

Microsoft®

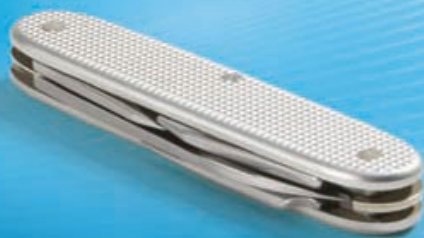
Updated for R2

Microsoft®

SQL

Server® 2008

2  
SECOND  
EDITION



William R. Stanek

*Author and Series Editor*

Administrator's  
Pocket Consultant

PUBLISHED BY  
Microsoft Press  
A Division of Microsoft Corporation  
One Microsoft Way  
Redmond, Washington 98052-6399

Copyright © 2010 by William Stanek

All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher.

Library of Congress Control Number: 2010923882

Printed and bound in the United States of America.

1 2 3 4 5 6 7 8 9 WCT 5 4 3 2 1 0

Distributed in Canada by H.B. Fenn and Company Ltd.

A CIP catalogue record for this book is available from the British Library.

Microsoft Press books are available through booksellers and distributors worldwide. For further information about international editions, contact your local Microsoft Corporation office or contact Microsoft Press International directly at fax (425) 936-7329. Visit our Web site at [www.microsoft.com/mspress](http://www.microsoft.com/mspress). Send comments to [mspinput@microsoft.com](mailto:mspinput@microsoft.com).

Microsoft, Microsoft Press, Access, Active Directory, ActiveX, Azure, BitLocker, Excel, Expression, MS, Outlook, SharePoint, SQL Server, Visual Basic, Visual Studio, Win32, Windows, Windows PowerShell, Windows Server, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious. No association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred.

This book expresses the author's views and opinions. The information contained in this book is provided without any express, statutory, or implied warranties. Neither the authors, Microsoft Corporation, nor its resellers, or distributors will be held liable for any damages caused or alleged to be caused either directly or indirectly by this book.

**Acquisitions Editor:** Ken Jones, Devon Musgrave

**Developmental Editor:** Karen Szall

**Project Editor:** Karen Szall

**Editorial Production:** John Pierce

**Technical Reviewer:** Rozanne Murphy Whalen; Technical Review services provided by Content Master, a member of CM Group, Ltd

**Cover:** Tom Draper Design

Body Part No. X16-84870

# Contents

	<i>Acknowledgments</i>	<i>xix</i>
	<i>Introduction</i>	<i>xxi</i>
<b>Chapter 1</b>	<b>SQL Server 2008 Administration Overview</b>	<b>1</b>
	SQL Server 2008 and Your Hardware . . . . .	2
	SQL Server 2008 Editions . . . . .	5
	SQL Server and Windows . . . . .	8
	Services for SQL Server	9
	SQL Server Logins and Authentication	10
	Service Accounts for SQL Server	10
	Using the Graphical Administration Tools . . . . .	12
	Using the Command-Line Tools . . . . .	16
	BCP	16
	SQLCMD	16
	Other Command-Line Tools	19
	Using SQL Server PowerShell . . . . .	21
	Running and Using Cmdlets	21
	Running and Using SQL Server PowerShell	22
	Working with SQL Server Cmdlets	23
<b>Chapter 2</b>	<b>Deploying SQL Server 2008</b>	<b>27</b>
	SQL Server Integration Roles . . . . .	27
	Using SQL Server Integration Services	27
	Using SQL Server 2008 for Relational Data Warehousing	28
	Using SQL Server 2008 for Multidimensional Databases and Data Mining	30
	Using SQL Server 2008 for Managed Reporting	32

---

## What do you think of this book? We want to hear from you!

Microsoft is interested in hearing your feedback so we can continually improve our books and learning resources for you. To participate in a brief online survey, please visit:

[microsoft.com/learning/booksurvey](http://microsoft.com/learning/booksurvey)

Planning for Your SQL Server 2008 Deployment . . . . .	33
Building the Server System for Performance . . . . .	33
Configuring the I/O Subsystem . . . . .	34
Ensuring Availability and Scalability . . . . .	36
Ensuring Connectivity and Data Access . . . . .	37
Managing SQL Server Configuration and Security . . . . .	39
Running and Modifying SQL Server Setup . . . . .	40
Creating New Instances of SQL Server . . . . .	41
Adding Components and Instances . . . . .	53
Repairing a SQL Server 2008 Installation . . . . .	54
Upgrading Your Edition of SQL Server 2008 . . . . .	54
Uninstalling SQL Server . . . . .	54

**Chapter 3 Managing the Surface Security, Access, and Network Configuration . . . . . 57**

Managing SQL Server Component Feature Access . . . . .	58
Configuring SQL Server Services . . . . .	62
Managing the Services Configuration . . . . .	63
Managing Service State and Start Mode . . . . .	67
Setting the Startup Service Account . . . . .	69
Configuring File Streaming . . . . .	70
Configuring Service Dump Directories, Error Reporting, and Customer Feedback Reporting . . . . .	73
Managing the Network and SQL Server Native Client Configuration . . . . .	75
Managing the Connections Configuration . . . . .	76
Specifying the Shared Memory Network Configuration . . . . .	77
Specifying the Named Pipes Network Configuration . . . . .	77
Specifying the TCP/IP Network Configuration . . . . .	78
Configuring Security for Native Client Configurations . . . . .	81
Configuring the Native Client Protocol Order . . . . .	81
Configuring the Shared Memory Native Client Configuration . . . . .	82
Configuring the Named Pipes Native Client Configuration . . . . .	82
Configuring the TCP/IP Native Client Configuration . . . . .	83

<b>Chapter 4</b>	<b>Configuring and Tuning SQL Server 2008</b>	<b>85</b>
	Accessing SQL Server Configuration Data . . . . .	86
	Working with the System Catalog and Catalog Views	87
	Working with System Stored Procedures	93
	Techniques for Managing SQL Server Configuration Options . . . . .	101
	Setting Configuration Options	101
	Working with SET Options	102
	Working with Server Options	105
	Working with Database Options	106
	Managing Database Compatibility	108
	Configuring SQL Server with Stored Procedures . . . . .	109
	Using SQL Server Management Studio for Queries	109
	Executing Queries and Changing Settings	110
	Checking and Setting Configuration Parameters	112
	Changing Settings with ALTER DATABASE	117
<b>Chapter 5</b>	<b>Managing the Enterprise</b>	<b>123</b>
	Using SQL Server Management Studio . . . . .	123
	Getting Started with SQL Server Management Studio	124
	Connecting to a Specific Server Instance	125
	Connecting to a Specific Database	126
	Managing SQL Server Groups . . . . .	127
	Introducing SQL Server Groups	127
	Creating a Server Group	128
	Deleting a Server Group	129
	Editing and Moving Server Groups	129
	Adding SQL Servers to a Group	130
	Managing Servers . . . . .	130
	Registering a Connected Server	131
	Registering a New Server in the Registered Servers View	132
	Registering Previously Registered SQL Server 2000 Servers	133
	Updating Registration for Local Servers	133

Copying Server Groups and Registration Details from One Computer to Another	133
Editing Registration Properties	136
Connecting to a Server	136
Disconnecting from a Server	136
Moving a Server to a New Group	136
Deleting a Server Registration	137
Using Windows PowerShell for SQL Server Management . . . . .	137
Starting, Stopping, and Configuring SQL Server Agent . . . . .	142
Starting, Stopping, and Configuring Microsoft Distributed Transaction Coordinator . . . . .	142
Managing SQL Server Startup . . . . .	143
Enabling or Preventing Automatic SQL Server Startup	143
Setting Database Engine Startup Parameters	145
Managing Services from the Command Line	148
Managing the SQL Server Command-Line Executable	149
Managing Server Activity . . . . .	150
Examining Process Information	151
Tracking Resource Waits and Blocks	153
Troubleshooting Deadlocks and Blocking Connections	157
Tracking Command Execution in SQL Server	159
Killing Server Processes	159
<b>Chapter 6 Implementing Policy-Based Management</b>	<b>161</b>
Introducing Policy-Based Management . . . . .	161
Working with Policy-Based Management . . . . .	164
Managing Policies Throughout the Enterprise . . . . .	171
Importing and Exporting Policies	171
Configuring Central Management Servers	172
Executing Statements Against Multiple Servers	176
Configuring and Managing Policy Facets	176
Creating and Managing Policy Conditions	178
Creating and Managing Policies	181
Managing Policy Categories and Mandating Policies	183

Evaluating Policies	185
Troubleshooting Policy-Based Management Policies	188
<b>Chapter 7 Configuring SQL Server with SQL Server Management Studio</b>	<b>191</b>
SQL Server Management Studio Essentials . . . . .	192
Managing the Configuration with SQL Server Management Studio	192
Determining System and Server Information	194
Configuring Utility Control Points	194
Configuring Authentication and Auditing . . . . .	200
Setting the Authentication Mode	200
Setting the Auditing Level	201
Enabling or Disabling C2 Audit Logging	201
Enabling or Disabling Common Criteria Compliance	202
Tuning Memory Usage . . . . .	203
Working with Dynamically Configured Memory	205
Using Fixed Memory	206
Enabling AWE Memory Support	206
Optimizing Memory for Indexing	208
Allocating Memory for Queries	208
Configuring Processors and Parallel Processing . . . . .	210
Optimizing CPU Usage	210
Setting Parallel Processing	213
Configuring Threading, Priority, and Fibers . . . . .	214
Configuring User and Remote Connections . . . . .	216
Setting Maximum User Connections	216
Setting Default Connection Options	217
Configuring Remote Server Connections	220
Managing Server Settings. . . . .	221
Enabling or Disabling File Streaming Support	221
Setting the Default Language for SQL Server	222
Allowing and Disallowing Nested Triggers	223
Controlling Query Execution	223
Configuring Year 2000 Support	224

Managing Database Settings . . . . .	224
Setting the Index Fill . . . . .	225
Configuring Backup and Restore Time-Out Options . . . . .	226
Configuring Backup and Restore Retention Options . . . . .	227
Flushing the Cache with Checkpoints . . . . .	227
Compressing the Backup Media . . . . .	228
Adding and Removing Active Directory Information . . . . .	228
Troubleshooting Configuration Problems . . . . .	228
Recovering from a Bad Configuration . . . . .	228
Changing Collation and Rebuilding the <i>master</i> Database . . . . .	230
<b>Chapter 8 Core Database Administration</b> . . . . .	<b>233</b>
Database Files and Logs . . . . .	233
Database Administration Basics . . . . .	238
Viewing Database Information in SQL Server Management Studio . . . . .	239
Viewing Database Information Using T-SQL . . . . .	240
Checking System and Sample Databases . . . . .	241
Examining Database Objects . . . . .	242
Creating Databases . . . . .	245
Creating Databases in SQL Server Management Studio . . . . .	245
Creating Databases Using T-SQL . . . . .	249
Altering Databases and Their Options . . . . .	251
Setting Database Options in SQL Server Management Studio . . . . .	251
Modifying Databases Using ALTER DATABASE . . . . .	252
Configuring Automatic Options . . . . .	257
Controlling ANSI Compliance at the Database Level . . . . .	259
Configuring Parameterization . . . . .	261
Configuring Cursor Options . . . . .	263
Controlling User Access and Database State . . . . .	264
Setting Online, Offline, or Emergency Mode . . . . .	266
Managing Cross-Database Chaining and External Access Options . . . . .	266



Configuring Recovery, Logging, and Disk I/O Error Checking Options	267
Viewing, Changing, and Overriding Database Options	269
Managing Database and Log Size. . . . .	270
Configuring SQL Server to Automatically Manage File Size	270
Expanding Databases and Logs Manually	271
Compressing and Shrinking a Database Manually	271
Manipulating Databases . . . . .	275
Renaming a Database	275
Dropping and Deleting a Database	276
Attaching and Detaching Databases	277
Tips and Techniques. . . . .	280
Copying and Moving Databases	280
Moving Databases	284
Moving and Resizing <i>tempdb</i>	285
Creating Secondary Data and Log Files	287
Preventing Transaction Log Errors	288
Preventing a Filegroup Is Full Error	288
Creating a New Database Template	288
Configuring Database Encryption	288
<b>Chapter 9 Managing SQL Server 2008 Security</b>	<b>291</b>
Overview of SQL Server 2008 Security. . . . .	292
Working with Security Principals and Securables	292
Understanding Permissions of Securables	295
Examining Permissions Granted to Securables	297
SQL Server 2008 Authentication Modes . . . . .	301
Windows Authentication	301
Mixed Security and SQL Server Logins	302
Special-Purpose Logins and Users . . . . .	302
Working with the Administrators Group	303
Working with the Administrator User Account	303
Working with the sa Login	303
Working with the NETWORK SERVICE and SYSTEM Logins	304

Working with the Guest User	304
Working with the dbo User	305
Working with the sys and INFORMATION_SCHEMA Users	305
Permissions .....	305
Object Permissions	306
Statement Permissions .....	311
Implied Permissions	312
Roles .....	312
Server Roles	313
Database Roles	314
Managing Server Logins .....	317
Viewing and Editing Existing Logins	317
Creating Logins	319
Editing Logins with T-SQL	321
Granting or Denying Server Access	323
Enabling, Disabling, and Unlocking Logins	324
Removing Logins	325
Changing Passwords	326
Configuring Server Roles .....	326
Assigning Roles by Login	326
Assigning Roles to Multiple Logins	328
Revoking Access Rights and Roles by Server Login	329
Controlling Database Access and Administration .....	329
Assigning Access and Roles by Login	329
Assigning Roles for Multiple Logins	330
Creating Standard Database Roles	331
Creating Application Database Roles	333
Removing Role Memberships for Database Users	334
Deleting User-Defined Roles	334
T-SQL Commands for Managing Access and Roles	335
Managing Database Permissions .....	336
Assigning Database Permissions for Statements	336
Object Permissions by Login	342
Object Permissions for Multiple Logins	344

<b>Chapter 10 Manipulating Schemas, Tables, Indexes, and Views</b>	<b>347</b>
Working with Schemas . . . . .	348
Creating Schemas	349
Modifying Schemas	350
Moving Objects to a New Schema	351
Dropping Schemas	353
Getting Started with Tables . . . . .	353
Table Essentials . . . . .	354
Understanding Data Pages	354
Understanding Extents	357
Understanding Table Partitions	357
Working with Tables. . . . .	358
Creating Tables	358
Modifying Existing Tables	364
Viewing Table Row and Size Information	367
Displaying Table Properties and Permissions	367
Displaying Current Values in Tables	368
Copying Tables	368
Renaming and Deleting Tables	369
Adding and Removing Columns in a Table	370
Scripting Tables	371
Managing Table Values . . . . .	371
Using Native Data Types	371
Using Fixed-Length, Variable-Length, and Max-Length Fields	376
Using User-Defined Data Types	377
Allowing and Disallowing Nulls	379
Using Default Values	380
Using Sparse Columns	380
Using Identities and Globally Unique Identifiers	381
Using User-Defined Table Types	383
Using Views . . . . .	386
Working with Views	387
Creating Views	388

Modifying Views	392
Using Updatable Views	393
Managing Views	393
Creating and Managing Indexes . . . . .	394
Understanding Indexes	394
Using Clustered Indexes	396
Using Nonclustered Indexes	397
Using XML Indexes	397
Using Filtered Indexes	397
Determining Which Columns Should Be Indexed	398
Indexing Computed Columns and Views	399
Viewing Index Properties	400
Creating Indexes	402
Managing Indexes	407
Using the Database Engine Tuning Advisor	411
Column Constraints and Rules . . . . .	415
Using Constraints	415
Using Rules	420
Creating Partitioned Tables and Indexes . . . . .	421
Creating Partition Functions	421
Creating Partition Schemes	422
Creating Partitions	423
Viewing and Managing Partitions	424
Compressing Tables, Indexes, and Partitions . . . . .	425
Using Row and Page Compression	426
Setting or Changing Compression Settings	427

**Chapter 11 Importing, Exporting, and Transforming Data 429**

Working with Integration Services . . . . .	429
Getting Started with Integration Services	430
Integration Services Tools	431
Integration Services and Data Providers	433
Integration Services Packages	433

Creating Packages with the SQL Server Import And Export Wizard .....	434
Stage 1: Source and Destination Configuration	435
Stage 2: Copy or Query	443
Stage 3: Formatting and Transformation	447
Stage 4: Save and Execute	450
Understanding BCP .....	453
BCP Basics	454
BCP Syntax	454
BCP Permissions and Modes	457
Importing Data with BCP	458
Exporting Data with BCP	460
BCP Scripts .....	460
Using the BULK INSERT Command .....	461
<b>Chapter 12 Linked Servers and Distributed Transactions</b>	<b>463</b>
Working with Linked Servers and Distributed Data .....	463
Using Distributed Queries	464
Using Distributed Transactions	466
Running the Distributed Transaction Coordinator Service	468
Managing Linked Servers .....	469
Adding Linked Servers	469
Configuring Security for Linked Servers	473
Setting Server Options for Remote and Linked Servers	475
Deleting Linked Servers	477
<b>Chapter 13 Implementing Snapshot, Merge, and Transactional Replication</b>	<b>479</b>
An Overview of Replication .....	479
Replication Components	480
Replication Agents and Jobs	481
Replication Variants	483

Planning for Replication . . . . .	485
Replication Models . . . . .	486
Preliminary Replication Tasks . . . . .	487
Distributor Administration . . . . .	490
Setting Up a New Distributor . . . . .	490
Updating Distributors . . . . .	495
Creating Distribution Databases . . . . .	497
Enabling and Updating Publishers . . . . .	497
Enabling Publication Databases . . . . .	498
Deleting Distribution Databases . . . . .	499
Disabling Publishing and Distribution . . . . .	499
Creating and Managing Publications . . . . .	499
Creating Publications . . . . .	500
Viewing and Updating Publications . . . . .	508
Setting Publication Properties . . . . .	508
Setting Agent Security and Process Accounts . . . . .	510
Controlling Subscription Access to a Publication . . . . .	511
Creating a Script for a Publication . . . . .	511
Deleting a Publication . . . . .	512
Subscribing to a Publication . . . . .	512
Subscription Essentials . . . . .	512
Creating Subscriptions . . . . .	513
Viewing Subscription Properties . . . . .	518
Updating, Maintaining, and Deleting Subscriptions . . . . .	518
Validating Subscriptions . . . . .	518
Reinitializing Subscriptions . . . . .	519
<b>Chapter 14 Profiling and Monitoring SQL Server 2008 . . . . .</b>	<b>521</b>
Monitoring Server Performance and Activity . . . . .	521
Reasons to Monitor SQL Server . . . . .	521
Getting Ready to Monitor . . . . .	522
Monitoring Tools and Resources . . . . .	523
Working with Replication Monitor . . . . .	525
Starting and Using Replication Monitor . . . . .	526
Adding Publishers and Publisher Groups . . . . .	526

Working with the Event Logs . . . . .	528
Examining the Application Log . . . . .	530
Examining the SQL Server Event Logs . . . . .	532
Examining the SQL Server Agent Event Logs . . . . .	534
Monitoring SQL Server Performance . . . . .	535
Choosing Counters to Monitor . . . . .	535
Performance Logging . . . . .	538
Viewing Data Collector Reports . . . . .	542
Configuring Performance Counter Alerts . . . . .	543
Configuring a Management Data Warehouse . . . . .	544
Understanding Management Data Warehouses . . . . .	544
Creating the Management Data Warehouse . . . . .	545
Setting Up Data Collection . . . . .	545
Managing Collection and Generating Reports . . . . .	546
Solving Performance Problems with Profiler. . . . .	546
Using Profiler . . . . .	546
Creating New Traces . . . . .	548
Working with Traces . . . . .	551
Saving a Trace . . . . .	551
Replaying a Trace . . . . .	551
<b>Chapter 15 Backing Up and Recovering SQL Server 2008</b> . . . . .	<b>557</b>
Creating a Backup and Recovery Plan . . . . .	557
Initial Backup and Recovery Planning . . . . .	558
Planning for Mirroring and Mirrored Database Backups . . . . .	562
Planning for Backups of Replicated Databases . . . . .	563
Planning for Backups of Very Large Databases . . . . .	564
Planning for Backup Compression . . . . .	565
Selecting Backup Devices and Media . . . . .	566
Using Backup Strategies . . . . .	568
Creating a Backup Device . . . . .	570
Performing Backups . . . . .	572
Creating Backups in SQL Server Management Studio . . . . .	572
Using Striped Backups with Multiple Devices . . . . .	577

Using Transact-SQL Backup	578
Performing Transaction Log Backups	582
Restoring a Database . . . . .	583
Database Corruption and Problem Resolution	584
Restoring a Database from a Normal Backup	586
Restoring Files and Filegroups	591
Restoring a Database to a Different Location	593
Recovering Missing Data	594
Creating Standby Servers	594
Using Transact-SQL Restore Commands	596
Restoring the <i>master</i> Database . . . . .	601

## **Chapter 16 Database Automation and Maintenance 603**

Overview of Database Automation and Maintenance . . . . .	604
Using Database Mail . . . . .	606
Performing the Initial Database Mail Configuration	606
Managing Database Mail Profiles and Accounts	611
Viewing or Changing Database Mail System Parameters	612
Using SQL Server Agent . . . . .	612
Accessing Alerts, Operators, and Jobs	613
Configuring the SQL Server Agent Service	613
Setting the SQL Server Agent Mail Profile	614
Using SQL Server Agent to Restart Services Automatically	615
Managing Alerts . . . . .	615
Using Default Alerts	615
Creating Error Message Alerts	616
Handling Alert Responses	617
Deleting, Enabling, and Disabling Alerts	618
Managing Operators . . . . .	619
Registering Operators	619
Deleting and Disabling Notification for Operators	620
Configuring a Fail-Safe Operator	620



Scheduling Jobs .....	621
Creating Jobs	621
Assigning or Changing Job Definitions	622
Setting Steps to Execute	623
Configuring Job Schedules	627
Handling Job Alerts	629
Handling Notification Messages	630
Managing Existing Jobs	631
Managing Job Categories	631
Automating Routine Server-to-Server Administration Tasks .....	632
Copying User Accounts, Tables, Views, and Other Objects from One Database to Another	633
Copying Alerts, Operators, and Scheduled Jobs from One Server to Another	636
Multiserver Administration .....	637
Event Forwarding	637
Multiserver Job Scheduling	638
Database Maintenance .....	641
Database Maintenance Checklists	641
Using Maintenance Plans	642
Checking and Maintaining Database Integrity	648
<b>Chapter 17 Managing Log Shipping and Database Mirroring</b>	<b>653</b>
Log Shipping .....	653
Log Shipping: How It Works	653
Preparing for Log Shipping	655
Upgrading SQL Server 2000 Log Shipping to SQL Server 2008 Log Shipping	656
Enabling Log Shipping on the Primary Database	657
Adding Log Shipping Secondary Databases	661
Changing the Transaction Log Backup Interval	664
Changing the Copy and Restore Intervals	664
Monitoring Log Shipping	664
Failing Over to a Secondary Database	665
Disabling and Removing Log Shipping	668

Database Mirroring .....	668
Database Mirroring Essentials	668
Configuring Database Mirroring	670
Managing and Monitoring Mirroring	675
Recovering by Using Failover	678
Removing Database Mirroring	680
Using Mirroring and Log Shipping .....	681
<i>Index</i>	683

---

**What do you think of this book? We want to hear from you!**

Microsoft is interested in hearing your feedback so we can continually improve our books and learning resources for you. To participate in a brief online survey, please visit:

[microsoft.com/learning/booksurvey](https://microsoft.com/learning/booksurvey)

# Acknowledgments

---

You know you've been at this thing called writing for a long time when people ask how many books you've written and you just have no idea. For many years, my bio stated that I was the author of more than 25 books. Several times my publishers have asked me to update the bio with a more precise number, so at around number 61 I started counting to keep everyone happy. That was a few years ago; I'm now somewhere over 100, depending on how you count. ;-)

For me, it's always been about the craft of writing. I love writing, and I love challenging projects most of all. Some of the most challenging books I ever wrote were the *SQL Server 7.0 Administrator's Pocket Consultant*, published in 1999, the *SQL Server 2000 Administrator's Pocket Consultant*, published in 2000, and the *SQL Server 2005 Administrator's Pocket Consultant*, published in 2005. Writing *SQL Server 2008 Administrator's Pocket Consultant* and updating it for a second edition was no less demanding.

The challenge in writing a day-to-day administrator's guide to SQL Server is that there's so much I'd like to cover, but pocket consultants aren't meant to be all-in-one references. Pocket Consultants are meant to be portable and readable—the kind of book you use to solve problems and get the job done wherever you might be. With that in mind, I have to continually make sure I focus on the core of SQL Server 2008 administration. The result is the book you hold in your hand, which I hope you'll agree is one of the best practical, portable guides to SQL Server 2008 and SQL Server 2008 R2.

As I've stated in the three dozen or so pocket consultants I've written, the team at Microsoft Press is top-notch. Karen Szall was instrumental throughout the writing process. She was my primary contact at Microsoft. Ken Jones was the acquisitions editor for the project. He believed in the book from the beginning and was really great to work with. Completing and publishing the book wouldn't have been possible without their help! Thank you so much!

Unfortunately for the writer (but fortunately for readers), writing is only one part of the publishing process. Next came editing and author review. I must say, Microsoft Press has the most thorough editorial and technical review process I've seen anywhere—and I've written a lot of books for many different publishers. Rozanne Whalen was the technical reviewer for the book, John Pierce was the project manager and copy editor, Curtis Philips prepared pages, and Andrea Fox was the proofreader.

I hope I haven't forgotten anyone, but if I have, it was an oversight. Honest. ;-)



# Introduction

---

*Microsoft SQL Server 2008 Administrator's Pocket Consultant, Second Edition*, is designed to be a concise and compulsively usable resource for SQL Server 2008 and SQL Server 2008 Release 2 (R2) administrators. It covers everything you need to know to perform the core administrative tasks for SQL Server and is the readable resource guide that you'll want on your desk at all times. Because the focus is on giving you maximum value in a pocket-sized guide, you don't have to wade through hundreds of pages of extraneous information to find what you're looking for. Instead, you'll find exactly what you need to get the job done.

This book is designed to be the one resource you turn to whenever you have questions about SQL Server administration. To this end, the book zeroes in on daily administration procedures, frequently used tasks, documented examples, and options that are representative while not necessarily inclusive. One of the key goals is to keep content concise enough that the book is compact and easy to navigate while also ensuring that the book contains as much information as possible. Instead of a 1,000-page tome or a 100-page quick reference, you get a valuable resource guide that can help you quickly and easily perform common tasks, solve problems, and implement advanced SQL Server technologies such as replication, distributed queries, and multiserver administration.

## Who Is This Book For?

---

*Microsoft SQL Server 2008 Administrator's Pocket Consultant, Second Edition*, covers the Standard, Enterprise, Developer, Web, and Workgroup editions of SQL Server. The book is designed for:

- Current SQL Server database administrators
- Accomplished users who have some administrator responsibilities
- Administrators migrating to SQL Server 2008 and SQL Server 2008 R2 from previous versions
- Administrators transitioning from other database architectures

To include as much information as possible, I had to assume that you have basic networking skills and a basic understanding of SQL Server. With this in mind, I don't devote entire chapters to understanding SQL Server architecture or running simple SQL queries. But I do cover SQL Server installation, configuration, enterprise-wide server management, performance tuning, optimization, maintenance, and much more.

I also assume that you're fairly familiar with SQL commands and stored procedures as well as the standard Windows user interface. If you need help learning SQL basics, you should read other resources (many of which are available from Microsoft Press).

# How Is This Book Organized?

---

*Microsoft SQL Server 2008 Administrator's Pocket Consultant*, Second Edition, is designed to be used in the daily administration of SQL Server, and as such the book is organized according to job-related tasks rather than SQL Server features. Before you use this book, you should be aware of the difference between the Administrator's Pocket Consultant titles and the Administrator's Companion titles. Both types of books are designed to be a part of an overall administrator's library, but books in the Administrator's Pocket Consultant series are the down-and-dirty, in-the-trenches books, and Administrator's Companion titles are the comprehensive tutorials and references that cover every aspect of deploying a product or technology in the enterprise.

Speed and ease of reference are essential parts of this hands-on guide. The book has an expanded table of contents and an extensive index for finding answers to problems quickly. Many other quick reference features have been added to the book as well. These features include quick step-by-step procedures, lists, tables with fast facts, and cross-references.

The first several chapters cover the fundamental tasks you need for SQL Server administration. Chapter 1 provides an overview of SQL Server administration tools, techniques, and concepts. Chapter 2 discusses deploying SQL Server. Chapter 3 shows you how to configure SQL Server's services, components, and networking capabilities. Chapter 4 examines the structures available for configuring and tuning SQL Server. The chapter looks at SQL Server's system catalog and then continues with a discussion of catalog queries and stored procedures.

In Chapters 5 through 9, I describe the essential tasks for administering SQL Server. Chapter 5 details management techniques for server groups and servers in the enterprise. In Chapter 6, you'll learn about essential tasks for implementing Policy-Based Management. Policy-Based Management defines policies that not only configure SQL Server settings but also control whether and how settings can be changed. Before you deploy SQL Server, you should familiarize yourself with the available options and determine whether your servers will be put into specific management groups. Chapter 7 explores configuring and tuning SQL Server using SQL Server Management Studio. You'll learn about optimizing memory usage, parallel processing, authentication, auditing, and more. The core administration tasks for creating and managing databases are covered in Chapter 8. Chapter 9 focuses on SQL Server security. To manage server security, you'll create user logins, configure login permissions, and assign roles. The permissions and roles you assign determine the actions users can perform as well as what types of data they can access.

Data administration is the subject of the next set of chapters. Chapter 10 covers techniques for creating, managing, and optimizing schemas, tables, indexes, and views. In Chapter 11, you'll find tasks for importing and exporting data, as well as the old standby bulk copy program (BCP). Chapter 12 focuses on integrating SQL Server databases with other SQL Server databases and with other data sources. You'll find detailed discussions on distributed queries, distributed transactions,

Microsoft Distributed Transaction Coordinator (MS DTC), and linked servers. Chapter 13 explores data replication. You'll learn all about the latest replication techniques, including merge replication and immediate-updating subscribers.

In the last four chapters I cover administration tasks you'll use to enhance and maintain SQL Server. Chapter 14 provides the essentials for working with server logs, monitoring SQL Server performance, and solving performance problems. Chapter 15 starts by explaining how to create a backup and recovery plan. The chapter then dives into common tasks for creating and restoring backups. Chapter 16 explores database automation and maintenance, showing you how to create alerts, schedule jobs, handle operator notifications, and more. You'll also learn how to create maintenance plans and resolve database consistency problems. Chapter 17 begins with a discussion about log shipping and database mirroring. Log shipping and database mirroring are disaster recovery solutions designed to help improve availability and quickly recover SQL Server.

## Conventions Used in This Book

---

I've used a variety of elements to help keep the text clear and easy to follow. You'll find code terms and listings in monospace type, except when I tell you to actually type a command. In that case, the command appears in **bold** type. When I introduce and define a new term, I put it in *italics*.

