

Inhalt

1	Vorwort und Einleitung	1
2	Neuerungen in SQL Server, Visual Studio und C#.....	5
2.1	SQL Server 2008.....	5
2.1.1	Neue Datentypen	5
2.1.2	Vereinfachte Initialisierung	6
2.1.3	Vereinfachte Berechnung	7
2.1.4	FileStream-Storage.....	7
2.1.5	CDC (Change Data Capture).....	7
2.1.6	Change Tracking	8
2.1.7	Auditing.....	10
2.1.8	Die Merge-Anweisung	10
2.1.9	Mehrere Zeilen mit einer INSERT-Anweisung einfügen	11
2.1.10	Grouping Sets.....	11
2.1.11	Tabellen als Parameter (Table-Valued Parameters – TVP)	12
2.1.12	Sparse-Spalten.....	13
2.1.13	Transparente Verschlüsselung der gesamten Datenbank.....	14
2.1.14	IntelliSense	14
2.1.15	Was sonst noch neu ist	15
2.2	Neuerungen des MS SQL Server 2008 R2.....	15
2.2.1	Datenbank.....	16
2.2.2	SQL Server Reporting Services.....	19
2.3	Visual Studio 2010.....	21
2.4	Neuheiten in C# 4.0.....	22
2.4.1	Optionale und benannte Parameter	22
2.4.2	Dynamic	24
2.5	Einige Neuheiten für .NET Framework 4.0	25
2.5.1	DLR (Dynamic Language Runtime).....	26
2.5.2	Numerische Werte	26
2.5.3	Datenstrukturen	27
2.5.4	Parallel Computing.....	29

2.5.5	Verzögerte Initialisierung (Lazy Initialization)	29
2.5.6	Code Contracts	30
2.5.7	Side-by-Side-Ausführung	31
2.5.8	Garbage Collection	31
3	Microsoft SQL Server 2008 (R2)	33
3.1	Woraus besteht der SQL Server eigentlich?	33
3.1.1	Instanzen	34
3.1.2	Datenbankmodul	36
3.1.3	SQL Server Agent	37
3.1.4	Die Systemdatenbanken	40
3.1.5	Volltextsuche	41
3.1.6	Analysis Services	42
3.1.7	Reporting Services	42
3.1.8	Integration Services	42
3.2	Verwaltungsprogramme	42
3.2.1	SQL Server Management Studio	42
3.2.2	SQL Server Profiler	44
3.2.3	Datenbankoptimierungsratgeber (Database Tuning Advisor)	45
3.2.4	SQL Server-Konfigurationsmanager	46
3.3	Die Installation	46
3.3.1	Vorbereitung	46
3.3.2	Installation starten	47
3.3.3	Während der Installation	48
3.4	Die Dienste hinter den Kulissen	50
3.5	Sicherheit	51
3.5.1	Ein wenig Begriffsklärung	51
3.5.2	Zwei Arten der Authentifizierung	52
3.5.3	Vordefinierte Anmeldungen	54
3.5.4	Vordefinierte Datenbankbenutzer	55
3.5.5	Berechtigungen vergeben	55
3.5.6	Serverrollen	56
3.5.7	Datenbankrollen	56
3.5.8	Besitzer	57
3.5.9	Benutzer anlegen	57
3.5.10	Datenbankschema	58
3.5.11	Schemasicherheit	59
3.6	Mit Datenbanken arbeiten	61
3.6.1	Grundlegendes	62
3.6.2	Datenbank erstellen	64
3.6.3	Sichern/Wiederherstellen	66
3.6.4	Löschen	68
3.6.5	Tabellen & Co	69
3.7	CDC (Change Data Capture)	71
3.7.1	Aktivierung	73
3.7.2	DML-Bruttoänderungen	76

