

# Index

## A

- AcceptChanges, DataSets, 183
- Access. *See* Microsoft Access
- Accessors
  - OleDb data providers, 358
  - strongly typed, 98–99
- ACID (atomicity, consistency, isolation, and durability) properties, 2–3, 150–151
- ActiveX Template Library (ATL), 366
- ADO, 415–417. *See also* interoperability, ADO with ADO.NET
  - asynchronous operations, 417
  - class equivalences with ADO.NET, 410–414
  - COM-based approach, 23
  - commands, 416–417
  - connections, 416
  - disadvantage of, 409–410
  - example using, 414–415
  - hierarchical data, 417
  - integration in OleDbDataAdapter, 195–196
  - object model of, 409
  - OLE DB and, 408
  - properties, 417
  - transactions, 416
  - XML and, 277
- ADO disconnected model, 417–424
  - class equivalences, 418–419
  - column and table names, 424
  - DataSets extensions, 423
  - filtering and sorting, 424
  - GetRows, 420–421
  - navigation, 419
  - updates, 422–423
- ADO Extensions. *See* ADOX (ADO Extensions)
- ADO for Multidimensional (ADOMD), 467
- ADO-format XML, 428–433
  - DataSet example, 430–431
  - hierarchical DataSet example, 432–433
  - hierarchical Recordset example, 431–432
  - Recordset example, 428–430
- ADO programmers
  - class equivalences, 410–414
  - example comparing ADO to ADO.NET, 414–415
- ADO Recordset. *See* Recordsets
- ADOMD (ADO for Multidimensional), 467
- ADO.NET
  - architecture for data access, 25–26
  - building upon OLE DB and ADO, 23–24
  - connected vs. disconnected access methods, 26–27
  - data access styles in, 25
  - layout of managed data classes, 65–67
  - support for nonrelational data, 13
- ADO.NET, connected mode, 27–42
  - console application example, 29–34
  - cursors and, 41–42
  - generic data access code, 35–40
  - OleDb and SqlConnection data providers, 34–35
  - SQL and OleDb connection strings, 27–29
- ADO.NET data providers
  - Command class, 371–375
  - Connection class, 368–371
  - DataAdapter class, 380–382
  - DataReader class, 375–380
  - enhancing functionality of, 382–383
  - overview of, 396
  - writing, 352–353, 366–368
- ADO.NET, disconnected mode
  - JDBC and, 452–453
  - updates, 42–46
- ADOX (ADO Extensions)
  - CommandBuilder, 218–219
  - DDLs and, 433
  - library, 23

- ADTG (Advanced Data TableGram), 23, 428
  - Aggregate types, 487
  - AND operator, 485
  - ANSI standards, SQL schema, 74
  - APIs (Application programming interfaces). *See also*
    - XML APIs
    - data access, 1
    - database access, 3
    - evolution of, 19–24
    - writing, 21
  - Architecture, ADO.NET, 25–26
  - Arithmetic operators, 485
  - Asynchronous operations, ADO vs.ADO.NET, 417
  - ATL (ActiveX Template Library), 366
  - Atomicity. *See* ACID (atomicity, consistency, isolation, and durability) properties
  - Attributes
    - mapping ODBC attributes to ADO.NET, 440–441
    - relational model and, 5
    - schema inference and, 492–493, 495–496
  - Authentication, 4
  - Authorization, 4
  - AutoIncrement columns, 137–138
- B**
- Basic Direct Access Method (BDAM), 9
  - Batch optimistic concurrency, 227
  - Batch updates, 219–220, 423
  - BDAM (Basic Direct Access Method), 9
  - Binary formats, 428. *See also* ADTG (Advanced Data TableGram)
  - Binary large objects. *See* BLOBs (binary large objects)
  - BLOBs (binary large objects)
    - connected model and, 101
    - managed types and, 358
    - metadata and, 446
    - object-relational model and, 460
    - storing data as, 12
  - Boolean operators, 485
  - Bulk loading, XML to SQL Server, 339
- C**
- C#
    - console application example, 29
    - processing server-side cursor, 425–426
  - Cancel methods, Command class, 92
  - Cascading updates, 160–161
  - CASPOL command-line utility, 125
  - Catch blocks, exceptions, 32
  - Changes
    - AcceptChanges, 183
    - RejectChanges, 287–288
  - Child elements, 493–494, 496. *See also* Parent-child
  - Class equivalents
    - ADO and ADO.NET, 410–414
    - ADO disconnected model and ADO.NET, 418–419
    - JDBC and ADO.NET, 449–452
  - Client cursor library, 41
  - Client-server applications, 16–17
  - CLOBs
    - connected model and, 101
    - managed types and, 358
    - metadata and, 446
    - object-relational model and, 460
  - Clone method, DataRows, 149–150
  - Close method, connection classes, 71
  - CLR data types, 171–172
  - CLS (Common Language Specification), 37
  - Clustered indexes, 6
  - CODASYL model, 9, 171, 460
  - Codd, E. F., 5
  - Code
    - access security, 125
    - declarative code, DataGrid, 260–262
    - declarative code, DataList, 257–259
    - generic code, JDBC, 443
    - generic code, OLE DB, 404
    - interoperability with legacy code, 25
    - stored procedures examples, 85–86
  - Column values
    - field indexers and, 99–100
    - fields as array of objects, 99
    - large data columns, 101–103
    - strongly typed accessors, 98–99
  - Columns. *See also* DataColumnms
    - attributes of elements inferred as, 495
    - elements without attributes inferred as, 496
    - expression syntax, 483–484
    - mapping information, 186–188
    - names, 424
  - COM-callable wrappers, 25, 425

- COM (Component Object Model)
  - interoperability, 402
  - libraries, 425
  - ODBC and, 22
- Command-based data models. *See* Connected data access model
- Command cotype, OLE DB, 353
- Command dialects, OleDb data providers, 359–360
- Command instances, DataAdapter, 179
- CommandBehavior parameters, 81–82, 372–374
- CommandBuilder class, 199–206
  - DeriveParameters and, 205–206
  - extended functionality with, 383
  - InsertCommand, UpdateCommand, and DeleteCommand, 199
  - populating DataAdapter with, 200–201
  - rebuilding commands, 201–202
  - table names containing spaces and, 202
  - updates generated by, 202–205
  - using, 200
- Command.ExecuteReader, 79
- Command.ExecuteScalar, 81
- Commands, 371–375
  - ADO vs. ADO.NET, 416–417
  - as basic data provider class, 367–368
  - customized, ADOX CommandBuilder, 218–219
  - customized, batch updates, 219–220
  - IDbCommand interface, 371
  - implementation, 373–374
  - JDBC vs. ADO.NET, 447
  - ODBC vs. ADO.NET, 435–436
  - OleDb data providers, 358–359
  - specialization, 374–375
  - specification, 371–373
- Commands, connected model, 77–82
  - behaviors, 81–82
  - Command.ExecuteScalar, 81
  - initializing, 77–78
  - multiple results, 80–81
  - preparation, cancellation, and cleanup, 92–94
  - returning total number of rows affected, 79
  - switching connections, 78
- CommandText, 79, 86
- CommandTimeout, 92
- CommandType, DataSets, 66
- Commit transactions, 107
- Common Language Specification (CLS), 37
- Comparison operators, 485
- Compile-time type checking, 457
- Component Object Model. *See* COM (Component Object Model)
- Composition, XML, 334
- Compute method, DataSets, 145–146
- Concurrency
  - checking, 197–198
  - optimistic vs. pessimistic, 177–178, 462–463
- Connected data access model
  - as ADO.NET data access method, 8
  - ADO.NET data providers and, 69–70
  - connection classes, 70–73
  - declarative transactions, 121–124
  - error handling, catching errors and warnings, 106–107
  - error handling, Sqlclient, 104–106
  - metadata methods, 74–77
  - permission classes, 124–126
  - relational databases and, 463
  - streaming data through DataReaders, 94–97
  - vs. disconnected model, 2, 26–27
- Connected data access model, column values
  - field indexers, 99–100
  - fields as array of objects, 99
  - large data columns, 101–103
  - strongly typed accessors, 98–99
- Connected data access model, commands
  - behaviors, 81–82
  - canceling, 92–93
  - Command.ExecuteScalar, 81
  - disposing of, 93–94
  - initializing, 77–78
  - multiple results, 80–81
  - Prepare method, 92
  - returning number of rows affected, 79
  - switching connections, 78
- Connected data access model, connection pooling
  - Min Pool Size, 119–121
  - overview of, 115–118
  - parameters, 73–74
  - sample listing, 118–119
- Connected data access model, parameters
  - adding to collection, 86–87
  - overview of, 82–83
  - retrieving result and output parameters, 84–85
  - SqlParameter, 87–88
  - stored procedures and, 85–86, 91
  - using with OleDb data provider, 89–90
  - using with SQL Server provider, 89–90

