

## 공개소스 PostGIS 기반 공간정보 처리 툴킷 사용자 인터페이스 구현

한선목 · 이기원<sup>†</sup>

한성대학교 정보시스템공학과

### Implementation of User Interface for Geo-spatial Information Processing Toolkit using Open Source-based PostGIS

Sun-Mook Han and Kiwon Lee<sup>†</sup>

Dept. of Information System Engineering, Hansung University

**Abstract :** Recently, open sources for geo-spatial information processing and analysis are being developed, and they are widely adopted for the various application development projects. Open sources in the geo-spatial communities consist in several levels or types: viewer, API-level, engine sources of SDK-level, or toolkits. Among them, spatial database engine of PostgreSQL-PostGIS is used in this study for the portable multi-geospatial information processing toolkit. This work can be extended to target-based applications with domain-specific spatial queries and analyses. Design and implementation are based on C Language Interface (LIBPQ) to PostGIS and OGC library on PostgreSQL database. Conclusively, PostGIS according to this approach is an important alternative to develop most applications dealing with multi-geospatial information due to its availability, extensibility, scalability, and stability.

**Key Words :** PostgreSQL, PostGIS, Open Source, Geo Toolkit, Geodata.

**요약 :** 최근, 지공간 정보 처리와 분석을 위한 여러 가지의 공개 소스가 개발되어 다양한 응용 개발 프로젝트에 적용되고 있다. 지형공간 분야에서 공개 소스는 뷰어, 응용 프로그램 인터페이스, 개발 환경으로 제공되는 엔진 소스코드, 툴킷 등과 같은 여러 가지 유형으로 구분된다. 본 연구에서는 간편한 다중 공간정보 처리 툴킷을 구현하기 위하여 공간 데이터베이스 엔진으로 발표된 PostgreSQL-PostGIS를 적용하였다. 이 결과는 특정 응용 분야의 질의나 분석 기능을 포함하는 응용 시스템 개발시에 적용될 수 있다. 설계 및 구현 작업은 PostgreSQL 데이터베이스에서 운영되는 OGC 라이브러리와 C 인터페이스인 LIBPQ를 이용하였다. 결론적으로 가용성, 확장성, 대용량 처리, 안정성 등의 이점에 따라 PostGIS는 다중 공간정보를 다루는 대부분의 응용 시스템 개발 시에 이용될 수 있는 중요한 대안이 될 수 있다.

접수일(2009년 4월 15일), 게재확정일(2009년 4월 20일).

<sup>†</sup> 교신저자: 이기원(kilee@hansung.ac.kr)

## 1. 서론

2000년대 중반 이후 포털 검색 엔진에서 웹 매핑을 중심으로 한 지도와 위성영상정보 서비스가 제공 되고, 소위 Web 2.0이라는 패러다임이 이러한 정보서비스와 결합되면서 지도 자료나 지도 정보에서 발전한 생활 밀착형 지도 콘텐츠의 유형이 정착되고 있다. 과거보다 더욱 많은 자료의 축척과 빠른 질의처리가 요구되고 있다. 이러한 주요 동향은 기존의 공간정보시스템 기술이나 응용 분야에도 많은 영향을 주게 되어 GIS 커뮤니티를 중심으로 공개 소스(Open Source)에 대한 관심이 증가하고 있고 국내외에서 실무 적용에 대한 다양한 사례 개발이 추진되고 발표되고 있다.

공간정보 분야에서 포괄적으로 보면 공개 소스는 다양한 수준이나 유형으로 나타나는데 무상으로 제공되는 다양한 공간 자료 가시화 가능 중심의 뷰어(Viewer), 개발자 인터페이스로 제공되는 응용 프로그램 인터페이스(API), 공개 소스 코드를 포함한 SDK(Software Development Kits)이나 공간데이터베이스 엔진 소스 등이 이러한 범주에 포함된다고 볼 수 있다. 이러한 공개 소스는 웹 기반 지도정보 콘텐츠 검색 및 제공 서비스와 매쉬업(Mashup)이라는 다양한 방식으로 연계가 가능하다. 또한 이러한 공개 소스나 매쉬업은 산업계를 중심으로 기존의 GIS 응용 시스템의 설계나 개발 과정에도 파급 효과를 주고 있다. 국내외 산업계의 개발이나 응용 전략은 여러 가지가 있을 수 있으나, 수요자의 다양한 요구사항에 능동적이고 유연하게 대처할 수 있는 사용자 인터페이스의 개발 분야가 우선적으로 고려되고 있는 분야라고 볼 수 있다. 또한 개발이나 운영 과정에서 비용이 많이 소요되는 공간 데이터베이스를 기반으로 하는 응용 시스템 분야에서의 공개 소스 적용에 대한 연구나 기술 개발도 지속적으로 진행되고 있다. 또한 국제 표준 공간 자료의 상호운용을 주 기능으로 하는 웹 기반 공간정보 서비스 분야도 공개 소스를 채택하는 주요 분야 중의 하나이다(Benthall, 2009).

