

OQL/Geo : 지리 정보 시스템을 위한 객체지향 공간 질의어

김 양 희**·김 명 선**·권 석 형**·정 창 성**

OQL/Geo : An object-oriented spatial query language for Geographic Information Systems

Yang-Hee Kim**·Myeong-Sun Kim**·Seog-Hyoung Kwon**
·Chang-Sung Jeong**

요 약

데이터 모델(data model)이란 실세계에 대한 공간 및 비공간적 특성(feature)을 추상화한 시스템 모델이다. 각 시스템들을 고유한 데이터 모델을 통해 외부 세계에 대한 내부 표현 및 외부 세계와의 접속관계에 대한 틀(framework)을 정의하게 되며, 공간 질의어는 지리 정보 시스템에서 정의하고 있는 외부 세계와의 접속 관계를 위한 효과적인 틀의 하나이다. 기존의 지리 정보 시스템에서는 관계형 데이터 모델에 기반한 공간 데이터 모델들이 주로 사용되었으므로, 데이터 추상화(abstraction) 및 상속(inheritance)을 통한 복합 객체의 표현에 문제점이 있었다. 본 논문에서는 ODMG의 객체 모델을 기반으로 평면 위상 모델(planar topological model)을 수용하여 객체지향 데이터 모델인 위상 객체 모델(Topological Object Model : TOM)을 제안하고, 이를 기반으로 객체지향 공간 질의어인 OQL/Geo을 설계하였다. OQL/Geo은 ODMG에 의해 개발된 질의어인 OQL을 기반으로 하여 위상 객체 모델을 효과적으로 표현할 수 있도록 확장하였으며, 기하연산, 위상연산 및 가시연산 등의 풍부한 연산자들을 제공하고 있으며, 복잡한 공간 분석에 대한 요구 뿐 아니라 질의 결과에 대한 출력 형식도 다양하게 표현할 수 있도록 하였다.

ABSTRACT: The data model is a system model which abstracts the spatial and nonspatial features of the real world. A system defines through its data model a framework for the inner representation of and connections with the outside world. The spatial query language is one of the most efficient framework for defining connection with outside world in the GIS. Existing GIS uses a spatial data model based on relational data model. Therefore, it has some difficulties in

* 본 연구는 정보통신부 초고속정보통신망 응용과제(1995)로 이루어진 것입니다.

** 고려대학교 전자공학과(Department of Electronics Engineering, Korea University, 1, 5-ka, Anam-dong, Seoul, 136-701, Korea, Tel. (02)921-0471)

data abstraction and representing complex objects through inheritance. In this paper, we propose an object oriented data model—Topological Object Model(TOM). TOM combines object model in ODMG and the planer topological object. Based on this model, we present an object—oriented spatial query language—OQL/Geo. OQL/Geo extends OQL in ODMG and represents TOM effectively. It also provides several operators such as geometric, topological and visible operators. Moreover, it represents with diverse flexibility the request for complex spatial analysis and presentation of query results.

서 론

지리 정보 시스템(Geographic Information System)은 공간 데이터베이스 시스템(spatial database systems)의 중요한 응용 분야중 하나로서 지도 등의 대용량 공간 데이터 및 관련된 비공간 데이터들에 대한 효율적인 저장과 검색 기능을 제공함은 물론, 이들 데이터에 대한 조작 및 분석을 통해 가공된 정보들을 의사 결정 과정에 효과적으로 이용할 수 있도록 개발된 시스템이다. 이러한 시스템들은 일반적으로 자신이 지니고 있는 고유 특성들이 기반하고 있는 일관된 모델을 지니게 되며, 이를 해당 시스템의 데이터 모델(data model)이라 한다.

데이터 모델이란 실세계에 존재하는 다양한 지형, 지물과 이들이 가지고 있는 공간 및 비공간적 특성들을 추상화한 시스템 모델이다. 각 시스템들은 고유한 데이터 모델을 통해 실체(entity)와 실체간의 관계(relationship) 및 이들로부터 파생되는 특성(feature)에 대한 의미(semantics)를 일관되게 정의할 수 있는 틀(framework)을 제공한다. 사용자들이 시스템에서 제공된 틀을 통해 원하는 정보의 특성들을 표현하고 정보

를 획득할 수 있도록 질의어가 제공되어야 한다.

지리 정보 시스템을 위한 기존의 공간 질의어는 관계형 데이터 모델(관계 모델)을 기반으로 하고 있으며, Pictorial SQL(N. Roussopoulos, et al, 1988), GEOQL(B.Ooi, 1990), Spatial SQL(M.Egenhofer, 1994) 등과 같이 관계 모델에 몇가지의 공간 관계만을 추가한 단순한 형태로부터 시작하여 GeoSAL(Z.Huang, 1993)과 Geo—relational algebra(R.Guting, 1988)등과 같이 기하 관계 및 위상 관계를 체계적으로 모델링하여 관계 모델과 결합시킨 복합 모델의 형태로 발전해왔다. 그러나, 이들 언어들은 관계 모델의 특성상 논리적으로 단일한 객체를 여러개의 테이블로 분할하여 관리하므로 참조 무결성(referential integrity)와 유일성(uniqueness)을 유지하기 어려울 뿐 아니라 복합 응용 객체를 위한 데이터 추상화가 어려운 문제점 등을 안고 있으며, 이러한 문제점들을 효과적으로 해결하고 보다 폭넓은 표현능력을 확보하기 위하여 객체지향 데이터 모델(객체 모델)에 기반한 공간 질의어의 개발이 필수적이다.

