A comparative study analysis of materialized view for selection cost

[1]T.Nalini, , [2]Dr. A.Kumaravel, [3]Dr.K.Rangarajan

Dept of CSE, Bharath University,
173, Agaram Road, Selaiyur, Chennai – 600 073, India.
Email-id: nalinicha2002@yahoo.co.in

Abstract

Materialized view selection is one of the most crucial techniques to design data warehouse in an optimal manner. Selecting views to materialize for the purpose of supporting the decision making efficiently is one of the most significant decisions in designing Data Warehouse. Selecting a set of derived views to materialize which minimizes the sum of total query response time & maintenance of the selected views is defined as view selection problem. Selecting a suitable set of views that minimizes the total cost associated with the materialized views is the key objective of data warehousing. In this paper we compare the various research works on several parameters for controlling the selection process and also we compare time , query frequency and spatial cost.

Keywords :

materialization view, data warehousing, selection cost, I-mine item set index, FP growth

I.INTRODUCTION

Data warehouse (DW) can be defined as subject-oriented, integrated, nonvolatile, and time-variant collection of data in support of management's decision [1]. It can bring together selected data from multiple database or other information sources into a single repository [2]. To avoid accessing from base table and increase the speed of queries posed to a DW, we can use some intermediate results from the query processing stored in the DW called materialized views. Therefore, materialized view selection involved query processing cost and materialized view maintenance cost. Selecting views to materialize for the purpose of supporting the decision making efficiently is one of the most significant decisions in designing Data Warehouse [3]. Selecting a set of derived views to materialize which minimizes the sum of total query response time & maintenance of the selected views is defined as

view selection problem. Therefore, to select an appropriate set of a view is the major target that diminishes the entire query response time and also maintains the selected views. So, many literatures try to make the sum of that cost minimal.[1-4]

 Materialized views act as a data cache that gather information from distributed databases and support faster and reliable availability of already computed intermediate result sets (i.e. responses to queries). Data sources in current scenario are becoming quite vast and dynamic in nature i.e. they change rapidly. Consequently, frequency of deletion, addition and update operations on the base relations rises unexpectedly. Whenever the underlying base relation is modified the corresponding materialized view also evolves in reaction to those changes so that it can present quality data at the view level. Hence, we need certain techniques to deal with the

problem of keeping a materialized view up-to date in order to propagate the changes from remote data source to the destined materialized view in the warehouse. These techniques can be broadly classified as- *view selection, view maintenance, view synchronization* and lastly, *view adaptation. In this we are discuss only view selection.*

The layout of the paper is as follows. In section 2, we address the above mentioned techniques and also give a brief on the literatures being reviewed for the same. Section 3, presents a comparative study of the various research works explored in the previous section. Lastly, we conclude in section 4 and section 5 will provide the references.

2. View Selection

The most important issue while designing a data warehouse is to identify and store the most appropriate set of materialized views in the warehouse. This is to optimize two costs included in materialization of views: namely, the query processing cost and materialized view maintenance cost. Materialization of all possible views is not recommended due to memory space and time constraints [6]. The prime aim of view selection problem is to minimize either one of the constraints or a cost function as shown below:
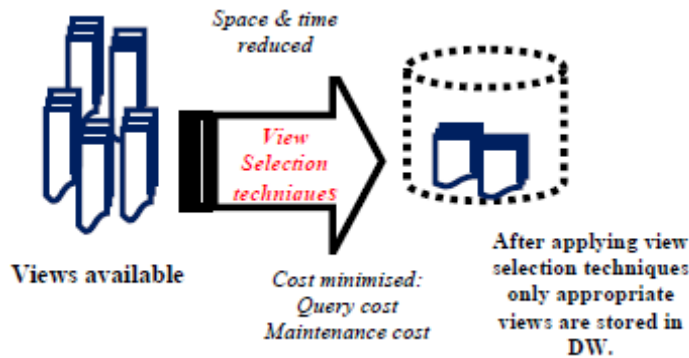

Fig 1. Space & time reduction scheme

Hence, view selection problem is formally defined as a process of identifying and selecting a group of materialized views that are most closely-associated to user-defined requirements in the form of queries in order to minimize the query response time, maintenance cost and query processing time under certain resource constraints [6].

