

Selection and Maintenance of Materialized View Using Genetic Algorithm

Ramesh s. Gawali¹, Prof. Mrunali Vaidya²

Department of Computer Science and Engg, BIT, Bahmni, India ¹

Lecturer, Department of Computer Science and Engg, Bahmni, India ²

Abstract: Data warehouse is a repository of large amount of data collected from multiple heterogeneous and distributed data sources. Quick response time and accuracy are the key points for success of any database. Performance of query can be improved by different approaches like query optimization, use of proper data structure etc. But leaving all these alternatives we are planning to use materialized view approach. Materialized views (MV) are precomputed result of queries. Use of materialized view avoids the visit of data source resulting in the saving the time of query response. In our work we are going to materialize the queries based on their threshold value of accessing the query.

Keywords: Threshold value, materialized view, data warehouse, query optimization.

I. INTRODUCTION

Data warehouse is the huge storage of information. This information is collected from various sources of information. These sources can be autonomous or distributed systems.

The primary goal of this research is to develop a system which will reduce the execution time for query i.e, getting the response from database quickly. We are going to take the help of materialized view. To avoid accessing the original data sources and increase the efficiency of the queries fired on data warehouse, some results in the query processing are stored in data warehouse. These results in data warehouse are known as materialized view. On abstract level data warehouse can be seen as collection of materialized view for quick access of data.

II. LITERATURE REVIEW.

Many researches are working to improve the performance of query. Some the work is illustrated as follows.

Dr. T.Nalini et al. [1] proposes an IM-LXI index for incremental maintenance of materialized view selection of materialized views so that query evaluation costs can be optimized as well as view maintenance and view storage was addressed in this piece of work

Ashadevi, B and Balasubramanian[2] proposed framework for selecting views to materialize which takes in to account all the cost metrics associated with the materialized views selection, including query processing frequencies, base relation, update frequencies, query access costs, view maintenance costs and the system's storage space constraints and then selects the most cost effective views to materialize and thus optimizes the maintenance storage, and query processing cost. This piece of work also addressed the preservation of existing materialized view.

Himanshu Gupta and Inderpal SinghMumick [3] developed a greedy algorithm to minimize the maintenance cost and storage constraint in the selection of materialized views for data

warehouse. In this paper view selection under disk space & maintenance cost constraints are addressed.

Yang, J et al.[4] proposed a heuristics algorithm based on individual optimum query plans. Framework is based on specification of multiple views processing plan (MVPP), which is used to present the problem formally.

Harinarayan et al. [5] proposed a greedy algorithm for the materialized views selection so that query evaluation costs can be optimized in the special case of "data cubes". This paper provides good trade-offs between the space used and the average time to answer query. Here, the costs for view maintenance and storage were not addressed in this piece of work.

Amit Shukla et al. [6] proposed a very simple and fast heuristic algorithm, PBS, to select aggregates for pre computation. PBS algorithm runs faster than BPUS, and is fast enough to make the exploration of the time-space trade -off feasible during system configuration. .

Y.D.Choudhari et al.[7] proposed a novel CBFSMV algorithm is proposed for selection of materialized view using query clustering strategy that reduces the execution time as compared to response time for actual database.

A greedy algorithm is used to incorporate the maintenance cost and storage constraint in the selection of data warehouse Materialized View. It reduces complexity of algorithm compare to previous algorithm [9].

III. PROPOSED SYSTEM

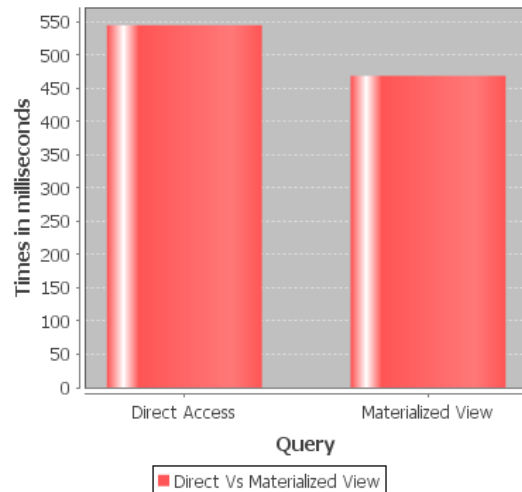
The architecture of proposed system is shown in figure.1 Data sources of information are distributed .When query is fired by user the data is search in materialized view first. This materialized view contains the result of queries which are fired frequently by the user. Thus, the time of visiting the original base table get saved and we get the response of query earlier as compared to conventional way of query processing.

original data source. The time required to search the data using MV get reduced as visiting to base table is avoided.

