EXIMA™ Supply at INEX 2002: Using an Object-relational DBMS for XML Retrieval

Heesop KIM *, Daesik JANG **, Gi Chai HONG***, Jong Cheol SONG***, Seong Yong LEE***,
Hyun Soo CHUNG***, Jae Hwan LEE***, Byung Ju MOON***
* Department of Library and Information Science, Kyungpook National University, Daegu, 702-701, KOREA
heesop@knu.ac.kr
** INCOM I&C Co. Ltd. R&D Center, 996-1, Daechi-dong, KangNam-Gu, Seoul, 135-280, KOREA
dsjang@duli.incom.co.kr
*** IT Information Center, ETRI, 161 Gajeong-dong, Yuseong-gu, Daejeon, 305-350, KOREA
{gchong, jcsong, leesy5, hsjung, jeahlee, bjmoon}@etri.re.kr

Abstract

In this paper we report our approach using an object-relational DBMS for INEX collection. EXIMA™ Supply is a kind of native XML DB and supporting Xpath Standard to search elements in XML documents, however, it is not offer any functionality of intelligent searching techniques. We briefly describe the test collection preparation, indexing, retrieval processes, and the evaluation results. Although EXIMA™ Supply has many benefits, for example, no delay in storing and searching XML documents, it showed relatively poor performance in overall evaluation at INEX 2002. This result may be caused since the given topics had to be decomposed and modified to be processed by the Xpath processor in EXIMA™ Supply, and during this modification the original meaning of topics can be changed inevitably and some important information may missing. Furthermore, EXIMA™ Supply targets only for Korean documents, and we were not able to implement any aid tools for construction of indices, knowledge bases for INEX 2002 test collection.

Keywords

XML Retrieval; EXIMA Supply; Object-relational DBMS; UniSQL; IR Evaluation

1. Introduction

The topics provided by INEX (Initiative for the Evaluation of XML retrieval) were deployed and tested by the native XML DB named EXIMA™ Supply developed by Incom I&C Co. Ltd.

EXIMA™ Supply is a kind of native XML DB to store and manage XML documents effectively. It can store and retrieve XML and its related documents (e.g., DTD, XSL) fast enough to process XML information. EXIMA™ Supply is supporting Xpath Standard to search elements in XML documents. However, it is not provide any functionality of a searching engine. This means that it cannot search information as intelligently as most searching engines do. As a result, the given topics had to be decomposed and modified to be processed by the Xpath processor in EXIMA™ Supply. The modified topics were expressed in one or several Xpath queries. Some complicated topics had to be decomposed into several Xpath queries. During this process of modification, the original meanings of topics were changed inevitably and some information was lost.

2. System environments

2.1. Software

The topics provided by INEX were tested under the following software environment.

- OS: Windows 2000 Professional
- XML Server: EXIMA™ Supply 1.0
- DBMS: UniSQL 5.1
- Web Server: Tomcat
- Searching client: Web application developed with JSP,

2.2. Hardware
- Server
  : Machine - Pentium III PC
  : Memory – 256 MB
- Client
  : Machine - Pentium III PC
  : Memory – 256 MB

3. Experimental Design

3.1. Test collection preparation

3.1.1. Preparing of test collection

The XML documents in test collection are stored in EXIMA™ Supply. EXIMA™ Supply is a native XML DB based on object-relational DBMS technologies. Therefore, it can preserve the native features of XML documents by representing and storing them in object-oriented structures. This is one of the important features of EXIMA ™ Supply. Thanks to this feature, the data and hierarchical information of XML documents can be stored without modification or distortion.

Besides, EXIMA™ Supply helps manage and utilize XML documents with ease by providing the standard Xpath query language. With EXIMA™ Supply, there is no need to transform XML documents into other formats such as relational tables of commercial DBMS (many XML servers are using relation DBMS and therefore XML documents must be transformed into relational tables), because it can treat the hierarchical structures of XML documents as it is. As a result, there is no delay in storing and searching XML documents and it is possible to process XML data on the fly.