Other conventions include the following:

- **Best Practices** To examine the best technique to use when working with advanced configuration and administration concepts
- **Cautions** To warn you about potential problems you should look out for
- **More Info** To provide more information on a subject
- **Notes** To provide additional details on a particular point that needs emphasis
- **Real World** To provide real-world advice when discussing advanced topics
- **Security Alerts** To point out important security issues
- **Tips** To offer helpful hints or additional information

I truly hope you find that *Microsoft SQL Server 2008 Administrator's Pocket Consultant*, Second Edition, provides everything you need to perform the essential administrative tasks for SQL Server as quickly and efficiently as possible. You are welcome to send your thoughts to me at [williamstaneke@aol.com](mailto:williamstaneke@aol.com) or follow me at [www.twitter.com/WilliamStaneke](http://www.twitter.com/WilliamStaneke). Thank you.

## Other Resources

---

No single magic bullet for learning everything you'll ever need to know about SQL Server 2008 exists. While some books are offered as all-in-one guides, there's simply no way one book can do it all. With this in mind, I hope you use this book as it is

intended to be used—as a concise and easy-to-use resource. It covers everything you need to perform core administration tasks for SQL Server, but it is by no means exhaustive.

Your current knowledge will largely determine your success with this or any other SQL Server resource or book. As you encounter new topics, take the time to practice what you've learned and read about. Seek out further information as necessary to get the practical hands-on know-how and knowledge you need.

I recommend that you regularly visit the SQL Server site ([www.microsoft.com/sqlserver/](http://www.microsoft.com/sqlserver/)) and Microsoft's support site ([www.support.microsoft.com](http://www.support.microsoft.com)) to stay current with the latest changes. To help you get the most out of this book, you can visit my corresponding Web site at [www.williamstaneck.com/sqlserver](http://www.williamstaneck.com/sqlserver). This site contains information about SQL Server 2008 and updates to the book.

## Support for This Book

---

Every effort has been made to ensure the accuracy of this book. As corrections or changes are collected, they will be added to a Microsoft Knowledge Base article accessible via the Microsoft Help and Support site. Microsoft Press provides support for books, including instructions for finding Knowledge Base articles, at the following Web site:

[www.microsoft.com/learning/support/books/](http://www.microsoft.com/learning/support/books/)

If you have questions regarding the book that are not answered by visiting this site or viewing a Knowledge Base article, send them to Microsoft Press via e-mail to [mspinput@microsoft.com](mailto:mspinput@microsoft.com).

Please note that Microsoft software product support is not offered through these addresses.

## We Want to Hear from You

---

We welcome your feedback about this book. Please share your comments and ideas via the following short survey:

[www.microsoft.com/learning/booksurvey](http://www.microsoft.com/learning/booksurvey)

Your participation will help Microsoft Press create books that better meet your needs and your standards.

**NOTE** We hope that you will give us detailed feedback via our survey. If you have questions about our publishing program, upcoming titles, or Microsoft Press in general, we encourage you to interact with us via Twitter at [www.twitter.com/MicrosoftPress](http://www.twitter.com/MicrosoftPress). For support issues, use only the e-mail address shown above.



## CHAPTER 1

# SQL Server 2008 Administration Overview

- SQL Server 2008 and Your Hardware 2
- SQL Server 2008 Editions 5
- SQL Server and Windows 8
- Using the Graphical Administration Tools 12
- Using the Command-Line Tools 16
- Using SQL Server PowerShell 21

Microsoft SQL Server 2008 was released in 2008, and SQL Server 2008 Release 2 (R2) followed in 2010. For ease of reference in this book, I refer to both SQL Server 2008 and SQL Server 2008 R2 as SQL Server or SQL Server 2008. When I need to differentiate between the two releases, I'll indicate that Release 1 (R1) does this but R2 does that. Sometimes, I'll simply state the additional options or features that R2 provides.

SQL Server 2008 builds on SQL Server 2005 and finally makes Microsoft's vision of SQL Server as an end-to-end data platform a reality. SQL Server 2008 R2 is an incremental release that includes some additional functionality and performance enhancements. By functioning as a mission-critical data platform, allowing dynamic development, providing extensive business intelligence, and going beyond relational data, SQL Server 2008 provides the bedrock foundation on which small, medium, and large organizations can build their next generation IT infrastructure. At the core of SQL Server 2008 and R2, you will find the following:

- **Database Engine Services** Includes the core database, notification, replication, and full-text search components. The core database—also known as the Database Engine—is the heart of SQL Server. Replication increases data availability by distributing data across multiple databases, allowing you to scale out the read workload across designated database servers. Full-text search allows plain-language queries on data stored in SQL Server tables.

- **Analysis Services** Delivers online analytical processing (OLAP) and data-mining functionality for business intelligence applications. Analysis Services enables your organization to aggregate data from multiple data sources, such as relational databases, and work with this data in a wide variety of ways.
- **Integration Services** Provides an enterprise data transformation and integration solution for extracting and transforming data from multiple data sources and moving it to one or more destination data sources. This functionality allows you to merge data from heterogeneous data sources, load data into data warehouses and data marts, and more.
- **Reporting Services** Includes Report Manager and Report Server, which provide a complete server-based platform for creating, managing, and distributing reports. Report Server is built on standard Internet Information Services (IIS) and .NET Framework technology, allowing you to combine the benefits of SQL Server and IIS to host and process reports.
- **Service Broker** Provides reliable queuing and messaging as a central part of the database. Queues can be used to stack work such as queries and other requests and perform the work as resources allow. Messaging allows database applications to communicate with each other. The Database Engine uses Service Broker to deliver notification messages. This notification functionality doesn't require or use Notification Services, which has been removed from SQL Server 2008.
- **Sync Framework** Enables data synchronization for collaboration and offline use. Developers can use Sync Framework to synchronize databases and other types of data stores as well as files, folders, and metadata.

As you get started with SQL Server 2008, you should concentrate on these areas:

- How SQL Server 2008 works with your hardware
- What versions and editions of SQL Server 2008 are available and how they meet your needs
- How SQL Server 2008 works with Windows operating systems
- What administration tools are available

## SQL Server 2008 and Your Hardware

---

Successful database server administration depends on three things:

- Knowledgeable database administrators
- Strong database architecture
- Appropriate hardware

The first two ingredients are covered: you're the administrator, you're smart enough to buy this book to help you through the rough spots, and you've implemented SQL Server 2008 to provide your high-performance database needs. This

brings us to the issue of hardware. You should run SQL Server 2008 on a system with adequate memory, processing speed, and disk space. You also need an appropriate data and system protection plan at the hardware level.

**NOTE** Well-written database applications and proper database design make a database administrator's job much easier. Poor performance is caused more often by poor application and data-structure design than by anything a database administrator can remedy. In a way, overall design is a fourth ingredient for success—but it's an ingredient that's largely beyond your control as a database administrator.

The hardware guidelines for SQL Server 2008 R1 and R2 are the same. Key guidelines for choosing hardware for SQL Server are as follows:

- **Memory** All editions of SQL Server 2008 except for Express editions require a minimum of 512 megabytes (MB) of RAM and have a recommended level of RAM of 2,048 MB. In most cases, you want to have at least twice the recommended minimum amount of memory. The primary reason for having extra memory is performance. SQL Server 2008 and standard Windows services together use about 512 MB of memory as a baseline on Windows Server 2003 and 1 GB of memory as a baseline on Windows Server 2008 and Windows Server 2008 R2.

Additional database features—such as Analysis Services, Reporting Services, and Integration Services—increase the baseline memory requirements (by about 30 MB of RAM each). Running SQL Server Management Studio on a server with SQL Server uses 50 MB to 60 MB of RAM as a baseline. Also consider the number of user connections. Each user connection consumes about 24 KB. Data requests and other SQL Server processes use memory as well, and this memory usage is in addition to all other processes and applications running on the server.

- **CPU** The 32-bit versions of SQL Server 2008 run on Intel x86 or compatible hardware. The 64-bit versions run on Intel Itanium (IA64) and the X64 family of processors from AMD and Intel, including AMD64 and Intel Extended Memory 64 Technology (Intel EM64T). SQL Server provides solid benchmark performance with Intel Xeon 3.4 GHz and higher processors or AMD Opteron 3.1 GHz and higher processors. Any of these CPUs provide good starting points for the average SQL Server system. You can achieve significant performance improvements with a high-level on-processor cache. Look closely at the L1, L2, and L3 cache options available—a higher cache can yield much better performance overall.

The primary advantages of 64-bit processors over 32-bit processors are related to memory limitations and data access. Because 64-bit processors can exceed the 4 gigabyte (GB) memory limit of 32-bit processors, they can store greater amounts of data in main memory, providing direct access to and faster processing of data. In addition, 64-bit processors can process data and execute instruction sets that are twice as large as those that 32-bit

processors can handle. Accessing 64 bits of data (versus 32 bits) offers a significant advantage when processing complex calculations that require a high level of precision. However, not all applications are optimized for 64-bit processors, and this can present an implementation and maintenance challenge.

- **Symmetric multiprocessing (SMP)** SQL Server 2008 supports symmetric multiprocessors and can process complex parallel queries. Parallel queries are valuable only when relatively few users are on a system and the system is processing large queries. On a dedicated system that runs only SQL Server and supports fewer than 100 simultaneous users who aren't running complex queries, a single CPU should suffice. If the server supports more than 100 users or doesn't run on a dedicated system, you might consider adding processors (or using a system that can support additional processors as your needs grow). Keep in mind that the size of the queries and data sets being processed affects how well SQL Server scales. As the size of jobs being processed increases, you have increased memory and CPU needs.
- **Disk drives** The amount of data storage capacity you need depends entirely on the number and size of the databases that the server supports. You need enough disk space to store all your data plus work space, indices, system files, virtual memory, and transaction logs. For log shipping and mirroring, you need space for the backup share and, in the case of a cluster, the quorum disk. I/O throughput is just as important as drive capacity. For the best I/O performance, FC (Fiber Channel) is the recommended choice for high-end storage solutions. Instead of using a single large drive, you should use several smaller drives, which allows you to configure fault tolerance with RAID (redundant array of independent disks). I recommend separating data and logs and placing them on separate spindles. This includes the backup share for log shipping and the quorum disk for clustering.
- **Data protection** You should add protection against unexpected drive failure by using RAID. For data, use RAID 0 + 1 or RAID 5. For logs, use RAID 1. RAID 0 (disk striping without parity) offers good read/write performance, but the effect of any failed drive is that SQL Server can't continue operation on an affected database until the drive is replaced and data is restored from backup. RAID 1 (disk mirroring) creates duplicate copies of data on separate drives, and you can rebuild the RAID unit to restore full operations. RAID 5 (disk striping with parity) offers good protection against single drive failure but has poor write performance. For best performance and fault tolerance, RAID 0 + 1 is recommended. This configuration consists of disk mirroring and disk striping without parity.
- **Uninterruptible power supply (UPS)** SQL Server is designed to maintain database integrity at all times and can recover information by using transaction logs. However, this does not protect the server hardware from sudden power loss or power spikes. Both of these events can seriously damage hardware. To prevent this, get an uninterruptible power supply that conditions

the power. A UPS system gives you time to shut down the system properly in the event of a power outage, and it is also important in maintaining database integrity when the server uses write-back caching controllers.

If you follow these hardware guidelines, you will be well on your way to success with SQL Server 2008.

## SQL Server 2008 Editions

---

SQL Server 2008 is distributed in four main editions: Workgroup, Standard, Enterprise, and Developer. SQL Server 2008 R2 adds two premium editions: R2 Data-center and R2 Parallel Data Warehouse. In all these editions, you will find instance features and shared features for installation. The instance features include the full version of SQL Server and support services. The shared features include Sync Framework and Integration Services, as well as the client tools, development tools, and documentation required for working with SQL Server from a workstation.

The Workgroup edition is designed as an entry-level database solution. This edition is ideal for small departments in large enterprises and for small businesses that need a robust database solution but do not need the extended business intelligence features of the Standard or Enterprise edition. The Workgroup edition has the following features:

- Runs on multiple versions of the Windows operating system, including desktop and server versions.

**NOTE** For all editions of SQL Server 2008 running on Windows Server 2003 and Windows XP Professional, Service Pack 2 (SP2) or later must be installed. For additional requirements pertaining to the operation of SQL Server 2008 on Windows Server 2003 and Windows XP Professional, refer to SQL Server 2008 Books Online.

- Supports an unlimited database size, up to the operating system maximum for RAM; two CPUs for symmetric multiprocessing; limited replication publishing; and full-text search.
- Enables log shipping, which allows SQL Server to send transaction logs from one server to another. Use this feature to create a standby server.

The most widely deployed edition is the Standard edition, which is designed for an average-size organization. The Standard edition has the following features:

- Runs on multiple versions of Windows, including desktop and server versions.
- Supports an unlimited database size, an unlimited amount of RAM, four CPUs for symmetric multiprocessing, full replication publishing, and full-text search.
- Provides basic features for Analysis Services, Reporting Services, and Integration Services.

- Includes database mirroring (single-threaded), log stream compression, and two-node failover clustering.
- Supports limited data encryption and auditing features.

In R2, the Standard edition now allows installation as a managed instance for application and multiserver management capabilities. As discussed in Chapter 7, “Configuring SQL Server with SQL Server Management Studio,” this allows a Standard edition instance of the Database Engine to be registered with and managed by a Utility Control Point (UCP) and to be configured as a data-tier application (DAC). UCPs work in conjunction with the new SQL Server Utility and the new Utility Explorer. Although the Standard edition is a strong database server solution, large organizations should consider the Enterprise edition. The Enterprise edition adds the following features:

- Unlimited scaling and partitioning, which provides for exceptional performance and the ability to scale SQL Server to support very large database installations. By horizontally partitioning tables across multiple servers, you can configure a group of servers to work together to support a large Web site or enterprise data processing.
- Advanced database mirroring for complete online parallel operations and advanced analysis tools for data mining and full-featured OLAP.
- Failover clustering for up to 16 nodes, multi-instance support, database snapshots, indexed views, online page and file restore, backup compression, and hot-add memory and CPU support.
- Extended data encryption and auditing features, including transparent data encryption (TDE) and C2-compliant tracing.
- Multiserver administration, unlimited virtualization, automatic server group management, and mirrored backup media.

New capabilities for the Enterprise edition in R2 include:

- Data compression with support for UCS-2 Unicode
- Master Data Services for data consistency across heterogeneous systems
- PowerPivot for SharePoint to enable querying and managing PowerPivot workbooks published to a SharePoint site
- Application and multiserver management for enrolling and managing server instances across the enterprise.

A Database Engine instance running the Enterprise edition can act as a UCP. As you might expect, SQL Server 2008 Enterprise edition runs on multiple versions of Windows, including desktop and server versions. The Developer edition supports all the features of the Enterprise edition but is licensed only for development and test use.

R2 Datacenter builds on the enhancements provided with the Enterprise edition and is designed to deliver a high-performance data platform. Datacenter offers high-scale complex event processing with SQL Server StreamInsight, support for

more than 8 physical processors and up to 256 logical processors, and support for RAM up to the operating system maximum. Most all editions of R2 can enroll for multi-instance management and be configured for DAC operations, but only the Datacenter edition can use the multi-instance dashboard views and drilldowns for centralized management. A Database Engine instance running the Datacenter edition can also act as a UCP. See Chapter 7 for more information.

R2 Parallel Data Warehouse, a specialized Datacenter edition, also builds on the enhancements provided with the Enterprise edition and is designed as a highly scalable, application-based solution for data warehouses. Parallel Data Warehouse offers support for massively parallel processing (MPP) architecture from tens of terabytes (TB) to one or more petabytes (PB) and advanced data warehousing capabilities, including star join queries, change data capture, and parallel database copy.

Other editions of SQL Server 2008 are available. These editions include the Web edition, the Compact edition (which replaces the Mobile edition in SQL Server 2005), the Express edition (which includes the redistributable database engine), and SQL Server Express with Advanced Services. The Compact edition allows you to use SQL Server as the data store for mobile devices, desktops, and Web clients. The Express edition is the version you run when you want an easy-to-use, low-end database solution for desktop and small server applications. The Express edition is free and can be distributed with third-party applications. Both editions support up to a 4-GB database size, up to 1 GB of RAM, and a single CPU.

**NOTE** With the exception of the Express and Compact editions, the differences between various editions of SQL Server are mostly below the surface and don't affect the user interface. I refer to specific editions and differentiate between their installations only when necessary. As you would expect, the Express and Compact editions have simple management interfaces.

All editions of SQL Server 2008 (as in SQL Server 2005) automatically and dynamically configure user connections. This behavior is different from SQL Server 7.0 and earlier versions, in which specific limitations were placed on the number of simultaneous user connections. Therefore, you don't have to be concerned about managing user connections as much as in early versions. Just keep in mind that as the number of user connections increases, so does the amount of resource usage on the server. The server has to balance the workload among the many user connections, which can result in decreased throughput for user connections and for the server as a whole.

Like SQL Server 2005, SQL Server 2008 uses the Windows Installer and has a fully integrated installation process. This means you can configure SQL Server 2008 components much like you can any other application you install on the operating system. You can perform the installation remotely from a command shell as well as locally.

SQL Server 2008 uses the same modular component architecture found in recent releases of Windows. Because of this modular architecture, all non-Express

editions of SQL Server 2008 can be part of a single build with the installed edition determined by the key. Further, you now can upgrade from one edition to another edition, which you could not do in earlier releases of SQL Server. For example, you can upgrade from the Developer edition to the Enterprise edition.

Chapter 2, “Deploying SQL Server 2008,” provides detailed instructions for installing SQL Server 2008. In an initial installation, the installer first checks the system configuration to determine the status of required services and components, which includes checking the configuration and availability of the operating system, operating system service packs, installation permissions for the default install path, memory, and hardware.

After checking the system configuration, the installer offers a choice of components to install. Whether you use the Developer, Workgroup, Standard, or Enterprise edition, you have similar options. You can install instance features, shared features, or both. With instance features—such as Database Engine Services, Analysis Services, and Reporting Services—you can install one or more instances of each on the Enterprise and Developer editions. Each instance of SQL Server consists of a distinct set of services that have specific settings for collation and other options. Directory structures in the file system and in the registry reflect instance names, as do the names associated with SQL Server support services. Shared features, on the other hand, are shared across all instances of SQL Server and include the client tools, development tools, Sync Framework, and Integration Services.

SQL Server replication and full-text search are part of Database Engine Services. Unlike in SQL Server 2005 and earlier releases of SQL Server, full-text search is now fully integrated into the Database Engine. Additionally, if you choose to install Reporting Services, SQL Server Setup configures the server as a report server. Unlike in SQL Server 2005 and earlier releases of SQL Server, Reporting Services does not require or use IIS.

By installing the client tools, you can work with remote computers running SQL Server from your workstation. When you install the client tools, you might also want to install SQL Server Books Online and Business Intelligence Development Studio. Books Online provides extended help documentation. Business Intelligence Development Studio allows you to develop business solutions for SQL Server.

**TIP** The SQL Native Client header and library files are installed with Database Engine instances of SQL Server. You'll find them under %ProgramFiles%\Microsoft SQL Server\100\SDK. When you are developing business solutions for SQL Server, you need to ensure that you copy and install all the required files.

## SQL Server and Windows

---

When you install SQL Server on server operating systems, SQL Server makes several modifications to the environment. These modifications include new system services, integrated authentication, new domain/workgroup accounts, and registry updates.



## Services for SQL Server

When you install SQL Server on Windows, several services are installed on the server. These services include the following:

- **Distributed Transaction Coordinator** Coordinates distributed transactions between two or more database servers.
- **SQL Active Directory Helper** MSSQLServerADHelper100 adds and removes objects used to register SQL Server and Analysis Server instances. It also updates object permissions related to SQL Server service accounts.
- **SQL Full-Text Filter Daemon Launcher** Starts the full-text filter daemon process. This process performs document filtering and word breaking for SQL Server full-text search.
- **SQL Server** The SQL Server service is the primary database service. For the default database instance, this service is named SQL Server (MSSQLServer). When multiple instances of SQL Server are installed, you also see MSSQL\$*instancename*, where *instancename* is the name of the SQL Server instance.
- **SQL Server Agent** The SQL Server Agent is used with scheduling and alerting. For the default database instance, this service is named SQLServer-Agent. When multiple instances of SQL Server are installed, you also see SQLAgent\$*instancename*, where *instancename* is the name of the SQL Server instance.
- **SQL Server Analysis Services** Microsoft SQL Server Analysis Services are used for OLAP and data mining. For the default database instance, this service is named MSSQLServerOLAPService. When multiple instances of SQL Server are installed, you also see MSOLAP\$*instancename*, where *instancename* is the name of the SQL Server instance.
- **SQL Server Browser** The SQL browser (SQLBrowser) provides connection details and information to clients.
- **SQL Server Integration Services 10.0** MsDtsServer100 provides an enterprise data transformation and integration solution for extracting and transforming data.
- **SQL Server Reporting Services** Microsoft Reporting Services creates, manages, and delivers reports. For the default database instance, this service is named ReportServer. When multiple instances of SQL Server are installed, you also see ReportServer\$*instancename*, where *instancename* is the name of the SQL Server instance.
- **SQL Server VSS Writer** SQLWriter provides the necessary interfaces for backing up and restoring SQL Server by using the Volume Shadow Copy Service (VSS).

**NOTE** You will learn more about managing services and configuring service-related options in Chapter 5, “Managing the Enterprise.”

# SQL Server Logins and Authentication

SQL Server authentication uses simple connection strings containing user IDs and passwords—a technique that is compatible with non-Windows clients and applications. Integrated Windows authentication provides a more robust solution for authentication by using Windows domain accounts and local computer accounts. Completely integrating SQL Server security with Windows domain security allows for authentication based on user and group memberships as well. Together, these authentication techniques make managing access and security much easier. You can do the following:

- Combine Windows and SQL Server authentication so that users in Windows domains can access the server by using a single account and other users can be logged on using a SQL Server login ID.
- Use authentication based only on Windows domain accounts so that only users with a domain account can access the server.

In SQL Server 2008, SQL Server authentication by default supports encryption, using certificates generated by SQL Server to encrypt the channel when transmitting login packets. Because of this, you do not have to acquire and install a valid Secure Sockets Layer (SSL) certificate to ensure that SQL Server credentials are secure and encrypted regardless of whether SQL Server logins or Windows logins are used.

When running on Windows Server, the Database Engine also uses Group Policy for password policy enforcement rules on SQL Server logins. SQL Server validates passwords automatically during authentication and whenever you set or reset passwords, unless you suspend enforcement for a specific login. For more information, see Chapter 9, “Managing SQL Server 2008 Security.”

## Service Accounts for SQL Server

When SQL Server services run on Windows Server, they can be configured to log on as the LocalSystem, LocalService, or NetworkService account or to use Windows logon accounts. There are advantages and disadvantages to each of these techniques, as follows:

- **Domain account** This option sets the service to use a standard domain account with privileges you configure. Use domain accounts when the server requires access to resources across the network, when you need to forward events to the application logs of other systems, and when you want to configure e-mail or pager notifications.
- **Local system account** This option provides administrative privileges to SQL Server on the local system but no privileges on the network. If the server requires resources only on the local server, use a local system account. Use local system accounts when you want to isolate SQL Server and restrict it from interacting with other servers.

- **Network service account** This option provides the same level of access to resources and objects as is granted members of the Users group and also allows services that run under this account to access the network and communicate with other servers. Specifically, processes running under this account can interact throughout a network by using the credentials of the computer account.
- **Local service account** This option provides the same level of access to resources and objects as is granted to members of the Users group. Services that run under this account access network resources as a null session without credentials. Use this account when a SQL Server service doesn't need access to other servers.

You should configure SQL Server services to log on using accounts with the lowest possible privileges, making sure only those privileges required are assigned. Table 1-1 provides a summary of the account types under which specific SQL Server services can run as well as the default account. Although SQL Server and SQL Server Agent can run under the NetworkService account, Microsoft doesn't recommend this when other services that use the account are installed on the computer.

**TABLE 1-1** Account Types for SQL Server Services

SERVICE NAME	DOMAIN USER	LOCAL SYSTEM	NETWORK SERVICE	LOCAL SERVICE	DEFAULT ACCOUNT
SQL Server	Yes	Yes	Yes	No	Domain user
SQL Server Agent	Yes	Yes	Yes	No	Domain user
Analysis Services	Yes	No	Yes	Yes	Domain user
Reporting Services	Yes	Yes	Yes	Yes	Domain user
Integration Services	Yes	Yes	Yes	Yes	NetworkService
SQL Server Browser	Yes	Yes	Yes	Yes	Domain user
SQL Server Active Directory Helper	No	Yes	Yes	No	Domain user
SQL Writer	No	Yes	No	No	LocalSystem

**NOTE** Security in SQL Server is managed through logins, server roles, database access permissions, and object permissions. Windows domain accounts can be used for user authentication and logging on to SQL Server. You can, for example, specify a Windows account to use for authentication and log on to SQL Server. You will learn more about SQL Server logins, server roles, and security in Chapter 9.

# Using the Graphical Administration Tools

---

SQL Server 2008 provides several types of tools for administration. The graphical administration tools are the ones you use most often. In R1, you can access these tools by clicking Start, choosing All Programs, and then using the Microsoft SQL Server 2008 menu. In R2, you can access these tools by clicking Start, choosing All Programs, and then using the Microsoft SQL Server 2008 R2 menu. With a full instance and shared feature installation, the Microsoft SQL Server 2008 and Microsoft SQL Server 2008 R2 menus have the following options and suboptions:

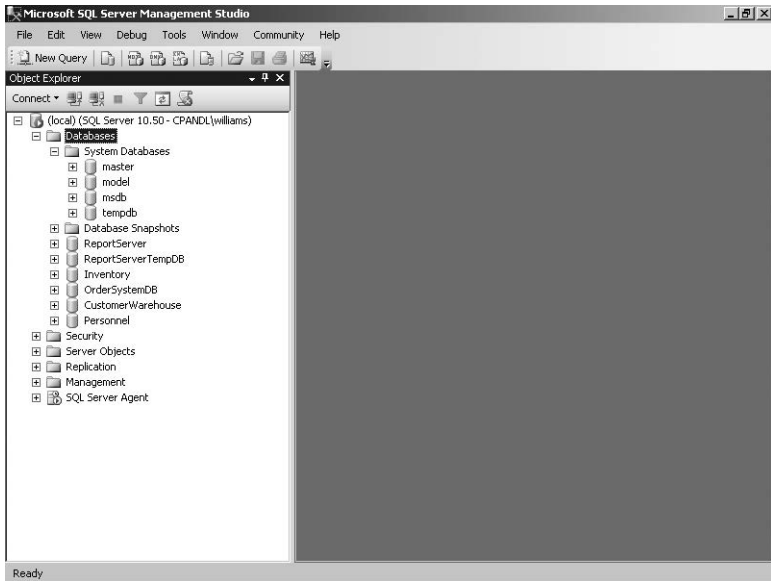
- Import And Export Data
- SQL Server Business Intelligence Studio
- SQL Server Management Studio
- Analysis Services
  - Deployment Wizard
- Configuration Tools
  - Reporting Services Configuration Manager
  - SQL Server Configuration Manager
  - SQL Server Error And Usage Reporting
  - SQL Server Installation Center
- Documentation And Tutorials
  - Microsoft SQL Server Samples Overview
  - SQL Server Books Online
  - SQL Server Tutorials
- Integration Services
  - Data Profile Viewer
  - Execute Package Utility
- Performance Tools
  - Database Engine Tuning Advisor
  - SQL Server Profiler

SQL Server Management Studio is the tool you use to perform most core SQL Server administration tasks. SQL Server Management Studio provides several different views. When you start working with this tool, you see the Object Explorer view, shown in Figure 1-1. If this view is not displayed, you can access it (and other views) from the View menu. The following descriptions explain how to use each view:

- **Object Explorer** Allows you to view and connect to SQL Server, Analysis Services, Integration Services, Reporting Services, and SQL Server Compact edition. Once you have connected to a particular server, you can view its

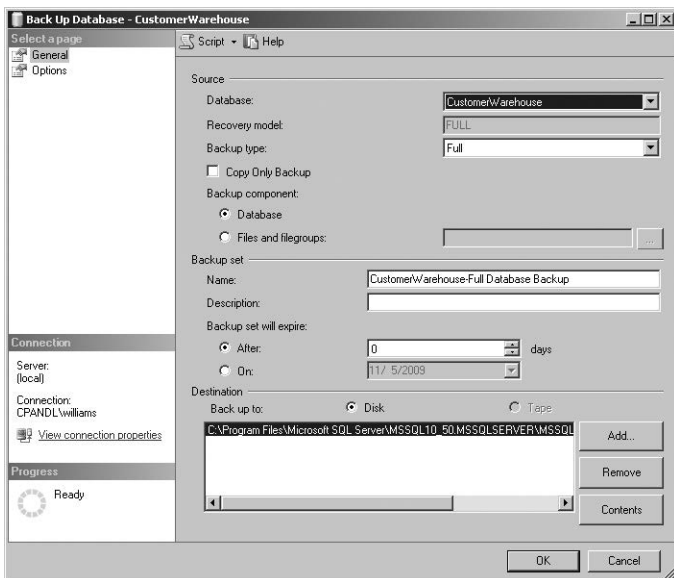
components as an object tree and can expand nodes to work your way to lower levels of the tree.

- **Registered Servers** Shows the currently registered servers. The top bar of the view allows you to quickly switch between servers of a particular type (SQL Server, Analysis Server, Integration Server, Report Server, SQL Server Compact Edition).
- **Template Explorer** Provides quick access to the default Query Editor templates and any custom templates you create. You can create templates in any script language supported by SQL Server Management Studio. SQL Server, Analysis Server, and SQL Server Compact edition support templates.
- **Solutions Explorer** Provides quick access to existing SQL Server, Analysis Server, and SQL Compact edition projects. A project details the connections, queries, and other functions that are performed when the project is executed.



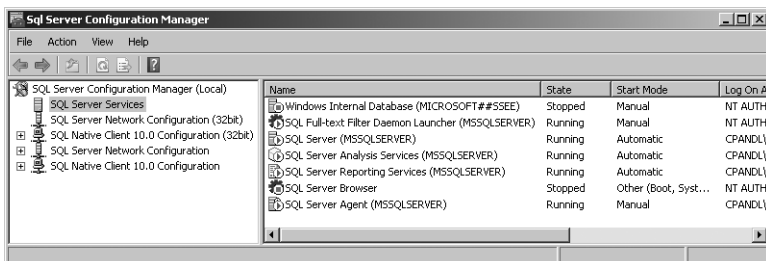
**FIGURE 1-1** Use SQL Server Management Studio to perform core administration tasks.

If you have worked with SQL Server 2000 or SQL Server 2005, you will find that SQL Server Management Studio is very different from the tools it replaces. Most of the friendly wizards are gone and have been replaced with nonmodal dialog boxes that provide quick access to configuration elements. As shown in Figure 1-2, Script and Help options are provided on the top bar of these dialog boxes to make it easy for you to generate a script based on your configuration choices and get help when you need it.



