# SQL Server Performance Tuning

<https://www.faceofit.com/sql-server-performance-tuning/>

### **SQL Server Performance Tuning**

* SQL Server Performance tuning is a big topic and sometimes it becomes difficult to decipher the problem.
* So, in this blog topic, I am going to talk about generic steps to ensure, your SQL Server is running optimally.
* There are various 3rd Party [monitoring tools](https://www.faceofit.com/tools-to-monitor-sql-server-performance/) like SQL Sentry, Redgate etc, but today we are only going to discuss about the [tools and resources](https://www.faceofit.com/tools-to-monitor-sql-server-performance/) available out of the box for the majority of the SQL Server installations.

### **Disks:**

Disks are the slowest part of the entire subsystem. In order for SQL Server to perform optimally, monitoring and optimizing the SQL Server Disk sub-system is very important. The Microsoft SQL Server Product team have very specific numbers recommended by them for Optimal Disk Performance.

**Avg. Disk Sec/Read** is the average time, in seconds, of a read of data from the disk. The following list shows ranges of possible values and what the ranges mean:

***Less than 10 ms – very good  
Between 10 – 20 ms – okay  
Between 20 – 50 ms – slow, needs attention  
Greater than 50 ms – Serious I/O bottleneck***

### **Performance Monitoring & Analysis:**

* You need to use \*Performance Monitor\* (Start>Run>Perfmon) and use the Data collector to capture the below mentioned counters. [[Reference](https://na01.safelinks.protection.outlook.com/?url=http%3a%2f%2ftechnet.microsoft.com%2fen-us%2flibrary%2fcc749337.aspx&data=01%7c01%7cinghosh%40064d.mgd.microsoft.com%7c62a2f9c8a4bf4928a9a608d2ec51aa84%7c72f988bf86f141af91ab2d7cd011db47%7c1&sdata=RHob2Yz8dUxX9nr6j6tNgdTNYZd%2bY%2f6j989W8PkwINg%3d)]
* You can use a time interval of 15 seconds or 30 seconds.
* [Create a Data Collector Set from Performance Monitor](https://na01.safelinks.protection.outlook.com/?url=http%3a%2f%2ftechnet.microsoft.com%2fen-us%2flibrary%2fcc722148.aspx&data=01%7c01%7cinghosh%40064d.mgd.microsoft.com%7c62a2f9c8a4bf4928a9a608d2ec51aa84%7c72f988bf86f141af91ab2d7cd011db47%7c1&sdata=4oqNb5uyXnyyEaQ1qGh9XzN83rRJAWsDXfNg24c1YMw%3d)
* [Schedule Data Collection in Windows Performance Monitor](https://na01.safelinks.protection.outlook.com/?url=http%3a%2f%2ftechnet.microsoft.com%2fen-us%2flibrary%2fcc722312.aspx&data=01%7c01%7cinghosh%40064d.mgd.microsoft.com%7c62a2f9c8a4bf4928a9a608d2ec51aa84%7c72f988bf86f141af91ab2d7cd011db47%7c1&sdata=g4%2bt7PP1A%2ft9jHNyUTchHSInShBPgTcDvht5z1ZwuWg%3d) (per day separate .blg file)
* Analyze through **PAL tool at code plex** Link: [https://pal.codeplex.com/](https://na01.safelinks.protection.outlook.com/?url=https%3a%2f%2fpal.codeplex.com%2f&data=01%7c01%7cinghosh%40064d.mgd.microsoft.com%7c62a2f9c8a4bf4928a9a608d2ec51aa84%7c72f988bf86f141af91ab2d7cd011db47%7c1&sdata=ZjzAEpv0cjDEnS59timd4igbKsToB30P5hI2T0Pnxks%3d)

