

COPYRIGHTED MATERIAL

Index

Index

SYMBOLS

- “ (double quote) characters, quoted identifiers, 78–79
- @ (at) character, T-SQL variable names, 239
- [and] square brackets, quoted identifiers, 78–79
- _ (underscore) character, table name avoidance, 16

A

- A.C.I.D (atomicity/consistency/isolation/durability), transactions, 140–141**
- Abstract Factory model, ADO.NET 2, 35–37**
- accent sensitivity, query results ranking, 421**
- `Accumulate()` method, aggregate functions, 322
- acronyms, naming convention avoidance, 15**
- Active Directory**
 - provider shortcomings, 201
 - versus user properties, 221–226
- ActiveX Data Objects (ADO), development history, 8–10**
- `AddNewCustomer` procedure, Service Broker, 382
- ADF (application definition file), Notification Services, 351–352**
- ADO (ActiveX Data Objects), development history, 8–10**
- ADO.NET**
 - `Command` object, 11
 - `Connection` object, 11
 - constants/enums listing, 549–584
 - data types, 109–116
 - `DataAdapter` object, 11
 - `DataReader` object, 11
 - `DataSet` object, 11
 - `DataSets`, 32
 - development history, 10–11
 - MARS (Multiple Active Result Sets) support, 9
- ADO.NET 2.0**
 - Abstract Factory model, 35–37
 - asynchronous commands, 266–272
 - database-independent code, 35
 - `DataSet` optimizations, 32
 - development history, 11–12
 - Generic Factory Model, 42–47
 - interfaces, 35–36
 - legacy API support, 40–42
 - schema metadata system, 67–77
- `AdsCommand` class, data providers, 211–219
- `AdsConnectionStringBuilder` class, data providers, 209–211
- `AdsDataAdapter` class, data providers, 228–231
- `AdsDataReader` class, data providers, 219–228
- `AdsFactory` class, data providers, 231–232
- `AdsParameterCollection` class, data providers, 212
- `AdsProvider` property, data providers, 205, 206
- aggregate functions, CLR (Common Language Runtime) Objects, 321–323**
- `ALERT QUEUE` statement, message queues, 378
- alerts, Ingres, 494**
- anchors, queries, 253–254**
- APIs, legacy support issues, 40–42**
- application definition file (ADF), Notification Services, 351–356**

applications, instance configuration file, 351–352

architecture

- CLR (Common Language Runtime) Objects, 328–329
- Firebird, 486–487
- Generic Factory Model providers, 44–45
- legacy data access model, 43–44

archives, data deletion protection method, 35

arrays, Oracle, 526–529

ASP.NET

- asynchronous operations, 272–276
- change notification, 539–543
- SqlCacheDependency object, 293–295, 538–543

ASP.NET 1.x, data binding, 163–164

ASP.NET 2.0

- data binding, 163–164
- Data Source controls, 164–169
- Data Source window elements, 192–196
- Web applications, 430–432

ASSEMBLYPROPERTY, CLR (Common Language Runtime) Objects, 311

AsyncCallbackObject() method, asynchronous commands, 268–270

asynchronous commands. See also commands

- AsyncCallbackObject(), 268–270
- BeginExecuteNonQuery() method, 267–269
- BeginExecuteReader() method, 269–271
- BeginExecuteXMLReader() method, 271–272
- callback objects, 269–270
- described, 266–267
- EndExecuteNonQuery() method, 268–269
- IO Completion ports, 267
- SqlCommand class support, 267–272

asynchronous communications, Service Broker, 370

asynchronous operations

- ASP.NET, 272–276
- IAsyncResult interface, 275–276

at (@) character, T-SQL variable names, 239

attributes

- SqlProcedure, 318–319
- SqlUserDefinedAggregate, 322

automatic change tracking, Full-Text Search population method, 406

B

Backup class, SMO utility class, 330

Backup() object, SQL Server Management Objects (SMOs), 338–340

backup/restore, SQL Server Management Objects (SMOs), 338–341

backups

- MaxDB, 480
- queues, 378

batch updating, ADO support, 9

BCP (Bulk Copy Program), data migration, 29

BeginExecuteNonQuery() method, asynchronous commands, 267–269

BeginExecuteReader() method, asynchronous commands, 269–271

BeginExecuteXMLReader() method, asynchronous commands, 271–272

BeginTransaction() method, data providers, 209

BerkeleyDB

- concurrency, 503
- data stores, 503
- database elements, 502–504
- development history, 502
- fine-grained locking system, 503
- in-memory transaction logs, 503
- installation, 504
- key-value pairs, 503
- logically atomic transactions, 503
- .NET interaction, 504–507
- resource information, 508
- single-master replication, 503
- Sleepycat Public License, 502
- small footprint, 503
- SQL compliance, 502
- transactional data store, 503
- write-ahead logging, 503
- XML layer, 503

BFile data type, Oracle, 543

BI (Business Intelligence), described, 427–428

BIGINT data type, SQL Server, 100

BINARY data type, SQL Server, 102–106

binary serialization, DataTables, 92–93

binary storage, data types, 102–106

BindingList control, Windows Forms, 198

BindingNavigator control, Windows Forms, 198

BindingSource control, Windows Forms, 197–198

BIT data type, SQL Server, 107

BLOBs, Full-Text Search, 422–424

Boolean fields, naming conventions, 16

Boolean values

- BIT data type, 107
- OrderByColumnsInSelect property, 71
- BuildUpdateCommands() method, data providers, 231
- bulk copy operations, SQL Server, 291–292
- Bulk Copy Program (BCP), data migration, 29
- Business Intelligence (BI), described, 427–428

C**C# language**

- business object class, 181–182
- data class, 180–181
- null coalescing operator, 113
- nullable types, 113–114
- reachability, 244
- sealing classes, 203
- transaction processing, 242–243

caching

- Data Source controls, 167–169
- page-level, 168–169
- user-control, 168–169

callback objects, asynchronous commands, 269–270

`Cancel()` method, data providers, 215

capture execution, SQL Server Management Objects (SMOs), 330**CAS (Code Access Security), XML/ADO.NET integration, 127–128****case sensitivity, `QuoteIdentifierCase` property, 72****catalog views, Service Broker, 396–399****catalogs, Full-Text Search index setup, 408****change notification**

- Oracle, 538–543
- XML/ADO.NET integration, 133–134

change tracking, Full-Text Search population methods, 407

`ChangeDatabase()` method, data providers, 209

`ChangePassword()` method, password management, 281

channels

- delivery, 350
- instance configuration file, 352

CHAR (character) data type, SQL Server, 98**character data storage, data types, 98–99****character sets, Firebird, 487****chronicles, generator component, 350****classes**

- `AdsCommand`, 211–219
- `AdsConnection`, 203–209
- `AdsConnectionStringBuilder`, 209–211
- `AdsDataAdapter`, 228–231
- `AdsDataReader`, 219–228
- `AdsFactory`, 231–232
- `AdsParameterCollection`, 212
- `AdsProvider`, 202
- `ConfigurationManager`, 54
- `ConnectionStringBuilder`, 53–60
- `ConnectionStringCollection`, 54

data provider, 202

`DataSourceEnumerator`, 232

`DataTableReader`, 93

`DbConnection`, 202

`DirectorySearcher`, 212

`Helper`, 48–51

Inference, 132–133

instance, 330

Microsoft Oracle provider, 517–518

`ObjectDataSource` control, 180–182

ODPNET, 518–519

Oracle provider, 516–519

`OracleConnection`, 523

`OracleConnectionStringBuilder`, 523

`OracleDependency`, 538–539

`OracleException`, 533–534

provider-independent, 44–45

sealing, 203

SMO utility, 330

`SqlBulkCopy`, 291–292

`SqlBulkCopyColumnMapping`, 291–292

`SqlBulkCopyColumnMappingCollection`, 291–292

`SqlCacheDependencyAdmin`, 297

`SqlCommand`, 267–272

`SqlConnectionStringBuilder`, 281–283

`SqlDataSourceEnumerator`, 278–279

`SqlDependency`, 297–300

`SqlTriggerContext`, 311–313

`Utils`, 357–359, 362

`XmlReader`, 121–124

`XmlSchemaSet`, 131

`XmlWriter`, 121–124

`ClearAllPools()` method, connection pooling, 280

`ClearPool()` method, connection pooling, 280

client applications, Firebird, 487**client-side transactions, 1.x Framework, 145–146**

`Close()` method, data providers, 208, 228

CLR (Common Language Runtime) Objects

advantages, 303

aggregate functions, 321–323

architecture, 328–329

business logic placement, 328–329

column updates, 313

complex operations, 329

context connections, 327–328

creating, 304–306

DDL event triggers, 313–314

debugging process, 309–310

deployment methods, 307–308

CLR (Common Language Runtime) Objects (continued)

CLR (Common Language Runtime) Objects (continued)

- design goal code, 316–317
- functions, 310–311, 319–324
- mapping methods, 307
- permissions, 309
- procedural programming, 303
- results output, 318
- scalar-valued functions, 319–320
- scientific calculations, 329
- security calculations, 329
- set theory, 302–303
- set-based programming, 302–303
- SQL Server integration, 301
- SQL Server projects, 306–310
- SqlCLRObjects assembly, 307
- SqlContext object, 327
- SqlPipe object, 318
- SqlProcedure attribute, 318–319
- SqlTriggerContext class, 311–313
- stored procedures, 318–319
- sys.assemblies table, 310
- sys.assembly_files table, 310
- sys.assembly_modules table, 310
- sys.assembly_preferences table, 310
- sys.assembly_types table, 310
- sys.sql_modules table, 310
- sys.types table, 310
- tables, 310–311
- tabular results, 326–327
- triggers, 311–317
- versus T-SQL objects, 329
- TVFs (table-valued functions), 320–321
- user-defined aggregate functions, 323–324
- user-defined types, 324–326

clustering, Ingres, 493

CODASYL (Conference on Data Systems Language), development history, 2–3

Code Access Security (CAS), XML/ADO.NET integration, 127–128

CodeBeside page, ReportServer, 446

Column() object, SQL Server Management Objects (SMOs), 335

columns

- foreign key naming conventions, 16
- naming conventions, 16
- primary key definitions, 19–20
- primary key naming conventions, 16
- Service Broker queue tables, 394–395
- SQL Server Management Objects (SMOs), 334–336

