

## Symbols

---

- 227  
^ 213, 227  
! 227  
? 227  
“ 227  
( ) 227  
{ 227  
} 227  
\* 227  
\  
&& 227  
+ 227  
|| 227  
~. *See* tilde

## Numerics

---

2PC 143

## A

---

AbstractJMSHibernateSearch-  
Controller 323  
accent 130  
access strategy 41  
ACID 141  
acronyms 125, 128  
active-passive 317  
ad hoc queries 386  
generation 212  
adapter class 369, 430  
Adobe 418  
Aelfred2 430

AJAX 143  
AliastoBeanConstructorResult-  
Transformer 192  
AliastoBeanResultTransformer  
192  
Altavista 12  
Amazon 5, 183, 249  
book search screen  
example 5, 7  
@Analyzer 218  
@AnalyzerDef 127  
analyzers 15, 31, 45, 53, 83, 116,  
125, 216–223, 415  
apply 127  
applying manually 219  
automatically applied to a  
query 221  
defining for specific  
field 83  
definition 127  
filter 127  
global 83, 220, 223  
mixing 84  
non-English language 130  
oddities 217–218  
performance 277  
query-synonyms 217  
specify a definition 127  
tokenizer 127  
annotations 67  
ANT 400  
Apache Commons  
Codec 131  
Collection 270  
Apache Lucene. *See* Lucene  
Apache Software Foundation 29

Apache Solr 20, 126  
analyzers 216  
ApacheCon 2007 388  
apache-solr-analyzer.jar 216  
apostrophe 125, 128  
appliance solutions 19  
application server 30  
approximation 44  
architecture 121, 145, 310  
array 109  
synchronized 228  
associated objects 110  
association 284  
bidirectional 110  
performance 277  
*See also* relationship  
async 146  
asynchronous 277  
clustering 314  
copy (index) 124  
mode 316  
attribute, access 75  
Automatic optimization 291

## B

---

backend 144, 314  
BackendQueueProcessorFactory  
150, 318  
Backus-Naur notation 213  
base query components 214  
batch 153, 156  
mode 293  
batch size 174, 179  
best value 175

- @BatchSize 279
  - Bauer, Christian 175
  - benchmark 308
  - Berkeley 117
  - Bialecki, Andrzej 48
  - bidirectional association 110
  - bit mask 251, 287
  - BitSet 253, 270
  - blind
    - feedback 388
    - relevance feedback 388
  - blob 312
  - Boolean
    - operators 44
    - queries 203–205, 283
    - retrieval 364
  - Boolean keyword, NOT 404
  - Boolean.Occur
    - MUST 246
    - MUST\_NOT 246
    - SHOULD 246
  - BooleanClause 246
  - BooleanClause.Occur flags 228
  - BooleanQuery 244–247, 249, 252, 267, 340, 372
    - increased memory usage 234
    - word syntax 203
  - BOOST 347
  - @Boost 85, 211, 247–248
    - example 248
  - boost 365, 405
    - APIs 247–249
    - in @ClassBridge 248
    - index time vs. query time 86
    - negative 85
    - one entity over an other 86
    - order of precedence 248
    - property 85
  - boost factor 46, 210, 247, 309, 375
    - default 210
    - document 366
    - example 210
    - field 366
    - range of values 210
  - boosting 354, 409
    - single term query 381
  - BoostingQuery 249, 404–409
    - boost greater than 1.0F 409
    - code signature 405
    - context 405
    - example of 407
    - match 405
    - scores after 408
  - bottleneck 308
  - brackets, do not mix 210
  - bridge 89
    - built-in 76, 90
    - class level 92, 261
    - custom 89, 241
    - definition 66
    - exception 94
    - flexible custom 99
    - JMS 320
    - Map 99
    - null 66, 78, 94
    - one-way flexible 99
    - one-way simple custom 93
    - one-way vs two-way 96
    - parameters 97
    - simple custom 93
    - thread-safety 98
    - two-way 90, 188
    - two-way flexible 101
    - two-way simple custom 95
    - type inference system 91
    - using 91
  - browse index documents 52
  - buffer 223
  - built-in directory providers 124
- C**
- 
- C, source code for determining
    - relevance 387
  - C# 135
  - C++ 208
  - cache 286–287
    - distributed 318
    - filter 287
  - CachingWrapperFilter 254
  - callback
    - handlers 438
    - methods 430, 434
  - caret symbol in boost factor 210
  - case 128
  - category
    - filter 288
    - navigation system 5
    - tree 5
  - CD class 431
  - change tracking 140
  - Chinese 126
  - class name 74
  - CLASS\_FIELDNAME 74, 173
  - @ClassBridge 92–93
  - ClassCastException 171
  - classpath 31
  - clauses, number of 283
  - clear() 180
  - close() (FullTextSession) 167
  - cluster 120, 147, 166, 310, 329, 345–346
  - clustering 311
    - asynchronous 314
    - asynchronous
      - (variations) 317
    - directory provider
      - (master) 325
    - directory provider (slave) 321
    - in-memory distributed 312
    - master configuration 322
    - master node 317
    - queue 324
    - slave configuration 318
    - slave node 315
    - synchronous 311
  - code readability 276
  - Cognitive Science
    - Laboratory 409
  - collection 108
    - flattened 106
  - command-line 32
  - commit 142, 278
  - commodity hardware 311
  - commons-codec.jar 132
  - comparison 76–77, 95
  - Compass 312
  - components 107
  - composite
    - identifier 101
    - primary key (mapping) 101
  - compound file format 293
  - compress 80
  - concurrency 277
    - ACID 25
  - concurrent
    - access 287
    - threads 278
  - configuration 34, 68
  - ConnectionFactory 319
  - ConstantScoreRangeQuery 244
    - speed increase 244
  - constructor 192
  - @ContainedIn 110, 155, 279
  - contention 277
  - context 149
  - continuous script 126
  - contributions 418–427
  - conversation 143, 267
  - conversion 64, 76
    - structure 65
    - type 66
  - coord 365, 367, 373–375
    - calculation, changing 373

copy  
   asynchronous 322  
   operation 124  
 CopyDirectory 344  
   inner class 346  
 core 311  
 corpus 387, 394  
 correlated query 104  
 corruption (index) 311  
 count(\*) 185  
 crash 317  
 crawl 12–13  
 create read update delete. *See*  
   CRUD  
 createFullTextEntityManager  
   167  
 createFullTextQuery 168  
 createFullTextSession 167  
 createWeight 378  
 Crimson 430, 439  
 Criteria 161, 197  
 cross-cutting restriction 251  
 CRUD 71  
 cursor, forward only 378  
 custom  
   bridge 241–244, 438  
   query generation 224  
 Cutting, Doug 394, 405

