Symbol Index

$\wedge$, 44
$\vee$, 64, 74
$\neg$, 74
$\neg.\mathbf{I}$, 413
$\langle\,\rangle$, 32
$\hookrightarrow$, 293
$\bot$, 84, 388
$\{e_1, \ldots, e_m \mid \varphi\}$, 45
$\delta_f(\cdot)$, 58
$\mathcal{Q}_1 \sqsubseteq \mathcal{Q}_2$, 47
$\mathcal{Q}_1 \equiv \mathcal{Q}_2$, 47, 75
$\exists$, 44
$\forall$, 74
Γ_P, 356
$\rightarrow$, 75
$\bowtie, \bowtie_F$, 55, 57
$\bowtie[X_1, \ldots, X_n]$, 170
$\rtimes$, 128
$\leq_{\mathbf{att}}$, 30
μ, *349*, *353*
$\pi_{\bar{j}}, \pi_{j_1,\ldots,j_n}$, 54, 57, 550
$\pi^*(c)$, 550
$\vdash, \vdash^{\Gamma}$, 167
ψ_y^x, 47
$q^{\emptyset}$, 48
$\rightarrow_{\mathcal{S}}, \xrightarrow{*}_{\mathcal{S}}$, 56
$\mathbf{I} \models \varphi[\nu]$, 46, 163
$\models_{\text{fin}}, \models_{\text{unr}}$, 198
$\models_{\mathbf{d}}$, 77
$\sigma_{j=a}, \sigma_{j=k}$, 53, 57, 376, 382
σ_F, 55
$\Sigma^*, \Sigma^{*,U}$, 165
$\subseteq_\Sigma, \subseteq_{\mathcal{F}}$, 175
$\equiv_\Sigma, \equiv_{\mathcal{F}}$, 175
$\times$, 54
$\uparrow$, 557
$X \to Y$, 163
$X \twoheadrightarrow Y$, 170
$R[\cdot] \subseteq S[\cdot]$, 193
$X^*, (X, \Sigma)^{*,U}$, 165

Index

Page numbers in italics indicate the location of definitions of terms.

AC_0, 96, 431
Access, 36, 143, 150, 152–153, 155
access plan, 107
active database, 8, 600–606
  action, 601
  composite event, 606
  condition, 601
  coupling mode, 603
  ECA, 601
  event, 601
  execution model, 601, 603–606
    accumulating model, 604–606
    concurrent firing, 603
    deferred firing, 603, 604
    immediate firing, 603–604
  rule, 601
  rule base, 601
  vs. expert system, 600
active domain, 41, 46, 249
  interpretation, 79
  preservation, 249
active domain semantics
  of relational calculus, 74, *79*
  vs. domain independence, 79
acyclic
  vs. dependencies, 137
  distributed databases, 136
  hypergraph, 36, 132
  inclusion dependencies, 208–210, 211
  join, 105, 126, 128–135, 136
  join dependency, 169, 182–183, 186
    and mvd's, 182
adom, 41, 46, 77, 249
adorned rule, 318, 321
adornment, 317, 318
A-egd, 218
aggregate function, 91–93
aggregate operator, 97, 153, 154
  in query language, 155
AGM postulates, 599
agreement set, 188
Alexander method, 336
ALG^{cv}, *514*
ALG^{cv-}, 519
algebra
  complex value, *514*, 519
  conjunctive, 52–61
  cylindric, 96, 103
  named conjunctive, 56–59, *57*
  nested relation, 519
  relational, 28, 35, 36, 64, 70, 71, 81
    named, 71
    unnamed, 71
    translation into calculus, 80
  SPC, 52–56, *54*
  SPCU, 62, 222
  SPJR, 56–59, *57*
  SPJRU, 62
  SPJU, 492
  typed restricted SPJ, 64, 67
  unnamed conjunctive, 52–56, *52*
  unsorted, 103
algebraic dependency, 228–233

algebraic dependency *(continued)*
axiomatization, 231
ALGRES, 337
allowed calculus query, 97, 101–102
alternating fixpoint, *390*, 413
ancestor program, 63
nonlinear version, 314
anomaly
deletion, 162, 254
insertion, 162
modification, 162
update, 162
anonymous variable, 39, 44
ans, 40
*ans_*R^{γ}, 321
anti-symmetric, 11
any, 548
AP5, 605, 615
APEX, 335
arithmetic in query language, 153, 154
arity$(\cdot)$, 31
of instance, *32*
of relation name, *31*
of tuple, *32*
Armstrong relation, 168–169, 186, 232
for typed dependencies, 233
Armstrong's axioms for fd's, 186
articulation set, 132
artificial intelligence (AI), 97
atom, 22, 33
constraint, 112
equality, 217
ground, 34
relation, 112, 217
att, 30
attribute, 29
in relational model, 30
in semantic data model, 243
attribute renaming, 58
autoepistemic logic, 408
automorphism, 12, *420*, 426–428, 461
average in SQL, 91, 154
awk, 155
axiom, 24
vs. inference rule, 167
axiomatizable, 167
axiomatization, *167*, 226
abstract formulation, 203
for algebraic dependencies, 231, 235
complete, 167
for fd's, 166, 168, 186
for fd's and mvd's, 172–173, 186
finite, 202
for full typed dependencies, 227–228
Gentzen-style for jd's, 186
IDM transaction for, 581
for inds, 193–195, 211
k-ary, 202, 204
proof using, 167
provable using, 167
sound, 167
for typed embedded dependencies, 226, 235
for uinds, 210, 215
for uinds and fds, 210
vs. fds and inds, 192, 202–207, 211
vs. fds and sort set dependencies, 213
vs. finite implication, 226
vs. jd's, 169, 171, 186

B$(P, \mathbf{I})$, 280
B$(P_{\mathbf{I}})$, 387
B-tree, 107
bag, 92, 136
in SQL, 145, 155
BCNF, 250, 251–*252*
algorithm, 255
belief revision, 588, 599
Berge-acyclic, 131, 137
Bernays-Schönfinkel class, 219
Binary Data Model, 264
binary relation, 10
body of rule, 39, *41*, 276
bottom-up datalog evaluation, 324–335
vs. top-down, 311, 327, 336
bound coordinate in datalog evaluation, 318
bound variable occurrence, 23, *45*, 75
boundedness, *285*, 304
Boyce-Codd normal form (BCNF), 250, *251*–252,
BP-completeness, 428, 560
buffering of main memory, 106, 107

C-genericity, 419–420
C+SQL, 466
c-table, 493
and dependencies, 501
update, 593–594
CALC$_{adom}$, *79*, 80, 100
CALC$_{di}$, 79, 80
CALC$_{sr}$, 81, 86, 100
CALC+μ, 348–352, *349*
normal form, 368
simultaneous induction, 351
CALC+μ^{+}, 352–354, *353*
normal form, 368

CALC+$\mu^{(+)} + W$, 456
CALC$_{adom}$, 79
CALC$_{di}$, 79
CALC$_{sr}$, 85
CALC^{cv}, 519
CALC^{cv-}, 528
calculus
  complex value, *519*, 523
  conjunctive, 44–47, *45*
  domain, 39, 74
  for OODBs, 557
  positive existential, 68, 91
  relational, 28, 35, 36, 39, 64, 70, *73–91*
  tuple, 39, 74, 101
calculus formula, 74–75
  parse tree, 83
Cartesian product, 52
chain program, 303
chase(T, t, Σ), 176
chase, 43, 159, 163, 173–185, 186, 220, 263, 497
  Church-Rosser, 183–185
  complexity, 176, 190
  fd rule, 175
  generalized to embedded dependencies, 223–225
  generalized to full dependencies, 220
  incomplete database, 498
  ind rule, 208
  and ind's, 208
  jd rule, 175
  and logical implication, 180–182, 186
  query optimization, 163, 177–180
  and tableau minimization, 177–180
  of tableau queries, 173, 186
  tgd-rule, 223
  uniquely determined, 176
  vs. datalog, 186
  vs. resolution and paramodulation, 186
chase homomorphism, 184
chasing sequence, 175
  infinite, 208, 223, 225
  terminal, 175
  vs. dependency satisfaction, 175
choice operator, 458
Church-Rosser property, 175, 176
  chase, 183–185
CINEMA example, 31
circumscription vs. fixpoint operators, 354
Clark's completion. *See* datalog$^{\neg}$, negation as failure.
class, 543, *545*, 547
  in semantic data model, 243
class extension, 556
class hierarchy, 549
  well formed, 549
classification, 572, 575
clause, 288
Closed World Assumption (CWA), 27, 283, 489, 497, 599
clustering, 107
CNF, 83
co-r.e., 16
Codd, 64
Codd-table
  query, 488
  update, 593–594
COL, 538
compactness theorem, 25
complement of views, 591–593
complement operator, 103, 104
complete axiomatization, 167
complete lattice, 286
completeness, 18
  object-oriented language, 560–561, *560*, 574
  of a query language, 466
  relational, *96*, 147, 150, 151
  update language, 583
  of while$_N$, 468
  of while$_{new}$, 470–473
  of while$_{uty}$, 478
completion in Query-Subquery (QSQ), 318
completion of program, 407
complex constant, 517
complex value, 508–541, 542, 543, 545
  algebra, *514*, 519
  calculus, *519*, 523
  datalog, *532*, 533
  elementary query, 534
  Equivalence Theorem, 526–531
  fixpoint, 531–532
  instance, 512
  relation, 512
  safe-range, 528
  schema, 512
  semantic data model, 243
  sort, 511
  strongly-safe-range, 530
  term, 519
complex value model, 97, 548
complexity, 13–20
  data vs. expression, 122
  of query languages, 136
composition of tableaux, 226–227
composition of queries, 37, 48–52
  conjunctive queries, 64
  conjunctive queries with union, 64
  conjunctive query program, 49

