Index

A
About mode, 55
abstraction, 207
access control, Ajax and, 448, 462
access control lists (ACLs), 22
accessors, Workplace Forms and, 415
ACORD 111 XML messages, Workplace
Forms and, 415
action calls, 52–53, 53
action events, JavaServer Faces (JSFs) and,
135, 152–156, 152-156
action handling, 52–53, 53
Action List builder, Portlet Factory and,
221, 224–225, 224, 225, 237–238, 238,
268, 268, 279, 279
action phase
inter-portlet communication and,
310–312, 311
JavaServer Faces (JSFs) and, 138,
164–165
persistent data and, 113
ActionRequest/Response, 164
ActionResponse, 167
ActionURLs, 63
active credentials, Credential Vault and,
367–374, 370
Active Server Pages (ASPs), Portlet
Factory and, 214
ActiveX, Ajax and, 434
Adaptive Path, LLC, 433
addCustomer, Portlet Factory and example,
252
adding/removing projects in RAD, 44, 45
admin console, WAS, 200, 200
administrative slots, Credential Vault and,
356, 374–377, 377
aggregate portals, 185, 186
aggregation, page, 446
in inter-portlet communication, 310
Ajax, 93, 96, 96, 102–103, 103, 433–464
access control and, 448, 462
accessibility, search and versioning using,
440, 449
ActiveX and, 434
advantages, strengths, weaknesses of,
439–441
AJAX Toolkit Framework (ATF) for, 462
alternatives to, 441–442
ATOM feeds and, 446

NOTE: Boldface numbers indicate illustrations or code listing
Ajax, continued
  browser support for, 434, 437
  client-side page aggregation and, 446
  concepts of, 434
  data values coded in, 451, 452
  Document Object Model (DOM) and,
  434, 435, 445, 454, 459, 459
  Dojo toolkit for, 460–461, 461
  dynamic HTML (DHTML) and, 440
  Eclipse toolkit for, 462
  example of, 449–460
  Extensible Application Markup Language (XAML) for, 442
  Flex vs., 439, 441–442
  frameworks for, 460–462
  hidden frame technique vs., 435
  HTML and, 442, 446, 457, 459
  HTTP and secure HTTP (sHTTPS) in, 463
  integrated development environments (IDEs) for, 440
  introduction to, and development of, 433–436
  JavaScript and, 434, 435, 437, 440, 445, 453–460, 463
  JavaScript function in, 457–460, 458–459
  JavaScript Object Notation (JSON) and,
  443, 459
  JavaServer Page (JSP) using, 453–456, 455–447
  JSR 286 and, 446, 469
  Kabuki toolkit for, 461–462
  Lightweight Third-Party Authentication (LTPA) and, 462
  logic for requests in, 452, 452
  Macromedia Flash vs., 441
  markup languages and, 442
  mashups using, 441–442
  namespace encoding nd, 457
  Open Ajax and, 439, 441, 461–462
  performance using, 448
  portal environments using, 443–449, 444
  portlets using, 446–449, 450–454, 450–454
  predecessor techniques to, 434–345
  processing flow of, 436–443, 437
  Rational Application Developer (RAD) using, 450–454
  Representational State Transfer (REST) services and, 446
  response format in, 453, 453
  RSS feeds in, 446
  security and, 440, 448, 462–463
  separation of client state/URL in, 440
  separation of content and presentation using, 440, 448
  server load and efficiency using, 440
  server logic and, 440
  server response format using, 442–443
  servlet execution and, 447
  state and, in client/server, 448
  state and, in portlets, 447
  toolkits for, 439
  URLs and, 434–435, 447, 449
  use of, 436
  user interfaces in, 438–439, 47
  Web Services for Remote Portlets (WSRP) remote service and, 456
  WebSphere Application Server (WAS) and, 462
  WebSphere Portal and, 444–445
  XML and, 440, 442, 445, 451–452, 451, 452, 459
  XML User Interface Language (XUL) and, 442
  XMLHttpRequest objects and, 434, 436, 436
  436, 444, 458–460, 458, 469
  AJAX Toolkit Framework (ATF), 462
  analysis tools, Service Oriented Modeling and Analysis (SOMA) and, 179
  anatomy of a portlet, 26–28, 26
  Apache, 483
  Apache Struts Bridges Framework, 94
Business Intelligence and Reporting Tools (BIRT), 172
business service repositories, 208
business-to-business (B2B) portals, 182
business-to-consumer (B2C) portals, 182
business-to-employee (B2E) portals, 182–183
Button builder, in Portlet Factory, 273, 274, 278–280, 280, 281
buttons
  action events in JSF for, 152–156, 152–156
  Portlet Factory and, 241, 273, 278, 279, 280, 281
C
C language, 381
C Programming Language, 25
caching, 197–206, 198, 199, 470–471
  DistributedMap cache in, 197, 198–199, 198, 199
  DistributedObjectCache in, 197
dynamic cache service for, 197
Java Virtual Machine (JVM) and effect of, 197
JNDI access to, 198
SystemOut.Log for, 202
viewing the DistributedMap in, 203–206, 203–204, 205, 206
WebSphere Application Server (WAS) for, 197, 200
Calendar Portlet, 64–86
  actions in, 66
  adding ToDo items to, 85–86, 85, 86
code for, 71–79, 72–74
customization in, 66
data sharing in, 65, 66–68, 66, 67
deploying, 85–86, 85, 86
deployment descriptors for, 83, 84
Edit mode JSP for, 82–83, 82–83
internationalization and localization of, 123–131
JSP modifications for, 125–130, 126–130
locale selection for, 123–124, 124
  properties files in, 125, 130–131
  resource bundles created in, 124–125, 124, 126, 129–130
  specifying portlet support for, 123
Java bean code for, 77, 78
Java Standard Tag Library (JSTL) tags in, 65, 68–69, 68, 81
JavaServer Pages (JSPs) in, 65, 79–81, 80
Model View Controller (MVC) pattern in, 65
  packaging of, 83, 84
  render parameters and, 65, 69–71, 70–71
  running, 85–86, 85, 86
tasks for, 65, 65
ToDo portlet code for, 75–77, 75–77, 79, 79
View mode in, 66
View mode JSP for, 81–83, 81, 82–83
  viewDate JSP code in, 79, 80
Cancel button, 158–159, 158, 159
caption attributes, inter-portlet communication and, 332
Chart builder, in Portlet Factory, 245, 259
choreography, process, 208
CityListPortlet example, 306–308, 306, 321–326
classes, 108
  JavaServer Faces (JSFs) and, and RAD, 142–143, 142, 143
Classic Models business template, 172–173, 173
  click-to-action (C2A), 276
  inter-portlet communication and, 305
client-capabilities access, 470
code generators, 213–214. See also Portlet Factory
<table>
<thead>
<tr>
<th>Term</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>collaborative portals</td>
<td>185, 186</td>
</tr>
<tr>
<td>commitment control, Portlet Factory and</td>
<td>278–279</td>
</tr>
<tr>
<td>Compact HTML (cHTML)</td>
<td>134</td>
</tr>
<tr>
<td>compiling code</td>
<td>29–30, 29</td>
</tr>
<tr>
<td>Component Object Model (COM)</td>
<td>381</td>
</tr>
<tr>
<td>composite applications, Service Oriented Architecture (SOA) and</td>
<td>181, 187–189, 188</td>
</tr>
<tr>
<td>Composite Capabilities/Preference Profiles (CC/PP)</td>
<td>134, 470</td>
</tr>
<tr>
<td>compression</td>
<td>169</td>
</tr>
<tr>
<td>computed-aided design (CAD) software</td>
<td>211–215, 213</td>
</tr>
<tr>
<td>configuration, SystemOut.Log to check</td>
<td>202</td>
</tr>
<tr>
<td>connection pools, database</td>
<td>195</td>
</tr>
<tr>
<td>connections or persistent channels, in inter-portlet communication</td>
<td>313</td>
</tr>
<tr>
<td>Console view in RAD</td>
<td>46, 46</td>
</tr>
<tr>
<td>container, portlet, 9–10, 10, 10</td>
<td>53 modes and, 54</td>
</tr>
<tr>
<td>Content Access Services</td>
<td>171</td>
</tr>
<tr>
<td>context, portlet, 15, 15</td>
<td></td>
</tr>
<tr>
<td>contextual portals, 184</td>
<td></td>
</tr>
<tr>
<td>controller, portlet, 27</td>
<td></td>
</tr>
<tr>
<td>action phase in, 310–312, 311</td>
<td></td>
</tr>
<tr>
<td>architecture of, 315–320, 317</td>
<td></td>
</tr>
<tr>
<td>binding information in, 311, 329–333</td>
<td></td>
</tr>
<tr>
<td>329, 330, 333</td>
<td></td>
</tr>
<tr>
<td>building, using RAD, 320–326</td>
<td></td>
</tr>
<tr>
<td>caption attributes in, 332</td>
<td></td>
</tr>
<tr>
<td>connections or persistent channels between</td>
<td>313</td>
</tr>
<tr>
<td>deployment descriptors for, 320</td>
<td></td>
</tr>
<tr>
<td>guard variables and, 337</td>
<td></td>
</tr>
<tr>
<td>hypertext references (HREFs) in, 340</td>
<td></td>
</tr>
<tr>
<td>Java Naming and Directory Interface (JNDI) lookups in, 337–338</td>
<td></td>
</tr>
<tr>
<td>JSR 168 and, 309</td>
<td></td>
</tr>
<tr>
<td>messaging and, 308</td>
<td></td>
</tr>
<tr>
<td>page aggregation in, 310</td>
<td></td>
</tr>
<tr>
<td>properties in, input and output, 309–312, 311, 330–333</td>
<td></td>
</tr>
<tr>
<td>Portlet Wiring Tool for, 313–315, 314</td>
<td></td>
</tr>
<tr>
<td>portType in, 324–325, 324, 328, 328</td>
<td></td>
</tr>
<tr>
<td>properties in, 326–337, 336–337, 338</td>
<td></td>
</tr>
<tr>
<td>source portlet code for, 336–337, 339</td>
<td></td>
</tr>
<tr>
<td>source portlet doView method in, 338–339, 339</td>
<td></td>
</tr>
<tr>
<td>source portlet init method for, 337–338, 338</td>
<td></td>
</tr>
<tr>
<td>source portlet processAction method in, 340–341, 340–341</td>
<td></td>
</tr>
<tr>
<td>source portlet WSDL for, 326–337, 326, 327–330, 332</td>
<td></td>
</tr>
<tr>
<td>target and source portlets in, 314–315, 320–326</td>
<td></td>
</tr>
<tr>
<td>target portlet JSP for, 344–347, 345–346, 347</td>
<td></td>
</tr>
<tr>
<td>WAR files and, 318–320</td>
<td></td>
</tr>
<tr>
<td>Web Service Description Language (WSDL) files and, 312, 317, 318, 320, 321–337, 322, 323–324, 326</td>
<td></td>
</tr>
<tr>
<td>WebSphere Portal Core in, 316–320, 317</td>
<td></td>
</tr>
<tr>
<td>WebSphere Portal Datastore in, 317–320, 317</td>
<td></td>
</tr>
<tr>
<td>NOTE: Boldface numbers indicate illustrations or code listing</td>
<td>493</td>
</tr>
</tbody>
</table>
Cooperative Source builder, in Portlet Factory, 276
coordination concepts in JSR 286, 466–468
core, WebSphere Portal Core in, 316–320, 317
Craigslist, 441
createCredentialSlot method, 359–360, 360
Credential Vault, 171, 349–377
adding a vault slot in, 375, 375, 376
administrative slots in, 356, 374–377, 377
administrator-managed segments in, 355–356, 359
authentication and, 349
authentication proxy servers in, 350
authorization and, 349
binding rules or policies concerning portlet access in, 363
configuration of segments in, 354
configuring a slot in, 375–376, 375
createCredentialSlot method in, 359–360, 360
credentials defined for, 352–353, 358, 364–374
HTTP and secure HTTP (sHTTPS) in, 350, 369–374
HttpBasicAuth active credential example in, 369–374, 371–372, 373
IDs and passwords in, 349, 354
inactive credentials in, 367–374, 370
opaque identifiers in, 359
passive credentials in, 364–367
portal administration link in, 374, 374
private slots in, 357
resource name of slots in, 357, 359
retrieving of, in a portlet, 353, 353
secrets in, 352–353, 354, 361
segments in, 354, 355–356, 359, 360
setting/getting credentials in, 358, 358
shared slots in, 356
single sign on and, 349–351, 351
slotID in, 359, 359, 361–363
structure of, 354–364, 354
system slots in, 356, 374–377
Uniform Resource Identifiers (URIs) and, 350
user-managed segments in, 355–356, 359, 360, 360
vault adapters in, 354
VaultSlotName parameter in, 357, 357–358
credentials. See Credential Vault
cross-site scripting, 109–110
customer lookup cache, 198–199, 198, 199
Customer Relationship Management (CRM), 182
CustomerData model, Portlet Factory and, 252, 256, 256
CustomerList portlet creation, Portlet Factory and, 238–239, 239
CustomerListView model, Portlet Factory and, 249
customization, 51, 56–59, 112, 259
Config mode and, 58
flyweight pattern and, 56, 57, 59
inheritance settings and, 56, 57
levels of, 57, 57, 58–59
patterns and, 56
portlet definitions and, 57–58
portlet entities in, 58
Portlet Factory and, 259
portlet windows and, 58–59, 471
preferences and, 60–61, 66

