

Index

Symbols and Numbers

- # (hash) sign, meaning of, 14
- \$ (dollar sign) metacharacter in Perl, action of, 760
- \$\$ (double dollar sign), using with map sed script, 68
- *\$ (dollar sign and asterisk), using with .class and .java files, 571
- \$< (dollar sign and less-than symbol) in make utility, meaning of, 568
- \$1 variable, using with map.sed script, 68
- \$BASE_DEV directory after copying beans, illustration of, 491
- \$JAVA_HOME environment variable, using with setclasspath.sh, 55
- %0 notation, meaning of, 62
- %C format descriptor, 669
- & (ampersand), using with startupServer.sh script in WAS, 92–93
- ' (apostrophe) operator, meaning in .xml documents, 353
- (...) (ellipsis)
 - metacharacter in Perl, 760
 - output in Perl, 179
 - purpose in wscp_Servlet-list.pl program, 178
- () (parentheses)
 - meaning in .xml documents, 353
 - purpose in sample shell script, 343
- * (asterisk)
 - metacharacter in Perl, 760
 - operator in .xml documents, 353
- * (wildcard general alias), advisory about, 97
- + (plus sign)
 - versus ; (semicolon), 291
 - metacharacter in Perl, 760
 - operator in .xml documents, 353
- ./ (dot slash), adding to classpath, 49
- . (dot)
 - metacharacter in Perl, 760
 - using with classpath, 49
 - using with setj.sh script, 53
- / (forward slash), using with Windows NT, 80
- : (colon)
 - versus ; (semicolon), 67
 - purpose in UNIX, 230
 - using with classpath, 49, 54
- ; (semicolon)
 - versus + (plus sign), 291
 - versus : (colon), 67
 - using with classpath, 49, 54
- <% %> (angle brackets and percent signs), tagging Java code between, 322
- > (greater than symbol), meaning in comparison of list1.txt to list2.txt, 347
- ? (question mark)
 - metacharacter in Perl, 760
 - operator in .xml documents, 353

770 IBM WebSphere Application Server Programming

- @ (at) character, appearance before WASMON internal variables, 708
 - \ (backslash)
 - metacharacter in Perl, 760
 - separator used with Windows NT PATH, 80
 - \\; (escaped semicolon), using with Windows NT PATH environment variable, 56
 - \\. (slash dot) notation, meaning of, 68
 - ^ (caret)
 - metacharacter in Perl, 760
 - using with WASMON, 660–661
 - { } (curly braces), purpose in WSCP, 172
 - | (pipe)
 - metacharacter in Perl, 760
 - operator in .xml documents, 353
 - 0000, advisory about use with WASMON, 661
 - 0-8 fields used in SharkUrl example, meaning of, 610
 - 4k+ page sizes for sessions, managing, 450–451
 - 8k session size, setting buffer pool for, 450
 - 9001-9999, advisory about use with WASMON, 661
- ### A
- A in WAS LEDs, meaning of, 647
 - AAT (Application Assembly Tool)
 - assembling EJB module with, 496–498
 - enabling `serveServletsByClassname` with, 348
 - modifying `petstore.ear` file with, 155–156
 - overview of, 139
 - role in deprecating Web Resources in WAS v4, 100
 - using with IBM extensions, 344
 - verifying EJB module with, 482–483
 - AAT navigation pane, structure of, 345
 - AATL WAS component ID, description of, 647
 - AC (application classloaders), directory location of, 359
 - `accept_all_cookies` Lynx option, description of, 386
 - `activity.log` file, overview of, 199–200
 - AdditionalClassPath IBM extension attribute, description of, 351
 - `adduser` command, using on UNIX, 578
 - ADGU WAS component ID, description of, 647
 - `admin_host`, setting to "localhost" for protection of WAS AE v4, 97–99
 - administration server problems, troubleshooting, 145
 - Administration Server screen, displaying, 28
 - administrative configuration files, overview of, 210–211
 - administrative console
 - enterprise application in, 146
 - explanation of, 20, 82
 - using with WSCP, 171
 - viewing `web.xml` deployment descriptor from, 269
 - administrative data, explanation of, 20–21
 - administrative practices for WAS
 - archiving WAS directory, 218–219
 - archiving WAS directory on Windows NT, 219–220
 - checking on repository, 217
 - invoking `showCfg`, 216–217
 - monitoring application archives, 217–218
 - administrative repositories
 - advisory about re-creation of, 214

- databases used as, 20–21
- re-creating, 215
- ADMR WAS component ID, description of, 647
- ADMS WAS component ID, description of, 648
- ADMT WAS component ID, description of, 647
- AE (application extensions)
 - directory location of, 359
 - example of, 360, 362
- AIX (Advanced IBM UNIX)
 - clone() system call used by, 205
 - confirming installation of IHS on, 41
 - definition of services in, 205
 - IHS installation directory for, 39–40
 - installing System Agent on, 667–668
 - installing UDB on, 29–32
 - Java machine installation for, 46
 - performance commands on, 666–667
 - pstat command used with, 199
 - system profile and user profile in, 50
 - tabulation of setw(v)(os).sh script in, 107
 - thread statistics on, 678–680
 - uninstalling WAS on, 111
 - using MrThread tool with, 680–683
 - vmstat report on, 672
- AIX server, sample Nanny process on, 129
- ALERT directives used in WASMON v1.2.1
 - configuration file, purpose of, 755
- alerts, e-mailing, 659–662
- Alias directive, using with Apache server, 87
- aliases
 - advisory about, 108
 - checking for resolution of, 102
- ALRM WAS component ID, description of, 648
- ampersand (&), using with startupServer.sh script in WAS, 92–93
- angle brackets and percent signs (<% %>), tagging Java code between, 322
- Apache HTTP Server, modifying
 - httpd.conf file in, 87
- Apache SOAP. *See also* SOAP (Simple Object Access Protocol)
 - providing web services to WASDG application with, 500
 - recognizing serialization with, 511
- Apache SOAP deployment descriptors. *See also* deployment descriptors
 - generating with gensoap-ejb, 508–511
 - overview of, 506–508
 - printing with gensoap-ejb, 511
 - specifying type mappings in, 513–514
- Apache SOAP User's Guide, code distribution web site for, 727
- Apache SOAP wasdgosoap.ear
 - application, deploying, 500–505
- apostrophe (') operator, meaning in .xml documents, 353
- app driver, overview of, 33–34
- AppBuild script, building and deploying, 576–578
- applet clients for WAS, overview of, 143
- appletviewer command, explanation of, 47
- APPLHEAPSZ UDB parameter
 - default value for, 689
 - explanation of, 691
- application archives, monitoring with WAS, 217–218
- Application Client Resource Configuration Tool, starting GUI for, 107
- application server process, setting execution property of, 119

772 IBM WebSphere Application Server Programming

- application servers
 - detecting problems with DrAdmin utility, 138–139
 - testing WAS with, 192–193
 - using models and clones with in WAS v4, 133–134
 - application.xml file type, location in WAS v4, 105
 - applParseWscp program, code for, 180–183
 - AppMaker script, code for, 572–574
 - AppMaker1 script, code for, 568–570
 - APPNAME column in SESSIONS table, details of, 434
 - arayech machine, synchronizing time on, 586–587
 - asterisk (*)
 - metacharacter in Perl, 760
 - operator in .xml documents, 353
 - at (@) character, appearance before WASMON internal variables, 708
 - autoFlush page directive attribute, value and description of, 338
- ### B
- b-var WASMON boolean internal variables
 - versus directives, 713
 - list of, 711
 - purpose of, 708–709, 758–759
 - using in logical expressions, 716–717
 - backslash (\)
 - metacharacter in Perl, 760
 - separator used with Windows NT PATH, 80
 - BASE_DEV directory
 - Makefile program for, 571–576
 - using with symbolic links to ejb directory, 493
 - BASE_DEV environment variable
 - setting for use with IBM extensions, 343
 - setting for use with J2EE web applications, 264
 - \$BASE_DEVBEAN directory
 - cleaning up classes after changing beans in, 482
 - maintaining after modifying code in, 493–494
 - base.properties file, using with BundleManager, 627
 - .bash_profile hidden file, purpose of, 50
 - batch file example, 62
 - batch processing
 - benefits of, 69
 - with Lynx, 317–318
 - BeanBase, extending to communicate returned objects, 473
 - beans, accessing with SOAP clients, 520–521. *See also* EJB entries
 - beans nomination, regulating, 479–481
 - Been There servlet, description of, 144
 - .bff files in UDB distributions, purpose of, 22
 - BFS script, Apache SOAP deployment descriptor artifact for, 507
 - binary distribution of WAS products, creating directory for, 582
 - @BODY WASMON s-var (static internal variable), content of, 710
 - @BOOL_* WASMON b-vars (boolean internal variables)
 - versus directives, 713
 - explanation of, 711, 758–759
 - Bourne versus Korn shell environment variable, 56
 - BruteConn.java servlet. *See also* servlets

- compiling, 295
- testing connectivity to DBGUIDE
 - database with, 293–295
- @BUFF WASMON g-var (global internal variable), content of, 712
- buffer page directive attribute, value and description of, 338
- buffer pools in SESSION database
 - enlarging, 690
 - getting number of pages used in, 690
- BUFFPAGE UDB parameter, default value for, 689
- build process for packages, understanding, 258–260
- BundleManager file
 - advantage of preprocessing properties files with, 630
 - bundling information with, 627–633
 - code for, 628–630
 - testing program for, 639
- BundleManager.getString() method, purpose of, 622
- @BVAR WASMON g-var (global internal variable), content of, 712
- BwjSort script
 - overview of, 750–751
 - interpreting SharkUrl reports with, 613
- C**
- C <cookie> option of ZappUrl stress-tester, description of, 600
- c <n> option
 - of SharkUrl stress-tester, 607
 - of ZappUrl stress-tester, 600
- C option of jar command, example of, 342
- call command, using with DOS batch files, 62
- canonical names, role in Java 2 encoding, 744
- capture directory, running in List.jsp example, 322–323
- .car file type, location in WAS v4, 105
- caret (^)
 - metacharacter in Perl, 760
 - using with WASMON, 660–661
- @CC WASMON s-var (static internal variable), content of, 710
- CCCC suffix for WAS components, meaning of, 646
- Cfile program, demonstrating Cipher encryption with, 554–555
- CGI environment variables
 - dumping with DumpEnv.java servlet, 286–288
 - dumping with printenv.sh script, 285
 - printing with printenv.cgi program, 284–285
- CGI programs, WAS session identifier visibility to, 408–409
- CGI script for Hello World!, code for, 72
- CGI scripts
 - reading cookies in, 389–390
 - setting cookies with, 387–389
- character sets, printing information about using MrUnicode, 742–744
- chekenv script, printing environment variables with, 748–749
- chfs command, allocating free disk space with, 32
- CHKJ WAS component ID, description of, 648
- CHKW WAS component ID, description of, 648
- ci -l command, using in Makefile program, 576
- Cipher encryption, overview of, 554–560
- class casting, role in JAAS, 552–553

774 IBM WebSphere Application Server Programming

- .class files
 - generating from .java files, 571
 - locating with Java programs, 49
- .class.jar (Servlet) file type, location in WAS v4, 105
- class reloading
 - example using module visibility, 370–377
 - explanation of, 366
- classes, cleaning up after changing beans in \$BASE_DEVBEAN directory, 482
- classloading hierarchy of WAS, diagram of, 359
- classloading in WAS, guidelines for, 359
- classpath definitions, searching configuration files contained in, 217
- classpath elements, specifying, 359
- CLASSPATH environment variable
 - adding .jar files to set SOAP client environment, 520
 - advisory about, 49
 - components of, 55
 - description of, 48
 - overview of, 48–49
 - setting globally, 54–55
 - setting with SETJAVA.BAT and SETCLASS.BAT, 62
- classpaths. *See also* WAS startup classpath
 - adding current directory to, 49
 - getting from within servlets, 364–365
 - overriding, 48
 - printing for PrintCp servlet, 369
 - purpose of, 49
 - testing proper addition of JAAS to, 526
 - tree hierarchy of, 359–362
- CliAcct object
 - retrieving data from, 252–258
 - type mappings for, 514
- CLIACCNT table in DBGUIDE database
 - description of, 225
 - query result for, 297
- client balances, crediting zero dollars to, 316–317
- client QName(), matching to server mappings, 518–519
- clientConfig tool, overview of, 142–143
- CLIENTID, prompting in CreditDebit servlet, 312–313
- CLIENTID, role in DBGUIDE database, 228
- clients, iterating and forking processes for, 598–599
- CLIPWD table in DBGUIDE database, description of, 225
- clock synchronization, confirming with ntpdc command, 588
- clock, synchronizing for use with make utility, 586
- CNTR WAS component ID, description of, 648
- code distribution web site, Internet addresses for, 726–727
- colon (:)
 - purpose in UNIX, 230
 - versus semicolon (;), 67
 - using with classpath, 49, 54
- com/tcnd/hellos/C1.java class, code for, 67
- com/tcnd/hellos/C2.java class, code for, 68
- com/tcnd/wasdg/common directory for logging and exception-handling packages, contents of, 621
- com/tcnd/wasdg/dbreg directory, description of, 537
- com/tcnd/wasdg/log directory for logging and exception-handling packages, contents of, 621
- COM.ibm.db2.jdbc.app.* driver, purpose of, 33–34
- COMM_FAILURE exception, generation of, 206
- command prompt, manipulating DOS variables from, 62

