

# XML 데이터의 갱신을 지원하는 확장된 SQL

조영주<sup>0</sup> 강지훈  
충남대학교 컴퓨터학과  
yjcho8@hanmail.net, jhkang@cs.cnu.ac.kr

## An Extended SQL for Updating XML data

Young-Joo Cho<sup>0</sup> Ji-Hoon Kang  
Dept. of Computer Science, Chungnam National University

### 요 약

XML은 자기 서술적인 표현 언어라는 중요한 장점을 가지고 있기 때문에 인터넷 관련 응용들에서 정보의 표현 및 문서 교환의 표준으로 빠른 속도로 자리잡아 가고 있다. 본 논문에서는 SQL에 XPath 표현 중 일부를 추가하여 SQL을 확장함으로써 XML 문서에 대한 질의를 가능하게 하였고, 또한 SQL의 CRUD (Create, Read, Update, Delete) 기능을 그대로 사용함으로써 XML 문서의 변경이 가능한 XML 질의를 위한 새롭게 확장된 SQL을 제안하였다. 그리고 폭 넓게 사용되고 있는 ORDBMS 환경에서 XML 색인 정보를 자동 생성 관리하는 XML 저장 관리기 및 확장된 SQL 처리기를 설계하고 구현하였다. 본 논문에서 제안하는 확장된 SQL 처리기는 XML 원문 자체의 보관을 원칙으로 하여 완벽한 문서 복구 등에 따르는 부담을 제거하였고, XML 저장 문서에 대한 원본성 및 무결성을 쉽게 유지할 수 있도록 하였으며, 색인 정보를 자동 생성 관리함으로써 파일 처리 시간과 XML 트리 운행 시간 및 XPath 파싱 처리 시간을 줄였다. 또한, "path" 라는 키워드를 추가로 관리함으로써 ORDBMS 환경에서 XML 질의를 위한 SQL 변환 과정에서 PATH의 깊이 만큼 조인이 중첩되는 문제를 배제하여 조인이 발생할 가능성을 줄이는 방법을 제시하였다.

### 1. 서론

고정된 문서 구조를 가지는 HTML의 표현의 한계를 극복하고 무거운 처리 모드를 필요로 하는 SGML의 복잡한 구문을 제거한 웹 문서 교환에 가장 적합한 XML은 자기 서술적인 특성을 가지는 잘 정의된 마크업 언어이다. 웹상에서 문서 표현과 교환을 위한 표준으로 XML의 사용이 급증하면서 XML 질의어에 대한 필요성은 더욱 중요한 이슈가 되었고, XPath[1], XML-QL[3], XQL[4] 등 관련 연구들이 이루어졌으며, W3C에서는 XQuery[6] 버전 1.0을 Working Draft로 발표하였다.

그러나 XQuery를 포함한 기존 XML 질의어들은 XML 문서를 대상으로 선택, 추출, 재구성, 조합 등의 다양한 검색 기능과 XML 부분 지정을 위한 풍부한 질의 기능을 포함하고 있지만 상대적으로 복잡한 질의 처리기를 요구할 뿐만 아니라[7] XML 변경 기능까지 포함하는 질의는 지원하지 않고 있다.

본 논문에서는 XML 문서에 대한 질의가 가능하도록 기존 SQL에 XPath 표현 일부를 추가함으로써 새롭게 확장된 SQL을 제안하였으며, ORDBMS 환경에서 XML 저장 관리기 및 확장된 SQL 처리기를 설계하고 구현하였다. 그리고 색인 정보를 활용함으로써 XML 질의를 위한 SQL 변환 과정에서 PATH의 깊이 만큼 조인이 중첩되는 문제[8]를 배제하여 검색 성능을 향상시켰다.

### 2. 관련 연구

현재까지 제안된 XML 질의어로는 [그림1]과 같이 LOREL[2], XML-QL[3], XQL[4], Quilt[5], XQuery[6] 등이 있으며 대부분은 풍부한 표현력을 지원하는 대신 기존 SQL 사용자들에게는 덜 친근하고 복잡하며, XQuery를 변경 개념까지 확장한 연구[9] 등이 이루어졌으나 아직까지는 XML 변경을 완벽하게 지원하는 제품은 없다.

