

MPEG-7 기반 디지털도서관 시스템을 위한 XQuery Engine

고은정^o 김철수 강지훈
충남대학교 컴퓨터학과

brain08^o@cs.cnu.ac.kr, chulsul@matabuild.co.kr, jhkang@cs.cnu.ac.kr

An XQuery Engine for MPEG-7 Based Digital Library Systems

Eun-Jung Ko^o, Chul-Su Kim, Ji-Hoon Kang
Dept. of Computer Science, Chungnam National University

요 약

MPEG-7 메타데이터는 이미지의 칼라 히스토그램과 같은 멀티미디어 객체의 속성을 기술할 수 있어서, MPEG-7을 사용하는 멀티미디어 디지털도서관 시스템은 멀티미디어 객체에 대한 내용기반 검색을 가능하게 해준다. MPEG-7 메타데이터는 XML 데이터이며, 이에 대한 질의를 위해서는 XQuery와 같은 XML 데이터를 위한 질의어가 필요하다. 본 논문에서는 MPEG-7 데이터에 대한 XQuery 질의를 처리할 수 있는 XQuery 엔진을 제안한다. 엔진은 XQuery를 입력 받아 정보검색을 위한 POT (Primitive Operation Tree)를 생성한다. 우리의 방식은 표준질의어인 XQuery를 처리함으로써 디지털도서관 시스템 사이의 상호 운영성을 확보할 수 있으며, XQuery 엔진과 정보검색기 사이의 표준 인터페이스로 사용할 수 있는 POT를 위한 primitive operation 집합을 정의하였으며, 또한 POT 상에서의 질의 최적화를 수행하여 검색 성능 향상을 도모하였다.

1. 서론

MPEG-7[1],[2]은 멀티미디어 데이터들을 내용 기반으로 검색하고, 처리하는 것을 가능하게 하는 멀티미디어 콘텐츠들 위한 메타데이터 표현 방식에 관한 표준이다. MPEG-7 메타데이터는 이미지의 칼라 히스토그램과 같은 멀티미디어 객체의 속성을 기술할 수 있다. 이러한 MPEG-7의 메타데이터의 구조는 XML[3]의 형태를 잘 정의하고 특성을 잘 나타낼 수 있는 XML Schema[4]로 정의되어있고, 응용분야에 필요한 Descriptor와 Descriptor Scheme을 이용하여 데이터를 기술할 수 있다.

이러한 MPEG-7을 사용하는 멀티미디어 디지털도서관 시스템은 멀티미디어 객체에 대한 내용기반 검색을 가능하게 해준다. MPEG-7 메타데이터는 XML 데이터이며, 이에 대한 질의를 하기 위해서는 XQuery1.0[5](XML Query Language)와 같은 MXL 데이터를 위한 질의어가 필요하다.

본 논문에서는 MPEG-7 데이터에 대한 XQuery를 질의할 수 있는 MPEG-7기반 디지털 도서관 시스템을 위한 XQuery Engine을 제안한다. 기존의 XPERANTO와 같은 XQuery Engine[6]은 XQuery의 결과 생성에서 단순 검색을 지원하는 것에 반해, 본 논문에서 제안하는 XQuery Engine은 이미지의 검색 시, 이미지의 contents검색을 지원하여 디지털 도서관 시스템에서 이미지의 모양, 색등의 유사도 검색과, 정보 검색을 이용한 ranking을 지원할 수 있게 한다.

XQuery Engine은 XQuery를 입력으로 받아, 분석한 다음 정확한 질의의 의미를 나타내는 Syntax Tree를 생성한다. 생

성된 Syntax Tree로부터 검색연산을 위한 POT(Primitive Operation Tree)와 검색결과를 만들어주기 위한 태그정보를 추출한다. 검색의 효율을 높이기 위해, POT로부터 Optimized Primitive Tree를 생성하여 정보검색기로 보낸다. 정보 검색기로부터 돌려 받은 결과는 태그 정보를 이용해서 최종 결과물을 만든다.

본 논문의 구성은 다음과 같다. 2절은 관련연구를 알아보고, 3절에서는 MPEG-7기반의 이미지 데이터를 위한 디지털도서관 시스템을 소개하고, 4절에서는 본 논문에서 정의한 Primitive Operations에 대해 설명하고, XQuery Engine의 구조, 설계에 관하여 알아본다. 5절에서는 결론을 내린다.

