

# 개방형 웹 맵 서버의 설계 및 구현

조 대 수<sup>†</sup> · 박 종 현<sup>††</sup>

## 요 약

지금까지 웹을 통한 지리정보의 접근을 위한 웹 맵 서비스에 대한 다양한 연구가 있었다. 기존의 연구에서는 클라이언트 구현 방법, 맵 서버 구현 방법, 통신 방법, 맵 데이터 표현 방법 등 각각의 구현 기술만을 다루고 있다. 따라서, 각각 구현된 웹 맵 서버간에 호환성이 없기 때문에, 지리정보에 대한 공유와 활용이 어려운 문제가 있다. 이 논문에서는 OGC를 통해 제안된 웹 맵 서비스를 위한 표준 인터페이스를 준수하는 개방형 구조의 웹 맵 서버를 설계하고 구현하였다. 특히, 하나의 맵 서버를 통해 하나 이상의 데이터 소스를 지원할 수 있도록 OGC에서 제안한 표준 인터페이스를 확장하였다. 또한, 데이터 소스의 종류에 관계없이 재사용이 가능한 맵 서버의 구현 방법을 제안하고, 이를 구현함으로써, 웹 맵 서비스의 구축 및 활용에 크게 기여하였다.

## Design and Implementation of Open Web Map Server

Dae-Soo Cho<sup>†</sup> · Jong-Hyun Park<sup>††</sup>

### ABSTRACT

Over the past few years, a number of studies have been made on web map services, which enable the GIS user to access spatial data through the web. They, however, have focused on only implementation techniques separately, such as map client implementation, map server implementation, communication between client and server, and the map data representation. Therefore, it is hard to share and practical use the spatial data, because they does not ensure interoperability in heterogeneous map servers. In this paper, we have designed and implemented the web map server with open architecture, which complies with the standard interfaces proposed by OpenGIS Consortium (OGC). In particular, we have extended the OGC's interfaces for a map server to support one or more data sources. This paper has contributed to construction and practical use of web map services by newly proposing the method of implementation of a map server, which could be reused regardless of the types of data sources.

**키워드 :** 웹 GIS(Web GIS), 웹 맵 서버(Web Map Server), 개방형 GIS(Open GIS)

### 1. 서 론

웹 맵 서비스는 웹 브라우저를 사용해서 지리정보를 검색, 획득 할 수 있는 것을 의미하며 대부분 웹 브라우저(맵 클라이언트), 웹 서버, 맵 서버로 구성되는 3계가 간편한 웹 브라우저를 통한 맵 서비스는 GIS 비전문가들도 쉽게 지리정보를 획득할 수 있는 장점을 가지므로, GIS의 활용 분야가 획기적으로 넓어질 수 있기 때문이다.

지금까지의 웹 맵 서비스에 관한 연구는 맵 서버 및 맵 클라이언트를 구현하기 위한 개별적인 방법에 관한 것이었다. 예를 들면, 맵 데이터의 유형에 따라서 래스터 맵과 벡터 맵을 서비스하기 위한 구현 방법이 있다. 래스터 맵을 서비스하는 경우에는 미리 맵을 생성해 두는 방식과 맵 클라이언트의 요청에 따라 동적으로 맵을 생성하는 방식 등 맵 서버를 구현하는 방법에 대한 연구가 있었다. 반면에, 벡터 맵을 서비스 하

는 경우에는, 맵 서버를 구현하는 방법뿐 아니라, 웹 브라우저만으로는 맵 클라이언트를 구현할 수 없으므로 플러그인, ActiveX, 자바 애플릿 등을 활용하여 맵 클라이언트를 구현하는 방법에 대한 연구가 진행되어 왔다.

그런데, 각각의 구현 방법에 따라 구현된 맵 서비스는 상호 운용되지 않는다. 즉, 하나의 맵 클라이언트가 서로 다른 여러 맵 서버에 접속하지 못하기 때문에, 각각의 맵 서버에 대해서, 별도의 맵 클라이언트를 구현해야 한다. 왜냐하면, 맵 서버로부터 맵을 검색, 요청하는 방식, 맵 전송 프로토콜, 벡터 맵의 경우 맵을 인코딩하는 방법 등이 서로 다르기 때문이다. 서로 상호 운용될 수 없는 맵 서비스는 다음과 같은 문제가 있다. 첫째, 맵 서버의 종류에 따라서, 서로 다른 맵 클라이언트가 사용되기 때문에, 웹을 통한 지리정보의 공유 및 활용이 어렵다. 둘째, 유사한 기능의 맵 클라이언트를 중복으로 개발해야 한다. 셋째, 다른 맵 서버의 지리정보를 검색, 요청할 수 없기 때문에, 동일한 지리정보를 각각의 맵 서버에서 중복으로 구축해야 한다. 따라서, 맵 클라이언트, 맵 서버에서

<sup>†</sup> 정 회 원 : 한국전자통신연구원 공간정보기술센터 GIS연구팀 선임연구원  
<sup>††</sup> 정 회 원 : 한국전자통신연구원 공간정보기술센터 GIS연구팀 팀장  
 논문접수 : 2002년 9월 13일, 심사완료 : 2002년 11월 25일

의 지리정보 구축에서의 중복 개발 문제를 해결하고, 웹을 통한 지리정보의 공유 및 활용을 극대화하기 위해서는 맵 서버와 맵 클라이언트간에 공통된 인터페이스가 정의된 개방형 구조의 웹 맵 서비스에 대한 연구가 필요하다.

이 논문은 개방형 웹 맵 서버의 구현을 목적으로 한다. 이 논문에서 제안하는 맵 서버는 크게 네 가지 특징을 갖는다. 첫째, OGC(OpenGIS Consortium)에서 제안하고 있는 웹 맵 서비스를 위한 표준 인터페이스[17]를 따르는 개방형 구조의 맵 서버이다. 표준 인터페이스를 따르는 맵 서버에 대해서는 클라이언트(웹 브라우저)가 동일한 방법으로 맵을 검색, 요청할 수 있다. 즉, 하나의 클라이언트에서 서로 다른 맵 서버에 접근하여 각각 맵을 생성한 후에 동일한 화면에서 출력할 수 있음을 의미한다. 따라서, 웹을 통한 지리정보의 공유가 용이한 장점을 갖는다.

