

Agent Based Architecture in Distributed Data Warehousing

Bindia, Jaspreet Kaur Sahiwal

Department of Computer Science, Lovely Professional University
Phagwara, India

Abstract- The distributed data warehousing is mainly based on how the data is used in the dynamic data distribution on a set of servers. Currently Query cycling process is used in distributed data warehousing for searching the relevant information from a large database. In this Process of query cycling, if the searching query is not in the required data mart then this agent will automatically redirects that request to the other data marts for searching queries, until it found. But the network load and execution time is more and the data management also needs the collaboration and interaction between the machines in order to reply the user queries. So in our approach we will use multi agent based architecture in distributed DWH. The data distribution is different from the classical one which depends on the data use. The distribution consists in distributing data when the server reaches its maximum storage capacity limit. And we will use an individual buffer for storage of results. Now, Client has no need to be connected with dispatcher all the time to get result. The result will be stored in its own buffer by dispatcher.

Index Terms- Data warehousing, dynamic distribution, Data access, Multi-agent System, Query cycling process

I. INTRODUCTION

In distributed database, the Individual data marts are built, managed, and maintained flexibly in distributed data warehousing. However, the data mart does not solve the problems of storage space and performance. It is stand-alone and has data integration problems in a global data warehouse context [9]. Also, the performance of many distributed queries is normally poor, mainly due to the load balance problems. Also individual data mart are designed and tuned to answer the queries related to its own subject area, whereas the response to global queries depends on the global system tuning and the network speed. The Complexity of the global schema remains a big problem in distributed data warehousing.

To solve this problem of complexity, we use multi agent based architecture in distributed data warehouse. It will also facilitate the collaboration, interaction and independency of the different machines and to improve the parallel execution of the user queries. In this case, the distribution consists in distributing data when the server reaches its storage capacity limit. This distribution assures the scalability and exploits the storage and processing resources available in the organization using the data warehouse. The materialized views and indexes will be used on each individual machine that must be tuned and optimized for performance. Our work aims is 1) to develop a dynamic system that can manage the DWH automatically, 2) taking advantage of

the storage and processing resources available in the organization, 3) Reduce the network load, (4) improve the query response time, 4) use of individual buffer for client, therefore no need to be connected with dispatcher all the time to get result.

This paper is organized as follows: Sect. 2, gives the information about our finding related to multi agent based architecture and distributed data warehouse. In sect. 3, we will show our experimental results related to our work. In sect. 4, we mention the reviews that we have got. In sect 5, we will improve our work. Finally, in sect. 6, a conclusion is made.

II. RESEARCH ELABORATIONS

Data Warehousing is a collection of decision support technologies, aimed at enabling the knowledge worker to make better and fast decisions [3]. Centralized databases are very expensive due to its large set-up cost and are very flexible due to its centralized nature [9]. Making data marts was the first attempt to solve the problem of space and performance. Distributed data warehouse represents the enterprise DW but has smaller data stores that are built separately and joined physically over a network, providing users with access to relevant reports without impacting performance. But Data marts are basically stand alone and have data integration problems in global data warehouse context. An overview of distributed data warehousing and OLAP technologies [3], describes the back end tools for extracting, cleaning and loading the data into a data warehouse, and the front end client tools for querying and data analysis. An Efficient Approach for Data Placement in Distributed Systems [6], explains different types of fragmentation. Fragment allocation is a distribution design technique to improve the system performance by reducing the total query costs [6]. The allocation problem involves finding an optimal distribution of fragments to sites. DWS-AQA: A Cost Effective Approach for Very Large Data Warehouses [7], provide the technique data warehouse striping with approximate query. This focuses on query redirection process in distributed system. This operation is done when the query optimizer determines that a materialized view or other table processing a query request with less overhead than the requested table. The query redirection process determines which table delivers the answers effectively. A scalable architecture handles very large amount of data but also to assure interactive response time to the users. Agent based data storage and distribution in data warehouses [8], the used data distribution technique is different from the "classical" one which depends on data use. The distribution in this approach consists in distributing the data when server reaches its storage capacity limit. The proposed multi-agent model is composed of stationary agent

classes: Client, Dispatcher, Domain and server, and a mobile agent class called manager. These agents collaborate and achieve automatically the storage, splitting and access operation on the distributed data warehousing. But we also find that the client has to connect to dispatcher all the item to get the result back. This is also the disadvantage of this approach. So with reference to the above mentioned issues we are going to focus on a query cycling process using multi agent based architecture in the distributed DWH and will limit the above problems also.

