

# XQuery2SQL 변환기 위한 알고리즘 구현

## Algorithm Embodiment for XQuery2SQL Converter

서현호, 김영국, 김덕만\*  
인하대학교\*

Seo hyun-ho, Kim young-kook, Kim duk-mann\*  
Inha University\*

### 요약

웹 기술의 급속한 발전으로 인한 인터넷의 사용과 정보의 양이 급증하는 요즘 표현중심적인 언어인 HTML에서는 웹의 정보를 이용하는 데 한계를 가져왔으며 이로 인한 대안으로 웹상에서 자유로운 문서 전송 및 교환을 위한 표준이며 W3C에서 데이터 자체의 의미나 상관관계를 표현하는 XML이 등장하였다. 이러한 XML문서를 RDBMS에서 저장해서 사용하기 위한 많은 노력이 있으나 구조적으로 XML문서는 트리구조이어서 관계형 DB에 자료를 질의하기 위한 언어인 SQL과 완벽한 호환을 이루지 못한다. 그래서 W3C의 XML 표준 질의인 XQuery가 등장하게 되었다.

이 논문에서는 XML 문서를 파싱하고 DOM 트리과정을 거쳐 RDBMS에 저장된 XML 정보들을 XQuery2SQL이라는 변환기를 통해서 SQL 질의로 변환한 후 RDBMS에 있는 정보를 추출하는 XQuery2SQL 변환 알고리즘을 구현하고자 한다.

### Abstract

HTML that is language that web technology is center expression these day that use of internet and quantity of information by fast development increase rapidly brought limit to use information of web and XML that express meaning or corelation of data itself in W3C by standard for free document transmission and exchange in World Wide Web by the alternative as long as is deviation appeared. There is many efforts to use storing this XML document in RDBMS but to relation style DB because XML document is tree structure structurally data SQL and perfect disaster caused by things that is language to ask a question accomplish.

In this paper XML document XML informations that is stored to RDBMS via Parsing and DOM tree process SQL quality through converter called XQuery2SQL of by change and embody XQuery2SQL conversion algorithm that draw information in RDBMS.

### I. 서론

XML은 WWW과 관련된 기술을 주도하는 단체인 W3C에서 확장 가능한 마크업언어를 설계하기 시작해서 SGML의 유동성, HTML의 폭넓은 대중성을

포함하여 설계되었다.

XML은 프로그래밍분야, 데이터분야, 프록시분야, 메타데이터분야, B2B등의 분야 등에서 폭 넓게 쓰이고 있다. 이중 데이터로서 XML인 XML 질의에 대해 알아보하고자한다.

XQuery는 XML 형태로 SQL의 기능을 제공하는 것이 목표로 XML을 DB에서 이용하기 위한 과정에

1) 본 연구는 대학 IT연구센터 육성·지원 업의 연구 결과로 수행되었음

XML의 저장 및 검색과 같은 새로운 형태의 데이터 관리기술이 필요하게 되었다 XML 문서의 저장소로는 관계형 데이터베이스, 객체지향형 데이터베이스, XML문서의 저장을 위한 전용시스템[1], 텍스트파일 시스템 등을 이용할 수 있으나 이 논문에서는 XML 문서가 보편화된 관계형 데이터베이스 시스템을 사용하였다.

RDBMS에 저장된 XML[2]정보를 검색하기 위해서는 SQL문이 필요하므로 XQuery To SQL과정을 통해서 SQL문을 생성해서 SQL 질의 결과를 얻고자 한다. 이 과정에 필요한 XQuery는 XPath와 FLWR로 구현된다.

이 논문의 구성은 서론에서는 XQuery의 구성 성분인 XPath와 FLWR에 대해 알아보고 본문에서는 XML관리시스템 구조와 XQuery To SQL로 변화하는 과정과 알고리즘에 대해 설명한다. 마지막으로 결론에서는 SQL의 결과로 형성된 XML을 이용할 수 있는 분야에 대해서 알아보자.

## 1. 관련연구

현재까지 제안된 XML질의어로는 다음의 LOREL[3], XML-QL[4], XQL[5], Quilt[6], XQuery[7] 등이 있으며 대부분은 풍부한 표현력을 지원 하는 대신 기존 SQL 유저에게는 사용하기불편하며 복잡하다.

## 2. XQuery

XQuery[8]를 구현하기 위한 대표적인 두 가지 표현은 XPath와 FLWR로 구현한다.

## 3. XPath

XQuery에서 경로 표현은 XPath표현과 비슷하나 주목할만한 차이는 XQuery 경로 표현은 Xpath의 축약형 구문을 사용한다. 다음에서 간단하게 XPath의 예를 알아보자.

