IMPLEMENTATION OF LIBRARY SYSTEM USING VERTICAL FRAGMENTATION METHOD IN DISTRIBUTED DATABASE

Shwe Mi San , Kyi Zar Nyunt
Computer University, Taung- Ngu
shweminsan@gmail.com, kyizarnyunt11@gmail.com

Abstract

Distributed Database Management System (DDBMS) is a software system that permits the management of the distributed database and makes the distribution transparent to users. DDBMS consists of a single logical database that is split into a number of fragments. Each fragment is stored on one or more computer under the control of a separate DBMS and is connected by communication networks. The distributed library system is used vertical fragmentation method. Book information, loan information and member information are vertically fragmented and stored at three different sites. The system is also proposed as a peer to peer distributed database by using Java programming language and Microsoft Access Database.

1. Introduction

Database and database technology are having a major impact on the growing use of computer. Databases are used in thousand of organizations ranging from large government agencies to small business. The databases play a critical role in almost all areas where computer are used, including business, engineering, medicine, law, education and library, etc. Today, many people can get the library resources from computer based library system.

Fragmentation is breaking a relation into small table fragments and allocation then on the different sites. There exist three fragmentation types: vertical, horizontal, hybrid. Vertical fragmentation consists of subdividing a relation into sub relations that are projections of the original relation according subset of attributes. The horizontal fragmentation divides a relation into subsets of tuples based on selection operations. The hybrid or mixed fragmentation consists of dividing a relation horizontally, and then splitting vertically each of the obtained horizontal fragment or vice-versa [2].

This thesis is implemented using vertical fragmentation. All tables in library database will be divided as separate tables. The distribution of database on network achieves the advantages of performance, reliability, availability, that are inherent in distributed system. The library system can support all users to view library information from both local sites and different sites.

2. Related Work

Vertical fragmentation is used in order to increase transaction performance. The more obtained fragments are close to transaction requirement, the more the system is efficient. The ideal case occurs when each transaction match exactly a fragment, i.e it needs only this fragment. If some attributes are always used together, the fragmentation process is trivial.

Using an objective function used on the group models we obtained the implementation of an evaluator of partitions that can be used in the verification of some scheme of the data fragmentation (“Adrian Runceanu ” university constsitin Brancusi Targu-Jiu, Adrin r@utgjiu. Ro “Towards vertical fragmentation in distributed database [1]). Navathe and Ra proposed in 1989 a graphical technique of partitioning. The attribute affinity matrix is considered as a complex graph where nodes represent attributes and edges’ weights represent the affinity values. The algorithm, by successively affinity edges, generates all the fragments in one iteration by considering a cycle as a fragment. The algorithm has a complexity of O (n2), where n is number of attributes, and has the advantages of not using on object function [6].

3. Distributed Database

Distributed Database System (DDBS) technology is the union of what appears to be two diametrically opposed approaches to data processing: database system and computer network technologies. A distributed database is a logically interrelated collection of share data that is stored on computer at several sites of a computer network and in which users can access data at any site in the network. An essential feature of a true distributed database is that users or programs work as if they have access to the whole database locally. Distributed Database System consists of a collection of sites, connected together via some kind of communication network, in which

- Each site is a full database system site in its own right, but

- The sites have agreed to work together so that a user at any site can access data anywhere in the network exactly as if the data were all stored at user's own site.

3.1 Distributed Database Management System (DDBMS)

Distributed Database Management Systems (DDBMS) are a marriage between database and network. The distributed database management system is expected to have the following function:

- Communication services to provide access to remote site and allow the transfer of queries and data among the sites using the network.
- Metadata about the details of distributed data storage
- Distributed query processing
- Security control to maintain appropriate access to the distributed data
- Concurrency control to maintain consistency of replicated data
- Recovery services to take account of failures of individual sites and the failures of communication links.

3.2 Architecture of Distributed Database

Distributed database systems have two types of architecture. They are:

(1) Peer to Peer (P2P) and

(2) Client –Server architecture.

This system is implemented using peer to peer distributed database.

3.2.1 Peer to Peer (P2P) distributed system

In a peer to peer distributed system, each site needs individual internal schema definition called local internal schema (LIS). The global conceptual schema (GCS) which describes the logical structure of the data at all the sites must also exist. Peer to peer systems have their own databases and application on any site. Each site is not only client but also server.

3.3 Fragmentation

Fragmentation consists of breaking a relation into smaller relations of fragments and storing the fragment, possibly at different sites. Data fragmentations are the techniques that are used to break up the database into logical units, called fragments, which may be assigned for storage at the various sites. Database is partitioned into disjoint fragments, each fragment assigned to one site.

There are three types of fragmentation:

(1) Horizontal Fragmentation

(2) Vertical Fragmentation

(3) Mixed or Hybrid Fragmentation

This system is implemented using vertical fragmentation.

3.4 Horizontal Fragmentation

Horizontal fragmentation consists of a subset of the tuples of a relation. The tuples that belong to the horizontal fragment are specified by a condition on one or more attributes of the relation. Horizontal fragmentation divides a relation "horizontally" by grouping rows to create subsets of tuples, where each subset has a creation logical meaning. Horizontal fragmentation allows parallel processing on fragments of a relation and allows a relation to be split so that tuples are located where they are most frequently accessed [2].

