제 18 회 한국정보처리학회 추계학술발표대회 논문집 제 9 권 제 2 호 (2002.11)

공간 데이터베이스와 GML 문서간의 지리 정보 변환 방법

안영수*, 박순영, 정원일, 배해영
인하대학교 전자계산공학과
{owlet76, sunny, wnchung}@dblab.inha.ac.kr, hybae@inha.ac.kr

A method of converting geographic information between spatial database and GML

Young-Su An*, Soon-Young Park, Warnill Chung, Hae-Young Bae
Dept. of Computer Science & Engineering, Inha University

요 약

빠른 인터넷의 보급과 정보화 사회의 발전에 따라 다양한 형태의 문서를 통합하여 유통할 수 있는 XML(eXtensible Markup Language)이 등장하게 되었다. 이러한 XML 의 장점을 이용하여 지리적 특성을 가진 정보에 대한 문서의 표준이 요구되었고, 이에 따라 OGC(Open GIS Consortium)에서는 GML(Geography Markup Language) 명세를 제시하였다. Geometry 와 geographic feature 정보를 포함하고 있는 GML 은 다양한 데이터 타입에 대한 상호 운용성을 제공 하고, 통합된 지리 정보를 유통할 수 있게 하는 장점을 가진다. 현재 이러한 장점을 가진 GML 지원 시스템과 응용에 대한 연구가 활발히 진행 중이나, 구체적인 방법론에 관한 제시는 미흡하다.

이에 본 논문에서는 공간 데이터베이스의 지리 정보를 GML 명세에 따른 GML 문서로 변환하는 방법과 GML 문서 내에 표현된 지리정보를 공간 데이터베이스에 저장 하는 두 가지 방법을 제안 한다.

1. 서론

과거의 지리 데이터는 각각의 지리 정보 시스템의 저장 형식에 따라 각각 독립적으로 구축 되어 데이터의 사용이 특정 사용자에게만 국한되어 있었다. 그러나 오늘날에는 인터넷 뿐만 아니라 무선 PDA 등 다양한 분야에도 지리 정보의 응용이 필요하게 되었다. 이를 위해 태그의 확장이 용이하고 구조화된 정보로 표현 할 수 있는 XML 의 장점을 이용한 지리정보 표현 방법을 이용하게 되었다[4, 12, 13].

이에 OGC 에서는 다양한 형태로 표현된 지리적 정보를 표준화된 형식으로 표현하여 상호 운용할 수 있는 GML 명세를 제시하였다. GML 은 Feature 의 Geometry 와 속성들을 포함하는 지리 데이터의 저장과 전달을 위한 XML 인코딩 메커니즘과 문법을 정의한 것이다[1].

본 논문[1]은 공간 데이터베이스와 GML 문서상의 지리정보 변환에 대한 것으로, 첫째, 공간 데이터베이스에 저장된 정보를 GML 명세를 기반으로 한 GML 문서로 표현 한다. 또한 GML 명세에서는 아직 제시되지 않고 있는 공간 객체에 대한 주석 정보의 표현 방법을 새롭게 정의하여 주석 정보를 출력할 수 있도록 지원 한다. 둘째, GML 문서를 통한 지리 정보를 공간 데이터베이스에서 관리할 수 있도록 DTD 또는 XML 스키마 기반의 저장 메커니즘을 제공하여 서로 이질적인 공간 데이터베이스 사이의 지리 정보 교환을 가능하도록 한다.

본 논문의 구성은 다음과 같다. 제 2 장 관련연구에서는 본 연구에 기본이 되는 GML 과 DOM(Document Object Model)에 대해 간략하게 소개한다. 제 3 장에서는 본 연구의 중심이 되는 공간 데이터베이스와 GML 문서간의 지리 정보 변환 처리 방법에 대해 설명하고, 제 4 장에서는 실제 시스템 구현에 대해 설명한다. 마지막으로 제 5 장에서 이 논문의 결론 및 향후 과제를 논한다.

2. 관련 연구

2.1 GML

GML 은 상호 이질적인 지리 정보를 공통적으로 표현하기 위한 XML 인코딩 표준 명세이다. GML 1.0 과 GML 2.0 의 차이는 GML 1.0 은 지리 정보를 XML 로 인코딩 하기 위한 메커니즘으로 DTD 를 제공하였고, GML 2.0 에서는 DTD 의 한계로 인한 해결을 위해 스키마를 제공하고 있다는 점이다. 현재 최종 버전으로는 GML 2.1.2 를 제시하고 있다.

