International Journal of Information Management 33 (2013) 28–31

Contents lists available at SciVerse ScienceDirect

International Journal of Information Management

journal homepage: www.elsevier.com/locate/ijinfomgt

Viewpoint

XML based advanced distributed database: implemented on library system

Saikat Goswami [a,*], Chandan Kundu [b]

[a] Dept. of Library, EIILM, Kolkata, India
[b] Dept. of System, EIILM, Kolkata, India

ARTICLE INFO

Article history:
Available online 7 June 2012

Keywords:
Data warehouse
Distributed database
XML
Data mining
Oracle XMLDB
CLOB
Library system

ABSTRACT

The modern sensation the field of Library and Information Science is the word e-library where information's are stored in a digital format. It actually enables the use of the new emerging technique XML for transporting information over internet and thus nullifies the disadvantages (viz. rigid transaction set, fixed business rule, high cost, and slow pace) of previously used EDI (*Electronic Data Interchange*) technique. Keeping in mind the necessity of rapid and dynamic information explosion the library in the modern era should be universal. This concept of spreading a centralized system in a geographically dispersed system is known as *distributed system*. In this work we represent a model that is distributed in nature and information is transferred by means of XML.

1. Introduction

With the advent of Computer Science and its increasing immense popularity people is attached with the concept of Information Science. So, at present, books and computer system, both are the backbones of library system. A *digital library* is a library in which collections are stored in digital formats (as opposed to print or other media) and accessible by computers (Greenstein & Thorin, 2002). The digital content may be stored locally, or accessed remotely via computer networks. A digital library is a type of information retrieval system.

Most digital libraries provide a search interface which allows us to find our desired search items. These information resources are typically deep web (or invisible web) resources since they cannot be located frequently by search engine crawlers. Some digital libraries create special pages or sitemaps to allow search engines to find all their information resources. Digital libraries frequently use the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) to expose their metadata to other digital libraries, and search engines like Google Scholar, Yahoo! and Scirus can also use OAI-PMH to find these deep web resources (Koehlar, 2006).

The digital libraries are now widely recognized by commercial interests and public bodies for rapidly accessing books, archives and images of various types. Secondly in terms of storage space and cost the digital libraries are more beneficial than the traditional one.

A traditional library must spend large sum of money paying for staff, book maintenance, rent, and additional books. Digital libraries may reduce or, in some instances, do away with these fees. Both types of library require cataloguing input to allow users to locate and retrieve material. Digital libraries basically adopt innovation in technology and thus provide the users a new form of communication such as audio book technologies, wikis, blogs, etc. Conventional libraries may consider that providing online access to their OPAC catalogue is sufficient. Also an important advantage to digital conversion is increased accessibility to users. They also increase availability to individuals who may not be traditional patrons of a library, due to geographic location or organizational affiliation (http://www.w3.org/XML/schema; http://www.w3c.org/TR/REC-XML).

XML allows structured information to be displayed in a web-based format, be it over the Internet, or an organization's own private intranet. This format makes it easy for organizations with different systems to import and export data in a simple format.

2. XML as a transporting media

2.1. XML-brief ideas

The *eXtensible Markup Language* (*XML*), standardized by the *World Wide Web Consortium* (*W3C*) in February 1998, is self-describing, human and machine readable, extensible, flexible, and platform neutral (Fong, Kuen, & Shiu, 2008; Parimal & Pahwa,

* Corresponding author.
E-mail addresses: goswami0408saikat@gmail.com (S. Goswami), chandankundu2008@gmail.com (C. Kundu).

0268-4012/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.ijinfomgt.2012.05.002

August 2009). XML has become the standard format for exchanging information across the networks. The prospects of XML for DW are tremendous. XML is a markup language much like HTML. It was designed to carry data, not to display data. XML tags are not predefined. We must define our own tags.

