATALINA_BASE environment variable**

- for configuration directory, 52–53
- finding internal classes, 387–388, 390–391

catalina.bat file (Windows), 37–38**CATALINA_HOME environment variable**

- for configuration directory, 52–53
- finding internal classes, 387–388, 390–391
- in startup files, 256
- for Tomcat installation, 34

\$CATALINA_HOME/conf directory, 58–59, 130**catalina.policy file**

- Catalina code permissions, 333–334
- for fine-grained access control, 84–87
- overview, 59
- Security Manager and, 332
- SSI security issues, 222–223
- system code permissions, 332–333
- Web application permissions, 85–87, 334–335

catalina.properties file, 59, 87**certificate authority (CA), 364–365****Certificate Signing Request (CSR), 365****certificates**

- configuration file for generation, 252
- generating a test certificate, 253–254
- JSSSE, 363–365
- for resource protection, 366–367

CGI (Common Gateway Interface)

- default Servlet definition, 79–80
- enabling support for, 215–216
- overview, 14
- Servlet for support, 220
- Servlet mappings, 82

cgi **Servlet in default web.xml file, 79–80**

Chain Certificate, 365

class loaders

- bootstrap class loader, 195–196
- Catalina class loader, 204
- class caching, 199
- common class loader, 203–204
- core class restriction, 201
- creating a custom class loader, 199–200
- delegation model, 197–198, 201, 205–206
- dynamic class reloading, 206
- enabling creation using Security Manager, 336
- Endorsed Standards Override Mechanism and, 198, 203, 208
- extension class loader, 196
- further information, 200
- importance of, 193
- lazy loading (on demand), 199
- namespace separation for, 199, 201–202
- overview, 194–198, 209
- packages split among, 207
- parent and child classes and, 199
- platform-independence and, 194
- role of, 194
- security issues, 200
- SecurityManager class for, 202
- shared class loader, 204
- singleton classes and, 207–208
- standard J2EE class loaders, 194–196
- system class loader, 196, 203
- Tomcat class loaders, 202–206
- troubleshooting, 206–208
- Web application class loader, 205–206
- XML parsers and, 208

classes. See also specific classes

- caching by class loaders, 199
- core class restriction for class loaders, 201
- directories for core Java classes, 195–196
- Java permission classes, 330–331
- logging from a Java class (Log4J), 534–535
- singleton, class loaders and, 207–208

classes directory, 94

CLASSPATH environment variable

- Log4J, 529
- standard extension mechanism versus, 196
- system class loader and, 196, 197

clean target (Ant), 567, 569

<Cluster> element (server.xml file), 477–479

clustering

- application-level load balancing, 490–497

back end configuration

- in-memory replication, 466–468, 475–482
- Persistent Session Manager with file store, 465–466, 482–486
- Persistent Session Manager with JDBC store, 466, 487–490
- testing, 490
- basics, 460–461
- benefits of, 457, 458–459
- configuring a Tomcat 5 cluster, 469–473
- cookies and modern browsers, 469
- fail-over behavioral pattern, 460–461, 482
- front end
 - configuring, 473–475
 - options for load-balancing implementation, 462–464
- high availability and, 457, 459
- load balancing
 - options for implementing, 462
 - sticky sessions overview, 463–464
 - via Apache mod_jk2, 463, 473–475
- master-backup topological pattern, 460
- multiple Tomcat instances on one machine, 469–470
- overview, 498–499
- performance and, 497–498
- response time and, 498
- scalability and, 457, 458–459
- session management in Tomcat 5, 468
- session sharing, 464–468
- shutting down the Tomcat cluster, 471
- Tomcat 5 model, 461–468

common class loader, 203–204

common directory, 42

Common Gateway Interface. See CGI

common log file format, 163

Commons, defined, 5

communications paths for Web server Connectors, 230–231

comparison mode authentication, 355

compile target (Ant), 156, 568, 569

compiling SendMailServlet, 185–186

conf directory, 42

Config Elements (JMeter), 408–409

config settings for workers2.properties file, 244–245

configuring. See also configuring Tomcat; configuring Web applications

- AJP 1.3 Connector, 243
- balancer filter, 493–495
- Coyote HTTP/1.1 Connector (Tomcat 4.1), 213–215
- HTTP/1.1 Connector (Tomcat 4.0), 212

- JAAS Realm, 359–361
 - JavaMail sessions, 184–187
 - JDBC DataSource, 182–184
 - JDBC, for legacy code integration, 311–312
 - JDBC, JNDI-based, 301–311
 - jdk2.properties file, 247–248, 280
 - JMeter thread groups, 398–400
 - JNDI Realms, 354–355, 357
 - lifecycle listeners, 187–191
 - Log4J, 529–533
 - manager application, 132–133
 - Persistent Session Manager, 173–175
 - Request Dumper Valve, 172
 - Request Filter Valve, 171–172
 - resources via JNDI, 179–182
 - Single Sign-on Valve, 169
 - Tomcat 4.x Connector for SSL, 215–216
 - UserDatabase Realm, 343–345
 - Web application authentication, 340–341
 - workers for Tomcat with Apache, 243–247
 - workers for Tomcat with IIS, 277–280
- configuring Tomcat. See also admin configurator application; specific configuration files**
- application server configuration, 66–76
 - authentication and the tomcat-users.xml file, 77
 - bootstrap configuration, 51–52
 - catalina.policy file, 84–87
 - catalina.properties file, 87
 - for CGI support, 220–221
 - clustering, 469–473
 - configuration components, 45–50
 - default deployment descriptor, 77–81
 - directory for configuration files, 52, 54
 - files in \$CATALINA_HOME/conf directory, 58–59
 - future of, 87–88
 - management versus configuration, 88
 - server configuration via default server.xml, 60–66
 - server.xml, context descriptors, and web.xml collaboration, 81–84
 - for SSL support, 222–223
 - for SSL
 - Tomcat 4.1, 215–216
 - Tomcat 5, 219–220
 - startup problems and, 87–88
 - typical configuration, 46
 - Web application context definitions, 76–77
- configuring Web applications. See also web.xml file**
- classes directory, 94
 - directory structure, 91–92
 - lib directory, 95
 - META-INF directory, 95–96
 - public resources, 92–94
 - ROOT application, 92
 - tllds directory, 94–95
 - URL mappings, 93–94
 - WEB-INF directory, 94–95
- connection pooling, 297–298, 300, 312–319**
- Connector API, 9**
- <Connector> **element (server.xml file)**
- for AJP 1.3 Connector, 243, 274
 - application server configuration, 66–68
 - application server versus standalone mode and, 65–66
 - for clustering, 471–472
 - Coyote HTTP/1.1 configuration
 - Tomcat 4.1, 213–216
 - Tomcat 5, 216–220
 - HTTP/1.1 configuration (Tomcat 4.0), 212
 - JMX-manageable attributes, 433–435
 - for load balancing with Apache, 256–257
 - optimization, 418–419
 - for SSL protection of resources, 366–367
 - for SSL support, 215–216, 219–220
 - standard connectors, 67–68
 - as subelement of <Service> element, 65
 - for Tomcat behind a proxy server, 224
 - for virtual hosting, 379
 - for virtual hosting (disabling HTTP Connector), 381
- Connector **interface, 212, 213, 230**
- Connectors. See also HTTP Connectors; Web server Connectors; specific Connectors**
- application server configuration, 66–68
 - application server versus standalone mode and, 65–66
 - in default server.xml file, 65–66
 - HTTP versus Web server, 211
 - interface, 212, 213, 230
 - JMX manageability, 433–435
 - for MBean Server, 428
 - Tomcat 4 architecture, 47
 - Tomcat default (Coyote), 47, 211
- console, logging to (Log4J), 538**
- constant throughput timer (JMeter), 402**
- constant timer (JMeter), 402**
- containers (Tomcat), 46, 48–49. See also specific kinds**
- context components, 49–50, 76–77**
- context descriptor. See <Context> element (server.xml file)**
- <Context> **element (manager.xml file), 132, 159**
- <Context> **element (server.xml file)**
- for balancer filter and standalone mode, 493
 - defining, 76–77
 - described, 76
 - JNDI resource configuration, 179–182

<Context> element (server.xml file)