- commands
 - setting in WAS development environment, 105–109
- common resources, configuring for WAS, 93–94
- components.properties file, using for exception handling, 632
- comtcmd/wasdg/wasdgexceptions directory for logging and exception-handling packages, contents of, 621
- com.tcmd.hellos package, changing to com.ibm.sos.hellos, 68
- com.tcmd.wadg.dbreg package, tree structure for, 538
- com.tcmd.wasdg.common.DataSet class, data types used in, 515
- configuration data, storage of, 20
- CONM WAS component ID, description of, 648
- Connection class, obtaining metadata for DBGUIDE database with, 230–233
- connectionProperties, managing for Sybase UNICODE character set, 216
- connect.java program
 - code for, 73–74
 - testing, 74
- CONTACT field in Employee Records, methods of, 257
- contentType page directive attribute, value and description of, 338
- context init parameters
 - changing webgenxml script for accommodation of, 302–303
 - inserting into web.xml file, 301
- <context-param> (*) element, meaning in web applications, 353
- context root, setting for Petstore sample, 159
- converters, printing information about using MrUnicode, 742–744
- cookie information, logging by redirecting System.out, 396–399
- cookie_file Lynx option, description of, 386
- cookie_save_file Lynx option, description of, 386
- cookies
 - with CGI, 387–391
 - deleting from browsers using Lynx, 395
 - disabling, 417
 - enabling in WAS AEs
 - administrative console, 399
 - fetching, 395–396
 - hash values of, 390–391
 - JSESSIONID, 400–407
 - limitations and naming conventions of, 399
 - Lynx option, description of, 386
 - overview of, 386
 - reading in CGI scripts, 389–390
 - setting through POST method, 392–395
 - using through HTTP Java servlets, 391–399
- COUNTER field in Employee Records, methods of, 257
- cpfl script
 - copying files with, 490–492
 - overview of, 751
- cpio command, backing up WAS installation with, 218
- CPU, monitoring with sar and vmstat commands, 671–673
- CPU requirements for WAS, requirements for, 82
- CPU thread attribute, description of, 679
- CPUID thread attribute, description of, 679
- crdeb URN service, <mappings> child of, 514

776 IBM WebSphere Application Server Programming

- Create DB2 Services screen, displaying, 27
- createDAObject() method
 - defining in DataAccessComponent package, 247
 - role in DataAccessComponent Bean, 471
- CREATIONTIME column in SESSIONS table, details of, 434
- CreditDebit servlet, overview of, 311–317
- CreditDebit9 servlet, using, 495
- CreditDebitBean as web service, explanation of, 520
- crontab command, automating WSCP output with, 183–185
- curly braces ({}), purpose in WSCP, 172
- custom logins, using with JAAS, 543
- custom tags
 - printing typical messages with, 336–338
 - role in JSP tag libraries, 330
- D**
- \d and \D Perl code, match for, 760
- d-var WASMON differential internal variables
 - versus directives, 713
 - list of, 711
 - purpose of, 708–709, 759
 - understanding, 718–719
 - using in logical expressions, 716–717
- DAC object, defining in init() method of QueryTable servlet, 457–458
- DAO (data access object), creating for use with DataAccessComponentBean, 471
- data access component development, diagram of, 472
- data queues, types of, 131
- data source parameters, obtaining when using WSCP, 173
- data source properties by database brand, table of, 216
- data sources
 - databases used as, 21
 - overview of, 193
 - testing WAS with, 192–193
- DataAccessComponent initialization part, code for, 245–246
- DataAccessComponent Java library package
 - building, 259
 - changing for use with JAAS, 551–552
 - characteristics of, 249–250
 - defining, 244–249
 - text substitution in, 250–251
- DataAccessComponentBean, overview of, 470–472
- DataAccessComponentRemote interface, code for, 472
- database connections, changing settings for, 247
- database tables, using QueryTable servlet with, 298–299
- databases
 - accessing through DataAccessComponent Java library package, 244–249
 - creating for Petstore sample, 152–154
 - rationales for installation of, 20–21
 - using as administrative repositories, 20–21
 - using as data sources, 21
- DataSet object
 - retrieving data with, 251–252
 - type mappings for, 514
- DataSetSerializer interface, programming, 515–518

- DataSource jdbc/Session, configuring setting for, 429
- DataSource objects
 - benefits of, 96
 - getting information about, 192
 - using wscp command with, 170
 - in WAS configurations versus WAS repository, 85
- datasources.xml configuration, suppressing in WAS versions, 132
- DB2 Administration Client, illustration of, 24
- db2 commands, using with DBGUIDE database, 239–240
- DB2 engine, purpose of, 27
- DB2 environment, setting for DBGUIDE database, 234
- DB2 installation, determining Java machine version for, 29
- DB2 Instance, illustration of, 25–26
- DB2 Nodes, verifying authorization credentials on, 151
- DB2 processes, displaying hierarchy of, 28
- DB2 run-time environment, purpose of, 27
- DB2 servers, verifying authorization credentials on, 151
- DB2 settable properties, table of, 216
- DB2 Universal Database Enterprise Extended Edition, illustration of, 25
- DB2 usernames, character limitation of, 24
- DB2_CLASSPATH environment variable, appending to WAS_CLASSPATH environment variable, 364
- db2_home directory, products for, 79
- db2as username, creating, 28
- db2inst1, authority of, 545
- db2java.zip file, locating for use with Petstore sample, 151
- db2setup script, requirements for, 22–23
- DBD::DB2 module, using with DBGUIDE database, 233–240
- DBGUIDE database
 - creating, 225–228
 - features of, 228
 - obtaining metadata for, 230–233
 - populating, 228–233
 - populating with Perl DBI, 233–240
 - testing connectivity with BruteConn.java servlet, 293–295
 - using DBD::DB2 module with, 233–240
- DBGUIDE tables
 - populating with ins_dbguides Perl program, 237–239
 - querying selectively, 295–299
- DbguidesAuth object, code for, 550
- dbguides.pl package, code for, 235
- dbguides.properties program, code for, 246
- dbguides.sql program, code for, 225–227
- DBI (Database Independent) in Perl, purpose of, 233
- dbies program, code for, 234–235
- DBMN WAS component ID, description of, 648
- dbreg user, privileges granted to, 537
- dbreg_privileged directory, description of, 537
- dbreg_sysprotected directory, description of, 537
- dbreg2was custom script, using with JAAS, 537
- DbregLoginModule class, using with FetchFile JAAS example, 528–531
- dbuser, authority of, 545, 547

778 IBM WebSphere Application Server Programming

- DDDD*, appearance when WASMON, 655, 661
- dds.xml Apache SOAP deployment descriptor, generating with gensoap-ejb, 509–511
- DebitCredit servlet, code for, 314–316
- deblog, authority of, 545
- deblog2was script, code for, 539
- debug tools for WAS, servlets used as, 144–145
- default_server_stderr.log file, troubleshooting WAS startup failure with, 197
- default_server_stdout.log file, troubleshooting WAS startup failure with, 197
- delegators, using with WASMON, 696–697
- DELTA directive wasmonhelper.conf file, purpose of, 715
- deployment descriptors, overview of, 340–343. *See also* Apache SOAP deployment descriptors
- DeregPrincipal, implementing for use with PrivilegedFetchFile JAAS program, 535
- <description> (?) element, meaning in web applications, 353
- descriptors. *See* Apache SOAP deployment descriptors; deployment descriptors
- deserialization, role in programming DataSetSerializer interface, 518
- Deserializers, role in Apache SOAP implementation, 515
- destroy() method, role in DumpCookLog servlet, 396–398
- developer roles, segregating for security of teller login/logout, 544–546
- df -k command, revealing free disk space with, 32
- df command, determining disk space with, 32
- Dfile program, demonstrating Cipher decryption with, 556–557
- DFT_DEGREE UDB parameter default value for, 689 explanation of, 691
- directories, replicating in UNIX, 589
- directoryBrowsingEnabled IBM extension attribute, description of, 351
- disk space determining, 32 requirements for WAS, 81
- <display-name> (?) element, meaning in web applications, 353
- Dist directory, creating for use with J2EE web applications, 265
- <distributable> (?) element, meaning in web applications, 353
- distributed environments, copying and moving files and directories in, 588–589
- distribution code, installing, 727–728
- documentation, providing with javadoc command, 589
- doGet() method of TellerLogged servlet, 326–327 using with PostSession servlet, 393
- dollar sign (\$) metacharacter in Perl, action of, 760
- dollar sign and asterisk (\$*), using with .class and .java files, 571
- dollar sign and less-than symbol (<) in make utility, meaning of, 568
- doPost() method of DebitCredit servlet, purpose of, 313
- DOS batch file programming, using set and call commands with, 62
- DOS prompt, converting into shell command processor, 65

- DOS variables, manipulating from command prompt, 62
 - dot (.)
 - metacharacter in Perl, 760
 - using with classpath, 49
 - using with setj.sh script, 53
 - dot slash (./), adding to classpath, 49
 - double dollar sign (\$\$), using with map sed script, 68
 - DR (dynamic reloading) and HD (hot deployment), overview of, 378–382
 - DrAdmin utility, generating thread dumps with, 138–139
 - drive requirements for WAS, requirements for, 82
 - DRSW WAS component ID, description of, 648
 - DTD (Document Type Definition)
 - determining WAS v5 support through, 764–766
 - identifying in WAS AEs, 352
 - dump command, backing up WAS installation with, 218
 - dump Lynx option, description of, 386
 - DumpCook servlet
 - fetching cookies with, 395–396
 - placing request to echo SESSIONID cookie, 393
 - result of, 389
 - DumpCookLog servlet, redirecting System.out to file with, 396–397
 - DumpEnv servlet
 - dumping CGI environment variables with, 286–288
 - sample Lynx request to, 290
 - showing output of, 266
 - DumpSession servlet
 - printing out session data with, 405–407
 - using, 496
 - using with ZappUrl script, 601
 - @DVAR WASMON g-var (global internal variable), content of, 712
 - @DVAR_* WASMON d-vars (differential internal variables)
 - versus directives, 713
 - explanation of, 711
 - purpose of, 759
 - DYNA WAS component ID, description of, 648
- E**
- E in WAS LEDs, meaning of, 647
 - e-mail
 - automating with Perl, 184–185
 - sending alerts by means of, 659–662
 - e-mail alerts, identifying supervisor ticket number with, 700
 - @E-MAIL WASMON s-var (static internal variable), content of, 710
 - e1.0 suffix in Gramercy Toolkit, meaning of, 729
 - EAR (Enterprise Application Archive), components of, 139–143
 - .ear files
 - deployment of, 271
 - role in enterprise application directory structure, 342
 - EAR sample file, sectioning, 146–150
 - EAR tree structure
 - choosing location for creation of, 263–265
 - setting for enterprise web applications, 263
 - ear2tree script, code for, 342–343

