특 집

지리정보시스템을 위한 XML

권 준 희*, 윤 용 익**

● 목 차 ●

1. 서 론
2. 지리정보시스템을 위한 XML 표준화 동향
3. 지리정보시스템을 위한 XML의 특징
4. GML 사양
5. 활용 사례
6. 향후 발전 방향
7. 결 론

1. 서 론

과거에는 지리정보 데이터가 각 지리정보시스템 엔진의 자체 저장 형식에 따라 독립적으로 구축되었고, 이에 따라 고가의 엔진을 사용하여 구축된 지리정보 데이터를 활용하였다. 따라서, 단지 전문가나 특정한 사용자 중심으로 지리정보시스템이 활용되었다. 그러나, 최근에는 언제 어디서나 지도를 이용하려는 일반 사용자들의 요구가 급격히 증가하고 있다[4].

이러한 움직임과 함께, XML(eXtensible Markup Language)은 W3C(World Wide Web Consortium)에 의해 웹 표준이 되면서, 차세대 웹 문서 표준으로 다양한 분야의 정보시스템들이 XML을 속속 채택하고 있다. 또한 XML은 웹 뿐만 아니라 다양한 분야에서 표준으로 주목받고 있다. 이를 위해, 지리정보시스템에서도 미래의 지리정보 데이터 교환 표준으로 XML을 채택하고자 하는 많은 연구가 진행되고 있다[5].

본 논문에서는 지리정보시스템을 위한 XML의 표준화 동향과 지리정보시스템을 위한 XML의 특징에 대해서 설명하고, 이에 대한 표준안 중 하나인 GML의 사양과 활용사례를 기술한다. 마지막으로 지리정보시스템을 위한 XML의 향후 발전 방향을 살펴보도록 한다.

2. 지리정보시스템을 위한 XML 표준화 동향

지리정보시스템을 위한 XML 표준화 움직임으로는 OGC(OpenGIS Consortium)에 의한 GML (Geography Markup Language)과 일본의 DPC (Database Promotion Center)에 의한 G-XML을 들 수 있다.

GML과 G-XML은 서로 상호 경쟁하는 가운데 협력을 계획 중이다. 이를 위해 OGC와 DPC는 GML과 G-XML간 공통적인 부분을 찾아내고 서로 균형 있는 발전을 취할 것에 동의하였고, 이를 통해 ISO(International Standardization Organization) 표준으로 제정될 것을 궁극적인 목표로 하고 있다[4].

GML과 G-XML 중 현재까지 어떤 것이 표준화로 결정될 것인가에 대해 결론을 내리기에는 이르지만, OGC의 규모와 영향력으로 볼 때 GML이

* 숙명여자대학교 컴퓨터과학과 박사과정
** 숙명여자대학교 정보과학부 교수

G-XML에 비해서 우세한 것으로 보인다. 따라서, 본 논문에서는 G-XML에 대해서는 간략한 소개로 그치고 GML을 중심으로 설명하고자 한다. 단, G-XML에서 다루고 있는 기술적 내용 중 많은 부분이 GML의 향후 발전 계획과 일치하는 부분이 많으므로, G-XML은 향후의 GML 개발에 있어 중요한 기초 역할을 수행할 것으로 예상된다.

2.1 G-XML

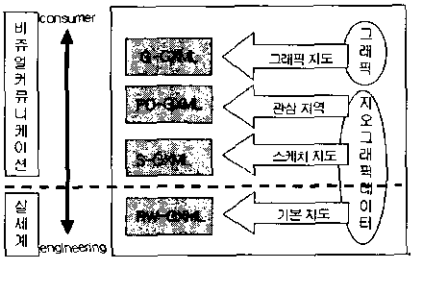
G-XML[4,5] 프로젝트는 G-XML 프로토콜로 불리우는 표준 XML 기반 프로토콜을 개발하기 위해 만들어졌다. 이러한 프로토콜 외에도 G-XML 프로토타입으로 불리우는 몇가지 프로토타입 시스템을 개발하였다. G-XML은 2000년 5월 버전 1.0이 출시된 이래 버전 2.0이 발표되었고, 현재 버전 3.0을 개발 중으로 그 구성도를 살펴보면 (그림 1)과 같다. G-XML은 JISC(Japan Industrial Standards Committee)의 승인을 준비 중이며, OGC와 ISO/TC211 워킹 그룹에 제출할 것을 계획 중에 있다.