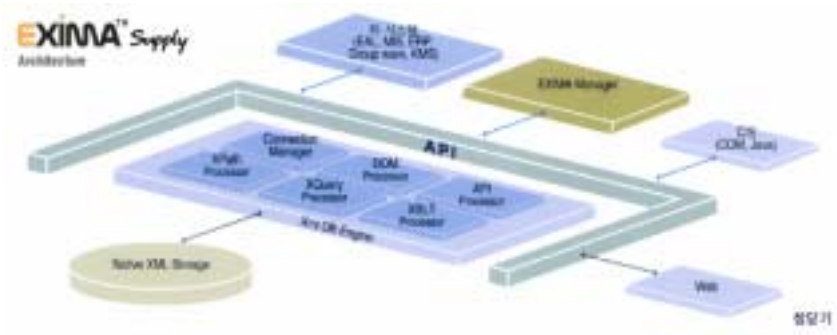

Figure 1: Architecture of EXIMA™ Supply

EXIMA™ Supply provides a logically hierarchical structure to manage the storage of XML documents. The logically hierarchical structure is the storage structure that is transparently accessible by users regardless of the internal physical storage structure. EXIMA™ Supply has two kinds of storage types, “Cabinet” and “Folder.” Cabinet is a logical storage that can contain cabinets and folders. Cabinet can be used to manage storage

hierarchically.

Folder is the storage where XML and related documents are actually stored. A folder can contain one DTD and corresponding XML and XSL documents. On the other hand, XML documents correspond to a DTD can be stored in multiple folders if necessary.

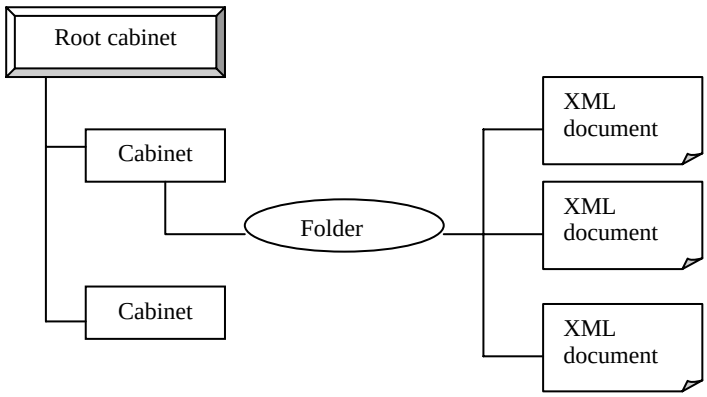

Figure 2: The Storage Types of EXIMA™ Supply

In set up the XML documents provided by INEX into EXIMA™ Supply, the directory structure of XML documents was mapped into the logical structure of EXIMA™ Supply. For example, XML documents in “E:\an\1995” directory are stored in the folder “1995” in the cabinet “an.”

The following picture shows the example storage structure of EXIMA™ Supply shown in EXIMA™ Manager.

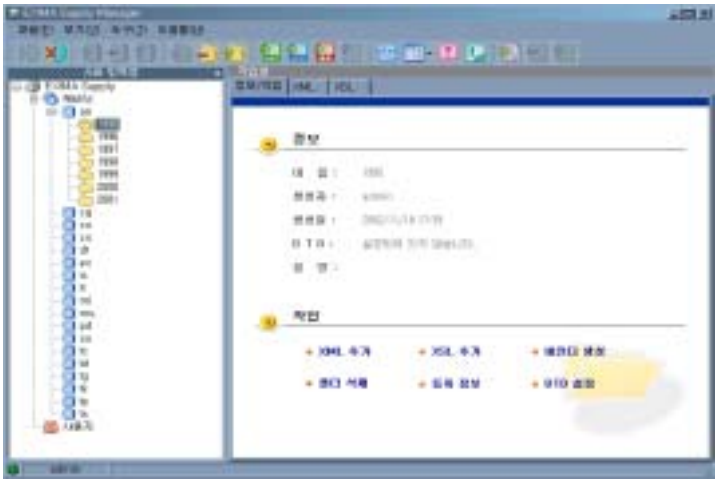

Figure 3: Example of the Storage Structure of EXIMA™ Supply

3.1.2. Indexing

EXIMA™ Supply has the functionality of indexing of elements of XML documents. EXIMA™ Supply makes indexes of elements when an XML document is stored. So it doesn’t need any extra indexing process. Elements in one folder are indexed together and the searching speed is almost same among elements in one folder. However, the indexing is done in each folders, the searching speed may be different from each folder.

