

Inhalt

Vorwort	23
---------------	----

1 Installation und Aktualisierung von SQL Server 2008 25

1.1 Überblick über die verfügbaren SQL-Server-2008-Versionen	25
1.2 Installationsvoraussetzungen	26
1.3 Installation von SQL Server 2008	27
1.4 Installation der SQL-Server-Beispieldatenbanken	39
1.5 Aktualisierung auf SQL Server 2008	45
1.5.1 Der Upgrade Advisor	45
1.5.2 Durchführung der Aktualisierung	50

2 Datenbankgrundlagen 57

2.1 Server und Client	57
2.2 Relationale Datenbanken	58
2.2.1 Struktureller Aufbau von Tabellen	61
2.2.2 Im Zusammenhang mit Tabellen gebräuchliche Fachbegriffe	63
2.2.3 Schlüssel	64
2.2.4 Beziehungsarten	73
2.3 Datenbankmanagementsystem	77
2.4 Integritätsarten	80

3 Logischer Datenbankentwurf 83

3.1 Grundlagen des Datenbankentwurfs	83
3.2 Normalisierungsregeln (Normalformen)	84
3.2.1 Erste Normalform	86
3.2.2 Zweite Normalform	88
3.2.3 Dritte Normalform	90
3.3 Normalisierung in der Praxis	91
3.4 Denormalisierung	95
3.5 Entity-Relationship-Diagramme	96

4 Die Oberfläche des SQL Servers 99

4.1 Das SQL Server Management Studio	99
4.1.1 Der Objekt-Explorer	107

4.2	Das Bearbeiten von SQL-Anweisungen im Management Studio ...	118
4.2.1	Der SQL-Abfrage-Editor	118
4.2.2	Der Ergebnisbereich	129
4.2.3	Der Abfrage-Designer	132
4.3	SQL-Server-Dokumentation und Hilfen zur Programmierung	135
4.3.1	Die SQL-Server-Onlinedokumentation	135
4.3.2	Dynamische Hilfe	137
4.3.3	SQL-Server-IntelliSense	138
4.3.4	Der Vorlagen-Explorer	140
4.3.5	Die Skript-Generierung	144
4.3.6	Die Verwendung von Lesezeichen	145
4.4	Überblick über die SQL-Server-Dienste	148
4.5	Das Dienstprogramm sqlcmd	149

5 Grundlegende Abfragetechniken 153

5.1	Einführung in SQL	153
5.2	SQL-Grundlagen	154
5.2.1	Formatierung	155
5.2.2	Bezeichner	155
5.2.3	Systemdatentypen	156
5.2.4	Neue Datentypen in SQL Server 2008	160
5.2.5	Informationen zu den Datentypen einer Tabelle	161
5.2.6	Verwendung und Kennzeichnung von Zahlen, Zeichenketten und Datum-/Uhrzeitwerten	162
5.2.7	Kommentare	163
5.3	Grundlegende Operatoren	163
5.3.1	Arithmetische Operatoren	164
5.3.2	Zeichenkettenoperator	164
5.3.3	Vergleichsoperatoren	164
5.3.4	Logische Operatoren	165
5.4	Einfache Abfragen	167
5.4.1	USE	167
5.4.2	SELECT...FROM	168
5.4.3	Berechnete Spalten	171
5.4.4	DISTINCT	173
5.4.5	ORDER BY-Klausel	174
5.4.6	WHERE-Klausel	177
5.5	Auswahloperatoren	182
5.5.1	IS (NOT) NULL-Auswahloperator	182

5.5.2	BETWEEN-Auswahloperator	184
5.5.3	IN-Auswahloperator	185
5.5.4	LIKE-Auswahloperator	186
5.6	Aggregatfunktionen	190
5.7	Unterabfragen	193
5.8	Zusammenfassung, Gruppierung und Beschränkung von Abfrageergebnissen	196
5.8.1	Verknüpfung mehrerer Abfrageergebnisse durch UNION	196
5.8.2	Beschränkung des Abfrageergebnisses mit TOP (n)	198
5.8.3	GROUP BY-Klausel	201
5.9	Ausgaben mit PRINT und SELECT	205

