**What's New (Database Engine)**

**SQL Server 2014**

[Other Versions](javascript:;)

Description: http://i.msdn.microsoft.com/Areas/Epx/Content/Images/ImageSprite.png

* [SQL Server 2012](http://msdn.microsoft.com/en-in/library/bb510411(d=printer,v=sql.110).aspx)
* [SQL Server 2008 R2](http://msdn.microsoft.com/en-in/library/bb510411(d=printer,v=sql.105).aspx)
* [SQL Server 2008](http://msdn.microsoft.com/en-in/library/bb510411(d=printer,v=sql.100).aspx)

This latest release of the SQL Server Database Engine introduces new features and enhancements that increase the power and productivity of architects, developers, and administrators who design, develop, and maintain data storage systems. These are the areas in which the Database Engine has been enhanced.

[**In This Topic**](javascript:void(0))

This topic has the following sections.

* [Database Engine Feature Enhancements](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#Feature)
  + [Memory-Optimized Tables](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#MemoryOpt)
  + [SQL Server Data Files in Windows Azure](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#DataFiles)
  + [Host a SQL Server Database in a Windows Azure Virtual Machine](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#AzureVM)
  + [Backup and Restore Enhancements](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#Backup)
  + [New Design for Cardinality Estimation](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#CE)
  + [Delayed Durability](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#Durability)
  + [AlwaysOn Enhancements](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#AlwaysOn)
  + [Partition Switching and Indexing](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#OIR)
  + [Managing the Lock Priority of Online Operations](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#Lock)
  + [Columnstore Indexes](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#CCI)
  + [Buffer Pool Extension](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#Buffer)
  + [Incremental Statistics](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#Stats)
  + [Resource Governor Enhancements for Physical IO Control](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#RG)
  + [Online Index Operation Event Class](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#OnlineEvent)
  + [Database Compatibility Level](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#Compat)
* [Transact-SQL Enhancements](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#TSQL)
* [System View Enhancements](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#SystemTable)
* [Security Enhancements](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#Security)
* [Deployment Enhancements](http://msdn.microsoft.com/en-in/library/bb510411(d=printer).aspx#Deployment)

[**Database Engine Feature Enhancements**](javascript:void(0))

**Memory-Optimized Tables**

In-Memory OLTP is a memory-optimized database engine integrated into the SQL Server engine. In-Memory OLTP is optimized for OLTP. For more information, see [In-Memory OLTP (In-Memory Optimization)](http://msdn.microsoft.com/en-in/library/dn133186.aspx).

**SQL Server Data Files in Windows Azure**

[SQL Server Data Files in Windows Azure](http://msdn.microsoft.com/en-in/library/dn385720.aspx) enables native support for SQL Server database files stored as Windows Azure Blobs. This feature allows you to create a database in SQL Server running in on-premises or in a virtual machine in Windows Azure with a dedicated storage location for your data in Windows Azure Blob Storage.

**Host a SQL Server Database in a Windows Azure Virtual Machine**

Use the [Deploy a SQL Server Database to a Windows Azure Virtual Machine](http://msdn.microsoft.com/en-us/library/dn195938(v=sql.120).aspx) Wizard to host a database from an instance of SQL Server in a Windows Azure Virtual Machine.

**Backup and Restore Enhancements**

SQL Server 2014 contains the following enhancements for SQL Server Backup and Restore:

* SQL Server Backup to URL

SQL Server Backup to URL was introduced in SQL Server 2012 SP1 CU2 supported only by Transact-SQL, PowerShell and SMO. In SQL Server 2014 you can use SQL Server Management Studio to backup to or restore from Windows Azure Blob storage service. The new option is available both for the Backup task, and maintenance plans. For more information, see [Using Backup Task in SQL Server Management Studio](http://msdn.microsoft.com/en-in/library/dn435916.aspx#backuptaskssms), [SQL Server Backup to URL Using Maintenance Plan Wizard](http://msdn.microsoft.com/en-in/library/dn435916.aspx#maintenancewiz), and [Restoring from Windows Azure storage Using SQL Server Management Studio](http://msdn.microsoft.com/en-in/library/dn435916.aspx#restoressms).

