# X-NIndex: A High Performance Stable and Large XML Document Query Approach and Experience in TOP500 List Data

Shaomei Wu, Xuan Li, and Zhihui Du

Department of Computer Sci. & Tech.
Tsinghua University, Beijing, P. R. C, 100084
wusm@mails.tsinghua.edu.cn

**Abstract.** This article describes X-NIndex, a novel approach for large XML documents with stable structure. The definition for the large XML document with stable structure is given while the concept of XML document tree coordinate(X-DTC) is introduced. The significant advantage of X-NIndex to other XML query schemas is shown and the experimental results are present.

## 1 Introduction

Extensive Markup Language (XML) [1] is emerged as the dominant standard for representing and exchanging data over Internet. However, as an organization of data with various semistructures, it is more difficult to store and query various XML documents, especially those consisted with a large amount of data.

To address this problem, there has been much work done on building XML database system, which lead to a research on schema design in the database for storage and query problem in XML. At present, several models have been advanced include XML-QL [8], XML-GL [3], Quilt [7], XPath [4], X-Rel [9], XQuery [6], and XML Indexed Structure with RRC [5]. They process XML data by changing XML document into different data schemas and do query on the schemas.

In our work, we found that much of these approaches are not very efficient in operation of large XML documents, and to solve such problem, we propose a new query structure named X-NIndex (XML Node Index Structure), which is proved to be able to improve query performance greatly.

The features of X-NIndex are: 1) for the XML document with large data amount and stable structure; 2) great efficient in query performance with such XML documents; 3) index structure and the XML document Tree Coordinate (X-DTC).

The rest of this paper is organized as follows: Section 2 describes the new approach of X-NIndex, including the definition of large XML document with stable structure, the concept of X-DTC and the exact process of X-NIndex; Section 3 compares the query performances in X-NIndex and other query structures, presents an example of processing Top500 computers data; Section 4 gives the conclusion of our work.

M. Li et al. (Eds.): GCC 2003, LNCS 3032, pp. 1043–1046, 2004.

## 2 X-NIndex Query Structure

In this Section, we will give out more details about X-NIndex, including the exact definition of Large XML document with stable structure, the concept of X-DTC and the process of X-NIndex.

### 2.1 The Definition of Large XML Document with Stable Structure

Here we adopt the XML document to be presented as a rooted tree with several node types such as Element, Attribute, Text, etc, the structure of it is shown in Fig. 1.

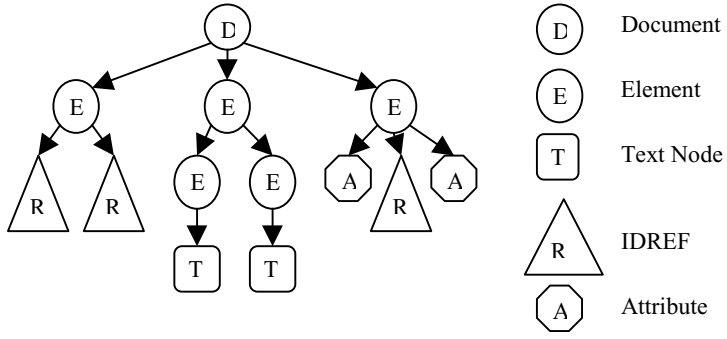


**Fig. 1.** The Rooted Structured Tree of XML Document

Especially, our discussion is base on a particular type of XML document, large XML Document with stable Structure, and we give the exact definition below:

**Definition 1:** Large XML Document with Stable Structure has such features: 1) it has a relative stable structure, which means, we do our query on the structure pre-known and the structure of it will not change once during the query; 2) it is consisted by a great amount of records as children of the root document element, and other parts of the document are all the attributes or descendants of such record elements; 3) each record has the same structure; 4) the number of attributes (or children attributes) is much smaller than the number of records.

If not mention clearly, all our study below is considered with such special XML document, for example, the Top500 XML ranking data documents.

### 2.2 XML Document Tree Coordinate (X-DTC)

To get the location of each node of XML document quickly, we introduce a new concept called X-DTC (XML Document Tree Coordinate). For the characteristic of stable structure, we assume that the children number of each node is given already.

First, we express the document in a tree as in Fig.1, and record each node's children number. Then, we add x and y coordinate into the tree and evaluate each node with a coordinate value, as shown in Fig. 2. Here, the coordinate x is the order of it in

its compeers and y is valued with the order of level in the document tree. We prescribe, the document is in level 0 and all the records are in level 1.

