

Query Optimization in MySQL Database Using Index

Siti Maesaroh¹, Heru Gunawan², Agung Lestari³, Muhammad Sufyan
Ats Tsaurie⁴, Mohamad Fauji⁵

University of Raharja^{1,2,3,4,5}
Jenderal Sudirman No.40, Cikokol, Kota Tangerang^{1,2,3,4,5}
Indonesia^{1,2,3,4,5}

e-mail: siti.mesaroh@raharja.info¹, heru.gunawan@raharja.info²,
agung.lestari@raharja.info³, m.sufyan@raharja.info⁴, m.fauji@raharja.info⁵



Author
Notification
October 2022
Final Revised
October 2022
Published
October 2022

To cite this document:

Maesaroh, S., Gunawan, H., Lestari, A., Tsaurie, M.S.A., & Fauji, M. (2022). Query Optimization in MySQL Database Using Index. *International Journal of Cyber and IT Service Management (IJCITSM)*, 2(2), 104-110. Retrieved from <https://ijast-journal.org/ijcitsm/index.php/IJCITSM/article/view/84>

DOI:

<https://doi.org/10.34306/ijcitsm.v2i2.84>

Abstract

The quality of a database design has a significant impact on the quality of an information system. Even if a system has been running for a long time, a programmer may forget to optimize a query in MySQL, even though optimization is critical as the amount of data is enormous. MySQL has an index function for tables that helps to speed up database searches. In essence, the index will be beneficial if utilized by the user's interests and wants; otherwise, the index, constructed randomly and in large quantities, would slow down database access. As a result, before constructing a database index, it is necessary to reconsider the rules and conduct a thorough examination.

Keywords : Optimization, MySQL, Index.

1. Introduction

A database is an essential part of any modern information system. A Relational Database Management System (RDBMS), often known as a Relational Database System, is used in practically all of today's information systems[1]. Oracle, MS SQL Server, PostgreSQL, DB2, FirebirdSQL, and MySQL are RDBMS software frequently used in information systems. A database collects data that broadly characterizes an organization's operations and actors; for example, a university database will include students, lecturers, lectures, and other information. SQL (Structured Query Language) is a structured language for accessing and manipulating data in databases and database entities [2]. SQL is also the standard language used in various existing databases, making it simple to transfer data between them. In SQL-Server, an index is a database object of tables and columns[3]. The database index is used to identify column values in specific tables because without it rapidly; the database would have to search from the first to the last row table, which would take a long time. Furthermore, if



you run a search on a table with many rows without an index, you will need a lot of RAM. As a result, this index may be compared to a book's table of contents [4].

2. Theoretical Basis

An index in a database is similar to a table of contents in a book in that it makes it simpler to discover data. Because if a database does not have an index, it will scan or search for each record in the required table [5]. Scanning or scanning a database takes a long time and slows down its speed. When an index is called in SQL, you can generate one or more columns in the index. Consequently, the index can produce one column (column index) or numerous columns (multiple-column index) from a table [6].

2.1 Column Indexes (One Column Index)

Indexing is possible for all MySQL data types. The best method to increase the efficiency of the SELECT operator is to use an index on the relevant column. The maximum index number and index length for each table are determined by the storage available (Storage Engine).

2.2 Multiple-Column Indexes

MySQL can generate composite indexes (creating an index by querying two columns of data or a table). Indexes with up to 16 columns are known as multiple-column indexes. A multiple-column index is an array of values merged with the values in the column from which the index is constructed [7].

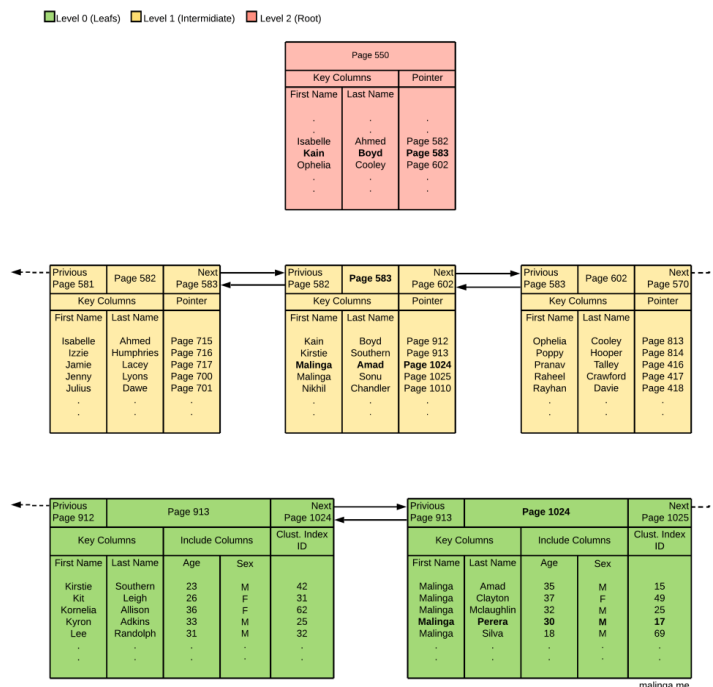


Figure 1. Structure of a Multi-column index in SQL server

MySQL uses indexes to execute the following tasks:

- a. To get the rows that meet the WHERE clause rapidly
- b. To take the rows out of the equation. MySQL will check for the index that has the fewest rows.
- c. The rows in the previously merged column must be retrieved.
- d. In some circumstances, the query can be streamlined to return values without initially entering the table.

3. Method

Experimental approaches were used in this investigation. One form of research approach is the experimental method. Researchers can use experimental approaches to modify and adjust factors and study the results. The variables in this experimental procedure are regulated so that extraneous variables that may impact the outcome are excluded[8].

The experimental approach seeks to uncover and obtain causal links by changing or manipulating one or more variables in one or more experimental groups, then comparing the findings to a control group that was not manipulated. Manipulation is systematically modifying the characteristics or values of independent variables [9]. Control is essential in experimental approaches since modification and observation without it will provide suspect data.

4. Results

Because not all data or indexes must be used, you must first build and consider the correct approach before constructing an index on a database. Creating an index may be done in a variety of ways, including:

a. Index Design

Determining the column to utilize, the kind of index to employ (for example, clustered or non-clustered), the proper index choices, and the filegroup or partitioning scheme placement are all part of index design.

b. Determining the best manufacturing method

The index is created by:

1. Using CREATE TABLE or ALTER creates a unique primary key or constraint for the column. Unless a clustered index already exists in the table or a unique non-clustered index is specified, SQL builds a unique index to enforce the uniqueness requirement of a PRIMARY KEY or a unique constraint [10]. Unless a unique clustered index is precisely defined and a table clustered index is not available, a unique clustered index is produced by default to enforce a PRIMARY KEY requirement.
2. Using the CREATE INDEX command or the New Index Dialog in SQL, create an index that is independent of constraints. The name, table index, and column index that apply are all determined. You may also provide the index and index location and the filegroup and partition scheme. If the clustered option or unique constraint is not supplied, a non-clustered, non-unique index is constructed by default. Use the WHERE clause to construct a more specialized (filtered) index [11].
3. Create index

There are no performance ramifications when an index is built on an empty table; however, performance will be affected when data is added to the table.

Indexing huge tables require careful planning to ensure that database performance is not impeded. The best method for indexing big tables is starting with a clustered index and subsequently building a non-clustered index [12].

In order to find data more quickly and efficiently, an index can be constructed in a table. Users cannot view the index; the brand solely uses it to speed up searching and queries. The syntax for creating an index on a table is as follows. Double values are permitted:

```
CREATE INDEX index_name  
ON table_name(column_name)
```

The above query will construct an index with the name of the index and in which table and column it will be generated.

The syntax for generating a unique index on a table is as follows. In this index, double values are not permitted:

```
CREATE UNIQUE INDEX  
index_name  
ON  
table_name(column_name)
```

The above query will construct an index with the name of the index and in which table and column it will be generated.

The syntax for generating a unique index on a table is as follows. In this index, double values are not permitted:

```
DROP INDEX  
index_name  
ON table_name
```

The above query will delete an index that was previously established using the DROP INDEX operator, followed by the name of the index and table to be destroyed.

You can deactivate an index that has already been generated in addition to generating and removing it. Users will be unable to access the index, clustered index, or underlying table due to this disabling. The nonclustered index retains metadata and statistical index data [13]. Only the physical data is removed from a clustered or nonclustered index when disabled.

The data remains in the table when the clustered index is disabled, but it is not available for DML operations until the index is destroyed or re-created. The ALTER INDEX REBUILD or CREATE INDEX WITH DROP EXISTING statements can re-create and reactivate the index.

c. Advantages of Using Indexed View Scenarios

1. Foreign Key Indexes

The Foreign key column is a specific circumstance that necessitates an orderly index. A foreign key was created to match a row from one table to a row from another table. If you have a parent table and wish to view the children of a row, it is critical to make sure that every time you specify a foreign key constraint, indexing is possible [14]. When removing the parent row in numerous relations, even one kind of domain, the access type becomes highly significant.

