

Optimization Design of Query Processing Performance Using Appropriate Materialized View Selection & Preservation

S D Choudhari¹, Mrs Rashi Agrawal²

Associate Professor, CSE, SBITM, Betul, India¹

HOD, IT, UIET, Kanpur, India²

Abstract: In this paper, we describe the effective design of materialized view selection and preservation in a data warehousing system. This design implementation aims to aid users in retrieving data effectively for business analysis. The skeleton design of this data warehousing system employs the dimensional modeling concepts of snowflake as well as star schemes. Here, some of frequently accessed queries are stored in various user files on which we apply materialized view selection process to create materialized views in order to minimize the query processing cost. A cost analysis model was developed to enable the estimate the total cost and benefit involved in selecting each materialized view. For effective materialized views selection and preservation methodology, WMVS and PMV algorithms has been implemented.

This algorithm takes into account an effective cost variables associated with the materialized views Selection and preservation method which includes query access frequencies, materialized view access frequencies, query processing costs, materialized view access cost ,query storage cost , materialized view storage cost and the availability of the system's storage. The algorithm has been applied to dummy tables containing student information to create cost effective set of materialized summary views, , thereby resulting in an efficient data warehousing system where storage and query processing of the system is optimized.

Keywords: Materialized View, View-selection, Access Frequency, storage cost, Materialized view preservation, query processing

I. INTRODUCTION

A data warehouse is a repository that stores a large volume of extracted and summarized data for On-Line Analytical processing and decision support systems [10]. To reduce the cost of executing join queries in a data warehousing environment, frequently used join queries are often pre-computed and materialized into physical summary views so that future queries can utilize them directly. Without a doubt, materializing these physical summary views can minimize query response time. On the other hand, if the source data changes frequently, keeping these materialized views updated will certainly incur a high maintenance cost. In addition, for a system with limited storage space and/or with thousands of virtual summary views, we may be able to materialize only a small fraction of the views and preserve the created materialized view. Therefore, different parameters used to select and preserve materialized view which includes query access frequencies, materialized view access frequencies, query processing costs, materialized view access cost ,query storage cost , materialized view storage cost and the availability of the system's storage.

Materialized views defined over distributed data sources are significant for many applications to ensure high Availability, efficient access and reliable performance. This work emphasizes an efficient optimization of query processing with the help of materialized view over the data warehousing environment.

There are many advantages of prominent materialized views such as

- It decreased CPU consumption
- Obviously it provides faster response times
- It required less physical reads (Base table read)
- Less writes
- Materialized Views offer us elasticity of basing a view on Primary key
- Users, Applications, Developers and others can take benefit of the fact that the answer has been already stored for them.
- In a read-only / read-intensive environment will provide reduced query response time and reduced resources needed to actually process the queries.

This paper is organized as follows. We describe a related work of materialized view selection and materialized view preservation in section 2, Materialized Views Selection and preservation framework implementation details is explaining in section 3. In section 4, we shown experimental result, and its discussion, in section 5, we concluded the paper and section 6 is used to provide the references from where we pick up the idea[1].

II. RELATED WORK

The difficulty of finding appropriate summary views to materialize for answering repeated queries has been studied under the name of materialized view selection methodology. Further created materialized views are needed to be preserve according to their access frequency

and storage which has been studied under the name materialized view preservation. The major task is to maintain the created materialized view whenever base table information changes. All these methodologies are studied by various researchers and provide the well suited solution according to the environment in which materialized view are created, preserve and maintain.

The various researches proposed as well as implementation work on materialized view selection, preservation and maintenance are described below.

Dr. T.Nalini et al. proposes an cost effective algorithm for the selection and maintenance of materialized views so that query evaluation costs can be optimized as well as storage cost was evaluated in this piece of work. [1]

Ashadevi, B and Balasubramanian developed framework for materialize view selection problem, which takes into account all the major 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 efficient views to materialize and thus optimizes the maintenance, storage and query processing cost.[2]

Himanshu Gupta and Inderpal SinghMumick developed an algorithm to integrate the maintenance cost and storage constraint in the selection of materialized views for data warehouse environment[3].