D
Dashboard Framework builder, in Portlet Factory, 245, 259
data access layer, 175
Data Access Object (DAO), 175, 474
Data Access Service example, 195–206
caching in, 197–206 198, 199. See also caching
connection pools and, 195
customer lookup cache in, 198–199, 198, 199
DistributedMap cache in, 197, 198–199, 198, 199
DistributedMap cache setup in WAS for, 200–202, 200–202
doView method in, 203–204, 203–204
local service layer in, 195
portlet access to, 195
processAction in, 204–205, 205
security and portlet access in, 195
singleton for, 195–197, 196, 195
viewing the DistributedMap in, 203–206, 203–204, 205, 206
WebSphere Application Server (WAS) for, 197, 200
data access, singleton for, 474–475
Data Column Modifier (DCM) builder, in Portlet Factory, 239–241, 240, 241, 245
data component setup for sample apps, 473–487
class loader in, 478–479, 478, 479
custom setup options for, 482–487
Data Access Objects (DAOs) and, 474
Derby database custom setup in, 483–487, 484, 486
extended library for, 475–480, 476–480
iBATIS and, 473–474, 487
interactive Java (ij) tool in, 485–487
Java Database Connectivity (JDBC) and, 474
modifying source code in, 487
quick start setup in, 474–482
Rational Application Developer (RAD) and, 481–482, 482
saving configurations in, 480, 480
singleton model of data access in, 474–475
SQL Maps and, 474
testing setup in, 481, 481
unzipping files for, 475
WebSphere Application Server (WAS) data sources and, 487
Data Field Modifier (DFM) builder, in Portlet Factory, 243, 244, 266–267, 269, 269
Data Hierarchy Modifier builder, in Portlet Factory, 265–266, 266, 271
Data Integration builders, in Portlet Factory, 245
data models, 59–62, 60
creating in Portlet Factory, 246–248
data sharing, 65, 66–68, 66, 67
Data View builder, in Portlet Factory and, 245
database management system (DBMS), 195
database, sample applications, 173–174, 174
datstore, WebSphere Portal Datastore in, 317–320, 317
DB2 Universal Database, 8, 195
decoration area, 15
definitions, portlet, 57–58. See also customization
deploying a portlet, 85–86, 85, 86
deployment descriptor files, 22, 27–28, 31–32, 31, 32, 83, 84
customization and, 112
inter-portlet communication and, 320
language support/specification in, 119–121, 120
MIME types and, 52–53, 53
packaging portlets and, 83, 84
read-only data in, 111–112
rendering and, 52–53, 53
Deployment Server, Workplace Forms and, 381

