Pro SQL Server 2005

Thomas Rizzo, Adam Machanic,
Julian Skinner, Louis Davidson,
Robin Dewson, Jan Narkiewicz,
Joseph Sack, Rob Walters

Apress®

Pro SQL Server 2005

Copyright © 2006 by Thomas Rizzo, Adam Machanic, Julian Skinner, Louis Davidson, Robin Dewson, Jan Narkiewicz, Joseph Sack, Rob Walters

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN (pbk): 1-59059-477-0

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Lead Editor: Tony Davis
Technical Reviewers: Sajal Dam, Cristian Lefter, Alejandro Leguizamo, Alexzander Nepomnjashiy, Andrew Watt, Richard Waymire, Joe Webb, Roger Wolter
Editorial Board: Steve Anglin, Dan Appleman, Ewan Buckingham, Gary Cornell, Tony Davis, Jason Gilmore, Jonathan Hassell, Chris Mills, Dominic Shakeshaft, Jim Sumser
Project Manager: Kylie Johnston
Copy Edit Manager: Nicole LeClerc
Copy Editors: Ami Knox, Nicole LeClerc
Assistant Production Director: Kari Brooks-Copony
Production Editor: Kelly Winquist
Compositor: Susan Glinert
Proofreaders: Kim Burton, Linda Marousek, Linda Seifert, Liz Welch
Indexer: Broccoli Information Management
Artist: Kinetic Publishing Services, LLC
Cover Designer: Kurt Krames
Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders-ny@springer-sbm.com, or visit http://www.springeronline.com.

For information on translations, please contact Apress directly at 2560 Ninth Street, Suite 219, Berkeley, CA 94710. Phone 510-549-5930, fax 510-549-5939, e-mail info@apress.com, or visit http://www.apress.com.

The information in this book is distributed on an "as is" basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is available to readers at http://www.apress.com in the Source Code section.

This book is dedicated to my loving wife, Stacy, who provided the support, love, and encouragement that made this endeavor possible.

—Thomas Rizzo

Contents

About the Authors xvii
About the Technical Reviewers xix
Acknowledgments xxi
Introduction xxiii

CHAPTER 1 SQL Server Overview and Installation 1

Evolution of SQL Server 1
SQL Server 2005 Overview 3
Editions 3
Features 4
Installation 8
Minimum System Requirements 8
Installation Process 10
Server Registration 19
Sample Databases 22
Side-by-Side Installation 23
Upgrading from SQL Server 2000 and 7.0 23
Summary 23

CHAPTER 2 SQL Server Management Technologies 25

Connecting to and Managing Your SQL Servers 25
Context-Sensitive Menus for Each Object Type 26
Mobile Database Support 27
SQL Server 2000 Support 28
User Interface Enhancements 28
Asynchronous Treeview and Object Filtering 29
Nonmodal and Resizable Dialog Boxes 29
Script and Schedule Enhancements 30
Code Authoring 31
Results Pane 33
Activity Monitor 34
Summary Views 34

Functional Enhancements 35
Dedicated Administrator Connection 35
Deadlock Visualization 36
Performance Monitor Correlation 37
Server Registration Import/Export 38
Maintenance Plan Designer 38
SQL Server Agent Changes 39
Database Mail 39
Catalog Views and Dynamic Management Views 41
Default Server Trace 43
Profiler Enhancements 44
SQL Configuration Manager 44
Surface Area Configurator 45
Enhanced Help and Community Integration 47
SQLCMD 48
Server Management Objects 49
Summary 52

CHAPTER 3 T-SQL Enhancements for Developers 53

Enhancements Affecting DML 53
Old-Style Outer Joins Deprecated 54
Common Table Expressions 55
TOP 63
Extensions to the FROM Clause 66
OUTPUT 76
Ranking Functions 77
EXCEPT and INTERSECT 83
Synonyms 85
General Development 86
Error Handling 87
.WRITE Extension to the UPDATE Statement 94
EXECUTE 96
Code Security Context 96
.NET Declarations 99
Summary 101

CHAPTER 4 T-SQL Enhancements for DBAs 103

Metadata Views 103
Compatibility Views 104
Catalog Views 104
Dynamic Management Views and Functions 106

DDL Triggers 107
Creating and Altering DDL Triggers 108
Dropping DDL Triggers 109
Enabling and Disabling DDL Triggers 109
Enumerating DDL Triggers Using Catalog Views 109
Programming DDL Triggers with the eventdata() Function 110
Indexing and Performance Enhancements 112
Online Indexing 112
Controlling Locking During Index Creation 113
Creating Indexes with Additional Columns Included 113
Altering Indexes 114
Using Statistics for Correlated DATETIME Columns 117
Improving Performance of Ordering for Tertiary Collations 118
Table and Index Partitioning 119
Partition Functions 120
Partition Schemes 120
Creating Partitioned Tables and Indexes 121
Adding and Removing Partitions 122
Modifying Partition Functions and Schemes 123
Switching Tables into and out of Partitions 124
Managing Table and Index Partitions 125
Enhancements to Tables and Views 125
Enhancements to Indexed Views 125
Persisted Computed Columns 126
Snapshots 127
SNAPSHOT Isolation Level 127
Database Snapshots 130
Data Integrity Enhancements 132
Verifying a Database's Pages 132
Putting a Database into an Emergency State 132
Summary 133

CHAPTER 5 .NET Integration 135

Introduction to SQL Server .NET Integration 135
Why Does SQL Server 2005 Host the CLR? 136
When to Use CLR Routines 136
When Not to Use CLR Routines 136
How SQL Server Hosts .NET: An Architectural Overview 137
SQL Server .NET Programming Model 137
Enhancements to ADO.NET for SQL Server Hosting 138
Overview of the New .NET Namespaces for SQL Server 138

