"PERFORMANCE EVALUATION OF DATABASE CLIENT ENGINE USING MODULAR APPROACH"

PROF. RAMANAND SAMDEKAR
S.B. Jain Institute of Technology Management & Research, Nagpur, India
samdekar.ram@gmail.com

ABSTRACT: *Materialized view creation is an important aspect for large data centric applications. Materialized views create an abstraction over the actual database tables to the users. The MV view creation and selection is based on the various parameters like access frequency, base update frequency etc. Database client engine use cluster base approach to create materializes view to reduce query execution time.*

Keywords: Database client engine, cluster base, Data Warehouse, Threshold, materialized views.

1. INTRODUCTION

By observing the use of heterogeneous data in Data warehouses are designed to facilitate reporting and analysis of data, focuses on data storage. The data warehouse is intended to provide decisions support services for large volumes of data. So how to rapidly respond to query request is a great challenge in data warehouse. Quick response time and accuracy are important factors in the success of any database.

This paper proposes an approach of grouping in broader sense clustering the similar queries depending on certain parameters like access frequency to find the result from MV. The proposed work explores the area of a) query clustering for the selection of materialized view to decrease the response time and storage space deployment environment b) Ease network goals c) Enabling data sub-setting d) Enabling disconnected computing. To achieve these benefits, a methodology is proposed in this paper to form a quantitative optimize total query response time under a disk space constraint for data warehouse applications presented in [1] [3].

2. RELATED WORKS

Ordinary views are loaded with data every time it is called. Thus in real life applications materialized views are found to be more suitable to reduce query execution time. Materialized view creation involves several issues to consider. However, the main concern is to ensure availability of higher amount of user requested data directly from materialized views. Automated selection [13] of materialized views in large data oriented application is desirable for dynamic changes. A very few research work has been done about selection of materialized view using clustering approach. A significant work about dynamic clustering of Materialized view is done by [1].Paper [5] proposes a greedy algorithm BPUS based on the lattice model. And paper [6] discusses the issue of materialized view selection with the B-tree index. Paper [7] proposes PBS algorithm which make the size of materialized view as selection criteria. Paper [8] proposes preprocessor of materialized view selection, which reduces the search space cost and time complexity of static materialized view selection algorithm. These algorithms are based on the known distribution of query, or uniform distribution under the premise, which essentially are static algorithms. However, the query is random in actual OLAP system, so materialized view set which static algorithm generates cannot maximally enhance the query response performance in data warehouse. In order to improve further query response performance in data warehouse, paper [9] proposes dynamic materialized view selection algorithm, FPUS algorithm, which is based on query frequency in unit space. Relationship among several attributes in the form of a quantitative metric using a robust mathematical model, which is implemented here using line fitting algorithm. This quantitative measure guides to construct the materialized views.

3. PROPOSED METHODOLOGY

Our solution is an approach based on user behavior and their interactions with the system, particularly the distribution of their queries, to create the set of views to materialize. Materialized views are able to provide the better performance for DW queries. However, these views have maintenance cost, so materialization of all views is not possible. An important challenge of DW environment is materialized view selection because we have to realize the trade-off between performance and view maintenance is needed to consider following things.

1) Classification of queries: it is to determine the categories of data which the user is interested. [11]

2) Classification of attribute groups: it is to determine the groups of attributes for each class.

3) Merging classes: merge the data classes to make the classes that are most compact.

A clustering method is suggested in which similar queries will be clustered according to their query access frequency to select the materialized views that will reduce the execution time and storage space. When the query is posed, it will be

compared with already clustered or existing query and the recomputed MV will be returned as a result which will reduce the execution time of the query. In this approach, a framework is created which will reduce the execution time of query when posed to this framework. [9]

4. ALGORITHM FOR DATABASE CLIENT VIEW

The steps of the algorithm are as below.

I) Generation of random set of records for given tables in database by record generator.

II) Extraction or generation of all possible set of queries resolved by system on above created records.

III) Optimization of above set of queries according to their access frequency.

IV) Creation of MV according to query access frequency called as Threshold Value and according to Maximum Cluster Area Threshold % According to above step a of MV creation, 3types of MV are created as follows.

1) Single query to Multi table MV. In this response of single query is obtained from multiple MV table.

2) Single query to single table MV. In this response of single query is obtained from single MV table. [10]

3) Multiple queries to single table MV. In this response of multiple similar queries will be obtained from single MV table.

4) After creation of these 3 different types of MV, we will store these MV. Creating candidate views for materialization in our approach, we assumed that a data pattern is present in user queries, i.e. certain categories of data will be queried more frequently than others. Thus, it will be very useful to extract these patterns given the basis of which we will create the candidate views for materialization. Extracting the attributes of interest. Generally in a mediation system, a global schema representing the domain of use is provided. It is in terms of the latter are expressed the user queries. We analyze these queries to determine, among all the attributes of this schema, those in which users are interested, i.e. the most frequent attributes.

5. Algorithm Materialized_View_Creation

Begin

Step 1.

/* In this step, construct a (2×m) matrix called Important Attribute and Affinity Matrix (IAAM) from the array Total_Use and the matrix AAM to compute the degree of importance of attributes. */
Call method IAAM_Computation.[6]

Step 2.