<Context> element (server.xml file) (continued)

for Persistent Session Manager, 174–175

reloadable attribute, 142–143

for virtual hosting, 380

context path

in CGI program URLs, 14, 139

deploy command (manager), 136

error messages, 157–158

in manager application parameters, 134

unique for Web applications, 135, 138, 157

<context-param> element (web.xml file), 101, 112

context.xml file, installing Web applications via, 142

Controller component of MVC architecture, 25

Controllers (JMeter), 401

cookies

balancer filter and, 496

HTTP Cookie Manager (JMeter), 409

role in modern browsers, 469

Servlet container use of, 15–16

for tracking state information, 468

user's session token, 15

core classes (Java), 195–196, 201

Coyote Connectors

backward compatibility, 211

for CGI support, 220–221

HTTP/1.1 Connector

disabling for load balancing with Apache, 257

Tomcat 4.1, 212–216

Tomcat 5, 216–220

JK2 Connector, 235, 239

for SSI support, 222–223

for SSL, 215–216, 219–220

support unified in, 68

as Tomcat default, 47, 211

CoyoteConnector class

org.apache.coyote.tomcat4 package, 213

org.apache.coyote.tomcat5 package, 212

CoyoteServerSocketFactory class, 215

CSR (Certificate Signing Request), 365

D

data listeners (JMeter), 405

database connection pooling, 297–298, 300, 312–319

DataSource factory (JDBC), 182–184, 302

debug attribute of Connectors, performance and, 224, 225

debugging. See also remote debugging

JSPs, 556

Request Dumper Valve for, 49, 163, 172

default context, virtual hosting and, 376, 380, 386

default home page (Tomcat), 38

<DefaultContext> element (server.xml file)

described, 69, 76

JNDI resource configuration, 179–182

JNDI-based JDBC configuration, 301–303, 306–308

delegation model for class loaders

overview, 197–198

security and, 201

Tomcat class loaders and, 205–206

deleting or removing

default lifecycle listeners, 190–191

JAAS Realm users and roles, 361

JNDI Realm users and roles, 358

jsp-examples Web application, 323

ROOT Web application, 322

servlets-examples Web application, 323

Web applications permanently (Tomcat 5), 147–148

Web applications temporarily (Tomcat 4), 146–147

Web applications using Web interface, 152

deploy command (manager). See **deploying Web applications**

deploying Web applications

installing versus, 136

from a local path (Tomcat 5), 140–141

manually, 130

redeploying, 142

remotely (Tomcat 5), 139–140

in Tomcat 4, 136–137

in Tomcat 5, 139–141

using manager Web interface, 153

deployment descriptors. See web.xml file

DeployTask class, 154–155

<description> element (web.xml file), 100, 112

destroy() method for Servlets, 15

DIGEST authentication, 338–339. See also User-Database Realm

digest.bat or digest.sh utility, 345–346

directories

Ant development-time directory structure, 566

for configuration files, 52, 54

for core Java classes, 195–196

for deploying Web applications manually, 130

installing Web applications from, 140

META-INF directory, 95–96

standard extension, 196

tomcat account permissions, 327, 328

Tomcat installation directory, 34, 39, 41–43

for virtual hosting with Tomcat, 376

virtual, under IIS, 280–283, 287

for WAR files, 376
 Web application structure, 91–92
 WEB-INF directory, 94–95

directory services, JNDI and, 176–177

displaying or listing
 deployed Web applications, 135
 JNDI resources available, 143–144
 OS and JVM properties, 144–145
 security roles available, 144
 session statistics, 148
 Tomcat internals using JMX proxy Servlet, 149–150
 Tomcat server status, 151

<display-name> **element (web.xml file), 100, 112**

dist **target (Ant), 568, 569**

<distributable> **element (web.xml file), 100, 113, 472–473**

distributed load testing (JMeter), 413–414

distributed services level (JMX), 430

distributed transactions, 303

distribution of Tomcat, 5–6, 160

DN (distinguished name), 355

DNS (Domain Name Service), 372, 374

“Document base does not exist...” error, 158

Document Type Definition (DTD), 99

doEndTag() **method (JSP), 22**

dollar sign (\$) in \$CATALINA_HOME/conf directory, 58–59, 130

domain names, fully qualified (FQDN), 370, 372

DOMConfigurator **for Log4J, 532, 533**

doStartTag() **method (JSP), 22**

downloading. See also Internet resources
 Ant tool, 559
 Apache binaries, 237
 embedded Tomcat, 512
 isapi_redirector.dll file, 275
 JavaBeans Activation Framework libraries, 185
 JavaMail libraries, 185
 JSSE, 363
 JVM, 29, 31
 Log4J, 528–529
 mod_jk2 Apache module binaries, 241
 MX4J (open source JMX), 450
 MySQL, 304, 349
 OpenLDAP 354
 PoolMan, 312
 Tomcat, 33, 40

DTD (Document Type Definition), 99

dynamic class reloading, 206

dynamic MBeans, 431

dynamic Web content, 11

E

Eclipse IDE, 553–554

EJB (Enterprise JavaBeans), 8

<ejb-local-ref> **element (web.xml file), 113**

<ejb-ref> **element (web.xml file), 113**

e-mail log messages (Log4J), 544–545

e-mail notification of build status (Ant), 570–572

embedded mode in Tomcat
 accessing embedded Tomcat, 517–519
 administrator role with, 503–504
 Ant scripts
 features, 506–507
 Jakarta Commons Modeler and, 507–510
 minimal embedded server startup, 512–517, 521
 operational flow, 510–511
 application scenarios, 503
 approaches, 505
 defined, 501
 downloading embedded Tomcat, 512
 hardware scenario requiring, 502
 importance in system design, 502–504
 installing embedded Tomcat, 512
 internal versus external control, 506
 Jakarta Commons Modeler, 507–510
 JBuilder IDE and, 556
 manager application with, 519–522
 manager role for authentication, 520
 MBean Server role, 505–506
 minimal embedded server startup, 512–517
 NetBeans IDE and, 554
 overview, 505–507, 522–523
 shutting down the embedded server, 519
 testing the embedded server, 517–519

enableLookups **attribute of Connectors, performance and, 225, 226**

encapsulation by JSP tag libraries, 22

“Encountered exception” error, 157

Endorsed Standards Override Mechanism, 198, 203, 208

Engine component
 application server configuration, 68–70
 JMX manageability, 433
 for load balancing with Apache, 257–258
 overview, 48

<Engine> **element (server.xml file)**
 Access Log Valve configuration, 164
 application server configuration, 68–70
 attributes, 68–69
 JMX-manageable attributes, 433

<Engine> element (server.xml file)

<Engine> element (server.xml file) (continued)

jvmRoute attribute, 257–258, 267
for load balancing with Apache, 257–258, 267
subelements, 69–70
for virtual hosting, 379

Enterprise JavaBeans (EJB), 8

<env-entry> element (web.xml file), 109–110, 114

<Environment> element (server.xml file),
179–180, 439

Environment object, 439

environment variables. *See also specific variables*

for class paths, 196, 197
for configuration directory, 52–53
finding internal classes, 387–388, 390–391
for JVM installation, 30–31
for JVM memory limits, 392–393
in startup files, 256
for Tomcat installation, 34

error messages. *See also troubleshooting*

HTTP 404 error, 377, 385, 386
manager application, 157–158
“port number is in use,” 43

<error-page> element (web.xml file), 105–106, 114
/etc/profile, running tomcat.sh from (Linux), 32

europa.dom virtual host example

directory structure, 376, 385, 387–388
httpd.conf file, 383, 384
IP addresses, 375
JSP file, 380
server.xml file, 377–380, 388–389
Tomcat 5 configuration, 379–381
workers2.properties file, 384, 385–386, 390

Event Listener design pattern (JMeter), 401

exclamation mark (!) for protecting manager URLs
(! /), 134–135

extension class loader, 196–197

external Tomcat

embedded Tomcat and, 506
NetBeans IDE and, 554–555

F

fail-over behavioral pattern

fail-over defined, 460
in-memory session replication and, 482
overview, 460–461
Persistent Session Manager with file store and,
485–486
sudden fail-over, 486
tracking state information, 461

FAT32 file system (Windows), 326

file system security, 326–327, 328, 336–337

files, logging to (Log4J), 539

<filter> element (web.xml file), 101–102, 115,
493–494

<filter-mapping> element (web.xml file), 115–116

filters

balancer, for load balancing, 490–497
Remote Address Filter, 170
Remote Host Filter, 170
Request Filter Valve, 163, 170–172
Servlet, as alternative to Valves, 49

form-based authentication, 339

FQDN (Fully Qualified Domain Name), 370, 372

G

Gaussian random timer (JMeter), 403

GET command (HTTP), 372–373

global resource name and type (JNDI), 143

<GlobalNamingResources> element (server.xml
file), 63

GlobalResourcesLifecycle class, 188

GNU Project, 6–7

GPL (General Public License), 6–7

GRANT command (MySQL), 350

grant entry (Security Manager)

granting permissions to applications, 329
permissions available, 330–331
syntax, 329–330

Graph Full Results listener (JMeter), 403, 404

Graph Results listener (JMeter), 403, 404

Grinder load tester, 419

gzip files, installing JVM from, 31–32

H

HA (high-availability) systems. *See also fail-over behavioral pattern*

clustering and, 457, 459
high availability defined, 457
in-memory session replication and, 482
Persistent Session Manager with file store and,
485–486

hash algorithms (DIGEST), 338–339

heap size

JVM performance and, 225, 226
optimizing Java memory allocation, 417–418
virtual hosting and, 391–393

hello.jsp example, 128–129

history of Apache, 2

horizontal scalability, 457, 459

Host component. See also virtual hosting

- application server configuration, 72–76
- JMX manageability, 435
- nested Logger in virtual host, 76
- overview, 49

<Host> element (server.xml file)

- application server configuration, 72–76
- attributes, 74–75
- described, 69
- JMX-manageable attributes, 435
- for manager application, 159
- for standalone server, 380
- subelements, 75–76
- for virtual hosting, 379

Host : header (HTTP), virtual hosting and, 373, 374, 377, 379

host restriction, 368

hostname parameter

- deploy command (manager), 136
- manager application syntax, 134

HTML format for Log4J, 540–542

HTMLManagerServlet class, 151

HTTP Authorization Manager (JMeter), 409

HTTP Connectors

- application server configuration, 66, 67, 68
- for CGI support, 220–221
- Coyote as Tomcat default, 47, 211
- Coyote Connector, 47, 68
- Coyote HTTP/1.1 (Tomcat 4.1)
 - adapters for earlier Tomcat versions, 212
 - configuring, 213–215
 - SSL configuration, 215–216
- Coyote HTTP/1.1 (Tomcat 5)
 - configuring, 216–219
 - disabling for load balancing with Apache, 257
 - SSL configuration, 216–219
- described, 67
- disabling for clustering, 471
- disabling for virtual hosting, 381
- HTTP/1.1
 - application server configuration, 66, 67
 - Coyote Connector versus, 212
 - Tomcat 4.0, 212
 - for virtual hosting, 378–379
- overview, 47, 212
- performance tuning, 224–226
- port specification for, 212
- for running Tomcat behind a proxy server, 223–224