본 연구에서는 경량화된 공간 데이터베이스 응용 프로그램의 개발과 공간정보 웹 서비스와의 연동을 위하여 확장 가능한 공개 소스 기반 툴킷을 설계, 구현하고자 한다. 이러한 목적을 위하여 적용할 수 있는 무상의 공간 엔진의 소스 코드는 몇 가지가 있지만 300 가지 이

상의 다양한 공간 질의 함수와 OGC 표준 사양을 지원하는 OGC 라이브러리를 제공하는 PostgreSQL-PostGIS를 적용하고자 한다. 여기서 PostgreSQL은 공개 소스 데이터베이스 관리시스템이며(<http://www.postgresql.org/docs/manuals/>), PostGIS는 PostgreSQL의 공간 엔진에 대한 공개 소스이다(<http://postgis.refrains.net/>). 즉 PostgreSQL과 PostGIS는 각각 설계나 구현 단계에서 데이터 서버와 공간 질의 함수를 지원하는 사용자 인터페이스 환경에 적용된다.

본 연구에서는 PostgreSQL-PostGIS를 개관하고 이를 기반으로 하는 툴 킷의 설계에 적용되는 기반 사항, 구현 환경을 설명하고, 툴 킷을 이용한 간단한 사용자 인터페이스를 제시하고자 한다.

## 2. PostgreSQL-PostGIS 개요

본 연구에서 사용한 공간 데이터베이스 관리시스템은 객체-관계형 데이터베이스를 지원하는 공개 소스 PostgreSQL이다. PostgreSQL은 Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), Windows등과 같은 대부분의 운영체제에서 구동이 가능하다. 또한 PostgreSQL은 강력한 객체 관계형 데이터 모델과 풍부한 데이터 타입과 거의 제한 없는 자료 크기를 지원하고, 고수준의 확장성을 가지고 있으며, 복합 객체(complex object), 규칙(rule) 등의 기능과 질의 최적화, 동시성 제어, 트랜잭션 처리, 다중 사용자 기능, GiST, R-tree 등과 같은 공간 검색 알고리즘을 제공하고 있다. 또한 PostgreSQL은 다양한 개발 언어에 대한 SDK 인터페이스를 제공한다(Philippe, *et al.*, 2002; Matthew and Stones, 2005).

Table 1은 현재 이용 가능한 공개 소스 데이터베이스 관리시스템을 비교한 것이다. PostgreSQL과 Ingres의 경우는 대부분의 데이터베이스 관리시스템에서 요구되는 기능과 함께 공간 엔진 기능을 제공하므로 비록 무상의 공개 소스지만 공간 데이터베이스 관리시스템에 기반한 응용 시스템 구축에 적용하는 데 적합한 것으로 나타나 있다. 한편 Hass(2007)는 공개 소스 데이터베이스 관리시스템의 성능 비교를 수행한 바 있는데, PostGIS

Table 1. Comparison of open source DBMS (Lemmen, 2007; Hall and Leahy, 2008)

		Open Source DBMS			
		PostgreSQL 8.2	FireBird 2.0	MySQL 5.0	Ingres 2006
ACID		Support	Support	Support (InnoDB table)	Support
Associated Integrity		Support	Support	Support (InnoDB table)	Support
DB transactions		Support	Support	Support (InnoDB table)	Support
Unicode Index		Support	Support	Support	Support
	R-/R+	Support	No support	Only MyISAM	Support
	Hash	Support	No support	Only InnoDB	Support
	GiST	Support	No support	No support	No support
Table partition	Range	Support	No support	Support	Support
	Hash	Support	No support	Support	Support
	List	Support	No support	Support	Support
Cluster		Support by add-on	No support	Support	Support
Spatial extension		Support	No support	Support	Support

와 MySQL의 비교에서는 PostGIS가 성능이 우수하다고 보고한 바 있다.

