Index

A

<a4j:include> component, 220
<a4j:log> component, 220
<a4j:mediaOutput> component, 220
<a4j:outputPanel> component, 213–214
<a4j:poll> component, 220
<a4j:region> container tag, 220
<a4j:status> component, 220
access control
   declarative access control, 280–283
   rule-based access control, 283–288
accessing
   private fields, 296
   Seam components
      from JavaScript, 234–236
      with JSF EL, 304–307
      with Seam API calls, 307–309
actors (jBPM), assigning to web users, 245–249
ADF Faces, 208–209
AJAX, 7, 197–198
Ajax4jsf, 211–223
   AJAX-enabled buttons, 217–219
   benefits/limitations, 223
   components in, 220
   configuring, 221–223
   container tags, 219–220
   programmatic AJAX, 214–217
   validation example, 212–214
Dojo toolkit integration, 236–242
ICEfaces, 199
   auto-complete text input example, 202–205
   bundling JAR files for Seam, 205–208
   page parameters, 208
   partial form submission example, 199–202
Seam remoting JavaScript library
   AJAX progress bar example, 232–236
   validation example, 225–231
AJAX calls, Seam remoting JavaScript library example, 229–231
AJAX-enabled buttons, 217–219
AJAX progress bar example, 232–236
Ajax4jsf, 211–223
   AJAX-enabled buttons, 217–219
   benefits/limitations, 223
   components in, 220
   configuring, 221–223
   container tags, 219–220
   programmatic AJAX, 214–217
   validation example, 212–214
alternative data sources in Tomcat environments, 343–344
alternative display output tags, 41
annotations, 11
   @ApplicationException annotation, 130
   @Begin annotation, 108–109
   @BeginTask annotation, 257–258
Index

@Conversational annotation, 111
@Create annotation, 93, 165–166
@DataModel annotation, 109, 150–151, 154–155
@DataModelSelection annotation, 152, 154–155
@Destroy annotation, 94
@End annotation, 111–114
@EndTask annotation, 258
exception annotations, 186–187
@Factory annotation, 94–95, 164–165
@IfInvalid annotation, 144
@In annotation, 16, 160
@Out annotation, 16, 95, 160
performance and, 351–352
@Remove annotation, 94
@RequestParameter annotation, 163
@Restrict annotation, 283–285
@Rollback annotation, 131–132
@Scope annotation, 91
@Stateful annotation, 91
@Test annotation, 294
@Transactional annotation, 129
@TransactionAttribute annotation, 129, 134
@Valid annotation, 144
validation annotations, 139–141
@WebRemote annotation, 227
XML files replaced by, 9

application context, 84
Application Development Framework
Faces. See ADF Faces
@ApplicationException annotation, 130
applications
CRUD applications, generating in Seam Gen, 71–74
deploying, 369–370
EJB3 applications, as templates, 371–377
Hello World sample application, 11–17
configuration files, 23–30
data models, creating, 13–14
data models, mapping to web forms, 14–15
Facelets, 34–35
packaging, 23–30
testing, 20
web events, handling, 15–17
Hotel Booking example application
conversations across workspaces, 123–124
customizing conversation IDs, 124–126
defining data sources, 347–348
installing JDBC drivers, 347
installing production databases, 345–347
long running conversations, 103–116
persistence engine configuration, 348–349
rolling back transactions, 130–132
workspace switcher, 120–123
workspaces, 117–120

Ant, 370
ANTLR (ANother Tool for Language Recognition), 54
Apache MyFaces Tomahawk project, 208
Apache Tomcat Connector, 362
API calls, accessing Seam components, 307–309

388
Seam applications, deploying, 369–370
Seam POJO applications, as templates, 378–383
stateful applications, 5–6
stateful example application, 87–98
  page navigation in, 95–98
  stateful components in, 88–95
Ticketing example application
  assigning jBPM actors to web users, 245–249
  binding data objects, 255–256
  creating business process instances, 254–255
  defining business processes, 252–254
  task management, 257–263
Tomcat applications, as templates, 383–384
Web 2.0 applications, 7
associating business processes with web pages, 270–274
Asynchronous JavaScript and XML. See AJAX
atomic conversations, 132–134
authentication, 277–280
auto-complete text input example, 202–205