* SQL Server Managed Backup to Windows Azure

Built on SQL Server Backup to URL, SQL Server Managed Backup to Windows Azure is a service that SQL Server provides to manage and schedule database and log backups. In this release only backup to Windows Azure storage is supported. SQL Server Managed Backup to Windows Azure can be configured both at the database and at instance level allowing for both granular control at the database level and automating at the instance level. SQL Server Managed Backup to Windows Azure can be configured on SQL Server instances running on-premises and SQL Server instances running on Windows Azure virtual machines. It is recommended for SQL Server instances running on Windows Azure virtual machines. For more information, see [SQL Server Managed Backup to Windows Azure](http://msdn.microsoft.com/en-in/library/dn449496.aspx).

* Encryption for Backups

You can now choose to encrypt the backup file during a backup operation. It supports several encryption algorithms including AES 128, AES 192, AES 256, and Triple DES. You must use either a certificate or an asymmetric key to perform encryption during backup. For more information, see [Backup Encryption](http://msdn.microsoft.com/en-in/library/dn449489.aspx).

**New Design for Cardinality Estimation**

The cardinality estimation logic, called the cardinality estimator, is re-designed in SQL Server 2014 to improve the quality of query plans, and therefore to improve query performance. The new cardinality estimator incorporates assumptions and algorithms that work well on modern OLTP and data warehousing workloads. It is based on in-depth cardinality estimation research on modern workloads, and our learnings over the past 15 years of improving the SQL Server cardinality estimator. Feedback from customers shows that while most queries will benefit from the change or remain unchanged, a small number might show regressions compared to the previous cardinality estimator. For performance tuning and testing recommendations, see [Cardinality Estimation (SQL Server)](http://msdn.microsoft.com/en-in/library/dn600374.aspx).

**Delayed Durability**

SQL Server 2014 introduces the ability to reduce latency by designating some or all transactions as delayed durable. A delayed durable transaction returns control to the client before the transaction log record is written to disk. Durability can be controlled at the database level, COMMIT level, or ATOMIC block level.

For more information see the topic [Control Transaction Durability](http://msdn.microsoft.com/en-in/library/dn449490.aspx).

**AlwaysOn Enhancements**

SQL Server 2014 contains the following enhancements for AlwaysOn Failover Cluster Instances and AlwaysOn Availability Groups:

* An Add Azure Replica Wizard simplifies creating hybrid solutions for AlwaysOn availability groups. For more information, see [Use the Add Azure Replica Wizard (SQL Server)](http://msdn.microsoft.com/en-in/library/dn463980.aspx).
* The maximum number of secondary replicas is increased from 4 to 8.
* When disconnected from the primary replica or during cluster quorum loss, readable secondary replicas now remain available for read workloads.
* Failover cluster instances (FCIs) can now use Cluster Shared Volumes (CSVs) as cluster shared disks. For more information, see [AlwaysOn Failover Cluster Instances (SQL Server)](http://msdn.microsoft.com/en-in/library/ms189134.aspx).
* A new system function, [sys.fn\_hadr\_is\_primary\_replica](http://msdn.microsoft.com/en-in/library/dn249345.aspx), and a new DMV, [sys.dm\_io\_cluster\_valid\_path\_names](http://msdn.microsoft.com/en-in/library/dn509548.aspx), is available.
* The following DMVs were enhanced and now return FCI information: [sys.dm\_hadr\_cluster](http://msdn.microsoft.com/en-in/library/hh212952.aspx), [sys.dm\_hadr\_cluster\_members](http://msdn.microsoft.com/en-in/library/hh231519.aspx), and [sys.dm\_hadr\_cluster\_networks](http://msdn.microsoft.com/en-in/library/hh213657.aspx).