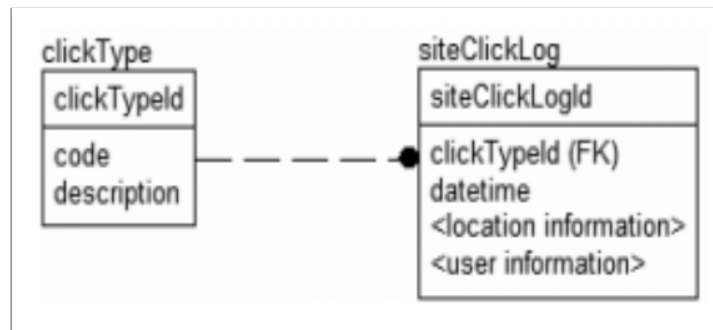


Figure 1. Relations Using Foreign Keys

2. Indexed Views

The process of indexing a view entails converting the view's virtual structure into a physical object. The result as updated data in the table is the data for performing queries with the view. Indexed views allow you to create summary tables without using any manual processes or triggers [15].

When utilizing an index view, there are two advantages. Even if SQL has not specified a specific view to use and the query does not reference a view, it will consider utilizing an indexed view whenever it performs a query.

d. Oracle Index: Database Performance

The appropriate indexing of the database, or Oracle index, is critical to database performance. Hundreds, if not thousands, of indexes may be found in most Oracle databases [14]. Index tuning and monitoring are challenging for DBAs due to the high number and complexity. Due to index distortion induced by numerous changes in the data in the indexed table, even an initially efficient index might become inefficient over time [17].

What are the different choices for utilizing Oracle indexes, and how do you manage them? The data in the related table is logically and physically independent of the index. Without impacting the base table or other indexes, the DBA can establish or eliminate indexes at any moment. All apps continue to function even if the DBA removes the index [18]. Access to previously indexed data, on the other hand, maybe slower. Because of its self-contained structure, the index takes up storage space [19].

After indexes are built, Oracle maintains and utilizes them automatically. Oracle automatically reflects data changes with no further action from the user, such as adding a new row, changing a record, or removing a row across all relevant indexes [20].

Depending on the Oracle application and the text source, Oracle Text permits three types of Oracle indexes. The DBA uses the Construct INDEX statement to create all types of Oracle index text.

5. Conclusions

Indexes in the database are highly beneficial for improving SQL speed, particularly in databases with many tables and entries in each table. Since if you do not utilize an index in the database, searching for data will take a long time because you will have to search or scan all of the table's rows. Furthermore, it will consume a significant amount of storage memory. Because Oracle provides more index types than MySQL, using SQL indexes in Oracle. Oracle, for example, may use a text index to search for terms.

However, the index's usage must be acceptable because not all databases require one, so if you are going to construct an index on the database, you will need to prepare ahead and strategize carefully. Because the index developed will make it more difficult for the user to discover the data they are seeking if it is not designed ahead of time and has a robust approach. As a result, it will provide inaccurate data for other users that want database data or information.

As a result, the index in the database is very beneficial for users who use it, but care must be taken when deciding which tables will be indexed and if the table should be indexed or not so that SQL performance is not slowed. Because too many indexes will cause a database's SQL performance to suffer.

References

- [1] S. Palanisamy, P. S.-2020 I. Conference, and undefined 2020, "A survey on RDBMS and NoSQL Databases MySQL vs MongoDB," *ieeexplore.ieee.org*, Accessed: Feb. 16, 2022. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/9104047/>
- [2] D. Immaniar, N. Azizah, D. Supriyanti, N. Septiani, and M. Hardini, "PoTS: Proof of Tunnel Signature for Certificate Based on Blockchain Technology," *International Journal of Cyber and IT Service Management*, vol. 1, no. 1 SE-Articles, pp. 101–114, May 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/28>
- [3] P. Openko, ... S. H.-... T. in I., and undefined 2019, "Problem of Choosing a DBMS in Modern Information System," *ieeexplore.ieee.org*, Accessed: Feb. 16, 2022. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/9030517/>
- [4] D. Cahyadi, A. Faturahman, H. Haryani, E. Dolan, and S. millah, "BCS : Blockchain Smart Curriculum System for Verification Student Accreditation," *International Journal of Cyber and IT Service Management*, vol. 1, no. 1 SE-Articles, pp. 65–83, Apr. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/20>
- [5] T. Ayuninggati, N. Lutfiani, and S. Millah, "CRM-Based E-Business Design (Customer Relationship Management) Case Study: Shoe Washing Service Company S-Neat-Kers," *International Journal of Cyber and IT Service Management*, vol. 1, no. 2 SE-Articles, pp. 216–225, Oct. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/58>
- [6] M. Mulyati, I. Ilamsyah, A. Aris, iketut gunawan, and M. Suzaki Zahran, "Blockchain Technology: Can Data Security Change Higher Education Much Better?," *International Journal of Cyber and IT Service Management*, vol. 1, no. 1 SE-Articles, pp. 121–135, May 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/36>
- [7] G. Maulani, G. Gunawan, L. Leli, E. Ayu Nabila, and W. Yestina Sari, "Digital Certificate Authority with Blockchain Cybersecurity in Education," *International Journal of Cyber and IT Service Management*, vol. 1, no. 1 SE-Articles, pp. 136–150, May 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/40>
- [8] T. Nurhaeni, N. Lutfiani, A. Singh, W. Febriani, and M. Hardini, "The Value of Technological Developments Based on An Islamic Perspective," *International Journal of Cyber and IT Service Management*, vol. 1, no. 1 SE-Articles, pp. 1–13, Apr. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/4>
- [9] N. P. Puji, A. Dudhat, H. Teja Sukmana, A. Mardiansyah, and M. Hardini, "Modern Technology Advances with Benefits for Humanity to Demonstrate Design with

