

Microsoft

Microsoft
SQL Server 2008

Mike Hotek



**Microsoft® SQL Server®
2008 Step by Step**

Mike Hotek

To learn more about this book, visit Microsoft Learning at
<http://www.microsoft.com/MSPress/books/12859.aspx>

9780735626041

Microsoft®
Press

© 2009 Mike Hotek. All rights reserved.

Table of Contents

3 Using the Tools in SQL Server 2008	31
SQL Server Documentation	31
Management Tools in SQL Server 2008	34
SQL Server Configuration Manager	35
SQL Server Management Studio	38
Database Mail	42
Performance Management Tools	44
Profiler	44
Database Engine Tuning Advisor	44
Performance Studio	45
Business Intelligence Tools	46
Business Intelligence Development Studio	46
Chapter 3 Quick Reference	47



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:

www.microsoft.com/learning/booksurvey/

Chapter 3

Using the Tools in SQL Server 2008

After completing this chapter, you will be able to

- Select the appropriate SQL Server 2008 tool for a given task
- Manage SQL Server 2008 services
- Launch, navigate, and utilize SQL Server Management Studio
- Utilize several shortcuts for better productivity
- Configure Database Mail

SQL Server 2008 ships with eight stand-alone tools used to configure, manage, and monitor SQL Server services. Within this group of eight core tools, you can also design SQL Server objects and execute code. The most wide-reaching tool, SQL Server Management Studio (SSMS), contains four additional tools designed for management and monitoring.



Note You can install instances of SQL Server Integration Services, Reporting Services, Analysis Services, and the Database Engine. To simplify the terminology, we will simply refer to a Database Engine instance as a SQL Server instance. All other instances will be referred to as either an SSIS, SSAS, or SSRS instance.

SQL Server Documentation

SQL Server 2008 ships with a very comprehensive Books Online. While many do not consider documentation as a “tool,” the saying “information is power” immediately comes to mind. There is a reason that you will hear someone say: “Read the manual.” Underservedly, Books Online has received a very bad reputation.

Books Online should be your primary source for information concerning SQL Server 2008, after this book of course. Books Online contains detailed explanations of every feature within SQL Server, syntax on every command, and thousands of code samples that you can apply. Additionally, SQL Server Books Online integrates a vast array of online content into the local documentation in order to provide extensive, constantly updated information that can be applied within your environment. While we will not spend a significant amount of time going through Books Online, we will explain a couple of very useful and often-overlooked features.

Hidden within Books Online, underneath the How Do I Link button, lurks a set of several dozen comprehensive tutorials that walk you through important feature sets such as policy-based management, *hierarchyID* data types, designing OLAP cubes, implementing Data mining models, deploying replication, and building SSIS packages or SSRS reports, as shown in Figure 3-1.

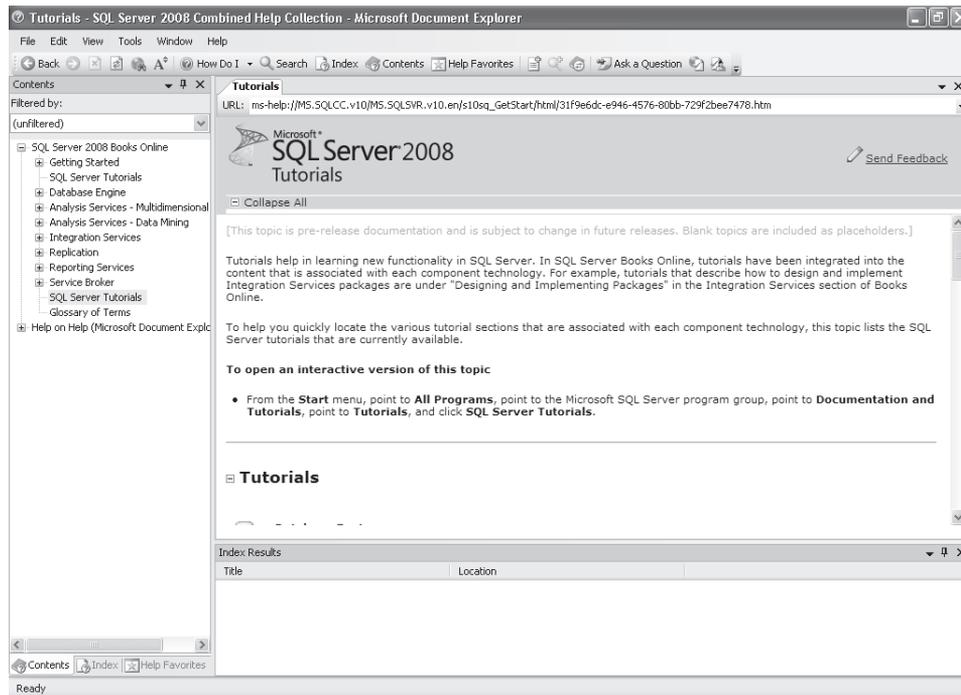


FIGURE 3-1 SQL Server 2008 tutorials

You can also see a set of three links at the far right of the toolbar in Figure 3-1. The first of these three links, Ask A Question, will launch a browser window into the right-hand pane and take you to the Microsoft Developer Network (MSDN) forums. The MSDN forums allow you to ask any question about SQL Server for which you cannot find an answer. Questions are answered by volunteers who include thousands of SQL Server professionals around the world as well as hundreds of members of the SQL Server development team.