[1] 본 연구는 정보통신부의 대학 S/W 연구센터 지원사업의 연구 결과임.

GML 명세는 Simple Feature 를 기반으로 하며, 지리 정보 표현을 위한 세 가지 Geometry, Feature, XLink(XML Linking Language) 스키마를 제공하고 있다.

geometry.xsd 는 GML 1.0 명세에서 점, 선, 면과 같은 기하학적인 요소를 나타내던 geometry.dtd 를 대체하는 것으로, OGC Simple Features Model 에 따라 Point, LineString, LinearRing, Polygon, MultiPoint, MultiLineString, MultiPolygon, MultiGeometry 클래스와 관계된 geometry 엘리먼트를 제공한다. Curve, Surface, MultiSurface, MultiCurve 와 같은 타입은 생략 된다.

feature.xsd 는 GML 2.0 에서 사용하는 Feature Collection 모델을 정의하고 있으며, feature 타입 정의를 위해 geometry.xsd 를 <include> 엘리먼트를 사용하여 geometry 스키마가 포함하고 있는 정의와 선언을 이용 가능 하도록 하고 있다.

xlink.xsd 는 Link Attribute 를 제공 한다. Xlink 는 W3C (World Wide Web Consortium)의 추천 안으로 리소스 사이에 링크를 생성하여 엘리먼트가 XML 문서 안에 삽입 될 수 있도록 한다.

이러한 세가지 기본 스키마를 바탕으로 다양한 사용자의 지리정보에 대한 요구를 충족 시킬 수 있도록 어플리케이션 스키마의 작성을 제공 한다. 즉, 사용자는 자신의 어플리케이션에 대해 geometry, feature, xlink 스키마들을 기반으로 특정 어플리케이션 스키마를 작성 할 수 있다.

2.2 DOM

웹에서 문서 구조를 액세스하고 조작하는 표준적인 방법으로 W3C 에서는 W3C DOM 이라는 명세를 제시하였다. DOM 은 XML 과 HTML 을 위한 프로그래밍 API 로 언어 중립적, 플랫폼 중립적인 정의이다. 즉 DOM 을 구성하는 다른 객체에 대해 인터페이스는 정의하지만, 구현에 대해서는 어떤 명세도 제공하지 않는다. DOM 안에 있는 객체는 개발자가 XML 문서로부터 데이터를 읽고, 변경, 추가, 삭제할 수 있도록 제공한다. DOM 은 XML 문서의 문법이 적절하며 well-formed 임을 보장하고, 내부 문서 조작을 단순화 한다. 데이터 요소 사이의 관계를 표현하는 방법에 있어서 DOM 은 객체 지향형 데이터베이스와 관계형 데이터베이스에서 정보를 표현하는 방법과 유사하다. 이는 데이터베이스와 DOM 을 사용하는 XML 파일 사이의 정보 이동을 용이하게 한다..

현재 DOM 명세는 3 가지 레벨을 제공하고 있다. DOM 레벨 1 은 임의의 구조적 문서를 액세스 할 수 있는 인터페이스에 대한 명세로 DOM 라이브러리는 적어도 이 부분에 대한 기능을 제공 하도록 하고 있다. DOM 레벨 2 는 DOM 레벨 1 의 모든 객체를 포함하고, 스타일 시트를 적용한 객체 모델을 추가 지원하며, 문서에 대한 질의 기능과 이벤트 모델에 대한 정의 기능도 포함 한다. DOM 레벨 3 는 문서를 로드하고, 저장하는 방법, 내용 모델을 조작하고 유효성을 검증하는 방법, 새로운 이벤트 형과 인터페이스 추가 등에 대한 명세가 아직 작업 초안 상태에 있다.