- updating, 313
- user properties versus Active Directory properties, 222–225

combined modes, MySQL, 472

Command object, query storage, 11

commands. See also asynchronous commands; objects

- asynchronous, 266–272
- CLR mapping methods, 307
- CreateCommand(), 77
- DataReaders(), 87
- DbCommand(), 77–78
- DDL CREATE ENDPOINT, 344–345
- DROP ENDPOINT, 344–345
- ExecuteScalar(), 83 96
- insert, 218
- named parameters, 80
- output parameters, 84–85
- positional parameters, 80
- provider-specific, 78
- provider-specific parameters, 80–81
- quoted identifiers, 78–79
- return codes, 85–87
- scalars, 87
- SqlCommand(), 78
- System.Data.CommandBehavior(), 82–83

CommandText property, data providers, 213

commitment, data access factor, 2

Common Language Runtime (CLR) Objects, SQL Server, 301–328

Common Table Expressions (CTEs), T-SQL, 252–254

communications, Server Broker, 369

compilers, legacy API support issues, 40–42

complex operations, CLR (Common Language Runtime) Objects, 329

CompositeIdentifierSeparatorPattern, schema metadata system, 69–70

concurrency, BerkeleyDB, 503

Conference on Data Systems Language (CODASYL), development history, 2–3

configuration files

- connection string storage, 54
- XML/ADO.NET integration, 120

Configuration() object, SQL Server Management Objects (SMOs), 332

Configuration Wizard

- ObjectDataSource control, 182–184
- SQLDataSource control configuration, 170–174
- TableAdapter method connection, 187–189

- ConfigurationManager class, retrieving connection strings, 54**
- Connection object, establishing/maintaining connections, 11**
- Connection property, data providers 213**
- connection objects**
- constructors, 204
 - MARS (Multiple Active Result Sets), 276–278
- connection pooling, SQL Server, 279–280**
- connection strings**
- configuration file storage, 54
 - ConnectionStringBuilder class, 53–60
 - ConnectionStringCollection class, 54
 - creation methods, 53–54
 - database, 306
 - dynamic construction, 57
 - provider-specific string parameters, 57–60
 - provider-specific types, 55–57
 - retrieving, 54
 - SqlDataSource method, 431–432
 - TableAdapter method, 187
- connections**
- context, 327–328
 - database opening/closing, 60–62
 - exception error information collection, 62–66
 - exception management methods, 62–66
 - failover notification, 526
 - Oracle, 523–526
 - provider-specific issues, 66–67
 - status checking, 60–62
- ConnectionString properties, data providers, 204–205**
- ConnectionStringBuilder class, building connection strings, 53–60**
- ConnectionStringCollection class, connection strings, 54**
- ConnectionTimeout property, data providers, 205**
- constants, ADO.NET listing, 549–584**
- constraints**
- MySQL, 473–474
 - PostgreSQL, 498
- constructors**
- AdsDataAdapter class, 229–230
 - connection object, 204
- consumers**
- data access, 6–12
 - OLE-DB workflow process, 5–6
- contacts**
- SOAP, 447–448
 - URL, 439–446
- CONTAINS function, Full-Text Search, 412–413**
- CONTAINSTABLE function, Full-Text Search, 415**
- context connections, CLR (Common Language Runtime) Objects, 327–328**
- contracts, Service Broker, 370, 377–378**
- controls**
- BindingList, 198
 - BindingNavigator, 198
 - BindingSource, 197–198
 - Data Source, 164–169
 - multi-version concurrency, 499
 - ObjectDataSource, 176–185
 - SQLDataSource, 170–175
 - UI, 197–198
- conversions, XML/ADO.NET integration, 123–124**
- coordination, Service Broker element, 370**
- CREATE AGGREGATE command, CLR mapping method, 307**
- CREATE ASSEMBLY statement, assembly deployment, 307**
- CREATE ENDPOINT command, SQL Server, 344–345**
- CREATE FUNCTION command, CLR mapping method, 307**
- CREATE PROCEDURE command, CLR mapping method, 307**
- CREATE TRIGGER command, CLR mapping method, 307**
- CREATE TYPE command, CLR mapping method, 307**
- Create() method, SMO Index() object, 336**
- CreateCommand() objects, Generic Factory Model, 77**
- CreateDbCommand() method, data providers, 208–209**
- CreateDbParameter() method, data providers, 215**
- CreateParameter() method, adding parameters to an object, 79–80**
- CreatePermission() method, data providers, 232**
- CTEs (Common Table Expressions), T-SQL, 252–254**
- currency, data types, 102**
- CURSOR data type, SQL Server, 108**
- cursors, server-side, 480**
- customer sign-up, Service Broker service, 383–386**
- CustomerInfoSchema, Service Broker, 376**
- ## D
- DAO (Data Access Objects), development history, 6–7**
- data access layer**
- consumers, 6–12
 - development history, 2–4
 - industry support, 2
 - performance factor, 1

data access layer (continued)

- reliability, 1
- vendor commitment, 2

Data Access Objects (DAO), development history, 6–7

data binding

- ASP.NET 1.x, 163–164
- ASP.NET 2.0, 163–164
- data source caching, 167–169
- Data Source controls, 164–169
- data sources, 160–162
- DataSource control generation, 189–190
- DBDirect methods, 186–187
- defined, 159
- ObjectDataSource control, 176–185
- one-way versus two-way, 163
- parameter data validation, 166
- parameter passing, 165–167
- provider-specific syntax issues, 175–176
- SQLDataSource control, 170–175
- TableAdapter method, 185–189
- typed DataSets, 185–186
- Windows Forms, 191–198
- Windows Forms versus Web applications, 159–160

data concurrency, two-way data binding issues, 163

Data Definition Language (DDL)

- event triggers, 313–314
- Full-Text Search index setup, 407–408

data exchange, XML/ADO.NET integration, 119

data migration, SELECT statement, 29

data providers. See also providers

- Active Directory shortcomings, 201
- ADO.NET registration, 233
- AdsCommand class, 211–219
- AdsConnection class, 203–209
- AdsConnectionStringBuilder class, 209–211
- AdsDataAdapter class, 228–231
- AdsDataReader class, 219–228
- AdsFactory class, 231–232
- AdsParameterCollection class, 212
- AdsProvider class, 202
- classes, 202
- connection object constructors, 204
- connection string fields, 203–204
- DataSourceEnumerator class, 232
- deleting records, 217–218
- DirectorySearcher class, 212
- indexers, 219–221
- Oracle, 515–516
- parameter non-support reasons, 212

- permissions, 232
- reader (select) methods, 216
- tables, 212
- updating records, 217–218
- user objects, 212
- user properties versus Active Directory properties, 221–226

Data Source caching, when to use, 168–169

Data Source Configuration Wizard, data sources, 193–196

Data Source controls

- ASP.NET 2.0 data binding, 164
- data caching, 167–169
- generating, 189–190
- parameter passing, 165–167
- Visual Studio 2005, 169–170

Data Source window, interface elements, 192–196

data sources

- data binding, 160–162
- data retrieval, 81
- Data Source Configuration Wizard, 193–196
- distributed transactions, 145–154
- listing available, 51–53
- Reporting Services, 436–437
- schema metadata system information, 69–73
- Windows Forms, 192–196

data storage, legal issues, 35

data stores, BerkeleyDB, 503

Data Transformation Services (DTS), data migration, 29

data types

- ADO.NET, 109–116
- BFile, 543
- BIGINT, 100
- BINARY, 102–106
- BIT, 107
- Boolean, 97
- CHAR (character), 98
- CLR (Common Language Runtime) Objects, 324–326
- CURSOR, 108
- DATETIME (TIMESTAMP), 100
- DECIMAL (DEC and DEC (p, s)), 101–102
- FLOAT (FLOAT and DOUBLE PRECISION), 101
- floating-point, 101
- IMAGE, 102–103
- INT (integer), 97, 99
- legacy support issues, 40–41
- MONEY, 102
- NCHAR (national character storage), 99
- .NET Framework, 109–116

- NTEXT, 99
- nullable types, 113–114
- NUMERIC, 101
- NVARCHAR (national character storage), 99
- NVARCHAR (MAX), 99
- Oracle, 523
- PostgreSQL, 498
- REAL (FLOAT (24)), 101
- schema metadata system, 69
- Service Broker queue tables, 394–395
- SMALLDATETIME, 100
- SMALLINT, 100
- SMALLMONEY, 102
- SQL Server, 98–108
- SQL_VARIANT, 107–108
- SqlTypes namespace, 109–113
- string, 97
- TABLE, 108
- TEXT, 99
- TIMESTAMP, 107
- TINYINT, 100
- UNIQUEIDENTIFIER, 108
- VARBINARY, 102–103
- VARBINARY (MAX), 102–103
- VARCHAR (char varying/character varying), 98
- VARCHAR (MAX), 99
- XML, 108
- DataAdapter **object, data retrieval, 11**
- DataAdapter () **object, DataSet population method, 89–90**
- Database **property, data providers, 205, 206**
- database connection strings, SQL Server projects, 306**
- database integration, Service Broker element, 370**
- Database () **object, SQL Server Management Objects (SMOs), 332, 334–335**
- databases. See also open-source databases**
 - CODASYL definition, 2
 - context connections, 327–328
 - creating in SQL Server Management Objects (SMOs), 332–334
 - data deletion protection methods, 35
 - data normalization, 24–30
 - design guidelines, 30–35
 - flat, 28–29
 - indexes, 22
 - normalized design, 26
 - old/new design migration, 34–35
 - opening/closing connections, 60–62
 - planning process, 13–14
 - quality data assurance, 26–27
 - relational, 3–4
 - reports, 23–24
 - security enforcement, 23
 - Service Broker setup, 372–373
 - SMO backup/restore, 338–341
 - strongly typed business objects, 33–34
 - views, 22–24
 - XML Web services, 343–345
- DataReaders () **object**
 - DataSet population method, 90
 - forward-only read capability, 11
 - value returns, 88
 - when to use, 95
- DataSet () **object, disconnected data storage, 11**
- DataSets**
 - binary serialization, 92–93
 - data filling, 82
 - data representation, 88
 - DataAdapter () **object, 89–90**
 - DataReaders () **object, 90**
 - in-memory database representations, 32
 - manual population methods, 89
 - serialization, 92–93
 - when to use, 95–96
- DataSource **property, data providers, 205**
- DataSourceEnumerator **class, data providers, 232**
- DataSourceProductName **class, schema metadata system, 70**
- DataTableReader () **class, data representation, 93**
- DataTables**
 - data filling, 82
 - data representation, 90
 - DataTableReader () **class, 93**
 - DataView () **object, 91–92**
 - Oracle fills, 529–531
 - RowState () **object, 91**
 - serialization, 92–93
 - streaming data, 93–94
- DataView () **object, DataTables, 91–92**
- date/time, data types, 100**
- DATETIME (TIMESTAMP) **data type, SQL Server, 100**
- DateTime **fields, naming conventions, 16**
- DbCommand **object**
 - calling from a DbConnection object, 78
 - calling from Generic Factory Model, 77
 - provider independent class, 44–45
- DbConnection **class**
 - DbCommand object calls, 78
 - metadata retrieval, 202
 - provider-independent class, 44–45