G-XML 프로토콜은 다음과 같은 4가지 전형적인 어플리케이션 형태를 목적으로 설계되었다.

- RW-GXML(Real World G-XML) : 일반적인 GIS 도메인을 목적으로 한 G-XML을 의미한다.
- PD-GXML(Point & Direction based G-XML) : 일반 사용자가 휴대폰과 모바일 컴퓨터를 사용해서 외부에서 지리정보 데이터를 쉽게 다룰 수 있도록 한다.
- S-GXML(Semantic G-XML) : 지오메트리 데이터(geometric data)가 없는 지도 데이터를 대상으로 한다. 이는 쉽게 이해할 수 있는 스케치 지도나 음성/문자 항해 기법에 적용될 수 있다.
- G-GXML(Graphic based G-XML) : SVG (Scalable Vector Graphics)를 사용한 지리정보 인코딩 프레임웍을 제공한다.

G-XML 프로토콜을 사용한 프로토타입은 다음의 7가지를 들 수 있다.

- G-XML 뷰어 : Java2 애플릿 기반 환경의 RW-GXML 뷰어
- G-XML 에디터 : RW-GXML을 읽고 쓰고 편집하기 위한 Java2 어플리케이션
- G-XML 메일러 : RW-GXML의 형태로 포맷된 지도를 포함하여 이메일 메시지를 교환하기 위한 Java2 어플리케이션
- G-XML 모바일 통신 : PD-GXML 인스턴스(instance)로 관심있는 지역을 만들고, 모바일 통신기에서 G-GXML 인스턴스를 보기 위한 Java2 마이크로판 어플리케이션
- G-XML Shape 파일 전환 : Shape파일과 RW-GXML 인스턴스간 데이터 전환을 가능하게 해주는 Java2 어플리케이션이다. 이 프로토타입을 통해 파일 형태의 기존 지리정보 데이터를 RW-GXML 인스턴스로 손쉽게 전환이 가능함을 보였다.
- 오라클을 위한 G-XML 래퍼 : 오라클 데이터베이스로부터 RW-GXML, PD-GXML, S-GXML 인스턴스를 추출하고 저장하기 위한 Java2 서블릿으로, 상용 데이터베이스와의 연계가 가능함을 보이기 위함이 목적이다.
- SVG를 위한 G-XML 래퍼 : RW-GXML, PD-GXML, S-GXML 인스턴스를 G-GXML 인스턴스로 전환하기 위한 Java2 서블릿



(그림 1) G-XML의 구성도

2.2 GML

GML[7,8]은 OGC에 의해 개발된 공간, 비공간 정보를 포함한 지리정보 데이터를 저장하고 전송하기 위한 XML 기반 인코딩(encoding) 표준이다. GML은 2000년 5월 버전 1.0으로 발표된 이후, 2001년 2월 버전 2.0으로 릴리즈 되어 발표되었다.

OGC는 개방형 지리정보시스템 환경을 위해 지리정보 데이터와 어플리케이션간 표준 인터페이스를 제시하는 것을 목표로 하고 있다. OGC와 ISO/TC211은 상호간 기술적 제휴를 승인하기로 서로 동의한 바 있고, 이를 토대로 OGC사양이 요구사항을 충족할 때마다, 이를 ISO사양으로 채택하게 된다. OGC는 이를 위해 16종류의 추상 사양(abstract specification)과 7종류의 구현 사양(implementation specification)을 제시하고 있다. 이 중 GML은 공식적인 표준으로 채택되기 직전인 권고안 상태이며, 공식적인 OGC 구현 사양으로 채택되기 위한 마지막 작업 중에 있다[14]. 또한, 지리정보시스템 업계에 있어 영향력 있는 많은 벤더들(ESRI, MapInfo, LaserScan 등)이 GML 사양을 만드는데 공헌하였다. 이는 GML이 공식적인 표준화 사양으로 자리매김하는데 어려움이 없을 것임을 시사한다고 볼 수 있다.

