Local Schemas Modifications Propagation and Maintenance of Metadata in Multidatabase System

Mohammad Ghulam Ali
Academic Post Graduate Studies & Research
Indian Institute of Technology,
Kharagpur - 721 302, INDIA

ABSTRACT

One of the challenging problems in the multidatabase systems is to find the most viable solution to the problem of interoperability of distributed heterogeneous autonomous local component databases. This has resulted in the creation of a global schema over set of local component database schemas to provide a uniform representation of local schemas. The main objective of this paper is to maintain the Global Schema and the Global Directory/Dictionary/Metadata as the local schemas are autonomous and evolve over time. In this paper we consider uniform object-oriented data models which is used during integration of different heterogeneous distributed databases from different data models and proposed a framework to maintain an objected-oriented Global Schema in the multidatabase system. The Bottom up local schema modifications propagation in the Global schema is proposed and during this process the Global Directory/Dictionary/Metadata is maintained without manual intervention.

General Terms

Maintenance of Metadata of Multidatabase System.

Keywords

Multidatabase, Object-Oriented Database, Replication Agent and Local Schemas Modifications Propagation.

1. INTROCUCTION

This paper addresses the problem of propagating local structural schema changes to an already acquired integrated schema of a federation. Our approach is to regard this problem from a schema integration point of view. The main contribution is the development of a framework for performing evolution of an integrated schema without information loss and avoiding as much as possible re-integration steps.

To efficiently retrieve information from heterogeneous and distributed data sources has become one of the top priorities for institutions across the business world. Information from these sources needs to be integrated into one single system such that the user can retrieve the desired information through the integrated system by a single query.

Keeping in mind the progress in communication and database technologies (concurrency, consistency and reliability) has increased the data processing potential.

Various protocols are proposed and implemented for network reliability, concurrency, atomicity, consistency, recovery and replication. The current demand is to access data from various existing heterogeneous database servers distributed among remote sites in a network and also to insert and update data at existing distributed heterogeneous databases which are

autonomous and evolve over times. If any organization has headquartered in any country and has many branches across the globe, wants efficient and quick retrieval of information for any kind of decision supports.

Heterogeneous data source means there is no homogeneity among the databases at various sites or at least component databases differ in some important respect (e.g. the DBMS they are running or perhaps the data model implemented by it – relational, object-oriented, etc).

A multidatabase system (MDBS) is a database system that resides on top of existing local autonomous component databases systems (CDBSs) and presents a single database to its users [1]. MDBS usually maintains a single global database schema [1], which is integration of all local component database schemas. In this paper we do not discuss on local schemas integration. Please see the details work of [2,3,4,5,6,7,8,9,10,12]. The schema contains the format, structure, and organization of the data in a system. MDBS maintains only the global schema and the CDBS actually maintains all user data. Creating and maintaining the global schema, which requires the use of database integration techniques, is a critical issue in multidatabase systems. A variety of approaches to schema integration have been proposed; e.g., [2,3,4,5,6,7,8,9,10,12]. The main schema integration problem is associated with combining diverse schemas of the different databases into a coherent global view by reconciling any structure or semantics conflicts between the local component databases [7].

Many researchers have already worked on local schemas integration in the multidatabase system and during this phase they have well addressed the problem of schematic and semantic conflicts, correspondence assertion or investigation, mapping between local schemas and global schema and schema transformation. Few researchers have addressed the issue of local schemas modification propagation into a global schema.

In this paper, we have proposed a framework to propagate local schemas modifications in the multidatabase system.