**D**

---

dash 125  
 data  
   grid 313  
   mining 224  
   population process 306  
 database 8, 24, 285, 295  
   all-or-nothing 201  
   local vs remote 308  
   Lucene as a 25  
   performance 155  
   relational 22  
   row 71  
   version 308  
 Date 90  
 date 23, 77  
 @DateBridge 78, 209  
 DbUnit 305  
 de facto standard 430  
 decode 366, 379  
   loss of precision 366  
 default 117  
   Formatter class 404  
 DEFAULT\_ANALYZER 395  
 DEFAULT\_FIELD\_NAMES 396

DEFAULT\_MAX\_NUM\_  
   TOKENS\_PARSED 396  
 DEFAULT\_MAX\_QUERY\_  
   TERMS 395  
 DEFAULT\_MAX\_WORD\_  
   LENGTH 395  
 DEFAULT\_MIN\_DOC\_  
   FREQ 395  
 DEFAULT\_MIN\_TERM\_  
   FREQ 395  
 DEFAULT\_MIN\_WORD\_  
   LENGTH 395  
 DEFAULT\_STOP\_WORDS 396  
 DefaultHandler class 430  
   extending 434  
 DefaultSimilarity 364–378  
 delay 315  
 delimiters  
   fieldname/term 227  
   phrase query 227  
 denormalization 71, 105  
 dependency injection 30  
 depth 113  
   limit 112  
 Derby 36, 119, 305  
 df. *See* document frequency  
 Directory 117, 311  
   RAM-based 329  
   warning 329  
 directory provider 117  
   cluster 120  
   custom 124  
   database 312  
   filesystem 68, 118  
   in-memory 68, 119  
 directory\_provider 117  
 DirectoryProvider 117, 124, 300,  
   312, 329–338, 342–350  
   custom 344  
   sharing 329  
   use cases 344  
   writing 343  
 disableFullTextFilter 260  
 disjunction query 247  
 disk  
   file structures 292  
   space 290  
 distinct 280  
 distributed cache 318  
 DocIdBitSet 253, 270  
 DocIdSet 287  
 DOCUMENT 347  
 Document 64–65, 89, 99, 101,  
   150, 162, 164, 190, 283  
   id 190

document 13, 39, 105  
   feedback 388–393  
   frequency 356, 360–361, 394  
   global information 355  
   length normalizing 359  
   local information 355  
   lowering the score 405  
   node 435  
   normalization 366  
   traversal recursion 435  
   unit 214, 216  
   update 140  
   weight 362  
 Document (Lucene) 23  
 DOCUMENT\_ID 347  
   projection 190  
 Document.setBoost 248, 366  
 DocumentBuilder.CLASS\_  
   FIELDNAME, effect of 339  
 DocumentHandler 430  
 @DocumentId 39, 72  
 DOM 434  
   efficiency 438  
   hierarchical tree 435  
   model memory  
     requirements 439  
   parsing 434–438  
   system resource  
     utilization 439  
 domain model 22, 162  
 dot product 357–358  
 DoubleMetaphone 131  
 DVD store application 29

**E**

---

EAN 40  
 EAR 33  
 EasyMock 303  
 eBay 249  
 edit distance 231  
 efficiency 175  
 EHCACHE 271  
 EJB 144  
 EJB 3.0 167  
 elementNode 438  
 ElementType  
   FIELD 218  
   METHOD 218  
   TYPE 218  
 Elision 125  
 ElisionFilter 130  
 embedded objects 107  
 enableFullTextFilter 260  
 English 83