LOREL 1997	<pre>SELECT &lt;offer&gt; Z/name, P/name, P'/price&lt;/offer&gt; FROM P in catalog/product, Z in discount_stores/store, Z/storecatalog/product P' WHERE P/category="camera" and P/make="canon" and P'/id = P/id</pre>
XML-QL 1998	<pre>WHERE &lt;book&gt;&lt;publisher&gt; &lt;name&gt;Addition-Wesley&lt;/name&gt;&lt;/publisher&gt; &lt;title&gt; \$t &lt;/title&gt;&lt;author&gt; \$a &lt;/author&gt;&lt;/book&gt; IN "www.a.b.c/bib.xml" CONSTRUCT &lt;result&gt;&lt;author&gt;\$a&lt;/&gt;&lt;title&gt;\$t&lt;/&gt;</pre>
XQL 1998	<pre>/book/author, /author, author, author//first_name, bookstore/*/title, author/**/, //author, @style</pre>
Quilt 2000	<pre>FOR \$b IN document(http://www.biblio.com/books.xml)//book, \$a IN \$b/author WHERE \$a/filename = "Crockett" and \$a/lastname = "Johnson" RETURN \$b</pre>
XQuery 2001	<pre>FOR \$a IN distinct-value (\$bib0/book/author/data()) RETURN &lt;biblio&gt; &lt;author&gt;{ \$a }&lt;/author&gt; { FOR \$b IN \$bib0/book, \$a2 IN \$b/author/data() WHERE \$a = \$a2 RETURN \$b/title } &lt;/biblio&gt;</pre>

[그림1] XML 질의어의 표현

### 3. XML 질의를 위한 확장된 SQL

#### 3.1 질의 정의

XML 문서는 개념적으로 트리 형태의 계층 구조로 표현되며 XML 문서내의 엘리먼트, 애트리뷰트, 패스 정보를 포함하는 내용 및 구조 정보 지정용 위해서는 경로 표현의 사용이 필수적이다. 따라서 본 논문에서는 XML 문서내의 특정 노드를 지정하기 위해 표준에서 권고하는 XPath의 부분을 취하였고, SQL을 확장하여 XML에 대한 CRUD 까지 자연스럽게 가능하도록 하였다.

XML 질의를 위한 확장된 SQL Syntax[11]는 오라클에서 제공하는 SQL 문법의 키킬 부분에서 [그림2]와 같이 XML 부분 지정용 위한 XPATH\_EXPR 표현을 추가로 정의하여 확장한 것이며, 해당 노드의 전체 경로를 질의하고 조인 연산을 줄이기 위한 방안으로 'path'라는 키워드를 사용하였다.

```

1. xpath_expr ::= xdoccolname # xpathstr
2. xpathstr ::= / | / | 'path' | xpath | (xpath / ) @ attname |
   xpath // @ attname | xpath // element
3. xpath ::= void | (xpath / ) element
4. xdoccolname ::= adstrings
5. element ::= adstrings
6. attname ::= adstrings
7. adstrings ::= ( alpha | '_' ) ( alpha | digit | '_' )
8. alpha ::= a|b|...|z|A|B|...|Z
9. digit ::= 0|1|...|9
    
```

[그림2] XPATH\_EXPR 표현의 GRAMMAR 정의

#### 3.2 질의 표현

질의 표현은 [그림3]과 같은 표현들이 가능하며 기본적으로 SQL 문장의 키킬 부분에 XPATH\_EXPR 표현을 확장한 것을 제외하고는 SQL Syntax 와 동일하다.

```

SELECT xdoc#사업자등록증/성명
FROM xtab
WHERE key like '10%'
AND xdoc#사업자등록증/사업의종류/업태='서비스';

SELECT xdoc#@국적코드
FROM xtab
WHERE key like '10%'
AND xdoc#사업자등록증/성명='홍길동';

SELECT xdoc#path
FROM xtab
WHERE rownum < 10;

UPDATE xtab
SET xdoc#사업자등록증/사업의종류/업태='서비스'
WHERE key like '10%'
AND xdoc#사업자등록증/사업의종류/업태='서비스';

INSERT INTO xtab ( key, xdoc )
( SELECT key, ydoc# FROM ytab WHERE key like '10%' );

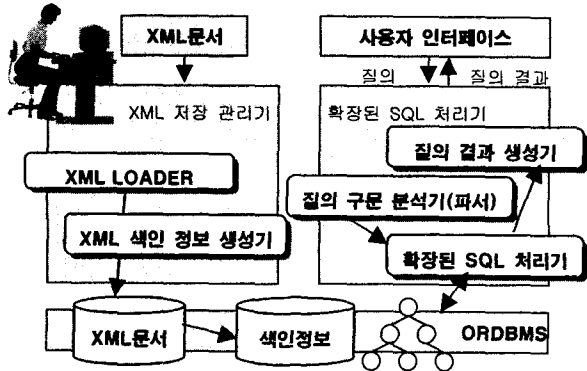
DELETE FROM xtab
WHERE xdoc#사업자등록증/사업의종류/업태='서비스';
    
```

[그림3] 확장된 XML 질의어의 표현

### 4 XML 질의를 위한 확장된 SQL 처리기

#### 4.1 시스템 구조

본 논문에서 제안하는 XML 질의를 위한 확장된 SQL 처리기의 범위는 [그림 3]과 같이 XML 저장 관리기와 확장된 SQL 처리기로 크게 구분되며, XML 저장 관리기는 XML 로더와 XML 색인 정보 생성기로 구성되고 확장된 SQL 처리기는 질의 구문 분석기(파서)와 질의 결과 생성기로 구성된다.