1.1 Why we have considered Object-Oriented Data Model?

In our approach, we considered the object-oriented data model as common data model constructing a global conceptual model by set of integration operators.
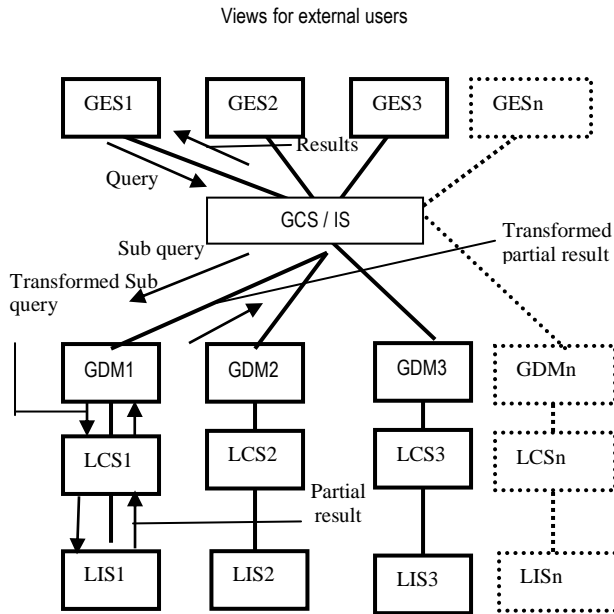
Views for external users

Figure 1 Object-oriented data model architecture of the Multidatabase System

Where

LIS	– is a physical data organization on each site
LCS	– is a logical organization at each site
GDM	– is a local schema in Global Data Model
GCS	- is a global conceptual schema
IS	- is an integrated schema
GES	- is global external schema, The n-th representation of the database through GUI of web-based interface

The object-oriented model has grown in popularity because of its simplicity in representing complicated concepts in an organized manner or we can say due to its semantic richness and availability. We can also say that in object-oriented data model there is only one concept (objects) is used as opposed to the two concepts (entities and relationship) of the entity relationship model. Using two concepts may cause problems in the process of schema integration and schema transformation because the same real world object may be modeled as entity in one schema but as a relationship in another schema [9]. Using encapsulation, methods, and inheritance, it is possible to define almost any type of data representation. Consequently, object-oriented models have been used not only to model the data, but also to model the data models. That is, object-oriented models have seen increased use as canonical models into which all other data models are transformed into and compared with. The object-oriented model has very high expressiveness and is able to represent all common modeling techniques present in other data model. Hence, it is a nature choice for a canonical model [7]. In general, tightly coupled systems integrate the diverse sources of intelligence through a global conceptual scheme, normally denominated canonical model, providing a uniform vision of the diverse component at a high level [8]. The use of a canonical model hides the structural differences between the different components and gives to the user the illusion to be accessing a simple centralized database. Since we adopt an object-oriented data model as the data model of the multidatabase system [2,3,4,5,6,10] also illustrated in figure 1, the global schema as we proposed in an object-oriented data model is a virtual schema because no actual data are stored for this schema. Thus, the classes in the global schema are called

virtual classes and the objects associated with the classes are called virtual objects.

