

객체그룹화에 기반한 지리정보시스템의 설계[†] The Design of Geographic Information System based on Object Grouping

강 신 봉*
Kang, Shin-Bong

주 인 학*
Joo, In-Hak

최 윤 철**
Choy, Yoon-Chul

要 旨

관계 데이터모델은 관계(relations)의 수학적 개념에 기반을 두고 잘 정형화되어 있으며 실용분야에서 많은 검토가 되었으나, 대부분의 지리객체의 특징인 복합 계층구조를 표현하는데는 적합하지 않다. 반면에 객체지향 데이터모델은 복합 계층구조를 자연스럽게 표현할 수 있으나, 현재 대부분의 상용 GIS시스템 사용자가 이용하고 있는 관계데이터모델과의 데이터 공유가 어려우며, 표준화된 구조(format)의 표준 질의어가 정립되어 있지 못하다.

본 논문에서는 RDBMS를 기반으로하여 기존의 관계 데이터모델의 데이터를 사용할 수 있으면서 객체지향 데이터모델의 각종 개념을 지원할 수 있는 객체그룹화(Object Grouping)를 제안하였으며, 이를 이용하여 지리정보시스템을 설계하였다.

ABSTRACT

The relational data model is based on mathematical concept of relations and is well formulated, and so there have been numerous practical applications and studies. However, it is not suitable for representing a complex hierarchical structure, which is the characteristic of most geographical objects. On the other hand, the object-oriented data model can naturally represent a complex hierarchical structure, but there is a difficulty in sharing data with the relational data model which is currently used by most commercial GIS users. Also it has no standard query language with standardized format.

In this paper, we propose an Object Grouping based on RDBMS to use data from a traditional relational data model while supporting various concepts of the object-oriented data model, and we applied this data model to design a GIS.

1. 서 론

지난 10여년간 컴퓨터 하드웨어의 저가격 고성능화와 관련기술의 발달로 인하여 다양한 응용 소프트웨어들이 개발되고 활용되고 있다. 이러한 응용 소프트

웨어들 가운데 복잡한 구조와 다량의 데이터를 취급하는 분야중의 하나로서 지리정보시스템이 최근 활발히 연구되고 있으며, 현실적으로 사용되고 있다. 지리정보시스템은 지리에 관한 대용량의 데이터를 수집, 저장하고 이를 분석, 가공하여 도시계획, 경로

† 본 논문은 1992년도 인공지능연구센터 멀티미디어 정보시스템 플랫폼 개발과제의 위탁과제로 지원받았음.

* 연세대학교 컴퓨터과학과 박사과정

** 연세대학교 컴퓨터과학과 교수

0.9999로 되어 있고, 투영원점이 행정계를 고려하여 최적화, 지도제작, 토지 이용도 평가, 시설관리, 국방 및 천연자원관리등 다양한 분야에 활용할 수 있는 정보를 제공하는 시스템으로 1970년대에는 화일시스템을 사용하다가 1980년대 이후부터는 주로 데이터베이스 관리시스템을 이용하여 지리정보시스템을 구현하고 있다(Won Kim 1995).

그러나 지리정보시스템은 공간객체의 표현을 위한 가변길이 필드와 지리객체의 특성을 이미지로 표현하기 위한 멀티미디어 데이터 타입의 지원, 지리객체의 디스플레이를 위한 컴퓨터 그래픽스 사용 및 효율적인 공간객체의 검색을 위한 공간인덱스등을 지원하여야 하므로 기존의 데이터베이스만으로는 구축하기 어려우며, 효율적인 데이터모델이 필요하다.

현재까지 연구된 GIS 데이터모델로는 여러가지가 있으나 대표적으로 크게 관계 데이터모델과(ESRI 1990, Waugh 1987, Haas 1991) 객체지향 데이터모델(Scholl 1991, David 1993, Worboy 1990, Mohan 1988, O. Gunter 1993)로 구분된다. 지금까지 상용 GIS시스템에서는 ARC/INFO와 같은 관계 데이터 모델이 주로 사용되어 왔으며, 최근 객체지향 데이터 모델을 이용한 시스템이 등장하고 있다.

관계 데이터 모델은 속성데이터를 테이블형태로 쉽게 구축할 수 있고, RDBMS의 데이터 보안, 복구, 트랜잭션 로그 기능을 이용할 수 있으며, 표준화된 질의어인 SQL을 제공하여 검색의 편의 및 유연성을 제공하며, 실용분야에서 많은 검토가 되었으나 대부분의 지리객체의 특징인 복합 계층구조를 표현하는데는 적합하지 않다. 반면에 객체지향 데이터모델은 복합 계층구조를 자연스럽게 표현할 수 있으나, 현재 대부분의 상용 GIS시스템 사용자가 이용하고 있는 관계 데이터모델과의 데이터 공유가 어려우며, 표준화된 구조(format)의 표준 질의어가 정립되어 있지 못하고, 데이터베이스의 구축이 어렵다.