둘째, 하나의 맵 서버가 다중 데이터 소스를 지원할 수 있도록 OGC의 표준 인터페이스를 확장하였다. 데이터 소스는 지리정보를 저장하고 있는 GIS 서버(또는 GIS 데이터베이스)를 의미한다. OGC의 웹 맵 서비스를 위한 표준 인터페이스[17]에서는 맵 서버마다 하나의 데이터 소스가 있는 것으로 가정하고 있다. 따라서, 다중 데이터 소스를 지원하기 위해서는 맵 서버를 여러 개 실행시켜야 하는 문제가 있다. 이 논문에서는 하나의 맵 서버만으로 다중 데이터 소스에 접근할 수 있도록 데이터 소스를 선택할 수 있는 인터페이스를 추가하였다.

셋째, 이 논문에서 구현한 맵 서버는 데이터 소스의 종류에 상관없이 재사용이 가능하다. 개방형 구조의 맵 서버는 표준 인터페이스만을 준수할뿐, 데이터 소스와 연결, 지리정보의 검색 및 획득, 맵 생성 등 내부 구현은 서로 다를 수 있다. 따라서, 특정 데이터 소스에 대해 구현된 맵 서버를 다른 데이터 소스에 적용하기 매우 어려운 문제가 있다. 이 논문에서는 서로 다른 데이터 소스에 대해서 동일한 방법으로 지리 정보를 검색, 획득할 수 있는 OGC 데이터 제공자를 이용하여 맵 서버를 구현함으로써, 데이터 소스의 종류에 상관없이 재사용이 가능하도록 하였다.

넷째, 이 논문에서 구현한 맵 서버는 벡터 맵에 대한 간단하고, 효과적인 인코딩 방법을 적용함으로써, 웹에서 데이터 전송 비용을 줄였다. OGC 표준 인터페이스[17]에서 웹 서버는 JPEG, GIF 등 이미지 포맷의 맵뿐 아니라, SVG, GML 등의 벡터 맵을 지원하고 있다. 그런데, 벡터 맵의 경우 각각의 좌표를 텍스트로 인코딩하기 때문에 맵의 크기가 커지는 문제가 있다. 이 논문에서는 맵 요청시의 영역정보를 활용한 간단한 인코딩 방법을 추가함으로써 벡터 맵의 크기를 줄였다.

이 논문의 구성은 다음과 같다. 2장에서는 관련연구를 살펴보고, 3장에서는 OGC에서 제안하는 웹 맵 서비스 인터페이

스 표준에 대해 살펴본다. 4장에서는 개방형 웹 맵 서버의 설계에 대해서 설명하고, 5장에서는 구현 결과에 대해서 설명한다. 그리고, 6장에서 결론을 맺는다.

## 2. 관련 연구

지리정보 시스템(GIS, Geographic Information System)은 전통적인 클라이언트/서버 구조에서 표준 인터페이스를 갖는 개방형 구조로 발전하고 있다. 기존의 클라이언트/서버 구조에서 각각의 GIS 서버는 자신만의 고유한 인터페이스(Proprietary Interfaces)를 갖고 있으므로, 각 GIS 서버별로 별도의 GIS 클라이언트를 필요로 한다. 따라서, 클라이언트/서버 GIS에서 하나의 클라이언트는 서로 다른 GIS 서버의 공간 데이터에 접근할 수 없기 때문에, 각각의 GIS 수요기관에서 서로 다른 GIS 서버로 구축된 방대한 양의 공간 데이터를 서로 공유하기가 어려운 문제점을 갖는다.

개방형 구조[6, 8, 12, 13]는 하나의 클라이언트에서 서로 다른 여러 서버의 데이터에 접근할 수 있도록, 각 서버에서 표준 인터페이스를 지원하는 데이터제공자를 제공하는 것이다. 즉, 데이터제공자는 서버에 의존적인 프로그램이지만, 클라이언트 측면에서는 서버에 관계없이 동일한 인터페이스를 제공해준다. [8]은 공간(geospatial) 데이터의 상호 운용 문제를 해결하기 위한 도구인 OGD(Open Geospatial Datastore Interface)를 제공하는 것을 목적으로 하고 있다. OGD에서는 응용 프로그램과 데이터 저장소간에 표준 API와 드라이버를 제공하여 문제를 해결하고자 한다. [6]에서는 자사의 여러 GIS 제품간의 데이터 상호 운용을 위해 개방형 구조의 공간 데이터베이스 엔진을 제안하고 있다. 그러나, [6]과 같이 특정회사에서 제안하는 방식은 자사의 제품만을 대상으로 하기 때문에 타사의 GIS 제품간에는 상호 운용이 되지않는 문제가 있다. 이와 같은 문제를 해결하기 위해서, 여러 GIS 업체로 구성된 OpenGIS 컨소시엄이 생겼으며, 공간 데이터 상호 운용을 위한 인터페이스 명세[12, 13]를 제안하고 있다.

현재까지 웹을 통한 맵 서비스에 관한 연구[1, 7, 22]는 대부분 웹에서의 맵 서비스 구현 방법에 관한 것으로서, 클라이언트 구현 방법, 맵 서버 구현 방법, 통신 방법, 맵 데이터 표현 방법 등에 따라 다양한 구조의 맵 서비스가 제안되었다.

