Chapter 1

Introducing SQL Server 2008

In This Chapter

- Understanding database basics
- Choosing a SQL Server 2008 edition
- Using SQL Server components
- Implementing SQL Server databases
- Finding additional information in SQL Server references

SQL Server 2008 is Microsoft's enterprise-class database server, designed to compete with products such as Oracle and IBM's DB2. According to a Gartner study, SQL Server is rapidly gaining momentum, possessing more than 17 percent of the worldwide database market in 2006.

SQL Server allows you to store, retrieve, and manipulate data to meet your organization's business objectives. The platform provides a number of tools and technologies to assist you in managing and manipulating your data on your own terms. For example, using SQL Server 2008, you can

- Import and export data from a variety of file formats
- Link to other databases (both SQL Server and those of other manufacturers)
- Manipulate data from within Microsoft Excel and Microsoft Access
- Produce professional-quality dynamic reports based on SQL Server data
- Create automated tasks that trigger when data satisfies specified conditions

That's only scratching the surface of the functionality offered by SQL Server 2008! In this chapter, I focus on the basic knowledge you need to get started with SQL Server.

Starting Off on the Right Foot

There are a couple of decisions you need to make if you're building a new SQL Server installation. Before making an investment of time or money, take a few moments to think about the following questions:

- What SQL Server edition effectively balances your business needs against cost?
- What hardware and software platform are best suited for your SQL Server installation?

I help you answer these questions in this section.

Examining SQL Server editions

SQL Server is a complex product with a wide variety of services. Most organizations need only a subset of that functionality. Rather than charge a single high price for a one-size-fits-all software package, Microsoft offers SQL Server 2008 in a variety of editions, ranging from the low-end (but free!) Express Edition to the expensive, fully functional Enterprise Edition.

The right edition for your organization will depend upon your data processing needs. In fact, many organizations host a combination of several different SQL Server editions, used for different purposes.

Table 1-1 summarizes the differences between the various SQL Server 2008 editions.

Table 1-1 Comparing SQL Server Editions

Feature	*Express*	*Workgroup*	*Standard*	*Enterprise*
Maximum Processors	1	2	4	Unlimited
Maximum RAM	1GB	3GB	Unlimited	Unlimited
Maximum Database Size	4GB	Unlimited	Unlimited	Unlimited
Database Mirroring	No	No	Yes	Yes
Log Shipping	No	Yes	Yes	Yes
Merge Subscriber	Yes	Yes	Yes	Yes
Merge Publisher	No	No	Yes	Yes

Feature	*Express*	*Workgroup*	*Standard*	*Enterprise*
Oracle Replication	No	No	No	Yes
SQL Agent	No	Yes	Yes	Yes
SQL Profiler	No	No	Yes	Yes
Analysis Services	No	No	Yes	Yes
Advanced Analytics	No	No	No	Yes
Partitioning	No	No	No	Yes
Data Compression	No	No	No	Yes
Resource Governor	No	No	No	Yes
Cost (per processor)	Free	$3,899	$6,000	$25,000

The prices listed in Table 1-1 are current as of the initial release date for SQL Server 2008 and are subject to change.

Table 1-1 presents only a high-level view of some common differences between the two platforms. For a complete feature comparison, see `http://msdn.microsoft.com/en-us/library/cc645993(SQL.100).aspx`.

One more SQL Server edition is available: Developer Edition. This edition is designed for application developers and offers functionality exactly the same as Enterprise Edition at an incredibly low price point of $50 per developer. What's the catch? You can use it only for development purposes. You may not use it in a production environment (even for disaster recovery purposes).

Microsoft plans to release two more editions of SQL Server 2008: Express Edition with Tools and Express Edition with Advanced Services. These two editions will include additional functionality.

Checking system requirements

Before you install SQL Server 2008, you need to verify that the hardware you intend to use meets Microsoft's minimum requirements for running SQL Server. In this section, I outline the requirements for each SQL Server edition.

Operating system

All editions of SQL Server 2008 will run on the following operating systems with at least the service pack (SP) level indicated:

- Windows Server 2003 Standard, Enterprise, or Data Center edition with SP2
- Windows Vista Ultimate, Home Premium, Home Basic, Enterprise, or Business
- Windows XP with SP2 (or later)
- Windows Small Business Server 2003 with SP2

Processor

SQL Server requires a minimum of a 1 GHz processor, but Microsoft recommends the use of 200 GHz or faster processors.

TIP

Microsoft charges per *physical processor* for SQL Server licenses. Current processor technology allows manufacturers to build multiple *cores* on the same physical processor. Each core is effectively an individual processor. So-called "dual core" processors include two discrete processors on the same chip, and "quad core" processors include four computing cores. Microsoft adopted a very generous licensing policy (unlike that of Oracle and IBM) that allows you to purchase licensing on a physical processor basis, regardless of the number of cores on those processors. Therefore, take this into account when choosing your hardware platform. You'll be much better off financially if you choose a single quad-core processor instead of four single-core processors!