2. STRUCTURAL/DATA LOGICAL MODEL BASED ON ANSI/SPARC

The structural/Data Logical model as shown in figure 2 of the MDBS is a 4-tiered client-server model [11] with a single multidatabase server. This model provides the advantages of a layered architecture, centralized control, global clients and distributed data access. The four layers of the structural model are the global clients of the MDBS, the application server, the Multidatabase Management System Server (MDBMS) and heterogeneous distributed local component databases. **Layer 1** includes a software process capable of performing queries to the MDBMS server through application server. **Layer 2** holds the application server including application program, business logic and API. **Layer 3** is a Multidatabase System Server. **Layer 4** holds the server node of the MDBS. Layer 3 includes Multidatabase Management System (MDBMS), Global Schema and Global Directory/Dictionary/Metadata. Layer 4 contains Local Database Management Systems (LDBMSs), local heterogeneous component databases that stores the distributed data.
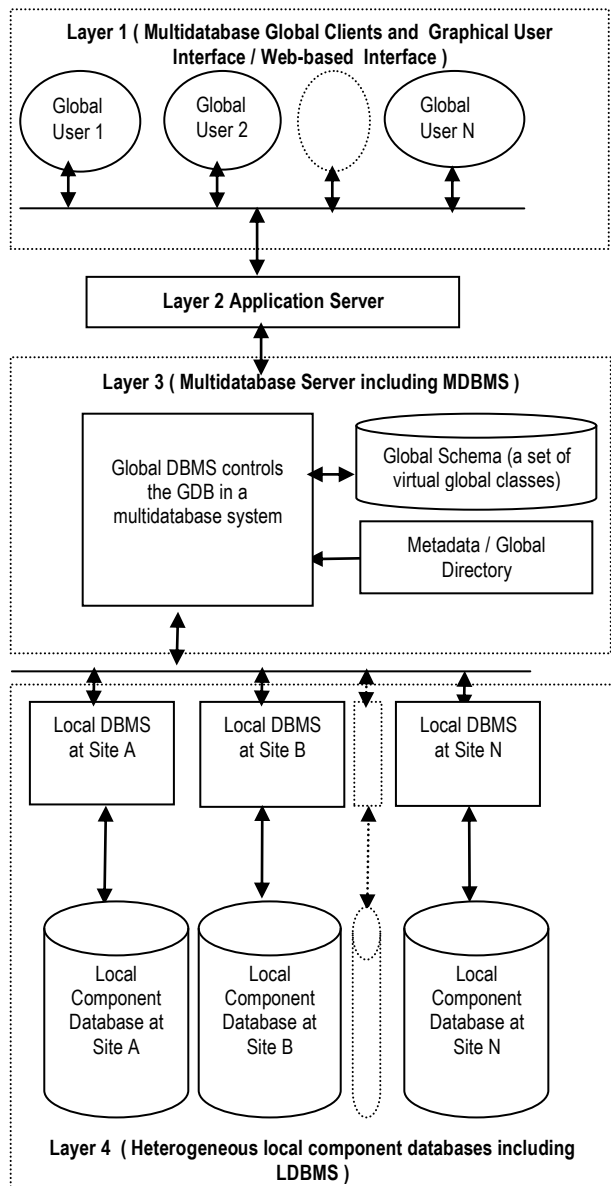

Figure 2. Structural Model/Data Logical Model

If client interface is a Graphical User Interface (GUI), then application server at layer 2 will have client program and API. If client interface is a web-based interface, then the application server at layer 2 will have web server and web application. If client program is a Graphical User Interface then the client application will be published through thin-client on end users desktop.

Now the most challenging issue in the aforesaid system is how to maintain Global Schema and Global Directory/Global Dictionary/Metadata without manual interventions as the local schemas are autonomous and evolve over time. However, based on changes made in the global schema it will require minor modifications in the Client Application Program.

3. LOCAL SCHEMAS MODIFICATIONS PROPAGATION INTO A GLOBAL SCHEMA AND MAINTENANACE OF GLOBAL SCHEMA AND GLOBAL DIRECTORY / GLOBAL DICTIONARY / METADATA

We extend the previous work [10] and proposed a framework through which we can maintain global schema and Global Directory/Global Dictionary/Metadata without manual intervention.

The main challenging issue is now how to maintain global schema as all local schemas are autonomous and evolve over time. How to automatically propagate local schemas modifications into a global schema in the multidatabase system. How to update Global Directory or Global Dictionary or we say metadata information of Global Schema based on local schemas. Metadata is information that describes data. Key metadata includes the name and location of the data source, the structure of the data, data item names and description. Because of the heterogeneous nature of various database systems, the task of metadata management can be more complex than homogeneous environment. The maintenance of this metadata can become an important issue, as the local schemas evolve over time, there should be a mechanism to import and synchronize metadata for global schema.

3.1. Mapping of Local Heterogeneous Schemas with Common Global Schema
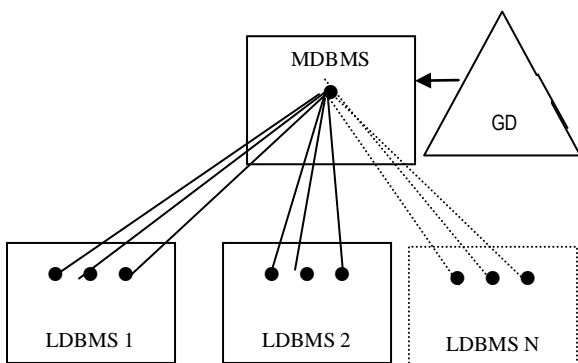

Figure 3 Mapping of local schemas with a global schema

Every Local Database Management System (LDBMS) will have many databases with distinct name. Distributed heterogeneous databases means there are many LDBMS with different data model. We propose a mapping methodology of N

numbers of local database from N numbers of Heterogeneous LDBMS with a single Multidatabase Management System maintaining single Global Database and Global Directory created by integration of all heterogeneous databases as shown in figure 3.