PostGIS는 PostgreSQL에서 공간 정보 처리를 위한 공간 엔진(Spatial Extension)으로, Fig. 1은 PostgreSQL 데이터베이스 서버 환경에서 공간 질의 처리 모듈인 PostGIS의 역할을 보여준다.

공간 질의 처리 미들웨어인 PostGIS는 공간 GIS 데이터를 관리하고, 공간 질의를 처리하기 위해 다음과 같은 기본 규칙을 가지고 구현되었으며 PostGIS의 주요 특성은 Table 2에 나타나 있다.

- SQL을 위한 OGC Simple Features 명세 구현
- 공간 객체들의 입력, 질의, 조작, 삭제에 대한 연산자들과 SQL 스키마 정의

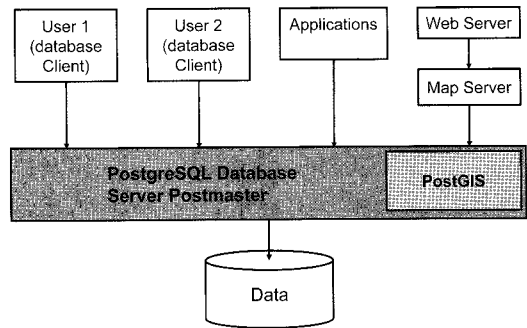


Fig. 1. PostGIS as spatial extension of PostgreSQL DBMS (Arnulf, 2006).

- 공간 객체들의 좌표정보는 Feature Table에서 저장
- 하나의 Feature Table은 한 가지 유형의 기하 요소만 표현

Table 2. Main features of PostGIS (Hall and Leahy, 2008)

DBMS/ Date of Introduction	PostGIS 1.2.1/2007.1
OGC compliant	SFSQL-TF 1.1 Certified
ISO standards support	SQL/MM
Spatial data-types (vector-oriented) 2D, 3D	As specified in OGC SFSQL: Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon, GeometryCollection
Spatial data-types (raster-oriented)	CHIP data type to store rasters in PostgreSQL
Spatial data-operators (vector-oriented)	OGC + ST_* + PostGIS Specific ?several hundred data-operators are supported
Spatial data-operators (raster-oriented)	Input/output only
Spatial data index 2D, 3D	2D GiST and R-Tree (with multi-version concurrency and recovery)
Supported co-ordinate systems/projections	All EPSG CRS systems
Topology support (node, edge, face)	SQL/MM Topology Model + Basic Functions
Spatial join algorithms	Spatial join using index operations spatial/attribute query optimization
Exchange formats	FME(ESRI Interoperability Extension) formats, GML, SVG, KML

- 각각의 객체의 좌표정보는 공간적 유형을 가진 하나의 필드에 저장
- 좌표정보의 필드 유형으로 WKT(Well Known Text)
- 각각의 Feature Table 메타 데이터는 유형과 포함된 기하 좌표로 구성
- 각각의 Feature Table 메타 데이터는 geometry\_columns(공간 테이블)에 저장

일종의 공간 질의 처리 미들웨어로서 PostGIS는 OGC WKT (Well-Known Text)와 WKB(Well-Known Binary) 형태인 공간 GIS 객체들과 공간 GIS 데이터를 관리하기 위한 공간 연산자와 함수를 제공한다. 또한 일부 상업적 공간정보 포맷을 포함한 다양한 종류의 데이터를 읽고, 변환하고, 추가를 가능하게 하는

OGC 라이브러리를 제공한다. 그리고 OGC 표준을 따르는 대부분의 기하 요소와 300개 이상의 다양한 공간 질의 함수와 쉽고 강력한 공간 색인을 제공한다. 지원되는 Geometry 형태는 Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon, GeometryCollection 이다.

