

GIS Vector Map 변환 엔진의 설계 및 구현¹⁾

최재영^o, 김명삼, 정영지
원광대학교 컴퓨터 공학과
{kassaka^o, ksixpost, yjchung}@wonkwang.ac.kr

A Design and Implementation of GIS Vector Map Converting Engine

Jae-Young Choi, Myung-Sam Kim, Yeong-Jee Chung
Dept. of Computer Engineering, Wonkwang Univ.

요 약

위치기반 서비스의 기초적인 부분이며 필수적인 부분이 바로 사용자가 자신의 위치에 대한 인지를 명확하게 할 수 있도록 위치를 표시하고 보여주기 위한 지도 데이터 서비스이다. 지도 데이터 서비스를 위해서는 지리정보를 특정 어플리케이션에 의존함이 없이 즉시 볼 수 있는 형태의 포맷으로 변환하는 것이 필요하다. 이를 충족시킬 수 있는 포맷으로 웹브라우저 상에서 바로 볼 수 있는 SVG를 고려할 수 있다.

본 논문에서는 GIS Vector Map의 대표적인 파일 형식인 DXF, SHP, MIF를 분석하였다. 또한, 이들 파일 형식을 공통된 하나의 파일형식인 XML로 변환하여 DB를 구축하고, 구축된 XML 데이터를 SVG로 변환하는 엔진을 구현하였으며 이를 통하여 여 98%이상의 변환 효율을 갖는 결과를 얻을 수 있었다.

1. 서론

현재 모바일 서비스로 주목을 받고 있는 것 중 하나가 Location Based Service(LBS)다. LBS는 사용자가 소지한 단말기를 통해 사용자의 위치를 파악하고, 해당 위치와 관련한 여러 부가가치 서비스를 제공하는 것인데, 이의 기본적인 서비스는 사용자의 위치에 대한 안내 서비스이다. 이때 사용자의 현재 위치에 대한 정보를 보다 직관적이고 명확하게 서비스하기 위해 지리 정보를 이미지화 하여 제공되어야 한다.

현재 사용되고 있는 지리 정보를 저장하는 GIS Map Data의 경우 DXF, SHP, MIF 등 여러 포맷이 공존하고 있다. 이러한 포맷들은 전용의 어플리케이션이 필요하며 GIS DB를 구축하는데 이용될 뿐 일반 사용자에게 지도 정보를 제공하는데 어려움이 있다

따라서 이러한 파일 포맷들을 특정 어플리케이션에 의존하지 않으며, 확대와 축소시 이미지의 왜곡이 적고, 위치기반 서비스를 위해 좌표계를 구현 할

수 있는 파일 포맷으로의 변환이 필요하다.

이러한 여러 요건을 충족시키는 이미지 파일 포맷으로 SVG 포맷을 고려해 볼 수 있다. SVG 파일의 경우 벡터 그래픽 포맷으로 표현하는데 사용하는 좌표 값을 이용하면, 이미지에서 사용자의 위치를 특 정확 할 수 있다.

본 논문에서는 사용자에게 SVG 파일 포맷으로 지도 이미지를 제공하기 위해 GIS Vector Map인 DXF, SHP, MIF 포맷을 SVG로 변환엔진을 구현하고자 한다

2. SVG Converter의 설계

2.1 SVG Converter의 구성

그림 1은 SVG converter의 구성을 나타낸 것으로 두부분으로 이루어 졌으며 크게 DXF, SHP, MIF 파일을 XML 파일로 변환하여 DB에 저장하는 부분과 저장된 XML 파일을 SVG 파일 포맷으로 변환하는 두가지 부분으로 나뉜다.

첫 번째 부분인 XML Converter는 DXF parser, SHP parser, MIF parser, Sentences DB, Sentences to XML의 다섯 부분으로 이루어져 있다.

DXF, SHP, MIF parser는 각각 DXF, SHP, MIF

1) 본 연구는 대학 IT연구센터 육성·지원사업의 연구결과로 수행되었음

파일을 파싱하는 역할을 한다. 이 세부분은 수치지도 DB에 저장된 DXF, SHP, MIF 포맷을 불러와 파싱하여 파일 내용 중 유의미한 부분만을 선별하고, 그 내용을 메모리 상에 존재하는 Sentences DB에 저장하게 된다. Sentences to XML은 DB에 저장된 각 구문을 트리구조로 정렬하고 이를 토대로 XML 파일을 생성하여 XML DB에 저장한다. 이렇게 서로 다른 데이터 포맷을 동일한 XML 포맷으로 저장함으로써 이후 SVG 파일 포맷으로 변환하는 converter의 설계가 간편해지며, 각 파일에 해당하는 형태의 DB를 따로 구축 할 필요 또한 없게 된다,

XML DB와 XML to SVG converter는 실질적으로 SVG를 생성하는 부분이다. XML DB는 각 각의 파일에서 생성된 XML 파일을 각 파일이 표현하는 지도의 이름을 파일명으로 하여 저장한다. 저장된 XML 파일은 이후 전자지도 파일을 대신하여 사용된다.