The second-to-last link will again take you to the MSDN forums, but will automatically apply a filter to display only those questions you are participating in so that you can easily follow up on the status of your questions.

The last link will launch a browser window in the right-hand pane and take you to Microsoft Connect where you can post feedback on SQL Server or a particular feature

as well as publish bug reports. The bug reports posted are continuously reviewed by the SQL Server development team so that they can proactively supply patches to functionality. The product feedback plays an important role during the planning of the next version of SQL Server.

You can bookmark topics that you reference frequently by using the Help Favorites feature. The Help Favorites feature allows you to also save frequently executed searches for later recall.

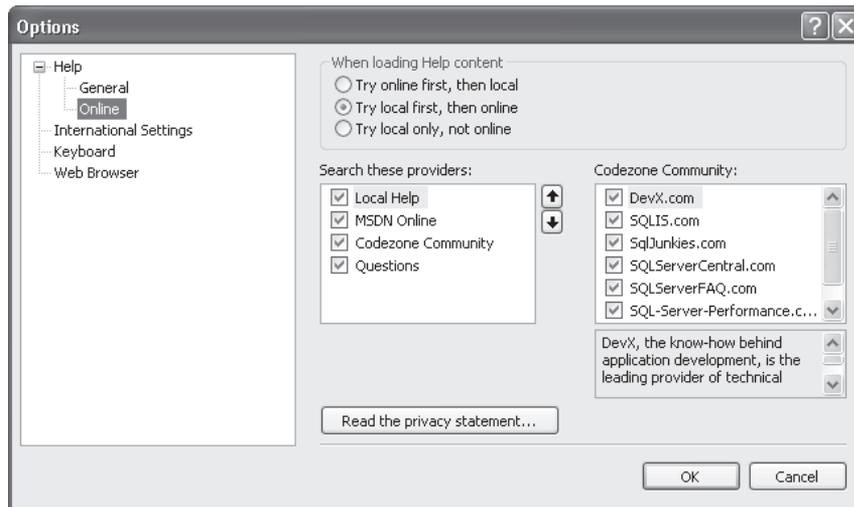


Note Because SQL Server Books Online contains features to integrate the local version of help with various Web sites, you can experience some performance issues. To mitigate any performance issues, you should set Books Online to look at the local help first before trying online resources.

In this procedure, you will configure Books Online for optimal performance.

Configure Books Online

1. Launch Books Online by selecting Start | All Programs | Microsoft SQL Server 2008 | Documentation And Tutorials | SQL Server Books Online.
2. Select Tools | Options.
3. Select Online and configure as shown here.



Management Tools in SQL Server 2008

SQL Server 2008 ships with a set of nine tools for managing SQL Server instances and interacting with data: OSQL, SQLCMD, Tablediff, Bulk Copy Program (BCP), SQLDiag, Resource Governor, SQL Server Configuration Manager, SSMS, and Database Mail.

OSQL is a command line utility which was added to SQL Server 2000 as a replacement for ISQL. OSQL allows you to connect to and execute queries against a SQL Server instance without requiring the overhead of a graphical interface.



Important OSQL has been deprecated as of SQL Server 2005. While OSQL is still available in SQL Server 2008, you should rewrite any OSQL routines to utilize SQLCMD.

SQL Server 2005 introduced SQLCMD as the command line query interface that replaced OSQL. While OSQL allowed you to submit interactive queries from a command line along with very limited automation capabilities, SQLCMD provides a rich automation interface complete with variable substitution and dynamic code creation/execution.



Note A discussion of the specifics of SQLCMD is beyond the scope of this book. For details on SQLCMD, please see the Books Online topic "SQLCMD Utility."

You can use Tablediff.exe to compare the data between two tables. Tablediff can be run to alert you if the data or structure of two tables is different. Additionally, Tablediff can generate a script file containing the statements necessary to bring the destination table into synchronization with the source table. Tablediff is primarily used within replication architectures.

The BCP utility is the oldest utility within the SQL Server product, dating all the way back to the very first version of SQL Server. BCP has been enhanced with each successive version to handle new data types and named instances, but the speed and feature set has not changed. BCP is used to export data from a table to a file as well as import data from a file into a table. If your import and export needs are reasonably simple and straightforward, BCP should be the only utility that you would need. For more advanced import and export capabilities, you should utilize SQL Server Integration Services (SSIS).



Note You will learn about BCP and its import cousin, BULK INSERT, in Chapter 10, "Data Manipulation." You will learn about SSIS in Chapter 24, "Business Intelligence."