3.1.3. Retrieval process

- Xpath query generation

EXIMA™ Supply is not equipped with any searching engine functionality and it just supports Xpath searching functionality. Therefore, searching topics from INEX has to be converted to Xpath queries for searching information. For instance, INEX topic 01 can be expressed in Xpath queries as follows:

Topic 01:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE INEX-Topic SYSTEM "inex-topics.dtd">
<INEX-Topic topic-id="01" query-type="CAS" ct-no="010">
    <Title>
      <te>article/fm/au</te>
      <cw>description logics</cw><ce>abs, kwd</ce>
    </Title>
    <Description>
      Retrieve the names of authors of articles on description logic, in particular articles in
      which the abstract or the list of keywords contains a reference to description logic.
    </Description>
    <Narrative>
      The rating should reflect the likeliness that a person is an expert on description logic.
    </Narrative>
    <Keywords>
      description logic DL ABox TBox reasoning
    </Keywords>
</INEX-Topic>
```

Xpath query:

```
"article/fm[abs//*/text('*')[contains('description logic')]]/au"
```

Complicated topics that can not be expressed in one Xpath query can be divided into several Xpath queries. For instance, topic 06 can be expressed in Xpath queries as follows:

Topic 06:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE INEX-Topic SYSTEM "inex-topics.dtd">
<INEX-Topic topic-id="06" query-type="CAS" ct-no="034">
    <Title>
      <te>tig</te>
      <cw>Survey on Software Engineering</cw>
      <cw>
        software engineering survey, programming survey, programming tutorial,
        software engineering tutorial
      </cw>
      <ce>tig</ce>
      <cw>programming languages</cw><ce>sec</ce>
    </Title>
    <Description>
      Retrieve the article title from all articles which are a tutorial or survey on software
      engineering or programming dealing with programming languages.
    </Description>
    <Narrative>
      To be relevant an article should offer a tutorial or survey on software
      engineering or programming containing sections dealing with programming languages.
    </Narrative>
    <Keywords>
      survey, tutorial software engineering, programming language
    </Keywords>
</INEX-Topic>
```

Xpath queries:
"article[//tig//*/text('*')[contains('Survey on Software Engineering')]]//tig"
"article[//tig//*/text('*')[contains('software')][contains('engineering')][contains('survey')]
[contains('tutorial')]]//tig"
"article[//sec//*/text('*')[contains('programming')][contains('languages')]]//tig"

If a topic can not be expressed in Xpath queries, just keywords can use for searching.

- Searching process of Xpath queries

In EXIMA™ Supply, the Xpath queries processed as the following Figure 4.

