SCS Case Study

Reporting Website Optimization

Solution Snapshot

Business Situation: Customer-facing reports on the client's website were running slower than expected causing long wait periods and timeouts for end users

Multi-Point Solution: Three different areas were looked at in order to optimize performance

Benefits: Reports are now running 30 to 200 percent faster

The Client

Our client is a nationally recognized freight transportation company located in the upper Midwest. They are leaders in providing tailored, technology-driven freight transportation solutions that help their customers achieve a competitive advantage, both locally and nationally.

Today, our client offers a variety of services, including dedicated contract transportation, brokerage, freight warehousing, fleet management and logistics consulting that allows them to meet virtually any customer need. Our client has a reputation for innovative technology and provides web reporting for their freights through the package tracking website.

The Challenge

Customer-facing reports on the package tracking website were running slower than expected causing long wait periods and timeouts for end users. Some online reports intermittently timed out even though the stored procedures were running quickly. Some stored procedures performed better when run in SQL Server Management Studio than in the reports. Developers had to run manual maintenance of indexes and cache when a timeout occurred. The client also indicated the performance issues in the stored procedures could be due to the need for re-structuring of the data in the database.

After observation and analysis of the SQL Server resources, the Reports executions, and the stored procedures, the following problems were identified:

- Reporting stored procedure code was scanning the main drive more than needed and was not utilizing tempdb and memory.
- The reporting site was running on a transactional database on real-time data. This situation caused stale cache on high transaction tables.

- Maintenance jobs needed to be more effective to target the main cause of stale statistics. They also needed to be more efficient so that they run in the background without interfering with transactional and reporting queries.

- Parameter sniffing was causing the website call to the stored procedures to run slow even though they ran fast in SSMS. The parameters were changing daily and cache was not effective. The difference in Estimated Rows and Actual Rows in execution plans was high.

- One of the data entities was hierarchical but maintained in non-hierarchical model, which could potentially cause long-term issues with stored procedure performance and maintenance.

The Solution

Superior Consulting Services (SCS) identified and examined three different areas in order to optimize performance. First, SQL Server configurations and allocations were recognized as potential performance enhancers. Second, certain stored procedures were identified as querying inefficiently and not utilizing the proper optimization. Lastly, the indexes being used in the large queries needed better utilization and specific maintenance.

SQL Server Health

- Moved the tempdb to a separate drive from C:\ Drive and split the tempdb into four files to make it perform faster
- Cleaned up the production environment with MSDB Purge and dropped the unnecessary copy of the production database to reduce resource conflicts

Stored Procedure Optimization

- Optimized stored procedure execution by reducing parameter sniffing stalls: the query hint (OPTION OPTIMIZE FOR UNKNOWN) used for large queries dealing with date ranges
- Optimized stored procedure execution by reducing the effects of stale cache by using filtered intermediary datasets through parameters provided to the stored procedure
- Performed query optimization via utilization of derived tables, temp tables and common table expressions
- Broke down big queries and nested joins to help run the queries with less reliance on indexes

Index Optimization

- Scheduled index maintenance to run daily by rebuilding and reorganizing only indexes needed based on 10-40 index rebuilding rule
- Scheduled maintenance statistics update jobs hourly, bi-hourly and daily in the production environment
- Split jobs for each client and scheduled at alternate hours to maximize available resources

- Scheduled another job to catch concentrated windows of scans that caused high row modifications for indexes
- Created an additional job to update statistics invoked by file feed row modifications

The Result

The package tracking reports are now running 30 to 200 percent faster based on internal testing done by our client. Parameter sniffing that was causing the website to stall was reduced drastically via the combination of query hints, filtering report related data and utilization of memory in stored procedures. The stale cache that was causing the stored procedures low performance was refreshed with the balance of index optimization, tuning and timely job maintenance.

Contact

Superior Consulting Services, LLC
350 West Burnsville Parkway, Suite 550
Burnsville, MN 55337
952-890-0606

www.teamscs.com