**Partition Switching and Indexing**

The individual partitions of partitioned tables can now be rebuilt. For more information, see [ALTER INDEX (Transact-SQL)](http://msdn.microsoft.com/en-in/library/ms188388.aspx).

**Managing the Lock Priority of Online Operations**

The ONLINE = ON option now contains a WAIT\_AT\_LOW\_PRIORITY option which permits you to specify how long the rebuild process should wait for the necessary locks. The WAIT\_AT\_LOW\_PRIORITY option also allows you to configure the termination of blocking processes related to that rebuild statement. For more information, see [ALTER TABLE (Transact-SQL)](http://msdn.microsoft.com/en-in/library/ms190273.aspx) and [ALTER INDEX (Transact-SQL)](http://msdn.microsoft.com/en-in/library/ms188388.aspx). Troubleshooting information about new types of lock states is available in [sys.dm\_tran\_locks (Transact-SQL)](http://msdn.microsoft.com/en-in/library/ms190345.aspx) and [sys.dm\_os\_wait\_stats (Transact-SQL)](http://msdn.microsoft.com/en-in/library/ms179984.aspx).

**Columnstore Indexes**

These new features are available for columnstore indexes:

* **Clustered columnstore indexes**

Use a clustered columnstore index to improve data compression and query performance for data warehousing workloads that primarily perform bulk loads and read-only queries. Since the clustered columnstore index is updateable, the workload can perform many insert, update, and delete operations. For more information, see [Columnstore Indexes Described](http://msdn.microsoft.com/en-in/library/gg492088.aspx) and [Using Clustered Columnstore Indexes](http://msdn.microsoft.com/en-in/library/dn589807.aspx).

* **SHOWPLAN**

SHOWPLAN displays information about columnstore indexes. The **EstimatedExecutionMode** and **ActualExecutionMode** properties have two possible values: **Batch** or **Row**. The **Storage** property has two possible values: **RowStore** and **ColumnStore**.

* **Archival data compression**

ALTER INDEX … REBUILD has a new COLUMNSTORE\_ARCHIVE data compression option that further compresses the specified partitions of a columnstore index. Use this for archival, or for other situations that require a smaller data storage size and can afford more time for storage and retrieval. For more information, see [ALTER INDEX (Transact-SQL)](http://msdn.microsoft.com/en-in/library/ms188388.aspx).

**Buffer Pool Extension**

The [Buffer Pool Extension](http://msdn.microsoft.com/en-in/library/dn133176.aspx) provides the seamless integration of solid-state drives (SSD) as a nonvolatile random access memory (NvRAM) extension to the Database Engine buffer pool to significantly improve I/O throughput.

**Incremental Statistics**

CREATE STATISTICS and related statistic statements now permits per partition statistics to be created by using the INCREMENTAL option. Related statements allow or report incremental statistics. Affected syntax includes UPDATE STATISTICS, sp\_createstats, CREATE INDEX, ALTER INDEX, ALTER DATABASE SET options, DATABASEPROPERTYEX, sys.databases, and sys.stats. For more information, see [CREATE STATISTICS (Transact-SQL)](http://msdn.microsoft.com/en-in/library/ms188038.aspx).

**Resource Governor Enhancements for Physical IO Control**

The Resource Governor enables you to specify limits on the amount of CPU, physical IO, and memory that incoming application requests can use within a resource pool. In SQL Server 2014, you can use the new MIN\_IOPS\_PER\_VOLUME and MAX\_IOPS\_PER\_VOLUME settings to control the physical IOs issued for user threads for a given resource pool. For more information, see [Resource Governor Resource Pool](http://msdn.microsoft.com/en-in/library/hh510189.aspx) and [CREATE RESOURCE POOL (Transact-SQL)](http://msdn.microsoft.com/en-in/library/bb895329.aspx).

The MAX\_OUTSTANDING\_IO\_PER\_VOLUME setting of the ALTER RESOURCE GOVENOR sets the maximum outstanding I/O operations per disk volume. You can use this setting to tune IO resource governance to the IO characteristics of a disk volume and can be used to limit the number of IOs issued at the SQL Server instance boundary. For more information, see [ALTER RESOURCE GOVERNOR (Transact-SQL)](http://msdn.microsoft.com/en-in/library/bb934013.aspx).

