International Journal of Academic Information Systems Research (IJAISR)
ISSN: 2643-9026
Vol. 4 Issue 11, November - 2020, Pages: 39-43

Performance Analysis of Percona Database Performance as Input Output Utilities Refinement for MySQL

Karnoto*, Ajub Ajulian, Sudjadi, Achmad Hidayatno, Aris Triwiyatno, Susatyo Handoko, Sukiswo, Agung Nugroho, Tejo Sukmadi, Imam Santoso, Yosua Alvin Adi Soetrisno

Department of Electrical Engineering, Diponegoro University, Semarang, Indonesia
***Corresponding author**: karnoto69@gmail.com

***Abstract**: This paper do some performance analysis based on I/O tuning on Percona Database according to MySQL. However, MySQL servers often face performance issues, leading to many websites looking for alternate high performance databases. Percona server become more popular in 2013, as a high performance database. Percona could become alternative to MySQL that has features comparable to the MySQL Enterprise version. Query speed are tested using large database with setting of different indicator in each database. Employee test database is used because have more than three million data. Native MySQL version is version 5.7 and Percona version is 5.7. I/O tuning in thread, cache and configurable setting in Percona could give performance increase in execution time although the number of concurrent connection that used is above 150 connection.*

Keywords—Performance analysis, I/O tuning, MySQL, Percona

1. INTRODUCTION

Transactional data that inputted to the information system require specific database management system. Database management increase in scale is happening when the business is growth. For example in this research, database employee is increasing because store an employee and salaries data for more than ten years. Database increasing with concurrent access increasing need a performance tuning. Performance tuning need some modification in memory configuration, buffer pool configuration, and also in specific tuning configuration [1]. Business nowadays face a various challenge including consider the best RDBMS for the information system that developed. MySQL is one favor of open source and easy to use structured database technology that could be used for ACID type database [2]. There is various kind of MySQL modification that done by company to get the best tuning performance. One kind of forking MySQL technology is Percona database. Percona and MySQL have same basic engine and feature because the technology behind that is the same [3]. Percona has more engagement in 2013, because there is a need of database concurrent access.

Transactional processing is the first process that need to make a decision making. Nowadays there is a lot of artificial intelligence application that used to make a certain decision. AI model cannot exist without database technology that firstly stored the data. Database is used in a lot of application and now stored a huge amount of data in the term of big data technology. Every web application and also web service tied with data and need a lot of query to access different scenario exist at the backend database [4]. From this situation the need of performance analysis is to ensure that the data is ready and available although there is a heavy traffic [5]. Light and fast database technology is rely within the database architecture, we must modify to make the power of thread, and I/O could come out.

The approach of this research is to make the performance of the Percona database become optimal. Optimal performance configuration could be arranged with native configuration with native ability and modification of the certain infrastructure configuration. The complex queries and scenario is used to ensure the performance is not only working with simple query. There is some consideration to use complex queries because database tuning sometime also work in complex queries. Percona and MySQL is open source version distribution so that could be accessed easily and freely. Percona is community version that has a feature and capability of enterprise version of MySQL. The idea is to test that I/O configuration was really fit with certain scenario in Percona, so Percona could really work well and outperform MySQL.

2. BACKGROUND

There is a lot of similarity between databases that forked from MySQL technology. The main different between Percona and MySQL is that Percona used fractal graph rather than binary tree graph when there is an index in database. This section deliver the background and also technology between MySQL and Percona database system.

2.1 MySQL

Oracle MySQL has the most longevity out of these three database management systems. Oracle MySQL usually becomes a default feature of database installation exists on the web hosts and it also acts as the default selection for some content management system.

Oracle offers two versions of MySQL which is a free community edition and a commercially licensed version. Company may require you to use the commercial edition when needs an advanced security and full level backup and rollback. MySQL Enterprise Edition has a fully integrated transaction-safe, ACID-compliant database with full commit, rollback, crash-recovery, and row-level locking capabilities

which is also available in Percona database [6]. MySQL Enterprise Edition has extra features to provide monitoring and online backup, as well as improved security and scalability. MySQL Enterprise also adds a number of advanced modules that help users to work more efficiently with the tool. These modules range from workflow optimizations to extended object handling. Performance analysis of this research using "MySQLSlap" which is included in a normal packet installation.

