

연구논문

Open Source GIS를 이용한 고해상도 영상의 Tile Map Service 시스템 구축에 관한 연구

A Study on Tile Map Service of High Spatial Resolution Image Using Open Source GIS

정명훈* · 서용철**

Jeong, Myeong Hun · Suh, Yong Cheol

요 旨

Tile Map Service는 정적인 이미지들을 모아둔 저장소를 이용하여 매우 빠르게 지도 서비스를 수행하는 서비스이다. 지도 이미지 저장소는 특정 축척별로 지도 영역을 Tile 이미지로 저장한 디렉토리 구조이다. 따라서 디렉토리 구조로부터 이미지를 불러오는 것은 사용자의 요구에 따라서 지도를 데이터베이스로부터 직접 생성하는 것보다 훨씬 시간 단축이 이루어진다. 이와 같은 Tile Map Service를 이용하면 사용자가 복잡한 지도를 생성할 때 획기적으로 지도 생성 시간을 단축시키는 효과와 성능 향상을 위한 제반 노력을 제거할 수 있다. 본 논문은 Open Source GIS를 사용하여 고해상도 영상의 Tile Map Service 시스템을 구축하는 방안을 제시하는 것이다. Open Source GIS중 GDAL(Geospatial Data Abstraction Library)을 사용하여 Tile Map Image를 생성하였으며, OpenLayers를 이용하여 웹 브라우저를 통한 서비스를 구축하였다. 또한 Tile Map Service 시스템을 구축 한 후 전통적인 방식으로 지도 렌더링을 수행하는 시스템과 성능테스트를 수행하여 결과를 비교 분석하였다. 수행 결과 본 연구를 통하여 제안된 방법으로 고해상도의 영상 데이터를 상용소프트웨어의 도입 없이 고가용성의 Tile Map Service 시스템을 구축할 수 있었다.

핵심용어 : Open Source GIS, Web GIS, GDAL, Tile Map Service

Abstract

A Tile Map Service is a regular map service that has been enhanced to serve maps very quickly using a cache of static images. The map cache is a directory that contains image tiles of a map extent at specific scale levels. Returning a tile from the cache takes the server much less time than drawing the map image on demand. Use of a Tile Map Service can dramatically improve the time that clients take to display complex base-maps. Using Tile Map Services thus eliminate the need to trade quality for performance. This study provides a way to construct Tile Map Service System using Open Source GIS. We used GDAL (Geospatial Data Abstraction Library) which is one of the Open Source GIS Softwares to make Tile Map Image and OpenLayers to publish Web Page. Moreover, We conducted a performance test on Tile Map System and Dynamic Map System and evaluated the results of it. As a result, the proposed method makes it easier to construct high performance Tile Map Service using Open Source GIS without commercial products.

Keywords : Open Source GIS, Web GIS, GDAL, Tile Map Service

1. 서 론

참여와 공유, 개방을 모토로 하는 Web2.0은 현재 인

터넷 산업 분야의 핵심 화두이다. Web2.0의 물결과 함께 최근 Open Source Software의 활용이 활성화됨으로 서 IT산업에서 Open Source Software의 도입과 활용이

2009년 2월 25일 접수, 2009년 3월 20일 채택

* SK C&C Solution 사업팀, Solution Business Team, SK C&C (mhjeong@skcc.com)

** 교신저자·정회원·부경대학교 위성정보과학과, Department of Satellite Information Science, Pukyong National University (suh@pknu.ac.kr)

표 1. Open Source의 장점과 단점

장 점	단 점
저비용 : 라이선스 비용이 거의 없음	기존 상업용 소프트웨어로부터의 전환 비용
특정 회사 소프트웨어에 대한 의존성 감소	너무 다양한 공개소프트웨어 : 사용자 요구에 맞는 공개 소프트웨어를 찾기가 어려움
소프트웨어의 수정과 배포가 용이	공개 소프트웨어에 대한 국내 전문가 부족
효율적이며 작은 비용을 갖는 다양한 응용 소프트웨어의 존재	상대적으로 불편한 사용자 인터페이스
표준을 활용한 더 나은 상호 운영성	상대적으로 부족한 신뢰성과 유지보수의 책임소재 불분명
이종 환경에 대한 지원	UI를 제외한 핵심 코어부분 커스텀마이징이 어려움

