

# GML을 이용한 인터넷 지리정보 서비스 향상방안에 관한 연구 Improving Internet GIS Services Using the GML

장민철\* · 전철민\*\*

Chang, Min Chol · Jun, Chul min

## 要 旨

최근 무선통신기술이 발달하면서 GIS기반의 응용분야도 주목을 받고 있다. 그러나 다양한 기종이나 플랫폼에서 응용 시스템을 구현하는 데에는 한계가 있음이 지적되어 왔다. 한편, eXtensible Markup Language(XML)는 이러한 플랫폼간의 데이터 교환이나 포맷 문제를 해결할 수 있는 기술로 주목받고 있다. Geographic Markup Language(GML)는 XML에 근거한 언어로서 인터넷이나 다양한 모바일 기기상에서 서비스가 가능한 기술로 자주 거론되고 있다. 본 연구에서는 네트워크나 무선 플랫폼을 통해 지리적 데이터를 전달하거나 표현하는데 GML을 사용하는 방안을 제시한다. 현재 사용되고 있는 지리정보 서비스들과 GML을 사용한 방법을 상호 비교하고, 이에 근거하여 GML을 사용한 지리정보 서비스 개선방안에 대해 제안한다.

**핵심용어** : 인터넷 GIS, GML, SVG, XML, 데이터호환

## Abstract

With the emerging mobile technologies, applications based on GIS are also getting more attention recently. But it is known to have some limitations in being served in different types of mobile devices and platforms. While information services are limited by data format and data exchange between platforms, the eXtensible Markup Language(XML) is getting focused as the most promising technology as the solution to such problems. Geographic Markup Language(GML) is based on XML technology and is being frequently mentioned recently as the solution to service problem on different mobile devices as well as the internet. In this study, we suggested the framework to use the GML in order to transfer and represent geographical data through the on-line or wireless platforms. We compared different application types which are currently used with GML and proposed to improve GIS services using the GML in geographical data transfer and description.

**Keywords** : Internet GIS, GML, SVG, XML, Data Exchange

## 1. 서 론

IT의 급격한 발전과 더불어 Geographic Information System(GIS) 서비스 또한 다양하게 전개되고 있다. 이에 기존 지리정보를 공유 및 교환하기 위해 데이터 포맷과 기법들이 개발되고 있지만 그 활용도는 기대에 못 미치고 있다. 또한 데스크탑, PDA, 휴대폰 등의 다양한 플랫폼들이 개발되고 있는데, 이렇게 다양한 플랫폼에 정보가 서비스되기 위해서는 데이터 포맷 및 플랫폼간 호환성의 문제가 야기된다. 이러한 문제점을 극복하기 위해 최근에는 eXtensible Markup Language(XML)기술이 주목을 받고 있다. XML은 웹상에서 구조화된 문서를 활용할 수 있도록 설계된 텍스트 형식의 문서이다. XML

은 World Wide Web Consortium(W3C)에서 공개한 표준이지만 HTML과 달리 문서에 태그를 정의하고 추가할 수 있는 표준 방법을 제공함으로써 모든 문서의 유형을 표현할 수 있다. 또한, XML은 텍스트로 구성되기 때문에 사용자들이 쉽게 읽을 수 있고 다양한 플랫폼 사이에서 파일을 교환할 수 있다. 즉, XML은 문서를 작성하고 소비하는 도구, 플랫폼, 또는 장치에 거의 제한이 없다. 한편 이러한 XML기술을 기반으로 지리정보를 유통하거나 표현하는 방안이 연구개발되고 있는데 이것이 Geographic Markup Language(GML)이다.

GML은 OGC에서 지리정보 저장과 전송을 위해 개발된 표준을 인코딩하는 XML이다. 최근 들어 많은 GIS 소프트웨어 벤더들이 응용 프로그램에 GML을 활용하

2005년 5월 10일 접수, 2005년 6월 12일 채택

\* 주저자, 국립해양조사원 (chamic@uos.ac.kr)

\*\* 서울시립대학교 지적정보학과 교수 (cmjun@uos.ac.kr)

여 데이터를 저장하고 교환하기 위해 사용하고 있으며, 이미 영국의 Ordnance Survey나 미국의 Census Bureau에서는 공간 정보를 GML로 변환하여 저장하고 서비스하는데 사용하고 있다. 그러나, GML은 지리적으로 참조된 정보만을 나타낼 뿐 묘사(portrayal)에 대해서는 규정이 되어 있지 않다. 따라서, 지형정보를 번역하고 표현하기 위해서는 동일한 XML기반의 Scalable Vector Graphics(SVG) 기술을 이용할 수 있으며 이는 W3C에서 권고되고 있는 사항이다. SVG는 이차원의 그래픽을 지원하는 언어로서 사용자가 원하는 방식으로 다이나믹하게 지도를 렌더링할 수도 있고 애니메이션 효과를 낼 수도 있다. 이렇게 GML문서를 표현하거나 사용하기 위해서는 우선 대상이 되는 GML문서를 SVG로 바꾸어야 한다.

본 논문의 목적은, 인터넷 GIS를 통한 지리정보의 표현 및 유통문제를 향상시키기 위해 GML 기반기술을 사용하는 방안을 제시하는데 있다. GML을 SVG로 변환하여 표현하는 방식과 기존에 사용되던 인터넷 GIS프로그램과의 비교를 통하여 GML기술의 효과를 분석하고자 한다.

## 2. 선행연구 및 GML 활용현황

### 2.1 선행연구의 검토

Garmash(2001)의 연구에서는 Wireless Application Protocol(WAP)과 Wireless Markup Language(WML)을 기반으로 한 지리정보 포맷에 대하여 논하고 있다. 물론 WML도 XML기반의 언어로써 Interpretability, 융통성, 그리고 특별한 문제에 대해서는 확장성도 제공하지만 웹에서 사용되는 HTML처럼 이미지가 코드에 삽입이 되어 사용자에게 보여지므로 지리정보를 조작하는데 있어서 구현의 복잡성과 정확도의 문제점을 가지고 있다. 이러한 문제점을 해결하기 위해 위 논문에서 SVG 포맷을 해결책으로 제시하고 있다.

Corcoles(2002)의 연구에서는 GML문서를 저장매체에 저장하는데 있어서 효율적인 방법을 소개하고, 매우 최적화된 관계가 있는 질의처리(Query Process)로 데이터 관리에 대한 실험 결과를 보여 준다. 또한, 김희원(2003)의 연구에서는 서울지역의 주제도를 SVG기술을 이용하여 구현하였으며, 인터넷 GIS에서의 고품질 지리정보 데이터를 활용할 수 있는 좋은 사례를 보여주고 있고, 이혜진(2003)의 연구에서는 사용자 요구에 따른 맵 통합 결과를 제공함으로써 웹 매핑 환경에서의 질을 높일 수 있으며 사용자 응답 시간도 줄일 수 있다는 것을 실험을 통하여 증명하고 있다. 하지만, 대부분의 선행 연구들은 GML문서, SVG문서 각각에 대한 실험과 효율성을 제시하고 있으며 이 둘간의 연동문제에 대해서는 다

루지 않고 있다. 물론 Zhimao Guo(2003)의 연구에서는 GML문서를 SVG로 변환하기 위한 변환 규칙을 세우고, 실험을 통하여 GML문서의 SVG문서로의 변환에 대한 효율성을 제시하고 있다.

### 2.2 GML 활용현황

일반적으로 선진국에서는 GML의 표준화 작업에 대한 참여와 GML활용에 대한 논의가 활발하게 진행중이며 최근에 열린 GML관련 워크샵에서는 ESRI, Microsoft, Oracle, Lockheed-Martin, Laser-Scan, NTT Data, DPC, CSIRO, Galdos System 등의 기업들이 참가하여 GML에 대한 관심을 나타내고 있다.

우리나라에서는 2001년 11월 22일에 스위스 제네바에서 열린 GIS Telecom 2001 conference의 educational workshop에 국내 Korea Telecom이 참여한 것을 계기로 GML에 대한 표준화 작업을 하고 있으며, 특히 LBS분야에서의 GML의 활용이 가속화되고 있다. 그러나 현재까지도 국내에서는 GML활용 사례가 미진한 상태이다.