6 Grundlagen der SQL-Programmierung 207

6.1	Das Stapeltrennzeichen GO	207
6.2	(Lokale) Variablen	209
6.2.1	Variablendeklaration	209
6.2.2	Wertezuweisung an eine Variable	210
6.2.3	Kombinierte Deklaration und Wertezuweisung	211
6.2.4	Inkrement und Dekrement	211
6.2.5	Gültigkeitsbereich von Variablen	212
6.2.6	Verwendung von Variablen	213
6.2.7	Lokale und globale Variablen in SQL Server	213
6.2.8	Wertezuweisung an Variablen durch Abfragen	214
6.3	Ablaufsteuerung	217
6.3.1	Blöcke	217
6.3.2	IF...ELSE	218
6.3.3	IF EXISTS	221
6.3.4	WHILE-Schleife	222
6.3.5	RETURN	225
6.3.6	GOTO	225
6.3.7	WAITFOR	226
6.4	Fallunterscheidungen	227
6.4.1	Die CASE-Funktion	228
6.4.2	Die ISNULL-Funktion	233
6.5	Funktionen	234
6.5.1	Konfigurationsfunktionen	234
6.5.2	Datums-/Uhrzeitfunktionen	235
6.5.3	Zeichenfolgenfunktionen	238

6.5.4	Mathematische Funktionen	242
6.5.5	Funktionen zur Statusabfrage	242
6.5.6	Konvertierungsfunktionen	243
6.6	Dynamische SQL-Anweisungen	246
6.6.1	Ausführung dynamischer Anweisungen mit EXECUTE	247
6.6.2	Ausführung dynamischer Anweisungen mit sp_executesql	247
6.7	Fehler in SQL Server und ihre Behandlung	248
6.7.1	Struktur von Fehlermeldungen in SQL Server	248
6.8	Fehlerbehandlung	251
6.8.1	Fehlerbehandlung mit TRY und CATCH	251
6.8.2	Funktionen zur Fehlerbehandlung	255
6.8.3	Gegenüberstellung von TRY...CATCH und @@ERROR ...	258
6.8.4	Abbruchverursachende Fehler	260
6.8.5	Erstellung benutzerdefinierter Fehlermeldungen	260
6.8.6	Programmgesteuerte Fehlerauslösung	262
6.8.7	Fehlerauslösung unter Verwendung von Parametern	264
6.8.8	Löschen von benutzerdefinierten Fehlermeldungen	267

7 Multitabellenabfragen 269

7.1	Tabellen verknüpfen (JOIN-Anweisung)	269
7.1.1	Einbindung zweier Tabellen in die SELECT-Anweisung	269
7.1.2	Innere Verknüpfung (INNER JOIN)	271
7.1.3	Äußere Verknüpfung (OUTER JOIN)	274
7.1.4	Mehrfache JOIN-Anweisungen	275
7.1.5	Kreuzverknüpfung (CROSS JOIN)	276
7.1.6	Umkehrung der Funktionsweise des INNER JOINs	277
7.2	Abfragen unter Verwendung mehrerer Server	279
7.2.1	Vollgekennzeichnete (vollqualifizierte) Namen	279
7.2.2	Einrichtung eines Verbindungsservers	279

8 Erstellen und Ändern von Datenbanken 281

8.1	Erstellen einer einfachen Datenbank	281
8.2	Einfluss der model-Datenbank auf das Erstellen neuer Datenbanken	289
8.3	Löschen von Datenbanken	290
8.4	Erstellen einer Datenbank mit mehreren Dateien	290
8.5	Nachträgliche Änderungen an Datenbankdateien	293
8.5.1	Dateien hinzufügen	293

8.5.2	Dateieigenschaften ändern	295
8.5.3	Dateien löschen	296
8.6	Erstellen einer Datenbank mit mehreren Dateigruppen	296
8.7	Datenbanken verkleinern	299
8.8	Datenbankdateien verkleinern	301
8.9	Gespeicherte Systemprozeduren (Stored Procedures) zur Datenbankverwaltung	302
8.10	Datenbank-Snapshots	303
8.10.1	Erstellen eines Datenbank-Snapshots	303
8.10.2	Interne Verwaltung eines Snapshots	305
8.10.3	Erstellen eines Snapshots für eine Datenbank mit mehreren Datendateien	309
8.10.4	Weitere Informationen zu Snapshots	311
8.10.5	Datenbank-Snapshot löschen	311
8.11	RAID-Systeme	312
8.11.1	RAID 0	312
8.11.2	RAID 1	313
8.11.3	RAID 3	314
8.11.4	RAID 5	315