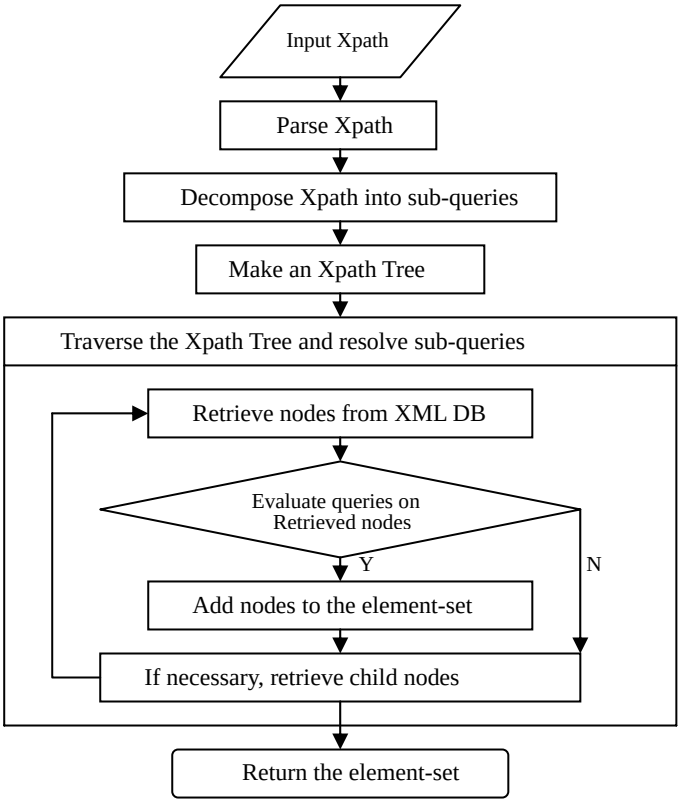

Figure 4: Flow of query processing in EXIMA™ Supply

As the above diagram illustrate, the given Xpath query is first parsed and then decomposed into several sub-queries. And based on these sub-queries, a query tree that represents the hierarchical relation of sub-queries is constructed. Once the query tree is constructed, the tree is traversed and evaluated to get the corresponding nodes. The traversing of query tree starts from the current context element. EXIMA™ Supply first retrieves the child elements of the current element as candidate elements from storage. And then the candidate elements are evaluated and elements that satisfy conditions are added to the element-set. The traversing is done recursively along to the child nodes of the query tree. If all nodes of the query tree are traversed and evaluated, the element-set is returned as the result of the search.

4. Results

We only submitted the results of CAS (content-and-structure) queries in INEX 2002. Figure 5 presents P-R

graphs for the evaluations results of the subsets of CAS topics, i.e., #01, #04, #05, #06, #11, #21. Applying the strict evaluation gave slightly higher score (average precision: 0.077) than the generalized evaluation result (average precision: 0.055) which provided by the official INEX organizers.

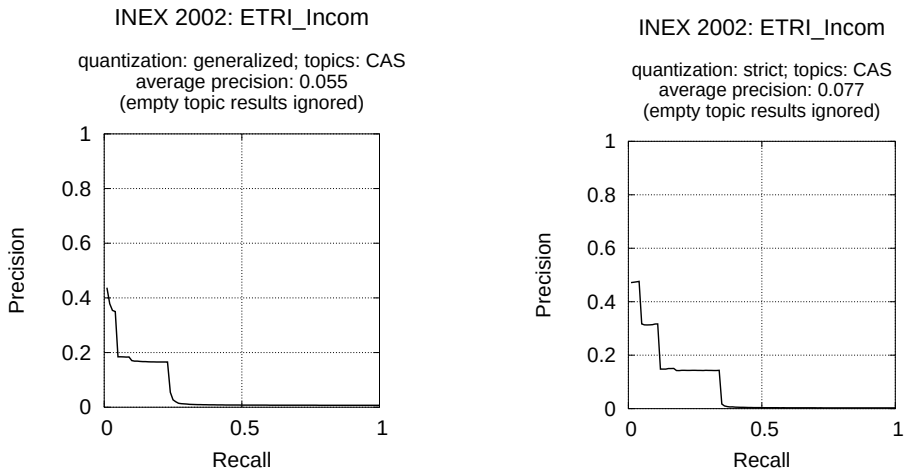

Figure 5: P-R Graph for (a) Generalized and (b) Strict CAS topic ignored empty results

Our overall, rather than empty topic results ignored, result showed relatively poor (average precision: 0.019). As shown in Figure 6 our results ranked with the 34th among 42 official submissions.

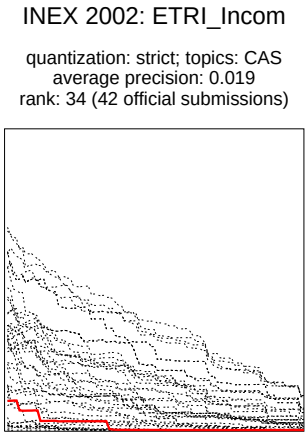

Figure 6: P-R Graph for Overall Results and Rank

5. Conclusion

In this paper, we described an approach of object-relational DBMS using EXIMA™ Supply for INEX test collections. Although EXIMA™ Supply has many benefits, for example, no delay in storing and searching XML documents, it showed relatively poor performance in overall evaluation at INEX 2002.