- Conventional Sources Islamic,” *International Journal of Cyber and IT Service Management*, vol. 1, no. 1 SE-Articles, pp. 14–36, Apr. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/8>
- [10] B. Rawat and S. Purnama, “MySQL Database Management System (DBMS) On FTP Site LAPAN Bandung,” *International Journal of Cyber and IT Service Management*, vol. 1, no. 2, pp. 173–179, 2021.
- [11] P. Edastama, A. Dudhat, and G. Maulani, “Use of Data Warehouse and Data Mining for Academic Data : A Case Study at a National University,” *International Journal of Cyber and IT Service Management*, vol. 1, no. 2 SE-Articles, pp. 206–215, Oct. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/55>
- [12] P. Edastama, A. S. Bist, and A. Prambudi, “Implementation Of Data Mining On Glasses Sales Using The Apriori Algorithm,” *International Journal of Cyber and IT Service Management*, vol. 1, no. 2 SE-Articles, pp. 159–172, Oct. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/46>
- [13] Y. Durachman, A. Sean Bein, E. Purnama Harahap, T. Ramadhan, and F. Putri Oganda, “Technological and Islamic environments: Selection from Literature Review Resources,” *International Journal of Cyber and IT Service Management*, vol. 1, no. 1 SE-Articles, pp. 37–47, Apr. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/12>
- [14] D. Apriani, A. Williams, U. Rahardja, A. Khoirunisa, and S. Avionita, “The Use of Science Technology In Islamic Practices and Rules In The Past Now and The Future,” *International Journal of Cyber and IT Service Management*, vol. 1, no. 1 SE-Articles, pp. 48–64, Apr. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/16>
- [15] A. Alwiyah, S. Husin, P. Padel, M. Anggaraeni, and S. Sulistiawati, “Alignment of Science and Technology With Islamic Principles Using Quantum Theory,” *International Journal of Cyber and IT Service Management*, vol. 1, no. 1 SE-Articles, pp. 115–120, May 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/32>
- [16] T. Ramadhan, Q. Aini, S. Santoso, A. Badrianto, and R. Supriati, “Analysis of the potential context of Blockchain on the usability of Gamification with Game-Based Learning,” *International Journal of Cyber and IT Service Management*, vol. 1, no. 1 SE-Articles, pp. 84–100, Apr. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/24>
- [17] M. R. Anwar, R. Panjaitan, and R. Supriati, “Implementation Of Database Auditing By Synchronization DBMS,” *International Journal of Cyber and IT Service Management*, vol. 1, no. 2 SE-Articles, pp. 197–205, Oct. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/53>
- [18] Y. Wu, J. Yu, Y. Tian, R. Sidle, R. B.-P. of the 2019, and undefined 2019, “Designing succinct secondary indexing mechanism by exploiting column correlations,” *dl.acm.org*, p. 18, Jun. 2019, doi: 10.1145/3299869.3319861.
- [19] S. Watini, T. Nurhaeni, and L. Meria, “Development Of Village Office Service Models To Community Based On Mobile Computing,” *International Journal of Cyber and IT Service Management*, vol. 1, no. 2 SE-Articles, pp. 189–196, Oct. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/51>
- [20] W. Setyowati, R. Widayanti, and D. Supriyanti, “Implementation Of E-Business Information System In Indonesia : Prospects And Challenges,” *International Journal of Cyber and IT Service Management*, vol. 1, no. 2 SE-Articles, pp. 180–188, Oct. 2021, [Online]. Available: <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/49>