캐나다에서는 GML이 빠르게 발전되기 시작하였으며, 많은 GML관련 학회가 열리고 있다. 또한 GML명세서 개발에 참여했던 개발자 Ron Lake가 세운 회사인 Galdos라는 업체에 의해 GML산업이 선도되고 있다.

미국에서는 업체뿐만 아니라 공공기관에서도 GML에 대한 활용을 확대해 가고 있다. 이미 ESRI, MapInfo, Oracle, Microsoft 등의 유명 벤더들이 GML에 대한 연구 및 개발을 하고 있으며 공공기관의 경우는, 미국의 통계청이라 할 수 U.S. Census Bureau에서 만든 TIGER/Line 데이터를 GML 형식으로 변경해 가고 있다. TIGER/Line의 경우는 활용도가 매우 높은 데이터이기 때문에 파급효과가 클 것으로 파악된다.

영국에서는 국가 차원에서 GML을 도입하여 Ordnance Survey에서는 DNF(Digital National Framework)의 콤포넌트 호환의 주요 요소로써 GML을 이용하기로 결정하였으며, 이는 LBS 기반의 응용에 대해 지원하려는 목적이 크다. OS에서는 MasterMap이라는 GML포맷을 만들어 서비스 하고 있다. 2000년 5월부터 호환을 위한 수치지도의 표준으로써 GML에 대한 사용을 고려해 왔으며 현재는 Intelligent Map Data라는 이름으로 GML상호 운용의 표준으로 이용하고 있다.

일본의 경우에는 자국의 GIS산업을 부흥시키기 위해 독자적으로 개발한 언어인 G-XML을 활용하여 이기종 대용량 GIS간 상호운용 및 PDA, 이동통신기기 지원을 목표로 하여 현재 2.0 규격까지 개발되었으며 3.0을 추진 중에 있다. 일본은 이를 국제표준화기구인 ISO에 표준안으로 제안할 계획이다. 그러나 업계 전문가들은 일본의

독자적 표준안보다는 산업계의 폭넓은 지지를 받고 있는 GML이 국제표준으로 정착될 것으로 보고 있다. OGC에서는 일본의 65개 업체들이 참여하고 있는 G-XML을 GML에 수렴하는 작업을 진행하여 2001년 전통적인 GIS분야 뿐 아니라 LBS, 사고관리 분야 등에서 두 표준안을 통합하는 것에 합의 하였으며, GML 3.0에 이 내용을 포함하여 현재는 호환이 되고 있다. 또한 이동통신 회사인 NTT DoCoMo의 i-mode에서는 SVG/SMIL 브라우저를 삽입하여 멀티미디어에 본격적으로 대비를 하고 있다. 그 예로 무선 인터넷(i-mode)을 통하여 교토역 주변의 관광지를 검색하면 SVG의 지도상에 목적지가 표시되는 등의 지리정보 서비스가 이루어지고 있다.

### 3. GML/SVG 기반 시스템 특성 및 기존 기술과의 비교

#### 3.1 GML과 SVG

GML은 지리학적인 요소의 공간적/비공간적 속성을 모두 포함하는 지리학적인 정보의 유통과 저장을 위한 XML 인코딩이다. XML 형식은 간단히 말해 DB의 내용을 플랫폼과 애플리케이션에 무관하게 전송□표현 할 수 있는 문서 규약으로 GML도 이러한 특징을 모두 가지고 있으며 특히 지리적 형태를 OGC표준화 규약에 따라 작성하고 전송한다면 이를 필요로 하는 사용자는 공간 데이터 내용을 그대로 표현하고 조작 할 수 있게 된다. 이는 래스터의 이미지를 표현 하는 것과는 질적으로 다르다. HTML 문에서는 문자와 이미지를 표현하므로 사용자가 보는 지도는 이미지 일 수밖에 없다. 그러나, GML로 전송된 데이터는 쉽게 말해 공간 DB의 내용을 텍스트로 변환한 형태이므로 DB의 속성 및 내용이 그대로 화면에 표현될 수 있는 것이다.

GML은 공간개체(spatial features), 공간 개체의 속성(feature properties), 공간 개체의 형상(feature geometries), 표준 데이터 모델(simple feature data model)에 기반한 요소들의 위치를 인코딩 하는 표준 방법을 제공한다. 또한 서로 다른 조직에서 데이터 이동과 교환을 할 때 데이터를 받는 부서에서는 당사자의 GIS 프로그램이 다룰 수 있는 데이터 형식으로 변환해야 한다. 이때 정보들은 데이터 전송과 변환 과정에서 없어지거나 왜곡이 된다. 한편 GML로 인코딩된 지리공간 데이터가 전송될 때는 Feature, Geometry 그리고 데이터의 공간 참조 체계를 포함한 모든 마크업 요소가 수신측에 전송된다. 이는 표준 마크업 요소에 기반하였기 때문에 받는 부서에서는 각 데이터 객체가 무엇을 의미하는지 어떻게 추출 되는지 알고 있으며 데이터 전송과 변환 과정에서 왜곡

되는 부분이 없어지게 된다.

또한 GML은 서로 다른 개체와 개체요소들 사이에 다방향적인 연결을 개발하기 위한 방법(XLink, XPointer사용)을 제공한다. 이는 지역적으로 만들어지고 유지되는 데이터들을 이용하는 유용한 방법으로서, 광범위하게 퍼져 있는 데이터를 쉽게 사용할 수 있다는 장점이 있다. 본 연구에서는 이러한 장점을 GIS 서비스 개선 방안에 반영하여 설계를 하였다. <그림 1>은 GML 문서의 예이다.

XML 그래픽 표준인 SVG는 XML의 개방성, 상호 운용성 등의 장점을 모두 수용하였고 GML, SensorML, MathML 등 다른 XML 언어들과 결합하여 다양한 응용 개발이 가능하고 HTML과 스크립트 등 기존의 웹 개발 툴과 상호운용이 가능하므로 웹 기반의 지리정보, 교육, 광고, 전자상거래, 프로세스 컨트롤 등 그래픽이 많이 사용되는 분야에 응용된다. SVG 또한 텍스트로 기술되기 때문에 그래픽에 대한 검색이 편리하고, 애플리케이션들이 SVG문서를 쉽게 사용할 수 있으며, XML과 SVG의 DOM(Document Object Model)의 인터페이스를 통하여 선, 폴리곤, 텍스트, 이미지 등의 모든 그래픽 요소에 쉽게 접근할 수 있으므로 데이터베이스와 연동하여 웹 그래픽 문서를 동적으로 생성할 수 있다. 이렇듯 SVG는 사용자의 요구에 따라 다양하게 구현될 수 있으므로 다양한 애

```

1: <?XML Version="1.0"?> <GML> <Layer>
2: <!-- village -->
3: <LayerMember>
4: <feature><point><coord><x>1300</x><y>3000</y></coord></point></feature>
5: </LayerMember>.....
6: <!-- road -->
7: <LayerMember>
8: <road> <streetName>Broadway</streetName><speedLimit>45</speedLimit>
9: <numberLanes>4</numberLanes>
10: <centerLineOf><LineString srsName="EPSG:4326">
11: <gml:coordinates>5.5,80.0 60.5,130.5</gml:coordinates>
12: </LineString></centerLineOf></road> ....
13: </LayerMember>
14: <!-- boundary -->
15: <LayerMember><Feature><name>DongDaeMun</name>
16: <Polygon><outerBoundaryIs>
17: <LinearRing><coord><x>-555.55</x><y>86.22</y></coord>... </LinearRing>
18: </outerBoundaryIs><innerBoundaryIs><LinearRing>
19: <coord><x>-555.96</x><y>48.44</y></coord>
...</LinearRing></innerBoundaryIs>
20: </innerBoundaryIs></LinearRing>
21: <coord><x>-555.73</x><y>48.27</y></coord> ...
22: </LinearRing></innerBoundaryIs>
23: </Polygon></Feature></LayerMember> ...</Layer> </GML>
    
```

그림 1. GML 문서의 예

플리케이션에서 동적으로 연동되어 강력한 기능을 발휘 하며 서버용 도구를 이용한 XML 생성 기술을 적용할 수 있어 웹에서 동적인 SVG 생성을 지원할 수 있다. <그림 2>는 <그림 1>을 SVG 문서로 변환한 결과이다.