Programming a CLR Stored Procedure 139
Starting a Visual Studio 2005 SQL Server Project. 140
Anatomy of a Stored Procedure 143
Adding Parameters. 144
Defining the Problem 144
Using the SqlPipe 146
Putting It All Together: Coding the Body of the Stored Procedure. 148
Testing the Stored Procedure 151
Debugging the Procedure 152
Throwing Exceptions in CLR Routines 153
Deploying CLR Routines 157
Summary 159

CHAPTER 6 Programming Assemblies 161

CLR User-Defined Types 161
Applications for User-Defined Types 162
Adding a User-Defined Type to a SQL Server Project. 162
Parts of a User-Defined Type 164
A Simple Example: The PhoneNumber Type 167
Another Example: The StringArray Type 175
Managing User-Defined Types 182
CLR User-Defined Functions 183
Adding a User-Defined Function to a Visual Studio Project 184
The Visual Studio 2005 User-Defined Function Template 184
The SqlFunction Attribute 184
Scalar User-Defined Functions 185
Table-Valued User-Defined Functions 188
Managing CLR User-Defined Functions 192
CLR User-Defined Aggregates 193
Adding a User-Defined Aggregate to a SQL Server Project 193
Parts of a User-Defined Aggregate. 195
CLR User-Defined Triggers 200
Adding a CLR User-Defined Trigger to a SQL Server Project. 200
Programming CLR Triggers. 201
Managing User-Defined Triggers 205
Managing Assemblies 205
A Note Regarding Visual Studio 2005 206
Summary 206

CHAPTER 7 SQL Server and XML 207

What Is XML? 207
What Are XPath and the XMLDOM? 208
XPath Syntax 210
XPath Functions 211
The XMLDOM–XML Document Object Model 211
The XPathDocument, XPathNavigator, and XPathExpression Classes 212
Getting XML into the Database 213
What Is SQLXML? 214
Configuring SQL Server 214
OPENXML 215
SQLXML: XML Views Using Annotated XML Schemas 220
SQLXML Updategrams 226
XML BulkLoad 228
Getting XML Out of the Database: FOR XML 230
FOR XML (Server-Side) 230
FOR XML (Client-Side) 236
Using Templates 236
Enhancements to FOR XML 237
Programming SQLXML from .NET and COM 238
SqlXmlCommand 238
SqlXmlParameter 239
SqlXmlAdapter 240
SqlXmlException 240
Code Samples 240
FOR XML: Server-Side and Client-Side 242
Using an XMLTextReader 242
Using Parameters with SQLXML 243
Executing XPath or SQL Queries with Templates 244
Interoperating with the ADO.NET Dataset 244
Programming Updategrams 245
Summary 245

CHAPTER 8 SQL Server 2005 XML and XQuery Support 247

Using the XML Datatype 248
Understanding How XML Is Stored by SQL Server 249
Creating XML Columns 250
Setting Permissions for Schema Creation 254
Constraining XML Columns 255
Examining the XML Datatype Limitations 256

Inserting Data into XML Columns 257
Using SSIS with XML Data 257
Bulkloading XML 259
Writing a Custom Query or Application 259
Querying XML Data 260
XQuery 101 260
Basic XML Query Methods 266
Cross-Domain Queries 268
Modifying XML Data 269
Limitations of XML Modification 270
Indexing XML for Performance 270
Understanding How XML Indexing Works 271
Examining Secondary XML Indexes 272
Full-Text Search and the XML Datatype 273
Dynamic Management Views and XML 274
Applications and XML 274
XML Web Services Support 275
Creating an Endpoint 276
Using Advanced Web Services 280
Monitoring Performance of XML Web Services 285
Summary 286

CHAPTER 9 SQL Server 2005 Reporting Services 287

Reporting Services Components 287
Report Server 289
Metadata Catalog 289
Report Designer 289
Report Manager Web Application 291
Reporting Services Security 292
Building a Basic Report with SSRS 2000 293
Launching the Designer 293
Working with Data Sources and Datasets 293
Laying Out and Previewing the Report 294
Working with Expressions 294
Deploying Your Report 295
Upgrading from SQL Server 2000 Reporting Services 295
Licensing Changes for Reporting Services 296
SQL Server Management Studio Integration 297
Walkthrough: Management Studio and Reporting Services 298
Management Changes 304
WMI Provider 305
Management and Execution Web Services 307
Reporting Services Configuration Tool 307

Report Design and Execution Improvements 308
Expression Editor 309
Multivalued Parameters 310
DatePicker for Date Values 312
Interactive Sorting 313
Analysis Services Integration 314
Walkthrough: Building a Report in BIDS 315
Floating Headers 322
Data Source Changes: Expressions, XML/Web Services, SSIS, and SAP 323
Custom Report Items 328
Visual Studio Integration and ReportViewer Controls 329
Using WinForm Controls 329
Working with the ReportViewer Controls Programmatically 332
LocalReport and ServerReport Objects 336
SharePoint Integration 337
End-User Ad Hoc Query and Reporting 337
The Report Builder Client 338
The Semantic Model Definition Language 338
Walkthrough: Report Builder 339
Summary 346

CHAPTER 10 Analysis Services 347

SSAS 2005 Enhancements 347
Architecture 348
Performance, Scalability, Availability 349
Usability 349
Development 350
Installation 351
What Is Analysis? 352
OLAP, OLTP, and Data Warehouses 352
OLAP Concepts 353
Cubes 353
Cells 354
Measures and Fact Tables 354
Dimensions and Attributes 354
Hierarchies 354
Analysis Services Projects 355
Defining Data Sources 356
Designers vs. Wizards 359
Defining Data Source Views 360
Defining Cubes 363
Deploying Projects and Configuring Projects for Deployment 367
Cube Operations 369

Browsing Cubes 370
Browsing Cubes with Hierarchies 372
Managing Displayed Data 374
Calculations and MDX 376
Key Performance Indicators (KPIs) 379
Analysis Services Scripting Language (ASSL) 382
Example ASSL Object: A Data Source View with a Named Query 382
ASSL Drilldown 384
Summary 385