활발히 진행되고 있으며, GIS(Geography Information System) 분야에서도 오픈소스 GIS를 이용한 시스템 구축이 점차 증가하는 추세에 있다.(남광우, 2006) 이와 같은 흐름 속에, Web GIS의 영역에서 고해상도의 영상을 빠른 속도로 조회할 수 있는 지도 서비스에 대한 사용자의 요구가 증가함에 따라 다양한 방안들이 도출되고 있다. 특히, Google Maps는 전통적인 이미지 기반 지도서비스를 능가하는 AJAX(Asynchronous Javascript and XML)와 Tile Map Service를 이용한 획기적인 유저 인터페이스를 선보여 전 세계 영상데이터 지도 서비스를 함으로써 GIS분야에 획기적인 변화를 몰고 왔으며(오정현, 2007) 현재 Tile Map Image를 이용하여 Web GIS 지도서비스를 수행하는 인터넷 사이트들이 많이 증가하는 추세이다. 따라서, 본 연구에서는 Web GIS 시스템 구축 시 Tile Map Service를 수행하기 위한 방법으로 상용소프트웨어를 사용하지 않고 용인시 고해상도의 항공사진 영상을 이용하여 Open Source GIS인 GDAL(Geospatial Data Abstraction Library)을 사용하여 Tile Map Service 표준을 준수 한 Tile Map Image를 만드는 방법을 제시하였으며, 전통적 방식의 Dynamic Map Service 시스템과 성능테스트를 수행하여 Tile Map Service 시스템의 성능에 대한 검증을 수행하였다.

2. Open Source GIS

Open Source GIS 혹은 GeoFOSS(Geospatial Free and Open Source Software)는 Free and Open Source Software를 지리정보시스템에 적용한 것을 말한다. 현재의 GIS 구축 환경에서 GIS 소프트웨어들은 ESRI, Intergraph, MapInfo, Autodesk 등과 같이 좋은 품질과 검증된 성능, 사후 관리 등이 보장된 상업용 제품들에 거의 의존하고 있다. 이러한 상업용 GIS 소프트웨어들은 GIS 소프트웨

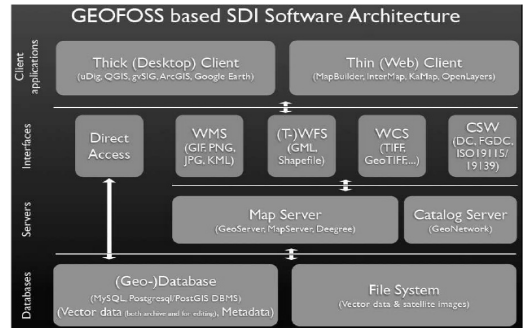


그림 1. Open Source GIS를 사용한 소프트웨어 아키텍처

어 벤더들이 소프트웨어와 소스코드에 대한 소유권을 갖고 있으며, 대부분 특정한 수의 컴퓨터에서 구동 가능한 컴파일 된 소프트웨어 라이선스들을 구매하는 형태로 이루어진다. 하지만, Open Source GIS 소프트웨어는 무료이며 소스코드가 공개되어 있어 자유롭게 소프트웨어 자체에 대한 확장이 가능하다.(Paul Ramsey, 2002) Open Source GIS의 장점과 단점은 다음 표와 같다(신상희, 2008).

위와 같은 장점을 가지고 Open Source GIS 소프트웨어로 GIS 구축을 위한 기반 프레임워크 개발이 다양하게 진행되고 있으며, 이러한 추세 속에서 현재 추진되고 있는 U-City 등의 공공부문 사업에서 Open Source GIS 소프트웨어를 적용하고자 하는 다양한 수요가 있을 것으로 예상된다. 그림 1은 Open Source GIS를 사용하여 구축할 수 있는 소프트웨어 아키텍처 그림이다.

3. 본 연구 Tile Map Service 구축 제안 방법

Tile Map Service 혹은 Cached Map Service는 데이터 베이스의 데이터에 직접 접근하는 것이 아니라 특정 축척별로 만들어진 이미지들을 호출하여 지도 서비스를 구현하는 것이다. 이 서비스는 미리 만들어진 Tile Map Image를 사용하기 때문에 동적으로 지도를 렌더링 할 필요가