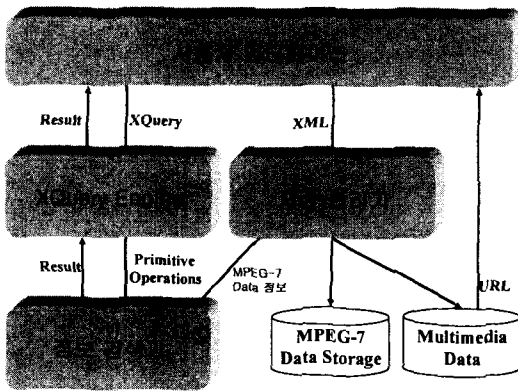
2. 관련연구

XML 데이터를 저장하고, 저장되어있는 XML 데이터에 질의를 할 수 있도록 지원하는 시스템들이 많이 나타나고 있다. 이러한 시스템들은 대부분 Database에 XML 데이터를 저장하고, XML Query Language를 이용하여 질의를 한다. 이러한 시스템들은 Relational Database위에 가상의 view를 이용하여 XQuery를 Relational Database에서 사용할 수 있는 SQL 질의로 변환하는 형태를 취하고 있다. 대표적인 시스템은 XPERANTO[6]와 XTABLES[7]가 있다. 위의 시스템들은 사용자가 작성한 XQuery를 임의로 정의된 중간 Data Model로 변환하여 그것을 SQL로 변환하는 점과 XQuery의 최종 결과인 XML을 구성하기 위해서 태그정보를 따로 저장한다는 점이 본 논문의 XQuery Engine에서 Primitive Operation Tree를 생성하고 태그정보를 추출하는 것과 비슷하다. 그러나 가상 view를 위한 Data Model은 XQuery의 의미를 정확하게 표현

하고 SQL로 변환이 가능 한 단위로써 노드를 만드는 반면, Primitive Operation Tree의 노드는 검색에서 바로 사용된다는 점이 다르고 또한 검색의 결과로서 가져 올 데이터는 저장 매체가 File System, Relational Database, Binary Format 등 어느것 인지 상관없이 저장기에서 직접 접근해서 가져오는 것에서 차이가 있다.

3. MPEG-7 기반의 이미지 데이터를 위한 디지털 도서관 시스템

MPEG-7 기반의 이미지 데이터를 위한 디지털 도서관시스템의 구조는 [그림1] 에서 보는 바와 같이 사용자 인터페이스, XQuery Engine, 정보검색기, 저장 관리기 등 4부분으로 나뉘어 진다.



[그림1] MPEG-7기반의 이미지 데이터를 위한 디지털 도서관 시스템

사용자는 사용자 인터페이스를 통해 XQuery를 작성하여 원하는 이미지에 관한 질의를 할 수 있다. 사용자로부터 만들어진 XQuery는 XQuery Engine의 입력으로 들어와서, XQuery Engine의 내부 분석 및 처리에 의해 정보 검색기에서 검색이 가능한 Executable Primitive Operation Plans로 변형이 되어 정보검색기에 넘겨준다. 정보검색기는 이렇게 받은 Plans을 그 순서에 맞게 저장 관리기를 통해 검색을 하여 검색 결과를 얻는다. 검색결과는 XQuery Engine에 전달하여 태깅을 한다. 태깅을 한 결과는 사용자 인터페이스를 통해 사용자에게 검색결과로 보여주게 된다.

MPEG-7기반의 이미지 관리자는 사용자의 검색대상이 될 MPEG-7 메타데이터를 사용자 인터페이스와 저장 관리기를 통해 생성하고 저장하는 등의 데이터를 관리할 수 있다.