본 논문에서는 ODMG(Object Database Management Group)에서 표준안으로 제안

하고 있는 객체 모델(Object Model)(R.G.G. Gattell, 1994)을 기반으로 지형 정보 데이터 베이스의 미국방 표준인 VPF(Vector Product Model)(1992)에서 사용하고 있는 평면 위상 모델(Planar Topology Model)을 수용하여 위상 객체 모델(Topological Object Model : TOM)을 제안하고, 제안된 위상 객체 모델을 기반으로 객체지향 공간 질의어인 OQL/Geo을 설계한다. OQL/Geo은 ODMG에 의해 개발된 질의어인 OQL을 기반으로 하여 위상 객체 모델을 효과적으로 표현할 수 있도록 확장하였으며, 기하 연산 및 위상 연산 기능과 함께 가시적 기능을 함께 고려하여 설계함으로써 복잡한 공간 분석에 대한 요구 뿐 아니라 질의의 결과에 대한 출력 형식 또한 사용자가 원하는 수준에 따라 다양하게 표현할 수 있도록 하였다.

본 논문의 구성은 다음과 같다. 2장에서는 위상 객체 모델(Topological Object Model : TOM)을 정의하고, 3장에서는 OQL/Geo에서 제공하는 지리 정보 연산(geographical operator)들을 정의하고, 4장은 OQL/Geo의 구문 및 기능과 OQL/Geo을 사용하여 표현된 질의문의 예를 보여준다. 마지막으로 5장에서 결론을 맺는다.

위상 객체 모델(Topological Object Model : TOM)

본 장에서는 ODMG에서 표준안으로 제안하고 있는 객체 모델을 기반으로 지형 정보 데이터베이스의 미국방 표준인 VPF에서 사용하고 있는 평면 위상 모델을 기반으로 위

상 객체 모델(Topological Object Model : TOM)을 제시한다.

객체 모델

객체 모델의 기본 단위는 '객체(object)'이며, 각 객체들은 고유한 상태(state)와 행동 양식(behaviour)을 가진다. 이들 객체들은 자신의 상태와 행동 양식에 따라 여러개의 타입(type)들로 분류되며, 주어진 타입에 속한 모든 객체들은 공통된 상태와 행동 양식을 갖는다. 어떤 객체의 행동 양식은 자신이 속한 타입의 객체들상에서 수행가능한 연산(operation)들의 집합으로 정의되며, 객체들의 상태는 성질들의 집합으로 정의된다. 여기서, 성질(property)이란 객체 자신의 속성(attribute)과 여타 객체들과의 사이에 존재하는 관계(relationship)들을 포괄하도록 추상화된 개념이다.

하나의 타입은 하나의 인터페이스(interface)와 하나이상의 구현(implementation)을 갖는다. 인터페이스는 타입에 속한 인스턴스(instance)들에 필요한 외부와의 접속 관계를 정의한다. 반면에, 구현은 타입의 인스턴스들이 물리적으로 표현되는데 필요한 자료구조와 인터페이스에서 정의된 상태와 행동양식을 외부에 가시화하는데 필요한 메소드(method)들을 정의한다.

타입 자체도 하나의 객체이며, 고유한 세 가지 성질을 갖는다. 그 하나는 슈퍼타입(supertype)이며, 이 특징을 통해 타입 객체 간의 계층관계가 유지된다. 두 번째는 익스텐트(extent)로서, 주어진 타입 객체에 속한 모든 인스턴스들은 모두 그 타입의 익스텐트에 속한다. 마지막으로 세번째 성질은 키

(key)이며, 관계형 데이터베이스 시스템에서의 키 개념과 흡사하다. 일반적으로 주어진 타입의 모든 인스턴스들은 성질들의 집합 내지는 한 성질에 할당된 값에 따라 단일하게 구별 가능해야 될 필요가 있으며, 이러한 성질 내지는 성질들의 집합을 키라한다.

하나의 타입은 자신의 인스턴스에 대한 상태와 행동 양식을 정의한다. 여기서, 한 객체의 상태와 행동 양식을 추상화하여 특성(characteristic)이라 한다.

객체는 다시 객체들에게 식별자를 할당해주는 방법에 따라 가변형(mutable) 객체와 리터럴(literal)로 세분된다. 가변형 객체는 자신이 가지고 있는 상태값에 관계없이 고유한 식별자를 가지며, 리터럴은 자신이 가지고 있는 값 자체가 식별자로서 사용된다.

평면 위상 모델

평면 위상은 평면 그래프로서 정의될 수 있으며, 3차원의 연속적인 객체들이 0-셀(노드 : node), 1-셀(에지 : edge), 2-셀(면 : face)과 같은 보다 낮은 차원의 기하 모델들의 조합으로 표현이 가능하다는 논리 위상학의 이론을 바탕으로 하고 있다. 이같은 위상적인 사상관계를 이용함으로써, 시스템은 객체들간에 보다 복잡한 관계를 허용하도록 모델링될 수 있다.

위상 모델의 목적은 각 셀들에 대하여 이웃하고 있는 셀들간에 존재하는 공간적 관계 및 비공간적 관계, 즉 위상 관계에 대한 정보들을 필요에 따라 이끌어내고 유지하는 것이다. 이같은 위상 모델이 유효하게 사용되기 위해서는, 이들 관계에 대한 정보들이

척도, 모양, 크기등의 변화에 관계없이 항상 유지되어야만 한다. 더욱이, 위상 관계 정보가 유지되는 반면에, 공간 객체의 분해과정에서 원 공간 객체의 크기, 모양과 같은 기하 정보들이 손실되어서는 안될 것이며, 평면 위상 모델에서는 공간 객체에 대한 기하학적인 정보들은 변경시키지 않고, 분해된 셀들 사이의 위상 관계들을 정의한다.

