

# GIS 기반 교통정보 제공 시스템의 설계 및 구현

## Design and Implementation of GIS based Traffic Information Service System

이 성 욱\*  
(Seong-Uck Lee)

이 철 기\*\*  
(Choul-Ki Lee)

### 요 약

대부분의 교통 관리 시스템은 웹을 통한 교통정보 제공 기능을 포함하고 있으며, 지도를 기반으로 교통정보를 표출하는 방식을 채택하고 있다. 일반적으로 지도 기반의 시스템을 구현할 때에는 지리정보시스템(GIS)을 이용하는 것이 효율적이다. 하지만 많은 교통 정보 제공 시스템은 어도비 플래시 등의 그래픽/애니메이션 전문 소프트웨어를 위한 별도의 전자 지도를 이용하고 있어서, 전자 지도의 변경에 유연하게 대처하기 어렵다. 본 논문에서는 이러한 문제점을 해결한 GIS 기반의 교통정보 제공 시스템을 제안한다. 제안하는 시스템은 교통정보 제공 클라이언트에서 지도 출력 엔진과 지도 데이터를 분리하고, 지도 데이터는 출력에 적합한 형태로 실시간 변환하여 전송함으로써 GIS 상에서 구축된 전자지도를 교통정보 제공 시스템에서 효율적으로 이용할 수 있도록 한다.

### Abstract

Most of traffic management systems include the map-based traffic information services using World Wide Web. Geographic Information System is an efficient tool to build a map-based system. However, many traffic information service systems have own independent maps for graphic/animation softwares such as Adobe Flash. These systems make it difficult to manage modifications of maps. In this paper, we suggest a new GIS-based traffic information service system without these problems. The proposed system makes the efficient use of GIS map data for traffic information service because a map display engine is separated from map data that is transferred after real-time conversion.

**Key words:** GIS, traffic information, adobe flash, electronic map, WWW

## I. 서 론

최근 들어 구축되는 교통관리 시스템들은 웹을 통한 교통정보 제공 기능을 포함하고 있으며, 이 교통정보는 지도를 기반으로 표출되는 것이 일반적이다.

이러한 지도 기반의 교통정보 표출 시스템은 사용

하는 전자지도의 형태에 따라 비트맵 지도를 이용하는 경우와 벡터 지도를 이용하는 경우로 나누어질 수 있으며, 구현 기법에 따르면 자바스크립트를 이용하는 경우, 별도의 ActiveX를 이용하는 경우, 그리고 플래시를 이용하는 경우로 나누어질 수 있다.

이와 같은 기준에 따르면, 현재 교통정보 표출을

---

\* 주저자 및 교신저자 : 신구대학 컴퓨터멀티미디어과 교수  
\*\* 공저자 : 아주대학교 ITS대학원 교수  
† 논문접수일 : 2010년 7월 25일  
† 논문심사일 : 2010년 8월 20일(1차), 2010년 8월 27일(2차)  
† 게재확정일 : 2010년 8월 29일

위해 가장 선호되는 방식은 어도비 플래시 상에서 벡터 지도를 이용하는 것이다. 어도비 플래시는 애니메이션을 위한 소프트웨어이므로 그래픽이 미려하고 지도의 축소, 확대, 이동 등의 동작이 손쉽게 구현 가능하며 플래시 플레이어는 한번 설치된 PC에서는 별도의 ActiveX 설치를 요구하지 않고 동작할 수 있기 때문이다.

그러나 이러한 방식을 채택한 대부분 사이트들은 교통 정보 표출을 위한 기반 맵(base map)을 구축할 때 GIS를 이용하지 않고 별도로 플래시 전용의 지도를 만들어 사용하고 있다. 이것은 교통정보센터를 위해 구축된 전자지도는 높은 정밀도를 가지고 있어 데이터의 양이 매우 많은데, 실제 일반 사용자에게 교통정보를 표출하는 목적으로는 이렇게 정밀한 지도가 필요하지 않기 때문이다. 일반 사용자들을 위한 웹 서비스에 GIS에 구축된 지도를 그대로 사용하면 데이터 로딩 시간이 길어지고 지도를 구동하는 시스템에 큰 부하를 줄 수 있다. 그러나 이런 문제의 회피를 위해 별도의 표출 전용 지도를 제작하는 것은 전자지도의 초기 구축과 유지보수 작업을 번거롭게 하고 있다. 즉, 최초 시스템 구축 시에 교통관리 센터 내부에서 사용하는 GIS와는 별개로 교통정보 표출 전용의 전자지도를 제작해야 함은 물론이고, 도로망 및 POI 변경 등으로 인해 지도에 수정을 할 경우에는 교통관리 센터에서 사용하는 전자지도와 교통정보 표출용 플래시 지도를 동시에 수정하여야 하는 이중 작업이 필요하게 된다.