III. PROPOSED METHOD

The query cycling agent will play an important role here. Cycling is the process of diverting a data from their normal destination to another one i.e. if the searching query is not in the required data mart then this agent is automatically send that request to the other data marts for searching queries, until it found. Using multi agent system the network loads and the execution time is decreased. Iteration agent is used to maintain that how many times the requested queries is to be recycle.

And also it facilitates the collaboration, interaction and independency of the different machines and to improve the parallel execution of the user queries.

Fig. 2 shows the proposed multi agent based architecture in distributed data warehousing. The Client agents act as an interface between the user and the DWH management system (Dispatcher agent). The users send the data storage and the data access queries to the Dispatcher agent. Client knows with which it can communicate and how to interact and cooperate with other agents. Its static knowledge is made up of its name and its address. This agent class does not have a dynamic knowledge.

As in existing approach in multi agent systems, the Dispatcher agent arranges the received operations according to their arrival order. These operations will be treated by the Messenger agent. When the Dispatcher agent receives the operation results from the Messenger agents, the Dispatcher can send the result to client only if client is connected otherwise; the result is stored in dispatcher's queue until the client will be connected again. Also if the queue is full then the result of the searched query which is arriving from the dispatcher agent will overflow.

Dispatcher agent has two queues. The first queue is used to store operations received from the Client agents. The second one is used to store the results provided by the Messenger agents. Then, the Dispatcher agent sends these results to the buffer of sending Client agent. So in our approach, there will be an individual buffer for client in multi agent system. Each client agent has its own buffer. This buffer is used to store a result of the searched queries which is received by the dispatcher agent. Now, Client has no need to be connected with dispatcher all the time to get result. The result will be stored in its own buffer by dispatcher.

The Messenger agents execute each operation found in the operations waiting queue of the Dispatcher agent. Each Messenger agent makes the execution plan of this operation. Then, it visits all the Domain agents concerned with this operation. The domain agents are responsible for sending the operations to the server agents which they control. Then they collect the replies sent by the server agents and transmit the final result to the messenger agent. It contains two queues. The first queue is used to store the operations brought by the messenger

agents. The second one is used to store the replies sent by the server agents.

The splitting operation will start when the machine reaches its storage capacity limit [8]. The role of this agent consists of the following steps. First, it creates a new Domain agent when it receives a splitting request. Then, it informs the Domain agent, asking for splitting of the location and the characteristics of the new one. Finally, it sends to the Dispatcher agent the new information concerning the two Domain agents in order to update the Domain agents list. The Splitting agent has as acquaintances the Dispatcher agent and the Domain agents that ask for splitting. Its static knowledge consists of its name and its address. Its dynamic knowledge is the list of splitting requests sent by the Domain agents.

The Meta base can be managed by the domain agents. It manages the data distribution on the domain agents, the network status, and the messenger agents load rate. This Meta base is also used by the messenger agents to make the execution plans of the received operations and determine the domain agents to visit. The splitting agent also used for updating it at the end of each splitting operation by Meta base.