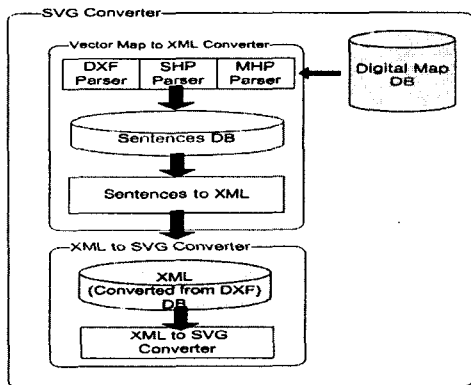


그림 1 SVG Converter의 구성

2.2 XML 문서 변환기의 설계

XML 문서 변환기는 DXF, SHP, MIF 파일을 Digital Map DB에서 읽어와 이를 세 파일이 공통으로 공유하는 특정형식의 XML 파일로 변환하는 역할을 한다.

GIS Vector Map의 변환시 지도의 레이어 정보를 구별하여 저장해야 한다. 이는 지도를 이용 POI 서비스를 제공하거나 지도의 필터링에 대비하기 위해서이다.

그리고 파일을 이루는 세가지 요소인 Geographic Information, Attribute Information, Display Information을 구별하여 저장해야 한다.

Geographic Information은 도면 내에 그려진 요소들의 형상 및 위치를 나타내는 자료나 좌표 값이 주

된 정보이며, Attribute Information은 지리 정보에 부가하여 각 요소들의 성격을 문자열로 정의하며, Display Information은 지리 정보와 속성을 화면이나 도면에 표시하는 방법에 대한 자료이다.

2.3 GIS Vector Map Data 분석

2.3.1 DXF

DXF(Data eXchange Format)파일은 AutoCAD의 데이터를 교환하기 위한 포맷으로, 캐드 파일의 삼차원 모델을 지원할 수 있는 형태로 AutoDesk 사에 의해 만들어졌다.

DXF는 위상구조나 속성자료 관리 등에 있어 한계를 갖고 있어 GIS 데이터로 활용되기에는 제약점이 많지만 현재 우리나라에서는 국가 GIS 사업의 일환으로 구축되는 수치지도 제작의 자료형식으로 사용하고 있다.

2.3.2 MIF

MIF는 MapInfo Interchange Format의 약자로 MapInfo 사의 데이터 교환 포맷의 일종이다. MIF는 지리정보상의 기하학적인 형상들에 대한 위치와 형상에 대한 정보를 저장하는 MIF 파일과 MIF 파일의 지리정보에 대응하는 지리속성에 대한 정보를 저장하는 MID 파일로 구성되어 있다. MIF 파일 하나만으로는 지리정보에 대한 속성의 표현이 불가능하기 때문에 MIF 파일의 처리와 함께 MID 파일의 처리 또한 필요하다.

2.3.3 SHP

Shapefile은 지리 공간의 형상을 표현하고 설명하기 위해 위상 구조를 지니지 않는 기하학적 위치 정보와 속성정보를 저장한다. 이러한 지리 공간의 형상은 일련의 벡터 좌표로 구성되어 shape 파일에 저장된다.

Shape 파일은 위상구조를 지니지 않기 때문에 빠르게 공간 데이터를 표시하고, 편집이 가능하며, 비연속적으로 중첩되어 있는 공간 형상 또한 쉽게 다룰수 있다.

2.4 DXF, MIF, SHP Parser의 설계

Parser는 크게 세 단계로 이루어져 있다. 처음 단계는 파싱을 위한 예비단계로 SHP, MIF 파일에만 한하여 처리하게 된다. 나머지 두 단계는 주 파싱단계로 헤더의 정보를 처리하는 부분과 데이터 섹션의

정보를 처리하는 부분으로 나눌 수 있다.

처음 예비단계에서는 파일이 속한 디렉토리 내에 존재하는 같은 포맷의 파일에 대한 정보를 저장한다.

GIS Vector Map 데이터 중 SHP, MIF 포맷은 하나의 파일이 하나의 지리 정보를 표현하는 경우도 있으나, 대부분의 경우 다수개 파일이 하나의 지리 정보를 표현하므로 한 디렉토리 내의 동일 포맷의 파일들은 하나의 지리정보를 표현하는 파일들로 가정하고 파일의 이름과 속성에서 레이어 정보를 유추하게 된다.

