Towards the Application of Distributed Database in University MIS

Xiaoyong Wang and Yaofeng Fang

Computer Science Department, Junior College of Zhejiang Wanli University, Ningbo, 315101, R.P. China

Abstract. The paper explores some crucial technologies of Distributed Database in University MIS, such as the distribution and replication of data, and studies its feasibility on the basis of campus networks.

Keywords: Distributed database, University MIS, Data distribution, Coherence and parallel of data, Data copy.

1 Introduction

Distributed Database is a set of data that belong to a complete unit in logical sense but are distributed to various nodes in physical sense, distinct from the traditional one which we call Centralized Database. Compared with Centralized Database, Distributed Database is characterized by distribution and logic coordination: the former refers to the arrangement that the data are distributed to some data subsets stored at various nodes with overall consideration, instead of being stored in a particular computer; the latter refers to the arrangement that the data subsets at various nodes are regulated separately by strict rules but belong to an integral unit by logic [1].

Therefore, the distribution of Distributed Database makes it different from Centralized Database, its logic coordination makes it distinct from Decentralized Database through network connections and it is superior to the Decentralized Database concerning the data's independence. Nevertheless, Centralized Database is the basis of Distributed Database and the network provides the indispensable environment for Distributed Database [2].

The University Management involves management of various aspects, the students management, the teaching management, the scientific research management, the management of human resources, the assets management, to name a few. The current situation in many universities where computers are utilized in their management is that each administrative division explores its own management system such as the students grading management system explored by teaching administration, the personnel management system by the personnel division, the library management system by library and so on, which operates mainly on the basis of the personal computer (PC) or the local area network (LAN), and the resource share and information exchanges visits can not be achieved among different divisions. As the modernization of university management has been advancing, the traditional mode of management, while abusing many resources, is inadequate to meet the requirement of open information management, resource share and information exchanges visits [3].

O. Castillo et al. (Eds.): Theor. Adv. and Appl. of Fuzzy Logic, ASC 42, pp. 513–519, 2007.
springerlink.com

In the past few years, the campus network has been utilized as the basic facilities of university administration, whose application has greatly promoted the modernization of education and provided the solid foundation for constructing the information college as well as a new network platform for the exploration of university MIS with Distributed Database as its core technology.

2 Some Crucial Technologies Concerning the Application of Distributed Database in the University MIS

Next, Some crucial technologies concerning the application of Distributed Database in the University MIS and the realization of these technologies are to be discussed as follows, supposing the operating system is WindowsNT and the network relation type database is SQL Server of Microsoft.

2.1 Drawing Up the Scheme of Data Distribution

Like Centralized Database, Distributed Database consists of two parts: the integral of all necessary applicable data, known as physical database, which is the principle part of the Distributed Database, and the definition of data structure, the distribution of the overall data and its description, known as descriptive database. The data of Distributed Database can be divided into the partial data provided for the local application of a particular node, and the overall data that are involved in the overall application and can be visited by many nodes as they are stored in various nodes physically. The data distribution of Distributed Database are not stored in a particular node, but partitioned into some logical segments according to different requirement, which are distributed to the various nodes through some strategies. As the data distribution affects the capability, the reliability and efficiency of the whole system, it is of extreme importance to decide on the proper strategy of data distribution [4] [5].

2.1.1 Models of Data Distribution

The data distribution of Distributed Database can be roughly divided into four types, the centralized model, the intersected model, the copied model and the composite model. The characteristics and demerits of each model are illustrated in table 1.

