

제 1 부 전국교통DB와 개방형 GIS 컴포넌트의 연계방안

제1장 서 론

제2장 개방형 GIS 컴포넌트 개요

제3장 데이터 제공자 컴포넌트

제4장 핵심공통 컴포넌트

제5장 교통DB와 개방형 GIS 컴포넌트

제6장 교통DB와 GDF

제7장 결 론

목 차

제 1 부

제1장 서론	1
제1절 연구의 필요성 / 3	
제2절 연구의 목표 및 내용 / 4	
제3절 연구의 기대효과 / 5	
제2장 개방형 GIS 컴포넌트 개요	7
제1절 개방형 GIS 컴포넌트의 정의 / 9	
제2절 컴포넌트 GIS 관련 국내외 기술 개발 동향/ 12	
제3절 개방형 GIS 컴포넌트 기술 개발의 필요성 / 18	
제4절 개방형 GIS 컴포넌트 소프트웨어의 구성 / 24	
제3장 데이터 제공자 컴포넌트	27
제1절 데이터 제공자 컴포넌트의 개념 / 29	
제2절 데이터 제공자 컴포넌트의 필요성 / 31	
제3절 데이터 제공자 컴포넌트의 기능 / 32	
제4절 데이터 제공자 컴포넌트의 장점 / 45	
제5절 데이터 제공자 컴포넌트의 구성 / 46	
제6절 결 론 / 54	
제4장 핵심 공통 컴포넌트	55
제1절 핵심 공통 컴포넌트의 개념 / 57	
제2절 핵심 공통 컴포넌트의 기능 및 구성 / 58	
제3절 핵심 공통 컴포넌트의 수행 예 / 68	
제4절 결 론 / 74	

제5장 교통DB와 개방형 GIS 컴포넌트	75
제1절 교통DB 개요 / 77	
제2절 GIS 정보의 공유 및 유통 방안 / 78	
제3절 교통DB의 공유 및 정보 유통 방안 / 91	
제4절 결 론 / 97	
 제6장 교통DB와 GDF	 99
제1절 GDF 개요 / 101	
제2절 GDF와 교통DB의 연계 방안 / 103	
 제7장 결 론	 107

제 II 부

제1장 서 론	1
제1절 연구의 배경 및 목적 / 3	
제2절 연구의 구성 및 결과 / 3	
 제2장 교통존 설정 개요	 5
제1절 교통존의 정의, 존재이유 / 7	
제2절 교통존의 일반적인 구분원칙 / 7	
제3절 교통존 설정에 따른 결과의 상이성 / 8	
 제3장 교통분석유형별 교통존 설정 유형 분석 및 교통망과의 연동방안	 11
제1절 거시적 국가교통계획에서의 교통존 설정 / 13	
제2절 미시적 교통영향평가에서의 교통존 설정 / 15	
제3절 교통망과 연동한 교통존설정: 이론과 실제 / 17	

제4장	지리정보시스템을 이용한 교통존설정 기법	21
제1절	개 요 / 23	
제2절	이론적 배경 / 25	
제3절	TAZ 구성 체계의 개발 / 31	
제4절	모형간 특성비교 / 41	
제5장	결 론	45
제1절	결 론 / 47	
제2절	향후과제 / 48	
참고문헌	49

표 목 차

제 I 부

<표 3- 1>	피쳐 테이블 관련 메타 정보 형식	34
<표 3- 2>	공간 정보 컬럼 관련 메타 정보 형식	36
<표 3- 3>	공간 좌표계 관련 메타 정보 형식	38
<표 3- 4>	피쳐 들의 최대 맵 영역 관련 메타 정보 형식	40
<표 3- 5>	OGC 데이터 제공자 컴포넌트의 공간 연산자 구현 사양	43
<표 3- 6>	Data Source 객체의 인터페이스 구성	50
<표 3- 7>	Session 객체의 인터페이스 구성	51
<표 3- 8>	Command 객체의 인터페이스 구성	52
<표 3- 9>	Rowset 객체의 인터페이스 구성	53
<표 4- 1>	OGC Geometry 클래스의 속성 및 메소드 구성	60

제 II 부

<표 3- 1>	서울시 교통센서스에서의 소존 구분 현황	13
<표 3- 2>	서울시 센서스에서의 대존 구분 현황	14
<표 3- 3>	실시 설계에서의 교통존 구분내역	16
<표 3- 4>	존과 네트워크의 관계	20
<표 4- 1>	TAZ 구성 체계 내의 실행모듈	32
<표 4- 2>	TAZ 구성 체계에서 이용하는 데이터 파일	33
<표 4- 3>	응집성 강행 모형	34
<표 4- 4>	행정구역 강행 모형	35
<표 4- 5>	응집성 및 행정구역 강행 기준값	40
<표 4- 6>	수정된 시뮬레이티드 애닐링 실행결과	41
<표 4- 7>	수정된 Ward 알고리즘 실행결과	42
<표 4- 9>	BSU 수 변화 허용비가 다를 때 알고리즘 수행결과	43
<표 4-10>	수정된 알고리즘의 수행시간 비교	43

그림목차

제 1 부

<그림 2- 1>	조립식 블록 개념의 컴포넌트 구성 예제	9
<그림 2- 2>	GIS 시장 흐름의 변화	10
<그림 2- 3>	OGC의 추상 사양(Abstract Specification)	16
<그림 2- 4>	OGC의 구현 사양(Implementation Specification)	17
<그림 2- 5>	급격한 정보화에 따른 소프트웨어의 위기	19
<그림 2- 6>	소프트웨어 개발의 패러다임 변화	21
<그림 2- 7>	기존 응용 시스템에서 컴포넌트 시스템으로의 재구성시 장점	22
<그림 2- 8>	개방형 GIS 컴포넌트의 출현 과정	23
<그림 2- 9>	개방형 GIS 컴포넌트의 전체적인 시스템 구성	24
<그림 3- 1>	데이터 제공자 컴포넌트의 개념도	29
<그림 3- 2>	OGC에서 제시하는 데이터 제공자 컴포넌트	30
<그림 3- 3>	데이터 제공자 컴포넌트를 이용한 공간 정보 유통 개념	32
<그림 3- 4>	피쳐 테이블 관련 메타 정보의 수행 예제	35
<그림 3- 5>	공간정보 컬럼 관련 메타 정보 수행 예제	37
<그림 3- 6>	공간좌표계 관련 메타 정보 수행 예제	39
<그림 3- 7>	피쳐들의 최대 맵 영역 메타 정보 수행 예제	41
<그림 3- 8>	공간 여과 객체: LineString, 공간 연산자	44
<그림 3- 9>	공간 여과 객체: Polygon, 공간 연산자	44
<그림 3-10>	SDE,ZEUS, GEOMania에 동시에 접근하여 공간 연산 수행 예	46
<그림 3-11>	GIS 소프트웨어에 따른 데이터제공자 컴포넌트 시스템 모델	47
<그림 3-12>	OLE DB 컴포넌트의 각 객체들간의 관계	49

<그림 4- 1>	핵심 공통 컴포넌트의 전체적인 구성도	58
<그림 4- 2>	Geometry Object 모델	59
<그림 4- 3>	Spatial Reference System Object 모델	61
<그림 4- 4>	GeoDAO의 컴포넌트 구성도	63
<그림 4- 5>	MapDraw의 컴포넌트 구성도	64
<그림 4- 6>	GeoDAO 컴포넌트와 MapDraw 컴포넌트의 연결 관계	65
<그림 4- 7>	Network 분석 컴포넌트의 네트워크 데이터 모델	67
<그림 4- 8>	Visual Basic을 이용한 핵심 공통 컴포넌트의 기본 수행 예	69
<그림 4- 9>	핵심 공통 컴포넌트의 주제도 수행 예	70
<그림 4-10>	핵심 공통 컴포넌트의 공간 검색 수행 예	71
<그림 4-11>	MBMap의 Chart 기능 수행 예	72
<그림 4-12>	MBMap의 Overlaps 공간 연산 수행 예	72
<그림 4-13>	Network 분석 컴포넌트의 수행 예	73
<그림 5-1>	데이터 변환을 통한 공간 정보의 상호 공유 개념	79
<그림 5-2>	공간 데이터의 변환 방법을 이용한 공간 정보 상호 공유의 예	81
<그림 5-3>	공간 데이터의 직접 접근을 통한 공간 정보 상호 공유의 개념	83
<그림 5-4>	공간 데이터의 직접 공유 방법을 이용한 상호 공유의 예	85
<그림 5-5>	OGC의 Catalog Service 구현 사양	86
<그림 5-6>	공간 정보 유동 체계 구성도	88
<그림 5-7>	개방형 GIS 컴포넌트를 이용한 교통DB 접근 방법	93
<그림 5-8>	다양한 기관들이 교통DB를 접근하기 위한 시스템 구성도	94
<그림 5-9>	교통DB의 공간 정보 유동 체계 구성도	96
<그림 6-1>	GDF 또는 전국교통DB의 포맷 변환을 통한 정보 공유 방안	104
<그림 6-2>	On-Line상에서 전국교통DB와 차량항법용 DB의 상호 공유 방안	105

제 II 부

<그림 2-1>	존의 centroid 위치	9
<그림 2-2>	3존체제에서의 교통량	9
<그림 2-3>	5존체제에서의 교통량	9
<그림 2-4>	3존체제시의 결과	10
<그림 2-5>	5존체제시의 결과	10
<그림 3-1>	Network for Regional Analysis	18
<그림 3-2>	Network for Corridor Analysis	19
<그림 4-1>	계층적 군집형성을 위한 트리	28
<그림 4-2>	TAZ 구성 체계 개념도	32
<그림 4-3>	수정된 Ward 알고리즘 수행결과	44
<그림 4-4>	수정된 시뮬레이티드 애닐링과 k-means 알고리즘 수행결과	44

제1장 서론

제1절 연구의 필요성

제2절 연구의 목표 및 내용

제3절 연구의 기대 효과

제1절 연구의 필요성

최근 GIS 기술이 급속히 발전함에 따라 다양한 GIS 소프트웨어들이 등장하게 되었으며 사용자들은 업무에 적합한 소프트웨어를 선정하여 응용 시스템을 구축하고 있다. 실제로 GIS의 주요 수요 기관인 지자체 및 각 행정 부처들은 ArcInfo, ZEUS, MapInfo, GEOMania, SmallWorld 등의 다양한 GIS 소프트웨어들을 채택하여 이용하고 있다. 그리고 GIS 사업에 핵심을 구성하는 GIS DB의 구축은 사업 추진의 주체인 기관 또는 용역 업체에 의하여 DB 표준, 용어 표준 또는 주제도 표준과 같은 표준 사양을 근거로 하지 않고 임의의 형태로 구축되는 경우가 많다. 이러한 경우에 사업을 추진한 기관 내에서 단일한 응용 시스템으로 이용하고자 한다면 전혀 문제가 되지 않는다. 그러나 서로 상이한 GIS 소프트웨어와 GIS DB를 사용하는 타 기관들과의 정보 교환이나 공유가 필요할 경우에는 서로 상이한 형식의 GIS DB와 GIS 소프트웨어로 인해 야기되는 문제를 해결하기 위하여 공간 정보의 유통 및 공유를 위한 새로운 시스템을 별도로 구축해야 하는 문제가 발생한다. 이러한 정보 유통 시스템은 상이한 GIS DB와 소프트웨어로 이용되는 데이터 또는 기관별로 모두 구축한다는 것은 비용과 시간 면에서 매우 불합리하다고 할 수 있다. 따라서 이 기종 분산 환경에서 상이한 GIS 소프트웨어간에 공간 정보의 유통을 효율적으로 지원할 수 있는 방안을 마련하는 것이 매우 중요하다고 볼 수 있다.

본 연구에서는 OGC(OpenGIS Consortium)와 ISO/TC211에서 제시하는 국제 표준 사양을 수용하고 상호 상이한 GIS 서버간의 공간 정보의 유통 및 교환을 용이하게 지원하기 위한 방안으로서 개발되고 있는 “개방형 GIS 컴포넌트 기술 개발 사업”의 결과물을 적용함으로써 현재 구축되고 있는 전국교통DB를 타 기관과 상호 공유하는 방안에 대하여 살펴보고자 한다. 구체적으로 본 연구의 일차적인 목적은 현재 SDE 서버와 Oracle 데이터베이스를 이용하여 구축되고 있는 전국교통DB를 이 기종 분산 환경에서 상이한 형식의 GIS DB와 GIS 소프트웨어를 사용하는 사업 또는 기관간에 상호 유통할 수 있도록 하는 방법론을 제시하는 것이다. 특히 “개방형 GIS 컴포넌트 기술 개발” 사업의 결과물인 “데이터 제공자 컴포넌트”, “핵심 공통 컴포넌트” 그리고 “정보 유통 컴포넌트”를 적용함으로써 최소한의 비용과 시간으로 전국교통DB의 정보 유통 문제를 해결하는 것을 주 목적으로 하고 있다.

제2절 연구의 목표 및 내용

본 연구의 전체적인 주요 내용은 개방형 GIS 컴포넌트의 특징 및 개념 조사, OGC에서 제시하는 OLE/COM 사양의 GIS 컴포넌트 구현 사양 조사, GIS DB의 정보 유통의 핵심 역할을 수행하는 데이터 제공자 컴포넌트 및 핵심 공통 컴포넌트의 특징 분석, SDE와 Oracle 데이터베이스를 이용하여 구축되고 있는 전국교통DB와 개방형 GIS 컴포넌트의 연계 방안 조사 그리고 교통DB와 차량 항법용 표준 수치지도 형식인 GDF(Geographic Data Files)와의 연계 방안 조사등을 통하여 교통DB 및 GDF 형식의 데이터를 효율적으로 공간 유통시키기 위한 방법을 도출하는 것이다. 이를 위한 세부 연구 내용과 범위는 다음과 같다.

첫째, 개방형 GIS 컴포넌트의 기본적인 개념을 파악하고 개방형 GIS 컴포넌트의 필요성 그리고 추진 방향에 대하여 살펴볼 것이다. 구체적으로 개방형 GIS 컴포넌트의 정의, 개방형 GIS 컴포넌트 소프트웨어 기술 개발의 필요성, 본 분야에 대한 국내외 기술 동향 그리고 개방형 GIS 컴포넌트의 소프트웨어 구성 방법 등에 대하여 자세히 살펴볼 것이다.

둘째, OGC에서 제시하는 "OpenGIS Simple Feature Specification For OLE/COM" 구현 사양에 대하여 조사하고, 특히 본 구현 사양 중에서 데이터 제공자 컴포넌트와 핵심 공통 컴포넌트의 구현 사양을 자세히 파악하여 적용함으로써 전국교통DB와 개방형 GIS 컴포넌트의 결과물을 연계할 수 있는 방안을 제시하고자 한다. 그러므로 본 연구에서는 이러한 데이터 제공자 컴포넌트와 핵심 공통 컴포넌트의 주 역할 및 기능 그리고 구축 방법에 대하여 자세히 살펴볼 것이다.

셋째, 구체적으로 SDE와 Oracle 데이터베이스를 이용하여 구축되어 있는 전국교통DB를 공간 유통시키기 위한 방법을 제시하고자 한다. SDE 데이터 제공자 컴포넌트를 이용한 전국교통DB의 공간 유통 방안 및 ZEUS, GEOMain, SmallWorld 등과 같이 다양한 GIS 소프트웨어와의 상호 공간 정보의 공유 방법 및 예시 모델을 제시할 것이다.

끝으로, 차량 항법용 수치지도 표준 데이터 저장 형식인 GDF 파일과 SDE와 Shape 형식으로 구성되어 있는 전국교통DB와의 데이터 공유 방법에 대한 방안을 제시하고자 한다.

제3절 연구의 기대 효과

우선 개방형 GIS 컴포넌트의 기본적인 개념을 파악하고 국내에서 개발되고 있는 개방형 GIS 컴포넌트의 결과물을 활용함으로써 이 기종 분산환경에서 다양한 GIS 소프트웨어를 사용하고 있는 다른 기관들과 전국교통DB의 정보를 자유로이 공유하기 위한 방법론을 습득하게 된다.

OGC와 ISO/TC211에서 제시하는 컴포넌트를 위한 국제 표준 사양에 대한 조사를 토대로 공간 정보 유통의 핵심 기술을 구성하는 데이터 제공자 컴포넌트와 핵심 공통 컴포넌트에 대한 기술을 습득할 수 있다.

다양한 차량 항법용 수치지도 표준 저장 형식인 GDF의 구조 및 교통DB와의 공유 방법에 대한 기술을 습득할 수 있다.

끝으로 이러한 교통DB의 공간 정보 유통을 위한 방법을 습득함으로써 향후 기관별 또는 사업별 공간 정보 유통 사업에 직접적으로 적용하기 위한 기반 연구로 활용되어질 수 있다.

제2장 개방형 GIS 컴포넌트 개요

제1절 개방형 GIS 컴포넌트의 정의

제2절 컴포넌트 GIS 관련 국내외 기술 개발 동향

제3절 개방형 GIS 컴포넌트 기술 개발의 필요성

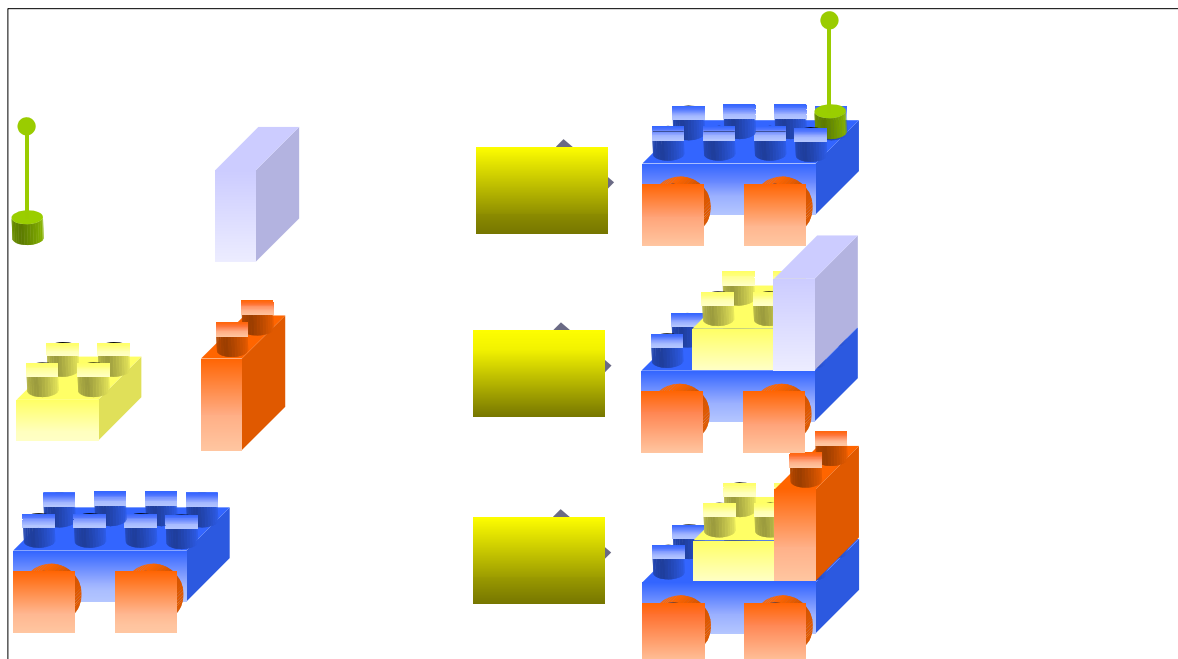
제4절 개방형 GIS 컴포넌트 소프트웨어의 구성

제1절 개방형 GIS 컴포넌트의 정의

1. 개방형 컴포넌트의 개념

컴포넌트는 정의된 인터페이스를 통하여 특정 서비스를 제공할 수 있는 소프트웨어의 최소 단위로서, 재사용성과 확장성이 매우 뛰어난 특징을 가지고 있다. 이러한 컴포넌트들은 외부에 자신들이 제공하는 서비스들을 명확하게 명시하고 있으며, 다른 컴포넌트들과는 독립적으로 동작하는 특징을 가지고 있다. 또한 동일한 서비스를 제공하는 다른 컴포넌트들로 언제든지 교체가 가능하며, 컴포넌트의 내부의 복잡한 물리적인 구현 사항을 파악하지 않고도 용이하게 사용할 수 있다.

컴포넌트 기반 소프트웨어는 쉽게 설명해서 다음 <그림 2-1>과 같이 조립식 블록과 유사한 개념을 가지고 있다고 할 수 있다. 예를 들어 <그림 2-1>의 첫 번째 조립의 경우처럼 '네트워크 처리 컴포넌트'와 '데이터베이스 접근 컴포넌트'를 조립함으로써 데이터베이스로부터 읽어들이는 데이터를 네트워크를 통해서 전송할 수 있는 응용 프로그램을 간단히 구성할 수 있는 특징을 가지고 있다.

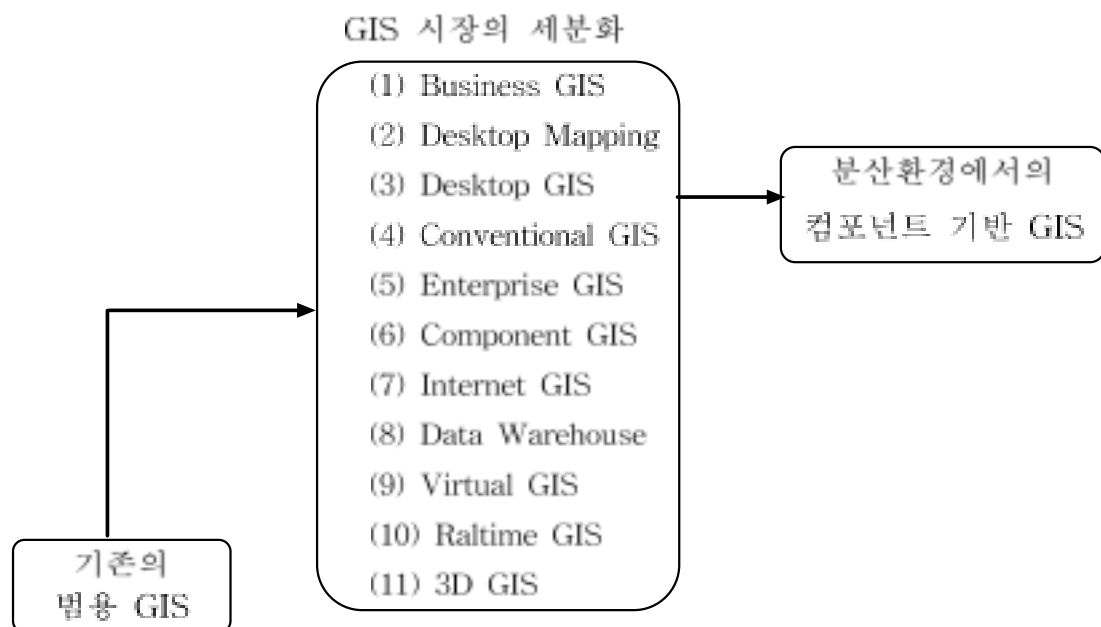


<그림 2-1> 조립식 블록 개념의 컴포넌트 구성 예제

특히 개방형(Open) 컴포넌트는 기존의 컴포넌트에 있어서 인터페이스를 표준화시키고 외부에 인터페이스의 접근 방법 및 내부 구현 사양을 완전히 공개함으로써 누구나 용이하게 접근하여 사용할 수 있고 같은 기능의 컴포넌트를 구현할 수 있는 특징을 가지고 있다. 그러므로 이러한 개방형 컴포넌트가 구축되기 위해서는 반드시 인터페이스 표준화의 문제가 선결되어야 하는 것이 매우 중요하다.

2. 컴포넌트 기반 GIS

최근에 GIS 기술이 급격히 발전함에 따라 기존의 축적된 GIS 기술들을 바탕으로 분산 환경 기반 GIS 기술, 상호운용성(Interoperability)을 지원할 수 있는 GIS 기술, 표준화를 지원하는 컴포넌트 기반의 GIS 기술 등과 같은 새로운 패러다임의 GIS 기술들이 발전하기 시작하였다. 이러한 GIS 기술 변화의 움직임은 이미 선진 GIS 업체들에 의하여 주도되고 있다. 기존의 하나의 범용 GIS 소프트웨어가 모든 업무를 처리하는 방식에서 Internet GIS, Desktop GIS, Enterprise GIS, Professional GIS 등과 같이 다양한 분야로 GIS 시장의 세분화, 세분화되고 전문화된 GIS 시장을 효율적으로 충족시키기 위한 컴포넌트 기반 GIS 소프트웨어의 개발 등이 현재 GIS 시장의 핵심으로 떠오르고 있다. 다음 <그림 2-2>는 GIS 소프트웨어 시장이 어떠한 방향으로 흘러가는지를 보여주고 있다.



<그림 2-2> GIS 시장 흐름의 변화

실제로 이러한 컴포넌트 기반 소프트웨어 시장은 GIS 분야뿐만 아니라 모든 소프트웨어 시장에서 2002년까지 모든 응용 시스템 개발의 60% 이상을 컴포넌트 기반 소프트웨어가 대체할 것이라고 가트너 그룹(Gartner Group)이 말하고 있다. 그리고 현재 세계의 선도적인 GIS 업체들은 이미 이러한 시장흐름을 파악하고 실제로 다양한 컴포넌트 기반 GIS 소프트웨어 솔루션을 발표하고 있다.

일반적으로 범용의 GIS 소프트웨어에 대하여 컴포넌트 기반의 소프트웨어 개발 방법론을 적용시켜 컴포넌트별로 분류하여 보면 크게 GIS 기반 컴포넌트, 핵심 공통 컴포넌트, 그리고 응용 컴포넌트로 분류될 수 있다. GIS 기반 컴포넌트는 다양한 GIS 응용 시스템들이 가장 기본적으로 요구하는 기능들을 지원하는 컴포넌트로서 공간 데이터 관리, 공간 검색, 그리고 공간 연산 처리와 같은 공간 데이터베이스로서의 기능을 포함하고 있다. 핵심 공통 컴포넌트는 GIS 기반 컴포넌트 기능 이외에 GIS의 고유 기능을 수행하는데 필요한 기능들을 지원하는 컴포넌트로서 매핑, 주제도 생성 그리고 공간 분석 등의 기능을 포함하고 있다. 그리고 응용 컴포넌트는 GIS 기능이외에도 MIS(Management Information Systems) 기능을 통합하여 상수도, 토지, 도로와 같은 특정 GIS 응용 업무를 공통으로 지원하기 위한 기능을 포함하는 것으로서 응용 업무별로 다양한 컴포넌트가 구축될 수 있다. 특히 이러한 응용 컴포넌트들이 효율적으로 구축되게되면 GIS, LIS, UIS에 관한 새로운 사업을 수행하고자 할 때 기존 컴포넌트의 재활용으로 인하여 커다란 시간 및 비용의 절감 효과를 얻을 수 있다.

제2절 컴포넌트 GIS 관련 국내외 기술 개발 동향

1. 국내 기술 개발 동향

현재 국내 대기업의 SI 사업 및 공공사업부분은 주로 외국산 GIS 소프트웨어를 도입하여 지자체와 공공정보 시스템 구축사업에 활용하는 경우가 일반적인 상황이다. ESRI사의 Arc/Info, SDE와 MapObject, GEsmaWorld사의 SmallWorld, MapInfo사의 MapInfo와 SpatialWare, Integrgraph사의 MGE와 GeoMedia 등의 다양한 GIS 소프트웨어들이 도입되어 이용되고 있다. 또한 1998년 미국 ESRI사, 미국 MapInfo사, 미국 Integrgraph사에서 각각 컴포넌트 기반으로 출시된 MapObject, MapX, GeoMedia등의 소프트웨어들도 최근 들어서 자주 이용되고 있다. 이와 같이 국내 대기업의 경우는 외국산 GIS 소프트웨어를 이용한 응용 기술 개발에만 주력하고 있을 뿐 외국산 GIS 소프트웨어에 대한 의존성 극복의 어려움과 자체 기술 개발에 따른 위험성으로 인하여 GIS 엔진 소프트웨어의 기술 개발이 어려운 실정이다.

국내 중소기업의 경우는 1990년대 중반이후 자체 GIS 엔진 소프트웨어를 개발하기 시작하였는데, 현재 대표적으로 한국통신데이터의 ZEUS, 지오매니아사의 GEOMania 등의 제품들이 발표되어 사용되고 있다. 그러나 국내 GIS 엔진 개발 업체의 경우 외국의 GIS 업체와 달리 컴포넌트 기반 GIS 소프트웨어에 대한 인식과 경험이 부족하고, 개방형 컴포넌트 GIS에 대한 기술적인 검토 수준에 머물러 있는 실정으로 현재까지 컴포넌트 개념을 포함하고 있는 GIS 소프트웨어가 발표되고 있지 못하고 있다. 결론적으로 1990년대 중후반까지는 관련 학회에서의 개방형 컴포넌트 GIS에 대한 활발한 논의가 있었음에도 불구하고 업계에서의 인식 부족 및 초기 투자에 대한 부담으로 인하여 이 분야에 대한 기술 개발이 뒤쳐지게 되었다.

그러나 1998년 이후 추진된 정부 주도과 다양한 민간 기업의 공동 연구로 개방형 GIS 컴포넌트 기술 개발 사업을 수행하고 있으며, 현재 위 사업을 통하여 GIS 데이터의 원활한 정보 유통을 지원하는 GIS 기반 컴포넌트의 개발, GIS의 고유 기능을 수행하는 GIS 핵심 공통 컴포넌트의 개발 그리고 지자체 업무에 직접 적용하기 위한 GIS 업무별 응용 컴포넌트의 개발 등이 이루어지고 있다. 특히 위 사업은 OpenGIS 컨소시엄과 ISO/TC211의 협력으로 발표되는 국제 표준 사양 중에서 "OpenGIS Simple Feature Specification For OLE/COM" 구현 사양을 수용하여 추진되고 있다. 현재 개방형 GIS 컴포넌트 기술 개발 사업은 3차년도를 수행 중에 있으며 2차년도까지의 결과물인 다양한 컴포넌트들은 이미 다양한 업체들을 통하여 상업용으로 재개발되고 있는 실정이다.

2. 국외 기술 개발 동향

GIS 기술개발 사업과 관련된 국가차원 및 표준적인 국외 GIS 기술 개발 현황에 대해서는 현재 국내 개방형 GIS 컴포넌트 기술 개발 사업에서 채택하고 있는 표준 사양을 발표하고 있는 OpenGIS 컨소시엄의 배경과 현황에 대하여 주로 살펴보고자 한다.

가. OpenGIS의 개념

OpenGIS(The Open GeoData Interoperability Specification)는 서로 다른 분야의 서로 다른 환경에서 만들어져 분산 저장되어 있는 다양한 형태의 공간 정보에 대하여 다양한 사용자의 접근 및 자료 처리 기능을 제공할 수 있도록 하기 위한 개념의 GIS를 말한다. 이러한 OpenGIS는 상호운용성(Interoperability)의 개념 기반 하에서 실현이 가능하다. 상호운용성은 소프트웨어에서의 Plug and Play 개념으로 설명될 수 있다. 하드웨어 분야에서 시작된 Plug and Play 개념은 새로운 하드웨어 장비를 설치함으로써 사용자가 원하는 기능을 바로 수행할 수 있는 개념으로 사용자에게 시간과 비용에 있어서 많은 장점을 주고 있다. 소프트웨어에서는 사용자가 수행하고자 하는 기능을 위하여 전체 시스템을 교체하지 않고 Plug and Play 개념과 유사하게 새로운 컴포넌트를 추가로 설치함으로써 문제를 해결할 수 있다. 그리고 이렇게 설치되는 컴포넌트는 분산 환경에서 그리고 네트워크 환경에서 자유로이 이용될 수 있다. 따라서 복잡하고 다양한 공간 정보의 형식과 소프트웨어를 사용하는 GIS 시스템에서는 이러한 Plug and Play 개념의 컴포넌트를 이용함으로써 보다 많은 공간 정보 형식과 소프트웨어를 상호 연계하고 공간 정보를 공유할 수 있으므로 그 효용성이 매우 높다고 볼 수 있다.

또한 OpenGIS는 공간 정보의 상호운용성 문제에 대한 해결책을 제시할 뿐만 아니라 네트워크를 기반으로 하는 분산 환경 상에서 공간 정보의 객체 지향적인 개념을 정의하고 있는 사양이다. OpenGIS는 최근에 만들어진 여타 분산 객체지향 기반의 GIS 소프트웨어와 기본적인 구조에 있어서는 비슷하다. 그러나 OpenGIS는 전세계적인 또는 국가정보의 전체 하부구조 수준에서의 GIS 정보의 상호운용성을 보장하기 위하여 적용된 최초의 방법론이다.

나. OpenGIS와 SDTS

OpenGIS는 GIS 정보의 상호 공유를 기본 목적의 하나로 채택하고 있다는 면에서, 미국이 지난 1992년 공간 정보의 변환 및 교환 표준으로 채택한 SDTS(Spatial Data Transfer Standard)와 일맥상통하는 점이 있다. 실제로 SDTS는 미국 뿐만아니라 몇몇 국가가 추진하고 있는 공간 정보의 변환 및 교환을 위한 표준 모델이 되고 있어서 GIS 산업전반에

대한 파급효과가 상당히 큰 편이다. 그러면 OpenGIS와 SDTS는 어떠한 차이점을 가지고 있는가? 공간 정보의 교환 및 변환을 위한 표준으로 이미 SDTS가 채택되었는데 OpenGIS의 또 다른 표준 사양은 왜 필요한 것인가? 이러한 의문에 대답하기 위해서 본 장에서는 OpenGIS와 SDTS의 유사점 및 차이점을 비교 분석해보고자 한다.

OpenGIS와 SDTS는 결론적으로 상호 다른 형식으로 구성되어 있는 공간 정보를 공유할 수 있다는 점에서는 유사한 면이 있으나 공간 정보를 공유하기 위하여 채택되는 과정이나 개념적인 면에서는 근본적인 차이점을 보이고 있다. 그럼으로써 OpenGIS와 SDTS는 GIS의 표준으로서의 역할과 기능에 있어서도 큰 차이를 보이고 있다. 우선 OpenGIS와 SDTS는 모든 공간 정보를 하드웨어 또는 소프트웨어 환경에 독립적으로 상호 공유할 수 있게 하고자 하는 공통의 목적을 가지고 있다. 그리고 OpenGIS와 SDTS는 광범위한 공간 정보 형식들의 공유를 지원하기 위하여 수많은 공간 정보 모델들을 포괄하고 있다는 점에서 공통점을 발견할 수 있다. 그러나 OpenGIS에서는 공간 정보 모델을 수용할 뿐만 아니라 프로세스 모델을 추가적으로 그 범위에 포함시키고 있다는 점에서 SDTS보다 훨씬 광범위하고 미래지향적인 개념이라고 할 수 있다. 프로세스 모델이란 쉽게 설명하자면 공간 정보만 상호 공유하는 것이 아니라 공간 연산 작업까지 상호 공유할 수 있는 것을 의미하는 것으로, 바로 상호운용성을 개념을 말하는 것이다. 결론적으로 SDTS의 목적은 공간 정보의 상호 공유에 있는 반면에, OpenGIS의 목적은 공간 정보의 공유뿐만 아니라 공간 연산자도 공유할 수 있는 상호운용성의 실현에 있다.

