생물 정보 저장용 XML 데이터를 위한 유연한 RDB 스키마 생성 규칙

정석훈°, 박성준, 한동수
한국 정보 통신 대학교
{jsh°, psj, dshan}@icu.ac.kr

A Flexible RDB Schema Generating Rule for Biological XML Data

Suk-hoon Han°, Sung-jun Park, Dong-su Han
Information and Communications University

요 약

유전자, 단백질 등의 생물정보를 이용하는 여러 툴은 효율성의 극대화를 위하여 각각의 시스템에 맞는 데이터베이스 스키마 구성 및 필요한 정보의 선택적 저장이 필요하다. 하지만 구조 복잡성, 동일한 객체 데이터의 분산 등, 생물 정보 XML의 일반적인 특성 때문에 기존의 XML정보 저장 기법으로는 유연한 데이터베이스 스키마구성에 한계를 지닌다. 이 때문에 생물정보 XML로부터 로컬 데이터베이스를 구성하는 과정은 1:1파서를 구현하여 진행하고 있어 많은 시간과 비용이 소모된다. 본 논문에서는 생물정보 XML 의 특성과 그에 따른 유연한 RDB 스키마 구성의 제약에 대해 논하고, 이를 극복한 자유로운 RDB 스키마 구성을 위한 규칙을 소개한다. 본 규칙은 사용자가 원하는 RDB 스키마를 구성하여 생물정보 XML의 데이터를 저장하게 해주며, SQL 형태를 따르고 있어 사용자에게 익숙하다. 또한 분산된 생물정보 XML의 통합에도 유리하다.

1. 서 론

최근 생물정보학은 정보과학 기술을 이용하여 생명과학을 지원하는 하나의 학문으로서 그 역할을 견고히 하고 있다. 생물정보학자는 생물실험 정보를 모아 분석하고, 인지하기 쉬운 형태로 표현하는 툴을 제공해주거나, 기존의 생물학 실험에서 벗어나 계산, 통계 등을 기초로 하여 새로운 정보를 얻어낼 수 있는 기법을 제안하는 등, 생물학 발견에 도움을 주는 방법론을 고안해내고 있다. 각각의 생물학 연구 그룹들은 자신들의 연구분야에 해당하는 특정 데이터베이스를 구축하여 인터넷상에 공개함으로써 여러 연구자들이 이를 자유롭게 이용할 수 있게 한다. 이러한 데이터베이스는 대부분 플랫파일이나 XML형태, 혹은 그 확장 포맷으로 정보를 배포하는데, 현재 인터넷 환경에 유리하고 스스로 자료의 의미를 정의 할 수 있는 XML포맷을 장려하고 있다.

생물 정보를 이용하는 시스템은 필요에 따라 로컬 데이터베이스를 구성하고, 배포된 데이터를 저장하여 사용한다. 특히 계산 통계학적 방법을 사용하는 대단위 작업을 수행할 경우, 데이터베이스의 이용이 시스템 효율성에 큰 영향을 주므로 시스템의 성능을 극대화 시킬 수 있는 데이터베이스 스키마의 구성이 필수 적이다. 또한 필요에 따라 연구자는 여러 생물 정보 데이터 베이스를 조합하여 의미 있는 정보만을 저장하여 사용한다. 때문에 생물정보학자들이 인터넷상의 생물정보를 이용하기 위해서는 분산되어있는 데이터베이스를 자신의 목적에 맞게 통합하고 정제하는 선행과정이 필수적이라 할 수 있다.

XML기반의 인터넷 생물정보 데이터베이스를 로컬 데이터베이스에 저장할 경우 일반적으로 관계 형 데이터베이스(RDB)가 주로 사용된다. 생물정보 XML 데이터를 로컬 RDB에 저장하기 위해 현재 제안된 XML-RDB 사상 기법[1,2]을 사용한다면, 기법과 생물 정보 XML의 특성상 사용자는 XML구조로부터 생성된 특정 RDB 스키마를 사용할 수 밖에 없다. 따라서 사용자가 시스템 성능향상 등의 이유로 로컬 RDB 스키마를 변경하고자 한다면, 사용자는 원 XML파일의 데이터를 수용하면서 자신의 목적에 맞는 RDB 구조를 만든 후, 일일이 XML을 파싱하여 데이터를 저장시켜야 한다. 이와 같이 생물정보 XML로부터 로컬 RDB를 구성하는 과정은 1:1파서 구현이 필수적이기 때문에 많은 시간과 비용이 소모된다.