현재 웹을 통한 맵 서비스를 위한 각각의 구현 기술들은 GIS S/W 개발사를 통해 상품화[2, 5, 9, 22]에 이르는 수준까지 발전하였으나, 웹 맵 서비스에서는 아직까지 개방형 구조의 맵 서버에 대한 연구가 활발히 이루어지고 있지 않다. 즉, 맵 서비스별로 클라이언트와 맵 서버는 각각의 고유한 인터페이스를 가지고 연결되어 있다. 따라서 기존에 연구된 웹 맵 서버는 상호 운용되지 않기 때문에, 각각의 맵 서버와 통신하기 위한 별도

의 맵 클라이언트를 각각 구현해야 하는 문제가 있다. 맵 클라이언트의 중복 개발 문제를 해결하고, 여러 맵 서버가 가지고 있는 지리정보의 공유 및 활용을 극대화하기 위해서는 개방형 구조의 맵 서버에 대한 연구가 필요하다. 최근 OGC를 웹 맵 서버에 대한 표준 인터페이스 명세[18]가 정의됨에 따라 웹 맵 서비스에서도 개방형 구조의 맵 서버에 대한 연구가 활발히 진행되고 있다.

<표 1> 웹 맵 서비스 기술의 분류

분 류	내 용
클라이언트 구현 방법	웹 브라우저 자체(이미지 맵 이용시), Java 애플릿, ActiveX 컨트롤, Plug-in 등
맵 서버 구현 방법	CGI, ASP(Active Server Page), 웹서버 API(NSAPI, ISAPI), 응용 서버, CORBA의 서버 객체 등
통신 방법	HTTP, JDBC, OLE/DB, CORBA IIOP, SOAP, TCP/IP 등
맵 데이터 표현 방법	이미지(JPEG, GIF, TIFF, ...), XML, SVG, ActiveCGM, GML 등

이 논문은 <표 1>에서 분류한 웹 맵 서비스를 구현하기 위한 세부적인 기술을 다루지 않는다. 대신에, 서로 다른 맵 서버간의 상호 운용성을 보장하기 위한 개방형 구조의 웹 맵 서버의 설계 및 구현을 다룬다. 이 논문에서 설계한 웹 맵 서버는 3장에서 설명할 OGC의 웹 맵 서비스 인터페이스 표준을 기반으로, 다중 데이터 소스를 지원하며, 재사용이 용이한 특징을 갖는다.

### 3. 웹 맵 서비스 인터페이스 표준

이 장에서는 OGC의 웹 맵 서비스 인터페이스 표준에 대해 설명한다. 이 논문에서 구현한 개방형 맵 서버는 OGC의 표준 인터페이스를 준수한다. OGC의 표준 인터페이스는 맵을 요청(request)하기 위한 연산자에 대한 표준과, 요청에 대한 응답(response)으로써 제공되어야 할 맵의 형식, 그리고 오류 발생시 전달되는 문서의 형식에 대한 표준을 제시하고 있다. 웹 브라우저는 표준 인터페이스를 준수하는 모든 맵 서버에 대해서 동일한 방법으로 지리정보를 획득할 수 있으므로, 표준 인터페이스를 따르는 맵 서버는 지리정보의 공유를 용이하게 하는 장점을 갖는다.

(그림 1)에서 보이는 Portrayal 모델[17]에 따르면 OGC에서는 지리정보의 형식을 크게 세가지로 분류하고 있다. 첫째는 데이터 소스에서 여과(Filter)를 거쳐 얻은 피쳐(feature) 데이터이다. 피쳐 데이터는 OGC의 심플 피쳐 명세서 [16]에 따라 정의되는 벡터 기반의 데이터로써, 화면 출력을 위해 필요한 스타일 정보는 가지고 있지 않다. 둘째는 피쳐에 스타일 정보를 추가하여 생성한 출력 요소(display element) 데이터

이다. 그리고, 마지막으로는 출력 요소를 렌더링하여 생성한 이미지 데이터이다.

(그림 1) Portrayal 모델[17]

OGC의 웹 맵 서비스 인터페이스 버전 1.0.0[17]에서는 세 가지 형태의 데이터를 모두 지원하고 있으나, 버전 1.1.0[18]에서는 출력 요소 데이터와 이미지 데이터만을 지원하고 있다. 대신에 피쳐 데이터는 웹 피쳐 서비스[15]로 분리해서 다루고 있다. 즉, 맵 서비스는 지리정보의 화면 출력을 위해 맵을 제공하는 것으로, 피쳐 서비스는 공간 분석 등 복잡한 공간 연산과 맵 데이터의 갱신을 지원하기 위해 사용되는 피쳐 데이터를 제공하는 것으로 구분되고 있다. 이 논문에서 구현한 맵 서버는 버전 1.0.0을 기반으로 구현함으로써, 맵 서버에서 세가지 유형의 지리정보를 모두 처리할 수 있게 하였다.

웹 맵 서버에 대한 일반적인 요청은 <표 2>과 같이 URL 형태를 가지며, REQUEST 매개변수에 의해 세가지 인터페이스로 구분된다.

<표 2> 맵 서버에 대한 일반적인 요청을 위한 URL

URL 요소	설 명
http://server_address/path/script?	맵 서버의 URL
WMTVER = 1.0.0	맵 서버의 버전
REQUEST = <name>	세 가지 인터페이스를 구분
Additional parameters	각 인터페이스별로 정의된 매개변수
Vender-specific parameters	확장 가능한 추가의 매개변수

OGC에서는 맵 서버에 대해서 “capabilities”, “map”, “feature\_info”의 세 가지 인터페이스를 정의하고 있다.

- (1) REQUEST = capabilities : 맵 서버가 제공할 수 있는 기능에 대한 정보를 요청한다.
- (2) REQUEST = map : 맵 서버에 대해서 맵을 요청한다.
- (3) REQUEST = feature\_info : 맵 상에 있는 특정 피쳐에 대한 상세 정보를 요청한다.