SQLDiag is a utility that collects diagnostic information about a SQL Server instance. SQLDiag is designed to capture Windows performance counters, event logs, SQL Server Profiler traces, SQL Server blocking, and SQL Server configuration information. Primarily used as a data

collection engine for Microsoft Customer Service and Support (CSS) to troubleshoot SQL Server issues, the data collected can also be used by a DBA to analyze SQL Server performance and stability issues.



Note The details of SQLDiag are beyond the scope of this book. Please refer to the Books Online article “SQLDiag Utility” for more information.

Resource Governor is a new feature in SQL Server 2005 that is found within SSMS. The purpose of Resource Governor is to allow a DBA to configure rules around resource allocation such as processor or memory that is then applied to specific queries, users, or groups of users. The goal of Resource Governor is to allow high-priority workloads to take priority over other workloads in order to provide the best response based on user expectations.



Note You will learn about Resource Governor in the article, “Performance and Data Capture Tools,” which can be found on the Microsoft Press Online Windows Server and Client Web site at www.microsoft.com/learning/books/online/serverclient.

SQL Server Configuration Manager

Shown in Figure 3-2 on page 36, SQL Server Configuration Manager is responsible for managing SQL Server services and protocols. The primary tasks that you will perform with SQL Server Configuration Manager are:

- Start/Stop/Pause/Restart a service
- Change service accounts and passwords
- Manage the startup mode of a service
- Configure service startup parameters

Once you have completed the initial installation and configuration of your SQL Server services, the primary action that you will perform within SQL Server Configuration Manager is to periodically change service account passwords. When changing service account passwords, you no longer have to restart the SQL Server instance for the new credential settings to take effect.



Important Windows Service Control Applet also has entries for SQL Server services and allows you to change service accounts and passwords. You should never change service accounts or service account passwords using the Windows Service Control Applet. SQL Server Configuration Manager needs to be used, because SQL Server Configuration Manager includes the code to regenerate the service master key that is critical to the operation of SQL Server services.

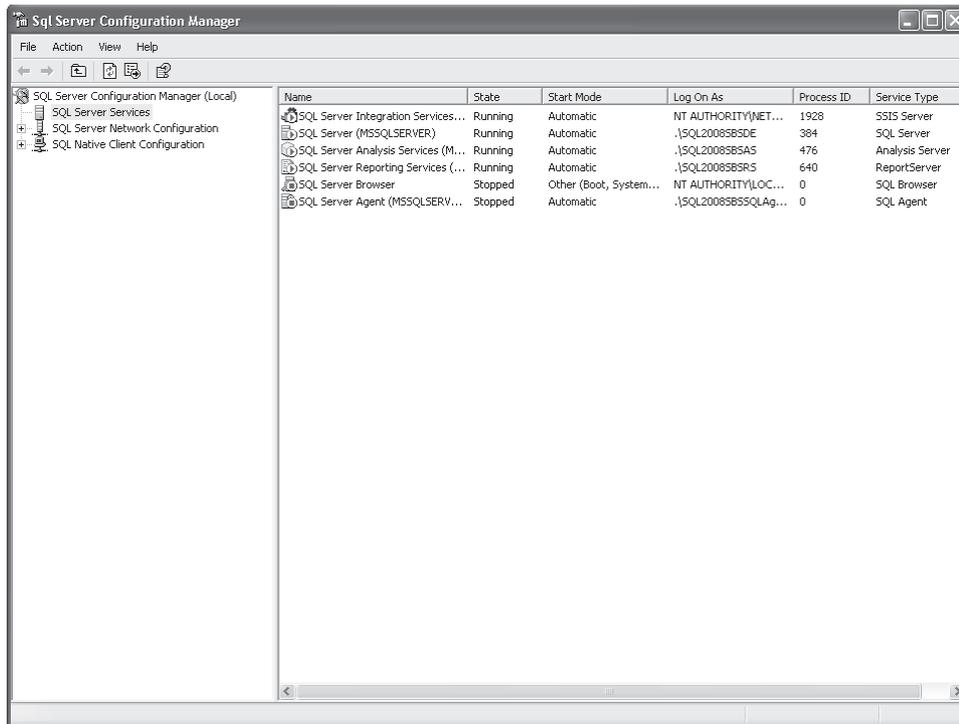


FIGURE 3-2 List of services within SQL Server Configuration Manager

While you can Start, Stop, Pause, and Restart SQL Server services, SQL Server has extensive management features which should ensure that you rarely, if ever, need to shut down or restart a SQL Server service.

In this procedure, you will review the options and settings available for SQL Server services.

Review Service Options

1. Launch SQL Server Configuration Manager by selecting Start | All Programs | Microsoft SQL Server 2008 | Configuration Tools | SQL Server Configuration Manager.
2. In the left-hand pane, highlight SQL Server Services.
3. Double-click the SQL Server service in the right-hand pane to display the Properties dialog box shown here.
4. Review the options on each of the tabs.
5. Click Cancel to close the Properties dialog box without making any changes.



SQL Server Configuration Manager also allows you to configure the communications protocols available to client connections. In addition to configuring protocol-specific arguments, you can also control whether communications are required to be encrypted or whether an instance will respond to an enumeration request, as shown in Figure 3-3.