3.7.3	DML-Nettoänderungen	78
3.7.4	DDL-Änderungen.....	79
3.7.5	Deaktivierung	79
3.8	Change Tracking.....	80
3.8.1	Aktivierung.....	80
3.8.2	Abfragen.....	83
3.8.3	Änderungskontext	84
3.8.4	Deaktivierung	86
3.9	Auditing	86
3.9.1	Ein Überwachungsobjekt erstellen	87
3.9.2	Eine Überwachungsspezifikation erstellen.....	88
3.9.3	Auswertung	90
3.10	Volltextsuche	91
3.10.1	Neuerung ab SQL Server 2008.....	92
3.10.2	Suchkatalog und Indizes einrichten	93
3.10.3	Stopplisten.....	95
3.10.4	Thesaurus	96
3.10.5	FTS-Abfragen.....	97
3.11	FileStream.....	99
3.11.1	Wann FileStream einsetzen?	99
3.11.2	Aktivierung auf der SQL Server-Instanz	100
3.11.3	Vorbereitung der Datenbank	103
3.11.4	Vorbereitung der Tabelle.....	104
3.11.5	Ablage der Daten im NTFS-Dateisystem	105
3.11.6	Zugriff per ADO.NET	106
3.11.7	Zugriff per FileStream API	109
3.12	HierarchyID	114
3.13	Geometry & Geography	121
3.14	Table-Valued Parameters	128
3.15	Tipps und Tricks	132
3.15.1	Benutzen der erweiterten Eigenschaften der Datenbank	132
3.15.2	Sehen, welche T-SQL-Anweisungen das Management Studio (Express) verwendet	134
3.15.3	Ein T-SQL-Skript für ein Objekt oder eine Aktion erstellen	134
3.15.4	Eine Datenbank als Skript ausliefern.....	136
4	T-SQL	139
4.1	SQL-Varianten.....	139
4.2	Anweisungen mit einem Semikolon abschließen?	140
4.3	T-SQL-Anweisungen entwickeln und testen.....	142
4.3.1	Syntaktische oder deklarative Fehler.....	143
4.3.2	Logische Fehler im Ablauf.....	145
4.3.3	Visual Studio 2010	146
4.4	Allgemeine Grundlagen	147
4.4.1	Kommentare	147
4.4.2	Operatoren.....	148
4.4.3	Groß- und Kleinschreibung	149

4.4.4	Literale	150
4.4.5	Zeichenketten	151
4.4.6	Unicode-Zeichenketten	152
4.4.7	Zahlen mit Komma.....	152
4.4.8	Datum und Uhrzeit.....	152
4.4.9	Ungültige Zeichen und reservierte Schlüsselwörter	154
4.4.10	Zeilenumbrüche und Einschübe mit Tab oder Leerzeichen.....	155
4.4.11	Ausführungsblöcke (Batches)	156
4.4.12	Ausdrücke	156
4.4.13	Lexikografische Ordnung.....	157
4.4.14	Tabellen oder Sichten.....	158
4.4.15	Gruppen von T-SQL-Anweisungen	158
4.4.16	Alles unsicher?	159
4.5	DML	159
4.5.1	Daten abfragen	160
4.5.2	Daten einfügen	188
4.5.3	Daten löschen	192
4.5.4	Daten manipulieren	194
4.5.5	Die Merge-Anweisung	195
4.5.6	Allgemeine Tabellenausdrücke (CTE) für rekursive Abfragen.....	198
4.6	Data Definition Language (DDL)	201
4.6.1	DDL-Anweisungen automatisch erstellen lassen	201
4.6.2	Datenbanken.....	204
4.6.3	Tabellen.....	207
4.6.4	Trigger.....	228
4.6.5	Sichten (Views).....	234
4.6.6	Gespeicherte Prozeduren.....	237
4.6.7	Benutzerdefinierte Funktionen	241
4.6.8	Synonyme.....	243
4.6.9	Benutzerdefinierte Datentypen.....	245
4.6.10	Datenbankschemata.....	247
4.7	Indizes.....	249
4.8	Transaktionen.....	252
4.9	Tipps und Tricks	253
4.9.1	Ermitteln der eingefügten Identitätswerte	253
4.9.2	Werte in die Identitätsspalte einfügen	253
4.9.3	Aktuellen Wert der Identitätsspalte auslesen und festlegen.....	254
4.9.4	In das Anwendungsprotokoll des Systems schreiben	256
4.9.5	Bei der Ausführung eine Pause einlegen.....	256
4.9.6	Einschränkungen deaktivieren.....	256
5	SQL Server 2008 (R2) mit ADO.NET	259
5.1	Übersicht über die Klassen	259
5.2	Die SqlConnection-Klasse	261
5.2.1	Wichtige Methoden und Eigenschaften.....	262
5.2.2	Zustand der Verbindung.....	264

