

GML을 적용한 이동체 관리 시스템의 설계 및 구현

이혜진^o 이현아 김동호 김진석
 한국전자통신연구원 우정기술연구센터
 (lhjin^o, halee, kdh, kimjs)^o@etri.re.kr

Design and Implementation of Moving Object Management System using GML

Hyejin Lee^o Hyunah Lee Dongho Kim Jinsuk Kim
 Postal Technology Research Center, ETRI

요약

본 연구는 이동체와 지도정보의 출력을 위한 XML 인코딩 표준인 GML(Geography Markup Language)을 이용하여 이동체 관리 시스템을 설계하고 구현하였다. OGC(OpenGIS Consortium) 웹 매핑 기술인 웹맵 서버(WebMapServer) 개념을 응용하여 공간 데이터와 이동체 정보를 통합하며, 통합 과정에서 요구되는 표준화된 데이터 모델과 인터페이스는 OGC가 제안한 표준화된 개념을 사용한다. 데이터는 별도의 응용스키마를 정의하여 적용하였다. GML은 공간데이터와 비공간데이터의 저장과 전송을 위한 XML 인코딩 표준이므로, 데이터 공유를 위한 확장성 및 상호운용성을 확보할 수 있다.

1. 서론

GML(Geography Markup Language)은 공간데이터와 비공간데이터의 저장과 전송을 위한 XML 인코딩을 의미한다[1]. GML은 현재 지리정보를 나타내는 국제표준으로 자리잡고 있으며, OGC에서는 웹맵서버(WebMapServer)의 개념을 사용하여 표현하고 있다[2]. 본 연구에서는 이동체 관리 시스템의 구성요소로서 GML에 대한 특징을 부각시켜 적용할 수 있는 방안을 제시하여, 표준을 이용하여 데이터 공유를 위한 확장성 및 상호운용성을 확보하고자 한다.

2. 관련연구

GML 표준화 연구는 2000년 5월 1.0버전을 시작으로 3년동안 많은 노력을 통해 오늘날에 이르렀다. GML 표준화 연구 초기에는 지리정보의 XML 인코딩이라는 동일한 목적하에 OpenGIS Consortium, ISO TC/211, 일본 세 개의 기관에서 서로 다른 이름으로 진행되었다[3][4]. 2001년도에는 각각 별도로 진행되었던 표준의 통합이 요구되어 연합된 형태로 GML 3.0 제정을 추진하였다. 그 결과, 2003년 1월 GML 3.0이 공표된 실정이다[5].

3. 스키마 설계

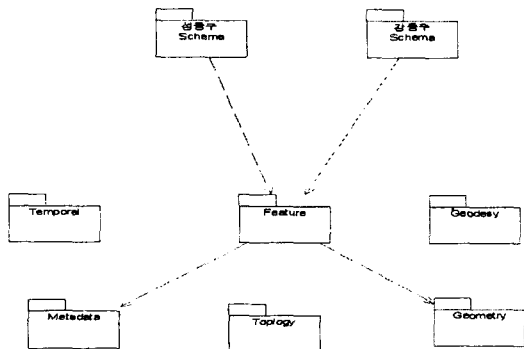


그림1 GML 객체와 응용스키마의 관계

응용스키마를 설계하기 위해서는 실제 스키마 모델링을 할 데이터를 분석해야 한다. 모델링 할 데이터로부터 공통된 요소를 도출하여 각각의 구성요소에 어떠한 의미를 부여할 것인가를 결정해야 한다. 실제 모델링 단계에서는 표준화된 모델링 언어인 UML을 사용하며 실제 우리가 설계하고자 하는 응용스키마가 **성동구 Schema**라고 가정할 경우, 성동구 Schema와 GML 기본 스키마의 관계는 그림1과 같다.

그림2는 실제 성동구 Schema를 상세 설계한 그림이다. 성동구 Shape 데이터는 이동체를 나타내는 부분과 공간데이터를 표현하기 위한 부분으로 크게 두 부분으로 나누어져 있다. 공간데이터는 Point, Line, Polygon 세 개의 공간 객체 유형으로 나누어져 있으며, 이동체 스키마는 이동체의 위치, ID, 궤적, URL 등의 구성요소를 가지고 있다.

