IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 04, Issue 03 (March. 2014), ||V6|| PP 28-32

Review on Fragmentation in Distributed Database Environment

Gurpreet Kaundal, Sukhleen Kaur, Sheveta Vashisht

[1]*(Research Scholar, Department of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab, India)*

[2]*(Research Scholar, Department of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab, India)*

[3]*(Asst. Professor, Department of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab, India)*

Abstract: - In Traditional environments, there are many advantages of distributed data warehouses. Distributed processing is the efficient way to increase efficiency of data. But the efficiency of query processing is a critical issue in data warehousing system, as decision support applications require minimum response times to answer complex, ad-hoc queries having aggregations, multi-ways joins overvast repositories of data. To achieve this, the fragmentation of data warehouse is the best to reduce the query execution time. The execution time reduces when queries runs over smaller datasets. The system performance is increased by allowing data to be spread across datamarts. So, it is very important to manage an appropriate methodology for data fragmentation and fragment allocation. Here focus is on the distributed data warehouses, which combines the known predicate construction techniques with a clustering method to fragment data warehouse relations by using the data mining-based horizontal fragmentation methodology for a relational DDW environment. DW decentralization gives the better performance; in the fragments are allocated to the corresponding site according to their frequency.

Keywords: - *Allocation, Distributed Data Warehouse, Fragmentation, K-mean.*

I INTRODUCTION

Data Warehouses (DWs) are usually built by centrally coordinated organizations. Basically, Data warehouse is a database which stores large amount of data. Data warehouse store current as well as historical data and this data is used by everyone for their works. For additional operations, the data is passed through an operational data store before they are used in DW for reporting .Also Data warehouse is used for reporting and data analysis.

To deal with the information technology applications, the DWs have been proposed the business planning and decision making. Many works have been proposed in this field, They intend optimizing the DW utilization. The Centralized data warehouse is very expensive because CDW responds the needs of several separate business units simultaneously by using a single data model and a unique database or centralized data storage optimization problem, remote access cost. Distributing requirements can be handled by paying attention to distribute a DW through the company several sites [2]. In DDW ,the decision makers provides a single view of data even though that data is physically distributed across multiple DWs in multiple systems at different branches. A distributed architecture supports integrated data flows between a CDW and individual data marts, allowing users to access rapidly enterprise wide information that where datamarts reside. By allowing data to be spread across data marts and analyzing smaller datasets, the DA is solution to improve the system performance. Consequently, this solution offers rapid response time as well as a broad access to data without compromising the integrity or the performance of production applications on the CDW.

1.1. Fragmentation

Fragmentation is a design technique to divide a single relation or class of a database into two or more Partitions such that the combination of the partitions provides the original database without any loss of information .This reduces the amount of irrelevant data accessed by the applications of the database, thus reducing the number of disk accesses. Fragmentation can be of any type: horizontal, vertical and hybrid/mixed.

1.1.1 Vertical Fragmentation

Vertical fragmentation splits a single relation R into sub-relations that are projections of relation R with respect to subset of attributes. These relations are in grouping with attributes and frequently accessed by queries. Projection built the vertical fragments[1] .By joining the fragments the original relation is reconstructed.

Name	Reg.No.	Course	Dept
Fragmentation1	Fragmentation2	Fragmentation3	

Table 1. Vertical Fragmentation

1.1.2 Horizontal fragmentation

Horizontal fragmentation, divides a single relation R into subsets of rows using query predicates. It reduces query processing costs by selecting the horizontal fragments that are built and the original relation is reconstructed by union of the fragments.

Name	Reg.No.	Course	Dept
Fragmentation 1			
Fragmentation 2			
Fragmentation 3			
Fragmentation 4			

Table 2. Horizontal Fragmentation

1.1.3 Mixed fragmentation (hybrid fragmentation)

The Mixed/Hybrid fragmentation is Combination of horizontal and vertical fragmentations. This type is most complex one, because both types are used in which horizontal and vertical fragmentation of the DB application [1]. The original relation is obtained back by join or union operations.

Name	Reg.No.	Course	Dept
Fragmentation 1		Fragmentation 2	
Fragmentation 3	Fragmentation 4		

Table 3. Mixed Fragmentation

In this report, the data mining fragmentation technique is used to improve the performance of the distributed DW system and reduces the execution time of queries. Here, allocation process and queries process are also involved. The allocation process allocates the data on the sites in network and maintains the replication of data. Queries are used to increase the accessing speed of data from the tables. Fragmentation, Allocation and queries improve the efficiency and performance of the system.

