

Building Distributed Database Application in Training Management

Nguyen Le

Faculty of Information Technology, HCM City University of Foreign Languages and Information, HCM, Vietnam

Abstract: Currently, the majority of the training management systems of the University and the College are built in the centralized database model, database is stored in a server and all operations as module registration, imported scores from the Training Department or course management of teachers, etc. are accessed directly to the same server and they can cause overload processor for the server, and have reduced performance and sometimes affected the safety of system. So in this paper, we research and build the distributed database model as well as proposal of the distributed database model in training management and synchronization solution.

Keywords: training management, distributed database, synchronization, replication, training department.

I. INTRODUCTION

Database technology has undergone a process of formation and development is quite long. Initially, the database system is often associated with the application, meaning that each application is defined and maintained their own data. Then database is managed centrally, many different applications can access data directly from the that centralized data. However, centralized databases also exist many defects, it may be mentioned that when the central storage and data processing have breakdown are whole system will shut down, or system may be congested when there are too many requests to retrieve data. The birth of distributed database was somewhat overcome the limitations of centralized database, the result of the integration of two approaches for processing the data: database technology and computer network technology. The majority of the training management systems of the University and the College are built in the centralized database model, database is stored in a server and all operations as registration module, imported scores from the training department or course management of teachers, etc. have accessed directly to the same a server and they can cause the overload processor for the server, and have reduced performance and sometimes affect the safety of system. In order to contribute to improving the efficiency in the training management of Universities and Colleges, we have focused on researching and building distributed database model and we also have proposed a distributed database model in training management and synchronizationsolution at Information Technology College - Ho Chi Minh city.

II. OVERVIEW OF DISTRIBUTED DATABASE

In this section, we will present the concept of distributed database and how to design distributed database model.

A. What is Distributed Database?

Define 1: A distributed database[1][2] is a collection of multiple, logically interrelateddatabases which are located in many sites via a computer network.

- Distribution: data must be distributed in many sites.
- Logical correlation: data in many places is shared to solve a problem.

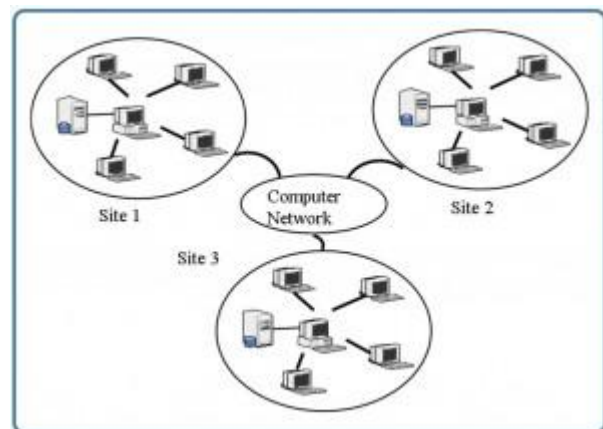


Fig.1. Example of a distributed database system [3]

Define 2: A distributed database[1] is a collection of multiple data which is distributed across different computers of a computer network. Each of the place of computer network is capable of handling the autonomous and can make local applications. Each place has to involve in implementation of at least one global application which the place can require data access to many other places by using the communication system.

B. Distributed Database Design

Two techniques are used in the design process is fragmentation and replication[1][4]. Fragmentation divides a relationship into independent pieces[1], each piece is distributed at various locations. When the distribution of the pieces on the location, it can be replicated or just maintaining a single replica.

Reasons for replication is to ensure the reliability and efficiency for data queries, specifically if more replicas of a data unit located in multiple locations[1], users still have the opportunity to access the data even the other location

of the incident. However, for the data update requests, the replication [1][4] has caused much trouble because the system must ensure that all replicas must be updated correctly.

Therefore, to come to a decision how fragmented, with replication or not, and the distribution of the pieces to the best position is the mathematical planning issues with NP-hard complexity [5] in order to lower the maximum processing and communication costs.