4. XQuery Engine

4.1 Primitive Operations

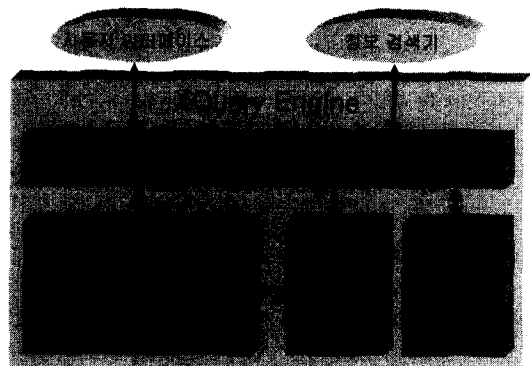
MPEG-7기반의 이미지 검색 시스템에서, 정보검색을 사용하여 이미지를 검색하기 위해서는, 사용자가 입력한 XQuery를 정보 검색기가 사용할 수 있는 형태: Primitive Operations으로 변환하여 사용하는 것이 효율적이다. 그래서 MPEG-7 메타데이터를 분석해서 검색의 대상이 될 수 있는 항목들을 발췌하고, 그 항목들을 대상으로 정보 검색기에서 이용할 수 있는 검색 모델을 설계하였다. 그리고 검색모델과 사용자 인터페이스에서 생성할 수 있는 XQuery를 가지고, XQuery에서

원하는 정보를 검색하고 요청할 수 있도록 검색모델을 이용할 수 있는 연산자들을 정의하였다. [표1]에 정보검색기에서 사용하기 위한 Primitive Operations을 나열하였다.

[표1] Primitive Operations

	Primitive Operations
1	_MatchElement(path, string)
2	_LessThanElement(path, value)
3	_LessThanEqualElement(path, value)
4	_GreaterThanElement(path, value)
5	_GreaterThanEqualElement(path, value)
6	_Contain(path, string)
7	_SimilarTo(path, string)
8	_MatchTitle(path, type, string)
9	_ContainTitle(path, type, string)
10	_MatchStructuredAnnotation(path, definition, string)
11	_ContainStructuredAnnotation(path, definition, string)
12	_Or(Temp1, Temp2)
13	_And(Temp1, Temp2)
14	_AndNot(Temp1, Temp2)
15	_Return(int, String)

4.2 XQuery Engine의 구조



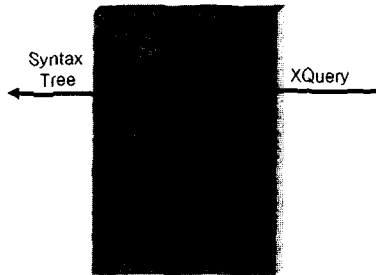
[그림2] XQuery Engine의 구조

XQuery Engine의 전반적인 구조는 [그림2]와 같다. 사용자 인터페이스를 통해 들어오는 XQuery는 Interface Manager를 통해 Query Analyzer에 전달이 되고, Query Analyzer에서는 XQuery를 분석하게 된다. 분석이 된 XQuery는 POTree Generator를 통해 정보 검색기에서 사용할 수 있는 query plan으로 변환하여 Interface Manager를 통해 정보검색기로 전달

한다. 정보검색기에서 검색한 결과는 Interface Manager를 통해 XML생성기로 전달된다. XML생성기에서는 태깅 정보를 이용하여 검색결과를 태깅하여 XML형태로 만들어준다.

4.3 질의 분석기(Query Analyzer)

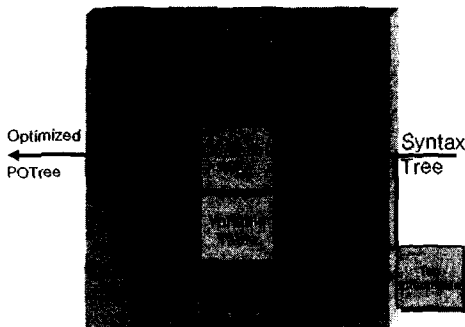
질의 분석기에서는, 입력으로 들어오는 XQuery가 XQuery의 문법에 맞는지 파서를 통해 parsing하면서, Syntax Tree를 생성한다. Syntax Tree는 XQuery를 분석하여 XQuery의 의미를 정확하게 나타내고, Primitive Operation Tree를 생성하기 용이하도록 본 논문에서 설계하였다. [그림3]은 질의 분석기의 구조를 나타내고 있다.



