International Journal of Information Technology and Knowledge Management
July-December 2010, Volume 2, No. 2, pp. 443-445

ALGORITHM FOR SYNCHRONIZING DATA AMONG BIOLOGICAL DATABASES

Ateet Mehta[1] & Bankim Patel[2]

Biologists increasingly use public databases to access data and to download it into their private databases. These public databases contain important experimental data that the availability of such database is of the highest priority. Hence having more than one instance of the same database at a different site is very essential and common in practice in order to provide maximum availability. In order to provide maximum availability and also for consistency, biological databases need to be always in sync with each other. Besides these, many biologists often share data among their private databases through flat files and such files are manually loaded into their private databases. There arise a need to have automatic mechanism to synchronize data among different databases in order to provide availability and consistency among different database. Further data replication solution provided by specific database vendor is expensive and it adds an extra cost when source and target database system are heterogeneous. In this paper, we propose an algorithm which uses publisher and consumer model using message queue consisting of XML messages to synchronize database.

Keywords: Data Capture, Data Apply, Message, Event, Synchronization, Replication, XML

1. Introduction

Biologists increasingly use public databases to access biological data. With almost every new scientific publication in genetics and related sciences, a new sequence is added and the rate at which the data is accumulating is on the rise. [1]. Currently there are global databases like NCBI, GenBank, and SWISS PROT storing such biological data and these databases need to be in Sync. The increasing number of biological databases, the emergence of new types of data that need to be captured, as well as evolving technologies, methods and biological knowledge add to the complexity of overall data management. [2]. Further many biologists and researchers share information among each other every time their database is updated with new information. Looking to this, data synchronization among databases is very important. Data Synchronization in biology has been observed to be difficult for many reasons not just because of its own complexity of subject and emergence of new data types, but also for the entity who manages the data. [3]. The quality of data management is therefore dependant on who actually manages data. The question such as who is managing data- a biologist, a researcher, a computer science and an IT expert and this is how it changes the perception of data management. Biologists often focus on retrieving, storing, processing and analyzing data instead of organizing and synchronizing data and hence most tools they use lack the functionality of data synchronization. [4]. Besides, database systems for biological data in place are heterogeneous in nature; data synchronization mechanism should be independent of the particular database system. In this paper, we attempt to present an algorithm to synchronize data among homogeneous and heterogeneous system using XML as a data exchange language.

[1]Atos Origin India Pvt Ltd, Mumbai

[2]Shrimad Rajchandra Institute of Management and Computer Application, South Gujarat University, Gujarat, India

Email: [1]ateet.mehta@gmail.com1, [2]bankim_patel@srimca.edu.in

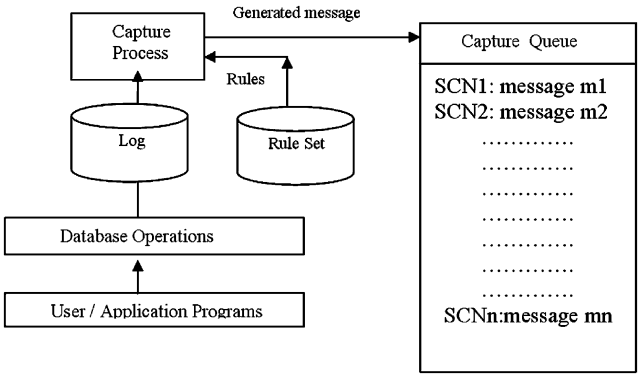
2. Related Work

The leading database management systems like Oracle, IBM DB2, Microsoft SQL Server, Teradata are being used to run biological databases. Each of these database vendors offers solutions for data synchronization and replication. Oracle offers Data Guard, Data Replication, Oracle Streams, Oracle Standby Database and Oracle Data Capture using Redo Logs. But these solutions are often expensive to implement. Further these solutions often come with advanced version of database systems as an additional pack. Biologists heavily depend on variety of tools to access data and they often are not concentrated on management aspect of data hence existing tools do not support functionalities like data synchronize and data replication.

