

OLE/COM을 기반으로 한 OpenGIS 미들웨어 설계¹

(The Design of OLE/COM-based OpenGIS Middleware)

윤우진*, 한성룡**, 조대수***, 홍봉희****

(Yun WooJin*, Han SungRyong**, Cho DaeSoo***, Hong BongHee****)

초 록

많은 지리자료에 대한 효율적이고 효과적인 검색 및 다양한 지리자료 데이터 포맷에 대한 데이터 변환에 드는 비용을 최소화 하기 위해 지리자료의 공유, 즉 상호운용(Interoperability)에 대한 필요성이 제기되었으며 이를 위한 표준의 하나로 OpenGIS라는 표준이 만들어 지고 있다. OpenGIS는 기존의 표준과는 다른 광범위하고 미래지향적인 특징으로 인해 실질적인 GIS 표준으로 자리잡아 가고 있는 실정이지만, 국내에서는 OpenGIS 구현명세에 대한 체계적인 연구 및 여러 GIS 서버를 대상으로 한 실제 구현 사례가 거의 없는 실정이다. 이 논문에서는 컴포넌트 기반의 분산환경에 가장 근접해 있는 OpenGIS OLE/COM 구현명세에 대한 분석을 통해 Gothic, MGE, Oracle에 적용하기 위한 OGIS 데이터 제공자의 설계 및 이를 바탕으로 상호운용을 달성하기 위한 OpenGIS 미들웨어의 설계를 제시한다.

키 워 드

OpenGIS, 상호운용, OLE/COM, OLE DB, ADO, OGIS 데이터 제공자, 데이터 접근 모듈

1. 서 론

지리공간 자료는 구축을 위해서 많은 비용이 지출되고 있지만, 특정 응용프로그램의 요구조건을 만족시키기 위해 다양한 데이터 포맷으로 존재해 왔다. 이로 인해 서로 다른 GIS S/W간 공간 정보의 유통이 어렵고 데이터 구축 작업의 중복으로 많은 비용이 낭비된다는 문제점이 있었으며, 이를 해결하기 위해 상호운용에 대한 필요성이 제기되었다.

상호운용을 위한 노력으로 많은 방법이 시도되어 왔지만 기존 방법은 자료 변환이나 추상적인 개념만을 다루고 있어 실제 시스템에 적용할 경우에 어려움이 있었다. 그러나 OpenGIS는 이질적인 컴퓨팅 환경에서 공간 데이터의 상호운용을 위해 현재 소프트웨어 기술의 큰 흐름인 분산 컴퓨팅 플랫폼 기술(Distributed Computing Platform Technology)과 객체 기술(Object Technology)에 기초한 구현 명세를 제시함으로써 실제 시스템에 바로 적용할 수 있도록 하고 있다. 현재 OpenGIS에서 고려 대상으로 하는 분산 컴퓨팅 환경으로는 CORBA, OLE/COM, 그리고 자바등이 있다. 이러한 특징으로 인해

¹ 이 논문은 정보통신부 개방형 GIS 컴포넌트 S/W 개발 프로젝트 연구비 지원에 의해 이루어졌음

* 부산대학교 컴퓨터공학과 석사과정

** 인터그래프 코리아 과장

*** 부산대학교 컴퓨터공학과 박사과정

**** 부산대학교 컴퓨터공학과 교수

OpenGIS는 빠른 속도로 실질적인 GIS 표준으로 자리잡아 가고 있다.

외국에서는 OpenGIS 구현명세에 대해 활발한 연구가 이루어지고 있으며 실제 제품까지 나오고 있지만, 국내에서의 OpenGIS에 대한 연구는 대부분 추상명세로 국한되어 있으며, 구현명세에 대한 연구도 하나의 GIS 서버만을 대상으로 하고 있을 뿐 여러 GIS 서버를 대상으로 상호운용을 구현한 사례가 없는 상황이다. 이것은 OpenGIS 구현명세가 추상명세와 해당 DCP에 대한 폭 넓은 이해를 전제로 하고 있기 때문에 OpenGIS 구현명세를 이용한 시스템 설계 시 구현명세를 이해하는 데에 많은 어려움이 따르기 때문이다.

이 논문에서는 OpenGIS 구현명세 중에서 컴포넌트 기반의 분산환경에 가장 적합한 OLE/COM 구현명세에 대해 분석을 통해 상호운용을 달성하기 위한 OpenGIS 미들웨어의 설계를 제시한다. 그리고 Gothic, MGE, Oracle에 적용하기 위한 OGIS 데이터 제공자의 설계를 제시함으로써 해당 DCP에 대한 이해도를 높이며 실제 시스템에 OLE/COM 구현명세를 적용할 경우에 참고자료로 활용하도록 한다.

이 논문의 구성은 다음과 같다. 2장에서는 관련연구로써 기술적인 측면에서 OLE/COM과 OLE DB에 대해, 상호운용 측면에서 SDTS (Spatial Data Transfer Standard)와 OGD (Open Geospatial Datastore Interface)에 대해서 간단히 설명한다. 3장에서는 Gothic, MGE, Oracle간의 상호운용을 위해 이 논문에서 제시하고 있는 OpenGIS 미들웨어의 구조를 제시한다. 4장에서는 OLE/COM 구현명세에서 가장 근간이 되는 OGIS 데이터 제공자가 되기 위한 요구조건에 대해서 설명한다. 5장에서는 OpenGIS 미들웨어 구현을 위한 각 서버별 OGIS 데이터 제공자의 설계 내용을 설명한다. 마지막으로 6장에서는 결론 및 향후과제를 제시한다.

2. 관련 연구

이 장에서는 본문에서 다루어질 내용의 핵심 기술인 OLE/COM과 OLE DB에 대해서 간단히 설명하고, OpenGIS 이전 표준인 SDTS와 OGD가 가지고 있는 문제점에 대해서 살펴본다.