따라서 현재 널리 사용되고 있는 관계 데이터 모델의 데이터를 그대로 사용할 수 있으면서 복합계층구조를 갖는 지리객체를 효과적으로 표현할 수 있는 데이터모델이 필요하다. 본 논문에서는 이러한 요구를 충족할 수 있도록 RDBMS를 기반으로하여 기존의 관

계 데이터모델의 데이터를 그대로 사용할 수 있으면서 객체지향 데이터모델의 각종 개념을 지원할 수 있는 객체-관계 데이터모델을 제안하고, 이를 이용하여 GIS시스템을 설계하였다.

2. GIS에서 객체지향 개념의 적용

2.1 GIS에서 객체의 개념과 기존 모델의 장 단점

객체란 실세계나 추상객체의 상태(state)와 행동(behavior)을 포함하는 패키지(package)를 말하는데 GIS에서의 객체에는 점(point), 선(arc), 면(polygon)과 같은 원시객체(primitive object)와 이들이 다양한 복합 계층형태로 혼합되어 이루어진 그룹객체(group object)가 있다. 특히 실세계의 지리객체는 대부분 그룹객체로서 표현되고 관리되기때문에, 그룹객체를 자연스럽게 표현하고 관리할 수 있어야 한다. 예를들면, 25번 도로는 여러개의 선 객체들로 구성되어 있는데 선 객체 하나하나에 대한 표현이나 분석보다는 이들이 모여서 이루어진 25번 도로라는 그룹객체를 실세계에서는 자주 사용하게 되며, 지리객체로서의 의미를 가지기때문에 그룹객체를 효과적으로 관리할 수 있어야 한다는 것이다. 따라서 그룹객체별 디스플레이, 속성관리, 질의어 수행등을 효과적으로 수행할 수 있는 데이터 모델이 필요하다.

현재 사용되거나 연구되고 있는 데이터 모델은 크게 관계 데이터모델과 객체지향 데이터모델의 두가지 모델로 구분할 수 있는데 각각의 장단점(Wang 1992)은 다음과 같다.

관계 데이터모델의 장점은 기본 속성데이터를 테이블형태로 쉽게 구축할 수 있고, RDBMS의 데이터 보안, 복구, 트랜잭션 로그 기능등을 이용할 수 있으며, 표준화된 질의어인 SQL을 제공하여 검색의 편의 및 유연성을 제공한다는 점이다.

반면 이 모델의 단점으로는 RDBMS가 고정된 길이의 레코드만을 지원하므로 가변형의 공간 데이터 타

입을 표현하기 어렵다. 따라서 보통 공간 객체를 위한 처리기를 따로 둔다. 또한 관계 데이터 모델은 복합 지리정보객체를 표현하기 어려우며, 조인 오버헤드가 있다.

객체 지향 데이터모델의 장점은 복합 계층구조 및 상속 계층구조를 자연스럽게 지원하므로 복합적인 지리정보객체를 표현하기 쉽고 클래스간의 상속이 가능하다. 따라서 실세계를 보다 자연스럽게 표현할 수 있기 때문에 사용자의 질의어 사용이 보다 편리해진다. 또한 네트워크 모델에서의 포인터 개념을 지원하므로 데이터 검색이 효율적이며, 가변길이의 데이터 저장을 허용하므로 공간 및 속성 데이터를 하나의 데이터베이스에 저장, 관리할 수 있다.

객체지향 데이터모델의 단점으로는 데이터구조의 생성 및 데이터베이스의 구축이 어려우며, 표준화된 구조가 없고, 현실적으로 각종 응용 프로그램에서 사용하고 있는 RDBMS와의 데이터 호환이 어렵다는 점이다(David 1993, Egenhofer 1994).

2.2 객체지향의 종류

객체지향 패러다임(paradigm)은 다양한 면을 가지고 있으나, 대부분 데이터 모델링, 프로그래밍 언어, DBMS 및 사용자 인터페이스로 구분한다(Maguire 1994).

Object-Oriented Data Modelling(OODM) : OODM에서 모든 개념적 실체(conceptual entities)는 객체와 객체간의 관계로 모델된다. 이는 컴퓨터 중심의 하위 레벨 표현보다 실세계나 사용자 중심의 상위레벨의 표현에 중점을 둔다.

Object-Oriented Programming Language(OOPLA) : 많은 사람들에게 있어서 객체지향이란 용어는 실제적으로 OOPLA를 말하는데, 이것은 객체의 데이터 구조를 정의하고 액세스(access)할 수 있는 모듈(modules)을 구현하는 분야이다. OOPLA의 핵심요소는 추상화(abstraction), 캡슐화(encapsulation), 상속성(inheritance) 및 다형성(polymorphism)이며, 구현된 프로그램은 재사용성을 높여주므로 시스템의 수정과

확장이 용이하다.

