

# 관계형 Schema 기반의 XML 문서작성기 설계

정유나<sup>o</sup> 황인준

아주대학교 정보통신전문대학원 정보통신공학과  
{serazad, ehwang}@madang.ajou.ac.kr

## The design for XML document generation based relational schema

Yuna Jung Eenjun Hwang

The Graduate School of Information and Communication, Ajou University

### 요약

최근 XML이 인터넷을 기반으로 하는 문서의 표준으로 급속히 확산됨에 따라, 기존 관계형 데이터베이스에 저장되어 있는 데이터와 XML과의 상호 호환문제가 급격히 부상되었다. 문제 해결을 위해서 다각적인 연구가 진행되고 여러 방법론이 제시되었으나, 이제까지의 방법들은 특정 경우에는 제대로 처리하지 못하거나, 많은 부분을 사용자가 직접 생각하여 작성하게 하는등의 불편을 안고 있었다. 이러한 한계점을 개선하기 위해서, 본 논문에서는 사용자의 부담을 최소화하는 동시에, 정보의 손실없이 자동적으로 XML 문서작성이 이루어질 수 있는 방법을 제시하였다.

### 1. 서론

인터넷이 보급되면서 웹 문서의 표준으로서 HTML이 사용되었다. HTML은 단순하고, 일반인도 사용하기 편리하다는 장점으로 초기의 인터넷을 전세계적으로 확산시키는데 큰 역할을 하였다. 그러나, 점점 사용자의 요구가 다양하게 늘어나고 문서의 양이 폭발적으로 증가하면서, HTML은 표준으로서의 한계를 드러내게 되었다. 이에 따라, W3C에서 SGML의 확장성과 HTML의 단순성을 적절히 접목시킨 XML[1]을 내놓았고, 이는 급격히 확산되어 현재는 그 자리를 확고히 하고 있다. 이처럼 XML이 널리 확산되자 기존에 있는 데이터 형식과 XML과의 호환이 절대적으로 필요하게 되었다. 이러한 필요성으로 인해서, 여러방면에서 연구가 진행되었는데, HTML과 XML의 상호 호환이나 관계형 데이터와 XML사이의 상호 호환에 대한 연구등이 그것이다. 본 논문에서는 관계형 데이터베이스로부터 얻은 관계형 데이터를 관계형 Schema를 이용하여 XML 문서로 작성하는 방법을 연구해보았다. 이 방법은 RDBMS의 관계형 schema로부터 XML 데이터의 구조의 뼈대를 만들어내고, 관계형 데이터 결과를 구조에 맞게 삽입하여 XML 문서를 작성하는 것을 주요 골자로 한다.

본 논문의 구성은 다음과 같다. 먼저, 2절에서는 본 논문과 관련되어서 기존에 행해진 연구들을 살펴본다. 3절에서는 본 논문에서 제안하고 있는 방법의 전체적인 흐름과 각 요소의 연산을 제시하고 4절에서는 3절에서 제시한 방법을 이용한 몇가지 예를 보이고 5절에서 결론을 맺는다.

### 2. 관련연구

기존에 있는 관계형 데이터로부터 XML 문서를 생성하는 것에 대한 연구는 최근 몇 년간 꾸준히 진행되고 있다. 그 중에서 두 가지 방법을 소개하겠다.

#### 2.1 SilkRoute

99년에 제안된 SilkRoute[2]는 RXL 형태의 View Query와 XML-QL 형태의 User Query를 받아서 XML 문서의 Template과 SQL을 만들어 낸 후, SQL을 던져서 받은 관계형 데이터를 Template에 맞춰 넣어 XML 문서를 작성하는 중간처리기이다.

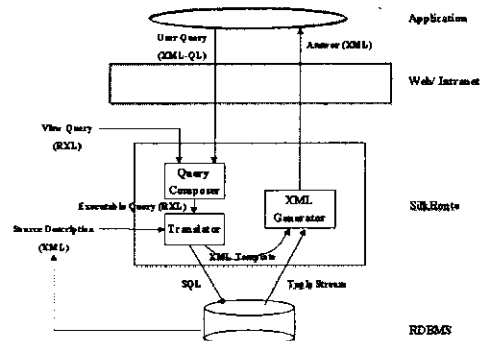


그림 1. SilkRoute의 구조

#### 2.2 Efficiently Publishing Relational Data as XML [3]

이 방법에서는 SQL에 XML Constructor을 덧붙여서, 질의를 수행하는 동시에 XML 문서가 생성되도록 하고 있다. 여기서 XML Constructor는 XML 문서의 fragment를 생성하도록 정의된다.

```

Select cust.name, CUST(cust.id,cust.name,
      (Select XMLAGG(ACCT(acct.id, acct.acctnum))
       From Account acct
       Where acct.custId=cust.id),
      (Select XMLAGG(PORDER(porder.id,porder.acct,porder.date,
      (Select XMLAGG(ITEM(item.id,item.desc))
       From Item item
       Where item.porderId=porder.id),
      (Select XMLAGG(PAYMENT(pay.id,pay.desc))
       From Payment pay
       Where pay.porderId=porder.id)))
       From PurchOrder porder
       Where porder.custId=cust.id))
From Customer cust
    
```