- for SSL support, 222–223
- for SSL support, 215–216, 219–220
- support unified in Coyote Connector, 68
- Web server Connectors versus, 211

HTTP Cookie Manager (JMeter), 409

HTTP 404 error, virtual hosting and, 377, 385, 386

HTTP Header Manager (JMeter), 409

HTTP monitor (JBuilder), 556

HTTP protocol, name-based virtual hosting and, 372–373, 374

HTTP Proxy Server (JMeter), 411–413

HTTP Request Defaults (JMeter), 409

HTTP requests with JMeter, 398–400

HTTP requests with manager application

- browser or scripts for, 129
- deploy command, 136–137, 139–142
- format for, 134–135
- install command, 136, 137–139
- list command, 135
- protecting URLs with !/, 134–135
- querying internals using JMX proxy Servlet, 149–150
- reload command, 142–143
- remove command, 146–147
- resources command, 143–144
- roles command, 144
- serverinfo command, 144–145
- sessions command, 148
- setting internals using JMX proxy Servlet, 150
- start command, 146
- stop command, 145
- undeploy command, 147–148

HTTP servers, combining Tomcat with, 418

HttpConnector class, 212

httpd.conf file

- <Listen> directive, 372
- mod_jk2 Apache module directives, 247
- SSL directives for Apache, 251
- <VirtualHost> section, 371, 372, 373–374, 382–383

HTTPS client certificate authentication, 339–340

I

IANA (Internet Assigned Numbers Authority), 375

<icon> element (web.xml file), 100, 116

IDEA (IntelliJ), 552

IDEs (Integrated Development Environments)

- Eclipse, 553–554
- IntelliJ IDEA, 552
- JBuilder, 555–557

IDEs (Integrated Development Environments)

IDEs (Integrated Development Environments) (continued)

NetBeans, 554–555
remote debugging support, 551–552
Web sites, 551

IE. See Internet Explorer

IIS (Internet Information Server)

combining Tomcat with, for performance, 418
connecting Tomcat with
 adding the ISAPI filter to IIS, 283–285
 configuring the AJP 1.3 Connector, 274
 configuring the `jk2.properties` file, 280
 configuring Tomcat workers, 277–280
 creating a virtual directory, 280–283
 installing the ISAPI filter, 274–275
 steps involved, 272
 testing the setup, 285–286
 testing Tomcat and IIS installations, 273–274
 updating the Registry, 275–277
default Web page, 273
keep-alive option, 290
log file, 287–288
market share, 2
performance tuning, 289–291
SSL with, 291
starting the server, 273
timeout, 290
troubleshooting, 287–288
Web site hits per day, 289
`iis_redirect2.reg` script, 277
`index.htm` or `.html` or `.jsp` file
 for testing the Apache load balancer, 265–266
 Web application example, 128
 welcome page handling, 83–84, 93
inheritance of logging behavior, 49
`init()` method for Servlets, 14
`init` target (Ant), 155
in-memory session replication
 advantages and disadvantages, 467–468
 <Cluster> element (`server.xml` file), 477–479
 HA fail-over, 482
 load balancing for requests with same session,
 481–482
 overview, 466–467
 SimpleTcpCluster implementation, 476
 test JSP for, 479–480
 testing, 480–481
 Tomcat components for, 475–476
 <Valve> element (`server.xml` file), 479
in-process workers versus plug-in workers, 239–240

`install` command (manager). See also **deploying Web applications**

`config_url` option (Tomcat 4.1), 139
`deploy` command versus, 136
general format, 138
installing an application in Tomcat 4, 137–139
in Tomcat 4.0 versus 4.1, 137–138, 139
Tomcat 5 and, 139
`war_url` option (Tomcat 4.1), 139

installing

Ant tool, 560
certificate from a CA, 364–365
embedded Tomcat, 512
ISAPI filter, 274–275
JavaBeans Activation Framework libraries, 185
JavaMail libraries, 185
JMeter, 396
JNDI driver, 355
JSSE, 363
JVM on Linux, 31–33
JVM on Windows, 29–31
`mod_jk2` Apache module, 241–242
testing the Tomcat installation on Windows, 34–39
Tomcat on Linux, 40–41
Tomcat on Windows using Windows Installer, 33–34,
 131
Tomcat on Windows using ZIP file, 39–40
troubleshooting Tomcat installation, 43–44
Web applications in Tomcat 4, 136, 137–139
Web applications in Tomcat 5, 139–142

instrumentation level (JMX), 426

Integrated Development Environments. See IDEs

IntelliJ IDEA, 552

interfaces

Connector, 212, 213, 230
JNDI as, 176
manager application, 129
Swing (JMeter), 397

Interleave Controller (JMeter), 406

internals (Tomcat)

finding internal classes, 387–388, 390–391
querying using `JMX proxy Servlet`, 149–150
querying using `queryjmx` utility, 451–454
setting using `JMX proxy Servlet`, 150

Internet Assigned Numbers Authority (IANA), 375

Internet Explorer (IE)

cookies and, 469
HTTP Proxy Server (JMeter) configuration, 413

Internet Information Server. See IIS

Internet resources. See also downloading

- Ant tool information, 565
- Apache documentation, 238
- Apache site, 3
- CGI information, 14
- class loader information, 200
- common log file format, 163
- history of Apache, 2
- IANA RFC 1918 (private networks), 375
- IDE sites, 551
- IDEs, 551
- JAAS Realm information, 358
- Java permission classes information, 330
- jkstatus Web page, 240, 267–268
- JNDI information, 353
- JSSE information, 363
- J2EE and J2SE platforms, 7
- J2EE licensees, 10
- license information, 7
- Netscape Directory Server site, 353
- open source load testers, 419
- OpenLDAP information, 353, 354–355
- OSI site, 7
- PoolMan site, 312
- Security Manager information, 332
- Tomcat default home page, 38
- Tomcat documentation, 1
- Tomcat support network, 44
- Tomcat troubleshooting information, 43
- user lists (Tomcat), 44

Internet Services Application Programming Interface filter. See ISAPI filter

“Invalid context path specified” error, 157

invoker Servlet, 78

IP addresses for private networks, 375**IP-based virtual hosting**

- aliasing in, 370–371
- avoiding common mistakes, 372
- defined, 370
- implementing in Apache, 371
- multihoming hosts and, 370
- overview, 370–371

ISAPI (Internet Services Application Programming Interface) filter

- adding to IIS, 283–285
- confirming installation, 287
- described, 271
- installing, 274–275
- logging, 287

- role of, 272

- updating the Windows Registry for, 275–277
- isapi_redirector.dll file, 274–275

J**JAAS (Java Authentication and Authorization Service) Realm**

- adding or deleting users and roles, 361
- changing the Java security policy, 360–361
- configuring the Provider, 360
- configuring the Realm directive, 361
- described, 72
- further information, 358
- obtaining a Provider, 359
- overview, 358
- setup for authentication technology, 359
- steps for configuring, 359
- terminology, 358

Jakarta Commons Modeler, 507–510**Jakarta project. See also ISAPI filter; JMeter load testing software**

- ISAPI filter provided by, 271
- JMeter project, 396
- subprojects, 3, 4–5
- Tomcat project, 4

Jakarta Web site, 33**Java API for XML Processing (JAXP), 198, 203, 208****Java Authentication and Authorization Service (JAAS) Realm. See JAAS Realm****Java Community Process (JCP), 423****Java compiler permissions, 84–85****Java Database Connectivity. See JDBC****Java Management Extensions. See JMX****Java Message Service (JMS), 8****Java Naming and Directory Interface. See JNDI****Java permission classes, 330–331****Java Secure Socket Implementation (JSSE), 363–365****Java Servlets. See Servlets****Java Transaction API (JTA), 9****Java 2 Enterprise Edition platform. See J2EE platform****Java 2 Standard Edition (J2SE) platform, 7, 9****Java Virtual Machine. See JVM****Java Web site, 29, 31****JavaBeans Activation Framework libraries, 185****javac command, 31, 32****JAVA_HOME environment variable, 30****java.lang.ClassLoader class, 200****java.lang.RuntimePermission class, 330**

JavaMail

- context descriptor for, 184–185
- described, 9, 178
- downloading and installing, 185
- libraries for, 185
- overview, 184
- Security Manager and, 336
- sending e-mail, 186–187
- `sendmail.jsp`, 186
- `SendMailServlet`, 185–186
- session configuration, 184–187
- `web.xml` file for, 185

`JAVA_OPTS` environment variable, 392–393

JavaServer Pages. See JSPs

JAXP (Java API for XML Processing), 198, 203, 208

JAX-RPC, 9

JBoss application server, 10

JBuilder IDE, 555–557

JCP (Java Community Process), 423

JDBC (Java Database Connectivity)