5.2.3	Ereignisse	264
5.2.4	Verbindungs-Pooling (Connection Pooling).....	266
5.2.5	Verbindungszeichenfolge (Connection String).....	267
5.2.6	Statistische Werte abrufen	274
5.3	Die SqlConnectionStringBuilder-Klasse.....	276
5.3.1	Wichtige Methoden und Eigenschaften.....	277
5.3.2	Praktischer Einsatz	278
5.4	Die SqlCommand-Klasse	280
5.4.1	Wichtige Methoden und Eigenschaften.....	281
5.4.2	ExecuteNonQuery-Methode.....	283
5.4.3	ExecuteScalar-Methode.....	284
5.4.4	ExecuteReader-Methode	285
5.4.5	ExecuteXmlReader-Methode	286
5.4.6	SqlCommand in Verbindung mit SqlDataAdapter	287
5.4.7	Parameter verwenden	288
5.4.8	Ereignisse	289
5.4.9	Praktischer Einsatz	290
5.5	Die SqlParameter-Klasse.....	296
5.5.1	Wichtige Methoden und Eigenschaften.....	297
5.5.2	Übergaberichtung des Parameters	298
5.5.3	Praktischer Einsatz	298
5.5.4	Parameter für mehrere Abfragen wieder verwenden	299
5.6	Die SqlDependency-Klasse.....	302
5.6.1	Wichtige Methoden und Eigenschaften.....	304
5.6.2	Ereignisse	305
5.6.3	Praktischer Einsatz	308
5.7	Die SqlDataReader-Klasse.....	310
5.7.1	Wichtige Methoden und Eigenschaften.....	311
5.8	Die XmlReader-Klasse.....	315
5.8.1	Wichtige Methoden und Eigenschaften.....	316
5.9	Die SqlDataAdapter-Klasse	318
5.10	Die SqlCommandBuilder-Klasse	319
5.11	Die SqlClientPermission-Klasse	319
5.11.1	Wichtige Methoden und Eigenschaften.....	320
5.11.2	Praktischer Einsatz	321
5.12	Die SqlBulkCopy-Klasse	321
5.12.1	Wichtige Methoden und Eigenschaften.....	323
5.12.2	Optionen für das Kopieren	324
5.12.3	Zuordnung von Quell- und Zielspalten.....	325
5.12.4	Ereignisse	326
5.12.5	Praktischer Einsatz	327
5.13	Die SqlTransaction-Klasse	329
5.13.1	Wichtige Methoden und Eigenschaften.....	332
5.13.2	Isolationsgrad	333
5.13.3	Praktischer Einsatz	334

