This bibliography reflects the age we live in: we draw our information not only from traditional publications and media (books, professional, and academic journal articles), but mostly from information available on the web. Hence the different categories:

- **Books.** These are good old paper books.
- **Articles and papers.** These include professional, trade, or academic publications.
- **Web sites.** Here we grouped web-bound organized sources of information, including blogs, conference web sites, technologies (e.g., WS-BPEL, EPF, J2EE, etc.), trade or standards organizations (e.g., OMG, the Business Rules Group), and on-line, or mixed-media (paper and on-line) publications.
- **Documents.** Although these are mostly available on the web, they do not represent elaborate sources of information like the many portals we find under web sites.
- **Tools.** We list some of the tools that are relevant to the book.

### Books

- Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, *Design Patterns: Elements of Reusable Design*, published by Addison-Wesley, 1994
• Naeem Siddiqui, Credit Risk Scorecards: Developing and Implementing Intelligent Credit Scoring by John Wiley & Sons (2006), ISBN 9780471754510

**Articles and Papers (Professional Journals, Trade Magazines, etc.)**

• Alasdair Urquhart, “Emil Post”, in Handbook of the History of Logic, vol 5: Logic from Russell to Church, Dov M. Gabbay and John Woods Eds, pp. 429–478

**Web Sites**

• Blogs: there are too many, high quality, informative blogs on business rules. Here we list the ones we referenced in specific parts of the book:
  • Pierre Berlandier explores different alternatives for structuring ruleset which can impact both performance and management. See http://www.ibm.com/developerworks/websphere/library/techarticles/1003_berlandier/ 1003_berlandier.html
  • John D McGregor athttp://www.cs.clemson.edu/~johnmc/joop/col3/column3.html details the component testing, needs, and approach
  • James Taylor on Everything Decision Management at http://jtonedm.com/
• Conferences: there are *many many academic* conferences on a variety of areas related to business rules, including the main AI conferences (AAAI, IJCAI, ICTAI, etc), that would be too numerous to mention. Here we simply list the two main industrial conferences on *business rules*:
The business rules forum (http://www.businessrulesforum.com) is an annual conference for people interested in the business rules approach and is a good opportunity for learning about new product features and cutting-edge thinking.

Rule Fest (http://www.rulefest.org)

Technologies

- DAO (Sun Core J2EE Patterns - Data Access Object), at http://java.sun.com/blueprints/corej2eepatterns PATTERNS/DataAccessObject.html
- IBM WebSphere ILOG JRules product documentation can be found at http://publib.boulder.ibm.com/infocenter/brjrules/v7r1/index.jsp
- The java architecture for XML binding (JAXB) specification and documentation is at http://www.oracle.com/technetwork/articles/javase/index-140168.html
- Java Management Extensions (JMX) at http://java.sun.com/javase/technologies/core/mntr-mgmt/javamanagement/
- Java Runtime API for Rule Engines (JSR 94) specification http://jcp.org/aboutJava/communityprocess/review/jsr094/index.html
- The latest of the Java Thread API is documented at http://download-llnw.oracle.com/javase/6/docs/api/
- OpenUp http://epf.eclipse.org/wikis/openup/
- Object pooling, Apache project for Commons pool at http://commons.apache.org/pool/
- Rule Interchange Format http://www.w3.org/blog/SW/2010/06/22/w3c_rif_recommendation_published
- SID, a reference model for telecom service providers and vendors, can be downloaded from the web site of the tele-management forum, at http://www.tmforum.org/InformationFramework/1684/home.html
- The Service Component Architecture (SCA):
  - A forum is hosted by the Open Service Oriented Architecture collaboration (OSOA), at http://www.osoa.org/display/main/service+component+architecture+home
  - Apache Tuscany open source implementation of the SCA specification at http://tuscan.apache.org/
- Service Data Objects:
  - Specification at http://www.osoa.org/display/main/service+data+objects+home
  - SDO Apache tuscany project in http://tuscan.apache.org/SDOOverview.html
- Web Ontology Language (OWL) http://www.w3.org/2004/OWL/
- Trade/industry/standard organizations
• MISMO is a “technology standards development body for the residential and commercial real estate finance industries, ... wholly owned subsidiary of the Mortgage Bankers Association,” http://www.mismo.org
• The Object Management Group (http://www.omg.org) has a number of active standards related to business rules, a number of which are based on (more readable) submissions of the business rules group
• On-line and mixed-media publications
  • http://intelligent-enterprise.infosecurityweek.com
  • http://www.information-management.com/channels/decision_management.html
  • http://www.kmworld.com
  • http://www.bptrends.com
  • InfoWorld publishes, with some regularity, the results of some academic benchmarks using the latest versions of the best known commercial and open-source engines (http://www.infoworld.com)

Documents (standards, web documents, etc.)