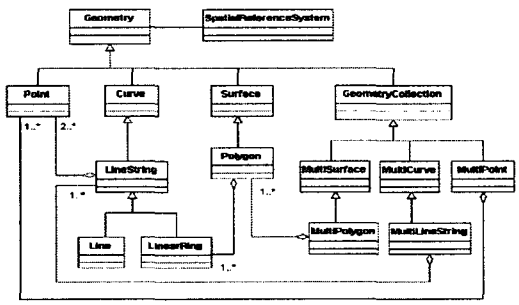
XML 문서를 읽고 조작할 경우 DOM 을 사용하면 다양한 플랫폼 간에 높은 수준의 정보처리 상호 운용을 보장할 수 있다.

3. 공간 데이터베이스와 GML 간의 지리 정보 변환

3.1 공간 데이터베이스에서 GML 로의 변환

공간 데이터베이스에 대한 사용자 요구를 처리하여 GML 문서로 생성하기 위해서는 공간데이타베이스에 대한 사용자 요구를 질의문으로 변환하는 과정, 변환된 질의를 공간 데이터베이스 인터페이스를 통해 보내어 반환된 결과를 분석하는 과정, 어플리케이션 스키마를 분석하는 과정, GML 문서로 변환 하는 과정으로 세분화 할 수 있다.

먼저 사용자 요청으로부터 레이어 정보를 추출 하고, 만약 확대 또는 축소 등 일정 MBR 에 대한 정보를 요청했을 경우에는 MBR 정보의 추출도 요구 된다. 이러한 정보는 질의문 형태로 변환 되어, 각 공간 데이터베이스 인터페이스를 통해 레이어 정보를 받는다. 반환된 결과는 공간 정보와 비 공간 정보로 분류 된다. 공간 타입은 GML 의 geometry 모델에 매핑 되고, 비 공간 타입은 XML 에서 제공하는 기본적이 타입으로 매핑 된다. [그림 1]는 GML 에서의 geometry 모델로서, 이는 기본적으로 OGC 에서 제시한 Simple Features 명세를 따르고 있다[1, 3].



[그림 1] Simple Features 에 대한 Geometry 모델

그러나, 공간 객체에 대한 주석에 대한 데이터 타입은 제시된 표준 타입을 가지고 있지 않다. 따라서, 본 논문에서는 이를 위해 새로운 사용자 정의 AnnotextType 타입을 정의 한다. [그림 2]은 실제 GML 문서 상에서 건물명에 대한 AnnotextType 이 어떻게 표현되고 있는지 보여 준다.

```
<건물ANNOTATION>
  <ANNOTATES>
   <gml:Point>
    <gml:coord>
      <gml:X>199057</gml:X><gml:Y>436819</gml:Y>
    </gml:coord>
   </gml:Point>
  </ANNOTATES>
  <ID>1</ID>
  <LABEL>인하대학교</LABEL>
  <ANGLE>0</ANGLE>
</건물 ANNOTATION>
```

[그림 2] AnnotextType 에 대한 GML 문서 예제

AnnotextType 은 주석을 표현할 위치 정보를 가지는 geo - metry 의 PointType 과 표시할 내용을 나타내는 StringType 으로 구성된 complex Type 으로 정의 한다.

질의 결과에 대한 분석 과정 후, GML 문서를 생성하기 위해서는 먼저 생성 되어진 어플리케이션 스키마에 대한 분석이 요구 된다. 이를 위해 본 논문에서는 DOM 인터페이스를 이용 한다[6]. 파싱된 결과는 계층적 트리 형태 구조를 생성한다. 이 결과를 바탕으로 구조적인 GML 문서를 생성한다. GML 의 구조적인 면을 설명하면, 우선 최상위 노드인 루트 엘리먼트가 있다. 루트 엘리먼트는 GML 에서 사용될 NameSpace[11]와 참조되는 스키마 위치에 대한 어트리뷰트를 가진다. [그림 3]는 건물 레이어에 대한 루트 엘리먼트 <건물 Schema> 이다.

```
<건물 Schema xmlns:gml=http://www.opengis.net/gml
   xmlns:xlink="http://www.w3.org/1999/xlink/"
   xmlns:xsi="http://wwww.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="file:/// c:\xml\\건물.xsd">
```

