Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 12, December 2014, pg.327 – 332

REVIEW ARTICLE

A Review Paper for Mapping of XML Data to Relational Table

Saurabh Vyas

PG scholar

GHRAET, Nagpur

saurabh.vyas7890@gmail.com

Shruti Kolte

Professor

GHRAET, Nagpur

shrutikolte32@gmail.com

ABSTRACT: ***This paper presents survey of techniques for Mapping XML data to relational table using supervised and unsupervised learning. XML is being increasingly used in diverse domains ranging from data and application integration to content management. The techniques are categorized based upon different approaches like Statistics, Self Organizing Mappings Based approaches. It provides the reader with the major advancement in the data mapping research, using these mapping techniques the features and categories in the surveyed work based upon the above stated categories. This served as the major contribution of this paper. Mappings in XML to tables can only deliver satisfactory query performance. We survey different mapping techniques for XML data into relational table or databases.***

Keywords: XML, Database System, Indexes, Query Processing, ShreX, XPEV

I. INTRODUCTION

In today's world or current scenario, the World Wide Web (WWW) is one of the important media used by lots of the human beings as the part of their regular life activity (i.e.; e-business, e-mail, e-management, e-learning, and e-library etc). Several Enterprise collaborates with other different enterprise in read or writes work flow in long term, they accessed through the services which are XML based. These inter enterprise

communication need large amount of data to be exchanged through web i.e. in XML format which needed to b stored somewhere as a digital copy. For the researchers and database vendors, storing huge amount of data or web services data is becoming a attractive area of research. The important issue is how efficiently they can retrieve and query these data. On the other side mapping data to the databases is also a big challenge.

Many of the techniques have been proposed for mapping XML data to relational tables. Using these techniques we aims to, no loss of information while shredding. Reconstruction of original XML documents is easier and faster. Maintaining a document structure. Preserve the ordering nature of XML data. Performing a semantic search for XML data.

II. LITERATURE REVIEW

A] *DataGuide-based Distribution for XML Documents*

[3] Data Guide: For XML Documents, DataGuide-based Distribution is effectively serves as a path-level index. Since XPath is the main navigational language for XML documents, were Data Guide is used in its optimizations. For example, in XML database nodes of Data Guide point to the specific nodes of a document, which allows quick evaluation for XPath queries.[1]

Data Guide for XML document D is an XML document DG with following properties:

1. for each path in D (Data) there is a equivalent or unique path in DG (Data Guide)
2. For each path in DG (Data Guide) there is a unique path in Data

Data Guide is a descriptive schema for SXML/XML data. While prescriptive schemas (DTD, XML Schema, Relax-NG) act more as a traditional database schema, restricting allowable XML data, a Data Guide infers rather than imposes structure. [3]

Data Guide shows actual (rather than possible) structure of XML data extracting the structure from the XML data. It may be used as schema for semi-structured data without any explicit schema declaration, such as non-validated XML documents.

B] *Query Translation from XPath to SQL in the Presence of Recursive DTDs*

[5] Xpath to Sql Translation:

An XPATH to SQL transformation algorithm that produces corresponding SQL queries for a subset of XPATH. This technique allows utilizing standard relational databases to store XML data. XPATH is a language which used to query a XML document for data.

The query is generally an expression. For every given XPATH expression, XPATH navigates through whole XML document to fetch the results which matching the predicates and conditions explain in the expression. Mapping data between XML and relational databases has become necessary for many organizations, because these days XML is used as a data store in many applications. [7]

C. Mapping XML to a wide sparse table

[1] Selective Sparse Mapping-

While mapping data in sparse mapping, entire document is mapped. Various applications may be or always interested in a small fragment only. In this section, we introduce selective sparse mapping, this is an index mechanism that allows users to map only a fragment of interest or data of interest to sparse columns. In original sparse mapping, each mapping column or mapping attributes corresponds to a unique path with only parent-child axes. In contrast, a column in the selective mapping may relates to a path expression containing // or *. Hence, elements from different two columns may not be disjoint. [1]

D. *Model-Mapping Approaches for Storing and Querying XML Documents in Relational Database*

[4] XPEV: This approach shows an improvement from Edge and XRel approaches in XML. It stores different XML documents in its relational schema and uses three tables schema to store XML documents in a relational database. Path table stores distinctive (root-to-any) paths, where Edge table represents parent-child relationships, and Value table stores all the values of elements and attributes. By using XPEV, any well developed XML document can be mapped to the relational database schema. It uses XQuery 2.0 and XPath 1.0 Data Model, and is having an efficient mapping approach for storing XML documents using relational database technology to provide well enhanced query processing. In addition, it is easy to understand, and it performs as unique model-mapping approaches.[4]

E. *A Comprehensive Solution to the XMLtoRelational Mapping Problem*

[7] ShreX:

XML-to-relational mapping framework and system. This system provides the first comprehensive solution to the relational storage for XML data available by web transaction. Mappings in *ShreX* are design through annotations to an XML schema or XML data. The use of XML Schema simplifies the mapping process, since user does not

need to understand new specialized mapping language. The use of annotations allows mapping choices to be combined in several or multiple ways. As a result, ShreX is goo in supports all the mapping strategies, but it never consider new useful strategies that need to be considered.[7]