2.1 Previous Work

In this paper**[5]** implements dynamic adjustment for static materialized views selection algorithm according to CVLC and IGA, that is ,   EMVSDIA Algorithm. The algorithm has been proved in reducing search space and time consumption by the experiment. Most of all, because the algorithm considers materialized views mutual relations in influencing view benefit. Consequently, the algorithm can be dynamically adjusted online and obtains anticipative purpose. Candidate view algorithm(CVLV) in mutil-dimensional data grid diagram is pruned and those views which have no effects on view overall cost decrease are excluded. So algorithm search space is reduced accordingly and algorithm implementation efficiency and speed is boosted. In IGA, view query and maintenance cost are considered in benefit definition. But IGA has a main defect, that is, when a view is selected , other view benefit may attenuate. The paper will overcome high complexity and frequent vibration of static materialize view

selection algorithms.EMVSDIA is designed based on CVLVand IGA,which could run and self-adjust on-line. And a lot of experiments test the excellent performance.

This paper makes a research on materialized view selection.Many static selection algorithms have many defects.Such as time complexity non-running on-line.Although FPUS makes some improvement on some aspects and could run on-line,it did not take into account dependence relationship among views .EMVSDIA overcomes the above shortcomings. Experiment demonstrates EMVSDIA has excellent implementing efficiency and reach to expectant effects.

In this paper[**6**], they develop algorithms to select a set of views to materialize in data warehouse in order to minimize the total view maintenance cost under the constraint of a given query response time. They call it query cost view selection problem (QC_VSP). In this paper, First, they propose query cost view selection problem model. Second, they give three algorithms for QC_VSP; we give view_node_matrix in order to solve it. Third, experiment simulation is adopted. The results show that our algorithm works better in practical cases. They implemented their algorithms and a performance study of the algorithms shows that the proposed algorithm delivers an optimal solution. Finally, they discuss the observed behavior of the algorithms. They also identify some important issues for future investigations.

They make use of the advantage of combining the genetic algorithm with the simulated annealing algorithm and solve QC_VSP problem in the paper. The SAGA_VSP algorithm proposed can produce better solution. The approximate optimization solution produced by combining algorithms is stable.

They could attain the conclusion from the experiments

(1)So as to QC_VSP we can solve problems applying for random algorithm. In the paper, They adopt genetic algorithm and the synthesis algorithm which is composed by genetic algorithm and simulated annealing algorithm and we get a approximate optimum materialized views .

(2)Through the simulation experiments, conclusion can be attained random algorithm is better than traditional greedy algorithm in VSP , whatever in the quality of solution and spatial cost . The solutions of synthesis algorithm are better than that of single random algorithm .

(3)They can adjust parameter to obtain balance between the quality and performing time of solution in the random method. This method is suitable to solve the large-scale problems. It will be valuable tool in data warehouse exploitation to perform materialized view selection in the random algorithm.

 In this paper **[7]** new algorithms are proposed for selectiong materialized views in a data warehouse. Two targets of research are considered. The first target is to propose an approach to sove the problem considering both multi-query optimization, and the maintenance process optimization. The other target considers using a simple search strategy that reduces the search space for the view selection, problem, and reduces the complexity to a linear instead of quadratic one.

Two approaches are being proposed in this work. Both approaches consider a cost model that combines query and maintenance costs of views. The multi-query optimization is achieved in both approaches by using an MVPP as a search space.The first approach is constructed on the following bases:

1.The cost model that combines query cost and maintenance cost. 2. The search space proposed considers multi-query optimization for minimizing the query cost.3. The view maintenance optimization suggested by incremental and recomputation strategies in the cost model.