CHAPTER 11 Security 387

A Word about sa 387
Surface Area Configuration 387
Remote Connections 388
Dedicated Administrator Connection 388
.NET Framework 388
Database Mail 389
SQLMail 389
Service Broker 389
HTTP Connectivity 390
Database Mirroring 390
Web Assistant 390
xp_cmdshell 390
Ad Hoc Remote Queries 390
OLE Automation XPs 390
SMO and DMO XPs 391
Principals and Securables 391
Principals 391
Securables 398
Permissions 401
Types of Permission 401
Managing Permissions 403
Code Access Security 405
Imperative and Declarative CAS 406
Using CAS with SQL Server 406
Encryption 410
The SQL Server 2005 Encryption Hierarchy 411
Encryption with a User-supplied Password 411
Encryption with a Symmetric Key 412
Asymmetric Key Encryption 413
Encryption with a Certificate 414
Certificates and Web Services 415
Summary 416

CHAPTER 12 Service Broker 419

What Is Service Broker? 419
Service Broker Architecture 420
Service Broker Scenarios 422
Creating Service Broker Applications 423
Enabling Service Broker 423
Creating Message Types 424
Creating Contracts 424
Creating Queues 424
Creating Services 425
Creating Service Broker Stored Procedures 425
A Simple Service Broker Example 428
Service Broker Routing and Security 434
Creating Distributed Service Broker Applications 434
Distributed Service Broker Example 437
Summary 457

CHAPTER 13 Automation and Monitoring 459

SQL Server Agent 459
Step 1: Connect to SQL Server 460
Step 2: Create the Agent Job 461
Security Enhancements 465
Agent Subsytems 473
Sharing Job Schedules 474
Logging to the sysjobstepslogs Table 476
WMI Events and Agent Alerts 476
Agent Performance Counters 478
Agent Upgrade 479
Maintenance Plans 480
Creating a Maintenance Plan 481
Toolbox 482
Maintenance Plan Designer Document Window 483
SQLCMD 488
Connecting to SQL Server 488
Passing Variables 489
Using the Dedicated Admin Connection 490
Creating Scripts 490
Database Mail 491
Overview 492
Configuring Database Mail 493
Sending Mail 497

SQL Profiler 498
Performance Monitor Correlation 501
ShowPlan 502
Deadlock Visualization 504
Summary 505

CHAPTER 14 Integration Services 507

What's New in SSIS? 508
SSIS's New IDE 509
Connecting to SSIS in Management Studio 509
Creating a New SSIS Project in BIDS 511
SSIS Fundamentals 511
Control Flow Design Surface 512
Data Flow Design Surface 514
Event Handlers Design Surface 526
Package Explorer 528
Connection Managers 529
Solution Explorer 529
Properties Window 529
Control Flow Toolbox Tasks 530
Containers 531
Analysis Services Tasks 535
Data Flow Task 535
Execute Package Tasks 536
Bulk Insert Task 536
Execute SQL Task 536
Execute Process Task 537
File System Task 537
File Transfer Protocol Task 538
Maintenance Plan Tasks 538
Message Queue Task 538
Send Mail Task 539
Scripting Tasks 539
Web Service Task 542
WMI Tasks 542
XML Task 542
Data Flow Designer Tasks 542
Source Adapters 543
Destination Adapters 543
Transformations 544
Logging 550

Configurations 554
Using the Package Configuration Organizer 554
Variables 556
Precedence Constraints 559
Checkpoints 559
Transactions 560
Debugging 560
Control Flow Visual Debugging 560
Data Flow Visual Debugging 562
Data Viewers 562
Breakpoints Window 562
Other Debug Windows 563
The SSIS Package Deployment Utility 563
Migrating SQL Server 2000 Packages 564
Scheduling an SSIS Package 564
Summary 565

CHAPTER 15 Database Mirroring 567

High Availability Defined 568
Database Mirroring Overview 569
Database Mirroring in Context 571
Setting Up Database Mirroring 572
Prerequisites, Connectivity, and Security 573
Back Up and Restore the Principal Database 576
Establish the Principal/Mirror Partnership 577
Changing Transaction Safety Levels 577
Database Mirroring States, Heartbeats, and Quorums 578
Initiating a Failover 579
Suspending and Resuming Mirroring 579
Terminating Database Mirroring 580
Full-Text Indexing and Mirroring 580
Service Broker and Database Mirroring 580
Setting Up Mirroring Using Management Studio 581
Client Applications and Database Mirroring 587
Monitoring Database Mirroring 588
Catalog Views 588
Performance Monitor Counters 592
Profiler 593
Windows Event Log and SQL Server Error Log 593
Performance Considerations for Mirroring 594
Limitations of Database Mirroring 594
Sample Application Walk-Through 595

Database Snapshots and Mirroring 601
Database Snapshots Overview 601
Working with Snapshots in T-SQL 603
Performance Considerations When Using Snapshots on Mirrors 604
Using, Monitoring, and Managing Database Snapshots. 604
Programming Database Snapshots 605
Limitations of Database Snapshots 605
Windows Clustering in SQL Server 2005 606
Replication in SQL Server 2005 607
Summary 607

CHAPTER 16 Notification Services 609

Notification Services Architecture 610
Subscribers 611
Subscriptions 611
Events 611
Notifications 612
Building a Notification Services Application 612
Defining an NS Instance: The Instance Configuration File 613
Defining the NS Application: The Application Definition File 617
Compiling and Running Your NS Application 633
Monitoring and Troubleshooting Your NS Application 636
Programmatically Working with NS 639
Programming NS from Visual Studio 639
Managing NS Programmatically 642
Summary 643

INDEX 645

About the Authors

TOM RIZZO is a director in the SQL Server group at Microsoft. Being an 11-year veteran at Microsoft, Tom has worked on a number of the different Microsoft server technologies such as BizTalk, SharePoint, and Exchange Server before joining the SQL Server group. Tom is a published author on topics ranging from SQL Server to developing collaborative applications using Microsoft's collaboration servers.
Tom authored Chapters 2, 7, 8, 9, 15, and 16.