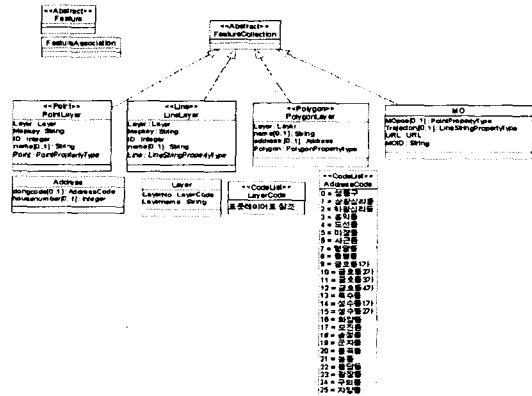


그림2 성동구 스키마

본 시스템에서는 공간데이터사이의 Topology 관계성은 배제하였다. 향후 연구 요소로 남겨두며 공간 연산자와 Xlink 방법을 도입하여 이동체와 공간 연산이 가능토록 하는 방법을 사용하게 될 것이다[6].

4. 시스템 설계 및 구현

이동체 관리 시스템은 실시간으로 차량이동 위치를 파악하여

Web, PDA 등의 단말기에 지도데이터상에 표현하는 것으로 목적으로 하는 시스템이다. 전체적인 시스템 구조는 아래와 같으며 Location Data Interface, Mo Engine, MODB, Web/Mobile Presentation Interface의 네 부분으로 구성된다.

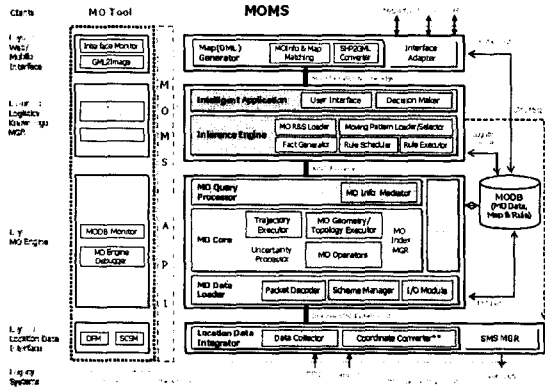


그림3 이동체 관리 시스템 구조

Web/Mobile Presentation Interface의 흐름은 다음 그림4와 같으며, 전체 단계는 크게 네 부분으로 나눌 수 있다.

표1 Web/Mobile Presentation Interface 흐름 단계

단계	기능	상세 설명
1	질의 입력 및 변환	사용자 인터페이스로부터 입력된 요구사항을 질의형태로 변환하여 서버에 전달한다.
2	맵 매칭	전달된 질의로 이동체 정보와 해당 지도를 검색하여 좌표를 변환한다.
3	지도 포맷 변환	맵 매칭 결과를 GML, Image 등의 포맷으로 변환한다.
4	출력	변환된 지도를 화면에 출력한다.

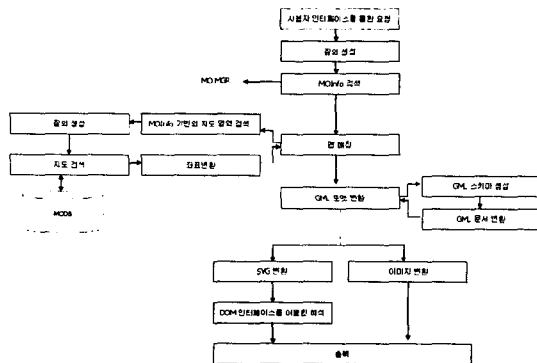


그림4 Web/Mobile Presentation Interface 흐름도

4.1 클라이언트

Web/Mobile Presentation Interface 클라이언트는 유무선 복합된 형태를 가지며 이동체 정보를 지도와 같이 출력하는 기능을 제공한다. 현재 사용되고 있는 대표적인 클라이언트 장치를 보면 핸드폰, PDA, PC 3가지가 있다. 일반적으로 PC는 관심있는 이동체의 위치확인을 위하여 Web 응용을 사용하여 정보를 제공받으며,

PDA와 핸드폰은 이동성(Mobility)을 가진 사용자를 위하여 위치 확인 및 부가정보를 제공할 수 있다. 클라이언트 모듈은 질의 입력 모듈, DOM 인터페이스, 질의결과 출력 크게 3부분으로 나누어진다. 사용자는 각 모바일 장비의 질의 입력 모듈을 통하여 이동체 위치 및 속성 정보에 대한 질의를 요청하고, 이는 질의입력 모듈을 통해 입력된다.

DOM Interface는 Web(PC) 클라이언트 측 모바일 장비로 전달된 SVG, GML과 같이 벡터형태의 위치 정보는 DOM Interface를 통하여 디지털 지도화면에 출력한다. 휴대폰, PDA의 경우에는 자체 브라우저별로 지원하는 이미지를 출력하므로 DOM Interface를 사용하지 않는다.