The biggest advantage to using Oracle MySQL is the stable code base and the large community. Web applications that depend on an incredibly reliable and predictable system, which is not too experimental in some configuration. Some configuration could mislead the normal performance use of MySQL such as I/O tuning formulation and query locking system [7]. Oracle's offering stands out performance in normal transaction process and not considering the number of concurrent connection.

2.2 Percona MySQL

Percona MySQL takes a more conservative approach to better support extremely demanding application requirements. Resource-intensive applications will maximize the performance improvements and optimization tools that Percona MySQL has available.

Percona has a special clustering features, which is XtraDB. In this research, we are using InnoDB rather than XtraDB. XtraDB is more performance-oriented which has plenty of ways to monitor and improve its operation. If there is a compatibility issues with other MySQL forks, Percona MySQL is closer to the MySQL base code. Percona offers a set of MySQL toolkit, which is very usefull for the database administrator to do administration activities.

Percona MySQL is bringing enterprise features into their fully open-source MySQL version. If the commercial licenses for Oracle MySQL exceed a project's budget, Percona may have several features that same which is fully open source.

The best choice for the applications depends on the size, scale, and resource demands that allocated for the database [8]. Each MySQL flavor offers plenty of benefits, and in some cases, it may come down to the development team's personal preferences. Existing scenario which has plenty of uses and row could test the condition when there is query that exploit the uses of calculation in entire database [9]. Query that calculate total salary at the quarter year from each employee in each department and make a profile analysis of each department could become the best scenario to test the database performance [10].

3. Experimental Setup

3.1 Dataset

The dataset used for this research is from the original data was created by Fusheng Wang and Carlo Zaniolo at Siemens Corporate Research. The data is in XML format that available at http://timecenter.cs.aau.dk/software.htm. Giuseppe Maxia has the contribution for the relational schema and Patrick Crews has contribution at the data in relational format. The database exists with 300,000 employee records with relation with 2.8 million salary entries. The total size of database is 167 MB, which is not very big at size, but heavy enough to do a performance based testing or summary report query for trivial testing.

3.2 System Specification

System is running inside Virtual Machine Technology with CPU Processor Intel i5 4210U. System runs inside VM because the environmental setup is running on the two different configuration with the need of the same specification environment. CPU has a core speed at 2394.29 MHz and has L1 D-Cache with 32 Kbytes x 2 and L1 I-Cache 32 Kbytes x 2. L2 Cache is 256 Kbytes x2 and L3 cache is 3 Mbytes. Virtual machine environment only used half of total eight virtual core processor with two real processor. Virtual memory only using 4 Gbytes of the computer's total memory which is 12 Gbytes with DDR3 technology with DRAM frequency at 798.1 Mhz. Hard disk type that used at the VM environment is Sandisk SSD type with average 4K random write speed at 71 MB/s.

The first parameter of performance testing is taken from "MySQLSlap" tools. "MySQLSlap" is a tool for load-testing in MySQL database infrastructure. It allows to emulate multiple concurrent connections, and run a set of queries multiple times in the iterations terms. "MySQLSlap" could do a general performance testing without specific table scheme or a specific performanc testing for specific table schema. Figure 1 shows the "MySQLSlap" testing on the default MySQL architecture with no schema in 150 concurrent connection, done in ten iterations to validate the concurrent connection ten times. The main parameter that used in table comparison is the average number of seconds to run all queries. The scenario for testing the concurrent connection is increasing the concurrent the connections above the default limit to test the I/O performance and also the CPU and memory performance.

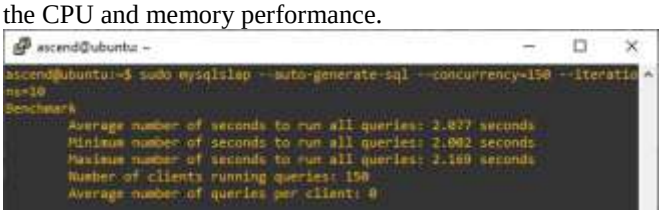

Fig. 1. *MySQLSlap Performance Testing*

The second parameter of performance testing is taken from "I/OStat" tools. The "iostat" command is used for monitoring system input/output device loading by observing the time the devices are active in relation to their average transfer rates. The "iostat" command generates reports that can be used to change system configuration to better balance the input/output load between physical disks. "I/OStat" is also tested with increasing of concurrent connection. Figure 2 shows the example of "I/OStat" tools which is testing the

MySQL architecture for default concurrent maximal connection.