composition of queries *(continued)*
functional paradigm, 50
imperative paradigm, 50
relational algebra queries, 71
and user views, 51–52
computability, 13–20
condensation, 136
condition box in QBE, 150
conditional table. *See* c-table.
conjunction, 44
flatten, 83
and negation, 74
polyadic, 46
conjunctive calculus, 64
with disjunction, 91
with equality, 48
equivalence of formulas, 46
normal form, 46–47
rewrite rule, 46
semantics, 45
with union, 81
conjunctive normal form (CNF), 21, 83
conjunctive query, 36, 37–64
algebraic, 52–61
with arithmetic, 105
calculus, 44–47, *45*, 64
normal form, 46–47
composition, 48–52, *50*
containment, 105
complexity, 121–122
and decidability, 36, 37, 117, 118
with disjunction, 61–64
equality, 47–48, 50
equivalence, 47, 82, 105
Equivalence Theorem, 60
evaluation, 56
Homomorphism Theorem, 105, 115–118, *117*, 127, 136
logic-based perspectives, 40–48
and Microsoft Access, 152
monotonic, 42
named algebra, 56–59, *57*
optimization, 36, 56, 105
in practical systems, 105–115
using chase, 163
using dependencies, 163
program, 49
range restricted, with equality, 41, 48, 65
rule-based, 39, 40–42, *41*
satisfiable, 42
and SQL, 143–146
static analysis, 105, 115–122
tableau, 43–44, *43*
with union, 36, 37, 38, 61–64
unnamed algebra, 52–56, *52*
vs. expert systems, 135
yes-no, 42
connectivity query, not first-order, 436, 460
$conseq_P$, 389
consistent
globally, 128, 136
pairwise, 128, 136
constant in relational model, 30
constraint, 186
inequalities over rationals, 96, 98
integrity, 28, 185, 236
vs. dependency, 157
polynomial inequalities, 96
temporal, 611–613
transition, 612
vs. first-order logic, 186, 234
constraint atom, 112
constraint database, 36, 71, *94–96*, 97–98
constraint programming, 97
constraint query language, 94–96, 97–98
containment
conjunctive queries, 105, 118
decidability, 117
differences of SPCU queries, 140
first-order queries
undecidability, 125
queries, 115
tableau queries, complexity of, 121–122
containment of queries
relative to dependencies, 175, 177
relative to family of instances, 174
context-free grammar, 19
context-free language, 20
continuous operator, 286
conventional perspective on relations, 32, 33
CORAL, 337
cost model for query evaluation, 106, 108–110
count, 91, 92, 154
counter machine, 15
counting vs. relational calculus, 154
counting technique, 327, *331–335*, 336, 341
covariance, 553
cover, 254
minimal, 257
create in SQL, 145
cross product, 52, *54*
physical implementation, 108
in SQL, 144
vs. equi-join, 108
vs. join, 58
cumulative assignment, 346

CWA. *See* Closed World Assumption.
cylindric algebra
  vs. relational algebra, 96, 103

dangling reference, 999, 572
data complexity, 122, 422–423
data definition language (DDL), 4, 28
data function, 306
data independence principle, 4, 9
data integrity, 162
data manipulation language (DML), 4, 28
data model. *See* database model.
data storage, 106
database access functional paradigm, 571
database instance, 29
  conventional perspective, 32
  logic-programming perspective, 32
database logic, 97
database management system, 3
database model, 4, 7, 28
  complex value, 508–541
  directory, 97
  Entity-Relationship (ER), 242
  functional, 574
  Functional Data Model, 264
  generic semantic model (GSM), 242
  hierarchy, 28, 97
  IFO, 242
  Logical Data Model (LDM), 97
  network, 28, 97
  object-oriented, 28; *See* object-oriented database.
  relational, 28–34
  semantic, 28, 207, 242–250
database schema, 29, *31*
  with dependencies, 241, 251
datalog, 39, *273–310*
  bottom-up, 312–316, 324–335
    vs. top-down, 311, 327, 336
  boundedness, *285*, 304, 309
    vs. first-order, 306
  chain program, *303*, 305, 309
  clause, 288
    definite, 288
    empty, 288
    goal, 288
    ground, 288
    unit, 288
  complex value, *532*, 533
  containment, 301–304
    uniform, 304, 305, 309
  and domain independence, 97
  evaluation, 112, 311–337
  adorned rule, 318, 321
  adornment, 318
  Alexander method, 336
  algebraic approaches, 336
  annotated QSQ, 330
  APEX, 335
  bottom-up, 312–316, 324–335
  bound coordinate, 318
  connected atom, 338
  counting, 327, *331–335*, 336, 341
  direct evaluation, vs. pre-compilation, 317
  Earley Deduction, 335
  extension tables, 335
  factoring, 337
  free coordinate, 318
  generalizations to logic programming, 336
  generalized supplementary magic set rewriting, 325, 336
  incremental, 337
  Iterative Query-Subquery (QSQI), 339
  left-to-right, 318
  magic set rewriting, vs. QSQ, 311, *324–335*, 336, 340
  memo-ing, 335
  naive, 312
  original magic set rewriting, 340
  parallel, 337
  pre-compilation, vs. direct evaluation, 317
  Query-Subquery (QSQ), 311, *317–324*, 335, 341
  rectified subgoal, 328, 330–331, 336
  Recursive Query-Subquery (QSQR), 323–324, *324*
  relevant fact, 317
  rule-goal graph, 335
  seminaive, *basic algorithm*, *improved algorithm*, *312–316*, 335
  sideways information passing, 318, 336, 340
  sip graph, 340
  SLD-AL, 335
  stratification, 337
  supplementary relation, 319–320
  top-down, 316–324
extensional database (ebb), 279
extensional relation, 277
extensional schema, 277
immediate consequence operator, *282*, 375
intensional database (idb), 279
intensional relation, 277
intensional schema, 277
least fixpoint semantics, 276, *282–286*
  Knaster-Tarski's Theorem, 286

datalog *(continued)*
linear program, 305, 316
linear rule, 316
magic set rewriting, 311, *324–335*, 336
generalized supplementary, 325, 336
original, 340
vs. QSQ, 324
minimum model semantics, 275, *278–282*
Herbrand interpretation, 282
Herbrand model, 282
monadic programs, 305
negative literal, 288
nonrecursive
with negation, 70, 72–73
normal form, 68
nonrecursive (nr) program, 62
optimization, 36, 112, 311–337
parallel evaluation, 337
positive literal, 288
precedence graph, 315
program, 276
proof tree, 286
proof-theoretic semantics, 275, *286–300*
prototype systems, 337
query, 317
Query-Subquery (QSQ), 311, *317–324*, 335
annotated, 330
completion, 318
Iterative (QSQI), 339
Recursive (QSQR), 323–324
template, 319–320
vs. magic set rewriting, 324
rule, 276
body, 276
head, 276
instantiation, 277
satisfiability, 300–301
semipositive, 379
sirup, 305, 309
SLD-resolution, 289–298
completeness, 297
datalog$^{\neg}$, 400
derivation, 290
most general unifier (mgu), 293
refutation, 290
resolvent, 289, 295
selection rule, 298
SLD-derivation, 295
SLD-refutation, 295
soundness, 296
unifier, 293
SLD-tree, *298*, 317
stratified evaluation, 337
syntax, 276
top-down vs. bottom-up, 311, 327, 336
and undecidability, 306, 308–311
vs. logic programming, 35, 278, 298
datalog$^{\neg}$, 308, 309, 355–360, *357*, 374–414
default model semantics, 408
inflationary semantics, *356*
locally stratified, 411
negation as failure, 406–408
Clark's completion, 406
finite failure, 406
SLDNF resolution, 406
noninflationary semantics, 357
nonrecursive, 70, 72–73
range-restricted, 372
rule algebra, *359*, 373
semipositive program, 377
on ordered databases, 406
vs. fixpoint, 405
SLB-resolution, 400
stable model semantics, 408, 413
vs. choice, 409
stratified, 374
stratified semantics, 377–385
independence of stratification, 382
on infinite databases, 411
on ordered databases, 406
precedence graph, 379
SLS resolution, 409
stratifiable program, 379
stratification, 378
stratification mapping, 378
vs. Fermat's Last Theorem, 411
vs. fixpoint queries, 400
supported model, *384*, 411
tie-breaking semantics, 409
update language, 582
valid model semantics, 409
well-founded, 374
well-founded semantics, *385–397*, 413
3-stable model, 389
3-valued instance, 387
3-valued model, 387
alternating fixpoint, *390*, 408, 413
global SLS-resolution, 409
greatest unfounded set, 413
on ordered databases, 406
total instance, 387
total program, 395
unfounded set, 413
vs. default, 412
vs. fixpoint queries, 400, 401
vs. stable, 412

