Server roles: <https://www.mssqltips.com/sqlservertip/1887/understanding-sql-server-fixed-server-roles/>

* Sysadmin 🡪 super permission
* bulkadmin 🡪 Bulk Insert permission + Insert on object
* dbcreator 🡪 New database creation
* diskadmin 🡪 backup device
* processadmin 🡪 Killing connection to SQL server
* securityadmin 🡪 manage security, Jr. DBA
* serveradmin 🡪 server configuration + Shut down SQL
* setupadmin 🡪 set up linked servers

Database roles:

<https://www.mssqltips.com/sqlservertip/1900/understanding-sql-server-fixed-database-roles/>

* db\_owner 🡪 super power
* db\_securityadmin 🡪 manage security
* db\_accessadmin
* db\_backupoperator 🡪 Backup
* db\_ddladmin 🡪 Alter, DROP and CREATE
* db\_datareader 🡪 READ ONLY
* db\_datawriter 🡪 INSERT, UPDATE AND DELETE
* db\_denydatareader 🡪 DENYREAD
* db\_denydatawriter 🡪 DENYWRITE

MSDB database role: <https://www.mssqltips.com/sqlservertip/1041/sql-server-agent-fixed-database-roles/>

|  |  |  |
| --- | --- | --- |
| **ID** | **Role** | **Description** |
| 1 | SQLAgentUserRole | * Ability to manage Jobs that they own |
| 2 | SQLAgentReaderRole | * All of the SQLAgentUserRole rights * The ability to review multiserver jobs, their configurations and history |
| 3 | SQLAgentOperatorRole | * All of the SQLAgentReaderRole rights * The ability to review operators, proxies and alerts * Execute, stop or start all local jobs * Delete the job history for any local job * Enable or disable all local jobs and schedules |

Grant Permissions to Integration Services Service:

### **To grant access to the Integration Services service**

1. Run Dcomcnfg.exe. Dcomcnfg.exe provides a user interface for modifying certain settings in the registry.
2. In the **Component Services** dialog, expand the Component Services > Computers > My Computer > DCOM Config node.
3. Right-click **Microsoft SQL Server Integration Services 11.0**, and then click **Properties**.
4. On the **Security** tab, click **Edit** in the **Launch and Activation Permissions** area.
5. Add users and assign appropriate permissions, and then click Ok.
6. Repeat steps 4 - 5 for Access Permissions.
7. Restart SQL Server Management Studio.
8. Restart the Integration Services Service.

GRANT Permission to SSISDB:

The catalog has **three** securable objects: **projects, environments and packages**

<https://www.mssqltips.com/sqlservertip/3153/managing-ssis-security-with-database-roles/>

SSIS\_ADMIN is most powerful.

Difference between rebuild and reorganize: <http://saurabhsinhainblogs.blogspot.com/2013/05/difference-between-reindexing-and.html>

# sys.dm\_db\_index\_physical\_stats

SELECT \* FROM sys.dm\_db\_index\_physical\_stats(2, null, NULL, NULL , 'LIMITED');

<https://docs.microsoft.com/en-us/sql/relational-databases/databases/database-states>

## **Database State Definitions**

The following table defines the database states.

| State | Definition |
| --- | --- |
| ONLINE | Database is available for access. The primary file group is online, **although the undo phase of recovery may not have been completed**. |
| OFFLINE | Database is unavailable. A database becomes offline by explicit user action and remains offline until additional user action is taken. For example, the database may be taken offline in order **to move a file to a new disk**. The database is then brought back online after the move has been completed. |
| RESTORING | One or more files of the primary filegroup are **being restored**, or one or more secondary files are being restored offline. The database is unavailable. |
| RECOVERING | **Database is being recovered**. The recovering process is a transient state; the database will automatically become online if the recovery succeeds. If the recovery fails, the database will become suspect. The database is unavailable. |
| RECOVERY PENDING | SQL Server has encountered a resource-related error during recovery. **The database is not damaged, but files may be missing or system resource limitations may be preventing it from starting**. The database is unavailable. Additional action by the user is required to resolve the error and let the recovery process be completed. |
| SUSPECT | **At least the primary filegroup is suspect and may be damaged**. The database cannot be recovered during startup of SQL Server. The database is unavailable. Additional action by the user is required to resolve the problem. |
| EMERGENCY | User has changed the database and set the status to EMERGENCY. The database is **in single-user mode and may be repaired or restored.** The database is marked READ\_ONLY, logging is disabled, and access is limited to members of the **sysadmin** fixed server role. EMERGENCY is primarily used for troubleshooting purposes. For example, a database marked as suspect can be set to the EMERGENCY state. This could permit the system administrator read-only access to the database. Only members of the **sysadmin** fixed server role can set a database to the EMERGENCY state. |

<https://www.stellarinfo.com/support/kb/index.php/article/procedure-to-recover-sql-database-from-suspect-mode>

# Trace Flags 1117, 1118, and Tempdb Configuration

<https://www.brentozar.com/archive/2014/06/trace-flags-1117-1118-tempdb-configuration/>

### **Trace Flag 1118 – Full Extents Only**

KB 2154845 advises that Trace Flag 1118 can help in some situations. That trace flag tells SQL Server that it should avoid “mixed extents” and use “full extents”. ([Learn more about extents here](http://technet.microsoft.com/en-us/library/ms190969(v=sql.105).aspx).)