NOTE: Boldface numbers indicate illustrations or code listing 495
Derby, custom database setup using, 483–487, 126
Designer screen, in Portlet Factory, 220–221, 220
Designer, Workplace Forms. See Workplace Forms; Workplace Forms Designer
directory structure of WAR file/portlet, 26, 27, 28
distributed code patterns, 96–103, 96
distributed computing principles, 192
DistributedMap cache, 197, 198–199, 198, 199
viewing, 203–206, 203–206
WAS setup of, 200–202, 200–202
DistributedObjectCache, 197
Document Object Model (DOM), 102, 103
Ajax and, 434, 435, 445, 454, 459, 459
Workplace Forms and, 415
Dojo toolkit, 460–461, 461
Domino, 190, 215
doView method, 203–204, 203–204
inter-portlet communication and, 338–339, 339
dynamic cache service, in WAS, 197
dynamic forms, 383
dynamic HTML (DHTML), 461
Ajax and, 440

E
Eclipse toolkit for Ajax, 462
Eclipse.org, 172, 173
Edit mode JSP (Calendar Portlet), 82–83, 82–83
Edit mode, 54–55
Edit_defaults mode, 55
encapsulation, 207
encodeURL, 106
encryption, 463
Enterprise JavaBeans (EJBs), 14, 14, 93
enterprise resource planning (ERP), Portlet Factory and, 216
Enterprise Service Bus (ESB), 208
enterprise services, 207, 208
templates, portlets, 58. See also customization
error handling, 21
Portlet Factory and, 221, 273
RAD and, 47–48, 48, 49
escape codes, XML, 69
Event Declaration builder, in Portlet Factory, 276, 277, 281
Event Handler builder, in Portlet Factory, 281, 282, 282
Event Handling builder, in Portlet Factory and, 276, 277, 278, 278
events, event handling, 466–467
evolution of portals, 185–186, 186
exceptions, 107
Excite, 183
execution stacks, Portlet Factory and, 214
expression language (EL), 68
Extensible Application Markup Language (XAMAL), 442
Extensible Forms Definition Language (XFDL), 379
Workplace Forms and, 385–386, 407
Extensible Markup Language. See XML
Extensible Stylesheet Language Transformation. See XSLT

F
factory portals, 184–185
feature-based modeling, 212
feeds, ATOM and RSS, Ajax and, 446
fetching data, Portlet Factory and builder code, 231–234, 232
fields, Workplace Forms and, 393, 393
file structure, in Portlet Factory, 220–221, 221
filters, portlet, 471–472
Firefox, 407
Flash Player, 441
Flex, 439, 441–442
flow logic and builder mapping, in Portlet Factory, 265
flow of data/information through portal, 10
flush method, 107
flyweight pattern, 56, 57, 59
FORM, JavaServer Faces (JSFs) and, 167–169
Form builder, in Portlet Factory, 245
Form Layout builder, in Portlet Factory, 265–266
formatting data, in Portlet Factory, 243, 244
forms using Portlet Factory, 259, 260–274, 268. See also Workplace Forms
Action List builder in, 268, 268
adding a new record in, 271, 272, 272–274, 273
Button builder in, 273, 274
Data Field Modifier (DFM) builder in, 266–267, 269, 269
Data Hierarchy Modifier builder in, 265–266, 266, 271
Data Page builder and inputs for, 260–274, 269, 270
deploying, 275
error handling in, 273
flow logic and builder mapping in, 265, 265
form submission entries in, 272–274, 273
form creation entries for, 262–264, 262
Form Layout builder in, 265–266
Imported Page builder in, 261, 261
Portlet Adapter builder in, 275
Rich Data Definition builder in, 266, 267, 270
Service Consumer builder in, 261
Submit form and database transaction in, 264–272
unique customer number generation for, 270–271, 271
validation in, 270
WebApp view in, 264
Fowler, Martin, 192
frameworks, 93–96, 104, 470
Ajax and, 460–462
G
Gadepalli, Venkata, 33
Garrett, Jesse James, 433–434
GenericPortlet class, 22
GET, 63, 83, 107
getCustomer, Portlet Factory and example, 252
getCustomerList, Portlet Factory and, 252
getCustomerList builder, Portlet Factory and, 231
getter and setter methods, JavaServer Faces (JSFs) and, 147–148, 148, 153
Global Settings administrative portlet, 119
Google, 183, 434, 441
granularity of services, SOA and, 180
graphics, 28
guard variables, in inter-portlet communication, 337
H
Hello World form and portlet example using Workplace Forms. See Workplace Forms
Hello World portlet, 25–50
Help mode, 54–55
Hibernate, 473
hidden frame technique vs. Ajax, 435
HTML, 16, 74, 133, 215
Ajax and, 440, 442, 446, 457, 459
Dojo toolkit for, 460–461, 461
dynamic HTML (DHTML) and, 440
JSF Widget Library (JWL) and, 138
tag libraries for, 109
Workplace Forms and, 400

NOTE: Boldface numbers indicate illustrations or code listing
HTTP and secure HTTP (sHTTP), 22, 350, 369–374, 463
GET, 63, 83, 107
POST, 63, 107, 417
Web services and, 98
HttpBasicAuth active credential example, 369–374, 371–372, 373
HttpSession, 12, 167
hypertext references (HREFs), in inter-portlet communication, 340

