Design of a Hybrid Search in the Online Knowledge Center

Jungkee Kim[1], Ozgur Balsoy[1], Marlon Pierce[2], and Geoffrey Fox[2]

Department of Computer Science[1]
Florida State University
FL32306, U.S.A.
jungkkim@cs.fsu.edu, ozgur@csit.fsu.edu

Community Grids Laboratory[2]
Indiana University
IN47404, U.S.A.
marpierc@indiana.edu, gcf@indiana.edu

ABSTRACT

A hybrid search in the Online Knowledge Center enables search for the content of linked documents in the metadata. In this paper, we present design issues for the combined search of unstructured data linked in semistructured data. We describe the problems of the initial design for the metadata storage and inquiry performance, and suggest a solution under a specific environment – the newsgroup service of the Online Knowledge Center system.

KEY WORDS

Hybrid search, metadata, XML, portal

1. INTRODUCTION

In the description of information using metadata, we may include some links associated with unstructured data. Conventional methods resolve the keyword search for unstructured data or metadata only. We suggest a solution, which combines metadata and unstructured inquiry, leveraging the functionalities in a relational database management system supporting metadata and unstructured data search.

Ever since the computer has been used for the storage of information, there have been attempts to make structured data from raw information - unstructured data. The Database clearly provides good structured data and the relational database has been developed to provide an efficient and convenient method of information storage and retrieval over several decades. Since the World Wide Web emerged as a Mecca for information resources, Web resources have become the recent targets for the information society. However, the main document formats for the Web are focused on how to show the information through the Web browser, not on the meaning of the Web contents. This makes it hard for a machine to understand the meaning of the resources on the Web. The answer to semantic problems of the Web coming from the World Wide Web society is the Extensible Markup Language (XML) [1] – a format that is both machine and human understandable. An XML document is semistructured data because it has a self-described format.

Since the XML format emerged as a de facto standard for the information exchange, many database companies have tried to support the XML data storage on their database management systems. Oracle, a leading database company also provides an XML-enabled database management system, Oracle9i [2]. In Oracle9i, XML documents can be stored into an object type of Character Large Object (CLOB), XMLType [3]. Besides native XML support in the database, there were a number of research studies into leveraging the existing relational databases, mapping XML instances into the relational tables [4, 5]. However, those studies are focused on general solutions of mapping the XML documents with different schemas. In the Online Knowledge Center (OKC) [6] project – an information Web portal system, we leverage the XML-enabled functionality of Oracle 9i, and non-automated XML document mapping to relational tables for the XML storage.

In a Web portal, a user may want to publish unstructured data to the portal attached metadata - for example, a title, authors, affiliations, addresses, and keywords for a paper. Most Web search engines don't provide search results combined with metadata. So, many users only get undesirably large search lists through the Web search engines. This paper suggests one solution to filter the unrelated items from the search in a portal system. Combining metadata and unstructured data search – *hybrid* search – may produce a desirable search result. For example, we want to find a light bulb story written by Thomas Edison. We can get the right result combining with the author of the metadata, Thomas Edison, and unstructured documents containing the word, light bulb through hybrid search.

An XML message-based newsgroup service is plugged in the OKC system. Through the Wizard user interface, the user is forced to produce a unified form of metadata that is associated with the attached file. The test prototype of the hybrid search [6] is based on the technology of the Oracle 9i XMLType and Oracle Text [7]. However, XMLType column data may cause

scalability problems. To reduce the problem, we can change the design for the semistructured search from XMLType column to elements and attributes mapping indexed tables. It will increase the efficiency because the Newsgroup schema is fixed and simple enough in the OKC system to convert between XML instances and tables. We also leverage the convert functions provide by the Oracle 9i SQL object package.