공간 정보의 공유 수준과 상호운용성의 기본적인 차이점은 공간 정보의 공유는 공간 정보의 공유 자체만을 다루는데 비하여 상호운용성은 공간 정보의 공유와 연산 처리과정 두 가지를 모두 포함한다는 것이다. 그러므로 SDTS는 공간 데이터 단위로 자료를 변환하고 공유하지만 OpenGIS의 상호운용성은 메소드를 포함하고 있는 공간 객체 단위로 공유가 가능하므로 어떠한 수준의 공간 데이터도 공유할 수 있다. 또한 SDTS가 공간 정보 소스에서 목표시스템으로의 일방통행으로 정보를 전송하는 방식인데 반하여, OpenGIS의 상호운용성은 클라이언트와 서버 사이의 커뮤니케이션을 수행할 수 있는 양방향 통행 방식으로 정보를 상호 공유할 수 있다. 이와 같이 OpenGIS는 SDTS에 비해 보다 광범위하고 포괄적인 공간 데이터 모델과 공간 데이터 처리 기능을 포함하고 있음을 알 수 있다.

그러므로 다양한 공간 정보 저장 형식과 GIS 소프트웨어들 사이에서 공간 데이터와 처리 모듈을 동시에 자유롭게 상호 공유하기 위해서는 SDTS 개념 보다는 OpenGIS의 개념이 더욱 적합한 것을 알 수 있다. 국내의 경우는 SDTS를 공간 정보 교환을 위한 표준 형식으로 선정하였지만 현실적용에 따른 어려움으로 인하여 그 활용이 지체되고 있으며, 현재는 OpenGIS 개념을 반영하여 공간 정보의 상호 공유 방식에 대한 연구가 활발히 진행되고 있는 추세이다.

다. OpenGIS 컨소시엄의 주요 현황

1994년 8월 25일 설립된 OpenGIS 컨소시엄(OGC)은 OpenGIS를 위한 여러 가지 추상 및 표준 구현 사양을 개발하고 추진시키고 GIS 각 분야의 의견일치를 목적으로 하여 설립된 회원제 비과세 법인이며, 회원으로 GIS의 선도적인 소프트웨어와 하드웨어 업체, 시스템 통합 업체, 공간 데이터 제공 기관, 학계 그리고 정부 기관 등을 포함하고 있다. OGC 회원이 된 이들 단체는 OpenGIS의 개발 및 GIS 정보와 GIS 정보 처리에 대한 전 세계적인 시장을 고무시킬수 있는 새로운 상호운용성 제품의 창조를 지원하도록 중용받고 있다. 그리고 이와 같이 GIS와 관련이 있는 모든 분야의 합의에 기반 하여 기술 개발을 진행 하여야지만 실질적 의미의 개방형 GIS를 위한 사양이 만들어질 수 있고, 또한 GIS 개발 부분, 데이터 구축 부분 또는 연구 부분의 모든 분야에서의 실제 사용에 있어서 이러한 사양을 수용할 수 있기 때문이다.

이러한 OGC는 미국의 국가 공간정보기반(NSDI) 구축 사업을 선도하고 있는 연방 지리 정보위원회(FGDC)와 긴밀한 관계를 유지하고 있으며 실제로 FGDC의 후원으로 웹 기반 지리정보시스템을 위한 OpenGIS 사양이 최근에 일부 발표되기도 하였다. 특히 GIS 관련 국제 표준화 기구인 ISO/TC211과 긴밀한 협력관계를 맺고 있으며, 실제로 ISO/TC211은 OGC가 발표하는 주요 추상 및 구현 사양을 기반으로 하여 국제 표준 사양을 제정하여 발표하고 있다.

라. OpenGIS의 목표

OpenGIS의 목표는 분산 환경에서 상호 이질적인 공간 정보들의 상호운용성을 충족시키기 위한 표준 사양의 개발을 추진하는 것이다. 실제로 OpenGIS는 기업들이 GIS 업무를 수행하고자 할 때 기업간 상호 다른 기업의 GIS 정보를 요구하게 되고 이를 효율적으로 처리하기 위한 공통 GIS 정보 처리가 반드시 필요하다는 인식에서 출발하게 되었다. 그러므로 OpenGIS가 가장 우선적으로 해결해야할 과제는 다음의 두 가지라고 보고 있다.

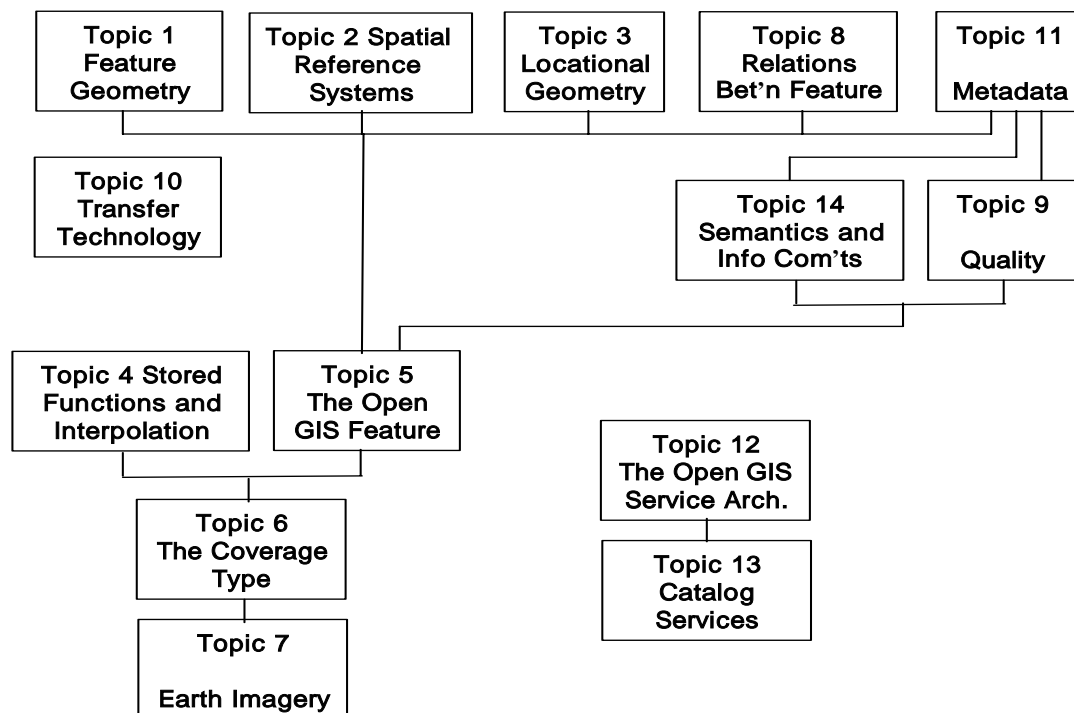
- 일부 사람에 의해 관리되었던 GIS 데이터베이스에 수 많은 사람들의 접근 지원
- 네트워크 상에서 직접 다른 기관의 GIS 데이터베이스에 접근 지원

GIS 정보를 관리하는 소프트웨어들은 모두 각자의 공간 정보 저장 방식, 관리 방식 그리고 사용자에게 사용 방식을 제공하고 있다. 그리고 각 GIS 소프트웨어별로 다른 방식을 이용하여 공간 정보를 사용자에게 제공하고 있다. 그러므로 OpenGIS에서는 위의 두 가지 문제 -GIS 데이터베이스에 자유로운 사용자들의 접근과 네트워크 상에서의 다른 기관 데이터베이스의 접근-를 해결하기 위하여 새로운 OpenGIS 인터페이스를 가지는 소프트웨

어를 제시하고 있다. 이러한 OpenGIS 인터페이스를 가지는 소프트웨어는 상호운용성을 보장하며 *Binary and Binary-Like* 인터페이스 레퍼토리 기반의 브라운관제과 도자식 인터페이스 특징을 가지고 있다. 실제로 OpenGIS는 컴포넌트 기반의 GIS 소프트웨어 구현 사양을 발표함으로써 위의 두 문제를 해결하도록 구체적인 방법을 제시하고 있다.

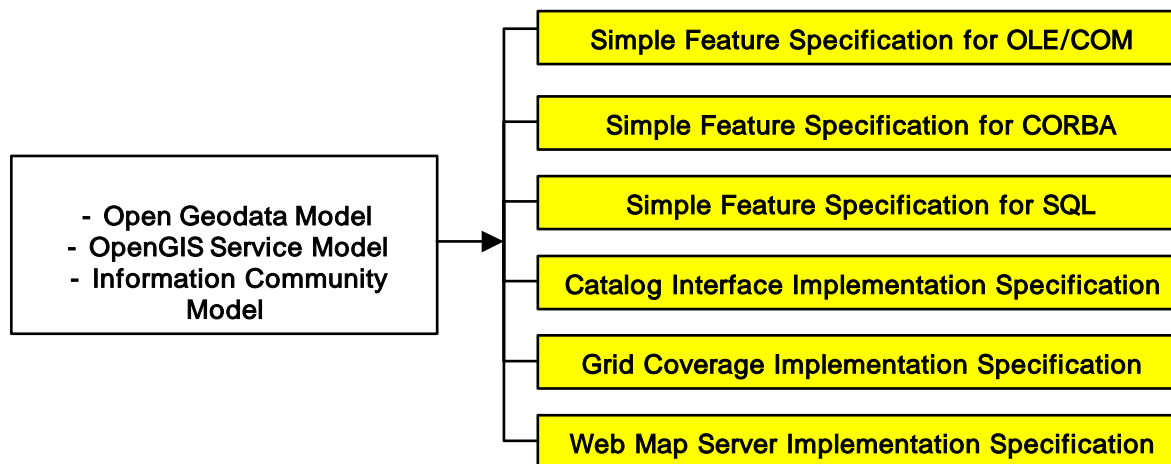
마. OpenGIS의 추상 및 구현 사양

OGC 기술위원회는 OpenGIS를 위한 추상 사양(Abstract Specification)과 다양한 구현 사양(Implementation Specification)을 개발하고 있다. 추상 사양은 상호운용 가능한 GIS 소프트웨어의 구현을 가능하게 하는 상세 지침으로, '핵심 모델(Essential Model)'이라 불리는 GIS 정보의 접근 및 처리를 위한 전체 모델을 정의하고 있다. 핵심 모델은 '개방형 지리 데이터 모델(Open Geodata Model)'과 개방형 GIS 서비스 모델(OpenGIS Services Model)'과 '지리공간 정보 커뮤니티(Geospatial Information Community) 개념을 포함하고 있다. 그리고 실제 이 핵심 모델은 다음 <그림 2-3>과 같이 13개의 토픽으로 나누어져 있다.



<그림 2-3> OGC의 추상 사양(Abstract Specification)

이러한 핵심 모델의 추상 사양들은 산업계에서 분산 처리 플랫폼을 지원하는 기술로 받아들여지는 OLE/COM 또는 CORBA 기술을 이용하여 구현되어야 실제로 분산 환경에서 공간 정보의 상호운용성을 위해 사용될 수 있다. 그러므로 OGC 기술 위원회는 이러한 추상 사양 이외에 다양한 구현 사양(Implementation Specification)을 병행하여 발표하고 있다. <그림 2-4>는 추상 사양의 기본 개념으로부터 개발되어진 OpenGIS를 위한 구현 사양들을 보여주고 있다.



<그림 2-4> OGC의 구현 사양(Implementation Specification)

OLE/COM 구현 사양은 마이크로소프트(Microsoft)사에서 소프트웨어의 분산 처리를 위해 개발된 OLE와 COM 기술을 이용하여 GIS 관련 컴포넌트를 개발하기 위한 세부 지침을 설명하고 있다. CORBA 구현 사양은 모든 개발 환경에 독립적으로 분산 처리를 지원하는 기술인 CORBA를 이용하여 GIS 공간 정보 및 메소드를 상호 공유하기 위한 방법을 상세히 설명하고 있다. 그 밖에도 OGC에서는 SQL(Structured Query Language) 처리를 위한 세부 구현 사양, 공간 정보의 유통에서 공간 정보의 검색을 위해서 필수적인 Catalog 구현 사양, GIS 이미지 처리와 관련된 Grid Coverage 구현 사양 그리고 끝으로 웹 상에서 GIS 서비스를 수행하도록 지원하기 위한 Web Map Server 구현 사양을 발표하고 있다. 이러한 구현 사양들은 시간이 지남에 따라 계속해서 발표되고 있으며, 사용자들의 요구를 수용하여 Revision되어 발표되고 있다. 국내에서는 현재 OLE/COM, CORBA, Catalog 그리고 Web Map Server 구현 사양이 일부 구현되어 있다. 본 연구에서는 교통DB와의 개방형 GIS 컴포넌트와의 연계 방안을 위해서 필요한 OLE/COM 구현 사양에 대하여 자세히 살펴볼 것이다.

제3절 개방형 GIS 컴포넌트 기술 개발의 필요성

1. 개방형 GIS 컴포넌트 기술 개발의 중요성

가. 사회·문화적 중요성

도로, 철도, 지자체 상하수도, 가스관, 유무선 통신망 또는 통신 케이블 등의 다양한 GIS 정보 구축은 시간과 비용이 많이 소모되는 일종의 SOC 작업으로 기존의 정보자원을 활용함으로써 중복 투자를 방지하고 국민이 체감할 수 있는 정보의 질을 향상시키는 것이 중요하다. 그리고 이러한 국가 사회 기반인 SOC 자원을 지자체에서 효율적으로 관리할 수 있는 GIS 도구를 개발함으로써 지자체 GIS 사업을 추진하고자 할 때 중복 투자 방지와 사업 기간 단축의 이점을 얻을 수 있다.

그리고 구축비용이 많이 소모되는 GIS 정보를 공유하고 GIS 정보와 소프트웨어 개발 사양에 대한 국제 표준을 수용함으로써 GIS 정보 유통 체계의 표준을 확립하고 지자체의 실무 적용을 통한 대 국민 삶의 질을 향상하고 국제적 정보화의 기반 구축에 기여할 수 있다.

나. 경제적 중요성

첫째, 고가의 외국산 GIS 엔진을 응용 시스템별로 대체할 수 있는 GIS 소프트웨어의 개발이 필요하다. 현재 지자체별로 사용하고 있는 응용시스템들은 대부분 외국산 GIS 엔진으로 운영되고 있으며 초기 도입가 이외에 수년간 수 백억원대의 유지보수 비용이 소요되고 있다. 특히 외국의 GIS 소프트웨어들은 범용의 목적으로 만들어진 소프트웨어이므로 국내에서 활용이 가능한 응용 시스템으로 변경하기 위해서는 외국산 소프트웨어에 대한 고급 엔지니어가 필요하고 추가의 비용과 노력이 많이 요구되어 국내 현실에 어울리지 않는 면을 가지고 있다. 실제로 외국산 GIS 소프트웨어 도입의 경우 국내 기술력 축적 및 기술 지원에 예로점을 많이 가지고 있다. 그러므로 장단기적으로 외국산 GIS 엔진을 국산화시키는 것이 반드시 필요한 실정이다.

따라서 개방형 GIS 컴포넌트 기술 개발에서는 이를 위하여 기존의 외국산 GIS 엔진을 지원하면서 국산 GIS 엔진과 상호 호환성을 가지는 컴포넌트들을 개발하여 기존에 외국산 소프트웨어를 사용하는 응용 시스템과 국산 소프트웨어를 사용하여 새로이 구축되는 응용 시스템과의 호환성을 지원해야 한다.

둘째, GIS 관련 정보 기술의 급격한 발전으로 공간 정보의 양이 급증하고 있으며 높은 질의 공간 정보 서비스에 대한 요구가 증대하고 있다. 특히 기존에 소수의 전문가들만을 대상으로 하는 범용의 GIS는 과거에 비해 GIS 소프트웨어 시장에서 점차 그 중요성이 떨어지고 있으며, 일반 사용자들을 주 대상으로 하는 인터넷 GIS 같은 것이 그 중요성을 더하고 있다. 그리고 국가지리정보체계 구축 사업으로 국내 공간 정보의 양이 계속 증대하면서 사용자들은 복잡한 기능을 요구하는 소프트웨어보다는 손쉽게 공간 정보 서비스를 받을 수 있는 소프트웨어를 더욱 필요로 하게 되었다. 결론적으로 기 구축된 공간 정보의 상호 공유를 통한 개인-기관간의 공간 정보 유통 또는 기관-기관간의 공간 정보 유통 시장이 형성되고 있다. 이러한 정보 유통 사업에서는 다양한 공간 정보의 형식과 다양한 GIS 소프트웨어로 구축된 공간 정보를 상호 공유하기 위하여 일관된 방식으로 공간 정보를 접근하고 서비스 받기 위한 데이터 제공자 컴포넌트와 같은 소프트웨어의 개발이 필수적이다.

따라서 개방형 GIS 컴포넌트 기술 개발에서는 다양한 GIS 소프트웨어 및 공간 정보의 저장 형식에 독립적으로 동작할 수 있는 컴포넌트 기술을 개발함으로써 공간 정보 유통 사업에서 소모되는 시간 및 비용을 최소화 할 수 있는 기반을 마련하도록 해야 한다.

2. 개방형 GIS 컴포넌트 기술 개발의 필요성 및 추진 배경

나. 소프트웨어 위기

컴퓨터 산업의 급격한 발전으로 인하여 모든 분야에 있어서 정보화가 가속화되고 있으며, 다양한 분야의 정보화 과정을 지원하기 위해서 소프트웨어 기술의 발전이 필수불가결하게 되었다. 그러나 이러한 정보화에 따른 소프트웨어 기술의 발전은 소프트웨어 위기라는 결과를 초래하게 되었다. <그림 2-5>



<그림 2-5> 급격한 정보화에 따른 소프트웨어의 위기

실제로 소프트웨어 기술에 대한 수요는 급증하였으나 공급이 이를 따라가지 못하여 상대적으로 소프트웨어의 기술이 낙후하게 되었으며, 기존에 만들어진 소프트웨어에 대한

유지보수에도 어려움이 있어 소프트웨어의 품질 향상에 있어서도 문제가 발생하고 있다. 현재 가트너 그룹(Gartner Group)에 따르면 100 배에 달하는 소프트웨어의 수요에 비해 소프트웨어의 기술 발전은 1.8배에 머무르고 있는 실정이다.

이러한 소프트웨어 구축에 소모되는 막대한 비용과 시간에 대한 해결책으로 IT 기술자들은 전체 시스템 구축 비용의 절감과 더불어 소프트웨어의 재활용을 강조하고 있다. 다시 말하자면, 우선 불필요한 부분을 포함하는 고가의 상업용 시스템 전체를 구입하지 않고 필요한 부분만을 구입함으로써 소프트웨어 구입 비용을 절감시키고, 둘째 같은 기능을 가지는 소프트웨어는 중복으로 구입하지 않고 재활용하여 사용하고, 셋째 각 업무별로 수행되는 소프트웨어를 상호 공유하여 사용하고, 끝으로 응용 시스템에서 변경이 요구될 때 변경되는 부분만을 대치하고 나머지는 그대로 사용하도록 하는 것이다. 그러나 위와 같이 소프트웨어의 재활용성은 기존에 클라이언트-서버 또는 Stand-alone 형식으로 구성되어 있는 응용 시스템에는 적용할 수가 없다. 즉, 전체 시스템을 재활용 할 수 있는 수준으로 분리하여 시스템을 개발할 수 있는 컴포넌트 기반의 소프트웨어 개발 방법론의 패러다임 전환이 요구된다.

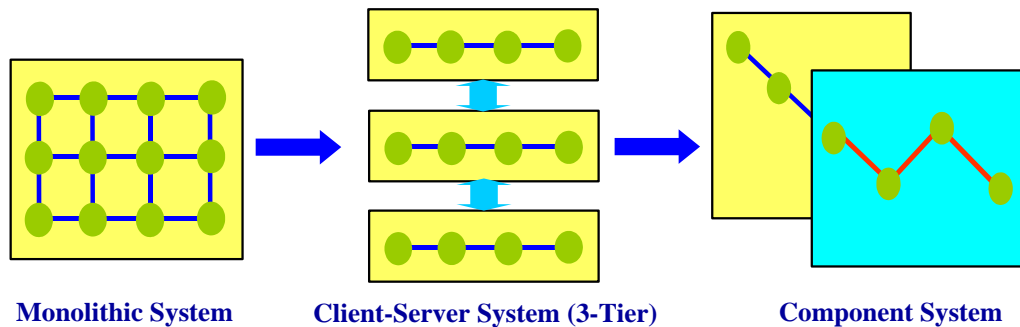
나. 소프트웨어 개발의 패러다임 변화

1980년대 초의 소프트웨어는 하나의(Monolithic) 시스템이 전체 응용 업무를 수행하도록 구성되어 있었다. 이러한 소프트웨어 구성은 새로운 업무 확장시 또는 업무 변경시에 전체 시스템의 구동을 멈추고 전체적인 소프트웨어 변경 작업을 수행하여야 했다. 초기에는 이와 같이 높은 비용을 지불하는 반면에 효율은 매우 떨어지는 고비용/저효율의 소프트웨어 구성방식을 주로 사용하였다.

이를 보완하기 위하여 1980년대 이후로 1990년대까지는 클라이언트-서버 환경의 시스템 구성이 주류를 이루게 되었다. 클라이언트는 사용자의 간단한 요구를 수용하고 중요한 업무나 데이터들은 서버에 저장하는 방식으로 구성된 클라이언트-서버 방식의 소프트웨어는 응용 업무에 따라서 2-tier, 3-tier 또는 n-tier의 구성을 가지게 되었다. 이 방식은 구축 및 유지 보수 비용이 많이 드는 단점이 있으나 시스템의 성능면에서는 좋은 효율을 보여주는 고비용/고효율의 특징을 가지고 있었다.

끝으로 최근 들어서는 분산환경에서 다양한 컴포넌트 소프트웨어를 구성하고 이들 컴포넌트들을 단순히 조립함으로써 응용 시스템을 구축하는 방법을 지향하고 있다. 이러한 컴포넌트 기반의 소프트웨어 구성 방식은 새로운 업무 확장시에 업무에 알맞은 컴포넌트만 추가함으로써 기존 시스템을 간섭하지 않고 손쉽게 시스템 변경이 가능한 장점을 가지고 있으며, 또한 기존에 구축된 다양한 컴포넌트를 사용하므로 소프트웨어의 재 사용성 측면에서도 뛰어난 장점을 보여준다. 결론적으로 이러한 컴포넌트 방식은 적은 비용으로 높은

효율을 보여주는 저비용/고효율의 소프트웨어 개발 방식으로 볼 수 있다. 다시 말하면 이와 같은 컴포넌트 기반의 소프트웨어 개발 방식이야말로 소프트웨어 위기의 해결책으로 이용될 수 있음을 알 수 있다. <그림 2-6>은 소프트웨어를 구성하기 위한 패러다임의 변화를 보여준다.



<그림 2-6> 소프트웨어 개발의 패러다임 변화

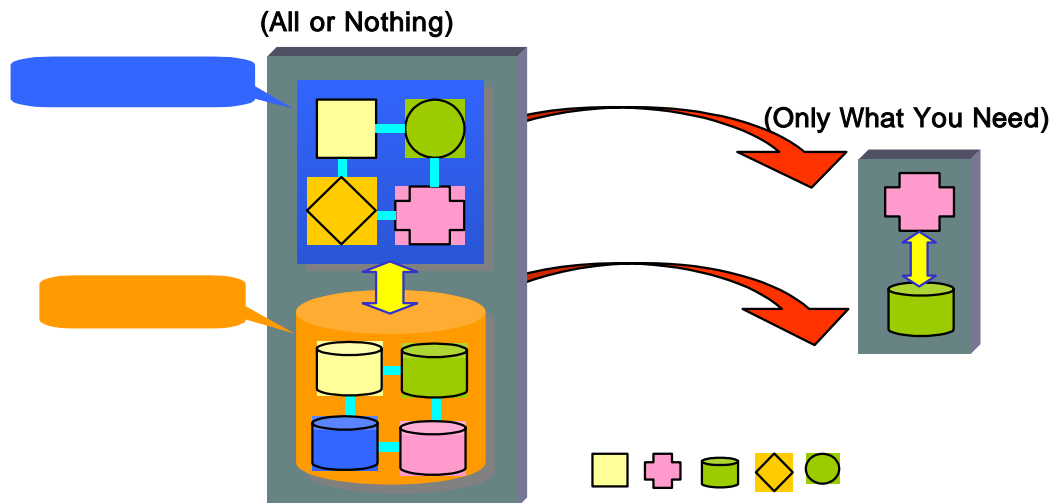
다. 컴포넌트의 필요성

컴포넌트 기반의 소프트웨어 개발 방식은 기존의 소프트웨어 개발 방식에 비해 다음과 같은 장점을 가지고 있다.

- 업무 환경이나 기술의 변화에 대한 적응력이 우수하다.
 - 새로운 업무 추가 => 새로운 업무 컴포넌트의 추가 장착
 - 업무의 변화 => 해당 업무 컴포넌트의 수정
- 저비용 / 고효율 시스템 구성
 - 전체 시스템을 구축하지 않고 필요한 컴포넌트 위주의 시스템 구성
 - 유지 및 보수 비용의 절감
- 전세계 소프트웨어 시장의 컴포넌트화
 - 가트너 그룹에 따르면 2001년까지 전세계 응용 소프트웨어 구축의 60% 이상이 컴포넌트화 예상

<그림 2-7>은 기존의 클라이언트-서버 시스템을 컴포넌트 기반 소프트웨어로 재구성함으로써 얻어지는 장점을 보여주고 있다. 기존의 응용 시스템은 전체 시스템을 구입(All or Nothing)하지 않으면 원하는 업무를 수행할 수가 없었다. 그러나 응용 시스템이 컴포넌트별로 구축되면 사용자는 원하는 업무를 수행하는 컴포넌트만을 구입(Only What You

Need)하여 응용 시스템을 구축할 수 있는 장점을 가지고 있다.

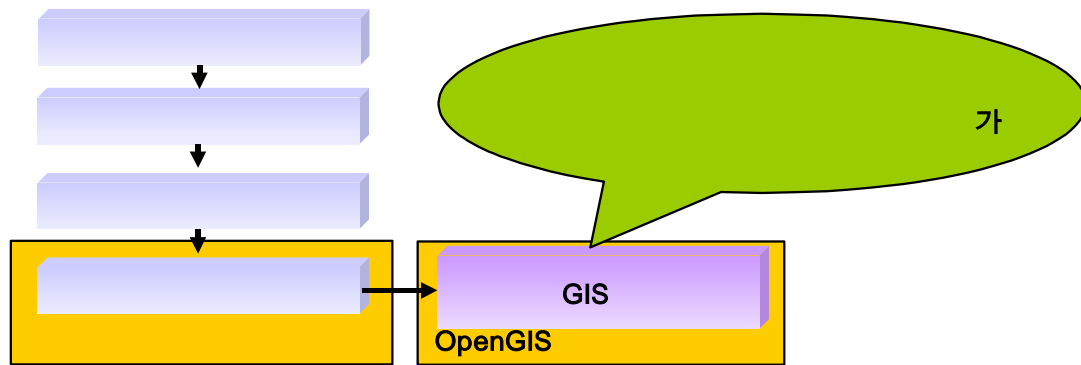


<그림 2-7> 기존의 응용 시스템에서 컴포넌트 시스템으로의 재구성시의 장점

라. 개방형 GIS 컴포넌트

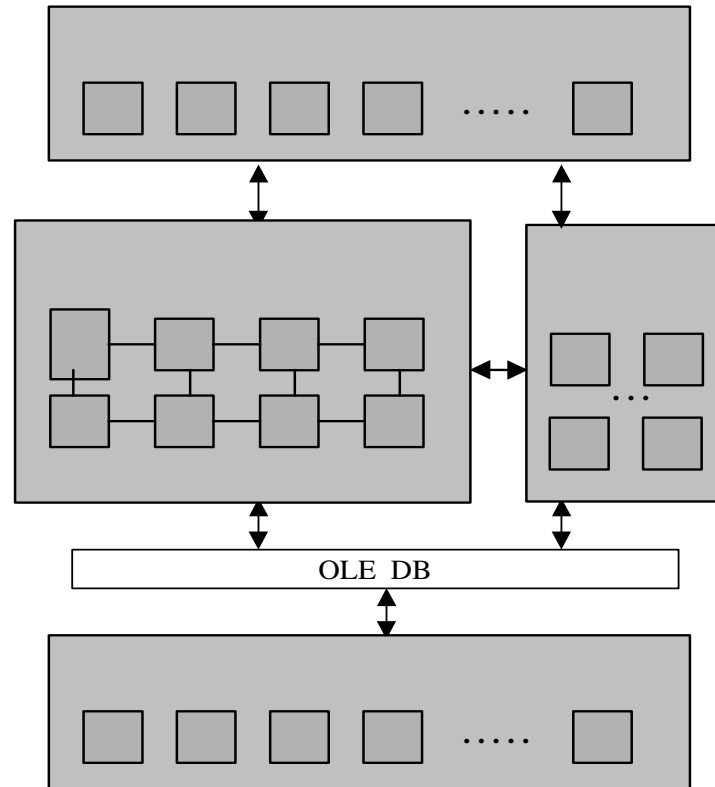
컴포넌트는 인터페이스를 가지고 있으며 사용자는 이 인터페이스를 정확하게 파악하고 있어야만 컴포넌트를 사용할 수가 있다. 그러면 이러한 컴포넌트들의 인터페이스들은 누가 정하는가? 해당 컴포넌트를 개발하는 기관에서 인터페이스를 정의하고 그 인터페이스에 관한 문서를 제공함으로써 다른 업체 또는 기관에서 그 컴포넌트를 이용할 수 있다. 여기서 만약 특정 기능을 수행하는 컴포넌트의 인터페이스를 표준화시킨다면 서로 다른 업체에서 같은 기능을 수행하고 표준 인터페이스를 가지는 컴포넌트들을 구현한다면 사용자들은 같은 기능을 수행하는 컴포넌트들 중에서 선택적으로 사용할 수 있다. 이와 같이 표준 인터페이스를 가지고 동일한 기능을 수행할 수 있는 컴포넌트들을 개방형 컴포넌트라고 한다. 개방형 컴포넌트는 기존 컴포넌트의 특징인 분산환경 지원, 상호운용성 지원, 재사용성 등의 특징 이외에도 표준화된 인터페이스를 지원하고 동일한 서비스를 지원하는 다른 컴포넌트로 교체 가능한 특징들을 가지고 있다. 이러한 개방형 컴포넌트의 개념을 GIS에 적용함으로써 개방형 GIS 컴포넌트 시스템이 어떻게 구성되는지에 세부 내용은 다음 절에서 살펴보도록 한다. <그림 2-8>은 정보화의 과정에서 소프트웨어 위기가 발생하고 이를 해결하기 위한 컴포넌트 기반 소프트웨어 기술 개발 과정이 나타나고, 컴포넌트

의 인터페이스를 표준화함으로써 개방형 컴포넌트 기술이 나타나고, 끝으로 이를 GIS에 적용함으로써 개방형 GIS 컴포넌트가 나타나는 과정을 보여주고 있다.



<그림 2-8> 개방형 GIS 컴포넌트의 출현 과정

제4절 개방형 GIS 컴포넌트의 구성



<그림 2-9> 개방형 GIS 컴포넌트의 전체적인 시스템 구성

개방형 GIS 컴포넌트의 개략적인 시스템 구성은 <그림 2-9>와 같다. 우선 데이터 제공자 컴포넌트는 다양한 공간 데이터베이스 또는 파일 시스템에 저장되어 있는 공간 정보를 표준화된 인터페이스를 통하여 접근할 수 있는 방법을 제공한다. 특히, 공간 연산 작업 그리고 SQL 처리 작업등의 GIS 엔진이 수행하는 역할들을 처리해주는 컴포넌트이다. 현재 SDE, Shape, ZEUS, SmallWorld, MapInfo 그리고 SQL Server 데이터베이스에 대한 데이터 제공자 컴포넌트가 지원되고 있다. 데이터 제공자 컴포넌트 OGC에서 제안하는 OLE/COM을 위한 표준 구현 사양을 구현하고 있다.

핵심 공통 컴포넌트는 GIS의 기본 기능 및 응용 컴포넌트에서 공통적으로 필요로 하는 기능들을 제공한다. 공간 데이터의 모델링, 공간 좌표계 변환 처리 그리고 공간 정보의 화면 도시 기능들을 수행하는 핵심 공통 컴포넌트들 역시 일부 컴포넌트를 제외하고 OLE/COM을 위한 표준 구현 사양을 구현하고 있다.

응용 컴포넌트는 상하수도, 도로 관리, 토지 관리, 도시 계획 등과 같이 실제로 지자체 응용 업무에서 이용될 수 있는 기능들을 컴포넌트로 구현하여 제공하고 있다. 그러나 응용 컴포넌트는 국내외 상황이 다르고 업무별로 구현해야 할 기능들이 다르고 같은 업무에 대해서도 지자체별로 상황이 다르므로 OGC에서 국제 표준 사양을 제시하지 않고 있다. 그러므로 개방형 GIS 컴포넌트 기술 개발에서는 국내외 상황만을 고려하고, 업무별로 그리고 지자체별로 공통 업무만을 추출하여 응용 컴포넌트로 구축하고 있다. 물론 이러한 응용 컴포넌트들은 지자체 또는 업무의 상황에 따라 조금씩 변화를 가할 수 있다.

정보 유통 컴포넌트는 각 기관 또는 업체별로 저장되어 있는 공간 데이터의 공유 및 상호 연계를 지원하기 위한 메타 데이터의 검색 및 관리 방법 그리고 원격지 공간 데이터의 접근 방법을 주요 기능으로 하는 컴포넌트이다. 특히 정보 유통 컴포넌트는 공간 정보를 효율적으로 유통하기 위해서 발표된 OGC의 Catalog Service를 위한 표준 구현 사양을 따르고 있다. 공간 정보의 유통 작업은 실제로 사용자가 원하는 공간 정보의 위치를 찾기 위한 메타 데이터 검색을 위해 정보 유통 컴포넌트를 이용하고, 검색된 공간 정보를 접근하여 획득하기 위해서는 앞에서 설명한 데이터 제공자 컴포넌트를 이용하고 있다.