인터페이스에 따라서 요청 URL의 매개변수가 정의되어 있으나, REQUEST = capabilities의 경우에는 추가의 매개변수가 존재하지 않는다. <표 3>은 REQUEST = map인 경우에 URL 매개변수를 보이고 있다. 맵 요청을 위해 필수적인 매개변수로는 LAYERS, STYLES, SRS, BBOX, WIDTH, HEIGHT, FORMAT이 있다. 이 중에서 FORMAT 매개변수는 지리정보의 형식을 의미하며, 피쳐 데이터 형식으로는 GML, 출력 요소 데이터 형식으로는 SVG, WebCGM, 이미지 형식으로는 GIF, JPEG, PNG, TIFF, GeoTIFF, PPM, WBMP 등이 있다.

<표 3> REQUEST = map인 경우 URL 매개변수

URL 요소	설 명
http://server_address/path/script? WMTVER = 1.0.0	맵 서버의 URL 맵 서버의 버전
REQUEST = map	맵요청
LAYERS = layer_list STYLES = style_list SRS = srs_identifier BBOX = xmin,ymin,xmax,ymax WIDTH = output_width HEIGHT = output_height FORMAT = output_format	(필수) 레이어 리스트 (필수) 스타일 리스트 (필수) 공간참조체계 (필수) 요청하는 맵의 영역 (필수) 요청하는 맵의 너비 (필수) 요청하는 맵의 높이 (필수) 지리정보의 형식
TRANSPAENT = true_or_false BGCOLOR = color_value EXCEPTIONS = exception_format Vender-specific parameters	(선택) 투명 (선택) 배경 색 (선택) 예외 정보의 형식 (선택) 확장 가능한 추가의 매개변수

<표 4>은 REQUEST = feature\_info인 경우에 URL 매개변수를 보이고 있다. 피쳐에 대한 상세 정보를 보기 위해 필수적인 매개변수로는 맵 요청시의 매개변수 복사본(<map request copy>), 맵 요청시 매개변수 중에서 WMTVER, REQUEST를 제외한 모든 매개변수, QUERY\_LAYERS, X, Y가 있다. 맵 요청시의 매개변수 복사본이 필요한 이유는 HTTP 프로

<표 4> REQUEST = feature\_info인 경우 URL 매개변수

URL 요소	설 명
http://server_address/path/script? WMTVER = 1.0.0	맵 서버의 URL 맵 서버의 버전
REQUEST = feature_info	피쳐 정보 요청
<map request copy>	맵 요청 매개변수 복사
QUERY_LAYERS = layer_list X = pixel_column Y = pixel_row	(필수) 피쳐 정보가 요청할 레이어 리스트 (필수) 피쳐의 X좌표 (픽셀) (필수) 피쳐의 Y좌표 (픽셀)
INFO_FORMAT = output_format FEATURE_COUNT = number Vender-specific parameters	(선택) 피쳐 정보의 리터 포맷 (선택) 리턴될 피쳐의 수 (선택) 확장 가능한 추가의 매개변수

토콜이 상태(state)를 갖지 못하기 때문이다. 즉, 이전에 요청했던 맵에 대한 상태를 맵 서버가 모르기 때문에, 맵 요청시의 매개변수를 다시 보내야 한다. QUEYR\_LAYERS는 맵 요청시 LAYERS로 기술된 레이어 리스트 중에서 상세 정보를 요청하려는 피쳐의 레이어를 기술한다. X, Y는 피쳐의 X, Y좌표를 의미하며, 맵 요청시 WIDTH, HEIGHT로 기술된 맵에 대해서 좌측 상단을 (0, 0)으로 우측 하단을 (WIDTH, HEIGHT)로 표시한다.

(그림 2)는 웹 맵 서비스 인터페이스의 사용 예를 보이고 있다. (그림 2)(a)는 “capabilities” 인터페이스 사용 예로써, 해당 맵 서버가 제공할 수 있는 기능을 명시한 XML 문서를 요청하는 것이다. (그림 2)(b)는 “map” 인터페이스 사용 예로써, EPSG : 4301 공간참조체계에서 ((309813, 536478) (326031, 549310)) 영역내에 존재하는 SP13, SC13, KN13 레이어의 피쳐에 대해서 기본(default) 스타일을 사용하여 렌더링 하여 600 × 530 크기로 생성한 JPEG 형태의 이미지 파일을 요청하는 것이다. (그림 2)(c)는 (그림 2)(b)으로 요청한 맵에서 좌측 상단을 (0, 0)으로 할 때, (200, 250)에 해당하는 SP13 레이어의 피쳐에 대한 상세 정보를 요청하는 것이다.

http://129.254.75.50/cgi-bin/wms.exe? WMTVER = 1.0.0 & REQUEST = capabilities  
(a) capabilities 인터페이스

http://129.254.75.50/cgi-bin/wms.exe? WMTVER = 1.0.0 & REQUEST = map & LAYERS = SP13, SC13, KN13 & STYLES = default, default, default & SRS = EPSG : 4301 & BBOX = 312800, 539413, 322570, 548910 & WIDTH = 600 & HEIGHT = 530 & FORMAT = JPEG

(b) map 인터페이스

http://129.254.75.50/cgi-bin/wms.exe? WMTVER = 1.0.0 & REQUEST = feature\_info

& LAYERS = SP13, SC13, KN13 & STYLES = default, default, default & SRS = EPSG : 4301 & BBOX = 312800, 539413, 322570, 548910 & WIDTH = 600 & HEIGHT = 530 & FORMAT = JPEG &

QUERY\_LAYERS = SP13 & X = 200 & Y = 250

<map request copy>

(c) feature\_info 인터페이스

(그림 2) 웹 맵 서비스 인터페이스 사용 예

#### 4. 개방형 웹 맵 서버의 설계

이 장에서는 OGC의 표준 인터페이스를 따르며, 다중 데이터 소스를 지원하기 위한 개방형 웹 맵 서버의 구조에 대해서 설명한다. (그림 3)은 이 논문에서 제안하는 웹 맵 서비스의 구조이며, 크게 웹 브라우저, 웹 서버, 맵 서버, OGC 데이터제공자, 데이터 소스로 구성된다. OGC 데이터제공자는 데이터 소스의 종류에 상관없이 동일한 방법으로 지리정보를 검색, 획득하기 위해 OGC에서 정의한 심플 피쳐를 위한 표준 인터페이스[16]를 지원하는 컴포넌트를 의미한다.

