**Some Suggested SQL Server 2008 R2 Instance Configuration Settings**

Posted on [February 24, 2011](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/) by [Glenn Berry](http://sqlserverperformance.wordpress.com/author/guderian1961/)

Depending on your anticipated workload and which SQL Server components you have installed, there are a number of instance level settings that you should consider changing from their default values. These include Max Server Memory, optimize for ad-hoc workloads,  default backup compression, max degree of parallelism, and the number of TempDB data files for the SQL Server instance.

Max Server Memory

Max Server Memory controls how much memory can be used by the SQL Server Buffer Pool, which is where cached data is held in memory after it is read in off of the disk subsystem. This is only for use by the relational Database Engine. It has no effect on other SQL Server components, such as SSIS, SSRS, or SSAS. It also does not affect other Database Engine components such as Full-Text Search, the CLR, the procedure cache, connection memory, etc.

The idea here is to limit how much memory the buffer pool can use in order to avoid starving other SQL Server components or the operating system for memory. The table in Listing 1 has some suggested starting values for a dedicated database server that is only running the Database Engine, with no other SQL Server components running. If you have other components installed, you should adjust the Max Server Memory setting downward. The setting is dynamic, meaning that you can change it and it will take effect immediately, with no restart of SQL Server required.

Optimize for ad-hoc workloads

Optimize for ad-hoc workloads is a new instance level setting that was added in SQL Server 2008 which is designed to help control the amount of memory that is used by single-use, ad-hoc query plans in the procedure cache. It allows SQL Server to only store a small stub of an ad-hoc query plan in the procedure cache the first time the ad-hoc plan is executed, which reduces the memory required by the plan in the procedure cache.

With SQL Server 2005, it was very common to see very large amounts of memory being used by single-use, ad-hoc query plans (often in the 6 to 8 GB range). Later builds of SQL Server have changes that reduced this problem somewhat, but it was still a big issue. Interestingly, one of the biggest offenders that generated ad-hoc query plans in SQL Server 2005 was SQL Server Agent! Another big offender was SharePoint.

In my opinion, you should always enable this setting on SQL Server 2008 and above. I really cannot think of a good reason not to do this.

Default backup compression

If you have a Version and Edition of SQL Server that supports native backup compression, I believe you should enable this setting as shown in Listing 7.1. This simply means that all SQL Server backups will use native backup compression by default. You can always over-ride this setting in your backup T-SQL command. SQL Server 2008 Enterprise Edition supports native backup compression, as does SQL Server 2008 R2 Standard Edition.

Max degree of parallelism

Max degree of parallelism is an instance level configuration setting that is meant to control whether and how much the Query Optimizer will attempt to spread a complex or expensive query across multiple processor cores to run in parallel. The default setting is zero, which allows SQL Server to parallelize queries across as many cores as it sees fit. This default setting is usually the best choice for DW and reporting workloads, but can often be problematic for OLTP workloads. With OLTP workloads, sometimes the Query Optimizer will choose a parallel query plan when a non-parallel query plan would have actually have been less expensive. If this happens a lot, it can cause a high percentage of CXPACKET waits for the instance.

In the past, many people would advise you to immediately change your max degree of parallelism setting to a value of 1 if you had an OLTP workload and you saw a high percentage of CXPACKET waits in your cumulative wait statistics for the instance. I think you should look a little deeper to try to find the actual source of the problem before you make that change. In my experience, it is quite common for “missing indexes” for important queries to cause the Query Optimizer to choose a parallel plan to try to compensate for the missing index. If SQL Server has to do a very large index or table scan, the Query Optimizer may think that it will be less expensive to parallelize the query.

If that is the case, it would probably be better to create the missing index to alleviate the source of the problem, instead of setting max degree of parallelism to 1 to treat the symptom. It is certainly worth some investigation. One big exception to this advice is if you are working with SharePoint databases (for either SharePoint 2007 or 2010). In that case, you should definitely set max degree of parallelism to 1 for the instance.

Number of TempDB data files

By default, SQL Server will create one small TempDB data file (and one log file) in the default location for TempDB on your SQL Server instance. If you ignored my advice about default file locations, the TempDB files will be in a sub-directory on the same drive where your SQL Server binary files are located. This is likely your C: drive on your database server, which is not a good location! If your TempDB is on your C: drive, you need to move it someplace better, which is hopefully a fast, dedicated logical drive.

You also need to create some additional TempDB data files, which should all be the same size. If all of the TempDB data files are the same size, SQL Server will use all of them equally. The reason to have multiple data files is to reduce possible allocation contention, as objects are created and destroyed in TempDB.

The old guidance from Microsoft was to create one TempDB data file per physical processor core. Now in the days of eight and twelve core processors, I think this is excessive. A general consensus seems to be forming that you should start out with perhaps four or eight TempDB data files (making them all the same size), and then look for signs of allocation contention before you consider creating any more TempDB data files.

Listing 1 shows some queries that let you check your current instance configuration settings, and some examples of how you might change them. These are example settings only; you need to use your own judgment for what values you choose.