III. BUILDING DISTRIBUTED DATABASE MODEL IN TRAINING MANAGEMENT

A. Training Management Process

Nowadays, the credit student management system in a number of University and College is made as follows:

- Training Department receives the admission student, arranges the class and opens the module.
- Each student will receive an account and log in the system to register the module, search timetables, schedules, scores.
- Finance Division is based on the number of credits that student registered each semester to collect fees.
- The learning outcomes of students are managed directly by Training department: enter score, aggregate, classify and review each semester.
- Faculties directly manage the teaching progress of teachers and manage students in attendance, thesis, assigned instructors

B. Distributed Database Model

Distributed database system [1] should be distributed to Faculties and Departments of the College, each department will have a database containing data belonging to the range of their part and operate independently. In addition, system must have a dedicated database for students to see the score and register module through the Internet. Database at Training Department contains the entire database of other parts. To ensure the accuracy of the data, database in the Training Department, database in the parts and database for students to have online synchronization [6][7] mechanism via a two-way system. Depending on the size of the organization that using of the database and allocating server to each department is designed for fitting.

C. Distributed Database Design

In training management system, all data should be stored in Training department database, distributing data to the departments is done by dividing the data table of the overall relational data into independent tables for matching functions and duties of each department and then replicating them and positioning the replicas to the appropriate locations. Any changes to the data in the departments will be synchronized on Training department and all data changes in the Training department will also be synchronized to the relevant departments.

The problem here is why to use replica of fragments in departments without using separate fragments, if Training department needs any information of parts, it will link to

that department to retrieve data. Because replication issue will lead to excess data and can be wasted storage space and the risk of causing much conflict when updating data.

The answer to this problem is the data access and updating information in the Training department are made very frequently, while retrieving data from two or more pieces in multiple servers and then connecting them with very high cost, so performance is significantly reduced. Nowadays, the cost of storage space was no longer a matter of top priority but instead is the processor speed, the risk of conflict occurs when updating can resolve the Training Management System (will be presented in section E). Distributed database model in Figure 2, all data is stored in a database located in Training department server, Student and Faculty servers save one replica part of the database that allowing students and employees to see or update data.

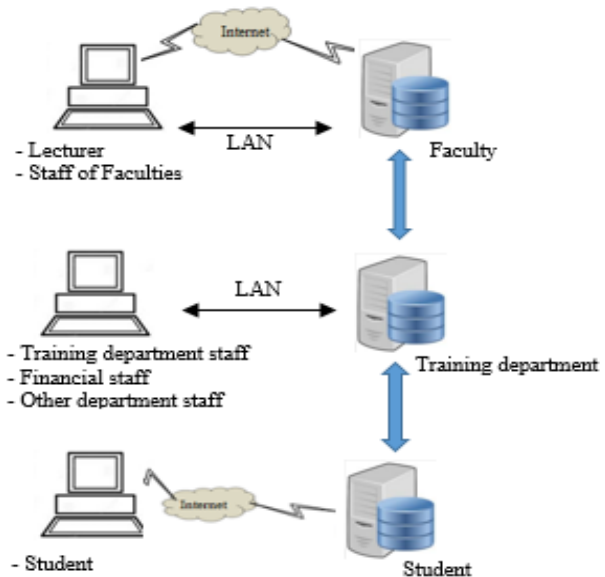


Fig.2. distributed database model of Training Management

D. Synchronization in Sql Server

The main components of Replication [6][7]

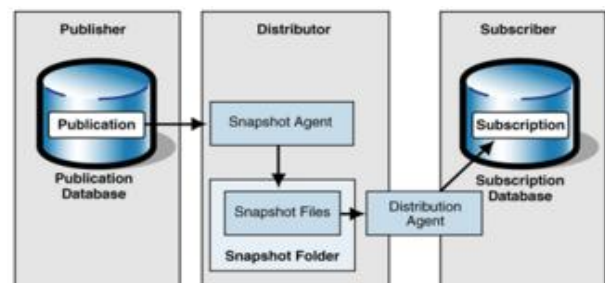


Fig.3. SQL Server snapshot replication [6]

- Publisher: server creates replica to another server. It decides which data will be replicated.
- Subscriber: server receives replicas and the data updating actions from the Publisher.
- Distributor: Server contains distributed database, is where to store and transfer replicas from Publisher to Subscriber.