780 IBM WebSphere Application Server Programming

- EARExpander command, invoking, 141, 350
- EditTeller9 servlet, using, 495
- egrep command, using with WAS startup classpath, 362–363
- EJB caching, considering, 687–688
- EJB classes, Apache SOAP deployment descriptor artifact for, 506–507
- EJB containers
 - accessing, 466
 - optimizing with larger bean, 473–479
 - role in WebSphere domains, 129
- EJB .Jar files level, result of changes made at, 380–381
- EJB levels supported in WAS v4, explanation of, 132
- EJB module
 - assembling, 496–498
 - deploying with ejbdeploy.sh shell script, 488–489
 - explanation of, 360
 - modifying code after assembly of, 497–498
 - testing, 489–493
 - verifying, 482–483
 - verifying code in, 493–494
 - WasdgBeans.jar as, 482
- <ejb-ref> (*) element, meaning in web applications, 355
- ejbdeploy.sh shell script, employing EJB module with, 488
- EJBs (Enterprise JavaBeans). *See also* beans; beans nomination
 - adding, 466
 - CreditDebitBean, 461–462
 - DataAccessComponentBean, 461–462
 - FetchTableBean, 461–462
 - guidelines for use of, 462
 - linking to WASDGD development tree with symbolic links, 492–493
 - locations of, 464
 - mapping names for, 463
 - merging to WASDGD development tree, 490–492
 - OperationEmpBean, 461–462
 - tree structure for, 463
 - tree structure for development of, 462
 - using HmLookup Java class with, 464–466
- ellipsis (...)
 - metacharacter in Perl, action of, 760
 - output in Perl, 179
 - purpose in wscp_Servlet-list.pl program, 178
- EMAIL field in Employee Records, methods of, 256
- emitproject12profile.sh file, code for, 51–52
- EmpAcct class, code for, 257–258
- EmpAcct object
 - getter/setter methods defined in, 256
 - retrieving data from, 252–258
 - showing for teller in session, 407
- EMPACCNT table in DBGUIDE database
 - description of, 225
 - fields of employee account defined in, 256
- Employee Record, fields of, 256
- EMPNO field in Employee Records, methods of, 256
- EMPPWD table in DBGUIDE database
 - description of, 225
 - inserting record for userid dave into, 250
 - inserting records into, 229–230
 - mapping entries in /etc/passwd file to records in, 230
 - querying, 235–236
 - retrieving structure of, 231–232
 - retrieving USERID and PASSWORD from, 251–252

- enterprise application archives, extracting into
 - directory structures, 342–343
 - enterprise application descriptor, displaying, 270
 - enterprise application level, result of changes made at, 381
 - enterprise application structure
 - archiving into files, 342
 - illustration of, 341
 - Enterprise Archive application, reinstalling, 270–271
 - enterprise WAS programming, Perl scripts used with, 15–16
 - enterprise web applications, setting EAR tree structure for, 263
 - Enumeration getAttributeNames() method, role in JSP programming, 326
 - <env-entry> (*) element, meaning in web applications, 355
 - environment variables, printing with chekenv script, 748–749
 - epmJvmpi library, profiling Java machine GC with, 163–164
 - error and I/O files, specification in WAS v3.5, 197–198
 - error codes, categorizing and tabulating, 631–633
 - <error-page> (*) element, meaning in web applications, 354
 - errorPage page directive attribute, value and description of, 338
 - errors in web containers, detecting with WASMON, 717
 - escaped semicolon (;), using with Windows NT PATH environment variable, 56
 - /etc/ntp.conf file on node1.tcnd.com, code for, 588
 - /etc/passwd file, mapping entries to records in EMPPWD table, 230
 - /etc/profile systemwide profile, execution of, 50–51
 - /etc/services ports, overview of, 205–207
 - EVENT thread attribute, description of, 679
 - events, filtering with WASMON, 660
 - Ex1 Insert program, code for, 250
 - Ex10AccountRetrieval program
 - code for, 252–254
 - output of, 255
 - Ex10Select program, retrieving USERID and PASSWORD from EMPPWD table with, 251–252
 - exception handling
 - logging, tracing, and debugging with, 620–623
 - overview of, 633–639
 - exception-handling properties files, overview of, 631–633
 - exceptions, fetching information about, 638
 - extends page directive attribute, value and description of, 338
- F**
- f <n> option
 - of SharkUrl stress-tester, 607
 - of ZappUrl stress-tester, 600
 - factual tree, diagram of, 581
 - fenced users, selecting for UDB, 26–27
 - fetch() method of FetchTypeBean, exposing, 467
 - FetchFile JAAS program
 - authorizing, 532–534
 - code for, 532–533
 - executing, 536
 - login configuration and policy files for, 535–536
 - overview of, 527–536
 - FetchFile servlet, code for, 299–301.
See also servlets

782 IBM WebSphere Application Server Programming

- FetchSignedFile program, decrypting ciphered files with, 557–559
 - FetchSignedServlet, requesting ciphered documents with, 559–560
 - FetchTableBean
 - code for, 466–467
 - dynamics of, 463
 - using with QueryTable9 servlet, 459–461
 - FetchTableHomeRemote home interface, code for, 467
 - FetchTableRemote interface, code for, 467–468
 - fetchtb URN service, <mappings> child of, 514
 - file contents, printing with servlets, 299–304
 - file sets, role in UDB distributions, 22
 - filemon command, description of, 667
 - files created, role in system environment for WAS deployment, 47
 - FileServingEnabled IBM extension attribute, description of, 351
 - @FILTER WASMON s-var (static internal variable), content of, 710
 - filtering command, using with WASMON, 660
 - find command
 - listing wasdg.ear tree with, 344
 - locating prereq.properties file with, 89–90
 - monitoring WAS application archives with, 217
 - FLAGS thread attribute, description of, 679
 - FlipSession servlet. *See also* servlets
 - FlipSession servlet, code for, 402
 - foo.txt report, generating with SharkUrl script, 612–613
 - foreach, iterating cookie hashes through, 391
 - forked processes, iteration started with, 599
 - forking concurrent processes for concurrent users, overview of, 598–599
 - Format-ServletList-aix formats of servlet list in Perl, code for, 179–180
 - forward slash (/), using with Windows NT, 80
 - free command, role in selecting heap sizes, 214
 - free disk space, allocating with chfs command, 32
 - from Lynx option, description of, 386
 - FRSTNAME field in Employee Records, methods of, 256
- ### G
- g-var WASMON global internal variables
 - list of, 712
 - purpose of, 708–709
 - gensoap-ejb
 - generating Apache SOAP deployment descriptors with, 508–511
 - printing SOAP deployment descriptor with, 511
 - gentagTld script, code for, 332–334
 - genwebxml script
 - code for, 272–274
 - example of, 351
 - role in handling context init parameter, 303
 - GET requests
 - obtaining URLs with, 597
 - role in CreditDebit servlet, 312–313
 - using with LoginScreen servlet, 311
 - using with service() method of DumpEnv.java servlet, 288
 - getComponentCode() method, explanation of, 638

getDescriptiveMessages() method, explanation of, 638
 getFile() method, definition in StrTool package, 303–304
 getjsessionId.cgi script, code for, 408
 getlaperla.cgi script, code for, 390
 getSeverity() method, explanation of, 638
 getter/setter, definitions in IDataAccessObject interface, 247–248
 Gramercy Toolkit
 BwjSort script, 750–751
 cpfl script, 751
 installing, 728–729
 modjar script, 751–752
 GRAMERCY_DIR environment variable, setting, 728
 greater than symbol (>), meaning in comparison of list1.txt to list2.txt, 347
 GTK (Gramercy Toolkit) v1, code distribution web site for, 726
 GUI of Application Client Resource Configuration Tool, starting, 107

H

-h <n> option
 of SharkUrl stress-tester, 607
 of ZappUrl stress-tester, 600
 hash (#) sign, meaning of, 14
 hash values of cookies, explanation of, 390–391
 Hashtable object, using with
 DataAccessComponent, 250–251
 HD (hot deployment) and DR (dynamic reloading), overview of, 378–382
 head Lynx option, description of, 386
 headers for columns, printing when monitoring processes with ps command, 668
 heading files, explanation of, 14
 heap sizes, management of, 214
 HelloPak Java package
 compiling and running, 67
 overview of, 66–68
 HelloPervasive servlet, description of, 144
 HelloURL example, creating, 69–72
 HelloWorld.cgi script, creating, 72
 HelloWorld.html file, code for, 70
 HelloWorld.java program, code for, 66
 Help menu, accessing for WSCP, 170
 here-string, example of, 72
 Hey.java program, code for, 370
 hg2 script, turning HTML file into
 Java string with, 291–295
 hidden fields, overview of, 419–420
 Hit Count servlet, description of, 144–145
 HmLookup Java class, code for, 464–466
 holding tree
 creating, 579–581
 diagram of, 581
 home directory
 conventions for, 78
 defining, 78–79
 hostnames, checking for resolution of, 102
 HTML content
 printing with servlets, 292–293
 tuning into Java strings, 291–292
 HTML documents (.html) file type, location in WAS v4, 104
 HTML forms, using JSESSIONID as input element in, 419–420
 HTTP Java servlets, using cookies through, 391–399
 HTTP methods, role in stress-testing, 597–598

784 IBM WebSphere Application Server Programming

- HTTP plug-in level, result of changes made at, 382
- HTTP requests
 - basic timing for, 595–597
 - checking with Lynx, 290
 - posting with Perl POST script, 596
- HTTP Server. *See* IHS (IBM HTTP Server)
- HTTP server log files, overview of, 42–44
- HTTP sockets, preparing bytes shipped to when stress-testing, 597–598
- @HTTP_ALTERNATIVE WASMON s-var (static internal variable), content of, 710
- @HTTP_SERVER WASMON s-var (static internal variable), content of, 710
- HTTPD log record
 - example of, 290
 - understanding, 289
- httpd.conf file
 - entries for IHS installation, 39
 - modifying for WAS v3.5 and v4, 86–87
- HttpSession objects
 - releasing when closing sessions, 437–438
 - serializing, 437
- I**
 - i <f1,f2...> option of SharkUrl stress-tester, explanation of, 607
 - I in WAS LEDs, meaning of, 647
 - i2jsp Perl script, generating
 - SessionFairy.jsp file with, 413–414
 - IBM corporate web site, display of, 17
 - IBM extensions
 - accessing, 343–351
 - attributes summary, 350–351
 - displaying in AAT, 346
 - overview of, 340–343
 - ibm-web-ext.xmi file
 - attributes of, 350–351
 - displaying contents of, 349
 - IBM WebSphere Servlet Engine
 - Configuration, purpose of, 216–217
 - <icon> (?) element, meaning in web applications, 353
 - ID column in SESSIONS table, details of, 434
 - IDACFetchTableBean, code for, 468–469
 - IDACFetchTableHomeRemote bean, code for, 469
 - IDACFetchTableRemote bean, code for, 469
 - IDACOperationEmpBean
 - code for, 477–479
 - flow diagram for, 477
 - IDataAccessObject interface
 - code for, 248–249
 - defining getter/setter in, 247–248
 - IDB (InstantDB) preconfiguration, removing in Petstore sample, 158
 - identity feature of UDB, role in DBGUIDE database, 228
 - IHS (IBM HTTP Server)
 - checking enabled status of logging on, 104
 - installing, 38–42
 - overview of, 38–44
 - starting and stopping, 42
 - stopping, starting and testing, 190
 - testing, 70
 - IHS log files, rotating on Linux, 43
 - ihs_docroot directory
 - explanation of, 39
 - products for, 79
 - ihs_home alias, explanation of, 39
 - ihs_home/conf directory, explanation of, 39
 - ihs_home directory, products for, 78
 - illegal access, role in JAAS, 542–543

IMAGE field in Employee Records, methods of, 256
 images of installation products, taking, 581–582
 implicit dependency, explanation of, 568
 import page directive attribute, value and description of, 338
 includes, advisory about using with SharkUrl script, 616
 inference rules, using with make utility, 568
 info page directive attribute, value and description of, 338
 Informix settable properties, table of, 216
 init() method
 defining DAC object in with QueryTable object, 457–458
 role in redirection of DumpCookLog servlet, 398
 using with servlets, 365
 initDefaultConnectionSettings() method, using with DataAccessComponent, 246–247
 ins_dbguid Perl program, populating DBGUIDE tables with, 237–239
 InsertEmpPwd.java program, code for, 229–230
 INST WAS component ID, description of, 648
 internal variables, using with WASMON, 697, 758–760
 internal variables, using with wasmon.conf file, 715–718
 internationalization, using MrUnicode for, 738–741
 interprocess communication, elimination by WebSphere domains, 119–120
 interservlet and intraservlet communication, overview of, 329
 invalidation schedule menu, displaying, 449
 iostat command, description of, 667

isErrorPage page directive attribute, value and description of, 338

J

j122 and j130 options for setj.sh script, purpose of, 61
 J2CA WAS component ID, description of, 648
 J2EE API 2.3, WAS v5 support for, 766
 J2EE application deployment, generalizing under WAS AEs, 730
 J2EE trees
 completing, 274–275
 creating, 262–265
 J2EE web applications
 adding new builds to already loaded types of, 275–277
 building, 265–274
 building and deploying, 266
 components of, 262
 creating Dist directory for, 264
 using thisbase command with, 264
 J2EE_CLASSPATH environment variable, description of, 48
 j2tree Perl script
 order of arguments used with, 264–265
 understanding, 277–281
 JAAS (Java Authentication and Authorization Service)
 FetchFile and PrivilegedFetchFile program, 527–536
 future of, 553
 installing, 525–526
 overview of, 524
 programming, 526
 role of class casting and symbolic name resolution in, 552–553