DbConnection property, **data providers, 213**
DbDataSourceEnumerator() **method, listing available data sources, 52–53**

DBDirect methods, data binding, 186–187

DbParameter, **provider independent class, 44–45**

DbTransaction **property, data providers, 214–215**

DDL (Data Definition Language)

event triggers, 313–314

Full-Text Search index setup, 407–408

DDL CREATE ENDPOINT **command, SQL Server, 344–345**

DDL triggers, T-SQL, 263–264

debugging

CLR (Common Language Runtime) Objects, 309–310

XSL, 125–126

DECIMAL (DEC and DEC(p, s)) **data type, SQL Server, 101–102**

Delete() **method, removing subscribers, 359**

delivery channels

distributor component, 350

instance configuration file, 352

delivery protocols, distributor component, 350

Dense_Rank() **function, T-SQL, 248**

Depth **property, data providers, 226**

device list, subscription management application, 360–363

DeviceInfo **object, ReportServer, 453–460**

devicelist.aspx **file, subscription management application, 360**

dialogs, Service Broker component, 370

direct execution, SQL Server Management Objects (SMOs), 330

DirectorySearcher **class, data providers, 212**

disconnected data access, ADO support, 9

Dispose() **method, data providers, 208, 228**

distributed data, Ingres, 493

Distributed Transaction Coordinator, performance monitoring, 149–150

distributed transactions

client-side, 145–146

flow-through, 156

IDBDataAdapter **object, 150–153**

local data effects, 150–153

versus local transactions, 141

multiple data source spanning, 145

nested transactions, 153–155

performance monitoring, 149–150

System.Transactions namespace, 146–149

TransactionScope **object, 146–149**

Web services, 155–156

workflow process, 147–149

distributors, Notification Service components, 350

double quote (") characters, quoted identifiers, 78–79

DROP ENDPOINT **command, SQL Server, 344–345**

DTS (Data Transformation Services), data migration, 29

E

e-mail, sending service, 386–389

EnableSubscriber() **method, adding/removing subscribers, 358–359**

EndExecuteNonQuery() **method, asynchronous commands, 268–269**

enumerators

ADO.NET listing, 549–584

SQL Server, 278–279

error codes, Firebird, 488

error tracing, ODP.NET, 534–535

errors

exception information collection, 62–66

Oracle, 533–534

severity levels, 63–64

event batches, Notification Services, 349, 367–368

event providers, Notification Services, 349

events

application definition file, 353, 356

DDL triggers, 263–264, 313–314

firing, 367–368

InfoMessage, 244–246

Notification Services, 349, 353, 367–368

ObjectSourceData control, 178–179

OnRowUpdating, 230

Page_Load, 358

SqlDependency **class, 298**

StateChange, 207

XML change notification, 133–134

exception codes, Firebird, 488

exception handling, T-SQL, 240–249

exceptions

error information collection, 62–66

management methods, 62–66

OracleException **class, 533–534**

provider-specific interaction, 64–65

EXECUTE AS function, Reporting Services, 429

ExecuteNonQuery() **method**

data providers, 217–218

data source data retrieval, 81

ExecuteReader() **method, search results display, 81–82**

`ExecuteScalar()` **method**

data providers, 216

single value return, 83, 96

execution, SQL Server Management Objects (SMOs), 330

explicit locking, MaxDB, 480–481

explicit schemas, MaxDB, 480

expressions

IdentifierPattern property, 71

Oracle, 536–538

ParameterMarkerPattern property, 71

ParameterNamePattern property, 71

QuoteIdentifierPattern property, 72

extended stored procedures (XP), SQL Server, 301–302

extensibility

Generic Factory Model, 46

PostgreSQL, 498

external events, Notification Services, 349

EXTERNAL_ACCESS permission, CLR (Common Language Runtime) Objects, 309

F

failover notification, Oracle, 526

FieldCount property, data providers, 226

fields

Boolean naming conventions, 16

connection string, 203–204

DateTime, 16

files

ADF (application definition file), 351–352

application definition, 352–356

CodeBeside, 446

devicelist.aspx, 360

instance configuration, 350–352

machine.config, 46

filters

defined, 404

Full-Text Search, 425–426

FindAll() method, data providers, 219

Firebird

character sets, 487

Classic Server architecture, 486–487

client applications, 487

database elements, 485–488

development history, 485

error codes, 488

exception codes, 488

foreign keys, 487

generators, 488

installation, 488–489

InterBase Public License, 485

.NET integration, 489–491

record locking, 485–486

record versioning, 485–486

referential integrity, 487

resource information, 491

selectable stored procedures, 487

SQL compliance, 485

Super Server architecture, 486–487

supported platforms, 487

UDFs (user-defined functions), 487

fixed-length storage, CHAR (CHARACTER) data type, 98

flags, BINARY data type, 103–106

flat databases, normalization, 28–29

FLOAT (FLOAT and DOUBLE PRECISION) data type, SQL Server, 101

floating-point, data types, 101

flow-control statements, T-SQL, 239–240

flow-through transactions, distributed transactions, 156

foreign keys

column naming conventions, 16

Firebird, 487

MaxDB, 478

MySQL, 473

PostgreSQL, 498

primary key relationships, 20–21

SQL Server Management Objects (SMOs), 337–338

ForeignKey() object, SQL Server Management Objects (SMOs), 337–338

formatters, distributor component, 350

forms

customer sign-up, 383–386

normalization types, 25

ReportServer, 445

Service Broker, 381–383

FORMSOF object, Full-Text Search, 416

FREETEXT function, Full-Text Search, 414–415

FREETEXTTABLE function, Full-Text Search, 415–416

Full Text Indexing Wizard, Full-Text Search index setup, 409–411

full-text catalog, defined, 404

full-text index, defined, 404

Full-Text Search

automatic change tracking population, 406

BLOBs search, 422–424

CONTAINS function, 412–413

CONTAINSTABLE function, 415

Full-Text Search (continued)

- filters, 425–426
- FORMSOF object, 416
- FREETEXT function, 414–415
- FREETEXTTABLE function, 415–416
- full population, 406
- incremental timestamp-based population, 407
- indexing setup, 407–412
- INFLECTIONAL parameter, 417
- ISABOUT object, 417
- language settings, 418–419
- manual change tracking population, 407
- multi-column queries, 418
- NEAR operator, 417
- properties, 411–412
- queries, 412–418
- query-linked servers, 418
- results rankings, 419–422
- structure elements, 405–406
- THESAURUS parameter, 417
- WEIGHT function, 417
- XML search, 422

full-text searching, described, 403–404

functions. See also methods

- CLR (Common Language Runtime) Objects, 310–311, 319–324
- CONTAINS, 412–413
- CONTAINSTABLE, 415
- Dense_Rank(), 258
- EXECUTE AS, 429
- FREETEXT, 414–415
- FREETEXTTABLE, 415–415
- NTile(), 258–259
- Oracle expressions, 537
- PIVOT, 254–256
- RANK(), 257
- READFAST, 262
- TABLESAMPLE, 259–262
- UNPIVOT, 256
- WAITFOR, 262–263
- WEIGHT, 417

G

- GenerateDBDirectMethods **property, data binding, 186–187**

- generational terms, Full-Text Search query, 413**

generators

- Firebird, 488
- Notification Services, 349–350

Generic Factory Model

- command parameter adding, 79–80
- CreateCommand() object, 77
- default parameters/values, 58–59
- default providers, 46
- exception catching methods, 64
- extensibility, 46
- Helper class, 48–51
- pros/cons, 47
- provider architecture, 44–45
- provider-agnostic platform, 42
- provider-independent classes, 44–45
- GetDataSources() **method**
 - data providers, 232
 - listing available data sources, 51–52
- GetDevices() **method, device list display, 360, 362**
- GetEnumerator() **method, data providers, 228**
- GetName() **method, data providers, 228**
- GetNsApplication() **method, subscription list display, 364**
- GetNsInstance() **method**
 - subscriber list retrieval, 357–359
 - subscription list display, 364
- GetOrdinal() **method, data providers, 228**
- GetSchemaTable() **method, data providers, 228**
- GetSqlValues() **method, SqlTypes, 289**
- GetSubscriber() **method, subscribers, 359–360**
- GetSubscription() **methods, subscription list display, 363–364, 366**
- GetValues() **method, data providers, 228**
- Globally Unique Identifiers (GUIDs), primary key avoidance, 19–20**
- GroupByBehavior **enumerator, schema metadata system, 70**

