On Two Query Interfaces for Genome XML Databases

Masatoshi Yoshikawa†, Toshiyuki Amagasa†, Dao Dinh Kha†, Kenji Hatano†, Hiroko Kinutani†, Noboru Matoba†, Junko Tanoue†, Masahiro Watanabe‡ and Shunsuke Uemura†

† *Graduate School of Information Science, Nara Institute of Science and Technology*
8916–5 Takayama, Ikoma, Nara, 630-0101, Japan
‡ *Department of Education Technology, National Institute of Special Education*
5–1–1 Nobi, Yokosuka, Kanagawa, 239-0841, Japan
E-mail: {*yosikawa,amagasa,kha-d,hatano,hiroko-k,noboru-m,junko-ta,uemura*}*@is.aist-nara.ac.jp, masahiro@nise.go.jp*

1. Introduction

XML has been recognized as a useful format to represent genome data. Until now, several XML-based markup languages for genome data have been proposed. They include GAME[7], GeneX[8], BioML[1] and GXML/GQL[17, 18].

To achieve XML-enabled wide area search, every database site is required to provide a standard query interface. There are two types of such query interface; one is system-oriented, and the other is human-oriented. System-oriented query interface means a formal XML query language interface such as XQL[13], XML-QL[3, 4] and Quilt[14, 2]. Whereas, human-oriented query interface is those which are similar to current HTML search engines. In this research, we are studying the following two issues about storage and retrieval of large volume of XML data:

1. An XML database system supporting a system-oriented query interface, where the query language is XPath[22].
2. An XML search engine supporting a user-oriented query interface.

2. An XML Database System – Path Approach

There have been several proposals on approaches to XML databases. They can be classified into the following three categories: i) development of dedicated XML databases, ii) use of object-oriented databases, or iii) use of relational databases. Among these, we consider the approach iii) is most promising for the following reasons:

1. We can utilize the functionality of query optimizers of relational databases for processing XML queries.
2. Currently, a large portion of non-XML data is stored in relational databases.

The basic logical data structure of XML data can be regarded as a tree. When developing a relational database for XML data, the main issue is how to map tree data structures into relational schemas. One of the important characteristics of Genome databases is that they are highly evolutional. Therefore, database schemas are updated very often. If data was represented in XML, this implies frequent changes of DTDs. For example, the DTD of Genome Annotation Markup Elements (GAME), which is used in Drosophila Genome Project, has been updated several times for a relatively short period. Hence, it is desirable the relational database schema for storing XML data is immune to the change of DTDs. Unfortunately, many mapping schemes[6][15] proposed so far lack of this immunity. We have developed a scheme[16] of mapping XML data into relational tables. We call the scheme as path approach since the basic idea is to store paths from the root to nodes as a character string in a relation table. The principal advantages of the path approach is i) that complex path expressions are transformed into string value comparison in SQL; and ii) that the expressive power of SQL need not be extended. We are now planning to extend this scheme in several possible directions. They include: incorporation of indices suitable for CLOB (Character Large OBject) data such as sequence data; support of XLink[20] and XPointer[21]; and support of indices for efficient handling of update[11].

3. Utilization of Structural Information in XML Search Engines

Although system-oriented query interface is necessary for application programmers developing systems for XML enabled wide area search, the majority of database users will simply be end-users who are not familiar with formal XML query languages such as XPath. From the observation of current HTML search engines, we learned the average

number of query terms is between 2 and 3[10]. We foresee end-users' queries will tend to be terse also in XML search engines. The issue here is how we can utilize the structural information of XML data, in queries, ranking and output of search engines. We are studying methods for identifying appropriate result subdocuments for a simple class of queries[12]. The basic assumption here is we cannot expect end-users understand all the details of DTDs. However, it is quite likely end-users know common tag sets such as Dublin Core[5]. Hence, the queries we consider consist of pairs of element names and terms. Much studies have already been carried out for query languages and query processing of XML data. However, they almost overlooked the integration of document logical structure data and vector data. The latter has been extensively used in traditional information retrieval community. We are now developing XML document ranking mechanisms which takes into account both document structures and document vectors[19] [9].

4. Conclusion

We hope our ongoing research on XML databases and XML search engines are relevant to and useful for handling a variety of genome data. Through the discussion at the Workshop, we hope we can identify the strength and weakness of our techniques for storing and querying large volume of genome data.

References

[1] "http://ala.vsms.nottingham.ac.uk/biodom/".

[2] Donald D. Chamberlin, Jonathan Robie, and Daniela Florescu. "Quilt: An XML Query Language for Heterogeneous Data Sources". In *Proceedings of the Third International Workshop on the Web and Databases, WebDB 2000*, pp. 53–62, May 2000.