3.5 Vertical Fragmentation

Vertical fragmentation is to partition a relation into a set of smaller relations so that many of the user applications will run on only one fragment. Each site may not need all attributes of a relation, which would indicate the need for a different type of fragmentation [2]. Vertical fragmentation divides a relation “vertically” by columns. A vertical fragment of a relation keeps only certain attributes of the relation. When all of the attributes are not needed to store, vertical fragmentation can be used. This vertical fragmentation is not quite proper because, if the two fragments are stored separately, it can not be put into the original tuples back together, since there is no common attributes between the two fragments. It is necessary to include the primary key or some candidate key attribute in every vertical so that the full relation can be reconstructed from the fragments.

The vertical fragmentation schema satisfies the correctness rules:

Completeness: If relation is decomposed into fragments, each fragment that can be found in original fragment must appear in at least one fragment.

Reconstruction: The book information relation can be reconstructed from the fragments.

Disjointness: The three fragments are disjoint, except for the primary key, bId, which is necessary for reconstruction.

3.5.1 Advantages of Vertical Fragmentation

Vertical fragmentation allows tuples to be split so that each part of the tuple is stored where it is most frequently accessed and allows parallel processing on a relation. Tuple-id attribute allows efficient joining of vertical fragment.

4. Proposed System Design

The overview of the proposed system is described in figure 1.

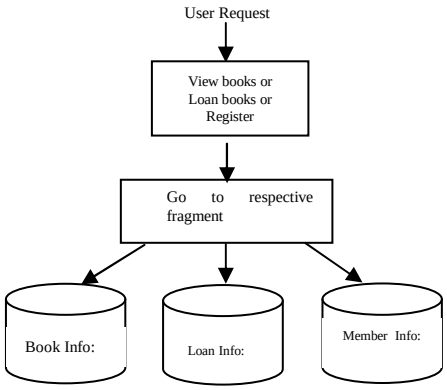

Figure 1. Overview of the proposed System

Book Database includes book information table, member information table and loan information table. Book information table is own table of book database. In addition, the member table and the loan table are replica tables. Book information table include bookId(string), Name(string), Category(string), Author(string), Date(date/time) and Edition(string). The original table divides two vertical fragment tables based on bId(primary key). Member table consists of memberId, password, name, class and rollno. Loan table has user events such as rent transaction and return transaction. If user rent the book, the tables in book database will interact with each other, book information table, member information table and loan information table. A Distributed Database System is a system, which is fragmented or replicated on the various configurations of hardware and software located at different sites within on organization.

In the proposed system, vertical fragmentation is used to fragment the library system in distributed database. In this system, book information, member information and loan information are vertically fragmented and stored at three different sites. The main tasks of this system are to view books, to loan books and to register. The system performed based on the user choosed task, go to respective fragment. If the user chooses register, the user must fill data in the registration form and record in the fragmented member database. The user can search book information by using the author name or book title. This system will extract the book information from the fragmented book database and will show to the user. The member can rent the book by using the memberId , password and bookname. The member can returned the book by using memberId, password and bookname. Any late fees would be computed by difference between due date and returned date.

5. Implementation of the library system using vertical fragmentation

The data flow diagram of the system is in figure 2. All users can view book information in the library. And then, the system allows the user to register as a memeber in the library system. If the user want to register, the user must fill data in registration form and will store in the member database. If the user known the book title or author name , the system will show details book information.

The system allows the member to rent the book and return the rented book. The system, book information, loan information and member information are vertically fragmented and stored at three different sites. This system is also proposed as a peer to peer distributed database.

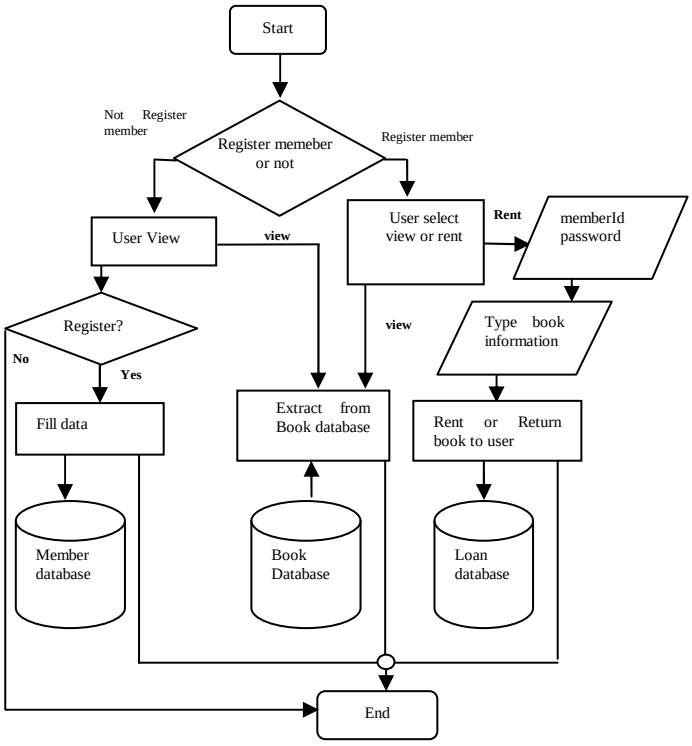

Figure 2. Data Flow Diagram of the system

The example of book information using vertical fragmentation is shown in table1, table2 and table3. The book information is vertically fragmented as two tables. In book information table, bId is the primary key for all tables. The Original book information has following attributes (bId, Name, Category, Author, Date and Edition). It fragments the relation to put information in two vertical fragment tables for security concern.