- Connected data access model, transactions
    - distributed, 111–113
    - EnterpriseServices, 112–115
    - named savepoints, 110
    - nested, 109–110
    - simple example, 107–109
  - Connected mode, ADO.NET, 27–42
    - console application example, 29–34
    - cursors, 41–42
    - generic data access code, 35–40
    - OleDb and SqlClient data providers, 34–35
    - SQL and OleDb connection strings, 27–29
  - Connection attributes, ODBC, 440
  - Connection classes, 367–371
    - as basic data provider class, 367–368
    - connected model and, 70–73
    - implementation, 369–370
    - specialization, 371
    - specification, 369
  - Connection Lifetime parameter, 120
  - Connection pooling, 115–121
    - JDBC vs. ADO.NET, 445
    - Min Pool Size, 119–121
    - overview, 73, 115–118
    - parameters, 74
    - sample listing, 118–119
  - Connection strings
    - ADO/ADO.NET interoperability, 416, 427–428
    - connected mode and, 27–29
    - Odbc data provider, 435
  - Connection Timeout, 120
  - Connection.Dispose method, 71
  - ConnectionState, 369–370
  - ConnectionTimeout property, 71
  - Consistency. *See* ACID (atomicity, consistency, isolation, and durability) properties
  - Console application example, 29–34
  - Constraints. *See also* Primary keys
    - creating, 141
    - overview of, 140
    - relaxing, 142
  - ContinueUpdateOnError property, 198–199
  - Control types, data binding, 233–235
  - Conversion, to ADO.NET, 401–402
  - Copy method, DataRows, 149–150
  - Cotypes
    - mapping OLE DB to ADO.NET, 403–404
    - mapping OLE DB to OleDb data provider, 362–364
    - OLE DB, 23, 353
    - OleDb data providers, 357
  - Crystal Reports, 356
  - Currencies positions, 11
  - Cursors, 41–42. *See also* Server cursors
- ## D
- DAO (Data Access Objects), 22
  - Data access
    - data provider model for, 351
    - styles in ADO.NET, 25
    - traditional, 273–274
  - Data Access Objects (DAO), 22
  - Data binding, 227–243
    - control types, 233–234
    - to DataReader, 241–243
    - with DataSet, 243–246
    - HTMLControls, 233–237
    - multiple results, 243
    - overview, 227–230
    - Web Form types, 230–233
    - WebControls, 233–235, 237–241
  - Data definition language. *See* DDL (Data definition language)
  - Data directed routing, 78
  - Data Form Wizards, 270–271
  - Data manipulation language, 9
  - Data mining, 13, 360
  - Data models
    - flat file, 9
    - hierarchical, 9–11
    - modern nonrelational, 11–12
    - multidimensional, 12–13
    - overview of, 1–2
    - relational, 5–8
  - Data presentation. *See* GUIs (graphical user interfaces)
  - Data providers, 60–65, 351–400
    - ADO.NET. *see* ADO.NET data providers
    - base classes of, 190
    - choices, 351–353, 396–398
    - connected model and, 69–70
    - JDBC vs. ADO.NET, 443–445
    - new architecture for, 25–26
    - nonrelational data and, 465
    - OLE DB vs. ADO.NET, 405
    - OleDb. *see* OleDb data providers

- OleDb connection strings and, 27–29
- SqlClient. *see* SqlClient data providers
- transparency of, 404–405
- XML. *see* XML Information Set (Infoset) providers
- data source name (DSN) files, 71
- Data storage, 399
- Data types. *See also* Managed data types
  - mapping JDBC to ADO.NET, 454–455
  - mapping Odbc to ADO.NET, 476–478
  - mapping OleDb to ADO.NET, 474–476
  - mapping Parameter.DB to ADO.NET, 479–481
  - mapping SQL Server to ADO.NET, 473–474
  - SQL and OLE DB, 87–88
  - supported by DataColumnms, 131–132
- Data warehouses, 13
- DataAdapters, 177–223, 380–382
  - as basic data provider class, 367–368
  - command instances, base classes, and interfaces, 179
  - customized commands, 218–220
  - DataReader and, 221–223
  - disconnected model and, 8, 177–178
  - implementation, 381–382
  - nonrelational types, 220–221
  - OleDbDataAdapter, 195–196
  - specialization, 382
  - specification, 380–381
  - synchronizing function of, 177
- DataAdapters, event model
  - refreshing DataSets, 216–218
  - RowUpdating handlers, 214–215
  - turning off errors during batches, 215–216
- DataAdapters, Fill method
  - error handling, 188–190
  - how it works, 190–195
  - overloads, 182–183
  - pessimistic concurrency and, 181–182
  - populating DataSets, 179–181
  - schema and mapping information and, 183–188
- DataAdapters, Update method
  - CommandBuilder class and, 199–206
  - concurrency checking, 197–198
  - ContinueUpdateOnError property, 198–199
  - disconnected mode and, 196–197
  - hand coded example, 208–209
  - how it works, 212–214
  - overview of, 45
  - retrieving identity columns and, 211–212
  - specifying row version, 206–207
  - timestamp construction, 209–211
- Database administrator accounts, 126
- Database cursors. *See* Server cursors
- Databases
  - authorization and authentication methods, 4–5
  - as data management mechanism, 1
  - DataSets vs. in-memory databases, 128
  - fundamentals of, 2–3
  - object-oriented, 399
  - physical records, 9
  - update methods, 177–178
- Databound controls, 26. *See also* data binding
- DataColumns. *See also* Columns
  - accessing and updating, 146
  - adding to DataTables, 134–136
  - computed, 134, 255–256
  - DataAdapter.Fill and, 192–193
  - DBNull and default values and, 132–133
  - errors, 163–164
  - mapping, 254–255
  - overview, 131–132
  - referencing, 303
  - schema information, 183
  - XML-specific properties of, 133
- DataDocumentXPathNavigator, 331
- DataExceptions. *See also* Exceptions
  - error handling, 163–164
  - list of, 161–162
- DataGrid, 259–265
  - binding to SqlDataReader, 241–242
  - custom pager, 263–264
  - declarative code, 260–262
  - overview, 259–260
  - sorting, 264–265
  - updating, 262–263
- DataList, 256–259
  - declarative code, 257–259
  - templates, 256–257
- Datamarts, 13
- DataReaders, 375–380
  - as basic data provider class, 367–368
  - binding to, 241–243
  - function of, 69
  - IDataReader and IDataRecord and, 375–376
  - implementation, 377–379
  - interfaces for, 94
  - OleDb data providers and, 357

