Distributed Healthcare Database Integration for Supporting Agile Decision Making

Ruey-Kei Chiu
Department of Information Management, Fu-Jen Catholic University
Email: im1004@mails.fju.edu.tw

Chien-Lung Chan
Dept. of Information Management, Yuan Ze University
E-mail: clchan@im.yzu.edu.tw

Chi-Ming Chang
Center for Disease Control of Department of Health, Taiwan
E-mail: kevein@cdc.gov.tw

ABSTRACT

We present an effective data model designed for designing an incrementally centralized national database for initially distributed immunization databases of CDC in Taiwan for supporting increasingly agile demand of decision making. Initially, the vaccination records for each individual are separately stored in the database of each county/city bureau of health. The main objective of this research aims to proposed an empirical data integration model and refresh Model for these distributed databases so that the widespread national vaccination data can be effectively aggregated into a centralized national immunization database. Leveraging this construction, the information for nationwide vaccination daily operation, data management, and decision making of the Center for Disease Control (CDC) of Taiwan can be more effectively supported. The system implementation and evaluation shows that the data model proposed by this research can truly reflect the requirement of each-level users.

1. INTRODUCTION

Since the early 1980s, the Department of Health (DOH) of Republic of China (ROC) started to build a national Health Information Network (HIN) shown in Figure 1, which is a nationwide broadband network system to support the health care administration and streamline the information flow in carrying out the affairs of national public healthcare. Through this network, the nationwide medical information may be more efficiently exchanged and delivered through this network. In the early years of 1990s, a stand-alone DOS-version Primary Health Information System (PHIS) was built and then started to independently operate on nationwide 357 county's or city's public health centers. It took almost five years to complete this implementation and popularize its operations to nationwide [1].

The main disadvantage of this stand-alone system is that without a central database and on-line system for data management and transition control, the vaccination records of each individual not only was unable to be refreshed in time but also could not be used and on-line transferred directly through the network. As a result, it is hard to provide the real-time vaccination records for each individual moving from one place to another or choosing different clinics/hospitals to get access to have vaccine inoculation. Meanwhile, there was no up-to-date information could be applied to effectively support the vaccination operation and management [1].

Starting from 2002 the Division of Immunization, Government of ROC, initiated a new project to redesign the system for the next generation immunization management and information system. The new system was formally named as the National Immunization Information System, which is shorted as NIIS. The new project was planned and set forth into official action by the Division of Immunization in May of 2001. The new system was redesigned to include all the functions of immunization subsystems have existed in its DOS version and many new functions were extended for new and future requirements. Especially, it is implemented in a new data model in which each individual records of vaccination was directly retrieved from bureau's centralized database during the operation of vaccine inoculation. The transaction record is then uploaded and restored back to the centralized bureau database through HIN or the Internet. The centralized database, which is installed at each bureau of health, is intentionally designed for centralized store, operation, and management for its entire vaccination records. By using this centralization for NIIS operational data to the bureau's database, during the operation of daily vaccine inoculation, the district health centers underneath a bureau can directly access to the bureau's database through HIN. To handle the process of vaccination shifting and homing, two or more bureaus of health involved in this process can exchange the related data in real time through bureau's databases and the central exchange center in Health Information Network. The NIIS database system with its network operational environment is shown in Figure 1 [1].