B
Back button. See browser navigation support
@Begin annotation, 108–109
@BeginTask annotation, 257–258
bijection. See dependency bijection
binding data objects in business process scope, 255–256
boilerplate code, reduction of, 84–86
bookmarkable web pages, 157–167
  Java solution to, 162–167
  page parameters, 158–162
  when to use, 158
bookmarks, 116
bootstrapping JBoss MicroContainer, 339, 342
browser navigation support
  in state management, 81–82
    Hotel Booking example application, 107
  stateful navigation rules and, 274
built-in components, 19–20
bundling
  ICEfaces JAR files, 205–208
  JAR files for Tomcat, 335–336, 340–341
business process context, 84
  binding data objects, 255–256
business process management tags, 40
business processes
  associating with web pages, 270–274
  jBPM, 245–249
    actors, assigning to web users, 249–251
    binding data objects, 255–256
    configuring, 263–265, 275–276
    creating business process instances, 254–255
    defining business processes, 252–254
    task management, 257–263
Index

long running conversations versus, 261
page navigation and stateful conversations, 274–275
business rules, security framework, 277–288
  declarative access control, 280–283
  rule-based access control, 283–288
  user authentication/roles, 277–280
buttons, AJAX-enabled, 217–219

C
  caches. See database caches
  Call by Value, avoiding, 352–353
  calls. See AJAX calls
  chatty web applications, 80
  checked exceptions, rolling back transactions, 130–131
  clickable data tables, 149–155
    data-binding framework, 154–155
    displaying, 150–152
    event handlers, injecting selected objects into, 152–153
    Seam JSF EL in, 153–154
  client-side state saving, server-side state saving versus, 356–357
  clustering, 361–364
  concurrent conversations, workspace switcher, 120–123
  configuration by exception, 8
  configuration files
    for advanced application templates, 385
    in clustering, 363
    Hello World sample application, 23–30
    JBoss Embeddable EJB3, 341–342
    JBoss MicroContainer, 337–339
  configuring
    Ajax4jsf, 221–223
    JBoss Rules, 287–288
    jBPM, 263–265, 275–276
    persistence engine, 348–349
  connections, detecting, 201–202
  container tags, AJAX, 219–220
  contexts, 6
    default conversation scope, 100–103
    stateful contexts, 83–84
  conversation context, 84
  conversation IDs, 123–124
    customizing, 124–126
  conversation management tags, 40
  @Conversational annotation, 111
  conversations, 99–100
    across workspaces, 123–124
    atomic conversations, 132–134
    concurrent conversations, workspace switcher, 120–123
    default conversation scope, 100–103
    ending, 111–114
    HTTP GET requests and, 101
    interrupted conversations, resuming, 120
    JSF messages, displaying, 102–103
    links in, 114–116
    long running conversations, 80, 103–116
      business processes versus, 261
      stateful navigation rules and, 274–275
    POJO components and, 101
    redirection and, 101
    starting, 108–109
    workspaces and, 119–120
<table>
<thead>
<tr>
<th><strong>Index</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>@Create annotation</strong>, 93, 165–166</td>
</tr>
<tr>
<td><strong>Create, Retrieve, Update, and Delete applications. See CRUD applications</strong></td>
</tr>
<tr>
<td><strong>Crud application framework, 169–182</strong></td>
</tr>
<tr>