특히, U-City 등 유비쿼터스 시스템이 점차 보편화 되던 교통 관리 센터에서 관리하여야 할 대상의 범위가 숫자가 매우 증가하며, 이러한 관리가 원활하게 이루어지기 위해서는 GIS가 필수적이 될 것으로 예측되고 있다[1,2]. 이를 감안하면, 이러한 문제를 해결하고 GIS와 연계되어 동작하는 유연한 교통정보 제공 시스템이 필요하다는 결론에 도달하게 된다.

따라서 본 논문에서는 GIS 구축 시 사용되는 일반적인 형태의 전자지도를 이용하여 교통정보를 표출할 수 있는 교통정보 시스템을 제안하고 그 구조 및 구현 시에 고려할 사항들을 제시한다. 제안하는 시스템은 GIS의 사실상 업계 표준인 ESRI의 SHP 파일

을 이용하여 지도를 표시하며, 교통정보 제공에 적합한 형태로 지도를 실시간 변환하여 전송할 수 있다. 이로 인해 지도에 변경이 필요할 때에도 교통정보 제공용 플래시 파일을 별도로 수정하는 작업 없이 GIS 상에서 모든 작업을 처리할 수 있어 유연한 교통정보 제공 시스템을 구축할 수 있다.

## II. 교통정보 표출 기술

서론에서 언급한 바와 같이 현재 지도 기반으로 교통정보를 표출하는 시스템은 지도의 형태와 지도 구동 기술에 따라 구분될 수 있다.

지도의 형식에 따르면 래스터 지도를 사용하는 방식과 벡터 지도를 사용하는 방식으로 구분할 수 있으며 각 형식의 특징은 다음과 같다[3,4].

- 비트맵 방식 : 지도를 픽셀로 이루어진 이미지로 저장하고 처리하는 방식이다. 미려한 지도를 보여줄 수 있고 지도에 표시된 내용에 관계 없이 동일한 데이터 크기를 가지게 된다. 그러나 하나의 이미지를 그대로 확대 또는 축소하면 그림이 깨지게 되므로 각각의 확대 레벨마다 별도의 지도 이미지가 필요하며 확대를 하면 할수록 더욱 많은 양의 이미지가 필요하게 된다.
- 벡터 방식 : 지도를 벡터 형태로 저장하는 방식으로, 현재 대부분의 GIS가 채택하고 있다. 벡터 형태로 저장된 지도는 충분한 정밀도를 가지고 있는 하나의 지도만으로 추가적인 데이터 로딩 없이 확대 및 축소가 가능하다. 그러나 HTML을 기반으로 동작하는 웹 브라우저 상에서는 벡터 드로잉(vector drawing)이 매우 어려우므로 별도의 그래픽 플러그인(plugin)을 필요로 하게 된다. 이러한 HTML의 단점을 극복하기 위하여 HTML5가 제안되었으나, 마이크로소프트의 인터넷 익스플로러는 차기 버전에서야 이를 지원하기로 발표하는 등 아직까지는 보편적으로 사용 가능한 기술이 아니라고 볼 수 있다.

지도 구현 기법은 다음과 같이 구분할 수 있다.