H

- HasRows **property, data providers, 226**

- Helper class, writing, 48–51**

hosted event providers

- application definition file, 353
- Notification Services, 349

- Hungarian notation, naming convention, 15**

I

- IAsyncResult interface, asynchronous operations, 275–276

- IDBDataAdapter **object, distributed transactions, 150–153**

- IdentifierCase, **schema metadata system, 70**
 - IdentifierPattern **property, schema metadata system, 71**
 - identifiers**
 - IdentifierCase property, 70
 - quoted, 78–79
 - UNIQUEIDENTIFIER data type, 108
 - IDisposable **interface, data providers, 207**
 - IMAGE **data type, SQL Server, 102–103**
 - implicit locking, MaxDB, 480–481**
 - implicit schemas, MaxDB, 480**
 - IMS (Information Management System), development history, 3**
 - incremental timestamp-based, Full-Text Search population method, 407**
 - independent event providers, Notification Services, 349**
 - Index() **object, SQL Server Management Objects (SMOs), 335–336**
 - indexers**
 - data providers, 219–221
 - Full-Text Search setup, 407–412
 - indexes**
 - database search performance enhancements, 22
 - SQL Server Management Objects (SMOs), 334–336
 - indexing engine, performance/speed enhancements, 94–95**
 - industry support, data access factor, 2**
 - Inference **class, XML/ADO.NET integration, 132–133**
 - INFLECTIONAL **parameter, Full-Text Search, 417**
 - inflectional terms, Full-Text Search query, 413**
 - InfoMessage **event, T-SQL, 244–246**
 - Information Management System (IMS), development history, 3**
 - Information() **object, SQL Server Management Objects (SMOs), 332**
 - Ingres**
 - alerts, 494
 - CATOSL (CA Trusted Open Source License), 492
 - clustering, 493
 - database elements, 493–494
 - development history, 492
 - distributed data, 493
 - installation, 494–495
 - key range table partitioning, 493
 - Knowledge Manager, 493
 - .NET integration, 495–496
 - Oracle support, 493
 - page sizing, 494
 - parallel query execution, 493
 - read-ahead caching, 493
 - replication, 493
 - resource information, 496–497
 - row-level locking, 494
 - SQL compliance, 493
 - supported platforms, 494
 - user-defined extensions, 493
 - inheritance, PostgreSQL, 498**
 - Init() **method, aggregate functions, 322**
 - INITIATOR **service, Service Broker contracts, 377–378, 380–381**
 - in-memory transaction logs, BerkeleyDB, 503**
 - insert command, data providers, 218**
 - instance classes, SQL Server Management Objects (SMOs), 330**
 - instance configuration file, Notification Services, 350–352**
 - instance methods, ObjectDataSource control support, 180**
 - instances, Notification Services, 350–352**
 - INT (integer) **data type, SQL Server, 97, 99**
 - interfaces**
 - ADO.NET 2, 35–36
 - IAsyncResult, 275–276
 - IDisposable, 207
 - ISqlReader, 320–321
 - IO Completion ports, asynchronous commands, 267**
 - ISABOUT **object, Full-Text Search, 417**
 - IsClosed **property, data providers, 226**
 - IsCompleted **property, asynchronous operations, 270–271**
 - IsDBNull() **method, data providers, 228**
 - isolation levels, transactions, 142–143**
 - ISqlReader **interface, table-valued functions, 320–321**
 - ItemDeleted **event, change notification, 133–134**
 - ItemDeleting **event, change notification, 133–134**
 - ItemInserted **event, change notification, 133–134**
 - ItemInserting **event, change notification, 133–134**
 - ItemUpdated **event, change notification, 133–134**
 - ItemUpdating **event, change notification, 133–134**
- ## J
- JavaScript, ReportServer validation, 444–445**
 - joinbacks, queries, 253–254**
 - joins**
 - MaxDB, 478
 - self, 253
 - SupportedJoinOperators property, 72–73

K

key range table partitioning, Ingres, 493

key-value pairs, BerkeleyDB, 503

L

labels, DataSourceProductName property, 70

languages, Full-Text Search settings, 418–419

legal issues, data storage, 35

licensing, open-source software issues, 467–468

lists, subscriber retrieval, 357–358

Load() method, streaming data, 93–94

local transactions

creating, 143–145

versus distributed transactions, 141

locking

BerkeleyDB, 503

MaxDB support, 480–481

logically atomic transactions, BerkeleyDB, 503

loops, CURSOR data type, 108

M

machine.config file, default providers, 46

manual change tracking, Full-Text Search population method, 407

MARS (Multiple Active Result Sets)

ADO.NET support, 9

connection objects, 276–278

materialized views, PostgreSQL, 498

MaxDB

backups, 480

Database Analyzer, 479

database clustering, 480

database elements, 479–481

Database Manager utility, 479

development history, 477–478

foreign keys, 478

hot standby configuration support, 480

implicit/explicit schemas, 480

installation, 481–483

isolation levels, 480–481

locking, 480–481

nested subtransactions, 479

.NET interaction, 483–484

outer joins, 478

query tools, 479

referential integrity, 478

resource information, 484

server-side cursors, 480

SQL interaction, 478–479

SQL modes, 478

SQL Studio, 479

SQLCLI, 279

stored procedures, 478–479

subtransactions, 479

Synchronization Manager, 480

temporary tables, 480

triggers, 478–479

updateable views, 479

WebDAV (Web Distributed Authoring and Versioning), 480

WebSQL, 479

measurement units, column naming conventions, 16

Merge() method, aggregate functions, 322

message locking, Service Broker element, 370

message ordering, Service Broker element, 370

message queues, 378–380

message types, Service Broker, 370, 374–377

message validation, Service Broker, 374–377

methods. See also functions

Accumulate(), 322

AdsDataReader class, 226–228

AdsFactory class, 231–232

AsyncCallbackObject(), 268–270

BeginExecuteNonQuery(), 267–269

BeginExecuteReader(), 269–271

BeginExecuteXMLReader(), 271–272

BeginTransaction(), 209

BuildUpdateCommands(), 231

Cancel(), 215

ChangeDatabase(), 209

ChangePassword(), 281

ClearAllPools(), 280

ClearPool(), 280

Close(), 208, 228

Create(), 336

CreatedbCommand(), 208–209

CreatedbParameter(), 215

CreateParameter(), 79–80

CreatePermission(), 232

DbDataSourceEnumerator(), 52–53

Delete(), 359

Dispose(), 208, 228

EnableSubscriber(), 358–359

EndExecuteNonQuery(), 268–269

ExecuteNonQuery(), 81, 217–218

- ExecuteReader(), 81–82
- ExecuteScalar(), 216
- FindAll(), 219
- GetDataSources(), 51–52, 232
- GetDevices(), 360, 362
- GetEnumerator(), 228
- GetName(), 228
- GetNsApplication(), 364
- GetNsInstance(), 357–359, 364
- GetOrdinal(), 228
- GetSchemaTable(), 228
- GetSqlValues(), 289
- GetSubscriber(), 359–360
- GetSubscription(), 363–364, 366
- GetValues(), 228
- Init(), 322
- IsDBNull(), 228
- legacy support issues, 40–41
- Load(), 93–94
- Merge(), 322
- MoveNext(), 221
- NextResult(), 228
- ObjectDataSource control, 183
- Open(), 206–207
- OracleDataReader class, 531
- OUTPUT, 249–250
- Page_Load(), 362, 365–366
- PopulateDevice(), 362
- PopulateSubscription(), 365–366
- Prepare(), 215
- PrepareCommand(), 215, 219
- QuoteIdentifier(), 78–79
- Read(), 219
- ReadAsObject, 122
- reader (select), 216
- Render(), 450–451
- ROW_NUMBER(), 256–257
- Script(), 342
- Send(), 318
- SqlCommand class, 267–272
- SqlDependency class, 298
- SqlType data retrieval, 287–290
- TableAdapter, 185–189
- Terminate(), 322–323
- TryGetValue(), 210–211
- UnQuoteIdentifier(), 78–79
- WriteFromObject, 122
- Microsoft Full-Text Engine for SQL Server (MSFTESQL), described, 405**
- Microsoft Oracle provider**
 - classes, 517–518
 - connection string properties, 524
- Microsoft Transaction Server (MTS), Oracle services, 543–544**
- Microsoft Visual Studio 2005, XML requirement, 118**
- Microsoft.SqlServer.Server namespace, **SQL Server, 306**
- models, MySQL support, 470**
- modes**
 - MaxDB, 478
 - MySQL, 470–472
- MONEY data type, SQL Server, 102**
- MoveNext() method, **data providers, 221**
- MSFTESQL (Microsoft Full-Text Engine for SQL Server), described, 405**
- MTS (Microsoft Transaction Server), Oracle services, 543–544**
- multi-column queries, Full-Text Search, 418**
- Multiple Active Result Sets (MARS)**
 - ADO.NET support, 9
 - connection objects, 276–278
- multiple recordsets, ADO support, 9**
- multi-version concurrency control, PostgreSQL, 499**
- MySQL**
 - combined modes, 472
 - database elements, 473–475
 - development history, 468–469
 - foreign keys, 473
 - installation, 475
 - invalid data handling, 473
 - licensing, 468–469
 - mode descriptions, 470–472
 - models, 470
 - .NET interaction, 475–476
 - online documentation, 469
 - primary keys, 473
 - resource information, 477
 - shorthand modes, 472
 - SQL compliance issues, 469–473
 - storage engines, 469–470
 - stored procedures, 472–473
 - transaction support, 470
 - triggers, 473
 - unique constraints, 473–474
 - views, 473

N

named parameters, command objects, 80

namespaces

- providers, 43–44
- ReportServer, 440–442
- SQL Server 2005, 306

naming conventions

- acronym use, 15
- Boolean fields, 16
- columns, 16
- Date/Time fields, 16
- foreign keys, 16
- Hungarian notation, 15
- measurement units, 16
- numeric columns, 16
- Pascal Case, 15
- primary keys, 16
- rules, 15–16
- space character use, 16
- stored procedures, 17–19
- table elements, 15–17
- triggers, 17
- underscore (_) character use, 16

national character storage, NCHAR/NVARCHAR data types, 99

NCHAR (national character storage) data type, SQL Server, 99

NEAR operator, Full-Text Search, 417

nested transactions, distributed transactions, 153–155

.NET Data Provider, SQL Server communications, 265

.NET Framework

- BerkeleyDB interaction, 504–507
- data types, 109–116
- data type/SQL Server mapping, 114–116
- Firebird integration, 489–491
- Ingres integration, 495–496
- MaxDB interaction, 483–484
- MySQL interaction, 475–476
- Oracle database extensions, 544
- PostgreSQL integration, 500–501
- SQLite integration, 510–511

.NET Framework 1.0, .Net Data Provider, 265

.NET Framework 2.0

- legacy API support issues, 40–42
- version display, 41

networks, SQL Server enumeration, 278–279

NextResult() method, data providers, 228

noise words

- defined, 404
- query results ranking, 421–422

nonhosted event providers

- application definition file, 353
- Notification Services, 349

normalization

- data views, 22–23
- flat databases, 28–29
- form types, 25
- performance considerations, 30
- real world application, 25
- reasons for, 24