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

Harinarayan et al. developed an algorithm for the materialized views selection so that query processing cost can be optimized in the unique cases of "data cubes". This paper provides good trade-offs between the space used by the data cubes and the average time to answer query. Here, the costs for view maintenance and storage were not addressed in this piece of work.[5]

Amit Shukla et al. developed 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 [6]

Wang, X et al. proposed view maintenance techniques which are classified into four major categories : self maintainable recomputation, not self-maintainable recomputation, self maintainable incremental maintenance and not self maintainable incremental maintenance. Self-maintainable Incremental maintenance performs the best in terms of both storage and number of rows accessed.[7]

III. IMPLEMENTATION FRAMEWORK FOR MATERIALIZED VIEW SELECTION AND PRESERVATION

This section is used to explain the design and implementation framework of the materialized view selection and preservation. The physically created summary views are very beneficial for the users to quickly

get the search results of frequent queries. The ultimate motive behind the proposed materialized view selection and preservation framework is to materialize the query having high frequency, high query processing time and low storage requirement of query.

The developed framework is applied on data warehouse system DWS. Let ASV be the set of all student views grouped by various dimension key attributes. Based on weight based approach we develop an algorithm for determining the optimal set of materialized views OMV, which is a subset of ASV, such that the materialized view selection cost can be minimized.

The first step is to generated huge number of dummy records for the data warehouse system and for that automatic record creator software is developed. This required the number of records that we want to insert into data warehouse system. After the completion of first step next step is to select the most promising materialized summary views out of bunch of views and for that following algorithm is used.

A. Algorithm for materialized summary view selection

- Find frequency of all the summary views in the data warehouse
- Sort frequency of all summary views in descending order
- Find storage cost of all the summary views in the data warehouse
- Sort storage cost of all summary views in ascending order
- Find processing time of all the summary views in the data warehouse
- Sort frequency of all summary views in ascending order
- Specify the threshold value for each sorted list containing summary view frequency, storage cost and processing time information.
- Select sorted summary views from top to bottom with specified threshold value.
- Calculate the materialized view creation cost of selected summary views using following formula $MV_{CC} = I_{W1} * \text{selected summary view frequency} + I_{W2} * \text{selected summary view processing time} + I_{W3} * (1 - \text{selected summary view storage cost})$ where I_{W1} , I_{W2} and I_{W3} are the impact weight specified by the materialized view selection analyzer.
- Specify the threshold value T for selection of materialized view.
- Create materialized summary views having materialized view creation cost (MV_{CC}) is greater than threshold value T.

Thus, the above materialized view selection algorithm can be used to achieved the desired multi-objective.

B. Algorithm for materialized summary view preservation

- Find frequency of all materialized summary views in the data warehouse
- Sort frequency of all materialized summary views in ascending order

- Find storage cost of all materialized summary views in the data warehouse
- Sort storage cost of all summary views in descending order
- Specify the threshold value for each sorted list containing materialized summary view frequency and storage cost
- Select sorted materialized summary views from top to bottom with specified threshold value.
- Calculate the materialized view preservation cost of selected materialized summary views using following formula $MV_{PC} = I_{W1} * \text{selected materialized summary view frequency} + I_{W2} * (1 - \text{selected summary view storage cost})$ where I_{W1} and I_{W2} are the impact weight specified by the materialized view preservation analyzer.
- Specify the preservation threshold value P_T for preservation of materialized view.
- Delete only those materialized summary views having materialized view preservation cost (MV_{PC}) is lesser than threshold value P_T .

IV. EXPERIMENTAL RESULT AND DISCUSSION

The section elaborates the running experiment results and their discussions that are carried out using dummy database schema by applying materialized summary view selection and materialized summary view preservation algorithm. The various typical user query views are shown below with its view frequency, processing time, and storage space along with materialized view creation cost which is calculated by finding frequency cost, processing cost and storage cost and then applying materialized view selection algorithm with impact weight 0.5, 0.3 and 0.2 respectively. Thereafter specific materialized view selection threshold is provided by the analyzer to create useful materialized views.