본 연구에서는 교통DB와 GDF 형식의 공간 정보를 개방형 GIS와 상호 공유하기 위한 방법론을 습득하기 위해서, 특히 데이터 제공자 컴포넌트와 핵심 공통 컴포넌트의 구성 및 기능에 대하여 3장과 4장에서 자세히 살펴볼 것이다.

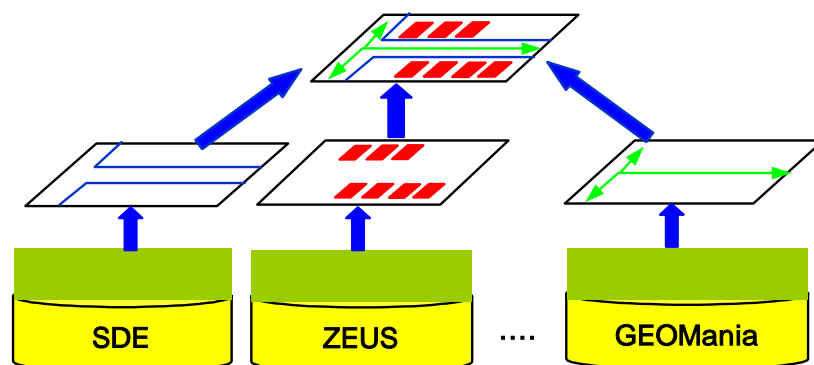
제3장 데이터 제공자 컴포넌트

- 제1절 데이터 제공자 컴포넌트의 개념
- 제2절 데이터 제공자 컴포넌트의 필요성
- 제3절 데이터 제공자 컴포넌트의 기능
- 제4절 데이터 제공자 컴포넌트의 장점
- 제5절 데이터 제공자 컴포넌트의 구성
- 제6절 결 론

제1절 데이터 제공자 컴포넌트의 개념

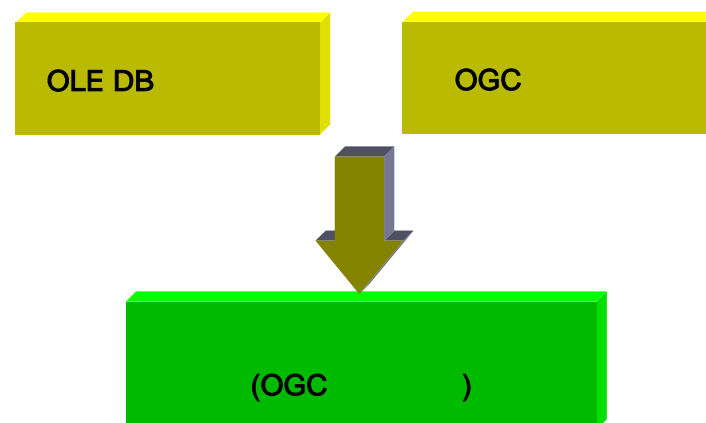
GIS 관련 기술이 급속도로 발전되고 안정화됨에 따라 기 구축된 우수한 GIS 기술들의 재사용 문제가 제기 되고 있다. 현재 이러한 재사용 문제들은 GIS 모듈의 컴포넌트화와 표준 인터페이스 제정을 기본 방향으로 하는 GIS의 표준화 연구 방향으로 해결책을 모색하고 있다. 현재 GIS 표준화 연구는 초기 단계로서 GIS 하부구조의 표준화, 다시 말하면 범 세계적으로 통용될 수 있는 GIS 정보의 표준 하부구조의 구축에 역점을 두고 있다. 이러한 표준화 작업은 앞에서 설명하였듯이 국제적인 GIS 선도 업체 위주로 구성된 OGC와 ISO/TC211을 주축으로 하여 국제적인 GIS의 표준 모델을 제시하고 있다. 그러므로 본 장에서는 OGC 데이터 어브스트라크트 표준화 워크그룹이 OGC 데이터 교환용 GDF/GDM Standard Feature 구현 사양 1.1을 주축으로 하여 응용 시스템에 표준화된 방식으로 GIS 데이터를 서비스할 수 있는 데이터 제공자(Data Provider) 컴포넌트의 기본 개념에 대하여 살펴보고자 한다. 실제로 이러한 데이터 제공자 컴포넌트의 구현은 분산 컴퓨팅 환경에서 GIS 정보의 공유 및 상호운용성을 지원하기 위해서 가장 먼저 선결되어야 하는 사항이다.

데이터 제공자 컴포넌트의 핵심 개념은 다양한 데이터베이스 시스템 또는 파일 시스템에 저장되어 있는 GIS 데이터를 표준화된 인터페이스를 통하여 접근이 가능하게 해주는 것이다. 예를 들면 <그림 3-1>과 같이 ESRI의 SDE 공간 엔진, GEOMania의 GEOMania GIS 소프트웨어, 한국통신데이터의 ZEUS 공간 데이터베이스 또는 Shape 파일과 같이 다양한 GIS 소프트웨어와 다양한 공간 정보 저장 형식의 GIS 정보를 모두 동일한 접근 방법으로 한 클라이언트 시스템이 접근 가능하게 해주는 것이다.



<그림 3-1> 데이터 제공자 컴포넌트의 개념도

OGC에서는 실제로 “OpenGIS Simple Feature Specification For OLE/COM Revision 1.1” 구현 사양을 통하여 이러한 데이터 제공자 컴포넌트의 구체적인 구현 방법을 제시하고 있다. 표준 데이터 제공자 컴포넌트를 구축하기 위해서는 우선 마이크로소프트사에서 지원하고 있는 OLE DB 컴포넌트의 구현 사양에서 제시하는 표준 인터페이스를 충족시켜야 한다. OLE DB 컴포넌트 구현 사양은 마이크로소프트사에서 모든 데이터베이스, 모든 파일 포맷 그리고 심지어는 E-mail의 데이터까지 동일한 인터페이스를 통하여 접근할 수 있도록 하기 위하여 전략적으로 발표한 것으로 기존의 ODBC(Open Database Connectivity) 보다 훨씬 발전된 개념을 포함하고 있다. 그러나 OLE DB 컴포넌트는 마이크로소프트사에서 GIS 데이터가 아닌 일반적인 데이터를 처리하기 위해서 발표한 사항이므로, GIS 적인 요소를 빠트리고 있다. 그러므로 OGC에서는 기존의 OLE DB 구현 사양에 GIS 데이터를 처리하기 위한 몇 가지 기능들을 추가로 구현하기를 요구하고 있다. <그림 3-2> 이러한 GIS용 OLE DB 데이터 제공자 컴포넌트의 자세한 기능 및 구성에 대해서는 3절과 5절에서 상세히 살펴볼 것이다.



<그림 3-2> OGC에서 제시하는 데이터 제공자 컴포넌트

제2절 데이터 제공자 컴포넌트의 필요성

90년대 들어 정보기술 산업의 급격한 발전과 더불어 GIS 관련 기술도 급격하게 발전되었다. 특히, 인터넷과 같은 정보 통신 산업의 기반이 구축됨에 따라 일부 전문가들에게 국한되고 대용량의 서버 시스템 기반에서 구동되던 GIS에 대한 일반 사용자들의 수요가 급격히 증가하게 되었다. 이러한 정황은 GIS라는 특수한 정보 기술이 다양한 시장의 요구를 충족 시켜줄 수 있는 대중적인 정보 기술로 변모하고 있음을 의미하는 것으로 이러한 공간 데이터 리마스터 레즈지리 데이터베이스 기술이 있어서 GIS의 기반 시스템에서 큰 변화가 일어나고 있다. GIS의 기술적인 움직임은 주로 네트워크를 기반으로 하는 분산 컴퓨팅 환경에서 GIS 응용시스템을 구축하기 위한 방향에서 나타났다. 특히 네트워크상의 상이한 이 기종 시스템간에 저장되어 있는 공간 정보를 상호 공유하고 분산 처리할 수 있는 시스템을 개발하고자 하는 노력은 기존 GIS의 기술을 혁신적으로 발전 시켰다. 이와 같이 GIS 기술이 급속도로 발전하게됨에 따라 다음과 같은 현상들이 나타나게 되었다.

- GIS 환경의 변화

- Shape, DXF, DGN, MGE, GDF 등과 같은 다양한 형식의 공간 정보 저장 포맷이 등장하게 되었다.
- SDE, ZEUS, GEOMania, SmallWorld, MapInfo, Intergraph 등과 같은 다양한 GIS 소프트웨어들이 등장하게 되었다.

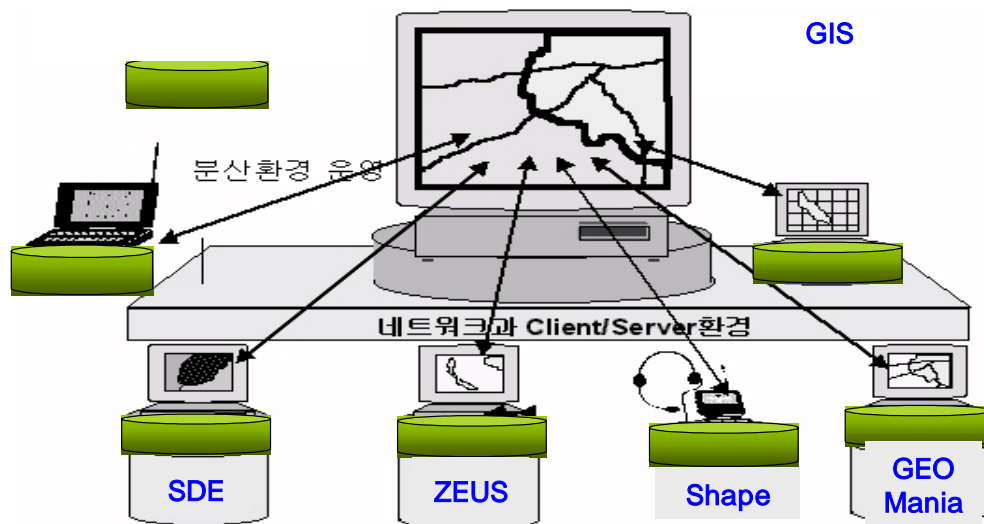
- 사용자들의 공간 정보의 상호 공유에 대한 요구 급증

- 다양한 GIS 소프트웨어간 공간 정보에 대한 상호 공유가 요구되게 되었다.
- 공간 정보의 상호 공유 및 연계를 통한 GIS 업무에 대한 요구가 급증하였다.
- 공간 정보의 상호 유통을 통한 공간 정보의 중복 구축 방지가 가능해졌다.

- 분산 환경에서 일반 사용자가 아닌 기관별 공간 정보의 유통이 요구됨.

이와 같이 사용자들의 공간 정보에 대한 상호 공유에 대한 요구와 분산 환경에서 기관별 공간 정보 유통에 대한 요구를 충족시켜 주기 위해서는 다양한 GIS 소프트웨어 그리고 다양한 공간 정보 저장 포맷을 손쉽게 그리고 동일한 방식으로 이용할 수 있는 방법이 절대적으로 필요하게 된다. 이러한 요구를 수행하기 위해서 SDTS와 같은 교환 포맷이 발표되기도 하였으나, SDTS는 한 방향으로의 일방적인 데이터의 변환을 지원할 뿐 분산 환경에서 공간 정보의 유통을 지원할 수 없다. 그리하여 OGC에서는 SDTS의 단점을 보완하고 사용자들의 공간 정보 유통을 충족시킬 수 있는 데이터 제공자 컴포넌트 구현 사양을 발표하기에 이르렀는데, 현재까지 이러한 데이터 제공자 컴포넌트를 이용하지 않고는 표준화된 방식으로 On-line상에서 공간 정보를 상호 공유시키고 유통시킬 수 있는 대

안이 없는 실정이다. 끝으로 <그림 3-3>은 데이터 제공자 컴포넌트들을 이용하여 구성될 수 있는 공간 정보 유통의 개념을 보여준다.



<그림 3-3> 데이터 제공자 컴포넌트를 이용한 공간 정보 유통 개념

제3절 데이터 제공자 컴포넌트의 기능

OGC에서 제시하는 데이터 제공자 컴포넌트를 구현하기 위해서는 순수 OLE DB 컴포넌트가 제공하는 기능 이외에 GIS 관련 메타 정보 지원, 공간 데이터 및 공간 좌표계 (Spatial Reference System) 정보 접근 방법 지원 그리고 공간 여과기(Spatial Filter) 지원 등의 기능들을 추가로 구현해야 한다. 본 절에서는 이러한 OGC용 데이터 제공자 컴포넌트가 필수적으로 제공해야 할 기능들에 대하여 자세히 살펴보려고 한다.

1. GIS 메타 정보 지원

일반적으로 GIS 응용 시스템은 사용자가 접속하고 있는 GIS 데이터베이스에 대한 메타 정보를 우선 접근하고, 획득된 이 메타 정보를 기본으로 하여 GIS 정보를 접근하거나 응용 업무를 수행하도록 구성되어 있다. 그러나 이러한 메타 정보를 접근하는 방법이 사용되는 GIS 소프트웨어별로 다르고, 게다가 메타 정보의 내용과 형식에 있어서도 GIS 소프트웨어별로 다른 방식을 취하고 있다. 그러므로 다양한 GIS 파일 포맷 또는 GIS 소프트

웨어에 대하여 동일한 방법으로 GIS 정보에 대한 접근 방법을 지원해야 하는 데이터 제공자 컴포넌트 역시 일관된 방식으로 메타 정보를 접근할 수 있는 방법을 구현하고 있어야 한다. 본 연구에서 데이터 제공자 컴포넌트가 반드시 제공해야 할 메타 정보는 다음과 같다.

- 공간 정보를 가지고 있는 전체 피처(Feature) 테이블 정보
- 피처 테이블에서 공간 정보를 저장하는 컬럼에 관한 메타 정보
- 공간 좌표계 메타 정보
- 각 피처 테이블들의 최대 맵 영역(Map Extents)에 관한 메타 정보

가. 공간 정보를 가지고 있는 전체 피처 테이블 메타 정보

공간 정보를 포함하는 전체 피처 테이블 메타 정보는 공간 엔진 또는 공간 데이터베이스 내에 저장되어 있는 모든 공간 피처 또는 레이어(Layer)에 관한 기본 정보를 제공하는 것이다. 다시 말하면 사용자가 이 메타 정보를 이용하게 되면 자신이 접속하고 있는 공간 데이터베이스에 어떠한 피처 또는 레이어 들이 존재하는지를 파악할 수 있게 된다. 이러한 피처 이름에 관한 메타 정보는 GIS 시스템에서 필수적이며 가장 기본적인 정보로 이용된다. <표 3-1>은 사용자가 피처 테이블 메타 정보를 요구했을 때 데이터 제공자 컴포넌트가 제공해야 할 메타 정보의 자세한 형식을 나타내고 있다.

<표 3-1> 피쳐 테이블 관련 메타 정보 형식 (OGC 표준 구현 사양)

Column_Name	Type_Indicator	Description
FEATURE_TABLE_ALIAS (NULLABLE)	DBTYPE_WSTR	User Friendly Feature Name may be NULL. => 실제 GIS 데이터베이스에 저장되어 있는 피쳐 테이블의 이름이 아닌 클라이언트 또는 응용 시스템 등에서 사용될 수 있는 피쳐 이름을 제공하면 된다. 예를 들면 실제 피쳐 테이블 이름이 "kn13"인 경우, 그 의미를 알 수가 없으므로 Alias 이름으로 "강남구 동경계"와 같은 것을 저장하는 경우가 있다. GIS 소프트웨어에 따라서 없는 경우도 있으므로 없는 경우는 NULL을 제공하면 된다.
TABLE_CATALOG (NULLABLE)	DBTYPE_WSTR	Catalog Name in which the table is defined. NULL if the provider does not support catalogs. => GIS 데이터베이스에서 정의하는 Catalog 이름을 제공한다. 일반적으로 GIS 데이터베이스 이름이 제공된다. GIS 소프트웨어에 따라서 없는 경우는 NULL을 제공하면 된다.
TABLE_SCHEMA (NULLABLE)	DBTYPE_WSTR	Schema name in which the Features Table is defined. NULL if the provider does not support schemas. => GIS 데이터베이스에서 정의하는 Schema 이름을 제공한다. 일반적으로 GIS 데이터베이스의 administrator 이름이 제공된다. GIS 소프트웨어에 따라서 없는 경우는 NULL을 제공하면 된다.
TABLE_NAME (NOT NULLABLE)	DBTYPE_WSTR	Feature Table Name => 실제 GIS 피쳐 테이블의 이름을 제공한다.
ID_COLUMN_NAME (NOT NULLABLE)	DBTYPE_WSTR	Preferred column name to reference rows. OGIS requires this column to have a name. => 실제 GIS 피쳐 테이블에서 각 행에 존재하는 GIS 객체들을 Unique하게 구별할 수 있는 컬럼 이름을 제공한다. 일반적으로 피쳐 테이블의 Primary Key 값을 제공하면 된다.
DG_COLUMN_NAME (NOT NULLABLE)	DBTYPE_WSTR	Default Geometry column name. OGIS requires this column to have a name. => 피쳐 테이블에서 Geometry 정보, 다시 말해서 공간 정보를 저장하는 기본 컬럼 이름을 제공한다. 일반적으로 공간 정보를 저장하는 컬럼 이름은 각 GIS 소프트웨어별로 모두 다르다.

그리고 다음 <그림 3-4>의 예제를 보면 <표 3-1>과 같은 형식의 피쳐 테이블 관련 메타 정보를 데이터 제공자 컴포넌트가 실제로 어떻게 지원하는지 자세히 파악할 수 있다.

Row Handle	FEATURE_TABLE_ALIAS	TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	ID_COLUMN_NAME	DG_COLUMN_NAME
0x1		STestDB	dbo	kn13	ID	POINTS
0x2		STestDB	dbo	kn31	ID	POINTS
0x3		STestDB	dbo	kn34	ID	POINTS
0x4		STestDB	dbo	kn64	ID	POINTS
0x5		STestDB	dbo	kn66	ID	POINTS
0x6		STestDB	dbo	sc13	ID	POINTS
0x7		STestDB	dbo	sc31	ID	POINTS
0x8		STestDB	dbo	sc34	ID	POINTS
0x9		STestDB	dbo	sc64	ID	POINTS
0xa		STestDB	dbo	sc66	ID	POINTS
0xb		STestDB	dbo	sp13	ID	POINTS
0xc		STestDB	dbo	sp31	ID	POINTS
0xd		STestDB	dbo	sp34	ID	POINTS
0xe		STestDB	dbo	sp64	ID	POINTS
0xf		STestDB	dbo	sp66	ID	POINTS

<그림 3-4> 피쳐 테이블 관련 메타 정보의 수행 예제 (OGC 표준 구현 사양)

<그림 3-4>의 수행 예제를 살펴보면 우선 사용자는 자신이 접속한 GIS 데이터 소스에 현재 15개의 피쳐들이 존재한다는 것을 알 수 있다. 또한 FEATURE_TABLE_ALIAS 정보로는 Null 값을, TABLE NAME 정보로는 각 피쳐 테이블 이름들을, ID COLUMN_NAME 정보로는 "ID" 값을, 그리고 DG_COLUMN_NAME 정보로는 "POINTS" 값을 제공하고 있다는 것을 이용함으로써, 사용자는 "kn13"이라는 피쳐 테이블이 존재하고, 이 피쳐 테이블은 Primary Key 값으로 "ID"라는 컬럼을 가지고 있으며 그리고 공간 정보는 "POINTS"라는 컬럼에 저장되어 있다는 것도 알 수 있다.

나. 피쳐 테이블에서 공간 정보를 저장하는 컬럼에 관한 메타 정보

둘째로 데이터 제공자 컴포넌트는 피쳐 테이블에서 공간 정보를 저장하는 컬럼에 관한 메타 정보를 지원해야 한다. 본 메타 정보는 사용자가 각 피쳐 테이블들로부터 공간 정보를 접근하여 원하는 작업을 수행하고자 할 때 주로 이용된다. 그러므로 본 메타 정보는

피쳐 테이블에서 공간 정보를 저장하는 컬럼 이름, 공간 정보의 타입 그리고 공간 좌표계 정보와 같이 공간 정보를 처리하기 위해서 반드시 필요한 정보들을 제공하고 있다. <표 3-2>는 데이터 제공자 컴포넌트에서 공간 정보 관련 메타 정보를 제공하기 위한 자세한 형식을 나타내고 있다.

<표 3-2> 공간 정보 컬럼 관련 메타 정보 형식 (OGC 표준 구현 사양)

Column_Name	Type_Indicator	Description
TABLE_CATALOG (NULLABLE)	DBTYPE_WSTR	Catalog Name in which the Features Table is defined. NULL if the provider does not support catalogs. => GIS 데이터베이스에서 정의하는 Catalog 이름을 제공한다. 일반적으로 GIS 데이터베이스 이름이 제공된다. GIS 소프트웨어에 따라서 없는 경우는 NULL을 제공하면 된다.
TABLE_SCHEMA (NULLABLE)	DBTYPE_WSTR	Schema name in which the Features Table is defined, NULL if the provider does not support schemas. => GIS 데이터베이스에서 정의하는 Schema 이름을 제공한다. 일반적으로 GIS 데이터베이스의 administrator 이름이 제공된다. GIS 소프트웨어에 따라서 없는 경우는 NULL을 제공하면 된다.
TABLE_NAME (NOT NULLABLE)	DBTYPE_WSTR	Feature Table Name => 실제 GIS 피쳐 테이블의 이름을 제공한다.
COLUMN_NAME (NOT NULLABLE)	DBTYPE_WSTR	Name of column containing geometry => 피쳐 테이블에서 Geometry 정보(공간 정보)를 저장하고 있는 컬럼의 이름을 제공한다.
GEOM_TYPE (NOT NULLABLE)	DBTYPE_UI4	Type of geometry column. Values taken from the OGIS_GEOMETRY Enumerated Type. (OLEDBGIS.H) => 피쳐의 Geometry 타입을 제공한다. Geometry Type은 OGC에서 제공하는 "OLEDBGIS.H" 파일에 정의되어 있다.
SPATIAL_REF_SYSTEM_ID (NULLABLE)	DBTYPE_I4	Foreign Key this is the ID of the Spatial Reference System of the geometry column. This ID can be used to find the Spatial Reference in the DBSCHEMA_OGIS_SPATIL_REF_SYSTEMS Rowset => 공간 좌표계 ID를 제공한다. 공간 좌표계 ID 값은 GIS 소프트웨어(SDE, MGE, GEOMania, GEUS 등) 별로 내부에서 사용하는 값을 제공하면 된다. 단, 이 공간 좌표계 ID 값을 이용하여 공간 좌표계 관련 메타 정보를 접근하였을 때 이 피쳐의 실제 공간 좌표계 정보를 획득할 수 있어야 한다. GIS 소프트웨어가 공간 좌표계를 지원하지 않는 경우는 NULL을 제공하면 된다.

<그림 3-5>의 수행 예제에서는 데이터 제공자 컴포넌트가 실제로 공간 정보 컬럼 관련 <그림 3-5> 공간 정보 컬럼 관련 메타 정보의 수행 예제 (OGC 표준 구현 사양) 메타 정보가 어떻게 제공하는지를 보여준다.

<그림 3-5>의 메타 정보를 통하여 사용자는 모든 피쳐 들이 공간 정보를 "POINTS" 컬럼에 저장하고 있으며, 각 피쳐 들의 공간 데이터 타입은 Polygon(6) 또는 LineString(4) 이며, 각 피쳐 들이 공간 좌표계 ID 값을 가지고 있음을 알 수 있다. 추후에 이 공간 좌표계 ID 값을 이용하여 공간 좌표계 관련 메타 정보를 접근함으로써 사용자는 각 피쳐 들의 공간 좌표계 정보를 획득할 수 있다.

RowHandle	TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	COLUMN_NAME	GEOM_TYPE	SPATIAL_REF_SYSTEM_ID
0x1	STestDB	dbo	ln13	POINTS	6	199
0x2	STestDB	dbo	ln31	POINTS	6	197
0x3	STestDB	dbo	ln34	POINTS	4	209
0x4	STestDB	dbo	ln64	POINTS	6	218
0x5	STestDB	dbo	ln66	POINTS	6	226
0x6	STestDB	dbo	sc13	POINTS	6	195
0x7	STestDB	dbo	sc30	POINTS	6	183
0x8	STestDB	dbo	sc34	POINTS	4	195
0x9	STestDB	dbo	sc64	POINTS	6	204
0xa	STestDB	dbo	sc66	POINTS	6	212
0xb	STestDB	dbo	sp13	POINTS	6	211
0xc	STestDB	dbo	sp30	POINTS	6	209
0xd	STestDB	dbo	sp34	POINTS	4	221
0xe	STestDB	dbo	sp64	POINTS	6	230
0xf	STestDB	dbo	sp66	POINTS	6	238

<그림 3-5> 공간정보 컬럼 관련 메타 정보 수행 예제(OGC 표준구현사양)

다. 공간 좌표계 메타 정보

셋째로 데이터 제공자 컴포넌트는 공간 좌표계에 관한 메타 정보를 제공해야 한다. 사용자들은 본 메타 정보를 이용함으로써 자신이 접근하는 공간 정보들이 TM 또는 UTM 등의 어떠한 공간 좌표계를 이용하고 있는지 알 수 있다. <표 3-3>은 공간 좌표계 메타 정보의 자세한 구성을 보여준다.

<표 3-3> 공간 좌표계 관련 메타 정보 형식 (OGC 표준 구현 사양)

Column_Name	Type_Indicator	Description
SPATIAL_REF_SYSTEM_ID (NOT NULLABLE)	DBTYPE_I4	ID of the Spatial Reference System. May be NULL only if SPATIAL_REF_SYSTEM_WKT is NULL. => 공간 정보 컬럼 메타 정보에서 제공하는 공간 좌표계 ID와 같은 의미의 ID 값을 제공하면 된다.
AUTHORITY_NAME (NULLABLE)	DBTYPE_WSTR	Defining Authority for this Spatial Reference System, e.g., POSC, USGS. May be NULL.
AUTHORITY_ID (NULLABLE)	DBTYPE_I4	Authority specific identifier. This is a Well-known id assigned to the spatial reference system by the authority. May be NULL.
SPATIAL_REF_SYSTEM_WKT (NOT NULLABLE)	DBTYPE_BSTR	The Well-Known Text Representation of the Spatial Reference System. May be NULL. => 공간 좌표계 정보를 WKT(Well-Known Text) 형식에 맞추어서 제공한다. WKT 형식은 OGC의 OLE/COM 구현 사양에 정의되어 있다.

<그림 3-6>은 데이터 제공자 컴포넌트가 공간 좌표계 관련 메타 정보를 제공하는 수행 예제를 보여준다.

Row Handle	SPATIAL_REF_SYSTEM_ID	AUTHORITY_NAME	AUTHORITY_ID	SPATIAL_REF_SYSTEM_WKT
Bt1	199	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt2	197	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt3	209	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt4	219	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt5	225	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt6	185	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt7	183	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt8	195	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt9	204	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt10	212	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt11	211	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt12	208	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt13	221	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt14	230	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...
Bt15	238	Authority Name	1	PROJCS('BESSEL_TM_Tokyo', GEOGCS('GCS_Tokyo', DATU...

<그림 3-6> 공간 좌표계 관련 메타 정보의 수행 예제 (OGC 표준 구현 사양)

<그림 3-6>의 수행 예제에서 보듯이 사용자는 공간 좌표계 메타 정보를 이용하여 각 피쳐 들이 어떠한 공간 좌표계를 이용하고 있는지 알 수 있다. 위의 예제에서는 모든 피쳐 들이 TM 좌표계를 이용하고 있음을 알 수 있다. 위의 예제에서 주의해서 보아야할 점은 SPATIAL_REF_SYSTEM_ID 정보는 GIS 소프트웨어에서 각각 사용되는 내부 ID로서 <그림 3-5>에서 제공되는 메타 정보의 SPATIAL_REF_SYSTEM_ID 정보와 일치해야 한다. 만약 데이터 제공자 컴포넌트 하부의 GIS 소프트웨어가 예를 들어 TM 좌표계만 지원한다면 공간 정보 컬럼 관련 메타 정보에서 공간 좌표계 ID 값을 모두 '1'로 세팅하고, 공간 좌표계 메타 정보에서 공간 좌표계 ID 값 '1'에 대한 공간 좌표계 정보(WKT)로서 "TM" 정보만 세팅해도 된다.

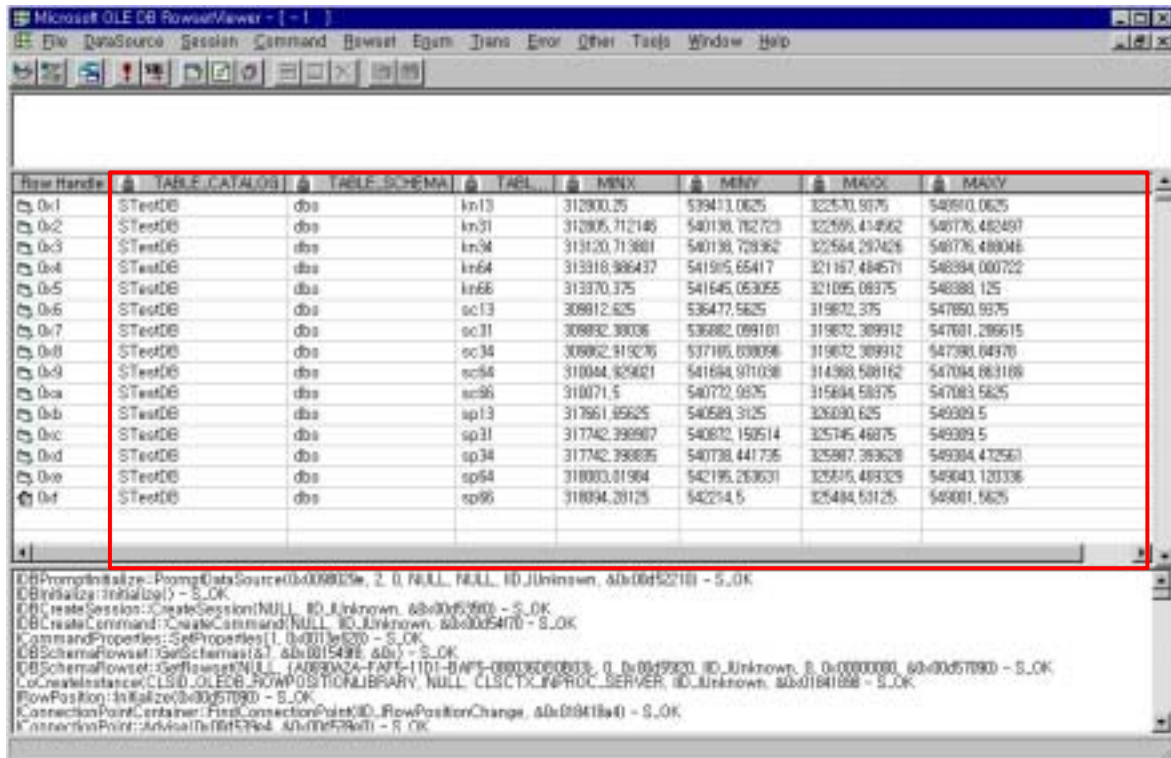
라. 피쳐 테이블들의 최대 맵 영역에 관한 메타 정보

끝으로 데이터 제공자 컴포넌트는 각 피쳐 들을 화면에 효율적으로 도시하기 위하여 각 피쳐들의 최대 맵 영역 메타 정보를 제공해야 한다. OGC의 OLE/COM 구현 사양에는 이러한 맵 영역에 관한 메타 정보를 요구하지는 않는다. 그러나 거의 모든 GIS 소프트웨어 들이 맵 영역에 관한 메타 정보를 가지고 있으며, 또한 국내 현실에 비추어 볼 때 거의 모든 GIS 관련 사업에서 피쳐 들의 맵 영역을 실제로 이용하고 있다. 그리하여 국내의 개방형 GIS 컴포넌트 개발에서는 이러한 맵 영역에 대한 메타 정보를 반드시 제공하도록 명시하고 있다. <표 3-4>는 피쳐 들의 최대 맵 영역에 대한 메타 정보 형식을 나타낸다.

<표 3-4> 피쳐 들의 최대 맵 영역 관련 메타 정보 형식 (국내 구현 사양)

Column_Name	Type_Indicator	Description
TABLE_CATALOG (NULLABLE)	DBYTPE_WSTR	Catalog name in which the Features Table is defined, NULL if the provider does not support catalogs.
TABLE_SCHEMA (NULLABLE)	DBYTPE_WSTR	Schema name in which the Features Table is defined, NULL if the provider does not support schemas.
TABLE_NAME (NOT NULLABLE)	DBYTPE_WSTR	The Feature Table Name.
MINX (NOT NULLABLE)	DBTYPE_R8	피쳐의 최소 X 좌표
MINY (NOT NULLABLE)	DBTYPE_R8	피쳐의 최소 Y 좌표
MAXX (NOT NULLABLE)	DBTYPE_R8	피쳐의 최대 X 좌표
MAXY (NOT NULLABLE)	DBTYPE_R8	피쳐의 최대 Y 좌표

<그림 3-7>은 최대 맵 영역 관련 메타 정보를 획득하는 수행 예제를 보여준다.



Row Handle	TABLE/CATALOG	TABLE-SCHEMA	TABLE	MINX	MINY	MAXX	MAXY
0x1	STTestDB	dbo	ln13	312900.25	539413.0625	32570.9375	549910.0625
0x2	STTestDB	dbo	ln31	312805.712145	540138.762723	325595.434952	549776.482497
0x3	STTestDB	dbo	ln34	313120.713681	540138.728362	325584.297428	549776.480048
0x4	STTestDB	dbo	ln64	313318.986437	541905.65417	321167.484571	548394.007722
0x5	STTestDB	dbo	ln66	313370.375	541645.053055	321095.08375	548388.125
0x6	STTestDB	dbo	ec13	308912.625	536477.5625	319872.375	547850.9375
0x7	STTestDB	dbo	ec31	308892.38036	536882.099181	319872.389912	547881.286615
0x8	STTestDB	dbo	ec34	308892.319276	537185.038056	319872.389912	547381.04970
0x9	STTestDB	dbo	ec64	310044.925021	541694.971036	314368.508162	547094.863168
0xa	STTestDB	dbo	ec66	310071.5	540772.9375	315894.58375	547083.5825
0xb	STTestDB	dbo	sp13	317861.85625	540589.3125	326330.625	549399.5
0xc	STTestDB	dbo	sp31	317742.398987	540672.150514	325745.46875	549399.5
0xd	STTestDB	dbo	sp34	317742.398985	540738.441736	325887.393628	549384.472561
0xe	STTestDB	dbo	sp64	318003.81984	542195.263631	325575.489328	549043.120336
0xf	STTestDB	dbo	sp66	318094.28125	542214.5	325484.53125	549081.5625

<그림 3-7> 피쳐 들의 최대 맵 영역 메타 정보의 수행 예제 (국내 구현 사양)

<그림 3-7>에서 보듯이 사용자는 위의 메타 정보로부터 GIS 데이터베이스 내에 저장되어있는 모든 피쳐 들의 최대 맵 영역 정보를 얻을 수 있다. 위의 예제에서는 피쳐 별로 최대 맵 영역이 모두 다른 경우이나, 실제로 어떠한 GIS 소프트웨어들은 전체 피쳐에 대하여 하나의 맵 영역 정보만을 관리하는 경우도 있다. 이러한 경우에는 메타 정보를 제공할 때 각 피쳐별로 최대 맵 영역 정보를 제공을 하되, 모두 전체 피쳐에 대한 맵 영역 정보를 제공하면 된다.