Memory

The bare minimum amount of memory needed to run SQL Server 2008 is 512MB. Microsoft recommends a minimum of 2GB, but I suggest adding as much memory as your budget allows.

Hard drive

You need about 350MB of free hard drive space for SQL Server's software components. If you intend to install optional (but useful!) components such as SQL Server Books Online (described later in this chapter) or sample databases, plan on having about 1GB free. Don't forget that this is the requirement for SQL Server itself; you'll still need to save space to store your data!

Display

SQL Server 2008 requires at least a VGA (1024 x 768 pixels) video adapter and monitor.

Software

Before installing SQL Server, be sure you've installed the .NET Framework 3.5.

Understanding the Basic Components of SQL Server

You should begin your SQL Server 2008 adventure with a basic understanding of the components of SQL Server and their purposes. In this section, I explain how each of the major SQL Server components interact to help you manage your installation and manipulate data.

SQL and Transact-SQL

The Structured Query Language (SQL) is the language of databases. Any interaction between a user, program, or server and a database takes place through the use of SQL, even if the actual SQL code is buried deep within a graphical environment.

All major relational databases today (SQL Server, Oracle, Microsoft Access, IBM DB2, and so on) implement the same basic SQL commands. This common language allows database developers to easily migrate between platforms and create links between disparate database environments.

That said, every manufacturer of database software adds its own customizations to support functionality unique to its platform. Microsoft uses the name Transact-SQL (sometimes abbreviated as T-SQL) to refer to its extended version of SQL. Similarly, Oracle calls its enhanced version PL/SQL.

I provide an in-depth exploration of both SQL and Transact-SQL in Parts III and IV of this book.

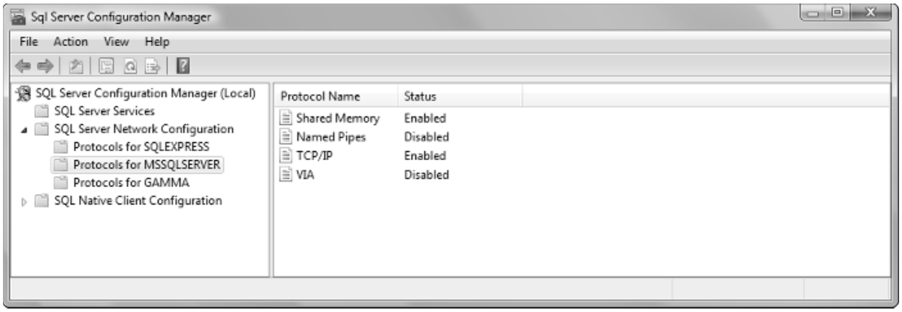
SQL Server components

SQL Server provides a number of tools that facilitate your interactions with SQL Server. Each is designed for a specific set of tasks, although they do have some degree of overlap.

SQL Server Configuration Manager

SQL Server Configuration Manager (shown in Figure 1-1) allows you to perform basic administrative tasks that affect the configuration of your SQL Server installation. For example, this tool allows you to do the following:

- Start, stop, pause, and restart SQL Server services
- Configure the use of network protocols to access SQL Server
- Configure SQL Server Native Client connectivity


Sql Server Configuration Manager
File
Action
View
Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration
Protocols for SQLEXPRESS
Protocols for MSSQLSERVER
Protocols for GAMMA
SQL Native Client Configuration

Protocol Name	Status
Shared Memory	Enabled
Named Pipes	Disabled
TCP/IP	Enabled
VIA	Disabled

Figure 1-1: SQL Server Configuration Manager.

I discuss the use of SQL Server Configuration Manager in Chapter 3.

SQL Server Management Studio

SQL Server Management Studio (SSMS), shown in Figure 1-2, is the database administrator's primary interface to SQL Server 2008. It offers a fully functional management interface, allowing you to configure and interact with your databases from a single console.