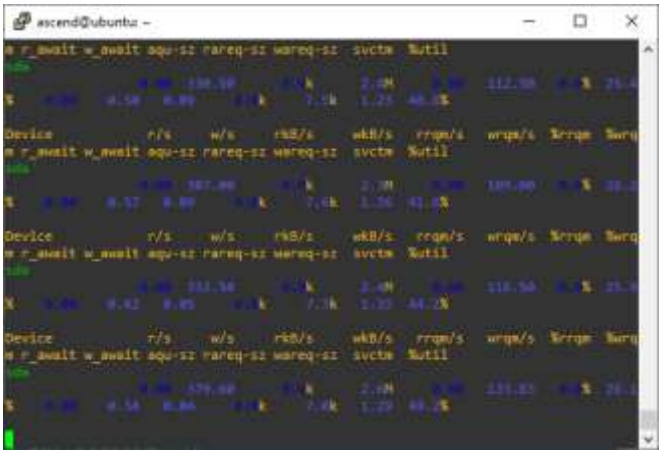

Fig. 2.*I/OStat Performance Tuning*

The third parameter of performance testing is taken from "top" tools. The "top" command is used to show the Linux processes with could be filtered with specific services. It provides a dynamic real-time view of the running system. The maximum CPU and memory resource is taken when the concurrent connection is coming. Usually, this command shows the summary information of the system and the list of processes or threads which are currently managed by the Linux Kernel. Figure 3 shows "top" performance testing of MySQL architecture which is encountering the concurrent connection.

Fig. 3.*Top Performance Testing*

Table 1 shows the first scenario which is comparing MySQL with Percona in the number of concurrent connection increases. Average number of seconds was slightly increased when the concurrent connection is increased. It is followed by maximum I/O capacity to write to the disk every seconds, I/O utilities, CPU utilities, and also Memory utilities. Percona is has a slower time in execution in seconds if compared with MySQL. Percona on the other has a slightly increase the maximum I/O capacity which is followed with I/O and CPU utilities. It is shows that Percona has a better performance in queueing the query to be written in the I/O although the average number in certain times is slower than MySQL. Memory utility is slightly slower because Percona has more effective ways in feeding the query to the I/O, so there is more little data queueing inside the memory.

Table 1: Performance analysis on concurrent connection increasing effect on the harddisk, memory, and CPU utilities in MySQL and Percona environment

Database	MySQL				
Condition	Average Number of Seconds	Maximum I/O w/s	I/O Utility	CPU Utility	Memory Utility
Concurrent Connection 150	**2.077**	**379.6**	**49.20%**	**156.80%**	**9.60%**
Concurrent Connection 200	**3.393**	**310**	**40%**	**160%**	**9.70%**
Concurrent Connection 250	**4.703**	**319.5**	**41.60%**	**167.30%**	**11.80%**

Percona				
Average Number of Seconds	Maximum I/O w/s	I/O Utility	CPU Utility	Memory Utility
2.258	**415**	**51.80%**	**158.70%**	**8.20%**
3.564	**477**	**58.40%**	**163%**	**8.50%**
4.892	**406.5**	**46.50%**	**165.10%**	**10.90%**

Table 2 shows the second scenario, which is comparing MySQL with Percona when there is thread set in the global configuration. This scenario is to test the time increasing and thread increasing when the number of thread is set to the same pool when the concurrent connection is still increasing. Percona has a better execution time when the number of thread is sufficient with the number of the connection. The concurrent connection increasing was made slower execution of time when the number of thread is not maximized well. Although the number of thread increasing is same when the concurrent connection is increasing, the execution time was still affected.

Table 2: Thread performance testing between MySQL and Percona environment

Database	MySQL			
Condition	Average Number of Seconds	Time Increasing	Thread Created	Thread Increasing
Thread 100 Connection 150	**2.164**		**20465**	
Thread 100 Connection 200	**3.393**	**1.229**	**21466**	**1001**
Thread 100 Connection 250	**4.914**	**1.521**	**22957**	**1491**

Percona			
Average Number	Time	Thread	Thread

of Seconds	Increasing	Created	Increasing
1.996		**22745**	
3.426	**1.43**	**23746**	**1001**
4.949	**1.523**	**25244**	**1498**

Table 3 shows the redo log testing to test the sustainability of I/O throughput. The redo log is a disk-based data structure used during crash recovery. Redo log encodes requests to change "InnoDB" table data. Redo log is flushed before a transaction is committed. MySQL architecture including in the Percona distribution write to the redo log files in a circular fashion. If there is a flushed mechanism, there is some I/O utility increasing. The performance of execution time is slightly faster when the redo log is disabled. The flushed mechanism is consuming more I/O utility in Percona environment. Although I/O utility is increasing, it is not affecting the execution time directly. The modification of redo log works well in MySQL and Percona environment.