786 IBM WebSphere Application Server Programming

- securing teller login/logout with, 543–553
- testing proper addition to WAS classpath, 526
- using custom logins with, 543
- JAAS programs, deploying in WAS, 538–539
- jaas.policy file
 - code for, 535–536
 - and java.policy file, 541
- jall script, compiling .java programs with, 260
- Jar archives, adding to WAS startup classloader, 364
- jar command
 - archiving enterprise application directory structure with, 342
 - explanation of, 47
 - invoking, 67
- .jar file type, overview of, 140–141
- jarsigner command, explanation of, 47
- java -X command, invoking JVM arguments with, 212
- Java 2 component support in WAS v4 versus v5, table of, 765
- Java 2 encoding, Sun Microsystems versus IBM nomenclature in, 744–745
- Java 2 SDK v1.4, user authentication APIs supported in, 553
- Java classes
 - Apache SOAP deployment descriptor artifact for, 506
 - passing parameters of, 511–519
- java command, using jar command with, 141
- Java computing system environment, setting up, 46–61
- Java distribution, typical commands available with, 47
- Java environment setup, separating from profile, 61
- Java environment variables
 - overview of, 48
 - setting globally with setj.sh script, 56–61
- .java files
 - compiling in Java package development tree, 14
 - converting to .class files with make utility, 568
 - generating .class files from, 571
- Java machine
 - arguments passed to, 213
 - availability on WAS, 87–88
 - distributing on several machines with WebSphere domains, 120
 - installing, 45–46
 - overview of, 44–45
 - relationship to WSCP, 137
 - working with, 65–72
- Java machine GC (garbage collection), profiling with epmJvmpi library, 163–164
- Java machine processes
 - categories of, 128
 - dumping state of, 203
- Java package tree, creating, 259
- Java packages
 - building, 259
 - mediocre compilation of, 259
 - working with, 66–69
- Java Pet Store Demo page, displaying, 159
- Java programs
 - compiling and running without specifying classpath, 55
 - compiling with jall script, 260
 - embedding WSCP commands in, 176
 - locating class files with, 49
- Java, role in DBGUIDE database, 228–233

- Java strings, tuning HTML content into, 291–292
- Java threads. *See also* thread entries
 - displaying with MrThread tool, 683
 - effective implementation of, 204–205
- java* commands, explanations of, 47
- JAVA_COMPILER environment variable, description of, 48
- JAVA_EXE environment variable, description of, 48
- java_home directory, products for, 79
- JAVA_HOME environment variable, description of, 48
- javac command
 - compiling programs with, 66
 - role in compiling Java packages, 259
- JAVAC user-defined macro in Makefile program, purpose of, 567
- JAVACOPT user-defined macro in Makefile program, purpose of, 567
- javacore files, generating with DrAdmin utility, 138–139
- javadoc command, documenting packages on web applications with, 589
- java.io.* package, role in dumping CGI environment variables, 286
- java.policy, role in FetchFile JAAS program, 533–534
- java.security.manager, advisory when using JAAS programs, 540
- JavaServer Pages (.jsp) file type, location in WAS v4, 104
- java.sql.DatabaseMetaData interface, implementing, 230–233
- java.util.ArrayList, importing with page directives, 338
- javax.servlet.* package, role in dumping CGI environment variables, 286
- javax.servlet.http.* package, role in dumping CGI environment variables, 286
- JDBC driver types, classifying for UDB, 33–34
- JDBC drivers
 - determining when using WSCP, 172
 - functionality of, 74–75
 - setting for WAS, 94–96
 - setting properties for Name PetStore Datastore, 157
 - testing remote database connections with, 73–75
- JDBC (Java Database Connectivity), role in DBGUIDE database, 228–233
- JDBC provider, configuring for WAS
 - AEs sessions, 428
- JDBC URL, explanation of, 74
- JDWP (Java Debug Wire Protocol) port, identifying, 207
- JIT (Just-In-Time) compiler, disabling, 213
- .jlist file, code for, 566
- JNDI names, looking up when setting and configuring sessions, 430–433
- jndi option for gensoap-ejb, explanation of, 509
- JndiLoginModule API in Java 2 SDK v1.4, description of, 553
- JORB WAS component ID, description of, 648
- JSAS WAS component ID, description of, 648
- JSESSIONID
 - cookie for identifying sessions, 400–407

788 IBM WebSphere Application Server Programming

- reading with `getjsessionid.cgi` script, 408
 - revealing information about using `-vj` option of `SharkUrl` script, 611
 - role in `ZappUrl` script, 601
 - JSP levels supported in WAS v4, explanation of, 132
 - JSP pages
 - placing requests to, 322
 - programmatic handling of, 320
 - JSP programming
 - `List.jsp` example, 321–324
 - `UpdateJSP` servlet, 325
 - JSP tag libraries
 - automating process of, 332–335
 - overview of, 329–332
 - steps for programming with, 335
 - JSPG WAS component ID, description of, 648
 - JVM processes, bringing together, 14
- ### K
- Kernel requirements for WAS, overview of, 82
 - `KeyStoreLoginModule` API in Java 2 SDK v1.4, description of, 553
 - `kill` command, invoking, 189
 - Korn shell, advantages of enabling
 - Windows NT command processor with, 64–65
 - Korn versus Bourne shell environment variable, 56
 - `Krb5LoginModule` API in Java 2 SDK v1.4, description of, 553
- ### L
- `ladder` command, example of, 526
 - language code, printing with `MrUnicode` tool, 745–746
 - language page directive attribute, value and description of, 338
 - LAPERLA cookie, reading, 390
 - LARGE column in `SESSIONS` table, details of, 434
 - LASTACCESS column in `SESSIONS` table, details of, 434
 - LASTNAME field in `Employee Records`, methods of, 256
 - `LD_LIBRARY_PATH` environment variable, description of, 48
 - LEDs, WAS component message numbers as, 646–648
 - `LIBPATH` environment variable, description of, 48
 - lightweight process, explanation of, 204
 - `linkres` script, code for, 579–580
 - Linux
 - `/etc/profile` systemwide profile on, 50–51
 - confirming installation of IHS on, 41
 - IHS installation directory for, 39–40
 - Java machine installation for, 46
 - listing processes as tree representations with, 203
 - locating latest release of Tcl for, 168
 - `logrotate` command used with, 199–200
 - performance commands on, 666–667
 - rotating log files with, 43
 - tabulation of `setw(v)(os).sh` script in, 107
 - user profile in, 50
 - `vmstat` report on, 672
 - Linux processes, monitoring with `MrTop` utility, 684–686
 - Linux RedHat v.7.2, code distribution web site for, 727
 - `_List_jsp_0.java` program, code for, 323–324

- list1.txt file, comparing to list2.txt file, 347
- listbeans Perl script, code for and output of, 479–480
- listcp Perl script, code for, 369
- LISTENERCNT column in SESSIONS table, details of, 434
- List.jsp program, code for, 321
- loaders, constraining violation with, 494
- LOCKLIST UDB parameter
 - default value for, 689
 - explanation of, 691
- lockstat command, description of, 667
- LOCKTIMEOUT UDB parameter, default value for, 689
- Log Analyzer tool
 - checking WAS version with, 196
 - overview of, 160–161
 - using with activity.log file, 201–203
- Log class, code for, 625–627
- .log file type, location in WAS v4, 105
- log files
 - advisory about, 43
 - deleting while application server is running, 200
 - formatting, 197–199
 - location of, 196
 - in WAS, 196–203
- log formats, giving nicknames to, 43
- logbr suffix, meaning of, 160
- logging activity, starting, 722
- logging and exception-handling packages, development tree hierarchy for, 621
- logging, enabling for HTTP server log files, 43
- Logical Alert directives
 - conditional monitoring with, 662–664
 - using logical expressions in, 756–757
- LOGICAL directives used in WASMON v1.2.1 configuration file, purpose of, 756
- logical expressions
 - using b-vars and d-vars in, 716–717
 - using in SUPERVISOR directive, 757–758
 - using with WASMON, 756–758
- @LOGICAL WASMON s-var (static internal variable), content of, 710
- <login-config> (?) element, meaning in web applications, 354
- login modules, alternatives for use with JAAS, 543
- login scripts, setting up on Windows NT, 63–64
- login.conf file
 - establishing JAAS login authentication with, 551
 - writing for FetchFile JAAS example, 528
- LoginScreen servlet
 - code for, 306–308
 - logging in as teller, 290
 - requesting by class name using Lynx, 349
- LoginScreen9 servlet test scenario, overview of, 494–496
- Log.log() method
 - logging information with, 623–627
 - purpose of, 621–622
 - testing program for, 639
- Logoff servlet, invalidating HttpSession objects with, 437–438
- Logout servlet
 - build tree for, 546
 - code for, 548–550
 - overview of, 547–548
- LogOut9 servlet, using, 496
- log.properties file, overview of, 631
- logrotate command, using with Linux, 199–200
- LogWriter class, code for, 623–624

790 IBM WebSphere Application Server Programming

- LogWriter file, code for, 624–625
 - lsattr command, role in selecting heap sizes, 214
 - lspp command, invoking, 31, 41
 - LTXT WAS component ID, description of, 648
 - LUM (License Use Management), advisory about, 216
 - Lynx
 - batch processing with, 317–318
 - checking HTTP requests with, 290
 - deleting cookies from browsers with, 395
 - ensuring reachability of RPC router with, 505
 - examining teller login scenario with, 444–447
 - overview of, 14–15
 - requesting LoginScreen servlet by class name with, 349
 - session harassment with, 447
 - testing PostCookie servlet with, 394
 - Lynx commands
 - example of, 366
 - issuing to log in teller1, 445
 - Lynx options for developers, table of, 386
- M**
- make files, role in system environment for WAS deployment, 47
 - make utility
 - compiling code with, 566–568
 - overview of, 565–566
 - synchronizing time during use of, 586
 - Makefile program
 - advisory about coding of, 575
 - for BASE_DEV, 571–576
 - code for and user-defined macros in, 566–567
 - generating in base directory of source tree, 574–575
 - invoking, 574
 - makeit file, code for, 571
 - makejar file, invoking, 571
 - manage-node2.properties property file, specifying with WSCP, 171
 - <mappings> item
 - artifact code, 513
 - child of URN service crdeb code, 514
 - child of URN service fetchtb code, 514
 - map.sed script, code for, 68
 - map.sh script, code for, 69
 - marshalling, role in programming
 - DataSetSerializer interface, 517–518
 - MAXAGENTS UDB parameter, default value for, 689
 - MAXAPPLS UDB parameter, default value for, 689
 - MAXINACTIVETIME column in SESSIONS table, details of, 434
 - MAXLOCKS UDB parameter
 - default value for, 689
 - explanation of, 691
 - maxou@node1.tend.com, e-mail sent by WASMON to, 704–706
 - maxou@node.tend.com, alert sent as e-mail to, 700
 - MEDIUM column in SESSIONS table, details of, 434
 - memory, monitoring with sar and vmstat commands, 671–673
 - memory requirements for WAS, explanation of, 81
 - memory usage, displaying with MrTop utility, 685