Table 3은 PostGIS에서 제공하는 다양한 공간질의 함수를 정리한 것이다. 이러한 공간 GIS 데이터 처리 기능은 보통 미들웨어 형태로 구성되어 있고, 공간 GIS 데이터를 정의하기 위한 개체 구조와 접근을 위한 JDBC나 ODBC 등과 같은 연결자 및 추가, 검색, 조작을 위한 다양한 함수들을 제공한다. 한편 Table 4는 PostgreSQL-PostGIS에서 제공하는 다양한 개발 언어에 대한 SDK

Table 3. Some types of spatial operators supported in PostGIS (Ramsey, 2004; Paul, 2007)

Function Category	Elements	Functions
Measurement	ST_Area (geometry) returns Numeric	Area of the geometry, in the units of the geometry projection
	ST_Length (geometry) returns Numeric	Length of the geometry, in the units of the geometry projection
	ST_Perimeter (geometry) returns Numeric	Perimeter of the geometry, in the units of the geometry projection
	ST_Distance (geometryA, geometryB) returns Numeric	Distance between geometryA and geometryB
Comparison	ST_Intersects (geometryA, geometryB) returns Boolean	True if the geometries intersect
	ST_Contains (geometryA, geometryB) returns Boolean	True if geometryA contains geometryB
	ST_Within (geometryA, geometryB) returns Boolean	True if geometryA is within geometryB
	ST_Touches (geometryA, geometryB) returns Boolean	True if the geometries touch at their boundaries only
	ST_Crosses (geometryA, geometryB) returns Boolean	True if the geometries cross (useful for lines)
Utility	ST_SRID (geometry) returns Integer	Returns the SRID of the geometry
	ST_SetSRID (geometry, srid) returns Geometry	Returns a geometry with the SRID set to the specified value. Note that SetSRID does not transform the geometry to the new SRID, it simply sets the value
	ST_GeometryType (geometry) returns String	Returns the type of the geometry.
	ST_AsText (geometry) returns String	Returns the geometry as a human-readable string.
	ST_AsBinary (geometry) returns ByteA	Returns the geometry as a machine-readable byte array
	ST_AsGML (geometry) returns String	Returns the geometry as a Geometry Markup Language XML string
Geometry Operation	ST_Intersection(geometryA, geometryB) returns Geometry	Returns the geometric intersection of geometryA and geometryB
	ST_Union(geometryA, geometryB) returns Geometry	Returns the geometric union of geometryA and geometryB
	ST_Difference(geometryA, geometryB) returns Geometry	Returns the geometric difference of geometryA less geometryB
	ST_Buffer(geometry, distance) returns Geometry	Returns a geometry expanded by the specified distance, or shrunk if distance is negative
	ST_Expand(geometry, distance) returns BBOX	Returns a bounding box, larger than the box of the input geometry by the specified distance
	ST_Centroid(geometry) returns Geometry	Returns a point, near the center of the geometry. Not necessarily inside polygonal features. Use ST_PointOnSurface(geometry) for that
	ST_Transform(geometry, srid) returns Geometry	Returns a geometry with the coordinates transformed to the new SRID
	ST_Simplify(geometry, tolerance) returns Geometry	Returns a geometry with fewer vertexes, using the Douglas/Poiker vertex weeding method

Table 4. SDK Interfaces supported in PostgreSQL-PostGIS (Paul, 2007)

Interface	Language	Processing	Advantages
LIBPQ	C	compiled	native interface
LIBPGEASY	C	compiled	simplified C
ECPG	C	compiled	ANSI embedded SQL C
LIBPQ++	C++	compiled	object-oriented C
ODBC	ODBC	compiled	application connectivity
JDBC	Java	both	portability
PERL	Perl	interpreted	text processing
PGTCLSH	TCL/TK	interpreted	interfacing, windowing
PYTHON	Python	interpreted	object-oriented
PHP	HTML	interpreted	dynamic Web pages

인터페이스이다.