### 3.2 GML과 기존기술의 비교

HTML과 비교하여 볼 때, GML은 몇가지 특징을 가지고 있는데, 그들은 확장 XML도 가지고 있는, 계층형 구조, 데이터 확인(validation), 구조와 표현의 분리, 내용과 관계의 분리 등을 포함한다 (Lake, 1999, 2000, 2001).

공간데이터를 인코딩하는 방법에는 CCOGIF(Canadian Council on Geomatics Interchange Format), MACDIF (Map And Chart Data Interchange Format), SAIF(State Access Inspection Fitting), DLG, SDTS를 포함하여 많은 종류가 있다(Lake, 1999). 그런데 “왜 우리는 다른 기준(표준)이 필요한가?”에 대한 의문이 생긴다. GML은 XML기술을 기반으로 하고 있어, 몇 가지의 특유의 특징과 이점을 가지는데, 그것은 문자 기반이고, 개방형이며, 소유권이 없으므로 요소와 스키마(schema)들의 용어에 있어 확장 가능성이 있다는 것 등이고 이러한 것들은 큰 유연성을 가져다준다. 또한 엄격한 내부 구조와 구성(construct) 규칙을 갖고 있어 GML문서의 확인을 위해 스키마(schema)들을 제공하고 상호 운용 가능성을 보증한다. 구조와 표현의 분리는 내용과 관계를 분리하는 것과 마찬가지로이다(Lake, 1999).

즉, GML은 인터넷을 이용하여 지리정보를 효율적으로 유통하기 위하여 개발되어 졌다고 볼 수 있다. 그래서 본 연구에서는 인터넷을 이용한 기존기술과 GML기술의 특성을 비교하였다.

일반적으로 인터넷을 통한 지리정보의 사용을 보면, 도형을 웹에서 읽을 수 있는 형태의 이미지 파일 포맷(Tiff, JPG, GIF 등)으로 변환시켜서 HTML 코드에 삽입하는 형태로 사용자들에게 보여 지고 있다. 그러나 이러한 이미지 포맷상의 접근 방법은 지리정보를 조작하는데 있어서

구현에 복잡성과 정확도에 문제점을 야기시킨다. GML은 기본적으로 XML에서 파생된 것이므로 웹상에서 자료의 교환에 대해 많은 장점을 가지고 있으며 XML자체의 성격을 그대로 지니고 있어 XML 엔진이나 프로세서를 통하여 지리정보에 대한 다양한 조작이 가능할 뿐만 아니라 값 비싼 GIS소프트웨어의 구입 및 설치가 필요 없이 단지 웹 브라우저를 통하여 표현이 가능하다. 웹 브라우저는 거의 모든 사용자들이 가지고 있으며 별도의 복잡한 Web GIS 애플리케이션 설치나 아키텍처의 구현이 필요 없이 GIS 소프트웨어의 역할을 할 수 있다. 이러한 GML과 기존기술의 웹을 통한 지리정보의 활용적 측면에서 가지는 장·단점들의 비교는 다음 <표 1>과 같다.

## 4. GML기반 지리정보 시스템 구현과 분석

이전까지는 GML과 SVG의 구현사례와 본 연구의 시스템을 어떻게 구현 할 것인가에 대하여 알아보았다. 또한 GML기반 지리정보 서비스를 제공하기 위한 시스템 구현에 필요한 기술에 중점을 두어 논의가 되었다. 이번 장에서는 크게 시스템을 구현하기 위한 시스템 분석과 실제 구현의 내용으로 나뉘었다. 또한 본 연구를 통해 구현된 시스템을 기존의 시스템과의 비교를 통하여 본 연구에서 제안하는 방안의 우수성을 알아보았다. 구현에 대한 사항은 간단히 말해 기존의 지리정보를 GML데이터로 변환하여 XSLT를 이용 GML데이터를 SVG로 변형시켜 웹 브라우저를 통해 사용자에게 지리정보를 제공하는 시스템 구현에 대한 것이다. 또한 기존 시스템과의 비교를 위해 같은 지리정보 데이터를 이용, Java Applet으로 구현한 시스템과의 비교 분석을 하였다. <그림 3>은 본 연구에서 구현한 시스템에 대한 개략적인 흐름도이다.

### 4.1 GML기반의 시스템 구현

지리정보 서비스를 하기 위해서는 해당지역의 수치지도가 필요하다. 아울러 기존의 지리정보시스템과 본 연구에서 제안하는 시스템을 비교하기 위해서는 동일한 공간데이터를 가지고 서비스하는 시스템이 필요하였다. 본 연구에서는 이를 충족하기 위해 교통 경로 제공시스템을 구축하고, 이를 기존에 빈번하게 사용되는 방법인 Java Applet기법과 본 연구에서 제안하는 GML기반 기법을 통해 상호 비교하였다. 본 연구에서는 주로 지도 표현이나 속도에 초점을 맞추어 비교분석 하였다. 동일한 자료를 이용하여 GML데이터를 SVG로 표현하여 지리 정보 서비스가 가능한 시스템을 구현함으로써 구체적으로 어떠한 장점을 가지게 되는지 비교하여 살펴 볼 수 있으며 구현 시 애로사항과 기타 필요로 하는 기술적 요

```
1 : <?xml version="1.0" encoding="UTF-8"?>
2 : <svg style="height:100px;width:100px;"><g>
3 : <circle cx="1300" cy="3000" r="0.20" transform="matrix(6,0,0,-6,800,400)"
   style="fill:rgb(135,10,174)" /> ...
4 : <polyline points="5.5,80.0,60.5,130.5" transform="matrix(6,0,0,-6,800,400)"
   style="fill:none;stroke:yellow;stroke-width="0.15"/> ...
5 : <path d="M-555.55,66.22 L-555.46,66.22 ... -555.40,66.22 Z"
   transform="matrix(6,0,0,-6,800,400)" style="fill:blue;stroke:black;"
   stroke-width="0.1"/> ...
7 : </g> </svg>
```

그림 2. SVG 문서의 예

표 1. GML 기반 기술과 기존기술의 비교

구분	기존 기술	GML 기반 기술
플랫폼	도면의 확장자명에 따라 지정된 플랫폼을 통해 표현	지정된 플랫폼이 아닌 웹 브라우저를 통한 표현이 가능 (Platform Free)
표현의 정밀도	데이터가 Raster로 이루어지기 때문에 도형에 대한 속성 정보의 조회나 Pixel의 수 제한이 있음	데이터가 Vector로 표현되기 때문에 도형에 대한 속성 정보의 조회가 제약 없이 이루어지며, Pixel의 수 제한이 없어 고해상도의 표현이 가능
데이터베이스 구조	사용자의 데이터 조작 권한을 제한 할 수 없고, 데이터의 조작이 이루어지면 DB의 데이터 역시 수정됨	데이터베이스 구조를 그대로 구현하며, 사용자의 데이터 조작 권한을 제한 할 수 있음
이용자 형태	전문가 그룹을 대상으로 함	일반 사용자들도 쉽게 접근할 수 있음
사용자의 데이터 이용	서버에서 제공한 데이터만을 이용할 수 있음	다양한 조작이 가능하여 사용자는 자신의 요구에 맞게 데이터 표현을 할 수 있음
기호의 표현	이미지 파일 등을 통해 규정된 기호만을 표현	실 세계의 지형지물이나 객체들을 코드를 통해 나타내어 특정 개체에 대하여 ekfs 색으로 표현하거나 텍스트를 입력한다면가 추가적으로 도형의 삽입이 가능
데이터 연결	좌표 시스템이 같은 지리정보만을 연결할 수 있음	좌표 시스템이 다른 지리정보를 연결하고 각각 다른 지역의 속성자료를 연결할 수 있음
접속	이벤트가 발생할 때마다 접속하므로 자주 발생함	최소화

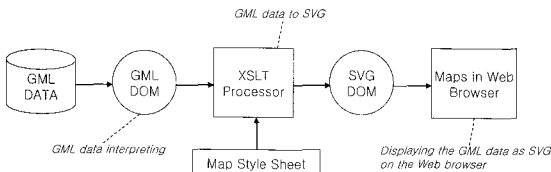


그림 3. GML문서를 이용 지리정보 표현 과정

인에 대한 파악이 가능하다. GML과 SVG기반의 지리정보 서비스를 구현함에 있어 전체적인 작업 절차를 정리하면 <그림 4>와 같다.