2. THE ONLINE KNOWLEDGE CENTER

The Online Knowledge Center (OKC) is an informational Web portal system that provides a Web component framework and distributed content management services. The architecture of the OKC portal is based on the Jetspeed [8] open source portal system – part of the Apache's Jakarta project [9]. It provides portlets and API's to define a single Web page or a group of Web pages for the dynamic components of the portal. The OKC server manages the portlets for presentation and layouts. For the management of the remote contents, Apache-Jakarta's Slide [10] model, which is based on Web-based Distributed Authoring and Versioning (Web-DAV) [11], is used. The architecture of the OKC system is shown in Figure 1.

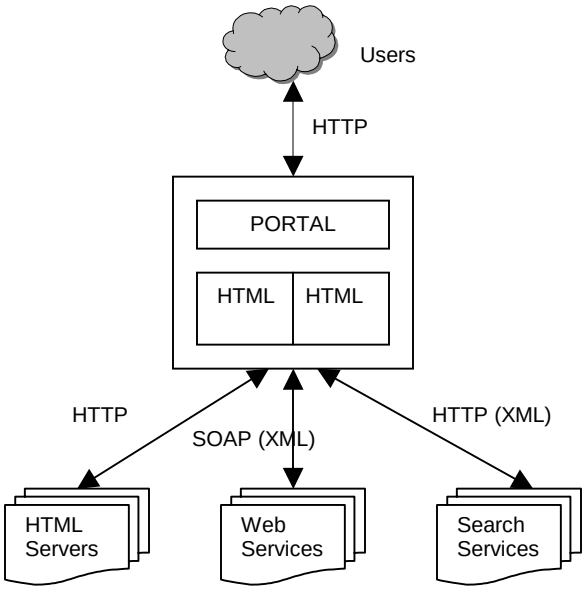

Figure 1. The Architecture of the OKC System

The OKC portal system also includes an XML messaging newsgroup service and content search. Newsgroup messages are exchanged in XML instances following a fixed XML schema through the JMS server. Sometimes, users may attach external files along with the main message. In the newsgroup service, two major event generators – the News Wizard and the E-mail Handler – produce XML event instances and convert e-mail messages to system events. Those procedures naturally conform to the internal XML schema, and the XML messages can be regarded as metadata. Through semistructured search against the metadata, we can extract the expected information. However, we need the information not only in the metadata but also in the attached documents, which may be associated with metadata. That requirement motivates the design of the hybrid or linked search.

Currently, a content search service is plugged in the OKC system. The content search is based on the Oracle Ultra Search [12], which consists of a crawler for the content collection and the Oracle Text for indexing the contents. The Oracle Ultra Search is an adequate tool for unstructured data search, but there is no method to refine the search keywords with metadata of unstructured documents. The crawler of the Oracle Ultra Search has the capability of primitive metadata collection through the search attributes. However, the search attribute only applies on particular columns of the relational database tables and meta tags of the Web resources.

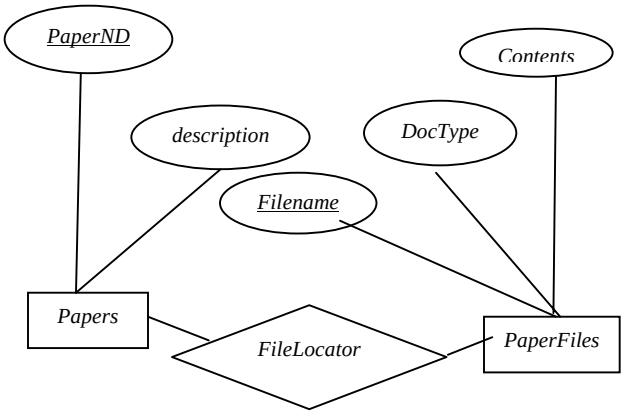

Figure 2. E-R diagram of the test prototype of hybrid search

3. THE HYBRID SEARCH

The initial design of the hybrid search is a simplified model - a paper search. This is a test prototype for evaluation of the hybrid search. The paper search is a content search across various types of documents, for example, Microsoft Word, Microsoft Power Point, PDF, and Post Script documents. Each document has metadata presented as an XML instance. Two Oracle database tables – *Papers* and *PaperFiles* - are used for the metadata and the documents. For the XML instances representing the metadata, an XMLType column of the Oracle9i is used. The BFILE large object type column is used for the external document table. Those large object