V.EXPECTED RESULT

As per the work , we are expecting the following result, When we execute the query without MV and with MV. The graph is plotted query versus time needed to execute that query.

Query Performance Measure Direct Access Vs. Materialized View Access



When query is fired on the database without the implementation of MV it take more time as compared to the time required to fetch the record when we implement the MV.

VI.CONCLUSION

Thus, implementation of materialized reduces the access time required to get the data from the database. Use of materialized view increases performance of system approximately 10-12%.

Implementation of materialized view is one of the best option to improve performance of data bases.

REFERENCES

- [1] Dr.T.Nalini, Dr.A.Kumaravel , Dr.K.Rangarajan,"A Novel Algorithm with IM-LSI Index For Incremental Maintenance of Materialized View" JCS&T Vol. 12 No. 1 April 2012
- [2] B.Ashadevi, R.Balasubramanian," Cost Effective Approach for Materialized Views Selection in Data Warehousing Environment", IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.10, October 2008
- [3] Gupta, H. & Mumick, I., Selection of Views to Materialize in a Data Warehouse. IEEE Transactions on Knowledge and Data Engineering, 17(1), 24-43, 2005.
- [4] Yang, J., Karlapalem. K., and Li. Q. (1997). A framework for designing materialized views in a data warehousing environment. Proceedings of the Seventieth IEEE International Conference on Distributed Computing systems, USA, pp:458.

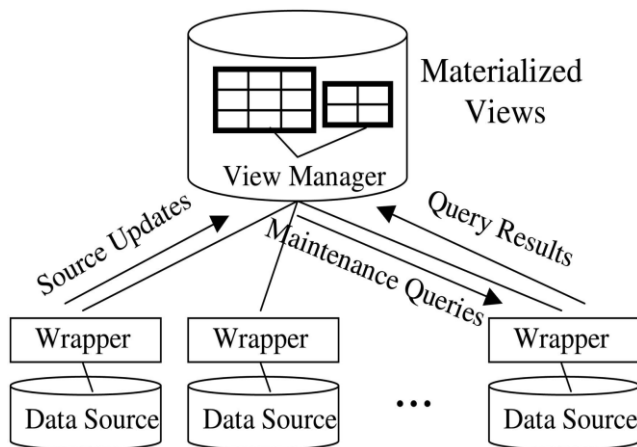


Figure1. Implementation of MV for query processing

IV IMPLEMENTATION FRAMEWORK

After studying different research paper and Analyzing different algorithm to select MV. By keeping in mind the goal of our work to enhance performance of query in data warehouse using MV. We have divided the work into four modules, as shown below.

- Creation of data warehouse
- Searching the data without the use of MV
- Materialized view creation
- Search records in the presence of MV

a.Creation of data warehouse

In this phase we collect raw data i.e.,dat file is used. This raw data is loaded in temp.table. After cleaning the data , data warehouse is created. Now data is available by executing query on them. Once the data warehouse is created it is used to create cluster data warehouse .These clusters are distributed on different systems. Clusters are formed on attributes.

b. Searching the data without the use of MV

We fire the queries on data warehouse and record the time needed to search the record based on attribute.

c. Materialized view creation

In this phase MV is created by firing the queries .Each of search query is added in .txt file as per that MV is created. The size of MV depends on what type of database is used. It works like cache memory

Search records in the presence of MV

User fires the query. If data is available in MV user does not visit the base table if miss occurs then record is search in

[5] V. Harinarayan, A. Rajaraman, and J. Ullman. "Implementing data cubes efficiently". Proceedings of ACM SIGMOD 1996 International Conference on Management of Data, Montreal, Canada, pages 205--216, 1996.

[6] A. Shukla, P. Deshpande, and J. F. Naughton, "Materialized view selection for the multidimensional datasets," in Proc. 24th Int. Conf. Very Large Data Bases, 1998, pp. 488–499.

[7] Y.D. Choudhari and Dr. S. K. Shrivastava, "Cluster Based Approach for Selection of Materialized Views", International Journal of Advanced Research in Computer Science and Software Engineering , Volume 2, Issue 7, July 2012