INDEX

Symbols

- (minus) operator, 8, 40
<< (node precedes) operator, 45
>> (node follows) operator, 45
$fs:dot, 289–292
$fs:last, 289–292
$fs:position, 289–292
$fs:sequence, 289–292
(: :) (begin/end comments), 8
* (multiply) operator, 40
* (wildcard) operator, 14, 221
' ' (apostrophe), delimiting string literals, 8
. (decimal), 8
" " (quotation marks), delimiting string literals, 8
// (double slash), 10, 13
/ (slash), 10, 99, 288
[ ] (brackets), 10–11
{ } (curly brackets), 15, 93, 123
+ (plus) operator. *See* plus (+) operator
= (equals) operator, 43–45

A

Abstract syntax, 241, 381, 443
Access control
  configuring, 410
  DBMSs and, 415
Accessor functions, built-in to XQuery, 49
Analysis-time. *See* Query-analysis phase
Annotated XSD Schema language, 331–334, 340
Anonymous types
  defined, 443
  use of, 69
  XML Schema, 117
APIs (application programming interfaces), 415
apostrophe (' '), delimiting string literals, 8
Application programming interfaces (APIs), 415
Arithmetic operators, 40–41
  atomization and, 113
  flexibility of, 297–298
  kinds of, 40
  normalizing, 295–296
  untyped data, 41
  XPath and, 105
At (positional variable), 28–30
Atomic types
  conversion rules, 48
  defined, 443
  referencing, 119
  XQuery data model, 90
Atomic values
  comparison operators and, 41–43
  defined, 443
  formal semantics, 267–269
  items and, 90, 207
  matching to types, 248–250
  XQuery data model, 6
Atomization
  comparison operators and, 41–43, 298
  conversion of arguments, 113, 217
  defined, 443
  value extraction and, 40, 104
  as XPath implicit coercion, 294–296
Attributes
  defined, 443
  document order of, 7
  importing types and, 66
  node attributes, 208
Auction use case
  converting input (XML) to output (XHTML), 196–197

debugging, 197–199
input data (in XML), 190–192
output data (in XHTML), 194–196
static typing, 201–203
validation, 199–201
XML Schema, 192–194
Audit logs, 355
avg( ), 56
Axes
reverse, 290–291
XPath steps, 150
XSLT vs. XQuery, 166–168

B

Basic validation, 368
Basic XQuery
defined, 444
static typing and, 129
type conformance and, 54
Begin/end comments ((: :)), 8
Bibliography, data sample, 5–6
Binary large objects. *See* BLOB (binary large objects)
Binding languages to databases, 421–422
BLOB (binary large objects)
data storage and, 356
defined, 444
XML physical models and, 361
Boolean expressions
branches of, 138
Effective Boolean Value and, 103
where clause and, 22
Boolean types, 168, 268
Boundary whitespace, 17
Brackets ([ ]), 10–11
Branches
boolean values, 138
conditional expressions, 218
Built-in functions, 47–49
accessor types, 49
numeric types, 48
SQL types, 47–48
Built-in types
constructors and, 70
function signatures and, 55
list of, 61
sequence types, 61–62
XML Schema, 96

C

Canonicalized XML, 361, 444
Cartesian cross products, 22, 33, 371
Case-sensitivity, 93, 444
CASE statement, 72
CDATA, 109
Character large objects. *See* CLOBs (character large objects)
Choice types, 284
Clark, James, 156, 175
Client APIs, XQuery, 414–420
CLOBs (character large objects)
data storage and, 316, 356
defined, 444
XML physical models, 360–361
Closure
defined, 444
principle of, 87
property, 127
CLR (Common Language Runtime), 362, 444
Code, internal efficiency, 174–175
Collations, of XML data, 358, 365–366, 444
collection( ), 9–10, 92, 386–387
Collections
configuring, 407
defined, 444
native XML databases, 413–414
overview of, 399
Columns, relational data model, 112
Command execution, DBMS, 415
Comment nodes, 7, 208
Comments, 8–9
Common Language Runtime (CLR), 362, 444
Comparison operators, 41–45
atomization and, 113, 295
equals operator, 43–45
error handling and, 137
general comparison operators, 42–43
node comparisons, 45
transitivity property, 104–105
value comparison, 41–42

value comparison operators, 41
XPath and, 106
Complex types
defined, 444
defining and naming, 96
formal semantics, 271
Composed representation
application types supported, 319
composition techniques and, 318, 320–325
LOB representation and, 316–317, 319
overview of, 317
querying/transforming XML documents, 349
shredding and, 318–319
Composition techniques
annotated XML templates, 331–334
concepts, 320–325
default mapping, 325–328
defined, 444
shredding and, 336
SQL/XML, 328–331
Compositionality
defined, 445
principle of, 86
Computed constructors
alternative syntax to XML constructors, 123
element construction with, 226–227
functionality of, 16–17
Conclusions, 242, 445
Concrete syntax, 242, 445
Conditional expressions
branches of, 218
overview of, 38–39
promotion rules, 218–219
static typing, 258
Configuration, native XML databases, 406–410
access control, 410
collections, 407
databases, 407
indexes, 407–409
locking, 410
schemas, 410
stored functions, 409–410
triggers, 409–410
Conjunction rules, evaluation order, 263–264
Connections, DBMS access, 415
Consolidation, 337, 445
Constructors
casting and typed value, 70–71
computed, 16–17, 123, 226–227
constructed element type and, 123
defined, 445
document creation, 15
dynamic content, 15
implicit validation, 75–77
Content management application, native XML database, 432–434
Context items
defined, 445
untyped data and, 109–112
XPath, 102–103
Conversion rules
atomic types, 48
function arguments, 69
return values, 70
Cotton, Paul, 82
Curly brackets ({ }), 15, 93, 123