datalog$^{\neg\neg}_{new}$, 483
DB2, 155
DBASE IV, 152, 155
dbms. 3
DDL, 28; *See* data definition language.
decidability, 16
  of implication for full dependencies, 220, 234
declarative vs. procedural, 35, 53
decomposition, 162, 251–259, *252*, 265–266
  dependency preserving, 254
  and functional dependency, 164, 171
  and join dependency, 169–171
  lossless join, 253
  mapping, 253
  multi-way join, 106, 114–115
  reconstruction mapping, 254
  vs. synthesis, 258, 265
DedGin, 337
deductive database, 8
  disjunctive, 502
deductive object-oriented database, 572, 574, 575
deductive temporal query language, 610
deep equality, *557*, 575
default logic, 408
definite clause, 288
definite query, 97
delete in SQL, 149
deletion, 580
  implicit, 556
deletion anomaly, 162, 254
dense linear order, 96, 98
dependency, 157
  afunctional, 234
  algebraic, 228–233
  axiomatization, 166, 171, 172, 186, 193, 202–207, 227, 231
  capturing semantics, 159–163
  classification, 218
  conditional table, 497
  and data integrity, 162
  and domain independence, 97
  dynamic, 234
  embedded, 192, 217, 233
  embedded implicational (eid), 233
  embedded join (ejd), 218, 233
  embedded multivalued (emvd), 218, 220, 233
  equality-generating (egd), 217–228
  extended transitive, 234
  faithful, 232, 233, 239
  finiteness, 306
  full, 217
  functional (fd), 28, 159, 163–169, *163*, 186, 218, 250, 257, 260
  general, 234
  generalized dependency constraints, 234
  generalized mutual, 234
  implication
    in view, 221
  implication of, 160, *164*, 193, 197
  implicational (id), 233
  implied, 234
  inclusion (ind), 161, 192–211, *193*, 218, 250
    acyclic, 207, 208–210, 211, 250
    key-based, 250, 260
    typed, 213
    unary (uind), 210–211
  inference rule, 166, 172, 193, 227, 231
    ground, 203
  join (jd), 161, 169–173, *170*, 218
  key, 157, 163–169, *163*, 267
  logical implication of, 160, *164*
    finite, 197
    unrestricted, 197
  multivalued (mvd), 161, 169–173, *170*, 186, 218
  mutual, 233
  named vs. unnamed perspectives, 159
  order, 234
  partition, 234
  projected join, 233
  and query optimization, 163
  satisfaction, 160
  satisfaction by tableau, 175
  satisfaction family, 174
  and semantic data models, 249–253
  and schema design, 253–262
  single-head vs. multi-head, 217
  sort set, 191, 213, 234
  subset, 233
  tagged, 164, 221, 241
  template, 233, 236
  transitive, 234
  trivial, 220
  tuple-generating (tgd), 217–228
  typed, 159
    vs. untyped, 192, 217
  unirelational, 217
  and update anomalies, 162
  and views, 221, 222
  vs. first-order logic, 159, 234
  vs. integrity constraint, 157
  vs. tableaux, 218, 234
dependency basis, 172
dependency preserving decomposition, 254
dependent class, 246
dereferencing, *557*, 558
derivation, 290

derived data, 246
determinate-completeness, 474, 561, 574
determinate query, 474, 559
diameter, 12
diff, 88
difference, 33, 36, 68
  in relational algebra, 71
  and SPCU algebra, 136
  in SQL, 146
  vs. negation, 70
direct product, 232, 238
directory model, 97
disjunction, 38
  in conjunctive queries, 37, 38, 61, 64
  flatten, 83
  and negation, 74
  in selection formulas, 62
disjunctive deductive database, 502
disjunctive normal form (DNF), 21, 83
disk, 106
distinct in SQL, 107, 145, 154
distributed database
  query optimization, 128
division in relational algebra, 99
DML, 4, 28
DNF, 83
dom, 30, 72
$Dom(\cdot)$, 30
domain
  active, 46
  in relational model, 29, *30*
  scalar, 153
  time, 607
  underlying, 74
domain calculus, 74
  vs. tuple calculus, 39
Domain Closure axiom, 26
domain independence, 70, 74, 75–77, *79*, 81–97
  and algebra, 78
  complex value, 526
  and datalog, 97
  and dependencies, 97
  with functions, 97
  and nr-datalog$^\neg$, 78
  with order, 97
  practical query languages, 153
  relational calculus, 81
  syntactic restrictions, 81–91
  undecidability, 97, 125
  vs. active domain semantics, 79
domain-inclusion semantics, 551
domain-key normal form, 265
dominance of query languages ($\sqsubseteq$), 47
DOOD. *See* deductive object-oriented database.
duplicate elimination, 107
  distinct, 107
duplicate tuples, 144
dynamic aspect of object-oriented database, 572
dynamic binding, 543, 546, *552*
dynamic choice operator, 464
Dynamic Logic Programming (DLP), 583, 613

ear of hypergraph, 130
Earley Deduction, 335
edb, 42, 49, *277*
edge of hypergraph, 130
egd, 217–228
  A-egd, 218
Ehrenfeucht-Fraissé games, *433–437*, 460
eid, 233
ejd, 218
EKS, 410
elementary functions, 18
elementary query, 534
embedded dependency, 192, 217
embedded implicational dependency (eid), 233
embedded join dependency (ejd), 218
embedded multivalued dependency (emvd), 218, 220, 233
embedding of tableau, 43
empty clause, 288
emvd, 218, 220, 233
enc_{α},, 418
encapsulation, 543, *546*, 553
entity, 543
Entity-Relationship (ER) model, 242, 264
equality atom, 217
equality-generating dependency (egd), 217–228
  A-egd, 218
equi-join, *55*, 108
  physical implementation, 107–108
  in SQL, 144
  vs. natural join, 57
equivalence
  algebraic, 106
  calculus formulas, 82
  conjunctive calculus formulas, 46
  conjunctive queries, 47, *60*, 64, 82, 105
    decidability, 118
  conjunctive queries with union, 63
  differences of SPCU queries, 140
  finite and unrestricted implicaton for full dependencies, 220, 234
  first-order languages, 36, 80, 96
  first-order queries, 74

undecidability, 125
of full typed and algebraic dependencies, 231
of hypergraph properties, 132
nr-datalog$^\neg$ and relational algebras, 73
queries, 37
relative to dependencies, 176, 177
query languages, 47
relational algebras, 71
SPC and SPJR algebras, 60
equivalence class, 10
equivalence relation, 10
Equivalence Theorem
conjunctive query languages, 60
conjunctive query languages with union, 63
first-order languages, 80
ER model, 242
ESQL, 368, 370
evaluable query, 97
evaluation
of conjunctive queries, 56
datalog, 112, 311–337
evaluation plan, 107, 108, 110, 135
generating, 110–111
parameterized, 135
exact cover problem, 121
existential quantification, 44
flatten, 83
vs. universal, 74
Exodus
and optimization, 135
and query evaluation plans, 111
expert system vs. conjunctive queries, 135
expression complexity, 122, *422–423*, 463
expressive power of object-oriented database, 569, 577
extended relational theory, 26
extension axioms, 26
extension tables, 335
extensional database edb, 42, 49, *279*
extensional relation, 42, 48, *277*