Fig. 2는 공개 소스의 특징 중 하나인 소위 공개 소스 에코시스템(Eco System) 측면에서 PostGIS와 연동이 가능한 공개 소스나 공간 GIS 엔진 또는 개발 환경은 Mapserver, Geotools (Geoserver, uDig), JUMP (OpenJUMP, Kosmo), OGR (QGIS, Mapserver), Autodesk MapGuide 등과 같이 다양하다. 하나의 공개 소스만으로 어플리케이션이 구축될 수도 있으나 각

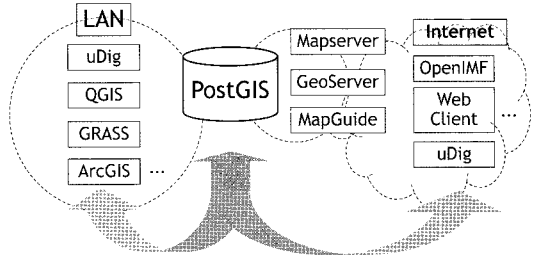


Fig. 2. Other open sources linkable with PostGIS (excerpted from anonymous internet sources).

공개 소스는 각각 고유한 특성과 적용상의 장점이 있으므로 공개 소스간의 통합 연계를 도모하는 개발 전략은 공개 소스 기반 어플리케이션 구축에서는 일반적인 특징이다(Brovelli and Magni, 2004; Paolo, *et al.*, 2006; Ramsey, 2006; Chu and Pao, 2007).

공간 데이터베이스를 기반으로 OGC 상호운영 프로세스를 통한 웹 매핑 어플리케이션에 대한 하나의 시나리오에서 웹 클라이언트의 지도정보 조회 및 요청이 있을 때 요청하는 데이터의 유형에 따라 웹 서버와 공간 엔진에서는 다양한 처리 과정이 필요하게 된다(Fig. 3). 여기서 공간 데이터베이스 엔진은 입력된 공간 질의를

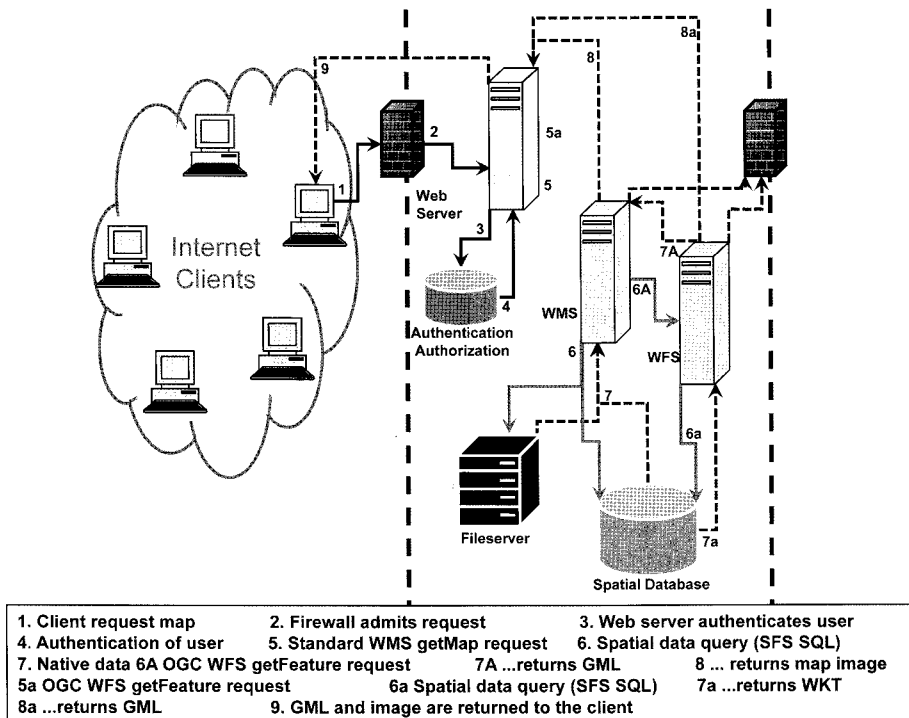


Fig. 3. A scenario regarding web mapping by OGC interoperable flow with Scalable databases. Modified from web mapping as spatial data infrastructure by Amulf (2006).