[3] Alin Deutsch, Mary Fernandez, Daniela Florescu, Alon Levy, and Dan Suciu. "XML-QL: A Query Language for XML". http://www.w3.org/TR/NOTE-xml-ql/, August 1998.

[4] Alin Deutsch, Mary Fernandez, Daniela Florescu, Alon Levy, and Dan Suciu. "A Query Language for XML". *WWW8 / Computer Networks*, Vol. 31, No. 11-16,17, pp. 1155–1169, May 1999.

[5] Dublin Core. Dublin core metadata. http://purl.oclc.org/metadata/dublin_core/.

[6] Daniela Florescu and Donald Kossmann. "Storing and Querying XML Data using an RDMBS". *IEEE Data Engineering Bulletin*, Vol. 22, No. 3, pp. 27–34, September 1999.

[7] "http://www.bioxml.org/Projects/game/".

[8] "http://www.ncgr.org/research/genex".

[9] Kenji Hatano, Masahiro Watanabe, Masatoshi Yoshikawa, and Shunsuke Uemura. "Automatic Extraction of Partial Documents based on Element Feature Vectors". In *Proc. of DBWeb2000*, December 2000. (in Japanese).

[10] Bernard J. Jansen, Amanda Spink, and Tefko Saracevic. "Real life, real users, and real needs: a study and analysis of user queries on the web". *Information Processing and Management*, Vol. 36, No. 2, pp. 207–227, March 2000.

[11] Dao Dinh Kha, Masatoshi Yoshikawa, and Shunsuke Uemura. "An XML Indexing Structure with Relative Region Coordinate". In *Proc. of IEEE 17th International Conference on Data Engineering*, 2001. (to appear).

[12] Hiroko Kinutani, Masatoshi Yoshikawa, and Shunsuke Uemura. "Identifying Result Subdocuments of XML Search Conditions". In *Austria-Japan Seminar (held in conjunction with 2000 Kyoto International Conference on Digital Libraries: Research and Practice*, October 2000.

[13] Jonathan Robie. "XQL (XML Query Language)". http://metalab.unc.edu/xql/xql-proposal.xml, August 1999.

[14] Jonathan Robie, Don Chamberlin, and Daniela Florescu. "Quilt: an XML Query Language". http://www.almaden.ibm.com/cs/people/chamberlin/quilt_euro.html, March 2000.

[15] Jayavel Shanmugasundaram, H. Gang, Kristin Tufte, Chun Zhang, David J. DeWitt, and Jeffrey F. Naughton. "Relational Databases for Querying XML Documents: Limitations and Opportunities". In Malcolm P. Atkinson, Maria E. Orlowska, Patrick Valduriez, Stanley B. Zdonik, and Michael L. Brodie, editors, *VLDB'99, Proceedings of 25th International Conference on Very Large Data Bases, September 7-10, 1999, Edinburgh, Scotland, UK*, pp. 302–314. Morgan Kaufmann, 1999.

[16] Takeyuki Shimura, Masatoshi Yoshikawa, and Shunsuke Uemura. "Storage and Retrieval of XML Documents using Object-Relational Databases". In *Proc. of the 10th International Conference on Database and Expert Systems Applications (DEXA'99)*, Vol. 1677 of *Lecture Notes in Computer Science*, pp. 206–217. Springer-Verlag, August-September 1999.

[17] Aaron J. Stokes, Hideo Matsuda, and Akihiro Hashimoto. "GXML: A Novel Method for Exchanging and Querying Complete Genomes by Representing them as Structured Documents". *IPSJ Transactions on Databases*, Vol. 40, No. TOD3, pp. 66–78, August 1999.

[18] Aaron J. Stokes, Hideo Matsuda, and Akihiro Hashimoto. "Making High-level Queries on Diverse Genome Data: A Structured Genome Document Database System based on GXML and GQL". In *Genome Informatics*, Vol. 10, pp. 176–185. Universal Academy Press, December 1999.

[19] Masahiro Watanabe, Kenji Hatano, Masatoshi Yoshikawa, Shunsuke Uemura, and Hitoshi Nakamura. "On Keyword and Vector Search Integration Method Based on XPath". In *Proc. of DBWeb2000*, December 2000. (in Japanese).

[20] World Wide Web Consortium. "XML Linking Language (XLink) Version 1.0". http://www.w3.org/TR/xlink/.

[21] World Wide Web Consortium. "XML Pointer Language (XPointer) Version 1.0". http://www.w3.org/TR/xptr.

[22] World Wide Web Consortium. "XML Path Language (XPath) Version 1.0". http://www.w3.org/TR/xpath, November 1999. W3C Recommendation 16 November 1999.