**FIGURE 1-2** Use the top bar options to quickly perform key tasks.

Another important tool is SQL Server Configuration Manager, shown in Figure 1-3. SQL Server Configuration Manager replaces Server Network Utility, Client Network Utility, and Services Manager. As a result, you can use SQL Server Configuration Manager to perform many essential service, setup, and network configuration tasks. When you select a service under the Services node, you can manage the service in the details pane by right-clicking it and then choosing an appropriate option, such as Start, Stop, or Restart. You can also choose Properties to configure the related settings, such as startup mode, login account, and login account password.



**FIGURE 1-3** Use SQL Server Configuration Manager to manage services and network configurations.

SQL Server 2008 is designed for local and remote management. You can use most of the administration tools to manage local resources as well as remote resources. For example, in SQL Server Management Studio, you can register a new

server and then connect to it. Afterward, you can remotely manage the server and all its databases from your system. Table 1-2 provides a summary of the graphical administration tools discussed, as well as other useful graphical tools.

**TABLE 1-2** Quick Reference for Key SQL Server 2008 Administration Tools

<b>ADMINISTRATION TOOL</b>	<b>PURPOSE</b>
Analysis Services Deployment Wizard	Allows you to deploy the output from an Analysis Services project to a target server by using specified deployment configuration settings and options.
Business Intelligence Development Studio	Allows you to develop and manage business intelligence objects. This feature includes SSIS Designer, which you can use to create and maintain SQL Server Integration Services (SSIS) packages.
Database Engine Tuning Advisor	Helps you tune the performance of SQL Server databases.
Import And Export Data Wizard	Allows you to define SSIS Services packages for importing and exporting data.
SQL Server Configuration Manager	Allows you to configure the client and server network libraries and manage SQL Server services. Replaces Server Network Utility, Client Network Utility, and Services Manager. This feature is covered in Chapter 3.
SQL Server Installation Center	Provides a central interface for upgrading to SQL Server 2008 from SQL Server 2000 or SQL Server 2005, installing instance and shared features, and configuring SQL Server clusters.
SQL Server Management Studio	The main administration tool for SQL Server 2008. This feature manages SQL servers, databases, security, and more. Key aspects are discussed in Chapter 7. It replaces SQL Server Enterprise Manager, Query Analyzer, and Analysis Manager.
SQL Server Profiler	Allows you to analyze user activity and generate audit trails. SQL Server Profiler is a graphical interface to SQL Trace. This feature is covered in Chapter 14, "Profiling and Monitoring SQL Server 2008."

# Using the Command-Line Tools

---

The graphical administration tools provide just about everything you need to work with SQL Server. Still, there are times when you might want to work from the command line, especially if you want to automate installation, administration, or maintenance with scripts. The primary command-line tool is SQLCMD (sqlcmd.exe), which replaces OSQL (osql.exe) and ISQL (isql.exe). Another command-line tool you can use is BCP (bcp.exe).

## BCP

BCP is the bulk copy program. You can use BCP to import and export data or to copy data between instances of SQL Server 2008. The major advantage of BCP is its speed. It is much faster than standard database import/export procedures. Unfortunately, its command-line syntax makes it much harder to use. The syntax for BCP is shown in Sample 1-1.

### SAMPLE 1-1 BCP Syntax

---

```
bcp {dtable | view | query} {in | out | queryout | format} datafile
[-m maxerrors] [-f formatfile] [-e errfile]
[-F firstrow] [-L lastrow] [-b batchsize]
[-n native type] [-c character type] [-w Unicode characters]
[-N keep non-text native] [-V file format version] [-q quoted id]
[-C code page specifier] [-t field terminator] [-r row terminator]
[-i inputfile] [-o outfile] [-a packetsize]
[-S server name\instance name] [-U username] [-P password]
[-T trusted connection] [-v version] [-R regional enable]
[-k keep null values] [-E keep identity values]
[-h "load hints"] [-x generate xml format file]
```

## SQLCMD

SQLCMD is an SQL query tool that you can run from the command line. Unlike OSQL and ISQL, which SQLCMD replaces, SQLCMD communicates with SQL Server only through the OLE DB application programming interface (API). Like OSQL and ISQL, SQLCMD has very little overhead, making it a good choice when system resources are a concern. Sample 1-2 shows the syntax for SQLCMD.



```

sqlcmd [-U login id] [-P password]
[-S servername[\instancename]] [-H hostname] [-E trusted connection]
[-d use database name] [-l login timeout] [-t query timeout]
[-h headers] [-s colseparator] [-w screen width]
[-a packetsize] [-e echo input] [-I Enable Quoted Identifier]
[-c cmdend] [-L[c] list servers[clean output]]
[-q "cmdline query"] [-Q "cmdline query" and exit]
[-m errorlevel] [-V severitylevel] [-W remove trailing spaces]
[-u unicode output] [-r[0|1] msgs to stderr]
[-i inputfile] [-o outputfile] [-z new password]
[-f <codepage> | i:<codepage>[,o:<codepage>]] [-Z new password and
exit]
[-k[1|2] remove[replace] control characters]
[-y variable length type display width]
[-Y fixed length type display width]
[-p[1] print statistics[colon format]]
[-R use client regional setting]
[-b On error batch abort]
[-v var = "value"...] [-A dedicated admin connection]
[-X[1] disable commands[and exit with warning]]
[-x disable variable substitution]

```

**NOTE** Unlike ISQL, SQLCMD supports connecting to named instances of SQL Server 2008. By default, SQLCMD connects to the default instance of SQL Server. If you specify the instance name as well as the server name, SQLCMD will connect to the specified instance on the designated server.

When you start SQLCMD, you can issue Transact-SQL (T-SQL) statements to run queries, execute stored procedures, and perform additional tasks. Because you are working at the command line, these commands aren't executed automatically. You need to use additional commands to tell SQLCMD when to execute statements, when to ignore statements, and so on. These additional statements must be entered on separate lines and are summarized in Table 1-3.

**NOTE** With some commands, the colon [:] is optional to maintain compatibility with OSQL scripts. This is indicated by the [:] notation in the table.

In previous releases of SQL Server, you used ODBCPING to verify an ODBC connection between a client and server. In SQL Server 2008, OLE DB is the preferred technique for establishing database connections. You can establish a connection to a server for the purposes of testing and troubleshooting by using SQLCMD -A.

**TABLE 1-3** SQLCMD Commands

COMMAND	DESCRIPTION	SUPPORTED BY INVOKE-SQLCMD
Ctrl+C	Ends a query without exiting from SQLCMD.	Yes
Go [ <i>count</i> ]	Executes all statements entered up to the previous GO or RESET command. If <i>count</i> is used, the cached statements are executed as a single batch the number of times specified in <i>count</i> .	Yes
!! <i>command</i>	Executes the specified system command or script.	No
:connect	Connects to an instance of SQL Server or closes the current connection. The syntax is as follows: <b>connect [timeout] [ServerName\Instance-Name] [Username] [Password]</b>	No
:ed	Calls the text editor, which is defined by the SQL-CMDEDITOR environment variable, such as <b>SET SQLCMDEDITOR=notepad</b> .	No
:error <i>filename</i>	Redirects all error output to the specified file.	No
:exit <i>statement</i>	Sets the exit statement. The batch or query is executed, and then SQLCMD quits.	Yes
:help	Displays a list of available commands.	Yes
:list	Prints the contents of the statement cache.	No
:listvar	Lists currently set variables.	No
:out <i>filename</i>	Redirects all query results to the specified file.	No
:perftrace <i>filename</i>	Redirects all performance trace information to the specified file.	No
:quit	Exits SQLCMD.	Yes
:r <i>filename</i>	Sets the name of a file containing T-SQL statements to execute, which can include the GO command.	Yes
:reset	Clears statements you've entered so that they aren't executed.	No
:serverlist	Lists the locally configured servers and any network servers.	No
:setvar	Sets variables.	Yes

## Other Command-Line Tools

Table 1-4 provides a summary of key command-line utilities included in SQL Server 2008. As the table shows, most command-line executables are stored in the %ProgramFiles%\Microsoft SQL Server\100\Tools\Binn directory or in the directory for the SQL Server component they relate to. On a 64-bit computer, you'll find a separate Program Files folder for 32-bit tools. For example, on X64 computers, you'll find a Program Files folder with 64-bit tools and a Program Files (x86) folder with 32-bit tools.

**NOTE** Some R2 tools are found under subfolders of %ProgramFiles%\Microsoft SQL Server\MSSQL10\_50, and some R1 tools are found under subfolders of %ProgramFiles%\Microsoft SQL Server\MSSQL10. Officially, R1 is SQL Server 10 and R2 is SQL Server 10.5

**TABLE 1-4** Key Command-Line Tools for SQL Server 2008

NAME	DESCRIPTION	LOCATION
Bulk Copy Utility (bcp.exe)	Used to import and export data or to copy data between instances of SQL Server.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn
Database Maintenance Plan Utility (sqlmaint.exe)	Used to execute database maintenance plans created in previous versions of SQL Server.	%ProgramFiles%\Microsoft SQL Server\MSSQL10_50\InstanceName\MSSQL\Binn
Replication Monitor (sqlmonitor.exe)	Used to start Replication Monitor from a command line.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn
Reporting Services Configuration Manager (rsconfigtool.exe)	Used to start Reporting Services Configuration Manager from a command prompt.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn
Reporting Services Configuration Utility (rsconfig.exe)	Used to configure a report server connection.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn
Reporting Services Key Management Utility (rskeymgmt.exe)	Used to manage encryption keys on a report server.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn
Reporting Services Utility (rs.exe)	Used to run Reporting Services scripts.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn

NAME	DESCRIPTION	LOCATION
SQL Diagnostics Utility (sqldiag.exe)	Used to perform comprehensive diagnostics testing for troubleshooting.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn
SQL Query Command-Line Utility (sqlcmd.exe)	Used to perform administration and enter T-SQL statements at the command prompt.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn
SQL Server Agent (sqlagent.exe)	Used to start SQL Server Agent from a command prompt.	%ProgramFiles%\Microsoft SQL Server\MSSQL10_50.InstanceName\MSSQL\Binn
SQL Server Database Engine (sqlservr.exe)	Used to start and stop an instance of the SQL Server Database Engine.	%ProgramFiles%\Microsoft SQL Server\MSSQL10_50.InstanceName\MSSQL\Binn
SQL Server Database Engine Tuning Advisor (dta.exe)	Used to analyze workloads and recommend optimization changes for that workload.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn
SQL Server Execution Package Utility (dtexec.exe)	Used to configure and execute a SQL Server Integration Services (SSIS) package. The corresponding GUI tool is DTEExecUI.	%ProgramFiles%\Microsoft SQL Server\100\DTS\Binn
SQL Server Log Shipping Agent (sqllogship.exe)	Used to configure and manage log shipping.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn
SQL Server PowerShell (sqlps.exe)	Used to start the SQL Server PowerShell and load the SQL Server provider.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn
SQL Server Profiler (profiler.exe)	Used to start SQL Server Profiler from a command prompt.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn

NAME	DESCRIPTION	LOCATION
SQL Server Replication Diff Tool (tablediff.exe)	Used to compare the data in two tables and display differences.	%ProgramFiles%\Microsoft SQL Server\100\COM
SQL Server Service Broker Diagnostics Utility (ssbdiagnose.exe)	Used to diagnose communication and configuration issues related to SQL Server Service Broker.	%ProgramFiles%\Microsoft SQL Server\100\Tools\Binn
SQL Server SSIS Package Utilities (dtutil.exe)	Used to manage SQL Server Integration Services (SSIS) packages.	%ProgramFiles%\Microsoft SQL Server\100\DTS\Binn

## Using SQL Server PowerShell

Windows PowerShell is being used more and more in enterprises, and SQL Server 2008 includes extensions to help you take advantage of this powerful shell environment. By default, when you install the Database Engine, SQL Server Setup installs Windows PowerShell and then configures extensions for SQL Server 2008. If you've already installed the appropriate version of Windows PowerShell, SQL Server Setup installs only the necessary extensions for SQL Server 2008.

### Running and Using Cmdlets

Windows PowerShell introduces the concept of a *cmdlet* (pronounced *commandlet*). A cmdlet is the smallest unit of functionality in Windows PowerShell. You can think of a cmdlet as a built-in command. Rather than being highly complex, most cmdlets are quite simple and have a small set of associated properties.

You use cmdlets the same way you use other commands and utilities. Cmdlet names are not case-sensitive. This means you can use a combination of uppercase and lowercase characters. After starting Windows PowerShell, you can enter the name of a cmdlet at the prompt, and it will run in much the same way as a command-line command.

For ease of reference, cmdlets are named using verb-noun pairs. The noun tells you specifically what the cmdlet works with. For example, the `Get-Variable` cmdlet gets a named Windows PowerShell environment variable and returns its value. If you don't specify which variable to get as a parameter, `Get-Variable` returns a list of all Windows PowerShell environment variables and their values.

You can work with cmdlets by executing commands directly at the shell prompt or by running commands from scripts. You can enter any command or cmdlet that you can run at the Windows PowerShell command prompt into a script by copying

the related command text to a file and saving the file with the .ps1 extension. You can then run the script in the same way you would any other command or cmdlet. However, when you are working with Windows PowerShell, the current directory might not be part of the environment path. For this reason, you might need to use the “./” notation when you run a script in the current directory, such as the following:

```
./runtasks
```

From the Windows command-line environment or a batch script, you can execute Windows PowerShell cmdlets with the `-Command` parameter. Generally, you should also suppress the Windows PowerShell logo with the `-Nologo` parameter and stop execution of profiles with the `-Noprofile` parameter. For example, at a command prompt or in a .bat script, you can get a list of running processes by using the following command:

```
powershell -nologo -noprofile -command get-process
```

**NOTE** Windows PowerShell also includes a rich scripting language and allows the use of standard language constructs for looping, conditional execution, flow control, and variable assignment. Discussion of these features is beyond the scope of this book. A good resource is *Windows PowerShell 2.0 Administrator's Pocket Consultant* (Microsoft Press, 2009).

## Running and Using SQL Server PowerShell

SQL Server PowerShell (sqlps.exe) is a version of Windows PowerShell with extensions for SQL Server 2008. You use SQL Server PowerShell to work with instances of the SQL Server Database Engine and the objects in those instances.

To invoke SQL Server PowerShell, you must first open a Command Prompt window or Windows PowerShell prompt and then start SQL Server PowerShell by typing **sqlps** at the command line. To exit SQL Server PowerShell and return to the standard prompt, type **exit**.

Usually, when the shell starts, you see a message similar to the following:

```
Microsoft SQL Server PowerShell
Version 10.50
Microsoft Corp. All rights reserved.
```

You can disable this message by starting the shell with the `-Nologo` parameter, such as

```
sqlps -nologo
```

Regardless of how you start the shell, you know you are using SQL Server PowerShell because the command prompt title bar changes to Command Prompt – sqlps and the current path is preceded by PS.

The current execution policy for SQL Server PowerShell controls whether and how you can run scripts. Although the default configuration depends on which operating system and edition you've installed, you can quickly determine the execution policy by entering **get-executionpolicy** at the Windows PowerShell prompt.

To set the execution policy to require that all scripts have a trusted signature to execute, enter the following command:

```
set-executionpolicy allsigned
```

To set the execution policy so that scripts downloaded from the Web execute only if they are signed by a trusted source, enter:

```
set-executionpolicy remotesigned
```

To set the execution policy to run scripts regardless of whether they have a digital signature and work in an unrestricted environment, you can enter the following command:

```
set-executionpolicy unrestricted
```

## Working with SQL Server Cmdlets

When you work with the SQL Server provider for PowerShell, the available cmdlets are different from those you use when you work with standard Windows PowerShell. The reason is that the set of registered snap-ins is different. Additional SQL Server–specific cmdlets are available, and some standard Windows PowerShell cmdlets might not be available.

In the original implementation of SQL Server PowerShell, the following additional cmdlets are included:

- **Convert-UrnToPath** Converts a SQL Server Management Object Uniform Resource Name (URN) to a SQL Server provider path. The URN indicates a management object's location within the SQL Server object hierarchy. If the URN path has characters not supported by Windows PowerShell, the characters are encoded automatically.

```
Convert-UrnToPath [-Urn] UrnToConvert
```

- **Decode-SQLName** Returns an unencoded SQL Server identifier when given an identifier that has been encoded.

```
Decode-SqlName [-SqlName] IdentifierToBeDecoded
```

- **Encode-SQLName** Encodes special characters in SQL Server identifiers and name paths to formats that are usable in Windows PowerShell paths. The characters encoded by this cmdlet include `\:/%<>?*?[]`. If you don't encode

these characters, you must escape them by using the single quotation mark (') character.

```
Encode-SqlName [-SqlName] IdentifierToBeEncoded
```

- **Invoke-PolicyEvaluation** Evaluates management policies applied to SQL Server instances. By default, this command reports compliance but does not enforce compliance. To enforce compliance, set `-AdHocPolicyEvaluationMode` to `Configure`.

```
Invoke-PolicyEvaluation [-Policy] PolicyName -TargetServerName  
ServerName [-TargetExpression QueryString] <AddtlParams>
```

```
Invoke-PolicyEvaluation [-Policy] PolicyName -TargetObjects  
SQLObjects <AddtlParams>
```

```
<AddtlParams>
```

```
[-AdHocPolicyEvaluationMode AdHocPolicyEvalMode] [-OutputXml]
```

- **Invoke-Sqlcmd** Runs a T-SQL or XQuery script containing commands supported by the SQLCMD utility. By default, this cmdlet doesn't set any SQLCMD variables by default or return message output. As shown in Table 1-3, many SQLCMD commands aren't supported.

```
Invoke-Sqlcmd [-ServerInstance ServerStringOrObject] [-Database  
DatabaseName] [-EncryptConnection] [-Username  
UserName] [-Password Password] [[-Query] QueryString]
```

```
[-AbortOnError] [-ConnectionTimeout Timeout]
```

```
[-DedicatedAdministratorConnection]
```

```
[-DisableCommands] [-DisableVariables] [-ErrorLevel (0..24)]
```

```
[-HostName ComputerNameForSP_Who] [-IgnoreProviderContext]
```

```
[-InputFile FilePath]
```

```
[-MaxBinaryLength <1024|MaxBin] [-MaxCharLength <4000|MaxChars>]
```

```
[-NewPassword NewPasswordForSQLId] [-OutputSqlErrors]
```

```
[-QueryTimeout Timeout]
```

```
[-SeverityLevel (0..24)] [-SuppressProviderContextWarning]
```

```
[-Variable VariableString]
```

Because the set of available cmdlets and cmdlet options changes as new versions of SQL Server PowerShell are released by Microsoft, you can use the following techniques to discover new cmdlets and determine how they are used:

- To view a list of all cmdlets, type **get-command** at the shell prompt.
- To get detailed information about a cmdlet, type **get-help cmdletname -detailed**, where *cmdletname* is the name of the cmdlet you want to examine.



- To get detailed information about the SQL Server provider, which provides SQL Server functionality for Windows PowerShell, type **get-help sqlserver ] more**.

You'll find detailed information about using Windows PowerShell for administration in Chapter 5. To load the SQL Server environment from a script, the script must add the SQL Server snap-ins, set certain global variables, and then load the SQL Server management objects. A sample initialization script follows:

```
#
# Add the SQL Server PowerShell Provider, if available
$erroractionpreference = "Stop"
$sqlpsreg="HKLM:\SOFTWARE\Microsoft\PowerShell\1\ShellIds\
Microsoft.SqlServer.Management.PowerShell.sqlps"

if (Get-ChildItem $sqlpsreg -ErrorAction "SilentlyContinue")
{ throw "SQL Server Powershell Provider is not installed."
} else {
    $item = Get-ItemProperty $sqlpsreg
    $sqlpsPath = [System.IO.Path]::GetDirectoryName($item.Path)
}

#
# Set global variables
Set-Variable SqlServerMaximumChildItems 0 -scope Global
Set-Variable SqlServerConnectionTimeout 30 -scope Global
Set-Variable SqlServerIncludeSystemObjects $false -scope Global
Set-Variable SqlServerMaximumTabCompletion 1000 -scope Global

#
# Load the SQL Server Management Objects
$assemblylist = "Microsoft.SqlServer.Smo",
"Microsoft.SqlServer.Dmf ",
"Microsoft.SqlServer.SqlWmiManagement ",
"Microsoft.SqlServer.ConnectionInfo ",
"Microsoft.SqlServer.SmoExtended ",
"Microsoft.SqlServer.Management.RegisteredServers ",
"Microsoft.SqlServer.Management.Sdk.Sfc ",
"Microsoft.SqlServer.SqlEnum ",
"Microsoft.SqlServer.RegSvrEnum ",
"Microsoft.SqlServer.WmiEnum ",
"Microsoft.SqlServer.ServiceBrokerEnum ",
"Microsoft.SqlServer.ConnectionInfoExtended ",
"Microsoft.SqlServer.Management.Collector ",
"Microsoft.SqlServer.Management.CollectorEnum"
```

```
foreach ($asm in $assemblylist)
{ $asm = [Reflection.Assembly]::LoadWithPartialName($asm) }

#
# Load SQL Server snapins, type data and format data
Push-Location
cd $sqlpsPath
Add-PSSnapin SqlServerCmdletSnapin100
Add-PSSnapin SqlServerProviderSnapin100
Update-TypeData -PrependPath SQLProvider.Types.ps1xml
update-FormatData -prependpath SQLProvider.Format.ps1xml
Pop-Location
```

Before you use this type of initialization script, you should check current documentation for the version and service pack of SQL Server you are running to determine the required components. On the Microsoft support site ([support.microsoft.com](http://support.microsoft.com)), you'll likely find examples of initialization scripts for your version and service pack.

# Configuring and Tuning SQL Server 2008

- Accessing SQL Server Configuration Data **86**
- Techniques for Managing SQL Server Configuration Options **101**
- Configuring SQL Server with Stored Procedures **109**

Microsoft designed SQL Server 2008 to balance workloads dynamically and to self-tune configuration settings. For example, SQL Server can increase or decrease memory usage based on overall system memory requirements. SQL Server also manages memory efficiently, especially when it comes to queries and user connections—and memory is just one of dozens of areas in which configuration settings are automatically adjusted.

Although the SQL Server self-tuning feature works well, there are times when you need to configure SQL Server settings manually. For example, if you are running a large database with special constraints and the database is not performing the way you expect it to, you might want to customize the configuration. You might also need to modify configuration settings for SQL Server accounts, authentication, and auditing. Key tools you use to configure and tune SQL Server include the following:

- **System catalog queries** Provide a direct way to determine database configuration characteristics and their related settings.
- **Stored procedures** Let you view and manage configuration settings through stored procedures such as `sp_configure`.
- **SQL Server Management Studio** Provides an easy-to-use interface that updates the database and registry settings for you.
- **SQLServr.exe** Starts SQL Server from the command line. You can use `SQLServr.exe` to set configuration parameters at startup.

In this chapter, I'll describe the structures available for configuring and tuning SQL Server. I'll start with a look at the SQL Server 2008 system catalog and then

continue with a discussion of catalog queries and stored procedures. This discussion provides the essential background for understanding how to configure and tune SQL Server 2008. Chapter 5, “Managing the Enterprise,” provides details about using SQL Server Management Studio and SQLServr.exe.

## Accessing SQL Server Configuration Data

---

SQL Server 2008 uses an object-based approach to representing servers and databases and all of their configuration characteristics and data contents. At the heart of this object-based structure is the system catalog, which describes the objects in a particular instance of SQL Server along with their attributes. For example, attributes of a database can describe the following:

- The number and names of the tables and views
- The number and names of columns in a table or view
- The column data type, scale, and precision
- The triggers and constraints that are defined on a table
- The indexes and keys that are defined for a table
- The statistics used by the query optimizer for generating query plans

In queries, you can access these attributes and other system catalog information by using the following:

- **Catalog views** Provide access to metadata stored in a database, which includes database attributes and their values. Catalog views can be used to access all user-available metadata except for metadata related to replication, backup, database maintenance plans, and SQL Server Agent.
- **Compatibility views** Provide access to many of the system tables included in earlier releases of SQL Server by using SQL Server 2008 views. These views are meant for backward compatibility only and expose the same metadata that is available in SQL Server 2000. They do not expose metadata for new SQL Server 2008 features, such as database partitioning and mirroring.
- **Information Schema views** Provide access to a subset of metadata stored in a database, which includes database attributes and their values. Information Schema views are based on catalog view definitions in the SQL-92 standard and do not contain metadata specific to SQL Server 2008. Applications that use these views are portable between heterogeneous SQL-92-compliant database systems.
- **ODBC catalog functions** Provide an interface that Open Database Connectivity (ODBC) drivers can use to return result sets containing system catalog information. The result sets present catalog information in a way that is independent of the structure of the underlying catalog tables.
- **OLE DB schema rowsets** Provide an IDBSchemaRowset interface that OLE DB providers can use to access system catalog information. The rowsets

present catalog information independently from the structure of the underlying catalog tables.

- **System stored procedures and functions** Provide Transact-SQL (T-SQL) stored procedures and functions that return catalog information.

Catalog views and stored procedures are the recommended methods for accessing a database's metadata, primarily because catalog views present metadata in a format that is independent of any catalog table implementation, which means that the views are not affected by changes in the underlying catalog tables. When you want to configure or manage a server, you typically use stored procedures to help you perform the necessary tasks. Stored procedures provide the functionality to view and manage the configuration of SQL Server and related databases with ease.

## Working with the System Catalog and Catalog Views

Catalog views contain information used by the SQL Server 2008 Database Engine. They provide the most general interface to the catalog metadata and are the most direct way to access and work with this information. All user-available metadata in the system catalog is exposed through catalog views. Catalog views do not contain information about replication, backup, database maintenance plans, or SQL Server Agent.

Like all structures in SQL Server 2008 databases, catalog views follow an object-based hierarchy in which lower-level objects inherit attributes from higher-level objects. Some catalog views inherit rows from other catalog views. For example, the Tables catalog view inherits all the columns defined in the Objects catalog view. Thus, in addition to columns that are specific to the Tables catalog view itself, the Tables catalog view includes the columns from the Objects catalog view. Table 4-1 summarizes the SQL Server 2008 catalog views and their uses.

**TABLE 4-1** SQL Server 2008 Catalog Views

<b>VIEW TYPE</b>	<b>DESCRIPTION</b>	<b>KEY CATALOG VIEWS</b>
Change Tracking	Describes change tracking components and values in databases and tables	sys.change_tracking_databases sys.change_tracking_tables
CLR Assembly	Describes common language runtime (CLR) assemblies	sys.assemblies sys.assembly_files sys.assembly_references
Databases and Files	Describes databases, database files, and backup devices associated with a SQL Server instance	sys.backup_devices sys.database_files sys.database_recovery_status sys.databases sys.master_files

VIEW TYPE	DESCRIPTION	KEY CATALOG VIEWS
Database Mirroring	Describes witness roles that a server plays as a database mirroring partner	sys.database_mirroring sys.database_mirroring_endpoints sys.database_mirroring_witnesses
Data Spaces	Describes filegroups and partition schemes	sys.data_spaces sys.destination_data_spaces sys.filegroups sys.partition_schemes
Endpoints	Describes endpoints used for mirroring, service broker messaging, and Web services	sys.endpoints sys.endpoint_webmethods sys.http_endpoints sys.service_broker_endpoints sys.soap_endpoints sys.tcp_endpoints sys.via_endpoints
Extended Properties	Describes extended properties and the class of objects from which they originate	sys.extended_properties
Linked Servers	Describes linked or remote servers and their related logins	sys.linked_logins sys.remote_logins sys.servers
Messages (for Errors)	Describes system-defined and user-defined error messages	sys.messages
Objects	Describes top-level database objects	sys.allocation_units sys.assembly_modules sys.check_constraints sys.columns sys.computed_columns sys.default_constraints sys.event_notifications sys.events sys.extended_procedures sys.foreign_key_columns sys.foreign_keys sys.function_order_columns sys.identity_columns sys.index_columns

VIEW TYPE	DESCRIPTION	KEY CATALOG VIEWS
Objects <i>(continued)</i>	Describes top-level database objects	sys.indexes sys.key_constraints sys.numbered_procedure_parameters sys.numbered_procedures sys.objects sys.parameters sys.partitions sys.procedures sys.service_queues sys.spatial_index_tessellations sys.spatial_indexes sys.sql_dependencies sys.sql_modules sys.stats sys.stats_columns sys.synonyms sys.table_types sys.tables sys.trace_events sys.traces sys.trigger_event_types sys.trigger_events sys.triggers sys.views
Partition Function	Describes partition functions, parameters, and range values	sys.partition_functions sys.partition_parameters sys.partition_range_values
Resource Governor	Describes workloads, resource pools, and states for managed resources	sys.resource_governor_configuration sys.resource_governor_workload_groups sys.resource_governor_resource_pools
Scalar Types	Describes user-defined scalar types for CLR assemblies as well as other system-defined and user-defined scalar types	sys.assembly_types sys.types
Schemas	Describes database schemas	sys.schemas

VIEW TYPE	DESCRIPTION	KEY CATALOG VIEWS
Security	Describes server-level, database-level, and encryption security attributes and values	<p>Database-level views:</p> <p>sys.database_permissions  sys.database_principals  sys.database_role_members  sys.master_key_passwords</p> <p>Server-level views:</p> <p>sys.server_permissions  sys.server_principals  sys.server_role_members  sys.sql_logins  sys.system_components_surface_area_configuration</p> <p>Encryption views:</p> <p>sys.asymmetric_keys  sys.certificates  sys.credentials  sys.crypt_properties,  sys.cryptographic_providers  sys.key_encryptions  sys.symmetric_keys</p> <p>Audit views:</p> <p>sys.database_audit_specification_details  sys.database_audit_specifications  sys.server_audit_specifications  sys.server_audit_specifications_details  sys.server_audits  sys.server_file_audits</p>
Service Broker	Describes Service Broker endpoints and messaging components	<p>sys.conversation_endpoints  sys.conversation_groups  sys.conversation_priorities  sys.remote_service_bindings  sys.routes  sys.service_contract_message_usages  sys.service_contract_usages  sys.service_contracts  sys.service_message_types  sys.service_queue_usages  sys.services  sys.transmission_queue</p>



VIEW TYPE	DESCRIPTION	KEY CATALOG VIEWS
Server-Wide Configuration	Describes serverwide configuration option values	sys.configurations sys.trace_categories sys.trace_columns sys.trace_event_bindings sys.trace_events sys.trace_subclass_values sys.traces
XML Schemas (XML Type System)	Describes XML Schema components and values	sys.xml_indexes sys.xml_schema_attributes sys.xml_schema_collections sys.xml_schema_component_placements sys.xml_schema_components sys.xml_schema_elements sys.xml_schema_facets sys.xml_schema_model_groups sys.xml_schema_namespaces sys.xml_schema_types sys.xml_schema_wildcard_namespaces sys.xml_schema_wildcards

Table 4-2 shows the mapping between SQL Server 2000 system tables and SQL Server 2008 system views. The entries are organized by database and view type. Mappings for the *master* database are followed by mappings for all databases. When you want to map uses of the `sysfulltextcatalogs` stored procedure, you should use the following data definition language (DDL) statements instead of the available SQL Server stored procedures: `CREATE FULLTEXT CATALOG`, `ALTER FULLTEXT CATALOG`, and `DROP FULLTEXT CATALOG`. These statements provide additional functionality and help ensure that your applications are compatible with future editions of SQL Server.