- Enterprise Archive. *See* EAR
  - entity filter 337
  - entity. *See* domain model
  - enumeration of terms 238
  - equals()/hashCode() 258
  - ERLANG 135
  - escapeSpecials 227
  - European Article Number.  
*See* EAN
  - event
    - characters 434
    - element end 430
    - element start 430
    - endElement 434
    - listener configuration 36, 140
    - startDocument 434
  - event model, disable 156
  - event-driven 430
  - Excel 425
  - exception 171
    - lock 312
  - execution order 244
  - explain 199, 367, 370, 372, 378, 381
    - method 366
    - query 55
  - Explanation 191, 198, 347, 368
    - fullTextQuery.explain 347
    - performance 347
  - explanation 374, 378, 381
    - first method 368
    - second method 370
  - EXPLANATION
    - (projection) 191
  - expressions, regular 226
  - Extension points 148
  - external change 157
- F**
- 
- @Factory 256
  - family of words. *See* stemming
  - feedback 309, 388, 395
  - fetch
    - associations 280
    - strategy 196, 284
  - FetchMode.SUBELECT 279
  - @Field 39, 75, 82
  - Field 64
  - field 39, 66
    - access 40, 75
    - bridge. *See* bridge
    - Lucene 214
    - normalization 372
    - setBoost 366
    - TermVector 389
    - TermVector values 389
  - field (Lucene) 23
  - @Field.index 79
  - @Field.store 80
  - fieldable 248
  - Fieldable.setBoost 248
  - @FieldBridge 91
  - FieldBridge 99, 434, 437
  - fieldNorm 368, 372
  - FieldSelectorResult 350
  - fieldWeight 368
  - files, number opened 289
  - filesystem
    - local 119
    - shared 123
    - shared vs local 308
  - filter 251, 269, 287, 364, 380
    - analyzer 126
    - apply 259
    - built-in 254
    - cache 256
    - cache size 257
    - caching 287
    - caching and parameters 258
    - caching instance 256
    - caching results 256
    - category 261, 288
    - chained filters 259
    - chaining 216
    - declare 255
    - disable caching 257
    - enable 259
    - examples 261
    - exclusion 263
    - external service 269
    - factory 255
    - Lucene 253
    - own caching system 271
    - parameter 257
    - performance 283
    - query based 262
    - range 264
    - search within search 267
    - security 261, 288
    - temporal 288
    - vs query 252
  - FilterCacheModeType 256
  - FilterCachingStrategy 257
  - FilterKey 258
  - firstResult() 185
  - flexibility 18
  - flush mode 143
  - flushing 142
  - flushToIndexes 154
  - Formatter class 404
    - injecting 404
    - method signature 403
  - forward only cursor 378
  - free-text search. *See* full-text search
  - Freider, Ophir 356
  - frequencies 361, 392
  - FSDirectoryProvider 68, 118
  - FSMasterDirectoryProvider 122
  - FSSlaveDirectoryProvider 123, 343–346
  - full-text queries 161
  - full-text search 12
    - Altavista 12
    - choosing a solution 21
    - efficient 16
    - flexibility 18
    - Google 12
    - main goal 12
    - management 17
    - Microsoft SQL Server 18
    - MySQL 18
    - Oracle DB 18
    - portability 18
    - scalability 18
    - Yahoo! 12
  - full-text search solution 17
    - dedicated box 19
    - library 20
    - relational engine 17
  - FullTextEntityManager 42, 166
    - build 166
  - FullTextEntityManager.close() 167
  - @FullTextFilterDef 255, 288
  - @FullTextFilterDefs 255
  - FullTextQuery 168, 171
  - FullTextQuery.EXPLANATION 370
  - FullTextSession 42, 166
    - build 166
    - obtaining SearchFactory 328
  - FullTextSession.close() 167
  - function package, warning 398
  - FuzzyQuery 130, 208–209, 237–240, 249
    - difference from proximity query 208
    - example 237
    - problem with 415
  - FuzzyTermEnum 238–240

**G**


---

garbage collection 313  
 get (method) 103  
 getDelegate() 198  
 getDirectoryProviders 329  
 getDirectoryProvidersForAll-Shards 301  
 getDirectoryProvidersFor-Deletion 300  
 getDocIdSet 254  
 getEnum(reader) 240  
 getFullTextEntityManager 42, 167  
 getFullTextSession 42, 166  
 getResultList() 176  
 getResultSize() 187  
 getSimilarity 379  
 getSingleResult 181  
 getter, access 40, 75  
 GigaSpace 122, 312, 318  
   Lucene 124  
 Gilleland, Michael 208  
 global analyzer 220, 223  
 good citizen rules 342  
   consequences of not following 343  
 Google 12, 16, 183, 249, 275, 295, 386  
   I'm Feeling Lucky 181  
   PageRank 12  
   Search Appliance 19  
 Gospodnetić, Otis 45  
 Grails 30  
 Grossman, David 356  
 guaranteed delivery 316  
 Guice 30

**H**


---

H2 36, 119, 305  
 hard drives 297  
 Harold, Elliot Rusty 439  
 hash 297  
 Hatcher, Eric 45  
 hbm.xml 30  
   mapping file 41  
 HDGF. *See* POI project  
 heterogeneous results 171  
 heuristic variables, defaults 395  
 heuristics 394–395  
 Hibernate  
   Annotations 30  
   Annotations or Core? 36  
   Core 30  
   EntityManager 30

Hibernate Annotations 140  
 \_hibernate\_class 74, 173, 339–340  
 <\_hibernate\_class> 341  
 Hibernate Core 29  
 Hibernate EntityManager 140  
 Hibernate Query Language. *See* HQL  
 Hibernate Search 3, 29  
   configuration-by-exception 34  
   DoubleMetaphone 131  
   download 31  
   executing a query 47  
   guide to annotations 441  
   listens to Core 42  
   Metaphone 131  
   property 214  
   providing properties 117  
   query data 43  
   Refixed Soundex 131  
   requirements 30  
   setting up 31  
   Soundex 131  
 hibernate.cfg.xml 140  
   file 34  
 hibernate.search 117, 291  
 hibernate.search.analyzer 83  
 hibernate.search.default.index-Base 343  
 hibernate.search.filter.cache\_bit\_results.size 257  
 hibernate.search.filter.cache\_strategy.size 257  
 hibernate.search.indexing\_strategy 156  
 hibernate.search.similarity 370  
 hibernate.search.worker.batch\_size 154  
 hibernate.search.worker.jndi.class 320  
 hibernate.search.worker.jndi.url 320  
 hibernate.worker.buffer\_queue.max 146  
 hibernate.worker.execution 146  
 hibernate.worker.jndi 147  
 hibernate.worker.thread\_pool.size 146  
 hibernate-search.jar 31  
 hibQuery.scroll() 348  
 hierarchical tree, DOM 435  
 Highlighter 400–404  
 highlighting 81  
 HPSF. *See* POI project

HQL 29, 47–48, 105–106, 161, 163  
   query 43  
 HSLF. *See* POI project  
 HSQLDB 36, 119, 305  
 HSSF. *See* POI project  
 HTML 404  
 HTTP request 322  
 HTTPSsession 267  
 hydrating objects 165, 188  
 hyphenation 125

**I**


---

I'm Feeling Lucky 181  
 ID 347  
   projection 190  
 identifier property 89, 95, 104  
   bridge 91  
 idf 339, 356, 360, 365, 368  
   low term frequency 394  
 ignorable whitespace 438  
 ILS 429  
 import.sql 305  
 incremental copy 316  
 index 10  
   accent 130  
   crawl 19  
   directory 34  
   disposability 346  
   distributed 122  
   in-memory 36  
   master 343, 346  
   merging 329, 337  
   method 151  
   model 23, 104  
   model vs. object model 64  
   optimize structure 288  
   out of date 166  
   PDF 20  
   problems with size 206  
   root directory 118  
   scan 11  
   seek 10  
   sharded 118  
   source copy 346  
   splitting 294  
   statistics 51  
   storage  
     recommendations 346  
   structure 73  
   synchronizing 17  
   where to store 117  
   Word document 20  
 Index.NO 80