2.1 OLE/COM

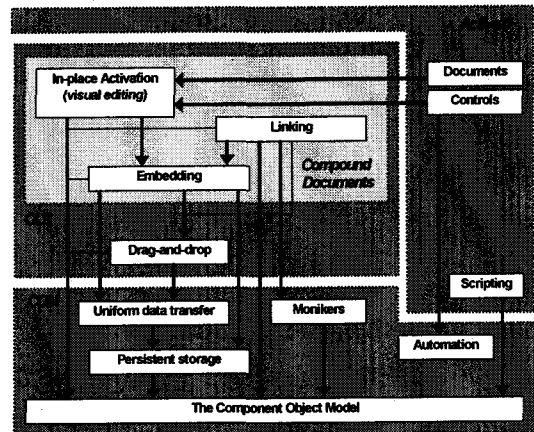


그림 1. OLE/COM 기술의 개요 및 상관관계 (GED98)

COM(Component Object Model)은 컴포넌트 소프트웨어를 위한 기반 기술이며 간단하면서도 매우 강력하고 확장성이 뛰어난 객체 모델을 제공한다. 또한 기존의 컴포넌트 소프트웨어가 가지고 있던 많은 문제점들을 컴포넌트간 호출을 위한 이진 표준, 확장 가능한 서비스 구조, 버전 관리, 언어 독립성, 위치 투명성 등을 통해 해결하였다[KB95]. COM의 확장 가능한 서비스 구조를 이용하여 다양한 분야에서 OLE/COM 기술을 이용하고 있으며 이 논문에서 설명하고 있는 OLE DB도 데이터 베이스와 OLE/COM 기술이 접목된 것이다. 그림 1은 OLE/COM 기술의 개요 및 상관관계를 나타내고 있는데 객체간의 통신을 위한 COM, 복합문서(Compound Document)를 위한 OLE(Object Linking and Em-

bedding), 인터넷 환경을 고려한 ActiveX로 나눌 수 있다[GED98].

2.2 OLE DB

데이터베이스 형태가 아닌 다양한 형태로 기존에 구축되어 있는 모든 종류의 데이터 타입에 대한 접근을 가능하게 하기 위한 방법의 하나로, 모든 데이터를 하나의 데이터베이스로 옮기고 해당 데이터베이스에 대한 API를 제공하는 방법이 있다. 이러한 접근법을 Universal 데이터베이스 접근 방법이라고 하는데, 이 방법은 많은 문제점을 가지고 있다. 첫째, 상당히 많은 양의 데이터를 해당 데이터베이스로 옮겨야 한다. 둘째, 인터넷 환경의 발달로 필요한 데이터를 반드시 가지고 있을 필요가 없다. 셋째, 데이터의 특성상 데이터베이스가 다루기에 적합하지 않을 수도 있다. 넷째, 데이터 중복에 의한 동기화 문제를 해결해야 한다.

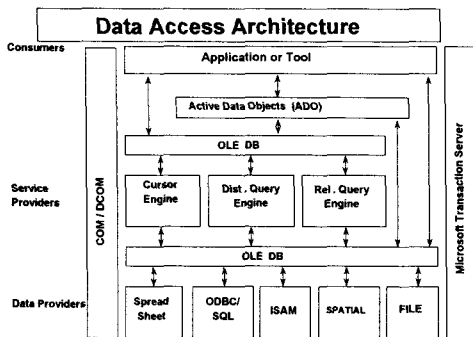


그림 2. OLE/COM 환경에서의 데이터 접근 구조[OGC98]

그림 2는 ODBC가 가지는 제한과 Universal 데이터베이스에서 제시된 문제점들을 해결하기 위한 OLE/COM 환경에서의 데이터 접근 구조를 나타내고 있다. 위의 그림에서는 OLE DB와 ADO 인터페이스를 기반으로 하여 세 가지 유형의 데이터 접근 컴포넌트로 나누고 있다. 데

이터 소비자(Data Consumer) 컴포넌트는 광범위한 데이터에 대해 접근하는 컴포넌트로서 어플리케이션이나 개발 툴에 해당되고, 데이터 제공자(Data Provider) 컴포넌트는 OLE DB 인터페이스를 통해 데이터를 제공해 주는 각 데이터 소스에 대한 일종의 래퍼(Wrapper)에 해당된다. 서비스 제공자(Service Provider) 컴포넌트는 데이터 제공자가 제공해 주지 못하는 기능을 보충해 주는 역할을 수행한다. OLE DB는 위에서 언급한 세 가지 유형의 컴포넌트가 상호작용을 할 수 있게 해주는 COM기반의 객체와 표준 인터페이스를 제공해 준다[DAS99].

2.3 SDTS와 OGD I

SDTS는 공간자료 변환 및 교환에 대한 표준으로 이질적인 환경 내에서 지리공간자료의 상호 소통 및 전송 수단으로 이용되고 있다. 이러한 특성은 기본 SDTS표준 타입을 프로파일을 통해 특정 타입으로 쉽게 확장할 수 있기 때문이다[김98]. 그러나 SDTS는 데이터 변환만을 위한 표준이고 변환단위가 데이터셋이기 때문에 1) 데이터 변환의 비능률성과 데이터의 중복저장, 2) 지리자료처리(Geoprocessing) 개념의 결여 라는 문제점을 가지고 있다.

OGDI는 다양한 지리공간 데이터에 대해 표준화된 접근 방법을 제공하는 응용 프로그래밍 인터페이스(Application Programming Interface)이다. 또한 TCP/IP 네트워크상에서 클라이언트/서버 구조를 지원하며 서로 다른 지리공간 데이터 포맷에 대해 드라이버 지향적인(driver-oriented) 접근 방법을 제공한다.[DG97] 그러나 OGDI는 OpenGIS와는 달리 객체 기술과 분산 컴퓨팅 플랫폼을 고려한 접근법이 아니기 때문에 웹이나 클라이언트/서버와 같은 제한된 환경 내에서만 상호운용을 지원한다는 단점을 가지고 있다.

