NTNU

Kunnskap for en bedre verden

TDT39 Empirical Studies of ICT

Plan for research studies

Title of the study:
Effect of Re-Optimization of MySQL-queries

Name of the student:
Olaf Rosendahl

Number of words (excluding front page and references):
1161

November, 2023

Purpose

Database queries can lead to poor database performance for various reasons. One common issue is poorly optimized queries. When queries are not efficiently written, they may not make the best use of the database's capabilities, leading to unnecessary resource consumption and slow execution times. Another significant factor is the absence of indexing. Without appropriate indexes, queries may require full table scans, which can be exceptionally slow for large datasets. Properly designed indexes can substantially enhance query performance by enabling quick data retrieval. Queries that involve joining multiple tables can also be resource-intensive, especially when the join conditions are not well-defined or when dealing with large datasets. Inefficient table join strategies can result in Cartesian products or excessive data retrieval, causing performance degradation. When the query optimizer of a database generates an estimate of the most optimal possible query plan for the query, it is largely based on cardinalities which represent the estimated number of rows a query will return. When the cardinalities are wrong or inaccurate, it can lead to a suboptimal query plan, even if well-suited indexes exist, the query is well written, or the table joins involves a small amount of tables [2].

Re-optimization has been presented by multiple research studies since the early nineties as a possible solution when cardinalities are inaccurately estimated. Since queries may be executed with strategies that are ill-suited to the actual data distribution, re-optimization can be used to make changes to the query plan when needed. Many different strategies for re-optimizing a query after execution has started have been presented, including re-planning the entire plan [4], query plan operators which can dynamically switch strategies based on actual cardinalities during execution [1] and query optimizers producing multiple plans which can be run in parallel during execution [3]. Various strategies have different ways of deciding when to make changes to the original plan, usually based on the difference between estimated and actual cardinalities [9], but all with the common goal of ensuring that queries always are executed with the most appropriate execution plans, improving efficiency and query performance.

There are, to my knowledge, no studies that test the effect re-optimization of query plans can have in the MySQL database. When looking for research that uses specifically MySQL, no studies have been found. However, re-optimization techniques have been implemented in PostgreSQL by multiple other relevant studies [6], [7]. As an open-source and free-to-use database, MySQL is widely used and among the most popular databases in the world [8], which means that even small performance improvements can have big impacts. What I hope to achieve with this research is to demonstrate the effect of query plan re-optimization in the MySQL database. More specifically I want to implement re-optimization by adding checkpoints in the query plan which stops the execution if the actual cardinalities significantly exceed the estimated cardinalities. Based on my motivations, the following research questions have been created:

1. **RQ1:** How can query performance in MySQL be improved by re-optimization, measured with testing performed using the most common test suites?
2. **RQ2:** By how much must the actual cardinalities be off, relative to the estimated cardinalities, before re-optimization is profitable in terms of performance?

Contributions

The primary contribution of this study will be centered on new and improved evidence. The study will aim to provide fresh empirical insights into the effectiveness of query plan re-optimization techniques in MySQL. The research will hopefully offer new and refined evidence through comprehensive empirical analyses, shedding light on the actual impact of these techniques in real-world scenarios.

The research will stand out by being the first study, to my knowledge, that looks at re-optimization specifically in MySQL. By conducting rigorous performance testing with real-world datasets, the research will contribute to a more substantial and nuanced understanding of the practical implications and potential of query plan re-optimization techniques in MySQL. This improved evidence will provide practitioners and researchers with a more robust comprehension of the real-world benefits and potential challenges associated with these methodologies, thereby enabling more informed and effective decisions of future directions for the improvement of MySQL.

Research Method

For this study, the research questions were defined based on my own experience with databases and the somewhat "random" drop in query performance that I have seen occur. Based on recent studies regarding re-optimization in databases, large performance improvements can be found in general, but the lack of studies specifically on MySQL gave motivation to focus on MySQL.

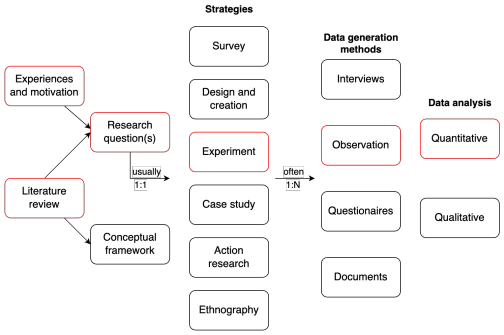

Figure 1: Research process [5]

Conducting experiments has been selected as the strategy. The goal of the research is to find how re-optimization can improve query performance in MySQL, and by running experiments and comparing the before and after measurements. Continuous experiments will also help answer RQ2. "Action and research" is another relevant strategy since it would allow implementing a solution and then testing it, but has not been selected for this study because evaluating the results is hard without measurements that experiments generate.