5.14	Die SQLException- und SqlError-Klasse.....	337
5.14.1	Wichtige Eigenschaften der SQLException-Klasse	338
5.14.2	Die SqlError-Klasse	338
5.14.3	Alle relevanten Informationen im Fehlerfall auswerten	339
5.15	MARS (Multiple Active Result Sets).....	340
5.16	Asynchrone Ausführung	340
5.16.1	Die IAsyncResult-Schnittstelle	340
5.16.2	BackgroundWorker-Komponente	342
5.16.3	Thread	343
5.17	Tipps und Tricks	345
5.17.1	Das Schema einer Tabelle abfragen	345
5.17.2	Das Kennwort über SqlConnection.ConnectionString erhalten.....	347
5.17.3	Das Kennwort bei SQL Server-Authentifizierung ändern.....	347
5.17.4	Feststellen, warum eine SQL Server-Anmeldung fehlschlägt.....	348
5.17.5	Verbindungs-Pool per Code leeren	348
6	CLR-Integration in SQL Server 2008 (R2)	351
6.1	Grundlagen	351
6.1.1	Was ist machbar, was nicht?	352
6.1.2	Wie wurde die CLR-Integration realisiert?	353
6.1.3	Wann T-SQL- und wann .NET-Objekte entwickeln?.....	355
6.1.4	Datentypen	356
6.1.5	Sicherheit	358
6.1.6	CLR-Integration aktivieren und deaktivieren	365
6.1.7	Assemblies verwalten im SQL Server Management Studio (Express).....	368
6.1.8	Assemblies mit T-SQL verwalten	371
6.2	Entwicklung & Debugging	375
6.2.1	Visual Studio als Entwicklungsumgebung	375
6.2.2	Erste Schritte bei der Entwicklung	377
6.2.3	Fehler und Ausnahmen.....	395
6.2.4	Gespeicherte Prozeduren.....	397
6.2.5	Trigger.....	404
6.2.6	Benutzerdefinierte Funktionen	416
6.2.7	Tabellenwertfunktion	422
6.2.8	Aggregate	429
6.2.9	Benutzerdefinierte Typen	434
6.3	Installation, Aktualisierung und Deinstallation.....	447
6.3.1	Installation.....	447
6.3.2	Aktualisieren	449
6.3.3	Deinstallation	450
6.4	Tipps und Tricks	451
6.4.1	Sicherheitskontexte bei Erstellung mit CREATE festlegen	451
6.4.2	Auf zugeordnete Dateien zugreifen	452
6.4.3	Auf Eigenschaften der Assembly zugreifen	452
6.4.4	LINQ to XML (oder andere Assemblies) verwenden.....	453

7	DataSets	459
7.1	Übersicht über die Klassen.....	459
7.2	Die DataSet-Klasse	461
7.2.1	Wichtige Methoden und Eigenschaften.....	461
7.2.2	Erweiterte Eigenschaften.....	464
7.2.3	Ereignisse	465
7.2.4	DataSet befüllen	466
7.2.5	Ein DataSet komplett durchlaufen und ausgeben	478
7.2.6	Datenänderungen abfragen, annehmen und verwerfen.....	479
7.2.7	Das DataSet kopieren	483
7.3	Die DataTable-Klasse	483
7.3.1	Wichtige Methoden und Eigenschaften.....	485
7.3.2	Ereignisse	488
7.3.3	Spalten hinzufügen, löschen und auflisten.....	488
7.3.4	Zeilen hinzufügen/löschen.....	490
7.3.5	Mit einem Primärschlüssel arbeiten.....	493
7.3.6	Zeilen per Abfrage suchen.....	493
7.3.7	Werte lesen und verändern	495
7.3.8	Beziehungen zwischen DataTable-Objekten	496
7.3.9	Änderungen abfragen	497
7.3.10	Per XML-Serialisierung lesen und schreiben	497
7.3.11	DataTable-Objekte kopieren.....	497
7.3.12	Werte berechnen.....	498
7.4	Die DataRow-Klasse	499
7.4.1	Wichtige Methoden und Eigenschaften.....	499
7.4.2	Ereignisse	500
7.4.3	Zeilenstatus und Zeilenversionen	500
7.5	Die DataColumn-Klasse.....	503
7.5.1	Wichtige Methoden und Eigenschaften.....	503
7.5.2	Ereignisse	504
7.5.3	Berechnete Spalten	504
7.5.4	Per Code Spalten hinzufügen/entfernen	509
7.5.5	Die Reihenfolge der Spalten einer Tabelle verändern	509
7.6	Die DataRelation-Klasse	509
7.6.1	Wichtige Methoden und Eigenschaften.....	509
7.6.2	Ereignisse	510
7.7	Die DataView-Klasse.....	510
7.7.1	Wichtige Methoden und Eigenschaften.....	512
7.7.2	Ereignisse	512
7.7.3	Filter und Sortierung.....	514
7.7.4	Zeilenstatus in einem DataView	515
7.7.5	Zeilen in einem DataView finden	516
7.7.6	Ein DataView in eine Tabelle exportieren.....	517
7.8	Die DataRowView-Klasse	518
7.8.1	Wichtige Methoden und Eigenschaften.....	518
7.8.2	Ereignisse	518