Object-Oriented Data Base Management System(OODBMS) : 최근에 OODBMS에 대한 많은 연구가 이루어지고 있으나 아직까지 실용화되어 있는 시스템은 거의 없다. OODBMS는 지속성(persistent) 객체관리와 객체의 저장과 접근, 객체기반 질의등의 특징을 가진다.

Object-Oriented User Interface(OOUI) : OOUI는 객체를 다룰수 있는 사용자 인터페이스를 제공하는 것을 말한다.

각각의 데이터 모델은 성격에 따라 여러가지 객체 지향개념을 지원하게 되는데 본 논문에서 제시하는 객체-관계형 데이터 모델에서는 OODM, OOPLA 및 OOUI를 지원하며 세부내용은 제3장에서 설명된다.

3. 객체-관계형 데이터모델

3.1 객체지향 데이터 추상화

객체지향의 정의에 의하면 아무리 복잡한 구조를 갖는 복합지리객체라도 한개의 객체로 표현되어야 한다. 즉, 기술적인 제약때문에 하나의 객체가 좀 더 단순한 형태의 부분으로 인위적인 분할이 일어나서는 안된다는 것이다. 객체지향 데이터모델에는 Classification, Generalization, Aggregation (Egenhofer 1988) 및 Ordered Grouping의 네가지 추상화가 있다.

3.1.1 Classification

클래스(class)는 모델링하고자하는 여러 객체의 공통적인 성질(멤버나 멤버함수)을 정의한 것으로 각 객체는 해당 클래스의 인스턴스로서 정의된다. 따라서 여러개의 객체인스턴스들은 공통의 특성을 갖는 하나의 클래스에 사상(mapping)되며 instance-of-relation을 갖는다. 그림 3.1에서 Seoul, Pusan,

Suwon 객체는 City 클래스의 인스턴스이다.

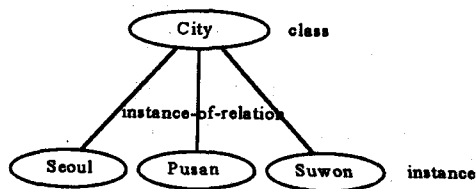


그림 3.1 Classification

3.1.2 Generalization

여러개의 객체들을 더 일반적인 상위클래스로 조합하는것으로서, 하위클래스의 객체들은 상위클래스로부터 상속받는다. 상속은 소프트웨어의 재사용성을 높여주고, 일반적인 개념에서 특수한 개념으로의 단계적인 정의를 가능하게 해준다. 이미 정의된 클래스로부터 상태와 행동에 관한 내용을 물려받고, 새로이 상태와 행동을 추가하거나 변경해서 새로운 클래스를 정의하는 상속은 여러 클래스의 공통적인 특성을 모아서 하나의 상위 클래스에 정의함으로써 각 클래스마다의 중복정의를 피하게 해준다. 상속관계에 있는 두 클래스중 하위클래스의 한 객체는 상위클래스의 한 객체로 볼 수 있는데, 이는 is-a-relation을 나타낸다.

그림 3.2에서 고속도로, 일반국도, 지방도, 철도는 도로 클래스의 하위클래스로써 모든 특성을 상속받는다.

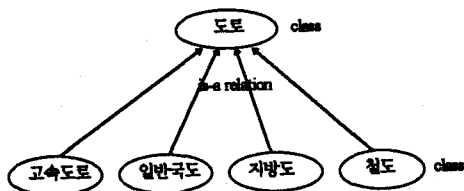


그림 3.2 Generalization

3.1.3 Aggregation

여러개의 객체를 새로운 복합객체로 그룹화하는것으로서, 대부분의 GIS데이터는 복합계층구조를 가지므로 aggregation을 통하여 자연스럽게 표현 할 수 있을뿐만 아니라 효과적인 관리가 가능하다. 이는 part-of-relation으로 나타낸다.

그림 3.3은 상수도의 소블럭을 Aggregation으로 나타낸것이다.

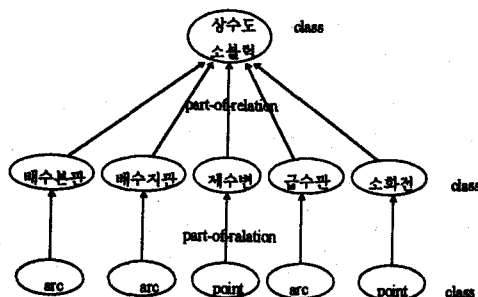


그림 3.3 Aggregation

3.1.4 Ordered Grouping

Aggregation과 유사하나 그룹객체의 소속객체들간에 일정한 순서가 부여되어 있기때문에 흐름분석이나 연결성분석등에 유용하며 ordered-part-of-relation으로 나타낸다. 주로 도로, 상수도, 가스관등 네트워크 그룹화에 사용된다.