<td><strong>DAOs, 169–170</strong></td>
</tr>
<tr>
<td><strong>declarative Seam DAO components, 171–176</strong></td>
</tr>
<tr>
<td><strong>POJOs versus session beans, 170–171</strong></td>
</tr>
<tr>
<td><strong>queries, 176–182</strong></td>
</tr>
<tr>
<td><strong>CRUD applications</strong></td>
</tr>
<tr>
<td>generating in Seam Gen, 71–74</td>
</tr>
<tr>
<td>Seam CRUD application framework, 169–182</td>
</tr>
<tr>
<td><strong>DAOs, 169–170</strong></td>
</tr>
<tr>
<td><strong>declarative Seam DAO components, 171–176</strong></td>
</tr>
<tr>
<td><strong>POJOs versus session beans, 170–171</strong></td>
</tr>
<tr>
<td><strong>queries, 176–182</strong></td>
</tr>
<tr>
<td><strong>CSS, 38</strong></td>
</tr>
<tr>
<td><strong>customizing conversation IDs, 124–126</strong></td>
</tr>
<tr>
<td><strong>D</strong></td>
</tr>
<tr>
<td><strong>DAOs, 169–170</strong></td>
</tr>
<tr>
<td><strong>declarative Seam DAO components, 171–176</strong></td>
</tr>
<tr>
<td><strong>Data Access Objects. See DAOs</strong></td>
</tr>
<tr>
<td><strong>data-binding framework, 154–155</strong></td>
</tr>
<tr>
<td><strong>data list components in Facelets, 39–40</strong></td>
</tr>
<tr>
<td><strong>data models</strong></td>
</tr>
<tr>
<td>creating, Hello World sample application, 13–14</td>
</tr>
<tr>
<td>mapping to web forms, Hello World sample application, 14–15</td>
</tr>
<tr>
<td><strong>data objects, binding in business process scope, 255–256</strong></td>
</tr>
<tr>
<td><strong>data sources. See also production databases</strong></td>
</tr>
<tr>
<td>alternative data sources in Tomcat environments, 343–344</td>
</tr>
<tr>
<td>defining, 347–348</td>
</tr>
<tr>
<td><strong>data tables. See clickable data tables</strong></td>
</tr>
<tr>
<td><strong>data validation. See validation</strong></td>
</tr>
<tr>
<td><strong>database access with EntityManager, 23</strong></td>
</tr>
<tr>
<td><strong>database caches</strong></td>
</tr>
<tr>
<td>application state and, 6</td>
</tr>
<tr>
<td>performance tuning, 358–360</td>
</tr>
<tr>
<td><strong>database services, unit testing, 297–299</strong></td>
</tr>
<tr>
<td><strong>database updates, manually flushing, 132–133</strong></td>
</tr>
<tr>
<td><strong>databases, generating CRUD applications from, 71–74</strong></td>
</tr>
<tr>
<td><strong>@DataModel annotation, 109, 150–151, 154–155</strong></td>
</tr>
<tr>
<td><strong>@DataModelSelection annotation, 152, 154–155</strong></td>
</tr>
<tr>
<td><strong>debug information page, 190–193</strong></td>
</tr>
<tr>
<td><strong>debug page, 192–193</strong></td>
</tr>
<tr>
<td><strong>debugging in NetBeans IDE, 67–69</strong></td>
</tr>
<tr>
<td><strong>debugging mode, JBoss AS in, 68–69</strong></td>
</tr>
<tr>
<td><strong>declarative access control, 280–283</strong></td>
</tr>
<tr>
<td><strong>declarative Seam DAO components, 171–176</strong></td>
</tr>
<tr>
<td><strong>declarative state management. See state management</strong></td>
</tr>
<tr>
<td><strong>decorators, 145–147</strong></td>
</tr>
<tr>
<td><strong>decoupling Seam components, 92–93</strong></td>
</tr>
<tr>
<td><strong>default conversation scope, 100–103</strong></td>
</tr>
<tr>
<td><strong>dependency bijection, 7–8, 16</strong></td>
</tr>
<tr>
<td>avoiding, 21–22</td>
</tr>
<tr>
<td>in bookmarkable web pages, 160</td>
</tr>
<tr>
<td>unit testing, 295–296</td>
</tr>
<tr>
<td>via getter/setter methods, 20–21</td>
</tr>
</tbody>
</table>
Index