F-logic, 574
fact, 32
factoring, 337
faithful dependency, 232, 233, 239
vs. typed, 233
fd, 28, 159, 160, 163–169, *163*, 186, 218. *See* functional dependency
fd closure
algorithm, 165
of set of attributes, 165
of set of fd's, 165
fd rule in chasing, 175
fd-schema, 251
field, real closed, 97
file systems, 3
filter, 518
finitary power set, 10
finite interpretation, 26
finite logical implication, 197–202, 219
vs. unrestricted, 197
finite model theory, 123, 197
finite representation of infinite database, 93–96, 97
finite-state automata, 13
finitely implies, 198
finiteness dependency, 306
first normal form, 265
first-order incremental definability, *588*, 613
first-order language, 70–98
Equivalence Theorem, 80
and undecidability, 122–126
vs. SQL, 147–149, 155
first-order logic, 22, 35
vs. conjunctive queries, 40
vs. constraints, 234
vs. dependencies, 159, 234
vs. integrity constraints, 186
vs. relational calculus, 77, 105, 123, 136
first-order predicate calculus, 22, 35
first-order queries, 70–98, *70*
and dependencies in views, 222
equivalence, 74
expressiveness, 433–437
Ehrenfeucht-Fraissé games, *433–437*, 460
on ordered databases, 462
logspace complexity, 430–431
parallel complexity, 431–433
static analysis, 105, 122–126
and undecidability, 105, 122–126
fixpoint
complex value, 531–532
datalog, 276
incomplete database, 495
semantics of datalog$^\neg$, 390
fixpoint of an operator, 283
fixpoint queries, 342, *367*
on ordered databases, 447
ptime complexity, 437
vs. while queries, 453
flatten, 524
FOID, 588
format model, 539
formula, 22
conjunctive calculus, 45
conjunctive normal form (CNF), 83

formula *(continued)*
disjunctive normal form (DNF), 83
interpretable, 77
matrix of, 82
prenex normal form (PNF), 82
relational calculus, 74–75
4NF, 252, *252*, 259
fourth normal form (4NF), 252, *252*, 259
Foxpro, 152
FQL, 264
free(·), 45, 75
free coordinate
in datalog evaluation, 318
free tuple, 33
free variable occurrence, 23, *45*, 75
fsa. *See* finite-state automata.
full dependency, 217
full reducer, 129, 136
full tuple generating dependency, 218
full typed dependencies
axiomatization, 227–228
function-based perspective on tuples, 32
Functional Data Model, 264
functional dependency (fd), 28, 163–169, *163*, 186, 218
agreement set, 188
axiomatization, 166–168
with mvd's, 172–173
vs. ind's, 192, 202–207, 211
and chasing, 175
closure, 165
cover, 254
and decomposition, 162, 164, 171, 253–262, 255
dynamic, 615
independent of ind's, 250
logical implication
with ind's, 192, 199–202
linear time, 165
satisfies, 163
saturated set, 188
and synthesis, 260–261
and two-element instances, 189
vs. decomposition, 164, 171
vs. join dependency, 171, 178
vs. multivalued dependency, 171
vs. propositional logic, 186, 189
vs. semantic data model, 249–253
vs. unrestricted implication, 199
vs. propositional logic, 189
functional paradigm, 569
functional query language, 569

G_P, 379
Galileo, 264
game-of-life, 343
garbage collection, 556
Gauss-Seidel algorithm, 335
generalized instance, 95
generalized SPC algebra, 55
generalized SPJR algebra, 59
generalized tuple, 94, *95*
generic OODB model, 547–556
generic semantic model (GSM), 242–250
genericity, 103, 419–421, *419*, 425
globally consistent join, 128, 136, 261
GLUE-NAIL, 337
goal clause, 288
Gödel Completeness Theorem, 123, 136
graph, 11
graphical query language, 150–153
Graphlog, 369, 370
ground, 22
ground atom, 34
ground clause, 288
ground inference rule, 203
group by in SQL, 154
grouping, 533
GSM, 242–250
GYO algorithm, *130*, 136
GYO reduction, 141

hash index, 107
head of rule, 39, *41*, 276
Heraclitus, 614
Herbrand interpretations, 23
Herbrand model
datalog, 282
hierarchy model, 28, 97
homomorphism, 12
of tableau queries, *117*, 127, 136
Homomorphism Theorem, 37, 105, 115–118, *117*, 127, 136, 177, 178
Horn clause, 279
hyp, 18
hyperedge, 130
hypergraph, 130
acyclic, 132
articulation set, 132
connected, 132
cyclic, 132
of database schema, 130
ear, 130
edge, 130
GYO algorithm, 130

path, 132
reduced, 130
hyperplane, 438

$\mathbf{I}^1$, $\mathbf{I}^{1/2}$, $\mathbf{I}^0$, 387
$\mathbf{I}_*$,$\mathbf{I}^*$,$\mathbf{I}^*_*$, 391
idb, 42, 49, *277*
IDM transaction, 580–582, 613, 615–617
axiomatization, 581
condition, 580
deletion, 615
insertion, 615
modification, 615
optimization, 581
parallelization, 616
schedule, 616
serializability, 616
simplification rules, 582
IDM transactional schema, *584*, 613, 617
vs. constraints, 585–586
completeness, 617
soundness, 617
vs. fd's, 585
vs. inclusion dependencies, 585, 617
vs. jd's, 617
IFO, 242, 264
ILOG, 576
image of calculus query, *78*
immediate consequence operator, 282
imperative method, 564–566, 573
implementation
cross product, 108
equi-join, 107–108
multi-way join, 111–115
physical, 106–108
projection, 107
relational algebra, 107–108
selection, 107
implication
and chase, 180–182, 186
closed under, 204
closed under k-ary, 204
of dependencies, 158, 160, *164*, 195
in view, 221
of fd's and ind's, 192
finite, 197–199, 226
finite vs. unrestricted, 202, 219, 234
of functional dependencies, 186
of ind's, 192, 195–197
for two-element instances, 189
unrestricted, 197–199
vs. fd's and ind's, 199–202
implicational dependency (id), 233
implies. *See* implication.
finitely, 198
without restriction, 198
inclusion dependency (ind), 161, 192–211, *193*, 218, 253
acyclic, 208, 210, 211, 250
vs. implication, 210
axiomatization, 193–195, 211
vs. fd's, 192, 202–207, 211
and chasing, 208
independent of fd's, 250
key-based, 250, 260
logical implication, 192, 195–197
with fd's, 192, 199–202
repeats-permitted, 212
restricted classes, 192
satisfies, 193
typed, 211
vs. referential integrity, 211
vs. semantic data model, 207
vs. unrestricted implication, 199
incomplete database, 487–507
c-table, 493
update, 593–594
complexity, 499
fixpoint, 495
logical theory, 594–600
and nondeterminism, 507
table, 488
incomplete information
and update anomalies, 162
incremental update. *See* first-order incremental definability.
ind, 161; *See* inclusion dependency.
ind-rule in chasing, 208
independent component, 265
indexing, 106, 107
inequality atom
in selections, 69
inequality in constraint databases, 96
inference rule, 24, 158
ground, 202, 203
schema, 202
substitution, 167
inference rules
for fd's and mvd's, 172–173, 186
for functional dependency, 166–168, 186
for inclusion dependency, 193–195
proof using, 167
provable using, 167
for unary inds, 210, 215
vs. algorithm for testing implication, 166

inference rules *(continued)*
vs. axiom, 167
infinitary logic, 458, *459*, 462
infinite database, 97
finite representation, 36, 93–96, 97
infinite tree, 575
inflationary datalog$^{\neg}$, 356
inflationary fixpoint logic (CALC+μ^{+}), *352*, *353–354*
inflationary fixpoint operator (μ^{+}), 353
information capacity
relative, 265, 268–269
INGRES, 34, 111, 155
distributed, 135
query optimizer, 114–115, 127, 135, 137
inheritance, 546, *552*, 553, 567, 573–575, 577
semantic data model, 245
input schema of query, 37
insert in SQL, 149
insertion, 580
insertion anomaly, 162
instance
complex value, 512
database, 29
conventional perspective, 32
logic-programming perspective, 32
generalized, 95
GSM, 245
object-oriented database, *554*, 555
relation
conventional perspective, 32
logic-programming perspective, 32
relativized, 77
semantic data model, 245
unrestricted, 197
instantiation, 277
integrity constraint, 6, 28, 157, 186
vs. first-order logic, 186, 234
intended model, 279
intensional database (id6), 42, 49, *279*
intensional relation, 42, 48, *277*
interpretable formula, 77
interpretation, 23
active domain, 79
natural, 78
relativized, 74, 77–78
unrestricted, 78
intersection, 33
in relational algebra, 71
and SPC algebra, 55, 69
in SQL, 146
vs. join, 58
invented value, 469
IQL, 573
irreflexive, 11
ISA, 543, 545
semantic data model, 245
isomorphic tableau queries, 120
isomorphism, 12
OID, 555
iterate, 518
Iterative QSQ (QSQI), 339

