

# 래퍼 상에서의 XML 실체뷰를 이용한 XQL 질의 처리\*

문찬호<sup>0</sup> 강현철  
 중앙대학교 컴퓨터공학과  
 {moonch, hckang}@rose.cse.cau.ac.kr

## XQL Query Processing Using XML Materialized Views in a Wrapper

ChanHo Moon<sup>0</sup> Hunchul Kang  
 Dept. of Computer Science and Engineering, Chung-Ang University

### 요약

웹 문서 표준으로 제안된 XML의 등장으로 XML 문서들은 점차 웹 데이터에서 많은 비중을 차지하고 있으며 웹 상에서의 통합 서비스를 지원하는 미디어터/래퍼 시스템에서 중요한 웹 자원으로 활용될 수 있다. 다수의 XML 문서를 저장하고 있는 XML 저장소 내에 XML 실체뷰가 있다고 가정할 때, 미디어터/래퍼 시스템은 XML 문서 검색의 성능 향상을 위해서 이들 실체뷰를 이용하여 질의를 처리할 수 있다. 본 논문에서는, 기존의 실체뷰를 이용한 질의 처리 유형 세 가지 중에서 질의 결과 일부를 실체뷰로부터 얻고 나머지 결과를 하부 XML 문서들로부터 얻는 유형에 대하여 연구한다. 즉, 주어진 XQL 질의와 관련 실체뷰에 대하여, 실체뷰에 대한 XQL 질의와 하부 데이터에 대한 XQL 질의를 생성하는 질의 변환 알고리즘을 제시한다.

### 1. 서론

미디어터/래퍼(mediator/wrapper) 기반의 정보검색 기술은 분산성(distribution), 자치성(autonomy), 이질성(heterogeneity)을 갖는 웹 자원의 통합 서비스를 지원할 수 있는 데이터베이스 기술이다[1]. 차세대 웹 문서를 위한 표준으로 XML(eXtensible Markup Language)이 등장함에 따라 향후 웹 상에는 수많은 XML 문서가 존재할 것이며, 이들은 미디어터/래퍼 시스템이 서비스하는 가장 중요한 웹 자원이 될 것이다.

그림 1은 다수의 XML 문서를 저장하고 있는 XML 저장소를 대상으로 한 미디어터/래퍼 시스템의 질의 처리 과정을 나타낸 것이다. 그림 1에서 보는 것처럼 XML 저장소에 어떤 DTD를 참조하는 여러 XML 문서들과 이들로부터 도출된 XML 실체뷰(materialized view)[2]들이 있다고 하자. 미디어터에서 래퍼로 전송된 질의 q가 주어졌을 때 XML 저장소에 대한 질의 결과를 관련된 실체뷰로부터 얻을 수 있다면, 래퍼 상에서의 질의 변환을 통해 q를 실체뷰에 대한 질의 q'으로 변환하여 수행하면 방대한 양의 XML 저장소 내 문서들에 대한 검색을 피할 수 있어 질의 응답 시간을 줄일 수 있다[3].

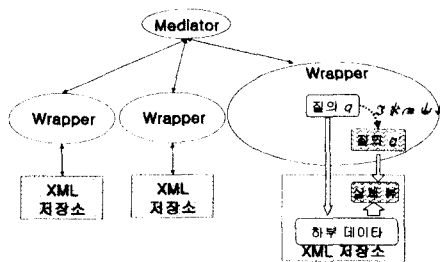


그림 1 래퍼 상에서의 XML 실체뷰를 이용한 XQL 질의 처리

본 논문에서는 XML 저장소 내에 XML 실체뷰가 있다고 가정할 때, XML 문서에 대한 질의 결과의 내용 중 일부가 실체뷰에 존재할 경우,

\*본 연구는 한국과학재단 목적기초연구(2000-1-303-001-3) 지원으로 수행되었음.

주어진 질의를 (1) 실체뷰에 대한 질의와 (2) 실체뷰에 포함되지 않은 내용을 직접 하부 XML 데이터로부터 검색하기 위한 질의로 변환하는 질의 변환 알고리즘을 제시한다. 본 논문에서는 XML 질의어로 경로 표현식을 이용하여 질의 대상을 표현하는 XQL(XML Query Language)[4]을 사용하였다.