2.2. Advantages of XML

Although there are a number of ways to transport useful information over network, still XML is gaining popularity day by day because of the following reasons:

i) *Expressive power*: XML can represent relational data, report formats and semi-structured documents directly without losing information and with uniform syntax.
ii) *Self-describing*: XML can combine data and metadata. Thereby, heterogeneous and even irregular data can be represented and processed without a fixed schema, which may become obsolete or simply get lost.
iii) XML provides a mechanism for representing data as a simple stream of text, allowing easy transmission and retrieval, as well as allowing existing Internet protocols to be used to exchange data between systems. XML allows semi-structured information to be displayed in a web-based format, be it over the Internet, or an organization's own private intranet. This format makes it easy for organizations with different systems to import and export data in a simple format.
iv) It allows data interchange between different computers.
v) It enables smart searches.

The ability to define data structures and deliver it between different systems makes XML particularly valuable to the DW.

3. Storage mechanism

Native XML databases are databases that store XML documents and data in a very efficient way (Alapati, 2005; http://download.oracle.com/docs/cd/B19306_01/appdev.102/b14259/xdb01int.htm). As classic (relational) databases, they allow data to be stored, queried, combined, secured, indexed, etc. Native XML databases are not based on tables, but on so-called containers. Each container can contain large amounts of XML documents or XML data, which have some relation between them.

Oracle XML DB simply refers to the set of built-in XML storage and retrieval technologies for the manipulation of XML data. Oracle XML DB provides the advantages of object-relational database technology and XML technology. For example, one of the major problems involved in dealing with XML data from within a relational database is that most XML data is hierarchical in nature, whereas the Oracle database is based on the relational model. Oracle manages to deal effectively with the hierarchical XML data by using special SQL operators and methods that let you easily query and update XML data in an Oracle database. Oracle XML DB builds the XML Document Object Model (DOM) into the Oracle kernel. Thus, most XML operations are treated as part of normal database processing.

Oracle XML DB provides the ability to view both structured and non-structured information as relational data. You can view the data as either rows in a table or nodes in an XML document.

3.1. Storing XML in database-XMLDB

The latest versions of Oracle provide a way to store XML documents directly in the database. XMLDB supported by database technologies is a way to high performance XML storage and retrieval. It provides native XML support by encompassing both SQL and XML data models in an interoperable manner.

Oracle uses a special native data type called XMLType to store and manage data in a relational table. XMLType enables us to leave the XML parsing, storage and retrieval to the Oracle database. We can use the XMLType data type just as we would the usual data type in an Oracle database.

With the CLOB XMLType you can store XML documents as a Character Large Object.

If we use the CLOB XMLType storage there is no need for validation of the XML. The XML does have to be well-formed, and it is optional to validate the XML document against an XML schema or DTD yourself.

With this type Oracle implicitly stores the XML in a CLOB column. The original XML is preserved including whitespace. All restrictions on a CLOB column apply also to this XMLType.

XML DB includes the following features:

1) Support for the World Wide Web Consortium (W3C) XML and XML Schema data models.
2) SQL supports to store, query, update, and transform XML data.
3) Helps to perform XML operations on SQL data.
4) A storage-independent, content-independent and programming language-independent infrastructure for storing and managing XML data which provides new ways of navigating and querying XML content stored in the database.
5) XML-specific memory management and optimizations.

The XML documents are stored in database using XMLTYPE datatype. It provides the way to store XML documents in an XMLType column or table. For non-schema-based storage, XMLType offers a Character Large Object (CLOB) storage option. Fig. 1 describes the overall architecture of the proposed model.

A particular application can be either schema based or non-schema based. In case of schema based application we can use either CLOB or structured storage. But if the application is non-schema-based then we can store our data in either CLOB values in XMLType tables or views or in files in XML DB Repository folders. In our approach we have adopted non-schema based application.