3. OpenGIS 미들웨어의 구조

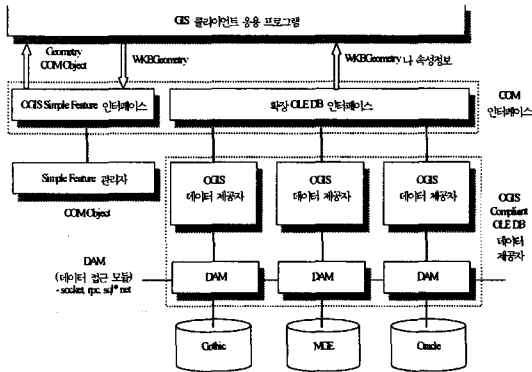


그림 3. OLE/COM을 기반으로 한 OpenGIS 미들웨어의 구조

그림 3은 Gothic, MGE, Oracle간의 상호운용을 달성하기 위해 이 논문에서 제시하고 있는 OpenGIS 미들웨어의 구조 및 상호관계를 보여주고 있다. 시스템의 구조는 크게 클라이언트가 이용할 수 있는 두 개의 COM 인터페이스를 기반으로 하고 있으며 각각의 역할은 다음과 같다. 첫째, 확장 OLE DB 인터페이스는 OpenGIS에서 정의한 GIS 개념을 추가하여 표준 OLE DB 인터페이스를 확장한다. 그리고 각각의 GIS 서버는 상호운용을 위해 확장 OLE DB 인터페이스를 구현한 OGIS 데이터 제공자와 실제 연결을 위한 데이터 접근 모듈(DAM)을 가지게 된다. 클라이언트는 확장 OLE DB 인터페이스를 통해 기하정보 및 속성정보를 얻어오게 된다. 둘째, OGIS Simple Feature 인터페이스는 OpenGIS 추상명세중 Simple Feature 부분에 해당하는 COM 인터페이스이다. Simple Feature 인터페이스의 기능은 클라이언트부터 기하정보를 WKBGeometry(Well Known Bindary Representation)형태로 입력받아 기하관련 처리 기능을 위한 COM 객체를 생성하는 것이다. Simple Feature 인터페이스에 대한 구현 모듈은 WKBGeometry라는 표준 형태로 모든 기하정보를 입력받기 때문에 서버에 관계없이 Simple

Feature 관리자라는 하나의 모듈만이 존재한다. 이 장에서는 OGIS의 Simple Feature 인터페이스와 Simple Feature 관리자에 대해서 설명하고, GIS 서버별 OGIS 데이터 제공자와 데이터 접근 모듈에 대해서는 5장에서 자세히 설명한다.

3.1 Simple Feature 인터페이스

OpenGIS 추상명세 중에서 Simple Feature에 관련된 부분이 COM인터페이스로 정의되어 있다. 구성을 살펴보면 클라이언트가 확장 OLE DB 인터페이스를 통해 얻어온 WKBGeometry로부터 기하관련 COM객체를 생성하여 기하관련 처리 기능을 이용할 수 있도록 해 주는 여러 개의 인터페이스로 이루어져 있다. Feature모델과 기하 모델로 나누어서 인터페이스를 정의하고 있는 CORBA 구현명세와는 달리, OLE/COM 구현명세에서는 기하 모델만을 정의하고 있으며 Feature 모델에 대해서는 별도의 인터페이스를 정의하지 않고 있다.

표 1. 기하 모델 인터페이스에 대한 분류

역 할	해당 인터페이스의 이름
기하관련 COM 객체 생성 및 조회	IGeometry, IGeometryCollection
기하관련 정보 검색	IGeometry, IPoint, ICurve, ILineString, ILinearRing, ISurface, IPolygon, IMultiCurve, IMultiSurface
공간 관련성 및 공간 연산	ISpatialRelation, ISpatialRelation2, ISpatialOperator
WKBGeometry 관련 변환	IWks

OLE/COM 구현명세의 기하 모델 구성을 역할에 따라 나누어 보면 1) WKBGeometry로부터 기하관련 COM 객체를 생성하고 얻어오기 위한 부분, 2) 좌표나 공간좌표 참조체계와 같은 기하관련 정보를 얻어오기 위한 부분, 3) Egenhofer의 DE-9IM에 근거한 공간 관련성 및 공간 연산을 위한 부분, 4) WKBGeometry 관련

변환을 위한 부분, 그리고 5) 기하 타입을 위한 COM 클래스 부분으로 구분할 수 있다. 표 1은 OLE/COM 구현명세중 기하모델의 인터페이스에 대한 분류를 나타내고 있다.

3.2 Simple Feature 관리자

Simple Feature 관리자는 Simple Feature관련 COM 인터페이스를 구현하는 부분으로 클라이언트로부터 WKBGeometry를 받아서 기하 관련 COM 객체를 생성해 주는 역할을 한다. 주로 클라이언트가 공간 연산과 같은 기하관련 처리 기능을 필요로 할 경우에 이용된다.

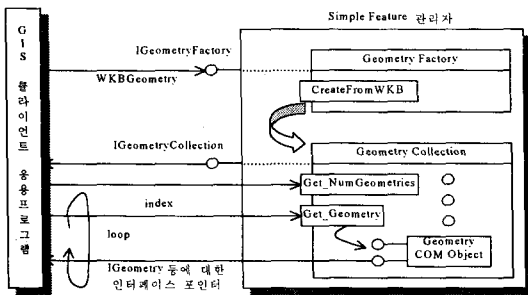


그림 4. Simple Feature 관리자를 이용한 COM 객체 생성

그림 4는 클라이언트가 WKBGeometry로부터 COM 객체를 생성하는 과정을 나타낸다. COM 객체를 생성하는 과정을 단계별로 살펴보면 다음과 같다. 첫째, 클라이언트는 WKBGeometry를 IGeometryFactory 인터페이스를 통해 Simple Feature 관리자에게 전달한다. 둘째, Simple Feature 관리자는 IGeometryFactory내의 CreateFromWKB 메소드를 호출하여 Geometry Collection을 생성하고 그 안에 해당 COM 객체들을 관리하게 된다. 그리고 IGeometryCollection에 대한 인터페이스를 클라이언트에게 전달한다. 셋째, 클라이언트는 IGeometryCollection내의 메소드를 호출하여 해당 COM 객체에 접근한다.