위상 객체 모델

위상 객체 모델은 객체모델(Object Model)을 기반으로 평면 위상 모델(Planar Topology Model)을 수용하여 제안한 객체 지향 공간 데이터 모델이다. 위상 객체 모델에서의 객체들은 주제축(thematic axis)과 위상축(topological axis)으로 이루어진 2차원의 객체 공간(object space)상에 존재하며, 모든 객체는 주제축과 위상축의 좌표쌍으로 표현된다.

위상 객체 모델에서는 익스텐트를 세분화하여 주제 익스텐트(thematic extent)와 위상 익스텐트(topological extent)의 두가지 익스텐트를 뒀으로써 객체 공간 개념을 효과적으로 지원한다. 여기서, 객체 모델에서의 익스텐트는 주제 익스텐트에 해당되며, 각 익스텐트는 객체 공간상의 주제축과 위상축을 이룬다. 객체 공간상의 위상축을 이루는 위상 익스텐트는 특정 수준의 위상 정보를 갖는 지도를 추상화한 개념이다.

공간 정보 처리를 위하여 위상 객체 모델에서 지원하는 공간 객체의 기본단위는 노드, 에지, 면, 텍스트이며, 노드는 다시 실체 노드(entity node)와 연결 노드(connected node)로 세분된다. 여기서, 위상 정보를 표

현하기 위하여 사용되는 실 공간 객체는 노드, 에지, 면이며, 텍스트는 지명이나 여타의 지도와 관련된 문자정보를 나타낸다.

위상 익스텐트는 레벨 0에서 레벨 3에 이르는 4단계의 위상 계층을 지원한다. 레벨 0에서는 공간 객체들간에 위상 관계는 정의되지 않으며, 단지 노드와 에지들간의 기하학적인 관계만이 정의된다. 레벨 1과 2에서는 노드와 에지간의 기하학적인 관계만이 아니라, 에지간의 연결성까지도 정의된다. 레벨 1은 비평면 그래프(non-planar graph)를 기반으로 있는 반면, 레벨 2는 평면 그래프(planar graph)를 기반으로 하고 있다. 따라서, 레벨 2에서는 레벨 1에서 제공되는 기능외에 별도로 평면인지 비평면 그래프인지를 확인하는 별도의 일관성 검사 기능을 제공해야 한다. 레벨 3은 노드, 에지, 면의 공간 객체로 이루어지며, 에지간의 연결성에 대한 위상 관계 정보뿐만 아니라 면들의 인접성에 대한 위상 관계 정보까지도 정의된다.

위상 익스텐트는 할당된 위상 레벨에 따라 필요한 인덱스들을 유지한다. 예를 들어, 레벨 0,1,2의 경우는 노드와 에지에 대하여 각각 하나씩의 인덱스를 유지하게 되며, 레벨 3의 경우에는 이외에도 면 객체들에 대한 인덱스가 하나 더 필요하다.

위상 객체 모델에서는 공간상의 좌표를 효과적으로 표현하기 위하여 좌표를 나타내는 문자열을 Coord(coordinate) 타입으로 정의하였으며, 이를 리터럴로서 제공하였다. 또, Node 타입과 Edge 타입 및 Face 타입을 가변형 객체로서 정의하였다. 위상 객체 모델에서 제공하는 타입 계층 구조는 Fig.

1과 같다.

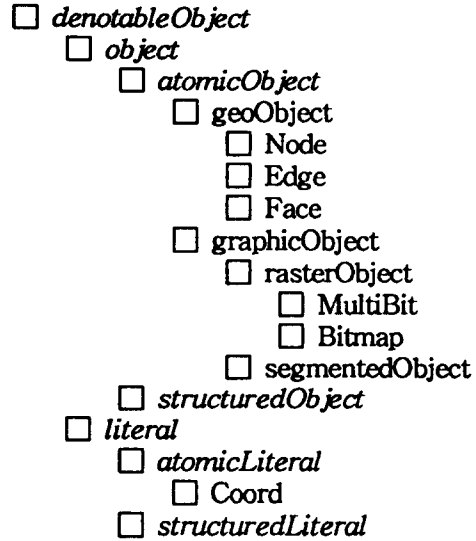


Fig. 1 Type-hierarchy provided by TOM

OQL/Geo의 지리 정보 연산

본 장에서는 복잡하고 다양한 공간 질의를 수행하기 위해 OQL/Geo에서 제공해 주는 여러가지 지리 정보 연산을 그의 특성에 따라 분류 정의한다.

기하 연산(geometric operator)

기하 연산은 공간 객체들 간의 좌표값을 이용하여 정보를 얻어내는 연산으로 위치 연산(location operator), 산술 연산(arithmetic operator), 객체 변환 연산(object transformation operator)과 다항 산술 연산(m-ary arithmetic operator)이로 분류된다.

위치 연산(location operator)

위치 연산은 공간 데이터 처리를 위한 가장 기본적인 연산으로 공간 객체의 위치값을 검색하여 좌표값을 결과로 돌려준다. 위치 연산의 종류는 다음과 같다.

- X_C(N), Y_C(N): 점 N의 x좌표값(x coordinate)과 y좌표값(y coordinate)을 구한다.
- S_P(E), E_P(E): 선 E의 시작점(start point)과 끝점(end point)을 구한다.
- EXT_P(E or F): 선 E나 영역 F의 극점(extreme point)을 구한다.
- EXT_M_P(E, m): 선 E의 m번째 극점을 구한다.
- SEGMENTS(E): 선 E의 모든 선분(segment)를 구하며, 결과는 선분들의 집합이 된다.
- WITHIN(geoObject, r): 공간 객체 geoObject로부터 반경 r내에 있는 공간 객체를 구한다.

