ISSN:0975-9646
Vishal Sharma et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (4) , 2015, 4094-4095

A Survey: Query Optimization

Vishal Sharma[1], Mr. Lovenish Sharma[2]

Shri Vaishnav Institute of Technology and Science Indore, India[1]
Asst. Prof. Shri Vaishnav Institute of Technology and Science Indore, India[2]

Abstract-Almost all applications use database and Information Retrieval system for storing and retrieving operational data i.e. query processing. But in current time most of the users on internet are not interested in huge amount of results. They look for specific type of information with in short time that can be retrieving after resolving cardinality estimation. The new challenge requires several changes that vary from introducing new query language constructs to enhancing the query processing and optimization engines with new query operators. This paper presented a survey of query optimization.

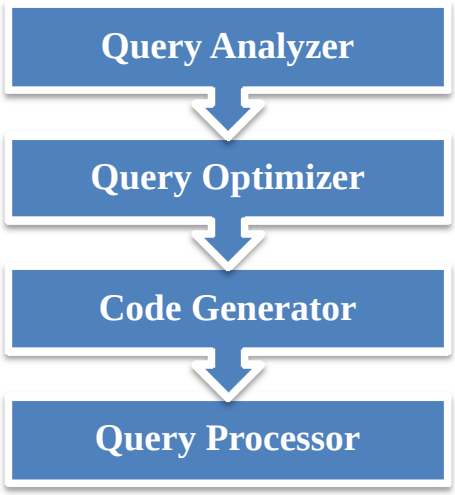
Keywords-Query Optimization, Query Analyzer, Rank Aware etc.

I. INTRODUCTION

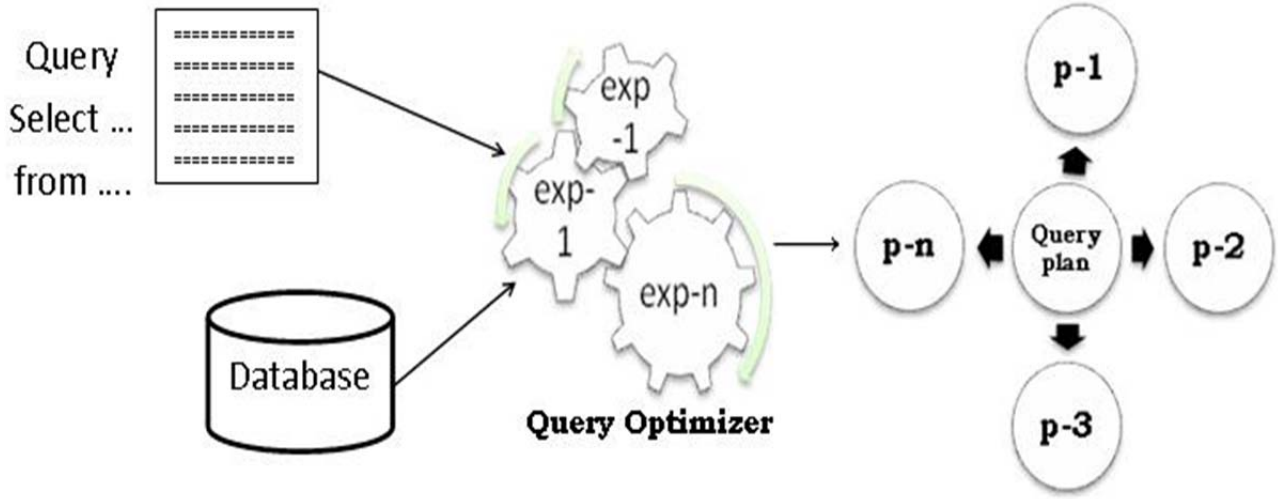
The query optimizer is the component of a database management system to optimize the query [2]. Query optimization is a function of many relational database management systems in which multiple query plans for satisfying a queries are examined and a good query plan is identified. Query optimizer is a selecting an efficient execution strategy for processing a query. The query optimization attempt to minimize the use of certain resources like I/O by selecting the set best query access plan. Query optimization starts during the validation phase by the system to validate whether the user has appropriate privileges. Now an access plan is generated to perform the query. The access plan is the put into effect with the execution plan of generated during query processing phases [1].