- DataReaders (*continued*)
  - querying large relational databases with, 462
  - specialization, 379–380
  - specification, 376–377
  - streaming data through, 94–97
  - vs. DataAdapters, 190, 221–223
  - vs. DataSets, 245
  - vs. XmlReader, 47
- DataRelations. *See also* Relationships
  - counting relations, 142–143
  - defining, 141
  - inference process and, 492
  - nested, 282–283
  - specification of cardinality and, 143
  - XML and, 282
- DataRows
  - adding to DataTables, 136
  - Copy and Clone methods, 149–150
  - DataRowState and DataRowVersions, 154–158
  - deleting, 147
  - editing, 151–152
  - errors, 163–164
  - filtering, 249
  - inserting, 148
  - overview, 131
  - updates and versions, 196–197
  - working with, 146
- DataRowState, 154–158, 249–250
- DataRowVersions, 154–158
  - changing order of inserts with, 213–214
  - current, default, original, and proposed, 155
  - disconnected mode and, 196–197
  - functions of, 158–159
  - specifying, 206–208
- DataSet object model, 129–167
  - array of objects in, 139
  - events, list of, 165–166
  - events, sample program, 166–171
  - filling DataSets programmatically, 138–139
  - ImportDataRow method, 148–150
  - keys and constraints, 139–140, 142
  - LoadDataRow method, 148–150, 158–159
  - merges, 152–154
  - nonrelational types, 171–175
  - row versions, 158–159
  - rules and relationships, 159–161
  - synopsis of classes in, 129–131
  - updates and changes, 150–152
- DataSet object model, DataColumnns
  - accessing and updating, 146
  - computed, 134
  - overview of, 131–132
  - using DBNull and default value with, 132–134
- DataSet object model, DataRelations
  - counting relations, 142–143
  - defining, 141
- DataSet object model, DataRows
  - Copy and Clone methods, 149–150
  - deleting, 147
  - editing, 151–152
  - inserting, 148
  - overview of, 131, 136
  - state and versions, 154–158
  - working with, 146
- DataSet object model, DataTables
  - adding columns, 134–136
  - adding rows, 136–138
  - overview of, 131
  - retrieving and updating values, 146
- DataSet object model, error handling
  - causing exceptions, 163–164
  - list of errors thrown, 161–162
  - processing errors, 164–165
- DataSet object model, navigation methods
  - getting child rows, 144
  - hierarchies, 144–145
  - using Select and Compute, 145–146
- DataSet.InferXmlSchema, 491
- DataSet.Merge, 217
- DataSets
  - ADO-format XML, 430–433
  - CommandType, 66
  - data binding with, 243–246
  - disconnected mode, 8, 44
  - extensions to ADO Recordset, 423
  - filling from DataAdapter, 180–181
  - filling from WebDav provider, 427
  - functions of, 129
  - integration with XML stack, 274
  - interoperability with Recordsets, 427–428
  - mapping OLE DB to, 364–366
  - overview of, 127–128
  - pessimistic vs. optimistic concurrency, 462
  - populating programmatically, 397
  - producing XML with, 60–62
  - PropertyCollection, 129

- refreshing with Update and Merge, 216–218
- schema, defining, 174–176, 274–277
- schema, inferring. *see* Schema inference rules
- schema, refining for XML, 280–283
- specialized use of, 464
- updates using, 42–46
- vs. DataReaders, 245
- vs. in-memory database, 128
- DataSets, XML and, 324
  - loading, 324
  - reading XML into DataSet, 283–288
  - use of XML properties, 281–282
  - writing XML data, 292–298
  - writing XML schemas, 288–291
  - XmlDataDocument class, 326–327
- DataSet.WriteXml method, 292–293
- DataTables
  - AutoIncrement columns and, 137–138
  - columns, adding, 134–136
  - DataAdapter.Fill and, 193
  - disconnected mode, 44–45
  - embedded, 173–174
  - event model, 214
  - filling from SQL statements, 191
  - mapping, 254–255
  - methods, 148
  - overview, 131
  - primary keys, 139–140
  - references, 303
  - retrieving and updating values, 146
  - rows, adding, 136
  - schema, 183–185
  - views, 246, 248
- DataManager, 250, 252–254
- DataViews, 246–254
  - DataManager, 252–254
  - DefaultView, 246–247
  - disconnected mode, 44–45
  - filtering and sorting, 249–250
  - modifying DefaultView, 251
  - nonrelational data, 265–266
  - reading items, 249–250
  - restricting editing, 248–249
  - two views on same DataTable, 248
  - vs. database views, 175–176
- Date, C. J., 5, 7
- Date values, 484
- DB2, 333
- DbDataAdapter base class, 179, 380–381
- DbDataAdapter.Update. *See* Update method
- DBDataPermission, 125
- DbEnumerator class, 379
- DBNull
  - columns and, 132–134
  - exceptions and, 162
- DbType, 87, 471–472
- DDL (data definition language)
  - ADOX and, 433
  - SQL, 53
  - XML schemas as, 400
- Decision support, data warehouses, 13
- Declarative code
  - DataGrid, 260–262
  - DataList, 257–259
- Declarative data binding, 230
- Declarative transactions, 121–124
- Decomposition, relational model, 8
- DefaultView, DataViews, 246–247, 251
- DELETE method, DataTables, 148
- DeleteCommand
  - customizing, 206
  - DataAdapter Command instances, 179
  - Update method and, 199
- DeriveParameters, CommandBuilder class, 205–206
- Designer libraries, 266
- DiffGram
  - applying, 295–297
  - changes and, 293–295
  - format of, 284
  - row changes in, 284–286
  - SQL Server and, 342, 347–348
  - XML representation of DataSet and, 293
- Disconnected data access model
  - ADO, 419
  - ADO.NET, 8
  - advantages of, 461
  - vs. connected data access model, 2, 26–27
- Disconnected mode, ADO.NET, 42–46, 196–197, 422–423
- Dispenser manager (DispMan), 116
- Distributed transactions
  - ADO.NET support for, 469
  - connected model and, 111–113
  - relational databases and, 462
- DL/1, 9–10

DLLs (Dynamic link libraries), 34  
 Document composition, XML, 334  
 document type definitions (DTDs), 14  
 DOM (Document Object Model), 15, 323  
 DSN (data source name) files, 71  
 DTDs (document type definitions), 14  
 Durability. *See* ACID (atomicity, consistently, isolation, and durability) properties  
 Dynamic link libraries (DLLs), 34