[그림 3] root element 에 대한 GML 문서 예제

위 예제는 OGC 에서 기본적으로 사용하는 엘리먼트에 대한 구분자로 'gml' 이라는 NameSpace 를 사용하도록 어트리뷰트를 정의하고 있으며, schemaLocation 어트리뷰트 값으로 어플리케이션 스키마의 위치를 갖는다. 이러한 루트 엘리먼트의 하위 엘리먼트로 MBR 영역을 갖는다. MBR 은 범위를 인코딩 하기 위해 사용되는 Box 엘리먼트로 정의 한다. [그림 4]는 건물 레이어에 대한 GML 표현이다.

```
<gml:boundedBy>
  <gml:Box srsName="http://www.opengis.net/gml/srs/epsg.xml#4326>
    <gml:coord>
      <gml:X>196575</gml:X><gml:Y>435627</gml:Y>
    </gml:coord>
    <gml:coord>
      <gml:X>208821</gml:X><gml:Y>448112</gml:Y>
    </gml:coord>
  </gml:Box>
</gml:boundedBy>
```

[그림 4] MBR 에 대한 GML 표현

SRS(Spatial Reference System) 는 지도에 그려질 geometry 들의 좌표들을 실제 지구의 위치와 일치시키는 좌표변환 시스템이며, geometry 들은 자신이 기준으로 하고 있는 SRS 가 어떤 것인가에 대한 정보를 가지고 있어야 한다. SRS 는 이러한 좌표변환 시스템에 대한 메소드를 가기고 있다. 이러한 참조되는 SRS 정보는 srsName 어트리뷰트로 정의 한다. 위 그림에서는 EPSG:4326(European Petroleum Survey Group)을 참조함을 정의 한다. 루트 엘리먼트의 또 다른 하위 엘리먼트로 레이어 엘리먼트를 정의 한다. 레이어 엘리먼트는 하위 엘리먼트로 해당 레이어에 대한 필드 엘리먼트와 해당 데이터 값을 정의 한다. 레이어는 공간 타입과 비 공간 타입으로 구성된 complexType 으로 정의 한다. [그림 5]은 Geometry 타입이 Polygon 인 건물 레이어에 대한 GML 표현이다.

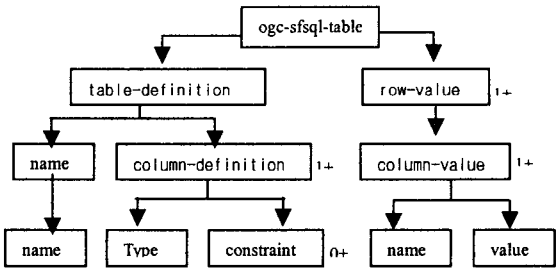
```
.<건물>
  <OBJ>
    <gml:Polygon>
      <gml:outerBoundaryIs>
        <gml:LinearRing>
          <gml:coord>
            <gml:X>199047</gml:X><gml:Y>438327</gml:Y>
          </gml:coord>
          <gml:coord>
            <gml:X>199052</gml:X><gml:Y>438322</gml:Y>
          </gml:coord>
                        ......
        </gml:LinearRing>
      </gml:outerBoundaryIs>
    </gml:Polygon>
  </OBJ>
  <OID>1</OID>
  <ID>1</ID>
</건물>
```

[그림 5] 건물에 대한 GML 표현

3.2 GML 데이터를 공간 데이터베이스로 저장

공간 데이터베이스에 저장 하기 위해서는 입력되는 GML 문서는 규격화되어 정의된 DTD 또는 스키마에 유효한 GML 문서임을 보장 한다. 세부적인 처리 과정을 살펴보면, 이러한 GML 문서를 분석하여 분석된 결과를 기반으로 데이터 저장을 위한 질의문을 생성하는 과정, 공간 데이터베이스 인터페이스를 통해 실제 데이터베이스에 데이터를 저장하는 과정으로 구성 된다.

먼저 입력된 GML 문서를 파싱하여 계층적 트리 구조를 생성하고, DOM 인터페이스를 통해 각 노드와 값을 분석 한다. 파싱된 GML 문서의 계층적 트리 구조는 [그림 6]과 같다.