본 논문에서 소개하고자 하는 GML사양은 OGC의 권고안 버전 2.0[8]에 기초하고 있다. GML 2.0은 XML 스키마(Schema)를 기반으로 하여 복잡한 피처와 그들간의 연관관계를 인코딩할 수 있도록 3가지 기본적인 스키마를 제공한다. 이는 기존의 GML 1.0이 XML DTD에 기반하여 3가지 형태의 프로파일로 표현되어 있는데 비해서 보다 강력한 기능이다.

3. 지리정보시스템을 위한 XML의 특징

갤도스 시스템 사에서는 웹 상의 지리정보 처리에 GML을 사용해야 하는 12가지 이유에 대해 설명하고 있다[2].이를 토대로 지리정보시스템을 위한 XML의 특징을 정리해보면 다음과 같다.

첫째, 벡터 데이터를 지원한다. 따라서, 웹 상의 일반적인 방식인 GIF나 JPG 등의 지도 이미지 방식에 비해 요청된 해상도대로 디스플레이 해주며 피처당 컨트롤이 가능하다.

둘째, 데이터와 이에 대한 표현(presentation)을 분리하는 원칙하에 설계되어 있다. 이를 통해 VML (Vector Markup Language), SGML(Scalable Vector Graphics)과 같은 표준 웹 그래픽 언어의 사용으로 보다 유연하고 높은 품질의 데이터 표현이 이루어진다. 또한, 데이터를 표현하기 위해 서로 다른 스타일시트(stylesheet)를 사용하여 지도를 랜더링할 수 있어 스타일시트 선택으로 지도를 커스터마이징할 수 있게 된다.

세째, 피처와 관련된 링크를 임베딩하기가 쉽다. 이러한 링크는 간단한 URL 웹 주소일 수도 있고, 보다 복잡한 형태일 수도 있다. 예를 들어 지도 상에서 오페라 하우스를 클릭하면 해당 홈페이지로 링크되는 것을 생각해볼 수 있다.

네째, 서버 측의 별도 데이터에서 질의를 처리하는 방식과는 달리, 자신의 데이터에서 직접 질의를 처리하므로 보다 좋은 질의 처리 능력을 보여준다.

다섯째, 비정형의 데이터를 지원한다. 예를 들면, 시간에 따라 변화하는 피처를 다룰 수 있다. 즉, 애니메이션이 필요한 피처를 정의하고 피처의 속성(attribute)으로 애니메이션에 필요한 알고리즘을 정의하여 시간에 따라 변화하는 피처를 다룰 수 있다.

여섯째, 지리정보 데이터의 상호 교환 포맷으로 사용될 수 있다. 지리정보시스템에 있어 가장 큰 문제점 중 하나는 데이터 포맷의 다양성에 있었고 이는 개방 환경에서 데이터 상호교환을 어렵게 만드는 중요한 요인 중 하나였다. 이에 대해 XML은 해결책을 제시한다.

4. GML 사양

4.1 GML 스키마

4.1.1 기본 스키마(base schema)

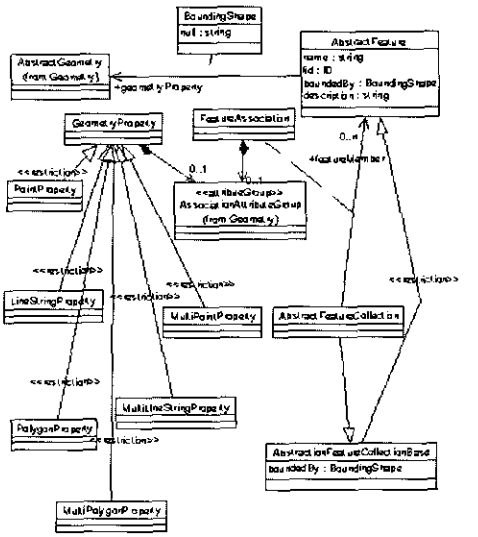
GML은 XML 스키마(schema) 후보 권고안 [10][11]과 XML 네임스페이스(namespace) 권고안 [6]에 기반하고 있다. 이를 토대로 3가지 기본 스키마를 제공하고 이들을 UML의 패키지와 클래스 다이어그램으로 표현하면 (그림 2), (그림 3), (그림 4)와 같다. 지오메트리(geometry) 스키마와 피처 스키마는 gml 네임스페이스를 사용하고 XLinks 스키마는 xlink 네임스페이스를 사용하여 기본 스키마와 어플리케이션 스키마를 분명하게 분리하여 준다.