본 논문의 구성은 다음과 같다. 2절에서는 실체뷰를 이용한 질의 처리의 유형 분류 및 유형 결정에 대해 기술한다. 3절에서는 기존의 실체뷰를 이용한 질의 처리 유형 중 실체뷰와 하부 데이터를 모두 이용하는 다섯 가지 경우와 그 예를 기술한다. 4절에서는 XQL 질의의 변환 알고리즘을 제시한다. 마지막으로 5절에서 결론을 맺는다.

### 2. 실체뷰를 이용한 질의 처리의 유형 분류 및 결정

그림 1에서와 같이 래퍼상에서의 질의 처리 시 실체뷰를 이용하기 위해서는 주어진 질의 결과의 전부 또는 일부가 실체뷰로부터 도출 가능해야 한다. 질의 결과와 실체뷰 내용 간의 포함 관계는 그림 2와 같이 다섯 가지로 분류할 수 있다[5].

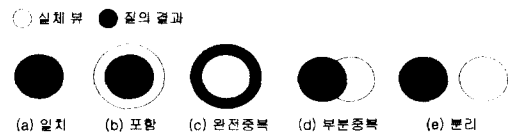


그림 2 실체뷰와 질의 결과 간의 포함 관계[5]

그림 2에서 (a)는 실체뷰와 질의 결과가 일치하는 경우를 나타낸 것이며, (b)는 실체뷰가 질의 결과를 포함하는 경우이다. (c)는 (b)와 반대로 실체뷰가 질의 결과에 완전히 포함됨을 나타내며, (d)는 실체뷰와 질의 결과가 일부만 겹침을 나타낸다. (e)의 경우에는 실체뷰와 질의 결과 사이에 아무런 포함 관계가 성립하지 않는다.

그림 2와 같은 실체뷰와 질의 결과 간의 포함 관계에 따라 질의 처리 시 실체뷰를 이용하는 방법은 그림 3과 같이 크게 세가지 유형으로 나누어 볼 수 있다[3].

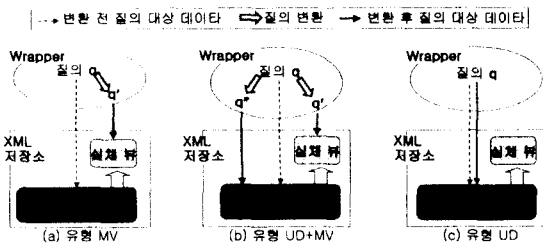


그림 3 Wrapper 상에서의 XML 실제뷰를 이용한 XQL 질의 처리

그림 3에서 (a)의 유형 MV(Materialized View Only)는 그림 2의 (a)일치나 (b)포함에 해당하는 경우의 질의 처리 방법으로 주어진 질의 q를 실제뷰에 맞는 질의 q'으로 변환하여 실제뷰로부터 원하는 결과를 모두 얻을 수 있음을 나타낸다. (b)의 유형 UD+MV(Both Underlying Documents and Materialized View)는 그림 2의 (c)완전 중복이나 (d)부분 중복에 해당하는 경우의 질의 처리 방법으로 질의 결과의 일부는 실제뷰에 존재하지만 일부는 실제뷰에 존재하지 않는 경우이다. 따라서 유형 UD+MV는 주어진 질의 q를 실제뷰에 대한 질의 q'과 하부 데이터에 대한 질의 q''으로 변환하여 각각으로부터 결과를 얻는다. (c)의 유형 UD(Underlying Documents Only)는 주어진 질의 결과를 모두 하부 데이터로부터 얻는 경우이다. 그림 2의 (e)분리의 경우 이러한 질의 처리 유형을 사용하여야 하며, (b)포함의 경우에도 질의 처리 비용을 고려하여 유형 UD와 같은 질의 처리가 가능하다.

XQL 질의 q와 실제뷰 v가 주어졌을 때, v를 이용한 q의 처리 가능 여부를 판별하고 그에 따라 해당 질의 처리 유형을 결정하는 것은, q와 v의 경로 간의 비교 그리고 q와 v의 조건 간의 비교를 통해 가능하다. [3]에서는 XML 실제뷰를 이용한 XQL 질의 처리의 유형을 결정하는 방법을 표 1과 같이 제시하였다.<sup>1</sup>

표 1 XML 실제뷰를 이용한 XQL 질의 처리의 유형 결정

조건이 명시되지 않은 경우	-	MV	MV		UD
조건이 명시된 경우	v.con = q.con	MV	MV		UD
	v.con ⊂ q.con	MV	MV		UD
	v.con ⊃ q.con			UD	UD
	otherwise	UD	UD	UD	UD