- memory usage, monitoring with ps
 - command, 668
 - Merant settable properties, table of, 216
 - message codes, interpreting, 197–198
 - message mnemonics, fetching information
 - about, 638
 - messages, printing with customized tags
 - libraries, 336–338
 - messages.properties file, using for exception
 - handling, 631–632
 - META file for sampleApp.ear file, purpose of, 147–148
 - metadata, obtaining for DBGUIDE database, 230–233
 - metafiles, role in assembling EJB
 - module, 496
 - MIGR WAS component ID, description
 - of, 648
 - migration considerations, overview of, 734–735
 - <mime-mapping> (*) element, meaning in web applications, 354
 - mime_header, accessing, 385–386
 - mime_header Lynx option, description
 - of, 386
 - mnemonic error code, explanation of, 631
 - modjar script, overview of, 751–752
 - module visibility
 - example using reloading, 370–377
 - overview of, 367–370
 - monitoring
 - with logical alert directives, 662–664
 - objective of, 646
 - scenario of, 761
 - monitorNode procedure, code for, 173–174
 - MrThread utility
 - features of, 12
 - monitoring threads with, 680–683
 - versus MrTop utility, 684
 - printing thread table with, 682
 - MrTop utility
 - displaying memory usage
 - with, 685
 - monitoring Linux processes with, 684–686
 - printing information about forked processes and CPU usage
 - with, 685
 - MrUnicode utility
 - converters and character sets
 - listed by, 744
 - guidelines for use of, 738
 - printing information about converters and character sets
 - with, 742–744
 - printing language code with, 745–746
 - purpose of, 12
 - syntax of, 738–739
 - using for internationalization and Unicode, 738–741
 - ms argument for Java machine,
 - purpose of, 213
 - MSGS WAS component ID,
 - description of, 648
 - mx argument for Java machine,
 - purpose of, 213
- N**
- Name PetStore Datasource, setting
 - properties of JDBC driver
 - for, 157
 - Name PetstoreMail, setting properties
 - of mail transport for, 157
 - name server problems,
 - troubleshooting, 145

792 IBM WebSphere Application Server Programming

- Nanny process
 - purpose of, 128, 203
 - role in WebSphere domains, 129–130
 - nanny.trace file, troubleshooting WAS
 - startup failure with, 196
 - native2ascii command, explanation of, 47
 - NDX field in Employee Records, methods of, 257
 - nebbish.sh script, code for, 447
 - nested braces, formatting WSCP output with, 180–183
 - net driver, overview of, 34
 - net START and net START commands, invoking for IHS, 42
 - netpmon command, description of, 667
 - netstat -e sample output, 209–210
 - netstat command
 - description of, 667
 - guidelines for use of, 209
 - invoking, 206, 207
 - using for Windows NT network usage, 675–678
 - Network Analyzer, displaying results in, 614–615
 - network hardware for WAS, requirements for, 82
 - nfsstat command, description of, 667
 - NMSV WAS component ID, description of, 648
 - NNNN message number for WAS
 - components, meaning of, 646
 - no command, description of, 667
 - noasynccg argument for Java machine, purpose of, 213
 - noclasscg argument for Java machine, purpose of, 213
 - Node remove command, invoking in WCSP, 174
 - nodes
 - checking in WebSphere domains, 193–194
 - explanation of, 115–116
 - monitoring within WAS, 173–174
 - overview of, 126–129
 - nojit argument for Java machine, purpose of, 213
 - nojit Java 1.1 run-time argument, explanation of, 213
 - NTLoginModule API in Java 2 SDK v1.4, description of, 553
 - ntpd command, confirming clock synchronization with, 588
 - ntpd.conf file on arayech.tcnd.com time server, code for, 587
 - @NUMMAIL WASMON s-var (static internal variable), content of, 710
 - @NUMTRIG WASMON s-var (static internal variable), content of, 710
- ### 0
- o option, using with ps command, 668–669
 - OBJDIR user-defined macro in Makefile program, purpose of, 567
 - object binding, role in WasdgBeans.ear generation and registration, 483
 - Object getAttribute() method, role in JSP programming, 326
 - OLT/OLD (Object Level Trace/Object Level Debugger), overview of, 640
 - online command, using with WASMON, 660
 - online filters, using with WASMON, 661
 - online triggers, using with WASMON, 661
 - operations in UpdateTeller9 servlet, explanation of, 476

Oracle settable properties, table of, 216
 ORB communication problems,
 troubleshooting, 145
 org.apache.soap.server.http package, classes
 in, 503
 os in set(v)(os).sh script, meaning of, 105
 OSE proprietary protocol, purpose of, 131–132
 -oss argument for Java machine, purpose
 of, 213

P

-P option, using with sar command, 672
 packages
 documenting on web applications, 589
 listing for DB2 servers, 30
 understanding build process of, 258–260
 page directive attributes for JSP tag libraries,
 table of, 338
 page sizes, setting for sessions, 450–451
 pak.jar package, creating, 67
 parameters, passing, 511–519
 parentheses (())
 meaning in .xml documents, 353
 purpose in sample shell script, 343
 parole, entering with SUPERVISOR directive,
 701–708
 ParseWscp element, purpose of, 183
 passwords, checking for change during UDB
 installations on AIX, 32
 PATH environment variable, description of, 48
 PATH variable, explanation of, 48
 PCKCACHESZ UDB parameter, default value
 for, 689
 performance
 considering for SESSION database,
 688–691
 gaining with WebSphere domains, 120
 performance commands on AIX and
 Linux, overview of, 666–667
 performance problems,
 troubleshooting, 145
 performance-tuning tools, role in
 system environment for WAS
 deployment, 47
 Perl
 metacharacters used in, 760
 running WSCP from, 177–178
 Perl and Perl modules, code
 distribution web site for, 727
 Perl DBI (Database Independent),
 populating DBGUIDE database with,
 233–240
 Perl/MIME::Lite, automating e-mail
 with, 184–185
 Perl POST script, posting HTTP
 requests with, 596
 Perl scripts
 for Hello World!, 72
 role in enterprise WAS
 programming, 15–16
 Perl short coding scheme, table
 of, 760
 persistent database, writing to, 439
 persistent HTTP session objects,
 serializing, 437
 persistent sessions, committing
 changes in, 438–439. *See also*
 session entries
 PETSTORE databases
 creating, 153
 tables in, 153–154
 Petstore sample
 database authentication for,
 153–154
 installing, 150–159

794 IBM WebSphere Application Server Programming

- preinstallation verification of, 151–159
- removing from application server, 156
- setting context root of, 159
- petstore.ear file
 - installing, 158
 - modifying with AAT, 155–156
- PID thread attribute, description of, 679
- pipe (|) metacharacter in Perl, action of, 760
- pipe (|) operator, meaning in .xml documents, 353
- Plain.jsp program, code for, 325
- PLGN WAS component ID, description of, 648
- plugin-cfg.xml file type, location in WAS v4, 105
- plus sign (+)
 - metacharacter in Perl, 760
 - operator in .xml documents, 353
 - versus semicolon (;), 291
- PMI.API (Performance Monitoring Infrastructure Application Programming Interface), using with WebSphere domains, 164
- PMON WAS component ID, description of, 648
- POLICY thread attribute, description of, 679
- port accessibility, checking, 206
- port numbers, range of, 205
- ports
 - 2012, 130
 - 7000, 130
 - 7000 set for DrAdmin utility, 138
 - 900, 130
 - 900 in WSCP property files, 171
 - 9000, 130
 - 9080, 130
 - 9090, 84, 97, 130, 208–209
 - 9090 information gathering, 207
 - 9443, 130
 - assigning for WAS instances, 583–585
 - debugging 2102, 207
 - debugging 7777, 207
 - debugging defaults for, 207
 - identifying use of, 582–583
 - mapping to other sets of ports, 582–583
 - monitoring and reporting network status of, 206
 - monitoring in WAS, 207–210
 - random assignment of, 130
 - selecting for TCP/IP connections in UDB DB2 instances, 26
 - using with WAS, 130
- POST requests
 - of LoginScreen servlet, 309
 - setting cookies through, 392–395
 - using with service() method of DumpEnv.java servlet, 288
- POST script, posting HTTP requests with, 596
- post_data Lynx option, description of, 386
- PostCookie requests, getting mime header of, 394
- PostCookie servlet, code for, 392–393
- PostSession servlet, testing, 393
- PreCompileJSPs IBM extension attribute, description of, 351
- prereq_checker, suppressing in WAS, 89–90
- PRI thread attribute, description of, 679
- PrintCp servlet
 - code for, 365
 - printing classpath visible to, 369

printenv.cgi program, printing CGI environment variables with, 284–285
 printenv.sh script, dumping CGI environment variables with, 285
 PrivilegedFetchFile JAAS program, securing, 534–535
 processes
 forking for concurrent users, 598–599
 iterating to place multiple requests within, 599
 monitoring with ps command, 668–670
 overview of, 678
 processName threads, reporting with MrThread tool, 681
 PROCNAME thread attribute, description of, 679
 product.xml file, getting product build and release date from, 195
 profile, separating from Java environment setup, 61
 #PROJECT12-JAVA-SYSTEMWIDE-ENVIRONMENT-001 comment, meaning of, 51
 Properties directory for logging and exception-handling packages, contents of, 621
 property files
 accessing for WSCP, 171
 advisory about editing of, 200
 PROPID column in SESSIONS table, details of, 434
 ps -ef | grep http command, purpose of, 70
 ps command
 description of, 667
 dumping state of Java machine processes with, 203
 format descriptors used after -o option of, 669
 monitoring processes with, 668–670
 purpose of, 136

ps_db2.sql file, locating for use with Petstore sample, 151
 pstat command
 description of, 667
 using -A option with, 678, 680
 using with AIX, 199
 pstree command sample code, 203–204
 psv script, syntax and code for, 669–670

Q

QName class, qualifying names on SOAP clients with, 512–513
 -qname option for gensoap-ejb, explanation of, 509
 qry session Perl DBI program, code for, 435
 qry_emppwd program, code for, 235–236
 qry_session program, querying SESSIONS table with, 445–446
 QueryTable servlet. *See also* servlets defining DAC object in init() method of, 457–458
 querying selected database table with, 298–299
 QueryTable9 servlet
 printing messages to WAS standard output with, 490–491
 using FetchTypeBean with, 459–461
 question mark (?)
 metacharacter in Perl, 760
 operator in .xml documents, 353

R

-R <filename> option
 generating SharkUrl reports with, 610–614

796 IBM WebSphere Application Server Programming

- of SharkUrl stress-tester, 607
 - of ZappUrl stress-tester, 600
 - RC (runtime classpath)
 - directory location of, 359
 - example of, 360
 - RCP (runtime classpath patches)
 - directory location of, 359
 - example of, 360
 - RCS link, role in holding tree
 - example, 580
 - RCS subdirectories, creating with make
 - make_rcs command, 576
 - RE (runtime extension)
 - directory location of, 359
 - example of, 360
 - recursive option, role in WCSP, 174
 - REFRESH INTERVAL directive used in WASMON v1.2.1 configuration file, purpose of, 756
 - refresh.sh script, using with deployed EAR, 269–271
 - refreshsoappear script
 - code for, 501
 - result of running of, 502
 - registry tree structure, restoring after archiving WAS directory on Windows NT, 219–220
 - reloading
 - example using module visibility, 370–377
 - explanation of, 366
 - ReloadingEnabled IBM extension
 - attribute, description of, 351
 - ReloadInterval IBM extension attribute, description of, 351
 - remote database connections, testing with JDBC drivers, 73–75
 - REMPID field in client record, role in DBGUIDE database, 228
 - renicing WAS Java machines, explanation of, 205
 - reports, generating and interpreting with SharkUrl script, 610–614
 - RequestDispatcher feature, role in JSP programming, 324–328
 - ResetSession servlet, code for, 403–404.
 - See also* servlets
 - resettagtd script, code for, 335
 - resetwebxml script, code for, 272
 - Resource Analyzer WAS tool, overview of, 161–164
 - <resource-ref> (*) element, meaning in web applications, 354
 - risk management with WASMON, overview of, 697–701
 - RLD* notation, meaning of, 378
 - rm command, advisory about removing log files with, 200
 - rmic command, explanation of, 47
 - rmiregistry command, explanation of, 47
 - rmss mon command, description of, 667
 - root, authority of, 545
 - RPC routers, using Lynx to ensure reachability of, 505
 - RPC services, specifying for type mapping, 513
 - rpm installation tool, using with IHS, 40–41
 - run-time errors, troubleshooting, 145
- ## S
- s option, using with netstat command, 675
 - \s and \S Perl code, match for, 760
 - s-var WASMON static internal variables
 - list of, 710
 - purpose of, 708–710