(그림 3) 웹 맵 서비스 구조

4.1 개방형 구조

이 논문에서 제안하는 개방형 웹 맵 서버의 구조는 (그림 4)와 같다. 개방형 구조의 맵 서버는 웹 브라우저에서 동일한 방법으로 맵을 획득할 수 있도록 웹 맵 서비스를 위한 표준 인터페이스를 제공한다. 따라서 웹 브라우저는 서로 다른 웹 맵 서버와의 상호 운용이 가능하다. (그림 3)에서 웹 브라우저는 웹 맵 서비스를 위한 표준 인터페이스[17]에 따라서 구현된 서로 다른 세 개의 맵 서버 MS-1, MS-2, MS-3에 동일한 방법으로 접근할 수 있다.

웹 브라우저는 웹 서버를 통하여 맵 서버에 요청 URL을 보내고, 맵 서버는 HTML 또는 XML 형태로 그 결과를 넘겨준다. 맵 서버에 대한 요청 URL은 맵 서버의 종류에 관계없이 동일하다. 요청 URL은 2장에서 살펴본 바와 같이 REQUEST 매개변수의 값에 따라서 “capabilities” 요청, “map” 요청, “feature\_info” 요청으로 나뉜다. 각각의 요청 URL을 해석하고, 각 요청을 처리하는 것은 CGI 프로그램에서 담당한다. 그리고, 맵 요청은 이미지, 출력요소, 피쳐 데이터와 같이 세가지 지리정보 유형을 처리해야 하기 때문에 매우 복잡해지므로, 별도의 맵 생성 컴포넌트에서 담당한다. 스타일 설정 컴포넌트는 “capabilities”와 “map” 요청에서 사용되는 각 레이어에 대한 스타일 정보를 설정하는 역할을 담당한다. 연결관리 컴포넌트는 데이터 소스와의 연결 정보를 관리하는 것으로, 4.2절에서 다시 설명한다.

(그림 4) 맵 서버의 구성

CGI 프로그램은 요청 URL에서 “REQUEST = map”인 경우에 맵 생성 컴포넌트를 호출하게 된다. 맵 생성 컴포넌트는 맵 요청 URL에서 데이터 소스(DataSource, 4.2절에서 설명함), 레이어(LAYERS), 공간참조체계(SRS), 맵의 영역(BBOX) 매개변수에서 명시된 내용에 따라서 사용자가 요청한 지리정보를 획득한 후에, “FORMAT = output\_format”에서 명시된 형식으로 맵을 생성한다. 이 논문에서는 이미지 데이터 형식으로는 JPEG, 출력 요소 데이터 형식으로는 SVG, 피쳐 데이터 형식으로는 GML을 지원한다.

맵 서버에서 각 레이어 별 스타일 정보는 다음의 경우에 사용된다. 첫째, 맵 서버가 “capabilities” 요청에 대한 응답으로써 Capabilities 문서를 생성할 때 레이어별 스타일 정보를 참조한다. (그림 5)는 “capabilities” 인터페이스를 통해 맵 서버로부터 얻은 Capabilities 문서의 예를 보이고 있다. Capabilities 문서에는 레이어의 이름과 각 레이어에 대해서 적용 가능한 스타일 이름도 명시되어 있다. 예를 들어, (그림 5)에서 KN 13 레이어는 Style 1, Style 2, Style 3의 세 가지 스타일을 지원함을 보여주고 있다. 따라서, 맵 서버는 Capabilities 문서를 생성하기 위해서, 레이어 이름은 OGC 데이터 제공자를 통해 데이터 소스에서 얻어오고, 스타일 이름은 스타일 설정 컴포넌트를 통해 정의된 스타일 정보에서 얻어오게 된다.

둘째, 맵 생성 컴포넌트가 맵을 생성할 때 스타일 정보를 참조한다. 예를 들어, 사용자가 KN 13 레이어를 Style 1 스타일로 요청한다면, 맵 생성 컴포넌트는 Style 1 스타일에 대한 스타일 정보(채우기 색, 선 색, 선 굵기)를 참조해야만 한다. 이

논문에서는 각 레이어에 대한 스타일 정보를 설정하기 위한 스타일 설정 컴포넌트를 맵 서버의 기본 기능으로 설계하였다.

(그림 5) Capabilities 문서 예

#### 4.2 다중 데이터 소스

이 논문에서 제시하는 웹 맵 서비스 구조(그림 3)에서 맵 서버 MS-2는 서로 다른 세 개의 데이터 소스 DS-A, DS-B, DS-C에 접속하고 있다. 사용자는 하나의 맵 서버에 접속해서 여러 개의 데이터 소스로부터 지리 정보를 획득할 수 있다. 즉, 맵 서버가 다중 데이터 소스를 지원한다.

그런데, 웹 맵 서비스를 위한 표준 인터페이스[17]에 의하면, 다중 데이터 소스를 구별하기 위한 인터페이스를 지원하지 않는다. 왜냐하면, OGC 표준 인터페이스에서는 하나의 맵 서버에 대해서 하나의 데이터 소스가 연결되어 있음을 가정하고 있기 때문이다. 즉, (그림 2)의 맵 요청 URL에서 보이는 바와 같이 레이어와 영역만을 선택하여 지리정보를 검색하고 있다. 따라서, 하나의 맵 서버가 다중 데이터 소스를 지원하기 위해서는 OGC의 표준 인터페이스의 확장이 필요하다.