Following the first approach, three algorithms are proposed that are based on the same search strategy but are different in the maintenance strategy used. The proposed algorithms are : (1) The IRVSA incremental Recomputation strategy view selection algorithm. The second approach presents one algorithm: the IMDVSA(Incremental strategy Materialized Descendants view selection algorithm). It presents a new strategy for searching the space of views, the depends on the fact theat views' maintenance benefits from materializing the views' descendants when using an incremental update strategy. In this paper two approaches are proposed for the problem of selecting materialized views in a DW. Several simulation experiments were conducted to explore the relative performance of the proposed algorithms compared to each other.

In this paper[**8**]as the pattern discovery takes place mainly in the data warehouse environment, such long processing times are unacceptable from the point of view of interactive data mining. On the other hand, the results of consecutive data mining queries are usually very similar. This observation leads to the idea of reusing materialized results of previous data mining queries in order to improve performance of the system. In this paper they present the concept of materialized data mining views and they show how the results stored in these views can be used to accelerate processing of data mining queries. They demonstrate the use of materialized views in the domains of association rules discovery and sequential pattern search. We have addressed the problem of employing materialized data mining views to optimize data mining queries in large data warehouses. Materialized data mining views are physical data warehouse structures, created explicitly or implicitly, used to store precomputed results of selected data mining queries. They showed that in some situations, a new data mining query can be mapped to an existing materialized data mining view and it can be answered without the need to run a complete data mining algorithm. They classified mining methods exploiting materialized results of previous data mining queries, and identified situations in which those methods are applicable in the context of two main data mining techniques: frequent itemset discovery and sequential pattern discovery.

In [**11**] the authors have proposed two algorithms, one for view selection and maintenance and the second one for node selection for fast view selection in distributed environments. They have considered various parameters: query cost, maintenance cost, net benefit & storage space.

In [**13**] authors have presented a framework for selection of views to improve query performance under storage space constraints. It considers all the cost metrics in order to provide the optimal set of views to be stored in the warehouse. They have also proposed certain algorithms for selecting views based on their assigned weightage in the storage space and query.. The presented framework considered all the cost metrics associated with materialized views such as query execution frequencies, base-relation update frequencies, query access costs, view maintenance costs and the system's storage space constraints. The most cost effective views have been selected for materialization by the framework and the maintenance, storage and query processing cost of the views have been optimized.

In [**14**] a clustering based algorithm ASVMRT, based on clustering. Reduced tables are computed using clustering techniques and then materialized views are computed based on these reduced tables rather than original relations.

In [**17**] a MA based algorithm , based on **memtic algorithm(MA) is presented.** The memetic algorithm has been successfully applied to several NP-hard combinatorial optimization problems, and efficiency of the algorithm has been confirmed. A novel materialized view selection algorithm based on memetic algorithm is proposed in this paper. Extensive experiments show that the proposed algorithm performs much better than heuristic algorithm(HA) and GA.

In[**15**] author presented in this paper based on I-mine algorithm , the choice of algorithm is a major concern in finding the frequent queries for further reducing the time complexity. By considering these, we make use of the I-Mine algorithm, Index Support for Item Set Mining to mine the frequent queries. The structure of the IMine index is characterized by two components: the Item set-Tree and the Item-Btree. The two components provide two levels of indexing. The Item set-Tree (I-Tree) is a prefix-tree which represents relation R by means of a succinct and lossless compact structure. The Item-Btree (I-Btree) is a B+Tree structure which allows reading selected I-Tree portions during the extraction task. For each item, it stores the physical locations of all item occurrences in the I-Tree. The advantage of the I-Mine algorithm is that it can mine the frequent queries with less computation time due to its I-Mine index structure compared with the traditional algorithms like, Apriori and FP-Growth.

In this paper[**16**] , is based on I-mine algorithm with little modifications. The structure of the IMine index is described by two structures: the Item set-Tree and the M-tree. The two structures offer two levels of indexing. The Item set-Tree (I-Tree) is a based on FP growth algorithm. Again no. of items are reduced using multiple constraints, it will stored in M-tree structure. Comparing with i-mine index structure it will occupied less memory space.

3. COMPARATIVE STUDY

We have analyzed the various research works on several parameters and presented their comparison in the table below.