2. 공간 데이터 및 공간 좌표계 정보 접근 방법 지원

첫째, OGC를 위한 데이터 제공자 컴포넌트는 일반 OLE DB 컴포넌트가 지원할 수 없는 공간 정보를 지원할 수 있어야 한다. 이를 위하여 데이터 제공자 컴포넌트는 우선 임의의 GIS 소프트웨어에 저장되어 있는 공간 정보를 OGC 사양에서 요구하고 있는 WKB(Well-Known Binary) 형태의 기하 정보로 변환 시켜 사용자에게 제공해주어야 하

고, 사용자가 이 데이터 제공자 컴포넌트를 통하여 이러한 WKB 형태의 기하 정보를 접근할 수 있는 방법을 제공해야 한다.

둘째, OGC를 위한 데이터 제공자 컴포넌트는 공간 정보 이외에 공간 좌표계에 관한 정보도 추가로 지원할 수 있어야 한다. 이를 위하여 데이터 제공자 컴포넌트는 우선 GIS 소프트웨어에서 지원하고 있는 모든 공간 좌표계 정보를 OGC 사양에서 요구하는 WKT 형태의 공간 좌표계 정보를 변환시키는 부분과 사용자가 이 WKT 정보를 접근할 수 있는 방법을 제공해야 한다.

3. 공간 여과기 지원

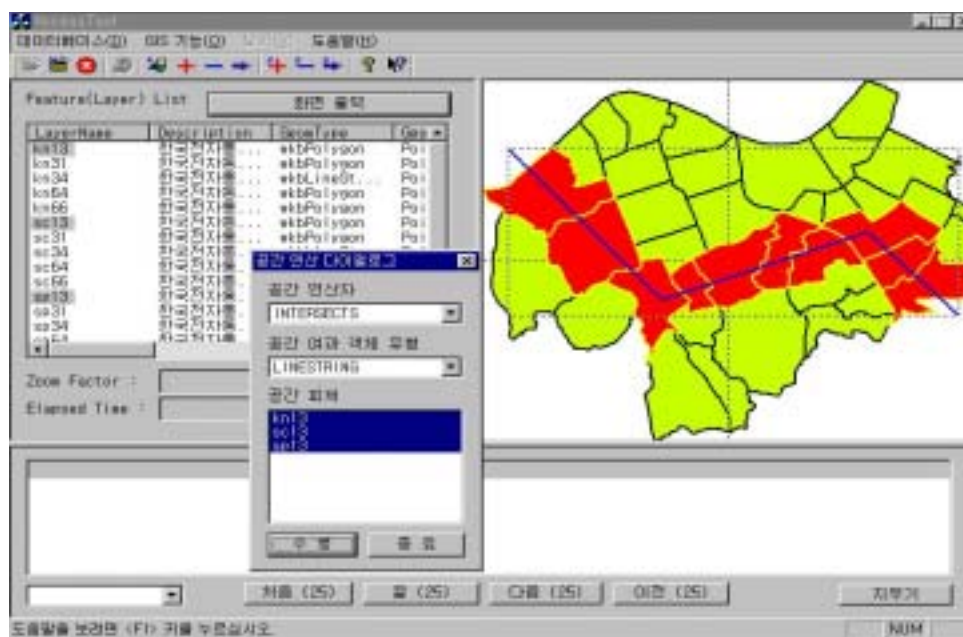
일반적으로 공간 연산(Spatial Operation) 작업은 대부분의 GIS 응용 시스템에서 핵심적인 기능들이다. 그리고 이러한 공간 연산 작업은 효율을 높이기 위하여 나름대로의 공간 색인 방법을 이용하고 GIS 데이터베이스 또는 GIS 엔진의 주요 기능으로 인식되고 있다. 그러나 이러한 공간 연산 기능은 GIS 소프트웨어별로 사용하는 방법이 전혀 다르다. 공간 연산의 입력 파라미터 그리고 공간 연산의 수행 결과물 등 모든 것이 GIS 소프트웨어별로 다른 사용법을 가지고 있다. 이와 같이 서로 상이한 방식으로 처리되는 공간 연산 처리기는 분산 환경에서 공간 정보에 대한 공간 연산 처리 작업을 자유로이 수행할 수 있는 상호운용성을 보장할 수 없게 된다.

따라서 OGC의 OLE/COM 구현 사양에서는 분산 환경에서 공간 연산 처리의 상호운용성을 보장하기 위하여 데이터 제공자 컴포넌트가 GIS 소프트웨어별로 다른 방식으로 이용되는 공간 연산 작업을 동일한 방식으로 모든 GIS 소프트웨어의 공간 연산 작업을 수행할 수 있도록 하는 공간 여과기(Spatial Filter)를 제공하도록 규정하고 있다. OGC 데이터 제공자 컴포넌트에서의 공간 연산 처리 작업은 공간 여과 객체(A), 공간 연산자(B), 피쳐 테이블의 공간 정보 컬럼 이름(C)의 세 가지 파라미터가 입력으로 주어지며, 공간 연산의 결과로는 WKB 형태의 공간 데이터가 반환되는 형식을 취하고 있다. 공간 여과 객체(Spatial Filter)로는 OGC에서 제안하는 공간 정보 타입 - Point, LineString, Polygon, Multi Point, Multi LineString, Multi Polygon - 들이 주어지고, 공간 연산자(Spatial Operator)로는 Touches, Within, Contains, Crosses, Overlaps, Disjoint, Intersects, Envelop Intersects, Index Intersects 의 9가지가 주어질 수 있다. 이와 같이 9가지 연산자를 필수적으로 모두 구현할 필요는 없으며 GIS 엔진 또는 GIS 데이터베이스가 지원하는 것만 구현해도 무방하다. 위 9가지 공간 연산자의 세부적인 구현 사양은 <표 3-5>와 같다. 그리고 피쳐 테이블의 공간 정보 컬럼 이름(Spatial Geometry Column Name)은 어떤 피쳐에 대한 공간 연산 작업을 수행할 것인가를 결정하기 위해서 필요한 정보로서, 예를 들면 "kn13.POINTS"와 같이 "피쳐 테이블 이름.공간 정보 컬럼 이름"의 형식을 가진다.

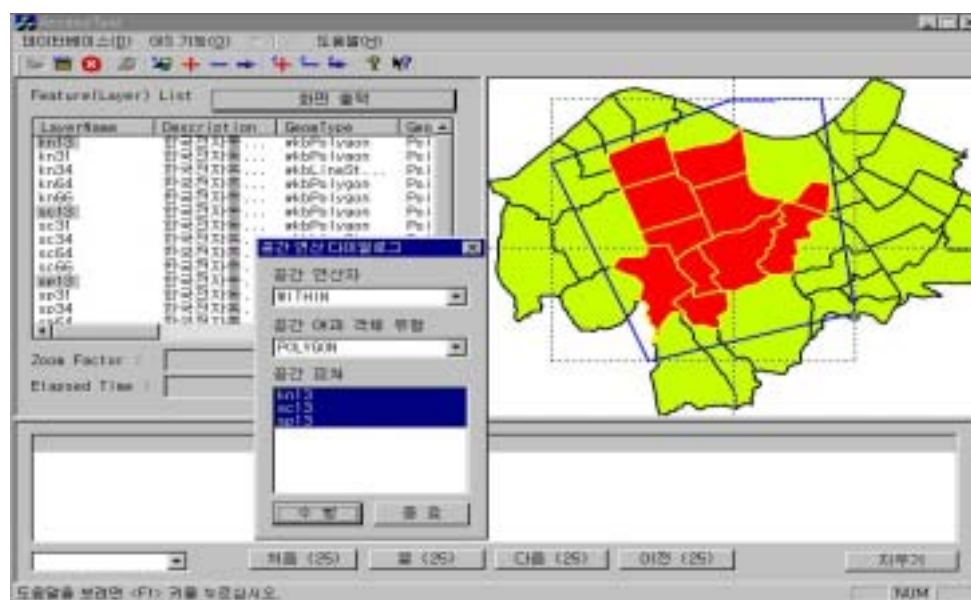
<표 3-5> OGC 데이터 제공자 컴포넌트의 공간 연산자 구현 사양

Property ID	Type_Indicator	Description
DBPROP_OGIS_TOUCHES	VT_BOOL	All points in the intersection of geometries of Data Source and the Spatial Filter lie on a geometry boundary and the interiors of the geometries of the Data Source and the Spatial Filter do not intersect.
DBPROP_OGIS_WITHIN	VT_BOOL	Geometries of the Data Source are wholly contained by the Spatial Filter.
DBPROP_OGIS_CONTAINS	VT_BOOL	The Spatial Filter is wholly contained by geometries of the Data Source.
DBPROP_OGIS_CROSSES	VT_BOOL	Geometries of the Data Source and the Spatial Filter intersect, but do not wholly contain each other, and the dimension of the intersection of their interiors is one less than the maximum dimension of their interiors.
DBPROP_OGIS_OVERLAPS	VT_BOOL	Geometries of the Data Source and the Spatial Filter intersect and the dimension of the intersection is the same as that of the input geometries but the intersection is different than the input geometries.
DBPROP_OGIS_DISJOINT	VT_BOOL	Intersection of geometries of the Data Source and the Spatial Filter is the empty set.
DBPROP_OGIS_INTERSECTS	VT_BOOL	Intersection of geometries of the Data Source and the Spatial Filter is not the empty set.
DBPROP_OGIS_ENVELOPE_INTERSECTS	VT_BOOL	Intersection of the envelop of geometries of the Data Source and the envelop of the Spatial Filter is not the empty set.
DBPROP_OGIS_INDEX_INTERSECTS	VT_BOOL	Intersection of the spatial index entries of the geometries of the Data Source and the geometry of the Spatial Filter is not empty set.

끝으로 OGC 데이터 제공자 컴포넌트의 공간 여과기의 처리 과정은 공간 정보 컬럼의 모든 객체들(C)과 입력으로 주어지는 공간 여과 객체(A)와의 관계를 선택된 공간 연산자(B)를 이용하여 연산자의 조건에 충족되는 후보 객체들을 여과시키는 방식으로 구성되어 있다. 다음 <그림 3-8> 과 <그림 3-9>는 각각 Intersects와 Within의 공간 여과의 수행 예제를 보여준다.



<그림 3-8> 공간 여과 객체: LineString, 공간 연산자: Intersects, 공간 객체: kn13, sc13, sp13 피쳐 테이블



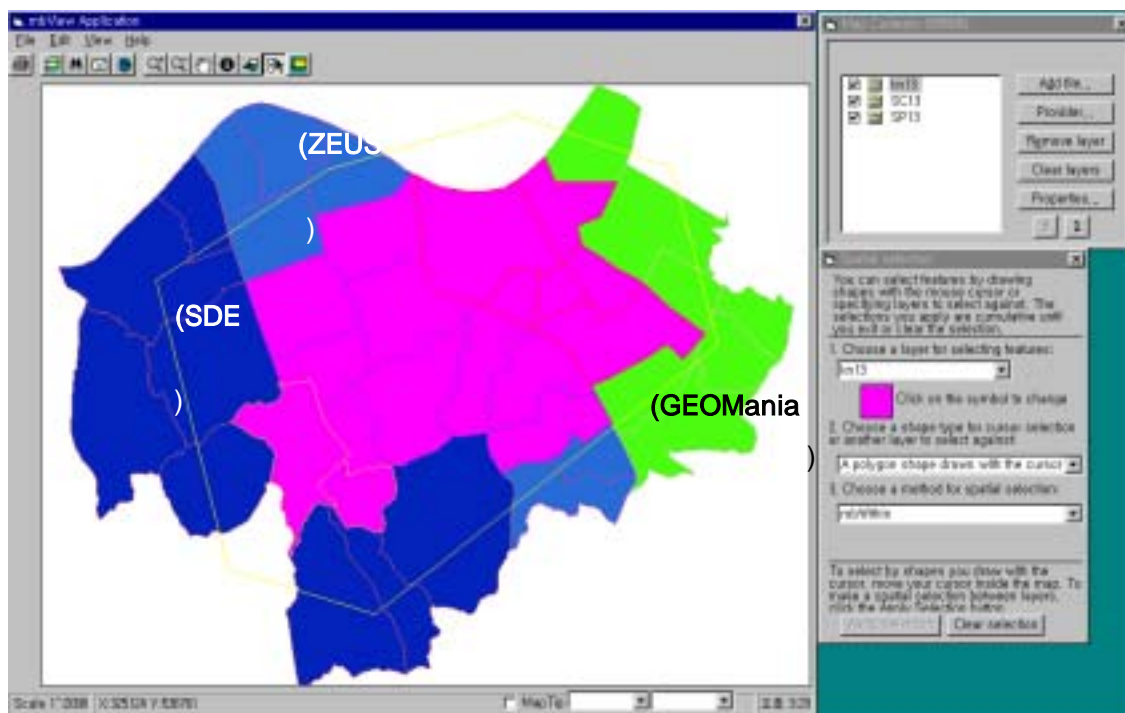
<그림 3-9> 공간 여과 객체: Polygon, 공간 연산자: Within, 공간 객체: kn13, sc13, sp13 피쳐 테이블

제4절 데이터 제공자 컴포넌트의 장점

OGC의 OLE/COM 구현 사양을 수용하고 다양한 GIS 소프트웨어 또는 GIS 파일 포맷에 대해 만들어질 수 있는 OGC용 데이터 제공자 컴포넌트의 장점을 요약해보면 다음과 같다.

- OGC용 데이터 제공자 컴포넌트의 동일한 인터페이스 메소드들을 이용하여 모든 상용 또는 비상용의 GIS 서버에 저장되어 있는 GIS 정보를 접근할 수 있다.
 - OGC의 데이터 제공자 컴포넌트 구현 사양의 수용
- OGC의 OLE/COM 구현 사양을 참조하여 누구나 자신의 GIS 서버에 대한 데이터 제공자 컴포넌트를 새로이 구축이 가능하다.
 - 기존의 GIS 소프트웨어에 대해 데이터 제공자 컴포넌트가 구축되어 있는 경우는 기존 구축된 데이터 제공자 컴포넌트 이용 가능
 - 새로운 GIS 파일 포맷 또는 새로운 GIS 소프트웨어에 데이터 제공자 컴포넌트 구축이 용이
- 분산 환경에서 동시에 2개 이상의 GIS 서버에 접속이 가능하다.
 - On-line 상에서 서로 다른 곳에 위치한 GIS 서버로부터 동시에 GIS 정보의 획득이 가능
 - 2개 이상의 GIS 서버에 대하여 공간 분석 및 공간 연산 작업 수행이 가능

<그림 3-10>은 분산 환경에서 SDE, ZEUS, GEOMania 데이터 제공자 컴포넌트를 이용하여 각각 서초구, 강남구, 송파구의 동 경계선 데이터를 로드하여 Polygon 여과기에 대하여 Within 공간 연산을 수행한 예제를 보여주고 있다. 이 예제에서 주의해야 할 점은 Within 공간 연산 작업이 각각의 GIS서버에서 수행된다는 점이다. 다시 말하면, 서초구 데이터의 Within 작업은 SDE에서, 강남구는 ZEUS에서, 그리고 송파구는 GEOMania에서 Within 연산이 수행된 다음에 각각의 결과가 클라이언트에 전송된다는 것이다. 결론적으로 데이터 제공자 컴포넌트를 이용하는 클라이언트 시스템은 분산 환경에서 언제나 다른 GIS 소프트웨어에 존재하는 데이터를 얻어올 수 있고 공간 연산 작업이 가능하다는 커다란 특징을 가지고 있다는 것이다.



<그림 3-10> SDE, ZEUS, GEOMania에 동시에 접근하여 공간 연산 작업을 수행한 예제

제5절 데이터 제공자 컴포넌트의 구성

1. GIS 데이터 소스별 데이터 제공자 컴포넌트 구성

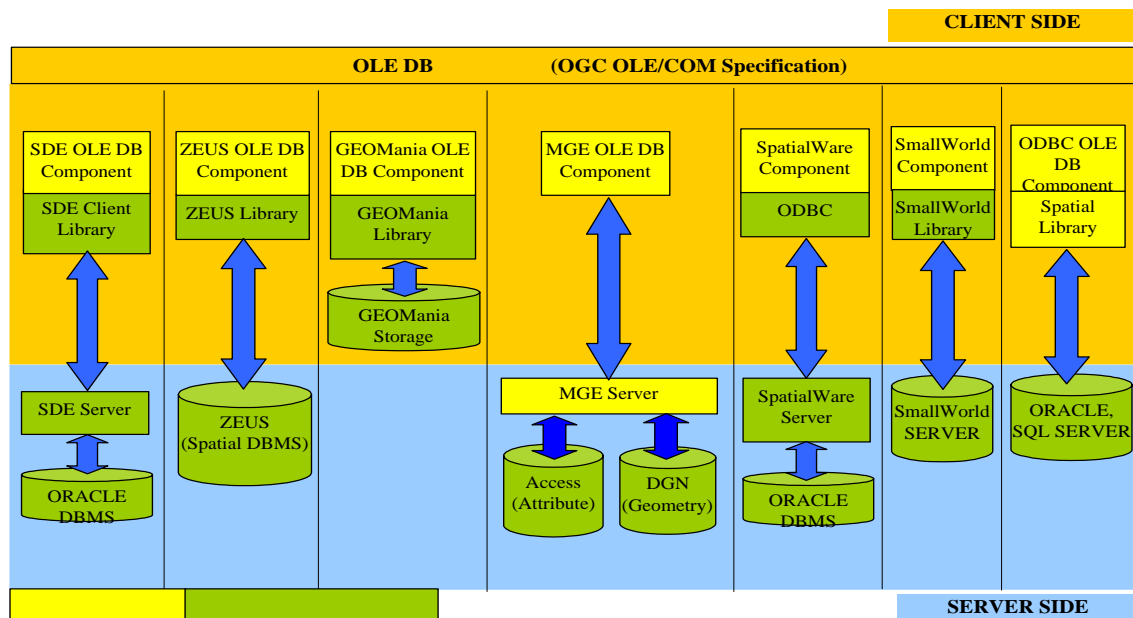
가. GIS 소프트웨어 vs GIS 파일 포맷

OGC의 사양을 수용한 데이터 제공자 컴포넌트를 구현하고자 할 때, 실제 구현되는 내용에 따라서 두 가지로 분류될 수 있다. 하나는 일반적인 GIS 소프트웨어에 대한 데이터 제공자 컴포넌트를 구축하는 것이고 또 다른 하나는 GIS 파일 포맷에 대한 데이터 제공자 컴포넌트를 구축하는 것이다. 주목할 점은 이 두 가지 분류에 대하여 데이터 제공자 컴포넌트를 구축할 때 소요되는 비용과 시간은 크게 차이가 난다는 것이다.

먼저 일반적인 GIS 소프트웨어를 사용하는 경우, 다시 말하면 GIS의 공간 엔진 기능을 기본적으로 모두 가지고 있다는 것을 의미한다. 그러므로 OGC의 데이터 제공자 컴포넌트

가 요구하는 메타 정보 관리 기능, 공간 연산 처리 기능, 공간 색인 기능 그리고 SQL 질의 처리 기능들이 모두 지원되고, 개발자는 그 GIS 소프트웨어의 API 메소드만을 이용하여 데이터 제공자 컴포넌트를 구현함으로써 비교적 용이한 작업이라고 볼 수 있다. 예를 들어, SDE, ZEUS, SmallWorld, MapInfo, GEOMania를 위한 데이터 제공자 컴포넌트 개발이 이에 해당된다. 그러나 단순히 GIS 파일 포맷을 이용하는 경우는 위의 경우와 정반대의 경우가 발생한다. 우선 메타 정보 관리 기능, 공간 연산 처리 기능, 공간 색인 기능 그리고 질의 처리 기능 중의 그 어느 것도 지원되지 않는다. 단순히 GIS 데이터를 저장하기 위한 파일 형식만이 주어질 뿐이다. 그러므로 데이터 제공자 컴포넌트를 구현하기 이전에 GIS 파일 포맷에 대하여 최소한 공간 연산 처리기와 질의 처리기를 구현해야 하는 어려움이 있고, 이러한 구현 작업은 많은 시간과 비용을 소모하게 된다. 만약 빠른 시간 내에 GIS 파일 포맷의 정보를 공유하기 원한다면 첫째로 공간 연산 및 질의 처리 기능을 지원하지 않는 읽기 전용(Read-Only)의 데이터 제공자 컴포넌트를 구축하는 방법과, 둘째로 GIS 파일 포맷을 데이터 제공자 컴포넌트가 구현되어 있는 다른 GIS 소프트웨어로 Importing 시킨 다음에 사용하는 방법의 두 가지가 가능하다. 이러한 GIS 파일 포맷으로는 대표적으로 GDF, Shape, DXF 등이 존재한다.

<그림 3-11>은 현재 국내에서 주로 이용되고 있는 GIS 소프트웨어를 이용하여 OGC용 데이터 제공자 컴포넌트를 구축하고자 할 때 구성될 수 있는 개략적인 시스템 구성 모델이다.



<그림 3-11> GIS 소프트웨어에 따른 데이터제공자 컴포넌트 시스템 구성 모델

<그림 3-11>을 살펴보면, 우선 SDE의 경우는 서버 측에 GIS 데이터 서버인 Oracle DBMS와 GIS 엔진인 SDE 서버를 내장하고 있으며, 클라이언트 측에는 클라이언트 시스템을 위한 SDE 라이브러리를 내장하고 있다. 그러므로 SDE 데이터 제공자 컴포넌트 SDE 클라이언트 라이브러리의 API 메소드를 통하여 SDE의 공간 엔진 기능과 Oracle SQL 처리 기능을 직접 이용함으로써 구현이 이루어질 수 있다.

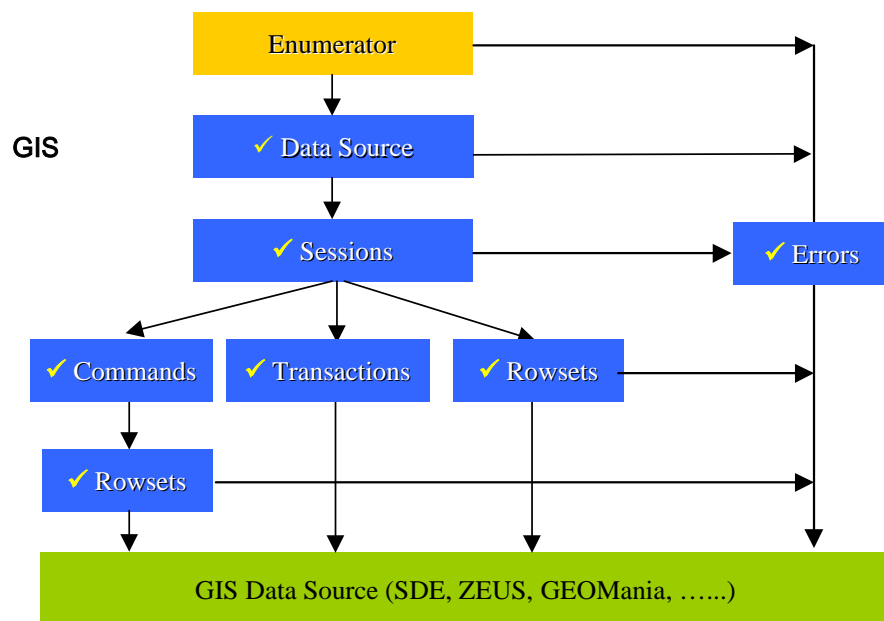
ZEUS의 경우도 SDE와 유사하게 ZEUS 클라이언트 라이브러리를 이용함으로써 데이터 제공자 컴포넌트를 구축할 수 있으나, 서버 측에서 SDE와 달리 공간 엔진 기능과 SQL 처리 기능이 ZEUS 라는 공간 데이터베이스에 통합되어 있다는 것만 다르다.

GEOMania는 현재 데스크 탑 버전과 클라이언트-서버 버전이 존재하며 위의 <그림 3-11>에서는 데스크 탑 버전에 대한 구성 모델을 보여주고 있다. 그러므로 GEOMania 라이브러리와 저장 구조가 모두 클라이언트에 위치하여 있으며, 공간 엔진 및 질의 처리 기능 역시 GEOMania 라이브러리에서 처리된다. 데이터 제공자 컴포넌트는 이 GEOMania 라이브러리의 API 메소드를 이용하여 구현된다.

그 밖에 MGE, SpatialWare, SmallWorld의 경우도 앞에서 설명한 바와 같이 유사한 방식으로 데이터 제공자 컴포넌트가 구성된다. 끝으로 위의 그림에서 마지막으로 구성된 ODBC 데이터 제공자 컴포넌트 모델은 좀 더 주의해서 볼 필요가 있다. ODBC 데이터 제공자 컴포넌트는 클라이언트 측의 공간 라이브러리(Spatial Library)를 이용하고 있으며, 서버 측의 SQL Server 또는 Oracle의 상용 데이터베이스를 이용하고 있다. 그러므로 ODBC 데이터 제공자 컴포넌트는 우선 서버 측의 데이터베이스를 이용하여 SQL 질의 처리 기능을 이용하고, 그 밖에 공간 엔진의 기능은 클라이언트 측에 직접 구현되어 있는 기능을 이용하는 방식이다. 이는 GIS 파일 포맷에 대한 데이터 제공자 컴포넌트를 구축하기 위한 전 단계로서, 추후에 데이터베이스 대신에 GIS 파일들을 연결하고 공간 라이브러리에 SQL 질의 처리 기능을 추가하면 GIS 파일 포맷을 위한 데이터 제공자 컴포넌트를 손쉽게 구축할 수도 있다.

2. 데이터 제공자 컴포넌트 세부 구성

OGC용 데이터 제공자 컴포넌트를 구현하기 위해서는 먼저 마이크로소프트사의 OLE DB 컴포넌트의 세부 구성을 파악해야 한다. OLE DB 컴포넌트는 Data Source 객체, Session 객체, Command 객체, 그리고 Rowset 객체를 기본적으로 구현해야 한다. <그림 3-12>는 OLE DB 컴포넌트의 각 객체들간의 구성 관계를 알 수 있다.



<그림 3-12> OLE DB 컴포넌트의 각 객체들간의 관계

Enumerator 객체는 시스템 내에 존재하는 모든 OLE DB 제공자 컴포넌트를 검색하는 역할을 수행하는 것으로서 사용자는 이 객체를 이용하여 자신이 원하는 데이터 제공자 컴포넌트를 선택할 수 있다. Enumerator 객체는 마이크로소프트사에서 기본적으로 구현되어 있는 객체이다. Data Source 객체는 데이터 제공자 컴포넌트에서 처음으로 생성되는 객체로서 사용자는 이 객체를 이용하여 자신이 원하는 GIS 데이터베이스에 접속하기 위한 초기화 작업과 연결 작업을 수행한다. 그리고 데이터베이스에 접속이 성공하면 Data Source 객체는 다음 작업을 위하여 Session 객체를 생성한다. Data Source 객체의 인터페이스 구성 및 반드시 구현해야 할 인터페이스는 <표 3-6>과 같다.

<표 3-6> Data Source 객체의 인터페이스 구성

컴포넌트 객체	인터페이스	선택사항	특징 및 구현 여부
Data Source	I DB Create Session	Mandatory	O (Session생성)
	I DB Initialize	Mandatory	O (접속처기화 및 연결)
	I DB Properties	Mandatory	O
	I Persist	Mandatory	O
	I Connertion Point Container	Optional	
	I DB AsynchStatus	Optional	
	I DB Data Source Admin	Optional	
	I DB Info	Optional	
	I Persis File	Optional	
	I Support ErrorInfo	Optional	

위의 그림에서 OLE DB의 필수(Madatory) 인터페이스는 기본적으로 반드시 구현해야 하며, 선택(Optional) 인터페이스는 필요에 따라서 구현하면 된다.

Data Source로부터 생성된 Session 객체는 데이터 제공자 컴포넌트의 기능에서 설명하였던 GIS 관련 메타 정보들을 지원하는 중요한 역할을 수행한다. 또한 SQL의 질의 처리를 수행하기 위하여 Command 객체를 생성하기도 하고, Transaction 처리를 위한 Transaction 객체를 생성하기도 한다. Session 객체의 인터페이스 및 반드시 구현해야 할 인터페이스는 <표 3-7>과 같다.

<표 3-7> Session 객체의 인터페이스 구성

컴포넌트 객체	인터페이스	선택사항	특징 및 구현 여부
Session	I Get Data Source	Mandatory	O
	I Open Rowset	Mandatory	O
	I Session Properties	Mandatory	O
	I DB Create Command	Optional	O(OGC 사양, Command 객체생성)
	I DB Schema Rowset	Optional	O(OGC 사양, 메타정보 접근)
	I Index Definition	Optional	
	I Support Error Infor	Optional	
	I Table Definition	Optional	
	I Transaction Join	Optional	
	I Transaction Local	Optional	
	I Transaction	Optional	
	I Transaction Object	Optional	

위의 표에서 “IDBCreateCommand”와 “IDBSchemaRowset” 인터페이스는 순수 OLE DB 컴포넌트에서는 선택 사항이지만, OGC에서는 SQL을 지원하기 위하여 그리고 GIS 관련 메타 정보를 지원하기 위하여 반드시 구현할 것을 명시하고 있다.

Session 객체로부터 생성되는 Command 객체는 SQL 명령어 또는 사용자가 제공하는 명령어를 처리하는 부분으로 SQL 명령어의 번역과 수행 작업을 수행한다. 물론 실제 SQL 명령어의 번역 및 수행 작업은 데이터 제공자 컴포넌트의 하부 구조를 구성하는 GIS 소프트웨어에서 처리된다. 이와 더불어 Command 객체의 중요한 기능은 SQL 명령어를 수행할 때 ‘데이터 제공자 컴포넌트의 기능’에서 설명하였던 공간 여과를 위한 세 가지 파라미터 -공간 여과 객체, 공간 연산자, 피쳐 테이블의 공간 정보 칼럼 이름- 를 동시에 처리하는 것이다. <표 3-8> 은 Command 객체의 인터페이스 구성과 구현해야 할 인터페이스를 보여준다.

<표 3-8> Command 객체의 인터페이스 구성

컴포넌트 객체	인터페이스	선택사항	특징 및 구현 여부
Command	I Accessor	Mandatory	○
	I Columns Info	Mandatory	○
	I Command	Mandatory	○
	I Command Properties	Mandatory	○
	I Command Text	Mandatory	○
	I Convert Type	Mandatory	○
	I Columns Rowset	Optional	○(OGC 사양)
	I Command Prepare	Optional	
	I Command With Parameters	Optional	○(OGC 사양, 공간 여과 Parameter 처리)
	I Support Error Info	Optional	

위의 표를 보면 Command 객체는 순수 OLE DB 사양에서 공간 연산 작업을 처리하기 위하여 선택 인터페이스인 “ICommandWithParameters” 인터페이스를 추가로 구현하고 있음을 알 수 있다.

끝으로 Command 객체에서 명령어가 성공적으로 수행되면 사용자가 원하는 결과 데이터들을 저장하기 위한 Rowset 객체가 생성된다. 그러므로 클라이언트 사용자가 GIS 데이터에 접근하기 위해서는 Rowset 객체에 접근하는 방법을 이해하고 있어야 한다. Rowset 객체에서는 생성된 객체들에 대하여 검색, 수정, 갱신, 색인 등의 다양한 작업을 수행할 수 있으며, 데이터 제공자 컴포넌트는 필요에 따라서 이러한 기능들을 위한 인터페이스를 구현하고 있어야 한다. Rowset 객체의 인터페이스 구성 및 필수 구현 인터페이스는 <표 3-9>와 같다.

<표 3-9> Rowset 객체의 인터페이스 구성

컴포넌트 객체	인터페이스	선택사항	특징 및 구현 여부
Command	I Accessor	Mandatory	○
	I Columns Info	Mandatory	○
	I Convert Type	Mandatory	○
	I Rowset	Mandatory	○
	I Rowset Info	Mandatory	○
	I Chaptered Rowset	Optional	
	I Columns Rowset	Optional	○(OGC 사양)
	I Connettion Point Container	Optional	
	I DB Asynch Status	Optional	
	I Rowset Change	Optional	○(편집)
	I Rowset Find	Optional	
	I Rowset Identity	Optional	
	I Rowset Index	Optional	
	I Rowset Locate	Optional	
	I Rowset Refresh	Optional	
	I Rowset Scroll	Optional	
	I Rowset Update	Optional	○(편집)
	I Rowset View	Optional	
	I Rowset Erro Info	Optional	

제6절 결 론

본 장에서는 다양한 GIS 소프트웨어 또는 GIS 파일 포맷 등에서 사용될 수 있는 OGC용 데이터 제공자 컴포넌트의 개념 및 필요성 그리고 데이터 제공자 컴포넌트의 기능과 구성에 대하여 자세히 살펴보았다. 실제로 이러한 OGC용 데이터 제공자 컴포넌트를 구현하기 위해서는 OGC에서 제공하는 "OLE/COM을 위한 Simple Feature 구현 사양" 이외에 마이크로소프트사의 OLE DB 컴포넌트 구현 사양을 참고하도록 해야 한다. 그리고 개발 툴로서는 마이크로소프트사의 비주얼 스튜디오를, COM 프로그래밍을 위해서는 ATL/COM의 개발 방법을 사용하는 것이 비용과 시간 면에서 효율적이다.

끝으로 지금까지 설명한 OGC용 데이터 제공자 컴포넌트는 다음과 같은 특징 및 장점을 가지고 있음을 알 수 있다.

- 분산 환경에서 다양한 GIS 데이터 소스를 접근하고 분석하기 위해서는 현재로서는 OGC에서 제시하는 데이터 제공자 컴포넌트 이외에는 다른 대안이 없는 실정이다.
- OGC에서 제시하는 OLE/COM 구현 사양과 마이크로소프트사에서 제시하는 OLE DB 컴포넌트 구현 사양을 이용하여 누구나 자신의 GIS 데이터 소스에 대한 데이터 제공자 컴포넌트를 구현할 수 있다.
- 컴포넌트 기반으로 분산 컴퓨팅 환경에서 기관별 공간 정보 유통을 직접적으로 지원할 수 있다.
- OGC용 데이터 제공자 컴포넌트는 OLE DB 구현 사양 이외에 GIS적인 요소를 처리하기 위하여 GIS 관련 메타 정보 지원과 공간 연산 기능이 필수적으로 요구된다.
- SDE, ZEUS, GEOMania, SmallWorld, MapInfo, Shape, ODBC를 위한 데이터 제공자 컴포넌트가 기 개발되었거나 개발 중에 있다.

