

# Database Performance Monitoring System

**Project Proposal, Group 23, February 10, 2017**

Drishti Wali (13266), Sahil Verma(150616), Shanu Vashishta (12657)

CS315A: Principles Of Database Systems, Jan-May, 2017

Instructor: Medha Atre, CSE, IIT Kanpur, India

## 1 Abstract

Monitoring of a Database includes measurement of the performance in terms of internal and OS-level statistics (CPU, RAM, Disk, cache). This may be used in order to determine problems in the database and find corresponding solutions as well as measure the query optimization techniques. In this project we aim to design a visual interface displaying the real-time statistics and potentially detecting resource saturation. We would display periodic snapshots as well as aim to track performance trends. Evaluation based on the monitored database performance has been used to "minimize response time and maximize throughput, yielding optimal performance" [3]

## 2 Literature review

Conventional DBMS applications organize and store large amount of data by normalizing and indexing them efficiently. In a Relational DBMS, data is organized according to tables on a logical level, and then into disks, files, extents and segments. In order to evaluate them, several Performance Monitoring systems built by Microsoft, Oracle [5], Appdynamics [1] etc are in use today. These systems support the most widely used databases like MySQL, Oracle, MongoDB, Microsoft Azure etc and help in tackling problems faced by organizations. A key component of these has also been the visualization power which they provide to their users. These not only help evaluate user activity but, also ease the debugging process. Support is also provided to view the live execution plan of an active query. Tools like QE3D [2], which help in the query plan visualization through a three-dimensional representation help in hotspot identification in real-world scenarios.

## 3 Plan for the project

We aim to develop the interface mainly using java and python interfacing with the database. We would like to follow the following plan for the project at this stage.

- **OS Level Statistics:** At the OS level, we would like to focus on the important elements like CPU Utilization, Memory Utilization and Disk space. These form the crux metrics which will help differentiate the database systems.
- **DB Level Statistics:** At the level of the Database, we would monitor the performance using Access methods counters, SQL Server buffer pool, keep track of the number of SQL statements that are being executed per second as well as the current processes waiting. We will also look at the Database locks, TempDb usage and the Server logs using SQL Server Management Studio. [6]
- **Visual Interface:** We will then prepare a visual interface which is user-friendly as well as helps interpret the performance for optimization and benchmarking. We aim to provide visual charts, heatmaps of the memory consumption and query plan execution visualization. [4]

## References

- [1] Appdynamics. Top 6 database performance metrics to monitor in enterprise applications. URL <https://blog.appdynamics.com/engineering/top-6-database-performance-metrics-to-monitor-in-enterprise-applications/>.
- [2] C. D. Daniel Scheibli and A. Boehm. Qe3d: Interactive visualization and exploration of complex, distributed query plans. *SIGMOD '15 Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data Pages 877-881*, 2015.
- [3] MSDN. Monitor and tune for performance. URL <https://msdn.microsoft.com/en-us/library/ms189081.aspx>.
- [4] MySQL. Visual explain plan. URL <https://dev.mysql.com/doc/workbench/en/wb-performance-explain.html>.
- [5] Oracle. Monitoring real-time database performance. URL [https://docs.oracle.com/cd/B2835901/server.111/b28275/tdppt\\_realtime.htm](https://docs.oracle.com/cd/B2835901/server.111/b28275/tdppt_realtime.htm) *TD PPT033*.
- [6] M. Otey. The essential guide to database monitoring. *Whitepapers*, 2015.