**TABLE 4-2** Mapping SQL Server 2000 System Tables to SQL Server 2008 System Views

<b>SQL SERVER 2000 SYSTEM TABLE</b>	<b>SQL SERVER 2008 SYSTEM VIEW</b>	<b>SQL SERVER 2008 VIEW TYPE</b>
<b>master DATABASE</b>		
sysaltfiles	sys.master_files	Catalog view
syscacheobjects	sys.dm_exec_cached_plans sys.dm_exec_plan_attributes sys.dm_exec_sql_text sys.dm_exec_cached_plans_ dependent_objects	Dynamic management view
syscharsets	sys.syscharsets	Compatibility view
sysconfigures	sys.configurations	Catalog view
syscurconfigs	sys.configurations	Catalog view
sysdatabases	sys.databases	Catalog view
sysdevices	sys.backup_devices	Catalog view
syslanguages	sys.languages	Compatibility view
syslockinfo	sys.dm_tran_locks	Dynamic management view
syslocks	sys.dm_tran_locks	Dynamic management view
syslogins	sys.server_principals sys.sql_logins	Catalog view
sysmessages	sys.messages	Catalog view
sysoledbusers	sys.linked_logins	Catalog view
sysopentapes	sys.dm_io_backup_tapes	Dynamic management view
sysperfinfo	sys.dm_os_performance_ counters	Dynamic management view
sysprocesses	sys.dm_exec_connections sys.dm_exec_requests sys.dm_exec_sessions	Dynamic management view
sysremotelogins	sys.remote_logins	Catalog view
syssservers	sys.servers	Catalog view
<b>ALL DATABASES</b>		
fn_virtualfilestats	sys.dm_io_virtual_file_stats	Dynamic management view
syscolumns	sys.columns	Catalog view

SQL SERVER 2000 SYSTEM TABLE	SQL SERVER 2008 SYSTEM VIEW	SQL SERVER 2008 VIEW TYPE
syscomments	sys.sql_modules	Catalog view
sysconstraints	sys.check_constraints sys.default_constraints sys.key_constraints sys.foreign_keys	Catalog view
sysdepends	sys.sql_expression_ dependencies	Catalog view
sysfilegroups	sys.filegroups	Catalog view
sysfiles	sys.database_files	Catalog view
sysforeignkeys	sys.foreign_key_columns sys.foreign_keys	Catalog view
sysindexes	sys.indexes sys.partitions sys.allocation_units sys.dm_db_partition_stats	Catalog view  Dynamic management view
sysindexkeys	sys.index_columns	Catalog view
sysmembers	sys.database_role_members	Catalog view
sysobjects	sys.objects	Catalog view
syspermissions	sys.database_permissions sys.server_permissions	Catalog view
sysprotects	sys.database_permissions sys.server_permissions	Catalog view
sysreferences	sys.foreign_keys	Catalog view
systypes	sys.types	Catalog view
sysusers	sys.database_principals	Catalog view

## Working with System Stored Procedures

You can use system stored procedures to view SQL Server configuration details and to perform general administration. SQL Server 2008 has two main categories of system stored procedures:

- Those meant for administrators
- Those used to implement functionality for database application programming interfaces (APIs)

Of course, you will work with system stored procedures meant for administration and not with those that implement database API functions. System stored procedures are written using Transact-SQL (T-SQL). Most return a value of 0 to indicate success and a nonzero value to indicate failure. As an example, `sp_dboption` is a stored procedure for managing the configuration options of SQL Server databases (except for the *master* and *tempdb* databases). When you use `sp_dboption` to set a database configuration value, a return code of 0 indicates that the option was set as expected. A return code of 1 indicates that the stored procedure failed and the option was not set as expected.

The following example takes the Personnel database offline if there are no current users:

**T-SQL**

```
USE master;  
GO  
EXEC sp_dboption "Personnel", "offline", "TRUE";  
GO
```

**PowerShell**

```
Invoke-Sqlcmd -Query "USE master; EXEC sp_dboption 'Personnel',  
'offline', 'TRUE';" -ServerInstance "CorpServer17\DataServices"
```

If the stored procedure returns 0, the database was successfully taken offline. A return value of 1 indicates that a problem occurred taking the database offline, and the database is still online. For more information about using stored procedures, see “Configuring SQL Server with Stored Procedures” later in this chapter. Note that you can change some options of `sp_configure` only when Show Advanced Options is set to 1, as in the following examples:

**T-SQL**

```
exec sp_configure "show advanced options", 1
```

**PowerShell**

```
Invoke-Sqlcmd -Query "exec sp_configure 'show advanced options', 1"  
-ServerInstance "DbServer18\OrderSystem"
```

**NOTE** When you use `Invoke-Sqlcmd`, you specify the Database Engine instance with the `-ServerInstance` parameter in the form `-ServerInstance "ServerName"` for the default instance or `-ServerInstance "ServerName\InstanceName"` for a nondefault instance. For more information on using `Invoke-Sqlcmd`, see Chapter 5.

Table 4-3 provides a summary of stored procedures for administration. The table entries are organized by the type of administration activity for which the stored procedure is designed.

**TABLE 4-3** Key System Stored Procedures by Type

<b>STORED PROCEDURE TYPE</b>	<b>DESCRIPTION</b>	<b>RELATED SYSTEM STORED PROCEDURES</b>
Active Directory stored procedures	Register instances of SQL Server and SQL Server databases in Active Directory.	sp_ActiveDirectory_Obj, sp_ActiveDirectory_SCP
Catalog stored procedures	Implement ODBC data dictionary functions.	sp_column_privileges, sp_columns, sp_databases, sp_fkeys, sp_pkeys, sp_server_info, sp_special_columns, sp_sproc_columns, sp_statistics, sp_stored_procedures, sp_table_privileges, sp_tables
Change Tracking stored procedures	Implement, manage, and monitor change tracking functionality.	sys.sp_cdc_add_job, sys.sp_cdc_change_job, sys.sp_cdc_cleanup_change_table, sys.sp_cdc_disable_db, sys.sp_cdc_disable_table, sys.sp_cdc_drop_job, sys.sp_cdc_enable_db, sys.sp_cdc_enable_table, sys.sp_cdc_generate_wrapper_function, sys.sp_cdc_get_captured_columns, sys.sp_cdc_get_ddl_history, sys.sp_cdc_help_change_data_capture, sys.sp_cdc_help_jobs, sys.sp_cdc_scan, sys.sp_cdc_start_job, sys.sp_cdc_stop_job
Cursor stored procedures	Implement cursor variable functionality.	sp_cursor_list, sp_describe_cursor, sp_describe_cursor_columns, sp_describe_cursor_tables
Database Engine stored procedures	Maintain SQL Server instances and perform general administration activities.	sp_add_data_file_recover_suspect_db, sp_add_log_file_recover_suspect_db, sp_addextendedproc, sp_addextendedproperty, sp_addmessage, sp_addtype, sp_addumpdevice, sp_altermessage, sp_attach_db, sp_attach_single_file_db, sp_autostats, sp_bindefault, sp_bindrule, sp_bindsession, sp_certify_removable, sp_configure, sp_control_plan_guide, sp_create_plan_guide, sp_create_plan_guide_from_cache,

STORED PROCEDURE TYPE	DESCRIPTION	RELATED SYSTEM STORED PROCEDURES
Database Engine stored procedures <i>(continued)</i>	Maintain SQL Server instances and perform general administration activities.	sp_create_removable, sp_createstats, sp_cycle_errorlog, sp_datatype_info, sp_dbcmptlevel, sp_dbmmonitoraddmonitoring, sp_dbmmonitorchangealert, sp_dbmmonitorchangemonitoring, sp_dbmmonitordropalert, sp_dbmmonitordropmonitoring, sp_dbmmonitorhelpalert, sp_dbmmonitorhelpmonitoring, sp_dbmmonitorresults, sp_dboption, sp_dbremove, sp_delete_backuphistory, sp_depends, sp_detach_db, sp_dropdevice, sp_dropextendedproc, sp_dropextendedproperty, sp_dropmessage, sp_droptype, sp_executesql, sp_filestream_configure, sp_getapplock, sp_getbindtoken, sp_help, sp_helpconstraint, sp_helpdb, sp_helpdevice, sp_helpextendedproc, sp_helpfile, sp_helpfilegroup, sp_helpindex, sp_helplanguage, sp_helpserver, sp_helpsort, sp_helpstats, sp_helptext, sp_helptrigger, sp_indexoption, sp_invalidate_textptr, sp_lock, sp_monitor, sp_procoption, sp_recompile, sp_refreshview, sp_releaseapplock, sp_rename, sp_renamedb, sp_resetstatus, sp_serveroption, sp_setnetname, sp_settriggerorder, sp_spaceused, sp_tableoption, sp_unbindefault, sp_unbindrule, sp_updateextendedproperty, sp_updatestats, sp_validname, sp_who
Database Mail stored procedures	Perform e-mail operations from SQL Server.	sp_send_dbmail, sysmail_add_account_sp, sysmail_add_principalprofile_sp, sysmail_add_profile_sp, sysmail_add_profileaccount_sp, sysmail_configure_sp, sysmail_delete_account_sp, sysmail_delete_log_sp, sysmail_delete_mailitems_sp,

STORED PROCEDURE TYPE	DESCRIPTION	RELATED SYSTEM STORED PROCEDURES
Database Mail stored procedures <i>(continued)</i>	Perform e-mail operations from SQL Server.	sysmail_delete_principalprofile_sp, sysmail_delete_profile_sp, sysmail_delete_profileaccount_sp, sysmail_help_account_sp, sysmail_help_configure_sp, sysmail_help_principalprofile_sp, sysmail_help_profile_sp, sysmail_help_profileaccount_sp, sysmail_help_queue_sp, sysmail_start_sp, sysmail_stop_sp, sysmail_update_account_sp, sysmail_update_principalprofile_sp, sysmail_update_profile_sp, sysmail_update_profileaccount_sp
Database Maintenance Plan stored procedures	Configure and manage database maintenance plans and related tasks.	sp_add_maintenance_plan, sp_add_maintenance_plan_db, sp_add_maintenance_plan_job, sp_delete_maintenance_plan, sp_delete_maintenance_plan_db, sp_delete_maintenance_plan_job, sp_help_maintenance_plan
Distributed Queries stored procedures	Implement and manage distributed queries.	sp_addlinkedserver, sp_addlinkedsrvlogin, sp_catalogs, sp_column_privileges_ex, sp_columns_ex, sp_droplinkedsrvlogin, sp_dropserver, sp_foreignkeys, sp_indexes, sp_linkedservers, sp_primarykeys, sp_serveroption, sp_table_privileges_ex, sp_tables_ex, sp_testlinkedserver
Full-Text Search stored procedures	Implement and query full-text indexes.	Deprecated. Use CREATE FULLTEXT CATALOG, ALTER FULLTEXT CATALOG, and DROP FULLTEXT CATALOG.
General Extended stored procedures	Provide an interface from SQL Server to external programs, primarily for server maintenance.	xp_cmdshell, xp_enumgroups, xp_findnextmsg, xp_grantlogin, xp_logevent, xp_loginconfig, xp_logininfo, xp_msver, xp_revokelogin, xp_sprintf, xp_sqlmaint, xp_sscanf

STORED PROCEDURE TYPE	DESCRIPTION	RELATED SYSTEM STORED PROCEDURES
Log Shipping stored procedures	Implement, manage, and monitor log shipping configurations.	sp_add_log_shipping_alert_job, sp_add_log_shipping_primary_database, sp_add_log_shipping_primary_secondary, sp_add_log_shipping_secondary_database, sp_add_log_shipping_secondary_primary, sp_change_log_shipping_primary_database, sp_change_log_shipping_secondary_database, sp_change_log_shipping_secondary_primary, sp_cleanup_log_shipping_history, sp_delete_log_shipping_alert_job, sp_delete_log_shipping_primary_database, sp_delete_log_shipping_primary_secondary, sp_delete_log_shipping_secondary_database, sp_delete_log_shipping_secondary_primary, sp_help_log_shipping_alert_job, sp_help_log_shipping_monitor_primary, sp_help_log_shipping_monitor_secondary, sp_help_log_shipping_primary_database, sp_help_log_shipping_primary_secondary, sp_help_log_shipping_secondary_database, sp_help_log_shipping_secondary_primary, sp_refresh_log_shipping_monitor, sp_resolve_logins
OLE Automation stored procedures	Create and manage OLE automation objects.	sp_OACreate, sp_OADestroy, sp_OAGetErrorInfo, sp_OAGetProperty, sp_OAMethod, sp_OASetProperty, sp_OAStop
Security stored procedures	Manage server and database security.	sp_addapprole, sp_addlinkedsvlogin, sp_addlogin, sp_addremotelogin, sp_addrole, sp_addrolemember, sp_addserver, sp_addsrvrolemember, sp_adduser, sp_approlepassword, sp_change_users_login, sp_changedbowner, sp_changeobjectowner, sp_dbfixedrolepermission, sp_defaultdb, sp_defaultlanguage, sp_denylogin, sp_dropalias, sp_dropapprole, sp_droplinkedsvlogin, sp_droplogin,



STORED PROCEDURE TYPE	DESCRIPTION	RELATED SYSTEM STORED PROCEDURES
Security stored procedures ( <i>continued</i> )	Manage server and database security.	sp_dropremotelogin, sp_droprolemember, sp_dropserver, sp_dropsrvrolemember, sp_dropuser, sp_grantdbaccess, sp_grantlogin, sp_helpdbfixedrole, sp_helplinkedsrvlogin, sp_helplogins, sp_helpntgroup, sp_helpremotelogin, sp_helprole, sp_helprolemember, sp_helpprotect, sp_helpsrvrole, sp_helpsrvrolemember, sp_helpuser, sp_MShasdbaccess, sp_password, sp_remoteoption, sp_revokedbaccess, sp_revokelogin, sp_setapprole, sp_srvrolepermission, sp_validatelogins
SQL Mail stored procedures	Perform e-mail operations from SQL Server. (In SQL Server 2008, Database Mail is preferred over SQL Mail.)	sp_processmail, xp_deletemail, xp_findnextmsg, xp_readmail, xp_sendmail, xp_startmail, xp_stopmail
SQL Server Profiler stored procedures	Used by SQL Server Profiler to monitor performance and activity.	sp_trace_create, sp_trace_generateevent, sp_trace_setevent, sp_trace_setfilter, sp_trace_setstatus,
SQL Server Agent stored procedures	Manage scheduled alerts and other SQL Server Agent activities.	sp_add_alert, sp_add_category, sp_add_job, sp_add_jobschedule, sp_add_jobserver, sp_add_jobstep, sp_add_notification, sp_add_operator, sp_add_proxy, sp_add_schedule, sp_add_targetservergroup, sp_add_targetsvrgrp_member, sp_apply_job_to_targets, sp_attach_schedule, sp_cycle_agent_errorlog, sp_cycle_errorlog, sp_delete_alert, sp_delete_category, sp_delete_job, sp_delete_jobschedule, sp_delete_jobserver, sp_delete_jobstep, sp_delete_jobsteplog, sp_delete_notification, sp_delete_operator, sp_delete_proxy, sp_delete_schedule, sp_delete_targetserver, sp_delete_targetservergroup,

STORED PROCEDURE TYPE	DESCRIPTION	RELATED SYSTEM STORED PROCEDURES
SQL Server Agent stored procedures <i>(continued)</i>	Manage scheduled alerts and other SQL Server Agent activities.	sp_delete_targetsvrgrp_member, sp_detach_schedule, sp_enum_login_for_proxy, sp_enum_proxy_for_subsystem, sp_enum_sqlagent_subsystems, sp_grant_login_to_proxy, sp_grant_proxy_to_subsystem, sp_help_alert, sp_help_category, sp_help_downloadlist, sp_help_job, sp_help_jobactivity, sp_help_jobcount, sp_help_jobhistory, sp_help_jobs_in_schedule, sp_help_jobschedule, sp_help_jobserver, sp_help_jobstep, sp_help_jobsteplog, sp_help_notification, sp_help_operator, sp_help_proxy, sp_help_schedule, sp_help_targetserver, sp_help_targetservergroup, sp_manage_jobs_by_login, sp_msx_defect, sp_msx_enlist, sp_msx_get_account, sp_msx_set_account, sp_notify_operator, sp_post_msx_operation, sp_purge_jobhistory, sp_remove_job_from_targets, sp_resync_targetserver, sp_revoke_login_from_proxy, sp_revoke_proxy_from_subsystem, sp_start_job, sp_stop_job, sp_update_alert, sp_update_category, sp_update_job, sp_update_jobschedule, sp_update_jobstep, sp_update_notification, sp_update_operator, sp_update_proxy, sp_update_schedule, sp_update_targetservergroup
XML stored procedures	Manage XML text.	sp_xml_preparedocument, sp_xml_removedocument

# Techniques for Managing SQL Server Configuration Options

---

You can think of configuration options as a set of rules that define how SQL Server is configured and used. Individual server instances can have different configurations, as can the databases they support, the connections made by applications, and any statements or batch programs that are executed.

## Setting Configuration Options

Configuration options can be set for the following:

- **A specific server instance** Server options are also referred to as *instance-wide* options and are set by executing the `sp_configure` stored procedure.
- **A specific database** Database options are also referred to as *database-level* options and are set by executing the `ALTER DATABASE` statement. The database compatibility level can be set by executing the `sp_dbcmtlevel` stored procedure.
- **A specific connection** Connection options are set by the Microsoft OLE DB Provider for SQL Server or the SQL Server ODBC driver properties and by ANSI SET options when a connection is established.
- **A specific statement or batch** Batch-level options are specified with SET statements. Statement-level options are specified in individual T-SQL statements.

Each of these configuration areas can be thought of as a level in the SQL Server configuration hierarchy. When an option is supported at more than one level, the applicable setting is determined by the following precedence order:

1. A server option
2. A database option
3. A connection (ANSI SET) or batch (SET) option
4. A specific statement (HINT) option

**NOTE** The stored procedure `sp_configure` provides the option *user options*, which allows you to change the default values of several SET options. Although *user options* appears to be an instance option, it is a SET option. In previous releases of SQL Server, batch-level options are called *connection-level options*. When you disable multiple active result sets (MARS), batch-level options are considered connection-level options as well.

You use `ALTER DATABASE` to change settings for a database, `sp_configure` to change server-level settings, and the SET statement to change settings that affect only the current session. If there are conflicts among configuration options, the options applied later have precedence over options set previously. For example, connection options have precedence over database and server options.

## Working with SET Options

Typically, SET options are configured by users within a batch or script and they apply until they are reset or the user's session with the server is terminated. SET options can also be configured within a stored procedure or trigger. In that case, the SET options apply until they are reset inside that stored procedure or trigger, or until control returns to the code that invoked the stored procedure or trigger.

SET options are applied at either parse time or execute time. The parse-time options are QUOTED\_IDENTIFIER, PARSEONLY, OFFSETS, and FIPS\_FLAGGER. All other SET options are execute-time options. Parse-time options are applied during parsing as they are encountered. Execute-time options are applied during the execution of the code in which they are specified.

Batch statements are parsed in their entirety prior to execution. This means that control flow statements do not affect parse-time settings. In contrast, both control flow and execution affect whether execute-time options are set. Execute-time options are set only if control is changed to a section of the batch containing execute-time options and the related statements are executed without error. If execution fails before an execute-time option is set or during the processing of the statement that sets the option, the option is not set.

When a user connects to a database, some options might be set to ON automatically. These options can be set through user options, server options, or the ODBC and OLE DB connection properties. If the user changes the SET options within a dynamic SQL batch or script, those changes apply only for the duration of that batch or script.

**NOTE** MARS-enabled connections maintain a list of default SET option values. When a batch or script executes under that connection, the default SET option values are copied to the current request's environment. These values remain in effect unless they are reset within the connection. When the batch or script ends, the execution environment is copied back to the session's default. This ensures that multiple batches executing simultaneously under the same connection run in an isolated SET options environment. However, because the execution environment is copied back to the session default when batch or script execution is complete, the current default environment for a connection depends on the last batch or script that completes execution.

Table 4-4 lists the batch/connection SET options available and indicates the corresponding database and server options supported in SQL Server 2008 as well as the default setting (as applicable). The SET ANSI\_DEFAULTS statement is provided as a shortcut for setting SQL-92 standard options to their default values. The options that reset when this statement is used are as follows: SET ANSI\_NULLS, SET CURSOR\_CLOSE\_ON\_COMMIT, SET ANSI\_NULL\_DFLT\_ON, SET IMPLICIT\_TRANSACTIONS, SET ANSI\_PADDING, SET QUOTED\_IDENTIFIER, and SET ANSI\_WARNINGS.

**NOTE** In a future release of SQL Server, ANSI\_PADDING will always be turned on and you will not be able to turn this setting off. ANSI\_PADDING must be on when you are creating or changing indexes on computer columns or indexed views. The SQL Server Native Client ODBC driver and SQL Server Native Client OLE DB Provider for SQL Server automatically set ANSI\_PADDING to ON when connecting. For connections from DB-Library applications, however, the default for SET ANSI\_PADDING is OFF.

**TABLE 4-4** SET Options

SET OPTION	DATABASE OPTION	SERVER OPTION	DEFAULT SETTING
ANSI_DEFAULTS	None	None	N/A
ANSI_NULL_DFLT_OFF ANSI_NULL_DFLT_ON	ANSI_NULL_DEFAULT	user options default	OFF
ANSI_NULLS	ANSI_NULLS	user options default	OFF
ANSI_PADDING	ANSI_PADDING	user options default	ON
ANSI_WARNINGS	ANSI_WARNINGS	user options default	OFF
ARITHABORT	ARITHABORT	user options default	OFF
ARITHIGNORE	None	user options default	OFF
CONCAT_NULL_ YIELDS_NULL	CONCAT_NULL_ YIELDS_NULL	None	OFF
CONTEXT_INFO	None	None	OFF
CURSOR_CLOSE_ ON_COMMIT	CURSOR_CLOSE_ ON_COMMIT	user options default	OFF
DATEFIRST	None	None	7
DATEFORMAT	None	None	mdy
DEADLOCK_PRIORITY	None	None	NORMAL
FIPS_FLAGGER	None	None	OFF
FMONLY	None	None	OFF
FORCEPLAN	None	None	OFF
IDENTITY_INSERT	None	None	OFF

SET OPTION	DATABASE OPTION	SERVER OPTION	DEFAULT SETTING
IMPLICIT_TRANSACTIONS	None	user options default	OFF
LANGUAGE	None	None	us_english
LOCK_TIMEOUT	None	None	No limit
NOCOUNT	None	user options default	OFF
NOEXEC	None	None	OFF
NUMERIC_ROUNDABORT	NUMERIC_ROUNDABORT	None	OFF
OFFSETS	None	None	OFF
PARSEONLY	None	None	OFF
QUERY_GOVORNOR_COST_LIMIT	None	query governor cost limit	OFF
QUOTED_IDENTIFIER	quoted identifier	user options default	OFF
REMOTE_PROC_TRANSACTIONS	None	None	OFF
ROWCOUNT	None	None	OFF
SHOWPLAN_ALL	None	None	OFF
SHOWPLAN_TEXT	None	None	OFF
SHOWPLAN_XML	None	None	OFF
STATISTICS IO	None	None	OFF
STATISTICS PROFILE	None	None	OFF
STATISTICS TIME	None	None	OFF
STATISTICS XML	None	None	OFF
TEXTSIZE	None	None	OFF
TRANSACTION ISOLATION LEVEL	None	None	N/A
XACT_ABORT	None	None	OFF

## Working with Server Options

Server options can be set using the Properties dialog boxes in SQL Server Management Studio or with the `sp_configure` stored procedure. The difference between these two methods is which options are available to set. Only the most commonly used server configuration options are available through SQL Server Management Studio, but all configuration options are accessible through `sp_configure`. Table 4-5 lists the server options available and provides the corresponding SET options and database options that are supported in SQL Server 2008, as well as the default setting (as applicable).

**TABLE 4-5** Server Options

<b>SERVER OPTION</b>	<b>SET OPTION</b>	<b>DATABASE OPTION</b>	<b>DEFAULT SETTING</b>
allow updates	None	None	0
backup compression default	None	None	0
clr enabled	None	None	0
cross db ownership chaining	None	None	5
default language	None	None	0
filestream access level	None	None	0
max text repl size	None	None	65536
nested triggers	None	None	1
remote access	None	None	1
remote admin connections	None	None	0
remote login timeout	None	None	20
remote proc trans	None	None	0
remote query timeout	None	None	600
server trigger recursion	None	None	0

SERVER OPTION	SET OPTION	DATABASE OPTION	DEFAULT SETTING
show advanced options	None	None	0
user options	ANSI_NULL_DFLT_ON ANSI_NULL_DFLT_OFF	ANSI_NULL_DEFAULT	OFF
	ANSI_NULLS	ANSI_NULLS	OFF
	ANSI_PADDING	ANSI_PADDING	ON
	ANSI_WARNINGS	ANSI_WARNINGS	OFF
	ARITHABORT	ARITHABORT	OFF
	ARITHIGNORE	None	OFF
	CURSOR_CLOSE_ON_COMMIT	CURSOR_CLOSE_ON_COMMIT	OFF
	DISABLE_DEF_CNST_CHK	None	OFF
	IMPLICIT_TRANSACTIONS	None	OFF
	NOCOUNT	None	OFF
	QUOTED_IDENTIFIER	QUOTED_IDENTIFIER	OFF

## Working with Database Options

Database options are set by executing the ALTER DATABASE statement. In new SQL Server installations, the settings in the *model* and *master* databases are the same. When you create new databases, the default database options for those databases are taken from the *model* database. Whenever you change a database option, the Database Engine recompiles everything in the database cache. Table 4-6 lists the standard database options that are available and provides the corresponding SET and server options supported in SQL Server 2008, as well as the default setting (as applicable).

**NOTE** Microsoft recommends that ANSI\_PADDING always be set to ON to avoid problems with future versions of SQL Server. Note that AUTO\_UPDATE\_STATISTICS\_ASYNC has no effect unless you set AUTO\_UPDATE\_STATISTICS to ON.



**TABLE 4-6** Database Options

<b>DATABASE OPTION</b>	<b>SET OPTION</b>	<b>SERVER OPTION</b>	<b>DEFAULT SETTING</b>
ANSI_NULL_DEFAULT	ANSI_NULL_DFLT_ON ANSI_NULL_DFLT_OFF	user options default	OFF
ANSI_NULLS	ANSI_NULLS	user options default	OFF
ANSI_PADDING	ANSI_PADDING	user options default	ON
ANSI_WARNINGS	ANSI_WARNINGS	user options default	OFF
AUTO_CLOSE	None	None	OFF
AUTO_CREATE_STATISTICS	None	None	ON
AUTO_SHRINK	None	None	OFF
AUTO_UPDATE_STATISTICS	None	None	ON
AUTO_UPDATE_STATISTICS_ASYNC	None	None	OFF
CONCAT_NULL_YIELDS_NULL	CONCAT_NULL_YIELDS_NULL	None	OFF
CURSOR_CLOSE_ON_COMMIT	CURSOR_CLOSE_ON_COMMIT	user options default	OFF
CURSOR_DEFAULT	None	None	GLOBAL
MERGE PUBLISH	None	None	FALSE
PUBLISHED	None	None	FALSE
QUOTED_IDENTIFIER	QUOTED_IDENTIFIER	user options default	ON
READ_ONLY	None	None	FALSE
RECOVERY BULK_LOGGED	None	None	FALSE
RECOVERY SIMPLE	None	None	TRUE
RECURSIVE_TRIGGERS	None	None	FALSE
RESTRICTED_USER	None	None	FALSE
SINGLE_USER	None	None	FALSE
SUBSCRIBED	None	None	TRUE
TORN_PAGE_DETECTION	None	None	TRUE

# Managing Database Compatibility

By default, when you create a new database in SQL Server 2008, the default compatibility level is 100 (unless the *model* database has a lower compatibility level). When a database is upgraded to SQL Server 2008, pre-SQL Server 2000 databases are upgraded to compatibility level 80. All other databases retain their existing compatibility level:

- 80 for SQL Server 2000 compatibility level
- 90 for SQL Server 2005 compatibility level
- 100 for SQL Server 2008 compatibility level

Although the compatibility level of the *master* database cannot be modified, the compatibility level setting of the *model* database can be changed. This flexibility allows you to create new databases with a nondefault compatibility level. To change the compatibility level, you can use the ALTER DATABASE statement.

The ALTER DATABASE statement allows you to set the database compatibility level for a specific database. The ALTER DATABASE statement sets certain database behaviors to be compatible with the specified earlier version of SQL Server. The following example changes the compatibility level of the Personnel database to SQL Server 2005:

### T-SQL

```
ALTER DATABASE Personnel  
SET COMPATIBILITY_LEVEL = 90;  
GO
```

### PowerShell

```
Invoke-Sqlcmd -Query "ALTER DATABASE Personnel;  
SET COMPATIBILITY_LEVEL = 90;" -ServerInstance "DbServer17\Cwhouse"
```

When there are possible conflicts between compatibility (and other) settings, it is important to know which database context is being used. Generally speaking, the current database context is the database defined by the USE statement if the statement is in a batch or script, or it is the database that contains the stored procedure if the statement is in a stored procedure applied to that statement. When a stored procedure is executed from a batch or another stored procedure, it is executed under the option settings of the database in which it is stored. For example, when a stored procedure in the Support database calls a stored procedure in the Personnel database, the Support procedure is executed under the compatibility level setting of the Support database and the Personnel procedure is executed under the compatibility level setting of the Personnel database.

## Configuring SQL Server with Stored Procedures

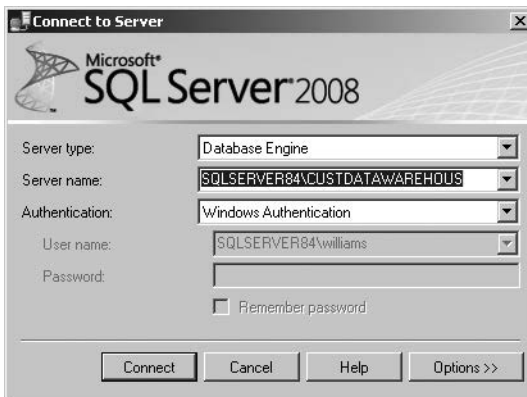
You can configure many areas of SQL Server using the SQL Server Properties dialog box, which is discussed in Chapter 5. As you have learned in this chapter, you can also configure SQL Server with stored procedures, such as `sp_configure`. You execute stored procedures and other queries in SQL Server Management Studio. SQL Server Management Studio has a built-in client tool that sends commands to a SQL Server instance, which in turn parses, compiles, and executes the commands.

The following sections explain how to use SQL Server Management Studio and stored procedures to configure SQL Server. You can find more detailed coverage of SQL Server Management Studio in other chapters.