본 논문에서는 XML기반 생물정보를 수용하는 RDB 스키마의 유연한 정의를 위한 규칙을 소개한다. 이를 위하여 RDB 스키마 구성 시 고려돼야 할 생물정보 XML의 특성과 관련 연구 또한 살펴본다. 본 규칙은 사용자가 원하는 RDB 스키마를 구성하여 생물정보 XML 데이터를 저장하게 해주며, SQL 형태를 따르고 있어 사용자에게 익숙하다. 또한 분산된 생물정보 XML의 통합에도 유리하다.

2. 관련연구

현재까지 많은 XML연구자들에 의하여 XML 데이터를 RDB에 저장하기 위한 다양한 기법들이 소개되어 왔다.[1,2] 하지만 그 대부분은 XML과 RDB의 1:1 사상으로 XML 구조를 유지하는데 목적을 두어 자유로운 RDB 스키마 구성에 제약이 따른다. XML의 구조를 유지하는 기법은 차후 RDB로부터 XML을 재구성할 때 강점이 있지만, 시스템의 효율성을 위한 자유로운 RDB 구조 구성에는 제약이 있다. 몇몇의 연구는 데이터베이스 구성 시 유연한 구조 변경을 허락하지만[3] 여전히 특정한 XML구조에 영향을 받는다. 특히 생물 정보 XML의 경우, XML의 엘리먼트 이름 외에 에트리뷰트 이름까지 참조해야만 그 자료의 의미가 정확히 파악되는 구조가 빈번해,

자유로운 RDB 스키마 구성에 어려움이 있다.

(표 2)는 (그림 1) 의 정보를 저장하기 위하여 현재까지 소개된 기법 중[3]을 사용하여 사용자가 원하는 구조(표 1)에 가장 유사한 테이블 스키마를 구성한 예이다.

```
<DATASET  NAME="dataset_1">
    <REFERENCE    DATABASE="A"    ID = "A_ID"/>
    <REFERENCE    DATABASE="B"    ID = "B_ID"/>
    <REFERENCE    DATABASE="C"    ID = "C_ID"/>
    <DATA1>data1</DATA1>
    <DATA2>data2</DATA2>
    <DATA3>data3</DATA3>
</DATASET>
```

그림 1. 생물정보 XML의 예

NAME	REF_A	REF_B	REF_C	DATA
dataset_1	A_ID	B_ID	C_ID	Data1

표 1. (그림 1)의 XML로부터 생성된 테이블

NAME	DATA	DATABASE	ID
dataset_1	data1	A	A_ID
dataset_1	data1	B	B_ID
dataset_1	data1	C	C_ID

표 2. (그림 1)의 XML로부터 생성된 테이블

XML 뷰를 만들어 메모리에 가상 테이블을 생성하는 기법[] 또한 활발히 연구되고 있어 XML 기반 생물 정보원에 대한 적용[4]도 진행되고 있다. XML-뷰를 만들기 위해서는 X-query를 이용하는데 이는 튜플 전체에 조건 문을 주게 된다. 따라서 이러한 기법은 로컬 RDB 구조 구성 시 몇몇 XML기반 생물 정보원에 대해 적용이 가능하지만 특정한 XML 구조에 영향을 받는다.

또한 위 연구의 대부분은 효율적인 XML구조 파악을 위해 DOM 파서를 사용하여 시스템을 구성한다. 그러나 생물정보 XML의 경우 그 엄청난 데이터 양 때문에 DOM과 같은 메모리상 구조 및 자료 구성기법은 매우 비효율 적이다.

3. 생물정보 XML 데이터 저장을 위한 유연한 RDB 스키마 생성 규칙

3.1. XML기반 생물정보원의 특성 및 규칙의 설계 목표

생물정보 XML은 일반 XML과 비교하여 일정한 특성을 나타낸다. 때문에 생물 정보 데이터를 RDB에 저장하기 위해서는 그 특성을 파악할 필요성이 있다. 일반 XML과 구분되는 생물정보 XML의 특성 및 그에 따른 규칙의 설계 시 고려사항은 다음과 같다

첫째, 그 구조는 트리, 혹은 그래프의 형태를 지닌다. 생물정보 XML은 같은 데이터 클래스를 상속한 여러 객체를 지니므로 트리 구조가 일반적이다. 또한 각 데이터 객체들 간의 관계를 표현한 그래프 구조도 사용된다. 따라서 데이터가 RDB에 저장될 경우 각 그래프, 혹은 트리의 노드는 하나의 데이터 세트, 즉 튜플이 되며, 그래프 형태의 경우 참조키가 표현되어야 한다.