**Here are the performance counters –**

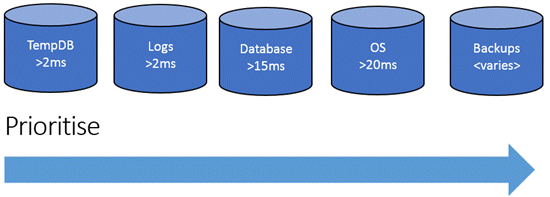
* Processor: % Processor Time  Should average below 75% (and preferably below 50%).
* System: Processor Queue Length             Should average below 2 per processor. For example, in a 2-processor machine, it should remain below 4.
* Memory—Pages/sec     Should average below 20 (and preferably below 15).
* Memory—Available Bytes           Should remain above 50 MB.
* Memory – Free System Page Table Entries
* Memory – Pages Input/Sec
* Physical Disk—% Disk Time          Should average below 50%.
* Physical Disk—Avg. Disk Queue Length Should average below 2 per disk. For example, for an array of 5 disks, this figure should average below 10.
* Physical Disk—Avg. Disk Reads/sec          Used to size the disk and CPU. Should be below 85% of the capacity of the drive.
* Physical Disk—Avg. Disk Writes/ sec        Used to size the disk and CPU. Should be below 85% of the capacity of the drive.
* Network Interface—Bytes Total/sec       Used to size the network bandwidth.
* SQL Server: Buffer Manager— Buffer Cache Hit Ratio     Should exceed 90% (and ideally approach 99%).
* SQL Server: Buffer Manager—Page Life Expectancy        Used to size memory. Should remain above 300 seconds.
* SQL Server: Buffer Manager Lazy Writes/Sec
* SQL Server: Buffer Manager Checkpoint Pages/Sec
* SQL Server: Buffer Manager Page writes/sec
* SQL Server: General Statistics— User Connections           Used to size memory.
* SQLServer:General Statistics – Logins/sec
* SQLServer:General Statistics – Logouts/sec
* SQL Server: Databases— Transactions/sec           Used to size disks and CPU.
* SQL Server: Databases—Data File(s) Size KB        Used to size the disk subsystem.
* SQL Server: Databases—Percent Log       Used to size the disk subsystem.
* SQLServer:SQL Statistics Batch Requests/Sec
* SQL Server:Latches Average Latch Wait Time (ms)
* SQL Server:Locks Number of Deadlocks/sec
* SQL Server:Locks Lock Requests/sec
* SQL Server:Locks Average Wait Time (ms)
* Paging File %Usage

**Disk Drive Placements:**

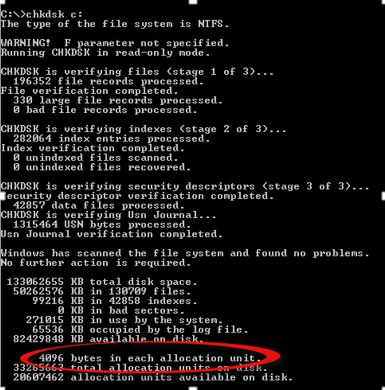
***Use separate drives for different purposes. (Log File, Data Files, Backup’s & TempDB)***

### **Individual disk latency requirements:**

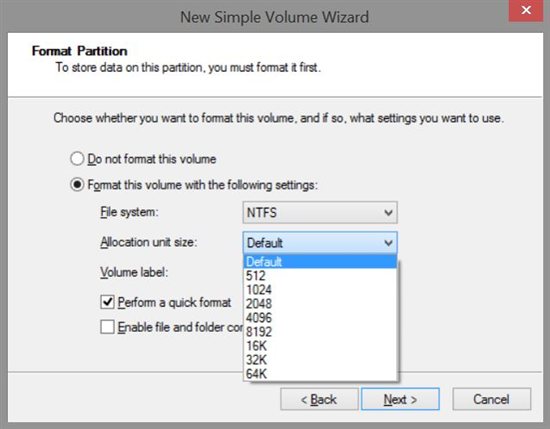
### Database >15ms, Logs> 2ms Tempdb> 2ms



***Do not use the default allocation unit when formatting the Disk drives for SQL Server Log files & Data files:***



### **Format drives with 64k Cluster Allocation Unit**



### **Antivirus programs:**

These programs can create issues with SQL Server functionality, and it is highly important to exclude them from their scan scope, by adding them to the exclusions list.