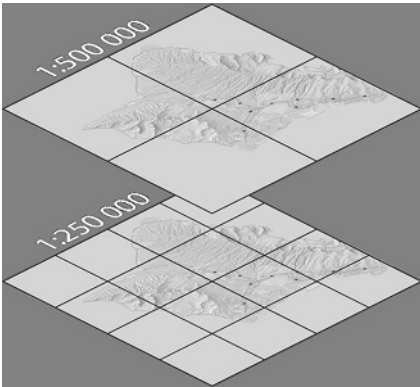


그림 2. Tile Map 개념

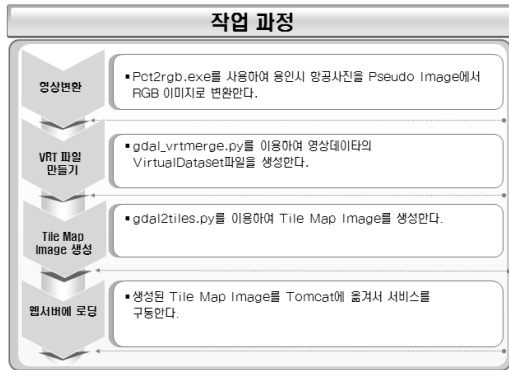


그림 3. 작업과정

없으므로 Web GIS 시스템 속도향상을 가져오며 영상지도와 같이 지속적인 변화가 없는 데이터에 대하여 적용하면 효과적이다.

본 연구에서 Tile Map Image를 만들기 위해서 Open Source GIS인 GDAL(Geospatial Data Abstraction Library)을 사용하였다. GDAL은 X/MIT Open Source License를 가지고 라스터 데이터에 대한 각종 변환 및 프로세싱에 대한 라이브러리를 제공하며 다양한 커맨드라인 유틸리티를 제공한다. 일부 기능들에 대한 버그들은 수정을 통하여 구현하였다. 또한, 본 연구에서 용인시 항공사진 영상 데이터를 가지고 Tile Map Image를 생성하였으며 Web Application Server로는 Tomcat을 사용하였고 Open Layers로 사용자 화면을 구축하였다. 본 연구의 작업 흐름은 다음과 같다.

3.1 영상데이터 변환

본 연구에서 사용한 데이터는 용인시 항공사진 데이터로 각 영상당, 70메가 의 Pseudo Color TIF파일이며 30cm

```
pct2rgb.py [-of format] [-b band] source_file dest_file
-of : 변환 파일 포맷
-b : RGB로 변환할 밴드
-source_file : 입력 파일
-dest_file : 생성될 파일
-예시 : pct2rgb D:\OSGEO\37709051.tif D:\OSGEO\37709051_rgb.tif
```

그림 4. RGB 데이터 변환방법

```
gdal_vrtmerge.py [-o out_filename] [-of out_format]
[-co NAME=VALUE]
[-ps pixelsize_x pixelsize_y] [-separate] [-v] [-pct]
[-ul_lr ulx uly lrx lry] [-n nodata_value] [-init value]
[-ot datatype] [-createonly] input_files

-o out_filename : 생성 파일 이름
-of out_format : 생성 파일 포맷. 기본은 GeoTIFF (GTiff) 이다.
-co NAME=VALUE : 생성 파일의 옵션을 만든다.
-ot datatype : 생성 이미지의 이미지 밴드를 설정한다.(ie. Byte, Int16,...)
-ps pixelsize_x pixelsize_y : 생성파일에 사용될 픽셀 사이즈
-ul_lr ulx uly lrx lry : 생성 파일의 영역
-v : VRT파일 생성과정의 이력을 보여준다.
-separate : 입력 파일을 분리된 밴드로 넣는다.
-pct : 첫 이미지의 pseudocolor table을 이용하여 생성파일에 적용한다.
-n nodata_value : 파일이 중첩될때 픽셀값을 무시한다.
-init value : 이 값으로 생성파일 초기화를 수행한다.
-createonly : 생성파일을 만든다 이미지 데이터의 값이 복사되지는 않는다.
-예시 : python gdal_vrtmerge.py -o D:\OSGEO\#tmp_rgb.vrt --optfile
D:\OSGEO\#rgblist.txt
```

그림 5. gdal_vrtmerge.py 사용방법

```
G:\rasters\#grs80_rgb\#37709051.tif
G:\rasters\#grs80_rgb\#37709052.tif
G:\rasters\#grs80_rgb\#37709053.tif
G:\rasters\#grs80_rgb\#37709054.tif
G:\rasters\#grs80_rgb\#37709059.tif
G:\rasters\#grs80_rgb\#37709060.tif
.
```

그림 6. 입력 이미지 리스트

의 해상도를 가진 136개의 파일로 구성되어있다. GDAL을 사용하여 Tile Map Image를 만들기 위해서는 RGB Color 형태의 파일이 필요하기 때문에 GDAL 유틸리티 중 하나인 pct2rgb.exe를 사용하여 용인시 항공사진 데이터를 RGB파일 형태로 변환하였다.