- Index.NO\_NORMS 80
  - Index.TOKENIZED. *See* TOKENIZED
  - Index.UN\_TOKENIZED. *See* UN\_TOKENIZED
  - index() 280
  - indexBase 34, 68, 118, 337, 345–346
  - @Indexed 39, 68
    - index property 337
    - when to use 71
  - indexed collections 109
  - @IndexedEmbedded 107, 155, 279
    - performance 277
  - indexing 13, 115
    - architecture 148
    - asynchronous 145, 278
    - concurrent 277
    - disabling transparent indexing 156
    - heavy writing 145
    - information 202
    - initial 151
    - initial indexing 153
    - JMS 146, 279
    - manual 42
    - mass indexing 153, 279
    - performance 155, 276–277
    - properties 39
    - remove old content 281
    - size 276
    - slow 155, 276
    - speeding performance 67
    - strategy 67
    - synchronous 144
    - third party changes 157
    - time 109, 276
    - transparent 139
    - troubleshooting 73
    - tuning operations 292
  - indexName 118
  - IndexReader 162, 253, 256, 286
    - caching 342
    - closing 332
    - cost of opening 342
    - explicitly closing 342
    - not closing 342
    - outside of the Hibernate Search framework 341
    - retrieving instance of 342
    - rules of use 343
    - sharing 342
    - warm up 286
  - IndexSearcher 162
  - IndexShardingStrategy 297
  - IndexWriter 292
  - information indexing 202
  - Ingersoll, Grant 388
  - inheritance 64
  - Initial indexing 153
  - InitialContext 320
  - injection 167
  - in-memory 318
    - database 305
  - InputSource 434, 437
  - INSTANCE\_AND\_DOCIDSET-RESULTS 256
  - INSTANCE\_ONLY 257
  - Integrated Library System. *See* ILS
  - integration testing 305
  - interface 108
  - International Standard Book Number. *See* ISBN
  - inverse document frequency. *See* idf
  - ISBN 7
  - ISOLatin1AccentFilterFactory 130
  - iterate() 177
- J**
- 
- jakarta-regexp.jar 412
  - JakartaRegexCapabilities 414
  - Japanese 126
  - JAR 31
  - Java 3, 8, 135, 157, 208
  - Java 5 30
  - Java Archive. *See* JAR
  - Java Compiler Compiler. *See* JavaCC
  - Java Development Kit. *See* JDK
  - Java EE 30, 33, 318, 323
  - Java Message Service. *See* JMS
  - Java Naming Directory Interface. *See* JNDI
  - Java Network Launching Protocol. *See* JNLP
  - Java Persistence 68
  - Java Persistence Query Language. *See* JPA-QL
  - Java Persistence specification 40
  - Java Runtime Environment. *See* JRE
  - Java SE 30, 320
  - Java WebStart JNLP, download Luke 48
  - JavaCC 212
  - JavaUtilsRegexCapabilities 414
  - JBoss 144
  - JBoss AS 74, 148, 318
  - JBoss Cache 122, 124, 271, 312–313, 318
  - JBoss Maven 31–32
    - repository 126
  - JBoss Seam 30, 143–144, 167, 267, 367
  - JBossCacheSearchable 313
  - jboss-service.xml 319
  - JDBC 71, 117, 144, 329, 378
    - author recommendations 329
    - driver 71
    - RowSet 378
  - JDK 30
  - JGroups 314
  - JMeter 308
  - JMS 146, 157, 316–317
    - backend 318
    - indexing 279
    - queue 319
  - JMSBackendQueueProcessorFactory 150
  - JMSqueue 316
  - JNDI 147, 319
  - JNLP 58
  - join 105
  - JOIN (SQL). *See* relationship
  - JPA-QL 47, 163
  - JRC-ACQUIS Multilingual Parallel Corpus 387
  - JRE 30
  - JUnit 305–306
- K**
- 
- keyword spamming 361–363
    - counting terms 363
    - simulating 362
  - King, Gavin 175
  - Knuth, Donald 276
- L**
- 
- languages 125
    - European 128
    - Greek 128
    - Russian 128
  - large term count effects, minimizing 361
  - Latin 126
  - lazy loading 25, 44
    - association 162
    - transparent 163