Table 1. Original Book information table

bId	Name	Category	Author	Date	Edition
b001	Digital Fundamentals	Hardware	Prentic Hall	2001	Fourth Edition
b002	Operating System	Software	William Stallings	2005	Fifth Edition
b003	An Introduction to Database System	Information System	C.J.Date	2000	Seventh Edition
b004	JavaScript	Programming	John Pollock	2004	Second Edition

Table 2 . Vertical fragmentation of book information

bId	Name	Category
b001	Digital Fundamentals	Hardware
b002	Operating System	Software
b003	An Introduction to Database System	Information System
b004	JavaScript	Programming

Table 3 . Vertical fragmentation of Book information

bId	Author	Date	Edition
b001	Prentic Hall	2001	Fourth Edition
b002	William Stallings	2005	Fifth Edition
b003	C.J.Date	2000	Seventh Edition
b004	John Pollock	2004	Second Edition

The sequence diagram is shown in figure 3. The user must fill register form as a member in the library. User register is Ok, return successfully message to user. The user request book information, show from book information database. The member can rent and return the book by using memberId, password and bookname.

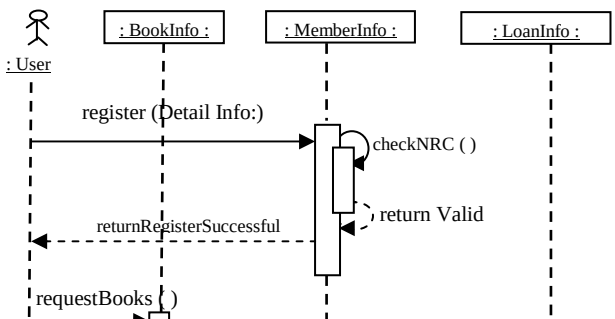

Figure 3. Sequence diagram of the proposed system

The GUI of the system is shown in figure 4 and 5. To rent book, the user must type member Id and password. The system check valid member or invalid member. If the member is valid, the user can rent the required books. If the total number of books is no more than the maximum number of books for a particular user type, the system allows the user to rent the required book. And then, the system show the successfully rented message.

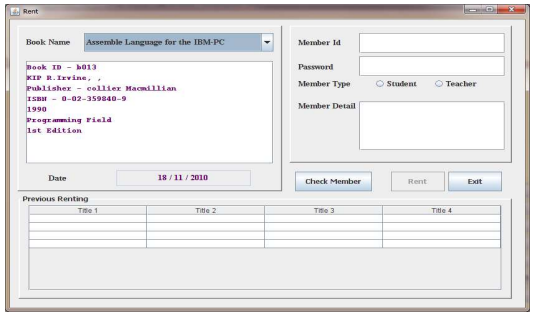

Figure 4. Rent the required book form

Figure 5. Successfully rented message form

6. Conclusion

The system is based on vertical data fragmentation on distributed database. The library will not allow to loan books from unauthorized user.

But all users at any site can request easily and rapidly without considering the actual location of the database. So, the system can save time because of processing the distributed database instead of the whole database. All distributed resources are available to everyone that reduce cost and time comparing to ordinary library system. This proposed system provides location transparency and fragmentation independence.

7. References

[1] "Adrian Runceanu " university constsitin Brancusi Targu-Jiu, Adrin r@utgjiu. Ro "Towards vertical fragmentation in distributed database.

[2] C.J, Date "An Introduction to Database System", seventh Edition, 2001. Addision-Wesley.

[3] David Bell and Jane Grimson : Distributed database systems. Reading, Mass: Addision-Wesley (1992). ISBN 0-13-738618-4

[4] "Horizontal Fragmentation in distributed object Database Systems" Ladjel Bellatreche an Ana Simonet Laboratorire TIMC-IMAG.

[5] Malinowski, E. an Chakravarthy, S. (1997), Fragmentation techniques for distributing object-oriented database, in D.W. Embley & R.C Goldstein, eds, 'Conceptual Modeling – ER '97', Vol . 1331 of lecture notes in computer science, springer, PP, 347-360.

[6] Navathe S. and Ra M. (1989) Vertical Partitioning for Database Design: A Graphical Algorithm. ACM SIGMOD, Portland.

[7] (TUFTF, KRISTIN; PAPADIMOS, VASSILIS; Distributed Query Processing and Catalogs for Peer to Peer System.)