• OMG, Semantics of Business Vocabulary and Business Rules, SBVR v1.0, January 2008
• The ACORD data model can be found at http://www.acord.org
• Withholding of Tax on Nonresident Aliens and Foreign Entities, Publication 515 (rev. Aril 2008), Department of the Treasury, Internal Revenue Service
• Test-driven development: http://en.wikipedia.org/wiki/Test-driven_development
• Semantics of Business Vocabulary and Business Rules (SBVR), v1.0 http://www.omg.org/spec/SBVR/1.0/PDF
• Business Motivation Model V 1.0, Object Management Group
• For an introductory definition of finite state machines, check http://en.wikipedia.org/wiki/Finite-state_machine
• A good introduction to the RETE network that explores more advanced concepts like the handling of ORs and negations. See http://en.wikipedia.org/wiki/Rete_algorithm
• Adaptive Object Model pattern, by Joe Yoder, see http://adaptiveobjectmodel.com/
• Introduction to boolean algebra:
  • http://en.wikipedia.org/wiki/Introduction_to_Boolean_algebra
  • http://www.internettutorials.net/boolean.asp.
• The GOF Command pattern is included in the GOF book, but also explained in http://en.wikipedia.org/wiki/Command_pattern

Tools

In this section, we list the tools that we either referred to in the book or that are relevant. In particular, we listed some of the commercial BRMSs (Blaze Advisor, PegaRules, Corticon) and some open source/free ones (DROOLS, JESS). There are many more that we did not list. This list is given for illustration purposes only. We also listed other tools that were referred to in the book, including Junit, and some early BRMS tools (see Chap. 7).
• Blaze Advisor, by FICO (http://www.fico.com/en/Products/DMTools/Pages/FICO-Blaze-Advisor-System.aspx)
• Corticon, http://www.corticon.com/
• DROOLS, the Jboss open source BRMS, http://jboss.org/drools/
• JESS, an open-source Java Expert System Shell, from Sandia National Laboratories, at http://herzberg.ca.sandia.gov/jess/
• JUnit, the unit testing framework mentioned many times in the book, is available at http://www.junit.org/
• International Council on Systems Engineering maintains a survey of requirements management tools (http://www.incose.org/productspubs/products/rmsurvey.aspx)
• BRS RuleTrack is a product of Business Rule Solutions, http://www.BRSolutions.com
• RuleXpress is a product of RuleArts, check http://www.RuleArts.com
• Versata, http://www.versata.com
Index

A
ABRD. See Agile business rule development
Abstraction, 191
ACID, 392
Action enabler coding pattern. See Rule coding patterns
Action enabler rule, 265
Action part, 341
Action phrase, 305. See also JRules business object model
Action rules, 341
grammar, 342
Actions, 347
Adaptive object model, 120
Agile alliance manifesto, 51
Agile business rule development, 49
activities, 52
build models, 127
build structures for rule development and execution, 130
communicate back to business, 142
define rule metamodel, 132
deployment, 56
design rule execution flow, 134
design rule project organization, 130, 269
determine rule implementation, 117
determine rule project organization, 130, 269
determine rule implementation, 117
ear to end testing, 56
rule analysis, 52, 73, 100
rule authoring, 53, 54
rule deployment, 53
rule design, 52
rule discovery, 52, 54, 73, 74
rule prototyping, 136
rule validation, 53
approach, 52
customization, 65
cycles
building, 55
enhancing, 56
harvesting, 53
integrating, 56
prototyping, 54, 115, 448
rule harvesting, 73
delivery process, 58
guidance, 58
iterations, 73
principles, 50
roles, 58, 61
business analyst, 61
rule administrator, 64
rule analyst, 61
rule architect, 62
rule writer, 63
subject matter expert, 64
structure, 59
tasks, 58
usage scenarios, 65
work products, 58, 64
business process description, 64
decision point table, 54, 64, 75
decision service, 56
executable rules, 55
rule execution flow, 134
rule metamodel, 132
rule project organization, 130, 269
unit test cases, 55
use cases description, 64
Agile rule maintenance, 45
Agile software development, 49, 50
AI. See Artificial intelligence
Alarm filtering and correlation, 463. See also Business rules approach
Algorithm selection, 372
decision framework, 383
AOM. See Adaptive object model
Apache Tuscany SDO, 399
Artificial intelligence, 148, 150
   expert system, 150
   knowledge-based system, 150
Atomicity, 393
Automated underwriting systems. See Business rule, application; Business rules approach
Autonomy, 191