ADAM MACHANIC is a database-focused software engineer, writer, and speaker based in Boston, Massachusetts. He has implemented SQL Server for a variety of high-availability OLTP and large-scale data warehouse applications, and he also specializes in .NET data access layer performance optimization. He is a Microsoft Most Valuable Professional (MVP) for SQL Server and a Microsoft Certified Professional (MCP).
Adam authored Chapters 4, 5, and 6.

JULIAN SKINNER studied Germanic etymology to PhD level before joining Wrox Press as an indexer in 1998 in order to get a real job. He became a technical editor shortly after that, later working as a technical architect and commissioning editor, before moving to Apress in 2003. He has consequently spent most of the last six years reading books about programming, focusing in particular on Microsoft technologies and, since 2000, on C# and the .NET Framework. He recently left Apress to concentrate on writing code.

Julian contributed many sections and code samples—and often whole chapters—to the books he worked on at Wrox, mostly hiding behind the relative anonymity of an "additional material" credit, but he is credited as a coauthor of *Professional ADO.NET*, *Professional ASP Data Access*, and *Beginning SQL*. He is also a coauthor of *The Programmer's Guide to SQL*, published by Apress.
Julian authored Chapters 11 and 12.

LOUIS DAVIDSON has been in the information technology industry for ten years, as a corporate database developer and architect. Currently, he is serving as a database administrator for Compass Technology Management in their Nashville data center, supporting the Christian Broadcasting Network and NorthStar Studios.

Davidson has a bachelor's degree from the University of Tennessee at Chattanooga in computer science with a minor in mathematics (though the minor in math is more of an indication of the amount of math required at UTC to get a computer science degree, rather than any great love or skill in the subject).

The majority of his experience, with slight deviations into Visual Basic, has been spent with Microsoft SQL Server from version 1.0 to whatever the latest version is in beta. Louis's primary areas of expertise are in database architecture and coding in Transact-SQL, and he has written thousands of numerous stored procedures and triggers throughout the years.

Louis was the sole author of *Professional SQL Server 2000 Database Design*, and he was a contributor to *SQL Server 2000 Stored Procedure Handbook*. It is said that in his ridiculously small amount of spare time he tends to play a lot of Nintendo (what is it with that silly princess—she's been captured by the freaking dragon again!) as well as watching a great deal of television that was popular in a

different era, most notably old English programs such as *The Avengers*, *The Saint*, *Monty Python's Flying Circus*, and *Black Adder*, to name a few. Quite often this spare time is also spent with his notebook computer writing something pertaining to SQL.
Louis authored Chapter 3.

ROBIN DEWSON has been hooked on programming ever since he bought his first computer, a Sinclair ZX80, in 1980. His first main application of his own was a Visual FoxPro application that could be used to run a fantasy league system.

From there, realizing that the market place for Visual FoxPro in the United Kingdom was limited, he decided to learn Visual Basic and SQL Server. Starting out with SQL Server 6.5, he soon moved to SQL Server 7 and Visual Basic 5, where he became involved in developing several applications for clients in both the UK and the United States. He then moved on to SQL Server 2000 and Visual Basic 6, through to SQL Server Yukon and Visual Basic .NET.

Robin is a consultant mainly in the city of London, where he has been for nearly eight years. He also has been developing a rugby-related website as well as maintaining his own site at `http://www.fat-belly.com`.
Robin authored Chapter 1.

JAN D. NARKIEWICZ (`jann@softwarepronto.com`) is chief technical officer of Software Pronto, Inc. His areas of expertise include Microsoft technologies, Oracle, and DB2. Jan also write books for Apress and serves as academic coordinator for U.C. Berkeley Extension's .NET/Windows program. His clients include E*Trade, Visa, eBay, and Oracle. Jan also acts as an expert witness in patent, copyright, and licensing-related litigation.
Jan authored Chapter 10.

JOSEPH SACK is a database administration and developer based in Minneapolis, Minnesota. Since 1997, he has been developing and supporting SQL Server environments for clients in financial services, IT consulting, manufacturing, and the real estate industry. Joseph received his bachelor's degree in psychology from the University of Minnesota. He is the author of *SQL Server 2000 Fast Answers for DBAs and Developers*, the coauthor of *Beginning SQL Server 2000 DBA: From Novice to Professional*, and is a Microsoft Certified Database Administrator (MCDBA).
Joe authored Chapter 14.

ROB WALTERS is a program manager in the SQL Server group of Microsoft. He has seven years of experience in software development and relational databases. When not talking about databases, Rob enjoys spending time with his wife, Tammie, their son, Bryan, and two overfed St. Bernard dogs.
Rob authored Chapter 13.

About the Technical Reviewers

SAJAL DAM works as an IT strategist at Dell, managing one of the largest SQL Server environments. The challenges of database performance tuning excite him the most. He has written a couple of books on SQL Server query performance tuning and is in the process of starting his next book on SQL Server 2005 performance tuning.

Besides his technical acumen, Sajal is passionate to learn how business decisions are made in successful corporations. To fulfill his passion, he has started his executive MBA from Duke alongside his other works.

In his free time, Sajal reviews other technical books and plays in the stock market. He can be reached at `sajaldam1@hotmail.com`.

CRISTIAN LEFTER is a SQL Server MVP, former developer, database administrator, and trainer. He is currently CEO of MicroTraining, a consulting and training company.

In his spare time, Cristian is a tech reviewer, author, and leader of two user groups (ITBoard and Romanian SQL Server User Group).

ALEJANDRO LEGUIZAMO has been working with SQL Server since 6.5, and with Microsoft Access since Office 97. He is certified in SQL Server 2000, mainly focused in data warehousing and business intelligence, plus ETL. He is certified in SQL Server 2000, and he has wide experience in training and consulting.

