

# Database query optimization term paper

[Business](#), [Management](#)



## **The query optimization problem**

Lack of precise statistical information about the database hampers greatly the optimization of queries in database systems (Raghu, & Johannes, 2003).

Good algorithms have to be applied in the process of query optimization if success is to be achieved in a greater manner (Walker, 2007).

In this paper, we define a query as a kind of language expression used to describe data elements to be retrieved in the database (Garcia-Molina, Ullhman, & Widom, 2004). Application of queries is done in many different settings. In many cases, users send direct requests to the databases in order to get the general structure of the database or even retrieve data (Cheung, 2007). Due to the limited nature of application of these queries, optimization can be done by use of programming techniques in to the system (Dayal, 2003). Some of the techniques that can be applied to optimize these common queries include use of drop down menus and other forms.

Another application of queries is during different transactions that are dynamic. These transactions change some stored values based on the applied queries (Chesnais, Gelenbe, & Mitrani, 2006). They are mostly applied internally in a DBMS for different purposes. An example is the application of the queries to access rights granted to an individual or to selectively change values of a particular row or column. In addition to therefore mentioned application of queries in transactional changes, application is also seen in the synchronization and concurrency access control of processes (Chen, & Akoka, 2006).

There are two kinds of query optimization. The first phase is query modification in which the initial query is rewritten to [provide an improved

efficiency of the provided query (Davis, & Kunii, 2005). Query modification does not at all change anything with regard to the procedure of the query process. Changes cannot happen in this phase, however the second phase which introduces the translation from a non procedural one to a procedural one. This second phase is called the query optimization level (Chiu, & Ho, 2004).

The difference between query optimization and modification is evident in the translation process. A straight forward manner is applied in the modification process without any regard for any available optional alternatives of modification process (Shneiderman, & Goodman, 2006). Query optimization on the other hand does not use a direct approach. Different alternatives are applied in the process. Exploration of different QEPS is done and the best approach chosen and implemented.

## **Optimization objectives**

The main objective of query optimization is to achieve efficiency. If maximization of output through the applied query while minimizing the input resources is achieved, then good efficiency is achieved. Procedures applied in the optimization process should be able to maximize the output and minimize the input resources (Severance, & Carlis, 2007). The response time should also be made less. In order to have efficient system, then the overall time taken to achieve a certain objective should be comparably low to the available alternatives or options. It cannot be forgotten that the total cost will also be reduced considerably by application of good optimization procedures (Rothnie, 2004). Some of the costs that will be reduced include communication cost, secondary storage access cost and computational cost.

Communication costs refer to the costs incurred when transmitting data from the storage sites to presentation sites. These costs can involve the costs associated with the transmission lines and the time overhead taken to transmit these resources. Secondary storage access costs refer to those expenses incurred while loading data to the main memory from the storage devices while storage costs imply those expenses associated with data occupation of the secondary storage and memory buffers (Raghu, & Johannes, 2003). Finally the computation cost which can be incurred stems from the costs incurred during the computation and presentation of data. The structure of any query optimization and the algorithm involved in it, clearly define how these costs thread off will be influenced and affected.

## **The principle of optimization**

There are basic building blocks that specify any algorithm especially at the optimization algorithms. The basic three principles identified in this paper are;

1. The QEP generation
2. The search strategy
3. The cost function

The determination of the source language and the transformations done are specified by the QEP creation. QEP creation involves validation which is done based on the language execution environment. The structure of the database, define the application and type of QEP and the access operators that can be applied (Selinger, et. al. 2009).

Search strategy defines the order in which the target data specified by the

query is to be found. Different choices are applied with each choice offering a unique way of finding the target values. The best is chosen and used in the process. Many search strategies can be applied in the process of query optimization in databases (Warren, 2005).

Cost functions are used in deciding the best available plan for execution based on the different QEPs available. Generally cost function use resources that the QEP plan is likely to consume to decide if it is the best plan for implementation or not. Consideration is placed on aspects like CPU consumption and input and output costs, sizes of tables and load distribution.

## **Reference**

Chen, P. s., & Akoka, J. (2006). Optimal design of distributed information systems. *IEEE trans. compute. c-29*, 12, 1068-1080.

Chesnais, A., Gelenbe, E., and Mitrani, I. (2006). On the modeling of parallel access to shared data. *Communication. ACM* 26, 3 (Mar.), 196-202.

Cheung, T.-Y. (2007). A method for equijoin queries in distributed relational databases. *IEEE Trans. Comput. C-31*, 8, 746-751.

Chiu, D. M., and Ho, Y. C. (2004). A methodology for interpreting tree queries into optimal semijoin expressions. In *Proceedings of the ACM SIGMOD International Conference on Management of Data* (Santa Monica, Calif., May 14-16). ACM, New York, pp. 169-178.

Davis, L. S., AND Kunii, T. L. (2005). Pattern databases. In *Data Base Design Techniques*

II, S. B. Yao and T. L. Kunii, Eds. Springer, New York, pp. 357-399.

Dayal, U. (2003). Evaluating queries with quantities: A horticultural approach. In

Proceedings of the ACM Symposium on Principles of Database Systems {Atlanta, Ga.}.

ACM, New York, pp. 125-136.

Epstein, A., Stonebraker, M., and Wong, E. (2008). Distributed query processing in a relational data base system. In Proceedings of the ACM-SIGMOD International

Conference on Management of Data. ACM, New York, pp. 169-180.

Garcia-Molina, H., Ullhman, J., and, Widom, J., (2004) Database Systems-The Complete

Book, Pearson Education,

Raghu R., and Johannes G., (2003) Database Management Systems, 3rdEdition, McGraw Hill,

Rothnie, J. B. (2004). An approach to implementing a relational data management system. In Proceedings of the ACM-SIGMOD Workshop on Data Description, Access, and Control (Ann Arbor, Mich., May 1-3). ACM, New York, pp. 277-294.

Selinger, et. al. (2009). Access path selection in a relational database management system. In Proceedings of the ACM-SIGMOD International Conference on Management of Data (Boston, Mass., May 30-June 1). ACM, New York, pp. 23-34.

Severance, D. G., and Carlis, J. V. (2007). A practical approach to selecting record access paths. ACM Comput. Surv. 9, 4 (Dec.) 259-272.

<https://assignbuster.com/database-query-optimization-term-paper/>

Shneiderman, B., and Goodman, V. (2006) Batched searching of sequential and tree structured files. *ACM Trans. Database Syst.* 1, 3 (Sept.), 268-275.

Walker, A. (2007). On retrieval from a small version of a large data base. In *Proceedings of the 6th International Conference on Very Large Data Bases* (Montreal, Oct. 1-3). IEEE, New York, pp. 47-54.

Warren, D. H. D. (2005). Efficient processing of interactive relational database queries expressed in logic. In *Proceedings of the 9th International Conference on Very Large Data Bases* (Cannes, Sept. 9-11). IEEE, New York, pp. 272-281.