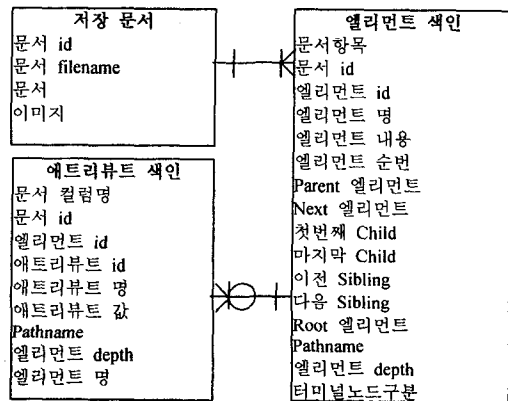


[그림4] XML 질의를 위한 확장된 SQL 처리기

XML 저장 관리기는 XML 로더를 통해 대상 XML 문서를 원문 자체로 데이터베이스에 저장하며 이 때 저장 문서에 걸려 있는 DB 트리거 형태의 XML 색인 정보 생성기를 통해 해당 XML 문서는 DOM 파싱되고 그 결과는 XML 색인 테이블에 별도로 관리된다. XML 질의를 위한 확장된 SQL 처리기는 SQL 문장과 유사한 형태로 제출되는 질의 문장을 입력으로 받아 파싱하고 확장된 SQL 처리기를 통해 색인 정보를 참조하여 질의를 수행하며 질의 결과 생성기를 통해 최종적인 결과를 반환한다.

#### 4.2 XML 색인 정보의 구성

XML 색인 정보는 엘리먼트, 애트리뷰트 및 PATH 정보를 기본으로 하고 [그림4]와 같이 검색 성능 향상을 위한 추가적인 정보들이 포함된다. 이러한 색인 정보의 관리의 파일 처리 시간과 XML 파싱 처리 시간을 줄여주며 조인 연산을 줄여 전체적인 성능을 향상시킨다.



[그림5] XML 색인 정보의 구성

4.3 확장된 SQL 처리 모듈

확장된 SQL 처리 모듈은 질의 종류에 따라 선택, 수정, 삽입, 삭제 질의어 처리 모듈로 크게 구성되며, SQL 문자열 변환을 위해 XPATH\_EXPR 표현이 포함된 위치에 따라 각각 해당되는 XQPATH\_EXPR 처리 모듈을 호출한다. XQPATH\_EXPR 표현은 [그림2]에서 정의된 GRAMMAR를 기준으로 모두 18가지 형태가 나타날 수 있고, 포함된 위치에 따라 SQL 문자열 변환 기준은 정해진다[11].

아래 [그림6]은 선택 질의어 처리 알고리즘의 수행 과정을 간략하게 나타낸 것이다.

1. 조건부에 포함된 XQPATH\_EXPR 표현 처리
  - A. 조건부에 포함된 XQPATH\_EXPR 표현을 각각 Subquery 형태의 SQL 문자열로 변환
2. 선택부에 포함된 XQPATH\_EXPR 표현 처리
  - A. 선택부에 포함된 XQPATH\_EXPR 표현을 SQL 문자열로 변환
  - B. A에서 변경되는 부분을 조건부에 반영
  - C. FROM 절에 색인 테이블 참조 추가

[그림6] 선택(SELECT) 질의어 처리 알고리즘

4.4 PATH 깊이만큼 중첩되는 조인 연산 문제

XML 질의를 위해 필수적인 경로 표현은 RDBMS 저장 환경에서 XML 질의 처리를 위한 SQL 변환 과정에서 PATH 깊이만큼 중첩되는 조인 문제를 발생시킨다[8][10]. 알려진 바와 같이 SQL 문장에서 조인은 상대적으로 처리 비용을 많이 필요로 하는 연산으로 과도한 조인은 성능의 저하로 연결된다.

```
<XML질의를 위한 확장된 SQL>
Select id, name, xdoc#/'e1/e2/e3, xdoc#/'e1/e2/e4 from xtab
where id like '10%' and xdoc#/'e1/e2/e3 = 'aaa'
and xdoc#/'e1/e2/e4 = 'bbb';
<SQL 문자열 형태로 변환>
Select id, name,
decode(path, '/e1/', decode(elename, 'e2', nvl(evalue, ''), ''), ''),
decode(path, '/e1/', decode(elename, 'e3', nvl(evalue, ''), ''), ''),
from xtab, xelement
where id like '10%'
and xtab.doc_id in ( select doc_id
from xelement
where doc_id = xtab.doc_id
and doccolname = 'xdoc'
and path = '/e1/e2/'
and elename = 'e3'
and evalue = 'aaa' )
and xtab.doc_id in ( select doc_id
from xelement
where doc_id = xtab.doc_id
and doccolname = 'xdoc'
and path = '/e1/e2/'
and elename = 'e4'
and evalue = 'bbb' )
and xtab.doc_id = xelement.doc_id(+)
and doccolname = 'xdoc';
```