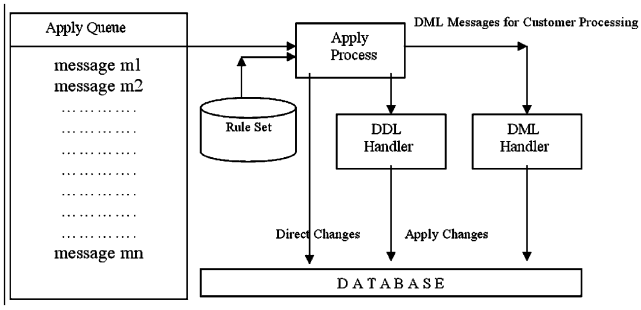
3. Proposed Work

We attempt to present an algorithm to synchronize data using message queue with messages in XML. The algorithm works even with heterogeneous databases. Algorithm uses two processes namely Capture and Apply which runs at source and target database respectively. Capture process captures any database event occurring in the database and checks within the rule set whether the message of this event needs

to be generated. Database event is trapped either by implementing database triggers or by tracing transactions logs implemented in underlying database systems. Rule set is a set of rules stored in the data dictionary which specifies which events should be synchronized with target database. Capture process creates messages for all events for which rules are set and periodically this message queue is propagated at the target database. Message queue can be propagated either at every commit happening in the source database or periodically, e.g. every 30 minutes. The message will be constructed in XML and will contain event type (DDL or DML), event handler type (Direct, Custom), Transaction type, data being manipulated by the transaction, and System Change Number (SCN). System Change number will be used to identify the sequence in which transactions have taken place in source database. Further it can also be used as the synchronization key between source and target database. Capture process can be enabled and disabled depending on the need. Same XML schema definition of the generated XML message should be used at source and target databases.



At target database, Apply Process scans the propagated queue and transfers them into the Apply queue all such messages for which rules are set in the target database. All messages are executed directly by Apply Process for which the message handle type is Direct. All custom messages are sent to custom procedural routines which are stored within the database and responsible for applying business rules and any data transformation needed on the data before applying the transaction on the database. DDL Custom messages are handled by DDL handler and DML custom messages are handled by DML handler.



Algorithm – Capture Process

1. Initialize Capture Queue – queue
2. Repeat the process while database is running
   - 2.1 Capture database event – event
   - 2.2 if e in {rule set} then
     - Create Message – message
     - Enque (queue,event,message);
   - 2.3 if counter then
     - transfer (queue);
     - end if
3. end

Algorithm- Apply Process

1. Initialize Apply Queue- queue
2. Repeat while queue is not empty

3. event= deque (queue)
4. if event in {rule set} then
   - if event_type_handler='DIRECT' then
     - apply message to database
   - else
     - /* Customer Event Handler */
     - if event_type='DDL' then
       - call DDL_Handler(message);
     - else
       - call DML_Handler(message);
     - end if
   - end if
   - end if
5. end

4. CONCLUSION AND FUTURE WORK

Data Synchronization is very essential in biological public databases are they are widely accessed by biologists all over the globe. The proposed algorithm can be easily implemented when source and target database is homogeneous or heterogeneous. This flexibility is needed as underlying database management system cannot be same among different biologist. Further the solution provided by the particular database vendor is often expensive and sometimes not needed for small databases. The algorithm can be extended with the inclusion of conflict detection at target database which is currently not added in the proposed algorithm. This algorithm can also be extended to have bidirectional synchronization from source to target. We have left many performance related issues that may exist during data synchronization.

REFERENCES

[1] Victor Markowitz-Biological Data Management in a Dataspace Framework.

[2] Jagadish, H.V., Olken- Database Management for Life Science Research.

[3] Aiguo Li-Facing the challenges of Data Integration in Biosciences.

[4] National Center for Biotechnology Information. http://www.ncbi.nih.gov.

[5] Torsten Suel-Improved File Synchronization Techniques.

[6] K Abdel- Optimal Strategy for Comparing File Copies.

[7] S.Balasubramaniyam- What is File Synchronization? ACM/IEEE MOBICOM'98.

[8] J Cho- An Effective Change Detection using Sampling.

[9] S Agarwal–Scalability of Data Synchronization Protocols-IEEE Network Magazine.