(그림 2) 기본 스키마 패키지

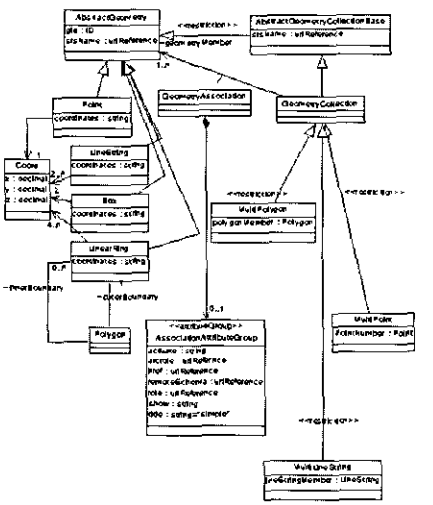
- 지오메트리 스키마(geometry.xsd) : 지오메트리 엘리먼트(element)를 정의한다. 이를 위해, OGC 심플 피처(Simple Feature)사양에 따라 Curve, Surface, MultiSurface, MultiCurve를 제외한 Point, LineString, LinearRing, Polygon, MultiPoint, MultiLineString, MultiPolygon, 이들에 대한 GeometryCollection 및 Box를 제공한다.
- 피처 스키마(feauture.xsd) : 피처 타입, 피처 식별자(fid),이름,설명과 같은 공통적인 피처 속성을 정의한다. 피처 스키마는 지오메트리 스키마를 참조로 하며, 이를 위해 <include>엘리먼트를 사용한다.

```
<include shemaLocation= geometry.xsd />
```

- XLinks 스키마(xlinks.xsd) : 링킹 기능을 지원하기 위한 Xlink 어트리뷰트(attribute)를 제공한다.



(그림 3) 지오메트리 스키마 패키지



(그림 4) 피처 스키마 패키지

4.1.2 어플리케이션 스키마(application schema)

기본 스키마가 지리정보를 표현하기 위한 기본적인 형태를 제공한다고 한다면, 어플리케이션 스

키마는 이를 활용한 어플리케이션 특수 스키마라고 볼 수 있다. 즉, 어플리케이션 스키마 작성자는, 기본 스키마를 이용함으로써 OGC표준에 대한 깊은 이해 없이도 표준에 맞는 어플리케이션을 보다 손쉽게 개발할 수 있게 되는 것이다.

4.2 GML인코딩

GML 사양에서는 다음과 같은 6가지 방법에 대한 인코딩을 소개하고 있다.

- 지오메트리 없는 피처 인코딩
- 지오메트리 인코딩
- 지오메트리를 가진 피처 인코딩
- 피처 콜렉션 인코딩
- 피처간 연관관계 인코딩

이 중, 지오메트리를 가진 피처 인코딩만을 그 예로 살펴보면 다음과 같다.

```
<element name="Dean" type="ex:DeanType"
 substitutionGroup="gml:_feature"/>

<complexType name="DeanType">
   <complexContent>
    <extension base="gml:AbstractFeatureType">
      <sequence>
       <element name="familyName"type="string"/>
       <element name="age" type="integer"/>
       <element name="nickname" type="string"
        minOccurs="0" maxOccurs="unbounded"/>
       <element ref="gml:location"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
```

(그림 5) 어플리케이션 스키마

GML은 피처와 관련된 지오메트리 정보와 미리 정해진 속성을 제공하므로, 이를 사용하여 어플리케이션을 개발할 수 있도록 한다. 이 예에서는 DeanType 피처가 Point Property 에 대한 다른 이름인 location 이라고 불리우는 미리 정해진 속성을