Throughout the experiments, I will observe and collect data in terms of how long the database needs to run queries from the most common test suite. This will then be evaluated through a quantitative analysis since that will best answer the research questions. It is natural to use the average of multiple repeated experiments to avoid random errors. For RQ2, the quantitative results can in addition be used to create a diagram showing the query performance with the x-axis as a variable for how much the actual cardinalities must differ for re-optimization to start.

Participants

I will conduct the research myself. My supervisor is Norvald Ryeng, Adjunct Associate Professor at NTNU and Software Development Director at Oracle Trondheim. There will be frequent meetings during the process. Oracle Trondheim does most of the work of MySQL and it will therefore be easy to get assistance during implementation of the re-optimization in the MySQL codebase.

Research Paradigm

The positivism paradigm, which emphasizes the objective and empirical analysis of phenomena, is the best paradigm for this study for several compelling reasons. First and foremost, it aligns with the nature of the research questions, as it seeks to uncover objective insights into a specific technical phenomenon. The use of quantitative data and measurements from experiments is particularly suited to this paradigm, allowing for precise assessment and comparison of performance metrics

with and without re-optimization. Moreover, the positivism paradigm promotes the principles of reliability and replicability, ensuring that the results can be verified and reproduced by other researchers, thereby enhancing the credibility and trustworthiness of the study's findings. By grounding the research in positivism, I can confidently address the study's objectives by providing a well-founded and objective analysis of the performance implications of re-optimization in MySQL, making it a suitable and robust choice.

Final Deliverables and Dissemination

The research will include a literature study which will be delivered in a project report, and a study of the effect of re-optimization with a selected technique in MySQL that will be delivered in a master's thesis.

References

[1] R. Borovica-Gajic, S. Idreos, A. Ailamaki, M. Zukowski and C. Fraser, 'Smooth scan: Robust access path selection without cardinality estimation', *The VLDB Journal*, vol. 27, no. 4, pp. 521–545, Aug. 2018, ISSN: 1066-8888. DOI: 10.1007/s00778-018-0507-8. [Online]. Available: https://doi.org/10.1007/s00778-018-0507-8.

[2] A. Deshpande, J. M. Hellerstein and V. Raman, 'Adaptive query processing: Why, how, when, what next', in *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, 2006, pp. 806–807.

[3] A. Dutt and J. R. Haritsa, 'Plan bouquets: A fragrant approach to robust query processing', *ACM Trans. Database Syst.*, vol. 41, no. 2, May 2016, ISSN: 0362-5915. DOI: 10.1145/2901738. [Online]. Available: https://doi.org/10.1145/2901738.

[4] N. Kabra and D. J. DeWitt, 'Efficient mid-query re-optimization of sub-optimal query execution plans', *SIGMOD Rec.*, vol. 27, no. 2, pp. 106–117, Jun. 1998, ISSN: 0163-5808. DOI: 10.1145/276305.276315. [Online]. Available: https://doi.org/10.1145/276305.276315.

[5] B. J. Oates, M. Griffiths and R. McLean, *Researching Information Systems and Computing*. SAGE, 2022.

[6] M. Perron, Z. Shang, T. Kraska and M. Stonebraker, 'How i learned to stop worrying and love re-optimization', in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, 2019, pp. 1758–1761. DOI: 10.1109/ICDE.2019.00191.

[7] S. Sikdar and C. Jermaine, 'Monsoon: Multi-step optimization and execution of queries with partially obscured predicates', in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '20, Portland, OR, USA: Association for Computing Machinery, 2020, pp. 225–240, ISBN: 9781450367356. DOI: 10.1145/3318464.3389728. [Online]. Available: https://doi.org/10.1145/3318464.3389728.

[8] SOLID IT, *DB-Engines Ranking — db-engines.com*, https://db-engines.com/en/ranking, [Accessed 19-09-2023], 2023.

[9] F. Wolf, N. May, P. R. Willems and K.-U. Sattler, 'On the calculation of optimality ranges for relational query execution plans', in *Proceedings of the 2018 International Conference on Management of Data*, ser. SIGMOD '18, Houston, TX, USA: Association for Computing Machinery, 2018, pp. 663–675, ISBN: 9781450347037. DOI: 10.1145/3183713.3183742. [Online]. Available: https://doi.org/10.1145/3183713.3183742.