Alejandro earned a degree in business management focused on executive information systems. He is based in Bogotá, Colombia, and has been invited to participate as speaker and expert in the United States, Puerto Rico, Peru, Ecuador, Spain, Venezuela, and other countries, at events like internal trainings for Microsoft, the launch of SQL Server Reporting Services, Developer Days, TechEd 2005, among others. Alejandro was awarded the Microsoft Most Valuable Professional (MVP) award first in 2004, and again in 2005. Currently, he is a mentor to the well-known group of experts in SQL Server, Solid Quality Learning (`http://www.solidqualitylearning.com`), for the Iberoamerican operations and the BI division.

Currently, **ALEXZANDER NEPOMNJASHIY** is working as Microsoft SQL Server DBA with NeoSystems North-West Inc., an ISO 9001:2000 certified software company. As a DBA, he is responsible for drafting design specifications for solutions and building database-related projects based on these specs. As an IT professional, Alexzander has more than 11 years of overall experience in DBMS planning, designing, securing, troubleshooting, and performance optimizing. He can be reached at `alexnep@onego.ru`.

RICHARD WAYMIRE is a lead program manager with Microsoft and has worked on the development of Microsoft SQL Server 7.0, 2000, and 2005. He is the author of several books on SQL Server, including most recently *Teach Yourself SQL Server 2000 in 21 Days*, and he is also a contributing editor to *SQL Server Magazine*.

JOE WEBB is the founder and chief operating manager of WebbTech Solutions. He has over 11 years of industry experience and has consulted extensively with companies in the areas of software development, database design, and technical training. Joe also serves on the board of directors for PASS, the Professional Association for SQL Server.

As a Microsoft MVP, Joe regularly speaks at technical conferences in the United States and in Europe. He is also the author of *The Rational Guide To: SQL Server Notification Services* and *The Rational Guide To: IT Consulting* (`http://www.rationalpress.com`).

When he's not working, Joe enjoys the farm life on his small farm in the middle of Tennessee, where he raises vegetables and livestock. He's been blessed with a wonderful wife and family.

ROGER WOLTER has 27 years of experience in various aspects of the computer industry, including jobs at Unisys, Infospan, and Fourth Shift. He has spent the last seven years as a program manager at Microsoft. His projects at Microsoft include SQLXML, the Soap Toolkit, SQL Server Service Broker, and SQL Server Express. His interest in the Service Broker was sparked by a messaging-based manufacturing system he worked on in a previous life. He's currently splitting his time between the Service Broker and the SQL Server Express projects in SQL Server 2005.

Acknowledgments

While there are too many people to acknowledge, I will give it my best shot. First, I'd like to acknowledge the great team at Apress, including Tony Davis, the editor of this book, for his dedication and hard work getting this off the ground. At some points during the early days of SQL Server 2005, we thought this book would never make it due to shifting contributing authors and the shifting product. This book is a testament to Tony's dedication to providing the highest quality educational materials to his readers. I also would like to thank Kylie Johnston, who worked hard to keep us all on track. She had to herd cats—and sometimes very, very reluctant cats—but in the end she pushed us hard and made this book better than we would have made it ourselves.

I'd like to also thank my technical reviewers, Sajal Dam, Cristian Lefter, Alejandro Leguizamo, Alexzander Nepomnjashiy, Andrew Watt, Richard Waymire, Joe Webb, and Roger Wolter. They kept me honest and pointed out where I could improve my explanations to the benefit of all readers.

Finally, there are a number of people that work with me on SQL Server that I have to thank for their tireless explanations of the nitty-gritty technical details at all hours of the day. These include Jason Carlson and Brian Welcker from the Reporting Services team, Shyam Panther from the NS team, Michael Rys and Shankar Pal from the XML team, Srik Raghavan and Brian Deen from the WebData team, Mark Wistrom and Christian Kleinerman from the SQL Server engine team, Mahesh Prakriya from the Management Studio team, and Euan Garden, who used to be on the Management Studio team but now heads up our product planning efforts for SQL Server.

Thomas Rizzo

Introduction

This book provides a critical examination of all of the major new functionality in SQL Server 2005, covering such diverse topics as CLR integration, the new management tools, SQL Server Integration Services, Service Broker, Transact-SQL (T-SQL) programming, and database mirroring.

The book does not profess or even try to be a comprehensive reference on any one of these areas—as you are probably aware, this would often require a sizable book in itself. Instead, it provides practical, in-depth coverage of the core topics in each area, illustrated with realistic examples. Hopefully, we've done this in such a way that you will immediately be able to translate what you learn here into your business environment and have a firm foundation for exploring a particular topic further, should it be necessary.

SQL Server 2005 is a vast new release. This book provides you with a starting point, a road map, and a strong foundation on which to build. Its practical nature and careful guidelines and advice will mean that the book continues to be useful long after your initial assessment of SQL Server 2005 is complete.

Who This Book Is For

This book is for anyone who wants to learn about SQL Server 2005. The topics are diverse and deep, and there is something in here for everyone, whether you are a DBA, developer, or business intelligence (BI) practitioner. As long as you have a sound base knowledge of SQL and relational database in general, then this book will teach you about the extensive new feature set of SQL Server 2005 and about how best to put these features to work in your environment.

How This Book Is Structured

This book is written is such a way that you can read through the book cover to cover or dip in and out for specific topics. It is structured as follows.

Chapter 1: SQL Server Overview and Installation

This chapter details a brief history on the evolution of SQL Server from a "desktop database" to a full-fledged enterprise-class relational database management system (RDBMS). It provides a quick reference guide to the new SQL Server 2005 feature set for each of the SQL Server editions, and then steps through the whole installation process. Many readers will already have SQL Server installed, but if you're downloading it for the first time from MSDN (we recommend using SQL Server Developer Edition), then this chapter will get you set up and ready to work through all of the examples in the book.

Chapter 2: SQL Server Management Technologies

SQL Server Management Studio (SSMS) is the major new management tool for SQL Server 2005. It combines most of the tools that you previously used separately (Enterprise Manager, Query Analyzer, and so on), and adds additional capabilities. This chapter details the functional and interface enhancements that have been made and how they might affect you. It also takes a look at the new Server Management Objects (SMO) technology, the successor to SQL-DMO.