- alternative configuration for legacy code, 311–312
- alternative connection pool managers, 312–319
- basics, 294–299
- connection pooling, 297–298
- connections to RDBMSs, 295
- `DataSource` factory, 182–184, 302
- deploying third-party pools, 319
- described, 178
- driver types, 296–297
- JNDI-based configuration
 - MySQL example, 304–310
 - `<Resource>` element (`server.xml` file), 301–302
 - `<ResourceParams>` element (`server.xml` file), 301–303
 - steps involved, 300–301
 - testing the configuration, 310–311
- with MySQL, 301–302, 304–310
- operation model, 294–295
- with Oracle, 302–303
- Persistent Session Manager with JDBC store, 466, 487–490
- `PoolMan` pool manager with, 312–319
- Realm component, 72, 347–353
- security realm, 347–353
- steps in programming, 295
- Tomcat support for, 182
- Tomcat versions and, 298–299
- transactions and distributed transaction support, 303
- versions, 295–296

JDBC Realms

- adding `tomcat` user to MySQL, 350–351
- configuring with digested passwords, 349
- mapping columns to the required view, 347–349
- MySQL-based, defining, 351–352
- overview, 72, 347
- testing, 352–353

`JDBCTest.jsp` JNDI lookup example, 309–310

JK Connectors for Web servers, 234

`jkstatus` worker, 240, 263, 267–268

JK2 Connectors. See also AJP

- for Apache Server, 237, 238–239
- application server configuration, 67–68
- Coyote Connector, 235, 239
- described, 67
- `jk2.properties` file configuration, 247–248
- `mod_jk2` Apache module, 238, 241–242, 247
- for Web servers, 234–235

`jk2.properties` file

- for load balancing with Apache, 258–259
- for Tomcat with Apache, 247–248
- for Tomcat with IIS, 280

JMeter load testing software. See also specific features

- alternative products, 419
- assertions, 410–411
- Config Elements, 408–409
- configuration options, 398
- Controllers, 401
- data validation using assertions, 410–411
- defined, 5
- for distributed load testing, 413–414
- establishing scalability limitations, 416
- Event Listener design pattern, 401
- further analysis, 416
- HTTP Proxy Server, 411–413
- HTTP requests, 398–400
- installing, 396
- interpreting test results, 414–416
- list of features, 401
- Listener feature, 403–406
- Logic Controller feature, 406–408
- Observer, 401
- overview, 396
- running tests, 400–401
- Sampler feature, 408
- setting goals and testing them, 414–416
- starting, 396
- test plans, 397–401, 406–407
- thread groups, 397–398

- Timer feature, 402–403
- viewing test results, 401, 402
- jmeter **shell script**, 396
- meter.bat **file**, 396
- JMS (Java Message Service)**, 8
- JMX (Java Management Extensions)**
 - accessing via manager proxy
 - examples, 446–447
 - JNDI initial context setup, 452
 - modifying MBean attributes, 447–450
 - operation of the JMX proxy, 444–445
 - query operation general form, 446
 - remote probing with queryjmx, 452–454
 - remotely, via RMI Connector, 450–452
 - URLs for accessing, 445
 - architecture
 - agent level, 427–430
 - distributed services level, 430
 - instrumentation level, 426
 - overview, 424–425
 - described, 9
 - embedded mode in Tomcat and, 505–506
 - JSR-3 specification, 423–424
 - J2EE manageability requirement, 421–423
 - manageable elements in Tomcat 5
 - architectural components, 432–435
 - nested components, 436–437
 - overview, 431–432
 - resource objects, 439–444
 - run-time data objects, 437–438
 - MBean Server component, 427–428
 - MBeans types, 430–431
 - MX4J (open source JMX), 450–451
 - proxy Servlet, 149–150, 189
 - Remote API, 430
 - security issues, 454
- JMX proxy. See also manager application**
 - examples, 446–447
 - JNDI initial context setup, 452
 - modifying MBean attributes, 447–450
 - operation of the JMX proxy, 444–445
 - query operation general form, 446
 - remote probing with queryjmx, 452–454
 - remotely, via RMI Connector, 450–452
 - URLs for accessing, 445
- <jmx-operation> **task (Modeler)**, 510
- <jmx-service> **task (Modeler)**, 508–509
- JNDI (Java Naming and Directory Interface)**
 - configuring resources via, 179–182
 - data source lookup using, 309–310
 - defined, 176
 - directory services unified by, 176–177
 - emulation in Tomcat 5, 299–300
 - further information, 353
 - global resource name and type, 143
 - information obtained through, 176
 - JavaMail session configuration, 184–187
 - JDBC DataSource configuration, 182–184
 - JNDI-based JDBC configuration, 301–311
 - LDAP driver and, 176
 - listing resources available, 143–144
 - overview, 176–177
 - PoolMan pool manager with, 317–319
 - Realm component, 72, 353–357
 - security realm, 353–358
 - Tomcat support for, 177–178
 - typical Tomcat JNDI resources, 178
- JNDI Realms**
 - adding roles and users, 357
 - configuration attributes available, 353–354
 - configuring, 354–355, 357
 - creating the LDAP schema, 355–356
 - creating user access to the directory, 357
 - described, 72
 - further information, 353
 - installing JNDI driver, 355
 - populating the LDAP directory, 356–357
 - removing a role or user, 358
 - storing roles in the JNDI directory, 356
- jni **worker**, 240–241
- JServ Connectors**, 231–234
- JSP containers in Web containers**, 20–21
- JSP tag libraries (extensions)**, 21–24, 94–95
- <jsp-config> **element (web.xml file)**, 116–117
- jsp-examples **Web application**, 322, 323
- JSPs (JavaServer Pages)**
 - ASPs versus, 18–19
 - compilation into Servlet classes, 19
 - debugging, 556
 - example page using Struts, 22–24
 - hello.jsp example, 128–129
 - for in-memory session replication, 479–480
 - limitations for large projects, 19–20
 - Model 1 architecture, 19, 20
 - Model 2 (MVC) architecture, 25–26
 - page assembly, 20, 21
 - Servlets versus, 19
 - tag libraries, 21–24
 - Tomcat as Servlet container for, 1, 502
 - troubleshooting IIS rendering, 287–288

JspServlet in default web.xml file, 78–79

JSR-3 specification, 423–424

JSSE (Java Secure Socket Implementation), 363–365

JTA (Java Transaction API), 9

J2EE (Java 2 Enterprise Edition) platform

APIs, 8–9

application servers, 9–10

Endorsed Standards Override Mechanism, 198, 203, 208

J2EE-compliant defined, 10

J2SE as foundation for, 7

licensees list online, 10

manageability requirement, 421–423

obtaining, 7

overview, 7–9

standard class loaders, 194–196

vendor neutrality and, 10

J2SE (Java 2 Standard Edition) platform, 7, 9

JVM (Java Virtual Machine). See also Security Manager

alternatives to Sun, 418

API services, 7

bootstrap class loader in, 195–196

delegation model for class loaders and, 197–198

downloading, 29

installing on Linux, 31–33

installing on Windows, 29–31

limited to one Server instance, 47

listing properties, 144–145

memory limits for virtual hosting, 391–393

memory settings and HTTP Connector performance, 225, 226

security, 328–337

separate, for virtual hosts, 387–391

server VM optimization, 417

K

keep-alive option (HTTP), 290

keytool command, 363–365

Knuth, Donald (computer guru), 416

L

Layout components (Log4J), 527, 531

lazy loading (on demand), 199

lb worker, 240, 261–263

LDAP (Lightweight Directory Access Protocol), 176, 353, 355–357

ldapadd tool, 357

ldapremove command, 358

Level components (Log4J), 527–528

LGPL (Lesser General Public License), 7

lib directory, 95, 96

libraries

for JavaMail sessions, 185

JSP tag libraries, 21–24, 94–95

lib directory for, 95

licensing for Tomcat, 5–7

lifecycle listeners

defined, 187

displaying MBeans using JMX proxy Servlet, 189

lifecycle events sent by Tomcat components, 187

<Listener> element configuration, 187–188

removing default listeners, 190–191

Tomcat 5 configuration, 188–191

Lightweight Directory Access Protocol (LDAP), 176, 353, 355–357

Linux

building mod_jk2 Apache module on, 242

digest utility bug fix, 346

file system security, 328

installing JVM from RPM, 32–33

installing JVM from tar/zip files, 31–32

installing Tomcat on, 40–41

JSSE certificate keystore preparation, 363

running JMeter, 396

starting Tomcat in debugging mode, 552

tomcat account for, 324, 325–326

virtual interface configuration, 371

list command (manager), 135

list target (Ant), 156

<Listen> directive (httpd.conf file), 372

<Listener> element (server.xml file)

attributes, 188

described, 63, 70

for lifecycle listener configuration, 187–188

<listener> element (web.xml file), 102, 118

Listener feature (JMeter)