처리하고 그 결과를 다양한 데이터 형식으로 반환하고 다양한 데이터 형식으로 분산되어 있는 공간 자료를 통합 관리하는 역할을 수행한다. 공간 데이터베이스 관리 시스템으로는 PostgreSQL이 이용될 수 있고, 공간 엔진인 PostGIS는 사용자 인증을 담당하는 웹 서버(과정 2, 3, 4)와 연동하여 OGC 라이브러리를 이용하여 공간 자료의 송수신, 변환, 추가 등의 기능(과정 5, 6, 7, 8, 9)을 통합 환경이나 개별 환경에서 수행하게 된다. 이 경우에 웹 서비스되는 공간 정보의 유형이나 기능이 수요자로부터 추가적으로 요구되는 경우에는 Fig. 2와 같이 다른 공개 소스와 연동이 가능하다. 이러한 경우에는 데이터베이스 무결성 문제와 최적화, 안정성 문제 등을 같이 고려해야 하는 데 이러한 작업이 가능한 공개 소스의 툴 킷이 별도로 제공되는 경우에는 중간 개발자는 보다 용이하게 통합 환경 어플리케이션을 구축할 수 있다.

### 3. PostGIS 툴 킷 사용자 인터페이스

본 연구에서 구현한 툴 킷은 본 연구의 개발 환경과 같은 WindowsXP - PostgreSQL-PostGIS를 사용하는 환경에서 사용자가 데이터를 조작하기 위한 GUI환경을 제공하고 있다. 해당 툴 킷을 사용하기 위한 환경 설정으로 우선 데이터 베이스 관리시스템인 PostgreSQL이 필요하다. 공개 소스인 PostgreSQL은 (<http://www.enterprisedb.com/getfile.jsp?fileid=485>)에서 원클릭 설치 파일을 얻을 수 있다. PostgreSQL의 설치를 완료한 후에 추가적인 기능 확장을 위한 'Application Stack Builder'가 자동으로 실행되며 이 프로그램을 이용하여 공간 GIS 미들웨어인 PostGIS를 설치할 수 있다. 한편 영어 이외의 다른 언어(한글, 일본어 등..)를 사용하기 위해서는 Windows 환경변수에 다음을 추가해 주어야 한다.

- PGCLIENTENCODING='uhc'

DB에 공간 GIS 데이터를 입력하는 방법은 PostGIS의 함수와 SQL을 이용하는 방법과 PostGIS Shape Loader를 이용하는 방법이 있다.

- SQL with PostGIS Function - Example:

```
CREATE TABLE user_locations (gid int4,
user_name varchar);
```

```
SELECT AddGeometryColumn
('db_mapbender', 'user_locations',
'the_geom', '4326', 'POINT',2);
INSERT INTO user_locations values
('1', 'Morissette', GeometryFromText
('POINT(-10.1 48.43)', 4326));
INSERT INTO user_locations values
('2', 'Sperb', GeometryFromText('POINT(-
48.67 -26.89)', 4326));
...
```

- PostGIS Shape Loader

```
shp2pgsql <shape_filename> <table_name>
<sql_filename>
psql -d <database_name> -f <sql_filename>
```

Fig. 4는 본 연구에서 구현한 툴 킷에서 DBMS에서 조작할 테이블에 연결하는 환경 설정과정을 제시한 것이다.

기존의 PostGIS의 함수들은 Command 방식으로 SQL문을 작성해서 DB에 질의를 보내는 방식으로 되어 있다. 하지만 이러한 방법은 SQL문을 충분히 숙지하지 않은 사용자들에게는 사용하기 힘든 방식이다. 그래서 본 연구에서는 PostGIS의 개발 언어 인터페이스 중에서 LIBPQ를 이용하여 GUI 환경에서 사용할 수 있는 DB연결자를 생성하였다. DB연결자인 PgConn은 GUI 환경에서 사용자의 마우스 조작과 추가적인 변수 입력을 받아 공간질의 SQL문을 생성하고, PostgreSQL Database Server에 질의를 보내 결과를 받아온다. 그리고 이 결과를 사용자에게 표현해 준다. Fig. 5는 본 연구에서 설계된 뷰어 프로그램의 구조이다. 본 연구의 개발 환경은 OS는 Windows XP이고, 데이터베이스 관리 시스템은 PostgreSQL 8.3.5이고, 공간 GIS 미들웨어는 PostGIS 1.3.5이다. 또한 개발 언어는 Microsoft Visual C++ 6.0 MFC이고, 라이브러리는 PostGIS의 C Interface인 LIBPQ와 다양한 형식의 자료를 변환 및 처리해 주는 OGC 라이브러리를 이용하였다. 또한 이용한 실험 데이터는 임의 지역의 Shapefile을 이용하였다. 사용자 인터페이스 부분에서 PgViewerView는 데이터베이스에 있는 공간 GIS 개체를 표현해주고 일반적인 마우스 이벤트를 받는다. LayersView는 현재 PgViewerView에 표현되는 개체 및 테이블에 대한 간