제4장 핵심 공통 컴포넌트

제1절 핵심 공통 컴포넌트의 개념

제2절 핵심 공통 컴포넌트의 기능 및 구성

제3절 핵심 공통 컴포넌트의 수행 예

제4절 결 론

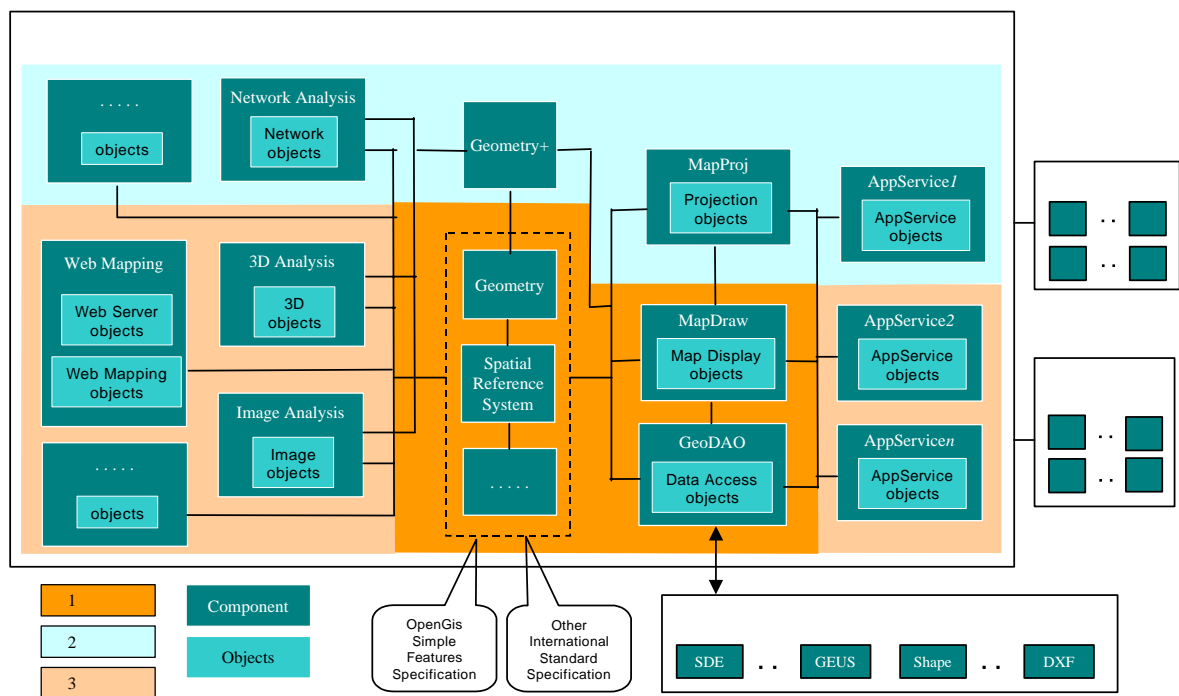
제1절 핵심 공통 컴포넌트의 개념

핵심 공통 컴포넌트는 데이터 제공자 컴포넌트와 GIS 응용 시스템의 가운데에 위치하여 GIS 응용 시스템들과 데이터 제공자 컴포넌트와의 다리 역할을 수행하는 컴포넌트이다. 실제로 GIS 응용 시스템을 구축하고자 할 때 데이터 제공자 컴포넌트를 직접 접근하는 방법보다는 훨씬 이용하기 편한 방법을 제공하는 것이 핵심 공통 컴포넌트의 주요 목적이다. 핵심 공통 컴포넌트는 여러 가지 다양한 기능을 수행하는 컴포넌트들로 구성되어 있으며, 일부 컴포넌트는 OGC에서 제안하는 “OLE/COM을 위한 Simple Feature 구현 사양”에 준수하여 구성된다.

핵심 공통 컴포넌트는 GIS 응용 시스템의 개발에 필요한 기본 기능들을 제공하는 것으로서, Geometry 컴포넌트, Spatial Reference System 컴포넌트, GeoDAO 컴포넌트, MapDraw 컴포넌트, Network 컴포넌트 그리고 Application Service 컴포넌트들로 구성되어 있다. 특히 이중에서 Geometry 컴포넌트와 Spatial Reference System 컴포넌트는 OGC에서 제시하는 구현 사양에 의해 구성되고, 이외의 컴포넌트들은 현재까지 표준으로 발표된 OGC 및 기타 사양이 없으므로 인하여 기존의 대중적인 GIS 소프트웨어를 기준으로 하여 구성되고 있다. 이러한 핵심 공통 컴포넌트의 자세한 기능과 구성에 대해서는 다음 절에서 살펴보고자 한다.

제2절 핵심 공통 컴포넌트의 기능 및 구성

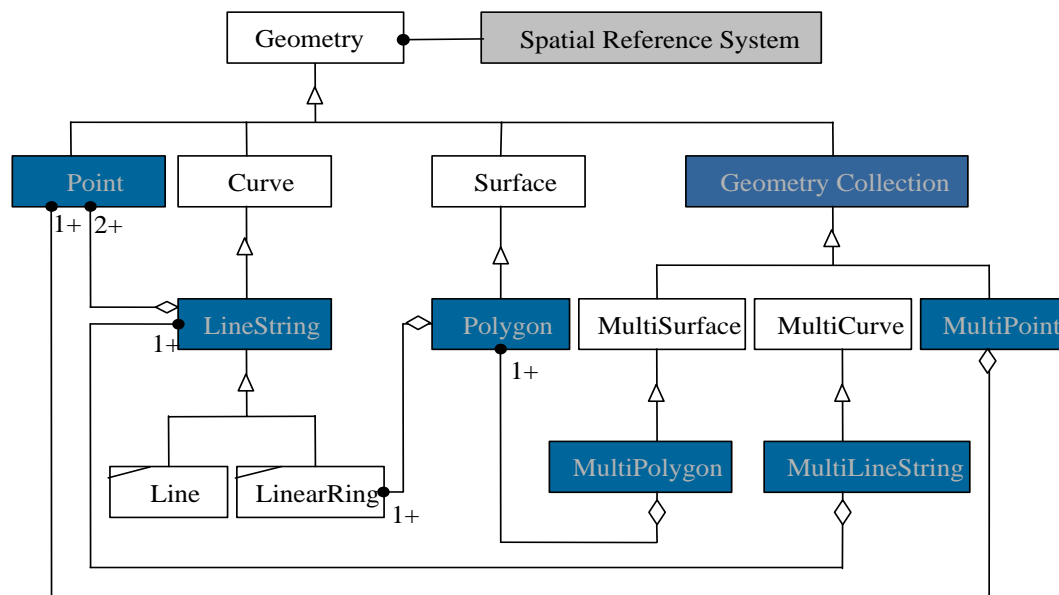
본 절에서는 Geometry 컴포넌트, Spatial Reference System 컴포넌트, GeoDAO 컴포넌트, MapDraw 컴포넌트, Network 분석 컴포넌트와 Application Service 컴포넌트로 구성되어 있는 핵심 공통 컴포넌트들의 주요 기능에 대하여 살펴보고자 한다. 특히 각 컴포넌트들이 응용 시스템에 어떠한 서비스들을 제공할 수 있으며, 또한 어떠한 근거 -OGC 사양 또는 사용 GIS물 사양- 에 의하여 컴포넌트로 구성되었는지에 관하여 살펴보고자 한다. <그림 4-1>은 현재 계획중인 핵심 공통 컴포넌트의 전체적인 구성도이며, 현재 3차년도 관련 컴포넌트 개발을 진행하고 있다.



<그림 4-1> 핵심 공통 컴포넌트의 전체적인 구성도

1. Geometry 컴포넌트

Geometry 컴포넌트는 공간 데이터를 접근하고, 분석하고, 그리고 연산 작업을 수행하기 위한 인터페이스들을 포함하고 있다. OGC에서는 “OpenGIS Simple Feature Specification For OLE/COM Revision 1.1” 사양에서 이러한 Geometry 컴포넌트에 대한 상세한 인터페이스 구성 및 기능을 정의하고 있다. Geometry 컴포넌트에서 사용될 수 있는 공간 객체 타입은 Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon과 GeometryCollection이 있으며, 이들 각 공간 객체들은 자신을 위한 필수적인 인터페이스들을 정의하고 있다. 이들 공간 객체들은 OGC에서 제시하는 <그림 4-2>와 같은 Geometry Object 모델을 따르고 있다.



<그림 4-2> Geometry Object 모델

위의 그림에서 Geometry 클래스는 Root 클래스를 나타내고 추상 클래스로서 객체로서 생성될 수 없는 클래스이며, 객체로서 생성될 수 있는 Geometry 관련 클래스는 <그림 4-2>에서 Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon, Geometry Collection의 2차원 상의 7 가지만 가능하다. 그리고 Geometry 클래스는 기본적으로 <표 4-1>과 같은 속성과 메소드들을 구현하고 있어야 한다.

<표 4-1> OGC Geometry 클래스의 속성 및 메소드 구성

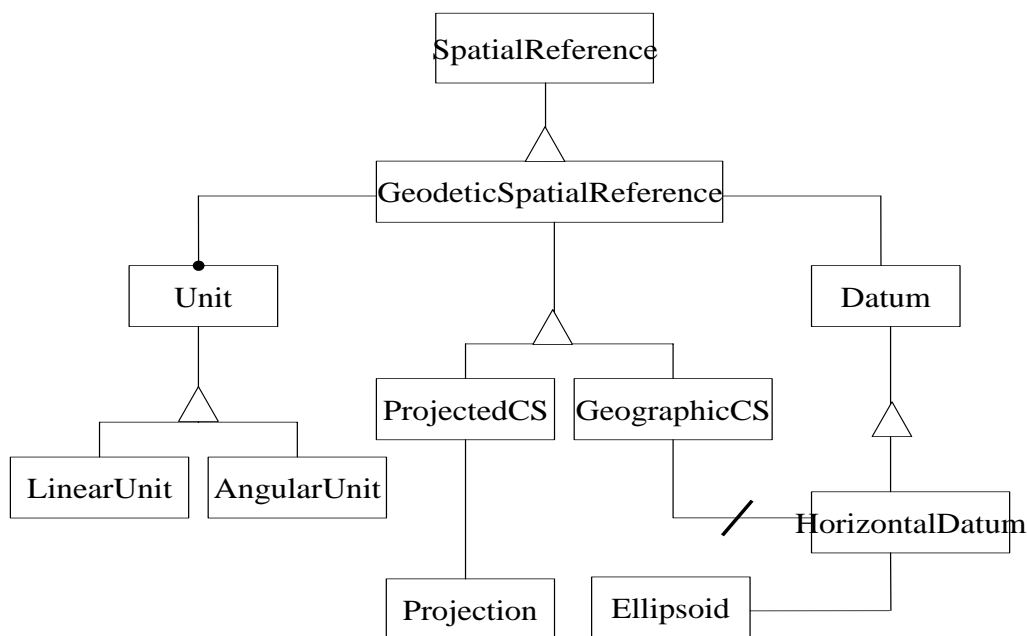
Attributes	Methods		
	Basic Methods	Methods for Testing Relations	Methods for Spatial Analysis
Dimension Is Empty Is Simple Spatial Reference	Clone() Envelop() Extent2D() Project() Set Empty()	Contains() Crosses() Disjoint() Equals() Intersects() Overlaps() Touches() Within()	Boundary() Buffer() Convex Hull() Difference() Distance() Intersection() Symmetric Difference() Union()

위의 표와 같이 모든 공간 객체의 Root 클래스인 Geometry 클래스는 Dimension, IsEmpty, IsSimple, SpatialReference와 같은 네 가지의 속성 정보, Envelop()과 같은 기능의 다섯 가지 기본 메소드, Contains()같은 여덟 가지의 관계 연산자 메소드, 그리고 Buffer()와 같은 여덟 가지의 공간 분석 메소드들을 구현해야 한다. 물론 이러한 Geometry 클래스는 상속받아야 하는 모든 다른 Point, LineString, Polygon 등의 클래스들은 위의 속성과 메소드들을 모두 이용할 수 있다. Point/MultiPoint 클래스는 속성 정보로서 'x'와 'y' 값을 가지고 있으며, 이 'x'와 'y' 값을 세팅하기 위한 'Coords()' 메소드를 구현해야 한다. LineString/MultiLineString 클래스는 속성 정보로서 'NumPoints' 값을, 메소드로서 LineString에서 임의의 위치에 있는 Point 정보를 추출할 수 있는 'Point()' 함수를 구현해야 한다. Polygon/MultiPolygon 클래스는 속성 정보로서 폴리곤 내부의 Ring 개수를 나타내는 'NumInteriorRings' 정보를, 메소드로서는 폴리곤 외부의 Ring을 반환하는 'ExteriorRing()' 함수와 인덱스된 위치의 내부 Ring을 반환하는 'InteriorRing()' 함수를 가지고 있다.

이와 같이 Geometry 컴포넌트는 다양한 공간 객체에 대한 다양한 속성과 메소드들을 정의하고 있는 클래스들의 집합으로 이루어져 있음을 알 수 있다. Geometry 컴포넌트를 구현하기 위해서는 OGC의 OLE/COM 사양에서 Geometry 컴포넌트를 위해 제시되는 인터페이스 문서를 참조하면 용이하게 구현할 수 있다.

2. Spatial Reference System 컴포넌트

Spatial Reference System 컴포넌트 역시 OGC의 OLE/COM 구현 사양에서 명확히 구현 사양을 명시하고 있다. <그림 4-3>은 OGC에서 제시하는 Spatial Reference System Object 모델에 관해서 설명하고 있다.



<그림 4-3> Spatial Reference System Object 모델

위의 Spatial Reference System Object 모델은 OpenGIS 추상 사양에 근거를 두고 EPSG(European Petroleum Survey Group)와 POSC(Petrotechnical Open Software Corp)의 Spatial Reference System을 위한 Geodetic 모델을 이용하고 있다.

OGC 구현 사양에서는 이와 같은 Spatial Reference System 컴포넌트를 위해서는 다양한 좌표 투영(Projection) 기법과 다양한 타원체간의 좌표 변환 기법을 제공하는 인터페이스를 구현하도록 제시하고 있다. 좌표 투영 기법으로는 Mercator 기법, Albers Conic 기법, Cylindrical 기법 등의 40 종류를 구현하도록 하고 있으며, 좌표 변환 기법으로는 Standard Molodensky 기법과 7-Parameters 기법을 구현하도록 하고 있다. 이러한 Spatial Reference System 관련 정보는 OGC 사양에 정의되어 있는 WKT(Well-Known Text)의 형식으로 제공하도록 규정하고 있다.

Spatial Reference System 컴포넌트 역시 OGC 사양에서 인터페이스를 제공하고 있으며 이러한 인터페이스를 참조하여 구현하는 데에는 큰 어려움이 없다.

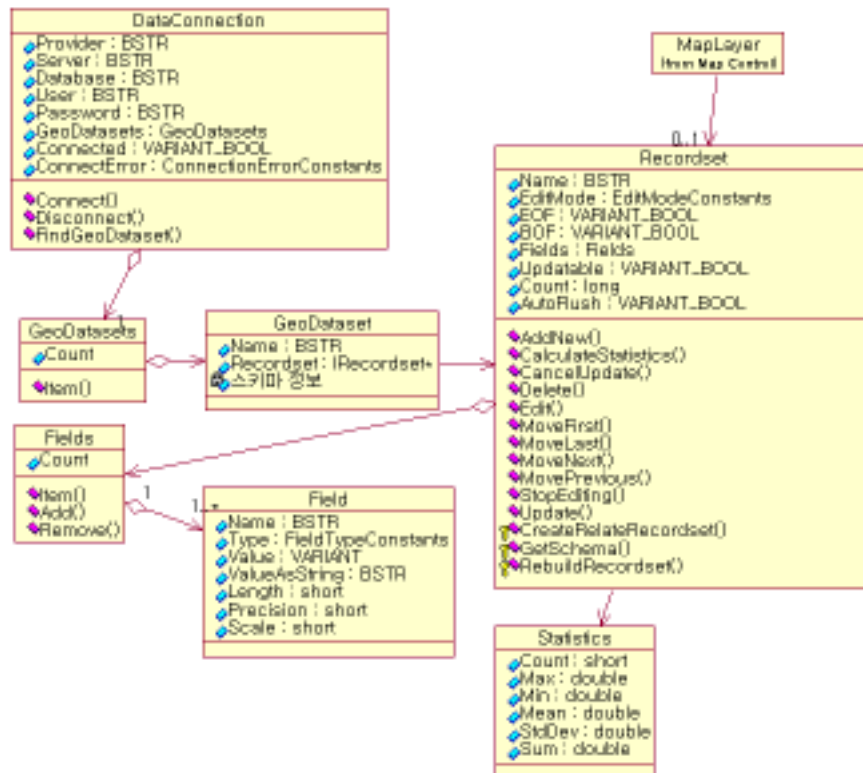
3. GeoDAO 와 MapDraw 컴포넌트

GeoDAO 컴포넌트와 MapDraw 컴포넌트는 OGC의 OLE/COM 구현 사양에 명시되어 있지 않은 사용자 위주의 컴포넌트들이다. 그러므로 이들 컴포넌트들은 기존에 사용자들이 가장 많이 이용하고 있는 상용의 컴포넌트 GIS 소프트웨어를 기준 모델로 삼고, 이를 바탕으로 기능 및 인터페이스가 정의되었다. 그러므로 이들 두 컴포넌트는 완전히 확정된 것이 아니라, 언제라도 사용자의 요구에 따라서 기능 및 인터페이스가 변경 또는 확장될 수 있는 특징을 가지고 있다.

GeoDAO 컴포넌트의 기본 기능은 제3장에서 설명되었던 OGC용 데이터 제공자 컴포넌트를 연결하고 관리하는 것으로 자세한 기능은 다음과 같다.

- OGC용 데이터 제공자(OLE DB) 컴포넌트 접속 기능
- 동시에 다중 데이터 제공자 컴포넌트를 이용한 다중 GIS 데이터베이스 연결 기능
- GIS 데이터베이스의 메타 정보 접근 및 관리 기능
- GIS 데이터베이스의 다중 피쳐 접근 및 관리 기능
- 공간 및 속성 정보의 접근 및 관리 기능

이러한 GeoDAO 컴포넌트의 기능을 간단하게 정리하면, 우선 데이터 제공자 컴포넌트에 접속하여 다양한 GIS 데이터베이스에 접속하고, GIS 데이터베이스로부터 메타 정보 및 GIS 정보를 획득하고, 공간 연산 작업을 수행하고 난 다음에 그 결과를 MapDraw 컴포넌트에 전달하는 것이다. 물론 3장에서 설명하였듯이 OGC용 데이터 제공자 컴포넌트는 이와 같은 GeoDAO 컴포넌트가 없어도 사용자가 OLE DB 인터페이스를 이용하여 자유로이 접근하여 이용할 수 있다. 그러나, 핵심 공통 컴포넌트에서 이러한 GeoDAO 컴포넌트를 제공하는 것은 사용자에게 조금 더 용이한 방식으로 데이터 제공자 컴포넌트를 사용할 수 있고 이외에 다양한 부가 기능을 제공하기 위함이다. <그림 4-4>는 이러한 GeoDAO의 컴포넌트 구성도를 보여주고 있다.



<그림 4-4> GeoDAO의 컴포넌트 구성도

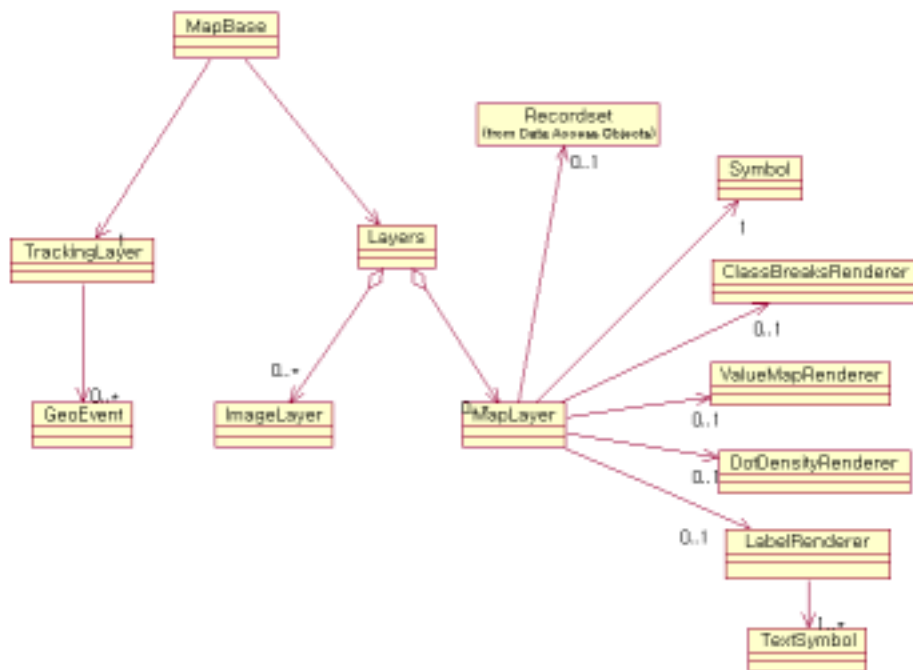
위의 그림에서 DataConnection 객체는 데이터 제공자 컴포넌트를 접근하기 위한 기능을 주로 수행한다. GeoDataSet 객체는 데이터 제공자 컴포넌트로부터 접근 가능한 GIS 메타 정보를 관리하고, GeoDataSets 객체는 GeoDataSet 객체의 컬렉션으로 구성되어 있다. RecordSet 객체는 데이터 제공자로부터 얻어오는 정보를 저장하고 관리하는 객체로서 데이터 제공자 컴포넌트의 Rowset 객체로부터 정보를 얻어온다. Field 객체는 RecordSet 객체의 각 컬럼에 관한 정보를 저장, 관리하며, Fields 객체는 Field 객체의 컬렉션으로 구성되어 있다. 끝으로 Statistics 객체는 RecordSet에 대한 간단한 통계자료를 관리하는 클래스로서 레코드의 개수, 최대 값, 최소 값, 평균 값 등을 관리한다.

둘째, OCX 컨트롤의 형태로 구성되어 있는 MapDraw 컴포넌트의 주요 기능은 공간 및 속성 정보를 화면에 표현하기 위해 필요한 모든 기능을 제공하는 것으로 자세한 내용은 다음과 같다.

- Zoom 기능, Pan 기능 그리고 Resize 기능의 MapDraw 컨트롤의 윈도우 관리 기능

- MapLayer, ImageLayer, TrackingLayer 등의 다양한 Layer 지원 기능
- Geometry, Image 등의 데이터 Mapping 기능
- RangeMap, UniqueMap, DensityMap, LabelMap 등의 주제도 생성 기능
- Mouse 등의 다양한 이벤트 처리 기능

이와 같이 MapDraw 컴포넌트는 GeoDAO 컴포넌트로부터 전달되는 공간 및 속성 정보에 대하여 다양한 매핑과 주제도 생성 기능 등을 이용하여 사용자에게 편리한 맵 도시 기능을 제공하는 것이 주 목적이다. <그림 4-5>는 이러한 MapDraw 컴포넌트의 컴포넌트 구성도를 설명하고 있다.

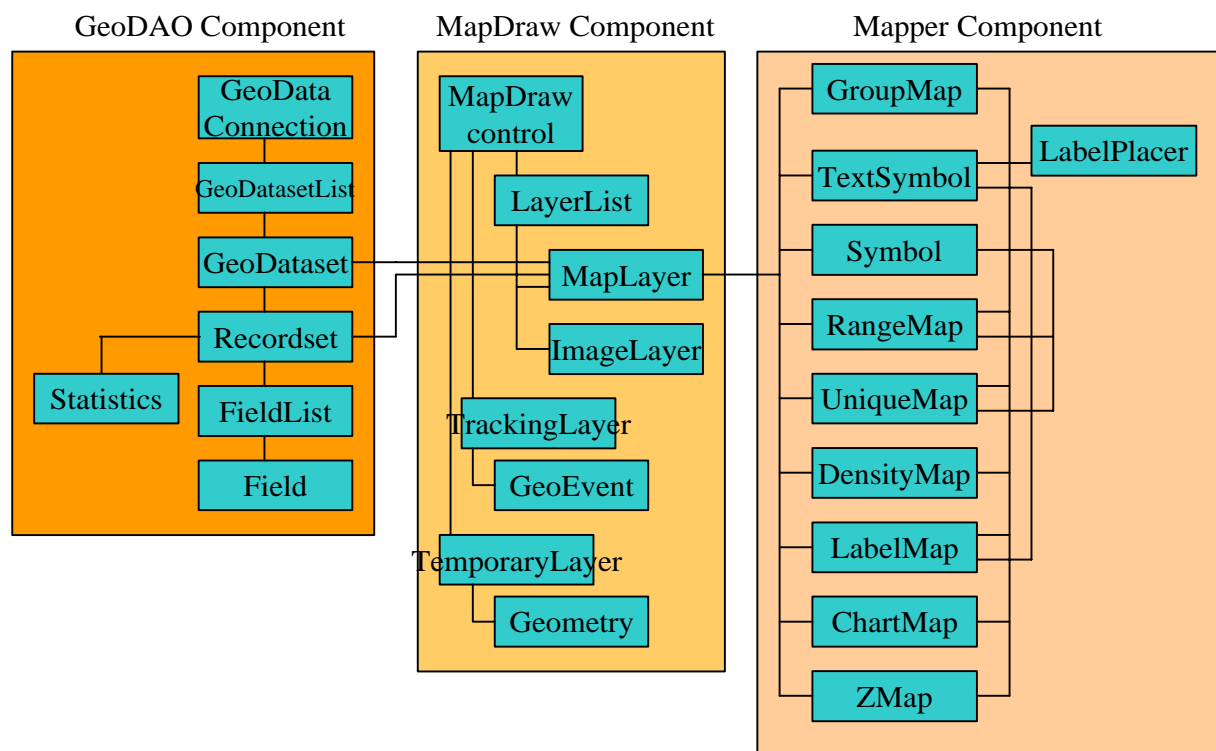


<그림 4-5> MapDraw의 컴포넌트 구성도

위의 그림에서 첫째로 MapBase 객체는 GeoDAO 컴포넌트가 제공하는 속성 및 공간 데이터를 이용하여 지리 정보를 화면에 표현하는 ActiveX 컨트롤로 구성되어 있다. MapLayer 객체는 MapBase 컴포넌트에 표시될 레이어에 대한 정보를, ImageLayer 객체는 MapBase 컴포넌트에 표시될 Image 파일에 대한 정보를, 그리고 TrackingLayer 객체는 MapBase 컴포넌트에 표시될 임시 공간 정보를 관리하며, Layers 객체는 이러한 MapLayer, ImageLayer 또는 TrackingLayer 객체의 컬렉션으로 구성되어 있다. Symbol

객체는 공간 정보가 표시되는 방법을, ClassBreaksRender 객체는 공간 정보들이 분류별로 표시되는 방법을 ValueMapRender 객체는 공간 정보들의 값에 따라 표시되는 방법을, DotDensityRender 객체는 공간 정보들이 점으로 표시되는 방법을, LabelRender 객체는 공간 정보들이 문자열로 표시되는 방법을 제공하고 있다.

끝으로 그림 <4-6>은 이러한 GeoDAO 컴포넌트와 MapDraw 컴포넌트가 어떠한 관계로 연결되어 있는지를 보여준다.



<그림 4-6> GeoDAO 컴포넌트와 MapDraw 컴포넌트의 연결 관계

위의 <그림 4-6>에서 Mapper 컴포넌트는 MapDraw 컴포넌트의 매핑, 렌더링, 심볼링 등의 기능을 세부적으로 구현하기 위한 컴포넌트 클래스들로서 본 장에서 세부적인 설명은 생략하도록 한다.

지금까지 설명한 바와 같이 사용자가 만약 OGC용 데이터 제공자 컴포넌트, Geometry 컴포넌트 그리고 Spatial Reference System 컴포넌트만을 직접 이용하지 않고, 비록 OGC 사양은 아니라고 할지라도 데이터 제공자 접근기능과 매핑 관련 기능을 제공해주는 GeoDAO 컴포넌트와 MapDraw 컴포넌트를 추가적으로 사용한다면 응용 시스템이나 용

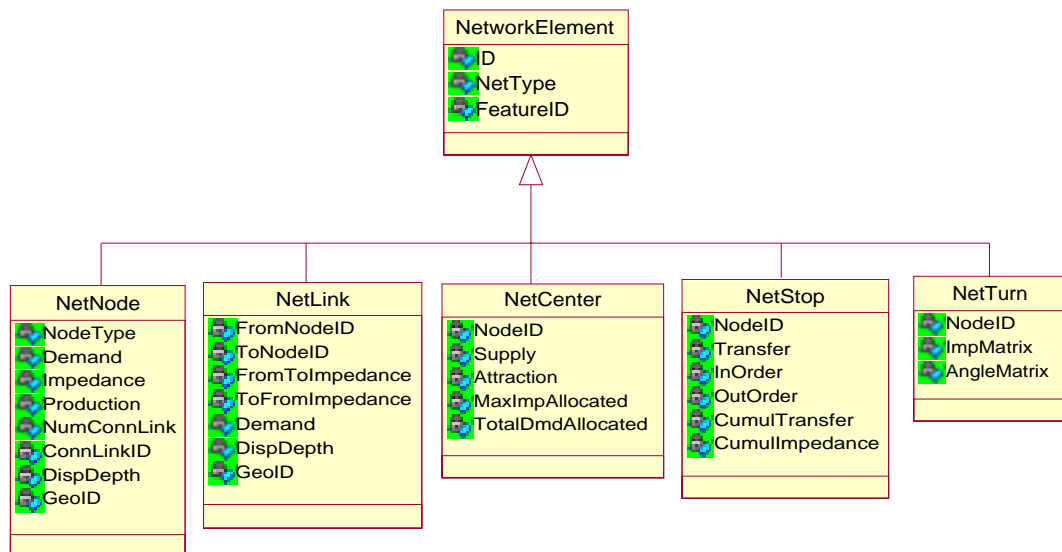
용 컴포넌트의 구축이 매우 용이해지는 장점을 가진다는 것을 알 수 있을 것이다. 그러므로 사용자는 특별한 목적이 없다면 GeoDAO와 MapDraw 컴포넌트를 이용함으로써 데이터 제공자 접근 기능 및 매핑 관련 기능을 별도로 구축함으로써 소모되는 노력과 시간을 줄이는 것이 바람직하다고 할 수 있다.

4. Network 분석 컴포넌트

GIS의 기능에서 중요한 부분을 차지하는 것들 중의 하나가 바로 네트워크 분석 기능이라고 할 수 있다. 특히, ITS나 지하 시설물 관리 시스템과 같은 응용 분야에서는 네트워크 분석 기능이 절대적으로 요구된다. 그러므로 핵심 공통 컴포넌트에서는 개방형 컴포넌트 GIS 개념에 적합한 컴포넌트 기반의 네트워크 분석 기능을 구현하고 있다. 본 네트워크 분석 컴포넌트의 기능 및 인터페이스들은 Arc/Info의 네트워크 모델을 참조하여 구성하였으며, 각 기능들은 다음과 같다.

- Utility 네트워크 분석 기능 : Trace, Allocate, Interaction, Accessibility
- Transportation 네트워크 분석 기능 : Path, Tour
- 네트워크 분석을 위해 필요한 위상 및 속성 정보 관리 기능
- 새로이 추가되거나 삭제되는 공간 정보를 반영하여 네트워크 분석을 가능하게 해주는 Incremental 위상 정보 구축 기능

다음 <그림 4-7>은 핵심 공통 컴포넌트에서 제공하는 Network 분석 컴포넌트의 데이터 모델을 보여준다. 이 중에서 실제로 네트워크 분석 작업에서 사용되는 객체는 NetNode, NetLink, NetCenter, NetStop, NetTurn의 다섯 가지이다.



<그림 4-7> Network 분석 컴포넌트의 네트워크 데이터 모델

위의 그림에서 NetNode는 네트워크에서 의미 있는 한 지점을 나타내며, NetLink는 네트워크 상의 한 지점에서 다른 지점으로의 이동 경로를 나타낸다. NetCenter는 네트워크 상에서 Link나 Node를 통하여 Resource를 받아들이거나 분배할 수 있는 Node로서, 예를 들면 발전소와 같이 ‘전기’라는 Resource를 분배할 수 있는 지점을 의미한다. NetStop은 NetCenter에서 분배한 자원을 Pick up하거나 Drop off하는 위치로서, 예를 들면 화물 하역장 같이 화물차가 ‘화물’이라는 Resource를 싣고 내리는 지점을 의미한다. 끝으로 NetTurn은 임의의 Link로부터 한 Node를 통하여 다른 Link로 이동하는 경로를 말한다. 다시 말하면 이와 같은 네트워크 데이터 모델은 Arc/Info의 네트워크 모델을 참조하여 구성된 것이다.

끝으로 Network 분석 컴포넌트는 시설물 관리 응용 시스템과 같이 네트워크 분석 기능을 포함하는 경우에만 필요로 하므로 그 외의 응용 사업에서는 필요에 따라서 Network 분석 컴포넌트를 사용하지 않아도 무방하다.