3.2 VRT 파일 만들기

용인시 항공사진 데이터를 RGB 파일로 변환하면 각 파일은 약 250메가 이며, 전체 영상파일의 크기는 약 30기가 를 차지한다. 이 파일들을 통합하여 Tile Map Image를 만들기 위해서는 많은 양의 컴퓨터 자원과 시간이 소요된다. 이와 같은 낭비를 막기 위해서는 각 영상 이미지 파일들에 대한 정보만을 가지는 텍스트 파일을 만든 후 이 파일을 이용하여 Tile Map Image를 만드는 것이 효율적이다. 따라서 VRT파일이란 각 영상 데이터에 대한 Virtual Dataset을 텍스트로 정의한 파일이다. VRT파일을

```
<VRTDataset rasterXSize="113720" rasterYSize="111417">
<GeoTransform>199950,0000000000000000, 0,300000000000000000,
0,0000000000000000, 430716,0000000000000000,
0,0000000000000000, -0,3000000000000000
</GeoTransform>
<SRS><PROJCS["unnamed"],GEOGCS["WGS 84",DATUM["WGS_1984",
SPHEROID["WGS 84",6378137,298.2572235630016,
AUTHORITY["EPSG","7030"]],
AUTHORITY["EPSG","6326"]],PRIMEM["Greenwich",0],
UNIT["degree",0.0174532925199433],AUTHORITY["EPSG","4326"]],
PROJECTION["Transverse_Mercator"],
PARAMETER["latitude_of_origin",38],
PARAMETER["central_meridian",127],
PARAMETER["scale_factor",1],PARAMETER["false_easting",200000],
PARAMETER["false_northing",500000],
UNIT["metre",1,AUTHORITY["EPSG","9001"]]]
</SRS>
<VRTRasterBand dataType="Byte" band="1">
<ColorInterp>Red</ColorInterp>
<NoDataValue>None</NoDataValue>
<SimpleSource>
<SourceFilename relativeToVRT="1">
G:\rasters\grs80_rgb\37709051.tif
</SourceFilename>
<SourceBand>1</SourceBand>
<SrcRect xOff="0" yOff="0" xSize="7717" ySize="9583"/>
<DstRect xOff="0" yOff="116" xSize="7717" ySize="9583"/>
</SimpleSource>
<SimpleSource>
<SourceFilename relativeToVRT="1">
G:\rasters\grs80_rgb\37709052.tif
</SourceFilename>
<SourceBand>1</SourceBand>
<SrcRect xOff="0" yOff="0" xSize="7720" ySize="9583"/>
<DstRect xOff="7380" yOff="113" xSize="7720" ySize="9583"/>
</SimpleSource>
</VRTDataset>
```

그림 7. 생성된 VRT파일

```
gdal2tiles.py [-title "Title"] [-publishurl http://yourserver/dir/]
[-nogooglemaps] [-noopenlayers] [-nokml] [-googlemapskey KEY]
[-forcekml] [-v] input_file [output_dir]

-title "Title": 생성되는 메타데이터를 위한 타이틀.
-publishurl http://yourserver/dir/: 생성된 결과물을 웹서버에 로딩할때 사용될 주소.
-nogooglemaps: Google Maps 에 기반한 html 페이지를 생성하지 않는다.
-noopenlayers: OpenLayers 에 기반한 html page를 생성하지 않는다.
-nokml: Google Earth 를 위한 KML 파일들을 생성하지 않는다.
-googlemapskey KEY: Google Maps API 웹 페이지에서 생성된 사용자 키 값.
-forcekml : KML 파일들을 생성시킨다.
-v : gdal2tiles.py 를 통한 Tile Map Image 생성과정을 보여준다.
-예시: python gdal2tiles.py -title "Yongin" -publishurl
http://192.168.1.134:8089/geo/osmap/ -nogooglemaps -nokml
-v D:\OSGEO\wtmp_rgb.vrt D:\OSGEO\raster3
--config GDAL_CACHEMAX 120
```

그림 8. gdal2tiles.py 사용 방법

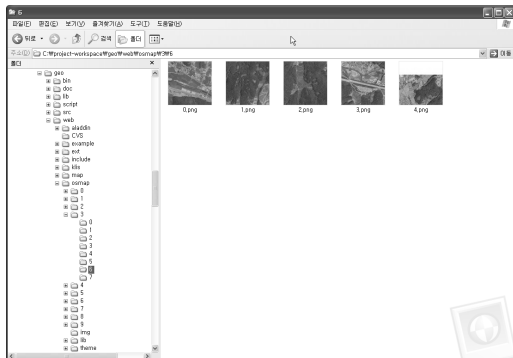


그림 9. Map Service 디렉토리 구조

작성하기 위해서 GDAL 유틸리티중 하나인 gdal_rtmerge.py 스크립트를 사용하였다.