iBATIS, 473–474, 487
IBM Global Services, 178–179
IBM Patterns for eBusiness, 182
IBM Portlet API and, 18
IBM Portlet Services. See Portlet
IBM WebSphere Portal Primer, 33
IBM Workplace Forms. See Workplace Forms
IBM’s SOA Foundation, 179
IDs and passwords, 349, 354
IFrames, 109
ij tool, 485–487
Imported Page builder, in Portlet Factory, 261, 261
inheritance, customization and, 56, 57
init method, source portlet, inter-portlet communication and, 337–338, 338
instance variables, 104
integrated development environment (IDE), 25
Ajax and, 440
Portlet Factory and, 218, 220, 221
inter-portlet communication, 275–283, 275, 305–348
Action List builder in, 279, 279
action phase in, 310–312, 311
binding information in, 311, 329–333, 329, 330, 333
building portlets for, using RAD, 320–326
Button builder in, 278–280, 280, 281
buttons in, 278, 279, 280, 281
caption attributes in, 332
Cooperative Source builder in, 276
deployment descriptors for, 320
Event Declaration builder in, 276, 277, 281
Event Handler builder in, 281, 282, 282
Event Handling builder in, 276, 277, 278, 278
guard variables and, 337
hypertext references (HREFs) in, 340
Java Naming and Directory Interface (JNDI) lookups in, 337–338
JavaServer Pages (JSPs) in, 320, 341–347, 345–346, 347
JSR 168 and, 309
messaging and, 308
page aggregation in, 310
parameters, actions, and properties in, 311–312, 311, 330–333
Portlet Wiring Tool for, 313–315, 314
portType in, 324–325, 328, 328
properties in, input and output, 309–312, 311
render phase in, 310
ShowTemperaturePortlet example of, 306–308, 307, 321–326
source portlet code for, 336–337, 336–337
Java Database Connectivity (JDBC), 175, 474
Java Development Kit (JDK), 30
Java Naming and Directory Interface (JNDI)
cache access through, 198
inter-portlet communication and lookups in, 337–338
Portlet Factory and, 234
Java Portlet API, 171, 190
code patterns used in, 90–103
Java Portlet Spec v1.0, 8–9, 16, 51–87
Calendar Portlet application using. See Calendar Portlet
customization and, 51, 56–59
data models for portlets and, 59–62, 60
deploying portlets and, 85–86, 85, 86
interaction of portlets and containers in, 53
JavaServer Faces (JSFs) and, 138
life cycle of portlet and, 51, 52–54
modes and, 51, 54–55
packaging portlets and, 83, 84
persistent state and, 59, 60–61
portlet interface using, 53–54, 54
preferences and, 60–61
running a portlet and, 85–86, 85, 86
states and, 51, 55–56, 55, 59–60, 60
transient state and, 59, 61–62
URLs and, 63–64
Java source code for portlet, 29, 29
Java Standard Tag Library (JSTL), 65, 68–69, 68, 81, 109
internationalization and, 123, 125, 126
Java Virtual Machine (JVM), 247
caching’s effect on, 197
instance variables and, 104
JavaBeans, 93, 217
managed beans and, 141
scope of, 147
JavaScript
AJAX Toolkit Framework (ATF) for, 462
Dojo toolkit for, 460–461, 461
JSF Widget Library (JWL) in, 138
Kabuki toolkit for, 461–462
security considerations for, 463
JavaScript function, Ajax and, 457–460, 458–459
JavaScript Object Notation (JSON), 443, 459
JavaServer Faces (JSFs), 3, 15, 89, 93, 94, 135–169, 190–191, 470
action events in, 135, 152–156, 152–156
action phase and, 138, 164–165
bookmarking in, 162
developing a portlet using, 139–151
features of, 135
FORM use in, 167–169
getters and setters in, 153
integration with portlet development in, 162
Java Portlet Specification and, 138
JSF Widget Library (JWL) in, 138
Model View Controller (MVC) pattern and, 135
MyFaces open source (Apache) download for, 135
namespace and, 163–164, 164
navigation in, 135, 156–161, 156–161, 162
navigational state in, 167
pagination in, 151–152, 151, 152
phases of life cycle in, 137
portlet concepts in, 163–169
Portlet Factory vs., 190–191
portlet window handling in, 163–164
PortletPreferences and, 166, 166
prebuilt components in, 135
Rapid Application Development (RAD) and, 135, 162
reference implementation of, download for, 135
render phase and, 135, 138, 164–165
request/response life cycle in, 136–138, 136, 137
scope in, 163
singletons used in, 197
standards for, 162
states in, 135, 163, 167–169
Struts vs., 162
URLs and, 162, 166–167, 166, 167
user interface in, 162, 165
validation in, 135, 162, 165
JavaServer Pages (JSPs), 12, 13, 22, 27, 65, 93, 214, 215, 465
Ajax and, 453–456, 455–447
best practices for, 89
Calendar Portlet using, 79–81, 80
coding of, 108
inter-portlet communication and, 320, 341–347, 341–343, 345–346, 347
internationalization of, 121, 122, 125–130, 126–130
Java Standard Tag Library (JSTL) tags in, 65, 68–69, 68, 81
JavaServer Faces (JSFs) and, editing of, 148–150, 149, 150
MVC code pattern and, 92
Rapid Application Development (RAD) editing of, 148–150, 149, 150
resource bundles for, 122–123
scriptlets and, 65, 68–69, 68
session state and, 111
source portlet, 341–344, 341–343
target portlet, 344–347, 345–346, 347
Workplace Forms and, 396, 397, 400, 409, 416
JNDI lookup, JavaServer Faces (JSFs) and, 145, 146
JSF Widget Library (JWL), JavaServer Faces (JSFs) and, 138
JSR 186, 18, 465
Composite Capabilities/Preference Profiles (CC/PP), 134
inter-portlet communication and, 305, 309
vendor-specific portal extensions in, 107
Web Services for Remote Portlets (WSRP) and, 101
WebSphere Portal Core in, 316–320, 317
Workplace Forms and, 383
JSR 286, 171, 465–472
Ajax and, 446, 465
caching in, 470–471
client-capabilities access in, 470
coordination concepts in, 466–468
events, event handling in, 466–467
J2EE and, 465
Java Architecture for XML Binding (JAXB) and, 467
Java Community Process (JCP) and, 465
JavaServer Pages (JSPs) and, 465
JSR 186 and, 465
portlet filters in, 471–472
portlet windows in, 471
resource rendering in, 468–469
shared render parameters in, 467
shared session state in, 467–468
Web framework support and, 470
Web Services for Remote Portlets (WSRP) and, 465

K
Kernighan, Brian, 25

L
languages and markups, 117–134
bidirectional language support and, 133
Calendar Portlet example, 123–131
Composite Capabilities/Preference Profiles (CC/PP) and, 134

NOTE: Boldface numbers indicate illustrations or code listing
languages and markups, continued

deployment descriptors for, 119–121, 120
dynamic changes to, 131–132, 131
Global Settings administrative portlet
for, 119
internationalization and, 117, 118, 121
ISO 639 Codes for languages in, 119
Java Standard Tag Library (JSTL) and,
123, 125, 126
JavaServer Pages (JSPs) example of,
125–130, 126–130
language selection in, defining, 119
locale selection for, 117, 123–124, 124
localization and, 117, 118
multiple device support and, 133–134
Navigational State SPI and Portal URL
for dynamic changes to, 132, 132
portlet language support definition in,
119–121
properties files in, 125, 130–131
Rational Application Developer (RAD)
and, 123–131
resource bundles for, 120–125, 121, 124,
126, 129–130
specifying portlet support for, 123
WebSphere Portal support for, 118–119
layered model, 194
library creation, JSF and RAD, 143, 143
library setup for data, 475–480, 476–480
life cycle of portlet, 51, 52–54
Lightweight Third-Party Authentication
(LTPA), Ajax and, 462
line of business (LOB) systems/workflow, 406
local service layer with cache, 192–193, 193
local service layer, 192–193, 192, 193, 192
locales, 114, 117
Navigational State SPI and Portal URL
for dynamic changes to, 132, 132
selection of, 123–124, 124
localization, 117, 118. See also languages
and markup

Lock Box (Tivoli Access Manager for
eBusiness; TAM), 354
logical model of SOA, 179–181, 180
Lotus Domino, 215. See also Domino

M

macro portals, 183
Macromedia Flash vs. Ajax, 441
Managed Beans, 141–148, 142–148
singletons and, 197
markup language support for Ajax, 442
markup restrictions, portlet-specific, 108
markup selection, 21
mashups using Ajax, 441–442
maximized window state, 55
mediation, 208
micro portals, 183
Microsoft, 190, 434
migrating features to portlets/portals,
19–20, 20
MIME types, 52–53, 53
minimized window state, 55
Model View Controller (MVC), 21, 65, 89,
90–92, 91, 104, 470
JavaServer Faces (JSFs) and, 135
modeling
parametric feature-based, 212
Service Oriented Modeling and Analysis
(SOMA) and, 179
models, in Portlet Factory, 217, 218–219,
218, 219, 231–234, 246–248
mode of portlet, 15, 51, 54–55
About, 55
Config, 55, 58, 112–113, 112
containers and, 54
Edit, 54–55
Edit_defaults, 55
Help, 54–55
optional custom type, 55
Preview, 55