전자지도 및 질의결과 출력은 맵매칭된 결과를 화면상에 출력 이동체 정보와 Map Matching 모듈에서는 MO Engine에서 질의 결과로 전달된 차량 위치 관련 정보를 텍스트 형태로 출력하고 차량 위치가 포함된 영역의 디지털 지도 화면을 사용자가 검색할 수 있도록 한다.

Web/Mobile Presentation Interface에서 제공되는 기본 데이터는 GML이며, 클라이언트 특징에 따라 선택적으로 변환한 이미지를 제공한다.

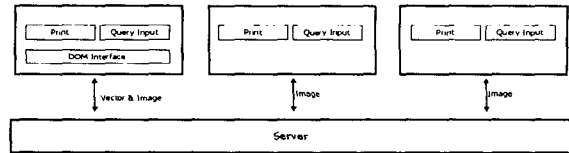


그림5 단말기별 클라이언트 모듈

4.2 서버모듈

Web/Mobile Presentation Interface 는 크게 이동체 정보와 지리 공간데이터를 매칭하는 Map Matching 부분과 지리공간정보를 GML로 변환하는 변환기(Converter) 두 부분으로 구성된다.

Web/Mobile Presentation Interface 서버 영역의 구성은 그림6과 같다. 서버는 이동체 정보를 질의하여 포맷 변환이 된 지도와 매칭하는 기능을 제공한다. 결과적으로 이동체 위치 정보와 지도 정보의 혼합하여 GML 형태로 제공한다.

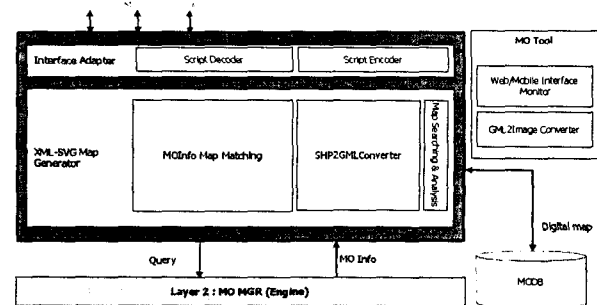


그림6 서버 모듈 구성도

Interface Adapter는 클라이언트와 서버 사이에서 전송되는 데이터의 Encoding/Decoding 기능을 제공하며, MO Engine에서 이동체 정보를 검색하기 위하여 질의를 제공한다. MO Engine에서 처리 가능한 질의형태로 변환하여 실행한다.

Script Encoder는 변환된 GML형태의 위치 정보는 Script Encoder를 통하여 각 클라이언트 장비에서 이해가 가능한 데이터 형태로 변환된 후 전송한다.

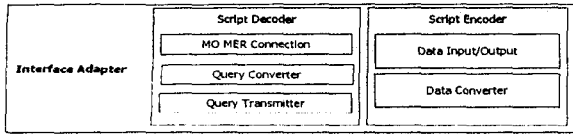


그림7 Interface Adapter 구성

Script Decoder는 각 모바일 장비에서 전송된 질의 데이터는 Script Decoder를 사용하여 MO Engine에서 이해가 가능한 형태로 변환되며 변환된 정보를 MO Engine으로 전송한다. 질의 결과는 Map(GML) Generator로 전송된다.

Map(GML) Generator는 검색된 이동체 정보를 가지고 지도와 매칭한다. 현재 지도는 MODB에 저장되어 있으며, 검색된 지도 영역은 GML 형태로 변환된다.

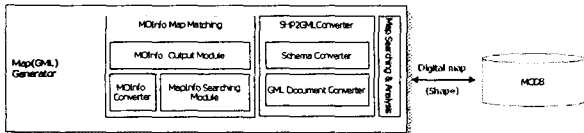


그림8 Map(GML) Generator 구성

MOInfo Map Matching 모듈은 MO Engine에서 처리된 질의 결과로 이동체 위치를 분석한다. 이동체가 존재하는 지도 영역을 검색한다. MOInfo Map Matching에는 MODB의 지도데이터의 제공 영역, 포맷 등에 관한 메타데이터를 가지고 있으며, 메타데이터를 기반으로 지도 영역을 검색한다. MOInfo Map Matching는 이동체 위치와 지도 좌표 사이에 좌표 변환이 필요하다. 지도의 메타데이터를 근거로 좌표계 기준점을 기반으로 좌표를 변환한다. 이동체 정보는 검색된 지도 데이터와 SHP2GML Converter로 전달된다.