Jacobi algorithm, 335
jd, 161, 169–173, 218. *See* join dependency
jd rule, in chasing, 175
join, 55,57
acyclic, 105, 126, 128–135, 136
algorithms for binary join, 135
complex value, 517
decomposition, 106, 114
equi-join, 55, 57, 108
implementation, 111–115
left-to-right evaluation, 112
lossless, 164, *253*
multi-way, 106, 108, 135
natural, 56, *57*, 169
vs. equi-join, 57
pairwise consistent, 128, 136
physical implementation, 107–108
semi-join, 128, 135
in SQL, 144
tuple substitution, 115, 135
vs. cross product, 58
vs. intersection, 58
vs. tableau, 64
join decomposition, 114–115
join dependency (jd), 161, 169–173, *170*, 218
acyclic, 169, 182–183, 186
and mvd's, 182
and chasing, 175
complexity of implication, 169
and decomposition, 169–171
embedded, 233
Gentzen-style axiomatization, 186
n-ary, 170
projected, 233
satisfies, 170
vs. axiomatization, 171, 186
vs. functional dependency, 169, 171, 178
vs. multi-valued dependency, 170, 182
vs. natural join, 169
vs. SPJR algebra, 181
vs. unrestricted implication, 199

join detachment, 114, 135
join tree, 130, 136

k-ary axiomatization, 202, 204
key, 257, 543
  attribute, 257
  in semantic data model, 247
key dependency, 163
  simple, 267
  vs. functional dependency, 161
key-based inclusion dependency, 250, 260
KL, 503
Knaster-Tarski's Theorem, 286

lambda-calculus, 574
language (formal), 13–20
late binding, 552
LDL, 337, 409, 533, 538, 613
  update language, 583
left-to-right evaluation
  datalog, 318
  join, 112
linear bounded Turing machine, 196
linear datalog, 305, 316
linear programming, 97
Lisp, 573
literal, 21
  in nr-datalog$^\neg$ rule, 72
local stratification, 411
logic. *See* mathematical logic.
  temporal, 612, 619
  three-valued, 389–391
logic programming, 97
  constraints, 97
  object-oriented database, 572
  vs. datalog, 35
logic-programming perspective on relations, 32, 33
Logical Data Model (LDM), 97
logical database, 503
logical implication, 21
  and chase, 180–182, 186
  closed under, 204
  closed under *k*-ary, 204
  of dependencies, 160, *164*, 193
    in view, 221
  of fd's, 165, 186
  of fd's and ind's, 192
  finite, 197–199
    vs. unrestricted, 202, 219, 234
  full dependencies
    complexity, 221
  of ind's, 192, 195–197
  of mvd's,172–173
  unrestricted, 197–199
logical level of three-level architecture, 106
logical theory and updates, 594
LOGSPACE complexity
  of first-order queries, 430–431
lossless join, 164, *253*
Löwenheim-Skolem theorem, 25

magic set rewriting, 311, *324–335*
  generalized supplementary, 325, 336
  original, 340
  vs. QSQ, 324, 327
main-memory buffering, 106, 107
many-sorted query language, 153–154
map, 540
map filter, 518
materizialized view, 51
mathematical logic, 20–27
matrix of formula, 82
maximum in SQL, 154
memo-ing, 335
message, 552
method, 543, 551
  languages, 563–571
method resolution, 546, 552
method schema, 563, *566–571*
  monadic, 543, 563, 565, *567*, 568, 577
  polyadic, *567*, 568, 577
mgu, 295
Microsoft Access, 36, 143, 150, 152–153, 155
minimal cover, 257
minimal tableau query, 118
minimization of tableau queries, 105, *119*, 136
minimum in SQL, 154
minimum model, 275
modal operator, 503
model, 24
  database, 28
  datalog, 279
  relational, 28–34
  semantic data, 243, 245–253, 267
modification, 580
modification anomaly, 162
modified RANF, 88
modus ponens, 24
monadic datalog program, 305
monadic method schema, 543, 563, 565, *567*, 568, 577
monoid, 199
monotone operator, 283

monotonic query, 42
monotonicity
  and conjunctive queries, 42
  and relational algebra, 71, 98
most general unifier (mgu), 293
multi-head dependency, 217
multi-way join
  decomposition, 114–115
  detachment, 114, 135
  implementation, 106, 108, 111–115, 135
  left-to-right evaluation, 112
  tuple substitution, 115, 135
multiset, 92, 136, 145
multivalued dependency (mvd), 161, 169–173, *170*, 186, 218
  and acyclic jd's, 182
  axiomatization with fd's, 172–173
  dependency basis, 172
  embedded, 218, 220, 233
  original definition, 189
  satisfies, 170
  and two-element instances, 189
  vs. functional dependency, 171
  vs. join dependency, 170
  vs. propositional logic, 189
mutual recursion, 315
mvd, *See* multivalued dependency.

N-datalog$^{\neg(\neg)}$, 463
N1NF. *See* nested relation.
NAIL!, 337, 409
naive evaluation
  of datalog, 312
  of SPC query, 109
naive table, 492
named perspective, 31, *32*
  and dependencies, 159
  projection, 57
  relational algebra, 71
  selection, 57
  SPJR algebra, 56–59, *57*
  tuple, 32
  vs. unnamed perspective, 32
named value, *554*, 556
  root of persistence, 556
natural interpretation, 78
natural join, 56, 57, 169
  polyadic, 58
  vs. equi-join, 57
  vs. join dependency, 169
natural semantics of relational calculus, *78*, 79
NC, 96, 431
negation, 36
  in Microsoft Access, 153
  pushing, 83
  in QBE, 150
  in selections, 68
  in SQL, 143
  stratified, 49
  vs. set difference, 70
Negation as Failure, 27, 406
negative literal, 288
nest, 518
nested loop implementation of join, 107, 108
nested relation, 512
  algebra, 519
nested SQL query, 143, 146–147
network model, 28, 97
new, 559
NF2. *See* nested relation.
no-information null, 502
non-existing null, 502
nondeterminism, 15
  semantics of negation, 409
nondeterministic query. *See* query, nondeterministic.
noninflationary datalog$^{\neg}$, 357
nonrecursive (nr) datalog
  with negation, 70, 72–73
    program, 72
nonrecursive datalog program, 62
normal form, 158
  Boyce-Codd (BCNF), 250,*251*
    decomposition algorithm, 255
  conjunctive (CNF), 83
  conjunctive calculus, 46–47
  disjunctive (DNF), 83
  domain-key, 265
  first, 265
  fourth (4NF), 252, *252*, 259
  nr-datalog, 68
  prenex (PNF), 82
  project-join (PJ/NF), 265, 267
  relational algebra (RANF), 86, 97
  relational schema, 251–259, 265
  safe-range (SRNF), 83
  SPC algebra, 55
  SPCU algebra, 62
  SPJR algebra, 59
  SPJRU algebra, 62
  third (3NF), 257
    decomposition algorithm, 257
    synthesis algorithm, 257
now, 607
NP, 18
NP-complete, 105, 121, 122, 127

NP-hard, 121
NPSPACE, 18
nr-datalog, 62
  normal form, 68
nr-datalog¬, 70, 72–73
  and domain independence, 78
  with equality, 72, 73
  equivalence to first-order languages, 80
  literal, 72
  program, 72
  query, 73
  range restricted, 72
    with equality, 72
  rule, 72
  semantics, 72
  translation into SQL, 147–149
  and undecidability, 122–126
NU-Prolog, 337
null value, 488

O_2, 562, 573
O_2SQL, 510, 536–537, 562
obj, 547
object, 246, 543, 545, *547*, 573
object creation, 573; *See* object-oriented database, object creation.
object equality, 557
object history, 615
object identifier (OID), 473, 543, *545–547*
  semantic data model, 243
object migration, 572, 613, 615
object-oriented data model, 28, 245, 477, 546
object-oriented database, 8, 242, 473, 542–578
  calculus, 557–558
  class, 545
  class hierarchy, 549
    well formed, 549
  classification, 572, 575
  completeness, 560–561, *560*, 574
  complex value, 545
  consistency, *See*, object-oriented database, type safety
  context-dependent binding, 552
  covariance, 553
  dangling reference, 999, 572
  dba mode, 546
  deductive, 575
  deep equality, 557, 575
  dereferencing, *557*, 558, 559
  determinate query, 559
  domain-inclusion semantics, 551
  dynamic aspect, 572
dynamic binding, 543, 546, *552*
encapsulation, 543, *546*, 553
expansion of value, 558
formal definition, 547–555
generic OODB model, 547–556
ILDG, 580
imperative methods, 564–566, 573
  expressive power, 565–566, 577
inheritance, 546, *552*, 553, 567, 573–575, 577
instance, *554*, 555
IQL, 573
ISA, 543, 545
languages for methods, 563–571
late binding, 552
logic programming, 572, 574
message, 552
method, 551
  signature, 551
  well formed, 553
method resolution, 546, 552
method schema, 563, 566–571
  expressive power, 569–571
  monadic, 543, 563, 565, *567*, 568, 577
  polyadic, *567*, 568, 577
named value, *554*, 556
object, 543, 545, *547*, 573
object creation, 558–562, 573, 574
object equality, 557
object identifier, 543, *545*, *547*
object migration, 572
OID assignment, 550
OID isomorphism, 555, 560
overriding, 546
parallelism, 573
query semi-deterministic, 574
query language, 556–563
querying schema, 572
reachability, 565
receiver, 552
role, 571
schema, 554
schema design, 571
specialization, 545
static binding, 552
subtyping relationship, 549
type, 548
  disjoint interpretation, 550
  semantics, 550
type safety, *563*, 565, 567, 573
user mode, 546
value, 547
value equality, 557
value-dependent binding, 552

object-oriented database *(continued)*
view, 571
object-oriented programming languages, 573
object-orienteddatabase
consistency. *See* object-oriented database,type safety.
ODE, *615*
OID
-assignment, 550
-equivalence, 246
-isomorphism, 246, 560
semantic data model, 243
OODB, 242; *See* object-oriented database.
Open World Assumption (OWA), 489, 497, 595
operator
continuous, 286
monotone, 283
OPS5, 369, 370
optimization
conjunctive queries, 36, 105
using chase, 163
using dependencies, 163
datalog, 36, 112, 311–337
and Exodus, 135
in practical systems, 105, 106–115
relational algebra, 106
transaction, 581
using chase, 177–180
or-sets, 505
ORACLE, 34, 155
ordered database, 397, 447
output schema of query, 37
overriding, 546
OWA, 489, 497, 595