이 논문에서는 다중 데이터 소스를 지원하기 위해 다음 두 가지 기능을 맵 서버에 추가하였다. 첫째, 맵 서버와 다중 데이터 소스와의 연결 정보를 관리하기 위해서 연결관리 컴포넌트를 추가하였다. 연결관리 컴포넌트는 맵 서버와 연결되어 있는 데이터 소스에 대한 리스트를 관리하며, 리스트에는 데이터 소스와의 연결정보가 기록되어 있다. (그림 6)은 연결관리 컴포넌트에 의해 관리되는 다중 데이터 소스에 대한 연결관리 리스트의 예를 보이고 있다. (그림 6)에서 현재의 맵 서버는 서로 다른 두 개의 데이터 소스 DS-A, DS-B를 지원함을 알 수 있다. 데이터 소스 DS-A에 대해서는 ZEUS 데이터베이스와의 접속 정보가, 데이터 소스 DS-B에 대해서는 SHAPE 파일과의 접속 정보가 기록되어 있다.

둘째, 이 논문에서는 사용자가 다중 데이터 소스에서 하나의 데이터 소스를 선택할 수 있도록 맵 서버에 대한 요청 URL에 DataSource 매개변수를 추가하였다. 예를 들어, 맵 요청 URL에 "DataSource = DS-A"가 명시된 경우에는 클라이언트가 요청한 지리정보를 데이터 소스 DS-A로부터 가져와서 맵을 생성한다. 이 논문에서 구현한 맵 서버는 OGC 표준 인터페이스를 준수하는 다른 맵 서버와의 호환성을 유지할 수 있도록 확장된 매개 변수 DataSource를 선택적으로 사용할 수 있도록 하였다. 즉, 하나의 맵 서버가 다중 데이터 소스를 지원할 경우에, 여러 데이터 소스 중에 하나를 기본 데이터 소스(연결관리 리스트에서 가장 먼저 기록된 데이터 소스으로써, (그림 6)에서는 DS-A가 기본 데이터 소스임)로 지정해 두고, DataSource 매개변수를 사용하지 않은 경우에는 기본 데이터 소스로부터 지리정보를 획득한다.

#### 4.3 재사용성

이 논문에서 제안한 맵 서버(그림 4)는 크게 4가지 컴포넌트로 구성된다.

- (1) 맵 서비스 인터페이스를 처리하기 위한 CGI 프로그램
- (2) 맵의 스타일을 설정하고, 저장하기 위한 스타일 설정 컴포넌트
- (3) 맵 생성을 담당하는 맵 생성 컴포넌트
- (4) 데이터 소스와의 연결을 담당하는 연결관리 컴포넌트

이 중에서 연결관리 컴포넌트를 제외한 맵 서버의 구현은 데이터 소스와 아무런 관련이 없기 때문에, 프로그램의 재사용이 가능하다. 그런데, 이 논문에서는 맵 서버가 데이터 소스와 직접 연결하는 대신에, OGC 데이터제공자를 통해 간접적으로 연결하도록 설계하였다. 즉, 연결관리자 컴포넌트는 서로 다른 인터페이스를 제공하는 각각의 데이터 소스와 직접 연결하는 대신에 표준 인터페이스를 지원하는 개방형 구조의 OGC 데이터제공자를 통해 간접적으로 연결하기 때문에, 연결관리자 컴포넌트 또한 데이터 소스의 종류에 관계없이 구현될 수 있으며, 재사용될 수 있다. 결과적으로 이 논문에서

(그림 6) 다중 데이터 소스 연결관리 리스트 예

제안한 맵 서버 내의 모든 컴포넌트는 데이터 소스의 종류에 관계없이 동일하게 사용될 수 있으므로, 한 번 구현된 맵 서버는 기능이 바뀌지 않는다면, 다른 데이터 소스를 위한 맵 서버로 소스 코드의 수정 없이 재사용 될 수 있다. 따라서, 이 논문에서 제안하고 구현한 맵 서버를 이용하면, 웹 맵 서비스를 쉽게 구축할 수 있다.

#### 4.4 벡터 맵 압축

웹 맵 서버는 JPEG, GIF 등의 이미지 맵뿐 아니라, SVG, GML 등의 벡터 맵도 지원한다. 벡터 맵의 경우 각 지리정보의 좌표를 텍스트로 표현하기 때문에, 맵의 크기가 커지는 문제가 있다. 이 논문에서는 맵 요청 URL에 포함된 BBOX(<표 2> 참조)를 이용하여, 실제 좌표를 BBOX에서의 상대좌표로 변환함으로써, 벡터 맵의 크기를 줄인다.

지리정보의 실제 좌표를 요청 URL의 매개변수인 BBOX에 대한 상대 좌표로 바꾸는 것은 다음과 같은 장점이 있다. 첫째, BBOX의 값은 웹 브라우저에서 설정한 것이므로, 맵 요청에 대해서 맵 서버로부터 받은 맵의 실제 좌표를 다시 계산할 수 있다. 둘째, 단순한 좌표 변환 노력에 비해서 맵 데이터의 크기를 크게 줄일 수 있으므로, 맵 전송시간을 줄임으로써, 사용자 응답시간을 크게 개선할 수 있다.

예를 들어, (그림 2)(b)의 맵 요청의 경우에 BBOX는 ((312800, 539413), (322570, 548910))으로써, 6자리의 좌표를 갖는 벡터 맵이 리턴된다. 이 경우에 BBOX에 대한 상대 좌표는 ((0, 0), (322570 - 312800 = 9770, 548910 - 539413 = 9497))이므로 4자리의 상대 좌표를 갖는 벡터 맵을 생성할 수 있다. 즉, 간단한 좌표 변환 연산으로 전체 벡터 맵의 크기를 1/3 줄일 수 있음을 알 수 있다.