D

Data binding, 421
Data definition, 401
Data definition interfaces, 401–405
database-design phase, 403–404
overview of, 401–403
query-analysis time, 404
query-authoring phase, 404
query-evaluation time, 405
Data extraction/insertion, DBMS access, 415
Data interchange
models, 353–357
relational systems, 355
XML format for, 81
Data manipulation, 401
Data models
defined, 445
encoding unknown and inapplicable data, 112
relational. *See* Relational data model
XML. *See* XML data model

XQuery. *See* XQuery (XQ-DM) data model
Data modification language, 379–380
Data-oriented (or data-centric) XML, 428–429, 445
Data sources, 315–316
Data structures, 175
Database command language, 411–412
Database configuration, 412–413
database objects, 412
indexes, 412
root objects, 412
triggers, 412–413
Database-design phase, 403, 403–404
Database languages, 154
Database objects, 412
Databases
configuring, 403, 407
defined, 393
managing. *See* DBMSs (database management systems)
native XML. *See also* Native XML databases
object oriented databases (OODB), 414
relational. *See also* Relational databases
Datatypes. *See* Types
Date, 70, 268
DBMSs (database management systems), 393–394
access features, 415
interoperability, 400–401
native XML databases and, 393–394
row updates, 405–406
Debugging. *See also* Error handling
auction use case, 197–199
static typing and, 201–203
validation and, 200–201
Decimals (.), 8
Decoupled approach, XQuery processing, 381–382
Default mapping, 325–328, 445
The Definition of Standard (Milner, Tofte, and Harper), 237
Dependencies, expressions, 178
Dependent joins, 343–344, 446
Derivation by extension, inheritance, 96, 446
Derivation by restriction, inheritance, 96, 446
Derived types, 96, 446
derives, (matching judgments), 275–276
Descendant operator, path expressions, 221
Design principles, XQuery, 86–88
distinct-values( ), 30–33
div operator, 40, 99
Division by zero, 136
doc( ), 9–10, 92
Document nodes, 7, 208
Document Object Model. *See* DOM (Document Object Model)
Document order
defined, 446
ordered sequences and, 138
Query data model and, 90
XML documents, 7
XPath, 100
Document-oriented (or document-centric) XML, 428–429, 446
Document Style Semantics and Specification Language (DSSSL), 155
Document Type Definitions. *See* DTDs (Document Type Definitions)
Documents, well-formed, 396
DOM (Document Object Model)
BLOBs, 361
building, 421
defined, 446
updates and, 314
XPath and, 150
Double slash (//), 10, 13
Doubles (E or e), 8
DSSSL (Document Style Semantics and Specification Language), 155
DTDs (Document Type Definitions)
absence of schemas, 109
data definition interfaces, 401
defined, 446
schemas, 4
XML documents, 128–129
Dublin Core, 14
Duration type, XML Schema, 97
Dynamic errors
defined, 446
handling in XQuery, 136
input functions and, 10
kinds of errors, 251–252
Dynamic rules, 246–248
Dynamic semantics, 241–254

abstract syntax, 241
defined, 240, 447
element constructors, 302–304
environments, 245–248
errors, 250–254
evaluation judgments, 241
FLWOR expressions, 281–282
inference rules, 242–245
matching values and types, 248–250
static semantics and, 254
Dynamic types
associating types with values, 212
compared with static types, 126
defined, 447
general comparison operators and, 43, 111

E

E or *e* (doubles), 8
Edge tables
defined, 447
table shredding and, 364–365
for XML data, 338–339
Effective Boolean Value
defined, 447
error handling and, 137
functions of, 103
element( ), 270
Element constructors, 122–125
computed, 123
data construction with, 4, 226–227
defined, 447
dynamic semantics, 302–304
enclosed expressions, 15–16
end tags and start tags, 93
formal semantics (XQ-FS), 301–304
implicit validation and, 75–77
node identity and, 299–300
Quilt proposal, 122
return clause and, 26–27
schema validation and, 124–125
static semantics, 304
stores and, 301–302
XSLT vs. XQuery, 160–163
Element nodes
kinds of nodes, 7, 208
referencing, 119–120
Element types, 270
Elements
declaring in schemas, 95, 447
defined, 447
global and local declarations, 204–205
importing types and, 65–66
matching judgments, 273–278
node values, 268–269
Emit as events, 421
empty( ), 48
empty sequence( ), 38
Empty sequences, 42, 113–114
Enclosed expressions, 15–16
Encoding, 358, 365–366, 448
End tags, 93, 448
Entity references, 9
Environments
defined, 448
dynamic, 245–248
inference rules and, 246–248
static, 254–255
eq operator, 41–42
Equals (=) operator, 43–45
error( ), 136
Error handling, 250–254. *See also* Debugging
auction use case, 201–203
detecting errors, 169
division by zero and, 136
dynamic errors and, 136
formalizing notation of raising errors, 250–254
if-then-else expressions, 138
kinds of errors, 251–252
non-deterministic (special rules), 136–137
static type errors, 125–126, 215, 219
try/catch mechanisms, 138
type errors and, 223
watershed issues, XQuery design process, 136–138
Error types, 136
Error values, 136
Evaluation judgments
defined, 448
notation for, 241
stores and, 301–302
Evaluation order, static semantics, 262–264
Evaluation-time. *See* Query-evaluation phase

except operator, 46
Execution order, XQuery/SQL integration, 384
Existential quantification
  defined, 448
  design of XQuery and, 88
  general comparison operators and, 114
  overview of, 36–37
exists( ), 49
Expressions
  compositionality and, 86
  delimiting enclosed, 93
  dependencies of, 178
  FLWOR. *See* FLWOR expressions
  matching and ranking, 423
  optimizing, 179–181
  path. *See* Path expressions
  static typing and, 212
Extensible Markup Language (XML). *See* XML (Extensible Markup Language)