4. OGIS 데이터 제공자의 요구조건

OGIS 데이터 제공자는 OpenGIS OLE/COM 구현명세의 근간을 이루며, 이 논문에서 제시하고 있는 OpenGIS 미들웨어 구조에서도 가장 중요한 역할을 한다. OpenGIS OLE/COM 구현명세는 다양한 GIS 서버의 기능을 만족시키기 위해 OGIS 데이터 제공자의 요구조건을 크게 최소 사양과 권장 사양으로 나누고 있다. 최소 사양은 파일 시스템과 같이 단순한 기능을 제공하는 GIS 서버를 위한 것이며, 권장 사양은 데이터베이스와 같이 풍부한 기능을 제공하는 GIS 서버를 위한 것이다.

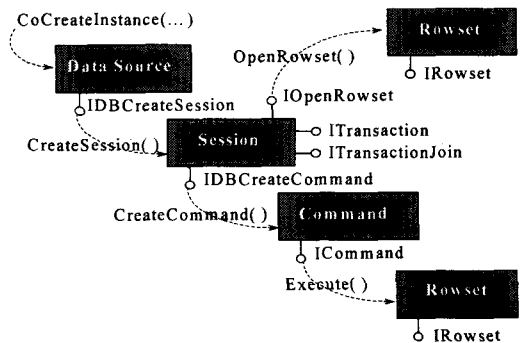


그림 5. OLE DB 를 통한 데이터 접근 (DAS99)

그림 5는 클라이언트가 OLE DB 인터페이스를 이용하여 최소 사양과 권장 사양을 지원하는 OGIS 데이터 제공자에 접근하는 모습을 보여주고 있다. 최소 사양을 지원하는 OGIS 데이터 제공자는 GIS 메타데이터를 이용할 수 없으며 Session 컴포넌트에서 정적인 형태의 Rowset 컴포넌트를 얻게 된다. 그러나 권장 사양을 지원하는 OGIS 데이터 제공자는 GIS 메타데이터와 SQL을 이용하여 Command 컴포넌트로부터 동적인 형태의 Rowset 컴포넌트를 얻을 수 있다. 최소 사양을 위한 요구조건은 OLE DB에서 모든 데이터 타입에 대해 적용되는 기본 인터페이스

이스(Base Interface)를 바탕으로 하며 다음과 같다. 첫째, OLE DB 표준에 의해 정의되어 있는 최소한의 인터페이스 집합, 즉 Session 컴포넌트 수준까지는 지원해야 한다. 둘째, 다른 OLE DB 데이터 제공자와의 구별을 위해 "OGISDataProvider"라는 컴포넌트 카테고리(Component Category)에 등록해야 한다. 셋째, 기하정보는 클라이언트/서버간 호환 가능한 표준 바이트 스트림 포맷인 WKBGeometry (Well Known Binary Representation) 형태로 노출되어야 한다. 최소 사양은 표준 OLE DB 인터페이스와 거의 동일하며 GIS 메타데이터 개념이 추가되어 있지 않다는 특징이 있다.

권장 사양을 위한 요구조건은 최소 사양과는 달리 OLE DB에서 확장된 기능을 지원하기 위한 공통 인터페이스(Common Interface)를 바탕으로, GIS 데이터 및 메타데이터를 다루기 위한 개념이 추가된 확장 OLE DB 인터페이스로 이루어져 있다. 추가된 GIS 개념으로는 1) GIS 메타데이터, 2) GIS 칼럼 메타데이터, 3) 공간 필터링 기능 등이 있는데 각각의 용도는 다음과 같다. GIS 메타데이터는 클라이언트로 하여금 GIS 정보가 어디에 있는지 그리고 어떻게 얻어내는 지를 알려주기 위해 사용된다. GIS 칼럼 메타데이터는 OLE DB의 Rowset 컴포넌트에서 실제 기하 정보 및 기하관련 스키마 정보를 얻어내기 위해 사용된다. 공간 필터링은 OLE DB의 Command 컴포넌트에서 명령어 파라미터를 통해 이루어진다.

이 논문에서는 GIS 메타데이터를 활용하기 위해 권장 사양을 지원하는 OGIS 데이터 제공자를 대상으로 한다.

5. OGIS 데이터 제공자의 설계

이 장에서는 크게 두 부분으로 나누어 설명한다. 첫째, 4장에서 설명한 OGIS 데이터 제공자의 요구조건중 권장 사양을 Gothic, MGE,

Oracle에 적용하고 각 서버별 OGIS 데이터 제공자에 대한 고려사항 및 설계를 OLE DB 컴포넌트별로 제시한다. 설계시 GIS 서버별로 공통적으로 적용되는 부분은 재사용이 가능하도록 하였다. 둘째, OGIS 데이터 제공자와 실제 GIS 서버를 연결하기 위한 데이터 접근 모듈(DAM)의 설계를 제시한다.

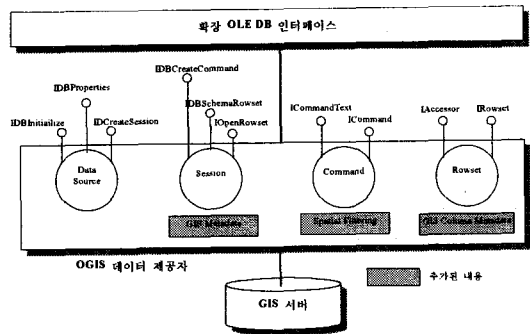


그림 6. 권장사양을 구현한 OGIS 데이터 제공자의 구조

그림 6은 권장 사양의 확장 OLE DB 인터페이스를 구현한 OGIS 데이터 제공자의 전체 구조를 보여준다. 기본적으로 OLE DB 4개의 컴포넌트와 해당 인터페이스를 모두 지원하며 Data Source 컴포넌트를 제외한 3개의 컴포넌트에 각각 GIS 개념이 추가되어 있음을 알 수 있다.

5.1 Data Source 컴포넌트

클라이언트가 원하는 데이터에 접근하기 위해 처음으로 이용하게 되는 컴포넌트로서 특정 데이터 제공자를 가리키며, GIS 서버에 적용할 때 고려사항은 다음과 같다. 첫째, 클라이언트가 GIS 서버에 연결하기 위한 올바른 권한을 가지고 있는지에 대한 사용자 인증 기능을 제공해야 한다. 둘째, 클라이언트에게 GIS 서버와의 연결을 설정하기 위해 필요한 정보를 입력받아서 GIS 서버와의 연결을 초기화 해야 한다. 셋째,

GIS 서버가 가지고 있는 속성 및 지원 가능한 기능에 대해서 알려주어야 한다. 넷째, GIS 서버에 대한 연결이 성공적으로 수행되면 Session 컴포넌트를 생성하게 되는데 다중 Session 지원 여부를 결정해야 한다.