9 Erstellen von Tabellen 317

9.1	Die grundlegende Syntax zur Tabellenerstellung	317
9.1.1	Definition einer Identitätsspalte	319
9.1.2	Bestimmung des physischen Speicherorts	320
9.2	Tabellen ändern	321
9.2.1	Spalten hinzufügen (ADD)	321
9.2.2	Spalten ändern (ALTER COLUMN)	322
9.2.3	Spalten löschen (DROP COLUMN)	323
9.3	Löschen von Tabellen	323
9.4	Implementierung der Datenintegrität	324
9.4.1	Die Einschränkungsarten im Überblick	324
9.4.2	Definition von Einschränkungen	324
9.4.3	Eigenschaften von Constraints	326
9.4.4	Lösch- und Änderungsweitergabe	330
9.5	Anwendungsbeispiel zu Einschränkungen	331
9.6	Nachträgliche Definition von Einschränkungen	333
9.7	Verwaltung von Einschränkungen	334
9.7.1	Deaktivierung und Aktivierung von Einschränkungen	335
9.7.2	Löschen einer Einschränkung	335

9.8	Temporäre Tabellen	336
9.8.1	Lokale temporäre Tabellen	336
9.8.2	Globale temporäre Tabellen	337
9.9	Partitionierung von Tabellen	338
9.9.1	Erstellung einer Partitionsfunktion	340
9.9.2	Erstellung eines Partitionsschemas	341
9.9.3	Erstellung einer partitionierten Tabelle	343

10 Verwendung der räumlichen und hierarchischen Datentypen 349

10.1	Räumliche Datentypen	349
10.1.1	Der geometry-Datentyp	350
10.1.2	Der geography-Datentyp	350
10.1.3	Koordinatenübergabe an räumliche Datentypen	351
10.1.4	Das Klassenmodell räumlicher Datentypen	351
10.1.5	Grundlegende Techniken zur Verwendung räumlicher Datentypen	357
10.1.6	Allgemeine Verwendung räumlicher Datentypen	360
10.1.7	Methoden zur Verarbeitung räumlicher Daten	361
10.2	Der hierarchische Datentyp	372
10.2.1	Die Methoden des hierarchischen Datentyps	374

11 Daten verwalten 381

11.1	Grundlegende Befehle zur Datensatzmanipulation	381
11.1.1	Einfügen von Datensätzen mit INSERT	381
11.1.2	Daten aktualisieren mit UPDATE	386
11.1.3	Löschen von Daten mit DELETE	389
11.2	Verwendung des Zeilenkonstruktors	390
11.3	Kombinierte Auswahl-/Einfüganweisungen	391
11.3.1	INSERT...SELECT	391
11.3.2	SELECT...INTO	392
11.4	Die MERGE-Anweisung	393
11.4.1	Grundlagen der MERGE-Anweisung	393
11.4.2	Filterung von Datensätzen	397
11.4.3	Die OUTPUT-Klausel	398
11.5	Massenkopierprogramme	400
11.5.1	BULK INSERT	400
11.5.2	bcp	401

12 Benutzerverwaltung und Schemas 403

12.1	Authentifizierung am Server	404
12.1.1	Einrichten eines Log-ins mit Windows-Authentifizierung	405
12.1.2	Einrichten eines Log-ins mit SQL-Server-Authentifizierung	406
12.2	Datenbankzugriff erteilen	408
12.3	Zusammenfassung von Benutzern zu Rollen	411
12.3.1	Serverrollen	412
12.3.2	Feste Datenbankrollen	413
12.3.3	Einen Benutzer zu einer Rolle hinzufügen	414
12.4	Rechtevergabe an Benutzer und Rollen	415
12.4.1	Objektberechtigungen	415
12.4.2	Anweisungsberechtigungen	415
12.4.3	SQL-Anweisungen zur Rechtevergabe	417
12.5	Deaktivierung und Aktivierung von Log-ins und Datenbankbenutzern	421
12.6	Vordefinierte Konten auf Server- und Datenbankebene	424
12.6.1	Das Log-in »system administrator« (»sa«)	424
12.6.2	Der Datenbankbenutzer »guest«	425
12.7	Schema	426
12.7.1	Objektverwaltung bis SQL Server 2000	427
12.7.2	Objektverwaltung seit SQL Server 2005	428
12.7.3	Erstellen eines Schemas	428
12.7.4	Namensauflösung bei fehlender Schema-Angabe	430

13 Sichten 433

13.1	Einsatz von Sichten	434
13.2	Verwalten von Sichten	436
13.2.1	Erstellen einer Sicht	436
13.2.2	Ändern einer Sicht	438
13.2.3	Löschen einer Sicht	438
13.3	Datenmanipulationsanweisungen auf eine Sicht	438
13.3.1	DML-Anweisungen auf eine Sicht ohne WHERE-Klausel	439
13.3.2	DML-Anweisungen auf eine Sicht mit WHERE-Klausel	440
13.4	Systemansichten des SQL Servers	444
13.4.1	Informationsschemasichten	444
13.4.2	Systemansichten im Schema sys	445