F

Facets
  checking with constructor function, 71
  defined, 448
  XML Schema, 95
Factored types
  defined, 449
  FLWOR expressions, 224, 282–286
  overview of, 220
Factorization theorem, 285–286
Fibonaci sequences, 55
Flat data structures, 84
FLWOR expressions, 4
  for clause, 19–22
  clauses, 19
  combining data sources, 33–34
  combining with distinct values( ) to remove duplicates from subtrees, 30–33
  defined, 449
  direct translation into XSLT, 164–165
  dynamic semantics, 281–282
  factored types and, 224, 282–286
  grammar for, 279
  iteration with, 140–141, 225
  let clauses, 19–22
  normalizing, 280–281
  optimizing joins, 180–181
  order by clause, 23–26, 226
  overview of, 18–19
  return clause, 26–28
  static semantics, 282–286
  where clause, 22–23, 225
fn:boolean, 291
for clauses
  combining data sources, 33–34
  dynamic semantics, 281–282
  FLWOR expressions, 19–22
  FLWOR grammar, 280
  normalizing, 280–281
  overview of, 19
  positional variables and, 28–30
Foreign keys, 84, 449
Formal semantics (XQ-FS), 237–305
  benefits of, 238–240
  combining types, 210
  defined, 449
  dynamic semantics. *See* dynamic semantics
  element constructors, 301–304
  FLWOR expressions, 279–286
  function calls, 296–298
  implicit operations, 293–296
  irregular data, 293–298
  matching judgments, 272–278
  node identity, 299–300
  normalization, 264–266
  notation, 204
  overview of, 240–241
  path expressions, 287–292
  sequence types, 209–210
  static semantics. *See* static semantics
  subtyping, 278–279
  types, 204, 269–272
  values, 268–269
  XML Schema conversion, 204–206
fs namespace, 289
Full-text searches, 422–428
  features, 427–428
  invoice example, 431
  overview of, 422–423
  query-execution plans, 424–426
  ranking algorithms, 426–427
  return clause, 424

techniques, 423
Function calls, 216–217, 296–298
Function libraries, 378–379
Function overloading
defined, 449
overview of, 132
resolving, 296–297
XQuery and, 133, 135
Function resolution, 132–135
defined, 449
overloading and, 296–297
overview of, 132
simulating polymorphism, 135
XPath 1.0 core function library, 133–134
Function signatures, 216
built-in types and, 55
declaring parameters with sequence types, 67–70
defined, 449
Functional languages
defined, 449
lazy evaluation, 177–178
pipelining, 175–176
XQuery as, 86
XSLT as, 146–147
Functions, 216–217
built-in, 47–49
calling, 216–217, 296–298
error handling, 137
external, 52
library modules and, 51–52
resolving overloaded, 296–297
signature of, 216
type definition and, 118–119
user-defined, 49–50
variable definition and, 50
XPath 1.0, 151
XSLT vs. XQuery, 163–164

G

General comparison operators
atomization and, 298
defined, 450
equals (=) operator, 43–45
overview of, 42–43
rules for handling data based on type, 111
transitivity property and, 105
unknown values and, 114
value comparison operators and, 41
Global element declarations, 204–205, 271, 450
GROUP BY, SQL, 330, 340
Grouping
defined, 450
example, 233–234
influence of SQL on XQuery, 107
XML data generation and, 324–325

H

Hierarchical groups, XML data, 324–325
Hierarchical joins, 323–324, 450
Hierarchical order, 93
HTML, 146
HTTP, 395
Hypotheses, inference rules, 242

I

Identity
defined, 450
node, 4, 6, 299–300
idiv operator, 40, 99
if clause
conditional expressions and, 38
normalizing, 280–281, 294
If-then-else expressions, 131, 138
Impedance mismatch problems, 421–422
Implicit operations, 293–296
examples, 294–296
flexibility of, 293
XPath, 104–105
Indexes
canonicalized XML and, 361
configuring, 407–409, 412
data definition and, 403
defined, 450
Inference rules
dynamic environments, 246–248
hypotheses and conclusions, 242
operational semantics and, 241
proof tree, 243–244

static environments, 255–257
static typing, 126, 234
type soundness theorem and, 260
Information items, 89, 450
Infoset (Information Set), 89, 395, 451
Infoset-level fidelity, XML storage, 360
Inheritance
defined, 451
type inheritance and, 96
XML Schema, 96
XPath, 99
XQuery, 87
Inlining, 337, 451
Input data
auction use case, 190–192
Query data model and, 91–92
uptyped data, 109
validating, 199
Input functions, 9–10
Instance-of expressions, 131
instance of (operator), 72
Integers
atomic values, 268
numeric literals, 8
XML and, 168
Integrated approach, XQuery processing, 381–383
Interfaces, native XML databases, 399–422
configuration interfaces, 406–410
data definition interfaces, 401–405
interoperability, 400–401
internal coding efficiency, 174–175
International Standards Organization (ISO), 400
Internet Explorer, 145
Interoperability, 400–401
intersect operator, 46–47
Inverting hierarchies, 34–35, 37
Invoice archive application, 430–431
is/is not, node comparisons, 45
ISO (International Standards Organization), 400
Item type, 270
Items
atomic values, 7, 268
context items, 102–103
defined, 451
information items, 89
node values, 268
testing for type, 72
values and, 207
XQuery data model, 90
Iterations
FLWOR expressions, 140–141, 225
nested (joins), 159
pipelining and, 176