II DECENTRALIZATION METHOD

The process of Redistributing or dispersing functions, people or thing, powers away from the central location an authority is called decentralization. Basically decentralization is of three forms like deconcentration, delegation, and devolution. In deconentration, the decision making authority redistributed the regional level to the same central organization and it is a weakest form of the decentralization [2]. Delegation is the extensive form of decentralization in which not wholly controlled by central organization but through delegation, the responsibility for decision making is transferred to the semi-autonomous organization. And in Devolution, the responsibility for the decision making is transferred completely to the semi-autonomous units.

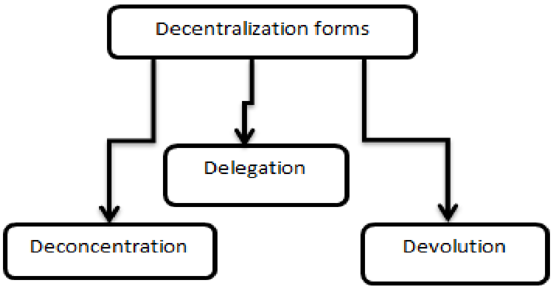

Fig. 1: Types of decentralization.

2.1 DW Decentralization

In which discussed about the DW decentralization. It involves the four phases [2]:

First phase is dimension predicate construction, in which t com_min algorithm is used for the complete selection of predicates according to rules.

Second Phase is dimension minterm predicate clustering, in which the multidimensional aspects are discussed.

Third phase is table's horizontal fragmentation, in which the DW horizontal fragmentation algorithm used for fragmenting dimension table according to its minterm predicates.

Fourth phase is fragment allocation, in which the three allocation strategies like simple fragment allocation, allocation with fragment replication and allocation with some fragment replication are used as shown in Fig1.

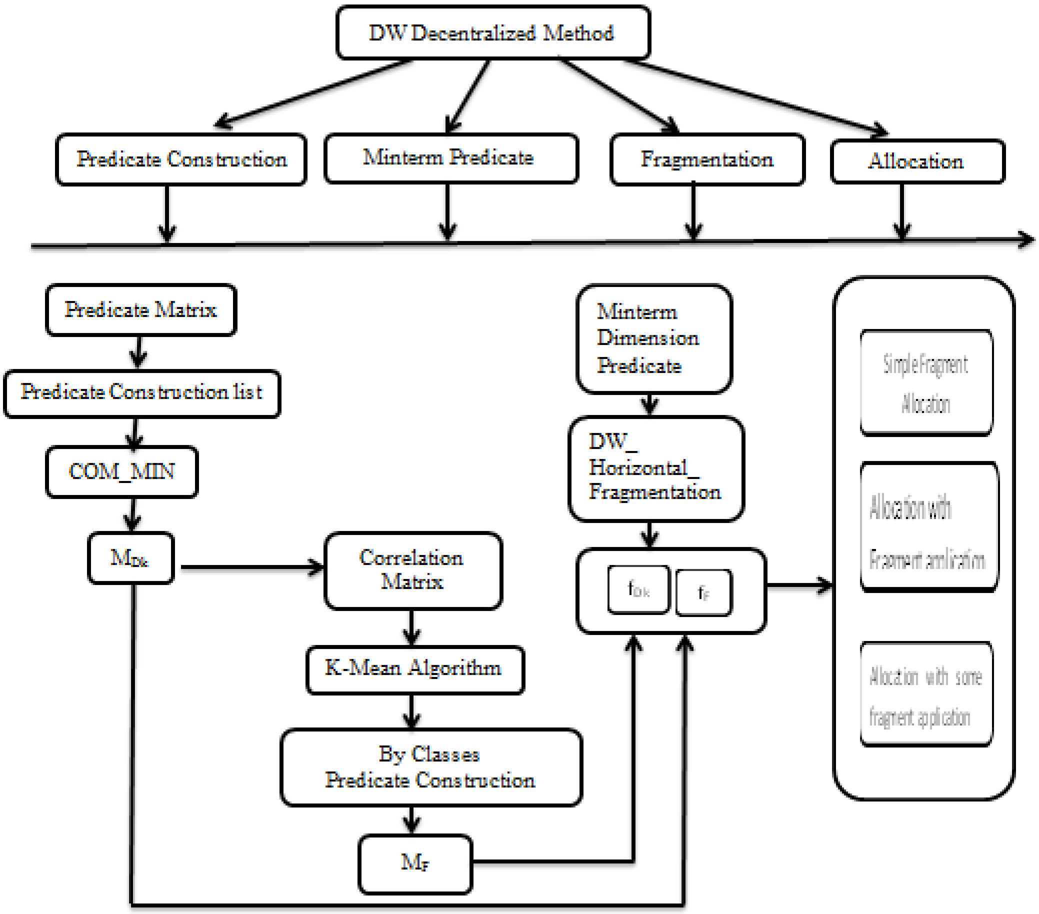
DW Decentralized Method
Predicate Construction
Minterm Predicate
Fragmentation
Allocation
Predicate Matrix
Predicate Construction list
COM_MIN
M_{Dk}
Correlation Matrix
K-Mean Algorithm
By Classes Predicate Construction
M_F
Minterm Dimension Predicate
DW_ Horizontal_ Fragmentation
f_{Dk}
f_F
Simple Fragment Allocation
Allocation with Fragment application
Allocation with some fragment aplication