B
BAL. See Business action language
BOM. See Business object model; JRules
Boolean logic, 106
BPEL. See Business process execution language
BPM. See Business process management
BPM center of excellence, 507
BPM governance, 504
BQL. See Business query language
BRLDF. See Business rule language development framework
BRM group. See Rule management group
Building test scenarios, 109
Build ruleset, 414
Business action language, 334, 341
   action part, 341
   actions, 347
   condition definition, 346
   condition part, 341, 345
definitions part, 341, 343
   grammar, 342
   IRL translation, 342
Business analyst, 61
Business conversation language, 246
Business event. See Business rule
Business modeling, 53. See also Agile business rule development, cycles
Business motivation model, 7, 76
Business object lifecycle, 135
Business object model, 247, 286, 301
   BOM design
      bottom-up, 228
top-down, 229
   BOM path, 290
   BOM sharing, 325
   BOM to XOM mapping, 228, 301, 308
custom mapping, 309
   virtual class, 311, 313
   virtual function, 309
Business policy, 76
Business process description, 64
Business process execution language, 67
Business process management, 49, 193

Business query language, 225, 288, 541
Business rule
   action enabler coding pattern, 265
   application, 13
      architecture, 177, 184
      BRMS integration, 184
code, 15
deployment, 15
design options, 184
   integration options, 177
J2EE application, 187
legacy data access layer, 208
maintenance, 16
message-oriented architecture, 185, 189
off-line, 187
re-engineering legacy application, 177, 204
re-engineering the application layer, 205
re-engineering the business layer, 207
re-engineering the data layer, 209, 249
rule engine integration, 194
run-time, 15
service-oriented architecture, 185, 191
standalone application, 185, 186
synchronous client-server application, 185, 187
atomic rule, 105, 259
behavioral rule, 7
business perspective, 6
category filter, 321
classification, 75
   action enabler, 265
   computation, 264
   constraint, 258
guideline, 264
   inference, 264
   risk assessment rule, 265
coding pattern, 257
   computation coding pattern, 264
   constraint coding pattern, 257
custom rule language, 256, 357
   definition, 5, 76
guideline coding pattern, 264
   implementing in application code, 117, 120
   implementing in GUI, 117, 121, 201
   implementing within BPM/workflow tool, 117, 123
   implementing within the data model, 117, 118
   implementing with rule engines, 118, 124
   inference coding pattern, 264
   information system perspective, 6
   lifecycle, 50, 337
lifecycle versions, 184
maintenance and release cycle, 17
metadata, 337
motivations, 8
ontology, 180
rule template, 180
structural rule, 7, 77
Business rule language development framework (BRLDF), 256, 334, 357
abstract syntax, 358
BAL customization, 360
concrete syntax, 358
parsers, 359
parsing and translation framework, 359
rule extension model, 359
translators, 359
Business rule management system, 13, 49, 178, 270
late business rule management system, 179
rule automation, 13, 15
rule management, 13
rule repository, 14
Business rules approach, 12, 13, 24
applications, 27
engineering applications, 28
financial services applications, 33
history, 27
insurance applications, 38, 43
methodology, 16
motivations, 24
re-engineering legacy applications, 21
target domains, 44
Business rules classification, 75, 77
action enabler, 78
compliance rule, 80
constraint, 78
correlation rule, 80
event condition action rule (ECA), 78
guideline, 78
inference rule, 79
operation rule, 76, 77
rating rule, 80
routing rule, 78
stateful rule, 80
structural rule, 76, 77
transformation rule, 79
Business rules expertise center, 19, 20
Business rules methodology, 16, 24
analysis and design, 23
authoring, 23
coding, 23
maintenance, 24
methodology matrix, 22
requirements, 22