- SAMPLE database, attempting to connect to, 74
- sampleApp.ear file
 - directory hierarchy of, 147
 - displaying through administrative console, 148
 - location of, 147
 - mapping of, 148–149
 - overview of, 146–150
- sampleTags.tld file
 - code for, 331–332
 - describing in web.xml web application descriptor, 335
- sar -B 1 100 command, output of, 671
- sar command
 - description of, 667
 - monitoring memory and CPU with, 671–673
- SAX parser, using with WAS, 352
- <SCRIPT> tags, security advisory about, 218
- scripts
 - automating cut and paste for, 51–52
 - role in system environment for WAS deployment, 47
 - triggering during use of WASMON, 660
- SD notation, appearance when using q-name option with gensoap-ejb, 509
- SEAppInstall.sh script, overview of, 141–142
- SECJ WAS component ID, description of, 648
- security advisory about <SCRIPT> tags, explanation of, 218
- <security-constraint> (*) element, meaning in web applications, 354
- security objects, using with WSCP, 171
- <security-role> (*) element, meaning in web applications, 355
- sed script, example of, 68
- Select Components, displaying in Windows NT when installing IHS, 41
- SelectTable servlet, code for, 296–297.
 - See also* servlets
- semicolon (;)
 - versus colon (:), 67
 - versus plus sign (+), 291
 - using with classpath, 49, 54
- serialization, recognizing with Apache SOAP, 511
- Serializers, role in Apache SOAP implementation, 515
- serialver command, explanation of, 47
- server-cfg.xml file
 - changing manually for Petstore sample, 158
 - differentiating versions for J2EE web applications, 266
 - HTTP transport port in, 207–208
 - location in WAS v4, 105
 - registering WasdgBeans.ear enterprise application in, 487
- server configuration level, result of changes made at, 382
- server grouping, role in WAS v4, 133
- servers
 - installing multiple instances of WebSphere domains on, 581–586
 - in WebSphere regions, 114–115
- serveServletsByClassname, enabling in AAT, 348
- serveServletsByClassnameEnabled IBM extension attribute, description of, 351
- service() method, role in DumpEnv.java servlet, 288
- services files, addition of ports to, 206

798 IBM WebSphere Application Server Programming

- Servlet (.class .jar) file type, location in WAS v4, 105
- servlet communication, overview of, 328–329
- Servlet engine, mode-switching in WAS v3.5.5, 132–133
- `<servlet-mapping>` (*) element, meaning in web applications, 354
- Servlet mappings, getting with MrUnicode, 739
- Servlet problems, troubleshooting, 145
- Servlet programming, purpose of, 292
- Servlet Redirector, suppressing in WAS versions, 131–132
- Servlet show (some_servlet_name_Vri) -all WSCP command, parsing, 180–183
- Servlet URLs, testing WAS with, 191
- `<servlet>` (*) element, meaning in web applications, 353
- ServletContext methods, table of, 326
- SERVLETNAME, role in generating web.xml web application deployment descriptor, 271
- Servlets. *See also* individual servlets
 - accessing UDB with, 293–299
 - blocking serving of, 218
 - enabling by classname with AAT, 348
 - getting application classpaths from within, 364–365
 - initializing and reloading, 365–367
 - listing with WSCP command, 177–178
 - printing file contents with, 299–304
 - printing HTML content with, 292–293
 - recompiling with svlbuild script, 290–291
 - role in WebSphere domains, 129
 - setting cookies from within, 391–392
 - tactics for programming of, 288–292
 - testing, 288–289, 350
 - using for teller/client example, 305
 - using wscp command with, 170
- Servlets used as debug tools for WAS, table of, 144–145
- Servlets.list file, processing, 272–274
- SESN WAS component ID, description of, 648
- session affinity, understanding, 426–427
- session beans, turning programs into, 466–469
- `<session-config>` (?) element, meaning in web applications, 354
- session data, printing out, 405–407
- SESSION database, considering performance for, 688–691
- session identifiers
 - explanation of, 424
 - gathering information from, 410–412
- session impact, understanding when reloading servlets or .jspxs, 439
- session introspection, testing, 443
- session management
 - explanation of, 424
 - overview of, 399–400
 - structure of, 385
- session manager
 - tuning, 448–450
 - WAS containment cache setting for, 450
- session page directive attribute, value and description of, 338
- session persistence
 - examining, 427–436
 - reasoning behind, 451–452
 - testing, 433–436
 - testing with teller login scenario, 439–448
- session scope, understanding, 424–426
- Session servlet, code for, 401
- SessionFairy servlet
 - code for, 410–412

- overview of, 409–410
 - testing introspection with, 443
- SessionFairy9 servlet, using, 496
- SessionFairy.jsp file
 - generating, 412–416
 - requesting when testing session persistence, 444
- SESSIONID cookie code, meaning of, 388–389
- sessions. *See also* persistent sessions
 - displaying value of, 441
 - dumping to test session persistence, 441
 - identifying with JSESSIONID cookie, 400–407
 - invalidating, 402–404
 - invalidating on logoff, 437–438
 - persisting with WAS components, 433
 - referencing multiple objects with, 448
 - setting page sizes of, 450–451
 - setting up and configuring with WAS AEs, 427–433
 - timing out, 404–405
 - using with single web container, 447–448
- SESSIONS table
 - examining status of, 442–443
 - fields in, 434
 - output result of, 15
 - querying, 434–436
 - querying with qry_session program, 445–446
 - role in session affinity, 427
- set command, using with DOS batch files, 62
- Set-Cookie: code, meaning of, 388–389
- SETCLASS.BAT file, code for, 62
- setclasspath_(p)(v).sh script, purpose of, 54
- setclasspath.sh script
 - purpose of, 53–55
 - specifying pathname to, 55
- SetCookie servlet, code for, 391–392
- setcookies.cgi script, code for, 387–389
- setcustom.sh script, contents of, 54
- SETJAVA.BAT file, code for, 62
- setj.sh script
 - branches of, 53
 - code for, 56–61
 - executing, 52–53
 - invocation of, 105
 - making available to all users, 60
 - options for setting development environments, 61
 - purpose of, 52
 - setting Java environment variables globally with, 56
 - testing setup of, 107
- setupCmdLine.sh script, role in WAS, 211–212, 361
- setw4lin.sh script
 - code for, 106–107
 - modifying when installing instances of WebSphere domains on same server, 585
- setw(v)(os).sh script
 - purpose of, 54
 - setting WAS environment variables globally with, 55
- severity.properties file, using for exception handling, 632
- sh command, invoking, 65
- shared_tools directory, products for, 79
- SharkUrl script
 - files preprocessed by, 606–608
 - generating and interpreting reports with, 610–614
 - options of, 607
 - purpose of, 606
 - running, 608–610
 - snapshot of foo.txt report generated by, 612–613

800 IBM WebSphere Application Server Programming

- shell interpreter on Windows NT, overview
 - of, 64–65
- shell scripts, advisory about, 56
- shell/sed, batch processing with, 68–69
- shoCfg request, testing WAS with, 191
- SHOW.BAT file, code for, 62
- showCfg servlet
 - description of, 144
 - invoking, 216–217
 - netstat -e output before and after invocation of, 209–210
- showlog script, browsing through activity.log file with, 200–201
- simreload script
 - code for, 373–374
 - result of running of, 375–377
- simulate1.inc include file, processing by SharkUrl script, 606–607
- simulate2.inc include file, processing with SharkUrl, 608–609
- slash dot (\.) notation, meaning of, 68
- SLT thread attribute, description of, 679
- SMALL column in SESSIONS table, details of, 434
- smitty command, invoking, 112
- smitty command, invoking for product installation, 30
- SMTL WAS component ID, description of, 648
- snoop test servlet
 - description of, 144
 - serving on WAS, 102–104
 - using with sampleApp.ear file, 149–150
- SOAP clients
 - accessing beans with, 520–521
 - debugging, 521
 - qualifying names on, 512–513
 - testing with teller login scenario, 521
- SOAP deployment descriptors. *See* Apache SOAP deployment descriptors
- soap-ejbfactory.list file, code for, 508–509
- SOAP message, demystifying, 504–505
- SOAP services, enabling within EAR archives, 143
- SOAP (Simple Object Access Protocol), overview of, 500. *See also* Apache SOAP
- soapcp script, code for, 520
- SoapEarEnabler.sh script, overview of, 143
- SoapEnabler.jar, extracting contents of, 140
- SOAPMappingRegistry mappings for Java types, list of, 512
- soap.war web application, overview of, 502–504
- software used in book, list of, 16–17
- someurls script, code for, 595
- source trees
 - diagram of, 581
 - linking holding trees to, 579–580
- SP (Scalable POWERparallel) system nodes versus WebSphere domains, 117
- splitcp script, code for, 363
- srm.conf file on Apache server, modifying, 87
- SRVE WAS component ID, description of, 648
- ss argument for Java machine, purpose of, 213
- SSN field in Employee Records, methods of, 256
- ST thread attribute, description of, 679
- startServer.sh script
 - overriding startup behavior of, 188
 - in purpose of, 112

- startup Java machine defined
 - in, 361
- startup failure, troubleshooting, 145
- startup messages in WAS, overview of, 196–203
- startupServer.sh script, running for WAS, 92–93
- stdout.log file, code for, 197
- StockQuote servlet, description of, 144
- stopServer.sh script
 - purpose of, 112
 - stopping, 189
- stress-testers, programming, 597–606
- stress-testing
 - overview of, 594
 - with sar -B 1 100 command, 671
 - with SharkUrl script, 606–616
 - web servers, 596
- String getErrorInfo() method in WscpCommand interface, purpose of, 176
- strings, replacing with subs script, 417
- StrTool class
 - location of, 303–304
 - role in JAAS example, 538
- StructureEmpPwd program, code for, 231–232
- StructureEmpPwd.java program, purpose of, 230
- @SUBJECT WASMON s-var (static internal variable), content of, 710
- subs script, disabling cookies with, 417
- subsreload Perl script, code for, 371–372
- subwasdir script, code for, 584–585
- subwasports script, code for, 584
- SUPERVISOR directive
 - entering in parole with, 701–708
 - purpose in WASMON, 698–699
 - putting in active state, 700
 - used in WASMON v1.2.1 configuration file, 756
 - using logical expressions in, 757–758
- supervisor mode, putting WASMON in, 697–700
- svlbuild script
 - code for, 276–277
 - dynamics of, 538
 - recompiling servlets with, 290–291
- svmon command, description of, 667
- swap paging space, considering, 686
- swap space for WAS, requirements for, 82
- Sybase settable properties, table of, 216
- Sybase UNICODE character set, role in managing WAS repository, 215–216
- symbolic links
 - creating for RCS directories, 580
 - linking EJBs to WASDGD
 - development tree with, 492–493
- symbolic name resolution, role in JAAS, 552–553
- sync() method, writing to persistent databases with, 439
- synchronization, role in EJB caching, 687
- sysinf command, printing Windows NT memory information with, 673
- System Agent, installing on AIX, 667–668
- system properties
 - adding to startup of Java machine when using JAAS programs, 540–541
 - java.policy, jaas.policy, and login.conf added to WAS as, 542
 - values of, 541
 - in Windows NT, 63

802 IBM WebSphere Application Server Programming

System.out redirection, logging cookie information by means of, 396–399

T

-t <server> option

of SharkUrl stress-tester, 607

of ZappUrl stress-tester, 600

tables, entering by name with SelecTable servlet, 296–297

<taglib> (*) element, meaning in web applications, 354

tags libraries

customizing to print typical messages, 336–338

registering in web application descriptor, 335

tailoring files, explanation of, 14

tar command, archiving WAS installation with, 218

Tcl (tool command language), relationship to WSCP, 168, 173–175

TCP/IP connection, selecting for UDB DB2 instance properties, 26

teller 1, logging in using Lynx, 15

teller authentication, implementing with TellerLogged servlet, 308–311

teller/client example, using servlets for, 305

Teller In Session window, displaying, 440
teller login/logout, securing with JAAS, 543–553

teller login scenario

examining using Lynx, 444–447

testing session persistence with, 439–448

teller login screen, displaying, 305

teller login servlet, overview of, 305–308

teller login, timing, 595

Teller Profile screen, displaying, 420

Teller to credit/debit client account, using CreditDebit servlet for, 311–317

TellerLogged servlet

doGet() method of, 326–327

implementing teller authentication with, 308–311

posting request to, 427

TellerLogged9 servlet, generating welcome screen with, 495

telnet prompt, accessing, 206

Test1 program, catching and throwing WasdgException exceptions with, 640

TestLog program, writing messages to log files with, 640

testsoap2 program, examining to match client QName() to server mappings, 519

text substitution, performing on SQL command strings, 250–251

thisbase command

invoking for use with wasdgsoap.ear application, 501

role in accessing IBM extensions, 343

role in loading J2EE web applications, 264

thread attributes, descriptions of, 679

threads. *See also* Java threads

creating with MrThread tool, 682–683

monitoring, 680

monitoring with MrThread tool, 680–683

threads and processes, overview of, 678

THREADS_FLAGS environment variable, description of, 48

@TICKET WASMON s-var (static internal variable), content of, 710

TID thread attribute, description of, 679

time command, invoking, 595–596

time.jsp program

code for, 330

- files for programming JSP tags library
 - used by, 335
- TIMESTAMP, role in DBGUIDE
 - database, 228
- timestamping, synchronizing for use with make utility, 586
- timeTag.java program, code for, 330–331
- TLD (Tag Library Descriptor) file
 - code for, 331–332
 - purpose of, 329–330
- TME (Tivoli Management Environment)
 - versus WebSphere domains, 117
- /tmp/foo.txt file, fetching contents of, 527–536
- .toc file, contents of head of, 29–30
- top command
 - description of, 667
 - invoking, 685
 - using with UNIX, 199
- touch commands, example of, 366
- touchreload script, testing servlet reloading with, 366
- trace Lynx option, description of, 386
- tracefile log file, troubleshooting WAS startup failure with, 196
- TRAS WAS component ID, description of, 648
- Tree view, using in WebSphere Domain, 84
- tree2ear script, code for, 342
- trees, diagram of, 581
- @TRIGARG WASMON s-var (static internal variable), content of, 710
- TriggersPool directory, displaying for use with WASMON, 661–662
- @TRIGNUM WASMON s-var (static internal variable), content of, 710
- @TRIGURI WASMON s-var (static internal variable), content of, 710
- type 2 JDBC app driver, overview of, 33–34
- type 3 JDBC net driver, overview of, 34