rows are indexed using Oracle Text. Through an Oracle Text index, we can search the target content. For definition of the relationship between the metadata and the documents, a relationship table – *FileLocator* - is created. Through the relationship table, we can query the related rows. The Entity-Relationship diagram for the test prototype of hybrid search is shown in Figure 2. In the E-R diagram, a *DocType* attribute is necessary for the filtering option of Oracle Text. Oracle Text filters binary files to pure text before making an index. Though a one-to-one relationship set is used for relation between the paper document and metadata, a one-to-many relationship set can be used for the newsgroup metadata with multiple attachments. With two data tables and a relationship table, we can query keywords of the content, which is associated with particular metadata through the nested subqueries. For example, we can find a document with a keyword "XML" and published in 2002. In relatively small size – about 100 rows – tables, nested subqueries are as fast as XMLType extraction queries and Oracle Text content queries.

Oracle 9i's XMLType objects provide convenient XPath grammar extraction query functions, but they have a scalability problem. For an XMLType object table with a relatively small number of rows, the time for the query processing is acceptable as shown in Figure 3. However, the query processing time grows as the number of rows increases, and the temporary tablespace consumes too much space. We used data-centric XML instances for the performance test. The size of the data table for the 10,000 rows of XML instances is about 16 M bytes, but the temporary tablespace should be increased to at least 300 M bytes for efficient operation under the Microsoft Windows 2000 operating system environment. The query time for relatively large number of data is also shown in Figure 3. More than one minute query time is not acceptable for the search query response.

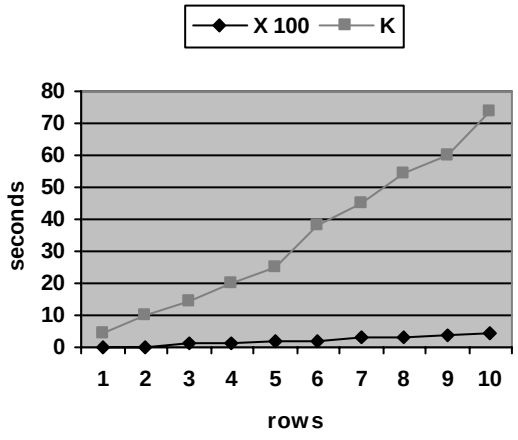

Figure 3. Processing time graph for the XMLType query

The result of the XMLType performance test naturally forced us to improve the design of the metadata query. Many studies have suggested methods for the general storage of XML instances and queries, but we apply fixed mapping SQL procedures. Our newsgroup metadata schema is unchangeable and simple enough to produce wrapping procedures without complicated preprocessors for the mapping and publication. We leverage an Oracle9i PL/SQL package – *DBMS_XMLGEN* for generating XML from relational tables. Mapping to relational tables makes the overall performance dependent on the performance and tuning considerations of the relational database. The indexing on the columns apparently produces reasonable query time. For example, the query against the 10,000 row data takes less than one second for the indexed column.

Another consideration for performance improvement is the cache. When we executed an XMLType object performance test, there was no time improvement for the same query during several attempts. That means there is no database system cache at least for the XMLType query. Many users usually inquire for similar categories and some of search keywords are used repeatedly. The attachment of the search cache will improve the search result. Current OKC system search leverages the Oracle Ultra Search and it produces a faster response against the same keyword query. The fast response for the content search is a major feature of the design of the search system. The improved architecture of the hybrid search is shown in Figure 4.