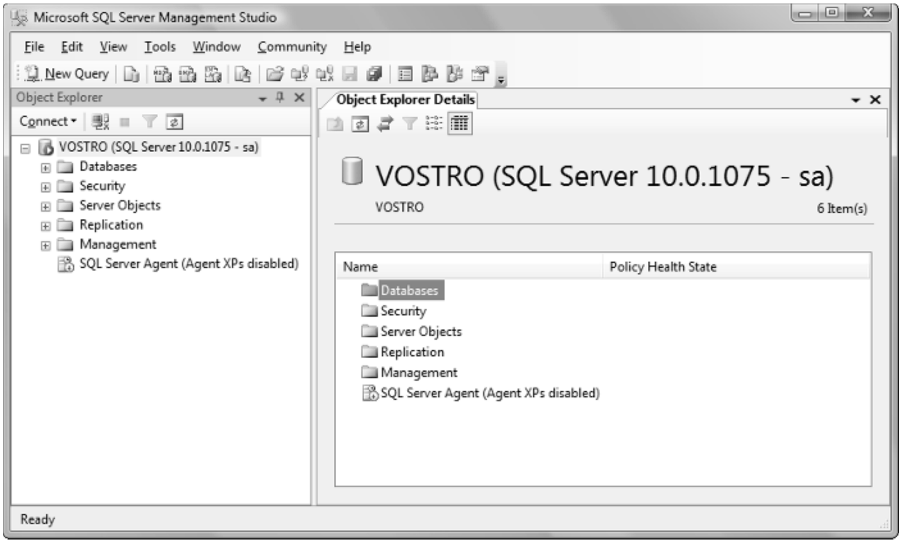
I describe the use SSMS throughout this book, both to directly issue Transact-SQL commands to SQL Server databases and to build databases using SSMS's graphic user interface.

I provide an overview of SSMS in Chapter 3.

SQL Server Books Online

My intention in this book is to provide you with a practical, hands-on introduction to SQL Server's functionality in an easy-to-read fashion. I don't intend it to be a "deep dive" into the technology and syntax of SQL Server. Rather, it should provide you with a working knowledge of this powerful database platform's functionality.

Microsoft includes detailed online documentation with SQL Server 2008 in the form of SQL Server Books Online. This documentation contains the latest information on SQL Server functionality for administrators and developers alike. It's a great place to turn when you're seeking specific information about command syntax or advanced SQL Server features.


Microsoft SQL Server Management Studio
File Edit View Tools Window Community Help
New Query
Object Explorer
Connect
Object Explorer Details
Ready

VOSTRO (SQL Server 10.0.1075 - sa)
- Databases
- Security
- Server Objects
- Replication
- Management
- SQL Server Agent (Agent XPs disabled)

VOSTRO (SQL Server 10.0.1075 - sa)

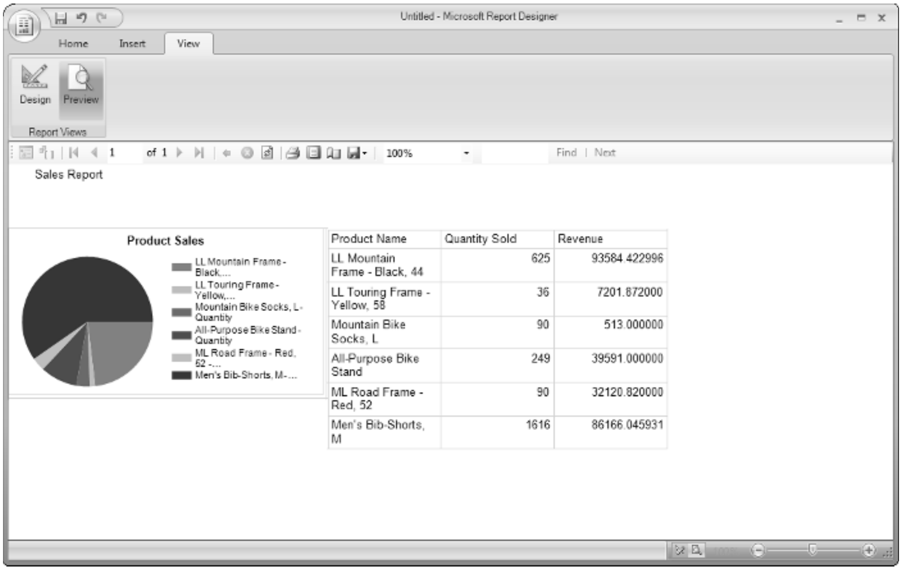
VOSTRO 6 Item(s)

Name	Policy Health State
Databases	
Security	
Server Objects	
Replication	
Management	
SQL Server Agent (Agent XPs disabled)	

Figure 1-2: SQL Server Management Studio.

Reporting Services

SQL Server Reporting Services underwent a significant overhaul before the release of SQL Server 2008. This platform allows you to design and publish dynamic reports based on SQL Server data. I show an example of a report created with SQL Server Reporting Services in Figure 1-3.


Untitled - Microsoft Report Designer
Home
Insert
View
Design
Preview
Report Views
1
of 1
100%
Find | Next
Product Sales
LL Mountain Frame - Black,...
LL Touring Frame - Yellow,...
Mountain Bike Socks, L - Quantity
All-Purpose Bike Stand - Quantity
ML Road Frame - Red, 52 -...
Men's Bib-Shorts, M-...