502
Print, 55
required, 54–55
vendor-specific, 55
View, 54–55, 66
Mozilla, 434
multiple device support, 133–134
mutators, Workplace Forms and, 415
MyFaces open source (Apache), 135

N
namespace, 108–109
Ajax and, 457
JavaServer Faces (JSFs) and, 163–164, 164
naming conventions, 108
nano portals, 184
navigation, JavaServer Faces (JSFs) and, 135, 156–161, 156–161
Struts, 162
Navigation and Model Services, 171
navigation handlers, 160–161, 161
navigational state, 61–62, 110–115
JavaServer Faces (JSFs) and, 167
Navigational State SPI and Portal URL for
dynamic language changes, 132, 132
.NET
Portlet Factory and, 214
Web services and, 98
Web Services for Remote Portlets (WSRP) and, 99–101
normal window state, 55
notarization, Workplace Forms and, 425–426, 426

O
OBJECT element, in Workplace Forms, 396–397, 396, 396, 419
Odyssey Browser Framework (OBF), 138
Olson, Bruce, 33
opaque identifiers, Credential Vault and,
359
Open Ajax, 439, 441, 461–462. See also Ajax
Open Ajax Alliance, 461
Oracle, 195

P
packaging a portlet, 83, 84, 114–115
grouping portlets for, 114–115
library patterns for, 114
page aggregation
Ajax and, 446
inter-portlet communication and, 310
Page builder, in Portlet Factory, 221–223,
222, 223, 224, 234–235
pages, portal, 5–9, 6, 11, 15–16, 15
pagination
JavaServer Faces (JSFs) and, 151–152,
151, 152
Portlet Factory and, 240, 241–243, 242, 243
span tags for, in Portal Factory, 242, 242
Paging Buttons builder, in Portlet Factory,
242–243, 243
parallel rendering, 106–107
parameters, actions, and properties in
inter-portlet communication, 311–312,
311, 330–333
parametric feature-based modeling, 212
Parametric Technology Corporation (PTC), 211
passive credentials, 364–367
passwords. See IDs and passwords
patterns, 89–116
Ajax in, 96, 96, 102–103, 103
Apache Struts Bridges Framework as, 94
Asynchronous JavaScript and HTML
(AJAH) in, 96, 96, 101–102, 101
asynchronous, 96–102, 96
choosing, criteria for, 95–96

NOTE: Boldface numbers indicate illustrations or code listing
patterns, continued
  combining vs. separating logic and markup in, 90–92, 90, 91
  customization and, 56
  distributed, 96–103, 96
  flyweight, 56, 57, 59
  frameworks, Web frameworks as, 93–96, 104, 470
  Java Portlet API, 90–103
  JavaServer Faces (JSF) and, 89, 93
  Model View Controller (MVC) as, 65, 89, 90–92, 91, 104, 135, 470
  PortalPatterns.org and, 209, 209
  Spring as, 94
  Struts and, 89, 93–94
  Web Services for Remote Portlets (WSRP) in, 96, 96, 99–101
  Web Services in, 96, 96, 97–101, 97
  WebWork as, 95
  PeopleSoft, 216
  PERL, Portlet Factory and, 214
  persistence layer, 195
  persistent channels or connections, in inter-portlet communication, 313
  persistent data, 110
  persistent state, 59, 60–61, 111–113
  action phase and, 113
  Config mode and, 112–113, 112
  read-only data in, 111–112
  String arrays and, 113
  personalization/customization. See customization
  perspectives, RAD, 39, 39
  Platform for Privacy Preferences (P3P), 106
  portals, 5–9, 6
  Portal Extension API, 20, 20
  PortalPatterns.org, 209, 209
  Portlet Adapter builder, in Portlet Factory, 221, 225–226, 226, 238, 275, 251–252
  Portlet Factory, 22, 175, 190–191, 211–257, 259–304
  Action List builder in, 221, 224–225, 224, 225, 237–238, 238, 268, 268, 279, 279
  Active Server Pages (ASPs) and, 214
  adding services in, 252–256
  advanced concepts in, 259–304
  background on and development of, 211–215
  Builder Editor for, 220, 220
  builders in, 213, 215–217, 231
  Button builder in, 273, 274, 278–279, 280, 280, 281
  buttons in, 241, 273
  change propagation through code using, 256
  changes to main (Action List builder) in, 250, 250
  Chart builder in, 245, 259
  Click-to-Action feature in, 276
  code generated by, 228, 228, 233–234
  code generator status of, 213–214
  commitment control in, 278–279
  computer-aided design (CAD) software as basis for, 211–215, 213
  configuration of, using Portal Unit Test Environment (UTE), 218, 221
  Cooperative Source builder in, 276
  Dashboard Framework builder in, 245, 259
  Data Column Modifier (DCM) builder in, 239–241, 240, 241, 245
  Data Field Modifier (DFM) builder in, 243, 244, 266–267, 269, 269
  Data Hierarchy Modifier builder in, 243, 244, 266–267, 269, 269
  Data Integration builders in, 245
  Data View builder in, 245
  deploying portlets from, 275
  Designer screen in, 220–221, 220
enterprise resource planning (ERP) and, 216
error messages in, 221, 273
Event Declaration builder in, 276, 277, 281
Event Handler builder in, 281, 282, 282
Event Handling builder in, 276, 277, 278, 278
execution stacks in, 214
file structure in, 220–221, 221
flow logic and builder mapping in, 265, 265
Form builder in, 245
Form Layout builder in, 265–266
formatting data in, 243, 244
forms using, 259, 260–274, 263, 268
Imported Page builder in, 261, 261
integrated development environment (IDE) of, 218, 220, 221
inter-portlet communication and, 275–283, 275
J2EE and, 214, 234
Java Naming and Directory Interface (JNDI) and, 234
JavaServer Faces (JSFs) vs. 190–191
model and builder interactions in a SOA using, 252, 252
model creation in, 231–234, 246–248
models in, 217, 218–219, 218, 219
modifying Factory-generated code in, 228–229
.NET and, 214
Page builder in, 221, 222–223, 222, 223, 224, 234–235
pagination in, 240, 241–243, 242, 243
Paging Buttons builder in, 242–243, 243
parametric feature-based modeling used in, 212
PERL and, 214
personalization and customization using, 259
Portlet Adapter builder in, 221, 225–226, 226, 238, 251–252, 275
profiling in, 259, 283–288
programming language selection for, 214
Rational Application Developer (RAD) and, 218
regeneration engine (regen) in, 212–214, 232–234, 233, 256
Rich Data Definition builder in, 266, 267, 267, 270
running a model in standalone mode in, 226–229, 227
running the model as a portlet in, 229–230, 230
SAP Framework builder in, 259
Service Call builders in, 245
Service Consumer builder in, 249, 249, 251, 261
Service Definition builder in, 246–247, 247
Service Operation builder in, 247–248, 248, 253, 254–255, 256
Service Oriented Architecture (SOA) and, 211, 245–252, 246
span tags in, 235, 235, 242, 242
SQL builder in, 245
SQL Call builder in, 231, 232–234, 232, 246, 253, 253
SQL Call example using, 213–214
SQL DataSource builder in, 253, 253
Submit form and database transaction in, 264–272
Summary and Chart builder in, 245
Summary and Drill Down builder in, 245
Target builder in, 276
template files in, changing or modifying, 263
testing operations in, 253, 273, 280
Two-Layer Data (SOA) portlet in, 252–254
validation in, 270
View builder in, 245