Query View	View Frequency	View Processing Time(ms)	Storage Space	Materialized View Creation Cost
QV1	1	60	200	0.52456
QV2	2	16	27247	0.41274
QV3	4	80	6700	0.7544
QV4	5	70	88	0.9623
QV5	3	31	92000	0.4161

Table 1 Shows Materialized View Selection Information

After finding the materialized view creation cost the next step is to identify the most promising views that need to be materialized for quick query processing which are shown in Table 2 where the specified materialized view threshold value is 0.75.

Table 2 shows only those views which satisfy the multiple purpose so here materialized view selection algorithm selecting only two views having selection cost is greater than the minimum materialized view selection threshold value from the bunch of views.

Query View	Materialized View Creation Cost
QV3	0.7544
QV4	0.9623

Table2 Show selected views out of bunch of views

Table 3 shows analysis of execution time of the query using specified materialized view selection framework as well as execution time of the query if it is executed on view of database without materialized view selection framework.

Query View	View Processing Time(ms)	Materialized View	Materialized View Processing Time(ms)
QV3	80	MV3	40
QV4	70	MV4	30

Table: 3 Comparison of view and materialized view processing time

V. CONCLUSION

Materialized summary view store the precomputed result of the query which is used to improve query performance cost by minimizing query processing time. But to create all materialized summary view is next to impossible due to huge materialized view storage cost and duplication of unnecessary base table data. Therefore to select the set of most prominent materialized summary views is essential, so that user query performance increases and storage cost for storing materialized summary view decreases significantly.

This paper gives the idea regarding how to select a most important materialized view with the help of various major parameters like: frequency of summary views, processing cost of summary view and storage space. We have implemented the above design algorithm that determines which views are more valuable for the creation of materialized view so as to achieve the good query performance.

For experimentation, the design framework is executed on the dummy data warehouse model using list of summary views, to find the efficiency of the implemented approach in selection of materialized view. For future research in this area could focus on materialized view maintenance and validating this model against some real-world data warehouse.

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 Centric. 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.
- [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] Wang, X., Gruenwalda. L., and Zhu.G. (2004). A performance analysis of view maintenance techniques for data centrics. Data centric knowledge, pp:1-41.
 - [8] Mr. P. P. Karde, Dr. V. M. Thakare. "Selection & Maintenance of Materialized View and It's Application for Fast Query Processing: A Survey". Proceedings of International Journal of Computer Science & Engineering Survey (IJCSSES) Vol.1, No.2, November 2010
 - [9] Abdulaziz S. Almazyad, Mohammad Khubeb Siddiqui. "Incremental View Maintenance: An Algorithmic Approach". Proceedings of International Journal of Electrical & Computer Sciences IJECS-IJENS Vol: 10 No: 03
 - [10] S. Chaudhuri and U. Dayal. "An Overview of Data Warehousing and OLAP Technology". SIGMOD Record, 26(1):65-74, 1997.
 - [11] Mahip Bartere et al," Optimization of Query Processing Time Base on Materialized Sample View" (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 2 (3) , 2011, 1222-1228
 - [12] K. Aouiche, P. Jouve, and J. Darmont. Clustering-based materialized view selection in data warehouses. In ADBIS'06, volume 4152 of LNCS, pages 81–95, 2006
 - [13] Y. Zhuge, H. Garcia-Molina, J. Hammer, and J. Widom, "View Maintenance in a Warehousing Environment." In Proceedings of the ACM SIGMOD Conference, San Jose California, May 1995.
 - [14] Y. Zhuge, H. Garcia-Molina, J. Hammer, and J. Widom, "View Maintenance in a Warehousing Environment." In Proceedings of the ACM SIGMOD Conference, San Jose California, May 1995