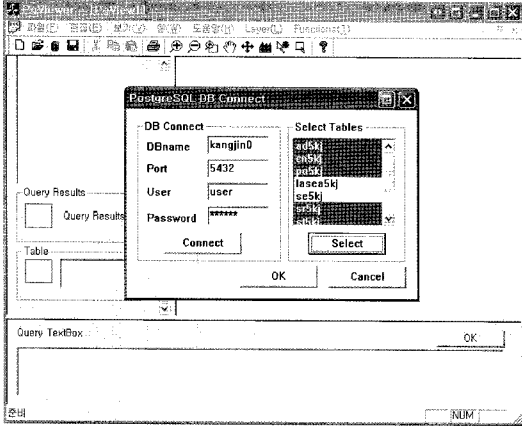


Fig. 4. Config environment for DBMS connecting.

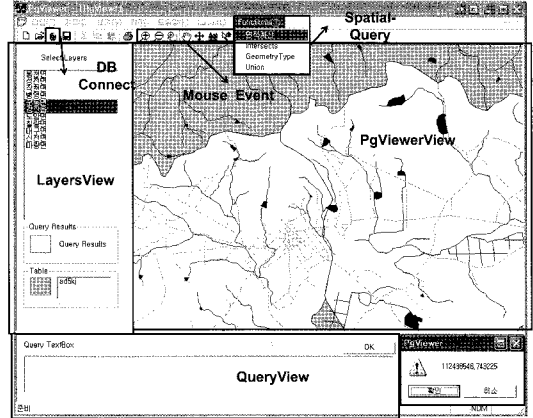


Fig. 6. Spatial-query processing in graphical user interface.

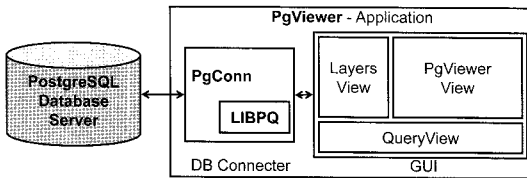


Fig. 5. Simple structure of PgViewer.

략한 정보를 보여주고 개체 선택을 보조한다. QueryView는 복잡한 공간 질의가 필요할 때 표준 SQL문을 이용하여 명령어 입력 방식으로 공간 질의를 수행할 수 있다.

본 연구는 전체적인 시스템 구현 보다는 사용자 인터페이스 중심으로 PostGIS에서 제공하는 함수 중 일부를 구현하였다. 복잡한 공간질의보다는 단순 공간질의 처리를 통해 PostGIS의 함수들이 사용자 인터페이스 환경에서 효율적으로 사용될 수 있는지 확인했다. DB Connect를 이용하여 데이터베이스에 연결하고 데이터베이스에 존재하는 데이터 중 PostGIS가 지원하는 GIS 데이터 타입(Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon, GeometryCollection)으로 되어 있는 테이블 여러 개를 가져와서 공간 질의를 할 수 있도록 구현하였다.

Fig. 6은 본 연구에서 구현한 툴 킷을 이용하여 마우스 이벤트를 통해 여러 개의 공간 GIS 개체를 선택해서 면적을 계산하는 한 가지 사례를 제시한 것으로 목표 시스템에서 기능 목록에 정의되면 PostGIS에서 지원하는 공간 질의 함수(Table 3)의 추가가 용이하도록 설계하였다.

Fig. 7은 본 연구에서 구현한 툴 킷에 Table 3에 있는 PostGIS에서 지원하는 공간 질의 함수중에서 사용

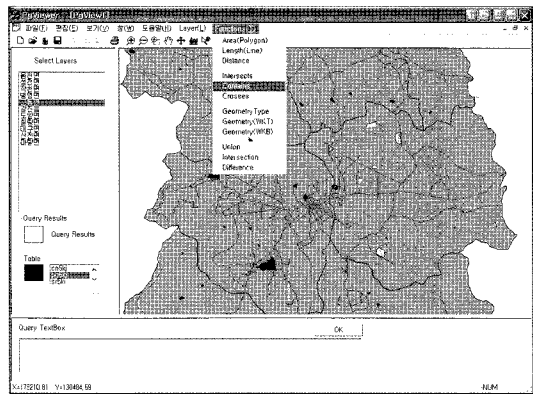


Fig. 7. Graphical user interface with spatial operators.

