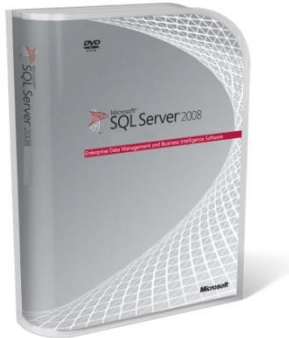




Microsoft® SQL Server® 2008

Programming SQL Server



Microsoft® SQL Server® provides database developers with a powerful set of new and improved features for efficiently creating next generation applications that manage any type of data.

<http://www.microsoft.com/sqlserver/2008/en/us/programmability.aspx>

TOP NEW FEATURES

- Precisely store and manage date and time information
- Store sparsely populated data efficiently using Sparse Columns
- Integrated Full-Text Indexes provide high-performance, scalability and manageability
- User Defined Types and User Defined Aggregates over 8KB
- Pass large amounts of data to functions or procedures using new Table-Valued Parameters
- Perform multiple operations with the new MERGE command
- Model hierarchical data using the new HierarchyID data type
- Build location-aware apps using new spatial data types, spatial methods and spatial indexes
- Manage files and documents efficiently using Filestream
- Identify dependencies across objects and databases using New Dependency Management
- Faster queries and reporting with Grouping Sets in GROUP BY
- Filtered Indexes provide faster data access for subsets of data
- Change Data Capture automates database change tracking

- Fully aligned and compatible with Windows Vista collation support

DEVELOP POWERFUL APPLICATIONS

Use SQL Server 2008's new and enhanced T-SQL programming features to create next generation database applications that support the most demanding end-user requirements.

New Table-Valued Parameters

Pass large amounts of data efficiently to functions and procedures using new Table-Valued Parameters.

T-SQL Enhancements

Perform multiple operations efficiently with the new MERGE command. Benefit from fully compatibility with Windows Server collation support, plus experience other programming improvements including the ability to:

- Insert multiple rows using a single INSERT statement
- Operate on values using rich assignment operators
- Initialize variables during declaration

SQLCLR Enhancements

Create powerful User Defined Types and User Defined Aggregates that are greater than 8KB in size. Create User Defined Aggregates that allow multiple input arguments.

Dependency Management

Streamline development by easily identifying dependencies across objects and databases using New Dependency Management.

CONTROL ANY TYPE OF DATA

Leverage SQL Server 2008's enhanced data type support to manage relational and non-relational data including precise date and time management, XML data, external documents and files, and new spatial information using either planar or geodetic representations.

New DATE and TIME Data Types

Store more precise date and time information with larger year range, user definable fractional precision, and time zone awareness using the following new ANSI SQL Standard compatible data types:

- DATE – new date-only type
- TIME(*precision*) - time-only type

- DATETIMEOFFSET(*precision*) - time zone aware DATETIME type
- DATETIME2(*precision*) - new type with larger fractional seconds and year range than DATETIME

New HIERARCHYID Date Type

Model hierarchical data such as org charts and files and folders using the new HIERARCHYID data type and easily perform operations on hierarchy data using powerful built-in methods for manipulating hierarchies efficiently.

New Support for Spatial Data

Build powerful location-aware applications using SQL Server's new spatial data types and built-in spatial functionality. Create responsive and highly engaging solutions that take advantage of the built-in spatial indexing capabilities.

Manage Files and Documents with new FILESTREAM Data Type

Manage files and documents efficiently while leveraging SQL Server security and transaction support using the powerful new FILESTREAM data type. Move files and documents to economical hardware and benefit from low cost file system storage that provides storage capacity limited only to volume size. Dual programming model enables the reuse of existing BLOB-centric T-SQL on new FILESTREAM BLOBs with the advantage of T-SQL transactional semantics, or take advantage of file system streaming access through Windows file management APIs that provide flexible operations on files and documents. Additional benefits include:

- Consistency between metadata and the data store
- Rich data streaming performance
- Reduction in database size

DELIVER OPTIMIZED SOLUTIONS

Enhancements in the SQL Server storage engine and data store

Filtered Indexes

Experience a new level of efficient data access and index storage

```

-- New date and time types
CREATE TABLE t1 (c1 DATE, c2 TIME(3),
c3 DATETIME2(7) NOT NULL DEFAULT GETDATE(),
c4 DATETIMEOFFSET CHECK (c4 < CAST(GETDATE() AS
DATETIMEOFFSET(0))));
INSERT INTO t1 VALUES ('0001-01-01', '23:59:59',
'0001-12-21 23:59:59.1234567', '9999-12-31 23:59:59.1234567 -07:00');
SELECT c4, DATEPART(TZOFFSET, c4), DATEPART(ISO_WEEK, c4),
DATEPART(MICROSECOND, c4) FROM t1;

-- Table Value Parameter
-- Create a user TABLE type
CREATE TYPE myTableType AS TABLE (id INT,
name NVARCHAR(100), qty INT);
-- Create a stored procedure that accepts a table-variable as a param
CREATE PROCEDURE myProc (@tvp myTableType READONLY)
AS
UPDATE Inventory SET qty += s.qty
FROM Inventory AS i INNER JOIN @tvp AS tvp ON i.id = tvp.id
GO

-- Multi-row insert through single insert stmt
INSERT INTO contacts VALUES ('John Doe', '425-333-5321'),
('Jane Doe', '206-123-4567'), ('John Smith', '650-434-7869');

```

enables developers to store, manage and find data efficiently using sparse columns, filtered indexes and a new fully integrated Full Text Search.

Sparse Columns

Store semi-structured and sparsely populated sets of data efficiently using Sparse Columns. Allows NULL data to consume no physical space in a database, thereby ensuring optimized data storage.

Fully Integrated Full-Text Search

Quickly find textual information across databases by leveraging SQL Server's new high-performance fully integrated Full-Text search technology. Manage how Full-Text Indexes are created and stored to build powerful and scalable applications.

using new Filtered Indexes that provide high-performance lookups of subsets of data.

Grouping Sets

Experience faster queries and reporting with Grouping Sets through powerful ANSI standard compliant extensions to the GROUP BY clause. Define multiple groupings in the same query to produce a high-speed result set that is equivalent to the results from a UNION ALL of differently grouped rows.

Change Data Capture

Automatically captures and maintains changes to data and schema across tables, which eliminates the need to develop custom change tracking logic. Built-in T-SQL methods enable developers to capture database changes efficiently.

For more information about programming Microsoft SQL Server 2008 visit <http://www.microsoft.com/sql/2008/technologies/programmability.aspx>. For the latest developer news and more information on Microsoft's broad range of resources for developers, including support programs, events, training, and the MSDN Library Online, visit MSDN Online at <http://msdn.microsoft.com/>.

This data sheet is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY.

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. (Use this only if fictitious content appears.)

0308 Part No. 098-00000 (color)
Part No. 098-00000 (black and white)

Microsoft