단 DXF 파일의 경우 하나의 파일이 하나의 지리 정보를 표현하므로 예비단계를 거치지 않는다.

본격적인 파싱의 처음단계로 파일의 헤더 섹션이나 그에 상응하는 부분을 읽는다. 여기서 DXF 포맷의 경우 레이어 정보와 그 속성에 대한 정보를 저장하게 되고 SHP, MIF 포맷의 경우에는 데이터 섹션에서 처리하는 지리정보에 대한 설정등을 저장하게 된다.

마지막 단계로 파일의 데이터 섹션에서 Geographic Information을 읽고 그 값을 Sentences DB에 저장하게 된다. DB에 저장되는 내용은 엔티티가 위치하는 레이어의 속성, 엔티티의 좌표, 엔티티의 속성이다. 이전 단계에서 저장한 레이어 정보를 이용하여, 이를 첨자로 하는 "Layername.entities's attribute"의 형식으로 엔티티 정보를 DB에 저장한다.

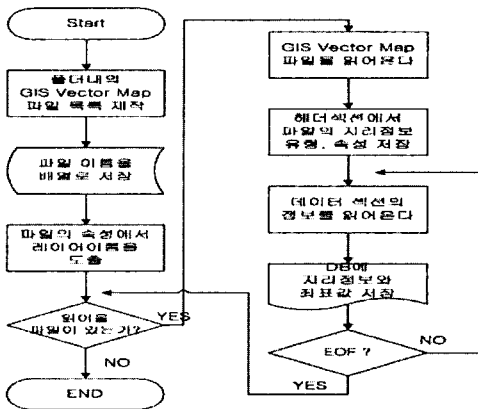


그림 2 DXF, MIF, SHP Parser

2.5 XML Converter의 제작

2.5.1 Sentences_to_XML Converter

이미 열거된 각각의 파서가 처리한 Sentences가 저장된 Sentences DB에 저장된 내용을 가나다 순의 내림차순으로 정리하면 각 Sentence는 레이어 이름이 그 처음에 첨자로 추가 되어 있기 때문에 레이어 이름별로 정렬된다. 먼저 XML의 헤더 부분과 원본 포맷을 알려주는 엘리먼트를 루트 엘리먼트로 추가한다. DB의 내용을 한 줄씩 읽어오면서 레이어 이름만 존재하는 경우 이를 레이어 엘리먼트로 저장하고, 이후 이 레이어 이름으로 시작하는 모든 요소들을 레이어 엘리먼트의 엘리먼트로 저장한다. 이때 첨자인 레이어 이름을 삭제한다. 생성된 XML 파일은 XML DB에 저장된다.

```
<?xml version="1.0"?>
<!DOCTYPE DXF [
<ELEMENT DXF (Layer*)>
<ELEMENT Layer(Polyline*, Line*, Circle*, Arc*)>
<!ATTLIST Layer name CDATA #REQUIRED>
<!ATTLIST Layer rgb CDATA #REQUIRED>
<ELEMENT Polyline EMPTY>
<!ATTLIST Polyline coord NMTOKENS #REQUIRED>
<ELEMENT Line EMPTY>
<!ATTLIST Line coord NMTOKENS #REQUIRED>
<ELEMENT Circle EMPTY>
<!ATTLIST Circle coord NMTOKENS #REQUIRED>
<!ATTLIST Circle radius CDATA #REQUIRED>
<ELEMENT Arc EMPTY>
<!ATTLIST Arc coord NMTOKENS #REQUIRED>
<!ATTLIST Arc radius CDATA #REQUIRED>
<!ATTLIST Arc StartAng CDATA #REQUIRED>
<!ATTLIST Arc FinishAng CDATA #REQUIRED>
]>
```

그림 3 XML Converter에서 사용된 DTD

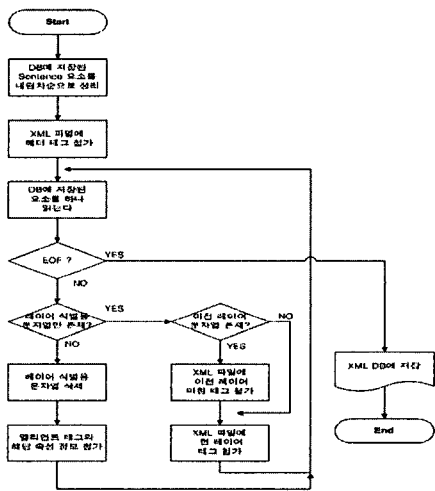


그림 4 sentences_to_XML Converter

2.7 XML_to_SVG Converter의 설계

XML to SVG Converter는 저장되어 있는 XML

파일을 읽어와 이를 SVG 형태로 변환하는 작업을 수행한다. XML 파일을 한 줄씩 읽어 들여와 그 내용이 레이어 정보 엘리먼트인지 판단하여 레이어 정보 엘리먼트인 경우에는 레이어 정보를 버퍼에 담고 레이어 엘리먼트 내의 지리정보 엘리먼트들을 가져온다.