## Using SQL Server Management Studio for Queries

You can start SQL Server Management Studio and access the built-in query client by completing the following steps:

1. Click Start, Programs or All Programs, Microsoft SQL Server 2008, SQL Server Management Studio. Or click Start, type **ssms** in the Search box, and then press Enter.
2. In the Connect To Server dialog box, shown in Figure 4-1, use the Server Type list to select the database component you want to connect to, such as Database Engine.



**FIGURE 4-1** The Connect To Server dialog box

3. In the Server Name field, type the name of the server on which SQL Server is running, such as `CorpSvr04`.

**NOTE** You can connect only to registered servers. If the instance of SQL Server you want to work with is not registered, you need to register the server before you can work with it. See “Managing Servers” in Chapter 5 for details.

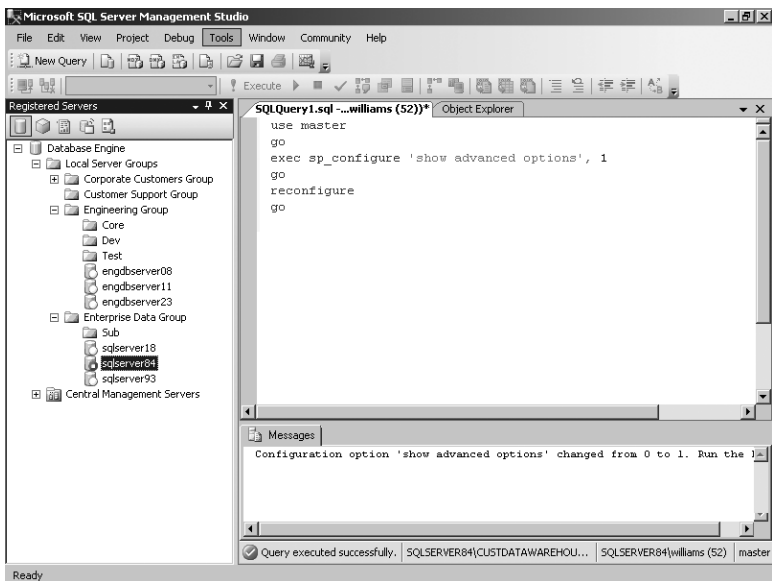
4. Use the Authentication list to specify the authentication type as Windows authentication or SQL Server authentication (based on the allowed authentication types when you installed the server). Provide a Windows user name or SQL Server login ID and password as necessary.
  - **Windows Authentication** Uses your current domain account and password to establish the database connection. This option works only if Windows authentication is enabled and you have appropriate privileges.
  - **SQL Server Authentication** Allows you to specify a SQL Server login ID and password.
5. Click Connect. You connect to the default database (unless you have configured another default previously). To change the database you connect to, click the Options button prior to clicking Connect, select the Connection Properties tab, and then use the Connect To Database list to select the database you want to connect to.
6. In SQL Server Management Studio, you can connect to the database you previously selected by clicking New Query on the toolbar, and then skip the remaining steps. To connect to a different database, click File, click New, and then select the query type, such as Database Engine Query.
7. In the Connect To Database Engine dialog box, specify the server name or select Browse For More in the drop-down list to search for all computers that are running SQL Server within an Active Directory forest as well as the different instances running on a particular server.
8. Specify the authentication technique to use. Click Connect. As before, you connect to the default database (unless you have configured another default previously). To change the database to which you connect, click the Options button, select the Connection Properties tab, and then use the Connect To Database list to select the database you want to connect to.

If you are working with an active database in SQL Server Management Studio and have already authenticated the connection, you can automatically connect to the currently selected database server instance and use your current authentication information to log on. To do this, right-click the database in Object Explorer view in SQL Server Management Studio, and then select New Query.

## Executing Queries and Changing Settings

The query window in SQL Server Management Studio is normally divided into two panes. (See Figure 4-2.) The top pane allows you to enter queries. The lower pane displays results.

If you do not see a separate pane in the lower part of the window, don't worry. It appears automatically when you execute a query. You can also set the pane to open by default by selecting the Show Results Pane option on the Window menu.



**FIGURE 4-2** Executing queries in SQL Server Management Studio.

As you know, you can use `sp_configure` to view and change SQL Server configuration settings. Two types of configuration settings are available: those that are dynamic and those that are not. In this instance, a dynamic setting is one that you can change without having to stop and restart SQL Server. To execute `sp_configure` or other types of queries, type a command in the top pane and then click the Execute button on the toolbar (the red exclamation point). You can also execute commands by using these key sequences:

- F5
- Ctrl+E
- Alt+X

**NOTE** By default, all users have execute permissions on `sp_configure` so that they can view settings. However, only users with the Alter Settings server-level permission can use `sp_configure` to change configuration options. By default, only members of the `sysadmin` and `serveradmin` fixed server roles have this permission. As with `sp_configure`, only users with the Alter Settings server-level permission can execute the `RECONFIGURE` or `RECONFIGURE WITH OVERRIDE` command.

Whenever you use `sp_configure` to modify settings, the changes do not take place until you also execute the `RECONFIGURE` command. You can change some highly risky settings by using only the `RECONFIGURE WITH OVERRIDE` command. Additionally, `sp_configure` settings are divided into two categories: standard and advanced. You can execute standard commands at any time, but you can execute

advanced commands only when Show Advanced Options is set to 1. With this setting in effect, you can modify both standard and advanced settings. Follow this procedure to allow modification of advanced settings:

1. In SQL Server Management Studio, type the following:

```
exec sp_configure 'show advanced options', 1
go
reconfigure
go
```

**TIP** You can disable advanced options later by setting the value to 0.

2. Execute the commands by pressing Ctrl+E.
3. Clear the query window.
4. Now type one sp\_configure command for each option you want to change.
5. Type **reconfigure** (or **reconfigure with override**).
6. Type **go**.
7. Execute the commands by pressing Ctrl+E.
8. If you changed any nondynamic settings, stop and restart the server. (See Table 4-7 and Table 4-8 for details.)

## Checking and Setting Configuration Parameters

Table 4-7 provides a summary of the standard configuration parameters. The parameters are listed in alphabetical order, with the minimum, maximum, and default values shown. The dynamic parameter column tells you whether the setting is dynamic. If you see an “N” in this column, you need to stop and restart the server to enforce changes.

**TABLE 4-7** Quick Reference Summary for Standard Configuration Parameters

CONFIGURATION OPTION	MINIMUM VALUE	MAXIMUM VALUE	DEFAULT VALUE	DYNAMIC YES/NO
allow updates	0	1	0	Y
backup compression default	0	1	0	Y
clr enabled	0	1	0	Y
cross db ownership chaining	0	1	0	Y
default language	0	9999	0	Y
filestream access level	0	2	0	Y
max text repl size	-1	2147483647	65536	Y
nested triggers	0	1	1	Y

CONFIGURATION OPTION	MINIMUM VALUE	MAXIMUM VALUE	DEFAULT VALUE	DYNAMIC YES/NO
remote access	0	1	1	N
remote admin connections	0	1	0	Y
remote login timeout(s)	0	2147483647	20	Y
remote proc trans	0	1	0	Y
remote query timeout(s)	0	2147483647	600	Y
server trigger recursion	0	1	1	Y
show advanced options	0	1	0	Y
user options	0	32767	0	Y

Table 4-8 provides a summary of advanced configuration parameters. To view or change these parameters, you have to set the parameter Show Advanced Options to 1. Self-configuring options have an asterisk (\*) after their name. With max worker threads, 1,024 is the maximum recommended for 32-bit operating systems. The default value zero (0) autoconfigures by using the following formula:  $256 + (\text{number of processors} - 4) * 8$ . Note also that you cannot change some advanced options, although you can view them.

**NOTE** If there are differences between the values used in R1 and R2, Table 4-8 shows the R1 value followed by the R2 value. For example, “0; 1” means the R1 value is 0 and the R2 value is 1.

**TABLE 4-8** Quick Reference Summary for Advanced Configuration Parameters

CONFIGURATION OPTION	MINIMUM VALUE	MAXIMUM VALUE	DEFAULT VALUE	DYNAMIC YES/NO
ad hoc distributed queries	0	1	0	Y
access check cache bucket count	0	65536	0	Y
access check cache quota	0	2147483647	0	Y
affinity I/O mask	-2147483648	2147483647	0	N
affinity64 I/O mask	-2147483648	2147483647	0	Y
affinity mask	-2147483648	2147483647	0	Y
affinity64 mask	-2147483648	2147483647	0	Y
Agent XPs	0	1	0; 1	Y
awe enabled	0	1	0	N

CONFIGURATION OPTION	MINIMUM VALUE	MAXIMUM VALUE	DEFAULT VALUE	DYNAMIC YES/NO
blocked process threshold(s)	0	86400	0	Y
c2 audit mode	0	1	0	N
common criteria compliance enabled	0	1	0	N
cost threshold for parallelism	0	32767	5	Y
cursor threshold	-1	2147483647	-1	Y
Database Mail XPs	0	1	0	Y
default full-text language	0	2147483647	1033	Y
default trace enabled	0	1	1	Y
disallow results from triggers	0	1	0	Y
EKM provider enabled	0	1	0	N
fill factor (%)	0	100	0	N
ft crawl bandwidth (max)	0	32767	100	Y
ft crawl bandwidth (min)	0	32767	0	Y
ft notify bandwidth (max)	0	32767	100	Y
ft notify bandwidth (min)	0	32767	0	Y
index create memory (kb)*	704	2147483647	0	Y
in-doubt xact resolution	0	2	0	Y
lightweight pooling	0	1	0	N
locks*	5000	2147483647	0	N
max degree of parallelism	0	64; 1024	0	Y
max full-text crawl range	0	256	4	Y
max server memory (mb)*	16	2147483647	2147483647	N
max text repl size (b)	-1	2147483647	65536	Y
max worker threads	128	32767	0	N
media retention	0	365	0	N
min memory per query (kb)	512	2147483647	1024	Y



CONFIGURATION OPTION	MINIMUM VALUE	MAXIMUM VALUE	DEFAULT VALUE	DYNAMIC YES/NO
min server memory (mb)*	0	2147483647	0	Y
network packet size (b)	512	32767	4096	Y
Ole Automation Procedures	0	1	0	Y
open objects	0	2147483647	0	N
optimize for ad hoc workloads	0	1	0	Y
ph_timeout(s)	1	3600	60	Y
precompute rank	0	1	0	Y
priority boost	0	1	0	N
query governor cost limit	0	2147483647	0	Y
query wait(s)	-1	2147483647	-1	Y
recovery interval (min)*	0	32767	0	Y
Replication XPs	0	1	0	Y
scan for startup procs	0	1	0	N
set working set size	0	1	0	N
SMO and DMO XPs	0	1	1	Y
SQL Mail XPs	0	1	0	Y
transform noise words	0	1	0	Y
two digit year cutoff	1753	9999	2049	Y
user connections*	0	32767	0	N
xp_cmdshell	0	1	0	Y

You can view the current settings of all configuration options by executing the following query:

**T-SQL**

```
exec sp_configure
go
```

**PowerShell**

```
Invoke-Sqlcmd -Query "exec sp_configure"
-ServerInstance "DataServer91\CorpServices"
```

**NOTE** Show Advanced Options must be set to 1 to see advanced options.

To view the current setting of a configuration option, execute the following query:

**T-SQL**

```
exec sp_configure 'optionName'  
go
```

**PowerShell**

```
Invoke-Sqlcmd -Query "exec sp_configure 'optionName'"  
-ServerInstance "DataServer91\CorpServices"
```

where *optionName* is the name of the option you want to examine, such as:

**T-SQL**

```
exec sp_configure 'allow updates'  
go
```

**PowerShell**

```
Invoke-Sqlcmd -Query "exec sp_configure 'allow updates'"  
-ServerInstance "DataServer91\CorpServices"
```

To change the value of a setting, execute the following query:

**T-SQL**

```
exec sp_configure 'optionName', newValue  
go  
reconfigure with override  
go
```

**PowerShell**

```
Invoke-Sqlcmd -Query "exec sp_configure 'optionName', newValue;  
reconfigure with override" -ServerInstance "DataServer91\CorpServices"
```

where *optionName* is the name of the option you want to examine, and *newValue* is the new value for this option, such as:

**T-SQL**

```
exec sp_configure 'allow updates', 1  
go  
reconfigure with override  
go
```

**PowerShell**

```
Invoke-Sqlcmd -Query "exec sp_configure 'allow updates', 1;  
reconfigure with override" -ServerInstance "DataServer91\CorpServices"
```

**NOTE** You do not always have to use RECONFIGURE WITH OVERRIDE. This value is required only when making ad hoc updates and setting an option to a value that is not generally recommended. Keep in mind that some setting changes are applied only when you restart the SQL Server instance.

## Changing Settings with ALTER DATABASE

For SQL Server 2008 and later releases, the ALTER DATABASE statement replaces the sp\_dboption stored procedure as the preferred way to change database settings. To change database settings, you must be a member of a role granted the ALTER permission on the database or be explicitly assigned this permission. When you execute an ALTER DATABASE statement, a checkpoint occurs in the database for which the option was changed, and this causes the change to take effect immediately. Table 4-9 provides an overview of the database options you can set with ALTER DATABASE.

**TABLE 4-9** Quick Reference Summary for Database Options

OPTION	WHEN TRUE OR SET TO VALUE	ACCEPTED VALUES
ANSI_NULL_DEFAULT	CREATE TABLE uses SQL-92 rules to determine if a column allows null values.	ON   OFF
ANSI_NULLS	All comparisons to a null value evaluate to UNKNOWN. (When OFF, non-UNICODE values evaluate to TRUE if both values are NULL.)	ON   OFF
ANSI_PADDING	Trailing blanks are inserted into character values, and trailing zeros are inserted into binary values to pad to the length of the column.	ON   OFF
ANSI_WARNINGS	Errors or warnings are issued when conditions such as “divide by zero” occur.	ON   OFF
ARITHABORT	An overflow or divide-by-zero error causes the query or batch to terminate. If the error occurs in a transaction, the transaction is rolled back. (When this option is set to OFF, a warning message is displayed, but execution continues as if no error occurred.)	ON   OFF

OPTION	WHEN TRUE OR SET TO VALUE	ACCEPTED VALUES
AUTO_CLEANUP	Change tracking information is automatically removed after the retention period.	ON   OFF
AUTO_CREATE_STATISTICS	Any missing statistics needed for query optimization are automatically generated.	ON   OFF
AUTO_UPDATE_STATISTICS	Any out-of-date statistics needed for query optimization are automatically generated.	ON   OFF
AUTOCLOSE	After the last user logs off, the database is shut down cleanly and its resources are freed.	ON   OFF
AUTOSHRINK	Automatic periodic shrinking is enabled for the database.	ON   OFF
CHANGE_RETENTION	When change tracking is set to ON, this option sets the retention period for change tracking information.	<i>RetPeriod</i> {DAYS   HOURS   MINUTES}
CHANGE_TRACKING	Turns on and enables change tracking.	ON   OFF
COMPATIBILITY_LEVEL	Sets the database compatibility level.	80   90   100
CONCAT_NULL_YIELDS_NULL	If either operand in a concatenation operation is NULL, the result is NULL.	ON   OFF
CURSOR_CLOSE_ON_COMMIT	Any cursors that are open when a transaction is committed or rolled back are closed. (When this option is set to OFF, cursors remain open when a transaction is committed. Rolling back a transaction closes any cursors except those defined as INSENSITIVE or STATIC.)	ON   OFF
CURSOR_DEFAULT	Cursor declarations default to LOCAL.	LOCAL   GLOBAL
DATE_CORRELATION_OPTIMIZATION	SQL Server maintains correlation statistics between tables in a database that are linked by a foreign key constraint and have datetime columns.	ON   OFF

OPTION	WHEN TRUE OR SET TO VALUE	ACCEPTED VALUES
DB_CHAINING	The database can be the source or target of a cross-database ownership chain. The instance of SQL Server recognizes this setting only when the cross db ownership chaining server option is 0 (OFF). Otherwise, all user databases can participate in cross-database ownership chains, regardless of the value of this option.	ON   OFF
EMERGENCY	Marks the database as read-only, disables logging, and allows access only by members of the sysadmin role.	EMERGENCY
ENCRYPTION	Transparent data encryption is turned on for the database.	ON   OFF
MULTI_USER	Multiple users can access the database.	MULTI_USER
NUMERIC_ROUNDABORT	An error is generated when loss of precision occurs in an expression. (When this option is set to OFF, losses of precision do not generate error messages and the result is rounded to the precision of the column or variable storing the result.)	ON   OFF
OFFLINE	The database is offline. (Otherwise, the database is online.)	OFFLINE
ONLINE	The database is online and available for use.	ONLINE
PAGE_VERIFY	SQL Server can discover damaged database pages.	CHECKSUM   TORN_PAGE_DETECTION   NONE
PARAMETERIZATION	SQL Server parameterizes all queries in the database.	SIMPLE   FORCED
QUOTED_IDENTIFIER	Double quotation marks can be used to enclose delimited identifiers.	ON   OFF

OPTION	WHEN TRUE OR SET TO VALUE	ACCEPTED VALUES
READ_ONLY	The database is set to read-only (but can be deleted by using the DROP DATABASE statement). The database cannot be in use when this option is set (except for the <i>master</i> database).	READ_ONLY
RECOVERY	Causes the recovery model to be reset. SIMPLE allows a checkpoint to truncate the inactive part of the log.	FULL   BULK_LOGGED   SIMPLE
RECURSIVE_TRIGGERS	Enables recursive firing of triggers. (When this option is set to OFF, it prevents direct recursion but not indirect recursion. To disable indirect recursion, set the nested triggers server option to 0 using sp_configure.)	ON   OFF
RESTRICTED_USER	Only the database owner can use the database.	RESTRICTED_USER
SINGLE_USER	Only one user at a time can access the database.	SINGLE_USER
TORN_PAGE_DETECTION	Allows incomplete pages to be detected. (This option is being replaced by PAGE_VERIFY.)	ON   OFF
TRUSTWORTHY	Database modules that use impersonation can access resources outside the database.	ON   OFF

Most of the options listed accept a value of ON or OFF, which is used to set the state of the option. For example, you can enable transparent data encryption on the CustomerSupport database by using the following command:

#### T-SQL

```
USE master;
GO
ALTER DATABASE CustomerSupport
SET ENCRYPTION ON;
GO
```

#### PowerShell

```
Invoke-Sqlcmd -Query "USE master; ALTER DATABASE CustomerSupport
SET ENCRYPTION ON;" -ServerInstance "DataServer91\CorpServices"
```

Some options explicitly set a specific state. For example, if no users are currently connected to the CustomerSupport database, you could set the database to read-only by using the following command:

**T-SQL**

```
USE master;  
GO  
ALTER DATABASE CustomerSupport  
SET READ_ONLY;  
GO
```

**PowerShell**

```
Invoke-Sqlcmd -Query "USE master; ALTER DATABASE CustomerSupport  
SET READ_ONLY;" -ServerInstance "DataServer91\CorpServices"
```





# Index

## A

- abbreviations (collation styles), 249
- access. *See also* permissions
  - access control lists (ACLs), 511
  - Analysis Services and, 60–62
  - authentication modes, 301–302
  - central management, 173
  - client access, 57
  - controlling, 329–336
  - Database Engine items, 58–60
  - database state and, 264–265
  - deployment planning, 37–39
  - external options, 266–267
  - granting or denying, 323–324
  - logins, 317–326
  - management data warehouses, 544–545
  - Performance Monitor, 536
  - publication access, 510
  - remote access, 57
  - Reporting Services, 62
  - restored databases, 589
  - revoking, 329
  - roles, 312–316, 326–329
  - security goals, 291
  - special-purpose logins, 302–305
- access control lists (ACLs), 511
- Access databases, 430, 433, 437, 472
- account provisioning, 49–50, 51
- accounts
  - copying, 633–636
  - database mirroring and, 673
  - install configuration, 47
  - log shipping configuration, 655
  - mail, 606, 611–612
  - process, 507, 510, 515
  - replication agents and, 510
  - service, 10–11
  - special-purpose, 302–305
  - startup service, 69
  - user groups and, 65–67
- Active Directory, 95, 228
- Active Directory Helper, 9, 63, 66
- Active File/Table view, 360
- ActiveX scripts, 165, 434, 625
- activity
  - monitoring. *See* monitoring
  - performance metrics, 522
- Activity Monitor, 151–157, 159, 522, 523
- ad hoc connections, 58
- ad hoc distributed queries, 113–115
- ad hoc policy evaluation, 9, 63, 66
- ad hoc reports, 62
- Address Windowing Extensions (AWE), 203, 206–207
- administration
  - adding, deleting, and manipulating databases, 275–280
  - altering databases, 251–270
  - automating tasks, 603
  - backing up, 572–583
  - backup planning, 557–566
  - basics, 238–245
  - command-line tools, 16–21
  - copying or moving databases, 280–286
  - creating databases, 245–251
  - database mirroring, 675–678
  - database size, 270–275
  - database templates, 288
  - dedicated connections, 37–39
  - encryption, 288–290
  - files and logs, 233–238, 270–275
  - graphical tools, 12–15
  - maintenance, 603, 641–652
  - multiserver administration, 637–640
  - Policy-Based Management, 161–164
  - secondary data and log files, 287
  - server-to-server tasks, 632–637
  - system stored procedures, 93–100
- administrator accounts, 10–11, 48, 303
- advanced configuration parameters, 113–115
- affinity masks, 167, 212
- affinity settings (processors), 210–211
- Agent History Clean Up job, 482
- agents, 481–483, 524. *See also* replication agents; *names of specific agents*
- aggregate functions, 399
- aggregates, securable, 294
- alerts. *See also* errors and error messages
  - ANSI warning options, 117, 260
  - central administration, 637–638
  - configuring, 604
  - copying, 636–637
  - Database Mail, 606–612
  - database mirroring, 676
  - default alerts, 615–616
  - deleting, enabling, or disabling, 618–619
  - error numbers, 529, 617
  - error severity levels, 617
  - Event Viewer display, 531
  - job execution, 629–630
  - log shipping, 654, 658–659
  - performance counters, 543–544
  - policy compliance, 188
  - policy events, 163

- alerts, *continued*
  - responses, 617–618
  - SQL Server Agent, 142, 612, 613
  - triggers, 613, 616
- aligned partitioning, 413
- ALTER DATABASE statement
  - ANSI compliance, 259–261
  - automatic options, 257–259
  - changing settings, 117–121, 251–257
  - compatibility levels, 108
  - configuration options, 101, 106–107
  - cross-database chaining, 266–267
  - cursor options, 263–264
  - database state, 264–265
  - manipulating databases, 275–280
  - moving databases, 284–285
  - online, offline, or emergency modes, 266
  - parameterization, 261–263
  - recovery, logging, and I/O error checking, 267–269
  - syntax, 252–257
  - transaction log recovery issues, 582
  - user access, 264–265
- Analysis Services, 1, 9
- AWE, 206–207
  - configuring instances, 51
  - data provider connections, 438
  - Deployment Wizard, 15
  - importing data to or from, 433
  - importing policies, 171–172
  - initializing, 64
  - installing, 45
  - job steps, 625
  - managing access, 60–62
  - multidimensional databases and data mining, 30–32
  - server groups, 127
  - user group, 66
- Analysis Services Deployment Wizard, 15
- anonymous authentication, 608
- anonymous connections, 61
- anonymous subscriptions, 509, 512–513
- ANSI compliance, 259–261
- ANSI nulls. *See also* nulls
  - ALTER DATABASE options, 117
  - ANSI compliance settings, 259
  - SET options, 102, 103, 106, 218, 219, 400
  - SQL-92 settings, 464
- ANSI padding
  - ALTER DATABASE options, 117
  - ANSI compliance settings, 259
  - script settings, 634
  - SET options, 102, 103, 106, 218, 219, 400
- ANSI SET options, 101
- ANSI SQL (SQL-92), 263, 464
- ANSI warnings
  - ALTER DATABASE options, 117
  - ANSI compliance settings, 260
  - distributed queries, 464
  - SET options, 102, 103, 106, 107, 218, 219
- APIs, 93–94
- AppleTalk, 77
- application database roles, 333–334
- application logs, 528–532
- application roles, 292, 293, 294, 306, 314
- applications, repointing, 667
- archive logs, 533, 534
- arithmetic operators, 262
- articles, 481, 484, 501, 505, 509
- ASCII text, 458
- assemblies, 294, 306
- asymmetric keys, 288, 293, 294, 306
- asynchronous mirroring, 669, 679–680
- attachable *subscription* databases, 509
- attaching databases, 277–280
- attachments (mail), 610
- auditing, 200–203, 531
- authentication
  - anonymous connections, 61
  - configuring, 201–202
  - Database Engine, 49–50
  - distributors, 493
  - exported registration details, 135
  - log shipping failovers and, 666
  - logins, 10
  - mixed security modes, 301–302
  - packages, 452
  - policy evaluation and, 186
  - policy settings, 168
  - PowerShell and, 138, 141
  - simulating with traces, 551–555
  - SMTP and, 606, 608
  - types of, 192
  - Web synchronization, 509
  - Windows vs. SQL Server, 110
- authentication modes, 301–302
- Auto Close option, 257–258
- Auto Create Statistics option, 258
- autogrowing, 166, 247–248, 270
- auto-shrinking, 166, 258, 271–275
- auto-updating statistics, 258
- auto-closing databases, 165
- autoloader tape systems, 567
- automated recovery process, 578
- automated Server Agent tasks, 142
- automatic failover, 674
- automatic server startup, 143–144
- automatic service restart, 615
- automation, 604–605
  - alerts, 615–619
  - copying items to servers, 636–637
  - Database Mail, 606–612
  - failover, 674
  - operators, 619–621
  - recovery, 578
  - scheduling jobs, 621–632
  - server startup, 143–144
  - server-to-server tasks, 632–637
  - service restarts, 615
  - single-user mode, 605
  - SQL Server Agent, 142, 612–615
- availability, 36–37, 669, 674
- AWE (Address Windowing Extensions), 203, 206–207

**B**

## backing up and backup sets

- Analysis Services and, 32
- assessing needs, 558–559
- automated recovery, 578
- backup devices, 234, 570–572
- backup intervals, 664
- backup sets, 572, 575–577, 587
- backup types, 559–560, 576
- compression, 228, 565–566, 575, 576
- creating backups, 572–577
- database mirroring, 562
- encryption and, 290
- erasing backups, 575
- expiration intervals, 574
- hardware and media, 566–568
- histories, backup, 588
- initializing replications, 483, 489
- large databases, 564–565
- log shipping databases, 661–663
- maintenance plans, 642, 644
- new backup sets, 573–575
- offsite backups, 559
- parallel backup/restore, 564–565
- parallel striped backups, 577–578
- permissions, 312
- planning backup and recovery, 557, 568–570
- policies, 165, 167
- recovering and restoring, 583–601
- recovery models, 561–562
- replicated databases, 563–564
- restoring transaction logs, 586
- retention, 227, 497, 513, 563, 658, 662
- rotation schedules, 568
- standby servers, 562, 594–596
- subsystems, 32

time-outs, 226–227

transaction logs, 582–583

T-SQL statements, 578–582  
verification, 575

BACKUP commands, 578–582

Backup Database dialog box, 572

backup devices, 234, 566–568

backup folders, 654, 656

backup jobs, 654, 658

backup sets, 572, 575, 576–577, 587

baseline memory, 205

baseline performance metrics, 522

batch scripts, 263–264

batch size limitations, 235

batch-level configuration, 101, 102–104

BCM (Bulk Changed Map), 355

BCP (bulk copy program), 16, 19, 453–454

- applications for DACs, 198
- exporting data, 460
- importing data, 458–460
- parameters, 455, 458
- permissions and modes, 457–458
- scripts, 460–461
- switches, 454, 456–457
- syntax, 454–457
- versions of, 459

binary data types, 375, 376–377

binary large objects (BLOBs), 50, 70–73, 592

binary literals, 262

bindings, 294, 309, 361, 390, 421, 634

BitLocker Drive Encryption, 39

BLOBs (binary large objects), 50, 70–73, 592

Blocked By values, 153, 157

blocked process thresholds, 167

blocked resources, 154–156

blocking connections, 157–158

Blocking information, 157

bookmarks, 397

breakpoints, traces, 554

broker priorities, 244

building queries, 445–447

built-in permissions, 297–298

Bulk Changed Map (BCM), 355

bulk copy program. *See* BCP

bulk logging, 267

bulk-logged recovery, 462, 561, 653, 655

Business Intelligence Development Studio, 15, 45

**C**

C2 audit logging, 201–202

cache, 227

cache-hit ratios, 204, 356

candidate keys, 417

catalog views, 86, 87–93, 95, 424–425, 652, 675

categories

- jobs, 631–632
- policies, 162, 163, 182, 183–185

central log shipping servers, 659–660

central management servers

- alert and job schedules, 637–640

- configuring, 172–175
- policy management, 164

- querying multiple servers, 176

- subordinate servers/groups, 174–175

central publisher or subscriber replication models, 486

certificates, 10, 81, 242, 288, 293, 294, 307

change tracking, 30, 87, 95, 240, 634

Change Tracking view, 87

character data types, 376–377

character mode (BCP), 458

characters

- case, 419
- data types, 374–375

- importing or exporting text files, 442

- limitations, 235

- schema names, 349

- table names, 358

check constraints, 415, 418–420, 634

checklists, maintenance, 641–642

checkpoint LSNs, 588

checkpoints, 148, 227, 268

checksum, 166, 268, 575

- chip information, 194
- classes, auditing security, 201
- cleanup agents, 482
- cleanup tasks, 642
- client configuration, 82–84
- Client Network Utility. *See* SQL Server Configuration Manager
- closing
  - cursors, 263
  - databases, 165, 257–258
- CLR (common language runtime), 58, 61, 89, 395
- CLR Assembly view, 87
- Cluster service, 36–37
- clustered indexes, 395–396
  - compression and, 425, 426–427
  - indexing views, 387
  - limitations, 235
  - selecting columns for, 399
- clustering
  - backups and, 559
  - Database Engine objects and, 236
- CmdExec commands, 165, 625
- cmdlets, 21–26, 137–139, 238
- code pages, 593
- cold standby servers, 562, 594–595, 654
- collation
  - changing, 230–231
  - columns, 362
  - designators and comparison styles, 49, 249
  - distributed queries, 464
  - linked servers, 470, 476
  - policies, 166
  - remote collation, 471
  - replicating settings, 481
  - restored databases, 593
  - scripts and, 634
  - settings, 48, 362
- Collation Designator, 49
- collections, 292
- Column Properties view, 360
- column sets, 380–381
- columns
  - adding or deleting, 370
  - BCP import or export, 459
  - column sets, 380–381
  - computed, 399–400
  - constraints and rules, 415–420
  - globally unique IDs, 362
  - identity columns, 416–417
  - import/export mapping, 447–450
  - indexing, 398–399, 402
  - limitations, 235, 359
  - maximum number, 414
  - modifying, 370
  - in nonclustered indexes, 395
  - nulls in, 379–380
  - number allowed in views, 388
  - properties, 360, 361–362
  - in publications, 502, 505
  - replaying traces and, 552
  - row-overflow storage, 237–238
  - sparse, 361–362, 380–381
  - traces, 549
  - variable length, 354
- COM (component object model), 61
- command-line tools
  - administration, 16–21, 137–141
  - management, 148–149
  - running SQL Server, 149–150
- commas, 442
- committing transactions, 467–468, 484–485, 666–667
- Common Criteria Evaluation Level 4 (EAL4+) requirements, 202–203
- common language runtime (CLR), 58, 61, 89, 395
- Compact edition (SQL Server), 7, 127, 505
- comparison operators, 398
- comparison styles (collation), 249
- compatibility levels, 108
- compatibility views, 86
- completed jobs, 630–631
- compliance, policies, 185–188
- component object model (COM), 61
- compression
  - backup media, 228
  - backup planning, 559, 565–566
  - backup settings, 575
  - copying settings, 634
  - database size and, 271–275
  - indexes, 425–428
  - log shipping settings and, 659
  - media set limitations, 576
  - partitioned tables, 357–358
  - partitions, 425–428
  - ratios, 565
  - tables, 425–428
  - types of, 426–427
- computed columns or views, 399–400
- Computer Management, 144, 147
- concatenate nulls options, 103, 107, 118, 219, 260
- conditions, 162, 172, 178–180, 182
- configuration. *See also* Database Engine Tuning Advisor
  - access, 58–62
  - Active Directory information, 228
  - ALTER DATABASE, 117–121
  - auditing, 201–203
  - authentication, 200–201
  - automation tools, 604–605
  - backup compression, 228
  - cache flushing, 227
  - catalog view types, 87–93
  - compatibility levels, 108
  - configuration servers, 163
  - connections, 76–77, 216–219, 220
  - Customer Feedback Reporting, 73–74
  - data collection, 542
  - Database Mail, 606–611
  - database mirroring, 670–675, 681
  - databases, 106–107, 117–121
  - distributors, 490–494
  - DTC, 142–143
  - dump directories, 73–74
  - dynamic settings, 111
  - error reporting, 73–74
  - event logs, 533
  - facets, 176–178
  - fibers, 214–215
  - file streaming, 70–73, 221–222
  - index fill factor, 225–226
  - I/O subsystems, 34–36
  - job schedules, 627–629