Fig. 2: Decentralization Method [2].

2.2 Problem and Formalization

This section discuss about fragmentation of a star schema [1].A star schema is fragmented with the fact table F and a set of dimension tables: D= {D1, D2………, Dn} with the number of dimension v in star schema model, in which first the dimension tables are partitioned and then their fragmentation schemas are used to derive the fragments of the fact table. A dimension table is partitioned by using the affinity-driven algorithm. This algorithm uses the quantitative and qualitative information about applications [3].The quantitative information is used to describe the selectivity factors and the frequency of each query accessing this table and the qualitative information is used to describe the selection predicates defined on this dimension table. When performing data warehousing queries, the dimension table contains the attributes used to constrain and group data [4]. A dimension table is set of companion tables to a fact table in data warehousing. A dimension table D_k is formalized by: D_k= {$A^D{}_{K}$, $A^D{}_{k1}$, $A^D{}_{k2\ldots}$, $A^D{}_{Kx}$, $A^D{}_{Ky}$} where $A^D{}_{Kx}$ is an attribute.

Some additive values are provided by the fact table and these values act as a independent variables by which dimensional attributes are analyzed. A fact table is formalized by $F(K^F,A,A^m{}_2,\ldots\ldots,A^mp,\ldots..A^mq)$ where A^mp: F->Qp is a measure of F and Qp is its domain and $1\leq p\leq q$ with q number of measure in $F.A^m{}_j$ element contain a numeric values. The m letter means that it is used for measurement. In fact table the key is K^F={$A^D{}_1$, $A^D{}_{2,\ldots\ldots..}$ $A^D{}_{u,\ldots.}$ $A^D{}_v$} and the $A^D{}_u$ is foreign key which references a primary key of a dimension table and$1\leq u\leq v$

with v the number of dimension table referenced by the fact table F. Each element A^{D}_{u} of K^{F} determines all the facts of F.

To fragment the DW tables query workloads are captured from the company distributed sites. From the query workload a complete list of simple predicates is extracted.

A simple predicate is formalized by $P^{Dk}_{Z:}$ K^{D}_{k} Θ Valuer(s) where K^{D}_{k} is the primary key of Dk; Θ is a relational operator belonging to the set {=, <, >, ≤, ≥, ≠, In ()} and value(s) is a value from K^{D}_{k} domain. The predicate list P_{Dk}= {$P^{Dk}_{1,}$ $P^{Dk}_{2,\ldots\ldots\ldots\ldots,}$$P^{Dk}_{z}$,…….,$P^{DK}_{w}$} for the dimension table D_{k} applied the set of applications with 1≤z≤w and w the number of predicates defined on D_{k}.In a natural form each predicate can be produced and $z=2^{m}$ is the number of minterm predicate.

A minterm predicate is formalized by M_{Dk}= {m^{Dk}_{t} / m^{Dk}_{t}=^p^{Dk}_{z} E p_{Dk} P^{Dk*}_{Z}} where P^{Dk*}_{Z}= p^{Dk}_{z} or P^{Dk*}_{Z}= ¬p^{Dk}_{z} and 1≤k≤v, v is the number of dimension tables where 1≤z≤w and w the number of simple predicates defined on Dk.