J

Java Virtual Machines (JVMs), 362, 451
JDBC (Java Database Connectivity), 400, 416
Joins
combining data sources, 33–34
defined, 451
dependent, 343–344
optimization, 180–181
simple, 342–343
Sorted Outer Unions (SOU), 344–347
XSLT and XQuery and, 159
Judgments. *See* Evaluation judgments
JVM (Java Virtual Machines), 362, 451

K

Keywords
defined, 451
referencing nodes by, 120
as syntax for XQuery, 82
XPath, 99
KindTest, step expressions, 12

L

last( ), 176
Lax validation mode, 200, 368
defined, 452
element constructors and, 125
uses of, 230–232
XML Schema, 75–77
Layered processing model, XML applications, 310–311
Lazy evaluation, 177–178

- let clause
  - dynamic semantics, 281–282
  - FLWOR expressions, 19–22
  - FLWOR grammar for, 279
  - iteration controls, 140–141
  - normalizing, 280–281
  - overview of, 19
  - static semantics, 283
  - variables, 214
- Library modules, 51–52
- List type, 99, 452
- Literal expressions
  - numeric types, 8
  - operators, 213–214
  - string literals, 8–9
  - types associated with, 212–213
- LOB representation
  - application types supported, 320
  - coexisting with composed representation, 319
  - compared with composed representation, 316–317
  - querying or transforming XML documents, 348
- LOBs (large object)s
  - data storage and, 310, 356
  - defined, 452
- Local elements declarations, 204–205, 221–222, 452
- Locking, 410, 433
- Logical models, XML data, 359–360
- Logical operators, 113

M

- Mapping combined XQuery/SQL queries, 380–385
- Mapping languages, 334
- Mapping XML to relational data, 309–352
  - annotated XML templates, 331–334
  - applications for, 312–315
  - composed representation, concepts, 320–325
  - data sources, 315–316
  - default mapping, 325–328
  - emitting XML documents, 339–347
  - overview of, 309–310
  - processing models for, 310–312
  - querying and updating XML documents, 347–350
  - representation approaches, 316–320
  - shredding, 335–339
  - SQL/XML, 328–331
- Markup data, 353–355, 452
- Matching judgments, 272–279
  - defined, 452
  - derives, 275–276
  - matches, 276–278
  - notation for, 249–250
  - overview of, 272–274
  - substitutes, 275
  - subtype, 278–279
  - yields, 274–275
- max( ), 47–48
- Metadata, 84, 452
- Minus (-) operator, 8, 40
- mod operator, 40
- Multiply (*) operator, 40
- Mutators, 379, 452

N

- Namespace
  - attributes, 16
  - defined, 453
  - influences of XML on XQuery, 93
  - schemas, 64
  - target namespace, 117
  - URL for, 211
- Namespace prefixes
  - defined, 453
  - library modules and, 52
  - QNames and, 93–94
  - schema imports and, 58
- *NameTest*, step expressions, 11–12
- Native XML databases, 393–435
  - collections and storage, 413–414
  - command language, 411–412
  - configuration interfaces, 406–410
  - content management application, 432–434
  - data definition interfaces, 401–405
  - full-text searches, 422–428
  - getting the data, 421–422
  - interfaces, 399–400

interoperability, 400–401
invoice archive application, 430–431
overview of, 393–395
sample applications, 428–430
text data, 395
update interfaces, 405–406
XML data model and, 396–399
XQuery client APIs, 414–420

Nesting
defined, 453
iterations, 159
sequences, 95
XML elements, 83–84

Netscape, 145

Node comparison operators, 45

Node hierarchy, 89–90, 453

Node tables, 363–365, 453

Node tests, XPath steps, 150

Node types, 270

Node values, 267–269

Nodes
combining and restructuring, 18
creating, 14–18
defined, 453
formal semantics, 267–270
identity, 299–300
of items, 90
kinds of, 6–7, 208
locating with path expressions, 10–14
referencing, 119–120
steps, 100
XML document structure, 4

Nonlinear hierarchy, 324–325, 454

Nonrecoverable errors, 98

Normalization rules
defined, 240, 454
FLWOR expressions, 280–281
irregular data, 294–298
overview of, 264–266
XPath, 288–292

not( ), 48

Notation. *See also* Formal semantics (XQ-FS)
comments, 8
path expressions, 10
string literals, 8–9
XML and XQuery and, 93
XPath, 99–100

Null
defined, 454
relational data and, 85
sorting, 25

Numeric functions, 48

Numeric literals, 8

O

Object oriented databases (OODB), 414

Occurrence indicators
defined, 454
sequence types, 61, 210
type references and, 120–121

ODBC (Open Database Connectivity), 400, 416

OLTP (Online transaction processing), 454

OODB (object oriented databases), 414

Open Database Connectivity (ODBC), 400, 416

Operational semantics, 241, 454

Operators, 39–47
arithmetic, 40–41
comparison, 41–45
existential quantification and, 88
literals and, 213–214
ordered sequences and, 138–141
overview of, 39–40
sequence, 46–47
uptyped data and, 109, 111
XPath 1.0, 151

Optimization, 170–183
defined, 454
expression rewriting, 178–181
internal coding efficiency, 174–175
overview of, 170–171
pipelining and lazy evaluation, 175–178
query-analysis phase and, 88
query-execution plans, 426
query processors and, 169
static typing, 125–126
time uses and, 171–174
type information, 181–183
XSLT and, 166

OQL, 107–108

order by clause
FLWOR expressions, 140–141, 226

  FLWOR grammar and, 279
  function of, 23–26
  overview of, 19
Outer joins, 107, 455
Output data
  auction use case, 194–196
  Query data model and, 91–92
  validating, 199
Overloaded functions. *See* Function overloading