SHP2GML Converter는 SHP 파일의 디지털 지도를 SHP2GML에서 GML형태의 데이터로 변환한다. 1단계는 원본 지도 데이터를 중심으로 스키마를 생성하고, 2단계는 생성한 스키마를 기반으로 GML 문서 자체를 변환한다.

Map Searching & Analysis는 이동체 위치와 같은 영역의 지도를 검색하고 지도의 스키마를 분석한다. 지도는 포맷에 따라서 각기 다른 스키마를 가질 수 있으므로, GML 변환을 위해서는 스키마 분석이 필요하다.

5. 구현

Web 구현 인터페이스와 PDA 구현 인터페이스는 다음과 같다.

5.1 Web 구현 인터페이스

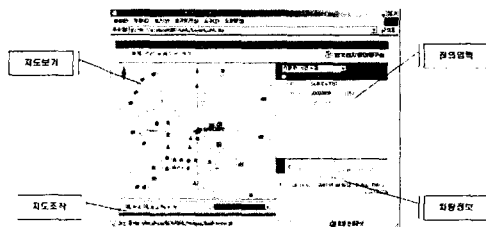


그림9 Web User Interface

Web PC용 사용자 인터페이스 설계화면은 그림9와 같으며, 화면 출력영역과 질의영역으로 크게 두 부분으로 나누어진다. 화면 출력영역은 이동체 정보를 지도와 같이 출력하는 기능을 하며, 지도 영역에 대한 확대, 축소, 이동 등 일반적인 기능을 제공한다. 질의 영역은 질의입력 부분과 질의결과 출력 부분으로 구성되며, 5가지 유형별 질의가 제공된다. 질의결과에 따른 화면 출력 기능을 가진다.

5.2 PDA 구현화면

PDA용 사용자 인터페이스는 그림10과 같은 기본 화면을 가지며, 웹 PC에 비하여 화면출력 영역이 작으므로 기능을 간략화하여 구성해야 하기 때문에 팝업 메뉴형태로 구성하였다. PDA(Compaq IPAQ)의 내장 Internet Explorer를 이용하여 클라이언트를 구동하며, 웹 클라이언트는 Html(JSP) + Image(서버생성)를 사용하여 동작한다.

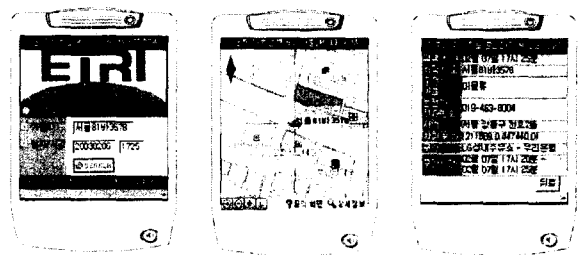


그림10 PDA 질의 Interface

6. 결론

본 연구에서는 GML을 적용한 이동체 관리 시스템 설계 및 구현하였다. 본 연구를 통하여 표준화된 방식을 적용한 데이터 공유 및 전송을 가능하며, 이를 통한 상호운용성을 확보할 수 있도록 하였다.

향후 GML은 현재 W3C 표준 권고안으로 이미 폭넓게 이용되고 있는 SVG(확장 벡터 그래픽, Scalable Vector Graphics)와의 연동을 통하여 표준화된 그래픽 형태를 지원할 예정이며, 이와 관련된 많은 연구가 진행되고 있는 실정이다[7]. 뿐만 아니라 2003년에 발표된 GML3의 경우 LBS를 고려한 요소를 추가하였으므로 이동체 관리 시스템에서 응용하기에 적합하리라 생각되므로, 향후 GML3, SVG와 관련된 국제 표준화 동향을 반영하여 추진할 계획이다.

[참고문헌]

[1]OpenGIS Consortium, Inc., Geography Markup Language(GML) 2.1.2, 2002
 [2]OpenGIS Consortium, Inc., Web MapServer Interface Specification, Revision 1.1.1 2002
 [3]ISO/TC 211, <http://www.isotc211.org/>
 [4]G-XML, <http://www.dpc.or.jp/>
 [5]OpenGIS Consortium, Inc., Geography Markup Language(GML) 3, 2003
 [6]World Wide Web Consortium, Inc., XML Linking Language (XLink), 1999(<http://www.w3c.org/TR/2000/WD-xlink-20000221>)
 [7]World Wide Web Consortium, Inc. Scalable Vector Graphics(SVG), 2003(<http://www.w3.org/TR/SVG11/>)