둘째, 구조가 다양하고 복잡하다. XML은 준 구조체로서 확장성을 지니므로 다양한 구조표현이 가능하다. 여러 데이터베이스가 같은 데이터 클래스를 표현한다 하더라도 그 자료 표현 형태가 다르며 표현 범위 또한 다양하다. 때문에 생물정보 XML을 사용할 때에는 그 구조에 대해 충분히 인지하고 있어야 한다. 또한 하나의 XML이 깊이가 깊은 다단계의 트리로 표현되어, RDB로 저장될 때 경우에 따라서 하나의 테이블이 아닌 여러 개의 테이블로 나뉘어 저장되어야 한다.

셋째, 생물정보 XML 구조 복잡성의 실례로 엘레먼트 이름만으로는 자료 의미의 전달이 정확하지 않음을 들 수 있다.. (그림 1)은 생물 정보 XML에서 빈번히 나타나는 구조이다. 데이터베이스의 정규화를 위해서는 'REFERENCE'에 표현된 자료들은 'DATABASE'의 값에 따라 서로 다른 의미로 저장되어야 한다.(표 1). (그림 1)은 같은 자료 도메인을 가지는 데이터베이스들의 통합 생물정보 XML로서 이러한 구조는 새 연구그룹 데이터베이스를 추가할 경우 유용하다. 예와 같은 구조가 쓰이는 또 다른 이유는 여러 생물정보 데이터베이스에서 다른 분류의 데이터 객체 임에도 같은 형태의 ID로 표현되는 것이 일반적이어서, 추가 분류 정보가 필요하기 때문이다.

넷째, 생물 정보 XML은 다른 XML과 서로 참조하는 관계를 가진다. 이러한 데이터 파일의 경우 RDB에서는 서로 관계를 가진 두 개의 테이블로 표현된다. 또한 서로 다른 XML에 같은 데이터 클래스를 가지는 경우도 있다. 이러한 경우 여러 개의 XML은 통합되어 하나의 테이블, 혹은 관계를 가진 몇 개의 테이블로 표현되어야 한다

마지막으로 데이터의 엄청난 양을 특징으로 들 수 있다. 지금까지 수많은 실험 데이터들이 축적되어 그 단위가 수 GB를 넘기도 한다. 또한 그 양은 계속에서 빠른 속도로 증가되고 있다. 따라서 본 규칙의 수행 시스템은 XML의 전체적인 구조 및 데이터를 메모리에 상주 시키지 않고 작업을 수행하도록 한다.

3.2 규칙의 정의

```
XCREATE TABLE TABLE_NAME(
        NAME CHAR(20) NOT NULL,
        REFERENCE_A CHAR(20),
        REFERENCE_B CHAR(20),
        REFERENCE_C CHAR(20),
        DATA CHAR(20),
        PRIMARY KEY (NAME) )
        AS ( SELECT (
                DATASET@NAME,
                [DATASET.REFERENCE]@ID
                        WHERE [].@DATABASE=A,
                [DATASET.REFERENCE]@ID
                        WHERE [].@DATABASE=B,
                [DATASET.REFERENCE]@ID
                        WHERE [].@DATABASE=C,
                DATASET  )
            FROM [file:/bio_data.xml].dataset  DATASET);
```

그림 2. (표 1) 생성을 위한 규칙

(그림 2)는 (그림 1)의 XML로부터 (표 3)의 테이블을 생성시키는 규칙이다. 규칙은 어떠한 형태로도 정의가 가능하나 사용자가 익숙한 SQL 의 형태를 따랐다. 하지만 WHERE 조건을 SELECT 문 전체는 물론 각각의 에트리뷰트에도 줄 수 있어 생물 정보 XML의 복잡한 구조 때문에 야기되는 자유로운 RDB 스키마 구성의 제약을

완화하였다. 'XCREATE TABLE' 절에는 원하는 RDB테이블 스키마를 정의 한다. 문법은 표준 SQL을 따르며, PRIMARY KEY, FOREIN KEY 등도 정의할 수 있다. 이때 테이블 의 한 튜플은 사용할 생물정보 XML에서 정의되는 하나의 데이터 셋과 같은 객체를 가리켜야 하며, 저장할 데이터는 선택이 가능하다. 'AS'절에는 각각의 에트리뷰트 당 생물정보 XML내의 데이터 위치 경로를 정의한다. WHERE절은 각각의 에트리뷰트의 조건 절이며, '[]'표기는 해당 위치 절과 중복되는 경로를 의미하여 규칙의 표기를 간편화 하였다. 'FROM절'에서는 사용될 생물정보 XML의 위치를 나타낸다. 문법은 [FROM 파일위치 별칭]으로 표기하며, 파일위치 표기 시 파일의 위치뿐 아니라 위 SELECT 문에서 사용될 XML내 데이터 경로의 공통 부분 또한 표기할 수 있다. 이때에 하나의 데이터 세트를 나타내는 경로까지 표기하기를 권장한다. 또한 여러 개의 파일을 정의하여 사용가능하며, 파일이 하나일 때에는 별칭 생략이 가능하다.

```
XCREATE TABLE TABLE_NAME(
        NAME CHAR(20) NOT NULL,
        ID CHAR(20) NOT NULL
        PRIMARY KEY (NAME, ID) )
        AS ( SELECT (
                DATASET@NAME,
                DATASET.REFERENCE@ID)
            FROM [file:/bio_dataset.xml].dataset  DATASET);
```

