

## □ 기술해설 □

## GIS를 위한 객체지향 데이터 모델링

서울시립대학교 최웅세\* · 장영권\* · 홍의경\*\*

## ● 목

1. 서 론
2. 지리 정보 시스템의 데이터 특징
3. 객체지향 데이터 모델링
4. 관련 연구
  - 4.1 M. Scholl의 모델
  - 4.2 B. David의 모델
  - 4.3 M. F. Worboy의 모델

## ● 차

- 4.4 L. Mohan의 모델
5. OMT를 이용한 객체지향 지리 정보 모델링
  - 5.1 OMT 기법
  - 5.2 공간 데이터 모델링
  - 5.3 지리 정보 시스템의 모델링
6. 결 론

## 1. 서 론

1970년대 이후로 CAD/CAM, 공학 정보 시스템, 원격 탐사(remote sensing), 환경 모델링, 이미지 데이터베이스들과 같은 새로운 응용 분야의 처리를 위한 수요가 증가되었으며 이러한 응용들을 지원하기 위한 새로운 기술들이 데이터베이스 분야에 필요하게 되었다[1]. 본 논문에서 다루고 있는 지리 정보 시스템도 새롭게 나타나고 있는 응용의 한 분야로서 공간상의 각종 지리 객체의 데이터를 입력, 저장 및 관리하고 목적에 따라 분석 처리하여 결과를 출력하는 시스템을 의미한다. 지리 정보 시스템을 포함한 이러한 새로운 응용들이 처리하는 공통의 데이터는 공간 데이터로서 1차원 구조의 데이터를 주로 다루는 기존의 데이터베이스 시스템으로는 관리하기 어려운 데이터이다.

공간 데이터[2,3,4]는 공간적 좌표값에 의해서 표현되는 데이터로서 점, 선, 면 등의 기본 객체들로 구성된다. 공간 데이터는 기존 데이터의

여러 가지 다른 특성들을 가지고 있으므로 적절한 데이터 표현 방식을 필요로 하며 적절한 데이터 표현을 얻기 위한 방법으로 데이터 모델링을 한다. 데이터 모델링은 “데이터의 의미와 데이터에 가해진 제약 조건을 나타내는 개념적인 도구[19]”라고 할 수 있다. 데이터 모델링은 시스템의 정적, 동적인 성질을 명세화할 수 있으므로 시스템이 처리할 정보의 정확한 저장과 효율적인 처리를 지원하며 사용자가 이해하기 쉽고 확장이 용이한 설계를 지원할 수 있다. 지리 정보 시스템이 처리하는 데이터는 공간 데이터 뿐만 아니라 기존의 데이터베이스 시스템이 처리하는 비공간 데이터도 포함하므로 공간 데이터와 비공간 데이터의 연결을 포함한 단계의 모델링이 필요하다.

지리 정보 시스템을 위한 데이터의 모델은 크게 관계형 모델[5,6,7]과 객체지향 모델[8,9,10,11,12]로 구분된다. 관계형 모델이 폭 넓은 데이터베이스 기술을 지원하지만 공간 데이터의 특징을 적절히 표현하지 못한다. 최근의 연구들은 비구조적, 계층적 데이터의 표현이 용이하며 설계 및 구현 단계에서 발생하는 모델의 불일치

\*준회원

\*\*중심회원

문제를 해결할 수 있는 객체지향 개념을 바탕으로 한 모델을 선택하였다. 따라서 본 논문에서는 기존에 제안된 모델들[8,9,10,11]의 장단점을 분석하여, 객체지향 소프트웨어 방법론인 OMT 기법[13]을 사용한 새로운 모델을 기술한다.

본 논문의 2절에서는 지리 정보 시스템이 처리하는 데이터의 특징을 설명하였고, 3절에서는 객체지향 모델의 특징을 설명하였다. 4절에서는 80년대 후반부터 제안된 객체지향 모델들을 설명하였으며, 5절에서는 본 논문에서 제안한 새로운 모델을 기술하였다. 마지막으로 6절에서는 결론 및 추후 연구 사항에 대해서 논하였다.

## 2. 지리 정보 시스템의 데이터 특징

지리 정보 시스템이 처리하는 데이터는 기존의 데이터베이스 시스템이 처리하는 데이터에 비하여 양도 크고, 복잡한 구조를 가진다. 지리 정보 시스템이 처리해야 하는 데이터는 크게 그래픽 데이터, 비공간 데이터, 공간 데이터로 분류할 수 있다[2,3,4,14]. 그래픽 데이터는 화면상에 출력되는 객체로서 이미지, 지도, 아이콘, 심볼 등을 말한다. 그래픽 데이터는 주로 래스터 방식으로 표현되며 지리 정보 시스템의 사용중에 입출력의 배경으로 사용된다. 비공간 데이터는 지리 정보의 개념적, 비공간적 속성을 갖는 객체를 말한다. 따라서 비공간 데이터는 문자, 숫자와 같이 기존의 DBMS에서 처리할 수 있는 객체이다.