Chapter 3: T-SQL Enhancements for Developers

Reports of the imminent demise of T-SQL have been greatly exaggerated. This chapter explores the feature and performance enhancements from a developer's perspective, covering such topics as common table expressions (CTEs), new join types, improved error handling, and more.

Chapter 4: T-SQL Enhancements for DBAs

This chapter switches focus to the numerous administration enhancements such as DDL triggers, table and index partitioning, snapshots, and the new `SNAPSHOT` isolation level.

Chapter 5: .NET Integration

Although T-SQL is alive and well, there are some things that it just isn't meant to do. Previously, when T-SQL ran out of steam, developers were plunged into the complex world of extended stored procedures. No longer. In many people's eyes, the biggest advancement in 2005 is the inclusion of the common language runtime, or CLR, within the database. As a result, developers can now create objects (stored procedures, user-defined functions, and so on) using any of the .NET languages (VB .NET, C#, C++, etc.) and compile them into .NET assemblies. These assemblies are deployed inside the database and run by the CLR, which is hosted inside the SQL Server memory space. This chapter introduces programming with CLR objects via a step-by-step tour through development of a CLR stored procedure. It describes the .NET object model provided for SQL Server CLR development, along with best practices for developing CLR objects and various deployment issues.

Chapter 6: Programming Assemblies

This chapter continues the exploration of CLR integration with some in-depth examples on the use of CLR user-defined types, functions, aggregates, and triggers.

Chapter 7: SQL Server and XML

This chapter provides an overview of the XML technology as it relates to SQL Server. It takes a broad look at XPath and XML Schema support in SQL Server 2005, and then drills down into how to get XML into and out of the database. It covers how to get XML into your relational data columns using `OPENXML`, updategrams, and SQLXML's XML `Bulkload` provider. It then shows how to query the relational columns and return the results as XML, using `FOR XML`.

Chapter 8: SQL Server 2005 XML and XQuery Support

This chapter investigates native XML support in SQL Server 2005, via the new `XML` datatype. It shows how to create XML columns, insert data into those columns, and then retrieve that XML data using XQuery.

Chapter 9: SQL Server 2005 Reporting Services

SSRS 2005 is the latest and most powerful reporting technology from Microsoft. An integral part of the SQL Server 2005 database, it allows you to design, author, render, and deploy reports via the Web or a company intranet. This chapter starts out by showing you how to create a report using SQL Server 2000 Reporting Services and then how to migrate that report to SSRS 2005. Next, it describes, and shows how to take advantage of, the numerous SSRS 2005 feature enhancements, such as multi-valued parameters, interactive sorting, and the use of the new ad-hoc Report Builder.

Chapter 10: Analysis Services

Databases store data, but they become truly profitable when that data can used and interpreted to provide business intelligence (BI). Powered by the new Business Intelligence Development Studio (BIDS), SQL Server Analysis Services (SSAS) is the major new suite of technologies designed to support the development and administration of BI applications. Described in this chapter are the SSAS mechanisms for exploiting Online Analytical Processing (OLAP) and data mining.

Chapter 11: Security

As with most areas of SQL Server, the security features built into SQL Server 2005 have undergone a fairly radical overhaul. This chapter takes a look at the new features for granting and denying permissions to access resources in the database, and the new system of schemas, which now resemble ANSI SQL schemas far more closely. It addresses new security functionality, such as the Surface Area Configurator (SAC) feature and the new encryption functions.

Chapter 12: Service Broker

One of the most important new features of SQL Server 2005 is Service Broker. Service Broker is a message queuing technology that is native to SQL Server and allows developers to integrate SQL Server fully into distributed applications. Service Broker provides an asynchronous system for database-to-database communication; it allows a database to send a message to another without waiting for the response, so the application will continue to function if the remote database is temporarily unavailable. All of this is demonstrated in this chapter with in-depth working examples.

Chapter 13: Automation and Monitoring

SQL Server 2005 brings with it advancements in many areas that will make the daily administration and maintenance of SQL Server much easier. The first half of this chapter takes an in-depth look at SQL Server Agent 2005, the task scheduling service used by SQL Server to execute a variety of jobs, including T-SQL, replication, and maintenance tasks. The chapter then moves on to examine tools such as Maintenance Plans, SQLCMD, and database mail, and demonstrates how they can make a SQL Server DBA's life easier.

Chapter 14: Integration Services

SQL Server Integration Services (SSIS), formerly known as Data Transformation Services (DTS), is Microsoft's extraction, transformation, and loading tool that comes bundled with SQL Server 2005. It has been massively overhauled and expanded, and this chapter will lead you through all of the significant changes. It guides you through all of the data flow, control flow, and transformation tasks, using plenty of hands-on examples along the way to really demonstrate the power of this new tool.

Chapter 15: Database Mirroring

Although disabled in the first SQL Server 2005 release, database mirroring is a very significant new feature. Microsoft is committed to re-enabling it after a period of extra testing, so many DBAs will want to find out what it can do and prepare for its adoption. This chapter investigates the new database mirroring capability and gives detailed instructions on how to set up and use it. It relates database mirroring to existing technologies, such as failover clustering, replication, and log shipping, and provides advice on which technology is best to solve a particular problem.

Chapter 16: Notification Services

SQL Server 2005 now comes with a built-in dynamic subscription and publication mechanism—namely, Notification Services (NS). This chapter fully describes the NS architecture, walks you through how to create a NS application, and then covers how to program with NS: creating and modifying your subscribers, devices, and subscriptions; submitting events to NS; working with custom components; and so on.

Prerequisites

Ideally, you will be running the examples in this book on the final release version of SQL Server 2005 and Visual Studio 2005. However, at a minimum, you need at least the September CTP of SQL Server 2005 and the release candidate of Visual Studio 2005.

While some chapters do not require Visual Studio, having Visual Studio will give you the best overall experience with this book. Of course, you should follow the software prerequisites and system requirements suggested by both the SQL Server 2005 and Visual Studio 2005 documentation.