After the formalization of fact tables, dimension tables, simple predicate and minter predicate select the set of predicate by using the com_min algorithm and then clustering the dimension minterm predicate by using the correlation matrix. After this, a Horizontal fragmentation algorithm is applied and then fragments allocation.

III DISCUSSION

In this section, consider a star schema having a one fact table Sales and the three dimension tables Product (Pro), Time (Time) and Customer (Cust) [2]. For example:- suppose that,the dimension tables (Pro) are fragmented into 50 partition using the attributes and also second dimension table is partitioned into 5 fragments using the attributes and the three dimension tables are partitioned into 4 fragments. The fact table is fragmented into 1000 fragments and it would be hard to maintain the sub star schemas. Therefore, it is important to reduce the number of fragments of the fact table. This problem is solved using the above mentioned algorithms[2]. The Horizontal fragmentation algorithm is to use to perform primary and derived horizontal fragmentation on dimension table and fact tables using algebraic restriction operation (α) according to the dimension or fact min term predicate list.

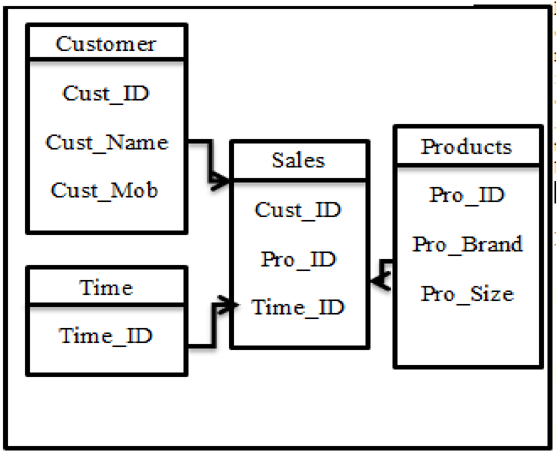

Fig. 3: Star Schema [2].

The queries workload is classified into two categories: One is specific queries and other is general queries. In specific queries, local needs are described and in general queries the needs of company sites are described. For example, compare sales amount by product of all the company sites. In DDW specific queries gives the better performance and the execution time of summed specific queries is reduced by 82%.Some general queries in DDW also give the better performance ,in which the execution time of the summed general queries is reduced by 76% and its compared to the Centralized context. In data warehouse, executing OLAP query can be very expensive because the data in data warehouse is not fragmented and modeled properly [2]. Problem of partitioning the data warehouse when the data is modeled using star schema is solved by using the horizontal fragmentation of star schema and OLAP queries can be executed efficiently [5]. For better performance the fragmentation is used as it gives the better results.

IV CONCLUSION

To design an effective distributed model, it is important to manage an appropriate methodology for data fragmentation and fragment allocation. Nevertheless, very little works address this problem in a distributed context; an optimization problem including the several interrelated problems like data fragmentation, allocation and local optimization. Each problem can be solved by using several different approaches. In this paper, we combine the known predicate construction technique with a clustering method to fragment data warehouse relations by using the data mining-based horizontal fragmentation methodology for a relational DDW environment. In which specific and general queries are used and the result of these queries are compared to centralized context. The specific queries gives better performance and it reduces the execution time.

REFERENCES

[1] Gawande A. D., Bhuyar P. R., and Deshmukh A. B., Horizontal Fragmentation Technique in Distributed Database, *International Journal of Scientific and Research Publications*, 2(5), 2012.

[2] Tekaya Karima, Abdelaziz Abdellatif and Habib Ounall, Data mining based fragmentation technique for distributed data warehouses environment Using predicate construction technique, *IEEE Publishers*, 2010, 63-68.

[3] Ladjel Bellatreche, Kamalakar Karlapalem, Mukesh Mohania and Michel Schneider, What can Partitioning do for your Data Warehouses and Data Marts, *IEEE Publishers*, 2000, 437-445.

[4] Kamber M. and Han J, Data mining concepts and techniques (Third Edition, Morgan Kaufmann Publishers, 255 Wyman Street, Waltham,, USA).

[5] Abdalla, H. I., Ali, and A., Amer, Dynamic Horizontal fragmentation and allocation Model in DDBMS, *International Conference on Information Technology and e-Services*, 2012.