Assertion Results listener, 406

data listeners, 405

overview, 403

types of listeners (table), 403

visualization listeners, 404

listeners

Ant tool, 572

defined, 50

lifecycle, 187–191

listing. See displaying or listing

load balancing. See also lb worker

with Apache

CATALINA_HOME environment variable for, 256

disabling the Coyote HTTP/1.1 Connector, 257

- Engine component configuration, 257–258
- `jdk2.properties` file configuration, 258–259
- need for, 255
- setting AJP Connector ports, 256
- steps involved, 255–256
- worker configuration, 258–265
- `workers2.properties` file configuration, 263–265
- balancer filter for application-level, 490–497
- for clustering front end, 473–475
- with IIS, 291
- for requests with same session, 481–482
- testing the Apache load balancer
 - with different load factors, 269–270
 - `index.jsp` file for, 265–266
 - for round-robin behavior, 267–269
 - for sticky sessions, 266–267
- Load Runner software (Mercury Interactive), 419**
- load testing. See also JMeter load testing software**
 - alternatives to JMeter, 419
 - defined, 395
 - establishing scalability limitations, 416
 - importance of, 395–396
 - interpreting test results, 414–416
 - optimization techniques, 416–419
 - overview, 419–420
 - setting goals and testing them, 414–416
- `<locale-encoding-mapping-list>` **element** (`web.xml` file), **118**
- log file rolling (Log4J), 539**
- Log4J logging API**
 - Appenders, 527, 533
 - CLASSPATH environment variable, 529
 - components, 526–528
 - configuration file example, 528
 - configuring
 - programmatically, 530–531
 - using simple properties file, 529–530
 - using XML configuration file, 532–533
 - defined, 5, 525
 - downloading, 528–529
 - format modifiers, 549
 - Layouts, 528
 - levels, 527–528
 - Loggers, 526, 533
 - logging
 - to the console, 538
 - to a file, 539
 - HTML format for, 540–542
 - from a Java class, 534–535
 - log file rolling, 539
 - meta level options, 538
 - to multiple destinations, 542–543
 - to native system logs, 545–548
 - to NT event log, 545–548
 - to SMTP server, 544–545
 - from a Web application, 535–538
 - overview, 526, 550
 - PatternLayout formatting, 548–549
 - performance tips, 549
- Logger components (Log4J)**
 - behavior, 533
 - configuring Log4J programmatically, 530, 531
 - overview, 526
 - XML configuration file, 532
- Logger components (Tomcat)**
 - Access Log Valves versus, 163
 - application server configuration, 70–71
 - Coyote HTTP/1.1 Connector and, 213–214, 217
 - JMX manageability, 436
 - nested in virtual host, 76
 - overview, 49
 - standalone server configuration, 379–380
 - for `workers2.properties` file, 244, 278
- `<Logger>` **element (server.xml file)**
 - application server configuration, 70–71
 - attributes, 71
 - described, 70, 76
 - JMX-manageable attributes, 436
 - standard implementations, 71
 - for virtual hosting, 379–380
- Loggers (Ant), 570–572**
- logging. See also Log4J logging API**
 - Access Log Valve, 163–166
 - IIS, 287–288
 - ISAPI filter, 287
 - virtual host, 379–380
- Logic Controller feature (JMeter), 406–408**
- logical mapping for Servlet access, 17, 80–82**
- login form for authentication, 340–341**
 - `<login-config>` **element (web.xml file)**
 - authentication and, 340
 - for manager role, 133
 - for multiple sign-on, 167
 - Servlet 2.3-style, 108–109
 - Servlet 2.4-style, 118–119
- LoginContext (JAAS Realm), 358**
- logs **directory, 42**
- Loop Controller (JMeter), 407**

M

mail sessions. See **JavaMail**

MailLogger class (Ant), 570–571

MailLogger.properties file, 571

manageability requirement (J2EE), 421–423

Managed Beans. See **MBeans**

management tools. See **admin configurator application;**
manager application

manager application. See also **JMX proxy**

Ant-based interface, 154–157
configuring, 132–133
deploy command, 136–137, 139–142
with embedded Tomcat, 519–522
enabling access to, 130–131
error messages, 157–158
format for commands, 134–135
install command, 136, 137–139
installing versus deploying applications, 136
list command, 135
querying internals using JMX proxy Servlet, 149–150
reload command, 142–143, 152
remove command, 146–147, 152
resources command, 143–144
roles command, 144
sample Web application, 128–129
security considerations, 158–159, 322–323
serverinfo command, 144–145
sessions command, 148
setting internals using JMX proxy Servlet, 150
start command, 146, 152
stop command, 145, 152
tasks performed by, 129–130
undeploy command, 147–148
ways of interacting with, 129
Web interface, 150–153
web.xml file for, 132–133

<Manager> element (Persistent Session Manager), 173–174

<Manager> element (server.xml file)
JMX-manageable attributes, 437
for Persistent Session Manager with file store, 483
for Persistent Session Manager with JDBC store, 487

manager role

adding to embedded Tomcat, 520
passwords, 130, 131, 157, 159, 322
security considerations, 132–133, 158–159
tomcat-users.xml file entries, 130

ManagerServlet class, 151

manager.xml configuration file, 132, 322–323

MANIFEST.MF file, 95–96

master server for clustering, 460

master-backup topological pattern, 460

maxKeepAliveRequest attribute of Connectors, performance and, 225

MBean Server component, 427–428, 505–506

<mbean> subelement (Modeler), 509

<mbean-descriptor> task (Modeler), 509–510

MBeans (Managed Beans)

connectors, 428
defined, 430
displaying using JMX proxy Servlet, 189
exposed objects in Tomcat 5
architectural components, 432–435
nested components, 436–437
overview, 431–432
resource objects, 439–444
run-time data objects, 437–438
JMX agent level and, 427–430
JMX instrumentation level and, 426
MBean Server component, 427–428, 505–506
modifying attributes with JMX proxy, 447–450
overview, 426
protocol adapters, 428–429
types of, 430–431

McCool, Rob (Web server developer), 2

MD5 algorithm, 339, 345

memory

HTTP Connector performance and, 225, 226
in-memory session replication, 466–468, 475–482
optimizing Java allocation, 417–418
virtual hosting and, 391–393

MemoryRealm, 72, 342–343

Mercury Interactive's Load Runner software, 419

<message-destination> element (web.xml file), 119

<message-destination-ref> element (web.xml file), 119–120

meta level logging options (Log4J), 538

META-INF directory, 95–96

Microsoft Active Server Pages (ASPs), 18–29

Microsoft ASP.NET, 18

Microsoft Internet Information Server. See **IIS**

Microsoft Windows. See **Windows**

migration, Servlet container issues for, 16

<mime-mapping> element (web.xml file), 82–83, 104, 120

`min.xml` **minimal embedded startup script (Ant)**, 512–517, 521

Model component of MVC architecture, 25

model MBeans, 431

Model View Controller (MVC) architecture, 25–26

Modeler. *See* **Jakarta Commons Modeler**

`<modelerRegistry>` **task (Modeler)**, 509–510

`mod_jk2` **Apache module**

Apache Server versions and, 242

balancer filter versus, 490

building on Linux/Unix, 242

building on Windows, 242

for clustering load-balancing front end, 463, 473–475

downloading binaries, 241

`httpd.conf` file directives, 247

load balancing supported by, 255

load factor, 463

overview, 238

round-robin behavior, 267–269

shared hosting compatibility and, 369

sticky sessions enabled by, 463–464

for sticky sessions with no session sharing, 464

for sticky sessions with shared file store, 466,

484–485

virtual hosting and, 369, 381–382, 385, 386

`mod_ssl` **extension, setting up in Apache**, 254

Module Controller (JMeter), 407

Mozilla browser

cookies and, 469

HTTP Proxy Server (JMeter) configuration, 413

multihoming hosts, 370. *See also* **IP-based virtual hosting**

multiple destinations, logging to (Log4J), 542–544

multiple Tomcat instances on one machine, 469–470

MVC (Model View Controller) architecture, 25–26

MX4J (open source JMX), 450–451

`myconfig.file` **certificate generation file**, 252

MySQL

adding JDBC JNDI resource to default context, 306–310

adding `tomcat` user, 350–351

defining a JDBC Realm based on, 351–352

downloading, 304, 349

JNDI-based JDBC configuration, 301–302

read-only user setup, 306

table setup for JDBC authentication, 349–350

test database creation, 304–306

transactions and distributed transaction support, 303

N

name-based virtual hosting

avoiding common mistakes, 374–375

defined, 370

implementing in Apache, 373–374

overview, 372–373

namespaces, unique for class loaders, 199, 201–202

NamingResources object, 439

NAT (Network Address Translation), 496–497

National Center for Supercomputer Applications (NCSA), 2

nested components

JMX manageability, 436–437

Logger in virtual host, 76

Tomcat 4 architecture, 46

Valves as, 162

NetBeans IDE, 554–555

Netscape Directory Server LDAP implementation, 353

Netscape 6/Mozilla browser

cookies and, 469

HTTP Proxy Server (JMeter) configuration, 413

Network Address Translation (NAT), 496–497

NICs (network interface cards), aliasing and, 370–371

“No context exists for path” error, 158

“No context path specified” error, 158

“No global JNDI resources” error, 158

“No user database is available” error, 158

NT event log, logging to (Log4J), 545–548

NT Service component, 34, 36–37

NTFS file system (Windows), 326

O

Observer (JMeter), 401

Once Only Controller (JMeter), 407

open MBeans, 431

Open Source Initiative (OSI), 7

open source JMX (MX4J), 450–451

open source load testers, 419

OpenLDAP protocol, 353, 354–355

OpenLoad load tester, 419

optimization

after load testing, 416–419

Java optimizations, 417–418

premature, ills of, 416

Tomcat optimizations, 418–419

Oracle, JNDI-based JDBC configuration for, 302–303

`org.apache.catalina` **package, Connector interface**, 212, 213, 230

org.apache.catalina.ant **package**, DeployTask **class**, 154–155

org.apache.catalina.connector.warp **package**, WarpConnector **class**, 230

org.apache.catalina.http **package**, HttpConnector **class**, 212

org.apache.catalina.mbeans **package**
GlobalResourcesLifecycle class, 188
ServerLifecycle class, 188

org.apache.catalina.servlets **package**
HTMLManagerServlet class, 151
ManagerServlet class, 151

org.apache.coyote.tomcat5 **package**, CoyoteConnector **class**, 213

org.apache.coyote.tomcat4 **package**
Ajp13Connector class, 230
CoyoteConnector class, 213
CoyoteServerSocketFactory class, 215