산술 연산(arithmetic operator)

좌표값을 이용하여 공간 객체의 길이, 넓이, 거리 및 방향등을 계산하는 연산으로 보다 복잡한 공간분석이 필요한 질의에 이용된다.

- LENGTH(E): 선 E의 길이를 구한다.
- CENTROID(geoObject): geoObject의 중심을 구한다.
- BOUNDARY(F): 영역 F의 경계선의 길이를 구한다.
- AREA(F): 영역 F의 면적을 구한다.
- REGION(E): 선 E의 시작점과 끝점을 연결하여 새로운 영역 F을 만든다.
- COMPLEMENT(F, WINDOW()): 주

어진 WINDOW()에서 영역 F을 뺀 나머지 영역을 구한다.

- DISTANCE(geoObject1, geoObject2): 두 공간 객체 geoObject1과 geoObject2의 직선 거리를 구한다.
- ARC-DIST(geoObject1, geoObject2, L): 두 공간 객체 geoObject1과 geoObject2의 최단 아크 거리를 구하고 그의 경로를 L에 저장한다.
- DIRECTION(geoObject1, geoObject2): x축을 기준으로 두 공간 객체 geoObject1과 geoObject2를 지나는 직선의 방향을 각도로 계산한 후 그 값을 돌려준다.

객체 변환 연산(object transformation operator)

공간 객체의 크기, 위치, 방향등을 변환하여 보다 효과적인 공간 출력을 제공해주는 연산이다.

- SCALE(geoObject, r): 공간 객체 geoObject를 r배 만큼 확대 및 축소한다.
- ROTATE(geoObject, N, r): 공간 객체 geoObject를 점 N을 중심으로 반 시계 방향으로 r만큼 회전시킨다.
- TRANSLATE(geoObject, v): 공간 객체 geoObject에 있는 모든 점에 벡터 v를 더함으로써 공간 객체를 이동시킨다.

다항 산술 연산(m-ary arithmetic operator)

다항 산술 연산은 두개 이상의 공간 객체를 이용하여 정의되며, 공간 객체간의 복잡

한 공간 분석 기능을 수행한다. 이 연산들은 전화시설, 전력 시설, 상하수도 등의 시설물 관리나 산림, 농지 등의 환경관리 분야에 이용된다.

- NEAREST(P*, N2):노드 집합 P*에 있는 점들중 점 객체 N2에서 가장 가까운 거리에 있는 노드 객체를 구한다. Fig. 2에서 이 연산의 사용 예를 보여

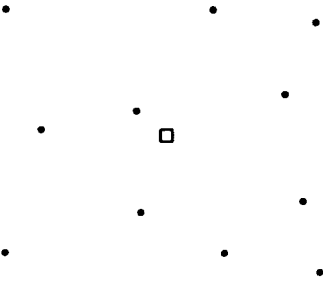


Fig. 2 Example of NEAREST operator

준다.

- FURTHEST(P*, N2):노드 집합 P*에 있는 점들중 점 객체 N2에서 가장 먼 거리에 있는 노드 객체를 구한다.
- CENTER(P*):노드 집합 P*에 있는 노드들의 중심되는 점을 구한다. Fig. 3에서 이 연산의 사용 예를 보여 준다.

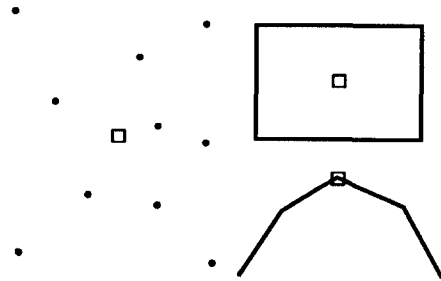


Fig. 3 Example of CENTER operator

위상 연산(topological operator)

위상 연산이란 두 공간 객체간의 위치와 관련된 위상 관계(topological relationship)를 파악 하는데 사용되며, 이 연산들을 사용하여 복잡한 위상관계를 필요로 하는 질의를 처리할 수 있다. 위상 연산은 모두 부울 연산이며, Fig. 2는 두 면 간의 위상 관계를 나타낸다.

- geoObject1 DISJOINT geoObject2:두 공간 객체 geoObject1과 geoObject2가 서로 분리되어 있는지 검사한다.
- geoObject1 MEETS geoObject2:두 공간 객체 geoObject1과 geoObject2의 경계 부분이 만나는가 검사한다.
- geoObject1 EQUALS geoObject2:두 공

간 객체 geoObject1과 geoObject2가 일치하는가 검사한다.

- geoObject1 COVERS geoObject2:공간 객체 geoObject1이 geoObject2를 포함하고, 경계부분에서 만나는가 검사한다.
- geoObject1 COVEREDBY geoObject2:공간 객체 geoObject1이 geoObject2에 포함되며, 경계부분에서 만나는가 검사한다.
- geoObject1 INSIDE geoObject2:공간 객체 geoObject1이 geoObject2안에 있고, 경계부분이 만나지 않는지 검사한다.
- geoObject1 OUTSIDE geoObject2:공간 객체 geoObject1이 geoObject2밖에 있

고, 경계부분이 만나지 않는지 검사한다.

- geoObject1 INTERSECTS geoObject2:
두 공간 객체 geoObject1과 geoObject2가 서로 교차하는지 검사한다.