① 지리정보 데이터를 GML로 변환

현재 우리나라는 지리정보가 GML로 표준화되는 단계이므로 “2001년 전국 교통DB 구축사업”에 의해 구축되어진 교통데이터(서울·경기 지역 중심)를 ArcGIS의 “ArcGIS OGC Interoperability Add-on”을 이용 GML데이터로 변환하였다. <그림 5>는 앞에서 언급한 방법에 의하여 변환하는 과정의 화면이다. 그러나 본 Add-on을 사용하면 ESRI사의 독자적인 스키마를 참조하여 GML로 문서화되므로 이를 다시 XML편집기인 “XMLSpy 2003 Enterprise Edition”을 사용 OGC표준 포맷에 맞추어 편집하였다. <그림 6>은 “ArcGIS OGC Interoperability Add-on”에 의해 변환된 GML을 “XMLSpy 2003 Enterprise Edition”을 사용하여 편집하는 화면이다.

- 서울경기 지역의 지리 Data를 GML문서로 변환
  - “2001년 전국교통 DB”의 서울경기 지역의 Data 기반
  - ArcGIS OGC Interoperability Add-on을 이용 웨이프라이를 GML문서로 변환

- OGC 스펙에 맞게 GML 편집
  - “XMLSpy Enterprise Edition 2004”를 이용 GML편집

- 시스템 설계
  - 시스템 전체 흐름 설계
  - 시스템 구현될 모듈설계

- 시스템 구현
  - GML을 SVG문서로 변환하기 위한 XSLT 제작
  - 인터넷 지리정보 서비스를 위한 JSP프로그램 제작
  - 인터랙티브한 서비스를 위한 Java Script제작

- 지속적인 피드백을 통한 부가적인 기능 추가

그림 4. GML과 SVG기반 지리정보 서비스 구현의 작업 절차

“XMLSpy 2003 Enterprise Edition”을 사용하여 편집하는 화면이다.

② 시스템 설계

<그림 7>은 시스템 설계를 위해 Web Feature Server (이하 WFS)를 상세하게 표현한 시스템 구성도이다.

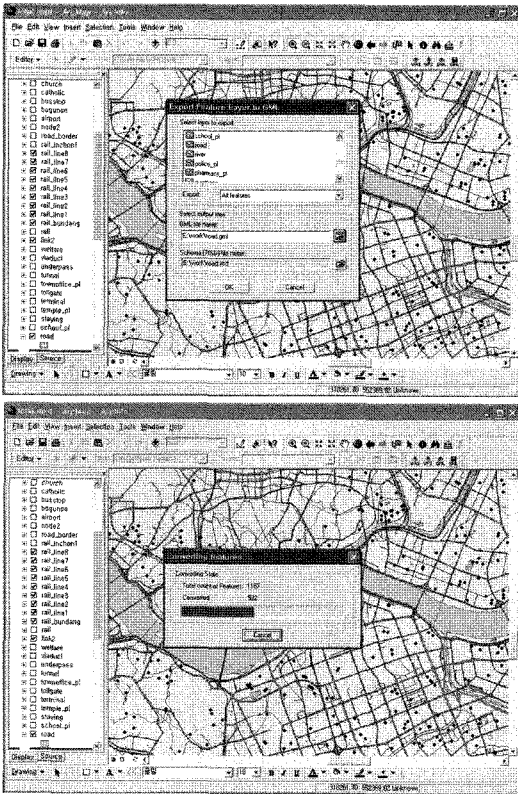


그림 5. ArcGIS의 OGC Interoperability Add-on을 이용 GML문서 생성

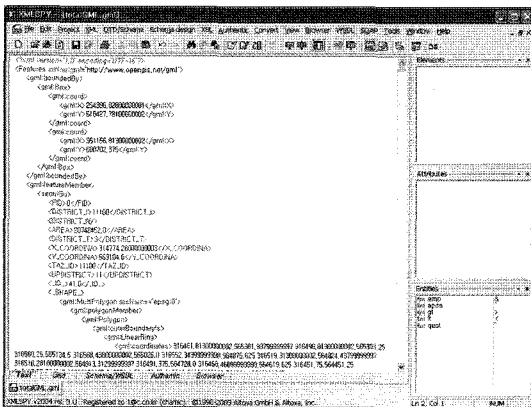


그림 6. XMLSpy 2003 Enterprise Edition을 이용 GML편집

WFS는 데이터베이스로부터 구성된 WKB\*, GML 데이터 또는 다른 지역에 퍼져있는 GML 데이터를 Styling

\* WKB(Well-Known Binary)는 OGC에서 규정한 Geometry 데이터를 전송하는 단위로써 연속된 바이너리형식의 Geometry 데이터를 의미함.

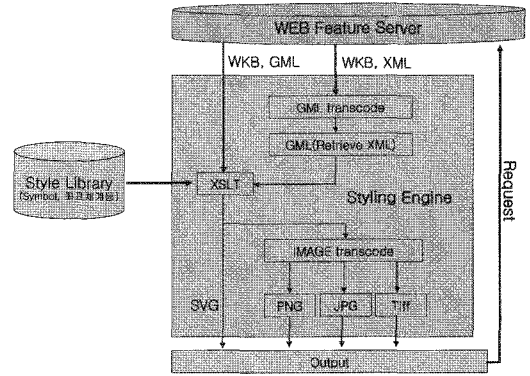


그림 7. 시스템 구성도

Engine 부분으로 전달하게 된다. Styling Engine은 외부에 있는 GML 데이터를 받아 미리 정의된 규칙대로 일관성 있게 지도로 표현하기 위하여 수행하는 모듈/프로세스의 집합이다. GML은 데이터 표현 방법을 기술 할 수 있는데 이는 Styling Engine에서 해석하여 표현된다. Styling Engine은 GML의 해당속성에 대한 표현을 정의하고 있는 부분을 라이브러리에서 가져오고 GML문서에 담겨진 속성이 정의된 규칙대로 심볼화하여 보여준다. Styling Engine에서 어떠한 데이터들이 들어와도 이미 정의된 같은 스타일로 변환하는 작업을 거치게 된다. 이로써 GML 문서에 표기된 내용대로 일관성 있게 표기된다. 이렇게 변환된 지도는 벡터 맵의 경우 바로 서비스를 하게 되며 이미지 맵으로 전송을 할 경우는 PNG, BMP, JPEG, GIF 등의 형식으로 변환하여 서비스된다. 이러한 사항을 토대로 구현하게 될 시스템을 모듈(module)별로 정의하면 다음 <그림 8>과 같다.

<그림 8>의 각 모듈의 내용은 다음과 같다.