3-1) 문서(artists.xml)에서 1997년에 발표된 모든 앨범을 가져오라.

```
document("artists.xml")/artists/artist/album
[year="1997"]
```

▶▶ 그림 1. artists.xml에 대한 XPath

```
<album live="no">
  <name>Live the Air</name>
  <year>1997</year>
</album>
```

▶▶ 그림 2. artists.xml에 대한 XPath결과

3-2) 1970년 전에 발표된 모든 앨범과 싱글은 가져오라.

```
document("artists.xml")/artists/artist/
*[year<1970]
```

▶▶ 그림 3. artists.xml에 대한 XPath

```
<single>
  <name>Ring of Fire</name>
  <year>1963</year>
</single>
```

▶▶ 그림 4. artists.xml에 대한 XPath결과

위에 대한 경로표현을 요약하면 다음과 같다 첫째, document()는 문자열 인수로 식별된 문서의 루트 노드를 반환한다.

둘째, 슬래시(/)는 자체로 질의를 실행할 환경에 의해 결정된 절대적인 루트 노드를 나타낸다.

셋째, 애스터리스크(\*)는 와일드 카드로 사용할 수 있다.

## 4. FLWR

FLWR은 노드의 컬렉션을 반복하고 반환을 수행하고, 임의 결과를 구성하고, 조건적인 프로세싱을 처리하는 능력을 제공함으로써 경로 표현을 보완한다. SQL 프로그래머가 쉽게 읽을 수 있고 익숙할 수 있도록 SQL 같은 구문으로 설계 하였다.

4-1) F-For

경로표현에 의해 반환된 노드를 변수에 순차적 리

스트로 반환한다. SQL문에서 From으로 표현되며 테이블 이름을 반환한다.

4-2) L-Let

경로표현에 의해 반환된 ordered forest 노드집합을 표현한다. SQL문에서 Select, From, Where, Order by 으로 노드집합을 표현한다.

4-3) W-Where

경로표현에 의해 반환된 노드의 조건을 검사한다. SQL문에서 Where문과 같다.

4-4) R-Return

경로표현에 의해 반환된 노드의 결과값의 표현으로 SQL문에서 Select문과 같은 기능을 한다.

다음에서 노드 컬렉션을 반복하고 반환을 수행하고, 임의 결과를 구성하는 FWLR와 조건적인 프로세싱을 처리하는 FLWR의 예를 보면 다음과 같다.

4-1-1) 각각의 예술가를 musician 요소로 변환하고, 각 예술가의 이름과 일치하는 텍스트를 포함하는 새로운 well-formed XML문서를 생성하라.

```
<musicians>
  FOR $a IN /artists/artist
  FOR $b IN /artists/artist/year
  WHERE $a/name="Chicago"
  and $b/@year > 1970
  RETURN
  <musician>$a/name/text()</musician>
</musicians>
```

▶▶ 그림 5. artists.xml에 대한 XQuery

```
<musicians>
<musician>Chicago</musician>
</musicians>
```

▶▶ 그림 6. artists.xml에 대한 XQuery결과

4-1-2) XQuery는 조건적인 프로세싱을 지원한다. 한 예로 각앨범에서 앨범이 지휘자를 가지고 있는지에 따라 다른 이름을 가진 요소들을 반환하라

```
FOR $a IN /artists/artist/album
IF $a/conductor
THEN
<classical>$a/name/text()</classical>
ELSE
<noClassical>$a/name/text()</noClassical>
```

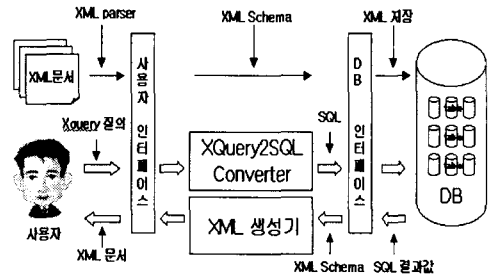
▶▶ 그림 7. artists.xml에 대한 XQuery

```
<classical>The 9th Symphony</classical>
<noClassical>Live the Air</noClassical>
```

▶▶ 그림 8. artists.xml에 대한 XQuery결과

## II. 본 문

### 1. XML 관리 시스템 구조



▶▶ 그림 9. XML-RDBMS 구조

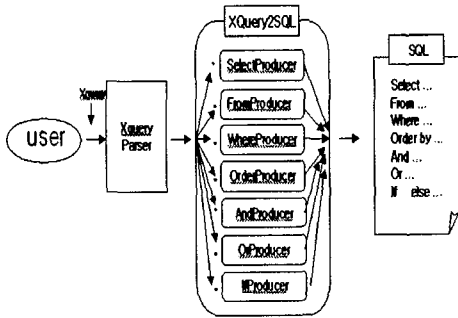
XML Parser에 의해서 파싱된 XML문서는 메모리상에서 DOM tree 형태로 존재하며 XML Schema 참조를 통해서 RDBMS시스템에 저장된다.