5. Application Service 컴포넌트

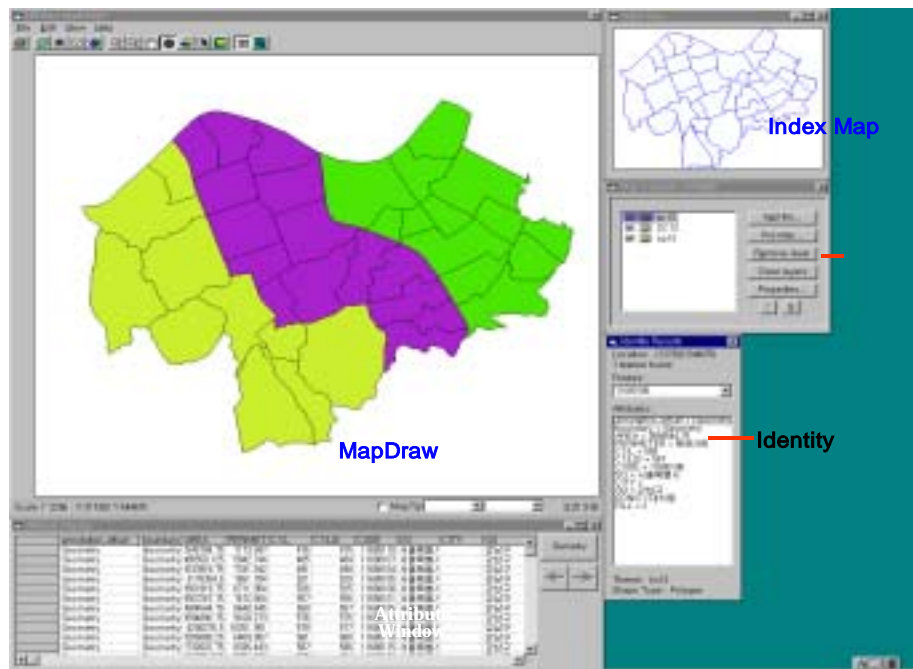
일반적으로 도로, 도시계획, 토지, 상하수도과 같은 지자체 응용 시스템 또는 기타의 GIS 응용을 위해 구축된 컴포넌트를 자세히 살펴보면 업무에 따라서 공통적인 기능을 요구하는 부분이 있음을 알 수 있다. 그러나 GIS 사업에 따라서 공통 업무가 존재함에도 불

구하고 매번 사업을 수행할 때 같은 업무의 기능을 새로이 구현하는 일이 비일비재하였다. 그럼으로 개방형 GIS 컴포넌트 기술 개발 연구에서는 이러한 GIS 응용에 있어서의 공통 업무를 추출하고 이렇게 추출된 공통 업무에 대한 표준 Application Service 컴포넌트를 구축함으로써 추후에 GIS 사업자가 편리하게 이용할 수 있도록 계획하고 있다. 그러나 Application Service 컴포넌트의 공통 업무를 추출하는 과정이 다양한 GIS 업체들의 이해 관계 그리고 공통 응용 업무에 대한 표준을 제정하는 문제 등으로 인하여 원활하게 수행하기에는 어려움이 매우 많다. 따라서 현재 개방형 GIS 컴포넌트 기술 개발 연구에서는 다양한 GIS 사업체들의 의견을 수렴하기에 노력하고 공통 업무를 추출하여 Application Service 컴포넌트의 인터페이스를 구성하기 위하여 연구를 진행하고 있다. Application Service 컴포넌트의 결과물로는 아직까지 일부만이 발표되었으며 계속하여 컴포넌트의 정의와 구현이 진행 중에 있는 상태이다.

제3절 핵심 공통 컴포넌트의 수행 예

OLE/COM 기반으로 구성되어 있는 핵심 공통 컴포넌트는 COM 환경의 특징으로 인하여 다양한 환경에서 클라이언트 프로그램을 작성하여 접근할 수 있다. 그러므로 본 예제에서는 Visual C++ 환경과 Visual Basic의 두 가지 환경으로부터 핵심 공통 컴포넌트와 데이터 제공자 컴포넌트를 접근하여 수행한 예제 결과를 보여줄 것이다. 또한 핵심 공통 컴포넌트로는 제1절에서 설명한 Geometry, Spatial Reference System, GeoDAO, MapDraw, Network Service 컴포넌트를 이용하였으며, GIS 데이터베이스 접근을 위해서 제3장에서 설명하였던 SDE, ZEUS, GEOMania 등의 OGC용 데이터 제공자 컴포넌트를 이용하였다.

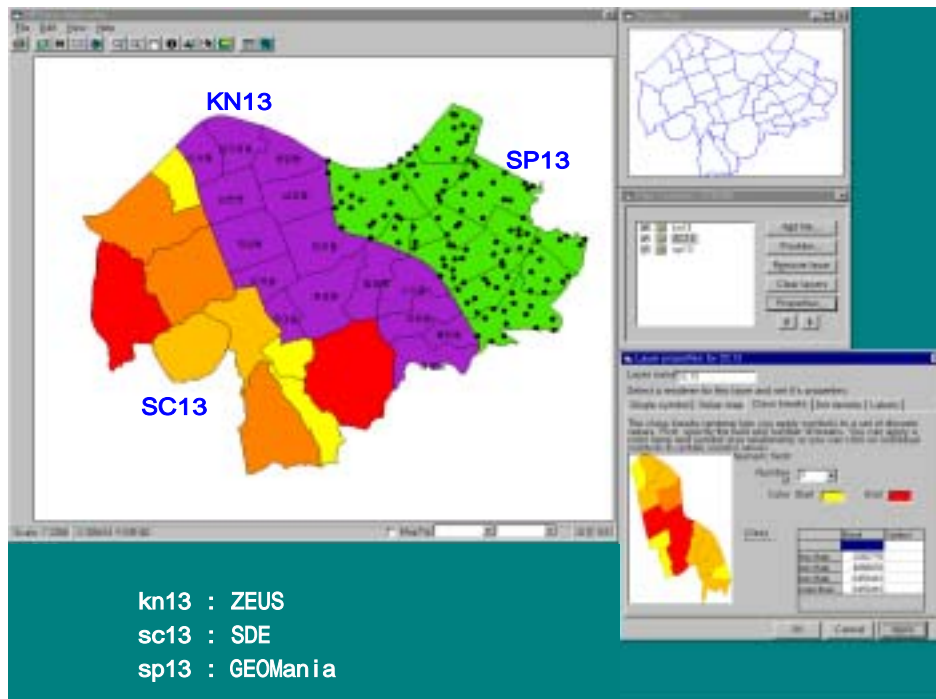
1. 핵심 공통 컴포넌트의 기본 수행 예



<그림 4-8> Visual Basic을 이용한 핵심 공통 컴포넌트의 기본 수행 예

위의 예제는 Visual Basic을 이용하여 구성되었으며, 위의 그림에서 보듯이 핵심 공통 컴포넌트의 MapDraw 컴포넌트와 IndexMap 컴포넌트가 사용되었음을 알 수 있다. 사실은 이외에도 화면에는 나타나지 않지만, GeoDAO 컴포넌트와 다양한 데이터 제공자 컴포넌트가 추가적으로 이용되었다. 본 예제는 데이터 제공자 컴포넌트를 이용하여 SDE와 ZEUS 그리고 GEOMania에 접속하여 공간 및 속성 정보를 읽어오고 이러한 정보를 단순히 화면에 표시하고 있는 결과를 보여주고 있다.

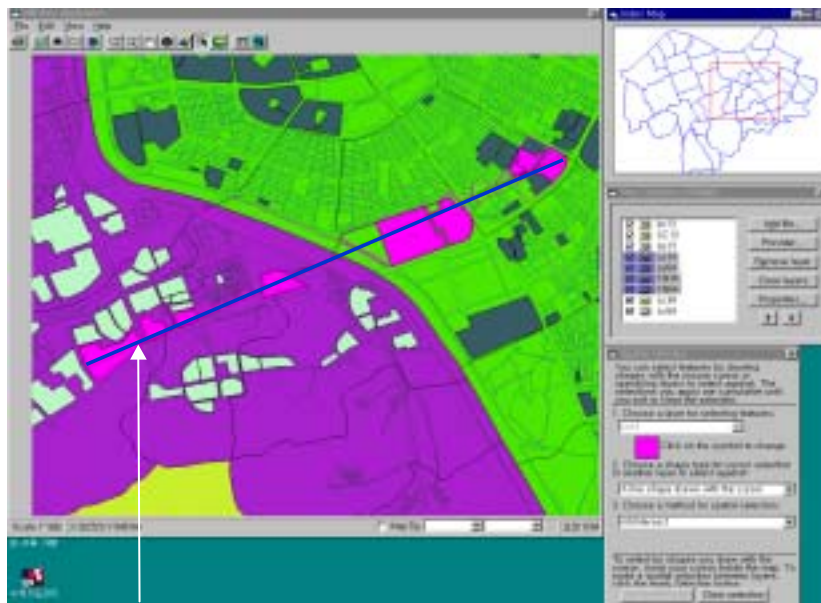
2. 핵심 공통 컴포넌트의 다양한 주제도 생성 예



<그림 4-9> 핵심 공통 컴포넌트의 주제도 수행 예

위의 예제에서는 MapDraw 컴포넌트의 다양한 Rendering 기능을 이용하여 주제도를 생성하고 있다. 실제로 ZEUS 데이터 제공자 컴포넌트로부터 읽어들인 kn13(강남구 동 경계선)은 동 이름을 출력하기 위한 LabelRender 기능을, SDE 데이터 제공자 컴포넌트로부터 읽어들인 sc13(서초구 동 경계선)은 동의 면적을 기준으로 하여 분류 기능을 수행하기 위한 ClassBreaksRender 기능을, 끝으로 GEOMania 데이터 제공자 컴포넌트로부터 읽어들인 sp13(송파구 동 경계선)은 동의 인구와 같은 속성을 기준으로 DotDensityRender 기능을 이용하고 있다.

3. 핵심 공통 컴포넌트의 다양한 공간 검색 수행 예



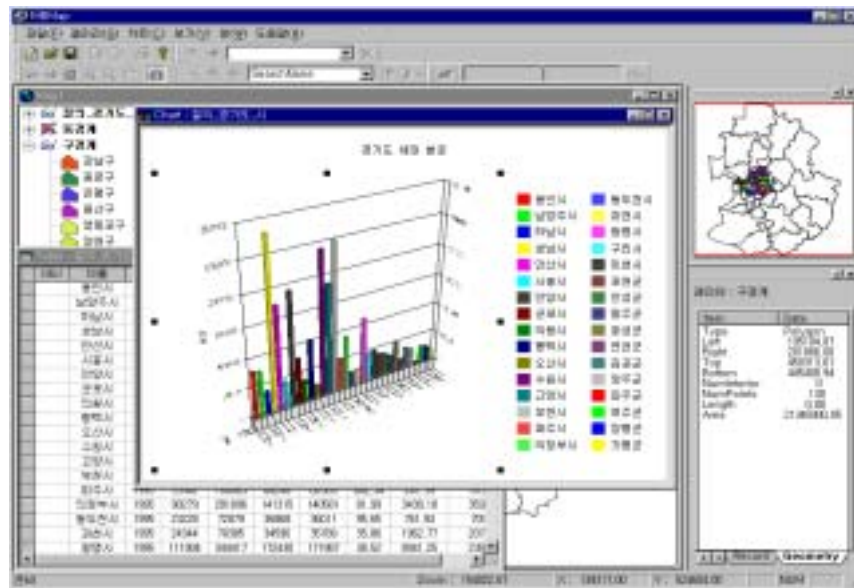
kn13:ZEUS
 sc13:SDE
 sp13:SQL SERVER
 sp34: SQL SERVER
 sp64: SQL SERVER
 kn34:SDE
 kn64:SDE
 sc34:GEOMania
 sc64:GEOMania

<그림 4-10> 핵심 공통 컴포넌트의 공간 검색 수행 예

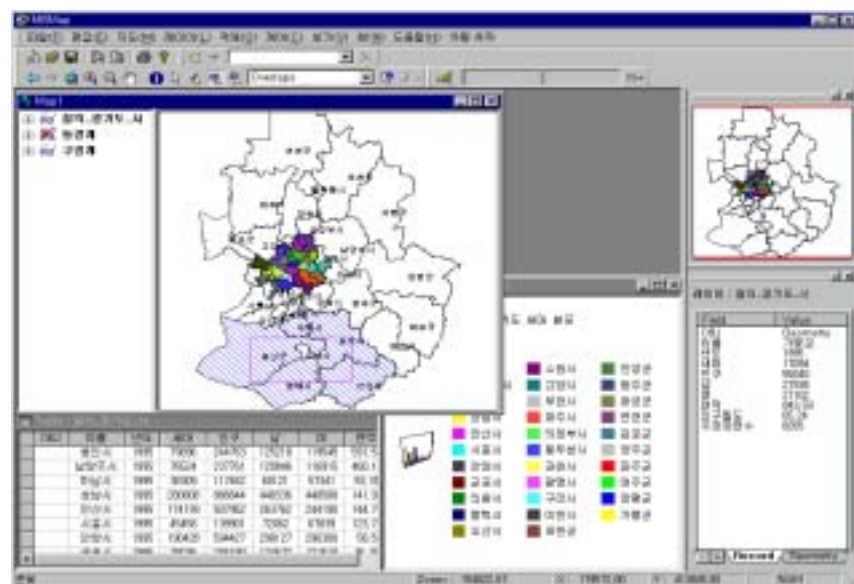
위의 예제에서는 SQL Server의 sp34, sp64 데이터와 SDE의 sp34, sp64 데이터에 대하여 LineString의 공간필터를 이용하여 Intersects 연산을 수행하고 있다. 이러한 공간 연산 기능은 3장에서 설명하였듯이 OGC용 데이터 제공자 컴포넌트의 기본 기능이다. 그러므로 위의 결과는 핵심 공통 컴포넌트들 중의 GeoDAO 컴포넌트가 공간 연산 작업을 SQL Server 데이터 제공자 컴포넌트와 SDE 데이터 제공자 컴포넌트에 요청하고, 공간 연산의 결과로서 반환되는 공간 데이터를 MapDraw 컴포넌트가 화면에 표시한 것이다. 물론 이와 같은 Intersects의 공간 연산 이외에도 핵심 공통 컴포넌트는 Within, Contains, Disjoint, Overlaps 등의 공간 연산도 모두 처리할 수 있어야 한다.

4. 핵심 공통 컴포넌트를 이용한 GIS 소프트웨어 둘 예제

본 절에서는 핵심 공통 컴포넌트와 데이터 제공자 컴포넌트를 이용하여 Visual C++로 구성된 GIS 소프트웨어 둘의 예제에 관하여 살펴보고자 한다. 우선 실제로 시스템의 수행 예제는 <그림 4-11> 그리고 <그림 4-12>와 같다.



<그림 4-11> MBMap의 Chart 기능 수행 예

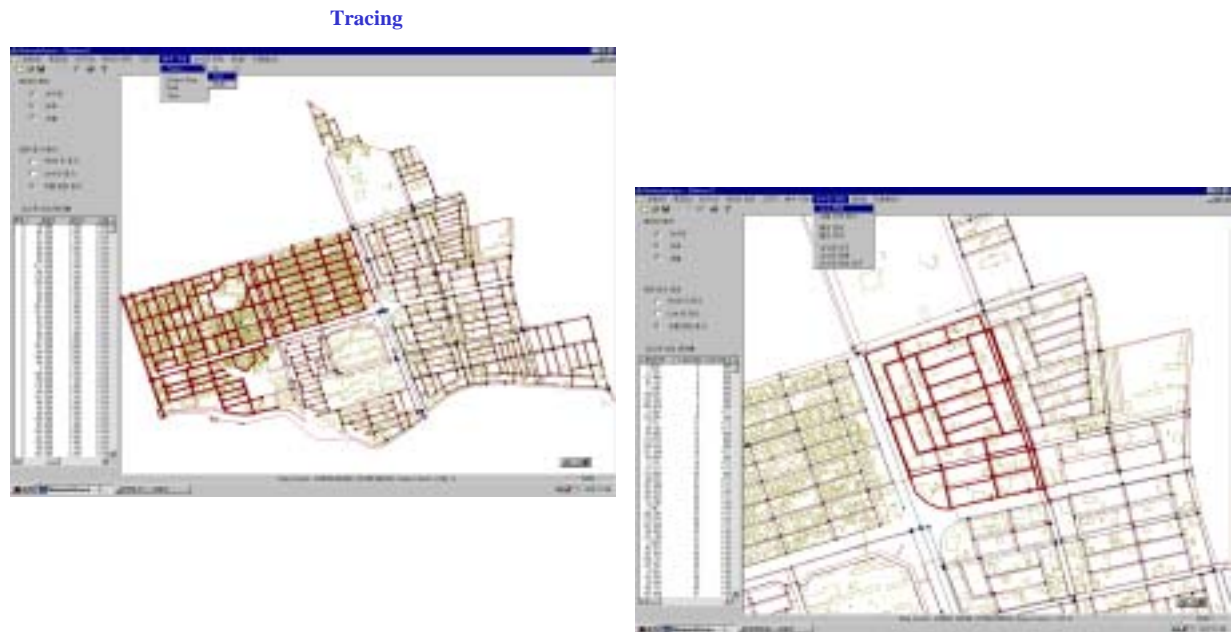


<그림 4-12> MBMap의 Overlaps 공간 연산 수행 예

위의 두 예제는 핵심 공통 컴포넌트와 데이터 제공자 컴포넌트의 인터페이스 기능들을 이용하여 VC++로 구성된 MBMap 소프트웨어의 일부 기능을 보여주고 있다. 특히 <그림 4-11>은 경기도에 존재하는 ‘도시’들에 대하여 세대 분포 정보를 이용하여 Chart 기능을 수행하고 있는 예제이며, <그림 4-12>는 Overlaps의 공간 분석을 수행한 예제이다. 위의 두 예제 이외에도 MBMap은 공간 분석의 결과를 편집 또는 저장할 수 있는 Report 기능도 제공하고 있다.

5. Network 분석 컴포넌트 수행 예제

끝으로 다음 <그림 4-13>은 핵심 공통 컴포넌트의 Network 분석 컴포넌트를 이용함으로써 상수도관 Tracing과 상수도 누수 관리 기능을 수행하는 예제를 보여주고 있다.



<그림 4-13> Network 분석 컴포넌트의 수행 예

제4절 결 론

본 장에서는 데이터 제공자 컴포넌트와 GIS 응용 시스템의 중간에 위치하여 GIS 응용 시스템이나 응용 컴포넌트를 구축하고자 할 때, 다양한 기능과 편리함을 제공할 수 있는 핵심 공통 컴포넌트의 기능과 구성 그리고 수행 예제에 대하여 살펴보았다. 그리고 핵심 공통 컴포넌트의 일부 컴포넌트들 - Geometry 컴포넌트와 Spatial Reference System 컴포넌트 - 은 OGC 의 OLE/COM 구현 사양에서 이미 표준 사양을 정의하고 있었으며, 그 밖의 컴포넌트들은 표준 사양이 현존하지 않는 관계로 기존에 가장 많이 사용되는 컴포넌트 기반 GIS 소프트웨어를 벤치마크로 하여 인터페이스 및 기능 사양을 구성하였다. 그리고 이러한 핵심 공통 컴포넌트도 역시 데이터 제공자 컴포넌트와 마찬가지로 마이크로소프트사의 비주얼 스튜디오와 ATL/COM 프로그램의 개발 방법을 이용하여 구축되었다.

끝으로 이러한 핵심 공통 컴포넌트는 어떠한 특징 및 장점을 가지는지 간략히 요약해보면 다음과 같다.

- OGC에서 제안하는 OLE/COM을 위한 국제 표준 사양을 준수하고 있다.
- 컴포넌트 기반으로 구성되어 재사용성과 상호운용성이 뛰어나고 사용하는 프로그램 언어에 관계없이 이용이 가능하다.
- 다양한 GIS 데이터베이스 접속 기능과 기능별 선택적인 컴포넌트의 사용으로 응용 시스템의 개발에 편리성을 더해준다.
- 컴포넌트 기반으로 분산 컴퓨팅 환경에서 기관별 공간 정보 유통을 간접적으로 지원할 수 있다.
- 핵심 공통 컴포넌트를 이용하여 MBMap과 유사한 GIS 소프트웨어들의 개발이 용이하다.

제5장 교통DB와 개방형 GIS 컴포넌트

제1절 교통DB 개요

제2절 GIS 정보의 공유 방안

제3절 교통DB의 정보 공유 및 유통 방안

제4절 결 론

제1절 교통DB 개요

1. 교통DB 개요

GIS 기반 교통DB 정보는 도면 정보와 속성 정보로 나누어져 있으며, 기본적인 GIS 도면 정보는 GIS에 종속된 데이터베이스를, 속성 정보는 GIS 데이터베이스와 독립적인 일반 데이터베이스를 이용하여 관리하고 있다. 속성 정보의 양이 적은 경우에는 GIS 데이터베이스가 속성 정보와 도면 정보를 동시에 관리할 수도 있다. 또한 데이터베이스는 이러한 기본 속성 및 공간 데이터 이외에, 이들 각 데이터 대한 이력을 관리하기 위한 메타 데이터를 관리하고 있다.

본 장에서는 이러한 GIS 데이터베이스 내에 저장되어 있는 교통망 DB와 같은 도면 정보가 어떠한 방식으로 저장되어 있는지 살펴보고, 다른 기관과 이들 정보들을 어떠한 방식으로 상호 공유하고 유통시킬 수 있는가에 대한 방안에 대하여 알아보하고자 한다.

2. 교통DB의 GIS 정보 저장 방식

결론부터 말하자면, 교통DB구축사업에서 GIS 관련 정보들은 ESRI사의 SDE 공간 엔진과 오라클 관계형 데이터베이스를 이용하여 관리되고 있다. 여기서 특히 SDE의 경우는 GIS 정보를 오라클 데이터베이스에 효율적으로 저장하고 관리하기 위하여 사용되는 GIS 시스템으로 다음과 같은 특징을 가지고 있다.

- 매우 방대한 양의 공간 정보를 지원할 수 있다.
- 공간 데이터를 쉽게 접근할 수 있다.
- 공간 색인 방법을 지원하므로 공간 검색 방법이 뛰어나다.
- 네트워크 환경에서 다중 사용자의 지원을 고려하여 설계되었다.
- 클라이언트-서버 환경을 지원한다.
- Unix와 NT 등의 다중 컴퓨터 플랫폼을 지원한다.
- C 언어와 같은 강력하고 유연한 어플리케이션 프로그래밍 인터페이스를 지원한다.

다음 2절과 3절에서는 이와 같이 SDE와 오라클 데이터베이스를 이용하여 저장되어 있는 교통DB의 정보들을 분산 환경에서 공유할 수 있는 방안에 대하여 살펴보고자 한다.

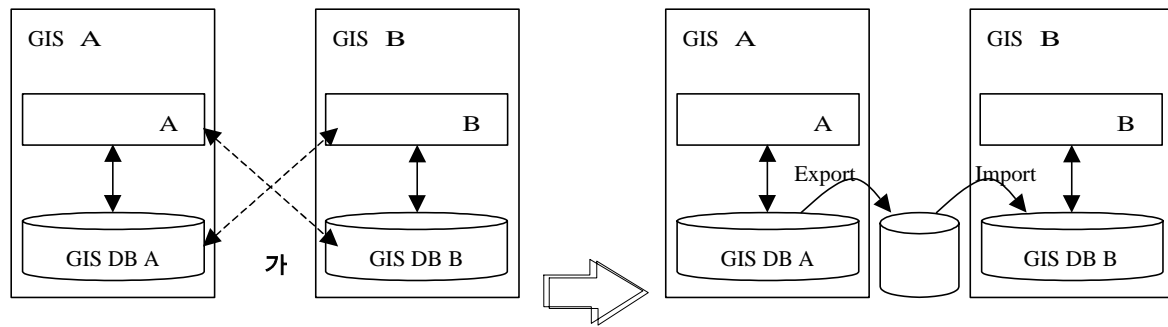
제2절 GIS 정보의 공유 방안

GIS의 발전으로 GIS 정보에 대한 수요가 늘어남에 따라 GIS 정보의 공급자들이 늘어나고 있다. 그리고 일반적으로 이러한 GIS 정보의 공급자들은 수요자의 요구에 맞는 형식을 사용하여 공간 데이터를 생산하고 있다. 그러므로, 이러한 GIS 정보들은 다양한 환경에서 다양한 형식으로 그리고 다양한 GIS 소프트웨어를 이용하여 저장되었고, GIS 정보의 수요자가 사용하는데 전혀 지장을 주지 않았다. 그러나, 최근 들어 GIS 수요자들이 분산 환경에서 다른 곳에 저장되어 있는 GIS 정보의 공유를 요구하게 되었다. 이러한 수요자들의 GIS 정보의 공유에 대한 커다란 요구는, 이 기종 분산 환경의 다양한 플랫폼에서 공간 정보를 상호 교환할 수 있고 서비스를 제공할 수 있는 공간 정보의 공유가 반드시 해결되어야 하는 문제임을 깨닫게 되었다. 그러므로 본 연구에서는 어떠한 방안을 이용하여 공간 정보를 상호 공유시키고 수요자들의 요구를 충족시킬 수 있는지에 대하여 살펴볼 것이다. 특히, 지금까지 주로 이용되어 왔던 공간 데이터의 상호 변환 방법을 통한 GIS 정보의 공동 활용 방법과 최근에 개방형 GIS 컴포넌트 기술 개발의 성과로 가능해진 공간 데이터의 직접 공유 방법을 통한 GIS 정보의 공동 활용 방법에 대해서 살펴보도록 한다.

1. 데이터의 변환을 통한 정보 공유 방안

가. 공간 데이터 변환의 개념

다양한 GIS 소프트웨어로 구성되어 있는 공간 정보들은 기본적으로 상호 공유가 불가능하다. 이는 GIS 소프트웨어간 공간 정보의 저장 방식이 다르며, 공간 정보를 접근하는 방식이 다르며, 공간 정보를 사용하는 환경이 다르기 때문이다. 그리하여 이러한 공간 정보의 상호 공유 문제를 해결하기 위하여 다양한 방법들이 모색되었으며, 가장 간단하고 쉬운 방법으로 공간 데이터의 포맷을 변환하는 방법이 나타나게 되었다. 공간 데이터의 포맷을 변환하는 방법은 다양한 GIS 데이터 저장 포맷으로 저장되어 있는 공간 정보를 상호 공유가 가능한 저장 포맷으로 변환함으로써 누구나 이 변환된 공간 데이터를 사용할 수 있도록 하는 방식이다. 다음 <그림 5-1>은 서로 다른 GIS 소프트웨어가 상호 공간 정보를 공유할 수 없는 단계에서 상호 공유가 가능한 포맷으로 공간 데이터를 변환함으로써 공간 정보의 공유가 가능해진다는 개념을 보여주고 있다.



<그림 5-1> 데이터 변환을 통한 공간 정보의 상호 공유 개념

<그림 5-1>의 왼쪽 그림을 보면 GIS 소프트웨어 B는 GIS 소프트웨어 A의 공간 데이터베이스를 접근할 수가 없으며, 그 반대의 경우도 마찬가지이다. 그러나 오른쪽 그림의 경우를 보면 GIS 소프트웨어 A의 데이터베이스를 공유가 가능한 공통 포맷으로 Export하고 GIS 소프트웨어 B는 공통포맷의 데이터를 자신의 데이터베이스 B로 Import시킴으로써, GIS 소프트웨어 B가 GIS A의 공간 데이터베이스를 접근할 수 있도록 하고 있다.

나. 공간 데이터 변환의 공통 포맷

그러면, GIS 정보를 상호 공유하기 위하여 이용될 수 있는 공통 포맷으로는 어떠한 포맷이 주로 이용되는가? 현재 국내의 경우에는 공간 데이터의 교환 포맷으로 DXF, Shape 또는 SDTS(Spatial Data Transfer Standard)의 방법이 주로 이용되고 있다. 특히, SDTS의 경우는 미국에서 공간 정보의 변화 및 교환 표준을 위하여 개발된 포맷으로 위의 공통 포맷으로 바로 이용할 수 있는 장점이 있다. 그러나 SDTS는 광범위한 공간 정보 형식들의 공유를 지원하고 수많은 공간 정보 모델들을 모두 포괄하기 위하여 복잡한 구성을 가지고 있어 실제 구현에 있어서 어려운 단점이 있다. DXF는 캐드에서 주로 사용되던 포맷으로 내부 구성이 공개되어 있으며 구현에 있어서도 용이한 장점이 있으나, 공간 정보가 아닌 속성 정보는 저장할 수 없다는 커다란 단점을 가지고 있다. Shape은 ESRI에서 사용하는 GIS 데이터의 저장 포맷으로 일반적으로 Shape과 DBF의 파일을 동시에 이용하고 있다. 그리고 Shape 파일은 공간 데이터를, DBF 파일은 속성 데이터를 저장하는 방식으로, 물론 내부 구성은 모두 공개되어 있으므로 누구나 손쉽게 사용할 수 있는 장점이 있다. 이와 같이 공간 정보와 속성 정보를 동시에 저장할 수 있고, 내부 구성이 완전히 공개되어 있고, 누구나 손쉽게 이용할 수 있는 특징으로 인하여, 이 Shape 방식은 국내에서 데이터의 교환 포맷으로 주로 이용되고 있다. 또한 대부분의 새로이 개발되는 많은 국산 또는 외산의 GIS 소프트웨어들 역시 이러한 Shpae 포맷을 Import 또는 Export하기 위한

기능을 자체에 내장하고 있음을 알 수 있다.

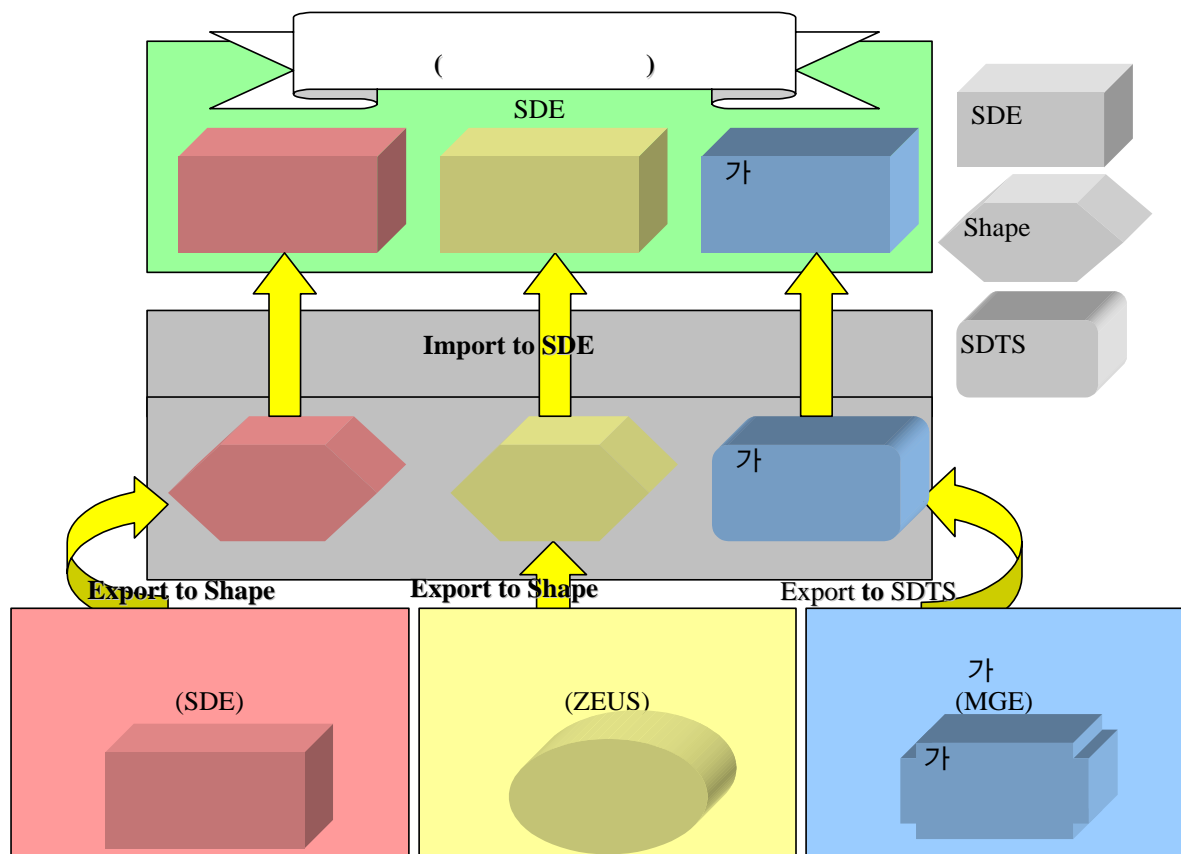
다. 공간 데이터 변환 방법의 특징

공간 데이터를 변환하여 공간 정보를 유통시키는 방법은 다음과 같은 장단점을 가지고 있다.

- 공간 데이터의 변환 과정을 거쳐야 한다.
 - 공간 데이터베이스에서 공통 포맷 파일로의 Export 및 공통 파일 포맷에서 공간 데이터베이스로의 Import의 과정을 거쳐야 한다.
 - 대부분 On-Line 상이 아닌 Off-Line 상에서 공간 데이터의 변환이 진행된다.
- 공간 데이터의 중복 저장이 일어난다.
 - 다른 데이터베이스에 저장되어 있는 공간 정보를 사용하기 위해서는 Import 과정을 통하여 자신의 데이터베이스에 공간 정보의 복사 본을 생성해야만 한다.
- 최신의 데이터를 반영하기가 어렵다.
 - 데이터의 공유 작업은 Off-Line으로 일정한 기간을 두고 발생하므로, 매 순간 발생하는 데이터의 최신 변경 내용을 직접 공유할 수 없다.
 - 사용자가 원하는 지역의 데이터만을 추출할 수가 없다.
- 대부분의 GIS 소프트웨어들이 공통 포맷을 지원하는 경우가 많다.
 - 현존하는 많은 GIS 소프트웨어들이 기본적으로 Shape 포맷에 대한 Import/Export 기능을 가지고 있으므로, 특별한 부가의 노력 없이 바로 공간 정보의 상호 공유가 가능하다.

라. 공간 데이터 변환 방법을 이용한 정보 공유 예

<그림 5-2>는 서울시의 상수도 정보, 한국통신의 통신망 관로 정보, 그리고 한국가스공사의 가스망 관로 정보를 공간 데이터의 변환 방법을 적용하여 공유하고자 할 때 설계될 수 있는 개념도를 나타낸다.



<그림 5-2> 공간 데이터의 변환 방법을 이용한 공간 정보 상호 공유의 예

위의 그림에서 보면 서울시 상수도 데이터는 ESRI의 SDE 공간 엔진을 이용하고 있으며, 한국통신의 통신망 관로 데이터는 한국통신데이터의 ZEUS 공간 데이터베이스를 이용하고 있으며, 그리고 한국가스공사의 가스망 관로 데이터는 Integrgraph의 MGE 소프트웨어를 이용하고 있다. 이러한 상황에서 SDE를 사용하고 있는 사용자가 서울시, 한국통신 그리고 한국가스공사의 데이터를 상호 공유하기를 원한다면 다음과 같은 과정을 거쳐야 한다.

- SDE로 저장되어 있는 서울시의 상수도 데이터를 Shape 포맷으로 Export한다.
 - SDE 공간 엔진은 SDE-to-Shape와 Shape-to-SDE 포맷 변화를 자체적으로 지원하고 있다.
- ZEUS로 저장되어 있는 한국통신의 통신망 관로 데이터를 Shape 포맷으로 Export한다.
 - ZEUS 공간 데이터베이스는 ZEUS-to-Shape와 ZEUS-to-SDE 포맷 변화를 자체적으로 지원하고 있다.