File types to exclude:

***\*.mdf, \*.ndf, \*.ldf, \*.bak***

### **Always Toggle the maximum memory setting on the SQL Server Instance level properties. Ensure at least 2GB – 4GB of RAM is available to the OS**

### SQL-Perf-4

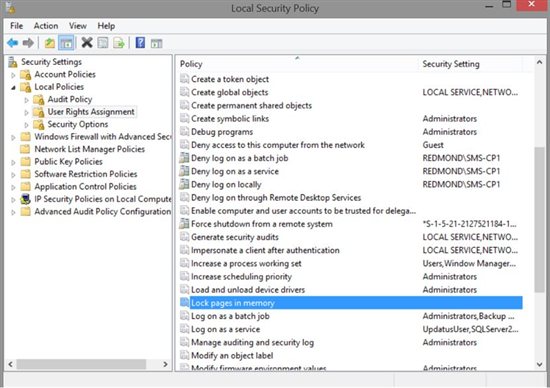
**Note: The maximum memory setting is only for the SQL Server buffer pool, and does not include memory requirements for other SQL Server functions like replication Services, Agent Job Service etc.**

### **Storage Planning for TempDB**

* Set the recovery model of TempDB to SIMPLE. This model automatically reclaims log space to keep space requirements small.
* Do not allow for TempDB files to automatically grow. This reduces the CPU overhead of managing a dynamic file growth.
* Have multiple data file for TempDB (Total Number of TempDB Primary Data File = Number of Processors available to SQL)
* Each data file should be of equal size.
* Try to keep each data files in separate disk drives for IO Parallelism.
* TempDB Data and Log files should be kept in faster disk drive (Preferably RAID 1 if possible)
* Use RAID-10 or SSD Disks.
* Pre-size TempDB files
* 25% of largest DB size.
* Set Auto Growth to fixed size < 200 MB
* You should have the same number of data files as the number of CPUs up to a maximum of 8.

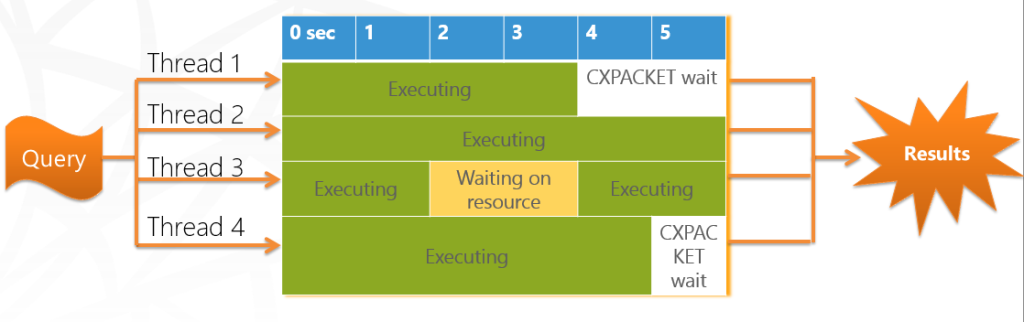
### **Enable the Lock Pages in Memory Option (Windows)**

This Windows policy determines which accounts can use a process to keep data in physical memory, preventing the system from paging the data to virtual memory on disk.



### **Max Degree of Parallelism**

If you have a multi-core CPU which have a high number of cores (<8), you can change the MAXDOP setting to 1 (Recommended for SharePoint 2010 & 2013, MS CRM). Too many threads can delay the performance of the SQL Server. See image below:



This should cover the basic Physical Server performance tuning best practices. Next we would discuss about the [various tools available](https://www.faceofit.com/tools-to-monitor-sql-server-performance/) for analyzing SQL Server performance.