사용자가 이 DB에 저장된 자료를 이용하기 위해서 XML표준 질의인 XQuery를 관계형 DB의 표준 질의인 SQL로 변화하는 과정이 필수적이다 이에 따라 XQuery2SQL Converter를 구현해야한다 이 변환기를 통해서 사용자는 RDBMS에 저장된 데이터를 가져와 사용할 수 있다. DB에서 가져온 결과값은 후에

XML생성기라는 모듈과 XML Schema를 통해서 사용자에게 XML 형태로 제공되어져서 여러 응용 분야에 폭넓게 이용할 수 있다.

## 2. XQuery2SQL Converter

사용자에 의한 XQuery를 입력하면 XQuery Parser가 구문과 데이터를 각각 분석하여 생성기 SelectProducer, FromProducer 등등에게 보내어지며 각각의 생성기는 그 구문과 데이터에 따라 각각의 SQL구문을 만든다 그래서 아래와 같이 완성된 SQL문이 생성된다.



▶▶ 그림 10. XQuery2SQL 과정

## 3. XQuery2SQL 과정 알고리즘

XQuery 파일을 직접 입력한 후 동적 메모리 할당을 하고 XQuery 파싱과정을 통해서 얻은 변수명, 함수명, 연산자... 등등의 정보를 구조체 객체의 내부에 저장하고 구조체마다 다음 구조체를 가르키는 포인터를 두어서 메모리를 효율적으로 관리한다. 이러한 XQuery2SQL 모듈의 질의 변환 과정은 getOperation() 이용하여 원하는 연산자(for, where...)들을 선택한 후 Trans()이 이용하여 선택한 연산자(한예로for) 대한 연산(trans\_for(for1), trans\_for(for2))등등을 개별적으로 수행한 후 SearchString(for1) SearchString(for2) 등등을 통한 검색결과로 각각의 Producer (SelectProducer, WhereProducer..) 등을 생성한다. 이 과정은 While(fopen(in\_file))이 존재하는 한

getOperator()->Trans()->SearchString() 과정에 의한 반복이 일어나며 파일이 존재 하지 않으면 XQuery2SQL 모듈과정은 정지한다.

위와 같은 XQuery파싱 과정을 통한 각각의 Producer의 생성알고리즘은 다음과 같다.

### 3-1) SelectProducer 알고리즘

Select문은 XQuery에서 return연산아래 for문이나 let문에서 지정한 '\$'다음의 변수속성값을 반환한다. 이 SelectProducer에서 파일 포인터는 먼저 return연산자를 찾는다.

그후 return연산자가 발견된 라인에서 for문이나 let문에서 지정한 '\$'다음의 변수값이 있으면 그에 대한 노드를 반환하고 return연산자가 발견된 지 않은 라인에서 for문이나 let문에서 지정한 '\$'다음의 변수값이 있으면 그에 대한 노드를 반환한다. 또 '\$'변수/변수2 에서 '/'기호다음의 변수2는 변수의 칼럼값이므로 '.'기호와 함께 변수에 접근할 수 있다.

또한 return문 아래 '\$'변수/속성에서 속성 개수가 릴레이션의 필드수와 일치하면 '\*'를 반환한다.

'(와 )'를 기준으로 각 변수의 속성 정보를 추출한다. 이와 같은 과정을 통해서 SQL문중 Select문은 형성된다.

### 3-2) FromProducer 알고리즘

From문은 XQuery에서 for연산자의 의해서 생성되는 것으로 마지막 요소의 '/', '//기호다음에 임의의 값은 데이터베이스에서 테이블로 정의하고 '\$'기호 다음의 변수에 그 값을 입력한다.

### 3-3) WhereProducer 알고리즘

Where문은 XQuery에서 where연산자에 의해서 생성되며 for, let연산자에 의해서 입력된 '\$'다음의 변수와 '/'기호 다음의 변수에 의해서 where문을 형성한다. 단 '\$'다음에 변수가 let연산자에 의해서 지정된 변수이면 Select-From-Where-Orderby 형식의 집합노드를 반환한다.

3-4) OrderProducer 알고리즘

order by문은 XQuery에서 order연산자에 의해서 생성되며 '\$'다음에 지정한 변수에 for, let연산자에 의해서 노드를 반환한다.