/* In this step, the views are created one after one by taking the attributes from IAAM in descending order of importance. It takes as input numbers of views user want to create and also the corresponding size of each view. */

Call method Materialized_View_Creation.
End

6. AN ILLUSTRATIVE EXAMPLE

Consider an example of a query set where 10 queries are participating and 10 attributes are used in these queries. Say the name of the attributes is A1, A2, and A10. Execution of the algorithm Attribute_Affinity_Scale is shown below.

A. Attribute_Affinity_ Scale:

Step 2.

/* In this step, the views are created one after one by taking the attributes from IAAM in descending order of importance. It takes as input numbers of views user want to create and also the corresponding size of each view. */
Call method Materialized_View_Creation.
End

7. SELECTION OF VIEWS TO MATERIALIZE

The views created in the first phase of our approach cannot be all materialized. Indeed, the space for materialization, the frequency of update and the cost of access to sources is critical.

• The frequency of change: the views that rarely change are good candidates for materialization.

• The size of views: the views of small sizes are favored for materialization than large ones.

• The availability of sources: The views, whose data re sides in sources that are rarely available, should be materialized.

• The cost of access: the materialization of views whose data resides in sources with a high cost of access will improve the system performance. Thus, a view will be materialized, if it satisfies at least two criteria. [7]

8. CONCLUSION

Thus the paper proposes algorithm for the materialized view design problem, e.g., how to select the set of views to be materialized so that the cost of processing a set of queries and storage space required storing the data for the materialized views is minimized.This approach realizes on analyzing the queries so as to derive common intermediate results which can be time and to eliminate the need for creation of same MV for the query. The proposed algorithm for determining a set of materialized views is based on the idea of reusing

temporary results from the execution of the global queries. The cost model takes into considerationof selection of queries with high accessfrequencies, clustering them and creation of Materialized Views for the same. These high access frequency queries are further analyzed for requiredcluster area to create MV.

9. REFERENCES

[1] A Gong, Weijing Zhao, "Clustering-based Dynamic Materialized View Selection Algorithm"Proceedings of Fifth International Conference on Fuzzy Systems and Knowledge Discovery, 2008, China, pp391- 395.

[2] Jian Yang, Kamlakar Karlapalem, Quing Li,''Algorithms for materialized view design in data warehousing environment." K. Aouiche, P.Emmanuel Jouve, and J.Darmont

[3] , "Clustering-Based Materialized View Selection in Data Warehouses"Technical Report, University of Lyon, 2007.

[4] Shukla A, Deshpande PM, Naughton JF, "Materialized view selection for multidimensional datasets", Proceedings of the 24th International

[5] Conference on VLDB, Morgan Kaufmann Publishers, San Francisco, 1996, pp. 51.

[6] Shukla A, Deshpande P M, Naughton J F, "Materialized View Selection or Multidimensional Datasets", Proceedings of the 24th VLDB Conference, 1998, pp. 488-499.

[7] Dynamic Materialized View Selection Algorithm" Proceedings of Fifth International Conference on Fuzzy Systems and Knowledge Discovery, 2008, China,pp391- 395

[8] Hadj Mahboubi, Kamel Aouiche and Jérôm Darmont, "Materialized View Selection by Query Clustering in XML Data Warehouses" FourthInternational Conference on Computer Science and Information Technology-Jor dan.

[9] Gupta H, Harinarayan V, Rajaraman A, et al, "Index Selection for OLAP", Proceeding of International Conference on Data Engineering,1997, pp. 208-219.

[10] Mistry H, Roy P, Sudarshan S, et al, "Materialized view selection and maintenance using multi-query optimization", Proceedings of SIGMOD'01, 2001, pp. 307-3

[11] .S.Agarawal, S.Chaudhuri, V.Narasayya "Automated Selection ofMaterialized Views and Indexes for SQL Databses" Proceedings of 26th International Conference on Very Large Databases, Cairo, Egypt, 2000

[12] P.A.Larsen, Jingren Zhou "Efficient Maintenance of Materialized Outer-Join Views" 23rd International Conference on Data Engineering (ICDE 2007), Istanbul, Turkey, April 2007.

[13] Xia Sun, Wang Ziqiang "An Efficient Materialized Views SelectionAlgorithm Based on PSO". Proceeding of the International Workshop on Intelligent Systems and Applications, ISA 2009, Wuhan, China, May 2009

[14] S.H. Talebian, S.A. Kareem "Using Genetic Algorithm to Select Materialized Views Subject to Dual Constraints". InternationalConference on Signal Processing Systems, Singapore, May 2000

both query access frequencies and % threshold.The work presented here is the first stageresearch in

10. AUTHOR PROFILE

Prof. Ramanand Samdekar received the Master of Technology in Computer Science and Engineering Rastrasant Tukodoji Maharaj Nagpur university, Nagpur. Currently he is an Assistant. Professor in S.B. Jain Institute of Technology Management & Research, Nagpur, India. He has published three papers in international journals, two papers in national journal and one in national conference. He is having 7 year teaching experience and his field of specialization is Software development, Data mining , Software System, Network Security,

.