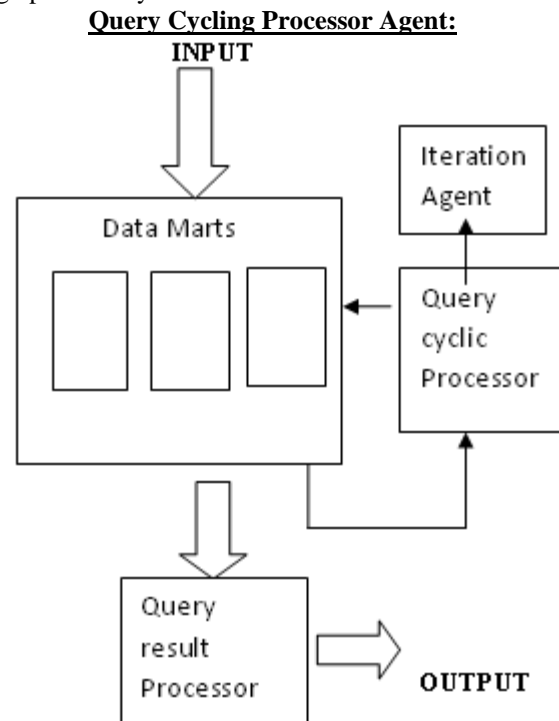


Fig. 1: Query Cycling Processor Agent

In fig.1, there are different data marts. The query cycling agent will play an important role here. Cycling is the process of diverting a data from their normal destination to another one i.e. if the searching query is not in the required data mart then this agent is automatically send that request to the other data marts for searching queries, until it found.

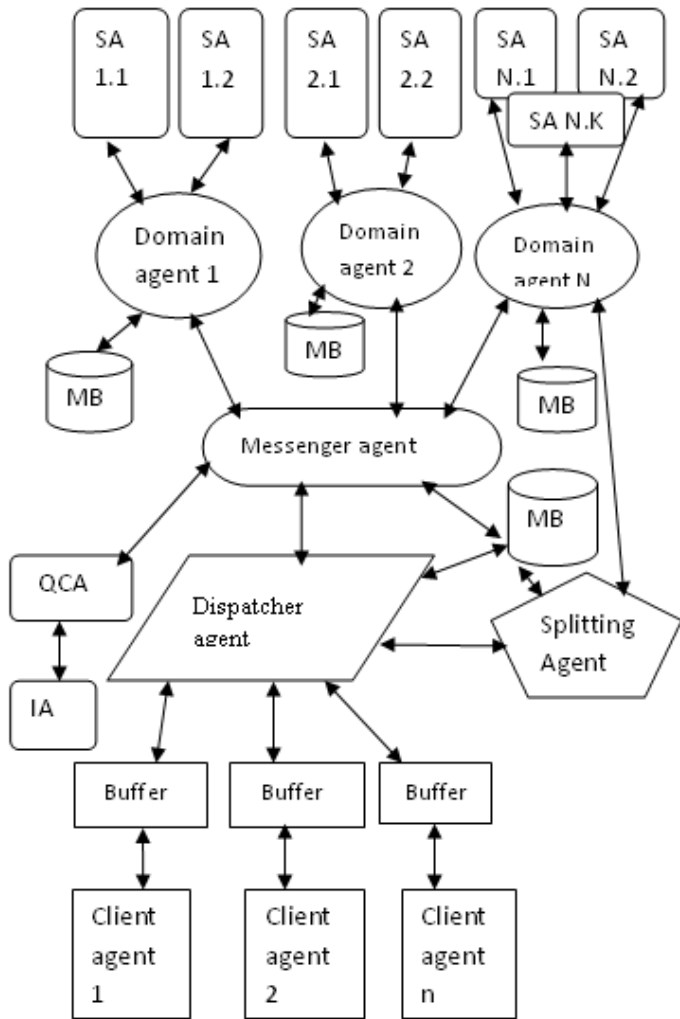


Fig. 2: Proposed method

IV. RESULTS

In this section, experimental results have been shown in the form of screen shots. Snapshots shows that the given input field data have been searched in the Datamart1 and produce the output. If the result is not found then it will send it to the next Datamarts2 and 3 respectively. We can search data by query or by attribute name as shown in fig.3. Finally it searched all the Data marts and produces the results is shown in the Figures.