<table>
<thead>
<tr>
<th>deploying</th>
<th>EJB session beans, 11</th>
</tr>
</thead>
<tbody>
<tr>
<td>in Java EE 5.0 environments, 313–317</td>
<td>EL. See Seam JSF EL</td>
</tr>
<tr>
<td>Seam applications, 369–370</td>
<td>@Email annotation, 141</td>
</tr>
<tr>
<td>Seam Gen applications, 63–64</td>
<td>email messages, template-based, 50–53</td>
</tr>
<tr>
<td>Seam POJO applications, 319–331</td>
<td>email support tags, 50–53</td>
</tr>
<tr>
<td>Hibernate API example, 327–331</td>
<td>Embeddable EJB3, 339–344</td>
</tr>
<tr>
<td>JPA example, 320–327</td>
<td>enabling Facelets debug page, 190–191</td>
</tr>
<tr>
<td>in Tomcat environments, 333–344</td>
<td>@End annotation, 111–114</td>
</tr>
<tr>
<td>EJB3 applications, 339–344</td>
<td>ending conversations, 111–114</td>
</tr>
<tr>
<td>Seam POJO applications, 335–339</td>
<td>@EndTask annotation, 258</td>
</tr>
<tr>
<td>design patterns, 11</td>
<td>enhancements. See JSF enhancements</td>
</tr>
<tr>
<td>@Destroy annotation, 94</td>
<td>Enterprise Application aRchive. See EAR files</td>
</tr>
<tr>
<td>detecting server connections, 201–202</td>
<td>Enterprise JavaBeans. See EJB 3.0</td>
</tr>
<tr>
<td>development tools. See Seam Gen</td>
<td>entity beans</td>
</tr>
<tr>
<td>directory structure</td>
<td>creating JavaScript objects for, 230</td>
</tr>
<tr>
<td>for example projects, 13</td>
<td>stateful entity beans, 91</td>
</tr>
<tr>
<td>in Seam Gen, 60–61</td>
<td>validation annotations, 139–141</td>
</tr>
<tr>
<td>disabling transaction manager, 133–134</td>
<td>entity objects, 11</td>
</tr>
<tr>
<td>Dojo toolkit integration, 236–242</td>
<td>initializing, 175</td>
</tr>
<tr>
<td>dynamic queries, 177–179</td>
<td>mapping, 173–174</td>
</tr>
<tr>
<td>eager loading, lazy loading versus, 77–79</td>
<td>retrieving, 174–175</td>
</tr>
<tr>
<td>EAR files</td>
<td>EntityManager</td>
</tr>
<tr>
<td>Hello World sample application, 24</td>
<td>database access with, 23</td>
</tr>
<tr>
<td>Seam Gen applications, 63–64</td>
<td>manually flushing, 132–133</td>
</tr>
<tr>
<td>Eclipse IDE, Seam Gen in, 69–71</td>
<td>error handling, 183–193</td>
</tr>
<tr>
<td>EJB 3.0, integration with JSF, 3–5</td>
<td>debug information page, 190–193</td>
</tr>
<tr>
<td>EJB3 applications</td>
<td>exception annotations, 186–187</td>
</tr>
<tr>
<td>deployment in Tomcat environments,</td>
<td>exception filters, 185</td>
</tr>
<tr>
<td>339–344</td>
<td>servlet error pages, 184–185</td>
</tr>
<tr>
<td>as templates, 371–377</td>
<td>system exceptions, 188–190</td>
</tr>
<tr>
<td>EJB3 components, POJO components</td>
<td>transactions, 128–129</td>
</tr>
<tr>
<td>versus, 17–19, 170–171, 319. See also</td>
<td>error messages</td>
</tr>
<tr>
<td>session beans</td>
<td>JSF error messages, displaying,</td>
</tr>
<tr>
<td></td>
<td>102–103</td>
</tr>
<tr>
<td></td>
<td>validation error messages, displaying,</td>
</tr>
<tr>
<td></td>
<td>145–147</td>
</tr>
</tbody>
</table>
event context, 84
event handlers, injecting selected objects into, 152–153
Exadel, 208
exception annotations, 186–187
exception filters, 185
exceptions, system, 188–190
exploded JARs, 64
expression language. See Seam JSF EL

G
Garret, Jesse James, 197
getter/setter methods
dependency bijection via, 20–21
in integration testing, 309
GlassFish, deployment in, 315–317
Google Maps, 197
Google Suggest, 197