그림 2. 관계형 데이터로부터 XML 문서를 만들어내는 SQL 위의 두 방법은 사용자가 만들어내고 싶은 XML 문서의 구조를 생각해서 그 문서에 필요한 데이터를 추출하기 위한 질의를 던지는 형태이다. 그러나, 이는 질의를 던질 때마다 원하는 XML 문서의 구조와 관계형 데이터베이스의 Schema를 생각해서 질의해야 하는 번거로움이 따른다. 따라서, 본 논문에서는 응용 프로그램에 몇 가지 모듈을 추가하여 자동적으로 XML 문서를 작성하는 방법을 제안한다.

3. 관계형 Schema 기반의 XML 문서작성기의 설계

기존에 존재하던 응용 프로그램에 Schema Information과 XML Document Generator라는 두 개의 모듈을 추가하여 관계형 데이터와 같은 정보를 가지는 XML 문서를 작성한다. 이 방법은 아래의 두 가정을 기반으로 한다. 첫째, RDBMS상의 모든 릴레이션은 정규화 되어있다. 둘째, RDBMS로 던져지는 SQL 질의는 최적화 되어있다.

3.1 구조

본 논문이 제안하는 방법을 사용하는 응용 프로그램의 구조와 관계형 데이터베이스와의 자료흐름은 아래와 같다.

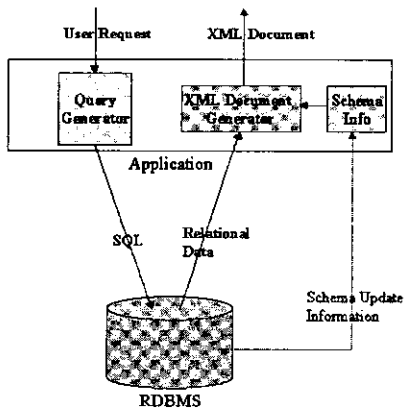


그림 3. 전체적인 구조와 자료흐름

기존의 응용프로그램에 추가되는 새로운 두 모듈을 살펴보자.

3.1.1 Schema Information

Schema Information 모듈은 XML 엘리먼트들간의 계층관계를 정의하는데 사용한다. 이 모듈이 가지고 있는 정보는 두가지이다. 하나는 서로 연관된 기본키와 외래키 정보이고, 또 하나는 같은 테이블내의 속성들간의 N:1 정보이다. 이 정보는 관계형 데이터베이스에서 관련정보가 수정되면 이에따라 갱신된다.

3.1.2 XML Document Generator

XML Document Generator 모듈은 Schema Information의 정보를 참조하여 질의결과로 받은 관계형 데이터와 같은 정보를 가지는 XML 문서를 작성한다.

4. 관계형 데이터로 XML 문서를 작성하는 예제

먼저, 관계형 데이터베이스가 아래와 같이 구성되어 있다고 가정하자.

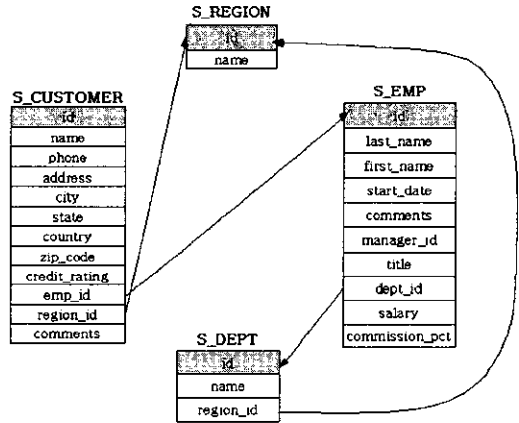


그림 4. 관계형 데이터베이스의 Schema

4.1 Schema Information

관계형 데이터베이스가 위와 같이 구성되어 있을 때, Schema Information 모듈이 가지는 정보는 아래와 같다.

기본키	외래키
S_EMP.id	S_CUSTOMER.emp_id
S_DEPT.id	S_EMP.dept_id
S_REGION.id	S_CUSTOMER.region_id
S_REGION.id	S_DEPT.region_id

그림 5. 기본키와 외래키관계

S\_CUSTOMER

id N-1 address N-1 city N-1 state N-1 county  
 id N-1 emp\_id  
 id N-1 region\_id  
 id N-1 zip\_code

S\_EMP

id N-1 last\_name, id N-1 first\_name, id N-1 start\_date,  
 id N-1 comments, id N-1 manager\_id, id N-1 title,  
 id N-1 dept\_id, id N-1 salary, id N-1 commission\_pct

S\_DEPT

id N-1 region\_id

그림 6. 한 테이블내의 속성간 다대일 정보

4.2 XML Document Generator

SQL 질의의 결과로 얻은 관계형 데이터를 아래의 순서에 따라 XML 문서로 작성한다.