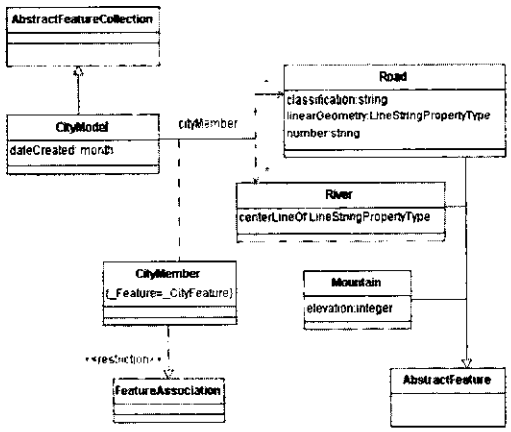
```
<Dean>
  <familyName>Smith</familyName><age>42</age>
  <nickName>Smithy</nickname>
  <nickName>Bonehead</nickname>
  <gml:location>
   <gml:Point>
      <gml:coord><gml:X>1.0</gml:X>
            <gml:Y>1.0</gml:Y></gml:coord>
   </gml:Point>
  </gml:location>
</Dean>
```

(그림 6) 어플리케이션 스키마를 사용한 XML

사용하여 손쉽게 어플리케이션 스키마를 만들어내고 있다. GML에서는 기본 스키마 외에도 유용한 다른 몇가지 스키마를 더 제공하고 있으며, 이는 [9]를 참조하기 바란다.

4.3 예제

본 절에서는 도시(city) 에 대한 어플리케이션 스키마와, 이를 이용하여 Cambridge 데이터를 만드는 과정을 살펴보도록 한다. 도시(city) 에 대한 UML 다이어그램은 (그림 7)이며, 이를 어플리케이션 스키마로 표현하면 (그림 8)과 같다. 이러한 어플리케이션 스키마를 사용한 XML파일은 (그림 9)와 같다.



(그림 7) city를 위한 UML다이어그램

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://www.opengis.net/examples"
xmlns:ex=http://www.opengis.net/examples
xmlns:xlink=http://www.w3.org/1999/xlink
xmlns:gml=http://www.opengis.net/ gml
xmlns="http://www.w3.org/2000/10/XMLSchema"
elementFormDefault="qualified" version "2.03">
 <annotation>
    <appinfo>city.xsd v2.03 2001-02</appinfo>
    <documentation xml:lang="en"> GML schema for the
       Cambridge example
    </documentation>
 </annotation>

 <import namespace="http://www.opengis.net/gml"
       schemaLocation="feature.xsd"/>

 <element name="CityModel" type="ex:CityModelType"
      substitutionGroup="gml:_FeatureCollection" />
 <element name="cityMember" type="ex:CityMemberType"
    substitutionGroup="gml:featureMember"/>
 <element name="Road" type="ex:RoadType"
    substitutionGroup="ex:_CityFeature"/>
 <element name="River" type="ex:RiverType"
    substitutionGroup="ex:_CityFeature"/>
 <element name="Mountain" type="ex:MountainType"
    substitutionGroup="gml:_Feature"/>

<element name="_CityFeature"
     type="gml:AbstractFeatureType" abstract="true"
    substitutionGroup="gml:_Feature"/>

 <complexType name="CityModelType">
   <complexContent>
     <extension base="gml:AbstractFeatureCollectionType">
       <sequence><element name="dateCreated"
            type="month"/></sequence>
     </extension>
   </complexContent>
 </complexType>

 <complexType name="CityMemberType">
   <complexContent>
     <restriction base="gml:FeatureAssociationType">
       <sequence minOccurs="0">
         <element ref="ex:_CityFeature"/>
       </sequence>
       <attributeGroup ref="gml:AssociationAttributeGroup"/>
     </restriction>
   </complexContent>
 </complexType>