- 자바스크립트 : 별도의 플러그인 없이 기본적인 브라우저만으로 지도 표출이 가능하지만 벡터 지도를 표출하기에는 적합하지 않다. 자바스크립트는 정적인 HTML에 동적인 요소를 주기 위해 만들어진 언어이고, 앞서 언급한 바와 같이 HTML과 자바스크립트만으로는 벡터 드로잉이 어렵기 때문이다.
  - ActiveX : ActiveX는 웹 브라우저 상에서만 동작한다는 차이가 있을 뿐, 궁극적으로는 윈도우즈 소프트웨어의 한 형태이므로 매우 자유롭게 지도를 화면에 표출하고 조작할 수 있다. 그러나 각각의 웹 사이트마다 서로 다른 ActiveX의 설치를 요청하게 되면 사용자에게 불편을 주며, 보안이나 시스템에 문제를 가져올 수도 있어 사용자들이 ActiveX 설치를 꺼리는 것이 최근의 경향이다[5].
  - 플래시 플레이어 : 넓은 의미에서는 플래시 플레이어도 ActiveX의 하나로 간주될 수 있으나, 단 한 번의 플래시 플레이어 설치만으로 다수의 플래시 무비(Flash movie)를 구동할 수 있다는 큰 차이가 있어, 각 사이트마다 별도의 ActiveX를 설치하고 동작하는 방식과는 구별하는 것이 현실적이다. 실제로도 교통정보 서비스와 관련 없는 많은 일반적인 웹 사이트들이 플래시 플레이어를 요구하고 있고, 대부분의 사용자들은 운영체제 설치 직후에 이를 기본적으로 설치하고 있다. 어도비사의 발표에 따르면 2010년 6월 현재 플래시 플레이어 10은 세계 모든 국가에서 96% 이상의 설치율을 보이는 것으로 나타났다[6]. 또한 시공간의 제약 없이 이동 중에 공간관련 정보를 제공받을 수 있는 모바일 GIS의 활용이 증가하고 있는 현실을 감안할 때[7], 플래시 기반의 시스템은 플래이어만 구동 가능하면 어떠한 모바일 환경에도 시스템 변경 없이 적용될 수 있다는 큰 장점을 가지고 있다.
- 이러한 지도 기반 서비스 제공 유형 중에서 현재 웹 사이트를 통한 교통정보 제공 시스템에 가장 널리 사용되는 것은 플래시에서 벡터 지도를 사용하는

방식이라 할 수 있다. 실제로 서울지방경찰청 종합 교통정보센터와 한국도로공사의 로드플러스 등의 많은 교통정보 제공 사이트에서는 이러한 방식을 사용하고 있다.

다만 여기서 주목할 점은 대부분의 플래시 기반의 벡터 지도 서비스들이 교통정보 표출을 위한 별도의 전자지도를 사용하고 있다는 점이다. 이런 방식을 사용할 경우, 교통정보 관리를 위하여 센터에 GIS가 구축되어 있어도 별도로 어도비 플래시 상에서 별도의 벡터 지도를 입력하고 이 지도를 구동시킬 수 있는 플래시 액션스크립트(Flash ActionScript) 프로그램을 작성하여 교통정보를 제공하여야 한다. 즉, 이러한 시스템 구현 방식은 시스템 초기 구축 시나 추후 지도의 수정이 필요할 때에도 지속적으로 두 지도를 업데이트하는 이중 작업을 요구하므로 시스템 운영에 지속적인 부담을 주게 된다.

### III. 교통정보 표출 시스템 설계

이상의 내용을 고려할 때, 유연한 교통정보 제공 시스템은 다음과 같은 성질을 만족하여야 한다.

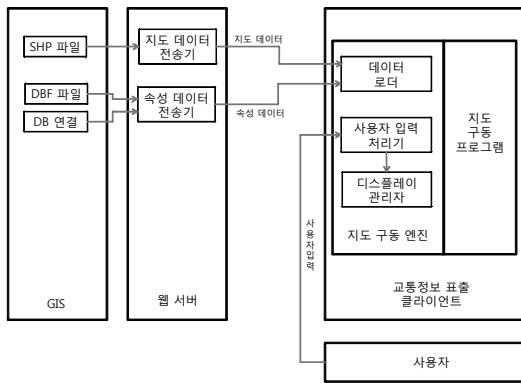
첫째, 전자지도와 지도 표출 소프트웨어 요소가 논리적으로 분리되어 있어서 지도의 수정이 필요한 경우에는 단지 GIS에서 한 번의 수정만으로 그것이 교통정보 제공 시스템까지 영향을 미칠 수 있어야 한다.

둘째, GIS와 교통정보 제공 시스템에서 공통적으로 사용할 전자지도는 독자적인 형식 또는 특정 소프트웨어 도구에서만 사용가능한 형식으로 지도가 구축되어서는 안 되며, 범용적으로 널리 사용되는 전자지도 형식을 이용하여야 한다.

셋째, 별도의 ActiveX 설치를 요구하지 않고 가급적 여러 형태의 플랫폼과 운영체제에 실행되는 기술을 채택하여 많은 사용자들에게 서비스를 제공하는 데 문제가 없도록 하여야 한다.

넷째, 많은 사용자들이 동시에 서비스를 받을 경우를 대비하여 가급적 다운로드 되는 데이터의 양을 최소화하여야 한다.