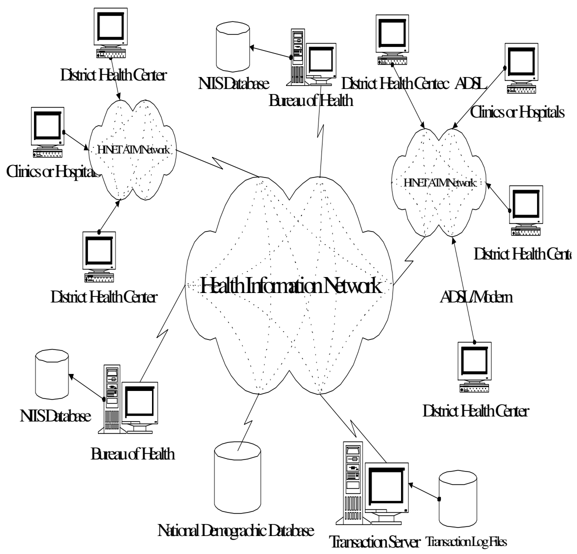

Figure 1. NIIS Database Framework and Operational Environment

With the completion of the installation of NIIS at the 375 district health centers nationwide in June 2003, the CDC further created a research project intending to design an empirical and feasible data model to aggregate its nationwide immunization databases scattered in each bureau of health to the central site of CDC in order to meet the user requirement in different levels from local distributed centers to Center of Disease Center and Department of Health. The challenge of this aggregation of distributed and great diversity of distributed databases is highly overcome in terms of technologies, data framework, and integrated process. This paper presents the data model to effectively support this aggregation.

2. PRIOR RELATED RESEARCH AND WORKS

2.1 THE APPROACHES OF DISTRIBUTED DATABASES AGGREGATION

Connolly and Begg defined that distributed database system (DDBS) was a logically interrelated collection of shared data, physically distributed over a computer network [2]. Franklin, et al. noted that distributed database system could be viewed as a single database logically, but physically located in different sites [3]. Users can query and access data through network. Each database in the site is in control of its own database management system (DDMS) with autonomy, and it needs to keep the data in consistency in the distributed database system. In addition, distributed database management system has to provide the functions of query optimization, data transparency, transaction management, recovery control, concurrency control, and data security to maintain the database system in good performance state for database users and administrators [4].

Through a serial survey of various integration mechanisms developed in 1990s, Thomas suggested a well-designed integration mechanism was capable to handle a seamless interoperation of widespread enterprise data located in several sites by the aids of database systems, operating systems and data communications. Thomas, et al.'s research also indicated that the key to manage the integration of different database systems was to have a standard language and protocol to access remote distributed databases.

Sheth and Larson proposed an integrated Model that could integrate database systems located in different sites with autonomy and heterogeneity for building the integrated enterprise database [5]. They indicated that there were two approaches frequently used to build integrated databases - bottom-up and top-down. The approach of bottom-up can transfer the database schemas located in different sites for being referenced as a part of the schema of integrated database. Afterwards, the database administrator has to define an integrated database schema. Once the integrated database schemas is established, the database designer needs to set a filtering and transforming mechanism to conduct the data extraction and transformation from the external databases located in different sites and load the data into the integrated database. Therefore, the integrated database system Model proposed by Sheth and Larson are worthy to be applied for establishing an integrated enterprise database.

3. THE DATA FRAMEWORK FOR NIIS DATABASE AGGREGATION

In this section, the data framework of this empirical research particularly designed for building the aggregated and centralized NIIS national database is presented and introduced.

3.1 THE DATA FRAMEWORK FOR CENTRALLY AGGREGATION

The NIIS Central Database is actually an aggregated database which data are selectively replicated from all bureau's databases of 26 counties and cities nationwide. By doing so, both the data requirements to support the national immunization management, operation, and research and the data accessing performance and storage cost between central and local sites can be fully considered. The NIIS data framework is shown in Figure 2.

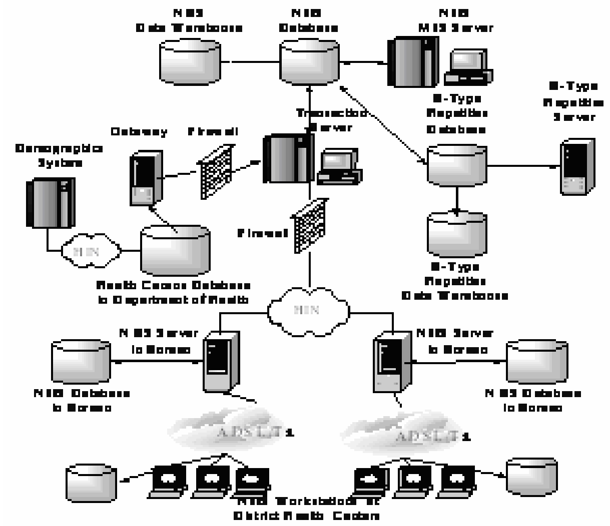

Figure 2. The NIIS Data Framework

3.2 SOME POTENTIAL BENEFITS OF BUILDING CENTRAL DATABASE

On the whole, it indeed provides an environment with high quality and efficiency of information management services. Some of potential benefits from these perspectives are described as follows.