3.2 객체 그룹화

3.2.1 데이터 모델의 구조

지리정보 모델링에 있어서 지형정보의 다양하고 복잡한 계층구조를 적절하게 표현하고 검색하며 분석하기 위해서는 객체단위의 처리가 필요하다. 지형정보의 계층구조에는 지도(map)나 레이어(layer)와 같은 물리적(physical)인 계층구조와 행정구역이나 도로, 상수도블럭과 같이 유사성 또는 종속성을 갖는 지리정보를 효율적으로 관리하기위해 정의하는 논리적(logical)인 계층구조가 있다. 따라서 이러한 지리정보간의 계층구

조를 쉽게 표현할 수 있고, 효율적인 검색이 가능한 모델링 방법이 필요하다(Wang 1992).

관계데이터 모델에서의 계층구조는 지리정보를 나타내는 테이블에 상위 지형정보를 나타내는 속성으로 표현한다. 그러나 이 방법은 계층구조를 위하여 테이블내에 인위적인 속성으로 표현하기 때문에 자연스럽게 못할뿐만 아니라 테이블간의 관계가 매우 복잡해진다(Wang 1992). 따라서 대량의 지리정보와 관련된 계층구조를 입력하려면 많은 시간과 비용이 소모된다. 또한 입력된 계층구조를 검색하려면 사용자가 계층구조를 표현하는 테이블간의 속성관계를 알아야 하며, 관련된 테이블간에 결합연산(join operation)을 수행해야 하는 문제점이 있다.

객체지향데이터 모델에서는 지리정보의 계층구조는 효과적으로 표현할 수 있으나 현재 대부분의 GIS사용자가 사용하고 있는 RDBMS와의 데이터 공유가 어렵다.

따라서 RDBMS와 데이터공유를 공유할 수 있으면서 계층적인 지리정보를 쉽게 표현하고, 효율적인 검색 기능을 지원하기 위해서 제안하는 객체-관계형 데이터 모델은 그림 3.4와 같다.

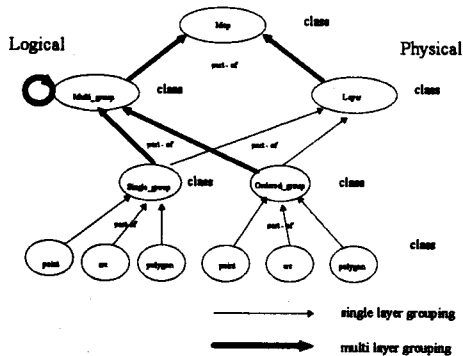


그림 3.4 Object-Relational Data Model의 계층구조

객체-관계형 데이터 모델에서 제일 하층에는 점, 선, 면의 원시 객체가 있으며, 중간층에는 이들에 대한 포인터로 논리적인 그룹을 구성하는 그룹객체 및 원시객체들을 물리적으로 구성하고 있는 레이어 클래스가 있으며, 상층에는 그룹객체와 레이어들을 전부

포함하는 지도(map) 클래스로 구성된다.

그림 3.4에서 전체 도면을 나타내는 지도는 물리적인 여러개의 레이어와 논리적인 multi_group의 계층구조로 구성되는데 레이어는 같은 성질을 가진 객체의 물리적인 단위로서 도로, 강, 행정구역, 상수도 배수관 등이 이에 속한다. multi_group은 여러가지 single_group과 ordered_group의 계층구조로 구성된다. single_group은 점, 선, 면의 원시객체들중 같은 종류의 객체들의 계층구조로 구성되며, ordered_group은 그룹을 구성하는 하위객체간에 순서가 부여된다.

그룹객체는 각각 고유의 의미있는 속성을 갖게되는데, 예를들면 선으로 된 원시객체가 상수도 배수관이라는 single-object가 될 때 사용자는 더 이상 단순한 선의 연결로 보지않고 매설년도, 환경, 유량을 가진 배수관으로 인식하게 된다. 원시객체와 그룹객체간에는 속성의 전파(propagation)가 일어나는데 그룹 구성시 사용자는 전파의 주기와 방법을 지정할 수 있다. 예를들어 행정구역에서 구를 그룹화하여 시라는 그룹객체를 생성했을때 시의 인구는 소속구의 인구들의 합이며, 전파주기는 5년이라고 지정하면 5년단위로 구의 인구의 합이 시의 속성으로 자동으로 전파된다.

원시객체는 도면상에서 표시되는 기본 공간데이터에 해당하는 객체로서 벡터 정보와 위상(topology)을 가지고 있다.