이들 조건과 현재의 기술 흐름을 종합해 볼 때,



<그림 1> 제안하는 교통정보 표출 시스템의 구조  
 <Fig. 1> Structure of the proposed traffic information service system

효율적이고 유연한 교통정보 제공 시스템을 구성하기 위해서는 입력 형식으로 GIS의 사실상 업계 표준인 ESRI의 SHP 파일을 채택하고, 지도 구동을 위한 기술은 어도비 플래시를 기반으로 하는 것이 가장 적절할 것이다. 제안하는 시스템의 구조는 <그림 1>과 같다.

그림 좌측의 GIS와 웹 서버는 센터에서 위치해 있으며 우측의 교통정보 표출 클라이언트는 어도비 플래시 파일로서 웹 서버에 탑재되어 있다가 사용자가 교통정보 제공 페이지를 요청하면 해당 사용자의 PC로 다운로드 되고 웹 브라우저 상에서 실행된다.

웹 서버는 교통 정보 제공을 위한 웹 서비스를 제공하는 서버를 지칭하며, 지도 데이터 전송기, 속성 데이터 전송기, 그리고 플래시 프로그램이 탑재된다.

지도 데이터 전송기는 GIS로부터 얻어진 SHP 파일을 읽어서 교통정보 표출 클라이언트와 약속된 형식으로 변환한 뒤 데이터를 전송하는 프로그램이다. SHP 파일은 웹 서버 자신 또는 웹 서버에서 접근 가능한 어떤 위치에 있어도 무방하다.

속성 데이터 전송기는 지도에 연관된 속성 데이터를 클라이언트에 전송하는 프로그램이다. 속성 데이터는 SHP 파일에 연관된 DBF 파일 형태이거나 DB에 담겨 있을 수 있으며, 역시 웹 서버에서 접근 가능하기만 하다면 DBF 파일의 위치나 DB서버의 위치는 시스템 동작에 영향을 주지 않는다. 속성 데이

터 전송기는 클라이언트가 지시한 쿼리를 받아서 DBF 파일 또는 DB를 조회하며, 그 결과를 교통정보 클라이언트와 약속된 형식으로 만들어 돌려주는 역할을 수행한다.

교통정보 표출 클라이언트는 교통정보 서비스 페이지가 열릴 때 웹 브라우저로 다운로드 되고 실행이 개시되는 이 플래시 프로그램으로서 지도 구동 엔진과 지도 구동 프로그램으로 구성된다.

지도 구동 엔진은 어느 클라이언트이나 공통적으로 사용되는 라이브러리들로 구성되어 있으며, 지도 데이터와 속성 데이터를 서버로부터 받아서 처리하는 데이터 로더, 사용자의 입력을 받아 처리하는 사용자 입력 처리기, 그리고 사용자 입력 처리기로부터의 지시를 디스플레이에 반영하는 디스플레이 관리자로 구성된다. 이 부분은 웹 사이트나 지도의 내용에 관련 없이 항상 동일한 내용을 가지고 있다.

지도 구동 프로그램은 지도 구동 엔진이 제공하는 API를 이용하여 해당 사이트에 적절한 지도 및 데이터 표출을 수행한다. 이 부분은 교통정보 표출 인터페이스를 어떻게 구성하는가에 따라 시스템 개발자들이 작성한다.

이상의 내용을 정리하여 제안하는 교통정보 표출 시스템의 동작 순서를 기술하면 다음과 같다.

- ① 웹 서비스 페이지에 포함된 플래시 프로그램이 클라이언트로 다운로드 된다.
- ② 플래시 프로그램이 미리 정해진 URL로부터 지도 및 관련 속성 데이터를 다운로드 하고 화면에 그린다.
- ③ 지도 구동 프로그램에 지정된 주기마다 DB 데이터를 다운로드하고 데이터 갱신에 연관된 이벤트 핸들러(event handler)를 실행하여 실시간 교통정보를 표출한다. 또한 사용자 입력이 있을 경우 디스플레이 매니저를 통해 적절하게 지도 화면을 갱신한다.

교통정보 표출 시스템이 이러한 구조를 가지게 되면, 전자지도와 속성 데이터(또는 DB 연결)는 웹 서버에 탑재되어 있고, 클라이언트 상에서 동작하는 플래시 프로그램은 데이터의 변경과 상관없이 항상 동일한 내용을 가지게 된다. 이러한 구조로 인해 지

도 변경이 필요할 때에도 단지 GIS의 전자지도만 변경할 뿐, 다른 구성 요소는 아무 변경 없이 사용할 수 있게 된다. 사실 정보 표출 정책이나 화면 인터페이스를 다르게 할 필요가 없다면, 단순히 전자지도만 교체함으로써 특정 지역을 위한 교통정보 시스템을 전혀 다른 지역을 위한 것으로 사용할 수도 있게 된다.