## E

EDI (Electronic Data Interchange), 468  
 EDIFACT (Electronic Data Interchange for Administration Commerce and Transport), 468  
 EJB (Enterprise Java Beans), 115, 468  
 Electronic Data Interchange (EDI), 468  
 Electronic Data Interchange for Administration Commerce and Transport (EDIFACT), 468  
 Embedded applications, ADO.NET support for, 16  
 Embedded DataTables, 173–174, 464  
 Enterprise Java Beans (EJB), 115, 468  
 EnterpriseServices transactions, 112–115  
 Entities, data representation, 5  
 Environment attributes, ODBC, 440  
 Environment handle, ODBC, 435  
 Error handling
 

- causing exceptions, 163–164
- connected model, 103–107
- connected model, catching errors and warnings, 106–107
- connected model, Sqlclient, 104–106
- Fill method and, 188–190
- list of errors thrown, 161–162
- OLE DB vs. ADO.NET, 405–406
- processing errors, 164–165
- types, 383

 Errors
 

- ADO.NET classes, 406
- JDBC vs. ADO.NET, 448
- ODBC vs. ADO.NET, 437
- OleDb data providers, 361

*Essential ASP.NET* (Onion), 233  
*Essential XML* (Box, Skonnard, and Lam), 16  
 Event handlers, 361  
 Event model
 

- refreshing DataSets, 216–218
- RowUpdating handlers, 214–215

turning off errors during batches, 215–216  
 Events
 

- list, 165–166
- sample program, 166–171

 Exceptions. *See also* DataExceptions; errors  
 ADO.NET classes, 406  
 catching, 32  
 error handling, 163–164  
 list of, 161–162  
 subclasses for OleDb and SqlClient, 103–104  
 Exchange mail stores, 401  
 ExecuteReader, IDbCommand, 374  
 ExecuteStream, SqlXml, 343  
 ExecuteToStream, SqlXml, 343–344  
 ExecuteXmlReader, 340–341  
 Expression syntax, 483–500
 

- aggregates, 487
- column and filter expressions, 483–484
- functions, 487–489
- operators, 485
- parent/child references, 486
- string operators, 486
- user-defined values, 484–485
- wildcard characters, 486

 Extensible Markup Language. *See* XML (Extensible Markup Language)

## F

Fetching data, ODBC, 436  
 Field indexers, 99–100  
 Fields, as array of objects, 99  
 File-based RDBMS, 461–464  
 Fill method
 

- error handling, 188–190
- how it works, 190–195
- overloads, 182–183
- pessimistic concurrency and, 181–182
- populating DataSets, 179–181
- schema and mapping information and, 183–188

 FillErrorEventArgs class, 188–189  
 FillSchema, 191, 436  
 Filter expressions, syntax, 483–484  
 Filtering and sorting
 

- ADO vs. ADO.NET, 424
- DataGrids, 264–265
- DataViews, 249–250

 Finally blocks, exceptions, 32



Find (ADO), vs. Select (ADO.NET), 424  
 First normal form, 7  
 Flat file systems  
   ADO.NET support for, 468–469  
   vs. relational models, 6–7  
 FMTONLY option, SQL Server, 194  
 FOR XML keyword, 335–336, 339  
 Foreign keys, 141, 492  
 Functions, 487–489

## G

GetChanges, 295–297  
 GetChildRows, 144  
 GetElementFromRow, 325–326  
 GetRows, 420–421  
 GIF (Graphics Interchange Format), 11–12  
 Globally unique identifiers (GUIDs), 6  
 Graphical user interface. *See* GUIs (Graphical user interfaces)  
 Graphics Interchange Format (GIF), 11–12  
 Grids. *See* DataGrid  
 GUIDs (Globally unique identifiers), 6  
 GUIs (Graphical user interfaces)  
   Data Form wizards, 270–271  
   data presentation, 226–227  
   table and column mappings, 254–256  
   Visual Studio integration, 266–270  
   Window Forms and Web Forms, 225–226  
 GUIs (Graphical user interfaces), data binding, 227–243  
   control types, 233–235  
   to DataReader, 241–243  
   with DataSet, 243–246  
   HTMLControls, 234–237  
   multiple results, 243  
   overview of, 227–230  
   Web Form, 230–233  
   WebControls, 235, 237–241  
 GUIs (Graphical user interfaces), DataGrid, 259–265  
   custom pager, 263–264  
   declarative code for, 260–262  
   overview of, 259–260  
   sorting, 264–265  
   updating, 262–263  
 GUIs (Graphical user interfaces), DataList, 256–259  
   declarative code for, 257–259

templates, 256–257  
 GUIs (Graphical user interfaces), DataViews, 246–254  
   DataViewManager, 252–254  
   DefaultView, 246–247  
   filtering and sorting, 249–250  
   modifying DefaultView, 251  
   nonrelational data and, 265–266  
   reading items, 249–250  
   restricting editing, 248–249  
   two views on same DataTable, 248

## H

Handles, ODBC vs. ADO.NET, 434–435  
 Heterogeneous data stores, 19  
 Heterogeneous hierarchies, 11  
 Hierarchical data  
   ADO vs. ADO.NET, 417  
   advantages of, 460  
   heterogeneous hierarchies, 11  
   homogeneous hierarchies, 9  
   OleDb data providers, 360  
   overview of, 1–2  
   parent-child structure of, 9  
   use of ADO.NET with, 464–465  
 Homogeneous hierarchies, 9, 464–465  
 HRESULTS, OLE DB error handling, 405–406  
 HTML (Hypertext Markup Language)  
   Infoset providers and, 383  
   screen scraping and, 14  
 HTMLControls, 234–237  
   HTMLSelect, 235–237  
   list of, 233–234  
 HTMLSelect, 235–237

## I

IBM, data access methods, 9  
 IDataAdapter, 179, 380  
 IDataReader  
   DataReader class, 94, 375–376  
   using with Shape provider, 100–102  
 IDataRecord  
   DataReader class, 94, 375–376  
   reading column values, 97  
   strongly typed accessors and, 98–99