P

Parse errors, 254
Partial updates, 406, 455
Path expressions, 4
  axes, 166
  defined, 455
  descendant operator, 221
  evaluating, 10–12
  examples, 12–14, 219–222
  factored type and, 220
  formal semantics, 287–292
  locally declared elements in, 221–222
  notation, 10
  predicates, 10–11
  rules for executing, 101–102
  steps, 150–151
  type error and, 11
  wildcards, 221
  XPath, 100
Path extraction, 421
PCDATA, 109
Physical mapping, XQuery/SQL queries, 380–385
Physical models, XML datatype, 360–365
  BLOB (binary large objects), 361
  CLOB (character large objects), 360–361
  relational table mappings, 363–365
  user-defined, 361–362
Pipelining, 175–177
Plus (+) operator
  arithmetic operators and, 40
  normalizing, 295
  numeric literals and, 8
Polymorphism
  not supported by XQuery, 134
  overview of, 132–133
  simulating, 135
Position, of nodes, 100
Positional variables, 28–30
Post-Schema Validation Infoset. *See* PSVI (Post-Schema Validation Infoset)
Predicate expressions, 222–224
  defined, 455
  empty sequences and, 114–115
  as integer-valued expression, 224
  kinds of, 102–103
  normalizing, 290–291
  overview of, 10–11
  picking elements by position, 223–224
  steps and, 150
  what it does, 222
PrimaryExpression, 12
prime( ), 285
Prime types, 283–284
Primitive types
  defined, 455
  XML Schema and, 96
  XQuery and, 87
Principles, XQuery design, 86–88
Processing instruction nodes, 208
Processing models, mapping XML to relational data, 310–312
Programming languages
  database languages, 154
  efficiency vs. determinism, 137–138
  functional vs. procedural, 4
  mapping languages, 334
  pipelining and, 175–176
  precise description of, 238
  query languages, 106–108
  transformation languages, 152–154
  XQuery-related. *See* XML (Extensible Markup Language); XML Schema; XPath
Prolog
  binding namespace prefix to URI, 94
  defined, 455
  external functions, 52
  variables, 52
Proof trees
  defined, 455
  environments and, 247–248

inference rules, 243–245
static typing and, 258
PSVI (Post-Schema Validation Infoset)
defined, 455
schema validation and, 89
typing datatypes, 368

Q

Qualified names (QNames), 93, 455
Quantifiers
computing, 283–284
error handling and, 137
inverting hierarchies and, 37
overview of, 36–37
quart( ), 285
Queries
basing on XML format, 82
optimizing by rewriting, 178–179
phases of, 88, 211–212, 402–404
ranking algorithms, 426–427
relational compared with XML query, 84–85
results, 399
testing existence of XML structure, 372–373
XML data from relational database, 313
XML documents, 347–350
Query-analysis phase
overview of, 402–404
query phases and, 88
static analysis, 211–212
Query-authoring phase, 403–404
Query data model, 89–92
as basis of XQuery, 89
defined, 456
input and output of data, 91–92
kinds of data models, 87, 89–92
specifications based on, 89
type annotation, 118
XML documents represented by, 90–91
XPath and, 89–90
Query-evaluation phase
answers provided by, 211–212
overview of, 402–403, 405
query phases and, 88
type soundness and, 258–262
Query-execution plans, 424–426
Query languages
defined, 456
influences on XQuery, 106–108
need for new language, 152–156
Query processors, 169, 174
Quilt language
defined, 456
element constructors, 122
influence on XQuery, 106–107
Quotation marks (" "), delimiting string literals, 8

R

Rank
algorithms for, 426–427
defined, 456
ranking expressions, 423
Recursive functions, 50, 456
Relational data, 84
Relational data model
compared with XML data model, 84–86, 353
defined, 456
mapping XML documents to. *See* Mapping XML to relational data
pipelining and, 175–176
query optimization and, 178–179
rows and columns of data in, 112
Relational databases
data source for XML applications, 309
defined, 456
emitting XML documents or XML fragments, 312–313, 339–347
list of, 357
querying or transforming XML documents, 313
shredding and, 335–336
storing XML data. *See* XML (Extensible Markup Language), relational storage
strongly typed nature of, 53
XML representation of, 3
Relational-level fidelity, XML storage, 360
Relational table mappings
data storage and, 356
node tables, 363–365

  - table shredding, 364–365
- Relational tables, XML views of, 388–390
- RELAX-NG, 129, 401
- Reserved words, 99, 456
- Result trees, XSLT, 160–161
- return clause
  - FLWOR grammar and, 279
  - full-text searches, 424
  - function of, 26–28
  - iteration controls, 140–141
  - overview of, 19
- Return values, 70
- Root nodes, 90, 456
- Root objects, 412
- Rows, relational data model, 112
- Rules
  - dynamic inference rules, 246–248
  - promotion rules for conditional expressions, 218–219
  - static inference rules, 255–257
  - untyped data, 109–110
  - XSLT as rule-based language, 147