**Online Index Operation Event Class**

The progress report for the online index operation event class now has two new data columns: **PartitionId** and **PartitionNumber**. For more information, see [Progress Report: Online Index Operation Event Class](http://msdn.microsoft.com/en-in/library/ms186279.aspx).

**Database Compatibility Level**

The 90 compatibility level is not valid in SQL Server 2014. For more information, see [ALTER DATABASE Compatibility Level (Transact-SQL)](http://msdn.microsoft.com/en-in/library/bb510680.aspx)

[**Transact-SQL Enhancements**](javascript:void(0))

**Inline specification of CLUSTERED and NONCLUSTERED**

Inline specification of CLUSTERED and NONCLUSTERED indexes is now allowed for disk-based tables. Creating a table with inline indexes is equivalent to issuing a create table followed by corresponding CREATE INDEX statements. Included columns and filter conditions are not supported with inline indexes.

**SELECT … INTO**

The SELECT … INTO statement is improved and can now operate in parallel. The database compatibility level must be at least 110.

**Transact-SQL Enhancements for In-Memory OLTP**

For information about the Transact-SQL changes to support In-Memory OLTP, see [Transact-SQL Support for In-Memory OLTP](http://msdn.microsoft.com/en-in/library/dn133180.aspx).

[**System View Enhancements**](javascript:void(0))

**sys.xml\_indexes**

[sys.xml\_indexes (Transact-SQL)](http://msdn.microsoft.com/en-in/library/ms176003.aspx) has 3 new columns: xml\_index\_type, xml\_index\_type\_description, and path\_id.

**sys.dm\_exec\_query\_profiles**

[sys.dm\_exec\_query\_profiles (Transact-SQL)](http://msdn.microsoft.com/en-in/library/dn223301.aspx) monitors real time query progress while a query is in execution.

**sys.column\_store\_row\_groups**

[sys.column\_store\_row\_groups (Transact-SQL)](http://msdn.microsoft.com/en-in/library/dn223749.aspx) provides clustered columnstore index information on a per-segment basis to help the administrator make system management decisions.

**sys.databases**

[sys.databases (Transact-SQL)](http://msdn.microsoft.com/en-in/library/ms178534.aspx) has 3 new columns: is\_auto\_create\_stats\_incremental\_on, is\_query\_store\_on, and resource\_pool\_id.

**System View Enhancements for In-Memory OLTP**

For information about system view enhancements to support In-Memory OLTP, see [New and Updated Properties, System Views, Stored Procedures, Wait Types, and DMVs for In-Memory OLTP](http://msdn.microsoft.com/en-in/library/dn133080.aspx).

[**Security Enhancements**](javascript:void(0))

**CONNECT ANY DATABASE Permission**

A new server level permission. Grant **CONNECT ANY DATABASE** to a login that must connect to all databases that currently exist and to any new databases that might be created in future. Does not grant any permission in any database beyond connect. Combine with **SELECT ALL USER SECURABLES** or VIEW SERVER STATE to allow an auditing process to view all data or all database states on the instance of SQL Server.

**IMPERSONATE ANY LOGIN Permission**

A new server level permission. When granted, allows a middle-tier process to impersonate the account of clients connecting to it, as it connects to databases. When denied, a high privileged login can be blocked from impersonating other logins. For example, a login with **CONTROL SERVER** permission can be blocked from impersonating other logins.

**SELECT ALL USER SECURABLES Permission**

A new server level permission. When granted, a login such as an auditor can view data in all databases that the user can connect to.

[**Deployment Enhancements**](javascript:void(0))

[Deploy a SQL Server Database to a Windows Azure Virtual Machine](http://msdn.microsoft.com/en-in/library/dn195938.aspx) enables deployment of a SQL Server database to a Windows Azure VM.