NOTE: Boldface numbers indicate illustrations or code listing
Portlet Factory, continued
View portlet creation using, 230–245
WAR file rebuild in, 229–230, 229, 238–239, 252, 275, 276
Web Services Description Language (WSDL) generation in, 247, 253, 276
WebApp elements added to builder in, 223, 224
WebApp elements in, 233, 233–234
WebApp view in, 220, 220, 228, 228, 264
writing a portlet using, 217–230
Portlet Messaging, 171
Portlet Services. See IBM Portlet Services
Portlet Wiring Tool, 313–315, 314
portlet.xml deployment descriptor, 28, 31–32, 31
See also deployment descriptors
PortletAdaptor class, 22
PortletPreferences, JavaServer Faces (JSFs) and, 166, 166
PortletRequest/Response, 166
portlets, 9, 10–15, 11–12
PortletURLs, 63
portType, 324–325, 328, 328
POST, 63, 107, 417
pre-populating Workplace Forms, 408–417.
See also Workplace Forms
preference data, internationalization and, 113–114
preferences, 60–61
persistent data and, 60–61
user-dependent, 61
user-independent (administrative), 60–61
validator for, 61
presentation layer, 194
SOA. See also user interface (UI), 180, 181
Preview mode, 55
Print mode, 55
privacy issues, 106
private slots, Credential Vault and, 357
process choreography, 208
process integration
Service Oriented Architecture (SOA) and, 187–189
Workplace Forms and, 430–431
process portals, 184
process-driven portals, 185–186, 186
processAction, 164, 204–205, 205
inter-portlet communication and, 340–341, 340–341
profiling, in Portlet Factory, 259, 283–288
properties, inter-portlet communication and, input and output, 309–312, 311
properties files, 125, 130–131
proprietary vs. standard APIs in creation of portlets/portals, 19, 19
protocol handling, 21
proxy servers, authentication, 350
publishing settings using RAD, 44, 44
publishing the portlet, 44, 45
Puma Services, JavaServer Faces (JSFs) and, 145, 146
PureEdge Solutions, 381–382. See also Workplace Forms
R
Rational Application Developer (RAD), 22, 33–47, 34, 13547, 34, 34, 418–482, 482
action events in, 152–156, 152–156
Actions and Preferences in, 38, 39
adding/removing projects in, 44, 45
Ajax and, 450–454
attribute retrieval in, 145, 145
Backing Beans and, 141
bean becoming a Managed Bean in, 146–148, 146–148
bean scope selection in, 147
Cancel button in, 158–159, 158, 159
class creation/selection for, 142–143, 142, 143
Console view in, 46, 46
creating portlet project in, 139–140, 139–141
error management in, 47–48, 48, 49
getter and setter methods assignment in, 147–148, 147–148, 148, 153
inter-portlet communication and, building portlets for, using, 320–326
internationalization and localization using, 123–131, 161, 161
JavaServer Faces (JSFs) and, 162
JavaServer Pages (JSPs) editing in, 148–150, 149, 150
JNDI lookup in, 145, 146
library creation for, 143, 143
managed beans creation in, 141–148, 142–148
method creation and assignment to beans in, 144, 144
naming the project in, 36
navigation handlers in, 160–161, 161
navigation in, 156–161, 156–161
new project in, 34–35, 35, 36
pagination in, 151–152, 151, 152
parameter selection in, 36–37, 37
perspectives in, 39, 39
Portlet Factory and, 218
publishing settings using, 44, 44
publishing the portlet using, 44, 45
Puma Services and, 145, 146
running the portlet in, 40–47, 40, 150, 150, 151
Security Alert in, 46, 46
server creation for, 41, 41
Servers view in, 44, 45
settings in, 38, 38
Struts, 162
Submit button in, 158–159, 158, 159
testing in, 34, 46
validation in, 161
viewing the portlet in, 46, 47
WebSphere Portal and, 34
WebSphere settings in, 42–43, 42, 43
wizard for, 36
Workplace Forms and, 398
read-only data, 111–112
regeneration engine (regen), in Portlet Factory, 212–214, 232–234, 235, 256
render calls, 52–53, 53
render parameters, 65, 69–71, 70–71
render phase
inter-portlet communication and, 310
JavaServer Faces (JSFs) and, 138, 164–165
rendering, 52–53, 53
JavaServer Faces (JSFs) and, 135
parallel, 106–107
render parameters and, 65, 69–71, 70–71
renderURLs in, 63, 80, 166–167
resource, 468–469
shared render parameters in, 467
RenderRequest/Response, 164, 165
renderURLs, 63, 69–71, 80, 166–167
repositories, 208
Representational State Transfer (REST) services, Ajax and, 446
request/response, JavaServer Faces (JSFs) and, 136–138, 136, 137
request-response paradigm, 17
requestRowAction, 167
resource bundles, 113, 120–121, 121, 124–125, 124, 126, 129–130
JavaServer Faces (JSFs) and, 161, 161
resource name of slots, Credential Vault and, 357, 359
resource rendering, 468–469
resource URLs, 64
reuse of code, 180, 188–189, 207, 208
Rich Data Definition builder, in Portlet Factory, 266, 267, 267, 270

NOTE: Boldface numbers indicate illustrations or code listing
Index

Ritchie, Dennis, 25
roadmap for SOA use in portals, 206–208, 207
Roberts, Andy, 211
RSS feeds, Ajax and, 446
running a portlet, 85–86, 85, 86, 150, 150, 151
Portlet Factory and, 226–230, 227, 230
on RAD, 40–47, 40, 40

S
Safari, 434
sample portlets, databases, 172. See also
book scenarios
data component setup for, 473–487. See also
data component setup for sample apps
website code source for, 2
SAP, 215, 216, 217
SAP Framework builder, in Portlet Factory, 259
schema, sample applications, 173–174, 174
scope, 62
JavaBean, selecting, 147
JavaServer Faces (JSFs) and, 163
scriplets, Java Standard Tag Library (JSTL) tags and, 65, 68–69, 68, 81
SCRIPT element, in Workplace Forms, 397–398, 397, 397
scrolling or paging through displayed data,
JavaServer Faces (JSFs) and, 151–152, 151, 152
secrets, Credential Vault and, 352–353, 354, 361
secure HTTP (sHTTPS), 350
secure sockets connections, 463. See also
HTTP and secure HTTP (sHTTPS)
security, 171
Ajax and, 440, 448, 462–463
authentication and, 349
authentication proxy servers in, 350
authorization and, 349
Content Access Services, 171
Credential Vault and, 171, 349. See
Credential Vault, 349
cross-site scripting and, 109–110
data Access Service example and, portlet access and, 195
cross-site scripting and, 109–110
data Access Service example and, portlet access and, 195
encryption and Ajax, 463
HTTP and secure HTTP (sHTTPS) in, 350, 369–374, 463
IDs and passwords in, 349
JavaScript and, 463
Lock Box (Tivoli Access Manager for eBusiness; TAM), 354
Navigation and Model Services, 171
portlet-based approaches to, 352–353
single sign on and, 349–351, 351
Uniform Resource Identifiers (URIs)
and, 350
WebSphere Portal and, 349, 352
Security Alert in RAD, 46, 46
segments, in Credential Vault, 354, 355–356, 359, 360
server creation in RAD, 41, 41
Server, Workplace Forms. See Workplace Forms; Workplace Forms Server
Servers view in RAD, 44, 45
Service Call builders, in Portlet Factory, 245
Service Consumer builder, in Portlet Factory, 249, 249, 251, 261
Service Definition builder, in Portlet Factory, 246–247, 247
service layers, 191–195
local service layer with cache and, 192–193, 193
local, 192–193, 192, 193, 192
persistance layer and, 195
presentation layer, 194
separation of concerns within, 194–195, 194
Service Operation builder, in Portlet Factory, 247–248, 248, 253, 254–255, 256
Service Oriented Architecture (SOA), 177–209. See also Data Access Service example
back-end considerations and, 191
composite applications and, 181, 187–189, 188
Data Access Service example using, 195–206
definition of, 178–181
distributed computing principles and, 192
evolution of portals and, 185–186, 186
goals of, 178–181
granularity of services in, 180
IBM Global Services and, 178–179
Interaction Services segment in, 180
JavaServer Faces(JSFs) and, 190–191
local service layer and, 192–193, 192, 193, 192
local service layer with cache and, 192–193, 193
logical view of, 179–181, 180
persistence layer and, 195
portals as on-rampquote to, 181–186
Portlet Factory and, 190–191, 211, 245–252, 246, 252, 252
presentation layer and, 180, 181
process integration in, 187–189
reuse issues and, 180, 188–189
roadmap for, in use with portals, 206–208, 207
service layers and, 191–195
Service Oriented Modeling and Analysis (SOMA) and, 179
services in, 178
services within portals and, considerations for, 194–195, 194
SOA portals in, 184
technology considerations and, 189–191
types of portals and use of, 181–186, 182, 186
WebSphere Portal and, 177–178, 189, 208–209
Workplace Forms and, 380
Service Oriented Modeling and Analysis (SOMA), 179
service repositories, 208
Service Stub builder, in Portlet Factory, 253
Service Test builder, in Portlet Factory, 253
services, 171
   granularity of, 180
   Service Oriented Architecture (SOA) and, 178
ServletResponse, 165
servlets, 13
   Ajax and, 447
   naming, 108
   portlets vs., 11–12, 18
session state, 61–62, 111
   shared, 467–468
   shared render parameters, 467
   shared session state, 467–468
   shared slots, Credential Vault and, 356
sharing data. See data sharing
ShowTemperaturePortlet example, 306–308, 307, 321–326
Siebel, 215
signatures, Workplace Forms and, 383
Simple API for XML (SAX), Workplace Forms and, 415
Simple Object Access Protocol (SOAP), 42, 98
single sign on, 349–351, 351
   singleton
data access using, 474–475
   for Data Access Service, 195–197, 196
   skills and training, 190–191