그림 3. (표 3)의 테이블 생성을 위한 규칙

NAME	ID
dataset_1	A_ID
dataset_1	B_ID
dataset_1	C_ID

표 3 다중 값 표현 테이블

(그림 3)는 (그림 1)로부터 (표 3)의 테이블을 생성시키는 예로서, 일반적으로 RDB 엔티티의 다중 값 표현 시 쓰이는 테이블 형태이다. 이러한 형태 또한 생물정보 XML 데이터 저장 시 자주 사용되는 테이블 스키마로써, 규칙은 XML의 한 데이터 세트 내에서, 한 에트리뷰트의 데이터 경로에 중복되는 값이 인식될 경우 다중 값 에트리뷰트로 인식하여 (표 3)과 같은 테이블을 생성한다.

```
XCREATE TABLE TABLE_NAME(
        ID CHAR(20) NOT NULL,
        DATA1 CHAR(20),
        DATA2 CHAR(20),
        PRIMARY KEY (A) )
        AS ( SELECT (
                DATASET1@ID,
                DATASET1.DATA1
                DATASET2.DATA2)
            WHERE DATASET1@ID = DATASET2@ID
             FROM [file:/bio_data1xml].dataset  DATASET1,
                [file:/bio_data2.xml].dataset  DATASET2,);
```

그림 4. 두 XML 파일의 통합을 위한 규칙

(그림 4)는 여러 개의 생물정보 XML에서 RDB 테이블을 생성하는 예로, 각 파일의 데이터 세트가 동일한 데이터 객체를 나타내지만 서로 가진 정보의 범위가 다를 때, 이를 하나의 테이블에 통합하는 경우이다. 이때 SELECT문 전체에 WHERE 조건을 주어 각 파일의 데이터 세트가 동일한 객체임을 인식한다.

4. 결 론

본 논문에서 소개된 규칙은 사용자가 자유롭게 RDB 스키마를 구성하여 생물정보 XML 데이터를 저장할 수 있게 한다. 본 규칙은 하나의 데이터 세트, 즉 하나의 튜플로 저장될 XML 노드뿐 아니라 각각의 에트리뷰트, 즉 테이블 컬럼에 저장될 데이터마다 조건 절을 주어, RDB 스키마 구성의 유연성을 높인다는 점에서 기존의 XML핸들링 방식과는 구분된다. 또한 다중 값 테이블 표현이 가능하며, 여러 생물정보 XML의 통합에 용이하다는 점에서 강점을 지닌다. 본 규칙은 생물정보 XML을 기반으로 고안되었으나, 일반 XML에 적용될 여지가 있다. 따라서 향후 일반 XML을 위한 확장, 적용 및 검토가 이루어 져야 할 것이다.

Reference

[1] Bourret, R, Bornhovd, C, Buchmann, A, A generic load/extract utility for data transfer between XML documents and relational databases; Advanced Issues of E-Commerce and Web-Based Information Systems, 2000. WECWIS 2000. Second International Workshop on 8-9 June 2000 pp.134 - 143

[2] D. Flourescu & D.Kossmann,. "Storing and Querying XML Data using an RDBMS.," Bulletin of the IEEE computer society Technical committee on Data Engineering 22(3), pp. 27-34, 1999.

[3] Bei Jia, Cai Fei, Tao Lie-Jun, Pan Jin-Gui, "A direct method of data exchange between XML and relational database" Information Technology Interfaces, 2004. 26th International Conference on 2004 Vol.1 pp.127 - 132

[4] Chun-Nan Hsu, Chia-Hui Chang,Harianto Siek, Jiann - Jyh Lu and Jen-Jie Chiou, "Reconfigurable Web Wrapper Agents for Web Information Integration." In Proceedings of IJCAI-2003 Workshop on Web Information Integration , 2003.