OS properties, listing, 144–145

OSI (Open Source Initiative) database, 7

P

packages

restriction by catalina.properties file, 87
split among class loaders, 207

packet structure for protocols, 232–233

passwords

admin role, 55, 131, 322
in Ant scripts, 157
changing for MySQL users, 351
digested (JDBC Realms), 349
digested (UserDatabase Realm), 345–347
manager role, 130, 131, 157, 159, 322
multiple sign-on without the Single Sign-on Valve, 168–169
Single Sign-on Valve and, 166

PATH **environment variable (Windows)**, 30–31

PatternLayout **formatting (Log4J)**, 548–549

performance tuning

clustering and performance, 497–498
for HTTP Connectors, 224–226
for IIS with Tomcat, 289–291
for Log4J, 549
virtual memory use, 391–393

permissions. See also ACLs; Security Manager

in catalina.policy file, 84–87, 332–335
Java permission classes, 330–331
for tomcat account, 326–327, 328

Persistent Session Manager

configuring, 173–175
with file store, 465–466, 482–486
with JDBC store, 466, 487–490
need for persistent sessions, 172–173

pluggable APIs, 8

plug-in workers versus in-process workers, 239–240

pool manager (JDBC), 298

PoolMan pool manager

deploying, 313
downloading, 312
obtaining JDBC connections with JNDI mapping, 317–319
obtaining JDBC connections without JNDI lookup, 315–316
overview, 312
testing with a legacy driver, 316–317
testing with JNDI-compatible lookup, 319
XML configuration file, 313–315

poolman.xml **configuration file**, 313–315

portnumber parameter

deploy command (manager), 136
manager application syntax, 134

ports

for AJP 1.3 Connector, 243, 274, 288, 471–472
for clustering, 471–472
conflicts with HTTP Proxy Server (JMeter), 412
for Connectors, unique numbers for, 47
for load balancing with Apache, 256–257
“port number is in use” error, 43
for SHUTDOWN command, 324
specifying for HTTP Connectors, 212
SSL, 215, 220
for Tomcat on Linux, 40–41
for Tomcat on Windows, 34, 39
for virtual hosting, 379

prepare **target (Ant)**, 567

primordial (bootstrap) **class loader**, 195–197

Principal (JAAS Realm), 358

privileges (MySQL), 350, 351

<project> **element (Ant build files)**, 561

<property> **element (Ant build files)**, 561

PropertyConfigurator **for Log4J**, 532

protocol adapters for MBean Server, 428–429

protocols. See also specific protocols

name in URLs, 14
Web server Connector, 231–233

Provider (JAAS Realm), 358, 359–360

proxy servers. See also JMX proxy

JMeter HTTP Proxy Server, 411–413
 running Tomcat behind, 223–224
 Tomcat access blocked by, 44

proxy Servlet (JMX)

displaying MBeans using, 189
 querying Tomcat internals using, 149–150
 setting Tomcat internals using, 150

public resources, accessing, 92–93

PUT command (HTTP), 137

Q**query string in URLs, 14**

querying Tomcat internals using JMX proxy Servlet, 148

queryjmx utility, 451–454

quick reference (table) (web.xml file), 110–111

R

Random Controller (JMeter), 407

random timers (JMeter), 403

RDBMSs (relational database management systems), 293. See also JDBC; MySQL

Realm component. See also specific Realms

application server configuration, 72
 JAAS Realm, 358–361
 JDBC Realms, 347–353
 JMX manageability, 436
 JNDI Realms, 353–358
 for manager application, 130, 158–159
 overview, 48, 341–342
 security realm users and roles, 342
 standard types of Realms, 72
 UserDatabase Realm, 72, 130, 342–347

<Realm> element (server.xml file)

application server configuration, 72
 described, 70, 76
 for JAAS Realm, 361
 for JDBC Realm, 348–349, 351–352
 JMX-manageable attributes, 436
 for JNDI Realms, 357

Recording Controller (JMeter), 408

recovery options (Windows), 36–37

redirection by balancer filter, 490–491, 496

regedit command (Windows), 275

Registry (Windows), updating for the ISAPI filter, 275–277

regular expressions with JMeter load testing software, 412

relational database management systems (RDBMSs), 293. See also JDBC; MySQL

reload command (manager), 142–143, 152

“Reload not supported on WAR...” error, 158

Remote Address Filter, 170

Remote API (JMX), 430

remote debugging

with Eclipse, 554
 IDE support for, 551
 with IntelliJ IDEA, 551–552
 with JBuilder, 556–557
 with NetBeans, 555
 starting Tomcat in debugging mode, 551–552

Remote Host Filter, 170

remotely deploying Web applications, 139–140

remove command (manager), 146–147, 152

removing. See deleting or removing

request distribution, hardware-assisted with NAT, 496–497

Request Dumper Valve, 49, 163, 172

request dumping, defined, 49

Request Filter Valve, 163, 170–172

request processors, 443

request sharing in Tomcat/Apache virtual hosting, 385–386

request.getServerName() function, 214, 223, 377

request.getServerPort() function, 214, 223

<Resource> element (server.xml file)

attributes, 180–181
 described, 179, 180–181
 JavaMail session configuration, 184
 JDBC DataSource configuration, 182–183
 JMX-manageable attributes, 439–440
 JNDI-based JDBC configuration, 301–302, 306–307

Resource object, 439–440

<resource-env-ref> element (web.xml file), 106–107, 120

<ResourceLink> element

manager.xml file, 132, 143, 144, 158
 server.xml file, 179, 181–182, 440

ResourceLink object, 440

<ResourceParams> element (server.xml file)

attributes, 181
 described, 179, 181
 JavaMail session configuration, 184
 JDBC DataSource configuration, 183
 JNDI-based JDBC configuration, 301–303, 306–308

<resource-ref> element (web.xml file)

<resource-ref> **element** (web.xml file)

JavaMail session configuration, 185
JDBC DataSource configuration, 184
JNDI-based JDBC configuration, 308
Servlet 2.3-style, 106–107
Servlet 2.4-style, 120–121

resources **command** (manager), **143–144**

response time, clustering and, **498**

reusability, MVC architecture and, **26**

Role object, **438**

roles **command** (manager), **144**

Root Certificate, **365**

ROOT **Web application**, **92, 321, 322**

round-robin behavior

for load balancing with sticky sessions, 463–464
testing the Apache load balancer, 267–269

RPMs, installing JVM from, **32–33**

rules.xml file for balancer filter, **494–495**

S

Sampler feature (JMeter), **408**

scalability. See also load testing

clustering and, 457, 458–459
defined, 395, 420
distributed load testing and, 413
establishing limitations, 416
horizontal, 457
performance versus, 395
types of, 458–459

scaling out, **459**

scaling up, **458**

seamless sessions. See sticky sessions

Secure Sockets Layer. See SSL

security. See also authentication; passwords;

SSL (Secure Sockets Layer)

for admin configurator, 55, 322–323
applications shipped with Tomcat and, 321–323
for class loaders, 200–202
file system, 326–328
invoker Servlet and, 78
JMX issues, 454
JVM, 328–337
listing available roles, 144
manager application considerations, 158–159,
322–323
manager role constraints, 132–133
running Tomcat with a special account, 324–326
Servlet container provisions, 16
SHUTDOWN command and, 323–324

virtual hosting and, 387

Web application

authentication and Realms, 337–362
encryption with SSL, 362–367
host restriction, 368

Security Manager

Catalina code permissions (Tomcat), 333–334

dangers from not using, 335

enabling class loader creation, 336

enabling for Tomcat, 332

enabling the Security Manager system, 332

explicit permission required by, 328

file system security, 336–337

further information, 330, 332

grant entry syntax, 329–330

granting permissions to applications, 329

JavaMail and, 336

overview, 328–332

permissions available, 330–331

recommended practices, 335–337

system code permissions (Tomcat), 332–333

Tomcat policy file, 332–335

using with Tomcat, 332–335

Web application permissions (Tomcat), 334–335

Web application requirements, 335

security realms. See Realm component; specific Realms

<security-constraint> **element** (web.xml file)

authentication and, 340

for manager role, 133

for multiple sign-on, 167

Servlet 2.3-style, 107–108

Servlet 2.4-style, 121

for SSL protection of resources, 366

SecurityManager **class**, **202**

<security-role> **element** (web.xml file), **109, 122, 133**

Segue's Silk Performer load testing software, **419**

sendmail.jsp, **186**

SendMailServlet, **compiling and configuring**,
185–186

Server component, **47, 62–63, 433**

server **directory**, **42**

<Server> **element** (server.xml file)