NOTE: Boldface numbers indicate illustrations or code listing
slotID, Credential Vault and, 359, 359, 361–363
slots, Credential Vault and, 354, 356–364, 360, 361
SOA portals, 184
source code, website downloads for, 2
source portlet code, for inter-portlet communication, 336–337, 336–337
source portlet doView method, for inter-portlet communication, 338–339, 338
source portlet init method, for inter-portlet communication, 337–338, 338
source portlet JSP, for inter-portlet communication, 341–344, 341–343
source portlet processAction method, for inter-portlet communication, 340–341, 340–341
source portlet WSDL, for inter-portlet communication and, 326–337, 327–330, 327
span tags, Portlet Factory and, 235, 235, 242, 242
spawning threads from portlets, 106
specification for portals (Java Portlet Spec v1.0), 8–9, 16. See also Java Portlet Spec v1.0,
Spring, 94, 470, 474
SQL, 123, 175
Call, using Portlet Factory, 213–214
SQL builder, in Portlet Factory, 245
SQL Call builder, in Portlet Factory, 231, 232–234, 232, 246, 253, 253
SQL DataSource builder, in Portlet Factory, 253, 253
SQL Maps, 474
SQL operations and their corresponding services, Portlet Factory and, 253, 254–255
standards
JavaServer Faces (JSFs) and, 162
Struts, 162
state change, 21
states, 51, 55–56, 59–60, 60
Ajax and, in client/server, 448
Ajax and, in portlets, 447
JavaServer Faces (JSFs) and, 163, 167–169
JavaServer Faces (JSFs) and,
preservation of, 135
maximized/minimized, 55
navigational, 61–62, 110–115, 167
normal, 55
persistent, 59, 60–61, 111–113
scope and, 62
session, 61–62, 111
shared session, 467–468
transient, 59, 61–62
String arrays, persistent data and, 113
Strings, Workplace Forms and, 381
Struts, 89, 93–94, 190, 470
bookmarks, in, 162
integration with portlet development in, 162
JavaServer Faces (JSFs) and vs., 162
navigation in, 162
Rapid Application Development (RAD) and, 162
standards for, 162
URLs and, 162
user interface in, 162
validation in, 162
WebSphere Portal support for, 93–94
Submit button, 158–159, 158, 159
Workplace Forms and, 393–395, 394, 422–426, 422
Submit form and database transaction, Portlet Factory and, 264–272
Summary and Chart builder, in Portlet Factory, 245
Summary and Drill Down builder, in Portlet Factory, 245
Supply Chain Management (SCM), 182
system slots, in Credential Vault, 356, 374–377
SystemOut.Log, 202

tag libraries, 109
JSF Widget Library (JWL) and, 138
TAM Lock Box, 354
target and source portlets, in inter-portlet communication, 314–315, 320–326
Target builder, in Portlet Factory, 276
target portlet JSP, for inter-portlet communication, 344–347, 345–346, 347
target portlet WSDL, for inter-portlet communication, 334–335, 334–335
(testing the portlet, 34, 46
text editor to create portlet, 25
threads, spawned from portlets, 106
Tivoli Access Manager for eBusiness (TAM), 354
Tivoli Directory Server, 8
ToDo portlet code (Calendar Portlet), 75–77, 75–77, 79, 79
Tomcat, 214
tools for portlets/portals, 18, 22
training, 190–191
transaction portals, 185, 186
transient data, 110
transient state, 59, 61–62
Two-Layer Data (SOA) portlet using Portlet Factory, 245–252
addCustomer in, 252
adding services in, 252–256
changes to main (Action List builder) in, 250, 250
CustomerData model in, 252, 256, 256
CustomerListView model in, 249
data model creation in, 246–248
Data Page builder in, 251, 251
getCustomer in, 252
getCustomerList in, 252
Java Virtual Machine (JVM) and, 247
Portlet Adapter builder in, 251–252
Service Consumer builder in, 249, 249, 251
Service Definition builder in, 246–247, 247
Service Operation builder in, 247–248, 248, 253, 254–255, 256
Service Stub builder in, 253
Service Test builder in, 253
SQL Call builder in, 246, 253, 253
SQL DataSource builder in, 253, 253
SQL operations and their corresponding services in, 253, 254–255
testing operations in, 253
updateCustomer in, 252
View model for, 249–252
Web Services Description Language (WSDL) generation in, 247, 253
types of portals, 7–8, 181–186, 182, 186
aggregate, 186
business-to-business (B2B), 182
business-to-consumer (B2C), 182
business-to-employee (B2E), 182–183
collaborative, 186, 185
contextual, 184
factory, 184–185
Internet, 183
macro, 183
micro, 183
nano, 184
process, 184
process-driven, 185–186, 186
SOA, 184
transaction, 186
Uniform Resource Identifiers (URIs), 350
namespaces and, 108–109
Workplace Forms and, form submission, 418, 418
uniform resource locators. See URLs

Unit Test Environment (UTE), in Portlet Factory, 218, 221
Universal Description, Discovery and Integration (UDDI) directory, 97
updateCustomer, Portlet Factory and, example, 252
URLs, 63–64, 106
  ActionURLs as, 63
  Ajax and, 434–435, 447, 449
  HTTP POST and GET in, 63, 83
  JavaServer Faces (JSFs) and, 162, 166–167, 166, 167
  navigational state and, 110–115
  PortletURLs as, 63
  RenderURLs as, 63
  resource type, 64
  Struts, 162
  Workplace Forms and, form submission, 418, 418
use cases, 173
User Agent Profile (UAProf), 470
user interface, 17
  Ajax and, 438–439, 447
  JavaServer Faces (JSFs) and, 162, 165
  Service Oriented Architecture (SOA) and, 180, 181
  Struts, 162
  Workplace Forms and, 388–390, 389, 390
user-dependent preferences, 61
user-independent preferences, 60–61