U

- u <uri> option of ZappUrl
 - stress-tester, description of, 600
- U character, appearance in
 - messages, 290
- UDB authority on UNIX, overview of, 224–225
- UDB configuration repository,
 - installing WAS with, 88–89
- UDB database parameter settings,
 - default values for, 689
- UDB installations
 - failure of, 32
 - testing, 29
- UDB product binary, confirming
 - successful installation of, 31
- UDB servers, communicating with
 - WAS servers, 615
- UDB stored procedures, Apache
 - SOAP deployment descriptor artifact for, 507–508
- UDB (universal database)
 - accessing with Perl DBI, 234–235
 - accessing with servlets, 293–299
 - binary distribution and basic
 - installation of, 22–24
 - classifying JDBC driver types
 - for, 33–34
 - installing components of, 24–29
 - installing on AIX, 29–32
 - WAS compatibility with, 21–22
- UDB usernames and groups, table of, 23
- UDB v71 or v7.2, code distribution
 - web site for, 726
- UMASK, using with user
 - processes, 119
- Unicode files, code distribution web site for, 727

804 IBM WebSphere Application Server Programming

- Unicode tables, downloading, 739
 - Unicode, use of MrUnicode with, 12, 738–741
 - UNIX
 - automating WSCP output with, 183–184
 - creating user accounts on, 578
 - definition of services in, 205
 - performance commands on, 667
 - replicating directories in, 589
 - setclasspath.sh script for, 55
 - top command used with, 199
 - UDB authority on UNIX, overview of, 224–225
 - WAS programming on, 14
 - xargs command used by, 205
 - UNIXLoginModule API in Java 2 SDK v1.4, description of, 553
 - UNIXSystem API in Java 2 SDK v1.4, description of, 553
 - unmarshalling, role in programming
 - DataSetSerializer interface, 518
 - update_emppwd_dbguide.properties file, code for, 547
 - UpdateJSP servlet, code for, 325
 - UpdateTeller9 servlet, code for, 474–476
 - URIs, printing, 614
 - URL names, requesting in browsers, 350
 - URL requests, placing in shell script for tellers, 595
 - URL rewriting, overview of, 417–419
 - URLs
 - obtaining with GET command, 597
 - timing with time command, 596
 - user accounts, creating on UNIX, 578
 - User Environment Profile dialog box, displaying in Windows NT, 64
 - user environment, setting on Windows NT, 63–64
 - user home directory, accessing on AIX, 50
 - USER ID field in Employee Records, methods of, 256
 - user logins, identifying, 385
 - user profile, execution of, 50
 - USERNAME column in SESSIONS table, details of, 434
 - users
 - authority of, 545
 - forking concurrent processes for, 598–599
 - securing registration of, 560
 - utility scripts, availability of, 12–13
- ### V
- V in setw(V)(os).sh script, meaning of, 105
 - v option
 - of SharkUrl stress-tester, 607
 - of ZappUrl stress-tester, 600
 - v option, using with ps command, 668
 - verbose argument for Java machine, purpose of, 213
 - version control commands, role in system environment for WAS deployment, 47
 - virtual hosting, setting for WAS, 96–102
 - virtual hosts
 - creating for WAS v3.5, 98
 - name resolvability of, 102
 - reasons for setting of, 97
 - virtual hosts default_host aliases, setting in WAS v4, 101–102
 - Visual Age products, advisory
 - about, 18
 - vj option
 - of SharkUrl stress-tester, 607, 611
 - of ZappUrl stress-tester, 600
 - vmstat command
 - description of, 667

- monitoring memory and CPU with, 671–673
- vmtune command, description of, 667
- void removeAttribute() method, role in JSP programming, 326
- void setAttribute() method, role in JSP programming, 326
- vv option
 - of SharkUrl stress-tester, 607
 - of ZappUrl stress-tester, 600

W

- W in WAS LEDs, meaning of, 647
- \w and \W Perl code, match for, 760
- w_ipstat script, monitoring Windows NT network usage with, 675–678
- w_mn script, monitoring Windows NT memory with, 673–675
- w122 option for setj.sh script, purpose of, 61
- w130 option for setj.sh script, purpose of, 61
- .war file type
 - location in WAS v4, 105
 - overview of, 140
- Warn.jsp program, code for, 327
- WAS administrative clients, launching, 123–125
- WAS administrative practices
 - archiving WAS directory, 218–219
 - archiving WAS directory on Windows NT, 219–220
 - checking on repository, 217
 - invoking showCfg, 216–217
 - monitoring application archives, 217–218
- WAS AE Administrative Console, invoking, 210–211
- WAS AE (Advanced Edition)
 - backing up and restoring, 733–734
 - overview of, 5
- WAS AE installation, steps for, 88
- WAS AE v3.5

- changing name of WebSphere domain names in, 122
- code distribution web site for, 726
- and v4 administrative consoles, 82–84
- WAS AE v4
 - setting admin_host to "localhost" for protection of, 97–99
 - setting JDBC driver for, 94
 - start.Server.sh and stopServer.sh scripts in, 136
 - storage of WebSphere domain names in, 121
- WAS AEd (Advanced Edition Developer), overview of, 5–6
- WAS AEs
 - generalizing J2EE application development under, 730
 - hierarchy tree of administrative server of, 424–425
 - identifying DTDs in, 352
 - redefining domain names in, 122–123
 - setting up session persistence in, 427–436
 - startup Java machine defined in startServer.sh script on, 361–362
- WAS AEs administrative console, enabling cookies in, 399
- WAS AEs (Advanced Edition Single Server), overview of, 5
- WAS AEs v4, code distribution web site for, 726
- WAS commands, aliasing, 108
- WAS Commands list, remembering, 107
- WAS components
 - ID numbers for, 646–648
 - identifying, 647–648

806 IBM WebSphere Application Server Programming

- persisting sessions with, 433
- result of making changes in, 378–381
- WAS configuration data, backing up and restoring, 732
- WAS configuration repository, diagram of, 21
- WAS configurations
 - checking functionality of, 145
 - extracting with Perl and WSCP, 177–185
- WAS containment, monitoring with WASMON, 654
- WAS development environment, setting, 105–109
- WAS directory, archiving, 218–219
- WAS Enterprise Edition (EE), overview of, 5–6
- WAS environment variables, setting globally, 55
- WAS events, filtering with WASMON, 654–659
- WAS failure to start, troubleshooting, 196–197
- WAS for IBM z/OS and IBM OS/390, overview of, 6
- WAS installations
 - backing up, 218
 - determining availability of HTTP Web Servers to be plugged into, 86–87
 - replicating on several machines, 112
 - testing with undocumented shortcut, 109–111
- WAS instances
 - assigning ports for, 583–585
 - modifying directory of, 583–585
- WAS Java machines
 - killing, 205
 - parametric tune-up of, 212
 - renicing, 205
- WAS JSP processing of JSP pages, stages undertaken by, 321
- WAS log files, purpose of, 136
- WAS parsing for web.xml files, overview of, 351–355
- WAS performance, sniffing network for measurement of, 614–616
- WAS processes
 - acting on groups of, 205
 - conventions of, 80
 - demystifying, 203–205
 - distribution by WebSphere domains, 118
- WAS production development, separating database repository in, 86
- WAS programming on UNIX, overview of, 14
- WAS redirection to standard input/output, overview of, 289–290
- WAS repository
 - checking, 217
 - versus DataSources in WAS configurations, 85
 - managing, 214–216
 - separating, 86
- WAS servers, communicating with UDB servers, 615
- WAS session identifier visibility to CGI programs, overview of, 408–409
- WAS startup classes, explanation of, 360
- WAS startup classloader, adding Jar archives to, 364
- WAS startup classpath. *See also* classpaths
 - loading classes in, 362
 - overview of, 362–363
- WAS startup configuration, disabling JIT from, 213
- WAS startup Java machine, overview of, 211–216
- WAS startup scripts

- advisory about configuration files used by, 218
 - location of, 188
- WAS suppressing prereq_checker in, 89–90
- WAS system policy, adding java.security.policy as, 541
- WAS threads, monitoring script for, 12
- WAS tools
 - Log Analyzer, 160–161, 196
 - Resource Analyzer, 161–164
- WAS trial versions, components of, 81
- WAS v3.5
 - comparing to WAS v4, 131–134
 - determining availability of HTTP Web Server for, 86
 - grouping of Web Resources in, 99–100
 - setting JDBC driver for, 95
 - setting virtual hosting in, 98–101
 - specification of error and I/O files by, 197–198
 - versus WAS v4 software components, 81
- WAS v4
 - commercial products of, 5–6
 - comparing to WAS v3.5, 131–134
 - deprecating Web Resources in, 100–101
 - determining availability of HTTP Web Server for, 87
 - DrAdmin utility used with, 138–139
 - EJB level 1.1 supported in, 132
 - exploring via Petstore sample, 150–159
 - file types and locations in, 104–105
 - JSP levels supported in, 132
 - monitoring with WASMON, 754
 - revealing port activities for, 209
 - SEAppInstall.sh script used with, 141–142
 - setting virtual hosts default_host aliases in, 101–102
 - setw4lin.sh script code in, 106–107
 - use of models and clones with application servers in, 133–134
- WAS v4 AEs Single Server, installing, 90–91
- WAS v5
 - determining support through DTDs, 764–766
 - packaging of, 6
 - reconsidering scripts with, 766–767
 - special consideration of, 11–12
 - support for J2EE API 2.3, 766
- WAS versions for Java machine, tables of, 44–45
- WAS Web Container, checking functionality of, 216–217
- WAS web container, checking when in parole with SUPERVISOR, 707–708
- WAS (WebSphere Application Server)
 - advisory about deploying web applications with, 552
 - Apache SOAP deployment descriptors used by, 506–508
 - applet clients for, 143
 - awakening to start logging activity, 722
 - checking versions and release level/dates of, 194–196
 - classloading hierarchy of, 359
 - classloading in, 360
 - compatibility with UDB, 21–22
 - configuring common resources for, 93–94
 - database requirements for, 84–85
 - default ports used by, 206
 - deploying JAAS programs in, 538–539
 - EAR (Enterprise Application Archive) used with, 139–143