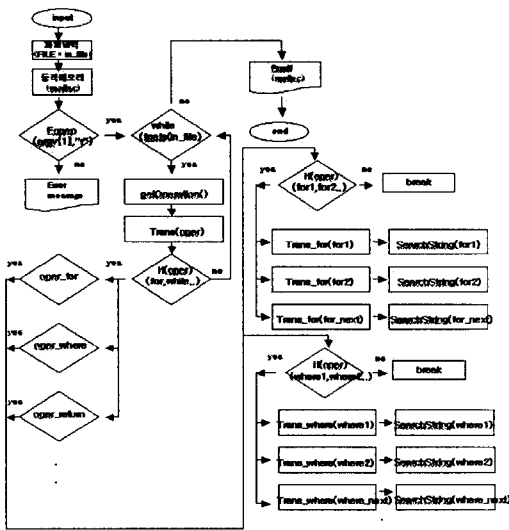
3-5) Let 알고리즘

let문은 XQuery에서 let연산자에 의해서 생성되며 \$기호다음의 변수에 마지막요소의 '/', '//'기호 다음의 노드를 Select-from-where등의 노드집합형식으로 반환하며 중첩질의인 경우 IN연산자와 함께 이용된다.

3-6) And, Or, IF Producer 알고리즘

and, or, if else 연산자들도 for, let 연산자에 의해서 반환된 '\$'다음에 변수에 노드를 반환하고 각각의 연산자 and, or, if else을 생성한다. 이와 같은 각각의 Producer연산과정을 통해서 SQL문을 완성할 수 있다.

4. XQuery2SQL 구현 알고리즘



▶▶ 그림 11. XQuery2SQL 알고리즘순서도

5. XQuery2SQL 변환

관계대수(Relational Algebra)는 릴레이션 조작을 위한 연산의 집합에 따라 첫째 일반집합 연산자인 합집합, 교집합, 차집합, 카디셔널 프로덕트와 둘째 기본 SQL문과 유사한 순수 관계 연산자인Selecting, Projecting, Join, Division 등이 있다 이 논문에서는 순수 관계연산자중 Selecting, Projecting는 단순질의와 중첩질의로, Join에서는 내부조인 으로 XQuery 2SQL문을 구현해 본다.

5-1) Selecting

5-1-1) 단순질의

XQuery문에서 Selecting 질의란 관계형 테이블에서 릴레이션 R에서 어떤 선택조건을 만족하는 튜플 들을 선택한 것으로 한 예로 인하대학의 학생중 김영국이라는 학생의 정보를 출력하시오

```

For $a in document("inha.xml")//인하
Where $a/이름='김영국'
Return
<num>{$a/학번}</num>
<name>{$a/이름}</name>
<year>{$a/출생년도}</year>
<subject>{$a/학과}</subject>
<professor>{$a/지도교수}</professor>
    
```

▶▶ 그림 12. 5-1-1에 대한 XQuery

[표 1] 5-1-1에 대한 XQuery2SQL표

From	//인하	인하
Select	\$a/학번, \$a/이름, \$a/출생년도, \$a/학과, \$a/지도교수	인하.학번, 인하.이름, 인하.출생년도, 인하.학과, 인하.지도교수
Where	\$a/이름='김영국'	인하.이름=김영국
Order by	NULL	NULL

```

SELECT *
FROM   인하
WHERE  (인하.이름 = '김영국')
    
```

▶▶ 그림 13. 5-1-1에 대한 SQL

5-1-2) 중첩질의

```

For $a in document("inha.xml")//이하
Let $b:=document("inha.xml")//개인신상표
Where $a/이름=$b/이름
Return
<num>{$a/학번}</num>
<name>{$a/이름}</name>
<year>{$a/출생년도}</year>
<subject>{$a/학과}</subject>
<professor>{$a/지도교수}</professor>
    
```

▶▶ 그림 14. 5-1-2에 대한 XQuery

[표 2] 5-1-2에 대한 XQuery2SQL표

From	//이하	이하
Let	//개인신상표	Select * from 개인신상표
Select	\$a/학번,\$a/이름,\$a/출생년도,\$a/학과,\$a/지도교수	이하.학번,이하.이름,이하.출생년도,이하.학과,이하.지도교수
Where	\$a/이름=\$b/이름	이하.이름 IN (SELECT 개인신상표.이름 FROM 개인신상표 )
Order by	NULL	NULL

```

SELECT *
FROM   이하
WHERE  이하.이름 IN
      (SELECT 개인신상표.이름
       FROM 개인신상표 )
    
```