Fig. 3: Search option

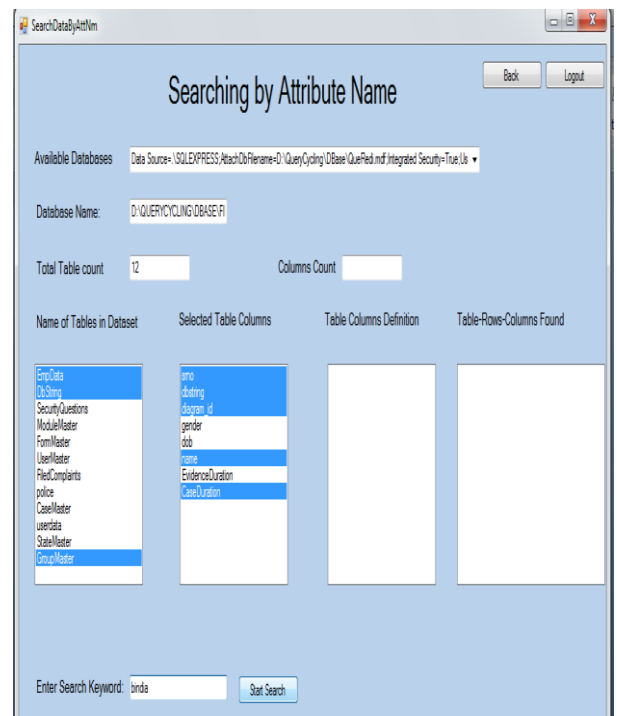


Fig. 4: Searching by attribute name

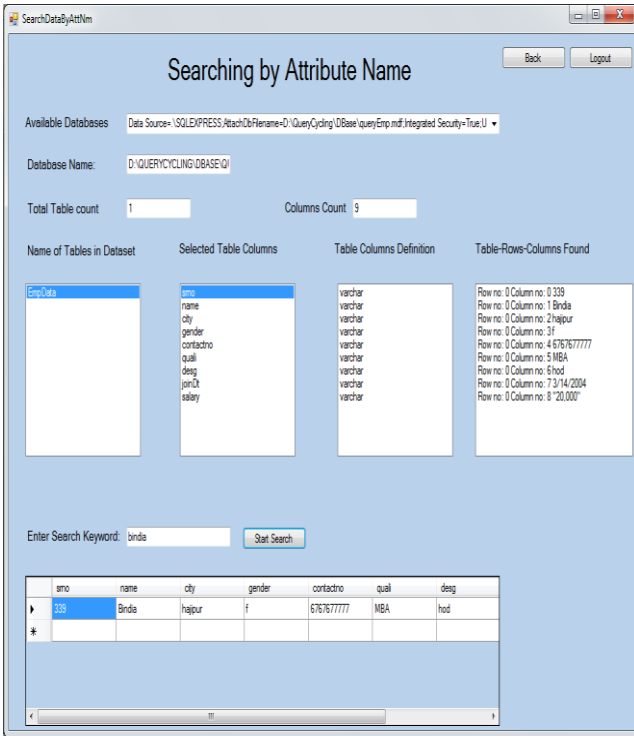


Fig. 5: Shows the result (By attribute name)

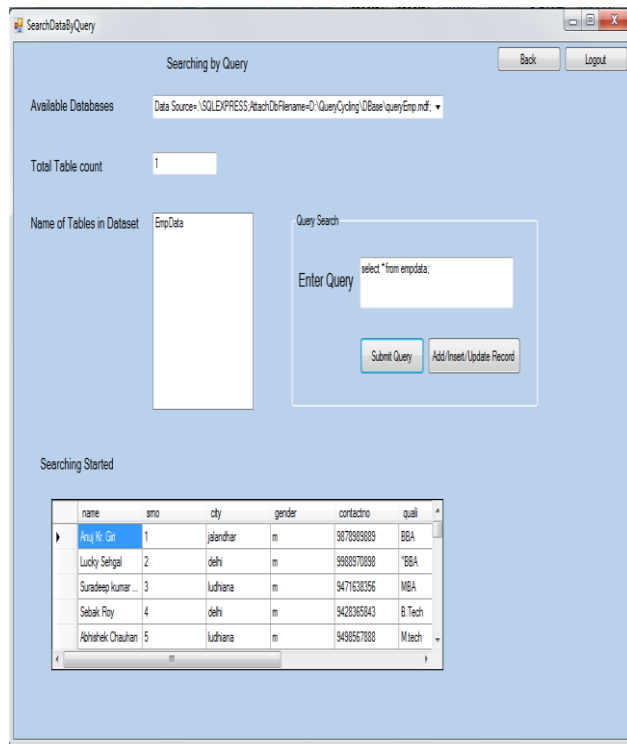


Fig. 6: Shows the result (searched by query)