VRT파일을 만드는 경우는 다량의 이미지 파일이 있는 경우에 주로 사용함으로 입력 파일들을 텍스트 파일에 리스트로 불러와서 VRT파일을 생성하는 것이 효율적이다. 그림 5의 gdal_vrtmerge.py 사용 방법 예시에서 사용한 입력 이미지 리스트 파일(rgblist.txt)의 내용은 다음과 같다.

또한, gdal_vrtmerge.py 스크립트를 사용하여 생성되는 tmp_rgb.vrt의 내용의 일부는 다음 그림과 같다. 생성된 파일의 내용을 살펴보면 영상 데이터의 전체 크기 정보와 Projection 정보 및 각 이미지 파일들의 컬러 밴드별로 각 이미지를 픽셀 값 등에 대한 정보가 표시된다. 따라서 전체 이미지들을 하나의 이미지로 만들지 않고도 각 이미지들의 정보를 모아서 Tile Map Image를 만들 수 있다.

3.3 Tile Map Image 생성

VRT 파일을 만든 후 gdal2tiles.py 스크립트를 이용하여 Tile Map Image를 생성하였다. 이 유틸리티는 OSGEO Tile Map Service Specification을 따르면서 작은 이미지 타일과 메타데이터를 가진 디렉토리를 생성하며, Google Maps와 OpenLayers에 기반한 화면을 가진 단순한 웹페이지도 생성된다. 따라서 생성된 디렉토리를 웹서버에 옮김으로써 상용 소프트웨어를 설치하거나 구성할 필요 없이 웹 브라우저를 통하여 아주 빠르게 지도를 조회할 수 있다.

또한, 본 연구에서 사용한 데이터와 같이 대용량의 이미지 데이터를 Tile Map Image로 만들기 위해서는 gdal2tiles.py 스크립트를 기동하는 동안 컴퓨터의 자원을 효율적으로 사용하기 위해서 구성 속성 중 GDAL_CACHEMAX 값을 조절하였다.

3.4 웹서버에 로딩

gdal2tiles.py를 이용하여 생성된 Tile Map Image 데이터를 웹 서버인 Tomcat으로 옮긴다. 이와 같이 생성된 폴더를 Tomcat으로 옮김으로써 사용자가 데이터베이스에 요청을 통하여 데이터에 대한 정보를 가져 올 필요 없이 기존에 생성된 이미지로 지도 이미지를 빠르게 조회할 수 있다. 다음 그림은 Tile Map Image로 생성된 폴더를 Tomcat에 옮긴 폴더 구조이다.

본 연구 방법으로 만들어진 Tile Map 폴더(약 1.2기가)를 웹서버인 Tomcat에 옮긴 후 브라우저를 통하여 OpenLayers 자바스크립트를 이용하여 다음 그림과 같이 서비스를 할 수 있다.

4. 성능 테스트 비교 및 평가



그림 10. Tile Map Service 화면

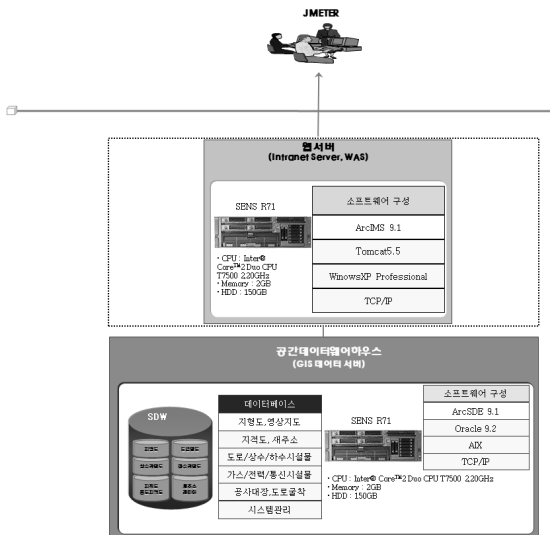


그림 11. H/W 및 S/W 총괄 구성도

본 연구에서 제안한 Tile Map Service 방식과 데이터베이스를 통하여 지도 렌더링을 하는 경우를 비교하여 총 3가지의 테스트 시나리오를 가지고 성능테스트를 수행하였다. 본 연구에서 성능테스트를 위한 시스템 아키텍처는 그림 11과 같다. SENS R71 PC에 Database 레벨에서는 Oracle9i, ArcSDE9.1을 사용하였고, Application Server 레벨에서는 Tomcat5.5, ArcIMS9.1로 구성하였으며 성능테스트 소프트웨어로는 JMeter-2.3.2 버전을 사

용 하였다.