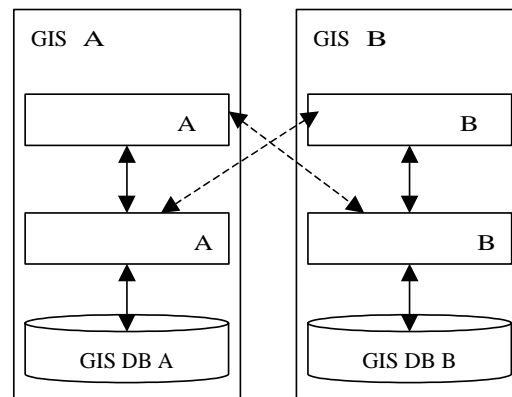
- MGE로 저장되어 있는 한국가스공사의 가스망 관로 데이터를 SDTS 포맷으로 Export 한다.
- 각 SDE, ZEUS, MGE로부터 Export된 Shape 및 SDTS 데이터 포맷을 클라이언트 사용자 시스템의 SDE로 Import 한다.

위의 예제와 같이 다른 기관에 저장되어 있는 공간 정보들을 상호 공유하기 위해서는 Off-Line 상에서 공간 데이터의 변환 과정을 두 번씩 거쳐야 하며, 또한 공간 데이터 역시 중복 저장되어야 함을 알 수 있다. 이에 다음에서는 이러한 Off-Line 방식이 아닌 그리고 공간 데이터의 중복 저장이 아닌 On-Line 상에서의 공간 데이터의 상호 공유 방식에 대하여 살펴보고자 한다.

2. 데이터의 직접 공유를 통한 정보 공유 방안

가. 공간 데이터 직접 공유의 개념

공간 정보를 공유하기 위한 방법으로 더욱 발전된 방식은 공간 데이터의 포맷을 교환이 가능한 공통 포맷으로 변환하는 방식보다 다른 곳에 저장되어 있는 공간 정보를 직접 접근하여 공유하는 방식이다. 그러나 이 기종의 GIS 소프트웨어, 분산 컴퓨팅 환경 그리고 다양한 GIS 데이터베이스 구조로 이루어진 공간정보 들을 직접 접근하여 상호 공유하는 것은 기본적으로는 불가능하다. 예를 들어, SDE를 사용하는 클라이언트는 ZEUS나 GEOMania를 접속할 수 없으며, 마찬가지로 ZEUS를 사용하는 클라이언트는 SDE나 GEOMania를 접속할 수 없다. 그러므로 이와 같이 다양한 GIS 소프트웨어에 저장되어 있는 공간 데이터를 데이터 포맷의 변환 없이 직접 접근하기 위해서는, GIS 소프트웨어에 독립적인 어떠한 공통된 또는 동일한 방법의 공간 데이터 접근 방법이 필요하게 되었다. 이를 위하여 OGC에서는 OLE/COM, CORBA 그리고 SQL을 위한 구현 사양을 발표하게 되었다. 특히, OLE/COM 사양의 데이터 제공자 컴포넌트는 바로 위에서 설명한 GIS 소프트웨어나 GIS 데이터 포맷에 관계없이 동일한 방식으로 공간 데이터를 직접 접근할 수 있는 방법을 제시하고 있다. 더군다나, 데이터 제공자 컴포넌트는 공간 데이터의 공유뿐만 아니라, 분산 환경에서 공간 연산자도 상호 공유시킴으로써 상호운용성을 제공하고 있다. 이러한 데이터 제공자 컴포넌트를 이용하여 공간 정보를 상호 공유하기 위한 개념도는 <그림 5-3>과 같다.



<그림 5-3> 공간 데이터의 직접 접근을 통한 공간 정보 상호 공유의 개념

위의 <그림 5-3>에서 보면 GIS 클라이언트 A는 데이터 제공자 컴포넌트 B를 통하여 B의 공간 데이터베이스를 접근할 수 있으며, 이와 반대로 클라이언트 B는 데이터 제공자 컴포넌트 A를 이용하여 A의 공간 데이터베이스를 접근할 수 있음을 알 수 있다. 이는 데이터 제공자 A와 B의 하부 구조를 이루는 데이터베이스나 공간 엔진은 다르지만, 3장에서 설명하였듯이 데이터 제공자 컴포넌트의 인터페이스와 메소드가 표준화되고 통일되어 누구나 동일한 방식으로 사용할 수 있기 때문이다. 또한 클라이언트 A와 B는 데이터 제공자 컴포넌트를 사용하기 위한 소프트웨어로 클라이언트 개발자가 직접 작성하던가 또는 4장에서 설명한 핵심 공통 컴포넌트를 이용해도 된다. 그러나 이미 데이터 제공자 컴포넌트를 접근하기 위한 GeoDAO 컴포넌트, OLE/COM 사양의 Geometry와 Spatial Reference System 컴포넌트, 그리고 화면 도시 및 각종 렌더링 기능을 제공하기 위한 MapDraw 컴포넌트로 구성되어 있는 핵심 공통 컴포넌트를 이용하는 것이 당연히 소모되는 시간과 노력에 비해볼 때 훨씬 효율적임을 알 수 있다.

결론적으로 다양한 GIS 소프트웨어와 다양한 환경에 저장되어 있는 공간 데이터를 직접 접근함으로써 공간 정보를 상호 공유하기 위해서 현존하는 가장 좋은 해결책은 개방형 GIS 컴포넌트의 데이터 제공자 컴포넌트와 핵심 공통 컴포넌트를 이용하는 것이라고 할 수 있다.

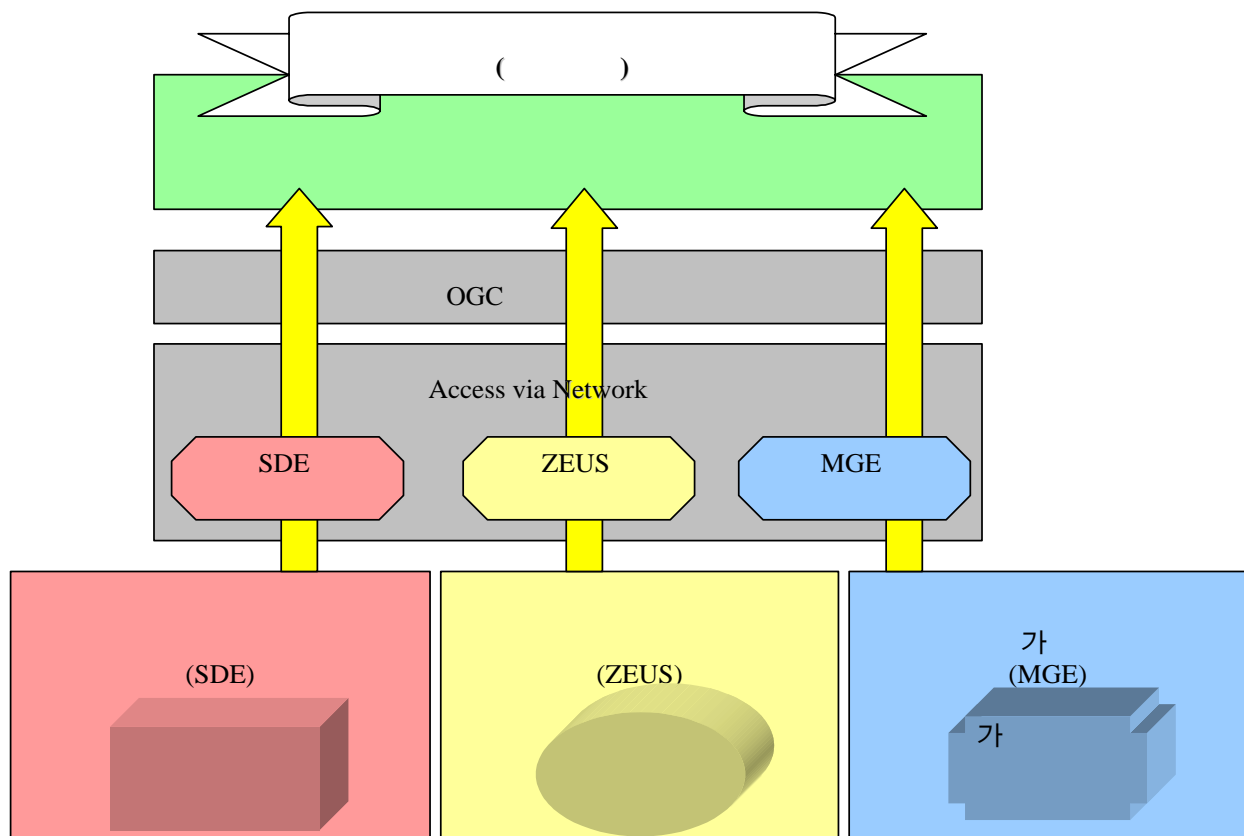
나. 공간 데이터 직접 공유 방법의 특징

공간 데이터를 공통 포맷으로 변환하지 않고 직접 접근하여 공유하는 방법은 다음과 같은 특징들을 가지고 있다.

- 공간 데이터의 변환 과정이 필요하다.
 - 공간 정보를 외부 파일 포맷으로 Export 또는 Import할 필요가 없으며, 단순히 공간 데이터를 OGC에서 제시하는 WKB 형식으로 변환할 필요가 있다. WKB 형식으로 파일 포맷으로 Export되는 것이 아니라 이터드이 쿼리 데이터로 다수처 저수처 기 위한 형식이다.
- On-Line 상에서 네트워크를 통하여 다양한 GIS 데이터베이스에 접속하여 사용자가 원하는 공간 데이터를 접근할 수 있다.
- 공간 데이터의 중복 저장이 일어나지 않는다.
 - 다른 GIS 데이터베이스에 저장되어 있는 공간 정보를 직접 접근하여 사용하므로 사용자는 자신의 데이터베이스에 사용하고자 하는 공간 정보의 복사 본을 저장할 필요가 없다.
- 항상 최신의 그리고 필요한 데이터만을 접근한다.
 - 데이터의 공유 작업이 사용자가 원하는 시점에 On-Line으로 발생하므로 매 순간 변경되는 최신의 데이터를 공유할 수 있다.
 - 전체 GIS 데이터베이스에서 사용자가 원하는 지역과 원하는 레이어의 공간 정보만을 추출하여 접근할 수 있다.
- 대부분의 GIS 소프트웨어들이 OGC 사양을 지원하지 않는다.
 - 현존하는 많은 GIS 소프트웨어들은 OGC 사양의 데이터 제공자 컴포넌트를 포함하고 있지 않다. 그러므로 사용자가 공유하기를 원하는 GIS 소프트웨어 또는 파일 포맷이 새로이 발생하면 우선 데이터 제공자 컴포넌트를 반드시 구현해야 한다.
 - 또한 GIS 소프트웨어들은 OGC용 데이터 제공자 컴포넌트를 직접 접근하기 위한 클라이언트 툴을 가지고 있지 않다. 그러므로 사용자가 이를 직접 작성하거나 핵심 공통 컴포넌트를 이용하여야 한다.

다. 공간 데이터의 직접 공유 방법을 이용한 정보 공유 예

<그림 5-4>는 앞에서의 예와 동일한 서울시의 상수도 정보, 한국통신의 통신망 관로 정보, 그리고 한국가스공사의 가스망 관로 정보에 대하여 공간 데이터의 직접 공유 방법을 적용하였을 때의 변화된 개념도를 나타낸다.



<그림 5-4> 공간 데이터의 직접 공유 방법을 이용한 공간 정보 상호 공유의 예

위의 그림에서의 공간 정보를 상호 공유하기 위한 과정은 앞의 <그림 5-3>에서 설명한 과정과는 많은 다른 면을 보여준다.

- 핵심 공통 컴포넌트로 구성되어 있는 클라이언트는 SDE 데이터 제공자 컴포넌트를 통하여 서울시 상수도 공간 데이터에서 원하는 부분만을 읽어들인다.
- 핵심 공통 컴포넌트로 구성되어 있는 클라이언트는 ZEUS 데이터 제공자 컴포넌트를 통하여 한국통신 통신망 관로 공간 데이터에서 원하는 부분만을 읽어들인다.
- 핵심 공통 컴포넌트로 구성되어 있는 클라이언트는 MGE 데이터 제공자 컴포넌트를 통하여 한국가스공사 가스관 관로 공간 데이터에서 원하는 부분만을 읽어들인다.
- 클라이언트가 원하는 작업들을 수행한다.

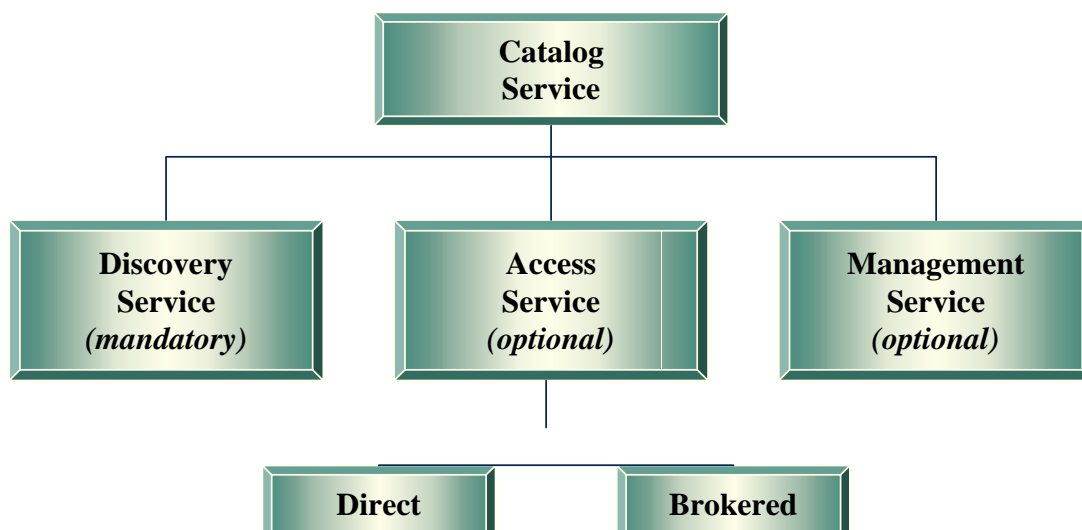
위의 예제에서 보듯이 다른 기관에 저장되어 있는 공간 정보들을 공유하기 위해서 핵심 공통 컴포넌트는 단순히 각 GIS 소프트웨어에 알맞은 데이터 제공자 컴포넌트만을 접근

함으로써 자신이 원하는 공간 데이터를 공유할 수 있고 더군다나 심지어는 공간 연산자도 처리할 수 있음을 알 수 있었다.

지금까지 우리는 개방형 GIS 컴포넌트들을 이용함으로써 공간 정보의 효율적인 공유 방법에 대하여만 살펴보았다. 그러면 이러한 서울시의 상수도 정보가 또는 통신망 관리 정보가 네트워크 환경에서 어느 기관의 어느 서버 시스템에 존재하는지는 어떻게 알 수 있을까? 지금까지 우리는 이러한 정보들이 어떤 서버 시스템에 존재하는지를 정확히 알고 있다는 가정 하에서 공간 정보의 공유에만 신경을 써왔다. 그러므로 다음 절에서는 이 기종 분산 환경에서 어떠한 공간 정보가 어디에 위치하는지를 검색하는 방법 그리고 그 후에 공간 정보를 상호 공유할 수 있는 방법을 포함하는 개념, 다시 말해서 공간 정보의 유통에 관하여 살펴볼 것이다.

3. 공간 정보의 유통

공간 정보 유통은 이 기종 시스템 및 플랫폼에서 공간 데이터를 상호 공유하고 서비스를 제공할 수 있는 공간 데이터의 공유 개념에 추가적으로 어떠한 공간 정보가 어디에 위치하고 있는지의 메타 정보 검색 개념을 포괄하고 있다. OGC에서는 이러한 공간 정보의 위치 및 내용을 관하 메타 정보를 관리하고 검색할 수 있는 방법을 제공하기 위하여 "Catalog Interface Implementation Specification"을 제공하고 있다. <그림 5-5>는 OGC Catalog Service 구현 사양에서 구현해야할 메인 기능들을 도시하고 있다.



<그림 5-5> OGC의 Catalog Service 구현 사양

<그림 5-5>에서 보면 정보 유통을 위한 Catalog Service 구현 사양은 Discovery Service와 Access Service 그리고 Management Service 구현되어 있음을 알 수 있다. 각각의 기능 및 역할은 다음과 같다.

- Discovery Service

- Catalog Service의 필수 구현 사양이다.
- 사용자가 원하는 공간 데이터 집합이 어느 위치에 존재하는지 검색하고 그 검색 결과(메타 데이터)를 전송 받는다.

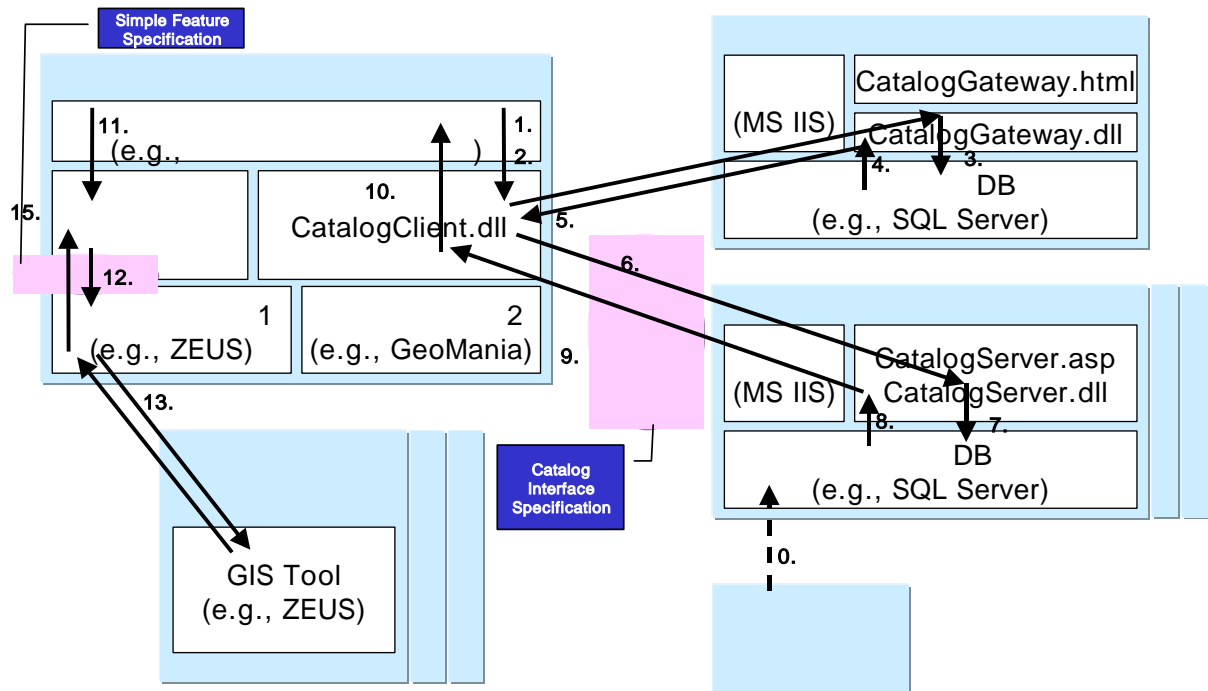
- Access Service

- Catalog Service의 선택 구현 사양이다.
- Discovery Service에서 결과로서 전송 받은 공간 데이터 집합에 대한 메타 데이터를 이용하여 공간 데이터가 위치한 서버에 접속하여 공간 데이터를 요구한다.
- 개방형 GIS 컴포넌트에서는 다양한 GIS 서버에 대한 공간 데이터 접근을 지원하기 위한 Access Service를 위하여 앞에서 설명하였던 OGC용 데이터 제공자 컴포넌트와 핵심 공통 컴포넌트를 이용하고 있다. 여기서 OGC용 데이터 제공자 컴포넌트는 공간 데이터의 접근을 수행하고 핵심 공통 컴포넌트는 GIS 관련 핵심 기능들을 수행한다.

- Management Service

- Catalog Service의 선택 구현 사양이다.
- 본 서비스는 새로운 공간 데이터 집합이 생성되거나 또는 기존의 특정 공간 데이터 집합이 소멸되거나 할 경우에 공간 데이터 집합에 대한 메타 정보를 관리하기 위한 기능을 수행한다. 다시 말하면 새로운 메타 정보를 추가하거나 또는 기존의 메타 정보를 삭제하거나 하는 등의 메타 정보 편집 기능을 수행하는 것이 주 기능이다.

이와 같이 OGC의 Catalog Service 구현 사양과 OLE/COM 구현 사양을 이용하여 공간 정보 유통을 수행하고자 할 때 구성될 수 있는 공간 정보 유통의 체계 구성도는 <그림 5-6>과 같다.



<그림 5-6> 공간 정보 유통 체계 구성도

위의 그림을 보면 공간 정보의 유통 체계는 사용자 시스템인 기관 응용 시스템, 유통 게이트 웨이, 유통 노드, 공간 데이터 서버 그리고 메타 데이터 입력기로 구성되어 있다. 이들 각각의 역할 및 기능은 다음과 같다.

가. 기관 응용 시스템

- 최종 사용자가 사용하는 클라이언트 시스템으로서 <그림 5-6>에서는 도로 굴착의 응용 업무를 예제로 들고 있다.
- 기관 응용 시스템을 자세히 살펴보면 도로 굴착 업무를 수행하기 위한 응용 시스템, 응용 시스템의 기능을 지원하기 위한 핵심 공통 컴포넌트, 공간 데이터 서버에 접속하기 위한 데이터 제공자 컴포넌트, 그리고 메타 데이터를 검색하고 결과를 얻어오는 정보 유통을 위한 클라이언트 컴포넌트로 구성되어 있다.
- 데이터 제공자 컴포넌트로는 응용 시스템이 접속하고자 하는 공간 데이터 서버에 따라서 다양한 GIS 소프트웨어를 위한 데이터 제공자 컴포넌트가 설치될 수 있다. 위의 예에서는 ZEUS와 GEOMania의 데이터 제공자 컴포넌트를 포함하고 있다.

- 단위 시스템

- 다중 서버 연결 및 정보 획득 : 데이터 제공자 컴포넌트를 이용한 다중 공간 데이터 서버 접속 기능 구현
- 공간 데이터 View 및 Analysis : 핵심 공통 컴포넌트를 이용한 매핑 기능, 렌더링 기능, 레이어 관리 기능 그리고 ActiveX 컨트롤을 이용한 웹 환경 지원 기능 구현
- 메타 정보 획득 기능 : 공간 메타 정보 검색 및 획득 기능

나. 공간 데이터 서버

- 공간 데이터 서버는 공간 정보를 저장하기 위한 어떠한 GIS 소프트웨어 또는 GIS 데이터베이스가 이용될 수 있다.
- 물론, 기관 응용 시스템에서는 다양한 GIS 소프트웨어 또는 GIS 데이터베이스에 접근하기 위한 데이터 제공자 컴포넌트를 포함하고 있어야 한다.
- 단위 시스템
 - 사용하고자 하는 GIS 소프트웨어나 GIS 엔진 또는 공간 데이터베이스 이용된다.

다. 유통 노드

- 유통 노드는 우선 공간 데이터 집합에 대한 메타 정보를 관리하기 위한 메타 데이터를 위한 데이터베이스 서버를 가지고 있으며, 이 메타 정보를 검색하기 위한 정보 유통 컴포넌트, 그리고 유통 노드에 접속하기 위한 웹 서버로 구성되어 있다.
- 특히, 메타 데이터를 위한 데이터베이스 서버는 정보 유통 컴포넌트를 통한 메타 정보 검색뿐만 아니라 메타 데이터 입력기를 통하여 새로운 메타 정보를 추가하거나 또는 기존의 메타 정보를 수정 또는 삭제할 수 있다.
- 단위 시스템
 - 메타 데이터 검색 및 전송 기능 : 정보 유통 컴포넌트가 수행하며, 사용자의 메타 정보 검색을 위한 질의문을 처리한다.
 - 유통 노드 관리 기능 : 사용자 권한 관리 기능 등을 수행한다.
 - 웹 페이지 구축 : 유통 노드를 접근하기 위한 홈 페이지를 구축한다.

라. 유통 게이트웨이

- 유통 게이트웨이는 다양한 공간 데이터 집합에 대한 메타 정보 리스트를 관리하고 있는 유통 노드 리스트 정보를 관리하고 있다. 다시 말하면 유통 게이트웨이는 많은 수

의 유동 노드 목록 정보들을 유지 및 관리하는 역할을 수행하는 것이다.

- 유동 게이트웨이는 사용자가 접근하기 위한 웹 서버, 유동 노드 목록 정보를 관리하기 위한 데이터베이스, 그리고 유동 노드 목록 정보를 검색하기 위한 정보 유동 컴포넌트로 구성된다.
- 단위 시스템
 - 유동 게이트웨이 웹 서버 : 사용자 인증 관리, 유동 노드 등록, 수정, 삭제, 조회 서비스 양식 및 결과 양식 제공
 - 유동 노드 목록 데이터베이스 : 유동 노드들의 목록 정보를 데이터베이스에 저장
 - 유동 컴포넌트 : 유동 게이트웨이의 웹 서버로부터 주어지는 사용자의 유동 노드에 관한 작업을 처리하여 결과를 전송

지금까지 응용 시스템, 유동 노드, 유동 게이트웨이, 그리고 공간 데이터 서버로 구성된 정보 유동 체계에서 각각의 구성 및 역할에 대하여 살펴보았다. 그러면 사용자가 요구하는 공간 데이터가 위와 같은 정보 유동 체계에서 어떠한 흐름에 의하여 처리될 수 있는지 <그림 5-6>을 보면서 살펴보도록 하자.

- 응용 시스템이 클라이언트의 정보 유동 컴포넌트(CatalogClient.dll)를 이용하여 유동 게이트웨이 유동 노드 목록을 요청(1, 2)한다.
- 사용자의 요구를 받은 유동 게이트웨이는 자신의 정보 유동 컴포넌트(CatalogGateway.dll)을 이용하여 유동 노드 목록을 검색(3)하고 검색 결과를 반환(4)한다.
- 유동 노드 목록 결과를 전송 받은 응용 시스템의 정보 유동 컴포넌트(5)는 유동 노드들의 정보 유동 컴포넌트(CatalogServer.dll)에게 사용자가 원하는 공간 데이터 집합에 대한 질의(6)를 통하여 공간 데이터 집합의 검색(7)을 지시한다.
- 공간 정보 집합 메타 데이터 데이터베이스로부터 검색된 메타 데이터 결과(8)는 다시 응용 시스템의 정보 유동 컴포넌트로 전송(9)된다. 그리고 그 결과는 다시 응용 시스템으로 전송(10)된다.
- 이제 응용 시스템은 공간 정보 집합에 관한 메타 정보로부터 자신이 원하는 데이터가 어느 위치에 어떠한 형식으로 저장되어 있는지 알 수 있다. 그러므로 응용 컴포넌트는 핵심 공통 컴포넌트를 통하여(11) 공간 질의 작업을 해당되는 데이터 제공자 컴포넌트에게 요청(12)한다.
- 공간 질의 처리 요청을 받은 데이터 제공자 컴포넌트는 공간 질의를 공간 데이터 서

버에 보내어 처리(13)하게 하고 그 결과의 공간 데이터를 데이터 제공자 컴포넌트를 통하여(14) 핵심 공통 컴포넌트에 전송(15)한다.

- 그러나 제시 교통 컴포넌트에 전송된 공간 데이터로 응용 시스템이 원하는 목적에 맞게 가공하여 작업을 수행함으로써 공간 정보의 유통을 통한 작업이 종료된다.

그러면 왜 이와 같이 복잡한 공간 정보의 유통 작업이 필요한가? 첫째, 대용량의 GIS 정보를 네트워크 상에서 효율적으로 공유함으로써, 기 구축된 공간 정보의 공동 활용을 높이고 국내 GIS 구축비용의 획기적인 절감을 꾀할 수 있다. 둘째, 향후에 국가 공간 기반(MCIV) 구축 및 공간 정보 오픈 하우스(Closing House) 서비스 인화 호선이 될 수 있다. 셋째, 표준화된 개방형 GIS 컴포넌트 기술 기반의 공간 정보 유통 체계를 확산할 수 있다. 넷째, 이러한 공간 정보 유통 컴포넌트 기술은 분산 환경에서 COM 또는 CORBA 기반뿐만 아니라 웹을 이용하는 인터넷 서비스 환경에서도 가능하므로, 일반인들을 대상으로 다양한 지형도, 주제도, 지적도 등과 같은 다양한 정부부처, 지방자치단체 등의 공간 정보들을 자유로이 서비스하기 위한 기술로 활용될 수도 있다.

제3절 교통DB의 정보 공유 및 유통 방안

1. 교통DB 정보의 상호 공유 방안

- 교통DB의 정보 공유를 위하여 개방형 GIS 컴포넌트 기술 기반의 유통 방안

분산 환경, 다양한 GIS 소프트웨어와 다양한 GIS 파일 포맷, 그리고 다양한 공간 데이터베이스의 환경에서 자신의 공간 정보를 공유하기 위해서 가장 쉽게 접근할 수 있는 방법은 자신의 공간 정보를 공유가 가능한 표준화된 포맷으로 데이터를 변환시키는 방법이다. 현재 국내에서 공간 데이터의 교환을 위하여 가장 많이 이용되는 형식은 SDTS와 Shape의 방식이 있다. 그러므로 교통DB를 타 기관 또는 다른 사용자와 공유하기 위해서 가장 가능한 방법은 교통DB를 SDTS 또는 Shape의 파일 형식으로 변환하여 공유하는 것이다. 다음은 교통DB를 SDTS와 Shape 저장 형식으로 변환하여 공유하고자 할 때의 장점 및 단점을 보여준다.

- SDTS 저장 형식을 이용한 교통DB의 공유
 - 대부분 On-Line 방식이 아닌 Off-Line 방식으로 공간 데이터를 공유하게 된다.
 - Oracle과 SDE 공간 엔진을 이용하여 구축된 교통 공간 정보를 SDTS로 변환하기

위해서 부가적인 응용 프로그램을 개발해야 하고, 이와 반대로 SDTS로 구성되어 있는 공간 정보를 SDE의 형식으로 변환하기 위한 프로그램도 부가적으로 개발해야 하므로 별도의 시간을 요구하게 된다.

- 다만, Arc SDE 8.0 버전 이상에서는 Arc/Info와 SDE가 상호 통합되었으며, Arc/Info가 SDTS 형식을 지원하므로, SDE 형식의 교통DB를 SDTS로 변환하기 위해서 요구되는 노력과 시간이 감소할 수도 있다.
- SDTS 변환 프로그램을 직접 작성하기 위해서는 SDTS 변환 프로그램을 위해 제공되는 기본 라이브러리를 이용할 수 있으므로 비교적 용이하게 개발이 가능하나, 제공되는 라이브러리를 숙지하는 별도의 노력이 요구된다.

- Shape 저장 형식을 이용한 교통DB의 공유

- 대부분 On-Line 방식보다는 Off-Line 방식으로 공간 데이터를 공유하게 된다.
- SDE 는 자체 기능으로서 Shape 형식의 공간 정보를 Import 또는 Export 할 수 있는 기능을 포함하고 있다. 그러므로 교통DB의 정보를 Shape 형식으로 변환하기 위한 프로그램을 개발하기 위하여 별도의 시간이나 노력이 필요하지 않는 장점이 있다.

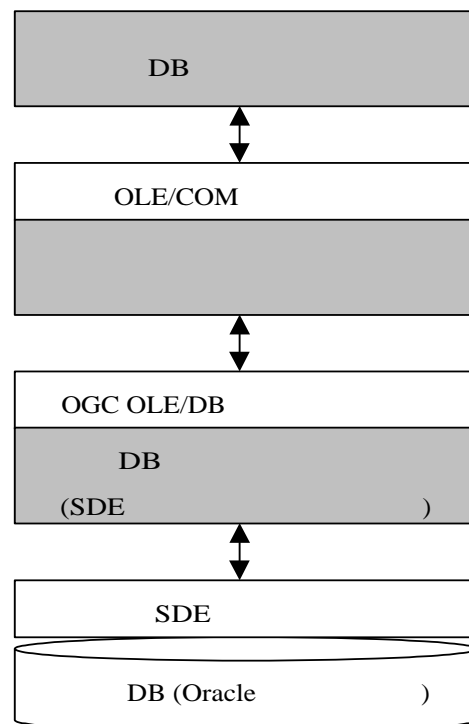
이와 같이 SDE 형식의 교통DB를 공유가 가능한 GIS 파일 포맷으로 변환하는 방식은 공유 체계가 단순하고 구현 방법도 단순한 장점을 가지고 있는 반면에 다음과 같은 단점도 가지고 있다.

- 사용자가 원하는 일부 지역, 다시 말해서 레이어의 전체 데이터가 아닌 일부 데이터에 공간 정보가 불가능하다. 이는 파일 포맷으로의 Import 또는 Export 기능이 레이어의 일부 데이터가 아니라 전체 데이터에 대해서만 공간 데이터의 변환을 수행할 수 있기 때문이다.
- Off-line 상에서 Shape 또는 SDTS의 데이터 파일을 교통DB에 Import 시켜 데이터를 공유하는 방식으로, On-line 상에서 실시간으로 타 기관의 공간 정보를 상호 공유할 수가 없다. 그러므로 공유된 교통DB는 가장 최신의 변경 사항을 반영하기가 어렵다.
- 변경되는 교통DB의 내용을 수시로 반영하기 위해서는 주기적인 업데이트 과정이 필요하게 된다.