공간 데이터는 기존의 DBMS로 처리하기 어려운 객체로서 2차원상의  $x, y$  좌표값을 사용하여 벡터 형식으로 표현된다. 공간 데이터는 각 데이터간의 관련성 및 계층 구조를 잘 표현할 수 있어야 한다. 이러한 세 가지 데이터의 유형을 바탕으로 지리 정보 시스템이 처리해야 하는 데이터의 특징[2]을 열거하면 다음과 같다. 첫째, 지리 정보 시스템에서 사용하는 정보는 기본 데이터로부터 여러 단계의 추상화가 필요하다. 예를 들면 좌표쌍에 의해서 선이 표현되며, 선들의 모임에 의해서 영역이 표현된다고 할 때, 영역의 면적은 각 단계의 내부 추상화 과정을 거쳐서 구해진다. 둘째, 그래픽 데이터는 비구조적인 자료 형식을 갖는다. 셋째, 내포된 데이터를 갖는다.

즉 하나의 객체가 다른 여러 객체들로 구성될 수 있으며, 다른 객체들의 구성 성분이 될 수 있다. 넷째, 일부 데이터의 경우 순서가 중요한 의미를 지닌다. 예를 들면, 고속도로에서 휴게소의 순서, 철도역의 순서 등은 지리 정보 시스템에서 유용한 정보이다. 다섯째, 데이터간의 복잡한 관계가 존재한다. 예를 들면, 포함 관계, 겹침 관계, 교차 관계 등은 기존의 데이터베이스 시스템에서 처리하는 1차원적인 데이터에서는 나타나지 않는다.

지리 정보 시스템은 이와 같은 특수한 형태의 데이터를 처리하므로 효율적인 데이터 모델을 선택하는 작업이 시스템 설계시에 선행되어야 한다.

## 3. 객체지향 데이터 모델링

지리 정보 시스템을 위한 데이터 모델은 크게 관계형 모델과 객체지향 모델로 구분할 수 있다. 관계형 모델은 기존의 관계 DBMS가 제공하는 회복, 보안, 무결성 제어 등의 기능을 활용할 수 있고, 개발이 용이하며 관계형 모델 자체가 표준화되어 있다는 장점을 가진다. 그러나 관계형 모델은 데이터 표현이 유연하지 못하며 실제계의 복잡한 관계를 쉽게 표현하지 못한다.

객체지향 모델은 아직까지 모델이 표준화되어 있지 않지만 지리 데이터를 처리하는데 있어서 추상 데이터형, 상속성, 다형성 이외에 다음과 같은 장점을 가지고 있으므로 관계형 모델보다 선호되고 있다[2,3]. 첫째, 비구조적이고 복잡한 데이터를 자연스럽게 표현한다. 둘째, 데이터의 계층 구조를 이용한 연산(aggregate 함수)이 용이하다. 셋째, 새로운 함수의 확장이 쉽다. 넷째, 데이터의 무결성 검사가 쉽다. 다섯째, 설계 단계의 모델과 구현 단계의 모델사이 발생되는 불일치 문제를 줄일 수 있다. 따라서 본 연구에서는 객체지향 방식으로 지리 정보 시스템을 모델링하고자 한다.

## 4. 관련 연구

### 4.1 M. Scholl의 모델[8,15]

지리 정보를 표현하기 위해서 이 데이터 모델은 비공간 데이터를 표현하기 위한 map 단계와 공간 데이터를 표현하기 위한 지리(geometric) 단계로 구분하였다.