FIGURE 3-3 Protocol properties



Tip Applications can broadcast a special command, called an enumeration request, across a network to locate any SQL Servers that are running on the network. While being able to enumerate SQL Servers is valuable in development and testing environments where instances can appear, disappear, and be rebuilt on a relatively frequent basis, enumeration is not desirable in a production environment. By disabling enumeration responses by setting the Hide Instance to Yes, you prevent someone from using discovery techniques to locate SQL Servers for a possible attack.

SQL Server Management Studio

SQL Server Management Studio is the core tool that you will be spending a large part of your time using. SSMS provides all of the management capabilities for SQL Server services along with the ability to create and execute Transact-SQL (TSQL), Multidimensional Expression (MDX) query language, Data Mining Extensions (DMX), and XML for Analysis (XMLA) code. This section will provide a brief overview of SSMS, as shown in Figure 3-4, to get you started. Each subsequent chapter within this book will extend your knowledge of SSMS capabilities.

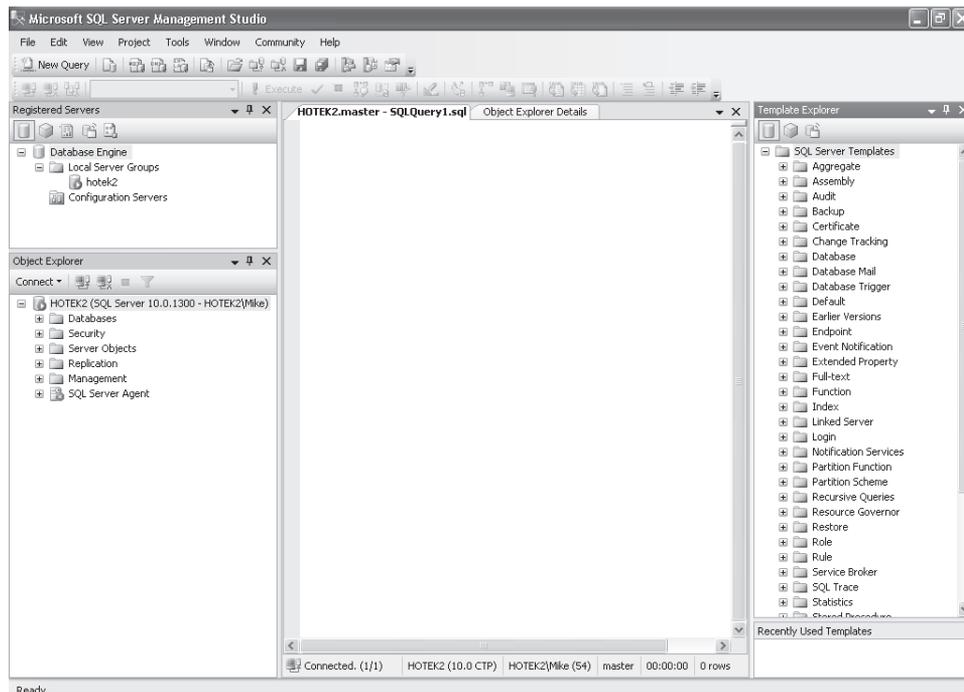


FIGURE 3-4 SQL Server Management Studio

In the following practice, you will launch SSMS and connect to the Database Engine instance that you installed in Chapter 2, “Installing and Configuring SQL Server 2008.”

Launch SSMS and Connecting to an Instance

1. Launch SSMS by selecting Start | All Programs | Microsoft SQL Server 2008 | SQL Server Management Studio.
2. When the Connect To Server dialog box is displayed, accept the default options and click Connect.



Note Because you have only installed a default instance at this point, this dialog should default to Database Engine for the server type, <machinename> for the server name, and Windows Authentication for the authentication option. Now that you have connected to an instance within SSMS, for all remaining exercises in this book, we will assume that you can perform these steps and will not repeat them.

SSMS has a variety of windows that you can open and position within the interface in order to access various feature sets.

The Registered Servers window provides a place to store connection information for all of the SQL Server services within your environment. Once stored, you can right-click any server and launch a connection to the server in either the Object Explorer or a query window.

The Template Explorer, shown in the right-hand pane of Figure 3-4, enables access to hundreds of predefined templates to create, alter, or drop objects as well as query various objects using TSQL, MDX, XMLA, or DMX. You can use the templates that ship with SQL Server, modify the templates to include your organization-specific coding standards, and add additional templates or template groups.

The Community menu on the toolbar allows you to launch a browser window into the center pane to access the MSDN forums and Microsoft Connect in the same way as previously described for Books Online.

The Tools | Options menu on the toolbar will display the Options dialog box, as shown in Figure 3-5 on page 40, so that you can set up the SSMS environment specifically the way you want to work.

Configure the SSMS Environment

1. Select Tools | Options from the toolbar.
2. Expand the Environment tree and select the General node. Use the At Startup drop-down list to configure the startup look and feel of SQL Server Management Studio.
3. Expand Text Editor | All Languages | Tabs.
4. Set the Tab Size to 4.