Sr. No.	Mapping Technique	Description	Advantage	Limitation
[1]	Data Guide	Used as Optimization of XML data for Xpath or Xquery.	Determining fragments to distribute and replicate and their placement. Management of corresponding fragments and replicas.	Replication of data fragment leads to error in mapping data using data guides.
[2]	Xpath to Sql Translation	Is a language used to query a XML document for data, The query is an expression.	When queries are written in XQuery, they require less code as compared to queries written in XSLT.	the limitations of the SQL language in handling semi-structured or markup information resulted in the development of the XQuery language
[3]	Selective Sparse Mapping	An index mechanism which allows user to map interested fragment of sparse column.	While accessing xml data its generally use XML data which assessed more from the table.	Due to selective approach, more relevant or appropriate data is not getting extracted.
[4]	XPEV	Is an efficient mapping approach for storing XML documents using relational database technology to enhance query processing	any well-formed XML document can be mapped to the relational database schema	its relational schema remains large, because it stores entire XML documents in two tables
[5]	Shredding XML(ShreX)	a system for shredding, loading and querying XML documents in relational databases	Incoming XML data is just feeding an existing relational database, at this instance shredding can be useful.	Shredding XML into a large number of tables can lead to a complex and unnatural fragmentation of your logical business objects that makes application development difficult and error-prone.

III. CONCLUSION

In this paper, different techniques for mapping XML to sparse table, their advantages and limitation is introduced. While sensing the surrounding environment, processing the sensed information and mapping the resultant data relational databases consumes high amount of data base space compared to sparse table. Therefore, the mapping techniques need to spend the least amount of database spaces as possible to spare enough space for the crucial operations of the XML mapping to relational table. As space is the scarce resource, query execution time of the mapping techniques is an important issue from a system design point of view. So, if high query execution time consumes more system time in process. This can surely save space and time of the rest of the system and ultimately increase the total lifetime of the system.

REFERENCES

[1] L. J. Chen et al. "Mapping XML to a wide sparse table," in IEEE 28th ICDE, Washington, DC, USA, 2012.

[2] P. A. Boncz et al "MonetDB/XQuery: A fast XQuery processor powered by a relational engine," in Proc. SIGMOD, Chicago, IL, USA, 2006.

[3] Alexander Kalinin, , "DataGuide-based Distribution for XML Documents," Proc. Proceedings of the Spring Young Researcher's Colloquium on Database and Information Systems, Saint-Petersburg, Russia, 2009.

[4] Amjad Qtaish, Kamsuriah Ahmad, "Model-Mapping Approaches for Storing and Querying XML Documents in Relational Database.

[5] Wenfei Fan, Jeffrey Xu Yu, et al, "Query Translation from XPath to SQL in the Presence of Recursive DTDs," in Proc of the 31st VLDB Conference, Trondheim, Norway, 2005.

[6] J. Qin, S. Zhao, S. Yang, and W. Dou, "Efficient storing well-formed XML documents using RDBMS," in Services Systems and Services Management, 2005. Proceedings of ICSSSM'05. 2005 International Conference on, pp. 1075-1080, 2005.

[7] Amer Yahia Fang Du, "A Comprehensive Solution to the XMLtoRelational Mapping Problem.

[8] A. Barta, M. P. Consens, and A. O. Mendelzon, "Benefits of path summaries in an XML query optimizer supporting multiple access methods," in Proc. 31st Int. Conf. VLDB, 2005.

[9] J. L. Beckmann, A. Halverson, R. Krishnamurthy, and J. F. Naughton, "Extending RDBMSs to support sparse datasets using an interpreted attribute storage format," in Proc. 22nd ICDE, Washington, DC, USA, 2006.

[10] V. Benzaken, G. Castagna, D. Colazzo, and K. Nguyen, "Typebased XML projection," in Proc. 32nd Int. Conf. VLDB, Seoul, Korea, 2006.

[11] K. S. Beyer et al. "System RX: One part relational, one part XML," in Proc. SIGMOD, Baltimore, MD, USA, 2005.

[12] P. Bohannon, J. Freire, P. Roy, and J. Sim´eon. From XML schema to relations: A cost-based approach to XML storage. In Proceedings of the International Conference on Data Engineering (ICDE), pages 64–75, 2002.

[13] P. Bohannon, J. Freire, P. Roy, and J. Siméon, "From XML schema to relations: A cost-based approach to XML storage," in Proc. 18th ICDE, San Jose, CA, USA, 2002.

[14] S. Amer-Yahia, F. Du, and J. Freire. A generic and flexible framework for mapping XML documents into relations.

[15] Technical report, OGI/OHSU, 2004 [15] Z. Chen, H. V. Jagadish, L. V. S. Lakshmanan, and S. Paparizos, "From tree patterns to generalized tree patterns: On efficient evaluation of XQuery," in Proc. 29th Int. Conf. VLDB, Berlin, Germany, 2003. 1414 IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 6, JUNE 2014

[16] D. DeHaan, D. Toman, M. P. Consens, and M. T. Özsu, "A comprehensive XQuery to SQL translation using dynamic interval encoding," in Proc. SIGMOD, San Diego, CA, USA, 2003.