3.2. Creating XML based table

The basic syntax of creating XML based table is as follows:

create table <table_name> (Xml_Column xmltype);

Now consider the example of Book details. The Book catalogue problem comprises of *Book details* as measure and *Title, Publication, Place_of_Pub., Year_of_Pub.* and *ISBN* as dimensions. The following *SQL* statement creates a dimension say *Model*:

create table book_catalog_table (acc_num number (20),book_details xmltype);

After creating the XML document in the database, we have inserted and selected values as follows in Fig. 2.

3.3. Advantages of XMLDB

Here is a brief list of the benefits offered by Oracle XML DB:

- It facilitates to store huge amount of data in the form of XML in database.
- Data retrieval speed is higher than that of traditional RDBMS.
- We can access XML data using regular SQL queries.
- We can use Oracle's OLTP, data warehousing, test, spatial data, and multimedia features to process XML data.
- We can generate XML from an Oracle SQL query.
- We can transform XML into HTML format easily.

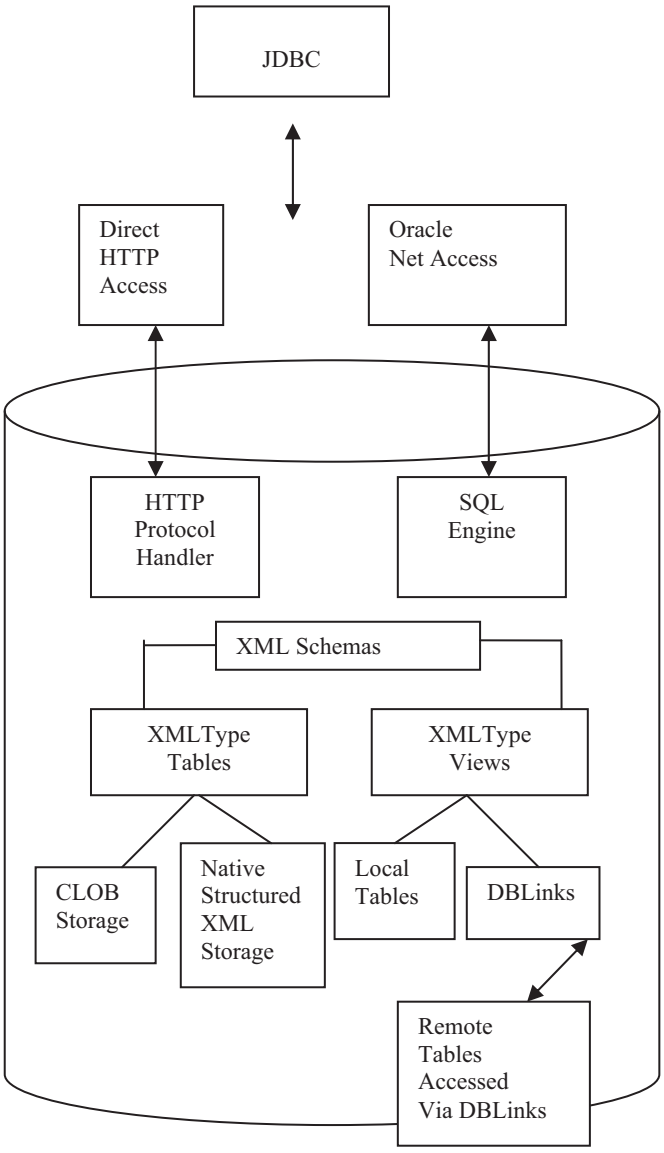

Fig. 1. Oracle XML DB architecture: XMLType storage.