- layer : Layer를 시각화하거나 스타일을 조작하는데 사용
- data : Layer의 속성 조작을 하는데 사용

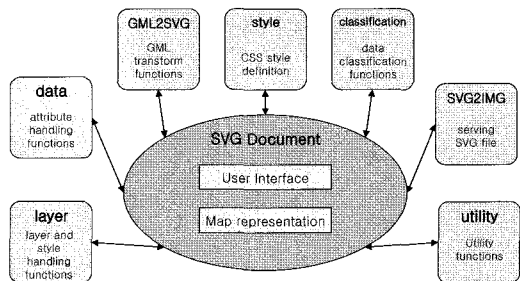


그림 8. 시스템 아키텍처

(자료참조 : <http://www.svgopen.org/2003/papers/SvgExplorerOfGmlData>)

- GML2SVG : GML을 SVG로 변형시키기 위해 사용됨. 즉 SVG원소를 새로 만들기 위해 GML문서 구조를 문법적으로 분석할 때 이 기능이 활성화됨. 이때 GML의 그래픽관련 태그만 사용
- classification : 지리정보의 속성을 분석하고 classification을 수행할 때 사용
- SVG2IMG : 이 모듈은 표현된 지도를 이미지 파일(PNG, JPEG, GIF 등)로 변환하여 저장할 때 사용
- utility : 표현된 지도상에 특정경로를 표시하거나 애니메이션 기능을 적용 할 때 사용
- style : 사용자 인터페이스의 그래픽의 원소에 적용된 유형의 정의를 포함하는 CSS 파일

③ GML을 SVG로 변환하기 위한 XSLT

XSL(eXtensible Stylesheet Language)은 XML문서의 포맷을 지정하는데 사용되는 언어로서 XML을 HTML로 변환하여 쓰기위해 사용되었으나 현재는 W3C에서 권고된 수많은 형식의 XML을 변환하는데 사용이 된다. W3C의 규정에 따르면, XSL은 XML문서의 XSL Transformation과 XML문서의 부분에 접근하기 위한 XML Path Language, XSL Formatting Objects의 세 부분으로 구성된다. 본 논문에서는 GML을 SVG로 바꾸는 새로운 방안에 대한 내용에 중점을 두고 XPath의 개념이나 XSLT에 대한 세부기술에 대한 내용은 기술하지 않았다. XSLT프로그램은 어떤 패턴과 템플릿으로 구성되는지 알기 위한 각 템플릿 규칙의 집합이다. 변환은 다음과 같이 진행된다. XSLT는 현재의 노드에서 참조하는 노드 목록으로부터 첫 번째 노드를 제거하고, 패턴을 노드에 적용한다. 이러한 방법이 성공하였다면 XSLT는 상응하는 템플릿을 실행한다. 이렇게 하여 XML의 결과를 보여 주고, 목록에서 현재의 노드 앞에 다음 노드를 부가시켜 동일한 방법으로 변환한다. 따라서 XSLT 변환은 입력된 문서에 따라 이끌리는 재귀적 과정이다. XSLT는 형태가 well-formed한 XML 문서이어야 하고 허용되는 구문과 어휘를 기술하는 XSLT 규정에 따라야 한다. 본 연구의 XSLT는 GML문서를 SVG로 바꾸기 위한 언어이고, 입력문서는 GML문서이며 출력문서는 SVG라고 가정한다. <그림 9>는 위 과정을 통해 GML문서를 SVG문서로 변환하는 XSLT문서의 일부분이다.

④ 웹 서비스를 위한 JSP 웹 프로그램

GML과 SVG기반의 지리정보 서비스는 클라이언트 기반의 애플리케이션이다. 그러므로 XSLT에 의하여 GML 문서가 동적으로 변환이 되어야 한다. 이는 사용자가 원

```

1 : <?xml version="1.0" encoding="UTF-8"?>
2 : <xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:gml="http://www.openengis.net/gml"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xiink="http://www.w3.org/2000/xiink/namespace/"
  xmlns:svg="http://www.w3.org/2000/svg" >
3 : <xsl:output encoding="UTF-8" standalone="no"
  media-type="image/svg+xml" indent="yes"/>
4 : <xsl:text disable-output-escaping="yes">
  &lt;!DOCTYPE svg PUBLIC "-//W3C//DTD
  SVG 1.1//EN" "http://www.w3.org/Graphics/SVG/1.1/
  DTD/svg11-flat-20030114.dtd"&gt;
5 : </xsl:text>
6 : <xsl:template match="Features">
7 : <xsl:processing-instruction name="xsl-stylesheet">
8 : </xsl:processing-instruction>
9 : <xsl:element name="svg">
10 : <xsl:attribute name="id">ZolPaperMap</xsl:attribute>
11 : <xsl:attribute name="width">100%</xsl:attribute>
12 : <xsl:attribute name="height">100%</xsl:attribute>
13 : <xsl:attribute name="viewBox">
  <xsl:value-of select="$XMin"/><xsl:text>
14 : </xsl:text><xsl:value-of select="- $YMin - $YDiff"/>
  <xsl:text>
15 : </xsl:text><xsl:value-of select="$XDiff"/>
  <xsl:text> </xsl:text>
16 : <xsl:value-of select="$YDiff"/></xsl:attribute>
17 : <xsl:if test="descendant::gml:featureMember">
18 : <xsl:copy-of select="document('Symbol.xml')/defs"/>
19 : <!-- seoulGu -->
20 : <xsl:element name="g">
21 : <xsl:attribute name="id"><xsl:text>seoulGu
  </xsl:text></xsl:attribute>
22 : <xsl:attribute name="visibility">visible</xsl:attribute>
23 : <xsl:apply-templates select="descendant::seoulGu"/>
24 : </xsl:element>
25 : <!-- road --> .....
26 : <!-- river --> .....
  .....
27 : <xsl:template match="seoulGu">
28 : <xsl:attribute name="transform">
29 : <xsl:text>matrix(1 0 0 -1 0 0)</xsl:text>
  </xsl:attribute>
20 : <xsl:apply-templates select="descendant::_SHAPE_"/>
21 : </xsl:template>
22 : </xsl:element></xsl:template></xsl:stylesheet>
  
```

그림 9. XSLT(GML to SVG) 문서

하는 GML데이터를 이용 SVG로 변환이 되어야 하므로 JSP를 이용하여 이를 구현하였다. 또한 지도와의 인터랙티브한 작용이 발생하므로 이에 대한 사항은 JavaScript를 이용하여 구현하였다. 이러한 사항은 물론 프로그램에 대한 지식을 가지고 있어야 하지만 기존의 Applet이나 ActiveX와 같은 프로그램보다는 구현하기 간단하다. JSP는 Web 페이지를 동적으로 처리하도록 하는 기술 중의 하

나로 서블릿 프로그램의 기능을 HTML 파일 내에 스크립트 형식으로 구현한 것으로 JSP엔진을 내장한 웹서버에서 처리가 가능하다. 본 연구에서는 웹서버를 IIS와 Jakarta-tomcat4.1을 이용하여 구현하였다. <그림 10>은 웹 서비스를 하기 위하여 구현된 JSP 소스의 일부분이다.

<그림 10>의 소스를 간단히 살펴보면 1에서 5라인까지는 JSP문서에 필요한 선언과 필요한 라이브러리를 импорт(import)해오는 부분이다. 6라인부터는 XSLT와 입력된 GML데이터를 이용 SVG문서로 바꾸어 사용자에게 전달되는 부분으로써 6에서 8라인까지는 GML데이터와 SVG문서 변환에 필요한 XSLT문서의 위치를 파악하는 부분이다. 9에서 12라인은 SVG문서를 변환하기 위한 인스턴스를 생성하는 부분과 스트림을 통해 GML데이터와 XSLT를 읽어오는 부분이다. 마지막으로 13과 14라인은 읽어들인 GML데이터를 SVG문서로 변환하는 부분이다. <그림 11>은 인터랙티브한 기능을 구현하기 위한 JavaScript의 일부분이다.