S

- Salton, Gerard, 426–427
- SAX (Standard API for XML), 421
- Saxon, 146, 177
- Schema imports
  - Basic XQuery, 129
  - defined, 457
  - overview of, 210–211
  - typed value extraction and, 40
- Schema validation
  - defined, 457
  - element constructors and, 124–125
  - invoice example, 431
  - modes of, 368
  - PSVI and, 89
  - type annotation and, 118
  - xs:anyType, 62
- Schemas
  - built-in types and, 53, 56–60
  - configuring, 410
  - conformance principle, 87
  - declaring elements, 95
  - defined, 457
  - importing types from, 62–67
  - information in, 64–65
  - W3C XML Schema language, 367–368
  - XQuery, 4
- SELECT statement, SQL, 164
- Semantic constraints, 86, 457
- Semi-structured data
  - defined, 457
  - optimizing, 169
  - XML and, 316, 353, 354–355
  - Sequence operators, 46, 46–47
- Sequence types, 60–67
  - built-in, 61–62
  - converting into choice types, 284
  - defined, 457
  - function signatures and, 67–68
  - imported from schemas, 62–67
  - lists of, 209–210
  - Occurrence indicators, 61, 210
  - overview of, 60
- Sequences
  - heterogeneous nature of, 95–96
  - of items, 7
  - operators for ordered sequences, 138–141
- Sets, XPath, 151
- SGML (Standard Generalized Markup Language), 155
- Shanmugasundaram, Jayavel, 344
- Shredding, 335–339
  - adding information to composition, 336
  - composed representation and, 318–319
  - database creation, 335–336
  - defined, 310, 457
  - full XML support, 337–338
  - inlining and consolidation, 336–337
  - overview of, 335
  - schema-independent representation, 338–339
  - table shredding, 364–365
- Signature. *See* Function signatures
- Simple joins, 342–343
- Simple types
  - constructor functions for, 70–71
  - defined, 457
  - formal semantics, 271
  - list of, 6
- Skip validation mode, 77, 200, 368

defined, 457
element constructors and, 125
uses of, 230–232
slash (/), 10, 99, 288
SOAP messages, 355, 458
sort( ), 139
SORT BY, SQL, 340
Sort keys, 139–140, 458
Sorted outer unions (SOU), 344–347, 458
Sorting, XPath vs. XSLT, 176–177
SOU (sorted outer unions), 344–347, 458
Sources, XML data
relational, 315
XML sources, 315–316
Spelling errors, 59
SQL-2003
defined, 458
mapping relational tables to XML, 388
relational storage and, 357
SQL WITH clause, 375
XML data storge, 366
XML datatype, 359
SQL Access Group, 400
SQL multimedia (SQLMULTI), 423
SQL query language
built-in functions (XQuery), 47–48
WITH clause, 375
as database query language, 154
defined, 458
development of XQuery and, 83
generating XML data, 328–331, 340–341
influence on XQuery, 107
INSERT clause, 387
mapping SQL types to XML Schema types, 377–378
SELECT clause, 164, 371
three-valued logic, 112, 115–116
updates, 314–315
XPath sets compared with SQL tuples, 151
XQuery designers' background with, 155–156
SQL query language, integration with
accessing SQL data in XQuery, 375–378
aspects not related to queries, 370
augmenting XQuery static context, 374–375
data modification language, 379–380
decoupled approach, 381–382
integrated approach, 381–384
issues with, 385–386
logical models for XML data, 359–360
physical mapping of combined XQuery/SQL queries, 380–385
physical models for XML data, 360–365
storing XML data, 357–358, 366
typing XML data, 366–369
XQuery and, 370–374
XQuery function libraries, 378–379
SQL/XML
composition technique, 328–331
defined, 458
mapping relational tables to XML, 388
mapping SQL types to XML Schema types, 377–378
SQL-2003 standard, 357
W3C XML Schema, 53
XML datatype, 359
Standard API for XML (SAX), 421
Standard Generalized Markup Language (SGML), 155
Standards
interoperability and, 400–401
SQL-2003 standard, 357
Start tags, 93, 458
Static analysis, 272
Static context, 374–375
Static environment
defined, 458
validation context of, 228
variables in, 214–215
Static semantics, 254–264
defined, 458
element constructors, 304
evaluation order, 262–264
FLWOR expressions, 282–286
inference rules, 255–257
notation for static environment, 254–255
overview of, 254–258
rules, 255–257
type soundness and, 258–262
typing judgments and, 254
Static type-checking, 128, 459
Static type error, 219

Static types
  checking, 128
  defined, 459
  evaluation-time types and analysis-time types, 211–212
  schema imports and, 210–211
  sequence types, 209–210
  tests and, 60
  values, 207–208
  XML Schema and XQuery types, 204–207
Static typing
  advantages of, 126, 188–189, 234
  associating types with values, 212
  Basic XQuery and, 129
  conditionals, 218–219
  debugging cycle and, 201–203
  defined, 240
  element constructors, 226–227
  error checking and optimization, 125–126, 130
  expressions supporting, 131–132
  FLWOR expressions, 224–226
  functions, 216–217
  grouping, 233–234
  inference rules, 126–127, 234
  literals, 212–214
  path expressions, 219–222
  predicates, 222–224
  schema imports and, 210–211
  sequence types, 209–210
  spelling error detection, 59
  static type-checking, 128
  validated annotations vs. structural comparisons, 127–128
  validation compared with, 201
  validation context, 227–230
  validation modes, 230–232
  values, 207–208
  variables, 214–215
  weaknesses of, 234
Static Typing option, 129–130
Steps
  defined, 459
  forward vs. reverse, 290
  of path expressions, 10–11, 13, 100
Storage
  native XML databases, 413–414, 431
  XML datatype, 366
Stored functions, 409–410
Stores
  capturing node identity, 299–300
  defined, 459
  evaluation judgments and, 301–302
Strict validation mode, 77, 200, 368
  defined, 459
  element constructors and, 124
  uses of, 230–232
String-level fidelity, XML storage, 360
String literals, 8–9
String operators, 105–106, 268
String values
  defined, 459
  operators and, 39
  Query data model and, 91
Strongly typed data, 182. *See also* Structured data
Structured data
  defined, 459
  relational management of, 353
  semi-structured data and, 169
  XML and, 316, 354–355
  XQuery and, 53, 56
Structured types, SQL, 83
Subelements, 319, 323–325, 388, 459
substitutes, (matching judgments), 275
Substitution groups
  creator element and, 64
  defined, 459
  type inheritance and, 96
Subtrees, 30–33
subtype, (matching judgments), 278–279
Subtyping, 272