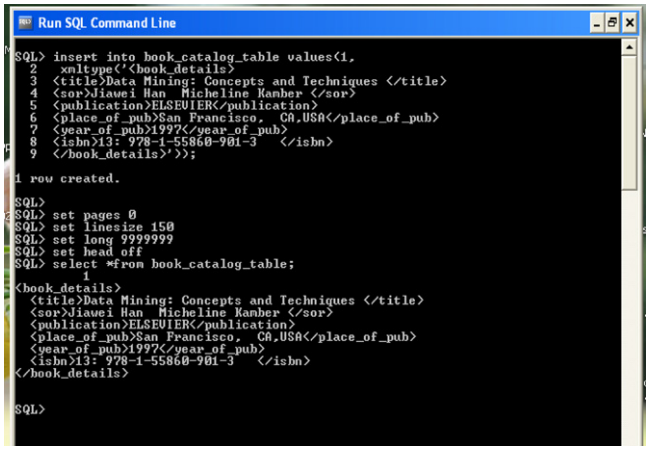

Fig. 2. Output from SQL command line.

3.4. Advantages of XML for libraries

Despite its great potential, there are only a few places on the Web where the impact of XML on libraries is discussed. There are even less concrete examples of XML projects. XML is currently being used for literary textual documents with a fairly simple structure; it is being used instead of HTML because people can define their own tags.

Some points in favor of XML (for libraries) are:

- The structure and markup of an XML document facilitates the creation of document databases, while at the same time XML content can be delivered on the Internet or on a CD-Rom.
- The "meta-data," or data about the document, can be explicitly read from the XML tags, provided that libraries can agree on a standard set of tags. For books and journals, the meta-data is captured in the form of a bibliographic "MARC" record that has to be created by hand and is added to the library database. XML would thus obviate the need for MARC records.
- There are some standard DTDs (Document Type Definition, an inheritance from SGML meaning the collection of tags of a particular XML application) that were designed specifically for literary scholarly documents (the TEI, Text Encoding Initiative) and scientific articles and books (ISO 12083). These applications can be used with XML. This is also an SGML MARC DTD.

4. Distributed database

A distributed database system allows applications to access data from local and remote databases. In a homogenous distributed database system, each database is of same type. In a heterogeneous distributed database system, at least one of the databases is of different type. Distributed databases use client/server architecture to process information requests.

4.1. Characteristics of distributed database

- Collection of logically related shared data.
- Data split into fragments.
- Fragments may be replicated.
- Fragments/replicas allocated to sites.
- Sites linked by a communications network.
- Data at each site is under control of a DBMS.
- DBMS can handle local application autonomously.
- Each DBMS participates in at least one global application.

4.2. Advantages of distributed system

- Data is located near the greatest demand site.
- Access is faster.
- Processing is faster due to several sites spreading out the work load.
- New sites can be added quickly and easily.
- Communication is improved.
- Operating costs are reduced.
- It is user friendly.
- There is less danger of a single-point failure.
- It has process independence.

In this paper we consider all the databases (distributed) homogenous in nature.

A homogenous distributed database system is a network of two or more of the same type of databases that reside on one or more machines. Fig. 3 illustrates a distributed system that connects three databases: LONDON, INDIA and USA. An application can