14 Programmierung von gespeicherten Prozeduren 447

14.1	Überblick über die Verwendung von gespeicherten Prozeduren in SQL Server	447
14.2	Erstellung und Aufruf von gespeicherten Prozeduren	451
14.2.1	Grundlegende Syntax zu Erstellung einer gespeicherten Prozedur	451
14.2.2	Verwendung von Eingabeparametern	452
14.2.3	Die verschiedenen Arten der Parameterübergabe	456
14.2.4	Verweisbezogene Übergabe	458
14.2.5	Verwendung von Ausgabeparametern	460
14.2.6	Verwendung eines Rückgabestatuscodes	464
14.3	Tabellenübergabe an eine gespeicherte Prozedur	468
14.4	Optionen von gespeicherten Prozeduren	471
14.4.1	ENCRYPTION	471
14.4.2	RECOMPILE	473
14.4.3	EXECUTE AS	475

15 Programmierung von benutzerdefinierten Funktionen 477

15.1	Überblick über benutzerdefinierte Funktionen	477
15.2	Skalare Funktionen (Skalarwertfunktionen)	478
15.3	Inlinefunktionen (Tabellenwertfunktionen)	482
15.4	Tabellenwertfunktion mit mehreren Anweisungen	485
15.5	Ändern und Löschen von benutzerdefinierten Funktionen	488
15.5.1	Ändern einer benutzerdefinierten Funktion	488
15.5.2	Löschen einer benutzerdefinierten Funktion	488
15.6	Verwendung des Datentyps TABLE als lokale Variable	488
15.6.1	Verwendung von Einschränkungen	490

16 Programmierung und Einsatz von Triggern 493

16.1	DML-Trigger	493
16.1.1	Programmierung von DML-Triggern	495
16.2	DDL-Trigger	514
16.2.1	DDL-Trigger mit Serverbereich	514
16.2.2	DDL-Trigger mit Datenbankbereich	517
16.2.3	Informationen zur Auslösung von DDL-Triggern	519

17 Ereignisbenachrichtigungen (Event Notifications) 521

17.1	Aktivierung des Service Brokers	522
17.2	Konfiguration des Zieldienstes	523
17.2.1	Einrichtung der Warteschlange	523
17.2.2	Erstellung des Dienstes	523
17.2.3	Einrichtung der Route	524
17.3	Erstellen einer Ereignisbenachrichtigung	525
17.4	Auslösen und Empfangen einer Ereignisbenachrichtigung	526
17.5	Löschen der erstellten Objekte	528

18 Erstellung und Einsatz eines Cursors 529

18.1	Funktionsweise eines Cursors	529
18.2	Erstellung eines Cursors	531
18.3	Öffnen eines Cursors	533
18.4	Das Abrufen von Datensätzen aus einem Cursor	534
18.4.1	Aufruf einer FETCH-Anweisung ohne Verwendung von Variablen	534
18.4.2	Aufruf einer FETCH-Anweisung unter Verwendung von Variablen	535
18.4.3	Die Verwendung von Positionsangaben zum Abruf von Daten aus einem Cursor	535
18.5	Schließen und Löschen eines Cursors	543
18.6	Schleifenprogrammierung zum automatischen Durchlaufen eines Cursors	544
18.7	Daten in der Cursorauswahl aktualisieren und löschen	549
18.7.1	Aktualisieren	549
18.7.2	Löschen	549

19 Indizes 551

19.1	Der nicht gruppierte Index auf einem Heap	552
19.1.1	Vorgehensweise beim Durchlaufen eines Index	555
19.2	Der gruppierte Index	556
19.3	Der nicht gruppierte Index auf einem gruppierten Index	559
19.4	Erstellung von Indizes	562
19.4.1	Manuelle Erstellung eines Index	562
19.4.2	Automatische Erstellung von Indizes	565
19.5	Verwaltung von Indizes	566
19.5.1	Fragmentierung	566

19.5.2	Neuerstellung von Indizes	569
19.5.3	Löschen eines Index	569
19.6	Statistiken	570
19.6.1	Löschen einer Statistik	573
19.6.2	Aktualisieren einer Statistik	574
19.6.3	Informationen zu Statistiken abrufen	574
19.7	Planung des Einsatzes von Indizes	575
19.7.1	Verwenden des Datenbankmodul- Optimierungsratgebers	577
19.8	Weitere Optimierungsmöglichkeiten mit Hilfe von Indizes	584
19.8.1	Indizes mit eingeschlossenen Spalten	584
19.8.2	Gefilterte Indizes	588
19.8.3	Indizierte Sichten	589
19.8.4	Partitionierte Indizes	591