Business rules modeling. See Rule discovery
Business transaction dispatching, 395
Business vocabulary, 76, 247
C
Car insurance. See Business rules approach
Categories, 320
Category filter, 321
CEP. See Complex event processing
Change management committee, 515
Change request, 534
Claim. See Claim processing
Claim adjudication, 41. See also Business rules approach
Claim conceptual data model, 111
Claim eligibility. See Claim processing
Claim processing, 41. See also Business rules approach
Claim validation. See Claim processing
Class condition, 337
Client-server application
J2EE, 189
web tier, 188
Client-server architecture, 187
J2EE, 187
Cognitive plausibility, 149
Cognitive psychology, 148
Collection condition, 339, 346
Command design pattern, 139
Common Client Interface, 429
Complex event processing, 79
Compliance rule, 80
Component-oriented development, 192
Component test, 453
Composability, 192
Computation coding pattern. See Rule coding patterns
Computation rule, 264
Condition part, 341, 345
Conflicting rule, 466
Conflict resolution strategy, 153
rule condition strength, 153
rule priority, 153
rule recency, 153
Consistency, 393
Consistency checking, 465, 472
Constraint coding pattern. See Rule coding patterns
Constraint rule, 258
necessary conditions, 259
Construction, 59
Continuous testing, 463
Control strategy, 152, 154
Correlation rule, 80
Credit scoring, 254
literal domains, 322
method verbalization, 306
refactoring, 316
refactoring BOM changes, 319
refactoring verbalization changes, 320
refactoring XOM changes, 316
static references, 323
use BOM to XOM mapping sparingly, 328
verbalization, 305, 306
virtual class, 311, 313
virtual function, 309
vocabulary, 301
XML Schema-based XOM, 313
decision validation service, 218
executable object model, 218
module, 216
rule artifacts, 285, 334
action rules, 341
Business Action Language, 334, 341
decision table, 334, 348
decision tree, 334, 353
scorecard, 334, 354
technical rules, 334, 335
rule authoring infrastructure, 284
rule engine, 155
agenda, 155
API, 155
concept of operations, 156
control strategy, 167, 168
Fastpath algorithm, 172, 174
IlrContext, 155
IlrRuleset, 155
inserting an object, 158
RETE algorithm, 162, 173
sequential algorithm, 168, 173
updating an object, 160
working memory, 155
rule execution server, 197, 217, 232, 236
rule extension model, 226
rule project, 284
dependencies, 289, 324
managing multiple users, 296
map, 222
organization best practices, 324
project security, 299
project synchronization, 291
references, 289
RTS locking, 301
RTS permission management, 300
RTS project security, 299
Rule Studio project, 285
Rule team server project, 292
source code management, 297
synchronization best practices, 295
synchronization conflicts, 294
Rule Solutions for Office, 217, 239
rule studio, 217, 221
rule team server, 217, 232
JRules BOM. See JRules, business object model
JSR 94, 198, 408
JRules support, 426
object filtering, 411
rule administrator, 412
rule runtime, 410
rule service provider, 409
session, 412
JUnit, 450
Just In Time, 171
bytecode generation, 171
K
Keywords
insert, 462
modify, 462
not, 463
retract, 462
update, 462
Knowledge-based system, 150
Knowledge engineering, 150
Knowledge representation language, 247
L
Late business rule management system, 179
Loan-to-value ratio, 35. See also Business rules approach
Loose coupling, 190, 191
LTV. See Loan-to-value ratio
M
Master data management, 50, 130
MDM. See Master data management
Message-oriented architecture, 185, 189
interoperability, 189
loose coupling, 190
robustness, 190
scalability, 190
Message-oriented middleware, 189, 405
Mockup, 450
Model-view-controller, 188
web application, 189
MOM. See Message-oriented middleware
Mortgage underwriting. See Business rules approach
MVC. See Model-view-controller
N
Navigation phrase, 305. See also JRules, business object model
Network of joins, 163