#### IV. 시스템 구현

이상의 설계를 구현하기 위해서는 서버 측과 클라이언트 측을 위한 개발 환경이 정해져야 한다. 지도 데이터 전송기와 속성 데이터 전송기는 웹 서버에서 동작하여야 하므로 웹 프로그래밍 언어를 이용하여 구현되어야 하고, 클라이언트 측은 어도비 플래시로 구현되어야 한다. 본 논문에서는 웹 서버 측의 소프트웨어들은 JSP로, 지도 구동 엔진은 어도비 플래시 CS4에서 액션스크립트 2.0으로 구현하였다.

##### 1. 데이터 전송기

지도 데이터 전송기와 속성 데이터 전송기는 앞서 언급한 바와 같이 JSP로 구현되었으나, 실제 응용에서는 PHP등 어떤 웹 프로그래밍 언어를 사용하여도 무방하며, 단순히 지정된 지도 데이터 또는 속성 데이터를 읽어 약속된 형식으로 클라이언트에게 전송하는 역할만을 수행하므로, 사이트 또는 데이터에 관계없이 한번 작성된 코드를 그대로 이용하게 된다.

지도 데이터 전송기가 하는 동작은 ESRI의 SHP 파일을 읽어서 플래시 클라이언트와 약속된 형식의 텍스트 파일로 변환하여 전송하는 것이다. 텍스트 파일의 구조는 SHP 파일에서 불필요한 예약영역을 삭제하고 의미가 있는 데이터 부분만을 뽑아 구성하였다[8]. 그 예를 보이면 <표 1>과 같다.

속성 데이터 전송기 또한 주어진 쿼리(query)를 받아 지시된 대로 DBF 파일 또는 DB로 부터 데이터를 추출하여 텍스트 파일 형태로 돌려주는 역할을 하게 된다.

다만, 이 두 프로그램이 SHP 또는 DBF 파일에 대

한 클라이언트의 요청을 처리할 때에는, 매번의 요청마다 이진 파일을 읽어 텍스트를 생성하지 않고, 첫 번째 요청이 있을 때 캐시(cache) 파일을 생성하여 두었다가 다음 번 요청부터는 같은 처리 작업을 반복하지 않고 캐시 파일의 내용을 그대로 전송한다. 이것은 매번 요청 시마다 이진 파일을 읽어 지정된 형식으로 변환하는데서 발생하는 로드를 현저하게 감소시켜 준다.

또한 효과적인 데이터 전송을 위해 한 가지 생각할 것은 대부분의 지도기반 교통 정보 표출 서비스들은 정밀도가 높은 지도를 필요로 하지는 않으며, 지도의 형태만 똑같이 유지된다면 본래 지도에서 사용하던 좌표계를 그대로 유지할 필요가 없다는 점이다. 일반적으로 GIS 상에서 구축되는 전자지도는 높은 정밀도로 제작되므로 데이터의 양이 많으며, 그 데이터가 텍스트 형태로 변환되면 하나의 값을 표현하는데 필요한 글자의 수도 많아지게 된다. 이러한 데이터를 그대로 클라이언트로 전송하게 되면 데이터 로딩 시간이 길어짐은 물론, 플래시 플레이어 상에서 동작하는 정보 표출 클라이언트의 동작 성능에 큰 저하를 가져오게 된다. 따라서 제안하는 시스템이 대용량의 지도에서도 원활하게 동작하도록 하기 위해서는 지도 데이터 전송기에 지도를 단순화 하는 기법을 구현하여야 한다. 지도의 선형을 단순화하는

<표 1> 지도 데이터의 예  
<Table 1> An example of map data

| 지도 데이터 전송기가 생성하여 클라이언트로 전송하는 내용   | 의미   |
|---|--|
| SHP,<br>5,<br>2011.1,4398.7,2077.0,4477.6,<br>1,<br>2020.1,4476.6,2021.9,4477.6,<br>1,<br>9,<br>2020.0,4476.6, .....<br>..... | - 도형 파일임<br>- 폴리곤 레이어임<br>- 맵 바운드<br>- 1번 도형<br>- 1번 도형의 바운드<br>- 이 도형은 1개 파트로 구성됨<br>- 첫 번째 파트는 9개 점으로 구성됨<br>- 파트를 구성하는 점들의 좌표 리스트 |

방법에 대해서는 많은 연구가 있어 왔으나[9], 앞서 설명한 바와 같이 제안하는 시스템에서는 선형을 단순화시킴과 동시에 전송되는 데이터의 양도 줄이는 기법이 필요하다. 따라서 이 두 가지 기능을 동시에 수행하기 위해 다음과 같은 방법을 사용하였다.