**4.2 구현된 시스템과 Applet기반 시스템 비교**

<그림 12>는 본 연구에서 구현된 시스템의 결과로서 앞절에 설계되어 구현된 각 기능들의 설명을 보여 주고 있다. 본 연구에서 구현된 시스템을 이용하기 위해서는 별도의 전문 GIS소프트웨어가 없이도 고품질의 벡터 그래픽의 대화식 지리정보 서비스가 가능하다. GML과 SVG기반의 지리정보 서비스는 간단한 구현으로 GIS소프트웨어의 기본 기능과 효과, 사용자 인터페이스를 제공한다. 구현 결과는 인터넷이 연결되어 있는 환경이라면 어디에서든지 확인 가능하다. 단, 현재 인터넷 익스플로러5.5

```

1 : var xOriginCorner = 254396.82800000001
2 : var yOriginCorner = -600702.375
3 : var allWidth = 96759.98500000001
4 : var allHeight = 82274.59399999998
5 : function initMap() {
6 :   svgdoc = mainMap.getSVGDocument();
7 :   .....
8 : }
9 : function toggleSVGElements(id,checkVal) { ..... } //CheckBox 컨트롤
10: function ctrlScale(scal) { ..... } // Scale 컨트롤
    //event 컨트롤
11: function zoomIn() { ..... }
12: function beginPan(evt) { ..... }
13: function doPan(evt) { ..... }
14: function endPan(evt) { ..... }
    
```

그림 11. Java Script 구현 예시

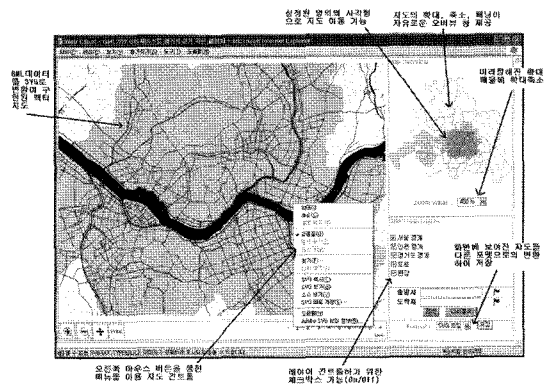


그림 12. GML/SVG 기반 시스템 구현

```

1 : <%@ page language="java" contentType="text/html; charset=euc-kr" %>
2 : <%@ page import="java.io.*"%>
3 : <%@ page import="javax.xml.transform.*"%>
4 : <%@ page import="javax.xml.transform.stream.*"%>
5 : <%@ page import="java.util.zip.*"%>
6 : <% String FS = System.getProperty("file.separator");
7 :   String gmlFile = request.getParameter("GML") + ".gml";
8 :   String xslFile = request.getParameter("XSL") + ".xsl";
9 :   TransformerFactory tFactory = TransformerFactory.newInstance();
10:   Transformer transformer
    = tFactory.newTransformer(new StreamSource(xslFile));
11:   StreamSource sSrc = new StreamSource(gmlFile);
12:   StreamResult sRst = new StreamResult(rstFile);
13:   transformer.setParameter("svg", svgFile);
14:   transformer.transform(sSrc, sRst); %>
    
```

그림 10. JSP프로그램 예시

이상의 인터넷 브라우저와 Adobe SVG Viewer 플러그인이 설치된 단말기에서 확인가능하다. GML데이터는 XSLT를 통해 SVG문서로 변환되어 웹 브라우저상에서 보여진다. SVG문서의 심볼들과 패턴은 미리 정의된 라이브러리에 의해 변환되지만 사용자가 쉽게 변경하거나 새로이 만들 수 있다.

기존 시스템과의 비교는 앞 단락에서 시스템을 구현하기 위한 요구분석과 <표 1>을 토대로 다음과 같은 사항을 비교하였다. 사용자 유저인터페이스(User Interface : 이하 UI)의 측면과 동일한 지역과 레이어를 표현할 때 평균 응답시간, 지도의 기능(확대, 축소, 이동 등)을 수행할 때의 평균시간을 비교하였다. 또한 래스터와 본 연구에서 제안한 기술인 벡터 기반의 기술과의 비교를 통한 표현의 정밀도, 클라이언트로 데이터가 전송이 될 때 구조나 표현에 대한 사항을 비교하였다.



기존 기술(본 연구에선 Applet과 DCOM을 이용한 기술)과의 비교를 보다 더 정확히 하기 위해 동일한 곳에 웹 서버를 설치하여 테스트 하였고 클라이언트 또한 같은 개인 컴퓨터에서 비교하였다.

① 사용 UI의 비교

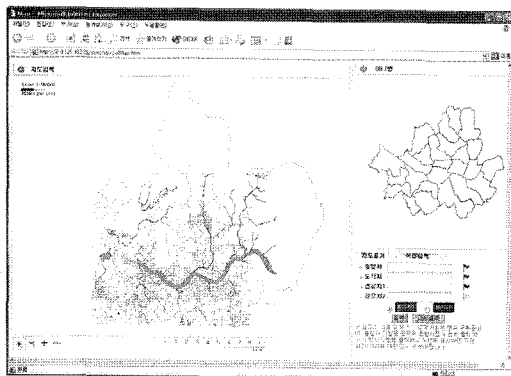
UI는 프로그램에서 데이터입력 등의 동작을 제어하기 위해 명령어를 입력하거나 출력을 표현하는 등 사용자와의 대화를 위한 화면을 말한다. 본 연구에서는 인터넷상에서 지리정보를 서비스 하므로 웹 브라우저 상에서 지도를 조작하는데 필요한 기본적인 기능들을 구현하였다. <그림 13>은 기존 Java Applet을 이용하여 구현한 시스템(a)과 본 연구에서 구현한 시스템(b)이다.

<그림 13>은 같은 환경에서 비교를 하기 위해 화면배치를 최대한 비슷하게 구성 하였으므로 외관상 기능이 거의 같아 보인다. 그러나 세부항목을 비교하여 보면 <표 2>와 같이 분석된다. 비교 항목은 레이어 컨트롤과 지리정보 서비스에 있어서 기본적인 확대, 축소, 이동에 관한 사항과 다른 포맷으로의 저장 등의 항목으로 정하여 비교하였다. <표 2>에서 요약된 바와 같이 본 연구에서 제안한 기술이 대체적으로 구현하기 쉬울 뿐 아니라

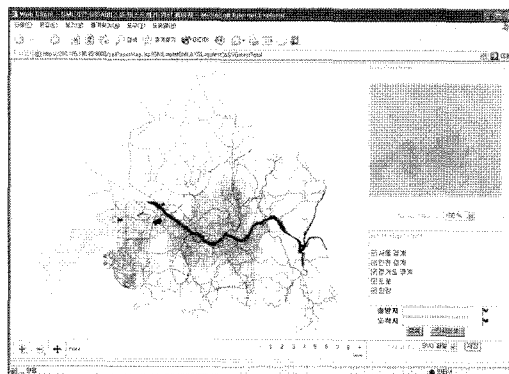
기능면에서도 앞서는 것을 알 수 있다. <그림 13>의 (a)의 경우 Applet과 DCOM을 이용하였고 미니맵은 플래시로 제작이 되어있다. 이렇기 때문에 구현을 위해서는 전문적인 프로그램인 C++, Java 등의 기술을 알아야하고 인터랙티브한 기능을 구현하기 위한 여러 가지의 프로그램 알고리즘이 필요하다. 이 밖에, 지도를 다른 포맷으로 저장하는 기능의 경우에는 일반적으로 그림 파일로만 저장이 가능하였으나 GML과 SVG기반의 지리정보 서비스는 XML문서를 다운받을 수 있도록 스펙자체에서 기본으로 제공하고 있다. 즉 SVG소스를 사용자 컴퓨터에 다운을 받아 키스트마이징 할 수 있다.

② 지도의 로딩속도

지도 로딩 속도의 비교는 서버에 서울·경기 지역의 영역을 요청하여 클라이언트에 지도가 표현되는 시간을 3회 반복하여 측정하였다. 또한 레이어별로 하나씩만 표현할 때의 응답시간을 측정하여 본 연구에서 제안하는 시스템의 만족도를 체크하였다. <표 3>은 로딩속도를 측정한 결과이며 <그림 14>는 응답시간 비교 화면이다. 시스템의 응답시간은 동일한 영역의 지리정보를 불러올 때 레이어별로 표현되는 시간을 측정하였다.