Sales Report

Product Name	Quantity Sold	Revenue
LL Mountain Frame - Black, 44	625	93584.422996
LL Touring Frame - Yellow, 58	36	7201.872000
Mountain Bike Socks, L	90	513.000000
All-Purpose Bike Stand	249	39591.000000
ML Road Frame - Red, 52	90	32120.820000
Men's Bib-Shorts, M	1616	86166.045931

Figure 1-3: Sample SQL Server Reporting Services report.

Analysis Services

SQL Server Analysis Services offers advanced analytical techniques, such as the use of online analytical processing (OLAP), data warehouses, and data mining. The use of this tool is beyond the scope of this book.

There are many other features of SQL Server — too many to list in this chapter. I discuss many of them later in this book. For example, I discuss SQL Profiler and the Database Engine Tuning Advisor in Chapter 14, and SQL Server Agent in Chapter 13.

Implementing Databases with SQL Server 2008

So how do you get started? SQL Server 2008 makes it simple to jump in feet first and begin working with databases.

Setting up your server

The first step is to create a SQL Server instance on an appropriate computing platform. Earlier in this chapter, I give you some advice for selecting the hardware, software, and SQL Server edition appropriate for your needs. In Chapters 2 and 3, I provide you with the information you need to set up a SQL Server instance.

Stocking it with databases

After you have SQL Server up and running, you need to create individual databases to house your data. In Chapter 4, I provide you with advice for planning and designing efficient databases. Chapters 5 and 6 describe the process for creating databases and tables and defining the relationships between different tables within the same database.

Accessing and updating your data

I dedicate a substantial portion of this book (Parts III and IV) to helping you put data in your database, update it, and retrieve it when necessary. My focus in this book is on the use of Transact-SQL and SQL Server Management Studio to manipulate your data.

Database developers use different techniques to manipulate databases. It still all boils down to Transact-SQL statements, but they use tools such as

Microsoft Visual Studio and the Microsoft Data Access Components (MDAC) to work with SQL Server 2008. Application development is beyond the scope of this book, but you can find more information in *Beginning Microsoft SQL Server 2008 Programming*.

Managing and protecting what you've built

Database administrators spend a large portion of their time keeping databases up and running daily. In Part V, I describe the tools and techniques you can use for ongoing administration of your SQL Server 2008 databases, including automation and troubleshooting tools. Part VI of this book discusses ways you can protect your data by applying SQL Server 2008's security and disaster recovery features.

What's New in SQL Server 2008

If you've used earlier versions of SQL Server, the first question in your mind is probably "What's new in SQL Server 2008?" The answer? Plenty! Microsoft promotes SQL Server 2008 as a major advance in its data platform vision and, as such, SQL Server 2008 offers a great deal of new functionality.

TIP

Rest easy, however, if you're already familiar with SQL Server 2005. Although SQL Server 2008 has a ton of new features, SSMS still has the same familiar look and feel. You should be able to get up and running quickly.

Declarative Management Framework

The Declarative Management Framework (DMF) is one of the most revolutionary features in SQL Server 2008. DMF allows database administrators to set high-level policies describing the allowed configuration status of DMF-managed SQL Server instances. DMF allows administrators to

- Create policies governing SQL Server configurations
- Evaluate an instance's current configuration against a policy and determine what deficiencies, if any, exist
- Apply a policy to a SQL Server instance
- Log or prevent any changes to a SQL Server instance that would bring it out of compliance with policy

I discuss the Declarative Management Framework in Chapter 20 of this book.

Encryption and Auditing

There are quite a few new security features in SQL Server 2008. Most notably:

- Transparent Data Encryption (TDE) allows the encryption of databases and backups with no user impact. I discuss TDE in Chapter 16.
- Enhanced auditing features allow the tracking of data access, in addition to data modification. I discuss SQL Server 2008's auditing features in Chapter 16.

Resource conservation

SQL Server 2008 includes two features designed to provide you with the ability to optimize server performance:

- Resource Governor allows you to set limits and priorities for different SQL Server workloads. This functionality offers you the ability to control the user experience by providing different users with a guaranteed level of performance. I discuss Resource Governor in Chapter 12.
- Backup compression shrinks the size of backup data before it is written to disk, reducing both the amount of time necessary to create a backup and the disk space used to store the backup. I discuss backup compression in Chapter 18.

Date/Time Data Types

I've been waiting for years for SQL Server to include date and time data types that match the way normal people think about dates and times! SQL Server 2008 provides four new data types that answer this formerly unmet demand:

- The `DATE` data type is a calendar date only, with no time information.
- The `TIME` data type is a time only, with no date information.
- The `DATETIMEOFFSET` data type is a date/time that allows for the inclusion of time zone information.
- The `DATETIME2` data type allows the specification of a date anywhere within the range of the year 1 A.D. to the year 9999 A.D.