- language settings, 222
  - linked servers, 469–476
  - log shipping, 655–656, 661–663, 681
  - maintenance tools, 604–605
  - management servers, 172–175
  - Management Studio, 109–112, 191–194
  - master and target servers, 639–640
  - memory usage, 203–209
  - monitoring changes, 542
  - nested triggers, 223
  - networks, 75
  - parallel processing, 210–214
  - parameter settings, 112–117
  - planning deployments, 39–40
  - Policy-Based Management, 161–189
  - priorities, 214–215
  - queries, 109–112
  - query execution, 223–224
  - replication, 485–490
  - retention periods, 227, 497, 513, 563, 658, 662
  - self-configuring options, 113
  - self-tuning feature, 85, 208
  - server options, 105–106
  - services, 62–74, 143–150
  - SET options, 102–104
  - SQL Server Agent, 142, 613–614
  - SQL Server options, 101
  - stored procedures, 109–121
  - system catalog data, 86–87
  - system stored procedures, 93–106
  - threading, 214–215
  - time-outs, 226–227
  - troubleshooting, 228–231
  - Year 2000 support, 224
  - configuration servers, 163
  - connections
    - ad hoc, 58
    - blocking connections, 157–158
    - configuration, 101, 216–220
    - connecting to servers, 136
    - data providers, 436, 438–439, 440
    - default query-processing, 217–219
    - file-based data, 437
    - limitations, 237
    - log shipping configuration, 655
    - Management Studio, 124–125, 192–194
    - monitoring, 151–152, 525
    - orphaned, 666, 669
    - remote, 76–77, 140, 141, 220, 238
    - replication updates and, 484
    - SET options, 102–104
    - simulating with traces, 551–555
    - SSIS packages, 433
    - trace data, 550
  - connectivity, 37–39
  - constraints
    - column constraints, 415–420
    - copying with scripts, 634
    - object values and limits, 235–237
    - as objects, 242
    - replicating, 481
    - as securable, 294
    - types of, 415–420
  - contracts, 294, 307
  - control flows, 434
  - Control Point monitor, 523
  - control points, 194–200
  - copy and restore jobs, 654–655
  - Copy Database Wizard, 280
  - copying
    - accounts, 633–636
    - bulk copy program, 453–454
    - databases, 280–284, 593, 661
    - log shipping copy jobs, 664
    - objects, 633–636
    - queries, 445–447
    - replication. *See* replication
    - server registrations, 132–136
    - tables, 368–369, 442–445, 633–636
    - views, 430, 442–445, 633–636
  - core database. *See* Database Engine
  - corrupt databases, 583–601, 653
  - costs of backup media, 566
  - counters
    - alert configuration, 543–544
    - data collectors, 539–541
    - mirroring monitors, 676
    - selecting, 535–538
  - CPUs
    - affinity policies, 167, 210
    - configuring, 210–214
    - determining number of, 194
    - displaying utilization, 199
    - hot-add functionality, 40
    - monitoring, 522, 524, 525
    - parallel processing, 213
    - processing time, 547
    - SQL Server requirements, 3–4
    - threading and, 214–215
    - utility control points and, 195
  - Create New Data Collector Set Wizard, 540
  - critical errors, 531
  - cross-database chaining, 266–267, 278
  - cursors
    - data types, 373
    - options, 263–264
    - replaying traces, 554
    - system stored procedures, 95
    - trace data, 550
  - Customer Feedback Reporting, 53, 73–74
  - cyclic redundancy checks, 676
- ## D
- DACs (data-tier applications), 43, 59, 194, 197–198, 199
  - daily backups, 558
  - daily changes, recovering, 569
  - daily jobs, 629
  - daily maintenance tasks, 641
  - data collector sets, 538–542, 542–544
  - data collectors, 316, 544–546
  - Data Compression Wizard, 427–428

- data conversion operators, 398
- data definition language (DDL), 91, 162, 336–342, 468
- data files
  - backup needs, 558
  - utility control points and, 195
- data flows, 434
- data manipulation language (DML), 384, 560
- data mining, 30–32, 60, 433, 438
- Data Mining Services, 433, 438
- data pages, 354–356
- data providers
  - building queries, 445–447
  - computers running SQL server, 440
  - copying tables or views, 443–445
  - file-based data connections, 437
  - flat files, 440–443
  - Integration Services and, 433
  - linked servers and, 463
  - .NET Framework, 436
  - non-SQL Server–based, 438–439
  - PowerShell and, 139
  - product names, 470
  - sources and destinations, 435–443
- data replication. *See* replication
- data sets, 480, 538. *See also* data collector sets
- data source names (DSNs), 436
- data sources or destinations. *See* destinations; sources
- Data Spaces view, 88
- data synchronization. *See* synchronizing
- data-tier applications (DACs), 43, 194, 197–198, 199
- data types
  - check constraints, 418–420
  - columns, 361
  - dates, characters, and binary data, 374–376
  - fields, 376–377
  - identity columns, 382
  - money, 372
  - native, 371–373
  - numeric, 372–373
  - as objects, 243
  - pointers to objects, 354
  - precision and scale, 371
  - properties, 379
  - script usage, 635
  - spatial, 373
  - transactional replication, 488
  - user-defined, 377–379
- data validation failures, 482
- data warehouses, 195, 200, 544–546
- database encryption keys (DEKs), 288–290
- Database Engine, 20
  - associating policies with, 181
  - configuring, 49–50
  - connecting to, 124
  - core database, 1
  - encryption and, 290
  - full-text search, 8
  - importing policies, 171–172
  - initializing, 64
  - installing services, 45
  - instances, 41–42
  - managing access, 58–60
  - object values and limits, 235–237
  - replication, 8
  - script compatibility, 635
  - server groups, 127
  - services, 1, 8, 45
  - SQL Server Agent service, 613
  - startup parameters, 145–148
  - system stored procedures, 95–96
  - Tuning Advisor, 15, 20, 398, 411–415
- Database Engine Services, 1, 8, 45
- Database Engine Tuning Advisor, 15, 20, 398, 411–415
- Database Mail
  - configuring, 604, 607–612
  - enabling, 58
  - SQL Server Agent and, 610, 614
  - stored procedures, 96–97
- Database Mail Configuration Wizard, 607–610
- Database Maintenance Plan
  - stored procedures, 97
- Database Maintenance Plan Utility, 19
- Database Management Plans, 604
- database mirroring, 37
  - vs. backup and recovery, 559
  - catalog views, 88
  - components of, 668–670
  - configuring, 670–675
  - failovers, 678–680
  - file streaming and, 71
  - linking servers, 472
  - log shipping and, 604, 681
  - mirror servers, 37, 472, 562, 668
  - monitoring, 523, 675–678, 676
  - pausing or stopping, 675
  - Performance Monitor, 676
  - principal servers, 668
  - quorum states, 677–678
  - removing, 680
  - standby servers, 562
  - synchronous or asynchronous, 669
  - troubleshooting, 676–677
  - views, 88
  - witness servers, 668
- Database Mirroring Monitor, 523, 676
- Database Mirroring view, 88
- database object owners, 305, 347
- database owner role. *See* dbo roles and users
- Database Properties dialog box, 257–261
- database roles, 292–294, 314–316
- database scope, 293–294
- databases. *See also specific types of databases (master, msdb, etc.)*
  - access, 124, 125–126, 329–336
  - adding, deleting, or manipulating, 275–280
  - administration, 329–336
  - altering, 251–270
  - attaching or detaching, 277–280
  - auto-growing, 166, 247–248, 270
  - automatic options, 257–259
  - backing up. *See* backing up and backup sets
  - basics of administering, 238–245

- collation, 249
- compressing, 271–275
- configuration options, 101
- copying and moving, 277–286, 633–636
- creating, 245–251
- cross-database chaining, 266–267
- database state, 264–265
- defragmenting, 170, 645
- diagrams of, 243
- dropping or deleting, 276–277
- encryption, 288–290
- expanding, 270–271
- file size, 270–275
- filegroup errors, 288
- files, 233–238
- indexes. *See* indexes
- integrity checks, 648–652
- large, 564–565
- limits, 233
- log shipping, 653–655
- logs, 233–238, 270–275
- maintenance, 641–652
- management data
  - warehouses, 195, 200, 544–546
- Management Studio, 124, 125–126, 239–240, 251–252
- mirroring, 668–680
- modifying, 251–252
- object management, 242–245
- object values and limits, 235–237
- online, offline, or emergency modes, 94, 266
- overwriting, 588
- permissions, 307–308, 336–345
- planning recovery, 557–566
- policy categories, 163
- primary data files, 234
- properties list, 241
- recovering, 583–601
- renaming, 275–276
- roles, 292, 293, 294, 314–316
- samples, 241–242
- schemas. *See* schemas
- scope, 293–294
- secondary data files, 234, 287
- as securable, 294
- security levels, 292–293
- SET commands, 269–270
- shrinking, 247–248, 270–275
- single-user mode, 605
- size, 270–275
- states, 585, 590
- system databases, 234–235, 241–242
- system tables, 92–93
- system views, 92–93
- tables. *See* tables
- taking offline, 94
- templates, 288
- transaction errors, 288
- transaction logs, 233–238
- T-SQL viewing, 240–241
- tuning, 15, 20, 398, 411–415
- viewing, 239–241
- views. *See* views
- working copies of, 593
- Databases and Files view, 87
- dates, 222, 224, 374, 454, 588, 629
- DATs (digital audio tape drives), 567
- day formats, 222
- DBCC commands, 524, 648–652
- DB-Library, 226–227
- dbo-only databases, 585
- dbo roles and users, 305
  - authentication modes, 302
  - db\_ roles, 313–316
  - implied permissions, 312
  - object owners, 305, 347
  - policies, 169
  - statement permissions, 311
- dbo schema, 188, 347
- DCM (Differential Changed Map), 355
- DDL (data definition language), 91, 468
- deadlocks, 157–158, 167, 546
- debug dump files, 431
- debugging traces, 554
- decimal data types, 372
- declarative ordering, 484
- dedicated administrator
  - connections, 37–39, 76–77
- defaults as objects, 242
- defragmenting databases, 170, 645
- DEKs (database encryption keys), 288–290
- deleting
  - alerts, 618–619
  - columns, 370
  - conditions, 180
  - databases, 276–277. *See also specific kinds of databases (master, msdb, etc.)*
  - groups, 129
  - indexes, 410–411
  - instances, 228
  - job definitions, 631
  - job schedules, 628
  - jobs, 631
  - linked servers, 476
  - log shipping configurations, 668
  - logins, 303, 325
  - maintenance plans, 648
  - mirroring configurations, 680
  - objects with scripts, 635
  - operators, 620
  - policies, 183
  - publications, 512
  - roles, 334
  - rows, 418
  - schemas, 353
  - server registration, 137
  - SQL Server 2000 tables, 656–657
  - startup parameters, 146
  - subordinate servers/groups, 175
  - subscriptions, 518
  - tables, 369–370
  - user-defined data collectors, 539
  - users, 334
- delimited files, 442
- deployment
  - access, 37–39
  - availability, 36–37
  - configuration, 39–40
  - connectivity, 37–39
  - data-tier applications, 197–198
  - integration roles, 27–33
  - performance and, 33–34
  - planning, 33–40
  - scalability, 36–37
  - security, 39–40
  - SQL Server Setup, 40–55
  - virtualized, 41–42
- descriptive headers, 634
- destinations, 435–443, 447–450, 596. *See also* data providers; replication
- detaching databases, 277–280

- Developer edition (SQL Server), 5, 206–207
  - developer roles, 331–333
  - devices. *See* disk drives; hardware
  - differential backups, 559, 576, 578, 584, 588
  - Differential Changed Map (DCM), 355
  - digital audio tape drives, 567
  - Digital Linear Tape (DLT), 567
  - directories, 50, 51, 73–74, 194
  - disaster recovery. *See* recovery
  - disconnecting
    - from control points, 197
    - from servers, 136
  - disk-based backup systems, 567
  - Disk Defragmenter tool, 170
  - disk drives
    - assessing needs, 558, 566–568
    - backup devices, 234, 567, 570–572
    - disk space requirements, 47
    - malfunctioning, 170
    - RAID, 4, 34–36, 558, 567–568
    - SQL Server requirements, 4
    - utility control points and, 195
  - disk I/O errors, 268–269
  - disk mirroring, 4, 240, 277
  - disk space
    - compression, 425–428
    - log shipping settings, 662, 665
    - monitoring, 524
    - tuning advice, 414
  - disk striping, 4
  - diskadmin role, 313–314
  - distributed data, types, 463
  - distributed partition views, 387
  - distributed queries, 97, 464–466, 550
  - Distributed Transaction Coordinator, 9
    - configuring, 142–143
    - linked servers, 468
    - MS DTC service, 467
    - replication and, 485
    - running, 468
    - stored procedures and queries, 220
  - distributed transactions, 463, 466–468, 471
  - distributing loads, 287
  - Distribution Agent, 482, 513, 515, 516, 517
  - Distribution Clean Up job, 482
  - distribution* databases, 242, 481, 492, 495–497, 499, 563, 564, 570
  - distributors, 481
    - backing up, 563
    - creating, 490–494
    - deleting databases, 499
    - disabling publishing, 499
    - distribution* databases, 497
    - enabling publishers, 497–498
    - planning architecture, 486
    - properties, 497
    - publication* databases, 498
    - retention settings, 497
    - snapshots, 493
    - updating, 495–496
  - DLTs (Digital Linear Tapes), 567
  - DML (data manipulation language), 384, 468
  - documenting policies, 182
  - domain accounts, 10–11, 48, 69, 302, 673
  - domain logins, 293, 303
  - double quote delimiters, 442
  - driver error messages, 170
  - dropping
    - databases, 276–277
    - schemas, 353
  - DSNs (data source names), 436
  - dta.exe (Database Tuning Advisor), 15, 20, 398, 411–415
  - DTC. *See* Distributed Transaction Coordinator
  - dtxexec.exe (execution package utility), 20, 431, 432
  - Dtxexecui utility, 431
  - DTS 2000, 27–28, 432
  - DTS 2000 Package Migration Wizard, 28
  - DTS Import/Export Wizard. *See* SQL Server Import And Export Wizard
  - DTS services, 430. *See also* SQL Server Integration Services (SSIS)
  - dutil.exe, 21
  - dump directories, 73–74
  - dump files, 431
  - duplicate keys, 404, 416
  - duty schedules, 620
  - dynamic configuration settings, 111
  - dynamic lock policies, 168
  - dynamic management view, 92–93
  - dynamic memory allocation, 203, 205
  - dynamic TCP/IP configuration, 80–81
- ## E
- EAL4+ requirements, 202–203
  - editing. *See* modifying editions of SQL Server
    - Compact edition, 7, 127, 505
    - Developer edition, 5, 206–207
    - Enterprise edition, 6, 206–207
    - Express edition, 7
    - localized versions, 222
    - R1 and R2 releases, 5, 6–7, 195, 198, 367, 633–636
    - R2 Datacenter, 5, 6–7
    - R2 Parallel Data Warehouse, 5, 7
    - recovery and, 584
    - script compatibility, 635
    - Standard edition, 5–6, 206–207
    - upgrading, 54
    - viewing edition information, 194
    - Web edition, 7
    - Workgroup edition, 5
  - effective permissions, 297–301
  - effective policies, 163
  - EFS (Encrypting File System), 39
  - e-mail
    - accounts, 606, 611–612
    - alerts, 615–619
    - Database Mail, 58, 96–97, 606–612
    - fail-safe operators and, 621
    - job notifications, 630–631
    - mail host servers, 604, 606–612
    - SQL Mail, 60, 99, 614



- SQL Server Agent profile, 606, 614
- stored procedures, 96–97, 99
- emergency mode, 266, 585
- Encrypting File System (EFS), 39
- encryption
  - available types of, 39
  - catalog views, 90
  - configuring, 288–290
  - database mirroring, 671
  - forcing, 81
  - packages, 451
  - policies, 165
  - transparent, 120
- endpoints
  - catalog views, 88
  - database mirroring, 671, 672, 673
  - permissions, 308
  - as securable, 294
  - status of, 59
- Endpoints view, 88
- enforcing
  - passwords, 169
  - policies, 184–185
- enlisting target servers, 640
- Enterprise edition (SQL Server), 6, 206–207, 659, 676
- enterprise management
  - centralized servers, 172–175
  - Management Studio, 123–127
  - monitoring activity, 150–159
  - MSDTC, 142–143
  - policies, 171–189
  - server groups, 127–130
  - server management, 130–137
  - SQL Server Agent, 142
  - SQL Server Startup, 143–150
  - Windows PowerShell, 137–141
- Enterprise Manager. *See* SQL Server Management Studio
- erasing backups, 575
- error logs
  - as monitoring tool, 524
  - startup parameters, 147, 148
- error numbers, 529, 617
- error state numbers, 529
- errors and error messages
  - alerts, 615–619
  - during backups, 575
  - catalog views, 88
  - components of messages, 529
  - concurrent connections, 217
  - critical and noncritical, 531
  - database mirroring, 676
  - disk I/O errors, 268–269
  - Error 824, 166
  - error reporting, 73–74
  - “filegroup is full” error, 288
  - list of, 170
  - operators for indexes, 398
  - packet errors, 525
  - policy-generated, 170
  - read/write errors, 525
  - report settings, 52–53
  - script execution, 634
  - studying before recovery, 585
  - viewing, 529
- eSATA drives, 567
- ETL platforms, 27–28, 429
- evaluation expressions, 178–180
- evaluation modes (policies), 162–163, 182, 185–188
- event logs
  - configuring, 533
  - database mirroring issues, 676
  - list of messages, 170
  - monitoring events, 528–532
  - as monitoring tool, 523, 524
  - policy compliance records, 188
  - policy-generated messages, 170
  - recycling, 534
  - viewing, 532–535
- Event Viewer, 523, 530–532
- EventId messages, 170
- events
  - alerts, 616
  - duration, 547
  - error message 825, 170
  - error message 833, 170
  - event classes, 549, 552
  - event subclasses, 549–550
  - forwarding, 637–638
  - job notifications, 630–631
  - recording with traces, 548–550
  - replaying order of, 554
  - system event messages list, 170
  - traceable, 541, 546
  - tracing, 539, 549
  - types of, 531
- Excel, 430, 433, 437, 473
- Execute Package Utility, 431
- executing packages, 431, 450–453
- existing backup sets, 576–577
- existing media sets, 578
- expanding databases, 166, 247–248, 270–271, 285–286
- expiration intervals, 574, 576, 588
- expired password settings, 169
- expired subscriptions, 482, 513
- exporting compliance results, 187
- exporting data
  - bulk copy utility (BCP), 453–461
  - BULK INSERT command, 461–462
  - Import And Export Wizard. *See* SQL Server Import And Export Wizard
  - SSIS and, 429–434
- exporting object states, 177
- exporting policies, 164, 171–172
- exporting registration detail, 132–136
- Express edition (SQL Server), 7
- extended properties, 88, 240, 402, 481, 635
- extended stored procedures, 97, 243
- extensibility of backup media, 566
- extents, 355–356, 357
- external access options, 266–267
- external processes, 434
- extracting
  - data, 430
  - data-tier applications, 197–198
- extraction, transformation, and loading (ETL) platform, 27–28, 429

**F**

facets, 161, 163, 176–180  
 fail-safe operators, 620–621  
 failed jobs, 630–631  
 failover clustering, 72, 595  
 failover methods, 484  
 failovers
 

- database mirroring, 669, 674
- forcing, 678–680
- mirroring states and, 677
- recovery, 678–680
- replication methods, 484

 Fiber Channel (FC), 4  
 fibers, 214–215  
 fields, 376–377, 459  
 file-based data connections, 437  
 file I/O streaming, 72–73  
 file size and space
 

- Auto Shrink option, 258
- database assignments, 247
- displaying utilization, 199
- expanding, 234
- limitations, 236
- mail messages, 609
- unused space, 272, 274

 file streaming, 50, 70–73, 221–222, 592  
 file systems, 473  
 filegroups
 

- backups, 560, 579, 591–593
- database assignments, 247
- DBCC commands, 652
- encryption, 290
- filegroup is full error, 288
- FILESTREAM filegroups, 71
- partition schemes, 422
- restoring, 591–593, 597–598
- views, 88, 239

 files
 

- attachments (mail), 610
- autogrowing, 166
- database files, 233–238
- encryption, 290
- file backups, 560, 579, 591–593
- location policies, 165
- restoring, 591–593, 597–598

size. *See* file size and space  
 streaming, 50, 70–73, 221–222  
 viewing, 239  
 Filestream Data, 592  
 FILESTREAM filegroups, 71  
 fill factor, 225–226, 401, 645  
 filtered application logs, 531–532  
 filtered indexes, 396, 397–398  
 filtered tables, 502–503, 506–507, 509  
 filtered trace information, 550  
 fixed-length fields, 376–377  
 fixed-length rows, 356  
 fixed memory configuration, 206  
 fixed roles, 314–316  
 fixed-point numeric literals, 262  
 flat files, 440–443  
 floating data types, 372  
 floating point numeric literals, 262  
 flushing cache, 227  
 forced parameterization, 261–263  
 forcing service to failover, 678–680  
 foreign keys, 235, 236, 354, 417–418, 635  
 formats, text file, 441  
 formatting data, 418–420, 447–450  
 formulas, 362  
 forwarding notifications, 637–638  
 fragmentation, 170, 402, 645  
 free space, 273, 274  
 FTP snapshots, 509  
 full backups, 559, 576, 578, 584, 588, 601–602  
 full logging, 267  
 full partitioning, 413  
 full quorum relationships, 678  
 full recovery, 561, 653, 655  
 full-text catalogs
 

- detaching databases and, 278
- moving, 285
- permissions, 308
- reattaching databases, 279
- as securable, 294
- storing, 234

 full-text indexes, 71, 97, 234, 235, 236, 560, 635

full-text search, 8, 97  
 full-text stoplists, 244–245  
 functions

- monitoring functions, 524–525
- as objects, 243
- partition functions, 421–422
- permissions, 295–297
- as securable, 294

**G**

GAM (Global Allocation Map), 355  
 General Extended stored procedures, 97  
 geographic data types, 373, 396  
 geographic location of servers, 654  
 geographic server groups, 127  
 geometric data types, 373, 396  
 Global Allocation Map (GAM), 355  
 global temporary tables, 358–359  
 globally unique column IDs, 362  
 globally unique identifiers (GUIDs), 373, 381–383  
 grantable permissions, 297–298  
 granular permissions, 351  
 graphical administration tools, 12–15  
 group accounts, 301–302  
 Group Policy, 10, 39  
 groups
 

- accounts, 301–302
- authentication, 301–302
- policy categories, 183–185
- schema ownership and, 348
- security levels, 293
- as security principals, 292
- server groups, 127–130
- subordinate servers, 174–175
- user groups, 65–67

 guest accounts, 167, 302, 304  
 GUID data types, 373, 381–383

**H**

hard disks. *See* disk drives  
 hard page fault errors, 170  
 hardware  
   back-up devices, 234, 570–572  
   back-up needs, 558, 566–568  
   driver error messages, 170  
   error messages, 170  
   malfunctioning, 170  
   monitoring, 522  
   performance, 2–5, 33–34  
   requirements, 2–5  
   secondary servers, 654  
   temperature, 34  
 hardware RAID, 34–36  
 heaps, 425, 427  
 help files for policies, 182  
 high-performance modes, 679  
 high-risk attachments, 610  
 high-safety modes, 679  
 history  
   backups, 588  
   cleaning up, 482, 645  
   jobs, 631  
   log shipping, 664–665  
   maintenance, 647–648  
   maintenance plan tasks, 642  
 hot standby servers, 562  
 hot-add functionality, 40  
 hot-add memory, 147, 207  
 hours, recovering up to, 569  
 HTTP services, 62  
 HTTPS protocol, 141

**I**

IAM (Index Allocation Map), 355–356  
 IBM DB2 databases, 473  
 IDBSchemaRowset interface, 466  
 identifier columns, 361, 416–417  
 identifiers, 236, 260, 348.  
   *See also* globally unique identifiers  
 identity columns, 382, 502, 506  
 identity increments, 361  
 identity seeds, 361, 383  
 identity values, 381–383, 651  
 idle time, 524, 525, 628  
 IIS servers, 509

image columns, 489  
 image data types, 375, 394  
 impersonation, 474, 507–508  
 implied permissions, 312  
 Import And Export Data Wizard, 15  
 importing data, 429–434, 453–462. *See also* SQL Server Import And Export Wizard  
 importing policies, 164, 171–172  
 importing registrations, 132–136  
 incremental backups, 559  
 Index Allocation Map (IAM), 355–356  
 Index data pages, 356  
 index fill factor, 225–226, 401  
 index fragmentation data, 402  
 index ID values, 394  
 index keys, 235  
 index locks, 395, 401  
 index pages, 394  
 indexed views, 29, 243, 387–388, 425  
 indexes, 353–354, 394–396  
   compression, 425–428  
   computed columns/  
   views, 399–400  
   creating, 402–407  
   deleting, 410–411  
   fill factor, 225–226, 401  
   index locks, 395  
   index pages, 394  
   maintenance plan tasks, 643  
   memory allocation, 208  
   monitoring, 524  
   naming, 410–411  
   as objects, 242  
   parallelism, 401  
   partitioned, 421–425  
   performance and, 395  
   properties, 400–401, 402, 410  
   rebuilding, 401, 411, 643, 645  
   recommendations, 414–415  
   recovering, 591  
   reorganizing, 411, 643, 645  
   replicating, 481  
   script usage, 635  
   selecting columns, 398–399, 402  
   statistics, 401  
   storage, 402  
   tree structure, 394–395  
   Tuning Advisor, 411–415  
   types of, 395–396  
   unique indexes, 416  
 information events, 531  
 Information Schema views, 86  
 inheriting permissions, 292  
 initializing  
   initialization scripts, 25–26  
   log shipping servers, 661  
   reinitializing subscriptions, 519–520  
 input packets, 525  
 input/output statistics, 525  
 installation  
   adding components, 53  
   Installation Wizard, 42  
   instance features, 8  
   new server instances, 42–53  
   PowerShell tools, 140  
   repairing, 54  
   shared features, 8  
   Windows installer, 7–8  
 Installation Wizard, 42  
 instances  
   adding, 53  
   applying policies to, 164  
   associating policies with, 181  
   catalog views, 87  
   central management servers, 173  
   collation, 230–231  
   configuration options, 101  
   connecting to, 124–125  
   creating new, 41–42  
   default, 42  
   displaying for backups, 588  
   distributing workloads, 480  
   enrolling in control points, 195, 197  
   installing instance features, 8  
   limitations, 236  
   managed, 195, 199  
   named or default, 46  
   permissions, 295–297  
   recovering, 229–230  
   repairing, 54  
   as securable, 294  
   SQL Server Agent service, 613

instances, *continued*  
 SQL Server Setup, 42–53  
 startup parameters, 148  
 system stored procedures, 95  
 uninstalling, 54–55  
 integer data types, 372  
 integer literals, 262  
 integration roles, 27–33  
 Integration Services. *See*  
 SQL Server Integration  
 Services (SSIS)  
 Integration Services  
 Designer, 431  
 integrity checks, 642,  
 644–645, 648–652  
 Internet access to snapshots,  
 509  
 invalidating log sequences,  
 582  
 I/O affinity, 210, 211  
 I/O processing time, 525  
 I/O request policy messages,  
 170  
 I/O subsystems, 33–34,  
 34–36  
 IPv4 and IPv6 addresses, 80  
 ISQL (SQLCMD), 15–18, 20

## J

Jet databases, 472  
 Job Activity Monitor, 523  
 job categories, 623, 631–632  
 job definitions, 622–623, 631  
 job shipping, 595–596  
 jobs  
 alerts, 629–630  
 categories, 623,  
 631–632  
 central administration,  
 637–638  
 configuring, 604  
 copying, 636–637  
 creating, 621–622  
 daily, weekly, or  
 monthly, 629  
 deleting, 631  
 deleting schedules, 628  
 disabling or enabling,  
 631  
 execution logs, 626  
 history, 631  
 Job Activity Monitor,  
 523  
 job definitions,  
 622–623, 631  
 log shipping, 654–655,  
 664–665

maintenance plans,  
 642, 645  
 monitoring, 526  
 notification messages,  
 630–631  
 one-time, 629  
 recurring, 629  
 replication, 481–483  
 responding to alerts,  
 617–618  
 schedules, 627–629,  
 638–640  
 scripts, 631  
 SQL Server Agent, 142,  
 612, 613  
 starting, 628, 631  
 steps to execute,  
 623–627  
 stopping, 631  
 successful, failed, or  
 completed, 630–631  
 joined tables, 506  
 joining data in views,  
 386–387

## K

keys, 243. *See also specific  
 types of keys (foreign, pri-  
 mary, etc.)*  
 killing processes, 159

