Research on Heterogeneous Data Exchange based on XML

Huanqin Li\textsuperscript{a}, Jinfeng Liu\textsuperscript{b}

\textsuperscript{a}Department of information technology, Zhengzhou Normal University, Zhengzhou, henan province, China
\textsuperscript{b}Department of The software college of, Henan university, Zhengzhou, henan province, China

Abstract

Integration of multiple data sources is becoming increasingly important for enterprises that cooperate closely with their partners for e-commerce. OLAP enables analysts and decision makers fast access to various materialized views from data warehouses. However, many corporations have internal business applications deployed on different platforms. This paper introduces a model for heterogeneous data exchange based on XML. The system can exchange and share the data among the different sources. The method used to realize the heterogeneous data exchange is given in this paper.

© 2012 Published by Elsevier B.V. Selection and/or peer-review under responsibility of Garry Lee
Open access under CC BY-NC-ND license.

Keywords: XML; heterogeneous data; data exchange; relation database.

1. Introduction

Database management systems (DBMS) pervade and proliferate tremendously throughout industry in the past decades. However, due to the storage capacity and cost, most of the prior database applications are mainly tailored to serve the information needs of people who handle day-to-day or short-term operations, such as inventory or purchasing. Thanks to the ever-increasing capability and decreasing price of storage devices, together with the speed promotion of Internet technologies, it is now feasible to bring historical data on-line to serve corporate decision-makers to access all the organization’s data, wherever it is located. The challenge for organizations now is the need to turn their archives of data into an integrated source of knowledge, such that a consolidated view of the organization’s data can be presented for decision-making[1].

However, since a data warehouse creation needs to integrate various enterprise-wide corporate data into a single repository, from which users can query via various dimensions and produce analysis reports[2]. There are problems may arise in building a data warehouse with pre-existing data, since it has various
types of heterogeneity. That makes it a common consensus that the ETL process (i.e., extraction, transformation, and loading) of data from various sources is indispensable before constructing a data warehouse. Therefore, the general conclusion is that the task has proven to be labor-intensive, error-prone, time-consuming and generally frustrating, leading a number of data warehousing projects to be abandoned mid-way through development. However, Trisolini et al. (1999) and Srivastava and Chen (1999) have pointed out that the situation is not as tough as it appears. In fact, the heterogeneity problems that are being encountered in data warehouse establishment are very similar to those encountered in heterogeneous database integration, which have been well studied in the past decade (ACM Computing Survey 1990; Batini et al. 1986; Breitbart et al. 1986; Breitbart 1990; Castano et al. 2001; IEEE Computer 1991; Hsiao 1992ab). Those works accomplished in dealing with heterogeneous schema integration have established a good framework to alleviate this task[3][4][5].

XML (eXtensible Marked Language) for the data integration of information provides a new solution. Because the XML can serve as a kind of information of high-rise encapsulation and transmission standards, or even different application system between the data interface standards, be able to accurately express multi-dimensional information between the logical structure and meaning. When a query task related tasks store data in different data sources, XML through its standard format for different between applications data exchange and different data integration between system provides a powerful mechanism. It appeared smoke soon in heterogeneous database applied in the integration of heterogeneous database based on XML, integration, and inquires both at home and abroad, it has become a hot topic[6].

2. The realization of heterogeneous data exchange

Based on XML heterogeneous data conversion technology as fig2 shows. Includes the following several main program modules.

1) connect to database. Both will database data saved to XML files, or XML data saved to the database, we first need to connect remotely user database. In system allows the user to select different database connection validation method.

2) From a database to read the source data, fill to recover Dataset cache list. Because a database have multiple relation table, during operation needs all the table filled to a data to recover Dataset cache list, from the XML into database and then separated into multi-table.
3) Database data saved to XML files. According to read watch ace database data in turn to recover Dataset cache list to filling, and the cache the data in the table to load the XML documents, realize the database file XML file conversion[7].

4) XML data storage in the database. Connected to a user database the premise condition, first loading XML data analytical source to recover DataSet then fill to watch each field name and types. According to the analytical results, with SQL commands create database and database table again finally by field analytical each field values, and SQL commands will these values inserted into the new data table, save to the corresponding database.

Figure 2. The heterogeneous database integration by xml technology

2.1 The database links

Systems use DBS XML files to convert j storage of source database and target database database connection information, including: database name, drives, server, database network address, username, password etc, users in the prior to the conversion should be DBS XML files fill, j conversion process needed to select type of database, the system will be selected according to the user and DBS. XML, via JDBC driver and JDBC - complete with various Bridge ODBC relational database connection. Below for DBS. XML files examples[8].

```xml
< ? xml v ersion= " 1. 0" encoding= "GB2312"? >
< dbs>
< source>
  <class> sun. jdbc.odbc.JdbcOdbcDriver </class>
  <url> jdbc: odbc: abc</url>
  < user> w ang</user>
  <password> 123456</password>
</ source>
< dest>
  <class> com. microsof t. jdbc.sqlserver. SQLServerDriver</class>
  <url> jdbc:microsof t:sql server: //localhost :1433; DatabaseN ame= w ang</url>
  <user> w ang</user>
  <password> 123456</password>
</ dest>
</ dbs>
```
2.2 XML documents to database information conversions

With XML document, you can take to exchange data deposit to destination database table. Process is as follows:

1) Generate database connection statements. Purpose of database information by the receiving party through inputting data interface. Must be in goal database query and generating SQL statement before inserting data generated database link statements, in order to use JDBC connection database.