T

Table shredding, 364–365
Tables
  edge tables, 338–339, 364–365, 447
  node tables, 363–365, 453
  relational, 112, 388–390
  relational table mappings, 356, 363–365
Tags, 93, 459
Tamino, 394, 408
Target namespace, XML namespaces, 117

Template rules, 165–166
Text
  formal semantics, 270
  node values, 268–269
  XML data, 395
Text nodes, 7, 208
then clauses, 38
Three-valued logic, 112, 115–116, 460
Top-level XQuery
  collection( ) and, 386–387
  overview of, 386
  XML views of relational data, 388–390
Transaction controls, DBMSs, 415
Transformation languages, 152–154
Transformations
  inverting hierarchies, 34–35
  tree-to-tree, 147–148
  XML data from relational database, 313
  XML datatype instances, 371
  XML transformations, 145
  XSLT and, 152
Transitivity, 104–105, 460
Treat expressions, 74, 131–132
Trees
  comparing XSLT and XQuery, 160–163
  design efficiency of, 175
  operator trees, 380–381
Triggers
  configuring, 406, 409–410, 412–413
  defined, 460
Try/catch mechanisms, 138
Tuples
  for clauses, 19–22
  FLWOR expressions, 19
  let clauses, 19–22
  order by clauses, 23–26
  where clauses, 22–23
  XPath sets compared with, 151
Two-phase commit, 431
Two-valued logic, 114, 460
Type annotation, 91, 118, 460
Type errors
  defined, 460
  detecting, 58
  kinds of, 251–252
  path expressions and, 11
  static typing and, 223
Type promotion, 69, 460
Type soundness
  defined, 461
  relating static and dynamic semantics, 258–262
  static typing and, 126
Type soundness theorem, 240, 260, 461
Type system, 203–211. *See also* Static typing
  analysis-time types, 211–212
  defined, 461
  error detection, 227
  evaluation-time types, 211–212
  schema imports and, 210–211
  sequence types, 209–210
  type strictness, 168–170
  values, 207–208
  XML Schema and XQuery, 204–207
Typed values
  defined, 461
  extracting, 39–40
  nodes converted to, 217
  Query data model and, 91
  XQuery, 70–71
Typed variables, 71–72
Types
  atomic types, 90
  defined, 461
  formal semantics, 269–272
  function definitions and, 118–119
  mapping SQL to XML Schema, 377–378
  matching values to, 248–250, 272–274
  optimizing, 181–183
  referencing, 119–121
  simple and complex, 4
  type strictness, 168–170
Types, XQuery
  atomic values, 208
  built-in, 61–62
  casting and typed value construction, 70–71
  formal semantics, 204–207
  implicit validation, 75–77
  importing from schemas, 62–67
  instance of operator and, 72
  nodes, 208
  overview of, 54–56
  schemas and, 56–60
  sequence types, 60

- treat as expression and, 74
- typed variables, 71–72
- typeswitch expression, 72–74
- wide range of XML documents to process, 53–54

Typeswitch expression, 72–74, 131
Typing judgment, 254, 461

U

Unicode
- defined, 461
- UTF-8/UTF-16, 365
- XML and, 354

Union operators, 46
Union types, 128, 461
Universal quantification, 36, 461
Universal Resource Identifier (URI), 93–94, 461
Universal Resource Locator (URL), 211
Unknown values, 112–117
- relational representation of, 112
- XML's representation of, 112–113
- XQuery comparison operators, 114

unordered( ), 139
Untyped data
- defined, 462
- handling according to context, 109–112
- operators and, 41
- XQuery's accommodation of, 53, 56

Update interface, native XML database, 405–406
Updates
- defined, 462
- invoice example, 431
- partial, 406
- support for, 309
- user-defined types and, 362
- XML data from relational database, 313–315
- XML documents, 331–334, 347–350

URI (Universal Resource Identifier), 93–94, 462
URL (Universal Resource Locator), 211
Use cases, 82, 462. *See also* Auction use case
User-defined
- functions, 49–50, 217
- XML datatype, 381
- XML physical models, 361–362

UTF-8/UTF-16, 365

V

Validate expressions, 76
Validation. *See also* Schema validation
- auction use case, 199–201
- debugging cycle and, 200–201
- defined, 462
- implicit, 75–77
- input and output data and, 199–200
- XML Schema, 87, 97, 303

Validation context
- defined, 462
- function calls and, 229–230
- static environment and, 125, 228
- validate expression and, 76

Validation modes, 200, 368
- defined, 462
- element constructors and, 124–125
- list of, 230
- skip validation example, 231–232

Value comparison operators
- compared with general comparison operators, 41
- defined, 462
- overview of, 41–42
- transitivity property and, 105
- unknown values and, 114

Value space, 396, 462
Values
- atomic values. *See* atomic values
- defined, 462
- dynamic types and, 212
- formal semantics, 267–269
- matching to types, 272–274
- node values, 268
- query phases and, 211–212
- type system and, 207–208
- unknown and inapplicable data and, 112–117

Variables
- binding multiple, 33–34
- binding with FLWOR expressions, 18
- declaring, 71–72

defining, 51, 159
external, 52
let clause and, 214
overview of, 214–215
Virtual XML view, 356, 463