20 Transaktionen 593

20.1	Einführung in Transaktionen	593
20.2	ACID	596
20.3	Interne Transaktionsverarbeitung	598
20.4	Verhalten bei Systemfehlern	600
20.5	Programmierung expliziter Transaktionen	602
20.6	Implizite Transaktionen	605
20.7	Wiederherstellung beim Systemstart (Recovery)	607
20.8	Sperren	610
20.8.1	Sperrebenen	610
20.8.2	Gemeinsame Sperren (Shared Locks)	611
20.8.3	Exklusive Sperren (Exclusive Locks)	611
20.9	Isolationsstufen auf Verbindungsebene	612
20.9.1	READ UNCOMMITTED	613
20.9.2	READ COMMITTED	615
20.9.3	REPEATABLE READ	617
20.9.4	SERIALIZABLE	619
20.9.5	Zusammenfassung der Eigenschaften von Sperrstufen ...	620
20.9.6	Setzen eines Timeout-Wertes	621
20.10	Sperrhinweise auf Tabellenebene	622
20.11	Informationen zu Sperren	623
20.12	Deadlocks	624
20.12.1	Vermeidung von Deadlocks	625
20.12.2	Beispiel zur Erzeugung eines Deadlocks	626

21 Der SQL Server als Laufzeitumgebung für .NET 629

21.1	Die Programmiermodelle im Vergleich	629
21.1.1	T-SQL	629
21.1.2	Erweiterte gespeicherte Prozeduren	630
21.1.3	Einbindung von COM-Objekten	631
21.1.4	Einsatz von Managed Code	631
21.2	Die CLR-Integration im Detail	633
21.2.1	CLR-Hosting	634
21.2.2	Thread- und Speichermanagement	635
21.2.3	I/O-Zugriffe	635
21.2.4	Assemblies Loading	636
21.2.5	Application Domain Management	636
21.3	Der Hosting Layer	637
21.4	Verwaltung von Assemblies	640
21.5	Sicherheitsstufen	641

22 .NET-Programmierung 643

22.1	CLR-Unterstützung aktivieren	643
22.1.1	CLR-Unterstützung per Skript aktivieren	645
22.2	CLR-Prozeduren und Funktionen	645
22.2.1	CLR vs. T-SQL	645
22.2.2	Prozeduren vs. Funktionen	645
22.2.3	Veröffentlichung von Methoden	646
22.2.4	Implementierungsregeln	647
22.2.5	Parameter und Rückgabewerte	648
22.2.6	Sicherheitsaspekte	649
22.3	Gespeicherte Prozeduren	650
22.3.1	Eine gespeicherte Prozedur mit Visual Studio erstellen	650
22.3.2	Das SqlParameter-Attribut	654
22.3.3	Parameter und Rückgabe	654
22.3.4	Temporäre Prozeduren	655
22.3.5	Assembly bereitstellen	655
22.3.6	Assembly ohne Visual Studio kompilieren	656
22.3.7	Installation mit sqlcmd durchführen	657
22.3.8	Prozeduren mit Visual Studio debuggen	657
22.4	Benutzerdefinierte Funktionen	658
22.4.1	Einschränkungen	658
22.4.2	Das SqlParameter-Attribut	659

22.4.3	Funktionen bereitstellen	660
22.4.4	Skalare Funktionen	661
22.4.5	Eine Skalarfunktion erstellen	661
22.4.6	Tabellenwertige Funktionen	662
22.4.7	Unterschiede zwischen T-SQL- und CLR-Funktionen	662
22.4.8	Eine tabellenwertige Funktion erstellen	663
22.5	Serverseitiger Datenzugriff mit ADO.NET	666
22.5.1	Der Kontext	667
22.5.2	Verbindung zur Datenbank herstellen	668
22.6	Pipes	668
22.6.1	Die Send-Methode	669
22.6.2	Die ExecuteAndSend()-Methode	672
22.6.3	Ausgabestrom steuern	672
22.7	Impersonalisierung	676
22.8	Benutzerdefinierte Typen	680
22.8.1	Das SqlUserDefinedType-Attribut	681
22.8.2	Einen benutzerdefinierten Datentyp erstellen	682
22.8.3	Das INullable-Interface	683
22.8.4	Die ToString()-Methode	684
22.8.5	Die Parse()-Methode	684
22.8.6	Erstellung des CalendarWeek-Datentyps	684
22.8.7	Den CalendarWeek-Typ installieren	687
22.8.8	Den CalendarWeek-Typ testen	687
22.8.9	Validierungen	688
22.8.10	Methoden implementieren	689
22.8.11	Serialisierung	691
22.8.12	Typen als Klassen abbilden	695
22.8.13	Zugriff vom Client	696
22.9	Benutzerdefinierte Aggregate	697
22.9.1	Das SqlUserDefinedAggregate-Attribut	697
22.9.2	Die Init-Methode	698
22.9.3	Die Accumulate-Methode	699
22.9.4	Die Merge-Methode	699
22.9.5	Die Terminate-Methode	699
22.9.6	Ein Beispielaggregat	699
22.9.7	Das MaxRange-Aggregat implementieren	701
22.9.8	Hinweise zur Implementierung	704
22.9.9	Aggregate installieren	704
22.10	Benutzerdefinierte Trigger	705
22.10.1	Das SqlTrigger-Attribut	706