STEP 1. 관계형 데이터가 추출된 릴레이션을 Schema Information의 기본기/외래키 정보에 따라 기본키를 가지는 릴레이션은 상위로하고, 외래키를 가지는 릴레이션은 하위로 해서 릴레이션간의 계층구조를 만든다.

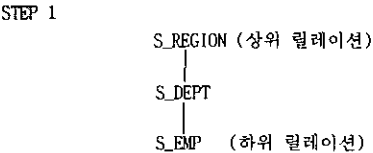
STEP 2. 먼저, 관계형 데이터의 각 컬럼을 분리한 뒤 재구성한다. 아래는 재구성 원칙이다. 재구성할 때, 먼저 위치되는 컬럼으로 정렬시킨후 그에 따라 다음 컬럼을 위치시킨다.

- 1) 계층구조상 가장 상위에 있는 릴레이션에 속하는 속성 컬럼부터 앞에 위치시켜 관계형 데이터를 재구성한다.
- 2) 같은 릴레이션에 속하는 컬럼이면, Schema Information의 다대일 정보를 살펴봐서 N을 가지는 컬럼이 상위로 되서 먼저 위치하게 된다.
- 3) 그 이외의 것은 계층구조에 상관없는 관계므로 임의의 순서로 나열한다.

STEP 3. 재구성된 관계형 릴레이션의 첫 컬럼부터 XML 엘리먼트를 만들어간다. 앞 컬럼값이 동일한 뒷 컬럼값들은 앞 컬럼의 자식 엘리먼트가 된다.

예외적으로, 사용자가 특정 정렬형식을 위해서, SQL 질의에 'order by'문으로 형식을 명시한 경우는 RDBMS가 넘겨주는 관계형 데이터를 재구성하지 않고, 컬럼 순서대로 정렬해서 그를 재구성한 결과 릴레이션으로 여기고 STEP 3을 바로 적용시킨다.

```
예1] SQL: select S_EMP.last_name, S_DEPT.name, S_REGION.name
from S_REGION, S_DEPT, S_EMP
where S_EMP.dept_id = S_DEPT.id
and S_DEPT.region_id = S_REGION.id;
```



STEP 2

S_EMP	S_DEPT	S_REGION		S_REGION	S_DEPT	S_EMP
lastname	name	name		name	name	lastname
김	판매	서울	재구성	서울	판매	김
박	기획	광주		서울	판매	최
최	판매	서울		서울	영업	박
남	영업	부산		서울	영업	김
박	영업	서울		부산	판매	류
류	판매	부산		부산	영업	남
관	판매	광주		부산	영업	박
김	영업	서울		광주	판매	문
박	영업	부산		광주	기획	박

STEP 3

```
<REGION>
<REGION_name> 서울
<DEPT>
<DEPT_name> 판매
<EMP>
<last_name> 김
```

```
</EMP>
<EMP>
<last_name> 최
</EMP>
</DEPT_name>
</DEPT>
<DEPT>
<DEPT_name> 영업
<EMP>
<last_name> 박
</EMP>
<EMP>
<last_name> 김
</EMP>
</DEPT_name>
</DEPT>
</REGION_name>
:
</REGION>
```

```
예2] SQL : select S_CUSTOMER.city, S_CUSTOMER.name, S_EMP.id
from S_CUSTOMER, S_EMP
where S_EMP.id = S_CUSTOMER.emp_id
and S_CUSTOMER.country = 'korea'
order by S_CUSTOMER.city;
```

S_CUSTOEMR	S_CUSTOMER	S_EMP
city	name	id
서울	홍길동	1
서울	이몽룡	2
서울	심철	1
서울	성준항	3
부산	장영실	8
부산	이희계	8
부산	이순신	9
광주	황진이	17
광주	유관순	16

최종 결과 릴레이션

5. 결론 및 향후 계획

본 논문에서는 기존의 방법들과는 다르게 사용자가 매번 XML 구조와 관계형 Scheme들을 생각해서 XML 문서의 구조를 지정하지 않고도, 관계형 데이터결과의 손실없이 XML문서를 작성하는 방법을 제시했다. 앞으로는 이 방법이 적용되는 구체적인 시스템을 구현하여 효율성에 대한 검증을 할 예정이다.

참고문헌

- [1] T.Bray et al., "Extensible Markup Language (XML) 1.0, "http://www.w3.org/TR/1998/REC-xml-19980210, 1998
- [2] M.Fernandez, W. Tan, D. Suci, "SilkRoute: Trading between Relations and XML," 9th international WWW Conference, May 2000
- [3] J.Shanmugasundaram, E. Shekita, R. Barr et al., "Efficiently Publishing Relational Data as XML Document,"26th International Conf. on VLDB, Egypt 2000
- [4] M.Fernandez, D.Suci., " Storing Semi-structured Data with STORED", ACM SIGMOD Conf.,Philadelphia, May 1999
- [5] K.Williams et al., Professional XML Database, Wrox press,2000