또한, 각 테스트의 시나리오 정의는 표2와 같다.

4.1. 시나리오1 결과

시나리오1은 시스템 구축 후 본 연구에서 제안한 방법으로 Tile Map Image를 만든 다음 Tomcat에 이미지들을 로딩하여 테스트하는 경우이다. 부하레벨 50명을 기준으로 1초 간격으로 10,000회 동안 지도 확대 기능에 대한 HTTP Request를 요청하는 경우 응답시간을 체크하였다. 그림 12는 성능 테스트 환경 설정 화면이다.

4.2. 시나리오2 결과

시나리오2은 Dyanmic Map Service 방식으로 시스템 구축 후 기본 세팅에서 ArcIMS와 ArcSDE와 Oracle을 사용하여 부하레벨 50명을 기준으로 1초 간격으로 10,000 회 동안 지도 확대 기능에 대한 HTTP Request를 요청하는 경우 응답시간을 체크하였다. 그림 14는 성능 테스트 환경 설정 화면이다.

시나리오2의 성능 테스트 결과는 평균 33msec응답 속도를 나타냈었다. 그러나 모든 Request가 100% 에러를 발생시켰다. 다중 접속으로 웹서버가 다운 되어서 웹 서버 및 데이터베이스에 관한 튜닝 작업을 수행해야 에러를 제거 할 수 있는 것으로 판단되었다.

4.3. 시나리오3 결과

시나리오3은 Dyanmic Map Service 방식으로 시스템 구축 후 기본 세팅에서 ArcIMS와 ArcSDE와 Oracle을 사

표 2. 성능 테스트 시나리오

테스트 시나리오	상세 설명
시나리오1	본 연구에서 제안한 방법으로 부하레벨 50명을 기준으로 1초 간격으로 10,000회 동안 HTTP Request를 요청하는 경우 지도확대 기능에 대한 응답시간
시나리오2	Dynamic Map Service 방식으로 WEB GIS엔진으로 ArcIMS와 RDBMS로 Oracle 및 SDE를 사용하여 기본적인 세팅 환경에서 부하레벨 50명을 기준으로 1초 간격으로 10,000회 동안 HTTP Request를 요청하는 경우 지도확대 기능에 대한 응답시간
시나리오3	Dynamic Map Service 방식으로 WEB GIS엔진으로 ArcIMS와 RDBMS로 Oracle 및 SDE를 사용하여 기본적인 세팅 환경에서 부하레벨 30명을 기준으로 10초 간격으로 10,000회 동안 HTTP Request를 요청하는 경우 지도확대 기능에 대한 응답시간

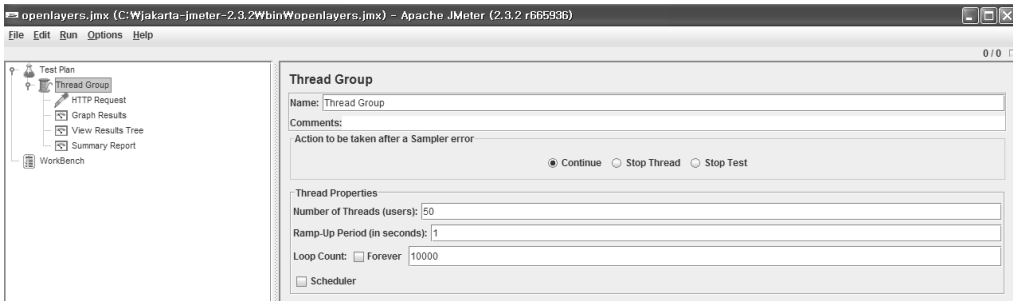


그림 12. 성능 테스트 환경 설정 화면

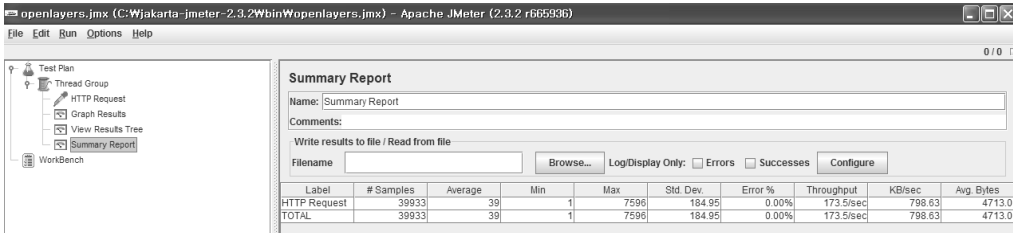


그림 13. 응답시간 결과(시나리오1)