22.10.2	Die TriggerContext-Klasse	706
22.10.3	Trigger implementieren	708
22.10.4	Hinweise zur Implementierung	711
22.10.5	Trigger installieren	711
22.11	Administration und Monitoring von CLR-Objekten	712
22.11.1	Informationen über die installierten Assemblies ermitteln	712
22.11.2	Laufzeitinformationen von CLR-Objekten ermitteln	713

23 T-SQL: Neuerungen und Verbesserungen der Kompatibilität 717

23.1	CROSS APPLY und OUTER APPLY	719
23.2	Kreuztabellen mit PIVOT erstellen	721
23.2.1	Entscheidungstabellen mit PIVOT erstellen	726
23.2.2	Kreuztabellen in flache Listen transformieren	727
23.3	Common Table Expressions	729
23.3.1	Verarbeitung von hierarchischen Daten	733
23.3.2	Maximale Rekursionsstufe einstellen	736
23.4	Fehlerbehandlung	737
23.4.1	Informationen über den Fehler ermitteln	738
23.4.2	Eigene Fehler auslösen	742
23.4.3	Verschachtelte Fehlerbehandlung	743
23.5	Ranking- und Windowing-Funktionen	744
23.5.1	Ergebnisse mit RANK() gewichten	744
23.5.2	Ranking ohne Lücken mit DENSE_RANK()	745
23.5.3	Gruppierte Ranglisten mit Windowing	746
23.5.4	Zeilen mit ROW_NUMBER() nummerieren	747
23.5.5	Paging mit ROW_NUMBER()	748
23.5.6	Daten mit NTILE() partitionieren	750
23.6	Die OUTPUT-Klausel	751
23.7	Die TABLESAMPLE-Klausel	753
23.8	EXCEPT- und INTERSECT-Statement	755
23.9	Tabellenwertparameter	756
23.9.1	Tabellenwertparameter definieren	757
23.9.2	Tabellenwertparameter verwenden	757
23.9.3	Prozeduren mit Tabellenwertparametern aufrufen	758
23.9.4	Aufruf vom Client mit ADO.NET	758
23.9.5	Limitationen	760

24 Einsatz von XML in der Datenbank 761

24.1	Warum XML?	762
24.2	HTML und XML	762
24.2.1	Auszeichnungssprachen	763
24.2.2	Auszeichnungen	764
24.3	XML-Fähigkeiten des SQL Servers	767
24.4	Der XML-Datentyp	767
24.4.1	Vor- und Nachteile der XML-Datenspeicherung	768
24.4.2	XML-Felder anlegen	770
24.4.3	XML-Felder mit einem Schema verknüpfen	772
24.5	Auf gespeicherte XML-Daten zugreifen	774
24.5.1	Eine kurze Einführung in XPath	774
24.5.2	XQuery als Abfragesprache	776
24.5.3	Die Methoden des XML-Datentyps	777
24.5.4	Indizierung von XML-Feldern	782
24.6	Darstellung von Abfrageergebnissen im XML-Format	785
24.6.1	Der RAW-Modus	787
24.6.2	Allgemeine Optionen der FOR XML-Klausel	790
24.6.3	Der AUTO-Modus	792
24.6.4	Der PATH-Modus	794
24.6.5	Der EXPLICIT-Modus	796
24.7	XML-Schema ermitteln	802
24.8	XML-Daten in einem relationalen Modell speichern	804
24.8.1	Ein XML-Dokument in eine Tabelle importieren	805