H
handling web events, Hello World sample application, 15–17
<h: dataTable> UI tag, 150–151
Hello World sample application, 11–17
collection files, 23–30
data models
creating, 13–14
mapping to web forms, 14–15
Facelets, 34–35
packaging, 23–30
testing, 20
web events, handling, 15–17
Hibernate API, Seam POJO application example, 327–331
Hibernate validators, 141
hidden fields, 166–167
hiding span element, 228–229
Hotel Booking example application
conversations across workspaces, 123–124
customizing conversation IDs, 124–126
long running conversations, 103–116
production databases
  defining data sources, 347–348
  installation and setup, 345–347
  JDBC driver installation, 347
  persistence engine configuration, 348–349
  rolling back transactions, 130–132
  workspace switcher, 120–123
  workspaces, 117–120
HTTP GET requests
  conversations and, 101
  query parameters in, 163
HTTP keepalive connections, 355–356
HTTP POST requests, 157
HTTP sessions, 81. See also state management
  memory leaks, 82–83
  state management versus, 99
  workspaces and, 117–119
HTTP thread pool, 355–356
@HttpError annotation, 187

I
ICEfaces, 199
  auto-complete text input example, 202–205
  bundling JAR files for Seam, 205–208
  page parameters, 208
  partial form submission example, 199–202
IDEs
  Eclipse IDE, Seam Gen in, 69–71
  NetBeans IDE, Seam Gen in, 65–69
@IfInvalid annotation, 144
ILOG JView JSF components, 209

@In annotation, 16, 160
initializing
  entity objects, 175
  Seam components, 164–166
input widgets, Dojo toolkit integration
  example, 238–242
installing
  JBoss AS, 12, 366–369
  JDBC drivers, 347
  production databases, 345–347
instantiating DAOs, 172
integration testing, 9, 303–309
  getter/setter methods, 309
  Seam component access with JSF EL
    example, 304–307
  Seam component access with Seam
    API calls example, 307–309
interceptable Java beans, business
  processes and, 256
interrupted conversations, resuming, 120

J
JAR files
  for advanced application templates, 385
  bundling for Tomcat, 335–336, 340–341
  bundling ICEfaces JAR files, 205–208
  exploded JARs, 64
  Hello World sample application, 28–30
Java EE 5.0 environments, deployment
  in, 313–317
Java for bookmarkable web pages, 162–167
Java Persistence API (JPA), 23
JavaBeans, creating JavaScript objects for, 230
JavaScript, accessing Seam components, 234–236. See also AJAX
JavaScript events, triggering, 228–229
JavaScript objects, creating for entity beans/JavaBeans, 230
JavaServer Faces. See JSF
JavaServer Pages. See JSP
JBoss AS
in debugging mode, 68–69
deploying into, 63–64
installing, 12, 366–369
requirements, 365–366
Tomcat versus, 334
JBoss AS 4.0.5, deployment in, 313
JBoss AS 4.2.x/5.x, deployment in, 314–315
JBoss Embeddable EJB3, 339–344
JBoss Enterprise Middleware Suite (JEMS) installer, 12, 366
JBoss MicroContainer, 333
bootstrapping, 339, 342
configuration files, 337–339
JBoss Rules, 277. See also security framework
configuring, 287–288
JBoss Seam. See Seam
jBPM, 245–249
actors, assigning to web users, 245–249
binding data objects, 255–256
configuring, 263–265, 275–276
creating business process instances, 254–255
defining business processes, 252–254
integration with, 6
task management, 257–263
jBPM Pageflow Definition Language. See jPDL
JDBC drivers, installing, 347
JDK 5.0, 365–366
JEMS (JBoss Enterprise Middleware Suite) installer, 12, 366
JPA (Java Persistence API), 23
Seam POJO application example, 320–327
jPDL, 98
JSF
benefits/limitations, 31
integration with EJB 3.0, 3–5
JSP conflicts with, 33
page navigation in, 22–23
JSF component libraries
Ajax4jsf. See Ajax4jsf
ICEfaces. See ICEfaces
list of, 208–209
JSF custom validators, 147–148
JSF enhancements, 31
Facelets, 32–40
adding support for, 44–47
data list components, 39–40
Hello World sample application, 34–35
JSP versus, 32–33
as template engine, 35–39
Seam email support tags, 50–53
Seam filter, 44
Seam formatted text tags, 53–54
Seam JSF EL, 42–43
in clickable data tables, 153–154
Seam PDF tags, 48–50
Seam UI tags, 40–41
  adding support for, 44–47
  stateful JSF, 44
JSF lifecycle phases, 305
JSF messages, displaying, 102–103, 175–176
JSF page navigation. See navigation
JSF replacement tags, 41
JSF web pages, 11
JSP
  error pages, 184–185
  Facelets versus, 32–33
  JSF conflicts with, 33
JVM options, performance tuning, 353–354