① 맵 바운드(map bound), 즉 전체 지도를 포함하는 최소 사각형을 구하고, 그 좌상단 점과 원점(0, 0)과의 x, y 좌표 값의 차이를 구하여 이 값을 dx, dy로 놓는다. 즉, 맵 바운드 좌상단 점의 좌표가 (x<sub>0</sub>, y<sub>0</sub>)라 하면 dx = x<sub>0</sub>, dy = y<sub>0</sub>가 된다.

② SHP 파일을 구성하는 모든 점들을 새로운 좌표로 이동시킨다. 현재 처리 중인 점의 좌표가 (x, y)라 하면, 새로운 좌표인 (nx, ny)는 다음과 같이 구할 수 있다.

$$nx = \text{round}((x - dx) * s)$$

$$ny = \text{round}((y - dy) * s)$$

여기에서 round는 소수점 첫 번째 자리에서의 반올림을 의미하며, s는 지도를 단순화시킬 스케일(scale)을 의미한다. 예를 들어 세로 크기가 1000인 지도를 100으로 줄이고 싶다면 s 값은 0.1로 지정하게 된다.

③ 이 같은 변환을 거친 폴리라인과 폴리건 개체를 구성하는 점들을 스캔하면서 동일한 점이 연속해서 있으면 하나로 만든다.

④ 3단계의 처리를 거친 폴리라인과 폴리건 개체가 단지 하나의 점만 가지고 있는 경우가 발견되면 개체 자체를 삭제한다.

위의 2, 3, 4 단계는 1-패스(1-pass)로 실행될 수 있으므로 실제로 많은 처리 시간을 필요로 하지 않으며, 앞서 언급한 바와 같이 캐시 파일을 사용하므로 지도 수정 후 최초의 지도 요청 시에 한번만 실행되어 서버에 주는 부담은 크게 증가하지 않는다.

이 알고리즘을 사용하여 지도를 단순화하였을 때, 지도 데이터의 감소를 시험한 결과가 <표2>에 제시되었다.

이 실험에 사용한 원본 지도 데이터는 6,961KB의 크기를 가진 SHP 파일이며, 지도 단순화를 시행하지 않은 경우(표2의 첫줄, 축소비율 1/1)에는 불필요한 필드들의 삭제를 통해 약간의 데이터 크기 감소가

<표 2> 지도 단순화 후의 데이터 크기  
<Table 2> Data size after map simplification

| 지도 축소 비율 | 지도 크기 (x, y 좌표 최대값) | 데이터 크기 (KB) |
|----------|---------------------|-------------|
| 1 / 1    | 166714, 150979      | 5227        |
| 1 / 10   | 16671, 15098        | 4000        |
| 1 / 100  | 1667, 1510          | 1097        |
| 1 / 200  | 834, 755            | 558         |
| 1 / 1000 | 167, 151            | 113         |

있었을 뿐 원본 파일과 아주 큰 차이를 보이지는 않는다. 그러나 지도를 축소시켜 해상도를 낮추면서 본 알고리즘을 수행하면 지도 데이터의 크기가 현저히 감소함을 볼 수 있으며 이것은 데이터 전송 및 로딩, 그리고 지도 구동 시간에 큰 이득을 가져오게 된다.

다만 1/1000 축소의 예와 같이 지도를 너무 과도하게 축소하면 지도의 크기가 너무 작아져서(가로 167, 세로 151) 화면에 지도를 표시하면 관독이 어렵게 되며, 이렇게 해상도가 낮아진 지도를 다시 확대하면 계단 현상이 나타나게 된다. 따라서 교통정보 제공 서비스에 사용할 지도 크기에 맞추어 적절하게 지도 축소 비율을 설정하여야 한다. 웹을 통한 교통정보 제공은 결국 PC의 모니터 화면을 통하여 이루어지는 것이므로 대부분 지도의 해상도가 1280 \* 1024를 넘길 필요가 없으며, 실제로 전체 화면 중에 지도 표출 영역은 이보다 작게 설정되는 것이 일반적이다. 따라서 위에 제시한 예에서는 1/100 또는 1/200으로 축소한 지도가 실제 서비스에 이용하기 적당하며, 이 때 지도 데이터의 크기는 본래 shp 파일의 8~16% 정도로 감소하는 것을 확인할 수 있다.