attributes, 63

for balancer filter and standalone mode, 493

JMX-manageable attributes, 433

for load balancing with Apache, 257

overview, 62

subelements, 63

server load testing. See load testing

server proximity, distributed load testing and, 413

serverinfo **command** (manager), 144–145

ServerLifecycle **class**, 188

server-minimal.xml **file**, 59

server-noexamples.xml.config **file**, 59

server-side includes (SSI), 79, 82, 222–223

server-side sessions. See sessions; sticky sessions

server.xml **file. See also specific elements**

for CGI support, 220–221

for clustering, 471–472

components nested in default configuration, 61

Connector port specification, 212

Coyote HTTP/1.1 Connector configuration

Tomcat 4.1, 213–215

Tomcat 5, 216–219

default configuration listing, 61

default server configuration stored in, 60

deploying Web applications manually, 130

described, 52

for in-memory session replication, 477–479

for JAAS Realm, 361

for JDBC Realm, 348–349, 351–352

for JNDI Realms, 357

for JNDI-based JDBC configuration, 301–303, 306–308

for load balancing with Apache, 256–258, 267

manager application configuration, 132

for MX4J (open source JMX), 451

optimization, 418–419

overview, 58–59

for Persistent Session Manager with file store, 483

for Persistent Session Manager with JDBC store,
487–488

for running Tomcat behind a proxy server, 223–224

for separate JVMs, 388–389

for SSI support, 222–223

for SSL protection of resources, 366–367

for standalone server, 378–380, 491–492

Tomcat 5 versus earlier versions, 60

for Tomcat with Apache, 243

for Tomcat with IIS, 275

server.xml.[date suffix] **file**, 59

Service component, 47–50, 64–65, 432

<Service> **element** (server.xml file)

attributes, 65

JMX-manageable attributes, 432

overview, 64

as subelement of <Server> element, 63

subelements, 65

for virtual hosting, 378

<service-ref> **element** (web.xml file), 122–123

Servlet containers

cookie use by, 15–16

defined, 15

migration issues, 16

overview, 15–16

Tomcat as, 1, 16, 502

in Web containers, 20–21

<servlet> **element** (web.xml file)

for CGI support, 220–221

JMX-manageable attributes, 442

overview, 77–80

Servlet 2.3-style, 102–103

Servlet 2.4-style, 123–124

for SSI support, 222–223

<servlet-mapping> **element** (web.xml file)

for CGI support, 220–221

overview, 80–82

Servlet 2.3-style, 93–94

Servlet 2.4-style, 124

for SSI support, 222–223

Servlets. See also specific Servlets

accessing, 16–18

for CGI support, 220

default definitions in web.xml file, 77–80

defined, 14

destruction, 15

initialization, 14

JMX manageability, 442

JSP compilation into Servlet classes, 19

JSPs versus, 19

limitations, 17–18

logical mapping, 17, 80–82

mappings in default web.xml file, 80–82

overview, 14–15

Tomcat as container for, 1, 16, 502

servlets-examples **Web application**, 174–175,
322, 323

session affinity. See sticky sessions

session sharing

in-memory replication, 466–468, 475–482

with JDBC-based store, 466, 487–490

overview, 464

with shared file store, 465–466, 482–486

<session-config> **element** (web.xml file), 82,
103, 124

sessions. See also Persistent Session Manager; sticky sessions

displaying statistics for Web applications, 148

JavaMail, configuring, 184–187

management in Tomcat 5, 468

sessions (continued)

- session sharing, 464–468
- timeout configuration, 82
- for tracking state information, 461, 468

sessions command (manager), 148

SET command (MySQL), 351

setting Tomcat internals using JMX proxy Servlet, 150

SHA algorithm, 339, 345

shared class loader, 204

shared directory, 42

shared hosting. See also virtual hosting

- expectations for Tomcat, 375
- fine-tuning, 386–393
- JMX and, 422–423
- standalone versus cooperative mode and, 375
- Tomcat versions and, 369

SHM (Shared Memory) file for workers2.properties file, 245, 278

shut_sites.sh script, 391

shutting down or stopping

- SHUTDOWN command security issues, 323–324
- shutting down embedded server, 519
- shutting down the Tomcat cluster, 471
- Web applications, 145, 152

sign-on. See also authentication

- multiple, without the Single Sign-on Valve, 166–169
- Single Sign-on Valve for, 163, 166, 169

Silk Performer load testing software (Segue), 419

Simple Controller (JMeter), 406–407

Simple Data Writer listener (JMeter), 403, 405

Simple Network Management Protocol (SNMP), 424

SimpleTcpCluster implementation, 476

Single Sign-on Valve

- for authentication, 163, 166, 169, 362
- configuring, 169
- described, 163
- multiple sign-on without, 166–169
- overview, 166
- Tomcat 4 architecture, 49

singleton classes, class loaders and, 207–208

slash (/)

- in context path for manager application, 134, 157
- for protecting manager URLs (!/), 134–135

SMTP Server, logging to (Log4J), 544–545

SNMP (Simple Network Management Protocol), 424

socketBuffer attribute of Connectors, performance and, 225

Spline Visualizer listener (JMeter), 403, 404

SSI (server-side includes)

- default Servlet definition, 79
- enabling support for, 222–223

- security issues, 222
- Servlet mappings, 82

ssi Servlet in default web.xml file, 79

SSL (Secure Sockets Layer)

- configuration file for certificate generation, 252
- configuring for Tomcat with Apache, 250–254
- configuring for Tomcat with IIS, 291
- configuring Tomcat 4.x Connector for, 215–216
- configuring Tomcat 5.x Connector for, 219–220
- Connector, 47
- encryption with, 362–367
- generating a test certificate, 253–254
- JSSE implementation, 363–365
- name-based virtual hosting and, 374
- port for, 215, 220
- for resource protection, 366–367
- testing the Apache-Tomcat setup, 254–255

stack size, virtual hosting and, 391

standalone mode

- application server mode versus, 65–66
- for balancer filter, 492–493
- components in standalone server, 377–378
- Connectors and, 65–66
- cooperative mode versus, 375
- Tomcat 5 as standalone server, 378–381
- virtual hosting and, 375, 377–381

standard MBeans, 431

start command (manager), 146, 152

starting

- IIS server, 273
- JMeter, 396
- minimal embedded server, 517
- stopped Web applications, 146, 152
- Tomcat as a service on Windows, 35–36
- Tomcat in debugging mode, 551–552
- Tomcat manually on Windows, 35
- Tomcat with JMX Remoting enabled, 451
- troubleshooting Tomcat, 43–44

start_sites.sh script, 390

startup command (Windows), 35

startup.bat file (Windows), 37–38

static Web content, 11

status command (manager), 151

sticky sessions. See also Persistent Session Manager

- defined, 255
- mod_jk2 Apache module and, 463–464
- with no session sharing, 464–465
- overview, 463–464
- with Persistent Session Manager
- JDBC-based store, 466, 487–490
- shared file store, 465–466, 482–486
- testing Apache load balancer for, 266–267

stop **command** (*manager*), 145, 152
stopping. *See* **shutting down or stopping**
`<Store>` **element** (*server.xml* file), 174, 483, 487–488
Struts Framework, 5, 22–24, 25–26
Subject (JAAS Realm), 358
support network (Tomcat), 44
Swing interface (JMeter), 397
Sysdeo Tomcat plug-in, 553–554
system class loader, 196–197, 203
system design, embedded mode in Tomcat and, 502–504
system logs, logging to (Log4J), 545–548

T

tag libraries (JSP), 21–24, 94–95
`<taglib>` **element** (*web.xml* file), 106
`<target>` **element** (*Ant* build files), 561
targets (Ant), 155–157, 567–569
tar/gzip files, installing JVM from, 31–32
TCP socket communication channel for
workers2.properties file, 245, 278
tcpNoDelay **attribute of Connectors, performance and**, 224, 225
test plans (JMeter), 397–401, 406–407
thrashing, 392
thread groups (JMeter), 397–400
thread pooling
 for HTTP Connector performance, 225, 226
 JMX manageability, 444, 447–450
ThreadPool object, 444
Throughput Controller (JMeter), 408
timeout, 82, 290
Timer feature (JMeter), 402–403
tlds directory, 94–95
tomcat account
 adding *tomcat* user to MySQL, 350–351
 configuring, 325–326
 creating, 324
 file system security, 326–328
 for Linux, 324, 325–326, 328
 running Tomcat with, 324–326
 for Windows, 324, 325, 326–327
tomcat-docs Web application, 321, 322
tomcat-sessions table, 488–490
tomcat.sh file (Linux), 32
tomcat-users.xml file
 authentication and, 59, 77
 enabling *manager* application access, 130–131

manager application configuration, 132
 overview, 59
top-level components, 46, 47–50. *See also specific components*
troubleshooting. *See also error messages*
 class loaders, 206–208
 IIS rendering JSPs, 287–288
 IP-based virtual hosting, 372
manager application errors, 157–158
 name-based virtual hosting, 374–375
 online resources for, 43
 Tomcat installation, 43–44
 Tomcat support network, 44
Type I to Type IV JDBC drivers, 296–297

U

undeploy command (manager), 147–148
uniform random timer (JMeter), 403
Unix. *See* **Linux**
updatability, MVC architecture and, 26
URI mappings for *workers2.properties* file, 246, 279
`<url-pattern>` **element** (*web.xml* file), 81, 167, 168
URLs. *See also* **downloading**; **HTTP requests with**
manager application; **Internet resources**
 for admin configurator, 55
 for CGI programs, 14
 for *manager* JMX proxy access, 445
 overview, 14
 parsing in Tomcat 5, 81–82
 protecting *manager* URLs (!/), 134–135
 for ROOT Web application, 92
 for Servlet access, 16–17
 Servlet mappings, 80–82, 93–94
 for Web application example, 129
 Web application mappings, 93–94
user accounts, 324–326
user lists (Tomcat), 44
user names
 admin role, 55, 131, 322
manager role, 130, 131, 322
User object, 438
UserDatabase object, 438
UserDatabase Realm
 adding the digested password, 346–347
 configuring, 343–345
 creating a digested password, 345–346
 described, 72
 as a file-based Realm, 342