- IDbCommand
    - ExecuteReader, 374
    - ExecuteScalar, 80
    - interface, 77, 371
    - parameterized queries and, 85
    - Prepare method, 92
  - IDbConnection
    - ADO vs. ADO.NET, 416
    - interface, 368–369
    - methods and properties, 369
    - System.Data namespace, 66
    - transaction method, 107
  - IDbTransaction, 107–109
  - Identity columns
    - retrieving, 211–212
    - updates and, 204
  - IDMS (Information Management System/Database), 9–10
  - IEnumerator, 379
  - IgnoreSchema, 285
  - ImportDataRow method, 148–150
  - in-memory database, 128
  - In-memory network library, 16
  - Indexed Sequential Access Method (ISAM), 9
  - InferSchema, 285
  - InfoMessageEvent, 105
  - Information Management System/Database (IDMS), 9–10
  - Information schema, DataSets, 174–176
  - Informix 9, 333
  - InfoSet. *See* XML Information Set (InfoSet) providers
  - INSERT method, DataTables, 148
  - InsertCommand
    - customizing, 206
    - DataAdapter Command instances, 179
    - retrieving identity columns with, 211–212
    - Update method and, 199
  - Interfaces. *See also* APIs (Application programming interfaces); GUIs (Graphical user interfaces)
    - DataReaders, 94
    - IDbCommand, 77, 368–369, 371
    - JDBC, 443
    - mapping JDBC to ADO.NET equivalents, 452–453
  - Interoperability
    - COM interoperability, 402
    - legacy code and, 25
    - Windows APIs and, 402
  - interoperability, ADO with ADO.NET, 425–433
    - ADO-format XML and, 428–433
    - connection strings and, 427–428
    - DataSets and RecordSets, 427
    - list of tips, 426
    - server cursors and, 425, 426–427
  - An Introduction to Database Systems* (Date), 7
  - ISAM (Indexed Sequential Access Method), 9
  - ISAPI application, SQL Server, 342
  - ISG Navigator, 352
  - Isolation. *See* ACID (atomicity, consistency, isolation, and durability) properties
- ## J
- Java Naming and Directory Interface (JNDI), 444–445
  - Java objects, 445–446
  - Java-relational databases (JDBMS), 46, 445–446
  - JDBC
    - Direct, 443–444
    - disconnected model, 456–457
    - drivers, 443–444
    - sample program, 455–456
  - JDBC programmers, ADO.NET for
    - commands and command behavior, 447
    - connection pooling, 445
    - disconnected model and, 456–457
    - errors, 448
    - example program, 455–456
    - generic code, 443
    - mapping classes to ADO.NET, 449–452
    - mapping data types to ADO.NET, 454–455
    - mapping interfaces to ADO.NET, 452–453
    - metadata, 446
    - nonrelational data types, 445
    - object databases, 445–446
    - provider types, 443–445
    - queries, 447–448
    - server cursors, 448
    - SQL-99 extensions, 446
    - SQL/J part O, 457–458
    - transactions, 446–447
  - JDBMS (Java-relational databases), 46, 445–446
  - Jet engine, 22, 463
  - JIT (Just-in-time) optimization, 12
  - JNDI (Java Naming and Directory Interface), 444–445
  - Join tables, 7
  - Just-in-time (JIT) optimization, 12

**K**

Key columns, relational data stores, 6  
 Keys, primary, 139–140

**L**

large object types. *See* LOBs (large object types)  
 LDAP readable directories, 273

## Libraries

ActiveX Template Library (ATL), 366  
 ADOX (ADO Extensions), 23  
 Client cursors, 41  
 COM, 425  
 designer libraries, 266  
 DLLs (Dynamic link libraries), 34  
 In-memory network library, 16  
 Jet Engine Library, 22  
 XML classes, 298

LIKE comparisons, 486

Listbound controls, 232–233

Lists. *See* DataList

LoadDataRow method, 148–150, 158–159

LOBs (large object types)

column values and, 101–103  
 object-relational mapping and, 466  
 storing, 11

Local transactions, relational databases, 462

Lookup tables, DataSets, 127

**M**

Managed data types

accessing, 358  
 JDBC and, 444  
 layout of, 65–67  
 mapping to OleDb and SqlClient, 379–380

Many-to-many relationships, 7

Mapping information

Fill method and, 183–188  
 tables and columns and, 186–188, 254–256

Marshaling

OLE DB vs. ADO.NET, 407  
 XML, 298–302

MDirProvDataReader, 377–379

MDX (Multidimensional Extensions to [SQL]), 13, 467

Merges, DataSets, 152–154

Metadata

collecting, 201

JDBC vs. ADO.NET, 446

methods, 74–77

multiple results and, 185–186

ODBC vs. ADO.NET, 436

relational databases and, 464

requesting from data providers, 194

Methods, mapping ODBC to ADO.NET,  
 437–439

Microsoft Access, 356, 420

Microsoft, data access APIs, 22

Microsoft Distributed Transaction Coordinator  
 (MSDTC), 111

Microsoft Foundation Classes, 434

Microsoft Management Console snap-in, 125

Microsoft-specific annotations, XML, 291–292

Microsoft Transaction Server (MTS), 115–116, 468

Microsoft Visual Studio. *See* Visual Studio

Migration paths, 401–402

Min Pool Size, 119–121

Move methods

ADO Recordset, 419

XPathNavigator, 393

MSDASQL, 352, 353

msdata:ConstraintName="Constraint1", 292

msdata:IsDataSet="true", 291–292

msdata:IsNested="true", 292

MSDE engine, 463

MSDTC (Microsoft Distributed Transaction Coordina-  
 tor), 111

MSOLAP provider, 360

MSSQLXML, 342

MTS (Microsoft Transaction Server), 468

connection pooling and, 115–116

Multidimensional data

ADO.NET and, 401, 467–468

OLAP and, 461

overview of, 12–13

Multidimensional Extensions to [SQL] (MDX), 13,  
 467

Multimember sets, 10

Mutator functions, SQL-99, 265–266

**N**

Named savepoints, 109–110, 462

Namespace

annotations, DataSet schema, 291

defining, 385

- Navigation
    - ADO disconnected model vs. ADO.NET, 419
    - facilities, 55–56
  - Navigation-based data access, 2
  - Navigation methods, DataSets
    - getting child rows, 144
    - hierarchies, 144–145
    - using Select and Compute, 145–146
  - Nested DataRelations, 282–283
  - Nested transactions, 109–110, 462
  - Netlibs, SQL Server, 17
  - Network data, 465–466
  - NO\_BROWSETABLE option, SQL Server, 194
  - Nonrelational data, 171–174
    - data providers and, 465
    - DataAdapters and, 220–221
    - DataViews and, 265–266
    - embedded DataTables and, 173–174
    - examples of, 273
    - JDBC vs. ADO.NET, 445
    - object types and, 172–173
  - Normal form rules, 7
  - NOT operator, 485
  - NT File System, 273
  - Numeric values, 484–485
- O**
- Object Data Management Group (ODMG), 466
  - Object database management systems (ODBMS), 12, 465–466
  - Object models
    - ADO, 409
    - XML Information Set (Infoset), 400
  - Object-oriented databases
    - data structures benefiting from, 460
    - JDBC vs. ADO.NET, 445–446
    - popularity of, 399
  - Object Query Language (OQL), 3
  - Object-relational database management systems (ORDBMS), 12, 464–465
  - Object-relational mapping, 466
  - Object-relational model, 460
  - Object types, DataSets, 172–173
  - ODBC
    - advantages of, 433–434
    - client-side pooling in, 18
    - data providers, 69, 356
    - legacy providers and, 397
    - mapping data types to ADO.NET, 476–478
    - OLE DB provider for, 352
    - relationship to JDBC, 443
    - universal data access and, 22, 398
  - Odbc data provider
    - example program, 441–442
    - metadata and schema information and, 436
  - ODBC-JDBC bridge, 443
  - ODBC programmers, 433–442
    - benefits of ODBC, 433–434
    - commands, 435–436
    - errors, 437
    - example program, 441–442
    - fetching data, 436
    - handles and environment, 434–435
    - mapping ODBC attributes to ADO.NET, 440–441
    - mapping ODBC methods to ADO.NET, 437–439
    - metadata and schema information, 436
  - OdbcCommand, 435–436
  - OdbcConnection, 435
  - OdbcDataReader, 436
  - OdbcExceptions, 437
  - OdbcHandleError, 437
  - OdbcType enumeration, 476–478
  - ODBMS (Object database management systems), 12, 465–466
  - ODMG (Object Data Management Group), 466
  - OLAP (Online analytical processing)
    - data analysis process, 13
    - OLE DB for, 467
    - providers, 360
  - OLE DB
    - ADO and, 408
    - classes and purposes, 354–355
    - client-side pooling, 18
    - COM-based data access, 23
    - components, 406–407
    - development of, 398–399
    - generic code, 404
    - heterogeneous and homogeneous hierarchies, 355
    - legacy providers, 397
    - mapping data types to ADO.NET, 474–476
    - mapping to DataSets, 364–366
    - mapping to OleDb data providers, 362–364
    - object model, 352–355
    - properties, exposed in ADO, 417
    - services, 406