(a) Applet을 이용한 서비스



(b) GML/SVG기반 서비스

그림 13. 기존 기술과 GML/SVG기반 서비스

표 2. 사용자 운영의 편리성과 구현 용이성에 대한 비교

항 목	Applet기반 기술	GML/SVG기반 기술
지도상 확대/축소/이동	어려움	쉬움
미니맵을 이용한 지도 확대/축소/이동	어려움	쉬움
지도를 다른 포맷으로 저장	어려움	쉬움
Layer의 컨트롤	어려움	쉬움
기본기능 구현 정도	대체로 어려움	쉬움

표 3. 응답 로딩 시간의 비교 (단위 : 초 sec)

레이어 명	Applet기반 기술			GML/SVG기반 기술		
	회수	응답시간	평균응답시간	응답시간	회수	
행정경계	1	47.88	19.7	4.20	1	
	2	5.72		4.18	2	
	3	5.50		4.12	3	
도로	1	50.82	24.27	10.01	1	
	2	11.11		9.90	2	
	3	10.89		9.68	3	
한강	1	46.60	24.66	14.71	1	
	2	13.63		14.63	2	
	3	13.74		14.63	3	

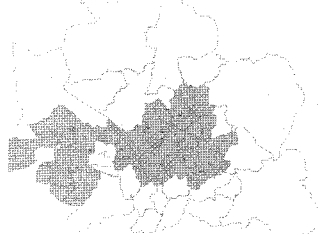
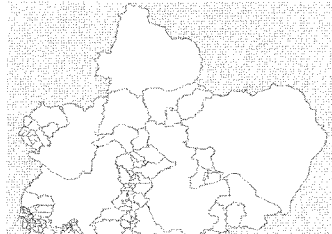
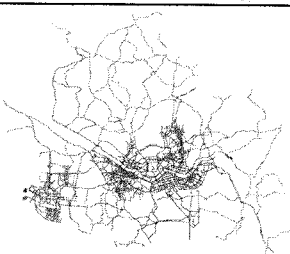
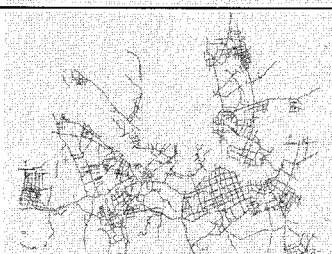
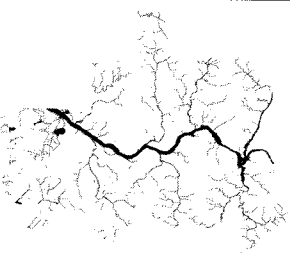
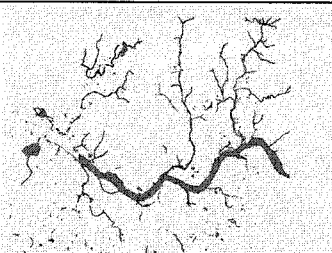
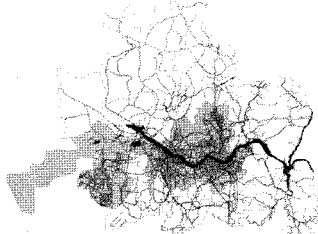
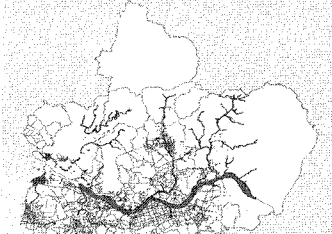
GML/SVG기반 서비스 화면	레이어 명	Applet기반 서비스 화면
	행정구역	
	도로	
	강	
	전체화면	

그림 14. 응답시간 비교화면

네트워크의 연결 속도는 100Mbps로 서버와 클라이언트로 연결되어있고 시간측정은 클라이언트의 브라우저에 표현되는 런타임(Run time)을 체크하여 소수점 둘째자리까지 3회 측정하였다. 또한 브라우저에 캐쉬(Cash)가 남아있으므로 이를 제거하기 위해 매회 측정시 브라우저를 새로 열어서 시간을 측정하였다.

측정결과에서 보듯이 Applet기반 기술의 2번째 3번째 측정시간이 거의 비슷하게 나오는 것을 볼 수 있었다. 그러나 3회 측정된 결과를 평균을 내었을 때는 GML/SVG 기반의 기술이 빠른 것을 볼 수 있다. 이렇게 시간 차이가 나는 것은 Applet기반의 기술의 첫 번째 측정시간이 오래 걸리기 때문이다. 이는 서버가 재가동되었을 때의 시간으로서 Applet기반의 기술은 사용자가 요청하는 데이터를 처음 실행 될 때 서버의 메모리에 데이터를 올려서 응답하기 때문에 시간이 오래 걸릴 수밖에 없다. 물론 이러한 문제를 해결하기 위해 많은 연구들이 진행되고 있다. 그러나 다음 <표 4>의 결과를 보면 본 연구에서 구현한 시스템의 장점은 이것뿐이 아니라는 것을 알 수 있다. <표 4>는 서울·경기 지역의 데이터를 처음 로딩시킨 후 강남(강남구·서초구 일대)지역을 확대·축소시켰을 때와 강남지역에서 종로지역으로의 지도를 이동시켰을 때 응답시간을 측정한 것이다.

기본 기능인 확대·축소·이동 기능을 수행할 때 평균 수행시간이 약 10초정도 차이 나는 것을 알 수 있었

다. 이러한 시간의 차이는 <그림 15>와 <그림 16>을 통해 알 수 있다. <그림 15>와 <그림 16>은 시스템과 사용자와의 동적관계를 보여주는 시퀀스 다이어그램(Sequence Diagram)이다. <그림 15>는 본 연구를 통해 구현한 시퀀스 다이어그램으로서 사용자가 POI의 데이터를 질의하여 GML 데이터를 서버로부터 다운받는다. 이 GML 데이터는 Styling Engine에 의하여 GML의 그래픽 요소들을 SVG로 바꾸는 작업을 한다. 이 때 그래픽요소의 색이나 선의 굵기는 “style.css”에 정의되어 있는 유형에 맞추어 사용자에게 보여진다. 즉 처음 실행할 때 받은 GML과 SVG를 이용하여 사용자에게 보여주게 되므로 서버와의 접속이 일어나지 않는다. 그러나 기존 기술의 경우에는 <그림 16>에서 표현되어 있듯이 위치가 바뀌거나 화면에 보여지는 지역의 축척이 틀려지면 서버로 접근하여 그 지역의 데이터를 클라이언트로 가지고 오게

표 4. 동일 영역의 기능수행 응답시간 비교(단위 : 초)

항목	Applet기반의 기술	GML/SVG기반 기술	차이
확대	4.29	0.88	3.41
축소	9.85	0.77	9.08
이동	17.16	0.70	16.46
평균	10.43	0.78	

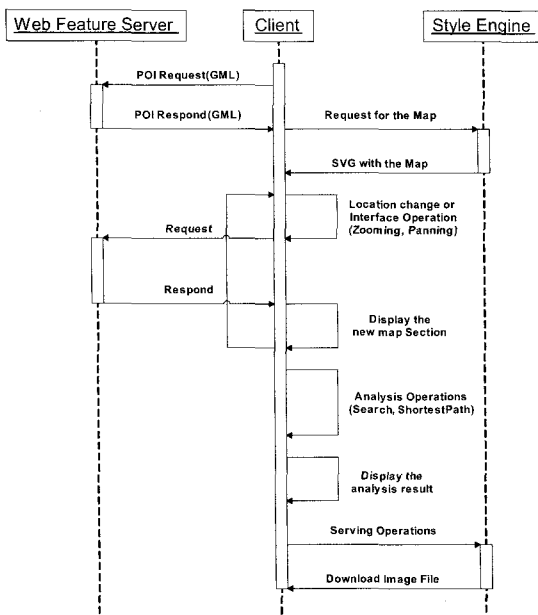


그림 15. GML/SVG기반의 시퀀스 다이어그램

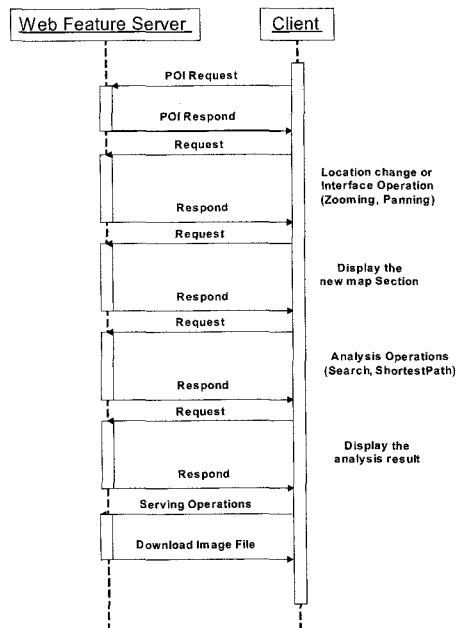


그림 16. 기존 기술의 시퀀스 다이어그램

된다. 이러한 이유로 지도 표현에 있어서 GML기반보다 더 많은 시간이 소요된다.