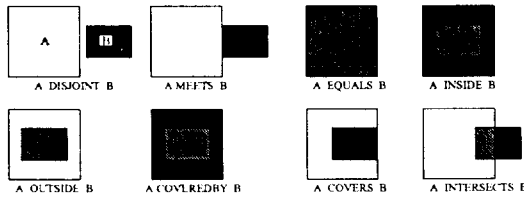


Fig. 4 Topology relationship between two faces

위에서 살펴본 위상 연산자 P는 xPy (x와 y는 피연산자) 형태를 취하며, 만일 모든 (x,y)에 대하여 $xPy=yPx$ 이면 연산자 P는 대칭(symmetric)이라 부르고 그렇지 않으면 비대칭(anti-symmetric)이라 부른다. 위에서 정의된 8가지 연산중 DISJOINT, MEETS, EQUALS, INTERSECTS는 대칭이며 그외는 비대칭이다. 다음 Table 1과 2는 대칭 및 비대칭인 위상 연산과 공간 데이터 타입간의 관계를 보여주며, 'O'은 위상 관계가 존재하는 것을 의미하며, 'x'는 위상 관계가 정의 되지 않음을 의미한다.

Table 1. Relation between symmetric topology operator and spatial object

위상 연산	공간 객체들의 데이터 타입					
	N-N	N-E	N-F	E-E	E-F	F-F
DISJOINT	O	O	O	O	O	O
MEETS	x	x	x	x	O	O
EQUALS	O	x	x	O	x	O
INTERSECTS	x	x	x	O	O	O

N : 노드, E : 에지, F : 면

Table 2. Relation between asymmetric topology operator and spatial objects

위상 연산	공간 객체들의 데이터 타입								
	N-N	N-E	N-F	E-N	E-E	E-F	F-N	F-E	F-F
COVERS	x	x	x	O	O	x	O	O	O
COVEREDBY	x	O	O	x	O	O	x	x	O
INSIDE	x	x	O	x	x	O	x	x	O
OUTSIDE	x	x	x	x	x	x	O	O	O

N : 노드, E : 에지, F : 면

가시 연산(visual operator)

가시 연산은 사용자가 화면을 보면서 마우스와 같은 위치 감식 장치(pointing device)를 사용하여 객체를 선택할 수 있는 연산으로 마우스 연산(mouse operator)과 상수 객체 생성 연산(constant object constructor operator)으로 분류된다.

마우스 연산(mouse operator)

마우스 연산은 전 질의에 의하여 화면에 출력된 지도나 사용자가 질의에 사용될 지도 위에서 키보드 이외에 마우스 등을 통하여 질의에 필요한 객체를 선택할 수 있는 기능을 제공하여 준다.

- PICK(): 질의어 수행 중에 이 연산을 만나면 마우스로부터 한 공간 객체를 입력으로 받아들여 수행을 계속한다.
- POINT(): 질의어 수행 중에 이 연산을 만나면 마우스로부터 점을 입력으로 받아들여 수행을 계속한다.
- LINE(): 질의어 수행 중에 이 연산을 만나면 마우스로부터 선을 입력으로 받아들여 수행을 계속한다.
- WINDOW(): 질의어 수행 중에 이 연산을 만나면 마우스로부터 직사각형 영역

을 입력으로 받아들여 수행을 계속한다.

- CIRCLE(): 질의어 수행 중에 이 연산을 만나면 마우스로부터 원 영역을 입력으로 받아들여 수행을 계속한다.

상수 객체 생성 연산(constant object constructor operator)

상수 객체 생성 연산은 사용자가 원하는 상수 객체를 생성하여 질의에 이용할 수 있는 기능을 제공한다.

- WINDOW(X_coordinate ± X_displacement, Y_coordinate ± Y_displacement): 좌표값을 이용하여 질의문 내에서 직사각형 영역 객체를 선택할 수 있는 연산이다.
- POINT(X_coordinate, Y_coordinate): 좌표값을 이용하여 질의문 내에서 점 객체를 선택할 수 있는 연산이다.
- CIRCLE(X_coordinate, Y_coordinate, r): 좌표값을 이용하여 질의문 내에서 반경이 r인 원 영역 객체를 선택할 수 있는 연산이다.
- LINE(X1_coordinate, Y1_coordinate, X2_coordinate, Y2_coordinate): 좌표값을 이용하여 질의문 내에서 점 좌표 (X1_coordinate, Y1_coordinate)와 (X2_coordinate, Y2_coordinate)를 잇는 선 객체를 선택할 수 있는 연산이다.

객체 생성 연산(object construction operator)

객체 생성 연산은 도시계획, 자원 개발 계획등으로 공간 객체를 나누어 새로운

공간 객체를 생성할 때나 경지계획, 재개발 등의 작은 영역을 합치거나 빼거나 교차구역을 생성할 필요가 있을 때 사용된다.

- CUT(geoObject1, geoObject2): 위상 관계 INTERSECTS가 있는 공간 객체 geoObject2를 확장 객체 geoObject1에 의해 분할한다. 이때 geoObject2는 geoObject1보다 한차수 높다. Fig. 5는 CUT연산의 사용 예를 보여준다.
- UNION(F1, F2): 두 영역 객체 F1과 F2를 합쳐서 새로운 영역을 생성한다. Fig. 6는 UNION의 사용 예를 보여준다.
- DIFFERENCE(F1, F2): 영역 객체 F1에서 F2를 제거하여 새로운 영역을 생성한다. Fig. 7은 DIFFERENCE의 사용 예를 보여준다.
- INTERSECTION(F1, F2): 영역 객체 F1과 F2가 겹치는 영역이 새로운 영역으로 생성된다. Fig. 7은 INTERSECTION의 사용 예를 보여준다.