 <complexType name="RiverType">
   <complexContent>
     <extension base="gml:AbstractFeatureType">
       <sequence>
         <element ref="gml:centerLineOf"/></sequence>
     </extension>
   </complexContent>
 </complexType>

 <complexType name="RoadType">
   <complexContent>
     <extension base="gml:AbstractFeatureType">
       <sequence>
         <element name="linearGeometry"
            type="gml:LineStringPropertyType"/>
         <element name="classification" type="string"/>
         <element name="number" type="string"/>
       </sequence>
     </extension>
   </complexContent>
 </complexType>

 <complexType name="MountainType">
   <complexContent>
     <extension base="gml:AbstractFeatureType">
       <sequence><element name="elevation"
type="integer"/></sequence>
     </extension>
   </complexContent>
 </complexType>
</schema>
```

(그림 8) city.xsd : 어플리케이션 스키마

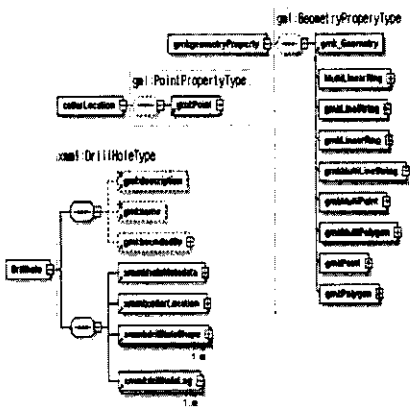
```
<?xml version="1.0" encoding="UTF-8"?>
<CityModel xmlns="http://www.opengis.net/examples"
           xmlns:gml="http://www.opengis.net/gml"
           xmlns:xlink="http://www.w3.org/1999/xlink"
           xmlns:xsi=

"http://www.w3.org/2000/10/XMLSchema-instance"
           xsi:schemaLocation=
                "http://www.opengis.net/examples city.xsd">

 <gml:name>Cambridge</gml:name>
 <gml:boundedBy>
   <gml:Box srsName=
        "http://www.opengis.net/gml/srs/epsg.xml#4326">
     <gml:coord><gml:X>0.0</gml:X><gml:Y>0.0</gml:Y>
     </gml:coord>
```

```
    <gml:coord><gml:X>10.0</gml:X><gml:Y>10.0</gml:Y>
      </gml:coord>
    </gml:Box>
  </gml:boundedBy>

  <cityMember>
    <River>
      <gml:description> Cambridge River</gml:description>
      <gml:name>Cam</gml:name>
      <gml:centerLineOf>
        <gml:LineString srsName=
            "http://www.opengis.net/gml/srs/epsg.xml#4326">

<gml:coord><gml:X>0</gml:X><gml:Y>50</gml:Y>
          </gml:coord>
          <gml:coord><gml:X>70</gml:X>
                    <gml:Y>60</gml:Y></gml:coord>
          <gml:coord><gml:X>100</gml:X>
                    <gml:Y>50</gml:Y></gml:coord>
        </gml:LineString>
      </gml:centerLineOf>
    </River>
  </cityMember>

  <cityMember>
    <Road>
      <gml:name>M11</gml:name>
      <linearGeometry>
          <gml:LineString srsName=
            "http://www.opengis.net/gml/srs/epsg.xml#4326">
            <gml:coord><gml:X>0</gml:X>
                    <gml:Y>5.0</gml:Y></gml:coord>
            <gml:coord><gml:X>20.6</gml:X>
                    <gml:Y>10.7</gml:Y></gml:coord>
            <gml:coord><gml:X>80.5</gml:X>

<gml:Y>60.9</gml:Y></gml:coord>
          </gml:LineString>
      </linearGeometry>
    <classification>motorway</classification>
      <number>11</number>
    </Road>
  </cityMember>