W

W3C (World Wide Web Consortium)
defined, 463
interoperability and, 400–401
query language workshop, 82
schemas, 4
specification consistency and, 156
SQL/XML proposal, 53
XML Schema language, 367
Wadler, Phil, 157
WebDAV
content management and, 433
defined, 463
updates and, 314
Well-formed documents, 396
where clauses
combining data sources, 33–34
FLWOR expressions, 225
FLWOR grammar, 279
function of, 22–23
iteration controls, 140–141
normalizing, 281
overview of, 19
quantifiers and, 36
Whitespace, 17, 463
Wildcard operators (*), 14, 221
Working data, 399
World Wide Web Consortium. *See* W3C (World Wide Web Consortium)

X

xdt:dayTimeDuration, 97, 129, 205, 225, 463
xdt:untypedAtomic
defined, 463
rules for handling, 111
schema validation, 62
xdt:yearMonthDuration, 97, 129, 205, 463
XHTML, 29
XIndice, 418
XML data model, 396–399. *See also* Infoset (Information Set)
levels of, 395
markup data and, 353
relational data model compared with, 84–86
textual representation, 395
XML datatypes
defined, 463
generating by extending SQL, 328–331
generating by hierarchical groups, 324–325
generating by hierarchical joins, 323–324
logical models, 359–360
mapping, 381
physical models, 360–365
queries and, 370
sources of, 315–316
SQL integration. *See* SQL query language, integration with
storing, 357–358, 366
typing, 366–369
user-defined, 381
XML documents
collections, 413–414
as data source, 315–316
DTDs and, 128–129
emitting from relational database, 312–313, 339–347
intrinsic order of, 85
mapping to relational data model. *See* Mapping XML to relational data
querying, 313, 347–350
range of, 53–54
representation by Query data model, 90–91
representation by relational data model, 3
storing, 413–414
transforming, 313
updating, 331–334, 347–350
XML (Extensible Markup Language)
comments, 8
data-centric vs. document-centric, 428–429
data interchange over Web, 81
defined, 448
element constructors, 122
text data, 168
transformations, 145, 146

unknown values, representing, 112–113
upgrade languages, 3
versatility of, 3
XQuery and, 92–94, 159–160
XSLT and, 159–160

XML (Extensible Markup Language), relational storage
encoding and collations, 365–366
logical models, 359–360
overview of, 357–358
physical models, 360–365
storing and exposing XML data, 356
typing XML datatype, 366–369

XML fragments
defined, 463
edge tables and, 338–339
emitting from relational database, 312–313

XML Infoset. *See* Infoset (Information Set)

XML Namespace, 93, 117

XML-QL, 107

XML Query Working Group (XQ-WG), 82–83
design principles for XQuery, 86–88

XML Schema
converting types to XQuery, 204–206
data definition interfaces, 401
defined, 464
duration type, 97
importing types, 66–67
mapping types to/from, 270, 377–378
primitive datatypes, 268
schema imports and, 210–211
simple types, 6
structural descriptions of types, 127
types, 4, 96
union types, 128
validation, 75–77, 200, 303
XQuery and, 4, 87, 95–97, 157
xsi:nil attribute, 117

XML storage fidelity, 360, 464

XMLAGG, SQL, 330–331

xmldb program, 418–420

xmlns, 94

XPath, 97–106
axes, 166
backward compatibility of version 2.0, 158
defined, 464
expressions, defining in terms of other expressions, 287
implicit operations, 104–105, 293–298
influence on design of XQuery, 97–98, 156–157
normalizing, 288–292
operators and functions, 151
predicates, 102–103
Query data model and, 89–90
steps, 150–151
syntax and semantics, 99–102
transitivity, 104–105
types, 98–99
version 1.0, 149–151
version 2.0, 156–158
XML Schema and, 157
XQuery and, 10, 87, 105–106, 121, 156–158
XSLT and, 148, 157

XPointer, 97

XQ-FS. *See* Formal semantics [XQ-FS]

XQ-LANG, 237, 403

XQ-REQ, 82

XQ-UC, 82, 189

XQ-WG (XML Query Working Group), 82–83, 86–88

XQuery
design principles, 86–88
development culture, 155–156
influences on, 83
integration with SQL. *See* SQL query language, integration with
reserved words and, 99
structure of, 3–4
technical requirements, 152–155
top-level. *See* Top-level XQuery
XSLT and. *See* XSLT (eXtensible Stylesheet Language), comparing with XQuery

XQuery 1.0 Language (XQ-LANG), 237, 402–403

XQuery design process, 108–141
element constructors, 122–125
error handling, 136–138
function resolution, 132–135
operators, ordering, 138–141
overview of, 108–109

static typing, 125–132
types and, 117–121
unknown and inapplicable data, 112–117
untyped data, 109–112

XQuery (XQ-DM) data model, 396–399. *See also* Query data model
compared with classical relational model, 4
defined, 456
illustration of, 397
native XML databases and, 398–399
overview of, 6–7
persistent document storage, 398
value space of, 396–398
XML data model and, 396

xs:anyType, 62, 91

XSD language, 331–334, 340

xs:decimal, 69

xs:double type, 43

xsi:nil, 117, 464

xsi:type attribute, 77

XSLT (eXtensible Stylesheet Language), 146–149
defined, 464
development culture, 155–156
integration with XML Schema and XPath, 157
key characteristics of, 146–148
processors, 171–174, 181–182
strongly typed data and, 182
technical requirements, 152–155
transforming XML to HTML, 146
XPath and, 97, 105

XSLT (eXtensible Stylesheet Language), comparing with XQuery
axes, 166–168
development culture, 155–156
element constructors, 160–163
FLWOR expressions, 164–165
functions, 163–164
similarities, 159
technical requirements, 152–155
templates rules, 165–166
type strictness, 168–170
XML-based syntax, 160

xslt:apply-templates, 165

xslt:for-each, 159

xs:string type, 43

XStreamDB, 416–418, 423

Y

yields, (matching judgments), 274–275