- LazyInitializationException 197
  - leading zeros 95
  - legal 295
  - lengthNorm 366, 369, 372
    - formula 366
    - ignoring 372
    - overriding rules 372
  - Levenshtein Distance 130
    - algorithm 208
  - lifecycle 110
  - lightweight 30
  - Linux 288–289
  - list() 176
  - Litchfield, Ben 418
  - local
    - copy 315
    - copy index 121
    - filesystem directory
      - provider 118
      - queue 320
    - lock 278, 295
    - file 311
      - global pessimistic 120, 146
    - log files, data mining 224
    - log search 308
    - look behind, negative 228
    - Lovins, Julie Beth 135
      - stemming algorithm 135
    - lowercase 128
    - LowerCaseFilter 128
    - LowerCaseFilter 223
    - Lucene 20–21, 29, 144
      - analysis package 216
      - analyzer 45
      - blob 346
      - clauses limit 283
      - code changes often 380
      - Contrib 131
      - Directory 68
      - directory 117
      - download 31
      - field 76, 214
      - filter 253
      - index 43
      - mailing list 353
      - native APIs 101
      - query 162
      - query parser syntax 54
      - Sandbox 400
      - scoring formula 364
      - sharing a directory 69
      - stop work list 129
      - testing a query 54
    - Lucene query 44
      - preparation 164
      - writing 44
  - LuceneBackendQueueProcessor 150
  - lucene-benchmark-javadoc.jar 388
  - LuceneDictionary 415
  - lucene-highlighter.jar 400
  - LucenePDFDocument
    - class 421–425
    - example of 422
    - output of 424
    - utilizing 422
  - LucenePDFDocument, contents
    - field 424
  - lucene-queries.jar 405
  - lucene-regex.jar 412
  - lucene-spellchecker.jar 415
  - lucene-wordnet.jar 409
  - Luke 48, 73, 118, 198, 202, 340, 418
    - analyzer 55
    - browse index documents 52
    - classpath 48
    - Document tab 51
    - download 48
    - explain 55
    - index size 55
    - index statistics 51
    - open an index 50
    - Overview tab 50
    - plugins 57
    - query 54
    - Search tab 54
    - term list 51
    - Tokenized tab 53
    - unlock an index 50
- M**
- 
- mailing list 353
  - maintenance 295
  - managed
    - entity 152
    - environment 323
    - objects 26, 43, 163, 188
  - manual 156
    - indexing 42
  - Map 90, 99
  - MapFieldSelector 350
  - mapping 38, 63
    - abstract class 71
    - analyzer 83
    - annotations 38
    - array 109
    - associated objects 110
    - boost factor 85
  - class hierarchy 69
  - collection 108
  - composite primary key 72, 101
  - custom bridge 89
  - depth 112
  - directory name 68
  - embedded objects 107
  - entities 67
  - identity property 71
  - indexed entity 68
  - indexing strategy 78
  - nested associations 112
  - primary key 71
  - property 75
  - same property multiple times 82
  - share Lucene directories 68
  - store 80
  - subclass 69
  - tokenize 79
  - master 146
    - node 122, 314, 317
  - MatchAllDocsQuery 246, 263
  - max\_field\_length 294
  - max\_merge\_docs 292
  - maxGramSize 131
  - maxResults() 185
  - MDB 317
  - memory 280, 292
    - usage 178
  - merge 292
    - segments 292
  - merge\_factor 292
  - MergedFurniture 338–339
  - Merging, benefit of 337
  - message persistence 316
  - @MessageDriven 323
  - message-driven bean 317, 322
  - MessageListener 323
  - metadata 361, 420–422
    - index 190
  - META-INF/persistence.xml
    - file 34
  - Metaphone 131
  - MFC property sets, reading 425
  - Microsoft
    - Excel 425
    - file formats 425
    - PowerPoint 425
    - SQL Server 18
    - Visio 425
    - Word 90, 425
  - Middle Ages 126
  - minGramSize 131

minimumSimilarity 208, 238  
 default value 238  
 definition 209  
 range of values 208  
 warning 209  
 mismatch 22, 161  
 retrieval 25  
 structural 23  
 synchronization 24  
 misspellings 415  
 help with 412  
 MMapDirectory 50  
 mocking 303  
 models, index vs. object 64  
 modulo 297  
 MoreLikeThis 393, 395–398  
 default values 395  
 heuristic methods 395  
 warning 394  
 MoreLikeThis.setboost 249  
 MultiFieldQueryParser 46, 228–231  
 example 229  
 static parse methods 228  
 multistage search engine 187  
 multiterm 374  
 query 365, 372  
 MultiTermQuery class 240  
 MySQL 18  
 blob 312

---

## N

n+1 loading issue 178  
 problem 284  
 query problem 279  
 naming conventions 107  
 navigation 105  
 nbr\_of\_shards,  
 configuration 334  
 negative look behind 228  
 nested associations 112  
 network 276  
 performance 155  
 topology 318  
 network filesystem. *See* NFS  
 next(Token token) 223  
 NFS 119, 311, 314, 317  
 n-gram 209, 412  
 algorithm 130  
 NGramTokenFilter 131  
 NGramTokenFilterFactory 131  
 NIO 50  
 n-node, problem 345  
 NO\_NORMS 372

no-arg constructor 255  
 node 121  
 master 121–122  
 processing 435  
 slave 121  
 noise 5  
 word 128  
 nonblocking I/O. *See* NIO  
 NONE (FilterCacheMode-  
 Type) 257  
 non-sharded 330, 335  
 NoResultException 183  
 norm 365, 372  
 normalization 104, 364, 369,  
 372, 379  
 preventing 372  
 normalize 381  
 normalizing document  
 length 359  
 <not available> 340, 422  
 null, special marker 66  
 number 23, 90  
 of files opened 289  
 of operations 291  
 of transactions 291  
 numeric fields 242  
 numSug 416

---

## O

object  
 graph 106, 284  
 identifier 190  
 model 104  
 total number of 164  
 object model  
 Java vs. Lucene 64  
 vs. index model 64  
 object-oriented paradigm 46  
 object-relational mapper.  
*See* ORM  
 objectToString 94  
 occurrence, number of 81  
 offline indexes 291  
 offset 81  
 OLE 425  
 OLE 2 Compound Document,  
 reading and writing 425  
 OpenBitSet 253  
 optimization  
 premature 276  
 query 162  
 optimize 174  
 premature 282  
 optimize() 281, 289

optimize(Class) 289  
 optimizer.operation\_limit.max  
 291  
 optimizer.transaction\_limit.max  
 291  
 optimizing 153  
 automatic 291  
 index structure 288, 292  
 index structure benefits 288  
 index structure manual 289  
 index structure, limit on open  
 files 288  
 index structure, need for 290  
 queries 282  
 Oracle DB 18  
 XML Parser for Java 430, 439  
 order 177, 267  
 of execution 211  
*See also* sort  
 org.hibernate.search.worker.  
 scope 149  
 ORM 22, 29  
 OutOfMemoryException 120,  
 146–147, 152–153, 180–181,  
 278, 280, 294, 316, 322  
 override 364, 372  
 coord factor 374  
 DefaultSimilarity 369