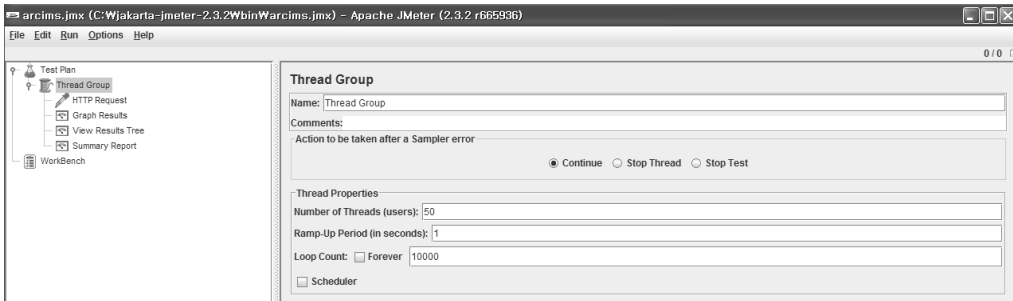


그림 14. 성능 테스트 환경 설정 화면

용하여 부하레벨 30명을 기준으로 10초 간격으로 10,000회 동안 지도 확대 기능에 대한 HTTP Request를 요청하

는 경우 응답시간을 체크하였다. 그림 16은 성능 테스트 환경 설정 화면이다.

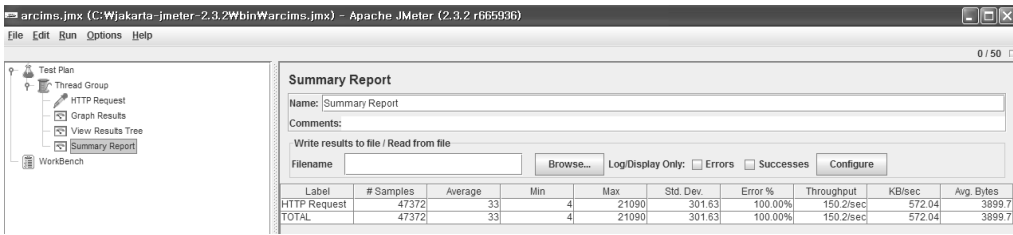


그림 15. 응답시간 결과(시나리오2)

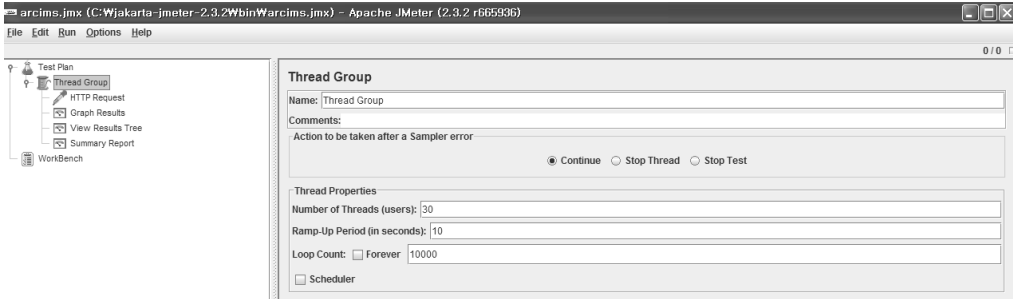


그림 16. 성능 테스트 환경 설정 화면

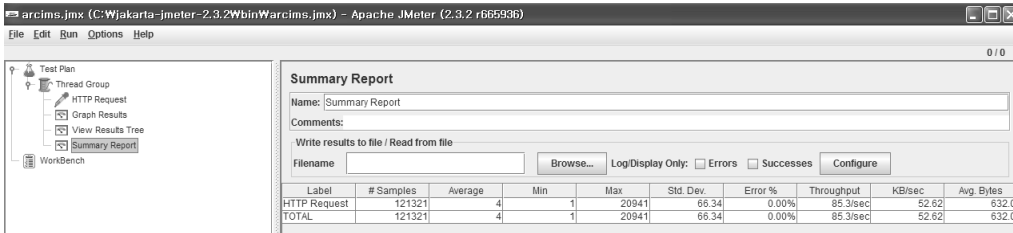


그림 17. 응답시간 결과(시나리오3)

시나리오3의 성능 테스트 결과는 평균 4msec 응답 속도를 나타내므로 1초 이하의 응답속도와 에러발생율도 0% 이었다. 하지만 시나리오1과 2와 다르게 동시 사용자수를 30명, 서버 요청 간격을 10초의 간격으로 작업하였으므로 서버에 부하를 적게 주어 작업하였다.

