

## SQL SERVER – HOW TO PERFORMANCE TUNE

Databases have been part of electronic computing for the longest time. Database servers contain actual databases and run relative software. A database management system is software that controls databases. It affords easy storage and retrieval of data in a structured manner in large systems.

An **SQL server** is a relational database management system. The smooth working of an SQL server can be ensured with certain enhancement techniques.

### Clusters

A **cluster SQL server** allows connectivity to other physical servers that act as intentional failover partners. A cluster offers high redundancy increasing uptime for critical applications. Regular upgrade to **SQL server** cluster can be performed without considerable downtime using modern enhancement tools.

- **Clustering** is a hardware and software technological innovation that provides high availability to web applications and database clusters.
- **Load balancing** clusters are designed to distribute sever load across multiple front end servers.

A variety of options can ensure effective functioning of the servers. There by clients and end users experience commendable performance without connection outages. The identification of faulty components can assure easy function of SQL servers.

### Performance Monitors

Maintenance and monetary factors play a part in determining performance enhancement procedures undertaken. Performance monitor procedure is the proven way to **performance tune SQL server**. Performance monitor is more like a quality control procedure. It allows regular check of performance statistics and log file recording.

Performance monitor process is a handy tool for database administrators to detect functional problems. The saved log files are monitored at periodic intervals to ascertain faulty components. They can also be subjected to further analysis of data. The log files can be exported as excel files.

This expands data analysis parameters and formulae. **Performance monitor** procedure can also enhance functioning of operating systems to an extent. There is scope for optimization of speed of application run time. This is done by working on application requirements at the server end.

The file log stores information about the files copied during an installation. It is a resident on the user's system. A counter log gives information about which performance counters to log onto. Collection of long term critical performance data can help capacity governance.

They can be used to gauge baseline performance statistics and curves. This promotes good business sense in proportion to performance measurements. Counter log set up automatically gathers performance data of **SQL server** on a regular basis. Performance statistics of SQL servers in the past and future can be assured through long term performance monitoring methods.

Performance data can be easily loaded in to a **SQL server** table since the log file is a comma delimited text file. Counter log and log files created have the same specifications. The first record holds names of each counter. The remaining is seen to contain a data row for each interval logged.

- Performance data can be transformed in to **SQL server** table which allows customization of database usage trend measures.
- Performance counter contains appropriate performance measurement for an operating system machine. They are usually like **pages/second** or **logins/second**.
- Performance counter is a form of performance object that contains multiple counters.

The **pages/second** is seen with memory performance object while **logins/second** is with general statistics performance object. Counter log tabulates frequency of counter value logs, counter log manuscript and counter log run time. Counter logs can be set to run manually, be initiated and suspended based on set criteria. They can also be rolled over.

Multiple counter logs have individual counter log for a certain appropriate activity. They can be used to determine memory use and control **SQL server** activity. Log creation and interpretation can help monitor network function. Logs work to identify and unclog bottlenecks in the system.

**Primary data collection** works around server specifications, processor and network. Logs allow analysis of performance at will. This is easier over continuous monitoring of the system.

**Logs** accommodate review of change in resource usage from the past.

**Logs** can date back to the past with account for origin time of fault. Heavy duty hardware can push the performance of an SQL server. This should be accentuated by suitable application and database design. Poor configuration of server hardware can significantly reduce over all performance.

Clients can be provided with advanced speed and performance through data validations.

Compatibility criteria should be regarded with statistics to **performance tune SQL server** and operating system optimization. An active work station is employed with performance monitor process. The work station watches over logs, performance criteria and traces.

The performance monitor can be accessed through administrative tools of control panel. Performance optimization can be underscored with administrative tools. Performance logs and alerts are available on the left side of performance window. **Counter logs** can also be accessed from there.

They can further be logged in a file with performance counter settings. A default counter log list is created and made available at regular intervals of time. There is scope for customization with new log settings and right click options. The name of the database server can be assigned to the log setting.

This works against ambiguity to its server. The work station name of the system is replaced by that of the **cluster SQL server** to **performance tune SQL server**. The server accumulates the data of performance objects with regards to the employed **SQL server** software after immediate name change. This collection of information takes a few minutes.

Certain sequential steps have to be pursued to **performance tune SQL server**. The memory object is selected from the dropdown menu of performance object. This brings about a change in the counter list. The add button under pages/sec counter is clicked.

The **Add** button can also be executed with available **Mbytes** counter. The above actions empower to monitor **performance tune SQL server** procedure. The counter log is seen to be in the running mode. A right click on the new counter's log name displays the start option to be grayed out.

The above actions fail to initialize with username and password restrictions. This is rectified by changing username and password through properties. A new username and password can also be assigned. The half empty rows and cells have to be deleted to **performance tune SQL server**.

The unnecessary contents of the cells, rows should be highlighted and followed by delete click to initiate erase.

The assessment of processor queue length can determine pressure exerted on the central processing unit. This is crucial to **performance tune SQL server**. A number that averages around one or more with a failed

**cluster SQL server** indicates certain things. This shows that the server has put on hold a series of listed tasks.

The average number displayed is often seen to go beyond one. This can be because of installation of other software applications while setting up web server on the **cluster SQLserver**. To **performance tune SQL server** in an efficient way calls for elimination of unwanted applications. This enhances performance of the **cluster SQL server**.

The performance figures of SQL server can be monitored to observe for processor queue length displaying lesser than one. The remote desktop can be added to the **cluster SQLserver** to derive optimum benefits. The process to **performance tune SQL server** can be initiated from anywhere.

An evaluation of essential requirements and application of appropriate object contemplations is advised to **cluster SQL server**. The most common problem encountered by end users is slow functioning of a system. A lot more needs to be done than just accommodations in the database. The individual components of a system have to work in coordination to deliver business specific needs.

There is no focal point in a system to **performance tune SQL server**. There is dissipation of attention to all regions to assure free flow of information. For performance tuning **SQL server** knowledge of effective functional parts should be known. The primary step is to be aware of all the parts in the system.

A manual that details all individual parts can be used as a guide. Evaluation is considered the second step to performance tuning **SQL server**. The individual parts to be tested should be evaluated. The metrics of all the components of a system should be considered.

The faulty component can be evaluated by running a check on the database and applications.

## Cluster SQL server

An **active/passive** configuration works best to optimize performance with SQL servers. The **clustering** of SQL servers can be very valuable to large organizations.

SQL servers can be effectively clustered by following certain guidelines. It is better to buy only approved cluster systems. Documentation of actions pursued while setting up cluster can be beneficial at a later stage. Regular testing and monitoring of the server cluster is advised.

A recovery plan can be designed to work against simultaneous failure of all individual servers.

The clustering hardware used should be part of the compatibility list of the operating system.

Poor configuration of SCSI cards lead to data corruption. The use of inappropriate network drivers affects speed. Individual workstations hardware has an influencing effect on **SQL server** performance.

A small number of **SQL server** databases can run effectively on many individual servers rather than multiple databases on a single large server.