25 Der Einsatz von Webservices in der Datenbank 807

25.1	Warum Webservices?	808
25.2	Den SQL Server als Webservice-Backend betreiben	809
25.3	Endpoints	811
25.4	Einen Endpoint erstellen	811
25.4.1	Die Status eines Endpoints	812
25.4.2	Adresse des Endpoints bestimmen	813
25.4.3	Ports konfigurieren	813
25.4.4	Methode definieren	814
25.4.5	Sicherheitskonfiguration	815
25.4.6	Authentifizierungstyp festlegen	815
25.4.7	Authentifizierungsverfahren festlegen	816
25.4.8	Berechtigungsverwaltung	819

25.4.9	HTTP-Kompression aktivieren	819
25.5	WSDL ermitteln	819
25.5.1	In die WSDL-Generierung eingreifen	820
25.5.2	WSDL vom Client abrufen	820
25.6	Zugriff vom Client	821
25.7	Session- und State-Management	824
25.7.1	Sessions aktivieren	825
25.7.2	Session-Steuerung durch den Client	825
25.8	Batch-Abfragen	827
25.9	Endpoints verwalten	829

26 Datenbankzugriff mit ADO.NET 831

26.1	Einleitung	831
26.1.1	Installation von Visual C# 2008 Express Edition	833
26.1.2	Projekte anlegen und speichern	834
26.1.3	Die Beispieldatenbank	835
26.2	Die Datenprovider	835
26.3	Die Verbindung zu einer Datenbank herstellen	837
26.3.1	Das Connection-Objekt	837
26.3.2	Die Verbindungszeichenfolge	838
26.3.3	Die Verbindung mit einer bestimmten SQL-Server-Instanz aufbauen	839
26.3.4	Öffnen und Schließen einer Verbindung	843
26.3.5	Das Verbindungspooling	848
26.3.6	Die Ereignisse eines Connection-Objekts	852
26.3.7	Verbindungszeichenfolgen aus einer Konfigurationsdatei abrufen	854
26.3.8	Die Klasse SqlConnection im Überblick	856
26.3.9	Verbindungen mit dem OleDb-Datenprovider	858
26.4	Die Datenbankabfrage	861
26.4.1	Das SqlCommand-Objekt	861
26.4.2	Abfragen, die genau ein Ergebnis liefern	866
26.5	Das SqlDataReader-Objekt	867
26.5.1	Datensätze einlesen	868
26.5.2	Schließen des SqlDataReader-Objekts	871
26.5.3	MARS (Multiple Active Resultsets)	871
26.5.4	Batch-Abfragen mit NextResult durchlaufen	873
26.5.5	Das Schema eines SqlDataReader-Objekts untersuchen ...	874

26.6	Parametrisierte Abfragen	877
26.6.1	Parametrisierte Abfragen mit dem SqlConnection-Datenprovider	877
26.6.2	Die Klasse SqlParameter	880
26.6.3	Asynchrone Abfragen	880
26.6.4	Gespeicherte Prozeduren (Stored Procedures)	884
26.7	Der SqlDataAdapter	892
26.7.1	Was ist ein DataAdapter?	892
26.7.2	Die Konstruktoren der Klasse DataAdapter	894
26.7.3	Die Eigenschaft SelectCommand	895
26.7.4	Den lokalen Datenspeicher mit Fill füllen	895
26.7.5	Tabellenzuordnung mit TableMappings	899
26.7.6	Das Ereignis FillError des SqlDataAdapter's	904
26.8	Daten im lokalen Speicher – das DataSet	905
26.8.1	Verwenden des DataSet-Objekts	906
26.8.2	Dateninformationen in eine XML-Datei schreiben	910
26.8.3	Dem DataSet Schemainformationen übergeben	911
26.8.4	Schemainformationen bereitstellen	913
26.8.5	Eigenschaften einer DataColumn, die der Gültigkeitsprüfung dienen	914
26.8.6	Die Constraints-Klassen einer DataTable	914
26.8.7	Das Schema mit Programmcode erzeugen	915
26.8.8	Schemainformationen mit SqlDataAdapter abrufen	917
26.8.9	Änderungen in einer DataTable vornehmen	921
26.8.10	Was bei einer Änderung einer Datenzeile passiert	926
26.8.11	Manuelles Steuern der Eigenschaft DataRowState	930
26.9	Mit mehreren Tabellen arbeiten	931
26.9.1	Der Weg über JOIN-Abfragen	931
26.9.2	Mehrere Tabellen in einem DataSet	933
26.9.3	Eine DataRelation erzeugen	934
26.9.4	DataRelations und Einschränkungen	935
26.9.5	In Beziehung stehende Daten suchen	937
26.9.6	Ergänzung zum Speichern von Schemainformationen in einer XML-Schemadatei	940
26.10	Aktualisieren der Datenbank	940
26.10.1	Aktualisieren mit dem CommandBuilder-Objekt	940
26.10.2	Manuell gesteuerte Aktualisierungen	943
26.10.3	Aktualisieren mit ExecuteNonQuery	945
26.10.4	Manuelles Aktualisieren mit dem DataAdapter	953
26.10.5	Den zu aktualisierenden Datensatz in der Datenbank suchen	956