3. 유형 UD+MV의 다섯 경우와 그 예

[3]에서는 그림 3 (a)의 유형 MV에 대하여 주어진 질의 q와 관련 실제뷰 v에 대해 q와 동일한 결과를 얻는 v에 대한 질의 q'을 생성하는 질의 변

<sup>1</sup> 질의의 대상이 되는 엘리먼트(element)를 목적 엘리먼트라 할 때, q.path(v.path)는 q(v)가 명시한 질의의 목적 엘리먼트까지의 경로를 나타내고 이를 q(v)의 목적 엘리먼트 경로라고 부른다. 연산 <sub>v</sub>은 q와 v와의 경로 비교를 위해 사용된다: q가 v의 prefix와 일치할 경우 q <sub>v</sub>가 성립한다. q.con(v.con)은 목적 엘리먼트 또는 그 자손 엘리먼트에 대한 조건을 나타내고 이를 q(v)의 목적 엘리먼트 조건이라고 부른다. q.con(v.con)은 q(v)에서 목적 엘리먼트 또는 그 자손 엘리먼트에 대한 조건이 필터 연산자를 통해 conjunctive normal form으로 [p<sub>1</sub>, \$and\$ p<sub>2</sub> \$and\$ ... p<sub>n</sub>]과 같이 주어졌을 때 이를 집합 {P<sub>1</sub>, P<sub>2</sub>, ..., P<sub>n</sub>}으로 표현한 것으로, P<sub>i</sub>는 p<sub>i</sub>에서 조건이 주어진 대상 엘리먼트를 해당 문서 내의 완전 경로(full path)로 바꾸어 표현한 것이다(i = 1, ..., n). 자세한 내용은 [3]을 참조.

환 알고리즘을 제시하였다. 본 논문에서는 [3]의 질의 변환 알고리즘을 확장하여 그림 3의 질의 처리 유형 세가지 모두에 대해 질의를 변환하는 알고리즘을 제시한다. 특히, 그림 3 (b)의 유형 UD+MV에 대하여 주어진 질의 q와 관련 실제뷰 v에 대한 질의 q'과 하부 데이터에 대한 질의 q''을 생성하는 알고리즘을 제시한다. 이를 위해 표 1에서 나타난 바와 같이 유형 UD+MV를 경로와 조건의 구분에 따라 다섯 가지의 경우로 나누었다. 이는 질의 변환 과정 시 다섯 가지 경우가 서로 다른 형태의 질의 변환 절차를 필요로 하기 때문이다. 표 2는 유형 UD+MV의 다섯 가지 경우에 해당되는 예를 XQL 질의 q와 실제뷰 v로 구분하여 나타낸 것이다.

표 2 유형 UD+MV의 다섯 가지 경우의 예

유형	질의 q	실제뷰 v
UD+MV case-1	bookstore/book	bookstore/book/author
UD+MV case-2	bookstore/book[author/first-name = "Michael"]	bookstore/book/author[first-name = "Michael"]
UD+MV case-3	bookstore/book[author/first-name = "Michael" \$and\$ author/last-name = "Kay"]	bookstore/book/author[first-name = "Michael"]
UD+MV case-4	bookstore/book/author[first-name = "Michael"]	bookstore/book/author[first-name = "Michael" \$and\$ last-name = "Kay"]
UD+MV case-5	bookstore/book/author[first-name = "Michael"]	bookstore/book/author[first-name = "Michael" \$and\$ author/last-name = "Kay"]

그림 4는 유형 UD+MV의 다섯 가지 경우 중 case-3의 예를 그림으로 나타낸 것이다. bookstore/book/author[first-name = "Michael"]로 정의된 실제뷰 v가 있다고 할 때, bookstore/book[author/first-name = "Michael" \$and\$ author/last-name = "Kay"]이라는 질의 q가 제거된 경우로서 v.con ⊂ q.con과 q.path <sub>v</sub> v.path가 만족된다. 즉, q의 결과 중 일부가 v의 일부 내용과 일치하므로 그림 2의 (d) 부분 중복의 경우에 해당되어 그림 3의 질의 처리 유형 UD+MV로 처리 가능하다. 그림 4에서는 XML 문서를 트리 구조로 표현하여 노드는 엘리먼트, 그리고 괄호 안의 내용은 값(content)을 나타낸다. v는 bookstore 엘리먼트의 자식인 book 엘리먼트의 자식 중 author 엘리먼트를 모두 검색하되 자식인 first-name 엘리먼트 값이 "Michael"인 것만 검색하는 것으로 그 결과에 해당되는 서브 트리를 그림 4에 빗금 영역으로 표시하였다. 한편, q는 bookstore 엘리먼트의 자식 엘리먼트 중 book 엘리먼트를 모두 검색하되 자식인 author 엘리먼트의 자식인 first-name 엘리먼트 값이 "Michael"이고 자식인 last-name 엘리먼트 값이 "Kay"인 것만 검색하는 것으로 그 결과에 해당되는 서브 트리를 그림 4에 음영 영역으로 표시하였다.

