An XQuery Engine for Digital Library Systems

Ji-Hoon Kang
Dept. of Computer Science, Chungnam National University
220 Gung-Dong, Yuseong-Gu, Daejeon, 305-764, South Korea
+82-42-821-5447
jhkang@cs.cnu.ac.kr

Chul-Soo Kim
Dept. of Computer Science, Chungnam National University
220 Gung-Dong, Yuseong-Gu, Daejeon, 305-764, South Korea
+82-42-821-7447
chulsu1@cs.cnu.ac.kr

Eun-Jeong Ko
Dept. of Computer Science, Chungnam National University
220 Gung-Dong, Yuseong-Gu, Daejeon, 305-764, South Korea
+82-42-821-7447
brain08@cs.cnu.ac.kr

Categories and Subject Descriptors
H.3.7 [Information Systems]: Digital Libraries

General Terms
Design, Languages

Keywords
Digital Library System, XML, XQuery, Information Retrieval

ABSTRACT
XML is now a standard markup language for web information. Many application areas are producing XML documents on the web. This situation urges digital library systems to deal with not only typical text documents but also XML documents. XML documents are semi-structured. Some queries based on the structures are useful and necessary.

MPEG-7 is a metadata standard for multimedia objects. MPEG-7 metadata can describe some features such as color histogram of image, so that a multimedia digital library system using MPEG-7 for metadata representation can provide content-based search for multimedia objects. MPEG-7 is defined by XML schema. In order to retrieve MPEG-7 metadata, a query language for XML data is required.

A standard query language is very helpful for interoperability among digital library systems over the Internet. XQuery, which has been influenced from most of the previous XML query languages, is a forthcoming standard for querying XML data.

In this paper we propose an XQuery Engine as depicted in the figure that can be used as an XQuery processing module in a digital library system that supports XML documents. We assume generic digital library system architecture. It consists of four modules: a user interface, an XQuery Engine, an Information retrieval Engine, and an XML Repository. The user interface module gives a user an easy way to search XML documents and transforms a given user query to an equivalent XQuery. The XQuery Engine module takes an XQuery as input and provides a query plan for an information retrieval module as output. The information retrieval engine executes a query plan by communicating with the XML repository, which stores XML documents.

The XQuery Engine parses an input XQuery and constructs a syntax tree for the query. Then, it transforms the syntax tree into a query plan, called a Primitive Operation Tree (POT). Each node of a POT represents an atomic operation in terms of the information retrieval engine and can be interpreted and processed by the information retrieval engine. The result set is given back to the XQuery engine, which in turn transforms the result into an XML document of the form being required by the user interface. The final result in XML is returned back to the user interface.

Our approach has the following useful aspects. First, any user interface that generates XQuery is able to access any digital library system including our XQuery Engine. Second, we define a set of primitive operations for POTs so that they can become a standard interface between an XQuery Engine and an Information Retrieval Engine for our generic digital library system that supports XML documents. Third, some query optimizations over POTs can be done in the XQuery Engine so that better searching performance is expected.

Currently we are developing an XQuery Engine prototype. It will be installed inside an MPEG-7 based Digital Library System that supports content-based searching for images. The XQuery Specification is an ongoing working draft and is not completed yet. Since the current version of the XQuery specification does not define full functions for information retrieval, we need to extend XQuery syntax by adding some functions such as rankby()

ACKNOWLEDGEMENTS
This research is supported by Software Research Center in Chungnam National University.