▶▶ 그림 15. 5-1-2에 대한 SQL

5-2) Projecting

5-2-1) 단순질의

XQuery문에서 Selecting 질의란 관계형 테이블에서 릴레이션 R에서 어떤 선택조건을 만족하는 필드값만을 선택한 것으로 한예로 인하대학의 학생중 김영국이라는 학생의 학번을 출력해보자.

```

For $a in document("inha.xml")//이하
Where $a/이름='김영국'
Return
<num>{$a/학번}</num>
    
```

▶▶ 그림 16. 5-2-1에 대한 XQuery

[표 3] 5-2-1에 대한 XQuery2SQL표

From	//이하	이하
Select	\$a/학번	이하.학번
Where	\$a/이름='김영국'	이하.이름='김영국'
Order by	NULL	NULL

```

SELECT   이하.학번
FROM     이하
WHERE    (이하.이름 = '김영국')
    
```

▶▶ 그림 17. 5-2-1에 대한 SQL

5-2-2) 중첩질의

```

For $a in document("inha.xml")//이하
Let $b:=document("inha.xml")//개인신상표
Where $a/이름=$b/이름
Return
<num>{$a/학번}</num>
    
```

▶▶ 그림 18. 5-2-2에 대한 XQuery

[표 4] 5-2-2에 대한 XQuery2SQL표

From	//이하	이하
Let	//개인신상표	Select * from개인신상표
Select	\$a/학번	이하.학번
Where	\$a/이름=\$b/이름	이하.이름 IN (SELECT 개인신상표.이름 FROM 개인신상표 )
Order by	NULL	NULL

```
SELECT 인하.학번
FROM 인하
WHERE 인하.이름 IN
      (SELECT 개인신상표.이름
       FROM 개인신상표 )
```

▶▶ 그림 19. 5-2-2에 대한 SQL

5-3) Join

5-3-1) 내부조인

XQuery문에서 Join(내부조인)란 서로 다른 테이블에서 공통적인 필드로 질의해서 원하는 값을 얻을 수 있는 것으로 한 예로 인하대학의 학생중 학번이 920324 인 김영국이라는 학생의 학과를 출력해보자

```
For $a in document("inha.xml")//인하
For $b in document("inha.xml")//개인신상표
Where $a/이름='김영국'
and $b/학번=920324
Return
<num>{$a/학과}</num>
```

▶▶ 그림 20. 5-3-1에 대한 XQuery

[표 5] 5-3-1에 대한 XQuery2SQL표

From	//인하//개인신상표	인하,개인신상표
Select	\$a/학과	인하.학과
Where	\$a/이름='김영국' and \$b/학번=920324	인하.이름='김영국' and 개인신상표.학번=920324
Order by	NULL	NULL

```
SELECT 인하.학과
FROM 인하
WHERE (인하.이름 = '김영국' and
      개인신상표.학번=920324)
```

▶▶ 그림 21. 5-3-1에 대한 SQL

III. 결론

XQuery2SQL 변환기를 통한 SQL로 RDBMS에 질의를 하여 얻은 결과값을 SQL 2000 Server의 XML 지원기능을 이용하여 XML 형태로 형성할 수 있다 또한 후에 XML 관리시스템에서 XML생성기를 구현하여 사용자가 원하는 결과 형식의 XML문서를 만들 수 있어 이 XML문서를 PDF, WML, LBS 환경하에 지리정보 검색 등에도 사용할 수 있다.

■ 참고문헌 ■

- [1] eXcelon Corporation, eXtensible information Server, <http://www.exln.com>
- [2] W3C, XML Specification("XMLspec"), 1998.
- [3] Serge Abiteboul, Dallon Quass, Jason McHugh, Jennifer Widom, and Janet L. Wiener, The Lorel Query Language for Semistructured Data, 1997.
- [4] Alin Deutsch, Mary Fernandez ,Daniela Florescu, Alon Levy, and Dan Suciu, XML-QL:A Query Language forXML.Submission to the World Wide Web Consortium 19-August-1998.
- [5] Eduard Derksen ,Peter Fankhauser, Ed Howland Gerald Huck, Ingo Macherius, Makoto Murata Michael Resnick, and Harald Schoning, XQL(XML Query Language) August 1999.
- [6] Don Chamberlin, Jonathan Robie, and Daniela Florescu, Quilt:an XML Query Language 31, March, 2000.
- [7] ScottBoat, DonChamberlin, Mary F.Fernandez DanielaFlorescu, Jonathan Robie,JeromeSimeon and Mugur Stefanescu,XQuery 1.0:An XML Query Language. W3C Working Draft 3.0 April 2002.
- [8] W3C, XQuery 1.0: AnXML Query Language, 2002.