그림 3.5에서 GIS_Object 클래스는 여러개의 계층적인 하위클래스로 정의되며, 하위클래스는 상위클래스로부터 공통적인 데이터와 메소드를 상속받는다.

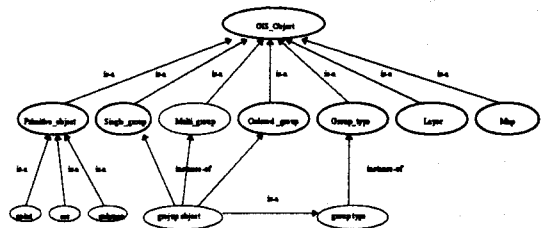


그림 3.5 지리정보의 inheritance 계층구조

그룹객체는 Single_group이나 Multi_group 또는 Ordered_group의 인스턴스으로써 group type으로부터 속성과 메소드를 상속받는다. group type은 그룹객체

의 속성과 메소드를 가지며 사용자에 의해 run time 시에 정의 가능하다. 따라서 다양한 지리정보에 관한 그룹객체를 쉽게 정의할 수 있으며, 확장이 용이하다. 또한 상위클래스에서 정의된 속성과 메소드는 하위클래스에서 그대로 사용할 수 있다.

3.2.2 그룹객체의 생성

그룹객체의 생성방법은 속성에 의한 방법(by attribute)과 선택에 의한 방법(by selection)의 두가지가 사용된다. 속성에 의한 방법은 속성을 이용하여 질의어(query)에 의해 자동으로 그룹화하는 방법이며, 선택에 의한 방법은 사용자의 선택에 따라 수동으로 그룹화하는 방법으로서 마우스를 이용하여 객체를 하나씩 선택하거나 원, 사각형, 다각형을 그린다음 그속에 포함되는 원시객체를 한꺼번에 선택하여 그룹화할 수 있다. 그룹객체를 선택할 때에는 object_viewer상에서 마우스를 이용하여 선택하거나, 그룹객체의 타입(type)을 정의한 다음 윈도우상에서 해당 타입의 그룹객체를 선택할 수 있다. 생성된 그룹객체는 object_viewer에 의해서 계층적인 구조로 관리된다.

3.2.3 그룹객체의 관리

생성된 그룹객체는 소속객체들을 추가, 삭제할 수 있는 Update와 그룹객체 자체를 Delete, Rename할 수 있으며, 소속객체들의 id를 알수 있는 GetChild등의 메소드가 제공된다.

3.2.4 속성 전파

그룹객체도 객체단위로 속성을 가진다. 그룹의 속성은 그룹고유의 속성과 하위 그룹의 속성을 전파받는 기능을 가진다. 이러한 속성전파 기능은 전파주기에 의해서 반복적으로 수행된다. 그룹객체와 소속객체간의 메시지 전달방식과 종류는 그림 3.6과 같은데 소속객체는 그룹객체에 속성을 전파하며, 그룹객체는 소속객체에 일괄처리 메시지를 보낸다.

속성전파 기능은 그룹객체의 생성시 정의한다. 속

성전파주기는 실시간, 일, 주, 월, 년등이 있으며, 그룹객체의 Update에서 수정할 수 있다.

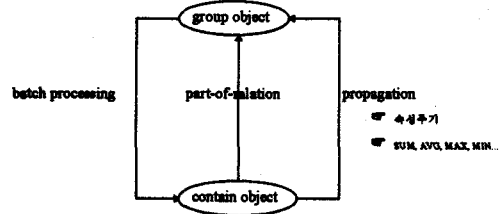


그림 3.6 그룹객체와 소속객체간의 attribute propagation

3.3 지원 도구

객체 그룹화를 위하여 사용되는 도구에는 그림 3.7과 같은 object_viewer가 있다. Object_viewer는 속성에 의한 방법과 선택에 의한 방법으로 그룹화된 그룹객체들의 계층구조를 나타내며, 그룹객체를 viewer상에서 선택할 수 있는 기능도 제공한다.

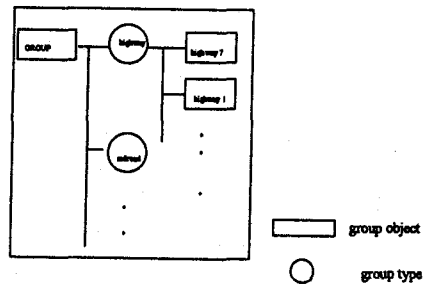


그림 3.7 Object_viewer

Object_viewer에서 사각형은 그룹객체를 나타내며, 원은 그룹타입을 나타낸다. 예를들면, highway라는 그룹타입을 가지는 그룹객체에는 highway-7과 highway-1등이 있으며 이들은 선들의 그룹으로 구성되어 있다. object viewer상에서 highway-7을 마우스로 클릭하면 이를 구성하는 선객체들이 윈도우상의 지도에서 강조(highlight)된다. 또한 윈도우 지도상에서 직접 highway-7이나 highway-1 그룹객체를 선택하기 위해서는 먼저 highway 객체타입을 지정한 후 map 윈도우에서 highway-7 이나 highway-1 그룹객체를 선택할 수 있다.