[그림7] SQL 문자열 형태로 변환하는 예제

본 논문에서 제안하는 시스템은 [그림7]에서 알 수 있듯이 XML 부분 지정을 위한 경로 표현 처리에서 PATH 깊이와 무관하게 SQL 문자열 변환이 이루어진다. 즉, 루트로부터 해당 노드까지의 전체 경로를 나타내는 색인 정보 "path" 라는 키워드를 활용함으로써 PATH 깊이 만큼 중첩되는 조인 연산 문제

를 해결하였다.

5. 결론 및 향후 연구

본 논문에서는 SQL에 XML 부분 지정을 위한 PATH 표현을 확장함으로써 XML 문서의 엘리먼트, 에트리뷰트 및 구조 정보에 대한 질의가 가능하도록 하였고 SQL과 자연스럽게 결합하여 XML 변경이 가능하도록 한 확장된 SQL을 제안하였다. 그리고 ORDBMS 저장 환경에서 색인 정보를 자동 생성 활용함으로써 질의 처리를 위한 SQL 문자열 변환 과정에서 PATH의 깊이만큼 중첩되는 조인 문제를 해결하여 검색 성능을 향상시켰다.

본 논문에서 제안하는 확장된 SQL은 범용 프로그램 안에서 내포된 형태의 사용이 가능하고, 단순한 문법을 가지기 때문에 사용자 인터페이스나 개발 도구에 의한 프로그램적인 처리가 용이하며, 기존에 SQL에서 지원하지 않았거나 그 개념이 약했던 XML 문서 대상의 다양한 질의를 가능하게 함으로써 엘리먼트나 에트리뷰트 Path 표현 등을 주 검색 조건으로 하는 내용 기반 정보 검색 및 활용을 위한 응용들에 적합하다.

향후에는 다양한 XPATH 표현을 지원할 수 있도록 XML 표현력을 강화하고 다른 XML 질의 엔진들과 비교 분석 및 성능 평가를 할 계획이며 대용량 문서의 변경을 위한 효율적인 처리 방향에 대한 연구도 계속 진행되어야 할 것이다.

6. 참고 문헌

- [1] W3C, XML Path Language (XPath) Version 1.0. Recommendation 16 November 1999.
- [2] Serge Abiteboul, Dallon Quass, Jason McHugh, Jennifer Widom, and Janet L. Wiener, The Lorel Query Language for Semistructured Data. 1997.
- [3] Alin Deutsch, Mary Fernandez, Daniela Florescu, Alon Levy, and Dan Suciu, XML-QL: A Query Language for XML. Submission to the World Wide Web Consortium 19-August-1998.
- [4] Eduard Derksen, Peter Fankhauser, Ed Howland, Gerald Huck, Ingo Macherius, Makoto Murata, Michael Resnick, and Harald Schoning, XQL (XML Query Language). August 1999
- [5] Don Chamberlin, Jonathan Robie, and Daniela Florescu, Quilt: an XML Query Language. 31 March 2000.
- [6] Scott Boag, Don Chamberlin, Mary F. Fernandez, Daniela Florescu, Jonathan Robie, Jerome Simeon, and Mugur Stefanescu, XQuery 1.0: An XML Query Language. W3C Working Draft 30 April 2002.
- [7] Jayavel Shanmugasundaram, Jerry Kiernan, Eugene Shekita, Catalina Fan, and John Funderburk, Querying XML Views of Relational Data. 2001.
- [8] J.Shanmugasundaram, K.Tufte, G.He, C.Zhang, D.De Witt, & J.Naughton, "Relational Databases for Querying XML Documents: Limitations and Opportunities", Proc. of 25th VLDB, pp. 302-314, Edinburg, Scotland, September 1999. (<http://www.informatik.uni-trier.de/~ley/db/conf/vldb/ShanmugasundaramGTZDN99.html>)
- [9] Igor Tatarinov, Zachary G. Ives, Alon Y. Halevy, Daniel S. Weld, Updating XML. 2001.
- [10] 김경원 외 4 명, 관계형 데이터베이스를 이용한 XQL질의 처리시스템의 설계 및 구현, 2002 봄 학술발표논문집(B)
- [11] 조영주, XML 질의를 위한 확장된 SQL, 충남대학교 석사 학위 논문, 2002.8