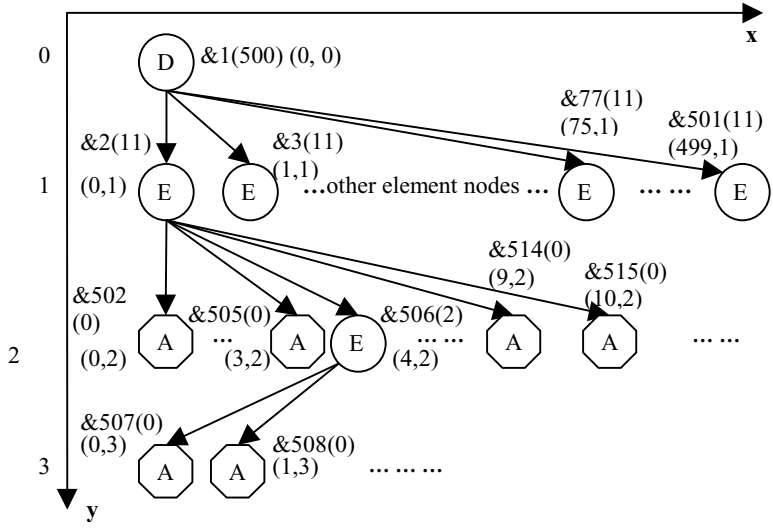


**Fig. 2.** Document Tree with X-DTC

### 2.3 The Process of X-NIndex Query Structure

From the analysis above, now we give the process of X-NIndex approach sketchily:
(1) Analyze the original XML document, present it into X-DTC form;
(2) Translate the data nodes of XML document into several indexed tables, do queries on the indexed tables with ordinary search method, for example, binary search;
(3) Using the coordinate data to compute and locate the record we search for in X_DTC, return the results set, which will be found from level 2 to level 3 or below.

## 3 A Performance Study of X-NIndex

To assess the effectiveness of the X-NIndex approach, we use JAXP and SAX to analyze the Top 500 XML documents. The query performances are compared and reported. It is important to notify that all our experiments are base on the large XML document with stable query, as we defined and referred before.

All experiments were conducted on a 1.60GHz PC with 256MB RAM, 30G hard disk. The experiment is to do different queries on the same sample of Top500 XML document. Results are shown in Fig 3, unit of y-axis is millisecond.

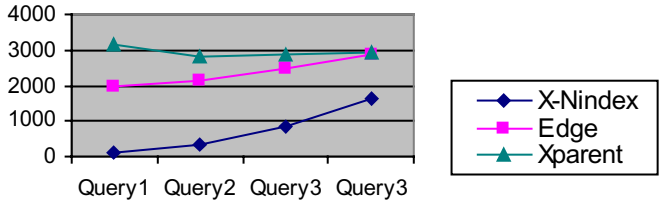


**Fig. 3.** Results for different queries in the same XML document sample

From the results we could see clearly that with X-NIndex, the effectiveness improved a lot of querying in the XML documents. Although it is not as stable as other approaches, as the efficient of X-NIndex may decrease more obviously with the scale of results set, but there is a limit of efficient descending reached by X-NIndex which is not significant more than the advantage of it in the average condition.

Such good results come from the special stable structure of our study objects, which make it possible to have much work pre-done during parsing the XML document, and much information is collected and helps to direct our queries.

## 4 Conclusion and Future Work

In this paper, we proposed a new coordinate-index approach called X-NIndex, the features of our work include the pre-analysis of stable XML document and the coordinate installation which helps to increase the efficiency a lot in the query afterward.

As some problems still exist with X-NIndex, for instance, the instability of it with increase of result set, in our future work, we will investigate the improvement with X-NIndex in such cases and conduct some new methods or concepts into this problem.

## References

1. W3C Extensible Markup Language XML1.1 in www.w3.org/TR/xml11/.
2. W3C XML Schema 1.1, in www.w3c.org/xml/schema.
3. Ceri, S., Comai, S., Damiani, E., and Fraternali, P. (1999). XML-GL: a graphical language querying and restructuring XML documents 1. in WWW8.
4. Clark, J. and DeRose, S., 1999, XML Path Language (XPath).
5. Dao Dinh Kha, Masatoshi Yoshikawa, Shunsuke Uemura, An XML Indexing Structure with Relative Region Coordinate in ICDE'01.
6. D. Chamberlin, XQuery: An XML query language in IBM System Journal, 2002.
7. Don Chamberlin, Jonathan Robie, Daniela Florescu , Quilt: An XML Query Language for Heterogeneous Data Source.
8. Deutsch, A., Fernandez, M., Suciu, D.(1999b).Storing Semistructured Data in Relations.
9. M. Yoshikawa, T. Amagasa, T. Shimura, S. Uemura,. XRel: A Path-Based Approach to Storage and Retrieval of XML Documents Using Relational Databases, 2001.