Notification Services

- ADF (application definition file), 351–352
- application definition file, 352–356
- application types, 347
- components, 348–350
- development history, 347
- distributors, 350
- event providers, 349
- firing events, 367–368
- generator components, 349–350
- instance configuration file, 350–352
- SQL instance, 356–357
- subscription management application, 348–349, 357–367

NTEXT data type, SQL Server, 99

NTile() function, T-SQL, 258–259

numbers

- data types, 99–102
- table names, 15

NUMERIC data type, SQL Server, 101

numeric columns, naming conventions, 16

NVARCHAR (MAX) data type, SQL Server, 99

NVARCHAR (national character storage) data type, SQL Server, 99

O

Object Linking and Embedding Database (OLE-DB), development history, 5

object serialization, XML/ADO.NET integration, 122–123

ObjectDataSource control, data binding, 176–185

objects. See also commands

- Backup(), 338–340
- callback, 269–270
- CLR versus T-SQL, 329

- Column(), 335
- Command, 11
- Connection, 11
- context connections, 327–328
- DataAdapter, 11
- Database(), 332, 334–335
- DataReader, 11
- DataSet, 11, 32
- DeviceInfo, 453–460
- ExecuteScalar(), 96
- ForeignKey(), 337–338
- FORMSOF, 416
- IDBDataAdapter, 150–153
- Index(), 335–336
- Information(), 332
- ISABOUT, 417
- ObjectDataSource control implementation, 182–183
- Restore(), 340–341
- Scripter(), 341–343
- scripting, 330
- Sequence, 532–533
- Server(), 331
- ServerConnection(), 331
- set-based programming, 302–303
- SMO creation, 331–332
- SqlCacheDependency, 293–297, 538–543
- SqlContext, 327
- SqlDataRecord, 326–327
- SqlDbType, 290–291
- SqlMetadata, 326–327
- SqlPipe, 318
- strongly typed, 33–34
- SubscriberDevice, 362
- SubscriberEnumeration, 357–359
- SubscriptionEnumeration, 363–364
- Table(), 334–335
- Transaction, 142
- TransactionScope, 146–149
- TriggerContext, 327
- URN referencing, 331
- User(), 336–337
- WMI Provider, 330
- ObjectSpaces, Visual Studio Longhorn release, 136**
- ODBC (Open Database Connectivity), development history, 4**
- ODP.NET (Oracle Data Provider for .NET)**
 - classes, 518–519
 - array binding, 526–529
 - connection string properties, 524, 525
 - Oracle data provider, 516
 - OracleDependency class, 538–539
 - provider classes, 516–517
 - tracing, 534–535
- OLE-DB (Object Linking and Embedding Database), development history, 5**
- one-way data binding, versus two-way data binding, 163**
- OnRowUpdating event, data providers, 230
- Open Database Connectivity (ODBC), development history, 4**
- Open Source Initiative (OSI), definition, 468**
- Open() method, data providers, 206–207
- OpenFTS, PostgreSQL downloadable add-on, 498**
- open-source databases. See also databases**
 - BerkeleyDB, 502–508
 - Firebird, 485–492
 - Ingres, 492–497
 - MaxDB, 477–484
 - MySQL, 468–477
 - PostgreSQL, 497–502
 - SQLite, 508–512
- open-source software, licensing issues, 467–468**
- operators**
 - NEAR, 417
 - TOP, 250–252
- Oracle**
 - array binding, 526–529
 - BFile data type, 543
 - change notifications, 538–543
 - connection string properties, 524–525
 - connections, 523–526
 - data providers, 515–516
 - data types, 523
 - DataTable fills, 529–531
 - error tracing, 534–535
 - errors, 533–534
 - expression functions, 537
 - failover notification, 526
 - Ingres integration, 493
 - large object support, 523
 - Microsoft Oracle provider, 517–518
 - MTS (Microsoft Transaction Server), 543–544
 - .NET database extensions, 544
 - .NET stored procedures, 544–547
 - ODPNET, 516
 - OracleConnection class, 523
 - OracleConnectionStringBuilder class, 523

Oracle (continued)

- OracleDataReader class, 531
- OracleException class, 533–534
- packages, 535–536
- parameter binding, 526
- provider classes, 516–519
- regular expressions, 536–538
- row insertion, 532–533
- Sequence database object, 532–533
- vendor-specific code avoidance, 516
- Visual Studio.NET developer tools, 519–533

Oracle Data Provider for .NET (ODP.NET), classes, 518–519

Oracle Explorer, Visual Studio.NET, 519–522

OracleConnection class, Oracle connections, 523

OracleConnectionStringBuilder class, Oracle connections, 523

OracleDataReader class, methods, 531

OracleDependency class, ODP.NET, 538–539

OracleException class, Oracle errors, 533–534

OrderByColumnsInSelect property, schema metadata system, 71

OSI (Open Source Initiative), definition, 468

outer joins, MaxDB, 478

output, CLR stored procedure results, 318

output formats, application definition file, 353–354

OUTPUT method, T-SQL, 249–250

output parameters, value returns, 84–85

output validation, XML/ADO.NET integration, 131–132

P

packages, Oracle definitions, 535–536

Page_Load() method

- adding/removing subscribers, 358–359
- device list editing, 362
- subscription list editing, 365–366

page-level caching, when to use, 168–169

paging, ObjectDataSource control support, 179

parallel query execution, Ingres, 493

ParameterMarkerFormat property, schema metadata system, 71

ParameterMarkerPattern property, schema metadata system 71

ParameterNameMaxLength property, schema metadata system, 71

ParameterNamePattern property, schema metadata system, 71

parameters

- adding to an object, 79–80
- application definition file, 352–355
- BeginExecuteReader() method, 269
- data provider non-support reasons, 212
- data validation, 166
- INFLECTIONAL, 417
- instance configuration file, 351
- named, 80
- ObjectDataSource control, 183–184
- Oracle, 526
- output, 84–85
- passing to Data Source controls, 165–167
- positional, 80
- provider-specific, 80–81
- Render() method, 450–451
- rs namespace, 440–441
- THESAURUS, 417
- URL ReportServer, 443–446

partitioning, key range table, 493

Pascal Case, naming conventions, 15

passwords

- ChangePassword() method, 281
- SQL Server management, 280–283

performance, data access factor, 1

permissions

- CLR (Common Language Runtime) Objects, 309
- data providers, 232
- Service Broker contracts, 377–378

pgAdmin III, PostgreSQL add-on tool, 499

PIVOT function, T-SQL, 254–256

PL/Perl language, PostgreSQL support, 499

PL/pgSQL language, PostgreSQL support, 499

PL/Python language, PostgreSQL support, 499

PL/Tcl language, PostgreSQL support, 499

PopulateSubscription() method, subscription list editing, 365–366

population, Full-Text Search methods, 406–407

population/crawl, defined, 404

positional parameters, command objects, 80

PostGIS, PostgreSQL downloadable add-on, 498

PostgreSQL

- BSD License, 497
- constraints, 498
- database elements, 498–499
- development history, 497
- extended data types, 498
- extensibility, 498

- foreign keys, 498
- inheritance, 498
- installation, 499–500
- materialized views, 498
- multi-version concurrency control, 499
- .NET integration, 500–501
- OpenFTS, 498
- pgAdmin III, 499
- PostGIS, 498
- procedural languages, 499
- referential integrity, 498
- replication, 499
- resource information, 501–502
- SQL compliance, 498
- stored procedures, 498
- triggers, 498
- updateable views, 498
- `PopulateDevice()` **method, device list editing, 362**
- prefix terms, Full-Text Search query, 413**
- prefixes**
 - stored procedure guidelines, 18
 - trigger naming conventions, 17
- `Prepare()` **method, data providers, 215**
- `PrepareCommand()` **method, data providers, 215, 219**
- primary keys**
 - column definitions, 19–20
 - column naming conventions, 16
 - foreign key relationships, 20–21
 - MySQL, 473
- private cache area, data source caching, 167–169**
- procedural languages, PostgreSQL supported types, 499**
- procedural programming, CLR (Common Language Runtime) Objects, 303**
- procedures**
 - naming conventions, 17–19
 - `sspGetSalesTotal`, 429–430
 - XP (extended stored), 301–302
- programmability tracking, SQL Server projects, 306**
- properties**
 - `AdsDataAdapter` class, 229
 - `AdsDataReader` class, 226–228
 - `AdsProvider`, 205, 206
 - `ASSEMBLYPROPERTY`, 311
 - `Column()` object, 335
 - `CommandText`, 213
 - `CompositeIdentifierSeparatorPattern`, 69–70
 - `Connection`, 213
 - `ConnectionString`, 204–205
 - `ConnectionTimeout`, 205
 - `Database`, 205, 206
 - `DataSource`, 205
 - `DataSourceProductName`, 70
 - `DbConnection`, 213
 - `DbParameterCollection`, 214–215
 - `DbTransaction`, 214–215
 - `Depth`, 226
 - `FieldCount`, 226
 - `Full-Text Search`, 411–412
 - `GenerateDBDirectMethods`, 186–187
 - `GroupByBehavior`, 70
 - `HasRows`, 226
 - `IdentifierCase`, 70
 - `IdentifierPattern`, 70
 - `Index()` object, 335–336
 - `IsClosed`, 226
 - `IsCompleted`, 270–271
 - legacy support issues, 40–41
 - `ObjectDataSource` control, 177–178
 - Oracle connection string, 524–525
 - `OrderByColumnsInSelect`, 71
 - `ParameterMarkerFormat`, 71
 - `ParameterMarkerPattern`, 71
 - `ParameterNameMaxLength`, 71
 - `ParameterNamePattern`, 71
 - query results ranking, 419–420
 - `QuoteIdentifierCase`, 72
 - `QuoteIdentifierPattern`, 72
 - `RecordsAffected`, 226
 - scalar-valued functions, 320
 - `ServerConnection()` object, 331
 - `ServerVersion`, 205
 - `SQLCacheDependency`, 168
 - `SqlConnectionStringBuilder` class, 282–283
 - `SqlContext` object, 327
 - `SQLDataSource` control, 174–175
 - `SqlDependency` class, 298
 - `SqlFunction` attribute, 320
 - `SqlTriggerContext` class, 311
 - `State`, 205
 - `StatementSeparatorPattern`, 72
 - `StringLiteralPattern`, 72
 - `SubscriberID`, 357
 - `SupportedJoinOperators`, 72–73
 - `Table()` object, 335
 - `TriggerAction`, 312
 - user versus Active Directory, 221–226
 - `User()` object, 336–337