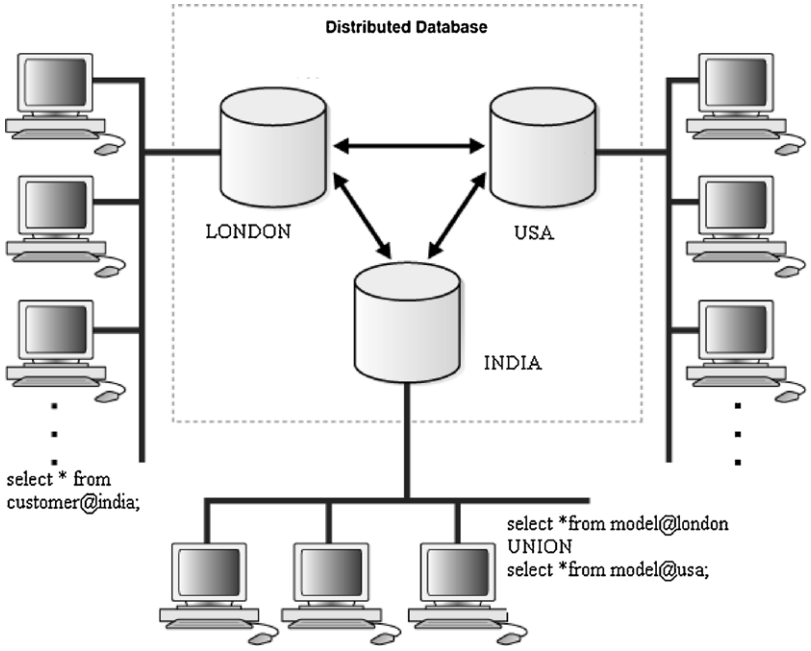

Fig. 3. Distributed XML based databases.

simultaneously access or modify the data in several databases in a single distributed environment. For example, a single query from 'INDIA' client on local database can retrieve joined data from the *Model* table on the remote USA database and the *Model* table on the remote LONDON database. The above fact has been explained in Fig. 3.

5. Future prospects of the current technology

In library Science, we come across millions of data stored in the database. This type of database is formally termed as data warehouse (Han & Kamber, 2006). A data warehouse is a repository of information collected from multiple sources, stored under a unified schema. From this huge database we try to find out some meaningful conclusions like Searching, Categorization, Trend, Demand, Indexing based on thought content, etc. Therefore in order to get some meaningful outcome we have to go through a process known as data mining (Dwivedi & Bajpai, 2004). Data mining has attracted a great deal of attention in the information industry and in society as a whole in recent years, due to the wide availability of huge amounts of data and the imminent need for turning such data into useful information and knowledge.

Data mining tools perform data analysis and may uncover important data patterns, contributing greatly to strategies, knowledge bases, and scientific and medical research.

This paper basically presents an idea that can be broadened to convert a simple database system into a promising field called data warehouse. In addition XML will help structured as well as unstructured information to be displayed in a web-based format, be it over the Internet, or an organization's own private intranet.

6. Conclusion

XML is gaining popularity at a tremendous speed. It provides a number of features that makes it suitable for the web based application. Native XML database (XMLDB) is a variation of the XML-database concept which has been accepted for several reasons. This approach can be expanded to give a shape of data warehouse to perform different data mining techniques in order to manipulate library information for varieties of work. In future we have planed to extend the model for non-XML output also.

References

Alapati, S. R. (2005). *Expert oracle database 10g administration*. Apress., 33 pp.
Dwivedi, R. K., & Bajpai, R. P. (2004). *Use of data mining in the field of library and information science: An overview* (pp. 512–517).
Fong, J., Kuen, S., & Shiu, H. (2008). *The XML tree model—Toward an XML conceptual schema reversed from XML schema definition* (pp. 1–4).
Greenstein, D. I., & Thorin, S. E. (2002). *The digital library: A biography*. Digital Library Federation., pp. 12–17.
Han, J., & Kamber, M. (2006). *Data mining: Concepts and techniques* (2nd ed.).
http://download.oracle.com/docs/cd/B19306_01/appdev.102/b14259/xdb01int.htm.
http://www.w3.org/XML/schema.
http://www.w3c.org/TR/REC-XML.
Koehlar, A. E. C. (2006). Some thoughts on the meaning of open access for university library technical services. *Serials Review*, *32*(1), 17.
Parimal, N., & Pahwa, P. (2009). From XML schema to cube. *International Journal of Computer Theory and Engineering*, *1*(August (3)), 236–238.

Mr. Saikat Goswami (MLIS, M. Phil.) Assistant Librarian, Dept. of Library, Eastern Institute for Integrated Learning in Management (EIILM), Kolkata.

Mr. Chandan Kundu (MCA, M.E., M. Phil., Pursuing PhD) Sr. Lecturer, Dept. of System, Eastern Institute for Integrated Learning in Management (EIILM), Kolkata.