[그림 6] 데이터 입력을 위한 GML 구조

GML 문서의 구조는 레이어에 대한 정의와 각 필드 정의 그리고 해당 데이터 정보를 갖는 구조로 정의 한다. 한 GML 문서에는 하나의 레이어에 대한 정보를 갖도록 정의 하고 GML 분석 후, 레이어 정의와 필드 정의를 통해 create table 질의문을 생성 하고, 생성된 테이블에 해당 데이터를 삽입하기 위한 insert 질의문을 생성 한다. 이 생성된 질의문은 공간데이터베이스 인터페이스를 통해 전달되어 실제 공간 데이터베이스에 테이블을 생성, 저장하게 된다. 이러한 GML 은 서로 이질적인 공간 데이터베이스 사이에 데이터 교환을 용이하게 하는 장점을 제공 한다. [그림 7]은 [그림 6]과 같은 계층적 구조를 바탕으로 생성된 강 레이어 입력을 위한 GML 문서를 보여준다.

```
<ogc-sfsql-table>
  <table-definition>
    <name> lakes </name>
    <column-definition>
      <name>shore</name>
      <type>POLYGON</type>
    </column-definition>
                                    ...
  </table-definition>
  <row-value>
    <column-value>
      <name>shore</name>
        <value>POLYGON((52 18,66 23,73 9,48 6,52 18),
        (59 18,67 18,67 13,59 13,59 18))</value>
    </column-value>
  </row-value>
                                    ...
</ogc-sfsql-table>
```

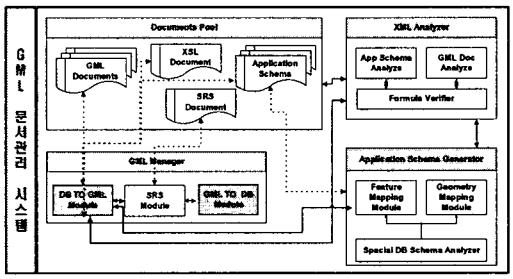
[그림 7] 강 레이어 데이터 입력을 위한 GML

4. 공간 데이터베이스와 GML 문서간의 지리정보 변환 방법에 따른 구현

본 논문에서는 GMS(GEO ManiaMillennium Server) 기반 GML 문서 관리 시스템의 개발을 통해 공간데이터와 GML 문서간의 변환 방법을 구현하였다. GMS 는 공간 및 비 공간 데이터에 대한 저장 및 연산을 처리하는 관계형 데이터베이스로 공간 데이터베이스 엔진으로 사용된다[14]. XML 파서로는 DOM 레벨 1 을 지원하는 MSXML 을 사용하여 GML 문서의 유효성을 검증 한다. GMS 기반 GML 문서관리 시스템의 구조는 다음 [그림 8]과 같다. GML 문서관리 시스템은 크게 Application Schema Generator, XML Analyzer 그리고 이 논문의 중심이 되는 GML Manager 로 구성 된다.

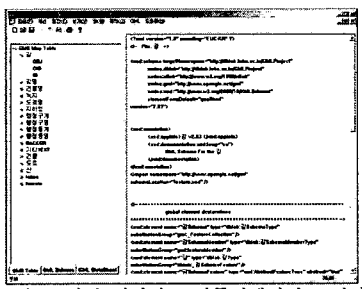
Application Schema Generator 는 GML 문서 생성시 참조하는 어플리케이션 스키마를 자동 생성하는 모듈이다. 스키마

는 각 레이어 단위로 생성되며, 표준화 되지 않은 공간 객체의 주석 정보도 새로운 사용자 정의 타입으로 스키마를 작성해 주는 특징을 가지고 있다. XML Analyzer 는 XML 파서를 이용하여 작성된 GML 문서에 대한 well-formed 검증과 유효성 검증, 분석을 위한 모듈이고, GML Manager 는 본 논문에서 제시한 방법의 구현된 GMS 의 데이터를 GML 로 변환하는 모듈과 GML 문서의 데이터를 GMS 에 저장하는 모듈, 좌표 체계 확인 및 수정, 변환을 제공하는 SRS 모듈로 구성된다.