O
OAA. See Object-oriented analysis
Object-oriented analysis, 103
Ontology Web Language, 76
OpenUp, 50, 59
  phases
    construction, 59
    elaboration, 59
    inception, 59
    transition, 59
Open XML, 239
Optimistic locking, 297. See also JRules, rule project
Orchestration of rule execution, 54, 360
  best practices, 375, 379
  design the signature first, 375
  execution algorithm selection, 379
  rule flow, 360
  rule flow granularity, 377
  ruleset parameters, 360
OWL. See Ontology Web Language

P
Pattern matching, 461, 496
Performance test, 456
Performance variables, 458
Permanent link. See Rule Team Server
Physical deployment, 415
Point to point messaging, 406
Policy underwriting, 38. See also Business rules approach
Policy underwriting system, 40. See also Business rules approach
Pool size, 493
Potential performance bottlenecks, 459
Preventive maintenance. See Business rules approach
Production system, 147, 151
  control strategy, 152
  database, 151
  interpreter, 151
  principles, 151
  ruleset, 151
Project structure design, 54, 270
Prototyping, 54
  activities
    build Java models, 127
    build models, 127
    build structures for rule development and execution, 130
    build XML data models, 128
    communicate back to business, 142
    define rule metamodel, 132
    design rule execution flow, 134
    design rule project organization, 130
    determine rule implementation, 117
    rule prototyping, 136
    synchronize rules with data models, 129
  work products
    rule coding patterns, 137
    rule execution flow, 134
    rule metamodel, 132
    rule project organization, 130
    unit testing framework, 137
Publish/subscribe messaging, 406