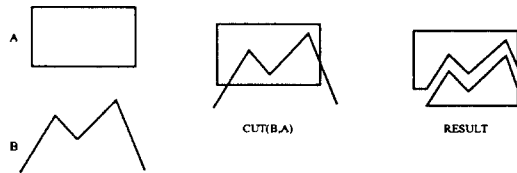


Fig. 5. Example of CUT operator

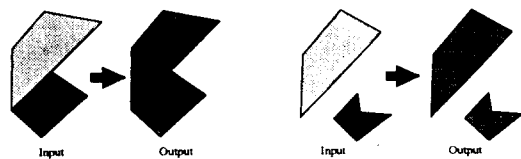


Fig. 6 Example of UNION operator

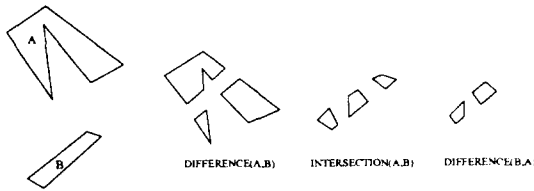


Fig. 7 Example of DIFFERENCE and INTERSECTION

객체지향 공간 질의어

객체지향 데이터 모델로 저장된 데이터의 검색과 조작을 위하여 여러가지 질의어가 개발 되었지만 그 중 객체지향 데이터베이스의 표준으로 제안된 ODMG-93에서 제공하는 질의어인 OQL은 SQL과 유사한 질의어를 제공하고 있어서 사용자가 쉽게 접근할 수 있다. 기존에 많이 쓰이는 질의어를 이용하여 공간 질의어를 작성할 때 다음과 같은 문제점이 발생한다(M.Egenhofer, 1994).

- 1) 공간 데이터의 내부 저장 구조와 무관한 추상적인 공간 데이터 타입과 공간 연산을 제공하여야 한다.
- 2) 질의 처리 결과중 비공간 및 공간 데이터를 선택하여 출력할 수 있는 기능을 제공해야 한다.
- 3) 질의시 공간 데이터의 스키마를 이용할 수 있어야 한다.
- 4) 공간 데이터의 처리 결과를 화면 출력할 수 있는 기능을 제공해야 한다.
- 5) 질의 결과를 원하는 레이어(layer)상에 출력할 수 있어야 한다.
- 6) 사용자가 전 질의 결과를 이용하여 다

양한 질의를 할 수 있어야 한다.

- 7) 키보드이외에 마우스 같은 위치 감식 장치를 이용하여 대화식 질의를 할 수 있어야 한다.
- 8) 사용자가 공간 데이터의 축적(scale)을 선택하고 원하는 축적대로 화면 출력할 수 있는 기능을 제공해야 한다.
- 9) 공간 객체를 화면 출력시 색깔이나 형태 등을 조절할 수 있는 기능을 제공해야 한다.

본 절에서는 위에서 제시한 문제점을 해결하기 위하여 OQL을 확장한 객체지향 공간 질의어인 OQL/Geo을 설계한다.

OQL/Geo의 설계

객체지향 공간 질의어인 OQL/Geo은 앞에서 살펴본 문제점들을 해결하기 위하여 OQL구문을 확장 정의 한다. OQL(Object Query Language)은 SQL과 유사한 객체지향 질의어로 그의 구문은 다음과 같다.

```
query ::= SELECT [DISTINCT] query
        FROM identifier IN query
        {, identifier IN query}
        [WHERE query]
```

OQL/Geo은 OQL구문에 DISPLAY-ON-STORE절을 첨가하여 정의한다. 사용자가 질의시 비공간 속성만을 이용할 경우는 기존의 OQL구문을 그대로 사용하며, 공간 속성을 이용할 경우는 DISPLAY-ON-STORE절을 사용하여 다양한 화면 출력을 할 수 있다. DISPLAY절은 사용자가 객체지향 스키마와 공간 및 비공간 데이터의 질의 결과를 선택하여 출력할 수 있을 뿐 아

니라 질의 결과를 화면상에 다양하게 출력할 수 있는 기능도 제공해 준다. ON절은 사용자가 기본 지도(base map)의 특정 레이어 상에서 질의 결과를 출력하고자 할 때 이용하며 생략시는 출력될 공간 데이터가 저장된 레이어상에 출력된다. STORE절은 질의의 처리 결과를 저장하기 위한 절로 사용자가 질의의 결과를 다시 이용하고자 할 때 사용되며 이 절이 생략되면 temp라는 임시 테이블에 저장된다. OQL/Geo의 구문은 아래와 같다.

```
[DISPLAY <display-opt>
    <display-form-commalist> ]
[ON <layer-name-commalist> :
    <display-cond> ]
[STORE <temporary-storage-name> ]
OQL statement
```

<display-opt>은 질의 처리 결과 중 공간 및 비공간 데이터를 선택하여 출력하는 방법을 지시하며 그 방법은 다음과 같이 세 가지 이다:1)*:공간 및 비공간 데이터를 동시에 출력한다. 이때 비공간 속성이 생략되면 화면 출력될 공간 데이터와 관련된 모든 비공간 속성이 테이블 형태로 출력된다;2) MAP:공간 데이터만 출력한다;3)TBL:비공간 데이터만 출력한다. <display-form-commalist>는 질의 처리 결과의 다양한 화면 출력 방법을 지시하여 주며 그 방법은 다음과 같이 여섯 가지다. 이 절이 생략되면 데이터베이스에 저장된 형태대로 공간 데이터가 출력된다;1)SCHEMA_BROWSER(t):타입 계층이나 데이터베이스 스키마 t를 화면 출력해준다;2)REMOVE(r):질의 처리