만일 다음 줄이 레이어 정보 엘리먼트의 마침 태그인 경우 다시 처음부터 레이어 정보 엘리먼트를 찾게 된다.

이때 각 레이어의 정보를 SVG 파일에 담아 레이어별 정보를 활용할 수 있다.

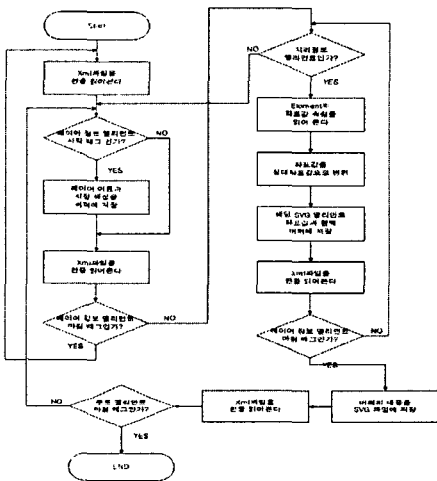


그림 5 Xml_To_SVG Converter의 구현

3. 시스템의 구현과 고찰

아래 그림은 SVG Converter로 변환한 SVG 파일과 변환전의 DXF 파일의 예이다.

이러한 변환 엔진의 경우 기능상 중요한 점은 원본과 변환본 사이의 동일성이다. 변환엔진이 원본 파일을 정확하게 번역하지 못하는 경우 변환 엔진의 효용성은 상실되기 때문이다.

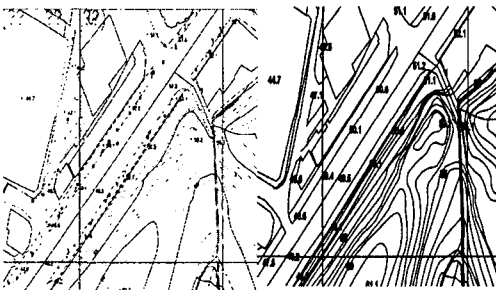


그림 6 원본 DXF Map File 그림 7 DXF에서 변환된 SVG

그래서 두 파일 사이의 차이를 분석하기 위해 파일이 가진 Geographic Information의 좌표값의 쌍인 Vertex의 숫자를 비교하였다. 두 파일의 좌표값의 경우 크게 변경하지 않고 변환하였으므로 좌표값의 이상은 없다는 가정하에 두 파일의 좌표값의 수를 비교하여 두 파일의 동일성을 검증하였다.

여러 파일을 시험 대상으로 한 결과 파일내에 심볼을 나타내는 요소가 있는 경우 이러한 심볼 요소의 구현 미비로 인해 평균 97%의 변환율을 보였고, 심볼요소가 없는 경우에는 평균 100%의 변환율을 보였다. 변환 엔진은 평균적으로 98%의 변환 효율을 가지는 것으로 나왔다.

4. 결론

본 논문에서는 GIS Vector Map 데이터를 세 파일의 공통 형식의 설정한 XML 파일로 변환하고 이를 다시 SVG 파일로 변환하는 변환 엔진을 구현하였고 그 변환 효율은 98% 였다.

이 변환엔진은 GIS Vector Map 데이터를 하나의 포맷으로 저장하는 것이 가능하도록 함으로써 각 파일의 포맷에 관계없이, 서로 혼용하거나 편집하기 용이하도록 하였고, SVG 포맷을 사용함으로써 사용자에게 SVG 어플리케이션을 이용하여 GIS Map 데이터를 처리할 수 있도록 하였으며, 이러한 Map Data 변환 엔진을 통하여 일반 사용자도 편리하게 GIS Map Data를 활용할 수 있을 것으로 기대된다.

참고문헌

[1]Autodesk, "DXF Reference Guide", 2001, Autodesk, INC
 [2]MapInfo, MapInfo Professional User's Guide, 2003, MapInfo Corporation
 [3]ESRI, ESRI Shapefile Technical Description,1998, ESRI, INC
 [4]W3c, Scalable Vector Graphics (SVG) Version 1.1, <http://www.w3.org/tr/SVG11>, Jan. 14, 2003
 [5]한성국, 정영식, 주수종, 이현창 공저, "XML 워크샵", 정익사
 [6]J. David Eisenberg, "SVG Essentials", February 2002, O'Reilly & Associates16