### ③ 표현의 정밀도

<그림 17>은 래스터 데이터와 GML/SVG기반으로 표현한 지도를 확대하였을 때를 비교하기 위하여 확대할 영역을 설정한 지역이다.

<그림 18>은 <그림 17>에서 설정한 영역을 800%로 확대한 그림이다. <그림 18>의 (a)에서 볼 수 있듯이 이미지로 표현된 지도는 픽셀로 되어 있으므로 확대를 하였을 때 선이 부드럽게 표현 되지 않고 색의 해상도 또한 떨어지는 것을 볼 수 있다. 그러나, <그림 18>의 (b)의 GML/SVG로 구현된 지도 데이터의 경우는 Vector로 표현이 되므로 확대를 할 경우 고해상도의 지도 표현이 가능하다. 이는 지적 분야와 같은 정밀성 있는 데이터 표현에 적합하다는 것을 알 수 있다.

## 5. 결 론

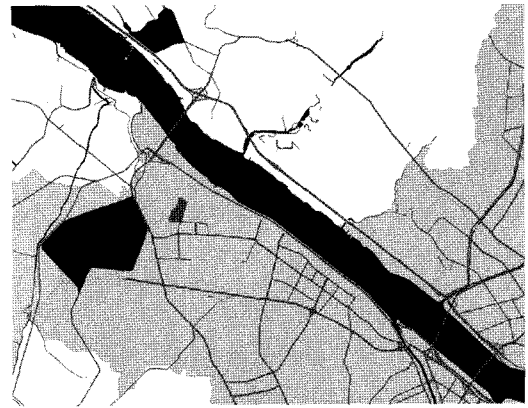
본 연구는 지리정보 분야에서 새로운 기술로 부각되는 GML을 인터넷 지리정보 서비스 분야에서 활용할 수 있는 방안을 제시하였다. 지리정보 서비스를 제공하는 기존 기술 중 지도를 이미지로 변환하여 서비스 하는 경우, 해상도가 대체로 낮고 파일 크기가 커서 휴대 단말기에 사용하기 어렵다는 것과 다양한 속성정보를 사용할 수 없다는 단점이 있다. 물론 Applet과 같은 기술을 통해 벡터로 HTML화면에 플러그인시켜 서비스하는 경우가 있으나 플랫폼마다 서로 다른 플러그인을 개발해야 하는 불편함과 지리정보 데이터를 표준에 따라 일관성있게 표현하는 것도 어렵다. 이런 상황에서 XML기반의 벡터 그



그림 17. 표현의 정밀도 비교를 위한 확대 영역



(a) 이미지 지도 확대



(b) GML/SVG기반 지도 확대

그림 18. 표현의 정밀도 비교

래픽 처리를 위한 SVG의 등장은 일상적인 환경에서 간단한 작업으로 벡터 그래픽을 사용할 수 있는 기반을 제공하였다.

본 연구에서는 플랫폼에 상관없이 서비스가 가능하면서도 표준화된 표현이 가능한 GML 기법을 이용하여 시스템을 구축하고, 이를 SVG로 변형하여 제공함으로써 기존방식의 GIS서비스에 대한 개선방안을 제시하였다. 특히 전문 GIS프로그램을 사용하지 않으면서 인터넷 브라우저를 이용하여 서비스할 수 있고, 시스템을 구축할 수 있는 공개된 소스를 제공하기 때문에 서비스 구축에 있어서의 개발 시간의 단축 및 재활용의 가능성을 높여 주었다.

우리나라에서는 아직까지 GML를 활용한 사례가 적은 뿐 아니라 인터넷상에서 SVG를 이용하는 사례도 없다. 이러한 상황에서 GML데이터를 이용한 지리정보 서비스의 제시는 의미가 있다고 할 수 있다. 본 연구의 결과는 다음과 같다.

- 표준화된 표현이 가능한 XML기반 기술의 GML을 인터넷 GIS환경에 적용
- 전문적인 GIS소프트웨어의 필요 없이 고품질의 벡터 그래픽을 제공하고 사용자와 상호작용이 가능한 지리정보 서비스를 구현
- 이는 사용자의 지리정보 서비스 활용가능성을 높여 줄 뿐만 아니라 서비스 제공자 입장에서도 시스템 구현에 있어서 최소의 노력과 비용만으로 가능

이렇게 GML데이터를 표현하기 위해 SVG를 이용하면 기존의 GIS시스템이 구축되어진 곳에서만 가능했던 인터넷 GIS를 일반 개인까지도 쉽게 구현하고 서비스 할 수 있게 된다. 또한 기존의 지리정보를 취급하는 기업이나 공공기관에서는 사용자에게 양질의 데이터를 서비스할 수 있을 것이며 대중이나 정책입안자들의 의사결정을 위한 GIS데이터 접근이나 분석에 보다 나은 도움이 될 것이다. 이를 위해 별도의 추가적 작업이 필요하지 않으며 지리정보 서비스에도 서버측의 부담을 최소화 할 수 있다.

## 참고문헌

1. J. Bowler, C. Brown, M. Capsimalis, et al, Scalable Vector Graphics, W3C Recommendation 04, Sep. 2001. <http://www.w3.org/TR/SVG/>.
2. O. Consortium, Geography Markup Language (GML) v2.1.2, Document Number 02-069, Sep. 2002. <http://www.openngis.net/gml/02-069/GML2-12.html>.
3. J. E. Corcoles and P. Gonzalez, A Specification of a Spatial Query Language over GML, In Proc. of ACM GIS, 2001.
4. J. E. Corcoles and P. Gonzalez, Analysis of Different Approaches for Storing GML Documents, In Proc. of ACM GIS, 2002.
5. A. Garmash, A Geographic XML-based Format for the Mobile Environment, In Proc. of HICSS-34, 2001.
6. Y. Isakowski and A. Neumann, Interactive Topographic Web-Maps Using SVG, SVG Open/Carto.net Developers Conference, Zurich, Switzerland, July 1517, 2002. <http://www.svgopen.org/>.
7. A. Neumann and A. Winter, Vector-based Web Cartography: Enabler SVG, Carto.net - Cartographers on the net, 2000. [http://www.carto.net/papers/svg/index\\_e.html](http://www.carto.net/papers/svg/index_e.html).
8. W.T.M.S.B.Tennakoon, Visualization of GML data using XSLT, 2003.
9. Zhimao Guo, Shuigeng Zhou, Zhengchuan Xu Aoying Zhou, G2ST: A Novel Method to Transform GML to SVG, 2003.
10. Zhong-Ren Peng, Chuanrong Zhang, The roles of geography markup language (GML), scalable vector graphics (SVG), and Web feature service (WFS) specifications in the development of Internet geographic information systems (GIS), Journal of Geographical Systems, 2004.
11. 김은영, 오해석, 2001, SVG웹 그래픽을 포함하는 XML문서를 위한 하이브리드 데이터 모델, 한국통신학회지 2001.
12. 김희원, 2003, SVG를 이용한 지리정보 활용에 대한 연구, 서울대학교 대학원 석사학위논문.
13. 반재훈, 조정희, 문상호, 홍봉희, 2000, GML데이터를 지원하는 확장된 DOM의 설계 및 구현, 한국정보과학지.
14. 이해진, 이현아, 김동호, 김진석, 2003, GML기반 통합 맵서버 설계 및 구현, 한국지리정보학회지.
15. 정원일, 배해영, 2004, 이질적인 공간정보시스템의 상호 운용성을 위한 효과적인 지리 데이터의 GML사상, 한국멀티미디어 학회지.