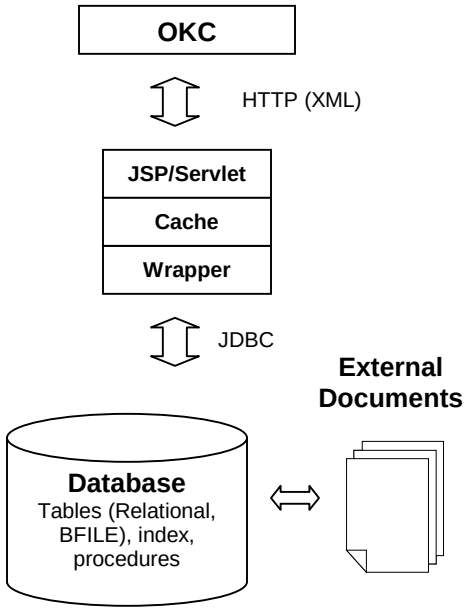

Figure 4. The architecture of the improved hybrid search

We also do research on the XML mediator system to solve the XML storage problem and integrate various database management systems and other collaborative tools.

4. CONCLUSION

In this paper, we describe an initial solution, which combines unstructured data and metadata inquiry, and its limitation for the expected enlargement. The initial design of the hybrid search was simple to implement, but may have a scalability problems due to the dependency on the Oracle XMLType object for the metadata manipulation. The improved hybrid search improves the scalability, but requires relatively more complexity. However, the biggest problem may come from the generation of the metadata. There is no automated manner to assign a proper metadata for a legacy unstructured system. We had to spend a lot of time to make a proper XML instance for each paper in the hybrid search test prototype. However, the new metadata can be produced by forcing the users to use the Wizard, and this discipline generates the appropriate and unique metadata for the unstructured data. A more general solution may be attained from further research on XML native storage or the mediator.

5. ACKNOWLEDGEMENT

The hybrid search is part of the Online Knowledge Center (OKC) project and the OKC is funded by the US Department of Defense's High Performance Computing Modernization Program through the Programming Environment and Training initiative. We gratefully acknowledge their support.

REFERENCES

[1] World Wide Consortium (W3C), Extensible Markup Language (XML) 1.0 (Second Edition), October 2000, http://www.w3.org/TR/2000/REC-xml-20001006.

[2] Oracle Corporation, Oracle9i Database, June 2002, http://www.oracle.com/ip/deploy/database/oracle9i/.

[3] Oracle Corporation, Oracle9i Application Developer's Guide – XML, June 2001.

[4] A. Deutsch, M. Fernandez, & D. Suciu, Storing Semistructured Data with STORED, In *Proceeding of SIGMOD Conference*, June 1999.

[5] J. Shanmugasundaram, K. Tufte, G. He, C. Zhang, D. DeWitt, & J. Naughton, Relational Databases for Querying XML Documents: Limitations and Opportunities, In *International Conference on Very Large Data Bases*, September 1999.

[6] O. Balsoy, J. Kim, & others, The Online Knowledge Center: Building a Component Based Portal, Accepted in *International Conference on Information and Knowledge Engineering*, June 2002, Word Wide Web, http://grids.ucs.indiana.edu/ptliupages/publications/OKCPaper1x1.pdf

[7] Oracle Corporation, Oracle Text Application Developer's Guide Release 9.0.1, June 2001.

[8] Apache Software Foundation, Jetspeed Overview, June 2002, World Wide Web, http://jakarta.apache.org/jetspeed/site/index.html

[9] Apache Software Foundation, The Jakarta Project, June 2002, World Wide Web**,** http://jakarta.apache.org/

[10] Apache Software Foundation**,** The Jakarta Slide Project, June 2002, World Wide Web, http://jakarta.apache.org/slide

[11] WebDAV organization, WebDAV Resources, June 2002, World Wide Web, http://www.webdav.org

[12] Oracle Corporation, Oracle Ultra Search Architecture, Technical white paper, May 2001, WWW, http://otn.oracle.com/docs/products/ultrasearch/.