7.8.3	Neue Zeilen hinzufügen	519
7.9	Die DataTableReader-Klasse.....	520
7.10	Eigene Fehler per Code signalisieren.....	521
7.11	Die SqlDataAdapter-Klasse	523
7.11.1	Wichtige Methoden und Eigenschaften.....	524
7.11.2	Ereignisse.....	525
7.11.3	TableMapping	527
7.12	Die SqlCommandBuilder-Klasse	528
7.12.1	Wichtige Methoden und Eigenschaften.....	528
7.12.2	Ereignisse.....	529
7.13	Typisierte DataSets	529
7.14	Tipps und Tricks	529
7.14.1	Nur die Änderungen eines DataSets schreiben.....	529
7.14.2	Inhalt und Schema getrennt schreiben/lesen.....	530
7.14.3	Ein DataSet komprimieren/dekomprimieren.....	531
8	Language Integrated Query	535
8.1	LINQ-Abfragen	538
8.1.1	Klassen/Tabellen für die Beispiele.....	538
8.1.2	LINQ-Operatoren	540
8.1.3	Sequenzen	544
8.1.4	Verzögerte Ausführung	544
8.1.5	Projektionsoperatoren.....	545
8.1.6	Filteroperatoren	546
8.1.7	Sortieroperatoren.....	548
8.1.8	Gruppierungsoperatoren.....	549
8.1.9	Join-Operatoren.....	551
8.1.10	Set-Operatoren	554
8.1.11	Aggregatoperatoren.....	557
8.1.12	Generierungsoperatoren	561
8.1.13	Quantifizierungsoperatoren	562
8.1.14	Aufteilungsoperatoren.....	564
8.1.15	Elementoperatoren	566
8.1.16	Konvertierungsoperatoren	569
8.1.17	Sonstige Operatoren	572
8.2	LINQ to SQL	574
8.2.1	Der Datenkontext	575
8.2.2	Entitäten-Klassen	578
8.2.3	Abfragen.....	592
8.2.4	Abfragen protokollieren	604
8.2.5	Datenbindung	604
8.3	LINQ to DataSets.....	604
8.4	Tipps und Tricks	608
8.4.1	Bei LINQ to SQL statt null einen Fallback-Wert erhalten	608

9	Datenbindung.....	609
9.1	Klassen/Tabellen für die Beispiele.....	610
9.2	Windows Forms	610
9.2.1	Steuerelemente für die Entwurfszeit.....	610
9.2.2	Einfache Datenbindung	611
9.2.3	Das DataGridView-Steuerelement	627
9.2.4	Bearbeiten der Zelleninhalte.....	650
9.3	ASP.NET	654
9.3.1	Datenquellen.....	655
9.3.2	Visuelle Steuerelemente	663
10	ADO.NET Entity Framework 4.0.....	603
10.1	Wozu dient ein O/R-Mapper?	604
10.2	ADO.NET Entity Framework im Projekt.....	605
10.2.1	Entitätenmodell (EDM)	605
10.2.2	Entitäten	613
10.2.3	Abgeleitete Entitäten	615
10.2.4	Entitäten mit eigenen Methoden und Eigenschaften erweitern.....	616
10.2.5	Komplexe Typen	617
10.2.6	Beziehungen (Assoziationen)	617
10.2.7	Gespeicherte Prozeduren und Funktionen	619
10.3	Daten abfragen mit LINQ	620
10.3.1	Lazy-Loading	621
10.4	Daten abfragen mit Entity SQL	622
10.4.1	Unterschiede zu T-SQL.....	622
10.4.2	Einführung in Entity SQL.....	624
10.4.3	Die CreateQuery()-Methode des Objektkontexts	634
10.4.4	System.Data.EntityClient	635
10.5	Daten verändern	639
10.5.1	Werte von Eigenschaften ändern	639
10.5.2	Entitäten hinzufügen/löschen	639
10.6	Änderungen speichern.....	641
10.6.1	Änderungskonflikte	641
10.6.2	Transaktionen	642
10.6.3	Self Tracking Entities	644
10.7	Code-Only	645
Register.....	249	