This result may be caused since the given topics had to be decomposed and modified to be processed by the Xpath processor in EXIMA™ Supply, and during this modification the original meaning of topics can be changed inevitably and some important information may missing. Some other possibilities are that because EXIMA™ Supply targets only for Korean, and we were not able to implement any aid tools of construction of indices, knowledge bases for INEX collection which will require to be investigating in the future study.

Acknowledgements

This work was supported in part by the Ministry of Information and Communication, Korea under the *Development of Information Distribution Framework Project*. Any opinions, findings, or conclusions expressed in this paper are those of the authors, and do not necessarily reflect those of the sponsor.

References

[1] Incom I&C Co. Ltd. EXIMA™ Supply. *Online available at*: http://www.incom.co.kr
[2] INEX homepage at: http://qmir.dcs.qmul.ac.uk/INEX/
[3] INEX {down, up} load area available at: http://ls6-www.cs.uni-dortmund.de/ir/projects/inex/download/
[4] Y. Despotopoulos, G. Patikis, J. Soldatos, L. Polymenakos, J. Kleindienst, and J. Geric. Accessing and transforming dynamic content based on XML: alternative techniques and a practical implementation. In W. Winiwarter, S. Bressan, and I.K. Ibrahim, editor, *Third International Conference on Information Integration and Web-based Applications and Services (IIWAS 2001).* Osterreichische Comput. Gesellschaft. 2001, pp. 95-105. Wien, Austria.
[5] T.T. Chinenyanga, and N. Kushmerick. Expressive retrieval from XML documents. *SIGIR Forum (ACM Special Interest Group on Information Retrieval)*, spec. issue., 2001, pp.163-71.
[6] S. Ha, and K. Kim. Mapping XML documents to the object-relational form. *ISIE 2001. 2001 IEEE International Symposium on Industrial Electronics Proceedings. Part Vol. 3*, 2001, pp. 1757-61. Piscataway, NJ, USA.
[7] S.J. Lim, and Y. K. Ng. An automated integration approach for semi-structured and structured data. *Proceedings of the Third International Symposium on Cooperative Database Systems for Advanced Applications. CODAS 2001*. IEEE Comput. Soc. 2000, pp. 12-21. Los Alamitos, CA, USA
[8] C. Zhang, J. Naughton, D. DeWitt, O. Luo, and G. Lohman. On supporting containment queries in relational database management systems. *ACM. SIGMOD Record (ACM Special Interest Group on Management of Data)*, Vol. 30, No. 2, June 2001, pp. 425-36.
[9] D. Shin. XML indexing and retrieval with a hybrid storage model. *Knowledge & Information Systems*, Vol. 3, No. 2, May 2001, pp. 252-61
[10] J. A. Miller, and S. Sheth. Querying XML documents. *IEEE Potentials*, Vol. 19, No.1, Feb.-March 2000, pp. 24-6.
[11] C. Petrou, S. Hadjiefthymiades, and D. Martakos. An XML-based, 3-tier scheme for integrating heterogeneous information sources to the WWW. In A. Cammelli, A. Tjoa, R.R. Wagner, editors, *Proceedings of Tenth International Workshop on Database and Expert Systems Applications. DEXA 99*. IEEE Comput. Soc. 1999, pp. 706-10. Los Alamitos, CA, USA.
[12] M. M. David. SQL-based XML structured data access. *Web Techniques*, Vol. 4, No. 6, June 1999, pp. 67-8, 70, 72.
[13] O. Alonso. Generation of text search applications for databases. An exercise on domain engineering. In C. Gacek, editor, *Software Reuse: Methods, Techniques, and Tools. 7th International Conference, ICSR-7. Proceedings (Lecture Notes in Computer Science Vol. 2319).* Springer-Verlag. 2002, pp. 179-93.
[14] M. Papiani, J. L. Wason, A. N. Dunlop, and D. A. Nicole. A distributed scientific data archive using the Web, XML and SQL/MED. *SIGMOD Record (ACM Special Interest Group on Management of Data)*, Vol. 28, No.3, Sept. 1999, pp. 56-62.
[15] N. Fuhr, N. Goevert, G. Kazai, and M. Lalmas. INEX: Initiative for the Evaluation of XML Retrieval, *ACM SIGIR Workshop on XML and Information Retrieval*, Tampere, Finland, August 2002.