1) Provide a unified information portal to apply the vaccination certificate.
2) Provide an integrated and informative statistics and analyses with respect to national vaccination affair.
3) Provide an effective central management and control of vaccine inventory.
4) Enhance the completion rate of NIIS vaccination.
5) Reduce data reentry workload in each local health center and bureau.
6) Support variety of data and information types.
7) Provide the data for the Control and prevention of diseases.
8) Provide data for the analysis of individual behavior of inoculation.

4. THE REFRESH MODEL FOR NIIS DATABASE AGGREGATION

In this creation, Microsoft Windows Server 2003 is employed as a database management platform to execute this implementation. We propose a NIIS central database refresh model to effectively maintain the data consistency between NIIS central database and the databases residing in all bureaus of health.

4.1 THE REFRESH MODEL

The refresh Model with its major components is designed and illustrated in Figure 3. As we can see from Figure 3, the central database system uses the central log file to record the data for refreshment. Similarly, at each bureau site of database system a bureau's log file is installed and maintained to keep the transaction records which have currently been changed but have not yet been refreshed to the central database.

Two major types of refresh modules are implemented for this Model. One type of Refresh module (type A) is installed at each bureau site as data collection agent which is periodically triggered by another type of refresh module (type B) which is installed at the central site to retrieve the transaction records from each bureau's transaction log file. The transactions retrieved are then passed to the data transformation service (DTS) which is also activated by refresh module B. The data transformation service then takes the responsibility of conducting the data extraction, transferring, and loading between two log files. In addition, refresh module A is also responsible for convert each transaction into XML format and log it into bureau's log file. And refresh module B is also responsible for reading the transaction records from central log file and activating the refresh stored procedures to conduct the precise refresh process according the transaction type recorded in each transaction record.

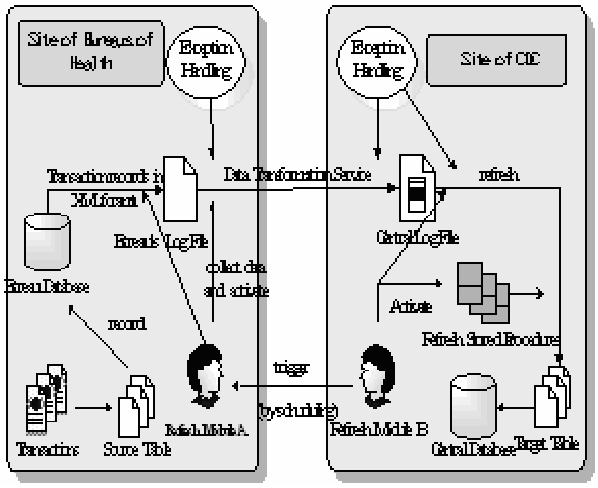

Figure 3. NIIS Central Database Refresh Model

The Model of Figure 3 can be further illustrated as a simplified process with its flow of data shown in Figure 4.

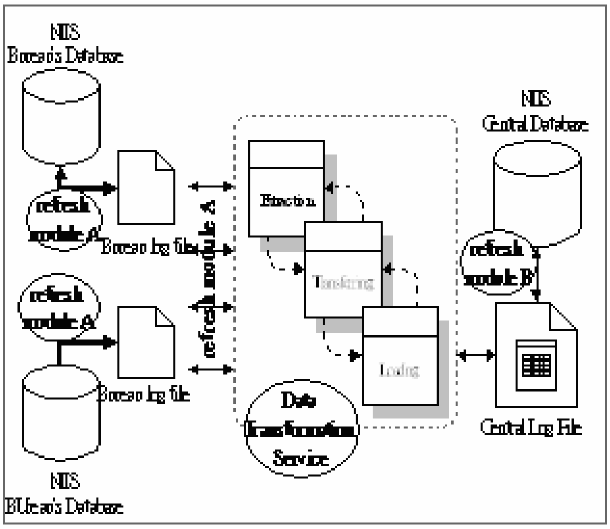

Figure 4. The Refresh Process with Flow of Data

As we can see in Figure 3 and 4, the refresh module A at the bureau site will also play the role of storing the transaction records into bureau's log file represented in XML format for each occurrence of transaction. There are two transaction records stored for each transaction. One is for recording the data before the transaction occurrence. The other is for recording the data after the transaction occurrence. Both are stored in XML format in order to be able to be transmitted over an open network during the refresh. The contents of each stored transaction record include transaction data, transaction time, table name, transaction type (i.e., insertion, deletion, update).

