Migrate to SQL Server's next release with ease and optimize the database workload at any time!

The Challenge:

Your company is upgrading to the next release of SQL Server on new servers. Whether the database workload has 500 or 5000 users, 500 or 5,000 SQL requests being processed/second, OLTP or business intelligence - the database upgrade must occur with no impact on the business operation. Your boss has asked how this can be performed with minimal risk of impacting the business operation and end-users. End-user response time must be at least what is presently being delivered. A methodology must be in place that allows **measurement** of the performance of the new database servers and **confirmation** that service is better than the current environment.

The **Compare Performance between CURRENT and NEW Release** section at the end of the white paper has screenshots of the results of the analysis for migration to a NEW release or to **optimize the database workload**.

Zero Impact SQL Capture Agent captures 100% of the SQL activity:

Our **Zero Impact SQL Capture Agent** (AGENT) uses non-intrusive network sniffing to capture 100% of the SQL activity. The SQL text, bind parameters, end-user response time and more are captured for every SQL request. The AGENT has a low 1% overhead and does **not connect** to the monitored database server. Think of it as a ZERO impact SQL profiler or trace facility. An AGENT license is included at no charge with every license to the Sql Power Tools **Agentless Bottleneck Monitor**.

Key Features of AGENT:

- Can always reconstruct with **complete accuracy** the exact SQL with its bind parameters that were running when server performance is poor or when end-users complain about **poor response time**.
- Top N Sql Analysis of 100% of the agent based SQL capture. The SQL performance is tracked for the 6,000 **unique SQL requests** that typically run on a server 100+ million times a day.
- **100% accurate** comparison of SQL performance between servers or time periods. Can be done on an ad hoc or scheduled basis. Both **improved** and **degraded** SQL end-user response time is reported for the 6,000 unique SQL requests that typically run on a server 100+ million times a day.
- Zero Impact SQL Capture Agent usage is **optional**. Can be installed at any time and dynamically enabled or disabled on servers of interest.
- Enable the agent on **priority** servers where end-user response time is important, on servers experiencing **performance issues** that cannot be pinpointed with agentless monitoring, or when a **server change** is about to be made.
- Agent can optionally be run on a secondary host (not the database server) for non-intrusive sniffing of port mirrored SQL packet flow.

Agentless Bottleneck Monitor pinpoints server bottlenecks before and after:

Sql Power Tools **Agentless Bottleneck Monitor** has a low, low, low monitoring overhead. Overhead is < 1%.

Server Bottlenecks are Monitored:

- Comprehensive wait state monitoring of the **600+ wait types** that a database server posts. The wait types causing SQL statements and stored procedures to not complete on a **timely basis** due to resource contention or being queued due to a lack of available resources are monitored. Wait condition **wait times** are tracked by SQL statement, stored procedure, application, wait type, database and the end-user client. Reduce **SQL wait time** -> reduce **end-user response time** -> improve server **throughput**. In most cases this can eliminate an expensive and unnecessary hardware upgrade.
- Database **I/O stall/wait time** monitoring on all database data and log files. If the average stall/wait time on a database file > 25 milli seconds, the disk subsystem is slow, I/O is not balanced, memory pressure exists or has poor performing SQL.
- **TempDb contention**. Pinpoints all SQL statements, wait types, applications, end-users and stored procedures experiencing a wait condition/wait time on the TempDB database and files. Average I/O stall times on all TempDb data and log files are monitored as well as the number of I/O operations and throughput on TempDB.
- **Excessive SQL statement I/O**. Automatically **mines SQL plans** for poor performing SQL. Table scans, missing indexes, sorts, timeouts and much more are pinpointed.

- CPU, I/O and memory pressure.

Other Agentless Bottleneck Monitor Features:

- Monitors SQL statement performance, blocking, deadlocks, performance counters and SQL jobs.
- View performance in real-time or over any prior time period plus real-time alerting.
- 3D server farm real-time views of performance counters, wait times and I/O stall times.
- 3D server farm views of buffer and procedure cache memory usage.
- 3D server farm views of database and disk drive space usage.
- 3D server farm views of index usage, missing indexes and index fragmentation.
- Baseline analysis of server farm performance plus ad hoc comparison of performance.
- View the performance of the entire server farm in a 100% web enabled interface.
- One click install and setup wizard makes installation a snap. Free product training.

The Database Workload Methodology:

The Zero Impact SQL Capture Agent is used to capture 100% of the **production** server's SQL activity with it's end-user response time for the CURRENT database release. The SQL will then be replayed against the NEW database release. The **Agentless Bottleneck Monitor** is run during the original capture and the multi-threaded replay against the NEW database release so that wait conditions, I/O device stall time, blocking, deadlocks, TempDb contention, SQL statement I/O, poor SQL plans and performance counter values are captured for review and comparison. **IMPORTANT**: Do not use a QA server since they rarely can duplicate the load and concurrency of a production server.