2.1.2 Elaboration on the Project of University Data Distribution

The flow of campus data information nodes are time dependent as the information exchange visits mainly focus on some periods of each semester and a large amount of information consultation are usually limited within each division. In accordance with this practical situation, we can employ the distribution strategy with slight variation on the basis of the composite model; namely, all data are divided into a number of subsets according to the operation requirement of each division, each of which is stored in a particular database server of the LAN where different divisions are located. The copes of all subsets for resource share are stored in the central database server of the campus network, thus decreasing the complication of data visits of inter-divisions. The concrete solution is that the database server of each division works as a network node of LAN, with SQL Server7.0 installed, in which the local data is stored;

the interactive visits among different divisions are made through the visit to the data copy stored in the central databank server, and the coherence is maintained through distributed transaction or copy. Besides, the central database server needs to register to local servers of various divisions so as to make the visit of each division to the central server available. If this strategy is taken, on one hand, the load of local servers can be relieved and the interactive visits among different divisions become less complicated; on the other hand, the whole system can still function well in case of the malfunction in a certain division, thus the reliability of the system is improved, since there is copy in the central database server for each local database, the prompt recovery of data through the copy of SQL Server is possible if the malfunction at a particular node leads to the destruction of data.

Table 1. Comparison among various data distribution models

type	characteristics	demerits
The centralized model	As all data segments are stored in the same node, the easy operation and management of data as well as the coherence and completion of data can be guaranteed.	The reliability is not high since the node where the data are stored is susceptible to overburden and occurrence of bottleneck, leading to the collapse of the whole system in case of malfunction
The intersected model	All data are integrated and intersected to be some logic segments, each of which is allocated to a particular node. Therefore, the memory equipment of each node can be fully exploited and the memory space becomes larger	There is difficulty in maintaining the coherence and completion of data since data are distributed to various nodes
The copied model	As the overall data have many copies and there is a complete data copy at each node, the reliability of system is higher, the response is faster and the recovery of databank is easier.	It costs more to maintain the simultaneous data revision of all nodes; there is more redundancy of data, and the memory space is affected.
The composite model	All data are divided into a number of subsets. Each one is stored in a particular node and none of the nodes store all data, thus combining the merits of both intersected model and copied model, and resulting in the more efficient operation.	As it is the combination of the intersected model and copied model, it inherits the demerits of both models.

The blue print of data distribution that is feasible in accordance with the characteristics of university Distributed Database is illustrated in the following Fig. 1.