-- Get configuration values for instance

SELECT name, value, value\_in\_use, [description]

FROM sys.configurations

ORDER BY name;

-- Set max server memory = 59500MB for the server

-- (example value only)

EXEC sp\_configure 'max server memory (MB)', 59500;

GO

RECONFIGURE;

GO

-- Some suggested Max Server Memory settings

-- Physical RAM Max Server Memory Setting

-- 4GB 3200

-- 6GB 4800

-- 8GB 6600

-- 16GB 14000

-- 24GB 21600

-- 32GB 29000

-- 48GB 44500

-- 64GB 59500

-- 72GB 67000

-- 96GB 90000

-- Enable optimize for ad-hoc workloads

-- (new in SQL Server 2008)

EXEC sp\_configure 'optimize for ad hoc workloads', 1;

GO

RECONFIGURE;

GO

-- Enable backup compression by default

-- (new in SQL Server 2008 Enterprise Edition)

-- (added to SQL Server 2008 R2 Standard Edition

EXEC sp\_configure 'backup compression default', 1;

GO

RECONFIGURE;

GO

-- Set MAXDOP = 1 for the server

-- Depends on workload and wait stats

EXEC sp\_configure 'max degree of parallelism', 1;

GO

RECONFIGURE;

GO

Listing 1: SQL Server Instance Configuration Setting Queries

[Windows Hyper Server 2008](s-p:%20Go%20to%20www.dell.com)  
Windows Server 2008 R2: Learn About Benefits & Solutions From Dell.com  
[www.dell.com](s-p:%20Go%20to%20www.dell.com)

[2N+2 Data Center](s-p:%20Go%20to%20www.databank.com)  
DataBank Colocation Managed Secure Data Center  
[www.databank.com](s-p:%20Go%20to%20www.databank.com)

[BCP for MSPs](s-p:%20Go%20to%20www.ZenithInfotech.com/)  
On/Offsite Backup w/ Standby Server NOC Mgmgt & Monitoring Included  
[www.ZenithInfotech.com/](s-p:%20Go%20to%20www.ZenithInfotech.com/)

[Software Developer](s-p:%20Go%20to%20dice.com/Netsoft_USA,_Inc.)  
New York, NY Work here -- love your job.  
[dice.com/Netsoft\_USA,\_Inc.](s-p:%20Go%20to%20dice.com/Netsoft_USA,_Inc.)

This entry was posted in [SQL Server 2008](http://en.wordpress.com/tag/sql-server-2008/), [SQL Server 2008 R2](http://en.wordpress.com/tag/sql-server-2008-r2/) and tagged [Configuration Settings](http://en.wordpress.com/tag/configuration-settings/). Bookmark the [permalink](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/).

[← New SQL Server 2005 Cumulative Updates](http://sqlserverperformance.wordpress.com/2011/02/22/new-sql-server-2005-cumulative-updates/)

[Like](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/?like=1&_wpnonce=ee32f08ade)

One blogger likes this post.

[hopefoley](http://gravatar.com/hopefoley)

**6 Responses to *Some Suggested SQL Server 2008 R2 Instance Configuration Settings***

1. http://0.gravatar.com/avatar/8944d06f663d43a175159814e0a71aef?s=40&d=identicon&r=G[*SQLRockstar*](http://thomaslarock.com) says:

[February 24, 2011 at 9:54 PM](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/#comment-4639)

Glenn,

If I have 32GB of RAM, and I allocate 29 to SQL Server, leaving 3 for the O/S, then shouldn’t I expect to see an O/S Memory consumption of close to 100% most of the time? Also, do those memory settings differ for x32 versus x64 versions, or from win2k3 or win2k8?

Thanks for the post.

[Reply](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/?replytocom=4639#respond)

* + http://0.gravatar.com/avatar/8512f3d2e5cd85aebc94c04afd2f0469?s=40&d=identicon&r=G[*Glenn Berry*](http://sqlserverperformance.wordpress.com) says:

[February 24, 2011 at 11:03 PM](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/#comment-4641)

Thomas,

Well, it depends on your workload. These are just suggested starting settings for a dedicated server that is only running the database engine. The Max Server Memory setting is dynamic, so it is pretty easy to change. The idea is to have roughly 2-4GB of free memory after the system is under a load for a while. I would use a lower setting for Windows Server 2003 vs. Windows Server 2008 and above. I would also use a lower setting for SQL Server 2005 vs. SQL Server 2008 and above. The newer versions have better memory management, and the OS and SQL Server communicate with each other better.

I would use the same settings for x86 and x64. AWE memory in x86 can only be used for the buffer pool, which is what we are trying to control here.

[Reply](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/?replytocom=4641#respond)

1. http://0.gravatar.com/avatar/2309325184e047170e3504a721263714?s=40&d=identicon&r=G*Jeff York* says:

[February 25, 2011 at 7:53 AM](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/#comment-4645)

Glenn-

Very nice. Good information presented clearly and succinctly. Thanks!

[Reply](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/?replytocom=4645#respond)

1. http://0.gravatar.com/avatar/ead2a33bde7c94164536d719715e67a9?s=40&d=identicon&r=G[*Edward Williams*](https://thesiteman.blogspot.com) says:

[February 25, 2011 at 9:17 AM](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/#comment-4648)

Glenn, thanks for sharing this info – very useful.

Do you think the suggested settings will change when dealing with sharepoint? I realize database server is a database server – regardless of what’s using it but still…

[Reply](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/?replytocom=4648#respond)

* + http://0.gravatar.com/avatar/8512f3d2e5cd85aebc94c04afd2f0469?s=40&d=identicon&r=G[*Glenn Berry*](http://sqlserverperformance.wordpress.com) says:

[February 25, 2011 at 9:26 AM](http://sqlserverperformance.wordpress.com/2011/02/24/some-suggested-sql-server-2008-r2-instance-configuration-settings/#comment-4649)

Some settings will change when dealing with SharePoint. A good example is max degree of parallelism, which (according to Microsoft and my own experience) should be set to 1, when you have SharePoint databases on your instance.