1. Capture 100% of the production server SQL activity with it's end-user response time for **CURRENT** database server release using **Zero Impact SQL Capture Agent** (AGENT). A business day of SQL activity is recommended.
2. **Agentless Bottleneck Monitor** is also run when AGENT is run so that server bottlenecks and performance metrics are available in the product database repository for the **CURRENT** release.
3. Perform a **multi-threaded replay** of 100% of the production server SQL activity captured by the AGENT in step 1 using the SQL Server Replay Tool. Replay is against the **NEW** database server release and server hardware.
4. Capture 100% of the SQL activity with it's end-user response time during the **multi-threaded replay** against the **NEW** database server release using **Zero Impact SQL Capture Agent** (AGENT).
5. **Agentless Bottleneck Monitor** is also run during **multi-threaded replay** so that server bottlenecks and performance metrics are available in the product database repository for the **NEW** release.

Compare Performance between CURRENT and NEW Release:

The **Agentless Bottleneck Monitor**'s 100% web enables user interface allows the performance of two servers or time periods to compared at any time. If an entire day of SQL activity was captured then the CURRENT and NEW release may be compared over the entire day or any subset of the day. E.g. the peak processing hour.

The following can be compared between the releases:

1. SQL Performance

The performance of 100% of the SQL activity captured using the Zero Impact SQL Capture Agent (AGENT). SQL statements flushed out of procedure cache during the business day due to high use of ad hoc SQL or memory pressure will not be missed.

Both improved and degraded SQL end-user response time is reported for the 6,000 unique SQL requests that typically run on a server 100+ million times a day. Click on right to view.

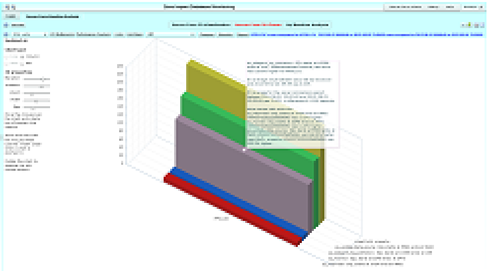
2. Wait Types with Wait Time

The wait types (600+ that a database server posts) causing SQL statements and stored procedures to not complete on a timely basis due to resource contention or being queued due to a lack of available resources are monitored. **Wait condition** wait times are tracked by SQL statement,

stored procedure, application, wait type, database and the end-user client. Reduce **SQL wait time** -> reduce **end-user response time** -> improve server throughput. In most cases this can eliminate an **unnecessary hardware upgrade**.

Automatic **drill down** into the wait types, wait times and performance counters for any poor performing SQL. Pinpoints the resource contention on the server that is causing the SQL statement to not complete on a timely basis.

A compare of the CURRENT AND NEW release identifies SQL statements, stored procedures, applications, wait types and databases having a decrease or increase in wait time. Click on right to view.

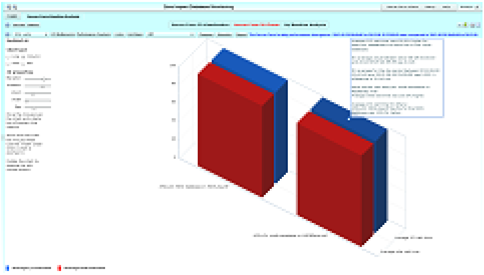


When in the web interface click on any performance counter having a poor value and the SQL statements plus wait conditions will be graphed.

3. I/O Stall and Wait Times on all Database Devices

If the average stall/wait time on a database data or log file > 25 milli seconds, the disk subsystem is slow, I/O is not balanced, memory pressure exists or has poor performing SQL.

Both degraded and improved I/O stall/wait times are reported. The I/O stall/wait times may be filtered and viewed over any prior time period. Click on right to view.



4. Blocking, Deadlocks and Performance Counter Values

An increase or decrease in blocking, deadlocks and performance counter values are identified. If an entire day was monitored with the Sql Power Tools **Agentless Bottleneck Monitor** then the performance counters for the CURRENT and NEW release may be compared over the entire day or any subset of the day. E.g. the peak processing hour.

Additional Agentless Monitoring Functionality

Additional screenshots and a product specification recap are available. Sql Power Tools **agentless monitor** has a **low, low, low** CPU, I/O and network overhead on monitored servers. Overhead is < 1%. We do NOT:

- Create tables in the TempDB database or issue millions of **I/O operations to TempDB** on monitored database servers.
- Issue millions of ad hoc SQL requests to the monitored servers that **bloat the procedure cache** and cause expensive SQL compilations to occur.
- Increase the database server wait conditions and **wait time (due to resource contention)** thereby impacting production users response time.
- Send millions of characters of SQL monitoring text **across the network**.
- Turn on SQL Profiler or SQL Trace that can **reduce server throughput** .
- SYSADMIN role is **NOT used** for agentless monitoring of the server farm.

Next - Optimize the Database Workload at any time!

Since 100% of the SQL activity with the end-user response time can be captured the impact of change can be accurately measured! Just run the Zero Impact SQL Capture Agent before and after an important change is made and use **The Database Workload Methodology**.

Example: Before and after a **hardware upgrade**, change in the server configuration, **table index changes**, SQL code changes, regression test of a **new application release** in QA, deployment of a new application release to production, or upgrade to a new release of the database server software such as Oracle 12, SQL Server 2012 or Sybase 15.7.

Now SQL statements having an improved or degraded end-user response time are accurately known. Performance issues can be immediately resolved prior to end-users possibly experiencing **poor performance** in production.

Contact Us:

support@sqlpower.com USA: (800) 733-5978 International: 800-7007-0020

Sql Power Tools, Inc.
1712 Pioneer Avenue, Suite 652
Cheyenne, WY 82001 USA