**4.1.1 map 단계**

map은 비공간 속성들과 하나 이상의 공간 속성을 가지는 릴레이션으로서 공간 속성은 지리 단계에서 정의될 데이터의 식별자를 가진다. 지리 정보를 저장하는 데이터베이스는 map의 집합으로 구성되고, map은 튜플의 집합으로 구성된다. 예를 들어, 숲[이름: 문자열, 주서식식물: 문자열, R: Regions]과 고속도로[이름: 문자열, L: Lines]라는 map 스키마 정의가 있다. 숲이라는 map은 비공간 속성으로 문자열형을 갖는 이름과 주서식식물 속성을 가지며, 공간 속성으로 클래스 Regions의 객체 식별자 R을 가진다. 튜플 <이름: 남산, 주서식식물: 소나무, R: r>은 남산이 소나무가 덮여 있는 숲임을 나타낸다. 이러한 map에 대한 질의는 다음과 같다.

```
SELECT 숲.이름, 숲.주서식식물, 숲. R
FROM   숲, 고속도로
WHERE  고속도로.이름= '경부고속도로'
AND    숲. R border Ladjacent 고속도로. L
```

이 SELECT문은 경부고속도로에 인접한 숲의 세 가지 속성을 검색하는 질의로서 Ladjacent는 지리 속성이 선인 객체들 사이의 인접 관계를 검색하는 메소드이며, border는 영역을 이루는 선을 추출하는 메소드로서 지리 단계에서 정의된다.

지리 속성 R1, R2를 각각 갖는 m1형의 튜플 <A, R1>과 m2형의 튜플 <B, R2>의 조인 연산의 결과로 생기는 튜플 m은 <A, B, R>이며 내부 과정은 다음과 같다[15].

$$m = \{t | t1 \in m1, t2 \in m2, t.A = t1.A, t.B = t2.B, t.R = t1.R1 * t2.R2 \neq 0\}$$

여기서 \*는 지리 단계에서 수행되는 공간 연산자로서 영역간의 교집합을 계산한다.

map 단계에서는 지리 단계의 내부 구조가 숨겨져 있으며 관계형 모델로 표현된다. 따라서

객체의 표현이 쉽고, 관계형 모델의 여러 이점을 사용할 수 있지만 관계형 모델에 따른 여러 제약을 갖게 된다. 또한 이 모델을 지원하기 위해서 하나의 시스템에 관계형 모델을 위한 기법과 객체지향 모델을 위한 기법이 필요하다.

**4.1.2 지리(geometric) 단계**

지리 단계는 공간 데이터를 표현하는 단계로서 객체지향 개념을 사용하며 각 내부 구조는 클래스들로 표현된다. 따라서 지리 단계에서 정의된 내부 속성 및 메소드가 map 단계에서 숨겨져 있으므로 map 단계에서는 공간 데이터들의 내부 구조를 알 필요가 없다. 그림 1은 지리 클래스들의 계층 구조를 나타낸다.

그림 1에서 표현한 것과 같이 모든 map에 공통적인 지리 정보(좌표 시스템, 축척)를 표현하는 클래스가 Spatial 클래스이다.

1) Geos

클래스 Regions(Lines, Points)의 객체들은 Regions(Lines, Points)형의 기하학적 값들을 표현하며, 클래스 Geos의 객체값은 클래스 Geo (Region, Line, Point)의 객체들의 집합이다. 예를 들어, 튜플 <행정구역: 대구, 주작물: 사과, Districts: r>은 대구 안에서 주작물이 사과인 지역을 나타내며, 각 지역들은 클래스 Region의 객체로서 표현된다. 클래스 Region(Line, Point)의 객체들의 집합으로 Regions(Lines, Points)를 모델링하는 이유는 동형의 객체들 사이에 연산들이 단혀 있기 때문이다.

2) Arc

클래스 Arc의 객체는 아크의 끝점들에 의해서 표현되는 선형 세그먼트를 표현하며 끝점과 시작점의 구별은 방향성 아크로 표현한다.

3) Arcs

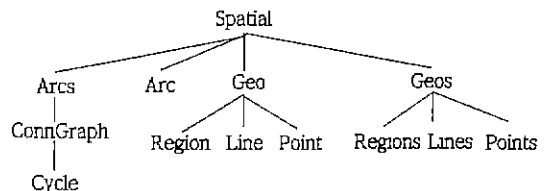


그림 1 지리 클래스의 상속 구조.

클래스 Arcs의 객체값은 클래스 Arc의 객체들의 리스트이다. 클래스 ConnGraph는 클래스 Arcs의 서브클래스이며, Arcs의 속성과 메소드를 상속받고, 아크들의 집합이 연결된 그래프인가를 검사하는 메소드 connected를 추가로 갖는다. connected의 값이 참인 아크들의 집합이 선이다. 클래스 Cycle은 클래스 ConnGraph의 서브클래스이며, ConnGraph의 메소드를 상속받고, 선들의 집합이 사이클을 형성하는가를 검사하는 메소드 Cyclic을 추가로 갖는다. Cyclic이 참인 선이 영역의 외곽선이 된다.