$P(\mathbf{I})$, 280, 378, 383, 387
$pg(P, \mathbf{I})$, 389
P^{wf}, 390
page fetch, 107
page size, 106
paging protocol, 106
pairwise consistent join, 128, 136
Paradox, 152, 155
parallel complexity
classes of circuits, 431
of first-order queries, 431–433
parameterized IDM transaction, 584
call, 584
parametrized query, 522
paramodulation vs. chase, 186
parity query
not first-order, 460
not in *while*, 437
partial fixpoint logic (CALC+μ), *348*, *349–352*
partial fixpoint operator (μ), 349
partial order, 11
partially ordered set, 11
path in hypergraph, 132
PCP, 16
and satisfiability of relational calculus, 123
permutation, 13
physical implementation, 106–108
cross product, 108
equi-join, 107–108
projection, 107
relational algebra, 107–108
selection, 107
physical level
of three-level architecture, 106
physical model of relational database, 106–107
PNF, 82
polyadic
conjunction, 46, 75, 83
disjunction, 75, 83
existential quantification, 83
natural join, 58
polyadic method schema, *567*, 568, 577
polynomial inequalities constraint, 96, 97
positive existential calculus, 91, 97
decidability, 99
positive literal, 288
positive selection formula, 67
poss(T), 490
Post Correspondence Problem (PCP), 16
and satisfiability of relational calculus, 123
POSTGRES, 153, 600
powerset, 514
precedence graph
in datalog evaluation, 315
in datalog$^{\neg}$, 379
negative edge, 380
positive edge, 380
predicate, 277
prenex normal form (PNF), 82
procedural vs. declarative, 35, 53
product
Cartesian, 52
cross, 52, 54, 58, 108, 144
direct, 235, 240
production rule system, 369
program schema, 574
project-join expression
extended, 229
project-join normal form (PJ/NF), 267
project-join query, flat, 126

projection, 52
  and aggregate functions, 93
  named perspective, 57
  physical implementation, 107
  pushing, 109
  in SQL, 144
  unnamed perspective, 54
proof, 24
  using inference rules, 167
proof tree, 286
propositional calculus, 21
propositional logic, 21
  vs. fd's and mvd's, 186, 189
PSPACE, 17
PSPACE complexity
  of while queries, 437
PSPACE-complete, 196
$P_{\mathbf{I}}$, 286
PTIME, 17
PTIME complexity
  of fixpoint queries, 437
pure universal relation assumption (URA), 126, 130, 242, 252
pushing
  negation, 83
  projection, 109
  selection, 109, 335

q_{adom}, 79
$q_{\mathbf{d}}(\cdot)$, 78
$q_{nat}(\cdot)$, 78
QC, 422
QL, 477
QPTIME, 406, 422
QSQ, 311, *317–324*, 335
  annotated, 330
  completion, 318
  Iterative (QSQI), 339
  Recursive (QSQR), 323–324
    algorithm, 324
  template, 319–320
  vs. magic set rewriting, 324, 327
QSQI, 339
QSQR, 323–324
  algorithm, 324
Quel, 74, 112, 155
query, 421
  complexity, 422–423
    data complexity, 422–423
    expression complexity, *422–423*, 463
  composition, 48–52, 71
  computability, 417–421
  conjunctive, 36, 37–64
  conjunctive calculus, 44–47
  containment relative to dependencies, 37, 177
  definite, 97
  determinate, 474
  equivalence, 37
    relative to dependencies, 176, 177
  first-order, 70
  genericity, 419–421, *419*, 425
    C-genericity, 419–420
  input schema, 37
  with invented values, 469
  monotonic, 42
  nondeterministic, 453–457
    CALC+$\mu^{(+)}+W$, 456
    choice operator, 458
    dynamic choice operator, 464
    N-datalog$^{\neg(\neg)}$, 463
    while$^{(+)}+W$, 454, 456
    witness operator, 454–456
  nr-datalog$^{\neg}$, 73
  optimization, 36, 105–115, 112, 313–339
  output schema, 37
  parametrized, 522
  project-join, flat, 126
  relational calculus, 75
  satisfiable, 42
  schema query, 572
  semi-deterministic, 574
  statistical properties, 106
  tableau, 43–44, *43*
  union-of-tableaux, 139
  untyped, 475
  vs. implementation, 110
  vs. query mapping, 37
  vs. update, 28
  well-typedness, 417
  yes-no, 42
query composition, 37
query decomposition, 114–115
query evaluation
  cost model, 106, 108–110
  naive, 109
  in practical systems, 106–115
query evaluation plan, 107, 108, 110, 135
  and Exodus, 111
  generating, 110–111
  parameterized, 135
query language
  aggregate operators, 153, 154, 155
  with arithmetic, 153, 154
  associative, 35
  BP-completeness, 428

query language *(continued)*
completeness, 466
completeness in a class, 424
conjunctive queries, 36, 37–64
with union, 36, 37, 38
constraint, 94–98
declarative, 29, 558
vs. procedural, 35, 53
determinate-completeness, 474
disjunction, 37, 38
dominated by ($\sqsubseteq$), 47
embedded, 466
C+SQL, 466
while$_N$, 467
equivalence ($\equiv$), 47
expressive power, 106, 427
graphical, 150–153
inflationary semantics, 342–344
many-sorted, 153–154
navigational, 558
noninflationary semantics, 342–344
object-oriented, 556–563
practical, 143–155
relational algebra, 28, 35, 36
relational calculus, 28, 35, 36
set-at-a-time, 35
static analysis, 36, 105, 122–126, 306–311
temporal, 606–613
three paradigms, 35–36
Query Management Facility (QMF), 155
query mapping vs. query, 37
query optimization, 36, 105
cost model, 106, 108–110
distributed database, 128
evaluation plan, 107, 108, 110–111, 135
and Exodus, 111
in INGRES, 114–115
join detachment, 114, 135
local vs. global, 115, 117
and negation, 106
in practical systems, 106–115
program transformation, 108
query rewriting, 108–110
query tree, 108–110, *108*
and relational calculus, 126
rewrite rule, 110
and sampling, 111
in System R, 112–114
by tableau minimization, 118–120
tuple substitution, 115, 135
using chase, 163, 177–180
using dependencies, 163
query rewriting, 108–110
query tree, 108–110, *108*
Query-By-Example (QBE), 36, 40, 43, 143, 150–152, 155
condition box, 150
and domain independence, 153
and first-order languages, 151
negation, 150
relationally complete, 151
view definition, 151
vs. tableau queries, 150
Query-Subquery (QSQ), 311, *317–324*, 335
annotated, 330
completion, 318
Iterative (QSQI), 339
Recursive (QSQR), 323–324
algorithm, 324
template, 319–320
vs. magic set rewriting, 324, 327

$R[\cdot]$, 31
r.e.. *See* recursively enumerable.
Rado graph, *442*, 461
RANF, 86, 97
algorithm, 88
modified, 88
range restricted
algorithm, 84
calculus query, 97
calculus variable, 83, 84
conjunctive query, with equality, 41, 48
formula, 102
nr-datalog$^\neg$, 72
with equality, 65, 72
rule, 41
range separable query, 97
rank, 402
RDL, 369, 370
real closed field, 96, 97
receiver, 552
reconstruction mapping, 254
rectangle, representation, 95
rectified subgoal in datalog evaluation, 328, 330–331, 336
recursive (formal) language, 16
Recursive QSQ (QSQR), 323–324
algorithm, 324
recursively enumerable, 16
reduced hypergraph, 130
redundancy and update anomalies, 162
referential integrity constraint vs. inclusion dependency, 161, 213
reflexive relation, 10