Property Page, `SQLDataSource` control configuration, 173

protocols

application definition file, 354–355
delivery, 350

provider classes, Oracle, 516–519

provider statistics, SQL Server, 292–293

providers. See also data providers

application definition file, 353
connection strings, 53–60
Generic Factory Model architecture, 44–45
`GetDataSources()` support, 51
hosted event, 349, 353
independent event, 349
machine.config file default types, 46
namespaces, 43–44
non-hosted event, 349, 353
OLE-DB workflow process, 5–6
provider-specific commands, 78

proximity terms, Full-Text Search query, 413

Q

quantum, generator component, 349

queries

anchors, 253–254
data caching, 293–297
Full-Text Search, 412–418
joinback, 253–254
MaxDB, 479
multi-column, 418
recursive, 253–254
reserved words, 75
results ranking, 419–422
schema metadata system, 68–73
schema metadata system information sources, 76–77
schema metadata system restrictions, 73–75
Service Broker, 372
`SQLDataSource` control configuration, 173

query-linked servers, Full-Text Search, 418

queues

e-mail sending service, 386–389
Service Broker, 370, 378–380, 394–396

quotation marks (“), quoted identifiers, 78–79

quoted identifiers

double quote (") characters, 78–79
square brackets [and] characters, 78–79

`QuoteIdentifier()` method, SQL Server, 78–79

`QuoteIdentifierCase` property, schema metadata system, 72

`QuoteIdentifierPattern` property, schema metadata system, 72

R

`RANK()` function, T-SQL, 257

`rc` namespace, ReportServer, 442

RDL (Report Definition Language), Reporting Services, 436–438

RDO (Remote Data Objects), development history, 7–8 reachability, T-SQL, 244

`Read()` method, data providers, 219

read-ahead caching, Ingres, 493

`ReadAsObject` method, XML block reading, 122

reader (select) methods, data providers, 216

READFAST function, T-SQL, 262

REAL (FLOAT(24)) data type, SQL Server, 101

receiving services, Service Broker component, 370

record locking, Firebird, 485–486

record versioning, Firebird, 485–486

records

data provider deletion, 217–218
data provider updates, 217–218
deletion protection methods, 35

`RecordsAffected` property, data providers, 226

recursive queries, T-SQL, 253–254

REFERENCE permission, Service Broker contracts, 377–378, 381

referential integrity

Firebird, 487
MaxDB, 478
PostgreSQL, 498

regular expressions

`IdentifierPattern` property, 71
Oracle, 536–538
`ParameterMarkerPattern` property, 71
`ParameterNamePattern` property, 71
`QuoteIdentifierPattern` property, 72

relational databases, development history, 3–4

relationships, foreign key/primary key, 20–21

reliability, data access factor, 1

Remote Data Objects (RDO), development history, 7–8

`Render()` method, ReportServer, 450–451

replication

BerkeleyDB, 503
Ingres, 493
PostgreSQL, 499

Report Definition Language (RDL), Reporting Services, 436–438**Reporting Services**

- account rights, 429
- versus ASP.NET reports, 433
- data sources, 436–437
- execute rights, 429
- implementation, 428
- log in, 429
- page rendering, 437–438
- RDL (Report Definition Language), 436–438
- report models, 460–464
- ReportServer, 438–460
- scenarios, 428
- servers, 438–460
- SQL Server setup, 428–430
- SqlDataSource connection string, 431–432
- sspGetSalesTotal procedure, 429–430
- URL contacts, 439–446
- URL setup, 432
- Web applications, 430–432

reports

- compiled views, 23–24
- ReportServer HTML rendering, 452–453

ReportServer

- CodeBeside page, 446
- DeviceInfo object, 453–460
- forms, 445
- HTML report rendering, 452–453
- JavaScript validation, 444–445
- rc namespace, 442
- Render () method, 450–451
- rs namespace, 440–441
- SOAP contacts, 447–448
- URL contacts, 439–446
- URL parameters, 443–446
- validations, 444–445
- Web application setup, 448–453
- WSDL (Web Service Description Language), 447–448

reserved words, schema metadata system, 75

- Restore **class, SMO utility class, 330**

- Restore() **object, SQL Server Management Objects (SMOs), 340–341**

- restore/backup, **SQL Server Management Objects (SMOs), 338–341**

- restrictions, **schema metadata system, 73–75**

- result sets, **multiple active, 276–278**

- results output, **CLR (Common Language Runtime) Objects, 318**

- results rankings, **Full-Text Search, 419–422**

- return codes, **command execution status reporting, 85–87**

- routes, **Service Broker, 370, 390–394**

- row set loops, **CURSOR data type, 108**

- ROW_NUMBER () **method, T-SQL, 256–257**

rows

- Oracle insertion, 531–533
- TIMESTAMP data type, 107

- RowState () **object, DataTables, 91**

- rs **namespace, ReportServer, 440–441**

- rules, **generator component, 349**

S

- SAFE **permission, CLR (Common Language Runtime) Objects, 309**

SAPdb (MaxDB)

- backups, 480
- Database Analyzer, 479
- database clustering, 480
- database elements, 479–481
- Database Manager utility, 479
- development history, 477–478
- foreign keys, 478
- hot standby configuration support, 480
- implicit/explicit schemas, 480
- installation, 481–483
- isolation levels, 480–481
- locking, 480–481
- nested subtransactions, 479
- .NET interaction, 483–484
- outer joins, 478
- query tools, 479
- referential integrity, 478
- resource information, 484
- server-side cursors, 480
- SQL interaction, 478–479
- SQL modes, 478
- SQL Studio, 479
- SQLCLI, 279
- stored procedures, 478–479
- subtransactions, 479
- Synchronization Manager, 480
- temporary tables, 480
- triggers, 478–479
- updateable views, 479
- WebDAV (Web Distributed Authoring and Versioning), 480
- WebSQL, 479

scalars, single value returns, 87

scalar-valued functions, CLR (Common Language Runtime) Objects, 319–320

schema inference, XML/ADO.NET integration, 132–133

schema metadata system

data source information, 69–73

data types, 69

IdentifierPattern property, 71

information queries, 68–73

information sources, 76–77

OrderByColumnsInSelect property, 71

ParameterMarkerFormat, 71

ParameterMarkerPattern property, 71

ParameterNameMaxLength property, 71

ParameterNamePattern, 71

provider-agnostic API, 67

QuoteIdentifierCase, 72

QuoteIdentifierPattern, 72

reserved words, 75

restrictions, 73–75

StatementSeparatorPattern, 72

StringLiteralPattern, 72

SupportedJoinOperators, 72–73

uses, 77

schemas

CustomerInfoSchema, 376

MaxDB, 480

scientific calculations, CLR (Common Language Runtime) Objects, 329

Script() method, SQL Server Management Objects (SMOs), 342

Scripter class, SMO utility class, 330

Scripter() object, SQL Server Management Objects (SMOs), 341–343

scripting objects, SQL Server Management Objects (SMOs), 330

scripts

Service Broker, 373

Service Broker communication routes, 390–394

SQL Server Management Objects (SMOs), 341–343

test, 306

searches

ASP.NET asynchronous operations, 272–276

results display, 81–82

XML, 422

security

CAS (Code Access Security), 127–128

CLR (Common Language Runtime) Object permissions, 309

database views, 23

XSD documents, 127–128

security calculations, CLR (Common Language Runtime) Objects, 329

SELECT statement, data migration, 29

selectable stored procedures, Firebird, 487

self joins, T-SQL, 253

Send() method, CLR (Common Language Runtime) Objects, 318

sending services, Service Broker component, 370

separators, CompositeIdentifierSeparator Pattern property, 69–70

Sequence database object, Oracle, 532–533

serialization

DataTables, 92–93

XML/ADO.NET integration, 122–123

Server() object, SQL Server Management Objects (SMOs), 331

ServerConnection() object, SQL Server Management Objects (SMOs), 331

servers

query-linked, 418

Reporting Services, 438–460

server-side cursors, MaxDB, 480

ServerVersion property, data providers, 205

Service Broker

asynchronous communication, 370

automatic activation, 370

catalog views, 396–399

communication patterns, 369

components, 370–371

contracts, 377–378

coordination, 370

customer sign-up service, 383–386

database integration, 370

e-mail sending service, 386–389

log in, 373

message locking, 370

message ordering, 370

message processing, 381–394

message types, 374–377

message validation, 374–377

queries, 372

queues, 378–380, 394–396

routes, 390–394

sample scenario, 371–372

scripts, 373

service automation, 390

services, 380–381

- setup, 372–373
- SQL Service Broker Explorer, 400
- SSB Admin, 399–400
- Web application form, 381–383
- Service Oriented Architecture Protocol (SOAP), Report-Server, 447–448**
- services**
 - automating, 390
 - customer sign-up, 383–386
 - e-mail sending, 386–389
 - Service Broker, 370, 380–381
 - Service Broker contracts, 377–378
- set theory, CLR (Common Language Runtime) Objects, 302–303**
- set-based programming, CLR (Common Language Runtime) Objects, 302–303**
- Settings() object, SQL Server Management Objects (SMOs), 332**
- shorthand modes, MySQL, 472**
- simple terms, Full-Text Search query, 413**
- simple transactions, versus distributed transactions, 141**
- single-master replication, BerkeleyDB, 503**
- SMALLDATETIME data type, SQL Server, 100**
- SMALLINT data type, SQL Server, 100**
- SMALLMONEY data type, SQL Server, 102**
- SMOs (SQL Server Management Objects)**
 - Backup() object, 338–340
 - backup/restore, 338–341
 - capture execution, 330
 - Column() object, 335
 - columns, 334–336
 - Configuration() object, 332
 - database creation, 332–334
 - Database() object, 332, 334–335
 - described, 301, 329
 - direct execution, 330
 - execution methods, 330
 - foreign keys, 337–338
 - ForeignKey() object, 337–338
 - Index() object, 335–336
 - indexes, 334–336
 - Information() object, 332
 - instance classes, 330
 - object creation, 331–332
 - Restore() object, 340–341
 - Script() method, 342
 - Scripter() object, 341–343
 - scripting objects, 330
 - scripts, 341–343
 - Server() object, 331
 - ServerConnection() object, 331
 - Settings() object, 332
 - table creation, 334–336
 - Table() object, 334–335
 - tables, 334–336
 - URN (Unique Resource Name) referencing, 331
 - user creation, 336–337
 - User() object, 336–337
 - UserOptions() object, 332
 - utility classes, 330
 - WMI Provider objects, 330
- SOAP (Service Oriented Architecture Protocol), Report-Server, 447–448**
- space character, table name, 16**
- SQL (Structured Query Language)**
 - development history, 3, 235–238
 - Firebird compliance, 485
 - MaxDB interaction, 478–479
- SQL Server**
 - ADO.NET integration, 265–266
 - BIGINT, 100
 - BINARY, 102–106
 - BIT, 107
 - bulk copy operations, 291–292
 - cache dependent data, 168
 - CHAR (character), 98
 - CLR Object tables/functions, 310–311
 - CLR (Common Language Runtime) Objects, 301–328
 - connection pooling, 279–280
 - CURSOR, 108
 - data type/.NET mapping, 114–116
 - data types, 98–108
 - database connection strings, 306
 - DATETIME (TIMESTAMP), 100
 - DDL CREATE ENDPOINT command, 344–345
 - DECIMAL (DEC and DEC (p, s)), 101–102
 - DROP ENDPOINT command, 344–345
 - enumerations, 278–279
 - FLOAT (FLOAT and DOUBLE PRECISION), 101
 - full-text searching, 403–404
 - IMAGE, 102–103
 - INT (integer), 97, 99
 - management objects, 329–343
 - Microsoft.SqlServer.Server namespace, 306
 - MONEY, 102
 - MSFTESQL (Microsoft Full-Text Engine for SQL Server), 405