Affiliate Disclosure: Faceofit.com is a participant in the Amazon Services LLC Associates Program. As an Amazon Associate we earn from qualifying purchases.

### **Free Tools to Monitor SQL Server Performance**

<https://www.faceofit.com/free-tools-to-monitor-sql-server-performance/>

In this post, I will talk about the various tools that is available for most of the SQL Server versions and releases.

***Here is my list of SQL Server tools which are free to use and can help you to manage and monitor***[***SQL Server performance***](https://www.faceofit.com/sql-server-performance-tuning/)***.***

* Performance Monitor
* SQL Server Profiler
* DMVs
* Performance Dashboard 2012
* Server Dashboard
* Activity Monitor
* Extended events
* Database Tuning Advisor
* SQL Diag
* SQL Nexus
* ReadTrace

### **Tools for measuring CPU Bottleneck:**

* Performance Monitor
* SQL Server Profiler
* DMVs
* Extended Events
* Data collector and the management data warehouse (MDW)

### *What to do?*

***Identify the queries taking High CPU using above tools***

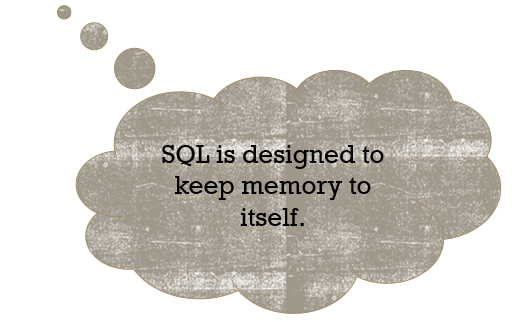
***Check Stats info. Check Missing Index info***

***Check for DTA recommendations***

***Manually tune the query.***

### **Memory Bottleneck:**

* General system and SQL Server state and memory-specific DMVs.
* The DBCC MEMORYSTATUS command.
* SQL Server ring buffers.
* Performance counters.
* The SQL Server error log, and Windows application and system logs.



#### **Windows Performance Monitor**

Performance monitor is used to uncover resource limits in the following:

* Disk IO
* Processor
* Memory
* Network IO

#### **SQL Internal Tools**

* SQL Performance Dashboard Reports. (Available till SQL 2012)
* SQL Profiler (Available in all versions)
* SQLDIAG.EXE
* Dynamic Management Views (DMV)

#### **Other Tools:**

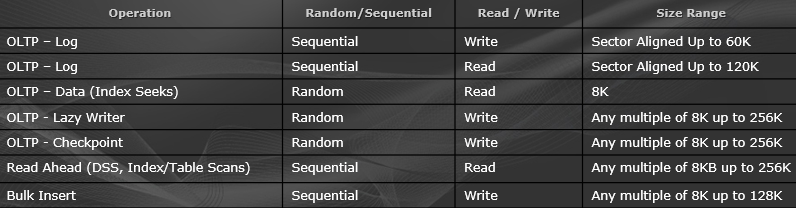
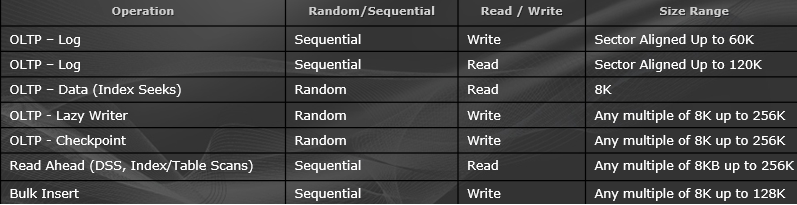
* Activity Monitor
* MDW/UCP (Deprecated in SQL Server 2016)

# SQL Server Performance I/O Characteristics

<https://www.faceofit.com/sql-server-performance-general-best-practices/>

* **SQL Server Performance I/O Characteristics :** SQL Server has different IOPS characteristics and it is difficult to generalize IO patterns of SQL Server. SQL is a back-end platform on which applications are built hence IO patterns may differ significantly from one application to another.[Monitoring of I/O](https://www.faceofit.com/tools-to-monitor-sql-server-performance/) is necessary to determine specifics of each scenario. Understanding the I/O characteristics of common SQL Server operations/scenarios can help determine how to configure storage requirements.