Because of the restriction of paper length, we omit algorithms corresponding to this refresh process.

5. BENCHMARK AND EVALUATION

In order to evaluate the efficiency of this research, we also conduct a benchmark test to compare with the Microsoft SQL Replication in an experimental environment of. The environment is setup as a client-server model. Each client has same database schema and number of data records which emulate NIIS database at the client site. The database schema in the server is mapped from the client and data records are aggregated from all clients. Hence the server's database has records number equal to the summation of all clients. The Microsoft SQL Server mainly handled for its replicated database synchronization among different replication sites by its Transact-SQL. It includes two main modes of transactional consistency. One is immediate transactional consistency which the data consistency is done by right after the transaction occurrence. The other is called the latent transactional consistency which can allow delay between the transaction occurrence and the execution of database synchronization. In this research, the refreshment mechanism between the bureau's databases and central databases is implemented as a mode of latent transactional consistency. So, we only setup this benchmark test with Microsoft SQL on the mode of latent transactional consistency.

We attempt to compare two execution times that involves in two major areas of conducting the database refreshment. One is " transaction data collection at each bureau site". The other is "central database refreshment". The transaction record number from 50,000, 100,000, 150,000 and 200,000 are selected at each client site for these two tests in order to compare the average refresh times between the use of replication functions provided by SQL Server and the database refresh Model proposed in this research. The results show that the average execution time of MS SQL Server is better than our research model when records are less than 100,000. However, when records are more than 100,000, our model shows have a better average execution time than that of MS SQL Server. Nevertheless, there are no significant differences between MS SQL Server Replication and the methods developed by our research model.

In our research model, the process of the data collection of bureau's transactions is triggered remotely by the central refresh module. The transactions are transferred and stored in XML format in order they can have high transportability over different database platforms. Therefore, during the central database refreshment, the refresh modules also has to interpret each transaction record retrieved in XML format retrieved from the transaction log files before they can be used for the database refreshment. Therefore, in comparison with SQL Server Replication, the logic of our research model is apparently more complicated but contrastively more open and transportable. Surprisingly, although the execution time of our research model would be worse than SQL Server Replication when the record number is low, whereas when the number is over a threshold our research model shows can have better results.

6. CONCLUSION

In this research, we present a data framework with the refresh mechanism to support the data aggregation of modeling and implementing an integrated NIIS central database of Taiwan from its exiting distributed databases. The data in the central database are selectively replicated from distributed bureau databases nationwide. Similarly, the central database schema is mapping from bureau's database schema. The refresh Model is designed and implemented on an assumption which the platforms of database management system used to support the implementations of central database and each distributed bureau's database are different. The method of latent transactional consistency is applied for designing this refresh system because the immediate synchronization between the central database and distributed ones are not required in NIIS.

The system benchmark is tested against the distributed database aggregation and update functions provided in MS SQL. The results show that the refresh Model developed in this research is indeed practicable and feasible.

References

1. Yuan, V.J., Chang, C.M., Chiu, R.K., Chan, C.L., and Chen, S.C., " National Immunization Information System", Proceedings of the 2003 Medical Informatics Symposium in Taiwan Conference, Taipei, Taiwan, October 2003.

2. Connolly, T. and Begg, C., Database Systems: A Practical Approach to Design, Implementation and Management, 2nd edition, Addison-Wesley, 1998.

3. Franklin, M.J., Carey, M.J., Livny, M., "Transactional Client-Server Cache Consistency: Alternatives and Performance", ACM Transactions on Database Systems, Vol. 22, No. 3, 1997, pp. 315-363.

4. Date, C.J., "An Introduction to Database Systems", 8th Edition, 2002.

5. Sheth, A. P. and Larson, J. A., "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases", ACM Computing Surveys, Vol. 22, No. 3, 1990, pp. 183-236.