4. GIS시스템에서 객체-관계형 데이터모델의 지원

객체-관계형 데이터모델을 이용하여 설계한 GIS 시스템은 그림 4.1과 같이 크게 4개의 층으로 구성되어 있다.

- ① 공간 데이터를 관리하는 공간데이터 관리자와 속성(attribute) 데이터를 관리하는 상용DBMS시스템과 이에 대한 인터페이스
- ② 공간데이터와 속성데이터를 통합관리하며, GIS 객체의 기본 클래스와 메소드를 관리하는 객체 관리자
- ③ 입력/편집기, 위상생성기, 질의어 처리기등 GIS시스템의 주요기능을 관리하는 기본요소
- ④ PC 윈도우즈 GUI 와 스크립트 언어를 지원하는 사용자 인터페이스로 구성된다.

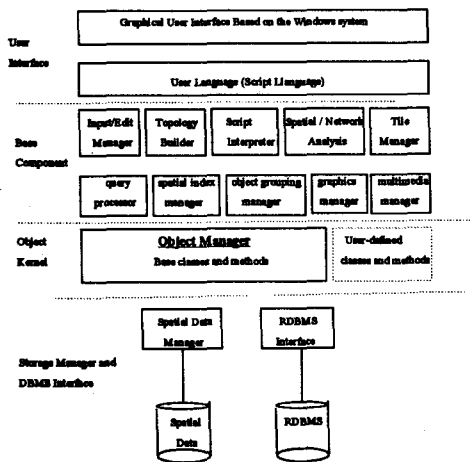


그림 4.1 System Architecture

4.1 저장관리자, DBMS 인터페이스 및 객체 커널(kernel)

본 GIS시스템에서는 공간데이터와 속성데이터는 분리되어 관리되는데, 저장관리자는 공간데이터를 관리하며, 속성데이터는 여러가지 상용 RDBMS를 동일한

인터페이스상에서 사용할 수 있는 투명성을 제공한다.

객체 Kernel은 분리되어 저장되어 있는 공간데이터와 속성데이터를 연결하여 유기적으로 운영할 수 있는 기능을 제공한다. 또한 커널(kernel)에는 GIS 객체를 관리하는데 필요한 각종 클래스와 메소드가 포함되어 있으며, 이러한 커널은 사용자에게 의해서 확장될 수 있다.

4.2 기본요소 및 사용자 인터페이스

기본요소에서는 GIS 데이터에 대한 입력과 편집작업을 수행하며, 위상을 생성하고 각종 공간분석 및 네트워크 분석을 수행한다. 넓은 지역의 지도를 타일(tile) 단위로 나누어 관리함으로써 처리속도를 향상시킬 수 있으며, RDBMS의 SQL을 확장한 geoSQL을 지원하도록 설계하였다. geoSQL의 성능향상을 위하여 R+-tree를 이용한 filtering기능을 제공하며, 복합객체에 관한 위상 관계들을 정의하였다. 모든 공간객체에 대해서는 R+-tree를 생성하여 신속한 공간객체 검색이 가능하며, 객체 그룹화를 통한 데이터 추상화기능을 부여하여 사용자 정의 타입과 복합객체를 효과적으로 표현할 수 있다. 그래픽스 관리자는 질의어의 결과를 다양한 형태의 지도로 표현할 수 있으며, 다른 기본요소와 연계하여 브라우징 기능을 제공한다. 멀티미디어 관리자는 하이퍼링크를 이용하여 공간객체와 관련된 이미지정보를 관리하며, 래스터 데이터를 기본지도(base map)로 사용할 수 있게 한다.

본 시스템의 주요 특징중 하나는 다양한 응용 시스템을 사용자가 쉽게 작성하고 사용할 수 있다는 것이다. 이를 위해서 Windows 3.1 환경에서 GUI를 이용하며, 비전문가가 쉽게 이용할 수 있는 스크립트 언어를 지원한다.

4.3 geoSQL

지리정보시스템에서 지리객체간의 공간적인 관계를 처리하는 공간질의처리는 매우 중요하나 기존의 SQL에서는 그 기능을 제공하지 않는다. 따라서 본 시스템

에서는 지리객체에 관한 공간 질의어를 처리할 수 있는 geoSQL을 설계하였으며, 설계시 고려요소는 다음과 같다(Egenhofer 1990).