Table 3: Redo Log effect performance testing between MySQL and Percona environment

Database	MySQL		
Condition	Average number of seconds	Maximum I/O w/s	I/O Utility
Redo Log Active Concurrency 100 Iterations 100	**1.143**	**436**	**51%**
Redo Log Inactive Concurrency 100 Iterations 100	**1.02**	**59**	**9.20 %**

Percona		
Average number of seconds	Maximum I/O w/s	I/O Utility
1.259	**621.5**	**59.60%**
1.086	**190.5**	**28.00%**

Table 4 shows the complex queries execution time comparison in MySQL and Percona. The complex query condition to test the overall performance of MySQL and Percona is a summary like query. The query is executed in the form of "SELECT sum(s.salary) FROM employees.employees e join employees.salaries s on e.emp_no=s.emp_no;" command. The query executed one time with one concurrent connection to test exactly the average number of seconds. This scenario shows that with the proper query, the execution time is better. Percona has a better time execution when there is an exploratory of calculation query. This kind of query must explore the whole of database and summarized them become one calculation. This scenario is become more interesting when tested with the variation of concurrent connection and some modification of basic parameter tuning.

Table 4: Complex queries execution time comparison in MySQL and Percona

Database	MySQL	Percona
Condition	Average number of seconds	Average number of seconds
Join Query from Employees and Salaries to make Summary of each Salaries	**13.418**	**12.885**

4. Conclusion

Execution time is not the main point concerning the performance test of the MySQL and Percona. There is some specific scenario and specific modification of database performance parameter that need to be done. Percona has a better I/O write capacity than MySQL in such different situation except the redo log parameter. Percona has a slower execution time in certain testing scenario. Proper query terminology with proper thread provided could maximize the Percona performance in slightly different ways. For the future work, there is some interesting ways to mix the scalability factor which in the Percona is the Xtra DB cluster and in the MySQL is MySQL clustering to test the distribution of I/O affected by I/O capacity tuning.

5. References

[1] J. Huang, "Achieving Performance Predictability in Transactional Databases," p. 116.

[2] S. A. Hosseini, F.-A. Parand, and F. Matinfar, "Data processing in Cassandra vs MySQL: A comparative analysis in the query performance," p. 12.

[3] M. Witham, I. Bender, and R. Gomes, "Comparative Analysis of MariaDB's Performance Efficiency as a Suitable Replacement for MySQL," p. 11.

[4] M. Dawodi, M. H. Hedayati, J. A. Baktash, and A. L. Erfan, "Facebook MySQL Performance vs MySQL Performance," in 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, Oct. 2019, pp. 0103–0109, doi: 10.1109/IEMCON.2019.8936259.

[5] O. Čapligins and A. Ermuiža, "MySQL Database Management System Forks Comparison and Usage," p. 9.

[6] S. Marsidin, "Need Analysis of Development of Principal Performance Assessment Model based on MySQL Software," vol. 372, p. 4.

[7] B. Jose and S. Abraham, "Performance analysis of NoSQL and relational databases with MongoDB and MySQL," Materials Today: Proceedings, vol. 24, pp. 2036–2043, 2020, doi: 10.1016/j.matpr.2020.03.634.

[8] I. Hossain, "Oracle, MySQL, PostgreSQL, SQLite, SQL Server: Performance based competitive analysis," p. 43.

[9] B. Sirish Shetty and K. Akshay, "Performance Analysis of Queries in RDBMS vs NoSQL," in 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), Kannur,Kerala, India, Jul. 2019, pp. 1283–1286, doi: 10.1109/ICICICT46008.2019.8993394.

[10] K. Lachewicz, "Performance analysis of selected database systems: MySQL, MS SQL, PostgerSQL in the context of web applications," p. 7.