첫 번째와 두 번째 고려사항인 사용자 인증 및 서버와의 연결을 위해 OLE DB에서 제공하는 속성 집합(Property Set)과 속성 그룹(Property Group)을 이용하고 있다. 속성 집합은 같은 식별자(GUID)를 공유하는 속성들의 집합이며 특정 OGIS 데이터 제공자를 위한 속성 집합을 만들어 사용할 수 있다. 속성 그룹은 논리적으로 연관성을 가지는 속성들의 집합이며 특정 OGIS 데이터 제공자를 위한 속성 그룹을 만들어 사용할 수 없다.

- <각 GIS 서버에 대한 연결을 초기화 할 때 사용하는 속성 정보>
- 속성 그룹 : 초기화(Initialization)
 - 속성 집합 : DBPROP_INIT
 - DBPROP_AUTH_USERID : 사용자 ID 정보
 - DBPROP_AUTH_PASSWORD : 사용자 패스워드 정보
 - DBPROP_INIT_LOCATION : 응용서버의 위치
 - DBPROP_INIT_DATASOURCE
 - Gothic : Dataset 의 이름
 - MGE : 스키마 파일의 이름
 - Oracle : Database 의 이름

그림 7. 연결 초기화를 위한 속성정보

그림 7은 각 GIS 서버에 대한 연결을 초기화 하기 위해 클라이언트가 제공해 주어야 하는 속성 정보와 그 속성 정보가 의미하는 바를 나타내고 있다. Gothic, MGE, Oracle의 경우 연결을 초기화 하기 위해 공통적으로 필요한 속성은 사용자 ID 정보, 사용자 패스워드 정보, 응용 서버의 위치 정보다. 그러나 Gothic, MGE, Oracle은 기하정보를 각각 Dataset, 스키마 파일, Database 형태로 관리하고 있다. 따라서 DBPROP_INIT_LOCATION 속성은 같은 이름을 가지지만 서버별로 다른 의미를 가지게 된다.

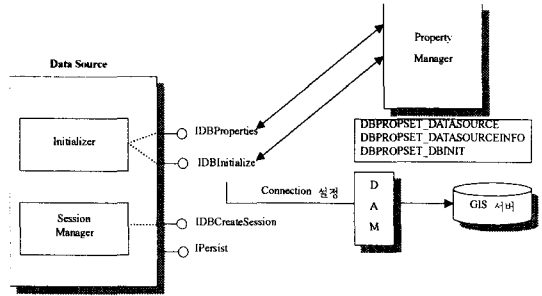


그림 8. Data Source 컴포넌트의 기본 구조

그림 8은 Data Source 관련 고려사항을 구현 하기 위한 구조를 보여주고 있다. 컴포넌트의 구조는 크게 1) Property Manager, 2) Initializer, 3) Session Manager로 이루어진다. Property Manager는 고려사항에서 제시된 기능과 관련된 속성을 관리하는데, GIS 서버가 지원하는 기능 및 연결을 초기화하는데 필요한 속성을 관리한다. Initializer는 두 번째 고려사항을 구현하기 위한 부분으로, Property Manager를 이용하여 연결에 필요한 속성 정보를 검색하고 GIS 서버에 대한 연결을 초기화한다. Session Manager는 네 번째 고려사항을 구현하기 위한 부분으로 Data Source 컴포넌트로부터 만들어 지는 Session에 대한 관리를 담당한다. 사실상 Data Source 컴포넌트는 OpenGIS와 관련해서 추가된 부분이 없기 때문에 OLE DB에서와 같은 역할을 수행하게 된다.

5.2 Session 컴포넌트

Session 컴포넌트는 GIS 서버에 대한 하나의 연결(Connection)을 의미하며 GIS 서버에 적용할 때 고려사항은 다음과 같다. 첫째, 스키마 관련 메타데이터 정보를 얻어오기 위한 방법을 결정해야 한다. 둘째, Session 컴포넌트에서 직접 Rowset 컴포넌트 형태로 데이터를 제공해 주어야 한다. 셋째, GIS 서버가 제공하는 공간 연산자의 종류를 결정해야 한다.

첫 번째 고려사항과 관련하여 각 서버별로 스키마 관리 및 접근 방법을 살펴보면 다음과 같다. 첫째, Gothic은 스키마 정보를 객체지향 데이터베이스(OODBMS)내에 저장하며 GSQL을 사용하면 Gothic 라이브러리 호출을 통해 스키마 정보를 읽어올 수 있다. 둘째, MGE는 스키마 정보를 RIS라는 인터페이스를 통해 외부 데이터베이스에 저장하며 해당 데이터베이스에 대한 SQL을 사용하여 접근할 수 있다. 셋째, Oracle은 데이터 사전(Data Dictionary) 형태의 테이블로 스키마 정보를 관리하는데 역시 SQL을 사용하여 접근할 수 있다. 따라서 세 가지 GIS 서버에 대한 GIS 메타데이터는 SQL을 사용하여 접근할 수 있다.

두 번째 고려사항은 데이터 접근 모듈(DAM)을 이용하여 구현된다. Session에서 생성되는 Rowset 컴포넌트는 Command에서 생성되는 Rowset 컴포넌트와는 달리 정적인 형태가 된다. 따라서 테이블에 있는 모든 정보를 읽어와서 정적인 형태로 클라이언트에게 제공한다. 데이터 접근 모듈에 대해서는 5.5절에서 자세히 설명한다.