이 논문에서는 사용자가 BBOX에 대해서 상대 좌표를 갖는 맵을 요청할 수 있도록, 맵 서버에 대한 요청 URL에 IsRelative 매개 변수를 추가하였다. IsRelative = TRUE인 경우에는 상대 좌표를 갖는 벡터 맵이 리턴되고, 매개변수 IsRelative를 사용하지 않을 경우에는 OGC 표준 인터페이스에 따라 실제 좌표를 갖는 벡터 맵이 리턴된다.

### 5. 구현 결과

이 장에서는 이 논문에서 제안한 개방형 웹 맵 서버에 대한 구현 결과를 보인다. 먼저, 데이터 소스와 연결을 설정하기 위한 연결관리 컴포넌트와 각 데이터 소스에 포함된 레이어에 대해서 스타일을 설정하기 위한 스타일 설정 컴포넌트를 보인다.

(그림 7)은 연결관리 컴포넌트와 스타일 설정 컴포넌트를 실행시킨 화면의 예를 보이고 있다. 연결관리 컴포넌트에서는 다중 데이터 소스를 등록, 편집할 수 있는 기능을 갖는다. 데이터 소스에 접속하여 스타일 설정 컴포넌트를 실행하면,

(그림 7) 데이터 소스 및 스타일 설정 컴포넌트 실행 예

해당 데이터 소스에 저장된 레이어 리스트를 보여준다. 스타일 설정 컴포넌트에서는 각 레이어에 대해서 여러 가지 스타일을 설정할 수 있다. 연결관리 컴포넌트에 의해 설정된 연결관리 리스트((그림 6) 참조)는 별도의 문서로 저장되며, 스타일 설정 컴포넌트에 의해 설정된 스타일 정보는 Capabilities 문서에 저장된다.

구현된 맵 서버에 대해서 맵을 요청하기 위해서는 (그림 2)(b)의 맵 요청 예와 같이 복잡한 URL을 사용해야 한다. (그림 8)은 세 개의 레이어 KN 13, SC 13, SP 13에 대해서 기본(default) 스타일을 적용하여 (a) JPEG 파일, (b) SVG 파일, (c) GML 파일로 맵을 생성하기 위한 요청 URL과 그 결과를 보이고 있다. 매우 긴 내용의 URL을 직접 입력하는 것은 매우 복잡한 작업이며, 입력 시 오류 발생의 확률이 높은 문제가 있다. 그러나, 실제 응용에서는 사용자가 GUI 인터페이스에서 조건을 선택하면, 응용에서 요청 URL을 자동으로 생성해주는 방법이 사용될 것이다.

이 논문에서는 맵 서버에 대한 요청 URL을 쉽게 생성하기 위한 사용자 인터페이스를 (그림 9)와 같이 구현하였다. 홈페이지의 구성은 맵을 출력하기 위한 윈도우와 접속된 맵 서버에 대한 정보를 보여주기 위한 윈도우, 맵을 요청하기 위한 버튼 등으로 구성되어 있다. (그림 9)와 같은 홈 페이지에서 사용자는 데이터 소스, 레이어, 스타일, 영역 등의 정보를 대화식으로 입력할 수 있다. 원하는 조건의 입력을 완료한 후에 UPDATE 버튼을 누르면 (그림 8)의 맵 요청 URL을 자동으로 생성하여 맵 서버에 보내게 된다. 맵 서버로부터 맵 요청에 대한 결과를 받은 후 그 결과를 화면에 출력하게 된다.

#### (그림 9) 맵 서버와의 인터페이스를 위한 홈 페이지

이 논문에서 구현한 개방형 구조의 맵 서버를 이용한 웹 맵 서비스는 기존의 고유한 인터페이스로 서로 연결된 클라이언트/서버 구조의 맵 서비스에 대해서 다음과 같은 특징을 갖는다. 첫째, 맵 서버의 성능에서는 기존의 방법에 비해 떨어지는 단점을 갖는다. 왜냐하면, 지리정보를 저장하고 있는 데이터 소스별로 가지는 특화된 기능을 활용하지 못하기 때문이다. 즉, 개방형 웹 맵 서버는 데이터 소스의 종류에 관계없이 동일한 기능만을 제공하고 있습니다. 둘째, 서로 다른 데이터 소스에 대해서 동일한 인터페이스를 갖는 맵 서버를 구현하였기 때문에, 하나의 맵 클라이언트를 통해 서로 다른 맵 서버에 접속할 수 있는 장점이 있다. 반면에, 기존의 고유한 인터페이스로 서로 연결된 클라이언트/서버 구조의 경우에는 새로운 맵 서버에 대해서 별도의 클라이언트를 추가적으로 개발해야 하는 문제가 있다. 셋째, 이 논문에서 제안한 맵 서버는 재사용성을 고려하여 설계, 구현하였기 때문에, 서로 다른 데이터 소스에 대해서 소스 코드의 수정 없이 재사용이 가능한 장점이 있다.

또한 이 논문에서 제안한 개방형 구조의 맵 서버는 OGC에서 제안한 개방형 구조의 맵 서버에 대해서도 다음과 같은



특징을 갖는다. OGC에서 제안한 인터페이스를 갖는 맵 서버는 하나의 데이터 소스만을 지원할 수 있다. 예를 들어, 한 사이트에서 서로 다른 두 개의 데이터 소스(ESRI사의 SDE, 한국통신데이터의 ZEUS)에 지리 정보가 저장되어 있을 경우에, 동일한 기능의 맵 서버 두 개를 가지고 있어야 하며, 클라이언트에서 서로 다른 CGI를 통해서 각각의 데이터 소스에 저장되어 있는 지리정보를 접근해야 하는 문제가 있다. 반면에 이 논문에서 제안한 맵 서버는 다중 데이터 소스를 지원할 수 있으므로, 하나의 맵 서버만을 운용하여 두 개의 데이터 소스를 관리할 수 있는 장점을 가지고 있다.