## L

language settings, 194, 222  
 large object data types,  
 356, 394  
 lazy schema validation, 476  
 legacy data, migrating,  
 431–432  
 license terms, 44  
 lifetime settings (mail), 610  
 lightweight pooling, 168,  
 215  
 limitations, databases,  
 235–237  
 Linear Tape Open (LTO), 567  
 linked objects, 61  
 linked servers  
 adding, 469–473  
 collation, 476  
 configuring, 475–476  
 deleting, 476  
 distributed queries,  
 464–466  
 distributed transactions,  
 463, 466–468  
 DTC service, 468  
 Linked Servers view, 88

loopback linked servers,  
 464  
 OLE DB parameters,  
 472–473  
 security, 473–475  
 Linked Servers view, 88  
 listen ports, 77, 78, 79, 671,  
 672, 673  
 literals, 262  
 loading process (databases),  
 585  
 local Administrators group  
 account, 301  
 local Administrator user  
 account, 301  
 local connections, 76–77  
 local logins, 293, 474  
 local partitioned views, 387  
 local server instances, 61  
 local service accounts, 11  
 local system accounts,  
 10–11, 48  
 localized versions of SQL  
 Server, 222  
 locks  
 dynamic lock policies,  
 168  
 index locks, 395, 401  
 lock escalation, 364  
 lock-related counters,  
 535  
 monitoring, 524  
 replication updates, 487  
 server logins, 324–325  
*tempdb* databases, 286  
 troubleshooting,  
 157–158  
 types of, 153–156  
 Log data, 592  
 log files. *See* logs  
 Log Reader Agent, 482,  
 503–504, 510  
 log sequence numbers  
 (LSNs), 588  
 log shipping  
 alerts, 658–659  
 vs. backup and recov-  
 ery, 559  
 backup intervals, 664  
 components in,  
 653–655  
 configuring, 604  
 database mirroring and,  
 37, 681  
 disabling, 668  
 disk space issues, 665  
 failing over, 665–668  
 file streaming and, 71  
 in Management Studio,  
 240

- monitor servers, 659–660
- monitoring activity, 664–665
- performance and, 566
- permissions issues, 665
- preparations for, 655–656
- primary databases, 657–660
- recovery, 665–668
- removing, 668
- secondary databases, 661–663
- standby servers, 562
- system stored procedures, 98
- upgrading SQL Server 2000, 656–657
- Log Shipping Agent, 20
- logical data errors, 677
- logins
  - access control lists (ACLs), 511
  - assigning roles by, 326–329
  - authentication, 10
  - authentication modes, 301–302
  - BCP import and export, 457
  - controlling access, 329–336
  - creating, 319–321
  - database mirroring and, 669
  - editing, 317–318, 321–322
  - enabling, disabling, or unlocking, 324–325
  - GRANT or DENY access, 323–324
  - linked servers, 473–475
  - login auditing, 201–203
  - management data warehouses, 545
  - managing for servers, 317–326
  - object permissions, 342–344
  - passwords, 326
  - permissions, 309
  - policy settings, 168
  - removing, 325
  - replication agents, 503–504, 510
  - resolving after failover, 667
  - script usage, 635
  - as securable, 294
  - security levels, 293
  - SMTP servers, 606
  - special-purpose, 302–305
- logs
  - audit log files, 202
  - backup folders (log shipping), 654
  - configuring number of, 533
  - creating new, 279–280
  - Database Mail, 610
  - expanding size, 270–271
  - invalidating log sequences, 582
  - job execution, 626
  - locations, 540
  - log marks, 568
  - log mirroring, 566
  - managing size, 270–275
  - monitoring, 528–532
  - as monitoring tool, 524
  - performance improvements, 287
  - performance logging, 538–542
  - policies, 165
  - Reader Agent, 482, 503–504, 510
  - reattaching databases, 278–280
  - recovery strategies, 585
  - recycling, 534
  - reports, 542–543
  - restoring, 600
  - size, 540
  - supplemental logging, 269
  - transaction log recovery, 582
  - transaction logs, 233–238
  - utility control points and, 195
  - viewing, 530–532, 532–535
- lookup caches, 28
- Lookup transformation, 432
- loopback linked servers, 464
- LSNs (log sequence numbers), 588
- LTO (Linear Tape Open), 567
- M**
- mail. *See* e-mail
- maintenance, 604–605
  - checklists, 641–642
  - integrity checks, 648–652
  - maintenance plans, 642–648
  - reports, 646, 647–648
  - scheduling jobs, 621–632
  - server-to-server tasks, 632–637
  - single-user mode, 605
  - SQL Server Agent, 612–615
  - system stored procedures, 97
- Maintenance Plan Package Designer, 642
- Maintenance Plan Wizard, 642, 643–647
- maintenance plans, 642–648
- Make TSX Wizard, 640
- managed code vs. T-SQL, 28–29
- managed instances, 199
- management. *See also* administration; databases
  - command-line tools, 148–149
  - Management Studio, 124
  - Management Tools, 45
  - policy-based, 161–164
  - server activity, 150–159
  - Windows PowerShell, 137–141
- management data warehouses, 544–546
- Management Tools, 45
- mandating policies, 183–185
- manual failovers, 679
- MAPI (Messaging Application Programming Interface), 614
- mapping
  - columns, 447–450
  - partitions, 424
- MARS (multiple active result sets), 37–39, 102
- master database, 242
  - backup and recovery, 558, 568, 570
  - compatibility levels, 108
  - configuration options, 106–107
  - rebuilding, 230–231
  - restoring, 601–602
  - startup parameters, 147
  - symmetric key policies, 169
  - system tables, 92–93
  - system views, 91–93
- master keys, 289–290, 672
- Master Server Operator, 639
- Master Server Wizard, 639–640
- master servers, 638–640

- maximum free space, 273
- maximum memory allocation, 203, 205
- maximum user connections, 216–217
- MDAC (Microsoft Data Access Components), 42
- MDX queries, 31
- mean time to failure (MTTF), 34
- media and media sets
  - backup needs, 566–568
  - creating new, 577
  - expiration, 574
  - restoring from, 587
  - rotation schedules, 568
- memory
  - AWE memory support, 206–207
  - configuring, 203–209
  - dynamic, 203, 205
  - fixed, 206
  - hardware requirements, 3
  - hot-add memory, 40, 207
  - indexing and, 208
  - monitoring, 522
  - physical memory, 205
  - query execution, 208–209
- Merge Agent, 482, 489–490, 513, 515, 516, 517
- merge publications, 504–508, 519
- merge replication, 484, 489–490, 563–564
- merging tables, 354
- message types, 294, 309
- messages, 88, 163, 170.
  - See also* errors and error messages
- Messages view, 88
- Messaging Application Programming Interface (MAPI), 614
- messaging services, 2
- Microsoft Access, 430, 433, 437, 472
- Microsoft Cluster Service, 595
- Microsoft Data Access Components (MDAC), 42
- Microsoft Data Mining Services, 433, 438
- Microsoft Distributed Transaction Coordinator. *See* Distributed Transaction Coordinator
- Microsoft Excel, 430, 433, 437, 473
- Microsoft OLE DB Provider for SQL Server, 440
- Microsoft SQL Server. *See* SQL Server 2008
- Microsoft SQL Server Integration Services. *See* SQL Server Integration Services (SSIS)
- Microsoft Visual Basic, 433
- Microsoft Visual Studio 2010, 197, 198
- Microsoft Visual Studio Tools for Applications (VSTA), 432
- Microsoft Windows. *See* Windows
- migrating
  - data, 274
  - legacy data, 431–432
  - scripts, 432
  - SQL Server 2000 log shipping, 656–657
- minimal logging, 30
- minimum memory allocation, 203, 205
- minutes, recovering up to, 569
- mirror endpoints, 671
- mirror security, 671
- mirror servers, 37, 472, 562, 668, 677–678. *See also* database mirroring
- mirroring databases. *See* database mirroring
- mixed extents, 357
- mixed security mode, 302
- Mobile edition (SQL Server), 505
- model* database, 242
  - backing up, 570
  - compatibility levels, 108
  - configuration options, 106–107
  - as prototype, 245
  - restoring, 230–231
  - symmetric key policies, 169
  - using as templates, 288
- modifying
  - columns, 370
  - Database Mail parameters, 612
  - database options, 251–257
  - job schedules, 628
  - job steps, 623–627
  - logins, 317–318, 321–322
  - maintenance plans, 648
  - policies, 183
  - schemas, 350–351
  - server groups, 129–130
  - server registration details, 136
  - tables, 364–367, 423
  - traces, 551
  - views, 392–393
- money-related data, 262, 372, 454
- monitor servers, 654, 656, 668
- monitoring
  - built-in functions for, 524–525
  - database mirroring, 675–678
  - deadlocks, 167
  - event log information, 528–532
  - log shipping, 664–665
  - management data warehouses, 544–546
  - performance, 535–544
  - performance metrics, 522
  - reasons for, 521–522
  - Replication Monitor, 525–535
  - reports, 542–543
  - resource waits and blocks, 153–156
  - server activity, 150–159
  - SQL Server Profiler, 546–555
  - tools and resources, 523–525
  - user commands, 159
  - witness servers, 668
- month formats, 222
- monthly jobs, 629
- monthly maintenance tasks, 641
- moving
  - databases, 277–280, 280–285
  - objects in schemas, 351–352
  - server groups, 129–130
  - servers to new groups, 136–137
  - subordinate servers/groups, 175
  - tempdb* database, 285–286
- msdb* database, 242
  - backing up, 570
  - configuring as mail host, 614
  - policy data, 164, 188
  - restoring, 230–231
  - special roles in, 316

symmetric key policies, 169  
 MTTF (mean time to failure), 34  
 multidimensional databases, 30–32  
 multinode server clusters, 42  
 multiple active result sets, 37–39, 102  
 multiprocessors, 4  
 multiserver administration, 637–640  
 multiserver tasks, 632–637  
 multitasking, 210, 214–215  
 multiuser databases, 265  
 multiuser mode, 605

## N

N/A value, 380  
 named instances, 46  
 Named Pipes protocol, 76–78, 82–83  
 names  
   collations, 249  
   columns, 361  
   control points, 196  
   databases, 246, 275–276  
   groups, 127  
   indexes, 410–411  
   packages, 452  
   registered servers, 132  
   schemas, 348, 349  
   script usage, 634  
   tables, 358–359  
 namespaces, 348  
 native client configuration, 42, 81  
 native data types, 371–373, 374–376  
 nested triggers, 162, 189, 223, 236  
 .NET Framework, 42, 61, 430, 436  
 Net Send alerts, 615–619, 621, 630–631  
 network configuration  
   Named Pipes clients, 82–83  
   Named Pipes protocol, 77–78  
   native clients, 81–82  
   Network Configuration node, 75  
   Shared Memory clients, 82  
   Shared Memory protocol, 77  
   TCP/IP clients, 83–84  
   TCP/IP protocol, 78–79

network interface cards (NICs), 212  
 network packets, 169  
 network service accounts, 11  
 network shares, 658  
 network traffic, monitoring, 522  
 new databases, 245–249  
 New Subscription Wizard, 513–517  
 NICs (network interface cards), 212  
 nodes in indexes, 394–395  
 nonaligned partitioned indexes, 208  
 nonclustered indexes, 395–399, 425, 426–427  
 noncritical errors, 531  
 nonindexed views, 388  
 nonlogged operations, 582  
 nonoperational databases, 590  
 nonpartitioned tables, 423  
 non-SQL Server subscriptions, 510  
 non-Unicode string literals, 262  
 not null constraints, 415, 420  
 “not recovered” databases, 585  
 notification messages, 630–631, 643. *See also* alerts; errors and error messages  
 ntext data types, 235, 355, 356, 381, 394, 421, 489  
 nulls, 361, 379–381, 415, 420. *See also* ANSI nulls  
 numeric data, 372–373, 419  
 numeric round-abort options, 219, 260, 400  
 nvarchar(max) data type, 354, 356, 375, 488  
 NWLink IPX/SPX, 77

## O

Object Explorer view, 12–13, 171–189, 193  
 OBJECT plan guide, 244  
 objects  
   copying, 633–636  
   database object owners, 305  
   monitoring object locks, 524  
   moving in schemas, 351–352  
   permissions, 306–311, 342–344  
   replicating, 481

schema objects, 300, 347, 481  
 tables as, 354  
 types of, 242–245  
 values and limits, 235–237  
 viewing, 12–13, 88–89, 171–189, 193  
 Objects view, 88–89  
 ODBC catalog functions, 86  
 ODBC connection states, 217  
 ODBC data sources, 472–473  
 ODBC drivers, 436  
 ODBC errors, 529  
 ODBC functions, 95  
 off-site backups, 559  
 off-duty operators, 620–621  
 offline databases, 266, 585  
 OLAP (online analytical processing), 2, 30–32, 438  
 OLE Automation, 59, 98  
 OLE clients, 224  
 OLE DB connections, 217, 438, 440, 465–466  
 OLE DB errors, 529  
 OLE DB providers, 463, 470, 472–473  
 OLE DB schema rowsets, 86–87  
 OLE DB verification, 17  
 On Change modes, 162–163, 188, 189  
 On Demand mode, 162–163, 188  
 On Schedule mode, 162–163, 188  
 on-demand maintenance plans, 643  
 on-demand replication updates, 516  
 online analytical processing (OLAP), 2, 30–32, 438  
 online databases, 266  
 online index recommendations, 414  
 opening databases, 165  
 operating system commands, 625  
 operating system information, 194  
 operators (arithmetic), 262  
 operators (individuals)  
   automation, 619–621  
   configuring, 604  
   copying, 636–637  
   deleting, 620  
   disabling notifications, 620  
   fail-safe, 620–621  
   policy conditions, 178–180

operators (individuals),  
*continued*  
 registering, 619  
 responding to alerts,  
 617–618  
 SQL Server Agent, 612,  
 613  
 optimization, 411–415, 604  
 Oracle databases  
   connections, 436, 438,  
   439  
   importing data to or  
   from, 430, 433  
   linked server param-  
   eters, 472  
   monitoring, 528  
   replication connections,  
   492

order of events, 554  
 orphaned connections, 669  
 OSQL tool (SQLCMD), 15–18,  
 20  
 “out of locks” errors, 168  
 output packets, 525  
 overflow errors, 260  
 overwriting databases, 588  
 ownership  
   database objects, 305  
   ownership chaining,  
   266–267, 278  
   permissions, 297

**P**

Package Configuration  
 Wizard, 431  
 Package Migration Wizard,  
 432  
 packages, 433–434  
   authentication, 452  
   building queries,  
   445–447  
   copying tables or views,  
   442–445  
   creating, 434–435  
   data formatting,  
   447–450  
   job steps, 625  
   managing, 431  
   migrating, 431–432  
   Package Configuration  
   Wizard, 431  
   Package Migration  
   Wizard, 432  
   saving and executing,  
   450–453  
   sources and destina-  
   tions, 435–443  
 packet errors, 525  
 packet monitoring, 525  
 page faults per second, 204

Page Free Space (PFS), 356  
 page verification, 166,  
 268–269  
 paggers, 615–619, 620,  
 630–631  
 pages, 166, 204, 268–269,  
 354, 426–427, 650–651  
 parallel backup and restore,  
 564–565  
 parallel I/O, 564–565  
 parallel processing, 210–214,  
 483, 488  
 parallel striped backups,  
 577–578  
 parallelism, 395, 401  
 parameters  
   advanced configuration,  
   113–115  
   BCP, 455, 458  
   changing settings, 116  
   current settings,  
   115–116  
   Database Mail, 609–610  
   forced parameteriza-  
   tion, 261–263  
   Invoke-Sqlcmd cmdlet,  
   138  
   limitations, 237  
   OLE DB providers,  
   472–473  
   simple parameteriza-  
   tion, 261–263  
   standard configuration,  
   112–113  
   startup, 145–148  
 partial backups, 560  
 partial differential backups,  
 560  
 partial restores, 594, 599  
 partial updates, 488  
 Partition Function view, 89  
 partition switching, 358  
 partition-aligned indexed  
   views, 29  
 partition-aligned structures,  
 413  
 partitioned indexes, 208, 423  
 partitioned tables, 29, 208,  
 423  
 partitioned views, 243,  
 387–388  
 partitions  
   catalog views, 89  
   compression, 425–428  
   creating, 423–424  
   indexes and, 395  
   mapping, 424  
   partition functions,  
   421–422  
   partition schemes,  
   422–423

partition switching, 358  
 tables, 357–358  
 types of, 413  
 viewing and querying,  
 424–425  
 partner-to-partner relation-  
 ships, 678  
 passwords, 10, 39, 169, 326,  
 451, 672  
 paths, 139–140, 147, 658  
 pausing  
   database mirroring, 675  
   traces, 551  
 PDS (Physical Design Struc-  
 tures), 413  
 peer-to-peer replication  
   model, 486  
 performance. *See also* Per-  
   formance Monitor  
   alerts, 543–544,  
   615–619  
   baseline metrics, 522  
   compression and, 228,  
   566  
   counters, 535–538  
   data collector sets,  
   538–542  
   fields and, 376–377  
   filtered indexes and, 396  
   fixed memory and, 206  
   hardware architecture  
   and, 2–5  
   indexes and, 395  
   management data  
   warehouses, 544–546  
   mirroring models, 669  
   mirroring monitors, 676  
   monitoring. *See* moni-  
   toring  
   planning systems,  
   33–34  
   query execution,  
   208–209  
   replication and, 486,  
   489  
   reports, 538–543  
   secondary data and log  
   files, 287  
   SQL Server Profiler,  
   546–555  
   tempdb database,  
   285–286  
   trace data, 541  
   Tuning Advisor, 411–415  
   utility control points,  
   194–200  
 Performance Monitor  
   access, 536  
   configuration changes,  
   542  
   counters, 535–538



- current activity, 536
- data collector sets, 538–542
- database mirroring, 676
- performance logging, 538–542
- reports, 542–543
- vs. stored procedures, 522
- Performance Monitor Users group, 536
- permissions
  - BCP, 457–458
  - built-in and effective, 297–310
  - DDL statements, 336–342
  - external access options, 266–267
  - GRANT, REVOKE, and DENY, 336
  - granular, 351
  - guests, 167
  - implied, 312
  - inheriting through roles, 292
  - keywords, 295–297
  - log shipping, 656, 665
  - object, 306–311, 342–344
  - policy alerts, 163
  - policy evaluation and, 186
  - public server roles, 167
  - schemas, 350–351
  - script usage, 635
  - securable, 295–297
  - standard database roles, 331–333
  - startup service
    - accounts, 69
  - statement permissions, 311–312
  - tables, 367–368
  - user groups, 65–67
  - viewing, 240, 297–301
  - views, 388, 393–394
- PFS (Page Free Space), 356
- Physical Design Structures options, 413
- physical memory, 205, 206, 207
- pipeline parallelism, 28
- plan guides, 244–245
- planning
  - backup and recovery, 557–566
  - deployment, 33–40
  - log shipping, 655–665
  - replication system, 485–490
- platform information, 194
- points in time, restoring to, 587
- points of failure and work, 568, 569
- policies and Policy-Based Management, 39, 161–164
  - administering, 164–171
  - alert numbers, 163
  - categories, 162, 182, 183–185
  - cmdlets, 24
  - conditions, 162, 178–180, 182
  - creating policies, 181–183
  - documenting, 182
  - editing, 183
  - enabling or disabling, 183
  - evaluation modes, 162–163, 182, 185–188
  - facets, 161, 176–178
  - global settings, 199
  - history, 164, 185, 187–188
  - identifying for databases, 185
  - import/export functions, 164, 171–172
  - list of policies, 164–170
  - management servers, 164, 172–175
  - managing access, 58
  - mandating, 183–185
  - policy groups, 164
  - roles, 316
  - subscribing to, 164
  - targets, 162, 182
  - troubleshooting, 188–189
  - violations, 162
- policy groups, 164
- pooling threads, 214
- PowerShell. *See* SQL Server PowerShell; Windows PowerShell
- precision, 361, 371, 450
- predefined policies, 164–170
- predefined roles, 314–316
- prefix compression, 426–427
- primary data files, 234, 248, 278–279
- primary databases
  - backing up, 661–663
  - configuring, 657–660
  - failing over, 665–668
  - swapping, 667–668
- primary keys, 353
  - clustered indexes, 396
  - constraints, 415, 416–417
  - indexing and, 399
  - limitations, 235
  - merge replication, 489
  - partitioned views, 387
  - script usage, 635
  - transactional replication, 488
  - unique IDs, 381–383
- primary servers (log shipping), 654
- primary XML indexes, 397
- principal servers, 37, 668, 677–678
- priorities, configuring, 214–215
- private mail profiles, 606, 611–612
- privileged databases, 169
- privileges, 11
- process accounts, 507, 510, 515
- processes
  - current activity, 523
  - deadlocks, 157
  - killing, 159
  - monitoring, 151–152, 524
  - policies, 167
  - SSIS packages and, 434
- processing time, 525, 547
- processors
  - affinity, 167, 210, 211
  - assigning SQL Server to, 212
  - configuring, 210–214
  - determining number of, 194
  - displaying utilization, 199
  - parallel processing, 213
  - SQL requirements, 3–4
- product keys, 44
- product names (data providers), 470
- profiler.exe. *See* SQL Server Profiler
- profiles
  - mail, 606, 611–612
  - SQL Server Agent, 614
- profiling with stored procedures, 547
- programmability, 244
- prohibited attachments, 610
- projects, viewing, 13
- promoting transactions, 471, 476

- properties
  - data types, 379
  - distributors (replication), 497
  - evaluating expressions, 178–180
  - facets, 176–178
  - indexes, 400–401, 402, 410
  - list of database properties, 241
  - publications, 508–510
  - publishers, 493
  - replicating, 481
  - servers. *See* Server
  - Properties dialog box
  - subscriptions, 518
  - tables, 367–368
  - user-defined tables, 385
  - viewing, 240–241
  - views, 393–394
- protocol encryption, 81
- protocols, 82
- providers. *See* data providers
- public mail profiles, 606, 611–612
- publication* databases, 490–494, 498–500, 563, 564, 570
- publications, 481
  - access control lists, 510, 511
  - creating, 499–500
  - deleting, 512
  - merge publications, 504–508
  - monitoring, 526
  - properties, 508–510
  - scripts, 511
  - snapshot publications, 500–504
  - subscriptions, 509, 512–520
  - transactional, 500–504
  - updating, 508
  - viewing, 508
- published databases, 277
- publisher groups, 526–528
- publishers, 480
  - backing up, 563
  - creating publications, 499–500
  - disabling publishing, 499
  - distribution* databases and, 492
  - enabling and updating, 497–498
  - groups, 526–528
  - monitoring, 526–528
  - planning architecture, 486

- properties, 493
  - publishing subscriber model, 486
  - pull subscriptions, 509–510, 511, 512
  - push subscriptions, 512
- Q**
- queries
    - building, 445–447
    - configuring execution, 223–224
    - copying, 445–447
    - costs of, 213, 223–224
    - default connections, 217–219
    - determining types of, 398–399
    - distributed queries, 463, 464–466
    - executing, 110–112
    - job steps, 625
    - limitations, 236
    - Management Studio and, 109–112
    - MDX queries, 31
    - memory and, 203, 208–209
    - minimum size, 209
    - multiple server queries, 176
    - parallel processing and, 213
    - partitioned tables/indexes, 424–425
    - plan guides, 244–245
    - policy conditions, 178–180
    - Query Designer, 445–447
    - Query Editor, 13, 605, 632
    - query governor, 223–224
    - secondary server processing, 654
    - slow-running, 546
    - SSIS packages and, 434
    - star join queries, 31
    - system stored procedures, 97
    - templates, 13
    - time-outs, 471, 476
    - T-SQL views, 240–241
    - Query Designer, 445–447
    - Query Editor, 13, 605, 632
    - query governor, 223–224
    - Queue Reader Agent, 482, 503–504, 510

- queued updating (replication), 484
- queues, 2, 294, 295–297
- quorum states, 677–678
- quotation mark delimiters, 442
- quoted identifiers
  - ANSI compliance, 260
  - vs. bracketed, 138
  - SET options, 102, 104, 106, 107, 118

**R**

- R1 and R2 releases
  - control points and, 195
  - copying objects, 633–636
  - extracting applications, 198
  - R2 Datacenter, 5, 6–7
  - R2 Parallel Data Warehouse, 5, 7
  - table rows and size, 367
- RAID
  - backup needs, 558, 567–568
  - hardware and software, 34–36
  - RAID 0, 0 + 1, and 5, 4, 34–36
  - SQL Server requirements, 4
- RAM, 3, 194. *See also* memory
- read errors, 170, 525
- read-only databases, 121, 167, 264–265, 569, 585, 590
- read-only partitioned views, 387
- read-write databases, 264–265
- reattaching databases, 278–279
- rebuilding
  - indexes, 401, 411, 643, 645
  - master* database, 230–231
- recommendations (indexes), 414–415
- recovery
  - automated process, 578
  - backup strategies, 569–570
  - Bulk-Logged mode and, 462
  - complications in, 583
  - database corruption, 584–585

- files and filegroups, 591–593
- log shipping, 653, 655, 663, 664
- master database, 601–602
- mirroring, 653, 674–675, 678–680
- mission-critical databases, 558
- options, 267–269
- parallel backup/restore, 564–565
- partial restores, 594
- policy settings, 167
- problem resolution, 584–585
- rebuilding configuration, 228–231
- recovery states, 590
- restoring from backups, 586–591
- restoring to different locations, 593
- secondary data and log files, 287
- SQL Server models, 561–562
- standby servers, 594–596
- time-outs, 226–227
- T-SQL commands, 596–601
- undoing recoveries, 590
- recovery states, 590
- recurring jobs, 629
- recursive triggers, 120, 260
- recycling logs, 534
- Registered Servers view, 13, 127–137, 176
- registering
  - management servers, 173–174
  - operators, 619
  - servers, 130–137
  - subordinate servers, 174–175
- registration details, servers, 130–137
- registry changes, 542
- reinitializing subscriptions, 482, 519–520
- related tables, 354
- relational data warehousing, 28–30
- releases for SQL Server 2008.
  - See R1 and R2 releases
- remote access, 524, 655
- remote collation, 476
- remote connections, 76–77, 140, 141, 220, 238
- remote distributor model, 486
- remote logins, 474
- remote procedure calls, 466, 470, 476
- remote servers, 88
- remote service bindings, 294, 309
- remote stored procedures, 463, 466
- removable disk drives, 567
- removing. *See* deleting
- reorganizing
  - data pages, 272, 274
  - indexes, 411, 643, 645
- repairing
  - installations, 54, 649
  - master database, 230–231
- replacing databases, 588
- replaying traces, 551–555
- replication, 479–480
  - agents, 481–483, 510, 625
  - alerts, 615–616
  - articles, 481
  - backup plans, 563–564
  - benefits of, 480
  - cleanup agents, 482–483
  - components, 480–481
  - counters and, 535
  - Database Engine Services, 8
  - detaching databases and, 277
  - distributors, 481, 490–499
  - encryption and, 290
  - file streaming and, 71
  - jobs, 481–483
  - linked servers and, 463
  - Management Studio
    - access, 124
  - merge replication, 484
  - models for, 486
  - monitoring, 19, 518, 524, 525–535
  - planning system, 485–487
  - preliminary tasks, 487–490
  - publication subscriptions, 512–520
  - publications, 481, 499–508
  - publishers, 480
  - recovery process and, 589
  - snapshots, 483
  - subscribers, 481
  - transactional, 483
  - types of, 483–485
  - updating options, 484
- replication agents, 481–483, 510, 625
- Replication Agents Checkup job, 483
- Replication Monitor, 19, 518, 523, 525–528
- Replication Monitoring Refresher job, 483
- Replication Services, 242
- reply addresses, 608
- Report Manager, 2, 62
- Report Server, 2
- Report Server Web Service, 62
- Report Services Configuration Manager, 19
- Reporting Services, 2, 9, 32–33
  - access, 62
  - configuring instances, 51–52
  - importing policies, 171–172
  - initializing, 64
  - installing, 45
  - server groups, 127
  - user group, 67
- Reporting Services Key Management Utility, 19
- Reporting Services Utility, 19
- reports
  - data collector reports, 542–543
  - enabling, 62
  - error and usage, 52–53, 73–74
  - formats, 32–33
  - log shipping activity, 665
  - maintenance plans, 604, 646
  - management data warehouses, 546
  - performance logging, 538–542
  - policy compliance reports, 186
- residual information protection (RIP), 203
- Resource Governor, 30, 89
- resource managers, 467
- Resource process information, 157
- resources
  - control points, 194, 199–200
  - resource health, 196
  - resource managers, 467
  - wait times and blocks, 153

responding to alerts, 617–618

restarting services automatically, 615

Restore Database dialog box, 586–591

restore operations, 664. *See also* recovery

restoring data. *See* recovery

result sets, 391

resumable snapshot delivery, 488

retention period, 227, 497, 513, 563, 658, 662

RIP (residual information protection), 203

roles

- access and, 314–316, 329–336
- application databases, 333–334
- assigning, 326–329
- creating, 331–334
- deleting, 334
- login assignments, 326–329
- management data warehouses, 544
- managers, 331–333
- msdb* database, 316
- multiple logins, 330–331
- package encryption and, 451
- permissions, 309
- primary and secondary databases, 667–668
- removing users from, 329, 334
- revoking, 329
- schema ownership and, 348
- schema security and, 292
- security levels, 293
- as security principals, 292
- server roles, 313–314
- standard database, 331–333

rolling back transactions, 234, 467–468

rotation schedules, 568, 576

routes, 294, 310

row compression, 426–427

rows, 353

- in data pages, 354
- fixed vs. variable length, 356
- limitations, 237
- row offsets, 354

- row-overflow storage, 237–238
- size information, 367
- text in row option, 356
- updating, 418
- variable length columns, 354

Rows Data, 592

rowsets, 464

RPCs (remote procedure calls), 466, 470, 476

rsconfigtoo.exe (Report Services Configuration Manager), 19

rskeymgmt.exe (Reporting Services Key Management Utility), 19

rules, 312, 420–421. *See also* check constraints

“running exposed” state, 677

## S

sa login, 302, 303

sample databases, 241–242

SATA II, 33

saving

- policies, 178
- SSIS packages, 450–453
- traces, 551

scalability, 36–37

scalable shared databases (SSDs), 32

Scalar Types view, 89

scale, 361, 371, 450

schedule planning, 559

scheduled duty rosters, 620

scheduled jobs

- backups, 569–570, 658
- configuring, 604, 627–629
- copying, 636–637
- masters and targets, 638–640

scheduled maintenance plans, 643

scheduled performance counters, 538

scheduled policy evaluations, 187

scheduled replication updates, 516

scheduled reports, 62

scheduled rotation, 568

scheduled trace logs, 538

schema objects, 300, 347, 481

schema scope, 293–294

schema-contained objects, 347

schemas

- catalog views, 89
- creating, 349–350
- dropping, 353
- identifiers, 348
- lazy schema validation, 476
- modifying, 350–351
- moving objects in, 351–352
- object names, 634
- permissions, 310
- as securable, 294
- security and roles, 292
- synonyms and, 348
- user ownership, 348
- views in, 388
- Windows PowerShell, 348–349
- XML views, 91