K
  keepalive connections, 355–356
King, Gavin, 5, 77

L
  lazy loading, 7
    eager loading versus, 77–79
  @Length annotation, 140
  links in conversations, 114–116
  load balancing sticky sessions, 362
  loading test infrastructure, 299–301
  logged-in users, checking for, 285
  logging, reducing, 354
  logout, 278
  long running conversations, 80, 103–116
    business processes versus, 261
    stateful navigation rules and, 274–275

M
  mapping
    data models to web forms, Hello World sample application, 14–15
    entity objects, 173–174
  @Max annotation, 141
  memory leaks, avoiding, 82–83
  merging into persistence context, 152–153
  messages, displaying JSF messages, 102–103, 175–176
  method-level access control, 282–283
  MicroContainer. See JBoss
    MicroContainer
  @Min annotation, 141
  multipage query results, displaying, 179–182
  multiple transitions in tasks (jBPM), 258–259
  MyFaces, server-side state saving in, 357

N
  navigation
    associating business processes with web pages, 270–274
    in JSF, 22–23
    in stateful example application, 95–98
    stateful navigation rules
      configuring, 275–276
      long running conversations and, 274–275
      in pages.xml file, 267–270
  navigation cases, 96
  navigation rules, 95–96
stateful navigation rules
  configuring, 275–276
  long running conversations and, 274–275
  in pages.xml file, 267–270
visual editors for, 97
NetBeans IDE, Seam Gen in, 65–69
@NotNull annotation, 141

O
Object Relational Mapping. See ORM
ORM
  Seam and, 5
  state management and, 77–79
Otrix, 209
@Out annotation, 16, 95, 160

P
packaging
  EJB3 applications for Tomcat, 339–344
  Hello World sample application, 23–30
  Seam POJO application with JPA example, 325–327
  Seam POJO applications for Tomcat, 335–339
page actions, 161–162
page context, 84
page navigation. See navigation
page parameters
  for bookmarkable web pages, 158–162
  in ICEfaces, 208

pages.xml file
  handling system exceptions, 188–190
  stateful navigation rules, 267–270
partial form submission example, 199–202
@Past annotation, 141
@Pattern annotation, 140, 141
patterns. See design patterns
PDF files, generating, 48–50
PDF tags, 48–50
per-instance access rules, 286–287
performance
  annotations and, 351–352
  JBoss AS versus Tomcat, 334
  state management and, 80–81
  tuning, 352–361
    Call by Value, avoiding, 352–353
    JVM options, 353–354
    logging, reducing, 354
    production databases, 357–358
    second-level database caches, 358–360
    server-side versus client-side state saving, 356–357
    thread pool, 355–356
    transactions, 360–361
performance tags, 41
persistence context, merging into, 152–153
persistence engine, configuring, 348–349
Plain Old Java Objects. See POJO
POJO applications
  deployment of, 319–331
    Hibernate API example, 327–331
    JPA example, 320–327
    in Tomcat environments, 335–339
  as templates, 378–383
Index

POJO components
   conversations and, 101
   EJB3 components versus, 17–19,
      170–171, 319
      as stateful, 92
POJO services with dependency bijection, 7–8
pooledTask component, 261–262
pooledTaskInstanceList component, 261
prerequisites for Seam Gen, 55–56
private fields, accessing, 296
process definitions (jBPM), 246
   creating, 247–248
process instances (jBPM), 246
   creating, 254–255
production databases
   data sources, defining, 347–348
   installation and setup, 345–347
   JDBC driver installation, 347
   performance tuning, 357–358
   persistence engine configuration, 348–349
production deployment. See deploying profiles in Seam Gen, 61–62
programmatic AJAX, 214–217
progress bar example (AJAX), 232–236