## 2. 지도 구동 엔진

지도 구동 엔진은 데이터 로더, 사용자 입력 처리기, 그리고 디스플레이 관리자로 구성된다.

데이터 로더는 웹 서버의 지도 데이터 전송기와 속성 데이터 전송기에게 지도 구동 프로그램이 지시한 데이터를 요청하고 그 응답을 해석하여 자료구조를 생성하는 역할을 담당한다. 어도비 플래시는 모

든 도형 객체를 무비클립(movie clip)이라는 객체로 관리하며, 하나의 무비클립을 다른 무비클립에 붙여(attach) 무비클립 간에 부모-자식 관계를 생성할 수 있다. 따라서 전체 지도에 대응되는 루트(root) 무비클립을 먼저 생성하고, 지도 레이어(layer)에 대응되는 무비클립을 루트 무비클립에 붙이며, 각각의 도형 객체에 대응되는 무비클립을 레이어 무비클립에 붙여나가는 것만으로 전체 지도를 하나의 트리(tree) 형태로 구성할 수 있다. 데이터 로더는 이렇게 자료 구조를 생성한 뒤 최초의 디스플레이를 위해 디스플레이 관리자를 호출한다.

여기에서 한 가지 고려할 점은, 교통 정보 표출용 지도에는 백그라운드 레이어(background layer)가 존재한다는 것이다. 백그라운드 레이어는 마우스 이벤트에 반응할 필요가 없고 단순히 지도의 시인성을 높이기 위해 사용되는 배경 그림만을 담고 있는 레이어를 의미한다. 이러한 레이어는 굳이 각각의 도형 객체를 하나의 무비클립으로 생성할 필요가 없으므로 모든 도형을 레이어 무비클립 하나에 그려 넣어 메모리 사용 또는 실행 속도에 이득을 가져올 수 있다.

사용자 입력 처리기는 사용자의 입력을 받아 처리한다. 지도의 확대, 축소, 이동 등을 요구하는 입력이 들어오면 이 요청을 디스플레이 관리자에게 전달하며, 특정 도형 객체에 대한 세부정보를 요청하는 입력이 들어오면 지정된 이벤트 핸들러를 호출함으로써 이러한 역할을 수행한다.

디스플레이 관리자는 지도의 디스플레이를 관리하는 모듈로서 현재 표출된 지도의 범위나 위치, 그리고 스케일 등과 같은 정보를 유지하고 있다가 사용자 입력 처리기로부터 요청이 들어오면 이에 맞게 지도 표출 영역을 변경한다. 어도비 플래시 상에서 무비클립의 확대, 축소, 이동 등은 전역 변수 변경만으로 쉽게 처리할 수 있으므로 디스플레이 관리자는 이에 관련된 변수 값을 유지, 변경하는 적은 양의 코드만으로 쉽게 구현가능하다.

### 3. 지도 구동 프로그램과 시스템 적용

이상과 같이 구현된 시스템을 활용하여 실제로 교

통정보 서비스 시스템을 구축하기 위해서는 해당 사이트에 맞는 지도 구동 프로그램을 작성하여야 한다. 지도 구동 프로그램 역시 플래시 플레이어 상에서 동작하므로 액션스크립트를 이용하여 작성되며, 여기에 기술되어야 하는 내용은 다음과 같다.

- 지도 로딩 명령 : 지도를 구성하는 레이어별 SHP 파일을 지정하고 로딩을 시작하도록 한다.
- 레이어별 표출 속성 지정 : 각 레이어를 표출할 때 사용되는 시각적 속성들을 지정한다. 선의 유형(점선, 실선 등), 폴리곤의 색상, POI 심볼 등이 이에 해당한다.
- 마우스 이벤트 핸들러 작성 : 마우스 커서를 특정 도형에 올려놓는 롤오버(roll over) 또는 마우스 클릭과 같은 동작이 이루어졌을 때 실행될 이벤트 핸들러를 작성한다.
- 실시간 교통정보 업데이트를 위한 이벤트 핸들러 작성 : 교통정보 업데이트 루틴이 실행될 주기를 지정하고 이 때 실행될 이벤트 핸들러를 작성한다.