2) parse XML document. In the Xml document contains to exchange data, and therefore must generate insert data in the SQL analytical it before.

3) Circulation processing elements. To search the information in the Document retrieval, need to include all the data root element. Once you have this element, an application can use `getElementByTagName()` and through the element name retrieving data. This method returns a NodeList that can be used to visit each element[9].

4) Retrieve data. Retrieval of actual data to merge `getElementByTagName()` and carefully visited the node's child nodes. For element attribute value, can pass `getAttribute()` method gets. Child elements of retrieval complex, need to expect some name retrieve all the NodeList child element. After a data can be added to database data.

2.3 A relational database and mapping rules of an XML document

In the XML data and database between conversion, you need to consider many problems, XML does not support any practical meaningful data model, all the data in an XML document will be as plain text processing. Usually data conversion middleware need to put in an XML document pure text convert database of data types, or database data type conversion for pure text XML format. In the XML document structure and database model structure between each mapping, we use as follows[10].

Based on the data elements tree mapping. After studying the existing methods based on the data mapping, we propose a mapping method based on elements of the tree, it is a model-based conversion methods. This method first creates element node tree and the tree elements and other types of mapping relations between the data and then mapping the elements of trees and on the basis of complete data conversion. First look at the elements of the definition of the tree.

The definition of element tree. A tree is a tree elements described DTD of elements of the relationship between the tree, its are defined as follows: tree of each node corresponds to a DTD an element of[11].

Each node contains the following information: a list of child elements, attributes, lists and other information.

Child element list contains all the elements of child elements, one for each child child node. Attribute list contains all the attributes of the element, each attribute is a triple (property name, property value, property type).

Other information includes: the father node, element content models, element content and element occurrences of text content.

DTD element content is not contained by other elements of the root.

Child element list is empty or the element attribute for the tree's leaf node.

A tree element trees in some cases are completely, i.e. when some element content model has child content model, system cannot determine the content of son content model definition should be how to appear. At this moment, need to insert content model node, explained the situation, then according to the actual data content to determine which should appear content, should appear how many times. So-called content model node is a kind of special nodes, its purpose is to reflect the son content model is established by. A content model nodes with belong to this child nodes content model all the elements of the son of the
corresponding node. For example the following definitions elements it, its content model is the content of particles, namely, child1 sequence list child2 or child3, according to the order child4 is appeared in an XML document. In the content of it element model contains a tall content model, this son content model is content of particles, namely select list in XML documents can only appear child2 and child3 - one of, we use a content model node[12].

On the basis of element tree, we also need to define an XML element and other types of data the mapping relationship between. For structured data, because it has a fine structure, we can define the XML document first with the structured data on the structure of mapping rules, then use mapping rules automatically determine XML element with business data the mapping relationship.

Based on the relational database, for example, the element database model between the tree and mapping rules can be defined as follows:

- The roots of a tree node corresponding element in the database tables, called a root table.
- If a node list or child elements of attributes list not null, so this node called table node, it is corresponding to the database; a table.
- If a child element node types for PCDATA type, the node called child nodes,
- A table of each attribute node and child nodes are corresponding to this node of the corresponding to the table of a field,
- If a table node or child nodes have a father node, then the parent node of the corresponding table is called the nodes of the corresponding table father list,
- The database, in addition to the root table outside the table should be included in a table with his father happened associated foreign key.
- Besides table node and content model outside the node called field node, it only corresponding node father node of the corresponding to the table of a field.
- Contents in the database model nodes don't correspond any object. However, in the actual application of the user data structures (such as database model) often has established good, then we need to specified by customer XML element and the mapping relationship between business data, and these data saved to a mapping table.

Tree-based data conversion element. Elements of the tree, based upon our conversion method, according to mapping to develop a series of executive orders. Through the implementation of these instructions, and the implementation of the results inserted into the corresponding position in the data model, you can get the corresponding XML documents. Similarly, a reverse command to connect to XML documents into other formats. With the introduction of content model nodes, the system limits on the structure of XML documents greatly relaxed[13]. And because the elements of tree generation is based on DTD, and therefore consistent with a DTD for a class of XML documents corresponding to the conversion element of the DTD tree can be reused many times. In the enterprise e-commerce, companies need to interact with the corresponding XML document DTD is relatively fixed. Once all the elements of DTD trees are generated, the system can reuse existing elements in the tree, thus greatly improving system performance.

3. Conclusion

The paper gives a xml-based heterogeneous data switching system realization plan in this system ensures that each member database independence and autonomy under the precondition of realizing the integration of heterogeneous data sharing and has better versatility and openness, suitable for the enterprise internal different platforms, data, the application of data exchange for enterprise integrated information for reference.
References