SQL Server (continued)

- NCHAR (national character storage), 99
- .NET Data Provider, 265
- NS instance, 356–357
- NTEXT data type, 99
- NUMERIC, 101
- NVARCHAR (MAX) data type, 99
- NVARCHAR (national character storage), 99
- password management, 280–283
- programmability tracking, 306
- provider statistics, 292–293
- quoted identifiers, 78–79
- REAL (FLOAT (24)), 101
- Reporting Services setup, 428–430
- reserved words, 75
- SMALLDATETIME, 100
- SMALLINT, 100
- SMALLMONEY, 102
- SMOs (SQL Management Objects), 301
- SQL_VARIANT, 107–108
- SqlConnectionStringBuilder class, 281–283
- SqlDataSourceEnumerator class, 278–279
- SqlTypes, 283–300
- stored procedure prefix guidelines, 18
- System.Data.Sql namespace, 306
- System.Data.SqlClient namespace, 306
- TABLE, 108
- test scripts, 306
- TEXT, 99
- TIMESTAMP, 107
- TINYINT, 100
- UNIQUEIDENTIFIER, 108
- VARBINARY, 102–103
- VARBINARY (MAX), 102–103
- VARCHAR (MAX), 99
- VARCHAR (char varying/character varying), 98
- Web Services/XML Endpoint, 301
- XML, 108
- XML Web services, 343–345
- XPs (extended stored procedures), 301–302
- SQL Server Management Objects (SMOs)**
 - Backup() object, 338–340
 - backup/restore, 338–341
 - capture execution, 330
 - Column() object, 335
 - columns, 334–336
 - Configuration() object, 332
 - database creation, 332–334
 - Database() object, 332, 334–335
 - described, 301, 329
 - direct execution, 330
 - execution methods, 330
 - foreign keys, 337–338
 - ForeignKey() object, 337–338
 - Index() object, 335–336
 - indexes, 334–336
 - Information() object, 332
 - instance classes, 330
 - object creation, 331–332
 - Restore() object, 340–341
 - Script() method, 342
 - Scripter() object, 341–343
 - scripting objects, 330
 - scripts, 341–343
 - Server() object, 331
 - ServerConnection() object, 331
 - Settings() object, 332
 - table creation, 334–336
 - Table() object, 334–335
 - tables, 334–336
 - URN (Unique Resource Name) referencing, 331
 - user creation, 336–337
 - User() object, 336–337
 - UserOptions() object, 332
 - utility classes, 330
 - WMI Provider objects, 330
- SQL Server Management Studio, Full-Text Search index setup, 408–411**
- SQL Service Broker Explorer, Service Broker administration tool, 400**
- SQL statements, TableAdapter method, 187**
- SQL Studio, MaxDB query tool, 479**
- SQL_VARIANT data type, SQL Server, 107–108
- SqlBulkCopy class, bulk copy operations, 291–292
- SqlBulkCopyColumnMapping class, bulk copy operations, 291–292
- SqlBulkCopyColumnMappingCollection class, bulk copy operations, 291–292
- SqlCacheDependency object
 - change notification, 539–543
 - query data caching, 293–297
- SqlCacheDependency property, data source caching, 168
- SqlCacheDependencyAdmin class, SQL Server, 297
- SQLCLI, MaxDB query tool, 479**
- SqlClient provider, SqlCommand object 78

- SqlClientFactory **provider**, GetDataSources() **method support**, 51
- SqlCLRObjects **assembly**, deployment methods, 307
- SqlCommand **class**, asynchronous command support, 267–272
- SqlCommand **object**, calling from SqlClient provider, 78
- SqlConnectionStringBuilder **class**, SQL connections, 281–283
- SqlContext **object**, CLR (Common Language Runtime) Objects, 327
- SqlDataRecord **object**, tabular results, 326–327
- SqlDataSource **connection string**, Reporting Services, 431–432
- SQLDataSource **control**, configuration methods, 170–175
- SqlDataSourceEnumerator **class**, SQL Server, 278–279
- SqlDbType **object**, SqlTypes, 290–291
- SqlDependency **class**, SQL Server, 297–300
- SQLite**
 - database elements, 509–510
 - development history, 508
 - installation, 510
 - minimalist approach, 510
 - .NET integration, 510–511
 - resource information, 512
 - small footprint, 510
 - SQL compliance, 508–509
 - zero configuration requirements, 510
- SqlMetadata **object**, tabular results, 326–327
- SqlPipe **object**, CLR (Common Language Runtime) Objects, 318
- SqlProcedure **attribute**, CLR (Common Language Runtime) Objects, 318–319
- SqlTriggerContext **class**, CLR (Common Language Runtime) Objects, 311–313
- SqlTypes **namespace**
 - bulk copy operations, 291–292
 - data retrieval methods, 287–290
 - data types, 109–113
 - provider statistics, 292–293
 - SqlCacheDependency **object**, 293–297
 - SqlCacheDependencyAdmin **class**, 297
 - SqlDbType **object**, 290–291
 - SqlDependency **class**, 297–300
 - structures, 284–287
- SqlUserDefinedAggregate **attribute**, aggregate functions, 322
- square brackets [and] characters**, quoted identifiers, 78–79
- SSB Admin**, Service Broker administration tool, 399–400
- sspGetSalesTotal **procedure**, Reporting Services, 429–430
- State** **property**, data providers, 205
- StateChange **event**, data providers, 207
- statements**
 - ALERT QUEUE, 378
 - StatementSeparatorPattern **property**, 72
 - StatementSeparatorPattern **property**, schema metadata system, 72
- static methods**, ObjectDataSource **control** support, 180
- statistics**, data deletion protection method, 35
- stemmer**, defined, 404
- storage engines**, MySQL, 469–470
- stored procedures**
 - AddNewCustomer, 382
 - CLR (Common Language Runtime) Objects, 318–319
 - Firebird, 487
 - MaxDB, 478–479
 - MySQL, 472–473
 - naming conventions, 17–19
 - Oracle, 544–547
 - PostgreSQL, 498
 - sspGetSalesTotal, 429–430
 - TABLE data type, 108
 - TableAdapter **method**, 188–189
 - XP (extended), 301–302
- string parameters**, provider-specific connection strings, 57–60
- StringLiteralPattern **property**, schema metadata system, 72
- strings**
 - connection, 53–60
 - connection fields, 203–204
 - database connection, 306
 - ParameterMarkerFormat **property**, 71
 - SqlDataSource **connection**, 431–432
- structured exception handling**, T-SQL, 240–249
- Structured Query Language (SQL)**
 - development history, 3, 235–238
 - Firebird compliance, 485
 - MaxDB interaction, 478–479
- subscriber devices**, subscription management application component, 349

SubscriberDevice object, device list editing

SubscriberDevice object, device list editing, 362–363

SubscriberEnumeration object, subscriber list retrieval, 357–359

SubscriberID property, subscribers, 357

subscribers

- adding/removing, 358–359
- device list display, 360
- retrieving, 357–358
- subscription management application, 357–359, 363–367
- subscription management application component, 348

subscription lists

- displaying, 363–364
- editing, 364–367

subscription management application

- adding/removing subscribers, 358–359
- device list, 360–363
- event firing, 367–368
- Notification Services, 348–349
- saving subscriptions, 366–367
- subscriber list retrieval, 357–358
- subscription list, 363–367

SubscriptionEnumeration object, subscription list display, 363–364

subscriptions

- application definition file, 355
- saving, 366–367
- subscription management application component, 349

SupportedJoinOperators property, schema meta-data system, 72–73

sys.assemblies table, CLR (Common Language Runtime) Object, 310

sys.assembly_files table, CLR (Common Language Runtime) Object, 310

sys.assembly_modules table, CLR (Common Language Runtime) Object, 310

sys.assembly_preferences table, CLR (Common Language Runtime) Object, 310

sys.assembly_types table, CLR (Common Language Runtime) Object, 310

sys.sql_modules table, CLR (Common Language Runtime) Object, 310

sys.types table, CLR (Common Language Runtime) Objects 310

system requirements, transactions, 156–157

System.Data.CommandBehavior() objects, enumerated types, 82–83

System.Data.Common namespace, provider classes, 202

System.Data.Sql namespace, SQL Server, 306

System.Data.SqlClient namespace, SQL Server, 306

System.Data.SqlTypes namespace, SqlType structures, 284–287

System.Transactions namespace, distributed transactions, 146–149

T

TABLE data type, SQL Server, 108

Table() object, SQL Server Management Objects (SMOs), 334–335

TableAdapter method, data binding, 185–189

tables

- CLR (Common Language Runtime) Objects, 310–311
- data providers, 212
- dragging/dropping into Windows Forms, 192
- foreign keys, 20–21
- MaxDB, 480
- message queues, 378–380
- namespace qualified, 94
- naming conventions, 15–17
- primary key conventions, 19–20
- Service Broker queues, 394–395
- SQL Server Management Objects (SMOs), 334–336
- SQLDataSource control configuration, 173–174
- user properties versus Active Directory properties, 221–222
- views, 22–23