UserDatabase Realm (continued)

- JMX manageability, 438
- for `manager` application, 130
- MemoryRealm versus, 342–343
- selecting the digest algorithm, 345
- steps for securing, 345
- testing the digested password, 347

`<user-data-constraint>` **element** (`web.xml` file), **108**

`user_roles` **table** (JDBC Realm), **347, 348**

users, creating with `admin` configurator, **56, 58**

user's session token, 15

`users` **table** (JDBC Realm), **347–348**

V

validating load testing data using assertions, 410–411

`<Valve>` **element** (`server.xml` file)

- Access Log Valve configuration, 163, 166
- described, 70
- for in-memory session replication, 479
- Single Sign-on Valve configuration, 169
- for virtual hosting, 379

Valves. *See also specific types*

- Access Log Valve, 163–166
- functionality from, 162
- JMX manageability, 436
- for `manager` application, 159
- as nested components, 162
- overview, 48–49, 162
- Request Dumper Valve, 49, 163, 172
- Request Filter Valve, 163, 170–172
- Single Sign-on Valve, 49, 163, 166–169
- standard, 162–163
- in Tomcat 5 architecture, 162

vendor neutrality for application servers, 10

View component of MVC architecture, 25

View Results in Table listener (JMeter), 403, 405

View Results Tree listener (JMeter), 403, 405

viewing. *See displaying or listing*

virtual directories under IIS, 280–283, 287

Virtual Directory Creation wizard (Windows), 280–282

virtual hosting. *See also shared hosting*

- application server configuration, 72–76
- approaches, 370
- basic concept, 73
- directory structure, 376, 377, 385, 387–388
- fine-tuning, 386–393
- IP-based, 370–372
- JMX and, 422–423

- memory problems with multiple JVMs, 391–393
- name-based, 370, 372–374
- nested Logger in virtual host, 76
- request sharing between Tomcat and Apache, 385–386
- security issues, 387
- separate JVMs for hosts, 387–391
- with Tomcat
 - components in standalone server, 377–378
 - directory structure, 376
 - example configuration, 375–377
 - IP addresses, 375
 - standalone versus cooperative mode and, 375
 - Tomcat 5 as standalone server, 378–381
 - Tomcat 5 with Apache, 381–386
 - Tomcat 4 architecture, 49
 - in Tomcat 5, 73
 - Tomcat versions and, 369
 - Web site defined for, 370

virtual memory, performance and, 391–393

`<VirtualHost>` **section** (`httpd.conf` file)

- for IP-based virtual hosting, 371, 372
- for name-based virtual hosting, 373–374
- for Tomcat with Apache, 382–383

visualization listeners (JMeter), 404

W

WAR files

- creating using Ant `dist` target, 568, 569
- deploying Web applications from, 136–137, 140–141
- deploying Web applications manually, 130
- directories for, 376
- installing Web applications from, 137–138, 140
- `manager deploy` command and, 136, 137, 140
- `manager install` command and, 137, 138
- MANIFEST.MF file creation and, 96

WARP protocol, 232–233, 234

WarpConnector **class**, 230

Web applications. *See also* `admin` configurator **application**; **configuring Web applications**; `manager` **application**

- Ant tool for managing, 154–157
- architecture, 24–25
- building with Ant, 566–570
- class loaders, 205–206
- configuring minimal applications, 471
- context definitions, 76–77
- context path unique for, 135, 138, 157
- default deployment descriptor (`web.xml`), 77–81

- deploying
 - installing versus, 136
 - from a local path (Tomcat 5), 140–141
 - manually, 130
 - redeploying, 142
 - remotely (Tomcat 5), 139–140
 - in Tomcat 4, 136–137
 - in Tomcat 5, 139
 - using `manager` Web interface, 153
- directory structure, 91–92
- displaying Tomcat server status, 151
- examples, 34, 128–129
- installing
 - deploying versus, 136
 - in Tomcat 4, 136, 137–139
 - in Tomcat 5, 139–142
- listing
 - deployed applications, 135
 - JNDI resources available, 142–143
 - OS and JVM properties, 144–145
 - security roles available, 144
- logging from (Log4J), 535–538
- optimization techniques, 416–419
- permissions in `catalina.policy` file, 85–87, 334–335
- redeploying, 142
- reloading, 142–143, 152
- removing
 - permanently (Tomcat 5), 147–148
 - temporarily (Tomcat 4), 146–147
 - Web interface for, 152
- `ROOT` application, 92, 321, 322
- security
 - authentication and Realms, 337–362
 - encryption with SSL, 362–367
 - file system security, 336–337
 - host restriction, 368
 - Single Sign-on Valve, 362
 - users and roles, 342
- session statistics, 148
- shipped with Tomcat, 321–323
- starting a stopped application, 146, 152
- stopping, 145, 152
- `tomcat-docs` application, 321, 322
- undeploying, 147–148
- Web interface for managing, 151–152
- Web containers, 20–21**
- Web interface** (`manager`)
 - Applications table, 152
 - deploying Web applications, 153
 - displaying Tomcat server status, 151
 - managing Web applications, 151–152
 - overview, 150–151
- Web server Connectors. See also specific Connectors**
 - AJP protocol, 231–232
 - Apache JServ Connectors, 233–234
 - architecture, 230–233
 - choosing a Connector, 233–235
 - communications paths, 230–231
 - configuring the AJP 1.3 Connector, 243
 - HTTP Connectors versus, 211
 - JK Connectors, 234
 - JK2 Connectors, 67–68, 234–235, 237, 238–239
 - reasons for using a Web server, 229–230
 - WARP protocol, 232–233
 - `webapp` Connectors, 234
- Web servers. See also Apache Server; IIS; virtual hosting**
 - combining Tomcat with, 418
 - JK Connector support for, 234
 - Netcraft survey of, 370
 - Tomcat integration with, 11
- Web sites. See downloading; Internet resources**
- `webapp` **Connectors, 234**
 - `<web-app>` **element** (`web.xml` file), 99, 110, 340, 472–473
- `webapps` **directory, 42**
- Web-based configurator. See admin configurator application**
- Web-based manager. See manager application**
- `WEB-INF` **directory, 94–95**
- `WebModule` **object, 440–442**
- `<web-resource-collection>` **element** (`web.xml` file), 108
- `web.xml` **file. See also specific elements**
 - for authentication, 340
 - for balancer filter, 493–494
 - CGI mappings, 82
 - for clustering, 472–473
 - default file versus Web application file, 77
 - default Servlet definitions, 77–80
 - described, 52
 - JavaMail session configuration, 185
 - JDBC DataSource configuration, 184
 - JNDI-based JDBC configuration, 308
 - for `manager` application, 132–133
 - mime mappings, 82–83
 - for multiple sign-on, 167–168
 - overview, 59, 77
 - Servlet 2.3- versus 2.4-style, 96–97

web.xml file (continued)

Servlet 2.3-style

- DTD declaration, 99
- elements, 99–110
- generalized form, 97
- order of elements, 97
- quick reference (table), 98
- XML header, 99

Servlet 2.4-style

- elements, 112–124
- order of elements, 110
- quick reference (table), 110–111

Servlet mappings, 80–82, 93–94

session timeout configuration, 82

SSI mappings, 82

for SSL protection of resources, 366

in WEB-INF directory, 94

welcome page handling, 83–84

welcome page handling, 83–84, 93

`<welcome-file-list>` **element** (web.xml file),
104–105, 124

Windows (Microsoft). *See also* IIS (Internet Information Server)

- building `mod_jk2` Apache module on, 242
- copying SSL files for Apache to, 252
- digest utility bug fix, 346
- file system security, 326–327
- installing JVM on, 29–31
- installing Tomcat using Windows Installer, 33–34, 131
- installing Tomcat using ZIP file, 39–40
- JSSE certificate keystore preparation, 363
- running JMeter, 396
- starting Tomcat in debugging mode, 552
- testing the Tomcat installation, 34–39

tomcat account for, 324, 325

updating the Registry for the ISAPI filter, 275–277

work directory, 43

workers (Tomcat instances). *See also specific types*

configuring for Tomcat with Apache, 243–247

configuring for Tomcat with IIS, 277–280

defined, 239

for load balancing with Apache, 258–265

multiple, 240

plug-in versus in-process, 239–240

ports for load balancing with Apache, 257

types of, 240–241

in virtual hosting example, 382

workers2.properties file

configuring for Tomcat with Apache, 244–246

configuring for Tomcat with IIS, 277–280, 288

examples, 246–247, 263–265, 279–280

format, 243–244

1b worker configuration, 261–263

for load balancing with Apache, 259–265

for virtual hosting with Tomcat, 383, 384, 385–386, 390

W3C (World Wide Web Consortium) Web site, 14

X

XA transaction operation model, 303

Xerces XML parser, 3, 208

XML header in web.xml file, 99

XML parsers, 208. *See also* JAXP; *specific parsers*

-Xms and -Xmx settings. *See* heap size

Z

ZIP files, installing Tomcat from, 39–40