  <dateCreated>2000-11</dateCreated>
</CityModel>
```

(그림 9) Cambridge.xml

5. 활용 사례

지리정보시스템에 있어서 지리정보 데이터에 대해 XML을 활용한 사례로는 XMML과 LandXML을 들 수 있다. CSIRO에서 개발된 XMML(eXploration & Mining Markup Language)은 GML을 활용하여 이를 탐사 및 광물 도메인에 적용하여 이를 별도의 마크업(markup) 언어로 사용하고 있는 대표적인 사례로 볼 수 있다[13]. XMML은 광석, 채광, 지질학에 관련되어 보다 차원높은 3D지형을 기술할 수 있는 특징을 가진다. XMML은 XML기반에서 만들어졌기 때문에 소프트웨어 어플리케이션, 사무실, 탐사자 등 다양한 사용자와 환경에서 모두 사용될 수 있어 소프트웨어와 데이터의 비호환성에 의한 문제를 해결할 수 있다. 즉, XMML은 GML의 기본 스키마를 임포트하여 사용함으로써 탐사 및 광물 분야라는 응용 도메인 컴포넌트를 만들어낸 사례로 볼 수 있다.



(그림 10) GML 기반 XMML

LandXML[12]은 Autodesk에 의해 개발된 토지 개발 분야에서 활용될 수 있는 XML이다. LandXML은 BizTalk,aecXML,GML에 기반하고 있으며 이 중 지리정보 데이터에 대해서는 GML에 기반하고 있다. 이를 통해 다양한 형태로 추출되는 공학과 설

계 데이터를 LandXML이라는 단일한 형태로 표현할 수 있게 된다.

6. 향후 발전 방향

지리정보시스템을 위한 XML의 발전방향을 살펴보기 위해 현재 개발 중인 GML 버전 3.0에서의 고려사항과, OGC에서 제시한 웹서비스 발전을 위한 RFT(Request For Technology)를 살펴보도록 한다.

GML 버전 3.0의 내용 중 몇가지를 살펴보면 다음과 같다[3].

- GML 모델과 문법 정형화
- 현재의 지오메트리 확장
- 위상(Topology) 추가
- 커버리지(coverage) 지원
- 표준적인 스타일링 지원
- 시간 개념 지원
- 메타데이터 지원

OGC의 웹서비스 발전을 위한 RFT에서 제시하고 있는 내용 중 XML과 관련된 내용 몇가지를 간추려보면 다음과 같다[1].

- WMT-3(Web Mapping Testbed Phase 3) : 비쥬얼라이징(visualization)을 위한 스타일링 기법 및 대용량 지리정보 데이터 처리를 위한 XML 파일 압축 및 스트리밍 기법에 대한 연구
- 3차원 및 4차원 지원 : 시간, 위상, 보다 복잡한 지오메트리 지원을 위한 GML 확장
- 웹기반 환경 : 특별히 OGC표준과 자바, XML을 이용한 상호호환성 기반하의 기술개발에 주력하며 표준에 기반한 컴포넌트 방식(SCOTS,standards-based COTS)의 개발, 기존 시스템과 SCOTS와의 게이트웨이에 대한 연구
- 센서 웹(Sensor Web) : 자동차 네비게이션 시스템, 무선 PDA등에 이용될 수 있도록 실시간 마크업 사용을 통한 실시간 환경 지원
- 공간분석과 의사결정 지원 : 상호 호환가능한 서비스 연계와 복잡한 공간 모델을 위한 메타데이터 확장을 개발할 것을 목표로 데이터 추출, 스키마 추출, 스키마 변환, 결과 비쥬얼라이징 등에 대한 연구

7. 결 론

본 논문에서는 지리정보시스템을 위한 XML의 동향과 특징, 그리고 향후 연구방향을 살펴보았다. XML은 차세대 웹기반 표준 언어로 주목받고 있으며, 불과 몇 년동안 눈부신 발전을 거듭해왔다. 이는 XML의 표준성 및 비정형화된 데이터를 처리하는 능력에 있다고 볼 수 있다.

지리정보시스템은 분산화되고 이질적인 개방형 환경으로 변화되어 가고 있다. 그동안 지리정보시스템은 다양한 데이터 형태를 상호교환하기 위해 많은 노력과 시간을 소비하여 왔다. 이는 그동안 지리정보시스템의 발전이 몇몇 벤더 중심으로 진행되어왔다는 패쇄성에 중요한 요인이 있다.

XML은 이러한 부분에 대한 해결책을 제시하며, 지리정보라는 비정형 데이터를 표현하는데 있어서도 매우 적절한 기술이다. 또한, 웹상의 정형화된 표준이라는 특징을 이용하여 새로운 데이터를 효과적으로 발굴해낼 수 있다는 측면에서 공간 데이터 마이닝과 같은 분야에도 효과적인 기술로 활용될 수 있다.