808 IBM WebSphere Application Server Programming

- enabling JAAS in, 525
- establishing successful database connectivity with, 86
- file locations on, 104–105
- gathering status of, 136
- GUI interface dependency of, 82
- installing with UDB configuration repository, 88–89
- Java machine availability on, 87–88
- loading order versus visibility order in, 379
- Log Analyzer, 201–203
- log files and startup messages in, 196–203
- monitoring, 12
- monitoring nodes within, 173–174
- monitoring ports in, 207–210
- operating system-level requirements for, 82
- overview of, 4–5
- performing silent installation of, 90
- ports used by, 130
- preinstallation verification of, 85–88
- requirements for, 81–85
- serving snoop test servlet on, 102–104
- setting desktop for, 108–109
- setting JDBC driver for, 94–96
- setting virtual hosting for, 96–102
- starting, 188–189
- startup and first-time configuration of, 92–93
- stopping, 189–190
- testing, 190–194
- testing tools for, 144–145
- uninstalling, 111
- WSCP (WebSphere Control Program) used with, 137–138
- XML grammar used with, 125
- WAS (WebSphere Application Server) repositories, purpose of, 20
 - @WAS_ALTERNATIVE WASMON s-var (static internal variable), content of, 710
 - WAS_CLASSPATH environment variable, defining in setupCmdLine.sh script, 361
 - WAS_EXT_DIRS environment variable, defining in setupCmdLine.sh script, 361
 - was_home directory, products for, 78
 - @WAS_HOSTNAME WASMON s-var (static internal variable), content of, 710
 - @WAS_NODE WASMON s-var (static internal variable), content of, 710
 - @WAS_URI WASMON s-var (static internal variable), content of, 710
 - WAS01 databases, testing connections to, 111
 - WAS01, increasing Heap Size of, 89
 - \$WASCOMMANDS command, invoking, 107
 - wasconnect.java command, testing WAS connections with, 109–111
 - The WASD Application, browsing, 268
 - WASDE Environment, explanation of, 38
 - WASDG application
 - build tree for JAAS-enabled Logout servlet, 546
 - installing, 490
 - J2EE tree structure of, 565
 - JAAS-enabling logout of, 544
 - monitoring with WASMON, 721
 - stress-testing with SharkUrl, 606–616
 - timing teller login to, 595
 - WASDG code v1, code distribution web site for, 726
 - WASDG development tree
 - merging EJBs to, 490–492
 - using symbolic links to link EJBs to, 492–493
 - WASDG environment

- overview of, 52–61
 - printing with chekenv script, 748–749
- wasdg_components_r100.rec file, code for, 720
- wasdg_leddb_s_r100.rec file, code for, 720
- WasdgBeans.ear enterprise application
 - assembling, 496–498
 - generating and registering, 483–488
 - generating logs during creation of, 486
 - registering in WAS AEs
 - server-cfg.xml, 487
 - starting with WAS EJB
 - Engine, 487
 - testing, 489–493
 - testing beans in, 491
- WasdgBeans.jar EJB module, generating, 481–482
- wasdg.ear tree
 - archiving, 344
 - collapsing into wasdg.ear Jar archive, 342
 - listing with find and wc commands, 344
- wasdg.ear web application, building and
 - deploying, 265–269
- WasdgException class
 - code for, 635–637
 - format of, 633
 - purpose of, 622–623
 - testing, 639
- WasdgMessages class, code for, 638–639
- wasdgsoap.ear application
 - deploying, 500–505
 - illustration of, 501
 - testing, 504–505
 - uninstalling and installing, 505
- WasJndi program, code for, 431–432
- WASLED/WASMON application
 - features of, 12
 - installing, 729
 - monitoring web applications with, 719–721
- WASLED/WASMON v1.2.1, code
 - distribution web site for, 727
- waslogbr command
 - analyzing activity.log file with, 201–203
 - invoking, 160
- WASMON
 - b-var (boolean) internal variable, 708–709
 - capabilities of, 654
 - d-var (differential internal variable), 708–709
 - detecting errors in web
 - containers with, 717
 - filtering events and e-mailing
 - users during use of, 660
 - filtering WAS events and firing
 - action with, 654–659
 - g-var internal variable, 708–709
 - installing and starting, 649–653
 - internal variables, 758–760
 - messages pane in, 650–653
 - monitoring WAS containment
 - with, 654
 - monitoring WASDG application
 - with, 721
 - performing generic monitoring
 - with, 719
 - risk management with, 697–701
 - running considerations, 721–722
 - sending alerts by means of e-mail
 - during use of, 659–662
 - starting and configuring, 754–760
 - starting in supervisor mode, 697–698
 - statewide interaction with
 - operator, 707
 - in supervisor mode monitoring
 - WebSphere region, 696

810 IBM WebSphere Application Server Programming

- trapping and monitoring WASDG application exceptions with, 634
- triggering scripts during use of, 660–662
- user input pane in, 650–651
- using logical expressions with, 756–758
- WASLED activities pane in, 650–652
- WASMON console, displaying, 650
- WASMON delegation, implementing with wasmonhelper program, 713–715
- WASMON internal variables. *See* b-var WASMON boolean internal variables; d-var WASOMON differential internal variables; g-var WASMON global internal variables; s-var WASMON static internal variables
- WASMON server, activating and connecting to, 653
- wasmon user account, delivering e-mail to, 702
- WASMON v1.2.1 configuration file, directives used in, 755–756
- wasmon.conf file
 - code for, 657
 - defining WASMON e-mail file and SUPERVISOR configuration property in, 701
 - directives of, 654–655
 - Logical Alert directives defined in, 663
 - using internal variables in, 715–718
- wasmonhelper program
 - considering, 717–718
 - implementing WASMON delegation with, 713–715
- wasmonhelper.conf file
 - code for, 714–715
 - directives mapped to corresponding variable names, 758–759
- wasmontkt report, example of, 657–658
- wasports script, code for, 582–583
- wasreport.cron program, code for, 184
- wc command, using with find command to list wasdg.ear tree, 344
- WCMD WAS component ID, description of, 648
- wcspMsg example, code for, 174–175
- web application deployment descriptors, generating, 271–274
- web application descriptor, registering tags libraries in, 335
- web application sessions. *See* sessions
- web applications
 - adding ZenFile servlet to, 539–540
 - advisory about deploying with WAS, 552
 - automating with make utility, 568–571
 - descriptive elements in, 352, 353–355
 - documentation of, 262
 - documenting packages of, 589
 - getting classpaths from within servlets for, 364–365
 - modules in, 360
 - monitoring with WASLED/WASMON, 719–721
- web containers
 - detecting errors in using WASMON, 717
 - role in WebSphere domains, 128–129
- web module level, result of changes made at, 380
- web modules
 - explanation of, 320, 360
 - implementation of, 263
 - management of, 267

- Web Resources
 - deprecating in WAS v4, 100–101
 - grouping in WAS v3.5, 99–100
- web servers, stress-testing, 596
- web service, explanation of, 520
- webgenxml script, changing to handle context init parameter, 302–303
- WebSphere administrative console, overview of, 137–138
- WebSphere domain names, defining, 122–123
- WebSphere domains
 - administering, 123–126
 - benefits of, 118–121
 - checking nodes in, 193–194
 - configuring common resources for, 93–94
 - conventions of, 80
 - defining name for, 121–123
 - installing multiple instances on same server, 581–586
 - measuring performance in, 164
 - naming, 586
 - overview of, 116–118
 - securing, 121
 - versus SP (Scalable POWERparallel) system nodes management, 117
 - structure of, 126–130
 - versus TME (Tivoli Management Environment), 117
- WebSphere nodes. *See* nodes
- WebSphere regions
 - monitoring in WASMON supervisor mode, 696
 - overview of, 114–116
- web.xml deployment descriptor
 - for soap.war web module, 503–504
 - viewing from administrative console, 269
- web.xml file type, location in WAS v4, 105
- web.xml files
 - browsing, 269
 - document type defined in, 352
 - inserting context init parameters in, 301
 - role in sampleApp.ear file, 148–149
 - WAS parsing for, 351–355
- web.xml web application deployment descriptor
 - describing sampleTags.tld file in, 335
 - generating, 271–274
- <welcome-file-list> (?) element, meaning in web applications, 354
- welcome.jsp file, code for, 336–337
- welcomeTag.java library
 - code for, 336
 - compiling and testing, 337
- wildcard general alias (*), advisory about, 97
- Windows NT
 - archiving WAS directory on, 219–220
 - batch commands in, 62
 - confirming installation of IHS on, 42
 - IHS installation directory for, 39
 - interpreting ntclasspath.sh script with, 55
 - Java machine installation for, 46
 - memory usage considerations, 673–675
 - mounting tools directory for, 52
 - network usage considerations, 675–678
 - performance command requirements for, 668
 - setting user environment on, 63–64

812 IBM WebSphere Application Server Programming

- shell interpreter on, 64–65
 - tabulation of `setw(v)(os).sh`
 - script in, 107
 - uninstalling WAS on, 111
 - user environment overview, 62–72
 - Windows NT PATH, designation of, 80
 - WINT WAS component ID, description of, 648
 - WOBA WAS component ID, description of, 648
 - workstation processes list, grepping, 70
 - WPRS WAS component ID, description of, 648
 - Write contents, changing for sessions, 438–439
 - WSCL WAS component ID, description of, 648
 - WSCP cache, updating, 171
 - wscp command
 - overview of, 169
 - running servlets with, 177–178
 - testing WAS with, 192
 - WSCP message codes, overview of, 185–186
 - WSCP output
 - automating, 183–184
 - parsing, 178–185
 - WSCP prompt, appearance of, 169–170
 - WSCP properties, table of, 172
 - WSCP property file, overview of, 171–172
 - WSCP WAS component ID, description of, 648
 - WSCP (WebSphere Control Program)
 - conversing with, 172–173
 - exception errors generated in, 171
 - installing on workstations, 175
 - introduction to, 168–169
 - overview of, 137–138
 - running from Perl, 177–178
 - starting, 169–173
 - and Tcl, 173–175
 - testing WAS with, 192
 - using with administrative console, 171
 - wscp_home directory, products for, 79
 - wscp_Servlet-list.pl program, code for, 177–178
 - WscpCommand interface, overview of, 176–177
 - WscpCommand(String *n*, String *p*) constructor, purpose of, 176
 - WSCPNNNN[s] notation, meaning of, 185
 - WscpNodeList.java program, code for, 176–177
 - WscpQualifiedName() class, purpose of, 176
 - WscpResult evalCommand(String command) method, purpose of, 176
 - ./wscp.sh command, invoking, 169
 - wscp.sh script, invoking, 169
 - WSVR WAS component ID, description of, 648
 - WTRN WAS component ID, description of, 648
 - WTSK WAS component ID, description of, 648
 - WWLM WAS component ID, description of, 648
- ### X
- xargs command, using with UNIX, 205
 - .xmi extension, locating files terminating with, 340
 - *.xml configuration files, updating, 210

.xml documents, operators describing elements
in, 353
XML grammar used with WAS, example
of, 125
XMLC WAS component ID, description
of, 648
\$XMLCONFIG item, exercising, 126
XMLConfig.sh script, syntax of, 125–126

Y

Y notation, meaning of, 378–379
Yo.java program, code for, 371

Z

ZappUrl script
implementing for stress-testing,
599–606
stress-testing web server with,
602–606
ZenFile.java servlet
adding to web applications,
539–540
class files in, 537
testing, 540

INTERNATIONAL CONTACT INFORMATION

AUSTRALIA

McGraw-Hill Book Company Australia Pty. Ltd.
TEL +61-2-9900-1800
FAX +61-2-9878-8881
<http://www.mcgraw-hill.com.au>
books-it_sydney@mcgraw-hill.com

CANADA

McGraw-Hill Ryerson Ltd.
TEL +905-430-5000
FAX +905-430-5020
<http://www.mcgraw-hill.ca>

GREECE, MIDDLE EAST, & AFRICA (Excluding South Africa)

McGraw-Hill Hellas
TEL +30-1-656-0990-3-4
FAX +30-1-654-5525

MEXICO (Also serving Latin America)

McGraw-Hill Interamericana Editores S.A. de C.V.
TEL +525-117-1583
FAX +525-117-1589
<http://www.mcgraw-hill.com.mx>
fernando_castellanos@mcgraw-hill.com

SINGAPORE (Serving Asia)

McGraw-Hill Book Company
TEL +65-863-1580
FAX +65-862-3354
<http://www.mcgraw-hill.com.sg>
mghasia@mcgraw-hill.com

SOUTH AFRICA

McGraw-Hill South Africa
TEL +27-11-622-7512
FAX +27-11-622-9045
robyn_swanepoel@mcgraw-hill.com

SPAIN

McGraw-Hill/Interamericana de España, S.A.U.
TEL +34-91-180-3000
FAX +34-91-372-8513
<http://www.mcgraw-hill.es>
professional@mcgraw-hill.es

UNITED KINGDOM, NORTHERN, EASTERN, & CENTRAL EUROPE

McGraw-Hill Education Europe
TEL +44-1-628-502500
FAX +44-1-628-770224
<http://www.mcgraw-hill.co.uk>
computing_neurope@mcgraw-hill.com

ALL OTHER INQUIRIES Contact:

Osborne/McGraw-Hill
TEL +1-510-549-6600
FAX +1-510-883-7600
<http://www.osborne.com>
omg_international@mcgraw-hill.com