3.2. Framework for Model independent Replication Agent

We propose every local heterogeneous database management system will have model independent replication agent that will replicate/relay schema modifications in bottom up unidirectional manner to Uniform Global Data Model and similarly if all Global Data Models will have the same replication agent that will replicate/relay in the same manner every schema modifications to global schema then the global schema will be maintained automatically without manual intervention and the Global Directory/Dictionary/Metadata will be maintained. In case, if any local site uniform global data model is not working at the time of local schema modifications then all local schema modifications will be stored in a schema update query log and all pending schema modifications queries log will be relayed as soon as the same site is restored. This method will only take care of relay schema modifications not data as the replication user that will be used by the replication agent will have only alter privilege permission to alter schema in bottom-up manner.

Our proposed framework will work as chain replication of the schema modifications as A → B → C. Where A is a database server where local schema does exist, B is a server where local schema in a uniform global data model does exist and C is a multidatabase system server where a global schema in the multidatabase system does exist.

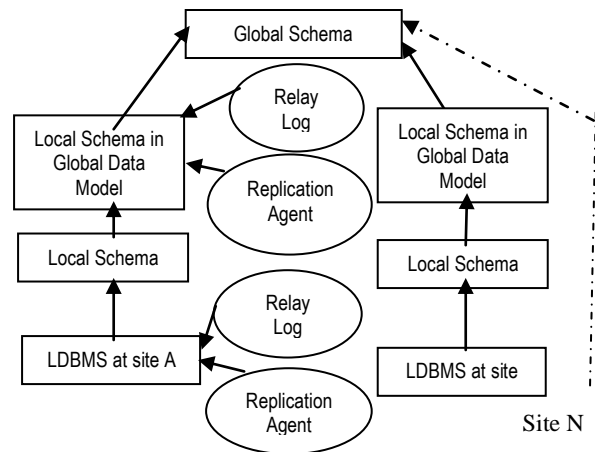We show a proposed framework through Figure 4.

Figure 4 Local Component Databases Modifications Propagation in the multidatabase system

A replication agent will participate as middleware between the LDBMSs and GDBMS. It is responsible for negotiating and establishing communication between LDBMS and GDBMS and transferring schema structure changes in bottom-up unidirectional manner.

We can also say that the replication agent will act as Log Transfer Manager (LTM), will read transaction log related to the local schema modifications and will transfer them to the replication server for further processing.

3.3. Model Designed and Tested for Chain Replication of Schema Modifications Propagation

The model of the chain replication for schema modifications propagation using three homogeneous
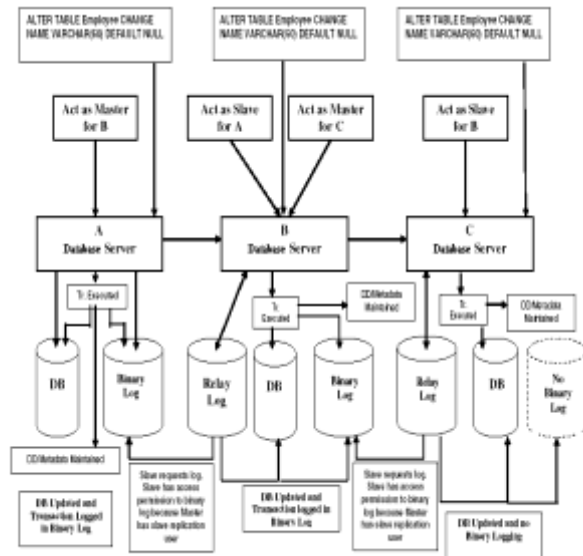

Figure 5 Model of chain replication

database management system platforms we designed and tested in our lab which is shown in figure 5. In this model Server A works as master for server B and Server B works as master for Server C. The Server B also works as Slave for Server A and Server C works as slave for Server B. Server A relay log to Server B and Server B relay log to Server C.