---

## P

padding 91, 210, 241  
 numbers 94–95  
 PadNumberBridge 242–244  
 PadNumbers, example 242  
 PageRank 12  
 pagination 183, 347  
 performance 283  
 user limit 183  
 @Parameter 127  
 ParameterizedBridge 97, 242  
 parameters 97  
 parent node 438  
 ParseException,  
 RangeQuery 210  
 parsing 202  
 partitioning, legal 295  
 PDDocument 420  
 PDF 90  
 from custom bridge 438  
 PDF metadata 421  
 contents field 424  
 field listing 422  
 inserting 421  
 PDFBox 418, 421  
 URL 418

PDFTextExtractor 421  
 PDFTextStripper class 419–420  
 PDFTextStripper, example  
   of 419  
 performance 67, 141, 145, 276  
   degradation 206  
   goal 276  
   hydrating objects 165  
   testing 308  
 Perl 135  
 persistence context 43–44, 89,  
   164, 190  
   loading matching objects 165  
 persistence unit 322  
 pessimistic lock 119, 145, 311,  
   314  
   scalability 346  
 Phillips, Lawrence 131  
 phonetic approximation 11, 131  
 PhoneticFilterFactory 132  
 PhraseQuery 207, 231–234  
   example 231  
 plain old Java object. *See* POJO  
 Plain Old Text File. *See* POTF  
 PlainTextDictionary 415  
 POI project 425  
   contributing to 425  
   mailing lists 425  
 POI, WordExtractor class 426  
 poi-FINAL.jar 426  
 POIFS. *See* POI project  
 poi-scratchpad.jar 426  
 POJO 38  
 polymorphism 47, 64, 69, 71,  
   173  
 POM 32  
 portability 18  
 Porter, Michael 135  
   stemming algorithm 135  
 position 81  
 POTF 427–428  
   dependencies 427  
 PowerPoint 425  
 Precision 387  
   defined 387  
   denominator 138  
 precision 386  
   and recall 386  
 prefix 107  
 PrefixQuery 207, 236–237  
   example 236  
 premature optimization 276,  
   282, 303  
 Princeton University 409  
 problem, n+1 284

Processing XML 429–439  
 projected property 95  
 projection 40, 80, 188, 347, 367,  
   370  
   in code 348  
   performance 284  
   performance  
     improvements 350  
   queries 90  
 ProjectionConstants 347, 349  
   validity of 349  
 Prolog 409  
 property  
   Hibernate Search 214  
   visibility 75  
 proximity searches 231  
 ProximityQuery 207  
   difference from fuzzy  
   query 208  
 ProximitySearch 207  
 pseudo relevance, feedback 388  
 punctuation 128  
 purge (method) 152  
 purgeAll() 152, 281

## Q

quality of service 145  
 Query 47  
 query  
   across multiple fields 228  
   ad hoc 212, 386  
   Boolean 283  
   boosting 375–378  
   building 166  
   correlated 89, 104  
   escape special characters 226  
   explaining 191  
   explanation 198  
   full-text 161  
   fuzzy 208  
   generating custom 224  
   grouping expressions 212  
   Hibernate Search 46, 163  
   HQL vs. Hibernate Search 48  
   improve readability 211  
   interface 164  
   JPA-QL query vs. Hibernate  
     Search 48  
   Lucene 44  
   mimicry 163  
   more like this 81  
   multiple analyzers 221  
   multiple terms 372  
   n+1 problem 279  
   negative 263  
   n-gram 209  
   normalizing 360  
   optimizing 282  
   order of execution 211  
   pagination 183  
   parser 44  
   phrase 207, 231  
   polymorphic. *See* polymor-  
     phism  
   production 90  
   programmatically API 15  
   proximity 207  
   range 209  
   receiving meaningful  
     answers 184  
   relevance 359  
   semantic 163  
   size reduction 361  
   slow 188  
   Soundex 209  
   special characters 226  
   string-based 15  
   term 225  
   text-based language 15  
   tokens 412  
   understanding results 198  
   wildcard 206  
 Query object 168  
 query results, demoting 404  
 Query.createWeight 378  
 Query.setboost 249  
 Query.toString 224  
 query-helper classes 409  
 Querying 43  
 querying 161  
   database and index  
     desynchronization 166  
   execution 175  
   fetch size 179  
   iterator 177  
   list 176  
   managed objects 164  
   number of results 186  
   object loading strategy 174  
   performance 282  
   persistence context 164  
   projection 188  
   restricting entities 171  
   result structure 191  
   returning multiple types 172–  
     173  
   scrollable resultset 178  
   single result 181

querying (*continued*)  
   sort 194  
   speed 183  
   subclass 173  
   transforming results 191  
 queryNorm 365, 377, 381  
 QueryParser 206, 209–214, 226–228, 249  
   drawback 214  
   escape 214  
   setDefaultOperator 214  
   syntax 202–214  
   warning 214  
 QueryParser, Operator, default value 214  
 QueryParser.jj 212  
 QueryWrapperFilter 254  
 queue 324  
   change 142  
   limit 278  
 QueueingProcessor 150

---

**R**  
 RAID 345  
 RAM 292  
 ram\_buffer\_size 292  
 RAMDirectoryProvider 68, 120  
 random file access 165  
 range 264  
   lexicographic 210  
 ranged query 77  
 RangeFilter 254, 266  
   vs RangeQuery 264  
 RangeQuery 78, 209–210, 240–244, 249, 264, 294  
   bracket types 209  
   definition 209  
   problem with 210  
   problem with numerics 240  
   with Strings 210  
 raw score 375, 379, 409  
 ReaderProvider 286, 342  
   closing reader 342  
 ReaderStrategy 286  
   configuration property 342  
   default value 342  
 read-mostly 277  
 Recall 387  
   defined 386  
   denominator 138  
 recursion 434  
 RefinedSoundex 131  
 refresh 122

regex 412  
   queries 399  
   query 412–415  
   warning 412  
 RegexQuery, example of 413  
 RegexTermEnum 240  
 regular expressions 226  
 rehydrated 89  
 reindexing 297  
 relational  
   database 277  
   model 104  
 relationship 23, 104  
   performance 277  
 relevance 4–5, 12, 14, 16, 80, 138, 194, 215, 267, 364, 386–398  
   accuracy requirements 387  
   effectiveness  
     measurements 386  
   feedback 388, 395  
   increasing 388  
   size of repository 387  
   subjectiveness 386  
   theory 386  
 relevant document  
   frequency 388  
 repository 354–356, 359, 361, 368, 386–387, 394  
   standard document 387  
 Representational State Transfer.  
   *See* REST  
 resource consumption 146  
 response time 145, 293  
 REST 15  
 restriction, cross-cutting 252  
 result set, scrollable 280  
 results per page 183  
 ResultTransform 192  
 ResultTransformer 286  
 rewrite 225  
 Rhino JavaScript 48  
 rollback 142, 317  
 root 134  
 rounding 91  
 rsync 124  
 Ruby 135, 157