Figures	Description
Figure.1	Query Cycling Processor Agent
Figure.2	Proposed method

Figure.3	We can search by attribute name or query
Figure.4	It shows all the available data marts, number of tables in a specific data mart and number of columns in specific table. we search through attribute name.
Figure.5	It will show the results from specific data mart automatically where it resides i.e it will redirect automatically and shows the Specific data mart in which it resides.
Figure.6	we search through SQL SERVER query. It will show the results from specific data mart automatically where it resides.

V. CONCLUSION

The overview of data warehousing and multi agent based architecture in it were discussed. We have presented some researches that deal with the data distribution in the data warehousing context and the multi agent based systems. We discussed about the existing problem and our proposed method for that problem. We described here is multi agent based system based query cycling process in distributed DWH and also remove the disadvantage of multi agent approach by placing an buffer, so that there will be no need for client to connect at all the time to get the results. Now it is not connected approach. Buffer will be on client side. This architecture ensures high performance to dynamic content applications even during overload conditions such as those during time-of-day effects. We have concentrated only on query cycling agent and an iteration agent in distributed data warehousing and limit some disadvantages of this architecture. In future we can implement the remaining agents which we can propose in our system.

ACKNOWLEDGEMENT

I am very thankful to my Dissertation mentor Miss. Jaspreet Kaur Sahiwal, who guided me time to time in developing my research related work. She left no stone unturned to help me. Because of my mentor kind support and a very good knowledge of the subject, it becomes possible to complete my research. I also want to say thanks to some of my friends who helped me when I was unable to move further in my research topic. And finally God and my parents, due to them I am able to stand at this stage in all respects. Thanks to all of you.

REFERENCES

- [1] G. Satyanarayana Reddy "Data Warehousing, data mining, OLAP and OLTP technologies are essential elements to support decision-making process in industries", (2010), pp. 2865-2873.

- [2] Mohammad Rifaie "Data Warehouse Architecture and Design", (2008), IEEE, pp. 58-63.
- [3] Surajit Chaudhuri, Umeshwar Dayal "An Overview of Data warehousing and OLAP Technology", ACM, (1997), pp. 23-33.
- [4] Jixue Liu, Millist Vincent "An Architecture for data warehouse systems", In the Proc. of IEEE Intl. Conf. on Global Connectivity in Energy, Computer, Communication and Control (TENCON'98), (1998), pp.107-110.
- [5] Hugh J. Watson" Data Warehouse Architectures: Factors in the Selection Decision and the Success of the Architectures", (2005), pp.10-22.
- [6] Hassan I. Aballla "An Efficient Approach for Data Placement in Distributed Systems", IEEE, (2011), pp. 297-301.
- [7] Jorge Bernardino, Pedro Furtado, Henrique Madeira "DWS-AQA: A Cost Effective Approach for Very Large Data Warehouses", IEEE, (2002), pp. 33-42.
- [8] Nader Kolsi, Abdelaziz Abdellatif, Khaled Ghedira "Agent based data storage and distribution in data warehouses", (2008), pp. 597-617.
- [9] Jorge Bernardino, Henrique Madeira "Data Warehousing and OLAP: Improving Query Performance Using Distributed Computing", (2001), pp 122-130.
- [10] WANG Fu shan "Application Research of Data Warehouse and Its Model Design", (2009), pp 1-6.

AUTHORS

First Author: Bindia, Department of Computer Science, Lovely Professional University, Phagwara, India.
Email id - kaushal_ne22@yahoo.co.in

Second Author: Jaspreet Kaur Sahiwal, Department of Computer Science, Lovely Professional University, Phagwara, India. Email id - jaspreet.14752@lpu.co.in,