결과인 공간 데이터 r을 화면에서 제거한다;3)COLOR(r, c):질의 처리 결과인 r을 c 색으로 출력한다;4)BLINK(r):질의 처리 결과인 공간 데이터 r을 깜빡거리게 출력한다;5)FILL(r, f):질의 처리 결과인 r을 f의 방법으로 채운 후 출력한다;6)SYMBOL(r, "s"):질의 처리 결과인 r을 기호 s로 출력한다. 사용자가 위의 여섯가지 중 SCHEMA_BROWSER의 출력 방법을 지시할때 DISPLAY절 전에 SET SCALE <scale-num>을 이용하여 원하는 축척대로 화면 출력할 수 있다. 이때 <scale-num>는 양의 정수 n을 취하며 1:n 축척으로 화면 출력해 준다.

또한 <layer-name-commalist>는 공간 데이터가 출력될 기본 레이어를 기술하며 <display-cond>는 <layer-name-commalist>와 함께 쓰이며 다음과 같은 질의 출력 모드(display mode)를 취한다:1) NEW:현재 화면을 지우고 질의 결과를 출력한다;2)OVERLAY:현재 화면 위에 질의 결과를 겹쳐서 출력한다;3)SYNC:<layer-name-commalist>에 나타나는 모든 출력 화면의 축척과 위치(location)을 맞춘다. 마지막으로 <temporary-storage-name>은 질의의 결과가 저장될 임시 테이블의 이름을 기술한다.

질의의 예

본 절에서는 앞 절에서 정의된 지리 정보 연산을 포함하는 질의들을 OQL/Geo구문을 사용하여 나타낸다. 질의 예에서 사용되는 레이어와 객체지향 데이터베이스 스키마는 Fig. 8과 같다.

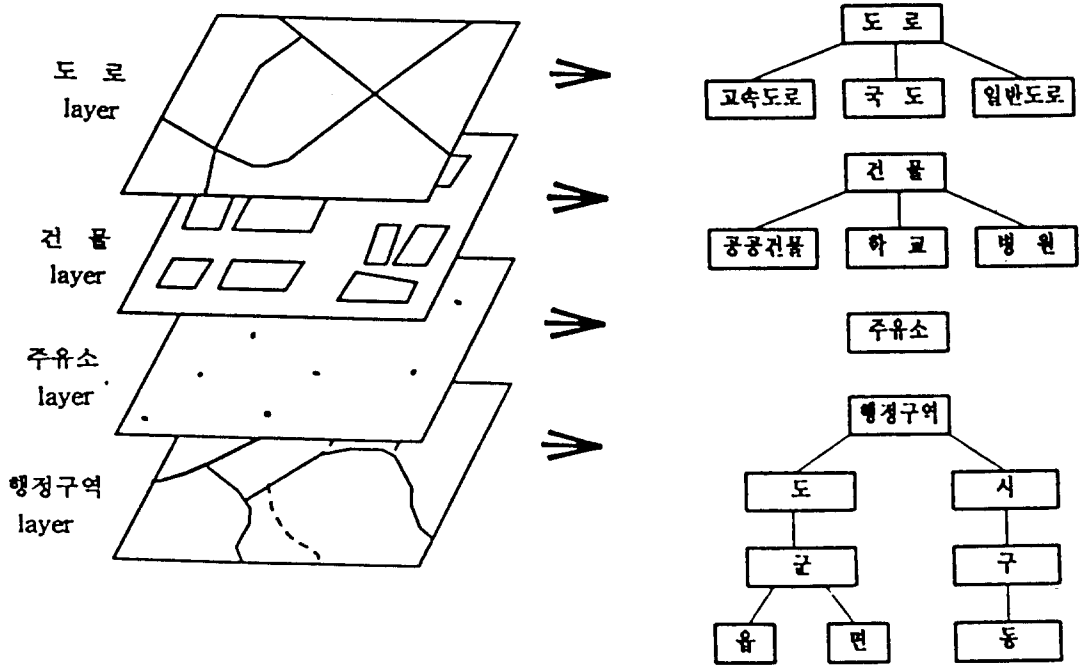


Fig. 8 Layer and OODB schema

만일 사용자가 공간 데이터와 비공간 데이터를 함께 출력하고자 할 때 다음과 같이 질의를 할 수 있다; “안암동 내에 위치한 국민학교와 그곳에서 가장 빠르게 갈 수 있는 병원을 건물지도 위에 출력하여라. 이때 학교는 빨강색, 병원은 깜빡이게, 학교에서 병원까지 가는 길은 노란색으로 출력하고, 학교와 병원간의 거리, 학교이름, 병원이름을 데이터로 출력하고 그 결과를 temp1에 저장하라”

```

DISPLAY* COLOR(a.LOC, "RED"),
        BLINK(d.LOC),
        COLOR(PATH, "YELLOW"),
        len, a.name, d.name
ON      건물, 도로
    
```

```

STORE   temp1
SELECT  a, d
FROM    a IN 학교, d IN 병원
WHERE   a.용도="국민학교"
AND     len=ARC_DIST
        (a.LOC, d.LOC, PATH)
AND     a.LOC INSIDE
        (SELECT b.LOC
FROM    b IN 동
WHERE   b.이름="안암동")
    
```

또한 다음과 같이 전 질의 결과와 마우스를 이용하여 질의를 할 수 있다; “화면에 출력된 전 질의 결과로 부터 마우스가 선택하는 범위 내에 있는 병원을 '#’기호로 출력하여라”

```

DISPLAY SYMBOL(a.LOC, "#")
SELECT a.LOC(SELECT b.LOC
              FROM b.LOC IN temp1
              WHERE b.구분="병원")
WHERE WINDOW()
    
```