Source Code and Updates

As you work through the examples in this book, you may decide that you want to type in all the code by hand. Many readers prefer this because it is a good way to get familiar with the coding techniques that are being used.

Whether you want to type the code in or not, all the source code for this book is available in the Source Code area of the Apress website (`http://www.apress.com`). If you like to type in the code, you can use the source code files to check the results you should be getting—they should be your first stop if you think you might have typed in an error. If you don't like typing, then downloading the source code from Apress website is a must! Either way, the code files will help you with updates and debugging.

Errata

Apress makes every effort to make sure that there are no errors in the text or the code. However, to err is human, and as such we recognize the need to keep you informed of any mistakes as they're discovered and corrected. An errata sheet will be made available on this book's main page at `http://www.apress.com`. If you find an error that hasn't already been reported, please let us know.

The Apress website acts as a focus for other information and support, including the code from all Apress books, sample chapters, previews of forthcoming titles, and articles on related topics.

Contacting the Authors

You can contact this book's lead author, Tom Rizzo, either via his e-mail address at thomriz@microsoft.com or via his blog at http://www.sqljunkies.com/WebLog/tom_rizzo/default.aspx.

CHAPTER 1

SQL Server Overview and Installation

SQL Server 2005 is a major advancement over SQL Server 2000. Right from the very beginning of your SQL Server 2005 experience, you will notice great changes in the installation process. You'll see as you progress through this book that these changes continue throughout the product.

In this chapter, we'll briefly overview how SQL Server has evolved in recent years, and then we'll look at the current editions of SQL Server 2005 and the features offered with each.

We'll then examine the compatibility of different editions (32-bit and 64-bit) of SQL Server with the various available operating systems flavors and take a look at the minimum system requirements for SQL Server 2005 installation.

Having done that, we'll walk through the installation process itself step by step, discussing the major considerations you'll need to take into account along the way. If you've performed or seen an installation of Visual Studio .NET, then the SQL Server 2005 installation process will be familiar to you—it's very similar in its methodology. No longer do you have to run several installations to ensure all the components you want are there. A treeview structure now lists all the possible components and combinations, thereby allowing you to install everything you need in one pass. There are also two new example databases you can install and use for testing new functionality provided with SQL Server 2005.

Evolution of SQL Server

Table 1-1 briefly charts the evolution of SQL Server, up to SQL Server 2000.

The very first version of SQL Server emerged in 1989/1990. It was available for OS/2, and its code base was essentially the same as Sybase SQL Server 4.0. The first edition of SQL Server for Windows NT emerged in 1993 and was a basic port of Sybase SQL Server 4.0 from OS/2 to NT.

The emergence of SQL Server 6.5 marked the split from Sybase; the database engine was completely rewritten specifically for NT. From that point on, SQL Server has evolved rapidly into a powerful enterprise-level database. SQL Server started life as a small, inexpensive desktop database, with some GUI management tools, and has been progressively expanding its enterprise feature set,

Table 1-1. *History of SQL Server*

Year	Version	Description
1993	SQL Server 4.2 (a desktop database)	A low-functionality *desktop* database, capable of meeting the data storage and handling needs of a small department. However, the concept of a database that was integrated with Windows and had an easy-to-use interface proved popular.
1995	SQL Server 6.5 (a small business database)	A major rewrite of the core database engine. This was SQL Server's first "significant" release, and it included improved performance and important feature enhancements. It still had a long way to go in terms of its performance and feature set, but it was now capable of handling *small e-commerce and intranet* applications, at a fraction of the cost of competitors' offerings.
1998	SQL Server 7.0 (a web database)	Another significant rewrite to the core database engine. Version 7.0 was a defining release, providing a reasonably powerful and feature-rich database that was a truly viable (and still cheap) alternative for *small-to-medium businesses*, between a true desktop database such as Microsoft Access and the high-end enterprise capabilities (and price) of Oracle and DB2. It gained a good reputation for ease of use and for providing crucial business tools (e.g., analysis services and data transformation services) out of the box, which were expensive add-ons with competing databases.
2000	SQL Server 2000 (an enterprise database)	Vastly improved performance scalability and reliability sees SQL Server become a major player in the *enterprise database* market (now supporting the online operations of businesses such as NASDAQ, Dell, and Barnes & Noble). A stiff price increase slowed initial uptake, but 2000's excellent range of management, development, and analysis tools won new customers.

scalability, and performance to the point where it is a serious competitor—most significantly to Oracle—in the medium-sized enterprise market (although, of course, SQL Server competes only on the Windows platform).

It is interesting to contrast SQL Server's journey from small business to enterprise with that of Oracle's, which in some ways has been pushing in the opposite direction. From the very start, Oracle was designed to handle large databases, and high numbers of transactions and users. In terms of "out-of-the-box" performance and scalability (i.e., the numbers of transactions and users per single instance), many consider Oracle to still be the superior database. However, some perceive that this superiority comes at the expense of high costs and complexity—but that the performance numbers are getting closer.

Whatever the truth might be, it is certain that the competition between Oracle and SQL Server is set to intensify with the release of SQL Server 2005. Certainly part of the drive behind the release of Oracle 10*g* appears to be to reduce the total cost of ownership (TCO) of the database and to make it easier to mange, introducing, as it does, a whole swath of "automated" management tools.[1]

In the meantime, SQL Server 2005 marks a significant advance in Microsoft's march into the enterprise database arena.

1. Of course, this is not the whole story. Oracle has also invested heavily in technologies such as Real Application Clusters (RAC), which ultimately is designed to reduce the cost of implementing highly scaleable enterprise systems—although at the moment it is still a very expensive technology!