In Training management system:

- Publisher: Training department server
- Subscriber: Student Server and Faculty Server

With these important data such as information on the score of the students are only allowed to update the Training department (Publisher), other parts (Subscriber) can watch but not allowed to update.

E. The Issue of Data Synchronization

One of the interesting issues of datasynchronization[8][9] is conflict, this problem occurs when two or more servers at the same time update the data on the same tuple. In distributed database system on training management, the server has the right to update the data: add, delete, edit. Server of Training department updates module, score, fees etc.; server of Faculty updates schedule of teachers, internship schemes for student et; server of Student updates module registration information. These rights are completely independent they are not overlapped thus conflicts occur rarely.

However, the case of conflict[9] is inevitable, use measures of conflict resolution in replication is supported priority for each server by SQL Server. If there is any conflict, it will select updated server with a higher priority. In the training management system, the priority for the Training Department.

IV. EXPERIMENTATION AND EVALUATION

A. Experimentation

Training management system of Information Technology College Ho Chi Minh City are using centralized database. To simulate distributed database, I use three computers connected over a LAN (Local Area Network), the first server computer represents Training Department, the second server computer represents Faculties and the third computer server represents Student.

TABLE 1 TEST DATABASE

Order	Tables	Attributes	Instances
1	tbStudent	35	5050
2	tbLecturer	44	157
3	tbClass	12	82
4	tbModule	21	3262
5	tbSchedule	19	2952
6	tbRegistration	11	14707
7	tbCurriculum	17	1102
8	tbScore	42	126725
9	tbMajor	16	20
10	tbStudentReviews	40	23687
11	tbTuition	37	22555

System consists one publication server and two subscription servers. Each subscriber has a database matching publication of the server functions. Database in three servers are shown in table 2.

TABLE 2 CONFIGURING REPLICATION IN SQL SERVER

Server	Replication	
	Local publications	Local subscriptions
Training department	Faculty publication Student publication	
Faculty		Faculty publication
Student		Student publication

Database of three servers are presented in table 3.

TABLE 3 DATABASE IN SERVERS

Database of Training department Server	tbStudent, tbLecturer, tbModule, tbSchedule, tbRegistration, tbScore, tbMajor, tbStudentReview, tbTuition , tbCurriculum
Database of Faculty Server	tbStudent, tbLecturer, tbModule, tbSchedule, tbScore, tbMajor, tbStudentReview, tbCurriculum
Database of Student Server	tbStudent, tbModule, tbRegistration, tbScore, tbMajor, tbStudentReviews, tbTuition

The following is the results of database synchronization between three servers of experiments

-A total of 102,000 records in the Faculty server are synchronized on the Training department server within 00:3:45 (Fig.4.a).

- A total of 214,100 records in the Training department server are synchronized on the Student server within 00:2:29 (Fig.4.b).