자 요구 사항이 있을 경우 이를 사용자 인터페이스에 추가하는 경우를 예시한 것이다.

## 4. 결론

Web 2.0 패러다임과 매쉬업의 보편화로 인하여 공간 GIS 처리 분야에서 다양한 데이터 형식으로 분산되어 있는 대용량의 공간 GIS 자료들이 혼합되어 처리되고 있고, 표준화 명세에 따른 공간 정보 상호운영이나 질의처리 성능의 향상 등이 주요 사항이 되고 있다. 이를 위해서는 안정적이고 다양한 데이터 형식을 지원하며 성능이 우수한 공간 데이터베이스 관리시스템을 이용한 공간 GIS 엔진의 적용과 보다 손쉬운 개발 환경에 필요성이 부각되고 있다. PostGIS는 다양한 데이터 형식을 지원하고 OGC 표준을 따르고 있기 때문에 OGC 표준 인터페

이스를 제공하는 다양한 공개 소스 기반의 공간 엔진이나 상업적 GIS 엔진과의 통합 연계가 가능하다. 또한 공개 소스이기 때문에 시스템 확장이 용이하고, 수요자의 다양한 요구에 대한 유연한 대응이 가능하다. 그러나 이러한 공개 소스를 이용하는 경우 엔진 처리 단계가 아닌 사용자 인터페이스를 포함하는 툴 킷 수준의 개발 환경이 제공되면 개발 과정이 보다 용이하게 진행될 수 있다.

본 연구에서는 공개 소스기반의 공간 엔진인 PostGIS에 대한 사용자 인터페이스 툴 킷을 시험 구현을 하였다. 구현 성과는 PostgreSQL 데이터베이스 서버에 있는 공간 객체들을 사용자 인터페이스 환경에서 관리하거나 조작할 수 있도록 하는 기본 설계를 바탕으로 한다. 이러한 인터페이스를 통하여 수요자의 요구에 맞는 공간 질의의 구현과 다른 데이터베이스 응용 시스템과의 연동이 가능하도록 PostGIS가 제공하는 다양한 공간 함수를 적용할 수 있다.

## 사 사

본 연구는 2009년도 한성대학교 교내연구비 지원과제임.

## 참고문헌

- Arnulf C., 2006. Introduction to Spatial Data Management with Postgis, *Proc. of 2006 GeoWeb*.
- Benthall, S., 2009. An Open Source Web GIS Solution-The OpenGeo Stack, *GEO informatics*, 12: 40-43.
- Brovelli, M. A. and D. Magni, 2004. *An Archeological Web GIS Application based on Mapserver and PostGIS*, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 34-Part5/WS12.
- Chu, C. and J. Pao, 2007. Transitioning Low Earth Orbit Satellite Archive Data from Informix (Geodetic DataBlade) to PostgreSQL (PostGIS), *presented at the FOSS4G 2007 Conference*.
- Hall, G. B. and M. G. Leahy, 2008. *Open Source Approaches in Spatial Data Handling*, Springer.
- Hass, W., 2007. Database Shootout: Benchmarking spatial DBMSs, *presented at the FOSS4G 2007 Conference*.
- Lemmen C., 2007. *Product Survey on Geo-databases*, GIM International: 21(5).
- Matthew, N. and R. Stones, 2005. *Beginning Databases with PostgreSQL from Novice to Professional*, Apress.
- Paolo C., E. Venturato, and M. Napolitano, 2006. Free, Cheap, and Powerful PostGIS, the Open Source geodatabase, *GIS Development Middle East*, 2(5): 34-39
- Paul R., 2007. Introduction to PostGIS, *presented at the FOSS4G2007 Workshops*.
- Philippe R., S. Michell, and V. Agnes, 2002. *Spatial Database with Application to GIS*, Elsevier Science.
- Ramsey, P., 2004. Introduction to PostGIS, *presented at the Open GIS 2004 Conference*.
- Ramsey, P., 2006. *PostGIS Case Studies*, Refraction Research.