refutation, 290
regular language, 14
regular tree, 558, 575
relation
  complex value, 512
  extended, 229
  extensional, 42, 48
  intensional, 42, 48
relation (instance), 29
  conventional perspective, 32
  logic-programming perspective, 32, 33
  over empty attribute set, 32
  unrestricted, 197
relation atom, 112, 217
relation schema, 31
  with dependencies, 241
relational algebra, 28, 35, 36, 70, 71, 81
  aggregate operators, 97
  with bags, 136
  complement operator, 103, 104
  composition, 71
  conjunctive, 52–61
  division, 99
  and domain independence, 78
  equivalence to first-order languages, 80
  equivalences, 106
  implementation, 106, 107–108
  and monotonicity, 71, 98
  named, 64, 71
  named conjunctive, 56–59
  optimization, 106, 126
    in practical systems, 105, 106–115
  physical implementation, 106–115
  and satisfiability, 98
  semi-join, 128, 135
  SPC, 52–56, 108, 118
  SPCU, 62, 97, 136
  SPJR, 56,–59, 118
    vs. join dependency, 181
  SPJRU, 62
  translation into calculus, 80
  typed restricted SPJ, 156
  and undecidability, 122–126
  unnamed, 71
  unnamed conjunctive, 52–56
  unrestricted, 103
  untyped algebra, 475
relational algebra normal form (RANF), 86, 97
  algorithm, 88
  modified, 88
relational calculus, 28, 35, 36, 64, 70, 73–91, *85*
  active domain semantics, 74, *79*
  aggregate operators, 97
  allowed query, 97, 101–102
  base formula, 74
  conjunctive, 45
  conjunctive normal form (CNF), 83
  and counting, 154
  disjunctive normal form (DNF), 83
  domain calculus, 39, 74
  domain independence, 70, 74, 75–77, *79*, 81–97
  equivalence to first-order languages, 80
  evaluable query, 97
  formula, 74–75
    equivalence, 82
    parse tree, 83
  image of query, 78
  inequalities constraint, 96, 97
  natural semantics, *78*, 79
  negation, 70–71
  polynomial inequalities constraint, 96
  positive existential, 68, 91, 97
  prenex normal form (PNF), 82
  query, 75
  and query optimization, 126
  range restricted
  range separable query, 97
    algorithm, *84*
    formula, 102
    query, 97, 102
    variable, 83, 84
  relational algebra normal form (RANF), 86, 97
  relativized interpretation, 74, 77–78
  rewrite rule, 82
    for RANF, 86–87
    for SRNF, 83
  safe DRC query, 97
  safe query, 64, 97
  safe-range, 81, 85, *83–85*, 97
    normal form (SRNF), 83
  safety, 70, 75–77
  and satisfiability, 123
  semantics, relativized, 77
  simulation of PCP, 123
  static analysis, 105, 122–126
  syntax, 74
  translation into algebra, 97
    active domain case, 80
    safe-range case, 81, 86–91
  tuple calculus, 39, 74, 101
  and undecidability, 36, 97, 105, 122–126, 136
  unrestricted semantics, 78
  unsafe, 75
  vs. first-order logic, 77, 105, 123, 136
  vs. select-from-where clause, 145
relational completeness, 96

relational completeness *(continued)*
QBE, 151
SQL, 147
vs. Turing computability, 96
relational model, 28–34
relative information capacity, 265, 268–269, 539
relativized instance, 77
relativized interpretation, 74, 77–78
relevant fact, 317
relname, 31
renaming
attribute, 58
complex value, 517, 524
operator, 57, *58*
SPJR algebra, 57
$rep(T)$, 489
repeat restricted tableau query, 67
representation system
strong, 489
weak, 490
representative instance, 263
resolution, 186, 552
vs. chase, 186
resolution theorem proving, 136
resolvent, 289, 294
RETE, 600
Reverse-Same-Generation (RSG)
program, 312
query, 317
revision vs. update, 599–600
rewrite rule
conjunctive calculus, 46
normal form vs. query optimization, 110
for optimization, 108, 110
relational calculus, 82
SRNF, 83
sound, 56
SPC algebra, 55–56, 110
SPJR algebra, 110
SRNF to RANF, 86–87
rewriting, query, 108–110
role, 571
root of persistence, 556
rule, 41
active database, 605
anonymous variable, 39
body, 39, *41*
head, 39, *41*
nr-datalog$^{\neg}$, 72
range restricted, 72
semantics, 72
range-restricted, 41
semantics, 41
update language, 582
rule-based conjunctive query, 39, 40–42, *41*
with equality, 48
semantics, 41
with union, 62
rule-goal graph, 335
running intersection property, 141

safe, 64
DRC query, 97
query, 97
safe-range, 85
and aggregate functions, 93
complex value, 528
normal form (SRNF), 83
query, 97
relational calculus, 81, *83–85*
and universal quantification, 85
safety, 70, 75–77
in SQL, 153
Same-Generation (SG)
program, 331
query, 331
Variant (SGV), 339
sampling in query optimization, 111
$sat(\mathbf{R}, \Sigma)$, $sat(\Sigma)$, 174
satisfaction, 24
conjunctive calculus formula, 46
relative to a domain, 77
satisfaction family, 174, 186, 222
satisfiability
and conjunctive queries, 42
datalog, 300
and first-order queries, 123
and relational algebra, 71, 98
and relational calculus, 123
satisfiable formula, 21
satisfiable query, 42
satisfiable SPC algebra, 56
satisfiable SPJR algebra, 59
satisfy
dependency, 160
by tableau, 175
functional dependency, 163
inclusion dependency, 193
join dependency, 170
multivalued dependency, 170
saturated set, 188
scalar domain, 153
schema
complex value, 512
database, 29, *31*

object-oriented database, 554
query, 572
relation, 31
schema design
decomposition, 162, 251–259, *252*
object-oriented database, 571
synthesis, 257–258
SDD-1, 135
select-from-where clause, 112, 144
vs. projection, 144
vs. relational calculus, 145
selection, 52, 57
constant based, 66
named perspective, 57
physical implementation, 107
positive conjunctive, 55, 58
pushing, 109, 335
in SQL, 144
unnamed perspective, 53
selection formula
atomic, 53
disjunction, 62
inequality atom, 69
with negation, 68
positive, 67
positive conjunctive, 55, 58, 108
selection rule, 298
Semantic Binary Data Model, 264
semantic data model, 28, 157, 192, 240, 242–250, 264, 542
abstract class, 243
attribute, 243
multi-valued, 243
single-valued, 243
class, 243
complex value, 243
derived data, 246
Entity-Relationship (ER), 242
and functional dependencies, 249–253
generic (GSM), 242
inheritance, 245
instance, 245
ISA, 245
object identifier (OID), 243
printable class, 243
and rfelational model, 249–253
and schema design, 247–250
subclass, 243
vs. inclusion dependencies, 207, 251–253
semantics
conjunctive calculus, 45
conjunctive query, 41
nr-datalog$^\neg$ program, 72
nr-datalog$^\neg$ rule, 72
relational calculus
active domain, 79
natural, *78*, 79
unrestricted, 78
rule-based conjunctive query, 41
SPC algebra, 54
SPJR algebra, 58
tableau query, 43
semi-deterministic query, 574
semi-join, 128, 135
program, 129
seminaive datalog evaluation, *312–316*, 335
basic algorithm, 315
improved algorithm, 316
semipositive datalog, 377
sentence, 23
Sequel, 144
set comprehension, 538
set constructor, 508, 509
set difference, 68
in relational algebra, 71
and SPCU algebra, 136
vs. negation, 70
set membership, 514
set-at-a-time, 35
set_create, 515
set_destroy, 515
sideways information passing, 111, 112–114
in datalog evaluation, 318, 336, 340
graph, 113, 340
strategy, 113
signature, method, 551
simple key dependency, 267
simple tableau query, 140
simultaneous induction, 351
single rule programs (sirups), 305, 309
single-head dependency, 217
singleton, 518
sip graph, 113, 340
sip strategy, 113
sirup, 305–309
SLD datalog evaluation, 289–298
SLD-AL, 335
SLD-resolution, 295; *See* datalog, SLD-resolution.
datalog$^\neg$, 406
SLD-tree, 298, 317
SLDNF resolution, 406
SLS resolution, 409
sort
complex value, 511
of instance, 32
of relation name, 31

sort *(continued)*
of tuple, 32
sort(·), 31
sort set dependency, 191
vs. axiomatization with fds, 213
sort-merge implementation of join, 108
sound axiomatization, 167
spatial database, 95
SPC algebra, 52–56, *54*, 108
base query, 54
generalized, 55
intersection, 55, 69
normal form, 55
rewrite rule, 55–56, 110
satisfiable, 56
unary singleton constant, 54
with union, 62
vs. SPJR algebra, 60
vs. tableau queries, 118
SPCU algebra, 62, 97
and dependencies in views, 222
and difference, 136, 140
normal form, 62
specialization, 545
SPJ algebra,
typed restricted, 64, 67
SPJR algebra, 56–59, *57*
base query, 58
generalized, 59
natural join, 56
normal form, 59
renaming, 57
rewrite rule, 110
satisfiable, 59
unary singleton constant, 58
with union, 62
vs. join dependency, 181
vs. SPC algebra, 60
vs. tableau queries, 118
SPJRU algebra, 62
normal form, 62
SQL, 2–3, 36, 70, 74, 112, 143–150, 155, 336, 370, 372, 536, 574
bags, 145, 155
and conjunctive queries, 143–146
contains, 146
count, 154
create, 145
delete, 149
distinct, 145, 154
and domain independence, 153
duplicate tuples, 144
from, 144
group by, 154
insert, 149
and negation, 143
nested query, 143–147
in personal computer DBMSs, 152
relationally complete, 147, 150
safety, 153
scalar types, 145
select, 144
set operators, 146
simulation of nr-datalog$^\neg$, 147–149
translation to algebra, 112
update, 149
update language, 580
views, 149
vs. cross product, 144
vs. first-order queries, 147–149, 155
vs. relational calculus, 145
vs. Sequel, 144
where, 144
SRNF, 83
stable model, 408, *413*
stage(P, $\mathbf{I}$), 285
Starburst, 368, 370
static analysis
conjunctive queries, 105, 115–122
datalog queries, 306–311
first-order queries, 105, 122–126
of queries, 36
relational calculus, 105, 122–126
static binding, 552
stored data, statistical properties, 106
stratified datalog$^\neg$, 378
stratified negation, 49
stratified semantics, 377–385. See datalog$^\neg$, stratified semantics
stream of tuples, 106, 135
strongly-safe-range
complex value, 530
structured object. *See* complex value.
Structured Query Language (SQL), 143. *See* SQL
subclass, 545
semantic data model, 243
subquery
in datalog evaluation, 318
substitution, 24, *116*
vs. valuation, 116
subsumption, 136
subtyping relationship, 549
succ, 397
sum, 91, 92
in SQL, 154
summary of tableau query, 43