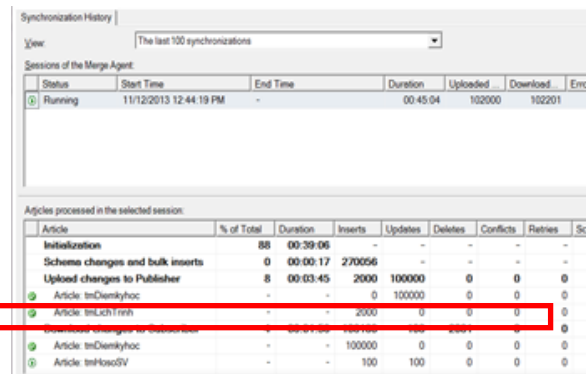


Fig.4.a. Data synchronization results between servers

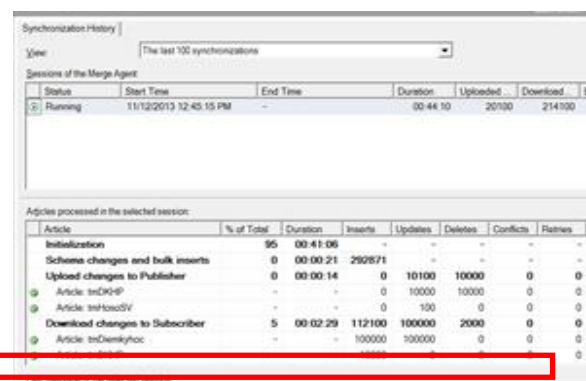


Fig.4.b. Data synchronization results between servers

B. Experimental Evaluation Result

The accuracy of the synchronization is 100%, which means that data between the Training department server correctly synchronized to the Student server and the Faculty server and vice versa. To test the accuracy, we used “tablediff” utility [10] to compare data in two tables at the servers after each synchronization time. Example, to test the similarity between the two tables with the same name tbStudent on two servers must execute as follows:

```
"C:\Program Files\Microsoft SQL Server\100\COM\tablediff.exe"
```

Source Options

```
-sourceserver          Training_Depart_Server  
-sourcedatabase       Database_Depart_Training  
-sourcetable          tbStudent
```

Destination Options

```
-destinationserver    Student_Server  
-destinationdatabase Database_Student  
-destinationtable     tbSudent
```

V. CONCLUSION

In this paper, we have proposed distributed database model in training management in order to overcome the limitations of centralized database model, improving the efficiency in the training management process for Universities and Colleges. In the future, we will aim to put the model into application at Information Technology College Ho Chi Minh City, detect and solve problems in the synchronization process as well as techniques to ensure data security in distributed database systems

REFERENCES

- [1] M.Tamer Ozsu and Patrick Valduries, Principles of Distributed Database Systems, New York Dordrecht Heidelberg London: Springer, 2011.
- [2] M.T. Ozsu and P. Valduriez, “Distributed database systems: where are we now?,” Computer (Volume:24 , Issue: 8), pp. 68-78, 1991.
- [3] Chapter 6 classification of database management systems. Available: <https://opentextbc.ca/dbdesign01/chapter/chapter-6-classification-of-database-systems/>.
- [4] B. Ciciani, D. M. Dias and P. S. Yu, “Analysis of replication in distributed database systems,” IEEE Transactions on Knowledge and Data Engineering (Volume:2 , Issue: 2), pp. 247 - 261, 1990.
- [5] F. Dehne, J.-R. Sack and R. Tamassia, Algorithms and data structures, Springer, 2001.
- [6] Implementing Master-Subordinate Snapshot Replication Using SQL Server. [Online]. Available: <https://msdn.microsoft.com/en-us/library/ff648057.aspx>.
- [7] S. P. Paul, Pro SQL Server 2008 Replication, Printed and bound in the United States of America , 2009.
- [8] S. Kublera, K. Främlinga and W. Derigentb, “P2P Data synchronization for product lifecycle management,” Computers in Industry, p. 82–98, 2015.
- [9] J.-Y. Lohrera, D. Kaltenthalera, P. Krögera, C. v. d. Meijdenb and H. Obermaiere, “A generic framework for synchronized distributed data management in archaeological related disciplines,” Future Generation Computer Systems, p. 558–570, March 2016.
- [10] Tablediff Utility. [Online]. Available: <https://msdn.microsoft.com/en-us/library/ms162843.aspx>.