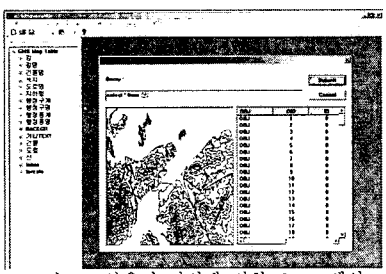
[그림 8] GML System Architecture

공간 데이터베이스와 GML 문서간의 지리 정보를 변환 방법을 사용하는 GML 문서관리 시스템은 공간 데이터베이스의 데이터를 OGC GML 명세에 따르는 GML 문서로 생성하고, 또한 입력되는 GML 문서의 데이터를 공간 데이터베이스에 저장하는 기능을 중심으로 구현 한다. [그림 9]은 Application Schema Generator 에서 GML 문서로 변환 시 참조하기 되는 어플리케이션 스키마의 자동 생성을 보여준다. 공간 데이터베이스와 연결이 성공하면 각 레이어에 대한 정보를 리스트로 보여주고, 해당 레이어를 선택하면 자동 생성해 준다.



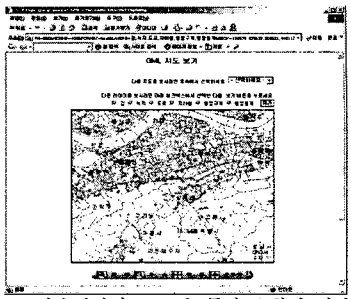
[그림 9] 자동 생성되는 어플리케이션 스키마

[그림 10]는 사용자 질의에 대한 GML 문서 생성 화면이다. 문서 생성과 함께 질의 결과에 대한 테이블 정보와 view 를 보여주고 있다.



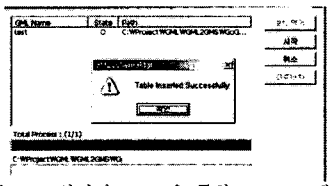
[그림 10] 사용자 질의에 의한 GML 생성

[그림 11]은 웹 환경에서의 GML 생성을 보여 준다. 각 레이어별로 GML 을 생성하여, 통합된 형태로 브라우저에서 출력해 주고 있다



[그림 11] 브라우저에서 URL 을 통한 요청에 대한 GML 생성

[그림 12]은 입력된 GML 문서의 데이터를 공간 데이터베이스(GMS)에 저장 하는 화면이다. 이 프로그램은 다른 환경의 공간 데이터베이스의 데이터를 상호 호환할 수 있도록 제공해 준다.



[그림 12] 입력된 GML 을 통한 GMS 로 데이터 저장

5. 결론 및 향후 과제

인터넷의 보급과 정보화 사회의 발전은 지리 정보 시스템에도 적용되어, 상호 이질적인 환경에서 지리 정보를 상호 운용할 수 있는 표준이 요구되었고, 이에 OGC 에서는 XML 의 장점을 이용하여 지리 정보에 응용할 수 있는 GML 명세를 제시하였다. 본 논문에서는 이러한 GML 명세를 기반으로, 공간 데이터와 GML 문서간의 변환 방법을 제안 하였고 이를 구현하였다.

향후 연구 과제로는 XML 문서에 대한 질의를 제공하는 XQuery 를 응용하여 이미 생성된 GML 문서에 대한 검색을 할 수 있는 연구가 필요 하다.

참고문헌

[1] OpenGIS Consortium, Inc., Geography Markup Language (GML) Implementation Specification 2.1.1, 2002.
[2] OpenGIS Consortium, Inc., A Web Mapping Scenario, 1998.
[3] OpenGIS Simple Features Specification For SQL Revision 1.1, 1999.
[4] W3Consortium, XML Specification ("XMLspec"), 1998.
[5] OpenGIS Consortium, Inc., Topic 1 - Feature Geometry, 1999.
[6] W3Consortium, Document Object Model (DOM) level 1 Specification Version 1.0, 1998.
[7] W3Consortium, XML Schema Part 0: Primer, 2001.
[8] W3Consortium, XML Schema Part 1: Structures, 2001.
[9] W3Consortium, XML Schema Part 2: Data types, 2001.
[10]Conformance Test Guidelines for OpenGIS Simple Features Specification for SQL, Revision 1.0. 1998.
[11] W3Consortium, Namespace in XML, 1999.
[12]GML 2.0 Enabling the Geospatial Web,www.geospatial online.com, 2001.
[13] GML Technology, http://www.galdosinc.com
[14] 이환재, 안준순, 강동재, 이경모, 정보홍, 박동선, 배해영, "GEO/Millennium 클라이언트-서버 공간 데이터베이스 시스템", 한국정보과학회 학술발표논문집, 27 권 1 호, pp.48 - 50, 2000. .