Q
queries, 176–182
   dynamic queries, 177–179
   multipage results, displaying, 179–182
query parameters in HTTP GET requests, 163

R
RAM, JVM options, 353–354
   @Range annotation, 141
   rapid application development tools. See Seam Gen
   @Redirect annotation, 186
   redirection, conversations and, 101
   @Remove annotation, 94
   @RequestParameter annotation, 163
   requirements for JBoss AS, 365–366
   @Restrict annotation, 283–285
   resuming interrupted conversations, 120
   retrieving entity objects, 174–175
   return values, rolling back transactions, 131–132
   roles, 277–280
   @Rollback annotation, 131–132
   rolling back transactions, 130–132
   rule-based access control, 283–288
   rules. See business rules

S
scalability
   of clustering, 361
   of state management, 80–81
<s:conversationPropagation> UI component, 116
scope. See contexts
   @Scope annotation, 91
<s:decorate> UI tag, 146–147
Seam
   bundling ICEfaces JAR files for, 205–208
   integration of EJB and JSF, 3–5
Index

ORM and, 5
purpose of, 3
Seam API calls, accessing Seam components (integration testing example), 307–309
Seam applications. See applications
Seam components
accessing from JavaScript, 234–236
accessing with JSF EL, integration testing example, 304–307
accessing with Seam API calls, integration testing example, 307–309
AJAX progress bar example, 232–234
decoupling, 92–93
initializing, 164–166
lifecycle, 93–94
Seam CRUD application framework. See CRUD application framework
Seam filter, 44
Seam Gen, 10
application deployment phase, 63–64
application development phase, 63
application testing phase, 64–65
CRUD application generation, 71–74
in Eclipse IDE, 69–71
loading test infrastructure, 299
in NetBeans IDE, 65–69
prerequisites, 55–56
profiles, 61–62
Seam POJO application deployment, 322, 326
setup, 56–59
skeleton application, generating, 60–61
Tomcat and, 334
Seam JSF EL, 42–43
accessing Seam components, integration testing example, 304–307
in clickable data tables, 153–154
Seam remoting JavaScript library
AJAX progress bar example, 232–236
Dojo toolkit integration, 236–242
validation example, 225–231
second-level database caches, performance tuning, 358–360
security framework, 277–288
declarative access control, 280–283
rule-based access control, 283–288
user authentication/roles, 277–280
selecting tasks (jBPM), 260–263
server connections, detecting, 201–202
server-side state saving, client-side state saving versus, 356–357
server-side validation, 139
Seam remoting JavaScript library example, 226–227
servlet error pages, 184–185
session beans. See also EJB3 components
POJO components versus, 170–171
stateful session beans, 91–93, 255
validation in, 144
session context, 84
setter methods. See getter/setter methods
<s:formattedText> UI component, 53–54
@Size annotation, 141
skeleton application, generating in Seam Gen, 60–61
<s:link> UI component, 115–116
<s:message> UI tag, 145–147
span element, hiding/viewing, 228–229
Index