표 2. 서버별 지원가능한 공간 연산자

OGIS	Oracle	Gothic	MGE
DBPROP_OGIS_TOUCHES	TOUCH	MEETS	touches
DBPROP_OGIS_WITHIN	INSIDE	WITHIN	entirely_contained_by
DBPROP_OGIS_CONTAINS	CONTAINS	CONTAINS	entirely_contains
DBPROP_OGIS_CROSSES		CROSSES	terminate_in terminate_of passes_through passes_through_by
DBPROP_OGIS_OVERLAPS	OVERLAPBDY DISJOINT OVERLAPBDY INTERSECT	OVERLAPS	overlaps
DBPROP_OGIS_DISJOINT	DISJOINT		
DBPROP_OGIS_INTERSECT	ANYINTERACT	IDENTIFY	
DBPROP_OGIS_ENVELOPE INTERSECTS			
DBPROP_OGIS_INDEX_INTE RSECTS			

세 번째 고려사항인 공간 연산자와 관련하여 OGIS에서는 DE-9IM에 근거한 9개의 속성을 제시하고 있다. 표 2는 OGIS에서 제시한 속성과 각 GIS 서버가 지원하는 공간 연산자의 관

계를 나타낸다. OGIS에서 제시하고 있는 속성은 해당 공간 연산자의 지원여부를 부울린 형태로 정의하고 있다. 그래서 표에서 OGIS에 해당되는 부분이 있으면 참이 되고, 없으면 거짓이 된다.

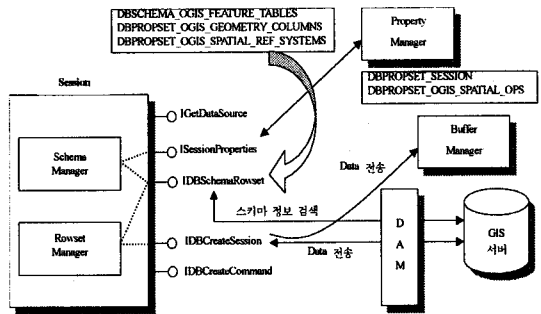


그림 9. Session 컴포넌트의 기본 구조

그림 9는 Session관련 고려사항을 구현하기 위한 구조를 보여주고 있다. 컴포넌트의 구조는 크게 1) Property Manager, 2) Schema Manager, 3) Rowset Manager, 4) Buffer Manager로 이루어져 있다. Property Manager는 세 번째 고려사항인 GIS 서버가 제공하는 공간 연산자의 종류와 관련된 속성을 관리한다. Schema Manager는 첫 번째 고려사항인 GIS 서버로부터 GIS 메타데이터를 검색하는 역할을 수행한다. 여기에서 얻어지는 GIS 메타데이터는 1) feature로써 질의를 할 수 있는 테이블, 2) 기하정보를 포함하고 있는 칼럼의 이름과 기하타입, 3) 데이터 제공자가 지원하고 있는 공간 좌표 참조체계, 4) 지원가능한 공간 연산자의 종류 등이 있다. Rowset Manager는 두 번째 고려사항을 구현하는 부분으로 Session에서 스키마나 기하 정보와 관련하여 생성되는 Rowset 컴포넌트를 관리한다. Buffer Manager는 첫 번째와 두 번째 고려사항과 관련하여 GIS 서버에서 읽어온 바이트 스트림을 관리함으로써 나중에 Rowset 컴포넌트에서 사용될 수 있게 한다.

5.3 Command 컴포넌트

Session 컴포넌트에서 얻어낸 스키마 관련 메타데이터 정보를 바탕으로 하여 SQL과 같은 텍스트 명령어를 처리하는데 사용된다. Command 컴포넌트를 GIS 서버에 적용할 경우 고려사항은 다음과 같다. 첫째, GIS 서버로 SQL과 같은 명령어를 전달하고 처리결과를 얻어내기 위한 방법을 결정해야 한다. 둘째, SQL같은 명령어에 대한 처리결과로 만들어질 Rowset 컴포넌트의 속성을 결정해야 한다. 셋째, 명령어 파라미터를 통한 공간 필터링을 지원하기 위한 방법을 결정해야 한다. 넷째, SQL 처리결과 만들어질 Rowset 컴포넌트에 대한 칼럼들의 속성을 제공해야 한다.

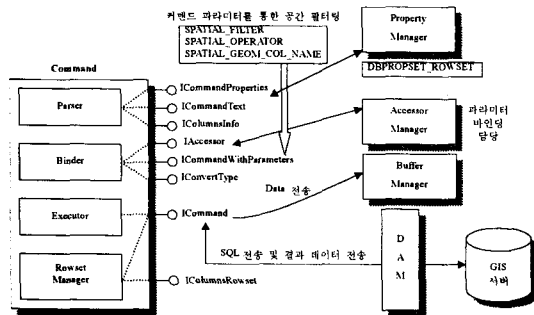


그림 10. Command 컴포넌트의 기본 구조

그림 10은 Command관련 고려사항을 구현하기 위한 기본 구조를 제시하고 있다. 기본 구조는 크게 1) Property Manager, 2) Parser, 3) Binder, 4) Executor, 5) Rowset Manager, 6) Accessor Manager로 이루어져 있다. Property Manager는 두 번째 고려사항과 관련하여 SQL 처리 후 만들어질 Rowset에 대한 속성을 관리한다. Parser는 네 번째 고려사항과 관련하여 GIS 서버로 전달될 SQL의 파싱을 통해 SQL 처리 후 만들어질 Rowset 컴포넌트의 칼럼들에 대한 정보를 제공한다. Binder는 세 번째 고려사항인 OpenGIS에서 정의하고 있는 명령어 파

라미터를 통한 공간 필터링을 지원한다. 이때 필요한 파라미터 정보는 Accessor Manager를 통해 얻게 된다. Executor는 첫 번째 고려사항과 관련하여 데이터 접근 모듈을 이용해 GIS 서버로 SQL을 전송하며 SQL 처리 결과를 Buffer Manager에 저장한다. Rowset Manager는 네 번째 고려사항인 SQL 처리 결과 및 칼럼 메타데이터 관련 Rowset 컴포넌트를 관리한다. Accessor Manager는 세 번째 고려사항인 공간 필터링을 위한 명령어 파라미터를 위해 클라이언트가 데이터 제공자로 전해 주는 바인딩 정보를 관리한다.

클라이언트는 그림 10에서 제시된 구조를 통해 Session에서 얻어낸 GIS 메타데이터 정보를 이용하여 질의문을 생성하고, 생성된 질의문을 해당 GIS 서버로 전달하여 결과를 얻어내게 된다. 각 GIS 서버에 대한 Command 컴포넌트는 그림 10의 구조를 바탕으로 해당 GIS 서버로 전달되는 SQL문만 차이를 가진다.