Table 1. COMPARISON OF VARIOUS RESEARCH WORKS

Features Authors	Technique	Issues addressed	Changes handled	Proposed work	Query Language Supported	Advantages	Disadvantages	Tool support / implementation
Zhou Lijuan, GeXuebin, Wang Linshuang, Shi Qian(2009)[5]	View Selection	reducing search space and time consumption	Related dimensions or relations	EMVSDIA (Efficient Materialized View Selection Dynamic Improvement Algorithm)	SQL based	reducing search space and time consumption		Not addressed

Lijuan Zhou1,Min Xu ,Qian Shi ,Zhongxiao Hao(2008)[6]	View selection & mainte nance cost	View selection under disk space & maintenance cost constraints.	Related dimension s or relations	QC_VSP (Query cost_ view selection problem)+ comparison with Greedy algorithm(G R_VSP)	SQL based	reduci ng searc h space and time consu mptio n	Sing le rand om algor ithm	Not addre ssed
Noha A.R.Yousri, Khalil M.Ahmed, Nagwa M. EI-Makky(2005) [7]	View Selecti on	View selection under disk space & maintenance cost	Related dimension s or relations	IRVSA incremental Recomputati on strategy view selection algorithm + IMDVSA(In cremental strategy Materialized Descendants view selection algorithm).	Not addre ssed	Decre ase time compl exity	IMD VSA uses only incre ment al upda te	Not addre ssed
Karde & Thakare (2010) [11]	View Selecti on	Query cost, maintenance cost, storage space &	In distributed environme nts	Algorithm for creation and maintenance of views + Algorithm for node selection	Not menti oned	Query perfor manc e impro ved	Only distri bute d envir onm ents highl ighte d	Not addre ssed
Agrawal, Chaudhari & Narasayya (2000) [8]	View Selecti on	Automated view and index selection		Framework for index & view selection + Candidate selection & enumeration techniques	SQL based	Robu st tool suppo rt + Both index es & view select ed	Only a part of phys ical desi gn spac e addr esse d	SQL Serve r 2000

Ashadevi & Balasubramanian (2009) [13]	View selection	Cost-effective view selection under storage space constraints		Framework for selecting views + Algorithm for the same + Cost metrics	Not addressed	All cost metrics considered	Query response time not considered + Threshold value not indicated clearly	Algorithms implemented in JAVA
Yang & Chung (2006) [14]	View selection	Attribute-value density + Clustered tables + Selection of views based on clustered /reduced tables	Related dimensions or relations	ASVMRT algorithm for view selction	SQL based	Faster computation time + Reduced storage space + 1.8 times performance better than conventional algorithms	Maintenance of reduced tables not addressed + Updating Reduced tables needs attention	In pubs database + ETRI
Elena Baralis, Tania Cerquitelli, and Silvia Chiusano	View selection	Cost-effective view selection under storage	Related dimensions or relations+ used in both dense	i-mine algorithm for selecting views	SQL based	Faster computation time	More memory space	Not mentioned

[15]		space constraints	and sparse					
T.Nalini & A.Kumaravel(2011)[16]	View selection	Cost-effective view selection under storage space constraints	Related dimensions or relations	i-mine algorithm (modification)for selecting views + using multiple constraints to reduce storage space	SQL based	Faster computation time + Reduced storage space	Selection of Threshold value is not evaluated	Algorithms implemented in JAVA + SQL server 2008
Qingzhou zhang & xia sun, ziqiang wang (2009)[17]	View selection	Cost-effective view selection under storage space constraints	Related dimensions or relations	MA algorithm for selecting views		Faster computation time + Comparison of GA & HA algorithm	Only optimal research	Not mentioned

4. Conclusion

In this paper we have presented an analysis of different approaches being proposed by various researchers to deal with the materialized views in data warehouse view selection cost. We have examined these techniques on various parameters and provided a comparative study in a tabular manner.

5 . References

[1] 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.

[2] Shukla, A., Deshpande, P.M., and Naughton, J.F. Materializes View Selection for Multidimensional Datasets. In *Proc. 24th VLDB Conference*, pp. 488-499, August 1998.