이러한 내용들은 지도 구동 엔진이 제공하는 API를 이용하여 액션스크립트로 작성되므로 하나의 교통정보 제공 사이트를 구축할 때 작성해야할 소스 코드의 양은 매우 적으며, 이는 본 논문에서 제안된



<그림 2> 제안된 시스템 활용사례-인천국제공항 고속도로

<Fig. 2> An application of the proposed system



<그림 3> 제안된 시스템 활용사례-화성동탄 u-City  
<Fig. 3> An application of the proposed system

시스템을 사용한 사이트들의 구축 과정에서도 확인되었다. 제안된 시스템이 적용된 실제 사이트를 제시하면 <그림 2>, <그림 3>과 같다.

## V. 결 론

최근의 웹 기반 교통정보 제공 시스템은 대부분 벡터 지도와 어도비 플래시를 이용하여 구축되고 있다. 이러한 방식은 별도의 플러그인을 필요로 하지 않아 사용자의 접근성이 높고 미려한 맵을 제공할 수 있다는 장점을 가지기 때문이다. 그러나 이들 시스템은 교통관리센터의 GIS와는 별개로 플래시 전용의 지도를 제작하여 사용하고 있어서 지도에 수정이 필요할 때마다 GIS와 교통정보 표출 전용 지도를 각각 수정하는 이중 작업을 야기하고 있다. 특히, GIS 기반의 교통 및 시설물 관리 시스템이 점차 확산될 것임을 감안할 때 이러한 이중 작업은 시스템의 유지보수에 적지 않은 부담으로 작용하게 될 것이다.

따라서 본 논문에서는 이러한 단점을 극복한 지도 기반의 교통정보 서비스 시스템을 제안하고 그 구현 사례를 제시하였다. 제안하는 시스템은 전자지도와

이를 구동하는 클라이언트 프로그램을 논리적으로 분리함으로써 지도의 변경에 유연하게 대처할 수 있으며, 플래시 액션스크립트를 이용하여 필요한 기능을 손쉽게 확장할 수 있으므로 다양한 목적의 웹 GIS 기반 서비스에도 활용될 수 있을 것이다.

## 참 고 문 헌

- [1] 심재용, 이동훈, 박주영, "Airport GIS 구축을 위한 서비스모델 설계에 관한 연구," *한국ITS학회 논문지*, vol.7, no.3, pp.82-94, 2008. 6.
- [2] 김은형, "u-City를 위한 지자체 GIS 연계통합 연구," *한국GIS학회지*, 제14권, 제3호, 2006. 11.
- [3] 김성호, "플래시 환경에서 래스터 지도를 기반으로 한 반경 검색시스템 개발," *한국콘텐츠학회 논문지*, 제8권, 제4호, 2008. 4.
- [4] 김주환, 양승묵, 남두희, "'Mash-up 기술을 이용한 4D Wall-Map 구성체계," *한국ITS학회 논문지*, vol.8, no.3, pp.27-33, 2009. 6.
- [5] 김수용, 손기욱, "ActiveX Control 취약점 검사 및 검증 기법 연구," *한국정보보호학회논문지*, 제15권, 제6호, 2005. 12.
- [6] Adobe, "Flash Player Version Penetration," [http://www.adobe.com/products/player\\_census/flashplayer/version\\_penetration.html](http://www.adobe.com/products/player_census/flashplayer/version_penetration.html), 2010. 6.
- [7] 이상수, 오영태, 이철기, "PDA기반 교통사고 조사장비 개발에 관한 연구," *한국ITS학회 논문지*, vol.8, no.1, pp.22-29, 2009. 2.
- [8] ESRI, "ESRI Shapefile Technical Description," *ESRI White Paper*, 1998. 7.
- [9] 박우진, 박승용, 유기윤, "건설 도면의 GIS 데이터 변환을 위한 건물 외곽선 단순화 기법 비교 연구," *한국지형공간정보학회 학술대회*, pp.437~444, 2008. 6.



저자소개



**이 성 욱 (Lee, Seong-Uck)**

2003년 ~ 현재 : 신구대학 컴퓨터멀티미디어과 조교수  
2003년 : 아주대학교 컴퓨터공학과 졸업 (공학박사)



**이 철 기 (Lee, Choul-Ki)**

1991년 : 아주대학교 대학원(석사)  
1998년 : 아주대학교 대학원(교통공학박사)  
2000년 : 미국 Texas A&M University TTI(Texas Transportation Institute) Visiting Scholar 과정  
2004년 : 서울지방경찰청 교통개선 기획실장 및 COSMOS 추진 기획단장  
현재 : 아주대학교 교통연구센터 부센터장  
아주대학교 ITS 대학원 교수