5.4 Rowset 컴포넌트

Rowset 컴포넌트는 클라이언트로 하여금 OGIS 데이터 제공자가 제공하는 모든 데이터에 대해 테이블 형태의 접근을 가능하게 해준다. 하나의 Rowset 컴포넌트는 여러 행들의 집합으로 이루어지며 각 행은 데이터를 가진 칼럼으로 구성된다. 클라이언트는 Rowset 컴포넌트 및 각 칼럼들에 대한 속성과 메타데이터 정보를 통해 원하는 데이터를 얻을 수 있으며, 네비게이션 기능을 통해 원하는 행의 위치로 쉽게 이동할 수 있다.

그림 11은 Rowset 컴포넌트를 구성하는 데이터 구조를 보여주고 있으며 각각의 기능은 다음과 같다. 첫째, Accessor는 OGIS 데이터 제공자가 클라이언트의 버퍼로 데이터를 전송하는데 필요한 바인딩 정보를 관리한다. 둘째, Row Cache는 GIS 서버로부터 읽어온 데이터를 일시

저장하며 행 단위로 관리해 주는 역할을 한다. 셋째, Row Handle은 Row Cache에 행 단위로 저장되어 있는 데이터를 구별해 주며 클라이언트가 얼마나 참조하고 있는지 관리하는 역할을 수행한다.[DAS99]

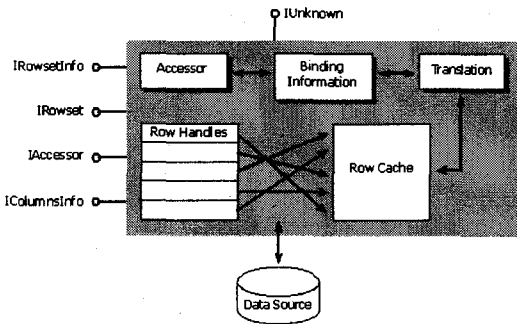


그림 11. Rowset 컴포넌트의 데이터 구조(DAS99)

Rowset 컴포넌트를 GIS 서버에 적용할 경우 고려사항은 다음과 같다. 첫째, Rowset을 구성하는 칼럼에 대한 메타데이터 정보에 기하 관련 속성을 추가해야 한다. 둘째, GIS 서버로부터 읽어들인 데이터를 행 단위로 관리해 주기 위한 방법을 결정해야 한다. 셋째, 클라이언트에게 실제 데이터를 전송해 주는 데 필요한 바인딩 정보를 관리해야 한다. 넷째, Rowset내 행 단위로 관리되는 데이터간에 어느 정도의 네비게이션 기능을 제공할 것인가를 결정해야 한다.

그림 12는 Rowset관련 고려사항을 구현하기 위한 기본 구조를 제시하고 있다. 기본 구조는 크게 1) Property Manager와 Rowset Property Manager, 2) Rowset Binder, 3) Row Cache Manager, 4) Row Handle Manager, 5) Accessor Manager로 이루어져 있다. 첫 번째 고려사항과 관련하여 Property Manager는 Rowset 컴포넌트간에 공유되어 지는 속성을 관리하며, Rowset Property Manager는 해당 Rowset 및 Rowset을 이루는 칼럼들에 대한 속성 정보를 관리한다. Rowset Binder는 Accessor Manager를 통해 바인딩 정보를 번역한다. 두 번째 고려사항과 관련하여 Row Cache Manager는 Row Cache에 저장되어 있는 데이터를 클라이언트에게 전송한다. 그리고 Row Handle Manager는 Buffer Manager로부터 데이터를 읽어와서 Row Cache를 활성화시킨다. Accessor Manager는 네 번째 고려사항과 관련하여 클라이언트가 실제 데이터를 버퍼로 읽어들이기 위해 제공하는 칼럼 바인딩 정보를 관리한다.

각 GIS 서버에 대한 Rowset 컴포넌트는 어느 정도의 네비게이션 기능을 제공하느냐에 따라 달라질 수 있지만, 이 논문에서는 순차적인 네비게이션 기능만을 고려하고 있다. 따라서 각 GIS 서버에 대한 Rowset 컴포넌트는 그림 12와 같이 동일한 구조를 갖는다.

5.5 데이터 접근 모듈(DAM)

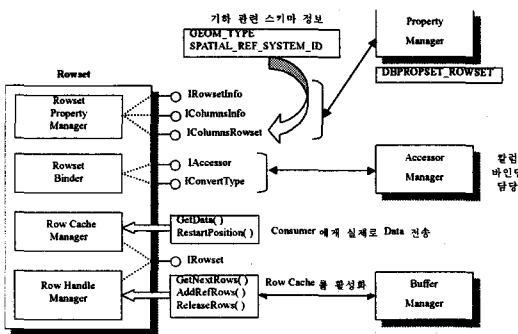


그림 12. Rowset 컴포넌트의 기본 구조

OGIS 데이터 제공자의 각 OLE DB 컴포넌트에서 실제 GIS 서버로부터 GIS 메타데이터나 기하 정보를 읽어 올 필요가 있을 경우에는 데이터 접근 모듈을 이용하게 된다. OGIS 데이터 제공자는 질의 처리기와 통신모듈을 사용하여 해당 GIS 서버로 SQL을 보내게 되며 각 GIS 서버의 응용 서버는 SQL을 받아서 GIS 서버로 SQL을 전달하고 처리 결과를 받아서 OGIS 데이터 제공자로 보내게 된다. 그림 13은 이러한

데이터 접근 모듈의 구조를 보여주고 있다.

수 있다.