superkey, 257
supplementary relation, 319–320
supported model, *384*, 411
sure(T), 490
surrogate, 247, 573
Sybase, 155
symmetric, 10
synthesis, 257–258
  vs. decomposition, 258, 265
System R, 111
  query optimizer, 112, 113–114, 122, 127, 135, 137

T_P, 375
table, 488–500; *See* Codd-table, naive table, c-table.
tableau, 43
  complexity, 121–122
  composition, 226–227
  embedding, 43
  typed, 44
  vs. dependencies, 218, 234
  vs. join, 64
tableau minimization, 105, 118–120, 136
  and chasing, 177–180
  vs. condensation, 136
  vs. local optimization, 117
  vs. number of joins, 118
  vs. resolution theorem proving, 136
tableau query, 43–44, *43*
  chasing, 173, 186
  complexity, 111–122
  composition, 226
  containment, 121–122
  difference, 64
  with equality, 48
  of an fd, 181
  homomorphism, *117*, 127, 136
  isomorphic, 120
  of a jd, 181
  minimal, 118
  minimization, 119
  repeat restricted, 67
  semantics, 43
  simple, 140
  summary, 43
  typed, 64, 121, 136
  union-of-tableaux query, 63, 64, 139
  vs. dependencies, 64
  vs. QBE, 150
  vs. SPC algebra, 118
  vs. SPJR algebra, 118
tagged dependency, 164, 221, 241
Tarski's Algebraization Theorem, 96
Taxis, 264
taxonomic reasoning, 572, 575
template dependency, 233, 236
temporal constraint, 611–613
  history-less checking, 615
temporal database, 95, 606–613
  query language, 607–611
    deductive, 610
    TSQL, 609
  representation, 608–609
  temporal CALC, 607
  temporal constraint, 611–613
    on events, 612, 615
    object histories, 615
    object migration, 613
    vs. transactional schemas, 612
  time domain, 607
    now, 607
  transaction time, 607
  transition constraint, 612
    dynamic fd's, 615
    pre/post conditions, 615
  valid time, 607
temporal logic, 608, 615
temporal query language, 607–611
term, 22, 34
  complex value, 519
tgd, 217–228
tgd-rule in chasing, 223
third normal form (3NF), 257
3-T_P, 388
3-satisfiability, 19
3NF, *257*
3NF Algorithm, 257
3-valued instance, 386, *387*, 388, 389
three-level architecture, 3
  logical level, 106
  physical level, 106
3-SAT, 139
TI Open Object-Oriented Data Base, 135
timestamp, 401
top-down datalog evaluation, 316–324
  vs. bottom-up, 311, 327, 336
topological sort, 11
total instance, 387
total order, 11
total program, 395
T_P, 283
transaction time, 607
transactional schema, 584–586, *584*, 617
  Gen(**T**), 585
  IDM transactional schema, 584, 613, 617

transactional schema *(continued)*
parameterized IDM transaction, 584
vs. constraints, 585–586
completeness, 585
soundness, 585
vs. methods, 584
vs. temporal constraints, 612
transformation rule.See rewrite rule.
transition constraint, 612
transitive, 10
transitive closure query
generalized, 310
not first-order, 436
tree, 12
truth assignment, 21
TSQL, 609
tup_create, 514
tup_destroy, 515
tuple, 29
free, 33
generalized, 94, *95*
named perspective, 32
with placeholders, 94
unnamed perspective, 32
tuple calculus, 74, 101
vs. domain calculus, 39
tuple generating dependency (tgd)
full, 218
tuple rewriting, 107
tuple substitution, 115, 135
tuple-generating dependency (tgd), 217–228
Turing machine, 15
linear bounded, 196
two-element instances
vs. fd's and mvd's, 189
two-way automata, 15
type in object-oriented database, 548
type safety, *563*, 565, 567, 573
typed dependency, 159
vs. faithful, 233
vs. untyped, 217
typed inclusion dependency, 211
typed restricted SPJ algebra, 64, 67, 156
typed tableau, 44
query, 64, 121, 136
types(*C*), 548

unary inclusion dependency (uind), 207, 210–211
undecidability
of properties of datalog queries, 306, 308
of properties of first-order queries, 105, 122–126
of implication for embedded dependencies, 220, 234
of implication for emvds, 220
of implication of fds and inds, 199, 211
underlying domain, 74
unfounded set, 413
unification, 293
uniform containment, 304
union, 33, 37, 38
in conjunctive queries, 61–64
in Microsoft Access, 153
in relational algebra, 71
in rule-based conjunctive queries, 62
in SQL, 146
union-of-tableaux query, 63, 64, 139
unique name axioms, 26
unique role assumption, 261
unirelational dependency, 217
unit clause, 288
universal quantification
removing, 83
and safe-range, 85
vs. existential quantification, 74
universal relation
assumption (URA), 137, 266
pure, 126, 130, 242, 252
weak, 261–264, *262*
interface, 266
scheme assumption (URSA), 260
unique role assumption, 261
universe, 23
universe of discourse, 77
Unix, 155
unknown value, 488
unnamed perspective
on relations, 32
projection, 54
relational algebra, 71
selection, 53
SPC algebra, 52–56, *54*
tuple, 32
vs. named perspective, 32
unnest, 518
unrestricted instance, 197
unrestricted interpretation, 78
unrestricted logical implication, 197–202, 219
vs. finite, 197
vs. functional dependency, 199
vs. inclusion dependency, 199
vs. join dependency, 199
unrestricted relational algebra, 103
unrestricted semantics of relational calculus, 78
untyped dependency, 192

vs. typed, 217
untyped relational algebra, 475
update
in SQL, 149–150
statistical properties, 106
vs. revision, 599–600
vs. query, 28
update in SQL, 149
update anomalies, 162, 241
and incomplete information, 162
and redundancy, 162
update language, 580–583
completeness, 583
IDM transaction, 580–582, 615–617
deletion, 615
insertion, 615
modification, 615
rule-based, 582–583
datalog$^{\neg\neg}$, 582
Dynamic Logic Programming (DLP), 583, 613
LDL, 583
SQL, 580
URA, 126, 130, 137
pure, 242, 252
weak, 261–264, *262*
URSA, 260
user view. See view.

V-relation, 513
val(O), 547
valid, 21
valid model semantics, 409
valid time, 607
valuation, 41
as syntactic expression, 45
of tableau, 43
vs. substitution, 116
value equality, 557
var, 33, 41
variable, 33
anonymous, 39, 44
bound occurrence, *45*, 75
free occurrence, *45*, 75
variable assignment, 24
variable substitution
rewrite rule, 46, 83
view, 4
complement, 583
and dependencies, 222
maintenance, 586–588, *586*
materialized, 51
object-oriented database, 571
in QBE, 151
and query composition, 51–52
in SQL, 149
update, *586*, 589–593
complement of views, 591–593
virtual, 51

weak instance, 262
weak universal model, 502
weak universal relation assumption (URA), 261–264, *262*
well-formed formula
conjunctive calculus, 45
relational calculus, 74–75
well-founded semantics, 385–397
where in SQL vs. selection, 144
while, 344–346, *345*
while queries, 342, *367*
normal form, 452–453
on ordered databases, 447
PSPACE complexity, 437
vs. fixpoint queries, 453
while^{+}, *346*, 346–347
while$^{(+)}+W$, 456
while$^{(+)}+W$, *454*
while$_N$, 467
completeness on ordered databases, 468
while$_{new}$, 469
completeness, 470–473
not determinate-complete, 474
well-behaved, 470
while$_{obj}$, *559*
while$_{uty}$, 475
completeness, 478
well-behaved, 477
witness operator, 454–456
word problem for monoids, 199

yes-no query, 42

0-1 law, 441
for CALC, 441–444
for *while*, 444–446