나. 개방형 GIS 컴포넌트를 이용한 교통DB의 직접 공유 방안

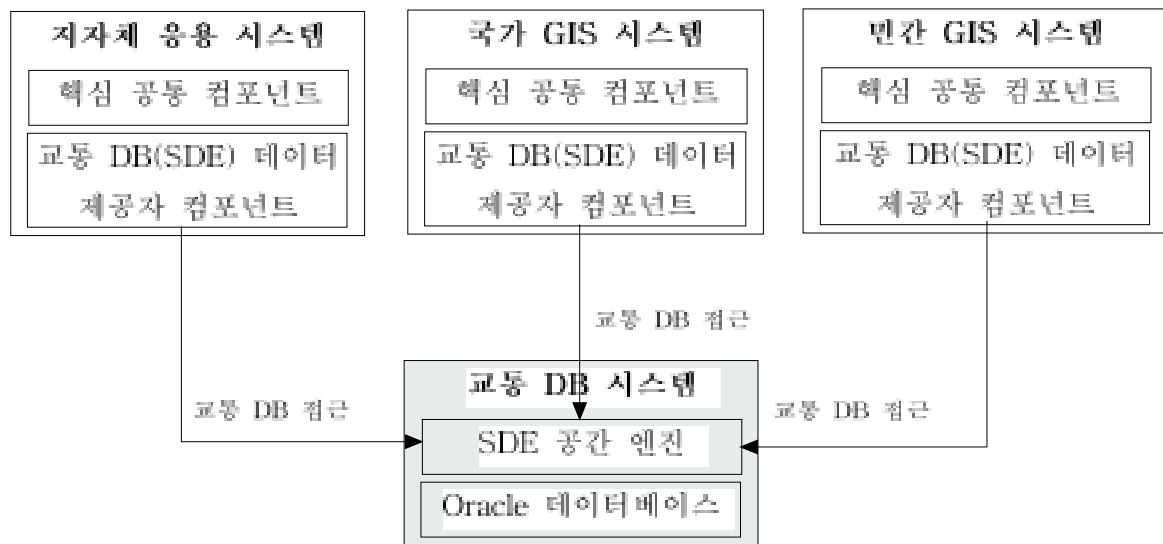
분산 환경에서 교통DB를 상호 공유하기 위한 두 번째 방법은 OGC에서 발표하고 있는 OLE/COM 구현 사양의 개방형 GIS 컴포넌트를 이용하여 교통DB의 공간 정보들을 직접

공유하는 방안이다. 교통DB의 공간 정보를 직접 접근하여 공간 정보를 상호 공유하기 위해서는 가장 먼저 교통DB의 공간 정보를 OGC의 표준 인터페이스로 접근이 가능하도록 해주는 교통DB를 위한 데이터 제공자 컴포넌트가 구축되어야 한다. 그러나 다행히도 교통DB의 경우는 SDE 공간 엔진을 이용하여 구축되어 있고 개방형 GIS 컴포넌트 기술 개발에서는 이미 SDE를 위한 데이터 제공자 컴포넌트를 기 구축하고 있다. 둘째로 중요한 것은 SDE 데이터 제공자 컴포넌트를 접근함으로써 얻어지는 교통DB의 정보를 처리하기 위한 GIS 툴 프로그램이 필요하게 된다. 개방형 GIS 컴포넌트 기술 개발에서는 이를 위해서도 OGC 사양을 준수하고 있는 Geometry와 Spatial Reference System 컴포넌트를 구현하고 있으며, 또한 기존 컴포넌트 기반의 상용 GIS 툴을 기반으로 하여 MapDraw 컴포넌트, Network 컴포넌트 그리고 Application Service 컴포넌트들을 구현하여 제공하고 있다. 이와 같이 데이터 제공자 컴포넌트와 핵심 공통 컴포넌트들의 기능들은 응용 시스템으로 하여금 교통DB를 자유자재로 접근하여 이용할 수 있도록 해준다. <그림 5-7>은 개방형 GIS 컴포넌트를 이용하여 접근될 수 있는 교통DB의 구조를 보여준다.



<그림 5-7> 개방형 GIS 컴포넌트를 이용한 교통DB 접근 방법

위의 <그림 5-7>에서 보면 교통DB를 직접 접근하여 상호 공유하기 위해서 데이터 제공자 컴포넌트와 핵심 공통 컴포넌트가 이용되고 있음을 알 수 있다. 그러나 여기서 교통DB를 이용하고자 하는 사용자가 OGC OLE/DB 표준 인터페이스를 이용하여 클라이언트 시스템을 직접 구성한다면, 위의 그림에서 핵심 공통 컴포넌트는 생략해도 무방하다. 다시 말하자면, 교통DB의 정보의 상호 공유는 핵심 공통 컴포넌트가 생략되고 데이터 제공자 컴포넌트 구현만으로도 가능하다는 것이다. 그러므로 적은 노력과 비용으로 분산 환경에서 교통DB를 상호 공유시키기 위해서는 데이터 제공자 컴포넌트만 이용하는 방식을 취하는 것이 좋다고 볼 수 있다. 다음 <그림 5-8>은 다양한 기관들이 개방형 GIS 컴포넌트들을 이용함으로써 교통DB에 접근할 수 있는 구성도를 보여준다.



<그림 5-8> 다양한 기관들이 교통DB를 접근하기 위한 시스템 구성도

위의 <그림 5-8>을 보면 지자체 응용 시스템이나 국가 기관 관련 GIS 시스템 그리고 민간의 GIS 시스템들도 핵심 공통 컴포넌트와 교통DB를 위한 데이터 제공자 컴포넌트를 이용하면 누구나 교통DB의 정보를 접근할 수 있음을 알 수 있다.

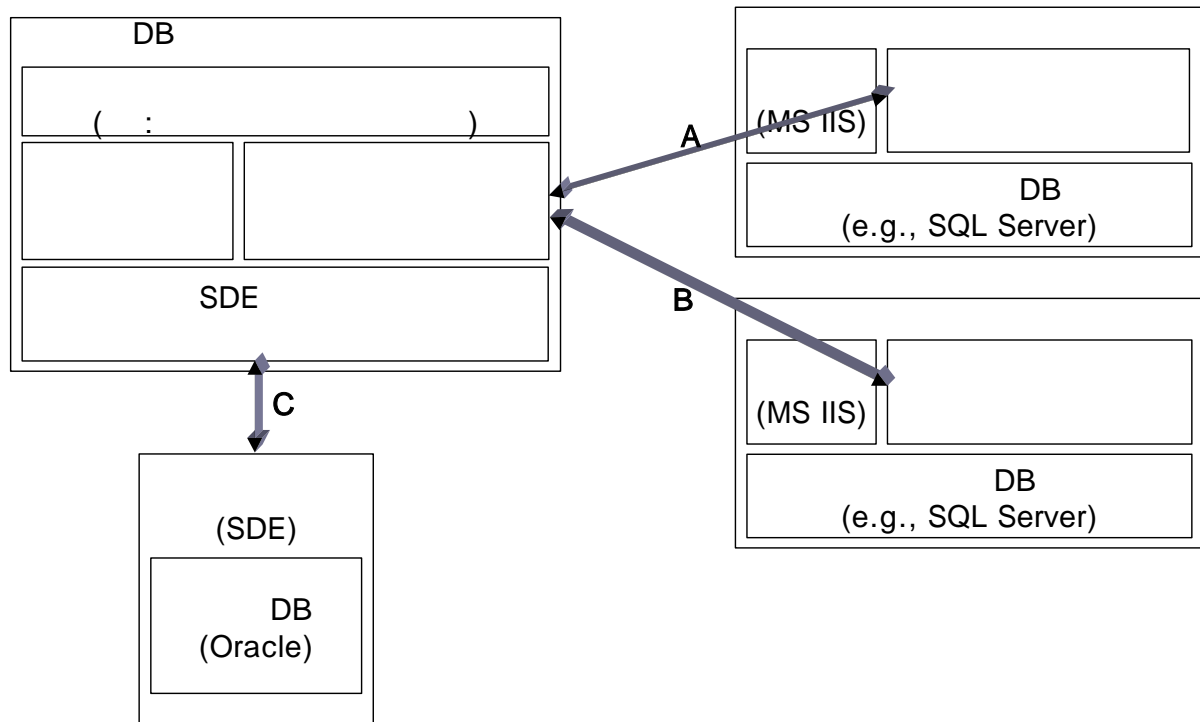
끝으로 이와 같은 개방형 GIS 컴포넌트들을 이용함으로써 교통DB를 직접 접근하여 공간 정보를 상호 공유시키는 방식은 공간 데이터 변환 방식과 다른 점에 대하여 살펴볼 필요가 있다.

- 교통DB를 표준화된 인터페이스로 직접 접근하기 위해서는 OGC의 OLE/COM 구현 사양을 미리 파악하고 있어야 한다. 특히, 데이터 제공자, Geometry 그리고 Spatial Reference System 컴포넌트 구현 사양을 정확히 파악하고 구현해야 한다. 물론 핵심 공통 컴포넌트를 이용하지 않는 경우에는 데이터 제공자 컴포넌트 구현 사양만 파악하고 있으면 된다.
- 교통DB의 경우 SDE의 GIS 엔진을 이용하여 공간 정보를 구축하고 있으며, 현재 개방형 GIS 컴포넌트 기술 개발에서 SDE 데이터 제공자 컴포넌트를 기 구축하고 있으므로, 데이터 제공자 컴포넌트를 구축하기 위해서 소모되는 시간과 노력이 필요하지 않는 장점을 가지고 있다.
- SDE 데이터 제공자 컴포넌트의 공간 검색 기능 및 공간 연산 기능을 이용함으로써 사용자가 원하는 레이어의 원하는 지역 정보만을 공유할 수 있다.
- 사용자가 필요할 때 교통DB를 접근하여 실시간으로 공간 정보를 읽어가므로 항상 최신의 교통DB 정보를 사용자에게 제공할 수 있다.
- GIS 데이터 변환 방식과는 달리 On-Line상에서 수행되며, 교통DB 정보의 공유뿐만 아니라 교통DB에 대하여 공간 연산 작업의 수행도 가능하다.
- 그러나 만약, 교통DB가 SDE가 아닌 다른 GIS 파일 포맷 또는 GIS 소프트웨어로 변경된다면, 새로이 변경되는 파일 포맷 또는 소프트웨어에 관한 OGC용 데이터 제공자 컴포넌트를 새로이 구축해야 하는 문제도 있다. 또한 임의의 교통DB 클라이언트 시스템이 다른 기관의 다양한 데이터를 접근하고자 할 때에는 그 기관의 GIS 데이터 포맷 또는 소프트웨어에 따른 데이터 제공자 컴포넌트를 모두 구축하고 있어야 한다.
- 이와 같이 개방형 GIS 컴포넌트들을 이용하여 교통DB를 직접 공유하는 방식은 공유 데이터 포맷으로의 파일 변환 방식에 비하여 구축이 어려운 면이 있으나 실시간, 최신의 데이터, 공간 연산 지원, 다중 공간 데이터 공유 가능 등의 많은 장점을 가지고 있음을 알 수 있다.

2. 교통DB 정보의 유통 방안

여기서는 교통DB의 정보를 단순히 공유하는 것보다 발전된 개념으로 분산 네트워크 환경에서 교통DB의 정보를 유통시키는 방안에 대하여 살펴보도록 한다. 우선 교통DB정보를 유통시키기 위해서는 핵심 공통 컴포넌트와 데이터 제공자 컴포넌트 이외에 OGC에서 제시하는 정보 유통 컴포넌트가 추가적으로 필요하게 된다. 그리고 정보 유통을 지원하기 위한 유통 게이트웨이, 유통 노드 그리고 메타 데이터 편집기가 요구되게 된다. <그림

5-9>는 교통DB를 이용하는 응용 시스템 그리고 유통 게이트웨이, 유통 노드 그리고 교통DB를 저장하기 위한 공간 데이터 서버로 이루어진 교통DB의 공간 정보 유통을 위한 구성도를 보여준다.



<그림 5-9> 교통DB의 공간 정보 유통 체계 구성도

- 교통DB를 이용하는 응용 시스템

- “도로 굴착 업무”와 같은 응용 업무는 교통DB의 도로 정보를 필요로 한다.
- 교통DB를 이용하는 응용 시스템은 자신이 원하는 교통DB의 도로 정보가 어디에 위치하는지를 검색하기 위하여 클라이언트 정보 유통 컴포넌트를 가지고 있다. 이 클라이언트 정보 유통 컴포넌트는 유통 게이트웨이(A)와 유통 노드(B)로부터 자신이 원하는 정보를 얻어올 수 있다.
- 핵심 공통 컴포넌트와 데이터 제공자 컴포넌트는 클라이언트 정보 유통 컴포넌트가 검색한 교통DB의 공간 데이터 서버에 접속(C)하여 응용 시스템이 원하는 정보를 접근하고 응용 시스템이 원하는 업무를 수행하도록 한다.

- 유통 게이트웨이

- 유통 게이트웨이는 여러 개의 유통 노드 리스트를 관리한다. 유통 노드는 어떠한 교통DB의 정보가 어떠한 서버에 저장되어 있다는 메타 정보 리스트를 관리하는 것이다.
- 국내의 경우 교통 정보를 관리하는 공간 데이터 서버가 크게 교통 개발 연구원, 도로 정보 협회 등 그리 많지 않으므로 유통 노드에서 관리 될 수 있는 메타 정보 리스트가 그리 많지 않다. 그러므로 이러한 메타 정보를 관리하는 유통 노드는 하나 정도만 있어도 충분할 것이라고 사료된다. 그러면 물론 유통 게이트웨이에는 하나의 유통 노드만을 가지는 구조가 되는 것이다.
- 유통 게이트웨이 정보 유통 컴포넌트는 클라이언트 정보 유통 컴포넌트가 요구하는 메타 정보 목록을 포함하는 유통 노드 정보 리스트를 클라이언트에게 전송하는 역할을 수행한다.

- 유통 노드

- 유통 노드는 실제로 메타 정보(교통DB 정보가 어느 서버에 저장되어 있는지의 정보) 리스트를 관리하는 역할을 수행한다.
- 유통 노드 정보 유통 컴포넌트는 클라이언트 유통 컴포넌트가 요구하는 메타 정보를 결과로서 반환함으로써 클라이언트의 응용 시스템이 자신이 원하는 교통DB 정보를 얻기 위해서 어느 공간 데이터 서버에 접속해야 하는지를 알려준다.

- 공간 데이터 서버

- 교통DB의 공간 데이터 서버는 SDE와 Oracle을 이용하여 구축되어 있다. 그러므로 이 교통DB를 접근하기 위해서는 SDE 데이터 제공자 컴포넌트가 필요하고 클라이언트의 응용 시스템을 보면 SDE의 데이터 제공자 컴포넌트를 포함하고 있음을 알 수 있다.

제4절 결 론

본 장에서는 분산 환경에서 일반적인 GIS 정보를 공유시키는 방법 그리고 유통시키는 방법, 그리고 이러한 GIS 정보의 상호 공유와 유통 방법을 교통DB에 적용하는 방안에 대하여 살펴보았다. 특히, GIS 정보의 상호 공유 방법으로는 SDTS 또는 Shape과 같은 주로 이용되는 공통 포맷으로 변환하는 방법과 핵심 공통 컴포넌트와 데이터 제공자 컴포넌트의 개방형 GIS 컴포넌트를 이용하는 방법의 두 가지에 대하여 설명하였다. 그리고 정보 유통 시스템을 구축하기 위해서는 데이터 제공자와 핵심 공통 컴포넌트 이외에 OGC에서

제안하는 정보 유통 컴포넌트를 추가로 구축해야 한다는 것에 대하여 설명하였다. 끝으로 본 장의 결론을 간략히 요약해보면 다음과 같다.

- 공간 정보를 상호 공유하기 위한 방법으로는 공통 포맷으로의 파일 변환 방법과 개방형 GIS 컴포넌트를 이용하는 방법의 두 가지 방법이 있다.
- 파일 변환 방법은 시간과 노력이 적게 드는 장점이 있으나, 개방형 GIS 컴포넌트를 이용하는 방법에 비하여 많은 단점을 가지고 있다.
- 공간 정보 유통을 위해서는 앞에서 설명한 핵심 공통 컴포넌트와 데이터 제공자 컴포넌트 이외에 정보 유통 컴포넌트가 필요하다.
- 교통DB의 정보를 상호 공유하기 위해서는 SDTS나 Shape 파일 형식으로의 데이터 변환 방식과 개방형 GIS 컴포넌트의 SDE 데이터 제공자 컴포넌트와 핵심 공통 컴포넌트를 이용하는 방식이 가능하다.
- On-Line 상에서, 다중 공간 데이터 서버를 지원하고, 실시간으로 정보를 접속하고, 공간 연산 기능을 지원하고, 사용자가 원하는 지역만 접근하는 등의 기능이 필요한 경우에는 반드시 개방형 GIS 컴포넌트를 이용한 방법을 채택해야 한다.
- 교통DB의 정보 유통을 위해서는 교통DB 정보의 메타 정보 검색을 위하여 유통 노드와 유통 게이트웨이를 추가로 구축해야 한다.
- 개방형 GIS 컴포넌트를 이용하는 방법은 대용량의 데이터를 한꺼번에 공유하고자 할 때 네트워크를 통하여 데이터가 전송되는 속도가 느리므로 성능상의 문제를 발생시킬 수도 있으나, 그 외의 다양한 장점을 가지고 있으므로 신중히 정보 공유 방법을 선택할 필요가 있다.

제6장 교통DB와 GDF

제1절 GDF 개요

제2절 GDF와 교통DB의 연계 방안

제1절 GDF 개요

자동차 대수의 폭발적인 증가는 잦은 교통 사고 유발과 교통 정체의 커다란 사회 문제로 확산되고 있다. 이러한 문제를 해결하기 위하여 국내외 교통 관계 전문가들은 교통 관련 문제를 해결하기 위한 합리적인 수단으로서 자동차 항법 장치에 관한 연구를 활발히 수행하고 있다. 그러나 자동차 항법용 수치도로지도에 대한 인식 부족과 작업 규모의 방대함으로 작업의 수행에 어려움이 있었다. 그리하여 지금까지 사단법인 한국도로정보협회에서는 자동차 항법용 수치도로지도를 구축하기 위하여 국제 표준인 ISO-GDF를 기반으로 국내 실정에 알맞은 GDF-K 표준안을 개발하였고, 이 표준에 맞추어 GDF-K 형식의 수치도로지도를 구축하게 되었다.

1. 국내외 동향

- 일본

- JDRMA(Japan Digital Road Map Association)를 통해 전자지도의 표준작업, DB 구축 및 유지관리를 진행한다.
- 일본 전지역의 폭 3m 이상의 고속도로, 국도, 지방도, 기타 도로망에 대한 도형정보와 속성정보를 구축하고 있다.
- 구축된 도로 DB는 차량항법장치, 교통규제정보 DB, 도로교통조사 DB, 특수차량운용 허가 시스템 등에 활용되고 있다.

- 유럽

- CEN, CERCO, ERTICO 등의 조직을 통하여 교통DB 표준을 추진하고 있다.
- 유럽의 DB 표준은 CEN-GDF로 현재 Version 3.0까지 개발되었으며, 국제 표준인 ISO-GDF의 기반이 되고 있다.
- 구축된 교통DB는 각종 GIS사업, 차량항법장치, PC를 이용한 여행자정보시스템, 차량관리시스템 등과 같이 ITS와 GIS의 폭 넓은 분야에서 활용되고 있다.

- 미국

- SAE, ITS America, FHWA 등의 전자지도 관련 기관들을 통해 교통DB 표준화 작업을 추진한다.
- 일종의 메타데이터에 대한 표준으로 SAE J1663을 완성하였고, 공간정보 DB에 대한 호환을 위해 USGS에서 개발한 SDTS를 ITS에 대한 표준으로 발전시키고 있다.
- 민간 업체들을 중심으로 정부의 협조하에 전자지도 DB를 구축하여 각종 ITS 사

업 및 GIS 사업에 활용하고 있다.

- 국내 상황

- 국립지리원에서 DXF 형태의 수치 지형도를 구축하였으며, 도로 선형 및 지형 지물 정보를 포함하고 있다.
- 국토연구원에서 SDTS 형태의 도로망 수치지도를 발표하고 있으며, 도로 중심의 도형정보와 속성 정보를 포함하고 있다.
- 교통개발연구원에서 SDE의 공간 엔진을 이용하여 전국교통DB를 발표하고 있으며, 전국 교통을 위한 속성 정보 및 도로의 도형 정보로 구성되어 있다.
- 표준화와 관련해서는 한국전산원에서 국내 교통전자지도용 DB 표준화 연구결과로서 GDF-K를 표준안 초안으로 채택(1999/11)하였으며, 현재 국가 표준으로 추진하고 있다.(TTA)

2. ISO-GDF 특징

- GDF에서의 데이터 모델은 도로나 빌딩 등과 같은 실세계의 지리학적인 객체를 데이터베이스 형태로 표현한 것으로, 각각의 객체는 정확히 하나의 객체 클래스와 하나의 객체 주제에 포함되어야 한다. 그리고 각각의 객체는 0개, 한개 혹은 여러 개의 속성을 가질 수 있으며, 또한 한 개 혹은 여러 개의 다른 객체들과 연관될 수 있다. 그리고 각각의 객체는 정확히 한 개의 객체 목록에 속하며, 객체 목록에는 점, 선, 면 객체, 그리고 복합 객체가 존재한다. 점, 선, 면 객체는 GDF에서 두 가지로 표현되어지며 이것은 GDF의 레벨에 따른다.
- GDF의 구성은 개념적으로 레벨 0부터 레벨 3까지의 세 개 레벨로 나뉘어진다.
 - 레벨 0 - 기본적인 그래픽 블록, 노드(0차원), 에지(1차원), 페이스(2차원)
 - 레벨 1 - 단순 객체, point, line, area feature, 레벨 0의 그래픽 블록으로 정의
 - 레벨 2 - 복합 객체, 단순 객체 혹은 복합 객체들의 집합
- GDF는 Graph 위상 관계를 가지고 있다.
 - Non-explicit topology : 대상간에 어떠한 관계도 없는 경우 또는 공간상의 관계가 중요하지 않은 경우, 예를 들어 단순한 지도의 디스플레이에만 사용되는 경우이다.
 - Connectivity topology : 0차원과 1차원간의 위상 관계가 명확하게 정의된 경우로 효과적으로 네트워크 연산을 수행하게 할 수 있다.
 - Full topology : 0 차원, 1 차원, 2 차원간의 모든 위상관계가 명확하게 정의된 경우로서 네트워크 연산에서 Area 정보를 효과적으로 통합할 수 있게 한다

- GDF에서 속성 모델은 기본적으로 GDF 자체에 속성 정보를 포함하는 것을 원칙으로 하고 있다. 그러나 이미지나 멀티미디어 정보와 같은 이진 정보는 이진 파일로 속성 정보를 첨부하는 경우도 있다.
- GDF에서는 객체간에 Relational 모델이 존재할 수 있다. 관계란 두개 혹은 그 이상의 객체들간의 의미 있는 연결을 말하며, 객체는 같은 클래스일 수도 있고 다른 클래스일 수도 있다.
- 결론적으로 이러한 자동차 항법용 및 지능형 교통정보 시스템용 수치도로지도인 GDF는 기본적으로 정확한 차량 위치의 표현 기능, 신속한 지도 변환 기능, 경로 안내 및 분석 기능, 그리고 속성 정보 제공 기능 등을 용이하게 지원하기 위한 모델을 채택하고 정보들을 포함하고 있는 것이다.

제2절 GDF와 교통DB의 연계 방안

차량 항법용 수치지도인 GDF와 교통 주제도의 전국교통DB를 상호 연계하여 사용하기 위해서는 GDF와 전국교통DB를 통합하는 방안, GDF 또는 전국교통DB의 저장 형식을 변환하는 방안 그리고 GDF와 전국교통DB를 그대로 두고 On-Line 상에서 직접 공유하는 방안의 세 가지가 존재할 수 있다. 본 절에서는 이러한 세 가지 방안의 가능성 및 시스템 구성 방법에 대하여 살펴보도록 하며, 자세한 세부 구현 내용 사항은 본 절에서는 생략하도록 한다.

1. GDF와 전국교통DB의 완전 통합 방안

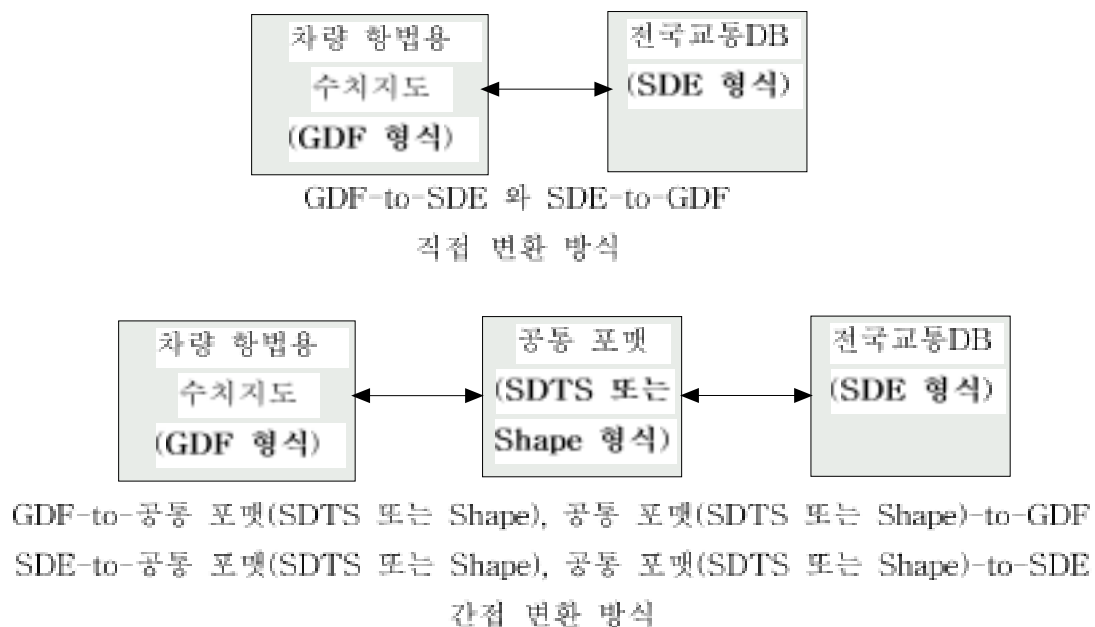
전국교통DB의 교통 주제도와 차량 항법용 지도인 GDF를 통합하기 위해서는 먼저 교통 수치지도에 대한 국내외 표준화 동향을 파악하고 사용자의 요구사항을 수용할 수 있는 새로운 국내 표준화 포맷을 설정할 필요가 있다. 실제로 전국교통DB의 수치지도는 NGIS의 수치지도를 기반으로 교통 계획과 같은 정보를 추가함으로써 교통 네트워크를 구축하는 것을 주목적으로 하고 있으며, 이에 비해 GDF는 차량 항법용 수치지도의 역할을 주목적으로 하고 있다. 또한 이 GDF와 전국교통DB의 두 포맷을 통합함에 있어서도 내부의 데이터 모델, 좌표 체계 또는 도엽 단위 체계 등 많은 부분이 서로 상이하므로 이들을 표준화를 고려하여 변환할 수 있는 기능들을 추가적으로 개발해야 한다.

그러므로 GDF와 전국교통DB를 완전히 통합하는 방식은 향후에 ITS 또는 교통 관련 시스템을 개발하고자 할 때 매우 유용하게 이용될 수 있다. 그러나 새로운 데이터 포맷을

위한 데이터 모델 및 기반 사항들을 새로이 정의하고 개발해야 하는 문제점도 가지고 있다.

2. GDF 또는 전국교통DB의 저장형식을 변환하는 방안

GDF 또는 전국교통DB의 저장형식을 변환함으로써 정보를 상호 공유할 수 있는 방안은 첫째 GDF와 전국교통DB 포맷을 직접 교통DB와 GDF 포맷으로 변환하는 방법이 존재할 수 있으며, 둘째 GDF와 전국교통DB 포맷을 상호 공유할 수 있는 공통 포맷으로 변환하는 방법이 존재할 수 있다. <그림 6-1>은 SDE를 사용하는 전국교통DB와 GDF의 포맷을 변환함으로써 정보를 상호 공유할 수 있는 두 가지 방안을 보여준다.

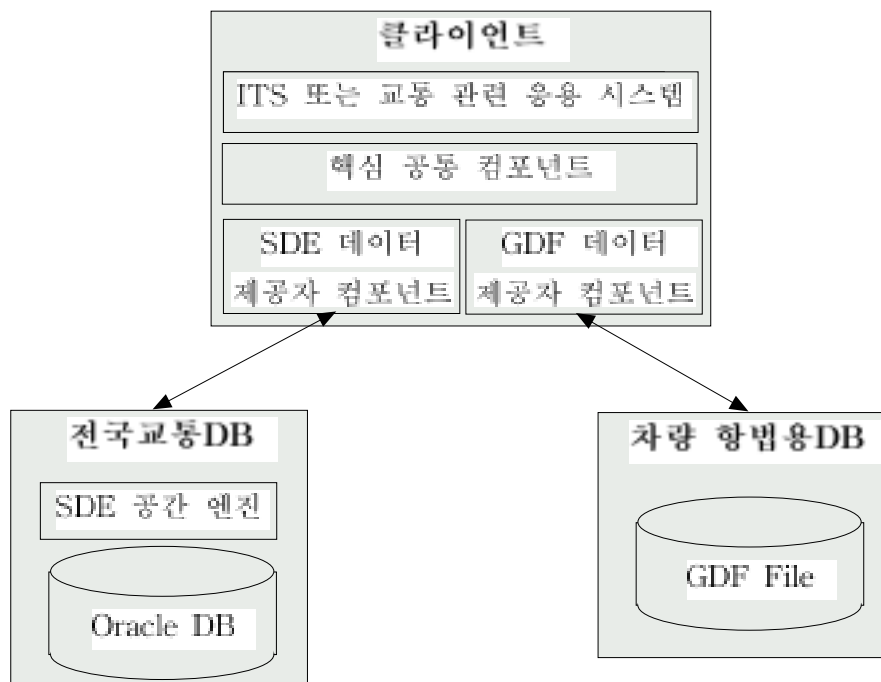


<그림 6-1> GDF 또는 전국교통DB의 포맷 변환을 통한 정보 공유 방안

<그림 6-1>에서 직접 변환 방식은 구성은 단순하나, 현재 GDF에서 SDE로 또는 SDE에서 GDF로 변환할 수 있는 기능이 구현되어 있지 않다. 이에 비해 간접 변환 방식은 SDE에서 Shape 또는 SDTS로 그리고 GDF에서 Shape 또는 SDTS로 변환할 수 있는 기능의 많은 부분이 구현되어 있다. 그러므로 현 시점에서는 간접 변환 방식이 구현하기가 좀 더 쉽다. 물론 이 직접, 간접의 두 방식은 Off-Line 상에서 수행되며, 실시간으로 공간 정보를 상호 공유할 수는 없다.

3. On-Line 상에서 직접 공유하는 방안

끝으로 On-Line상에서 교통DB와 GDF 정보를 실시간으로 상호 공유하기 위해서 개방형 GIS 컴포넌트의 데이터 제공자 컴포넌트와 핵심 공통 컴포넌트를 이용하는 방법이 있다. 다시 말하면, 현재 구축되어 있는 교통DB를 위한 SDE 데이터 제공자 컴포넌트, GDF 포맷을 위한 GDF 데이터 제공자 컴포넌트 그리고 핵심 공통 컴포넌트를 이용함으로써 교통DB와 GDF의 정보를 상호 공유할 수 있는 시스템을 구성할 수 있다. <그림 6-2>



<그림 6-2> On-Line 상에서의 전국교통DB와 차량 항법용DB 상호 공유 방안

<그림 6-2>에서 보듯이 On-Line 상에서 전국교통DB와 차량항법용DB를 상호 공유하기 위해서는 SDE 데이터 제공자 컴포넌트와 GDF 데이터 제공자 컴포넌트가 필요함을 알 수 있다. 현재 SDE 데이터 제공자 컴포넌트는 개방형 GIS 컴포넌트 기술 개발에서 이미 개발되어 있으므로 직접 이용이 가능하나, GDF의 경우는 신규로 데이터 제공자 컴포넌트를 개발하여야 한다. 이러한 On-Line 상에서의 직접 공유 방식은 네트워크 상의 속도 문제로 인하여 대용량의 데이터를 상호 공유하고자 할 경우에는 성능상의 문제를 가져올 수도 있다. 그러나 On-Line이 가능하고, 실시간의 정보를 접근할 수 있고, 공간 연산을 수행할 수 있고, 향후 확장성도 뛰어난 장점을 가지고 있다.

제7장 결 론

본 연구에서는 교통개발연구원에 구축되어 있는 전국교통DB를 이 기종 분산 환경에서 상호 공유하고 유통시킬 수 있는 방법에 대하여 알아보았다. 특히, OGC에서 제시하는 OLE/COM, CORBA 그리고 Catalog Service를 위한 국제 표준 사양을 수용하여 국내에서 개발되고 있는 개방형 GIS 컴포넌트의 개념 및 결과물을 이용함으로써 전국교통DB를 상호 공유하고 유통시킬 수 있는 방안에 대하여 살펴보았다.

이를 위하여 2 장에서는 개방형 GIS 컴포넌트의 개념 및 정의, 개방형 GIS 컴포넌트 관련 국내외의 기술 개발 동향, 그리고 왜 개방형 GIS 컴포넌트 기술 개발이 필요한가에 대하여 살펴보았다. 특히, 크게 나누어 데이터 제공자 컴포넌트, 핵심 공통 컴포넌트, 정보 유통 컴포넌트 그리고 응용 컴포넌트로 구성되는 개방형 GIS 컴포넌트 시스템 구성을 파악하게 되었다. 3 장에서는 개방형 GIS 컴포넌트의 하부구조를 이루며 공간 정보 공유 및 유통에서 가장 중요한 역할을 수행하는 데이터 제공자 컴포넌트의 기능 및 구성 방법에 대하여 집중적으로 살펴보았으며, 4 장에서는 데이터 제공자 컴포넌트와 직접 연결되는 핵심 공통 컴포넌트의 기능 및 구성 방법에 대하여 살펴보았다. 그리고 5 장에서는 분산 네트워크 환경에서 전국교통DB를 어떻게 공유 및 유통시킬 수 있는가에 대한 방안에 대하여 살펴보았다. 특히, 3 장과 4 장에서 설명된 데이터 제공자 컴포넌트와 핵심 공통 컴포넌트를 이용하여 교통DB의 정보 공유 및 유통 방안의 장점 및 단점에 대해서도 설명하였다. 끝으로 6장에서는 도로정보협회에 구축되어 있는 차량 항법용 수치지도 포맷인 GDF-K와 전국교통DB를 연계시킬 수 있는 방안에 대하여 살펴보았다.

그리고 본 연구에서는 전국교통DB의 공유 및 유통 문제를 해결하기 위하여 공간 데이터의 포맷 변환 방법과 개방형 GIS 컴포넌트의 이용 방법의 두 가지를 제시하고 있다. 이 두 가지 방법은 각각의 장단점을 가지고 있을 뿐만 아니라 전혀 다른 특징을 가지고 있다. 그러므로 향후 전국교통DB를 공유하고자 하는 목적 또는 응용에 따라서 사용자가 두 가지 방법중의 하나를 선택해야 하는 경우가 발생할 때 본 연구에서 제시하는 두 가지 방법의 특징 및 장단점을 신중히 고려하여 선택하여야 할 것이다.

끝으로 본 연구 결과를 이용하면 개방형 GIS 컴포넌트의 개념 및 시스템 구성을 파악할 수 있으며, 또한 이들 개방형 GIS 컴포넌트의 세부 컴포넌트들의 인터페이스 구성 및 구현 내용을 자세히 파악할 수 있다. 그리고 이러한 개방형 GIS 컴포넌트의 컴포넌트들을 이용함으로써 전국교통DB와 같은 공간 정보를 상호 공유하고 유통시킬 수 있는 방안도 습득할 수 있게 된다. 그러므로 향후 기관별 또는 사업별로 전국교통DB를 위한 정보 유통 사업을 추진하기 위한 기반 연구 결과로서 이용될 수 있다.