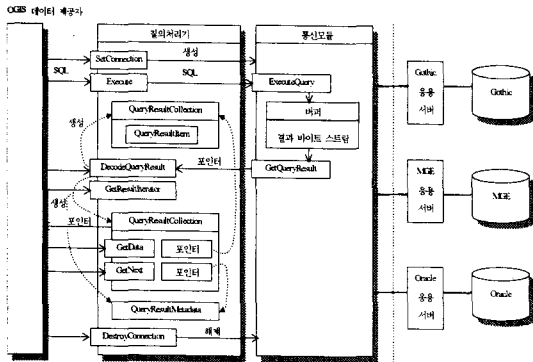


그림 13. 데이터 접근 모듈의 구조

데이터 접근 모듈의 구성을 살펴보면 GIS 서버마다 하나씩의 응용 서버가 있지만 OGIS 데이터 제공자를 위한 질의 처리기와 통신 모듈은 하나만 존재한다. 이것은 SQL 처리 결과를 그림 14와 같이 소켓을 통해 테이블 구조의 정해진 바이트 스트림 포맷으로 넘겨 받기 때문이다.

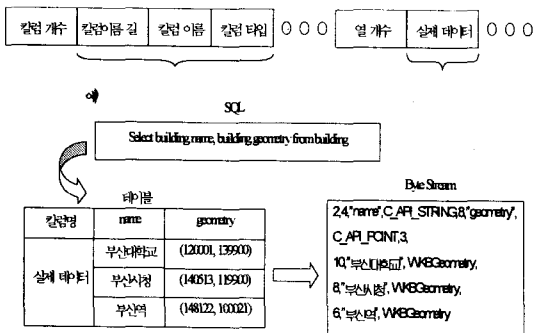


그림 14. SQL 처리 결과에 대한 바이트 스트림 포맷

그림 14의 구조를 사용할 경우 가지게 되는 이점은 다음과 같다. 첫째, 데이터 제공자 구현 시 GIS 서버와의 연결을 위해 필요한 부분을 최소화 시킬 수 있다. 둘째, Rowset 컴포넌트의 테이블 구조와 비슷한 형태로 데이터를 읽어옴으로써 좀 더 쉽게 Rowset 컴포넌트를 구현할

6. 결론

여러 GIS 서버간에 상호운용을 달성하기 위해 OpenGIS에서는 여러 분산 컴퓨팅 환경에 기초한 구현 명세를 제시하고 있다. 이 논문에서는 컴포넌트 기반의 분산 컴퓨팅 환경에 가장 근접해 있는 OpenGIS OLE/COM 구현명세에 대한 분석을 통해 Gothic, MGE, Oracle간의 상호운용을 지원할 수 있는 OpenGIS 미들웨어의 설계를 제시하였다. 그리고 각 서버별 OGIS 데이터 제공자에 대한 고려사항 및 설계를 OLE DB 컴포넌트별로 제시하였다.

앞으로의 향후 연구 과제로는 OpenGIS 데이터 제공자의 기능이 미약할 경우 이를 보완해 주기 위한 OLE DB 서비스 제공자에 대한 연구와 이 논문에서 설계한 OpenGIS 미들웨어에 대한 실제 구현을 통해 여러 GIS 서버간에 상호운용을 입증하는 것이다.

참고문헌

- [OGC98] Open GIS Consortium, OpenGIS Simple Features Specification for OLE/COM Revision 1.0, 1998
- [DAS99] Microsoft Data Access SDK 2.1 MSDN Help, 1999
- [MS98] Microsoft Press, Microsoft OLE DB 2.0 Programmer's Reference and Data Access SDK, 1998
- [JAB96] Jose A. Blakeley, Data Access for the Masses through OLE DB, SIGMOD'96 Montreal, Canada, 1996
- [DON98] Don Box, Essential COM, Addison Wesley, 1998
- [GED98] Guy Eddon, Henry Eddon, Inside Distributed COM, Microsoft Press,

- 1998
- [KB95] Kraig Brockschmidt, Inside OLE 2nd Edition, Microsoft Press, 1995
- [Orf96] Robert Orfali, Dan Harkey, Jeri Edwards, The Essential Client/Server Survival Guide, John Wiely and Sons, 1996
- [MGE98] Intergraph, MGE Foundations, 1998
- [OSC97] Oracle, Oracle Spatial Catridge User's Guide and Reference, 1997
- [DG97] Denis Gouin, Paul Morin, Gilles Clement, Christian Larouche, Solving the Geospatial Data Barrier Open Geospatial Datastore Interface(OGDI), CIG GEOMATICA Journal, Volume 51, Number 3, pp 278-287, 1997
- [김98] 김계현, 김희두, 지리정보 활용을 위한 영상데이터의 표준화, 98 개방형 GIS 학술회의 논문집, 1998
- [조98] 조대수, 최진호, 류우석, 안경환, 홍봉희, 객체 지향 GIS DB를 위한 GSQL의 설계 및 구현, 98 데이터베이스 연구회지, 제 13권 4호, 1998
- [안98] 안경환, 조대수, 홍봉희, CORBA를 이용한 클라이언트/서버 GIS의 설계 및 구현, 한국정보과학회 98 가을 학술발표논문집, 제 25권 2호, 1998

- [금97] 금완호, 김승민, 노대식, 노태원, 최원기, 김일곤, 박영철, www에서 OLE DB를 기반으로 한 지리정보시스템 컴포넌트 설계 및 구현, 한국정보과학회 97 가을 학술발표논문집, 제 24권 2호, 1997

윤 우 진

- 1998년 부산대학교 컴퓨터공학과 졸업(공학사)
 1998년~현재 부산대학교 대학원 컴퓨터공학과, 석사과정
 관심분야 : OLE/COM, 컴포넌트 소프트웨어, 개방형 GIS

조 대 수

- 1995년 부산대학교 컴퓨터공학과 졸업(공학사)
 1997년 부산대학교 대학원 컴퓨터공학과 졸업(공학석사)
 1997년~현재 부산대학교 대학원 컴퓨터공학과, 박사과정
 관심분야 : 객체지향 공간 질의어, 인터넷 GIS, 공간 데이터 모델링

홍 봉 희

- 1982년 서울대학교 전자계산기공학과 졸업(공학사)
 1984년 서울대학교 대학원 전자계산기공학과 졸업(공학석사)
 1988년 서울대학교 대학원 전자계산기공학과 졸업(공학박사)
 현재 부산대학교 공과대학 컴퓨터공학과 부교수
 관심분야 : 개방형 GIS, 병렬 GIS, 공간뷰, 공간 SQL