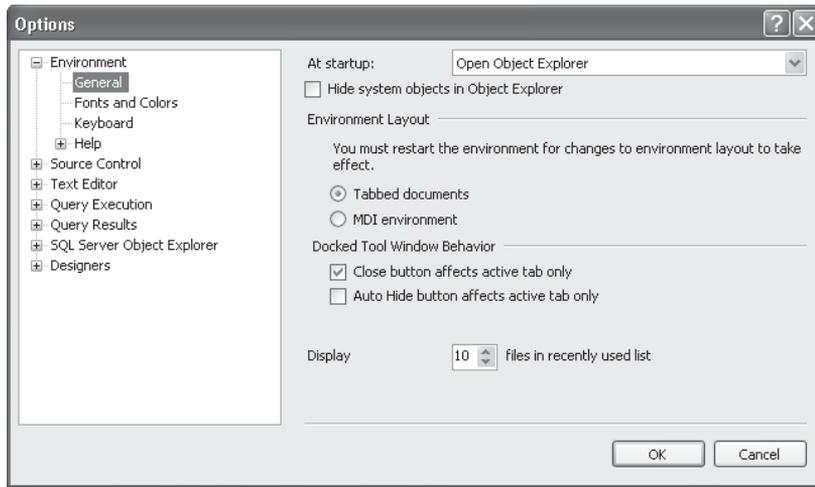


FIGURE 3-5 Options dialog box of the SQL Server Management Studio

5. Set the Indent Size to 4.
6. Select the Insert spaces option.
7. Explore the rest of the options that are available for configuration.
8. Click OK to save your settings.



Tip When you set SSMS to start up with an empty environment, you will not see a Connect To Server dialog and SSMS will immediately start. You will then need to explicitly connect to an instance for the Object Explorer or query window through either the Registered Server pane, File | Connect Object Explorer, or the New Query button. By setting the tab size and insert spaces options, SSMS will automatically replace any tabs with spaces in a query window, allowing you to more easily format and align code even when using a proportional font.

As you can see from Figure 3-6, the Object Explorer provides access to practically any action that you wish to perform against any SQL Server object. You will be using the functionality within the Object Explorer throughout virtually every chapter in this book.

Two additional capabilities of SSMS are object summaries and built-in reporting capabilities. The Object Explorer Details tab will display summary information according to the object that is currently selected within the Object Explorer. SSMS Reports, shown in Figure 3-7, allow you to display either Standard Reports that ship with SQL Server or to access your own custom reports that have been designed using the Reporting Services Report Designer that you will learn about in the article, "Reporting Services," which can be found on the Microsoft Press Online Windows Server and Client Web site at www.microsoft.com/learning/books/online/serverclient.