5. 결 론

본 연구는 Open Source GIS를 이용한 고해상도 영상의 Tile Map Service 시스템 구축을 위한 방안을 제시하였으며 본 연구에서 제안한 방식으로 생성된 시스템과 전통적인 Dynamic Map Service 방식의 시스템에 대한 성능 테스트를 수행하여 응답 시간을 비교 분석하였다. 성능 테스트 결과 같은 조건에서 Tile Map Service 방식은 1초 이하의 빠른 응답시간을 보였지만, 데이터베이스를 통하

여 지도를 렌더링 하는 방식은 서버가 다운 되는 현상을 보였다. 따라서, Open Source GIS 소프트웨어인 GDAL의 다양한 라이브러리들을 사용하여 상용소프트웨어의 도입 없이 Tile Map Service 시스템을 구축할 수 있었으며, 주기적인 변동이 없는 대용량 데이터에 대한 Web GIS 시스템에 Tile Map Service 구축 시 응용 가능할 것으로 판단된다. 또한, 방재관련 GIS 지도서비스 시스템 구축 시 기본 레이어를 Tile Map Service로 구축한 후 실시간 및 동적으로 변동되는 레이어를 WMS(Web Map Service)로 오버레이 시켜서 시스템을 구축하면 다양한 지도 레이어들과 MashUp 기능을 통하여 확장성이 높은 시스템을 구축 할 수 있으며 모든 레이어들을 서버에 직접 호출하지 않고 지도상에 표시함으로써 속도 향상에 기여할 수 있다. 향후 과제로는 Tile Map Image를 생성하는 경우 데이터가 없는 영역의 값을 사용자가 선택하여

색상을 세팅할 수 있도록 하는 기능, 대용량의 Tile Map Image들을 Cache하여 사용할 수 있는 부분, 분산 환경에서의 시스템 구축 적용에 관한 추가적인 연구가 필요한 것으로 사료된다.

감사의 글

이 연구는 기상청 기상지진기술개발사업(CATER 2009-3111)의 지원으로 수행되었습니다.

참고문헌

1. 남광우, 2006, "표준/Open Source 기반의 GIS 구축 지침 개발 연구", 2006년 정보통신부 연구결과 보고서, 정보통신부, pp. 10-102
2. 신상희, 2008, "Open Source GIS 최근 동향", OSGeo Korea 발표자료, OSGeo Korea, pp. 7-11
3. 손영기, 신영철, 2001, "Web GIS 구축시 UML을 이용한 모델링에 관한 연구-충북대학교를 중심으로", 한국지리정보학회지, 한국지리정보학회, 제 4권, 2호, pp. 46-60
4. 오정연, 2007, "새로운 GIS 패러다임 Where 2.0에 주목하라", 한국정보사회진흥원 정보사회 현안 분석, 한국정보사회진흥원, pp. 15-28
5. 최원근, 김경옥, 장윤섭, 2008, "Open API 기반의 사용자 데이터 웹 지도 서비스 방안", 2008 한국지리정보학회 춘계 학술발표논문 초록집, 한국지리정보학회, pp. 36-37
6. Paul Ramsey, 2002, "Open Source GIS Fights the Three-Horned Monster", August 2002 GeoWorld, GeoWorld, pp. 34-36
7. Seokchan Yun, 2007, "The User-Participated Geospatial Web as Open Platform", Scotland, Paper presented at the Proceeding of the 11th International Seminar on GIS, The Korea Research Institute for Human Settlements, pp. 33-48.
8. ESRI, 2007, A Cached Map Service, USA, http://webhelp.esri.com/arcgisserver/9.2/dotNet/manager/publishing/static_map_svcs.htm
9. OSG대 2007, Geospatial Data Abstraction Library, USA, <http://www.gdal.org/>
10. OGC, 1994, Open Geospatial Consortium, USA, <http://www.opengeospatial.org/>
11. OSGeo, 2008, Tile Map Service Specification, USA, http://wiki.osgeo.org/wiki/Tile_Map_Service_Specification