[그림3] 질의 분석기의 구조

4.4 POTree생성기(POTree Generator)

POTree는 Syntax Tree를 입력으로 받아서, Optimized Primitive Operation과 태그정보를 만든다. POTree생성기는 연산추출기, 태깅정보 추출기, 연산 최적화기로 구성되어 있다. 연산 추출기는 Syntax Tree를 받아서 Primitive Operation Tree와 해당 Variable과 Path정보를 저장하는 Variable Table을 뽑아낸다. 태그정보 추출기는 Syntax Tree를 받아서 XQuery의 최종 결과 물인 XML을 구성하기 위해 필요한 정보를 추출한다. 연산 최적화기는 POTree를 분석하여 성능 향상을 위해 연산의 순서를 변경한다. 연산순서의 변경 방법은 다음과 같다. And연산은 순서를 바꾸어도 전체 결과에는 변화가 없으므로, ((A or B) and (C or D)) and E 일 경우, 순서를 바꾸어서 E and ((A or B) and (C or D))처럼 만든다. 이럴 경우, E연산의 결과가 null 일 경우, 수행 해야 할 연산이 많은 뒤의 연산을 수행하지 않아도 된다. POTree생성기의 구조는 그림[4]와 같다.



[그림4] POTree생성기

4.5 인터페이스 관리자(Interface Manager)

XQuery Engine은 POTree생성기에서 만든 Primitive Operation Tree를 네트워크 상에서 독립적으로 운영이 되고 있는 정보검색기로 전달을 해야 한다. 그래서 인터페이스 관리자에서는 Primitive Operation Tree를 XQuery Engine과 정보검색기간의 정해놓은 형태에 맞게 변경하여, 정보검색기로 보낸다.

또한 인터페이스 관리기는 XQuery의 검색결과를 저장관리기로부터 받아서, 결과를 정보검색기로부터 받는 역할도 한다.

4.6 XML 생성기(XML Constructor)

인터페이스 관리기가 정보검색기로부터 검색 결과를 받으면 XML생성기로 검색 결과를 보내서, XQuery를 파싱할 때, 저장해둔 태깅 정보를 보고 검색 결과에 태깅을 하여 검색결과를 XML형태로 만든다.

5. 결론

MPEG-7 기반의 이미지 데이터를 위한 디지털 도서관 시스템은, XQuery를 이용하여 메타데이터의 검색을 지원하는 시스템이다. MPEG-7은 차세대 멀티미디어의 표준이며, XQuery는 XML을 사용하는 어떠한 시스템에서도 상호 운용성이 가능하도록 만들어진 질의 언어이다. 본 논문에서는 이러한 MPEG-7 기반의 멀티미디어 메타데이터의 검색을 지원하기 위한 XQuery Engine을 구현하였다.

본 논문의 의의는 다음과 같다. 첫째, XQuery를 생성하는 어떠한 사용자 인터페이스도 우리의 XQuery 엔진을 가지고 있는 디지털도서관 시스템에 접근이 가능하므로 디지털도서관 시스템 사이의 상호 운영성을 확보 할 수 있다. 둘째, XQuery 엔진과 정보검색기 사이의 표준 인터페이스로 사용할 수 있는 POT를 위한 primitive operation 집합을 정의하였다. 셋째, 연산 순서 변경과 같은 POT 상에서의 질의 최적화를 수행하여 검색 성능 향상을 도모하였다.

6. 참고문헌

- [1] ISO/IEC, Overview of the MPEG-7 Standard(version 6.0), approved, N4509,, December 2001.
- [2] ISO/IEC, Introduction to MPEG-7(version4.0), approved, N4675, March 2002.
- [3] W3C, Extensible Markup Language (XML) Version 1.0, Recommendation, February1998.
- [4] W3C, XML Schema Part0:Primer, Recommendation, May 2001.
- [5] W3C, XQuery 1.0: An XML Query Language, Working Draft, November, 2002.
- [6] J. Shanmugasundaram, J. Kiernan, E. Shekita, C. Fan, & J. Funderburk, "Querying XML Views of Relational Data," Proc. 27th VLDB, Roma, Italy, September 2001.
- [7] C. Fan, John Funderburk, Hou-in Lam, Jerry Kiernan, Eugene Shekita, Jayvel Shanmugasundaram, "XTABLES: Bridging Relational Technology and XML", IBM Systems Journal, 2002.
- [8] W3C, XML Path Language(Xpath)2.0, Working Draft, November 2002.
- [9] W3C, XML Query Use Cases, Working Draft, November, 2002.