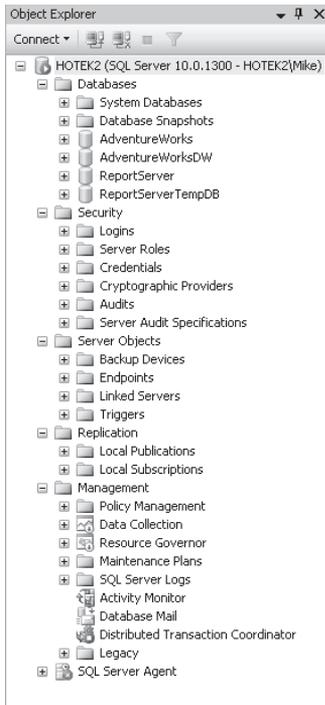


FIGURE 3-6 Object Explorer for a SQL Server instance

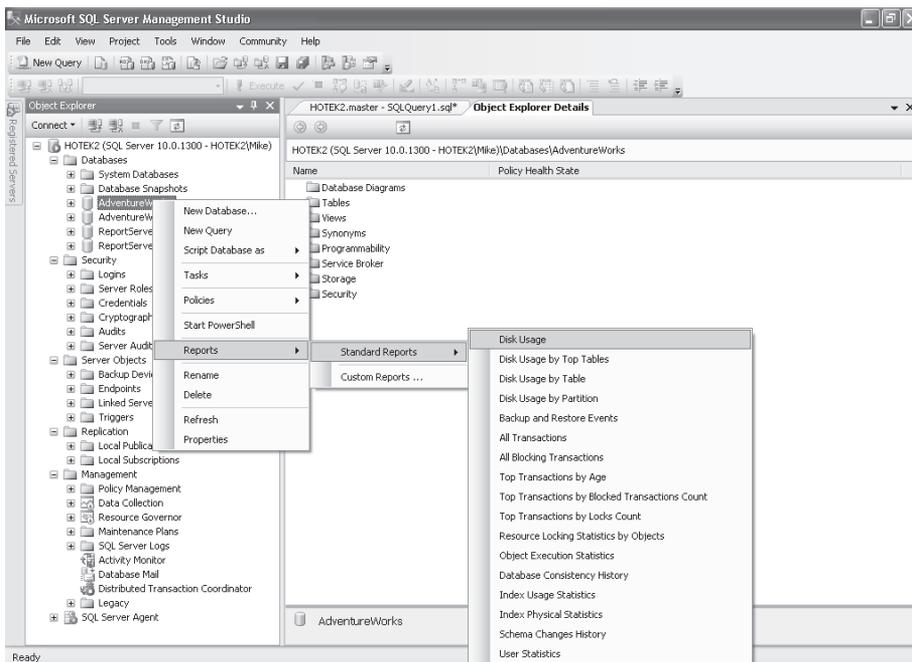


FIGURE 3-7 SSMS Reports

Database Mail

Database Mail enables a SQL Server to send outbound mail messages. While messages can contain the results of queries, Database Mail is primarily used to send alert messages to administrators to notify them of performance conditions or changes that have been made to objects. In the procedure that follows, you will learn how to configure Database Mail.

Configure Database Mail

1. Click the New Query button to open a new query window and execute the following code to enable the Database Mail feature:

```
EXEC sp_configure 'Database Mail XPs',1
GO
RECONFIGURE WITH OVERRIDE
GO
```
2. Within the Object Explorer, open the Management Node, right-click on Database Mail, and select Configure Database Mail.
3. Click Next on the Welcome screen.
4. Select the Set Up Database Mail by Performing the Following Tasks option and click Next.
5. Specify a name for your profile and click Add to specify settings for a mail account.
6. Fill in the account name, e-mail address, display name, reply e-mail, and server name fields on the New Database Mail Account page.
7. Select the appropriate SMTP Authentication mode for your organization and, if using Basic Authentication, specify the user name and password. Your settings should look similar to the following:

The screenshot shows the 'New Database Mail Account' dialog box. It has a title bar with a close button. The main area contains the following fields and options:

- Account name: mhotek@mssqlserver.com
- Description: (empty)
- Outgoing Mail Server (SMTP):
 - E-mail address: mhotek@mssqlserver.com
 - Display name: Michael Hotek
 - Reply e-mail: mhotek@mssqlserver.com
 - Server name: (empty) Port number: 25
- This server requires a secure connection (SSL)
- SMTP Authentication:
 - Windows Authentication using Database Engine service credentials
 - Basic authentication
 - User name: (empty)
 - Password: (empty)
 - Confirm password: (empty)
 - Anonymous authentication

At the bottom, there are three buttons: OK, Cancel, and Help.



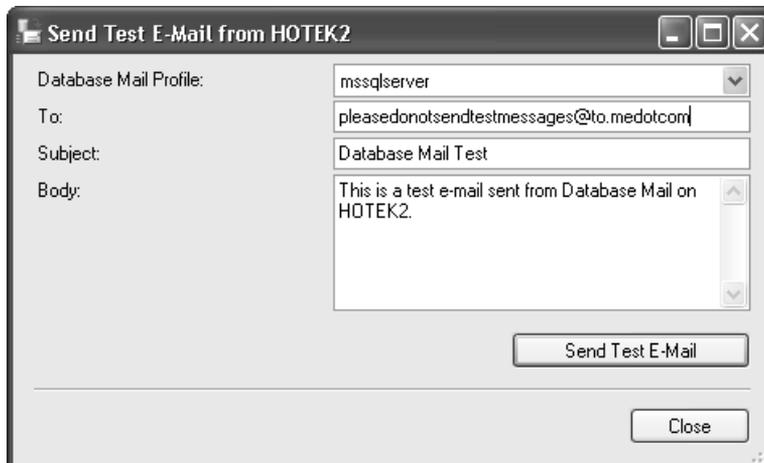
Note Your screen should look similar to the settings in the graphic. I am using my Internet e-mail account and have purposely left the Server Name, User Name, and Password out of the graphic. You will need to fill in the Server Name field if you are using an internal mail server.

8. Click OK and then click Next.
9. Check the box in the Public column next to the profile you just created and set this profile to Yes in the Default Profile column and click Next.
10. Review the settings on the Configure System Parameters page and click Next.
11. Click OK, then click Next, and then click Finish.
12. The final page should show success for all four configuration steps; click Close.
13. Within Object Explorer, right-click SQL Server Agent item and select Start from the shortcut menu in order to start the SQL Server Agent service, if it is not already running.



Note Database Mail utilizes the services of SQL Server Agent to send messages as a background process. If SQL Server Agent is not running, messages will accumulate in a queue within the msdb database.

14. Right-click Database Mail and select Send Test E-mail from the shortcut menu.
15. Select the Database Mail Profile you just created, enter an e-mail address in the To: line, and click Send Test E-Mail.



16. Go to your e-mail client and verify that you have received the test mail message.

Performance Management Tools

In addition to configuration and management tools discussed previously, SQL Server 2008 ships with three specialized tools for capturing, analyzing, and troubleshooting performance data.

Profiler

SQL Server Profiler is a graphical tool that acts as an interface to the SQL Trace Application Programming Interface (API). Profiler allows you to define SQL Server events, as shown in Figure 3-8, that you want to capture information on. You can also specify filtering options to target your data capture within the events that you have specified. You will learn about Profiler in the article, “Performance and Data Capture Tools,” which can be found on the Microsoft Press Online Windows Server and Client Web site at www.microsoft.com/learning/books/online/serverclient.