**General IO characteristics of common scenarios:**



### *There is no one single “right” way to configure storage for optimal performance*

### **General Performance Considerations:**

* Storage design considerations differ for large vs. small or consolidated  environments
* Understanding the I/O characteristics is key
* General guidelines
* More/faster spindles is always better for performance;
* Especially true for OLTP or workloads with random IO patterns
* Engage the engineers from all sides, early on
* Ensure storage engineers have at least some knowledge of SQL best practices
* Try not to “over” optimize, simpler designs generally offer good performance and more flexibility
* Validate configurations prior to deployment

### **Performance – RAID Level**

* Best Practice: log files on RAID 1+0 disks
* Best Practice: Isolate log from data at the physical disk level (more on isolation later)
* Tempdb may realize a performance if placed on RAID 1+0
* Our results indicate performance gain on RAID 1+0 for write intensive workloads but at a higher cost ($)
* The performance difference between RAID 1+0 and RAID 5 can vary by vendor
* Benchmarking of the storage can give a clear indication of the performance differences between RAID levels before SQL Server is deployed
* For RAID levels other than RAID 5, 1, or 1+0 test to ensure performance is acceptable

### **Designing Storage Design for Optimal Performance**

* Multiple smaller LUNs are preferred over a single large LUN
* Adverse impact of long running CHKDSK is minimized
* Potentially better load balancing across array service processors
* For huge databases, multiple large LUNs are acceptable (though smaller ones are preferred)
* Fewer large LUNs can accommodate large Databases & would be easier to manage
* Cost of CHKDSK may be acceptable if volumes contain a small number of files
* Consider specific array architecture and use multiple LUNs to ensure proper balancing of LUN’s across array service processors
* Design/Plan adequately for growth
* Other Considerations:
* More LUNs = multiple independent queues, thus potential for better parallel I/O operations
* Assuming scalability at the back end and no bottlenecks exist elsewhere, a system will scale better having multiple paths
* Rebuild as a result of failed disks will effect LUN’s spanning that RAID group

***SQL Considerations***

***Backup/restore – 1 thread per volume***

***File initialization – 1 thread per volume***

### **Designing Database Files/Groups for Performance**

* ***How many data files/filegroups should I have?***
* More data files does not necessarily equal better performance
* Determined mainly by hardware capacity
* Consider disaster recovery requirements
* Will the target environment for a disaster recovery restore accommodate the file sizes?
* Number of data files may impact scalability
* ROT:  .25 to 1 data files (per filegroup) for each CPU (core) on the host server
* Mainly a concern for applications with high rate of page allocations (insert) on systems with >= 4 CPUs
* Generally more of a consideration for Tempdb than for user databases
* However, consider overall data volume and file size
* Can be used to maximize # of spindles – Data files can be used to “stripe” database across more physical spindles
* Multiple filegroups may be optimal for backup / recovery scenarios of larger datasets
* Best practice: Pre-size data/log files, use equal size for files within a single file group and do not rely on AUTOGROW

### **Managing File Growth:**

* Design for growth from the beginning
* Dependent on features offered by storage array in terms of IO performance.
* Most modern storage arrays offer the ability to dynamically grow a LUN – consult with your storage vendor

#### **Two types of GROWTH**

* Capacity vs. Additional performance (more physical disks)
* Windows perspective
* Basic or Dynamic disks – Either can be expanded
* However – dynamic striped volumes cannot be extended
* Basic disks can be expanded using Diskpart.exe
  + Changes to underlying LUNs may require a rescan for Windows to recognize them