---

**S**

SaaS 295  
 Saenger, Paul 126  
 Salton 355  
 SAN 119, 345  
   directory provider 119  
   typical configuration 345

sandbox 393, 399  
   contents of 400  
   maintainability 400  
 SAX 428  
   parsers 430–434  
 SaxExampleBridge class 432  
 scalability 18, 121, 142, 310, 316  
 ScopedAnalyzer 220–223  
 ScopedEntity 218, 221  
 SCORE 347  
   projection 190  
 score 80, 196, 198, 251  
   definition 354  
   normalized 369  
   normalizing 375  
   raw 369  
 Scorer 354, 377–385  
   similarity to JDBC 378  
 scoring 16  
   formula 364  
 script, continuous 126  
 scroll 178  
 scrollable result set 154, 280  
 ScrollableResults 178  
 Seam. *See* JBoss Seam  
 search  
   all-or-nothing database 201  
   approximation 44, 130  
   based on business needs 21  
   boost factor 46  
   by word 9  
   category 5  
   choosing a strategy 8  
   crawl 12–13  
   detailed screen 5  
   edit distance 231  
   false positive 131  
   full-text 12  
   full-text and relational  
     engine 17  
   full-text solutions 17  
   fuzzy 130  
   index scan 11  
   index seek 10  
   indexing 13  
   in layers 139  
   n-gram algorithm 130  
   noise 9  
   not-shared 286  
   phonetic approximation 131  
   precision 131  
   proximity 207, 231  
   relevance 9  
   root 134  
   score 354

- search (*continued*)
  - scoring 16
  - shared 286
  - simple interface is key 6
  - sort by relevance 12
  - special characters 205
  - stemming 14
  - synonyms 11, 14, 133
  - table scan 10
  - text box 7
  - trigram 130
  - words with same root,
    - meaning 11
  - words, not columns 9
- Search (class) 42, 166
- Search Assist 268
- search engines
  - multistage 187
  - pitfalls 8
- search within search 267
- search.function package
  - warning 398
- Searchable Interface, abstract
  - methods 380
- Searcher.explain 368
- SearchException 171
- SearchFactory, gateway to
  - Lucene 328
- SearchFactoryImpl 328
- SearchFactoryImplementor
  - 328, 343
- searching 13
  - multistep process 15
  - slow 276
  - tips 275
- second-level cache 155
- security 258, 295
  - filter 288
- segment 288, 292
- Selenium 308
- Serial API for XML. *See* SAX
- server
  - cluster 346
  - topology 308
- Service Level Agreement 308
- SessionFactory 68
- set (method) 100
- setAllowLeadingWildcard 206, 214
  - default value 214
- setBoost 338
- setCriteriaQuery 197
- setFetchSize 179
- setFilter 261
- setFirstResult 184, 283
- setMaxClauseCount 234
  - default value 234
- setMaxResults 184, 283
- setOmitNorms 372
- setParameterValues 98
- setProjection 189, 285
- setResultTransformer 192
- setter 192
- setUp 307
- shard 329, 334–335
- sharding 294, 313
  - add a new shard 302
  - adding shards over time 301
  - complete reindexing 302
  - configuration 118, 296
  - drawback 301
  - number of 296
  - payload 300
  - strategy 297
- sharding\_strategy.nbr\_of\_shards
  - 296
- shared
  - drive 321
  - location 315
  - queue 314
- shared IndexReader 342
- Similarity 354, 364
  - decode 366
  - encodeNorm 366
  - saving work 379
- similarity coefficient 355, 358, 361
- SimilarityDelegator 364
- SimpleHTMLFormatter 403
  - posttag 403
  - pretag 403
  - three ways to override 404
- single
  - point of failure 317
  - result 181
- slave 146
  - optimizing (index
    - structure) 289
- slave node 123, 314–315
- slop distance. *See* slop factor
- slop factor 207, 231–234, 414
  - default value 231
- smart copy 316
- SNOBOL 135
- Snowball 135
- soft reference 257, 271
- Solr 126, 131
  - classpath 31
  - solr-common.jar 126
  - solr-core.jar 126
- sort 77, 79, 194
  - document id 196
  - Lucene 194
  - numeric value 196
  - score 196
  - string 196
- SortField 194
- Soundex 131
  - query 209
- source 122
- sourceBase 122, 325, 343–346
- SpanNearQuery 414
- SpanQuery 214
- SpanRegexQuery 414
  - example of 413
- SpanWeight 380
- special characters 205, 226
  - escape 226
- spellchecker 412, 415–418
  - methods 415
  - suggestions minimum 416
- SpellChecker class 415
- split indexes 294
- splitting text 125
- Spring 144, 167
  - Framework 30
- SQL 8–9, 15, 17–18, 161
  - HQL extension 29
  - performance 10
- stage 187
- stale data 142
- standard document
  - repositories 387
- StandardAnalyzer 83, 128–129, 216, 223, 414
- StandardFilter 128
- StandardFilterKey 258–259
- StandardTokenizer 128
- StandardTokenizer class,
  - javadoc 219
- static fetching strategy 175
- statistics 81, 280
- stemming 11, 14, 134
  - performance 277
- stop word lists, analyzer effect
  - on 415
- stop words 215, 309, 359, 395
  - default set 396
  - examples 215
- StopFilter 128
- storage area network. *See* SAN
- store 101, 188
  - performance 277
  - property 40
- Store.COMPRESS 80, 277