V
validation
  JavaServer Faces (JSFs) and, 135, 161, 162, 165
  Portlet Factory and, 270
  Struts, 162
  Workplace Forms and, 425–426, 426
validator, preferences, 61
variables, guard, 337
vault adapters, Credential Vault and, 354
VaultSlotName parameter, 357, 357–358
Veloticy, 93
vendor-specific modes, 55
vendor-specific portal extensions, 107
versioning information, 104–108
view beans, 27
View builder, in Portlet Factory, 245
View mode, 54–55, 66
View mode JSP (Calendar Portlet), 81–83, 81, 82–83
View portlet created with Portlet Factory, 230–245
Action List builder in, 237–238, 238
buttons in, 241
columns in, 239–241, 240, 241
CustomerList portlet creation in, 238–239, 239
Data Column Modifier (DCM) builder in, 239–241, 240, 241
Data Field Modifier (DFM) builder in, 243, 244
Data Page builder in, 234–237, 236, 237
Factory-generated code in, 233–234
fetching data code in, 231–234, 232
formatting data in, 243, 244
getCustomerList builder in, 231
model creation for, 231–234
Page builder in, 234–235
pagination in, 240, 241–243, 242, 243
Paging Buttons builder in, 242–243, 243
Portlet Adapter builder in, 238
regeneration engine at work in, 232–234, 233
span tags in, 235, 235, 235, 242, 242
WAR file rebuild for, 238–239
WebApp elements in, 233, 233–234
viewDate JSP code (Calendar Portlet), 79, 80
Viewer, Workplace Forms. See Workplace Forms; Workplace Forms Viewer viewing the portlet in RAD, 46, 47 views, application view in, 85, 85
virtualization, 208

W
WAR files, 12, 26, 32–33, 33, 106
inter-portlet communication and, 318–320
Portlet Factory and, rebuilding of, 229–230, 229, 238–239, 252, 275, 276
Web applications
cross-site scripting and, 109–110
error management in, 21
markup selection, 21
naming conventions in, portlet vs. servlet, 108
portlets/portals vs., 17, 21
protocol handling in, 21
scope in, 62
state change in, 21
Web frameworks, 93–96, 93, 104, 470
Web Service Description Language (WSDL) files, in inter-portlet communication, 312, 317, 318, 320, 321–337, 322, 323–324
Web services, 96, 96, 97–101, 97
data-oriented vs. presentation-oriented, 99–100, 100
HTTP and, 98
Java and, 98
.NET and, 98
portlet use of, 98
Simple Object Access Protocol (SOAP) and, 98
Universal Description, Discovery and Integration (UDDI) directory for, 97
Web Services Description Language (WSDL) in, 97–98
Web Services for Remote Portlets (WSRP), 99–101
Web Services and, support in, 408
Web Services Description Language (WSDL), 97–98
Portlet Factory and, 247, 253, 276
Workplace Forms and, 406
web.xml deployment descriptor, 27, 31–32, 31, 32
WebApp elements, Portlet Factory and, 223, 224, 233, 233–234
WebApp view, in Portlet Factory and, 220, 220, 228, 228, 264
Webform Server, 380–381, 401–403, 402, 403
WebSphere, 214
WebSphere Application Server (WAS), 8, 30
admin console of, 200, 200
Ajax and, 462
as data source, 487
caching and, 197, 200
DistributedMap cache in, 197, 200–202, 200–202
Workplace Forms and, 381, 407
WebSphere Portal, 1, 2, 3, 7–8, 8, 22, 171, 191
Ajax and, 444–445
click-to-action (C2A) in, 305
inter-portlet communication and, 305, 309
language support in, 118–119
multiple device support and, 133–134
RAD and, 34, 42–43, 42, 43
security and, 349. See also Credential Vault
Service Oriented Architecture (SOA) and, 177, 189, 208–209. See also Service Oriented Architecture (SOA)
Struts support in, 93–94
WebSphere Portal Core, 316–320, 317
WebSphere Portal Datastore, 317–320, 317
WebSphere Portal Framework, 171
WebSphere Portal, 7–8, 8, 7
WebWork, 95, 470

NOTE: Boldface numbers indicate illustrations or code listing 513
window states. See states
window, portlet, 15, 15, 58–59, 163–164, 471. See also customization
Wireless Markup Language (WML), 16, 133–134, 385
wiring, in inter-portlet communication, 305, 308, 310, 312–315, 314
Workplace Dashboard Framework, 217
Workplace Forms, 22, 379–432
accessors in, 415
ACORD 111 XML messages in, 415
API code library for, 427–428, 427, 428
architectures for, 405–408, 406, 407
background and development of, 381–382
benefits of, 379–380
browser use with, 407
creating a form in, 386–388, 387, 388
data extraction from forms in, 427–428, 427, 428
data integration from forms in, 428–429
data used in, 408–409
displaying a form in a portlet from, 395–403, 395
document model in, 384–386, 384
Document Object Model (DOM) and, 415
documentation (online) for, 395
dynamic forms in, 383
dynamic forms in, 383
enterprise architecture and, 383
extending portlet framework for, to display, 401–403, 402, 403
extensibility of, 383
Extensible Forms Definition Language (XFDL), 385–386, 407
fields inserted in, 393, 393
file size constraints in, 383
Hello World form and portlet example, 386–398
HTML and, 400
integration of applications within, 405–408, 406, 407
inter-portlet communication and, 420–430
Internet Form Extension (IFX) in, 410
JavaServer Pages (JSPs) in, 396–398, 396, 396t, 397, 400, 409, 416
JSR 168 and, 383
key features and uses of, 383
mutators in, 415
OBJECT element in, 396–397, 396, 396t, 419
offline or disconnected operation and, 408
pre-populating forms in, 408–417
process integration models for, 430–431
processing forms in, 425–429, 426–428
RAD preview of, 398–399, 399
Rational Application Developer (RAD) and, 398
SCRIPT element in, 397–398, 397, 409
server-side processing performance and, 383
Service Oriented Architecture (SOA) and, 380
signature validation/notarization in, 425–426, 426
signatures in, 383
Simple API for XML (SAX) and, 415
string parsing data extraction in, 428
Strings and, 381
submitting forms in, within context of portal application, 417–425
template storage in, 400–401
terminology of, 382, 382

514
Wireless Markup Language (WML) and, 385
Workplace Forms Designer in, 380, 405–408, 406, 407, 420–425
Workplace Forms Portlet Framework in, 407
Workplace Forms Server in, 380–381, 405–408, 406, 407
Workplace Forms Viewer, 380, 405–408, 406, 407
World Wide Web Consortium (W3C), 379

X
XForms, 379, 381, 382, 385–386, 390–393, 391, 392, 393, 407. See also Workplace Forms
Workplace Forms and, 420–421, 421, 429
XFormsXML, 22, 214, 215, 216
Ajax and, 440, 442, 445, 451–452, 451, 452, 459
escaped output from, 69
Workplace Forms and, 381, 382, 401, 406, 414–415, 414, 415, 429
XML User Interface Language (XUL), 442
XSLT, 22, 93
XMLHttpRequest, 102, 103
Ajax and, 434, 436, 436, 444, 458–460, 458, 469
XPath references, Workplace Forms and, 413–414, 413, 414

Y
Yahoo, 183

Z
Zimbra Inc., 461

NOTE: Boldface numbers indicate illustrations or code listing