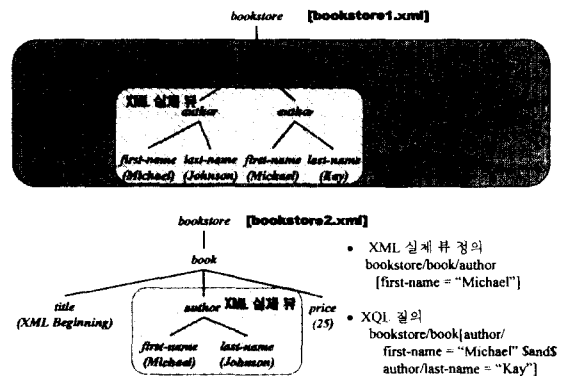


그림 4 유형 UD+MV case-3의 예

4. XQL 질의 변환 알고리즘

표 1에서 기술한 바와 같이 XQL 질의 q와 그에 연관된 XML 실체뷰 v가 주어지면 v를 이용한 q의 처리 유형을 결정할 수 있다. 본 절에서는 이들 질의 처리 유형(MV, UD+MV, UD)을 위한 질의 변환을 수행하는 알고리즘 rewrite\_XQL\_query()를 제안한다(그림 5). rewrite\_XQL\_query()는 [3]에서 제안한 유형 MV의 질의 변환 알고리즘을 확장하여 유형 UD+MV 및 유형 UD의 질의도 변환을 수행하도록 하였다.<sup>2</sup>

```

rewrite_XQL_query(v, q, q', q'') {
/* v XML 실체뷰 정의      q XQL 질의
q' 실체뷰 v에 대한 XQL 질의  q'' 하부 XML 문서에 대한 XQL 질의
*/

/* MV 유형 */
if ((v.path = q.path or v.path < q.path) and (v.con < q.con)) {
q' = "/" + v.elem + d_path(q, v.level) path + filter( and(q.con - v.con) )
q'' = NULL
/* +는 string concatenation */
}

/* UD+MV 및 UD 유형 */
...
}

q' = refine_query(q')
q'' = refine_query(q'')

return(q, q'')
}
    
```

그림 5 XQL 질의 변환 알고리즘

rewrite\_XQL\_query()는, 각각 질의와 실체뷰를 정의하는 XQL 표현식 q와 v를 입력받아 표 1에서 정리한 질의 처리 유형에 따라 질의 변환을 수행하여, v에 대한 XQL 질의 표현식 q'과 하부 데이터에 대한 XQL 질의 표현식 q''을 반환한다. 알고리즘에서 사용된 표기(notation)는 표 3과 같다[3].

표 3 표기

표기	설명
sub(con1, con2)	목적 엘리먼트 조건 con1에 con2의 내용이 포함될 때는 con1 - con2를 반환하고, 아닐 경우 NULL을 반환한다
and(con)	목적 엘리먼트 조건 con이 공집합이면 NULL을 반환하고, {P}이면 P를 반환하고, {P1, ..., Pn}일 경우 P1 Sand\$ ... Sand\$ Pn을 반환한다
filter(cnf)	conjunctive normal form으로 나타낸 조건 cnf가 NULL이면 NULL을 반환하고, 아닐 경우 [cnf]를 반환한다
q.path	XQL 질의 경로 표현식 q의 목적 엘리먼트 경로
q.con	XQL 질의 경로 표현식 q의 목적 엘리먼트 조건
q.level	XQL 질의 경로 표현식 q의 레벨
q.elem	XQL 질의 경로 표현식 q의 목적 엘리먼트
a_path(p, l)	경로 p를 레벨 값 l을 기준으로 분할 시 조상 경로
d_path(p, l)	경로 p를 레벨 값 l을 기준으로 분할 시 자손 경로