Schemas view, 89

schemes, partition, 422–423

scopes, 293–294

Script Wizard, 633–636

scripts

- ActiveX job scripts, 625
- bulk copy program, 460–461
- copying objects, 633–636
- cursor options, 263–264
- initialization sample, 25–26
- jobs, 631
- migrating, 432
- packages and, 434
- PowerShell, 137–141
- publications, 511
- server-to-server tasks, 632
- signature execution policy, 22
- tables, 371
- T-SQL job scripts, 625
- views, 13–14

SCSI host adapters, 170

SDLT (Super DLT), 567

secondary data files, 234, 248, 278–279, 287

secondary databases (log shipping), 661–663, 665–668

secondary servers (log shipping), 562, 654, 656

secondary XML indexes, 397

securables, 292–294, 297–311

Secure Sockets Layer (SSL), 10, 608

- security
  - account types and, 11, 302–305
  - auditing levels, 201–203
  - authentication, 200–201, 301–302
  - component access, 58–62
  - database administration, 329–336
  - Database Engine, 49–50
  - database mirroring, 671
  - encryption types, 39
  - goals for, 291
  - linked servers, 473–475
  - logins, 302–305, 317–326
  - Management Studio access, 124
  - Performance Monitor, 536
  - permissions. *See* permissions
  - planning deployments, 39–40
  - policy execution and, 164
  - principals, 292–294
  - replication agents, 510
  - roles, 292, 312–316, 326–329
  - schemas in, 292
  - securables, 292–297
  - Security view, 90
  - special-purpose logins and accounts, 302–305
  - system stored procedures, 98–99
  - utility control points and, 200
  - viewing objects, 244
- security identifiers (SIDs), 292
- security principals, 292–294
- Security view, 90
- self-tuning, 85, 208
- sensitive data imports, 451
- serialized procedure execution, 485
- server clustering. *See* Cluster service; clustering
- server groups
  - adding servers to, 130
  - creating, 128–129
  - deleting, 129
  - editing or moving, 129–130
  - managing, 127–130
  - moving servers, 136–137
  - querying multiple servers, 176
  - registration details, 132–136
  - subordinate groups, 174–175
- server logins
  - assigning roles by, 326–329
  - controlling access, 329–336
  - creating, 319–321
  - editing, 317–318, 321–322
  - enabling, disabling, or unlocking, 324–325
  - GRANT or DENY access, 323–324
  - passwords, 326
  - removing, 325
  - security levels, 293
  - viewing and editing, 317–318
- Server Network Utility. *See* SQL Server Configuration Manager
- Server Properties dialog box
  - Active Directory information, 228
  - authentication and auditing properties, 200–203
  - backup compression, 228
  - connecting to servers, 192–194
  - current settings, 193
  - file streaming, 221–222
  - flushing cache, 227
  - index fill factor, 225–226
  - language settings, 222
  - memory, 203–209
  - nested triggers, 223
  - parallel processing, 210–214
  - query execution, 223–224
  - remote connections, 220
  - retention periods, 227
  - system and server information, 194
  - threading, priorities, and fibers, 214–215
  - time-outs, 226–227
  - user connections, 216–219
  - utility control points, 194–200
  - Year 2000 support, 224
- server roles, 294, 313–314
- server scope, 293–294
- server-based connections to data providers, 440
- servers. *See also specific types of servers (publishers, principals, subscribers, etc.)*
  - adding to server groups, 130
  - backing up, 572–583
  - central management, 172–175
  - central monitor servers, 659–660
  - centralized notifications and jobs, 637–640
  - configuring, 105–106, 191–194
  - connecting to, 12–13, 136
  - copying alerts, operators, and jobs, 636–637
  - database mirroring, 668–680
  - disconnecting from, 136
  - job scheduling, 638–640
  - linked servers. *See* linked servers
  - log shipping and, 653–655
  - Management Studio, 130–137
  - monitoring activity, 150–159
  - moving to new groups, 136–137
  - multiserver administration, 637–640
  - permissions, 310
  - policy categories, 163
  - PowerShell management, 137–141
  - registration information, 130–137
  - as resource managers, 467
  - server objects, 124
  - server scope, 293–294
  - server-to-server tasks, 632–637
  - single-user mode, 604–605
  - viewing information, 194
- server-to-server tasks, 632–637
- Server-Wide Configuration view, 91

- service accounts
  - customizing, 47
  - database mirroring, 673
  - log shipping configuration, 655
  - startup service accounts, 69
  - types of, 10–11
  - user groups, 65–67
- Service Broker, 2, 21, 59, 244, 295–297
- Service Broker view, 90
- service master keys, 147
- services
  - automatic startup, 143–144
  - configuring, 62–74
  - file streaming, 70–73
  - log shipping configuration, 655
  - managing, 67–68
  - permissions, 311
  - restarting automatically, 615
  - as securable, 294
  - startup service accounts, 69
- Services Manager. *See* SQL Server Configuration Manager
- Services utility, 65, 67
- SET commands, 217–219, 269–270
- SET options, 101, 102–104
- severity levels (errors), 529, 617
- SGAM (Shared Global Allocation Map), 355
- shared features, installing, 8
- shared folders, 656
- Shared Global Allocation Map (SGAM), 355
- Shared Memory protocol, 77, 81–82
- SharePoint Server, 32
- shrinking databases, 166, 247–248, 258, 271–275, 643, 646
- shutdowns, 170, 585
- Simple Object Access Protocol (SOAP), 59
- simple parameterization, 261–263
- simple recovery, 268, 561
- single quote delimiters, 442
- single-user databases, 265, 585
- single-user mode, 604–605
- size
  - columns, 361
  - database files, 166
- SMP (symmetric multiprocessing), 4, 212
- SMTP accounts, 606, 607, 608, 614
- Snapshot Agent, 481–482, 487, 503–504, 507, 510, 517
- snapshot publications, 500–504, 508
- snapshot replication, 483, 487–488
- snapshots
  - detaching databases and, 277
  - distributors, 493
  - filegroups, 71
  - FTP snapshots, 509
  - options, 509
  - transactional replication, 488–489
  - Tuning Advisor, 411
- SOAP (Simple Object Access Protocol), 59
- software RAID, 34–36
- Solutions Explorer view, 13
- sort orders, 593
- sources. *See also* data providers
  - data providers, 447–450
  - data replication. *See* replication
  - database mirroring servers, 668
  - job shipping and, 596
  - log shipping databases, 653–655
  - mapping columns in, 447–450
  - product names, 470
  - restoring databases, 587
  - SSIS packages, 435–443
  - standby servers and, 596
- sparse columns, 361–362, 380–381
- spatial data types, 373
- spatial indexes, 396
- spatial properties (indexes), 402
- sp\_configure stored procedures, 101, 111, 217–219, 476, 489
- sp\_dboption procedure. *See* ALTER DATABASE statement
- special characters, 349, 358, 442
- sp\_serveroption procedure, 475–476
- sp\_spaceused procedure, 367, 524
- sp\_trace stored procedures, 547
- sp\_who stored procedures, 523, 524
- SQL Active Directory Helper, 9, 63, 66
- SQL Diagnostics Utility, 20
- SQL Full-Text Filter Daemon Launcher, 9
- SQL Mail, 60, 99, 614. *See also* Database Mail
- SQL Native Client, 57, 440, 464
- SQL plan guide, 244
- SQL Query Command-Line Utility (SQLCMD), 15–18, 20
- SQL Server 2000, 505, 656–657
- SQL Server 2005, 432, 504
- SQL Server 2008. *See also* Compact edition; Developer edition; Enterprise edition; Express edition; Mobile edition; R1 and R2 releases; Standard edition; Web edition; Workgroup edition
  - administrative tools, 12–15
  - authentication, 110
  - command-line tools, 16–21
  - configuring and tuning, 85
  - data provider connections, 436
  - editions of, 5–8
  - event logs, 528–535
  - failover clusters, 595
  - hardware architecture for, 2–5
  - importing data to or from, 430, 433
  - linked servers, 472
  - localized versions of, 222
  - logins, 302, 303
  - object values and limits, 235–237
  - recovery models, 561
  - repairing, 54
  - replication, 492
  - security level, 292–293
  - SSIS package scripts, 434
  - subscriptions, 504
  - uninstalling, 54
  - upgrading, 54
  - user groups, 66
  - Windows modifications and services, 8–11

- SQL Server Agent, 9, 20, 612
  - alerts, 612, 613, 616–617
  - configuring, 47, 142, 604, 613–614
  - creating new logs, 534
  - distributors, 493
  - event forwarding, 637–638
  - event log monitoring, 528–532
  - initializing, 64
  - job definitions, 622–623
  - job histories, 664–665
  - job monitoring, 526
  - job scheduling, 638–640
  - jobs, 612, 613, 631
  - log shipping functions, 654–655
  - logs as monitoring tool, 524
  - mail, 610, 614
  - Management Studio
    - access, 124
    - msdb* database, 242
    - operators, 612, 613, 619–621
    - policy evaluation jobs, 162, 189
    - restarting services, 615
    - system stored procedures, 99–100
    - user groups, 66
- SQL Server authentication, 110
- SQL Server Books Online, 45
- SQL Server Browser, 9, 65, 67
- SQL Server Configuration Manager, 14, 15
  - access, 57
  - configuring services, 63–67
  - database mirroring, 670
  - file streaming, 72–73
  - running, 62
  - services and start mode, 67–68
  - starting up services, 145
  - startup parameters, 145–148
- SQL Server Database Engine. *See* Database Engine
- SQL Server Database Engine Tuning Advisor, 15, 20, 398, 411–415
- SQL Server Execution Package Utility, 20, 431, 432
- SQL Server Failover Cluster Wizard, 595
- SQL Server Import And Export Wizard
  - building queries, 445–447
  - copying tables or views, 442–445
  - creating packages, 434–435
  - data formats, 447–450
  - moving packages, 433–434
  - saving and executing packages, 450–453
  - sources and destinations, 435–443
  - starting and running, 434–435
- SQL Server Installation Center, 15, 40, 41
- SQL Server Integration Services (SSIS), 2, 9, 27–28, 430
  - applications for DACs, 198
  - components, 430
  - data providers and, 433 vs. DTS 2000, 27–28
  - dump files, 431
  - encryption, 452
  - executing packages, 433–434, 450–453
  - initializing, 64
  - installing, 45
  - Integration Services Designer, 431
  - job steps, 625
  - roles, 316
  - saving packages, 450–453
  - server groups, 127
  - SQL Server SSIS Package Utilities, 21
  - tasks and functions, 429–430
  - tools, 431–432
  - user groups, 67
- SQL Server Internet Server API, 509
- SQL Server Log Shipping Agent, 20
- SQL Server Management Studio, 15, 123–124. *See also* Server Properties dialog box
  - access, 323, 329–336
  - administration, 329–336
  - altering databases, 251–252
  - ANSI compliance, 259–261
  - automatic options, 257–259
  - backups, 571, 572–577
  - compression, 427–428
  - configuration, 191–194
  - connecting to databases or instances, 125–126
  - cross-database chaining, 266–267
  - cursor options, 263–264
  - database information, 239–240
  - Database Mail, 607
  - database manipulation, 275–280
  - database states, 264–265
  - distributors, 490–494
  - DTC, 142–143
  - encryption, 290
  - event logs, 532–535
  - file sizes, 270–275
  - group management, 127–130
  - identity values, 382–383
  - indexes, 402–404, 407–408, 416
  - linked servers, 469–471
  - logins, 317–321, 324, 325
  - new databases, 245–249
  - online, offline, or emergency modes, 266
  - parameterization, 261–263
  - partitions, 423–424, 425
  - permissions, 336–337, 342–345
  - Policy-Based Management, 162, 171–189
  - publications, 500, 508–510
  - queries, 109–112
  - recovery, logging, and I/O error checking, 267–269
  - revoking rights, 329
  - role assignments, 326–329
  - rules, 420–421
  - schema management, 349–353
  - server management, 130–137
  - SQL Server Agent, 142
  - tables, 358–369
  - Tuning Advisor, 15, 20, 398, 411–415
  - user access, 264–265
  - user-defined data types, 377–378
  - user-defined tables, 384
  - views, 12–13, 389–391, 392

- SQL Server PowerShell, 20–26. *See also* Windows PowerShell
- SQL Server Profiler, 15, 20, 99, 398, 546–551
- SQL Server Replication Diff Tool, 21
- SQL Server service (Windows), 9
- SQL Server Setup
  - adding components, 53
  - deployment, 40–55
  - launching, 40
  - new server instances, 41–42, 53
  - repairing installations, 54
  - running, 42–53
  - user groups for services, 65–67
- SQL Server SSIS Package Utilities, 21
- SQL Server Startup, 143–144
- SQL Server Utility, 194–200
- SQL Server VSS Writer, 9, 65
- SQL variant data type, 373
- SQL-92, 263, 464
- sqlagent.exe. *See* SQL Server Agent
- SqlClient .NET Data Provider, 39
- sqlcmd.exe (SQL Query Command-Line Utility), 15–18, 20
- sqldiag.exe (SQL Diagnostics Utility), 20
- sqllogship.exe (SQL Server Log Shipping Agent), 20
- sqlmaint.exe (Database Maintenance Plan Utility), 19
- SQLmonitor.exe (Replication Monitor), 19, 518, 523, 525–528
- sqlps.exe (SQL Server PowerShell), 20–26
- sqlserv.exe. *See* Database Engine
- SQLServr.exe, 149–150
- SQLXML, 433
- SQLXMLOLEDB data provider connections, 438
- ssbdiagnose.exe (Service Broker Diagnostics Utility), 2, 21, 59, 244, 295–297
- SSDs (scalable shared databases), 32
- SSIS. *See* SQL Server Integration Services (SSIS)
- SSIS Package Upgrade Wizard, 432
- SSL (Secure Sockets Layer), 10, 608
- standard configuration, 112–113
- standard database roles, 331–333
- Standard edition (SQL Server), 5–6, 206–207
- standard procedure execution, 485
- standard roles, 314
- standard views, 243
- standby servers, 562, 594–596, 604, 653
- star join queries, 31
- start mode, 67–68
- starting jobs, 631
- startup parameters, 145–148
- startup service accounts, 69
- statement permissions, 311–312
- statements
  - configuration options, 101
  - troubleshooting, 546
- static TCP/IP configurations, 79–80
- statistics
  - automatically creating, 258
  - maintenance plan tasks, 643, 646
  - monitoring, 525
  - performance counters, 535–538
  - recomputing, 401
  - script usage, 635
  - as securable, 294
  - startup parameters, 148
  - table-valued variables, 384
  - tracking, 148, 237
  - updating, 258
- status messages, 676
- steps (jobs), 623–627
- stoplists, 244–245
- stopwords, 244–245
- storage
  - indexes, 402
  - viewing objects in, 244
- stored procedures. *See* extended stored procedures; system stored procedures
- storing traces, 548
- strings, 235, 236, 260
- striped backups, 577–578
- subordinate server groups, 174–175
- subordinate servers, 174
- subqueries, 236
- subscribers, 481, 486, 500, 514, 563
- subscribing
  - to policies and policy groups, 164
  - to policy categories, 185
- subscription databases, 516, 517, 563, 570
- subscriptions, 512–513
  - access control lists (ACLs), 511
  - anonymous, 512–513
  - cleaning up, 482
  - creating, 513–517
  - deactivated, 513
  - deleting, 518
  - expired, 513
  - initializing, 503, 516, 517
  - maintaining, 518
  - monitoring, 526
  - properties, 518
  - publication settings, 509
  - pull and push subscriptions, 512
  - reinitializing, 519–520
  - retention periods, 513
  - security, 515
  - subscribers, 481, 486, 500, 514, 563
  - subscription databases, 516, 517, 563, 570
  - synchronizing, 513
  - types of, 509–510
  - updating, 518
  - validating, 518–519
- subsets of data, 386–387
- Super DLT, 567
- supplemental logging, 269
- surface security, 58–62
- suspect databases, 166, 277, 585
- swapping primary and secondary databases, 667–668
- switches (BCP), 454, 456–457
- symbols, 349, 358, 442
- symmetric keys, 169, 294, 311
- symmetric multiprocessing (SMP), 4, 212
- Sync Framework, 2, 45
- synchronizing
  - Database Mirroring Monitor, 523
  - databases after failover, 665–668
  - forcing failovers, 678–680
  - log shipping databases, 655



- logins in database mirroring, 669
- replication changes, 480
- standby servers, 595
- subscriptions, 513
- SYNCHRONIZING state, 678
- synchronous mirroring models, 669, 679
- synonyms, 243, 294, 295–297, 348
- sys user accounts, 302, 305
- sysadmin fixed server role, 38, 303, 311, 313–314
- sys.indexes catalog, 394
- system catalog, 86–87, 87–93
- system databases, 241–242, 558, 570, 643
- system event messages list, 170
- system information views, 194
- SYSTEM logins, 302, 304
- system stored functions, 87
- system stored procedures. *See also* extended stored procedures
  - cursor options, 263–264
  - distributed queries and, 466
  - limitations, 235, 236, 237
  - list of, 94–100
  - Management Studio queries, 109–112
  - monitoring current activity, 524
  - as objects, 243
  - permissions, 295–297, 312
  - profiling, 547
  - in publications, 502
  - replication and, 485
  - trace data, 550
  - types of, 93–94
  - viewing server information, 194
- system tables, 91–93, 169, 243, 426
- system views, 91–93, 243
- system-defined data sets, 538

## T

- T-SQL statements
  - access, 323, 337–341
  - ANSI compliance, 259–261
  - automatic options, 257–259

- backups, 572, 578–582
- catalog information, 87
- cmdlets, 21–26, 137–139, 238
- copying objects, 633–636
- cursor options, 263–264
- data type selection, 376
- database manipulation, 275–280
- database states, 264–265
- distributed transactions, 468
- encryption, 289–290
- file sizes, 270–275
- file streaming, 70–73, 221
- globally unique IDs, 383
- importing data, 461–462
- indexes, 404–407, 408–410
- job scripts, 625
- linked servers, 471–473
- logins, 318, 320, 321–322, 324–326
- maintenance plan tasks, 642
- vs. managed code, 28–29
- new databases, 249–251
- parameterization, 261–263
- partition switching, 358
- plan guides and, 244
- policy conditions, 178–180
- PowerShell, 137
- querying multiple servers, 176
- recovery, logging, and I/O error checking, 267–269
- recovery commands, 596–601
- revoking access, 337–341
- roles, 327–328, 329, 335–336
- rules, 421
- schema management, 350–353
- single-user mode, 604–605
- SQLCMD tool and, 17
- system stored procedures, 94
- table management, 358–372
- table-valued variables, 383–386

- trace data, 550
- user access, 264–265
- user-defined data types, 378–379
- user-defined tables, 384–385
- view commands, 240–241, 391–393
- Table Designer, 361
- table joins, 354, 399, 506
- Table Properties view, 360
- Table view, 360
- tablediff.exe (SQL Server Replication Diff Tool), 21
- tables
  - BCP import or export, 459
  - column constraints and rules, 415–420
  - compression, 357–358, 425–428
  - copying, 368–369, 442–445, 633–636
  - creating, 358–364
  - current values in, 368
  - data pages, 354–356
  - data types, 373
  - DBCC commands, 650, 651
  - default values, 380
  - deleting, 369–370
  - deleting columns, 370
  - elements of, 353–354
  - extents, 357
  - fields, 376–377
  - global temporary tables, 358–359
  - identities and unique IDs, 381–383
  - joined tables, 506
  - limitations, 235–237
  - list of system views, 91–93
  - list of tables, 92–93
  - lock escalation, 364
  - modifying, 364–367, 423
  - names, 358–359, 369
  - native data types, 371–373
  - nulls, 379–380
  - as objects, 243
  - partitions, 357–358, 421–425
  - permissions, 295–297, 312, 367–368
  - properties, 360, 367–368
  - in publications, 502, 505
  - recovering, 591, 650, 651
  - relationships, 354

- tables, *continued*
  - replicating, 481
  - row and size information, 367
  - Rows Data, 592
  - scripting, 371
  - as securable, 294
  - sparse columns, 380–381
  - SQL Server 2000 log shipping, 656–657
  - Table Designer, 361
  - Tuning Advisor, 411–415
  - user-defined data types, 377–379
  - user-defined table types, 383–386
- table-valued variables and parameters, 383–386
- tape backups, 226–227, 566, 587
- tape library systems, 567
- tapes, ejecting, 575
- target databases, 653–655
- target servers, 668
- target systems, 552, 638–640
- targets
  - compliance evaluation, 187
  - policies, 162, 163, 182
- tasks
  - checklists, 641–642
  - copying, 636–637
  - maintenance plans, 644–646
  - server-to-server tasks, 632–637
  - SSIS packages, 434
- TCP ports, 608, 671
- TCP/IP configuration, 78–84
- TDE (Transparent Data Encryption), 288–290
- tempdb* database, 169, 242, 285–286, 290, 570
- temperature of hardware, 34
- Template Explorer view, 13
- TEMPLATE plan guide, 244
- templates
  - data collectors, 539
  - model* database, 288
- text data, 374, 394, 489
- text files, 433, 440–443
- Text/Image (large object data types) data pages, 356
- third-party components, 432
- threads, 168, 210, 214–215, 555
- time data types, 374
- time stamps, 373, 489, 588
- time-outs, 170, 226–227, 471, 476
- timing replication updates, 487
- traces
  - analyzing, 553–554
  - creating, 548–550
  - debugging, 554
  - modifying, 551
  - replaying, 551–555
  - saving, 551
  - stopping and starting, 551
  - storing, 548
  - target systems, 552
  - trace data, 541, 551
  - trace definitions, 549, 551
  - trace events, 546
  - trace files, 398, 411, 548
  - trace flags, 148
  - trace tables, 548
- tracing policies, 168
- tracking changes. *See* change tracking
- traffic loads, 522
- transaction log backups, 560
  - adding to existing sets, 576
  - creating, 582–583
  - log sequence numbers and, 588
  - log shipping, 654–655
  - points of failure and work, 568
  - recovery, 584
- transaction log shipping. *See* log shipping
- transaction logs, 233–238
  - backing up before restoring, 586
  - encryption and, 290
  - location, 248
  - log shipping. *See* log shipping
  - permissions, 312
  - preventing errors, 288
  - reattaching databases, 279–281
  - settings, 267–269
  - size of, 248, 268
  - startup parameters, 147
  - transactional replication, 488
  - truncating, 575, 577, 580–583
- transaction managers, 467. *See also* Distributed Transaction Coordinator
- transaction safety, 669, 677
- transactional publications, 500–504, 508, 519
- transactional replication, 483, 488–489, 564
- transactions
  - committing or rolling back, 467–468
  - distributed transactions, 463
  - transaction safety, 669, 677
  - uncommitted, 666–667
- Transact-SQL statements. *See* T-SQL statements
- transfer rates, 33
- transforming data. *See also* SQL Server Import And Export Wizard
  - bulk copy utility (BCP), 453–461
  - BULK INSERT command, 461–462
  - imported or exported data, 447–450
  - SSIS and, 429–434
- transparent data encryption (TDE), 39, 288–290
- tree structure of indexes, 394–395
- triggers
  - alerts, 613, 616
  - configuration, 223
  - cursor options, 263–264
  - limitations, 236, 237
  - nested, 162, 189, 223, 236
  - as objects, 243
  - policies and, 162, 189
  - recursive, 120, 260
  - replicating, 481
  - script usage, 635
  - updating partitioned views, 387
- troubleshooting. *See also* monitoring
  - blocking connections, 157–158
  - configuration, 228–231
  - corrupt databases, 575
  - database mirroring, 653, 676–677
  - deadlocks, 157–158
  - event logs, 528–532
  - EventID messages, 170
  - “filegroup is full” error, 288
  - integrity checks, 648–652
  - log shipping, 653, 665–668

- master* database, 230–231, 601–602
- online, offline, or emergency modes, 266
- policies, 188–189
- recovering and restoring, 583–601
- SQL Server Profiler, 546–555
- traces, 548–550
- transaction log backups, 582–583
- transaction log errors, 288
- truncated data, 449
- truncated transaction logs, 268, 575, 577, 580–581, 582, 583
- truncating databases, 274
- Trusted Computer System Evaluation Criteria standard, 201
- trusted connections, 201
- TRUSTWORTHY option, 267
- tuning
  - Database Engine Tuning Advisor, 15, 20, 398, 411–415
  - self-tuning, 85, 203, 208
- types
  - permissions, 311
  - as securable, 294

## U

- UCPs (utility control points), 43, 194–200
- UDDTs (user-defined data types), 377–379, 634
- UDM (Unified Dimension Model), 30–31
- Ultra320 SCSI, 33
- UMDWs (utility management data warehouses), 195, 200, 544–546
- uncommitted transactions, 666–667
- Undo files, 590
- Unicode collation, 593
- Unicode locales, 593
- Unicode string literals, 262
- Unicode text, 458
- Unified Dimension Model (UDM), 30–31
- uniform extents, 357
- uniform resource names (URNs), 23
- uninstalling SQL Server, 54–55
- uninterruptible power supplies (UPSs), 4–5
- unique constraints, 415, 416
- unique identifiers, 373, 381–383
- unique keys, 635
- uniqueidentifier columns, 502, 506
- unloading tapes, 575
- unsupported statements, 634
- untrusted connections, 201
- unused space, 272, 274
- updatable views, 393
- UPDATE statement, 29–30, 213, 261, 393, 489
- updating
  - distributors, 495–496
  - job categories, 632
  - partitioned views, 387
  - publications, 500–504, 508
  - rows, 418
  - server registration details, 132
  - statistics, 258, 643, 646
  - subscriptions, 518
  - system tables, 169
  - timing replication, 487
  - transactional replication, 488–489
- upgrading
  - applications for DACs, 198
  - connection strings, 432
  - migrating legacy data, 431–432
  - SQL Server, 54
  - SQL Server 2000 log shipping, 656–657
- UPSs (uninterruptible power supplies), 4–5
- URNs (uniform resource names), 23
- USB drives, 567
- USE statement, 108, 635
- user access settings, 264–265. *See also* access; accounts
- user accounts, 301–302. *See also* accounts
- user commands, 159
- user databases, 230–231, 558, 569, 643
- user groups, 65–67
- user IDs, 10
- user keys, 451
- user tables, 243
- user views, 243

- user-defined application roles, 314
- user-defined data sets, 538
- user-defined data types, 377–379, 634
- user-defined functions, 61, 312
- user-defined standard roles, 314
- user-defined table types, 383–386
- users
  - monitoring, 524
  - permissions, 311
  - removing from roles, 329, 334
  - reporting current activity, 523
  - schema ownership, 348
  - as securable, 294
  - security levels, 293
  - standard roles, 331–333
  - viewing permissions, 300
- utility control points (UCPs), 43, 194–200
- Utility Explorer view, 195, 199–200
- utility management data warehouses (UMDWs), 195, 200, 544–546

## V

- vacations, 620–621
- validating subscriptions, 518–519
- values
  - check constraints and, 418–420
  - columns, 361
  - current tables, 368
- varbinary(max) data type, 354, 356, 375, 394, 488
- varchar(max) data type, 354, 356, 374, 394, 488
- variable length rows, 356
- variable-length fields, 376–377
- verifying backups, 575
- versions of SQL Server. *See* editions of SQL Server
- VIA support, 77
- views, 386–387
  - binding to schemas, 390
  - computed, 399–400
  - copying, 430, 633–636
  - creating, 387–392
  - database mirroring, 675
  - filtered indexes and, 398

views, *continued*  
 limitations on, 388  
 modifying, 392–393  
 as objects, 243  
 partition information, 424–425  
 permissions, 295–297, 312, 388, 393–394  
 properties, 393–394  
 as securable, 294  
 table views, 360  
 updatable, 393  
 virtual address space, 147, 207  
 virtual files, 525  
 virtual memory configuration, 203  
 virtual servers, 595  
 virtualized deployments, 41–42  
 Visual Basic, 433  
 Visual Studio 2010, 197, 198  
 Visual Studio Tools for Applications, 432  
 Volume Shadow Copy Service (VSS), 9  
 volume utilization, 199  
 VSS (Volume Shadow Copy Service), 9  
 VSTA (Microsoft Visual Studio Tools for Applications), 432

## W

Wait Resource values, 153  
 Wait Time values, 153, 157  
 Wait Type values, 153, 157

warm standby servers, 562, 595–596  
 warnings. *See* alerts  
 Web Assistant, 60. *See also* Reporting Services  
 Web edition (SQL Server), 7  
 Web synchronization, 509  
 weekly backups, 558  
 weekly jobs, 629  
 weekly maintenance tasks, 641  
 wide mode (BCP), 458  
 Windows  
   application logs, 528–532  
   SQL Server modifications and services for, 8–11  
 Windows authentication, 110, 168, 186, 301–302, 608  
 Windows collation names, 249  
 Windows Event Log policies, 170  
 Windows Fibers, 215  
 Windows groups, 292, 293  
 Windows Installer, 7  
 Windows Management Framework, 140  
 Windows PowerShell indexes, 410–411  
   installing tools, 140, 141  
   integrating, 21–26  
   providers and, 139  
   remote computer access, 238  
   schema access, 348–349  
   server management, 137–141

  table management, 358  
   view commands, 388  
 Windows Server 2003, 204, 530–531  
 Windows Server 2008, 530–531  
 Windows-level security, 292–293  
 WinRM 2.0, 140, 141  
 witness servers, 37, 88, 668, 671–672, 677–678  
 worker threads, 168, 214–215  
 workflow containers, 434  
 Workgroup edition (SQL Server), 5  
 working copies of databases, 593  
 workloads, 89, 480  
 write errors, 525

## X

XML, Analysis Services usage, 31  
 XML data types, 373, 394  
 XML indexes, 237, 396, 397  
 XML schema collections, 294, 311  
 XML Schemas view, 91  
 XML stored procedures, 100  
 XQuery scripts, 24, 137

## Y

Year 2000 support, 224

# About the Author

---



**WILLIAM R. STANEK** ([www.williamstanek.com](http://www.williamstanek.com)) has more than 20 years of hands-on experience with advanced programming and development. He is a leading technology expert, an award-winning author, and a pretty-darn-good instructional trainer. Over the years, his practical advice has helped millions of programmers, developers, and network engineers all over the world. He has written more than 100 books. Recent books include *Microsoft Exchange Server 2010 Administrator's Pocket Consultant*, *Windows PowerShell 2.0 Administrator's Pocket Consultant*, and *Windows Server 2008 Inside Out* (all from Microsoft Press).

William has been involved in the commercial Internet community since 1991. His core business and technology experience comes from more than 11 years of military service. He has substantial experience in developing server technology, encryption, and Internet solutions. He has written many technical white papers and training courses on a wide variety of topics. He frequently serves as a subject matter expert and consultant.

William has a BS in computer science, magna cum laude, and an MS with distinction in information systems. He is proud to have served in the Persian Gulf War as a combat crewmember on an electronic warfare aircraft. He flew on numerous combat missions into Iraq and was awarded nine medals for his wartime service, including one of the United States of America's highest flying honors, the Air Force Distinguished Flying Cross. Currently, he resides in the Pacific Northwest with his wife and children.

William recently rediscovered his love of the great outdoors. When he's not writing, teaching, or making presentations, he can be found hiking, biking, backpacking, traveling, and/or trekking the great outdoors in search of adventure.

Follow William on Twitter at [WilliamStanek](#).