Store.YES 40, 80  
 stored properties 285  
 stream of token 125  
 stress testing 308  
 StringBridge 93  
 stringToObject 96  
 strong typing 163  
 Structured Query Language.  
   *See* SQL  
 subclass 65, 173  
 sumOfSquaredWeights 365, 381  
 surrogate key 103  
 Swing 30  
 sync 146  
 synchronized arrays 228  
 synchronous 277  
   clustering 311  
 synonym 11, 14, 133, 409–412  
   dictionary 133  
   generation 399  
   share reference 133  
   testing 410  
 synonym, index building 412  
 SynonymHelper 412  
 Syns2Index 409  
 synthetic flag 261

## T

---

table  
   information spread across  
     several 9  
   row 40  
   scan 10  
 @Target 218  
 targeted classes 282  
 targetElement 108  
 TDirectory 344  
 tearDown 307  
 temporal filter 288  
 Term 226  
 term  
   boost 227  
   boost problem 365  
   buffer 220  
   count normalization 364  
   weight 356–357  
 term frequency 355–356, 359,  
   365, 367, 369–372, 394  
   changing 369  
   pairs 389  
   storing 389  
 term vectors 389  
   enabling 389  
 term\_index\_interval 294

termEnum.term().text() 240  
 TermFreqVector 390, 392  
 TermPositionVector 392  
 TermQuery 225–226, 249, 264,  
   365, 367, 381  
   extending 382  
 TermScorer 378–379, 381  
 TermVector 81  
 TermVector.NO 81  
 TermVector.WITH\_OFFSETS  
   81  
 TermVector.WITH\_POSITION  
   81  
 TermVector.WITH\_POSITION\_  
   OFFSETS 81  
 TermVector.YES 81  
 Terracotta 122, 124, 312, 318  
 test 36  
 test data, DOM 439  
 testing 36, 119, 303  
   in-memory 305  
   integration 305  
 TestNG 306  
 testScopedAnalyzerAPI unit  
   test 223  
 Text Retrieval Conference.  
   *See* TREC  
 text, splitting 125  
 textNode 438  
 tf. *See* term frequency  
 third party contributions 399  
 THIS 347  
   projection 190  
 thoughtput 293  
 thread 278  
   local variable 168  
 thread-safety 98, 214  
 tilde 44, 227  
   in a fuzzy query 208  
   in a proximity query 207  
 token 125  
 @TokenFilterDef 127  
 TokenFilterFactory 127  
 tokenization 40, 215  
   example 216  
   performance 277  
 tokenize 79  
 TOKENIZED 79  
 @TokenizerDef 127  
 TokenizerFactory 127  
 tokens 215, 412  
 TokenStream 220–222, 402  
 tools, troubleshooting 48  
 TooManyClauses 254, 264, 283  
   exception 234

TopDocs 380  
 TopFieldDocs 380  
 transaction 142, 153, 281  
   mode 293  
 transaction incentive 346  
 transactional support 316  
 TransactionalWorker 149  
 transformTuple 192  
 transparent  
   fetching 44  
   lazy loading 163  
 TREC 386–387  
 trec\_eval\_latest.tar.gz 387  
 TriggerTask 344–346  
 trigram 130  
 troubleshooting 48  
 troubleshooting tool  
   Luke 202  
   Query.toString() 224  
 tuning index  
   operations 292  
   structures 292  
 two-phase commit. *See* 2PC  
 two-way bridge. *See* bridge, two-  
   way  
 TwoWayStringBridge 95  
 type 23, 64, 66  
 type-safe 163, 172  
 typo 11, 130

## U

---

ulimit 289  
 UN\_TOKENIZED 40, 79  
 unicity 44  
 unique identifier 79  
 uniqueResult 181  
 unit testing. *See* testing  
 Universal Product Code.  
   *See* UPC  
 UPC 40  
 URL 90, 125, 157  
 users 22, 184  
   acceptance test 308  
   test 308

## V

---

value  
   padding 91  
   rounding 91  
 vararg 172  
 vector coordinates 357  
 vector lengths, product of 358

vector space model 354–359,  
364–365  
pure 364  
view 71  
mapping 71  
objects 191  
virtual machine. *See* VM  
Visio 425  
Visual Basic 208  
VM 146  
vowel 125

## W

---

WAR 32  
warm up 162  
weak reference 271  
WeakHashMap 254, 271  
web application 30  
Web Archive. *See* WAR  
Web Beans 30  
Weight 354, 377–384  
as inner class 380  
weight 360, 378  
Weight Interface, method  
signatures 381  
WhitespaceAnalyzer 223

WildcardQuery 205–207, 234–  
236, 249, 294  
example 234  
wildcards 206, 234  
examples 206  
warning 206  
WildcardTermEnum 240  
window size, determining 181  
Windows XP 425  
WITH\_OFFSETS. *See* Field.  
TermVector  
WITH\_POSITIONS\_OFFSETS.  
*See* Field.TermVector  
WITH\_POSITIONS.  
*See* Field.TermVector  
wn\_s.pl 409  
WNprolog-3.0.tar.gz 409  
Word 90, 425  
word  
accentuation 126  
case 126  
common 126, 128  
noise 128  
origin 126  
reference 133  
separator 125  
Word extractor, from custom  
bridge 438

WordExtractor 427  
WordNet 409  
homepage 409  
jar 409  
lexical database 409  
preliminary steps 409  
website 410  
Worker 149  
write (index) 314  
write-intensive 277

## X

---

XA resource 143  
Xerces-J 430, 439  
XML 67  
XML documents, size  
concerns 439  
XML parsing  
memory usage 439  
Pros and Cons ??–439  
pros and cons 438

## Y

---

Yahoo! 12  
Yahoo! Search 268