[3] H. Gupta. "Selection of Views to Materialize in a Data Warehouse". Proceedings of International Conference on Database Theory, Athens, Greece 1997.

[4] C. Zhang, X. Yao, and J. Yang. An evolutionary Approach to Materialized View Selection in a Data Warehouse Environment. IEEE Transactions on Systems, Man and Cybernetics, vol. 31, no.3, pp. 282–293, 2001.

[5]Zhou Lijuan ,Ge Xuebin ,Wang Linshuang,Shi Qian "Efficient Materialized View Selection Dynamic Improvement Algorithm" 2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery

[6] Lijuan Zhou1,Min Xu ,Qian Shi ,Zhongxiao Hao ,"Research on Materialized Views Technology in DataWarehouse" Beijing Educational Committee science and technology development plan project 2010.

[7]Noha A.R.Yousri, Khalil M.Ahmed, Nagwa M. EI-Makky," Algorithms for selection Materialized views in a Data Warehouse",IEEE,2005

[8] Bogdan Czejdo," Materialized Views in Data Mining", Proceedings of the 13th International Workshop on Database and Expert Systems Applications (DEXA'02)2002

[9] C. Dhote, and M. Ali, "Materialized view selection in data warehouse", In International Conference on Information Technology, 2007.

[10] A. Gupta, and I. Mumick, "Selection of views to materialize in a data warehouse", In IEEE Transactions on Knowledge and Data Engineering, vol. 17, 2005.

[11] P. Karde, and V. Thakare," Selection of materialized views using query optimization in database management: An efficient methodology", In International Journal of Management Systems, vol. 2.

[12] S. Agrawal, S. Chaudhari, and V. Narasayya, "Automated selection of materialized views and indexes for SQL databases", In Proceedings of 26th International Conference on Very Large Databases, 2000.

[13] B. Ashadevi, and R. Balasubramanian, "Cost effective approach for materialized views selection in data warehouse environment", In International Journal of Computer Science and Network Security, vol. 8, 2008.

[14] J. Yang, and I. Chung, "ASVMRT: Materialized view selection algorithm in data warehouse", In International Journal of Information Processing System, 2006.

[15] Elena Baralis, Tania Cerquitelli, And Silvia Chiusano," Imine: Index Support For Item Set Mining", Ieee Transactions On Knowledge And Data Engineering, Vol. 21, No. 4, April 2009

[16]T.Nalini, Dr.A.Kumaravel, Dr.K.Rangarajan," An Efficient I-Mine Algorithm For Materialized Views In A Data Warehouse Environment**",** Ijcsi International Journal Of Computer Science Issues, Vol. 8, Issue 5, No 1, September 2011 Issn (Online): 1694-0814

[17] Qingzhou Zhang, Xia Sun, Ziqiang Wang," An Efficient Ma-Based Materialized Views Selection Algorithm", 2009 Iita International Conference On Control, Automation And Systems Engineering

Authors

Mrs. T.Nalini received M.Sc from the Karanataka university, M.Tech from Bharath University in 2000, 2007 respectively. Now, she is pursuing Ph.D. in Bharath University. She was a Lecturer between 2000 and 2006. Currently she is an Assistant Professor in the Department of CSE. She has published more than 4 research papers in international journals. She also presented the paper in 15 national conferences and 2 international conferences. She is a life member of many professional bodies like ISTE, CSI, IEEE.

Dr. A.Kumaravel received PG in Computer Science in Applied Sciences from the MIT Chennai , Ph.D in theoretical computer science from Anna university in 1988,1992 respectively. He was worked as a

professor in CS for more than 20 years in Singapore. He has published more than 10 papers, and he also presented 20 papers in national and international conferences. He has organized workshops, national and international conferences, seminars in various organizations. Now, he is working as Dean of Computing Studies. He had received senior fellowship in UGC.

Dr.K.Rangarajan is a Professor and Dean of Science & Humanities, Bharath University, TN, India. He has about 60 research publications and guiding many research scholars. His research areas include Automata Theory, Formal Languages, Petri Nets and Graph theory.