- 공간객체에 대한 그래픽 표현
- 디스플레이된 공간 객체를 마우스로 선택(pick, zone)하여 질의어의 입력으로 사용
- 질의어의 결과를 또 다른 질의어의 입력으로 사용
- 질의어의 결과를 다양한 지도 또는 그래픽, 도표형식으로 표현할 수 있어야 한다.

4.3.1. 공간 연산

geoSQL이 갖추어야 할 공간연산에는 다음과 같은 3가지가 있다.

① 단항(unary) 공간 연산자

단일 공간객체에 대한 연산자로서 객체종류별 연산자는 표 4.1과 같다.

표 4.1 객체종류별 연산자

객체 종류	연산자
선	length
면	area, perimeter
선 그룹	length, connect
면 그룹	area
ordered group	direction

② 이항(binary) 공간 연산자

두개의 공간객체간의 연산자로서 distance와 direction(N, S, E, W, NE, NW, SE, SW)이 있다.

③ 공간 relationships

두개의 공간객체간의 위상관계에 의해서 표현되는

공간 relationships은 표 4.2와 같다.

표 4.2 공간 relationships

구분	선,선그룹	면	면그룹
점	lies_on	within	within
선,선그룹	intersect connect	cross within	cross within
면		disjoint meet overlap cover/covered_by inside/contains equal	disjoint meet overlap cover/covered_by inside/contains between
면그룹			mutual_inclusion

이들 중 disjoint, meet, overlap, inside/contains, covers/coveredBy, equal, between, mutual_inclusion 등을 나타내면 그림 4.2와 같다.

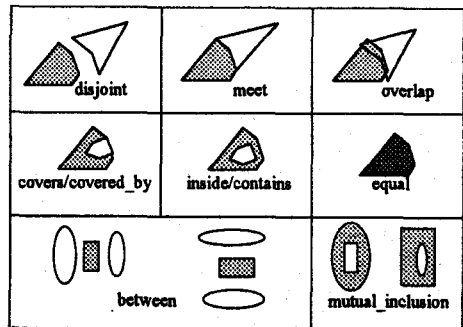


그림 4.2 공간 relationships의 예

4.3.2 공간인덱스를 이용한 2단계 질의 처리

공간 질의어에서는 적당한 공간 인덱스를 사용하지 않을 경우, 저장된 모든 대상객체가 질의 조건에 대해 검색되어지며 이것은 상당한 성능의 저하를 초래한다. 따라서 효율적인 질의처리를 위해서는 공간 접근 기법이 요구된다. geoSQL에서는 2단계과정에 의하여 질의어가 수행되는데, 1단계는 전체 대상객체중에서 R+-tree를 이용하여 후보객체들을 선택하는 filter 단계이며, 2단계는 선택된 후보객체중에서 질의어의 조건을 만족하는 최종객체를 검출하는 refinement 단계이다.

그룹객체에 대한 질의어 처리과정을 예를들어 살펴 보겠다.

```
SELECT geometry
FROM public-building
WHERE distance(public-building.geometry, PICK)<100
```

위의 질의어는 학교, 병원, 소방서등 공공건물중에서 마우스로 선택한 지점으로부터 100km이내에 있는 객체를 검색하여 그래픽으로 출력하라는 것이다.

여기에서 public-building은 그림 43과 같이 레이어 중에서 미리 그룹화를 수행하여 생성한 그룹객체이다. 질의어 수행과정은 그림 44와 같이 선택(pick)한 지점 P로부터 사방이 100km인 지역 R을 정한다. 다음에는 점선으로 표시되어 있는 public-building 그룹객체에 대한 R+-tree의 MBR(Minimal Boundary Rectangle)과 지역 R이 겹치는 MBR ①와 MBR ②에 포함되는 후보 객체 ②와⑥을 먼저 찾은다음, 2단계로 후보객체 ②와⑥이 지역 R에 포함되는지를 공간연산하여 최종 객체 ②를 검색한다.

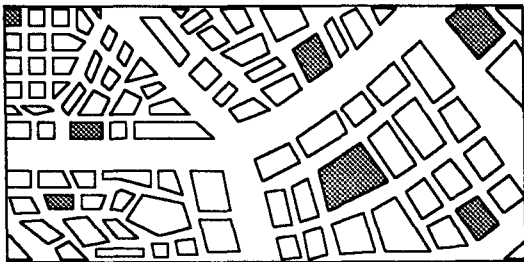


그림 43 layer 상에서 선택된 public-building group

따라서 위의 질의어 수행과정에서 공공건물은 미리 그룹객체로 정의되어 있으므로 공공건물을 찾기위한 별도의 질의어 수행이나 검색작업이 불필요하기 때문에 검색비용을 줄일 수 있다. 또한 그룹객체중에서 P 지점으로부터 100km이내의 객체검색시에도 공간인덱스를 이용하여 먼저 후보객체를 선택한후 선택된 후보객체에 대해서만 각 객체별로 질의어를 수행하기때

문에, 모든 객체를 검색할때보다 효율적인 공간검색이 가능하다.