<sup>2</sup> [3]에서는 XQL 실체뷰의 내용을 XQL 질의 결과로 검색된 서브 트리들을 루트("/") 엘리먼트로 묶은 것으로 정의하고 있다.

rewrite\_XQL\_query()는 다음과 같이 작동한다. 먼저, DTD 테이블을 참조하여 q와 v로부터 q.path, q.con, v.path, v.con을 구한 후, 이들 간의 비교를 통해 표 1에서 정리한 질의 처리 유형에 부합되는 질의 변환을 수행하여 q'과 q''을 생성한다. 알고리즘 내용영역은 본 논문에서 제시하는 UD+MV 및 UD 질의 처리 유형을 위한 질의 변환을 수행하는 부분이다. 생성된 q'과 q''에 필터 연산자를 통한 조건이 명시되어 있을 경우, 조건의 대상 엘리먼트를 나타내는 완전 경로(이들은 q.con과 v.con으로부터 도출된 것임)를 q'과 q''의 목적 엘리먼트 경로 이후의 자손 경로로 변환하고, 제의 연산자('^')<sup>3</sup>와 경로 연산자('/')가 연속해서 명시되어 있을 경우(즉, '^('/') 제의 연산자('^')로 변환한다(refine\_query()).

표 2에서 예시한 유형 UD+MV의 다섯 경우에 대해 rewrite\_XQL\_query()를 적용하였을 경우 반환되는 q'과 q''은 표 4와 같다.

표 4 표 2의 유형 UD+MV의 다섯 경우 예에 대한 질의 변환 결과

유형	원래 질의	변환된 질의
UD+MV case-1	/author	bookstore/book\author
UD+MV case-2	/author	bookstore/book[author[first-name = "Michael"]\author[first-name = "Michael"]]
UD+MV case-3	/author[last-name = "Kay"]	bookstore/book[author/first-name = "Michael" Sand\$ author/last-name = "Kay"]\author[first-name = "Michael" Sand\$ last-name = "Kay"]]
UD+MV case-4	/author	bookstore/book/author[first-name = "Michael" Sand\$ Snot\$ (last-name = "Kay")]
UD+MV case-5	/book/author[first-name = "Michael"]	bookstore/book[Snot\$(author/first-name = "Michael" Sand\$ author/last-name = "Kay")]\author[first-name = "Michael"]]

5. 결론

본 논문은 XML 저장소 내에 XML 실체뷰가 있다고 가정할 때, 랩퍼 상에서 XML 실체뷰를 이용한 XQL 질의 변환에 대해 연구하였다. 이를 위해 기존의 실체뷰를 이용한 질의 처리 유형 세가지 중에서 질의 결과 일부를 실체뷰로부터 얻고 나머지 결과를 하부 XML 문서들로부터 얻는 유형 UD+MV에 대해 주어진 질의 q와 관련 실체뷰 v에 대하여 v에 대한 XQL 질의 표현식 q'과 하부 데이터에 대한 XQL 질의 표현식 q''을 생성하는 질의 변환 알고리즘 rewrite\_XQL\_query()를 제시하였다.

참고문헌

- [1] D. Florescu et al., "Database Techniques for the World-Wide Web : A Survey," SIGMOD Record, Vol.27, No.3, pp.59-74, 1998.
- [2] 임재국 외, "점진적 갱신에 기반을 둔 XML 정성류 관리 프레임워크," 정보처리학회논문지, 제 8-D권, 제 4호, 2001. 8.
- [3] 김수희, "XML 실체뷰를 이용한 XQL 질의 변환," 석사 학위 논문, 중앙대학교 컴퓨터공학과, 2001. 2.
- [4] J. Robie et al., "XML Query Language (XQL)," <http://www.w3.org/TandS/QL/QL98/pp/xql.html>, 1998.
- [5] D. Lee and W. Chu, "Conjunctive Point Predicate-based Semantic Caching for Web Databases," Extended Abstract, Tech. Rep. TR-980030, UCLA, Sep. 1998.

<sup>3</sup> 제의 연산자('^')는 XQL 질의 변환을 위해 본 논문에서 추가한 XQL 연산자(operator)로서 제의 연산자 이후의 엘리먼트를 포함하는 서브 트리는 검색에서 제외하는 것을 의미한다. 예를 들어, bookstore/book\author라는 질의는 bookstore 엘리먼트의 자식인 book 엘리먼트를 검색하되 그 자식인 author 엘리먼트는 검색 결과에서 제외함을 의미한다.