When a DML for schema modification is issued at server A, the statement is executed and the schema is modified, metadata is maintained and also the DML is logged in the binary log and relayed to server B. At server B the same DML is executed there and schema at server B is modified, metadata is maintained and the DML is logged and relayed to server C. At server C the same DML is executed and schema is modified, metadata is maintained.

In the same manner we can implement a framework as shown in figure 3 and 4 for local schemas modifications propagation in the multidatabase system to maintain global schema and global directory / global dictionary / global catalog / metadata of the multidatabase system. The model independent replication agent will fetch only those DML statements which are related to the schema structure modifications and will relay them in bottom-up replication manner.

4. CONCLUSION

This paper proposed a framework for local schemas modifications propagation in the Multidatabase System and will maintain Global Schema and Global Directory/Dictionary/Metadata without manual intervention as the local schemas are autonomous and evolve over time. In future we shall address other issues in the multidatabase system.

5. ACKNOWLEDGEMENTS

6. REFERENCES

[1] M.W. Bright, A.R. Hurson, and Simin H. Pakzad, A Taxonomv and Current Issues in Multidatabase Svstems. *IEEE*, 1992.

[2] E. Pitoura, O. Bukhres, and A. Elmagarmid. Object Orientation in Multidatabase Systems. *ACM Computing Surveys*, 1995.

[3] Ching-Ming Chao. Schema Integration between Object-Oriented Databases. *Tomkang Journal of Science and Engineering*, 2001.

[4] Rehab M. Duwairi. A framework for Generating and Maintaining Global Schemes in Hetrogeneous Multidatabases Systems. *IEEE*, 2003.

[5] Soon M. Chung and Pyeong S. Mah. Schema Integration for Multidatabase Using the Unified Relational and Object-Oriented Model. *ACM*, 1995.

[6] B. Thuraisingham. Application of object oriented technology for integrating heterogeneous databases systems. *ACM*, 1995.

[7] L. Ramon, and J. R. Kalmus. Schema Integration Methodologies for Multidatabase and the Relational Integration Model – Candidacy document. 1999.

[8] A. Mufioz, J. Aguilar and M. Rodrigo. Integration Ontology for Distributed Databases. 2006.

[9] Keim A.Daniel, H. Kriegel and A. Miethsam. A. Integration of Relational Database in a Multidatabase System based on Schema Enrichment.

[10] Ali M. Ghulam. Object-Oriented Approach for Integration of Heterogeneous Databases in a Multidatabase System and Local Schemas Modifications Propagation. *IJCSIS*, USA, November, 2009.

[11] Ali M. Ghulam. A Multidatabase System as 4-Tiered Client-Server Distributed Heterogeneous Database System. *IJCSIS*, USA, November, 2009.

[12] Ali M. Ghulam, Murtuza M. Ghulam. A Framework for Creating Global Schema Using Global Views from Distributed Heterogeneous Relational Databases in Multidatabase System, *GJCST*, USA, April, 2010.

Mohammad G. Ali   Received a degree of Master Diploma in Computer Science (1991) and Master of Science in Mathematics (1993) obtaining first class and also obtained top rank in the university in the computer science. He is a member of IAENG, Hong Kong and IACSIT, Singapore. He has published two papers in the International Journal of Computer Science and Information Security, USA and one paper in the Global Journal of Computer Science and Technology, USA. His one paper is also accepted for publication in the International Journal of Computer and Electrical Engineering (IJCEE), Singapore. He is a System Engineer Grade I in the Indian Institute of Technology Kharagpur, West Bengal, India. He is associated with System Analysis & Design, Programming, Implementation and Maintenance of Client-Server DBMSs and Web Applications Development. He is also associated with Database Administration, Web Server Administration, System Administration and Networking of the Institute. He has deployed many small to big projects on the Institute Network. His area of research is Distributed Databases and Software Engineering.