II. QUERY PROCESSING

Query processing is the process of translating a query expressed in a high-level language such as SQL into low-level data manipulation operations. Query optimization refers to the process by which the best execution strategy for a given query is found from a set of alternatives. Typically query processing involves many steps.



- Query Analyzer check the validity of the query whether it is valid or not.
- Query Optimizer is process of selecting a suitable execution strategy for processing the query.
- Code Generator generates the code for various plan of query optimizer.
- Query Processor process the query.



III. LITERATURE REVIEW

Several solutions have been proposed in the literature for the Query Optimization. Some of them are as follows:

Davide Martinenghi[3] In this paper we propose an execution strategy for retrieving the top-k combinations that can be formed by joining the results of heterogeneous search engines. We optimize such a strategy with respect to an additive cost model that considers both sorted access and random access. To this end, we introduce a statistical framework to characterize the number of combinations and the number of random accesses.

Sumita Gupta[4] Generally, people rely on digital library search systems to explore the web for scientific information. In such a scenario, many advanced web searching and mining techniques have been employed in order to find only relevant information. Page ranking algorithms play a significant role in ranking scientific information, so that the user can retrieve the most relevant information in the top of the result list.

Hui Zhao[5] we presented on a global index based optimization strategy for range query and analysis, and do some tests to evaluate the correctness and efficiency in the end , the strategy is first checking whether user requests can be optimized by using the global index knowledge. If it does we use a customized tasks division method which is based on global index structure and query condition, result in reducing task number and input data, through which we not only gain better response time, but also optimize resource usage and task scheduling overhead. Finally, we experimentally verified the feasibility of this technology.

Christian Politz[6] We introduced dependencies of documents and rankers. The dependencies were used to estimate the use we gain when using postings of certain terms and applying specific rankers. We defined an optimization task of which the solution results in the optimal documents (resp. postings) to load and rankers to apply when facing an arbitrary budget jointly on loading and computational costs.

Saranya.R.S[7] It presented above algorithms for processing top-k spatial preference queries. The branch-and-bound (BB) algorithm derives upper bound scores for non-leaf entries in the object tree, and prunes those that cannot lead to better results. The algorithm BB* is a variant of BB that utilizes an optimized method for computing the scores of objects (and upper bound scores of non leaf entries). The algorithm performs a multi way join on feature trees to obtain qualified combinations of feature points and then search for their relevant objects in the object tree.

IV. CONCLUSION

To a large extent, the success of a DBMS lies in the quality, functionality, and sophistication of its query optimizer, since that determines much of the system's performance. In this paper, we have given a bird's eye view of query optimization. And a survey on various technique used by query optimization on map reduced, spatial data and so on. We proposed a generalized rank-aware optimization framework that define ranking as a supplementary plan enumeration dimension beyond enumerating joins and allowed for generating the full space of rank-aware query evaluation plans.

REFERENCES

[1] Sambit Kumar Mishra and Srikanta Pattnaik, “Evaluation of Cost of Plans in Multiple Dependent Queries Execution Using G.A. Techniques”, IACSIT International Journal of Engineering and Technology, Vol.3, No.2, April 2011.

[2] Pawan Meena, Arun Jhapate & Parmalik Kumar,"Framework for Query Optimization", (IJCSIS) International Journal of Computer Science and Information Security Vol. 9, No. 10, October 2011.

[3] Davide Martinenghi and Marco Tagliasacchi,” Cost-Aware Rank Join with Random and Sorted Access”, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 24, NO. 12, DECEMBER 2012.

[4] Sumita Gupta, Neelam Duhan, Poonam Bansal and Jigyasa Sidhu,” Page Ranking Algorithms in Online Digital Libraries: A Survey”, 978-1-4799-6896-1/14/$31.00 ©2014 IEEE.

[5] Hui Zhao, Shuqiang Yang, Zhikun Chen, Songcang Jin, Hong Yin and Long Li,” MapReduce model-based optimization of range queries”, 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2012).

[6] Christian Politz and Ralf Schenkel,” Ranking under tight budgets”, 2012 23rd International Workshop on Database and Expert Sytems Applications.

[7] Saranya.R.S and Saraswathi. S M.E.,” Ranking Spatial Data by Quality Preferences”, IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM -2012) March 30, 31, 2012.