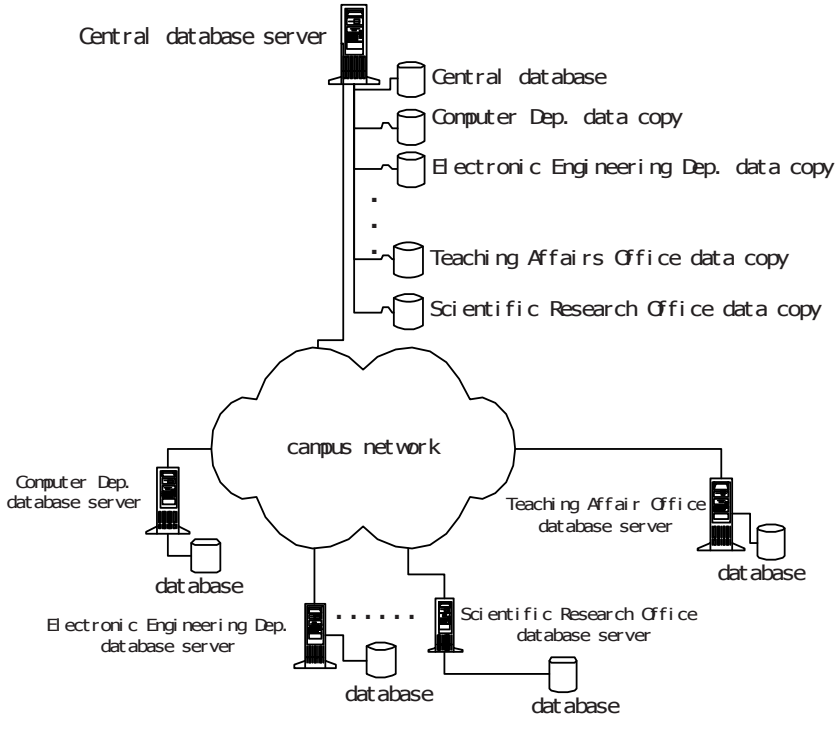

Fig. 1. The blue print of the Distributed Database applicable in university MIS

2.2 Coherence And Parallel of Data

In the MIS that employs Distributed Database, transaction is an array of distributed processing and the data to be processed are likely to be distributed to various nodes. As the distribution of data is transparent to users who often make overall demands on data likely to be stored in any of the nodes comprising the Distributed Database, problems tend to arise when quite a few transactions are operated in parallel, despite the improving efficiency of the system. There are three problems brought about by transaction parallel operating: loss of updating, incoherence of data processing, and dependence on the unsubmitted updating. In case the problems can not be properly solved, the completion and coherence of the data are ruined. In order to maintain the coherence and parallel of data, lock scheme contained in the MIS of SQL SERVER can be applied to close off the data being revised by a transaction in case other users should visit the incoherent data.

2.3 The Scheme of Data Copy

Another problem that should not be overlooked in the date distribution to various nodes and the processing of distributed data, is how to maintain the data synchronization of

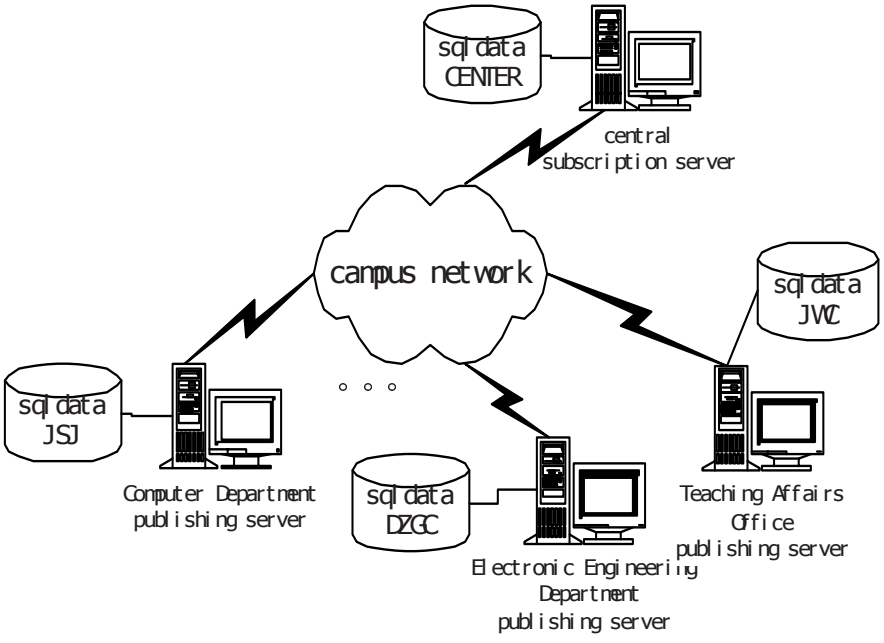

Fig. 2. The blue print of the publishing/ subscription model in university distributed MIS

various databanks. For instance, suppose the courses information of a specialty in the computer department has been revised, but the revision has not been reported to the master database server, consequently, when another division pays visit to the master database server, it only get the outdated courses information, which is inconsistent with the revised version owing to the failure of the data synchronization between local databases and the master one. The reprography service provided by SQL Server can ensure the replication of information among different databank servers, capable of acknowledging revision and sending it to the other systems including long-distance systems.

The reprography model of SQL is made up of three physical layers – subscription layer, publishing layer and distribution layer, which are closely linked with each other to constitute a functional replication model on the principle of "you publish and I subscribe" that are feasibly observed and applied. A brief introduction to these three layers is as follows.

1. Publishing layer provides information. When a system is designed to provide information to other systems through the reprography, it becomes a publisher and the database server where it is located is the publishing server, providing information for other systems and that information can be regarded as a publishing with all entries, ranging from the whole content of the databank to a record or a search result.
2. Subscription layer receives information. The users should turn their attention to subscribers and have them aware where to find received information after the information entries are published. Subscribers refer to the information receivers who need to install the connection with the distribution servers.