결 론

기존에 설계된 많은 공간 질의어가 관계 데이터베이스 시스템의 표준 질의어인 SQL을 이용하였으나 객체지향 모델이 복잡한 공간 데이터를 표현하기에 더 적절하며 그중 SQL과 유사한 OQL은 현재 표준화가 진행중인 객체지향 질의어이다. OQL/Geo은 OQL을 확장한 객체지향 공간 질의어로 앞에서 언급된 9가지 문제점을 다음과 같이 해결한다: 1)공간 데이터 타입의 타입 계층 구조와 다양한 지리 정보 연산을 제공한다; 2)출력시 공간 데이터와 비공간 데이터를 함께 출력 하려면*, 공간 데이터만 출력 하려면 MAP, 비공간 데이터만 출력하려면 TBL을 선택한다;3)SCHEMA_BROWSER를 이용한다;4)DISPALY절을 사용한다;5)ON절을 사용한다;6)STORE절을 이용한다;7)마우스 연산을 사용한다;8)SET SCALE절을 이용한다;9)COLOR, BLINK, FILL, SYMBOL, NEW, OVERLAY 및 SYNC와 같은 출력 형태를 이용한다.

OQL/Geo은 OQL에 DISPLAY-ON-STORE절을 첨가하여 설계 하였으므로 기존의 OQL에서 크게 벗어나지 않아 사용자에게 친숙함을 제공하며, 질의 처리 결과를 다양하게 화면 출력할 수 있는 기능을 제공한다. 또한 다양한 공간 연산과 확장된 구문

을 이용하여 복잡한 의미를 갖는 공간 질의를 간단하게 작성할 수 있는 장점이 있다. 본 논문에서 제시한 OQL/Geo은 현재 구현 중이며 질의어에 친숙하지 못한 사용자를 위하여 그에 대응되는 그래픽 사용자 인터페이스(GUI)도 개발 중이다.

참 고 문 헌

- F. Bancilhon, et al., "The Design and Implementation of O2, and Object-Oriented Database systems", in *Advances in Object-Oriented Database Systems-Proceedings of the 2nd International Workshop on Object-Oriented Database Systems, Germany*, edited by K. Dittrich, LNCS vol. 334. New York, Springer-Verlag, 1988, pp. 323-328.
- R. G. G. Cattell, editor, *The Object Database Standard : ODMG-93 Release 1.1*, Morgan Kaufmann Publishers, 1994.
- B. David, L. Raynal, G. Schorter and V. Mansart, "GeO2 : Why objects in a geographical DBMS ?", in *Proc. of the 3rd Internatinal Symposium SSD*, June 1993, pp.264-276.
- DMA, MIL-STD-6006, *Military Standard -Vector product Format(VPF)*, April 1992.
- M. Egenhofer and A. Frank, "Towards a spatial query language : User interface considerations", in *Proc. 14th Int. Conf. VLDB, Long Beach, CA*, 1988, pp.124-133.

- M. Egenhofer and R. Franzosa, "Point-set topological spatial relations", *Int. J. Geographical Inform. systems*, vol.5, no.2, pp. 161-174, 1991.
- M. Egenhofer, "Why not SQL!", *Int. J. GIS*, 1992, Vol.6, No.2, 71-85.
- M. Egenhofer, "Spatial SQL : A Query and Presentation Language", *IEEE Transaction On Knowledge and Data Engineering*, Vol.6, No.1, 1994, pp.86-95.
- P-C. Goh, "A graphic query language for cartographic and land information systems", *Int. J. GIS*, 1989, Vol.3, No.3, pp.245-255.
- R. Gutting, "Geo-relational algebra : a model and query language for geometric database systems", in *Proc. Int. Conf. Extending Database Technology*, Venice, Italy, 1988, LNCS Vol.303. New York : Springer-Verlag, pp.506-527.
- C. Harris and J. Duhl, "Object SQL", in *Object-Oriented Databases with Applications to CASE, Networks, and VLSI CAD*, Prentice-Hall, 1991, pp.296-312.
- Z. Huang, "Design of GeoSAL: A database language for spatial data analysis", Ph. D Thesis, Royal Institute of Technology, Sweden, 1993.
- Z. Huang, "Designing a Spatial Query Language for Spatial Data Query and Analysis", in the Far East Workshop on Geographic Information Systems, June 21-22, 1993, Singapore, and published in *Proceedings by World Scientific Publishing Co.*
- B. Ooi, *Efficient Query Processing in Geographic Information Systems*, Lecture Notes in Computer Science 471, Springer-Verlag 1990.
- F. P. Preparata and M. I. Shamos, *Computational Geometry: An Introduction*, Springer-Verlag, 1985.
- N. Roussopoulos, C. Faloutsos and T. Sellis, "An efficient pictorial database system for PSQL", *IEEE Trans. on Software Eng.*, Vol.14, No.5, pp.630-638, May 1988.
- J. Wu, T. Chen and L. Yang, "QPF A versatile query language for a knowledge-based geographical information system", *Int. J. GIS*, Vol.3, No.1, 1989, pp.51-57.
- D. S. Park, "The Design and Implementation of a Spatial Query Language for GIS", M.S. Thesis, Inha University, Korea, 1994.
- R. Williamson and J. Stucky, "An Object-Oriented Geographical Information System", in *Object-Oriented Databases with Applications to CASE, Networks, and VLSI CAD*, Prentice-Hall, 1991, pp.296-312.