stack trace, displaying, 187
starting conversations, 108–109
state management
  benefits, 77–86
  boilerplate code, reduction of, 84–86
  browser navigation support, 81–82
  memory leaks, avoiding, 82–83
  ORM usage, 77–79
  performance, 80–81
  stateful contexts, 83–84
  conversations. See conversations
  HTTP sessions versus, 99
  POJO components and, 92
  scalability of, 80–81
  server-side state saving, client-side
    state saving versus, 356–357
  stateful example application, 87–98
    page navigation in, 95–98
    stateful components in, 88–95
state replication in clustering, 363
@Stateful annotation, 91
stateful applications, 5–6
stateful components in stateful example application, 88–95
stateful contexts, 83–84
stateful entity beans, 91
stateful example application, 87–98
  page navigation in, 95–98
  stateful components in, 88–95
stateful JSF, 44
stateful navigation rules
  browser navigation support, 274
  configuring, 275–276
  long running conversations and,
    274–275
  in pages.xml file, 267–270
stateful session beans, 91–93, 255
stateless context, 84
states (jBPM), 246
sticky sessions, load balancing, 362
success messages, displaying, 102–103,
  175–176
Sun blueprint catalog, 209
<s:validate/> UI tag, 142–143
<s:validateAll/> UI tag, 143–144
system exceptions, 188–190
T
task IDs, 259–260
taskInstanceList component, 262
taskInstanceListByType component,
  262–263
tasks (jBPM), 246
  implementing business logic, 257–259
  selecting in UI, 260–263
  specifying, 259–260
template-based email, 50–53
template engine, Facelets as, 35–39
templates
  configuration files for advanced application templates, 385
  EJB3 applications as, 371–377
  Seam POJO applications as, 378–383
  Tomcat applications as, 383–384
@Test annotation, 294
test infrastructure, loading, 299–301
testing
   Hello World sample application, 20
   integration testing, 9, 303–309
       getter/setter methods, 309
   Seam component access with JSF
       EL example, 304–307
   Seam component access with Seam
       API calls example, 307–309
Seam Gen applications, 64–65
unit testing, 9, 291–301
   database services, 297–299
   dependency bijection, 295–296
   loading test infrastructure,
       299–301
   TestNG test case example, 293–295
TestNG, 293
   test case example, 293–295
text input example, 202–205
third-party JavaScript libraries, Dojo
   toolkit integration example, 236–242
thread pool, 355–356
Ticketing example application
   assigning jBPM actors to web users,
       245–249
   binding data objects, 255–256
   creating business process instances,
       254–255
   defining business processes, 252–254
   task management, 257–263
timeouts for conversations, 114
Tomcat
   Apache Tomcat Connector, 362
   deployment in, 333–344
       EJB3 applications, 339–344
       Seam POJO applications, 335–339
   JBoss AS versus, 334
   Tomcat applications as templates,
       383–384
tools support. See Seam Gen
transaction manager, disabling, 133–134
  @Transactional annotation, 129
  @TransactionAttribute annotation, 129, 134
transactions, 127–134
   atomic conversations, 132–134
   error handling, 128–129
   forced rollbacks, 130–132
   performance tuning, 360–361
transitions, multiple transitions in tasks
   (jBPM), 258–259
triggering
   JavaScript events, 228–229
   validation, 142–144
tuning performance, 352–361
   Call by Value, avoiding, 352–353
   JVM options, 353–354
   logging, reducing, 354
   production databases, 357–358
   second-level database caches, 358–360
   server-side versus client-side state sav-
       ing, 356–357
   thread pool, 355–356
   transactions, 360–361
UI
UI components, declarative access con-
   trol, 281–282
UI tags, 40–41
   adding support for, 44–47
   triggering validation, 142–144
   validation error messages, 145–147
unit testing, 9, 291–301
  database services, 297–299
  dependency bijection, 295–296
  loading test infrastructure, 299–301
  TestNG test case example, 293–295

URLs. See bookmarkable web pages

user authentication, 277–280
user roles, 277–280

V

@Valid annotation, 141, 144
validation, 137–148
  Ajax4jsf example, 212–214
  entity bean annotations, 139–141
  error messages, displaying, 145–147
  JSF custom validators, 147–148
  partial form submission example, 199–202
  Seam remoting JavaScript library example, 225–231
  server-side validation, 139
  in session beans, 144
  triggering, 142–144
validation tags, 40, 142–144
viewing span element, 228–229
visual editors for navigation rules, 97
visual effects, Dojo toolkit integration example, 236–238

W

WAR files
  Hello World sample application, 26–28
  Seam POJO application with JPA example, 325–327

Web 2.0 applications, 7
Web Application aRchive. See WAR files
web events, handling (Hello World sample application), 15–17
web forms, mapping data models to (Hello World sample application), 14–15
web pages
  associating business processes with, 270–274
  bookmarkable web pages. See bookmarkable web pages
declarative access control, 280–281
web transactions. See atomic conversations
@WebRemote annotation, 227
Wikitext, 53–54
Woodstock project, 209
workspace switcher, 120–123
workspaces, 117–120
  conversations across, 123–124
  customizing conversation IDs, 124–126
  workspace switcher, 120–123
wrapper code, reduction of, 84–86

X

XHTML. See Facelets
XML files, 8–9
  Hello World sample application, 23–30