Q
Query, 442

R
Racing, 396
Rating rule, 80
Rational unified process, 59
RDF. See Resource description framework
Redundant rule, 466
Refactoring, 316
  BOM changes, 319
  verbalization changes, 320
  XOM changes, 316
Referential data access, 130
Refraction, 167. See also JRules, rule engine
Regression test, 456
Resource adapter descriptor, 493
Resource description framework, 76
RETE algorithm, 153, 154, 161, 173, 381, 494
  RETE network, 162, 164
    condition sharing, 165, 167
    discrimination tree, 163
    network of joins, 163
    object insertion, 165
    object modification, 165
    object removal, 165
Reusability, 191
Risk assessment. See also Business rules approach
  financial services, 34
  insurance, 40
  rule, 265
Risk model, 266
Risk scoring, 254
Robustness, 190
RTS. See Rule team server
Rule administrator, 64, 508
Rule analysis
activities
analyze rule descriptions and fact models, 100
check rules completeness, 108
identifying rule patterns, 105
make rules atomic, 105
object oriented analysis, 103
remove redundant rules, 108
resolve inconsistent rules, 108
term and fact analysis, 104
transforming rules, 105
sample process, 181
Rule analyst, 61, 508, 515
RuleApp, 232, 421
Rule architect, 62, 508
Rule author, 508, 514, 518
Rule authoring, 54, 245
infrastructure, 283
language, 251
vocabulary, 284
Rule automation, 44
Rule-based programming, 147
history, 147, 148
Rule capture
sample process, 181
Rule change process, 535
Rule coding patterns, 137, 257
action enabler coding pattern, 265
computation coding pattern, 264
constraint coding pattern, 258
delay rule actions, 138
encoding business data, 267
encoding pricing tables, 269
encoding product tables, 268
explain decisions, 137
guideline coding pattern, 264
inference coding pattern, 264
rule maintenance, 259
test for data quality first, 139
Rule coverage, 475
Rule developer, 51
Rule discovery, 73
activities, 80
define rule discovery roadmap, 84
execute rule discovery roadmap, 88
gather the related documents, 86
plan rule discovery, 87
review business process map, 82
review decision points table, 82, 86
formal rule language, 90
roadmap
analysis technique, 81
automated analysis, 85
business event-driven, 82
business mission-driven, 82
business process driven, 82, 83, 96
data analysis driven, 82
discovering rules from code, 92
discovering rules from documents, 91
discover rules from SMEs, 90
document the business rules, 92
interactive analysis, 85
rule sources, 81
static analysis, 85
use case driven, 82
rule languages, 88
use case approach, 82
work products, 75
architecture description, 75
business process description, 75
data model, 75
decision points table, 75
object model, 97
workshop, 54
RuleDocs, 239
Rule engine
cognitive plausibility, 149
conflict resolution strategy, 153
deployment, 194, 196
business tier, 201
client tier, 201
EIS tier, 202
embedded rule engine, 196
impact of architecture of application, 198
options, 194, 196
rule execution service, 197
web tier, 201
which tier in a multi-tier application, 199
embedded rule engine, 196
execution algorithms, 161, 369
algorithm selection, 372
exit criteria, 373
Fastpath algorithm, 172, 174
ordering, 373
parameters, 373
performance, 174
RETE algorithm, 162, 173
sequential algorithm, 168, 173
Fastpath algorithm, 153
integration options, 177
pooling, 395
principles, 147
RETE algorithm, 153
rule execution algorithms, 161
rule execution service, 197
sequential algorithm, 153
Rule execution language, 246
Rule execution server, 197
console, 238, 423
execution unit (XU), 429
invocation pattern, 431
JMS support, 433
management model, 430
monitoring, 427
ra.xml, 430
resource adapter (RAR), 428 (see also Resource adapter)
service component architecture, 434
Rule execution service
hot deployment, 198
remote ruleset execution, 197
ruleset versioning, 198
scalability, 198
separate ruleset deployment, 198
Rule execution set, 408
Rule flow, 134, 136, 231, 334, 360, 365
branch node, 366
flotask, 365, 369
fork, 366
function tasks, 365
IRL translation, 368
join, 366
rule execution algorithm, 369
rule selection, 370
rule task, 365
run-time rule selection, 369, 371
tasks, 365
transition, 365
Ruleflows. See Rule flow
Rule governance, 504
activities, 506
change request, 515
identifying stakeholders, 508
organization map, 509
processes, 506
roles, 508, 511
RTS roles, 524
rule change, 517
rule management group, 509
status property, 526
Rule harvesting, 73
activities
building test scenarios, 109
verify rules against the data models, 110
Rule implementation criteria
adaptability, 118
auditability, 118
manageability, 118
reusability, 118
traceability, 118
Rule inconsistency, 465
Rule language, 246
authoring language, 247
best practices, 384
business conversation language, 246
business object model, 247
business vocabulary, 247
custom languages, 256
decision table, 253, 334, 348
decision tree, 254, 334, 353
do you really need a custom language, 384
execution language, 246, 247
formal rule authoring language, 246
if-then rule, 251
scorecard, 254, 334, 354
structured natural language, 246
Rule life cycle
classical, 514
definition, 513
rule team server, 526
simple, 513
Rule management, 44
asynchronous, 17
group, 511
organization, 17, 18
business rules expertise center, 19, 20
synchronous, 17
Rule metamodel. See JRules
Rule never applies, 466
Rule never selected, 466
Rule orchestration, 231
Rule package, 270
Rule priority, 153, 168
Rule project, 130, 270, 284
dependencies, 289
JRules rule project, 284
managing multiple users, 296
ruleset parameters, 360, 361
Rule project organization, 131, 270, 284, 289
best practices, 324
design drivers, 271
problem domain dimension, 272
promoting BOM sharing, 325
promoting rule reuse, 324
separating orchestration from rules, 326
Rule project organization pattern
Rule project organization pattern (cont.)
- effective division of labor, 274
- favor structure over computation, 279
- jurisdiction containment pattern, 272, 278
- metadata dimension, 278
- process decomposition pattern, 272, 277
- product and service portfolio, 272
- product specialization pattern, 276
- reusability, 274
- simple mapping to execution structure, 275
- understandability, 274