This means that each newly allocated object in every database on the instance gets its own private 64KB of data. Tempdb is usually the place where most objects are created, so it makes the most difference there.

Since this trace flag is advocated in KB 2154845, it’s clearly documented as safe to use. But for one reason or another it hasn’t made its way into the [list of Trace Flags in SQL Server Books Online](http://technet.microsoft.com/en-us/library/ms188396.aspx).

### **Trace Flag 1117 – Grow All Files in a FileGroup Equally**

Trace flag 1117 changes the behavior of file growth: if one data file in a filegroup grows, it forces other files in that filegroup to ALSO grow. This can be useful for tempdb, which is commonly configured with multiple data files as [KB 2154845](http://support.microsoft.com/kb/2154845/en-us) advises.

Not everyone likes to implement this trace flag, particularly because it impacts every database on the instance and not just tempdb. Personally, I prefer to pre-grow tempdb files so they fill the tempdb drive, just leaving room for any “free space monitoring” you have. Whenever I can avoid using a trace flag, I do it: just less room to hit an edge condition.

But Trace Flag 1117 is something that some folks like, and it’s recommended in the [Fast Track Architecture Guide from Microsoft](http://download.microsoft.com/download/D/B/D/DBDE7972-1EB9-470A-BA18-58849DB3EB3B/FTRARefConfigGuide.docx).

Trace Flag 1117 is also not in the [list of Trace Flags in SQL Server Books Online](http://technet.microsoft.com/en-us/library/ms188396.aspx).

**Deadlock Trace flag**: 1222

You can enable the trace flag when SQL Server starts by using the -T1222 startup option, or you can use the DBCC TRACEON (1222,-1) command after SQL Server has started.

DBCC TRACESTATUS (1222, -1)

<https://blogs.msdn.microsoft.com/sqlsakthi/2011/02/08/different-status-of-a-spid-in-sql-server-and-what-do-they-mean/>

**Wait types:**

DMV: sys.dm\_os\_waiting\_tasks 🡪 to get to know wait types.

Sys.dm\_exec\_query\_memory\_grants 🡪 amount of memory received by each query spid or not. How much it asked for and how much it has granted.

PageIOLatch 🡪 some bottleneck at physical IO, reading from disk.

CXPACKET 🡪 parallelism

Pagelatch 🡪 Buffer latch contention

\_UP

\_EX

\_SH

IOCompletion

Writelog

LCK\_M\_S 🡪 Updates block select due to shared locks.

Resource semaphoure🡪 waiting for memory grant

**SQLOS** 🡪 Running tasks

**Runnable queue** 🡪 got access on resource but waiting for CPU

**Waiter list (resource waits)🡪** waiting due to resource like many wait types.

Running 🡪 its consuming CPU cycle currently.

Suspended 🡪 Waiting for resource like IO, N/w etc.

Runnable 🡪 got resource but waiting for CPU quantum.

Pending 🡪 No worker thread available.

Backgroud 🡪 Like resource monitor and deadlock monitor

Sleeping 🡪 No work to be done.

New features of SQL 2014:

<https://www.itprotoday.com/microsoft-sql-server/top-ten-new-features-sql-server-2014>

New features of SQL 2016:

<https://www.quickstart.com/blog/10-new-features-of-sql-server-2016/>

in memory oltp engine

buffer pool extention SSD disks

always on : add azure replica, 8, avaiable

encrypted backup and too to Azure

storage IO control : Resource governer

Powerful BI: SSDT-BI

Always encrypted

Query store: track previous plan

Row level security

Json support

Enhancement on AG: 3 sync replica

Stretch database

CXPacket: parralism happening.

<https://support.microsoft.com/en-us/help/2806535/recommendations-and-guidelines-for-the-max-degree-of-parallelism-confi>

PAGEIOLATCH: waiting for IO completion, waiting for reading 8 KB pages from disks into buffer

SOS\_scheduler\_yield: 4 ms quantum completed. went back to CPU, didnt wait for resource.

Resource Semaphore, waiting for memory grant. sys.dm\_os\_memory\_Grants

**SPLIT TEMPDB**

<https://www.sqlservercentral.com/forums/topic/dividing-tempdb-datafiles-into-multiple-files>