Performance tuning case study

Issue

The company name is withheld to protect its identity. The company's e-commerce application was developed using Microsoft ASP, Visual Basic and SQL Server technologies. The application was plagued with serious performance problems due to poor design in application and database. The database was developed using a database from another vendor and custom added tables. There were hundreds of stored procedures that were processing the requests behind the scene. There were serious performance issues and customer complaints during the peak shopping times of the day. These performance problems resulted in low level of user satisfaction and hundreds of thousands of dollars of potential loss of revenue.

Solution

We were tasked with fixing the performance issue. We did an initial overview and came up with a solution to fix the problem in stages. First, we identified the low hanging fruits that could be fixed immediately. Next, over the period of several months we slowly identified the parts of code that caused the performance issues and either rewrote whole stored procedures or partially fixed the existing ones. We recommended hiring another firm from New York to review the front end application that used ASP and Visual Basic code. We worked closely with that firm to identify problem areas in both front application and back end database and fixed them strategically. We recommended moving the application to better server with more memory and SAN storage with a failover cluster. We proactively monitored the server for potential performance issues and fixed them even before they became major issues. We analyzed the database and added indexes wherever necessary. We rebuilt the indexes using online indexing method so as not to affect the database availability. We kept the fragmentation to a minimum. We eliminated many stored procedure recompiles. We eliminated redundant calls to the stored procedure and cached important tables for better performance. We wrote custom monitoring scripts to monitor blocking, stored procedure compiles, memory issues, etc. Finally we recommend moving the application and database to ASP.NET and SQL Server 2005 environment.

Result

We were able to fix some of the performance issues immediately. Over a period of time we slowly improved the performance and finally reached an optimal level where performance issues were completely eliminated. Performance issues during the peak shopping times were minimized. The application was available 24/7. This resulted in improved user satisfaction and increase in revenue of hundreds of thousands of dollars in online revenue.