- OLE DB programmers, 402–407
    - cotype equivalents, 403–404
    - data provider transparency, 404–405
    - error handling, 405–406
    - marshaling, 407
    - provider-specific functionality, 405
    - service providers, 407
    - system-supplied components, 406–407
    - system-supplied services, 406
  - OleDbCommand, 77
  - OleDb data providers
    - accessors, 358
    - command dialects, 359–360
    - command execution, 358–359
    - command preparation, cancellation, and cleanup, 92–94
    - command results, 359
    - connected mode, 34–35, 36, 69
    - connection pooling, 73, 118
    - cotypes and type mapping, 357
    - data access methods and, 27
    - DbDataAdapter base class, 179
    - errors and unsupported functions, 361
    - exceptions, 103–104
    - hierarchical data, 360
    - IDbCommand and, 374
    - mapping to managed types, 379–380
    - mapping to OLE DB, 362–364
    - metadata information, 74–76
    - overview of, 356–357
    - parameters, 88–90
    - retaining existing, 352
    - supported and unsupported providers, 361–362
    - transparency, 405
    - types, 351–352
    - updating, 360–361
  - OleDbCommand object, 358
  - OleDbCommandBuilder class, 200
  - OleDbConnection, 416
  - OleDbDataAdapter, 195–196
  - OleDbDataReader, 358
  - OleDbPermission, 125
  - OleDbTransaction, 107–109
  - OleDbType enumeration, 474–476
  - OleTX, 111–112
  - Online analytical processing. *See* OLAP (Online analytical processing)
  - Open method, 71
  - OpenRowset, 359
  - OpenXML, 336
  - Operators, 485
  - Optimistic concurrency. *See also* Disconnected data
    - access model
    - conflict resolution and, *see*, 177–178
    - DataSets and, 462–463
    - defined, 177
    - updates, 178, 207
  - Optimistic data models. *See* Disconnected data
    - access model
  - OQL (Object Query Language), 3
  - OR operator, 485
  - Oracle, 182
  - ORDBMS (Object-relational database management systems), 12, 464–465
  - Ordering, relational model, 6
  - Output parameters, 84–85
  - Overload methods, XML, 279–280, 283
- P**
- Paging, DataGrid, 263–264
  - Parameter classes, 382
  - ParameterCollection, 382
  - Parameter.DbType, 479–481
  - Parameterized statements, 4, 82–83, 206
  - Parameterized updates, 188
  - Parameters, connected model, 82–91
    - adding to collections, 86–87
    - OleDb data provider and, 89–90
    - overview, 82–83
    - result and output parameters, 84–85
    - SQL Server provider and, 89–90
    - SqlParameter, 87–88
    - stored procedures, 85–86, 91
  - Parent-child. *See also* Child elements
    - grids, 253
    - references, 486
  - Permission classes, 124–126
  - Permission types, 383
  - PermissionAttribute, 124
  - Persistent stored modules. *See* Stored procedures
  - Person class, 301–302
  - Pessimistic concurrency. *See also* Connected data
    - access model
    - DataSets and, 462
    - defined, 177
    - Fill method and, 181–182
  - Physical database records, 9

Platform invoke (PInvoke), 25, 402  
 Pooling. *See* Session pooling  
 Prepare method, IDbCommand, 92  
 Primary keys. *See also* constraints  
   defining, 139–140  
   populating, 191  
   relational data model and, 7  
   updating, 159  
 Programmers, ADO. *See* ADO programmers  
 Programmers, JDBC. *See* JDBC programmers  
 Programmers, ODBC. *See* ODBC programmers  
 Programmers, OLE DB. *See* OLE DB programmers  
 Properties, ADO vs. ADO.NET, 417  
 Property Builder, 270  
 PropertyCollection, DataSets, 129

## Q

### Queries

JDBC vs. ADO.NET, 447–448  
 large relational databases, 462  
 multimedia data types, 467  
 parameters for, 4–5, 82–83  
 XPathNavigator methods, 395

## R

RCWs (Runtime callable wrappers), 25  
 RDBMS (Relational database management systems)  
   ADO.NET and, 2, 461–464  
   SQL commands and, 3–4  
 RDO (Remote Data Objects), 22  
 RDS (Remote Data Services), 23, 351, 407  
 ReadSchema, 286  
 ReadXml, 284–287  
 Recordsets  
   ADO-format XML, 428–432  
   creating, 418  
   data exchange with, 351  
   disconnected, 418–419  
   extensions, 423  
   GetRows method, 420  
   interoperability with DataSets, 427–428  
   navigation, 419  
   update support in, 422  
   vs. DataSets, 417–418  
 RejectChanges, 287–288  
 Relational data, mapping XML to, 277

Relational database management systems. *See*  
 RDBMS (Relational database management systems)

Relational databases  
   restricted formats of, 333  
   vs. DataSets, 127

Relational model, 5–8  
   advantages of, 7–8, 460  
   design rules for, 6–7  
   features of, 5–6  
   support in ADO.NET, 8  
   vs. flat file systems, 6–7

Relationships. *See also* DataRelations  
   counting, 142–143  
   defining, 141  
   navigating through, 143–146  
   relational data model and, 7  
   rules, 159, 496–498  
   types of, 143