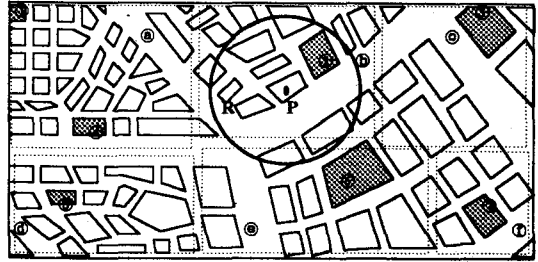


그림 44 R+-tree를 이용한 filter phase

5. 결 론

본 논문은 기존의 관계 데이터모델과 객체지향 데이터모델의 단점을 보완하고 장점을 활용할 수 있도록, 관계 데이터모델의 데이터를 기반으로 하여 객체 지향 개념을 적용할 수 있는 모델로서 객체-관계형 데이터모델을 제시하였다. 객체-관계형 데이터모델에서는 계층적인 구조를 갖는 다양한 지리정보를 객체 그룹화를 통하여 쉽게 표현할 수 있으며, 그룹객체에 대한 효율적인 관리가 가능하다. 또한 다양한 응용분야에 따라 group type의 속성과 메소드를 확장할 수 있으며, 그룹객체와 소속객체간에 속성주기별로 속성이 전파되어 일관성(consistency)이 보장된다.

그리고 위에서 제안한 객체-관계형 데이터모델을 기반으로 하여 GIS시스템을 설계하였다.

공간분석을 효율적으로 수행하기 위하여 위상을 자동으로 생성하며, 공간객체의 검색성능을 높이기 위하여 R+-tree 공간인덱스를 이용한다. 또한 SQL을 기반으로 하여 이를 공간 질의어를 수행할 수 있도록 확장한 geoSQL을 설계하였으며 질의어수행시 R+-tree를 이용하여 공간 filtering을 먼저 수행함으로써 처리 성능을 향상시키도록 하였고 현재 PC 윈도우즈 환경에서 Visual C++을 이용하여 구현 중에 있다.

앞으로의 연구방향으로는 앞에서 설계한 내용들에 대한 구현과, 분산시스템에서 대규모 데이터를 공유할 수 있는 GIS모델의 연구가 필요하다.

REFERENCE

1. Won Kim, MODERN DATABASE SYSTEMS: the object model, interoperability, and beyond, ACM press, 1995.
2. O. Gunter, and W. F. Riekert, The Design of GODOT : An Object-Oriented GIS, IEEE Data Engineering, Vol. 16, No.3, Sep. 1993, pp.4-9.
3. Mohan and R. L. Kashyap, An Object-Oriented Knowledge Representatin for Spatial Information, IEEE TSE, Vol. 14, No. 5, May 1988, pp.675-681.
4. B. David, L. Raynal, and G. Schorter, GeO2: Why Objects in a Geographical DBMS?, proc. of the 3rd Int. Symp. SSD, Jun. 1993, pp.264-276.
5. M. J. Egenhofer, Spatial SQL : A Query and Presentation Language, IEEE Trans. on Knowledge and Data Engineering, Vol. 16, No. 1, Feb. 1994, pp.86-95.
6. T. Sellis, N. Roussopoulos, and C. Faloutsos, The R+-tree : A Dynamic Index for Multidimensional Object, Proc. VLDB 1987, pp. 507-518.
7. J. Orenstein, Spatial query processing in an Object-oriented database system, Proc. ACM SIGMOD Conf., Washington D. C., May 1986, pp. 326-336.
8. M. Scholl and A. Voisard, Object-Oriented Database Systems for Geographic Applications: an Experiment with O2, Geographic Management System Workshop Proceeding, 1991, pp. 103-137.
9. Wang, Joyce C. C., GIS Major Data Structures VS. Applications, Urban & Regional Information Systems Associations, Annual Conf. Prof., Vol. 2, 1992, pp. 13-24.
10. D. J. Maguire, What is an Object-Oriented GIS?, Proc. of the Fourteenth Annual ESRI User Conf., 1994, pp. 76-89.
11. Max J. Egenhofer, Object-Oriented Modelling: A Powerful Tool for GIS, NCGIA Seminar Workbook, Nov. 1988.
12. L. Haas, and W. Cody, Exploiting Extensible DBMS in Integrated Geographic Information Systems, Proc. of the 2nd Symposium Spatial Database Systems, Aug. 1991, pp.423-449.
13. Stan Aronoff, Geographic Information System : A management perspective, WDL publications, 1989.
14. M. F. Worboy, et al., Object-Oriented Data Modelling for Spatial Databases, Int. J. of GIS, Vol. 4, No. 4, 1990, pp. 369-383.