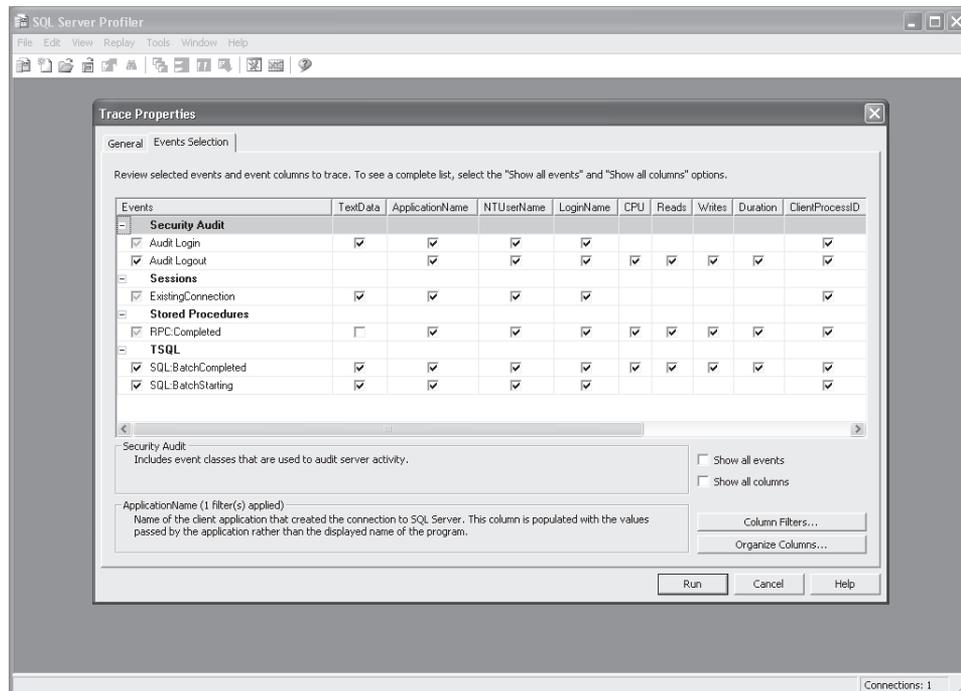


FIGURE 3-8 SQL Server Profiler Trace Properties dialog box Events selection

Database Engine Tuning Advisor

Database Engine Tuning Advisor (DTA) analyzes a query workload and makes recommendations on index and partitioning changes that can improve the performance of your queries (as shown in Figure 3-9). You will learn about indexes in Chapter 6, “Indexes,” partitioning in

Chapter 7, “Partitioning,” capturing a query workload in the article, “Performance and Data Capture Tools,” which can be found on the Microsoft Press Online Windows Server and Client Web site at www.microsoft.com/learning/books/online/serverclient, and how to apply DTA in the article, “Performance Analysis Tools,” which can be found on the Microsoft Press Online Windows Server and Client Web site at www.microsoft.com/learning/books/online/serverclient.

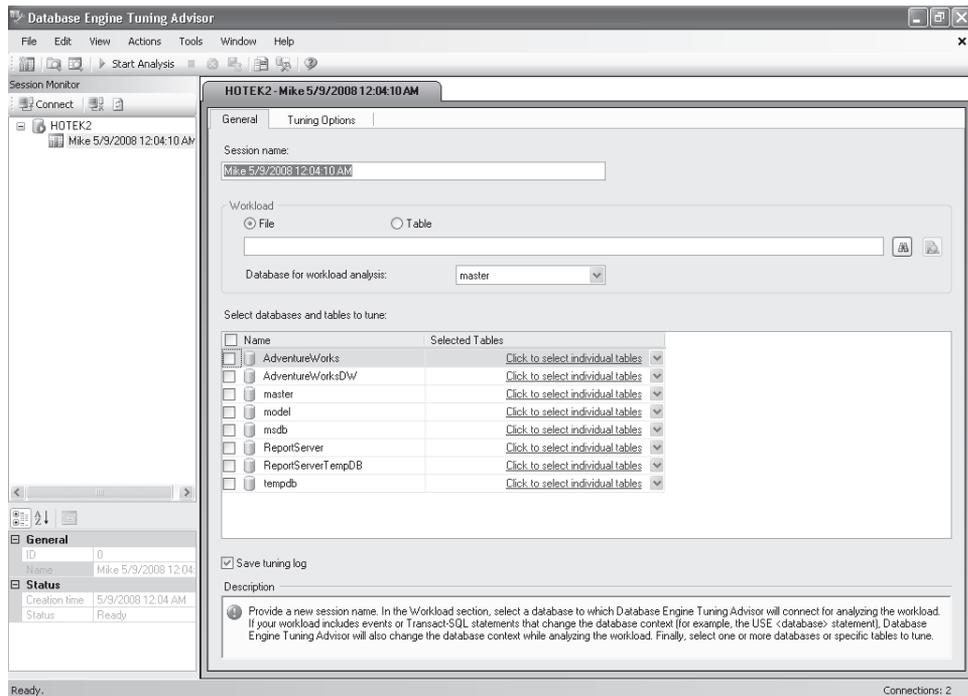


FIGURE 3-9 DTA tuning workload definition

Performance Studio

Performance Studio is the name given to a collection of technologies within SQL Server 2008 that are targeted at the analysis of enterprise-wide performance data. The components of Performance Studio are:

- Performance Data Warehouse
- Data Collectors
- Performance Reports

The Performance Data Warehouse is a database that you create. Data Collectors are SSIS packages, which are executed on a scheduled basis using SQL Server Agent. Performance Reports are a set of Report Designer reports written against the data stored in the Performance Data Warehouse.

You will learn how to configure, manage, and leverage the components of the Performance Studio in the article, "Performance Analysis Tools," which can be found on the Microsoft Press Online Windows Server and Client Web site at www.microsoft.com/learning/books/online/serverclient.

Business Intelligence Tools

Management of SSIS, SSRS, and SSAS occurs within SSMS. However, development of packages, reports, report models, OLAP cubes, and Data mining models occurs within the Business Intelligence Development Studio (BI Dev Studio).

Business Intelligence Development Studio

The BI Dev Studio is the Visual Studio 2008 shell with support for SQL Server 2008 BI projects, as shown in Figure 3-10. Each of the BI projects will be explored in Chapter 24, "Integration Services," Chapter 25, "SQL Server Reporting Services," and Chapter 26, "SQL Server Analysis Services."

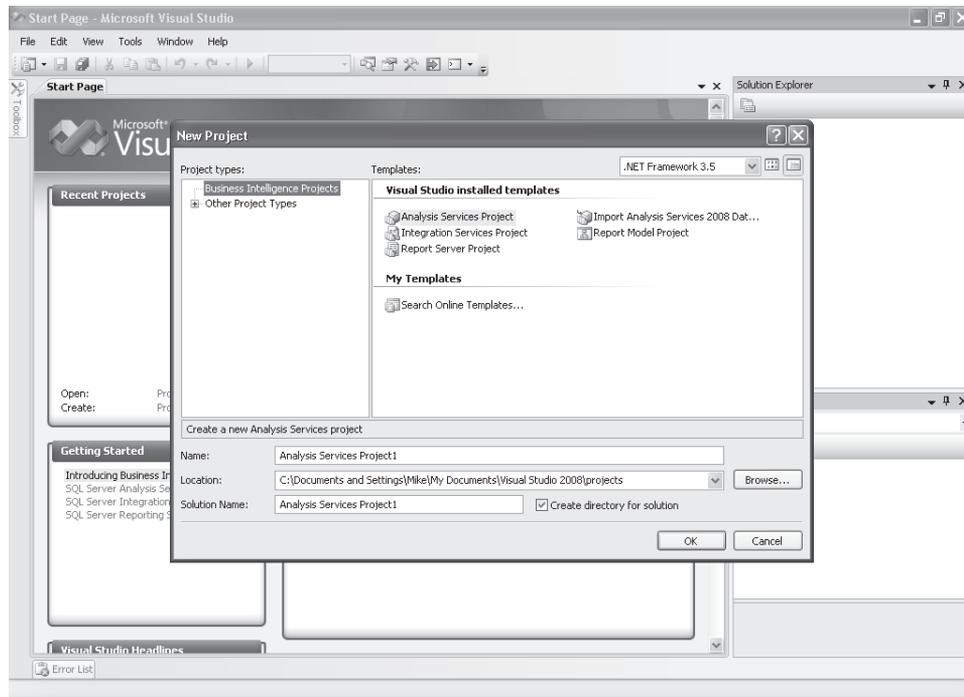


FIGURE 3-10 BI Dev Studio projects

Chapter 3 Quick Reference

To	Do This
Manage a SQL Server instance, OLAP cubes, Data mining models, Integration Services, or Reporting Services	Connect to the appropriate service within the Object Explorer in SQL Server Management Studio
Write and/or execute T-SQL, MDX, or DMX code	Open the appropriate query window (T-SQL, MDX, or DMX) and connect to the instance you want to write or execute code against
Configure and manage Database Mail	Connect to a Database Engine instance
Get help on a topic	Launch SQL Server Books Online. If you are writing a query, you can highlight a term and use SHIFT+F1 to launch Books Online to the highlighted term
Start, Stop, and Pause a service or manage service accounts	Open SQL Server Configuration Manager and either use the Start, Stop, or Pause buttons on the toolbar or double-click the service to access the property sheet
Limit the resources a particular query or user can consume	Configure the Resource Governor within SSMS
Build SSIS, SSRS, or SSAS projects and deploy them to a SQL Server instance	Open BI Dev Studio, create a new project, and design the objects that you wish to deploy
Manage Performance	Use Profiler to capture query workloads, DTA to analyze workloads captured with Profiler, and the Performance Studio to automate the capture of performance metrics