Rule project security, 299
Rule project structure, 284, 285
  - Rule Studio, 285
Rule project synchronization, 291, 294
  - best practices, 295
Rule property, 132, 225
Rule query, 225, 287
Rule recency, 168
Rule repository, 233, 270
Rule selection, 369
  - at runtime, 417
Rule session, 408
Rule set, 151
  - archive, 232
  - deployment using JMX, 416
  - extraction, 134
  - extractor, 232
  - map, 512
  - parameter, 360, 361
    - passing data to the engine, 362
    - rule references, 363
  - path, 422
  - project
    - variables, 364
    - versioning, 416
Rules never selected, 472
Rule steward, 508
Rule task, 231
  - algorithm selection, 372
  - dynamic run-time rule selection, 372
  - fastpath algorithm, 373
  - RETE algorithm, 373
  - rule execution algorithm, 369
  - rule selection, 369, 370
  - run-time rule selection, 369, 371
  - run-time rule selection filter, 371
  - sequential algorithm, 373
  - static rule selection, 369
  - static run-time rule selection, 372
Rule team server, 298
  - baseline, 530
  - data source, 533
  - deployment, 441
  - parallel release development, 532
  - permanent link, 536
  - permission management, 529
  - recycle bin, 531
  - role-based access control, 298
  - rule studio synchronization, 533
  - session, 528
  - session controller, 526, 528
  - workflow engine integration, 538
Rule template, 180, 225, 288
  - freezing parts, 289
Rule tester, 508, 514
Rule tracing, 467
Rule unit testing, 476
Rule vocabulary, 228
Rule writer, 63
Run-time rule selection, 370
  - dynamic, 372
  - filter, 371
    - hierarchical properties, 371
    - static, 372
S
SBVR. See Semantics of business vocabulary and business rules
Scalability, 190, 394
  - horizontal, 396
  - vertical, 394
Scenario
  - DVS, 478, 488, 490
  - service provider, 236, 479, 480
SCM. See Source code management
Scope expression, 339
Scorecard, 266, 334, 354
  - credit scoring, 254
  - decision model, 355
  - IRL translation, 356
  - reason codes, 355
  - reasoning, 355
  - reasoning strategy, 355
  - risk scoring, 254
Semantic query, 474
Semantics of business vocabulary and business rules, 75, 247
Sequential algorithm, 153, 168, 169, 173, 494
  - default tuple generator, 381
  - heterogeneous rule signature, 170, 172, 382
  - JIT bytecode generation, 171
  - parameters, 170
  - restrictions, 171
  - tuple generator, 170
Index

Service, 191
Service characteristics, 191
Service component architecture, 406, 448
   components, 406, 434
   composite, 407, 453
Service data object, 398, 452, 460
Service definition, 403
   bottom-up, 404
   meet in the middle, 403
   top-down, 405
Service-oriented analysis, 193
Service-oriented architecture, 128, 185, 191
   enterprise service bus, 193
Service-oriented computing, 191
   abstraction, 191
   autonomy, 191
   composability, 192
   discoverability, 192
   influences, 192
   loose coupling, 191
   reusability, 191
   standardized contracts, 191
   statelessness, 192
Service-oriented engineering, 193
   domain analysis, 194
   meet in the middle approach, 194
SME. See Subject matter expert
SOA. See Service-oriented architecture
SOC. See Service-oriented computing
Software development lifecycle, 51
Software process engineering, 57. See also
   Eclipse process framework
Source code management, repository, 297
SSP. See Scenario service provider
Standalone application, 185, 186
Standardized contracts, 191
Stateful rule, 80
Stateful session, 409
Statelessness, 192
Stateless session, 409
Static rule selection, 369
Static run-time rule selection, 372
Strategy pattern, 207
Structured natural language, 246
Subject. See JRules business object model
Subject matter expert, 51, 64
Synchronous client-server application,
   185, 187

T
Task
   final actions, 365
   initial actions, 365

Tax reporting and withholding. See Business
   rules approach
TDD. See Test driven development
Technical rules, 334, 335
Telecommunications network management.
   See Business rules approach
Term, 104
Term and fact analysis, 104
Test driven development, 55, 110, 450
Test goals, 449
Test scenario, 235
Thread safety, 396
Train car maintenance. See Business rules
   approach
Transaction, 392
Transition, 59
Transparent decision service, 438
Truth maintenance, 340
Tuple generator. See Sequential algorithm
Two-phase commits, 393

U
Underwriting rules. See Business rules
   approach
Unit test cases, 55
Unit testing, framework, 137
Use cases description, 64

V
Verbalization, 305. See also JRules, business
   object model
Virtual class, 311, 313. See also Business
   object model, BOM to XOM
   mapping
Virtual function, 309. See also Business object
   model, BOM to XOM mapping

W
Web service, 128, 193
Web Service Description Language, 128
WebService Java annotation, 405
Working memory, 155
WSDL. See Web Service Description
   Language

X
XML schema-based XOM, 313
XML schema description, 128, 129
XSD. See XML schema description
XSD-based BOM to BOM mapping, 314