그러나, 지리정보시스템에서 보다 활성화되어 사용되기 위해서는 많은 문제점이 존재한다. 즉, 위상 관계 등 중요한 정보의 부재와 대용량 데이터를 처리하기에는 비효율적인 형태, 또한 이러한 데이터를 처리하기 위한 질의어 등의 부재, 데이터에 대한 비쥬얼라이징 기술이 성숙되어 있지 못하다는 점들을 들 수 있다.

지리정보시스템에 있어서의 XML기술은 아직까지는 초보 단계로 볼 수 있다. 하지만, 표준화 노력

이 계속적으로 이루어지고 있고 개방화된 환경에 있어서 적절한 기술로 평가받고 있다는 점을 볼 때 향후 비약적 발전을 기대해볼 만하다.

참고문헌

[1] "A Request for Technology in support of an OGC Web Services Initiative", http://ip.opengis.org/ows/index.html

[2] Galdos Systems Inc. "12 Reasons for Using GML to Deliver Geospatial Data over the Web," http://galdosinc.com/,

[3] "GML 3.0 Work Item Plan", http://www.focalpoint.org/gml3/WorkItmeScheduleLiedge.html

[4] "G-XML Project Page," http://gisclh.dpc.or.jp/gxml/contents/index.htm

[5] M.Arikawa, K.Kubota, "A Standard XML Based Protocol for Spatial Data Exchange-Its Capabilities and Real Applications," International workshop on emerging technologies for geo-based applications, May 21-26, 2000. pp37-45

[6] "Namespaces in XML, W3C Recommendation(14 January 1999)", http://www.w3.org/TR/1999/REC-xml-names-1990114

[7] Open GIS Consortium, Inc. "Geography Markup Language Specification (GML) v1.0", 2000

[8] Open GIS Consortium, Inc., "Geography Markup Language Specification (GML) v2.0", 2001, "http://www.opengis.org/techno/specs.html

[9] "OpenGis.net's GML," http://www.open-gis.net/gml

[10] "XML Schema Part 1:Structures W3C Candidate Recommendation(24 October 2000)," http://www.w3.org/TR/xmlschema-1

[11] "XML Schema Part 2: Datatypes, W3c Candidate Recommendation(24 October 200)", http://www.w3.org/TR/xmlschema-2

[12] http://www.landxml.org

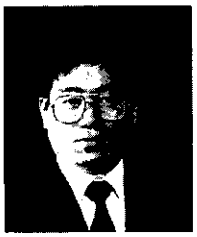
[13] http://www.ned.dem.csiro.au/XMML

[14] http://www.opengis.org/info/press, April 4, 2001, Mar 12, 2001

저자약력

권 준 희

1992년 숙명여자대학교 전산학과 졸업(이학사)
1994년 숙명여자대학교 대학원 전산학과 졸업(이학석사)
1996년-현재 숙명여자대학교 대학원 컴퓨터과학과 박사과정 재학 중
1994년-현재 쌍용정보통신 GIS팀 과장
2000년 전자계산조직응용기술사
관심분야 : 개방형 GIS, 컴포넌트 GIS, XML 데이터베이스, 공간 데이터베이스, 데이터 마이닝, 객체 지향 모델링 및 방법론



윤 용 익

1983년 동국대학교 통계학과 졸업(이학사)
1985년 한국과학기술원 전산학과 졸업(공학석사)
1994년 한국과학기술원 전산학과 졸업(공학박사)
1997년-현재 숙명여자대학교 정보과학부 교수
관심분야 : 정보통신, 멀티미디어통신, 분산시스템, 실시간 처리 시스템, 분산 미들웨어 시스템, 분산 데이터베이스 시스템, 실시간 OS/DBMS