### 6. 결 론

다양한 정보 시스템 분야에서 정보의 공유 및 활용의 수단으로써 웹이 이용되고 있다. 최근 GIS 분야에서도 웹을 통한 맵 서비스에 대한 요구가 증가하고 있으며, 맵 서버를 구현하기 위한 많은 시도가 이루어지고 있다. 웹을 통한 지리정보의 공유와 활용을 극대화하기 위해서는 각각 구현되고 있는 맵 서버간의 상호 운용이 보장되어야 하므로, 표준 인터페이스를 따르는 개방형 구조의 맵 서버의 구현이 요구되고 있다.

이 논문에서는 OGC의 웹 맵 서비스 인터페이스 표준을 준수하는 개방형 구조의 맵 서버를 구현하였다. 이 논문에서 구현한 맵 서버는 이미지, 출력 요소, 피쳐 데이터 유형을 모두 지원하며, 다중 데이터 소스를 지원하기 위해서 OGC의 표준 인터페이스를 확장하였다. 또한 OGC 데이터제공자를 사용함으로써, 서로 다른 데이터 소스에 대해서 동일한 인터페이스를 통해 지리정보를 검색, 획득할 수 있다.

이 논문에서 구현한 맵 서버는 재사용이 가능하다는 특징을 갖는다. 재사용이 가능한 이유는 웹 브라우저와 맵 서버간의 인터페이스와 맵 서버와 데이터 소스간의 인터페이스가 모두 표준을 따르고 있기 때문이다. 따라서 이 논문은 맵 서버 구현의 중복을 제거함으로써, 맵 서버 구축을 용이하게 하고, 웹을 통한 지리정보 서비스의 폭 넓은 활용의 계기가 될 것으로 기대한다.

### 참 고 문 헌

- [1] A. P. Steinke, "Developing Geographic Services on the World Wide Web," ASIA-PACIFIC WORLD WIDE WEB '95 CONFERENCE, 1995.
- [2] Autodesk MapGuide, URL: <http://usa.autodesk.com/adsk/section/0,939487-123112,00.html>.
- [3] B. Plewe, "A Primer on Creating Geographic Services," GIS World Magazine, pp.56-58, 1996.
- [4] D. Crossley and T. Boston, "A Generic Map Interface to Query Geographic Information Using the World Wide Web," Fourth International World Wide Web conference, 1995.
- [5] ESRI MapObjects, URL: <http://www.esri.com/software/mapobjects/overview.html>.
- [6] ESRI, "Spatial Database Engine," An ESRI White Paper, April, 1998.
- [7] K. E. Foote and A. P. Kirvan, "WebGIS, NCGIA Core Curriculum in GIScience," posted, <http://www.ncgia.ucsb.edu/giscc/units/u133/u133.html>, July, 1998.
- [8] F. T. Fonseca and C. A. Davis Jr., "Using the Internet to Access Geographic Information : An Open GIS Interface Prototype," Conf. and Workshop on Interoperating Geographic Information Systems, 1997.
- [9] GeoMedia WebMap, URL: <http://www.intergraph.com/gis/support/GeoMediaWebMap.asp>.
- [10] J. P. Alexander and V. J. Warwick, "Writing GIS Applications for the WWW," Proceedings of the 1997 ESRI User Conference, 1997.
- [11] J. Ferraiolo (ed.), "Scalable Vector Graphics (SVG) 1.0 Specification," W3C Proposed Recommendation, 2001.
- [12] LAS Inc., OpenGIS Features for Internet Implementation Specification, OpenGIS Project Document 97-026, August, 1997.
- [13] Open GIS Consortium, Inc., The OpenGIS Abstract Specification Model Version 3, 1998.
- [14] OpenGIS Consortium Inc., OpenGIS Discussion Paper # 01 - 022r1 : Basic Service Model Draft Candidate Implementation Specification 0.0.8, 2001.
- [15] OpenGIS Consortium Inc., OpenGIS Discussion Paper # 01 - 023 : Web Feature Service Draft Candidate Implementation Specification 0.0.12, 2001.
- [16] OpenGIS Consortium Inc., The OpenGIS Simple Feature Specification for OLE/COM Revision 1.1, 1999.
- [17] OpenGIS Consortium Inc., Web Map Server Interface Implementation Specification 1.0.0, 2001.
- [18] OpenGIS Consortium Inc., Web Map Service Implementation Specification 1.1.0, 2001.
- [19] R. Laurini and D. Thomason, "Fundamentals of Spatial Information Systems," Academic Press, p.680, 1992.
- [20] R. East, R. K. Goyal, Art Haddad, Alexander Kononov, Andrea Rosso, Mike Tait, Jay Theodore, "The Architecture of ArcIMS, a Distributed Internet Map Server," SST2 2001, pp.387-403, 2001.
- [21] Z. R. Peng, "An Assessment of the Development of Internet GIS," Proceedings of the 1997 ESRI User Conference, 1997.
- [22] MapInfo MapExtream, URL: <http://dynamo.mapinfo.com/miproducts/Overview.cfm?productid = 1162>.

### 조 대 수

e-mail : junest@etri.re.kr

1995년 부산대학교 컴퓨터공학과(학사)

1997년 부산대학교 컴퓨터공학과(공학석사)

2001년 부산대학교 컴퓨터공학과(공학박사)

2001년~현재 한국전자통신연구원 공간정보

기술센터 GIS 연구팀 선임연구원

관심분야 : GIS, 공간DB, LBS, 이동체DB 등

### 박 종 현

e-mail : jhp@etri.re.kr

1989년 경희대학교 우주과학과(학사)

1991년 연세대학교 천문기상학과(이학석사)

2000년 University of Chiba 환경원격탐사  
학과(공학박사)

1991년~1996년 KIST 시스템공학연구소  
연구원

1996년~2000년 Center of Environmental Remote Sensing,  
Chiba University, Research Associate

2000년~현재 한국전자통신연구원 공간정보기술센터 GIS  
연구팀장

관심분야 : LBS, Data Fusion, Remote Sensing 등