Remote Data Objects (RDO), 22

Remote Data Services (RDS), 23, 351, 407

Remote method calls, SOAP, 46

Remote procedure calls (RPCs), 46

Repeater control, WebControls, 239–241

Request for Comments (RFCs), 51

Result parameters, 84–85

RFCs (Request for Comments), 51

Rollback transactions, 107

Round-trip updates, XML, 297–298

Rows (tuples), relational model, 5. *See also*  
 DataRows

### Rowset

  disconnected model and, 456–457

  updating from, 360–361

  vs. DataSets, 452–453

RowState. *See* DataRowState

RowUpdating events, 214–216

RPCs (Remote procedure calls), 46

Runtime callable wrappers (RCWs), 25

Runtime.Serialization, 465

## S

Savepoints, named, 110, 462

Schema inference rules, 491–500

  columns, 495–496

  excluding information, 491

  lack of determinism in, 499–500

- relationships, 496–498
- summary of, 492
- tables, 492–495
- text, 498–499
- Schema information
  - changing using column mapping, 280–281
  - Fill method and, 183–188
  - ODBC vs. ADO.NET, 436
- Schema Object Model (SOM), 54
- Schemaless data, 14
- Screen scraping, 14, 273
- Scrolling-rowset presentation, 227
- SDK documentation, data providers and, 366
- Second normal form, 7
- Security
  - code access security, 125
  - database security, 126
- Segments, hierarchical databases, 9
- Select method, DataSets
  - navigation with, 145–146
  - returning an iterator with, 59
  - vs. ADO Find, 424
- SelectCommand, 179, 201
- Self-referencing tables, 7
- Semistructured data
  - ADO.NET support for, 468–469
  - defined, 14
- Serialization, XML, 298–302
  - ADO.NET libraries for, 298–299
  - ISerializable, 299–301
  - object databases and, 278
  - object persistence and, 465–466
- Server-based RDBMS, 461–464
- Server cursors
  - GetRows method, 420
  - JDBC vs. ADO.NET, 448
  - processing in C#, 425–426
  - processing in VB.NET, 426–427
  - unsupported by ADO.NET, 41–42, 70
  - using in ADO.NET, 421
- Server Explorer, Visual Studio, 267
- Server-side cursors. *See* Server cursors
- Service providers
  - OLE DB support for, 353–354
  - OLE DB vs. ADO.NET, 407
- Session pooling, 18
- Set-based data access, 2, 231
- Simple Object Access Protocol (SOAP), 46, 461
- Single-record presentation, 226–227
- Skonnard, Aaron, 384
- SOAP (Simple Object Access Protocol), 46, 461
- SOM (Schema Object Model), 54
- Sorting and filtering
  - ADO vs. ADO.NET, 424
  - DataGrids, 264–265
  - DataViews, 249–250
- Special characters, 483–484
- SQL-99 standard
  - DataSet schema, 174–175
  - JDBC and, 445, 446
  - Mutator functions, 265–266
- SQL-CLI (SQL Call Language Interface), 19
- Sql exception types, 32
- SQL/J part O, JDBC, 457–458
- SQL Query Builder, 268–269
- SQL Server
  - bulk loading, 339
  - concurrency checking, 197–198
  - cursors, 420–421
  - Fill method and, 193–194
  - FOR XML keyword, 335–336
  - ISAPI application, 342
  - mapping data types to ADO.NET, 473–474
  - nonrelational data, 465
  - OLE DB and, 356
  - OpenXML and, 336
  - pessimistic concurrency, 181
  - provider parameters, 89–90
  - Server Explorer connection to, 267–268
  - SQLOLEDB provider, 336
  - SqlXml managed classes, 337
  - SqlXml Web application, 337
  - SQLXMLOLEDB Provider, 339
  - updategrams, 337–338
  - updating with DiffGram, 347–348
- SQL (Structured Query Language)
  - accessing data via, 1
  - commands, examples, 4
  - commands, filling DataTables from, 191
  - commands, list of basic, 3
  - commands, syntax of, 3–4
  - connection strings, 27–29
  - dialects, 463
  - information schema, 74–75
  - use of SQL commands by APIs, 3

- SqlClient data providers
    - command preparation, cancellation, and cleanup, 92–94
    - connected mode, 34–35, 69
    - connection pooling, 73, 118
    - data access methods and, 27–28
    - DbDataAdapter base class, 179
    - error handling, 104–106
    - exceptions, 103–104
    - ExecuteXmlReader support, 340–341
    - IDbCommand and, 374–375
    - mapping to managed types, 379–380
    - metadata information, 76–77
    - parameter specification, 88–90
    - SQL Server and, 366
  - SqlClientPermission, 125
  - SqlCommand, 77
  - SqlCommandBuilder class, 43, 200, 202–205
  - SqlConnection, 32, 416
  - SqlDataAdapter class, 43–44
  - SqlDataReader
    - binding DataGrid to, 241–242
    - connected mode, 32–33
    - disconnected mode, 43
    - getting fields as arrays of objects, 99
    - using field indexers, 99–100
  - SQLFetch, 436
  - SQLGetData, 436
  - SQLOLEDB provider, 336, 354
  - SqlParameter, 87–88
  - SqlTransaction, 107–109
  - SQL\_VARIANT type, 220
  - SQLXML, 339–348
    - ExecuteStream, 343
    - ExecuteToStream, 343–344
    - ExecuteXML Reader, 340–341
    - SqlXmlAdapter, 346
    - XPath query and, 344–346
  - SqlXml data provider, 366
  - SqlXml managed classes, 337, 359–360
  - SqlXml Web application, 337
  - SqlXmlAdapter, 346
  - SQLXMLOLEDB Provider, 339
  - Standards. *See also* SQL-99 standard
    - ANSI standards, SQL schema, 74
    - SQL command language, 3
    - W3C standards, XSD, 52–53
  - State
    - of database entities, 5
    - DataRowState, 154–158, 249–250
  - Statement attributes, ODBC, 440–441
  - Stored procedures
    - adding to databases, 334
    - applying as CommandText, 86
    - calling, 91
    - client-server applications and, 17
    - code examples of, 85–86
    - DeriveParameters and, 205–206
    - parameters, 85, 91
    - queries, 83
    - SQL statements and, 3
  - Streaming data, DataReaders, 94–97
  - Streaming mode, SQL Server, 95
  - String operators, 486
  - String values, 484
  - Strong typing
    - accessors, 98–99
    - DataSets, 304–305
    - JDBC and, 457
  - Structured file systems, 467–468
  - Syntax. *See* Expression syntax
  - System. Web.Services, 461
  - System. Windows.Forms, 461
  - System.Data namespace, 66
  - System.Runtime.Serialization, 13, 298–301
  - System.Xml, 461
  - System.Xml.dll, 67
  - System.Xml.Serialization, 13, 298–301
- T**
- Tables. *See also* DataTables
    - elements with attributes are inferred as tables, 492–493
    - elements with child elements are inferred as tables, 493–494
    - mapping information, 186–188
    - names, ADO vs. ADO.NET, 424
    - repeating elements are inferred as single table, 494–495
  - TDS (Tabular data stream) packets, 29, 193
  - Templates, DataList, 256–257
  - Text, schema inference rules, 498–499
  - Third normal form, 7



- Three-tier applications, 17–19
- Timestamps, 204, 209–211
- TLBIMP, 467
- Transactions
  - ADO vs. ADO.NET, 416
  - extended functionality with, 382
  - JDBC vs. ADO.NET, 446–447
  - relational databases and, 462
- Transactions, connected model
  - declarative, 121–124
  - distributed, 111–113
  - EnterpriseServices and, 112–115
  - named savepoints, 110
  - nested, 109–110
  - simple example, 107–109
- Transformations, XSLT, 59–60, 400
- Try-catch-finally blocks, exceptions, 32
- Tuples (rows), in relational model, 5
- type mapping, OleDb data providers, 357
- Typed DataSets
  - applying, 317–322
  - defined, 302–303
  - producing input for, 305
  - strong typing, 304–305
  - subclass generated by XSD.exe, 305–317
  - table and column references, 303

**U**

- UDA (universal data access), 19–21, 398–400
- UDFs (User defined functions), 83, 334
- UDL (universal data link) files, 71
- UDTs (User-defined types), 334
- Uniform resource identifier (URI), 51
- Uniform resource locator (URL), 51
- Universal data access (UDA), 19–21, 398–400
- Universal data link (UDL) files, 71
- Universal servers, 19–20, 399
- Update method
  - CommandBuilder class, 199–206
  - concurrency checking, 197–198
  - ContinueUpdateOnError property, 198–199
  - DataAdapter class, 45
  - databases and, 177–178
  - DataTables and, 148
  - disconnected mode, 196–197
  - hand coded example, 207–209