3. Distribution server with its databank working as a connector between the publishing server and the subscription server, is the resource of information. The users, upon installation, can choose the local servers or the long-distance severs as their distribution servers, where all information related to the publishers and subscribers are stationed.

Above all, the publishing server and distribution server should be installed in order to realize the replication of Distributed Database of university MIS. The two servers are to be installed in different computers when there are a lot of subscriptions; otherwise, it's more proper to put them together (see Fig. 2).

2.4 Technology of Database Accessing of MIS

Next, the technology of Database Accessing of MIS is to be discussed with an example of the MIS subsystem applied in the Computer Department, which employs PowerBuilder6.5 as the client's developing tools on the basis of the distributed MIS of the campus network. Making use of transaction objects to make visit to the database, the clients program of PowerBuilder is directly connected with the sub-databases through the specific interface. A detailed description of how users of the MIS subsystems connect with and make visit to various databases upon registration is given in the following script.

```
//to read from the init files the parameters connected
to the databank and to connect with the database
through the specific interface.

disconnect;

SQLCA.DBMS=ProfileString(".\jsjprofile.ini","Database",
"DBMS", " ")

SQLCA.Database=ProfileString(".\jsjprofile.ini","Databa
se","DataBase", " ")

SQLCA.LogID=ProfileString(".\jsjprofile.ini","Database"
,"LogID", " ")

SQLCA.LogPass=ProfileString(".\jsjprofile.ini","Databas
e","LogPassword"," ")

SQLCA.ServerName=ProfileString(".\jsjprofile.ini","Data
base","ServerName", " ")

SQLCA.UserID=ProfileString(".\jsjprofile.ini","Database
","UserID"," ")

SQLCA.DBPass=ProfileString(".\jsjprofile.ini","Database
","DatabasePassword", " ")

SQLCA.DBParm=ProfileString(".\jsjprofile.ini","Database
","DbParm"," ")

SQLCA.AutoCommit=true

CONNECT;//connected to the above-mentioned parameters

……
```

JSJprofile.ini, an init file in the script, is a structured text similar to INI of windows and made up of many sections each of which consists of some variable Assignment Statements. Through Profile String parameters provided by PowerBuilder, each attribute value of SQLCA can be retrieved from the structured text.

3 Conclusions

It has been a big game to explore the large scale MIS on the basis of Distributed Database operating in the campus network. This mode of database application system that employs the distributed computing and Distributed Database technology ensures more direct, prompt and convenient data management in various divisions that are widely separated from each other. In addition, the MIS on the basis of Distributed Database is easy for system expansion and it is a new task to take the campus network as its operation environment. However, owing to the complication of the distributed environment, a great many difficulties will occur in the process of exploration concerning the maintenance of the coherence, completion, parallel and security of data in Distributed Database. The paper has probed into some crucial technologies and the feasibility of distributed database in University MIS. With the development of Distributed Database technology and the campus network, the explorations in this respect will surely flourish in near future.

References

1. F. Bennour, Un Gestionnaire de Structures Distribuées et Scalables pour les multiordinateurs Windows: Fragmentation par Hachage, PhD thesis in French, Paris Dauphine University, 2000.
2. M. Blaum, J. Brady, J. Bruck & J. Menon, EVENODD: An Optimal Scheme for Tolerating Double Disk Failures in RAID Architectures, IEEE 1994.
3. J. Blomer, M. Kalfane, R. Karp, M. Karpinski, M. Luby & D.Zuckerman, An XOR-Based Erasure-Resilient Coding Scheme, ICSI Tech. Rep. TR-95-048, 1995.
4. L. Hellerstein, G.A. Gibson, R.M. Karp, R.H. Katz & D.A. Patterson, Coding Techniques for handling Failures in Large Disk Arrays, Algorithmica, 1994, 12, pp.182-208.
5. M. Ljungström, Implementing LH*RS: a Scalable Distributed Highly-Available Data Structure, Master Thesis, Feb. 2000, CS Dept., U. Linkoping, Suede.