TABLESAMPLE function, T-SQL, 259–262

table-valued functions, CLR (Common Language Runtime) Objects, 320–321

tabular results, CLR (Common Language Runtime) Objects, 326–327

TARGET service, Service Broker contracts, 377–378, 390–381

Terminate() method, aggregate functions, 322–323

test scripts, SQL Server projects, 306

TEXT data type, SQL Server, 99

text markup, XML/ADO.NET integration, 120

THESAURUS parameter, Full-Text Search, 417

time-based population, Full-Text Search method, 407

time/date, data types, 100

TIMESTAMP data type, SQL Server, 107

TINYINT data type, SQL Server, 100

token, defined, 404

TOP operator, T-SQL, 250–252

tracing, ODP.NET, 534–535

transaction logs, BerkeleyDB, 503

Transaction object, isolation levels, 142

transaction processing

C# language, 242–243

VB.NET, 241–242

transactional data store, BerkeleyDB, 503

transactions

A.C.I.D (atomicity/consistency/isolation/durability),
140–141

concepts, 139–145

distributed, 145–154

isolation levels, 142–143

local transaction creation, 143–145

logically atomic, 503

MaxDB, 479

MySQL support, 470

parent/child scenarios, 140

simple versus distributed, 141

system requirements, 156–157

types, 141

**TransactionScope object, distributed transactions,
146–149**

**Transact-SQL (T-SQL) language, development history,
235–238**

Transfer class, SMO utility class, 330

TriggerAction property, operations, 312–313

**TriggerContext object, CLR (Common Language
Runtime) Objects, 327**

triggers

CLR (Common Language Runtime) Objects, 311–317

column updates, 313

DDL, 263–264

DDL events, 313–314

MaxDB, 478–479

MySQL, 473

naming conventions, 17

PostgreSQL, 498

TriggerContext object, 327

TryGetValue() method, data providers, 210–211

T-SQL (Transact-SQL) language

aggregate functions, 323–324

at (@) character, 239

CLR mapping methods, 307

versus CLR objects, 329

CREATE ASSEMBLY statement, 307

CTEs (Common Table Expressions), 252–254

DDL triggers, 263–264

Dense_Rank() function, 258

development history, 235–238

flow-control statements, 239–240

InfoMessage event, 244–246

NTile() function, 258–259

OUTPUT method, 249–250

PIVOT function, 254–256

RANK() function, 257

reachability, 244

READFAST function, 262

recursive queries, 253–254

ROW_NUMBER() method, 256–257

self joins, 253

structured exception handling, 240–249

TABLESAMPLE function, 259–262

TOP operator, 250–252

UNPIVOT function, 256

variable declarations, 238–239

WAITFOR function, 262–263

two-way data binding, data concurrency issues, 163

typed DataSets, data binding, 185–186

U

UDFs (user-defined functions), Firebird, 487

UI controls, Windows Forms, 197–198

underscore (_) character, table name avoidance, 16

Unique Resource Name (URN), SMO referencing, 331

UNIQUEIDENTIFIER data type, SQL Server, 108

universal data access, development history, 4–6

UNPIVOT function, T-SQL, 256

UnquotedIdentifier() method, SQL Server, 78–79

**UNSAFE permission, CLR (Common Language Runtime)
Objects, 309**

updateable views, PostgreSQL, 498

updates, columns, 313

URL, ReportServer parameters, 443–446

URL contacts, ReportServer, 439–446

URN (Unique Resource Name), SMO referencing, 331

user objects, data providers, 212

**user properties, versus Active Directory properties,
221–226**

**User() object, SQL Server Management Objects
(SMOs), 336–337**

user-control caching, when to use, 168–169

**user-defined aggregate functions, CLR (Common Lan-
guage Runtime) Object, 323**

user-defined extensions, Ingres, 493

user-defined functions (UDFs), Firebird, 487
user-defined types, CLR (Common Language Runtime) Objects, 324–326

UserOptions() **object, SQL Server Management Objects (SMOs), 332**

users

password changing, 280–281
SQL Server Management Objects (SMOs), 336–337

utility classes, SQL Server Management Objects (SMOs), 330

Utils class

device list editing, 362
subscriber list retrieval, 357–359

V

validations

parameter data, 166
ReportServer, 444–445
Service Broker messages, 374–377
XML/ADO.NET integration, 131–133

VARBINARY data type, SQL Server, 102–103

VARBINARY (MAX) data type, SQL Server, 102–103

VARCHAR (MAX) data type, SQL Server, 99

VARCHAR (char varying/character varying) data type, SQL Server, 98

variable declarations, T-SQL, 238–239

VB.NET language

reachability, 244
transaction processing, 241–242

vendor commitment, data access factor, 2

versioning records, data preservation method, 35

views

data normalization, 22–23
database abstraction, 31–32
MaxDB, 479
MySQL, 473
PostgreSQL, 498
reports, 23–24
security enforcement, 23
Service Broker catalogs, 396–399

Visual Studio 2005

CLR debugging, 309–310
CLR deployment, 307–308
CLR mapping methods, 307
CLR permissions, 309
creating CLR objects, 304–306
data components, 192

Data Source controls, 169–170

Report Server project, 433

SQL Server projects, 306–310

UI controls, 197–198

Web applications, 430–432

Visual Studio.NET

Oracle developer tools, 519–533

Oracle Explorer, 519–522

W

WAITFOR function, T-SQL, 262–263

Web applications

data binding environment, 159–160
one-way versus two-way data binding, 163
Reporting Services, 430–432
ReportServer setup, 448–453

Web Distributed Authoring and Versioning (WebDAV), MaxDB, 480

Web forms, ObjectDataSource control implementation, 182–183

Web Service Development Language (WSDL), ReportServer, 447–448

Web services

distributed transactions, 155–156
XML/ADO.NET integration, 119–120

Web Services/XML Endpoints, SQL Server, 301

Web sites

anti-stored procedure argument, 238
BerkeleyDB downloads, 508
BerkeleyDB drivers, 508
BerkeleyDB licensing information, 502
BerkeleyDB support, 508
Bill Vaughn, 249
BSD License, 497
CATOSL (CA Trusted Open Source License), 492
Dan Sullivan, 400
Firebird downloads, 491
Firebird drivers, 491
Firebird support, 492
Ingres downloads, 497
Ingres drivers, 497
Ingres support, 497
InterBase Public License, 485
MaxDB downloads, 484
MaxDB drivers, 484
MaxDB support, 484
Microsoft downloads, 427

MySQL documentation, 469
 MySQL downloads, 477
 MySQL drivers, 477
 MySQL licensing, 468–469
 MySQL support, 477
 .NET Framework change issues, 41
 Niels Berglund, 399
 nullable types, 114
 Oracle Developer Tools, 519
 OSI (Open Source Initiative) definition, 468
 PostgreSQL, 499
 PostgreSQL downloads, 501
 PostgreSQL drivers, 502
 PostgreSQL support, 502
 Scot Allen, 140
 SQL (Structured Query Language) specifications, 235
 SQLite discussion, 508
 SQLite downloads, 512
 SQLite drivers, 512
 SQLite support, 512
Web SQL, MaxDB query tool, 479
**WebDAV (Web Distributed Authoring and Versioning),
 MaxDB, 480**
WEIGHT function, Full-Text Search, 417
weighted terms, Full-Text Search query, 413
Windows Forms
 application data binding, 191–198
 BindingList control, 198
 BindingNavigator control, 198
 BindingSource control, 197–198
 data binding, 191–198
 data binding environment, 159–160
 data sources, 192–196
 dragging/dropping tables, 192
 one-way versus two-way data binding, 163
 UI controls, 197–198
wizards
 Configuration, 170–174, 181–184, 187–189
 Data Source Configuration, 193–196
 Full Text Indexing, 409–411
**WMI Provider objects, SQL Server Management Objects
 (SMOs), 330**
word breaker, defined, 404
write-ahead logging, BerkeleyDB, 503
**WriteFromObject method, XML object representa-
 tion, 122**
**WSDL (Web Service Description Language), Report-
 Server, 447–448**

X

XML (eXtensible Markup Language)

BerkeleyDB support, 503
 CAS (Code Access Security), 127–128
 change notification, 133–134
 configuration files, 120
 data exchange, 119
 design goals, 120–121
 Designer, 124–125
 DeviceInfo object tags, 453–460
 flexibility, 117
 Full-Text Search, 422
 future trends, 135–136
 Inference class, 132–133
 ItemDeleted event, 133–134
 ItemDeleting event, 133–134
 ItemInserted event, 133–134
 ItemInserting event, 133–134
 ItemUpdated event, 133–134
 ItemUpdating event, 133–134
 Microsoft Visual Studio 2005 requirement, 118
 .NET Framework schema type conversions, 123–124
 object serialization, 122–123
 ObjectSpaces, 136
 output validation, 131–132
 performance enhancements, 134–135
 schema inference, 132–133
 security focus, 127–128
 Service Broker message validation, 374–377
 text markup, 120
 validation, 131–133
 Web services, 119–120
 XML Views, 136
 XmlReader class, 121–124
 XmlSchemaSet class, 131
 XmlWriter class, 121–124
 XPath 2.0, 135
 XPathDocument, 128–130
 XQuery, 135–136
 XSD documents, 126–128
 XSL debugging, 125–126
 XSLT 2.0, 135–136
 XSLT enhancements, 134
XML data type, SQL Server, 108
XML Views, declarative maps, 136
XML Web services, database processing, 343–345
 XmlReader class, XML/ADO.NET integration, 121–124

XmlSchemaSet class, XML/ADO.NET integration

`XmlSchemaSet` class, XML/ADO.NET integration, 131
`XmlWriter` class, XML/ADO.NET integration, 121–124
XP (extended stored procedures), SQL Server, 301–302
XPath 2.0, standards, 135
`XPathDocument`, XML/ADO.NET integration, 128–130
XQuery, XPath 2.0 superset, 135–136

XSD documents, XML/ADO.NET integration, 126–128
XSL debugging, XML/ADO.NET integration, 125–126
XSLT 2.0, XPath 2.0 superset, 135–136

Z

zero configuration, SQLite, 510