- how it works, 212–214
  - retrieving identity columns, 211–212
  - specifying row version, 206–207
  - timestamp construction, 209–211
- Update statements, 423
- UpdateCommand
  - customizing, 206
  - DataAdapter Command instances, 179
  - Update method and, 199
- Updategrams, 337–338, 342
- Updates
  - ADO vs. ADO.NET, 422–423
  - batch updates, 423
  - cascading updates, 160–161
  - CommandBuilder class and, 202–205
  - DataSets, 150–152
  - disconnected mode, 42–46
- URI (uniform resource identifier), 51
- User defined functions (UDFs), 83, 334
- User-defined types (UDTs), 334
- User-defined values, 484–485

## V

- VB.NET, 426–427
- Visual C++, 366
- Visual Studio
  - concurrency and, 463
  - integration with, 266–270, 398
  - typed DataSets and, 304–305
  - XML schema support, 53

## W

- W3C (World Wide Web Consortium), 52
- Web browsers, 27, 42
- Web Forms
  - binding data to arrays, 232
  - binding List box to SqlDataReader, 231–232
  - as data binding control class, 228–230
  - overview of, 225–226
  - types, 230–233
  - vs. Window Forms, 226
- Web Service Description Language (WSDL), 461
- WebControls, 237–241
  - data binding, 235
  - DataFormatString Property, 238–239

- WebControls (*continued*)
    - list of, 235
    - overview of, 237–238
    - Repeater control, 239–241
  - WebDav, 427
  - What enumeration, XmlReader, 391
  - Wildcard characters, 486
  - Window Forms
    - as data binding control class, 228–230
    - overview of, 225–226
    - vs. Web Forms, 226
  - Windows registry, mapping to XML, 384–385
  - Windows (Win32), migration to ADO.NET, 402
  - World Wide Web Consortium (W3C), 52
  - WSDL (Web Service Description Language), 461
- X**
- XBase applications, 17, 420
  - XDR (XML Data Reduced)
    - as predecessor of XSD, 277
    - specifying DataSet schema with, 491
    - XML schema support, 52
  - XML APIs, 46–65
    - overview of, 46–47
    - XML, mixing with data providers, 60–65
    - XML schemas, 51–54
    - XmlDocument, XPath, and XPath Navigators, 54–60
    - XmlReader, 47–51
  - XML Data Reduced (XDR)
    - as predecessor of XSD, 277
    - specifying DataSet schema with, 491
    - XML schema support, 53
  - XML DataBlade, 333
  - XML Extender, 333
  - XML (Extensible Markup Language)
    - ADO classic and, 277
    - advantages, document composition, 353
    - advantages, universal support, 333
    - advantages, XML as distinct type, 333–334
    - bridging data models with, 2
    - features of, 14–15
    - flat files and semistructured data and, 6, 468
    - interoperability with nonrelational data, 13
    - mapping to relational data, 277
    - mapping to Windows registry, 384–385
    - method for reading XML data, 279–280
    - Microsoft-specific annotations, 291–292
    - portability and, 460
    - serialization and marshaling, 298–302
    - universality of, 15–16
    - XSD schemas and, 277–279
  - XML (Extensible Markup Language), DataSets and defining schema, 274–277
    - reading XML to, 283–288
    - refining schema, 280–283
    - writing XML data from, 292–298
    - writing XML schemas from, 288–291
  - XML (Extensible Markup Language), SQL Server and bulk loading, 339
    - ISAPI application, 342
    - OpenXML, 336
    - SQLOLEDB provider, 336
    - SqlXml managed classes, 337
    - SqlXml Web application, 337
    - SQLXMLOLEDB Provider, 339
    - updategrams, 337–338
    - updating with DiffGram, 347–348
    - FOR XML keyword, 335–336
  - XML (Extensible Markup Language), SQLXML and
    - ExecuteStream, 343
    - ExecuteToStream, 343–344
    - ExecuteXML Reader, 340–341
    - SqlXmlAdapter, 346
    - XPath query, 344–346
  - XML (Extensible Markup Language), typed DataSets
    - and
      - applying, 317–322
      - defined, 302–303
      - input for, 305
      - strong typing, 304–305
      - subclass generated by XSD.exe, 305–317
      - table and column references, 303
  - XML for Analysis specification, 13, 467
  - XML for Data Mining, 467
  - XML Information Set (Infoset) providers, 383–396
    - DTDs and, 14
    - not limited to homogeneous data, 324
    - as object models, 400
    - overview of, 383–386
    - XmlReader, implementing, 386–392
    - XPathNavigator, implementing, 392–396
  - XML Schema Definition Language. *See* XSD (XML Schema Definition Language)

- XML schemas
  - compiler, 274
  - as data definition language (DDL), 400
  - example, 62–65
  - refining for DataSet, 280
  - writing from DataSet, 288–291
  - XML APIs and, 51–54
- XML transformation language. *See* XSLT (XML transformation language)
- XmlDataDocument, 322–333
  - bridging data models with, 2
  - creating, 326–327
  - DataSets and, 324
  - disconnected model and, 8
  - GetElementFromRow, 325–326
  - loading, 327–328
  - merging data, 328–329
  - merging documents, 329–331
  - missing columns, 323–324
  - mixed content, 324
  - overview, 322–324
  - XPathNavigator and, 331–333
- XmlDocument
  - members, 56–57
  - navigation methods, 55–56
  - populating, 397
  - as XML APIs, 54–60
- XmlNode
  - document traversal, 56
  - DOM model and, 323
  - members, 56–57
- XmlNodeType, 387–388
- XmlReader, 386–392
  - customizing, 392, 396–397
  - document properties, 388–389
  - exposing data, 384–386
  - functional areas of, 387
  - hierarchy levels and, 389
  - mapping to XPathNavigator, 394
  - node recognition by, 387–388
  - properties and methods, 388
  - SqlClient and, 339
  - streaming XML, 47–51
  - What enumeration, 391
- XmlReadMode, 285–286
- XmlSerializer, 300
- XmlTextReader, 48–51
- XmlTextWriter, 48–51
- XmlWriter, 48–49
- XPath
  - nodes or nodesets, 323
  - overview of, 54–60
  - queries, 54–55, 65, 344–346
  - as XML API, 14–15
- XPathNavigator, 392–396
  - applying, 331–332
  - bridging data models with, 2
  - customizing, 396–397
  - DataDocumentXPathNavigator and, 331
  - exposing data, 384–386
  - in-place updating, 396
  - loading, 57–58
  - mapping to XmlReader, 394
  - members for document traversal, 58
  - Move methods, 393
  - navigating XML documents, 323
  - optimizing, 395–396
  - query methods, 395
  - updating with, 332–333
  - as XML API, 15, 54–60
  - XPathNodeTypes and, 392–393
  - XSLT and, 59–60
- XPathNodeTypes, 392–393
- XSD (XML Schema Definition Language)
  - flexibility of, 278
  - specifying DataSet schema with, 491
  - vs. DTDs, 14
  - XDR as predecessor of, 277
  - XML integration with ADO.NET, 277–279
- XSD.exe
  - object and relational paradigms and, 466
  - strongly typed subclass, 53
  - typed DataSets, 302, 305–317
- XSLT (XML transformation language)
  - as transformation language, 400
  - transforming XML documents with, 323
  - XPathNavigator and, 59–60