26.10.6	Den Benutzer über fehlgeschlagene Aktualisierungen informieren	961
26.10.7	Konfliktverursachende Datenzeilen bei der Datenbank abfragen	964
26.11	Objekte vom Typ DataView	970
26.11.1	Eine DataView erzeugen	971
26.11.2	Auf die Datenzeilen in einer DataView zugreifen	972
26.11.3	Die Eigenschaft Sort und die Methode Find	972
26.11.4	Die Methode FindRows	972
26.11.5	Die Eigenschaft RowFilter	973
26.11.6	Die Eigenschaft RowStateFilter	973
26.11.7	Änderungen an einem DataView-Objekt	974
26.11.8	Aus einer DataView eine DataTable erzeugen	975
26.12	Stark typisierte DataSets	976
26.12.1	Ein stark typisiertes DataSet erzeugen	977
26.12.2	Die Anatomie eines typisierten DataSets	981
26.12.3	Typisierte DataSets manuell im Designer erzeugen	988
26.12.4	Weitergehende Betrachtungen	990
26.12.5	Der TableAdapter	990
26.12.6	Einen TableAdapter mit Visual Studio erzeugen	991
26.12.7	Festlegen der Methoden des TableAdapters	996
26.12.8	TableAdapter im Code verwenden	996
26.13	Fazit: Typisierte oder nicht typisierte DataSets?	1003
26.14	Projekte mit grafischer Benutzeroberfläche	1004
26.14.1	Eine Verbindung mit einem Steuerelement herstellen ...	1004
26.14.2	SqlCommand-Objekte mit dem Designer erzeugen	1005
26.14.3	SqlDataAdapter mit dem Designer erzeugen	1007
26.14.4	Datenausgabe in WinForms mit Visual Studio 2008	1010
26.15	Datenbindung	1013
26.15.1	Binden an Datenquellen (Simple Binding)	1014
26.15.2	Komplexe Datenbindung (Complex Binding)	1024

27 LINQ 1029

27.1	Was ist LINQ?	1029
27.2	Neue Sprachfeatures in C# 3.0	1031
27.2.1	Implizit typisierte Variablen (Typinferenz)	1031
27.2.2	Lambda-Ausdrücke	1032
27.2.3	Erweiterungsmethoden	1039
27.2.4	Objektinitialisierung	1042

27.2.5	Anonyme Typen	1042
27.2.6	C# 3.0 und LINQ-Abfragen	1043
27.3	LINQ to Objects	1046
27.3.1	Musterdaten	1046
27.3.2	Die Abfrage-Syntax	1049
27.3.3	Übersicht über die Abfrageoperatoren	1050
27.3.4	Die from-Klausel	1051
27.3.5	Der Restriktionsoperator where	1052
27.3.6	Die Projektionsoperatoren	1055
27.3.7	Sortieroperatoren	1056
27.3.8	Gruppieren mit GroupBy	1058
27.3.9	Verknüpfungen mit Join	1059
27.3.10	Die Set-Operatoren-Familie	1062
27.3.11	Die Familie der Aggregatoperatoren	1064
27.3.12	Generierungsoperatoren	1067
27.3.13	Quantifizierungsoperatoren	1069
27.3.14	Aufteilungsoperatoren	1069
27.3.15	Die Elementoperatoren	1072
27.4	LINQ to SQL	1075
27.4.1	Allgemeine Grundlagen	1075
27.4.2	Objektzuordnung mit Entitätsklassen	1076
27.4.3	Das Mapping von Objekten	1078
27.4.4	Verknüpfungen zwischen Entitäten	1079
27.4.5	Tools zur Erzeugung von Entitätsklassen	1086
27.4.6	Die Klasse DataContext	1087
27.4.7	Konflikte behandeln	1093
27.5	Der LINQ-to-SQL-Designer (O/R-Designer)	1100
27.5.1	Handhabung des O/R-Designers	1100
27.5.2	Die abgeleitete DataContext-Klasse	1103
27.5.3	Die Entitätsklassen	1106
27.5.4	WinForm-Beispielprogramm mit einer Tabelle	1108
27.5.5	WinForm-Beispiel mit mehreren Tabellen	1115
Index		1119