SQL Server 2005 Overview

SQL Server 2005 brings with it a vast array of new features, graphical user interfaces (GUIs), and management tools, many of which are covered in this book. The following list should give you a brief taste of these:

- The ability to host the .NET Framework common language runtime (CLR) in the database so that you can now program assemblies in Visual Basic 2005 and C# *in the database*. This may have interesting consequences for the SQL Server database programmer, who previously was limited to SQL and T-SQL, and it will have dramatic implications for the way applications may be architected.
- Deep support for XML, via a full-fledged XML datatype that carries all the capabilities of relational datatypes. You can enter an XML document into your database, have it validated, and extract just part of the document. This means that you can marry semistructured data with relational data, storing them in the same place and treating them in the same way. Additionally, server-side support is provided for XML Query (XQuery) and XML Schema Definition language (XSD) standards.
- A completely revamped GUI management tool called SQL Server Management Studio (SSMS), which provides a single, integrated environment for most management/administration requirements.
- A reporting framework (SQL Server Reporting Services, or SSRS) as an integral part of the database.
- A new application framework, the Service Broker, for asynchronous message delivery.
- Vastly improved and expanded SQL Server Integration Services (SSIS; formerly Data Transformation Services), a tool for extracting, transforming, and loading data (again, a feature that is a costly add-on with other relational database management systems).

The latter three are excellent examples of features that SQL Server provides as an integral part of the product, rather than as (extra-cost) add-ons.

Editions

SQL Server 2005 is available in the following distinct editions:

- *Enterprise*: This is the most powerful, scalable, and expensive SQL Server 2005 edition. It is targeted, as its name suggests, at enterprise businesses where performance availability and scalability are of paramount importance. It supports all available features.
- *Developer*: This is the same as the Enterprise Edition, but with restrictions on CPUs and licenses.
- *Standard*: This edition is a cheaper option than Enterprise and Developer, and it is targeted at small- and medium-sized businesses. It removes support for such features as partitioning and online indexing, but it does support many of the "high-end" features, such as Analysis Services, Integration Services, database mirroring, and so on.
- *Workgroup*: This edition is designed for small- and medium-sized businesses and departmental solutions. It supports many of the core SQL Server features, but it doesn't include high-availability features, and it also has limited analysis functionality.
- *Express*: This edition replaces Microsoft SQL Server Desktop Engine (MSDE). However, it inherits many (nonenterprise) features from SQL Server 2005 and comes complete with its own dedicated (albeit limited) development and administration tools. It is freely available and is an ideal database for departmental solutions, prototype or evaluation projects, and hobbyists.

Table 1-2 outlines the CPU, memory, and size limitations for each edition.

Table 1-2. *Hardware Limitations for Each SQL Server Edition*

Feature	Enterprise/Developer	Standard	Workgroup	Express
Maximum number of CPUs	No limit	4	2	1
Maximum amount of RAM	No limit	No limit	3GB	2GB
64-bit processor supported	Yes	Yes	Windows on Windows	Windows on Windows
Maximum size for a database	No limit	No limit	No limit	4GB

Features

Table 1-3 provides an overview of the "core" new features of SQL Server 2005, with a brief description of each and an indication of the edition(s) in which it is supported.

Table 1-3. *Core SQL Server 2005 Features*

Feature	Description	Supported In
Advanced performance tuning	Mining models can receive advanced performing tuning.	Enterprise
Advanced transforms such as data mining, text mining, and data cleansing	The Enterprise Edition allows the inclusion of Analysis Services–based transforms and mining capabilities within the SSIS packages.	Enterprise
Database available for use while transaction undo operations in progress	Databases can be available for use during the undo phase while a restore is in progress.	Enterprise
Data flow integration	SSIS can be used to improve the mining model for creating prediction queries.	Enterprise
Indexes on a view	SQL Server allows creation of indexes on a view.	Enterprise
Parallel indexing operations	Indexing can run in parallel on multiprocessor computers.	Enterprise
Online database restore	A database can be restored without taking it offline.	Enterprise
Online indexing of tables and views	Tables and views can be indexed while users are still working with the system.	Enterprise
Oracle replication	SQL Server databases can now replicate to an Oracle database.	Enterprise

Table 1-3. *Core SQL Server 2005 Features (Continued)*

Feature	Description	Supported In
Partitioning	Tables can be split up (physically or logically) into smaller units to speed processing of data and indexing.	Enterprise
Text mining	Structured text can be created for SQL Server Analysis Services (SSAS) from unstructured text.	Enterprise
Database mirroring	Changes completed in one database are mirrored in another.	Enterprise Standard
Database Tuning Advisor (DTA)	The DTA tool provides tuning advice for the whole database and replaces the Index Tuning Wizard.	Enterprise Standard
Failover clustering	A database can failover to another database on a point of failure.	Enterprise Standard
Integration Services, including graphical Extract, Transform, and Load (ETL)	Integration Services is a tool for extracting, transforming, and loading data. This used to be known as DTS.	Enterprise Standard
Notification Services, for sending out notifications to subscribers	Notification Services is used for applications that generate and send notifications of events that happen within SQL Server to any subscriber, whether it is a PDA, mobile phone, etc.	Enterprise Standard
Web services	Support for native web services allows you to expose specific SQL Server objects such as stored procedures, user-defined functions, and queries via HTTP(S).	Enterprise Standard
Full-text searching	Words or phrases can be searched in any column defined for full-text searching.	Enterprise Standard Workgroup
Log shipping	Transaction logs can be moved from one database to another to allow the transactions.	Enterprise Standard Workgroup
SQL Server job scheduling	Jobs can be created and processed using specific scheduling requirements. Failures can also trigger notification by e-mail, pager, etc.	Enterprise Standard Workgroup
.NET integration	The .NET Framework CLR is hosted in the database, so assemblies can now be programmed in Visual Basic 2005 and C#, in the database.	All
Advanced auditing, authentication, and authorization	Windows authentication and authorization can be used for user logins.	All
Auto database tune	Databases can be tuned automatically.	All
Data import and export	Data can be imported and exported from external data sources, such as Excel.	All
Error handling, datatypes, and recursive queries	`TRY...CATCH` error handling, recursive queries, and new data types such as XML can be used.	All