4) Geo

클래스 Geo의 계층은 원자적 객체를 표현한다. 점은 두 좌표에 의해서 표현된다. 선은 클래스 Line의 객체로서 표현되며, 연결된 아크가 선이므로 하나의 속성값이 ConnGraph의 값인 튜플 구조를 가진다. M. Scholl 모델의 지리 단계에서는 공간 객체간의 위상 연산이 부족하며, 위상 관계를 저장하기 위한 자료 구조가 부족하다.

4.2 B. David의 모델[9]

이 모델은 지리 정보를 표현하기 위한 단계를 지리 데이터의 성질을 추상적으로 정의하는 개념적인 단계와 실제 구현에서 적용할 수 있는 내부 단계로 구분하였다.

4.2.1 개념 단계

개념 단계는 E-R 모델을 확장한 의미 데이터 모델과 추상 데이터 타입(ADT)에 의한 국부화(localization) 모델로 구성된다. 의미 데이터 모델은 기존의 E-R 모델에 상속 관계와 propagation 기법을 추가한 것이다. propagation은 계층

화된 구조에서 속성이 계층 구조에 관한 정보를 가짐으로써 계층 구조에 속한 객체에 대한 연산을 편리하게 한다. 국부화 모델은 추상 데이터 기법을 이용하여 내부 단계에서 다루어지는 세부 사항, 즉 내부 단계에서 다루어지는 속성과 연산을 개념 단계에서 숨김으로써 투명성을 제공하도록 한다. 개념 단계는 지리 데이터를 모델링할 때 구현에 신경쓰지 않고 모델링할 수 있는 단계이다.

4.2.2 내부 단계

내부 단계는 실제 지리 데이터를 처리하는 단계로써 스페리티 데이터 구조, 네트워크 데이터 구조, map 데이터 구조 등의 세 구조로 구성된다.

스파리티 데이터 구조는 가장 간단한 구조로서 프리미티브들 간의 위상 관계가 전혀 고려되지 않은 구조이다. 각 프리미티브들은 클래스를 사용하여 정의된다. Point 클래스는 점을 표현하며 x, y 좌표값을 가진다. Line 클래스는 클래스 Vpt, 클래스 Inv 두개의 데이터 구조를 가진 포괄(generic) 클래스이다. 클래스 Vpt는 선을 구성하는 중간점의 좌표 쌍들을 위한 것이며, 클래스 Inv는 Vpt 클래스 객체에 대한 참조로 반대 방향의 선을 위한 것이다. SimpPoly 클래스는 Line 클래스 객체의 집합들로서 면을 구성하며, 하위 클래스 CompPoly는 면내에 다시 면들이 존재하는 면을 표현하는 것이다.

네트워크 데이터 구조는 스페리티 데이터 구조를 바탕으로 선형 위상 관계를 포함하는 구조이다. 이 구조의 프리미티브들은 Node, Dart, SimpArea 클래스이다. 이 구조에서의 핵심은 클래스 Dart로서 하나의 선에 대해서 각 선은 두

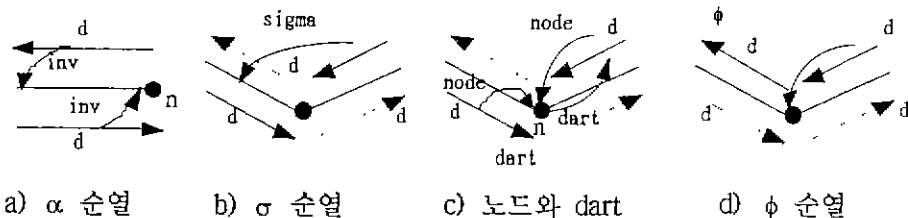


그림 2 dart들 간의 위상 관계, 한 노드와 dart들 간의 위상 관계.

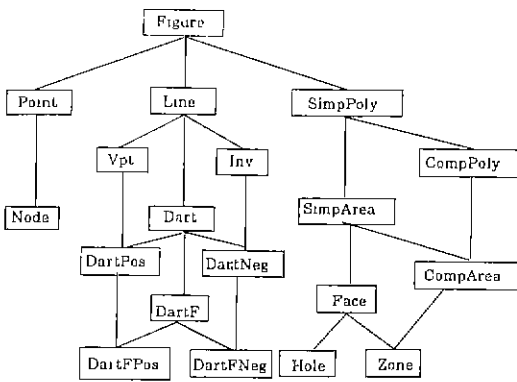


그림 3 map 프리미티브들을 위한 상속 계층 구조.

개의 클래스 DartPos, DartNeg를 가진다. dart는 점과의 연관성을 위하여 dart가 속한 노드를 참조하며, 노드 또한 연관된 dart를 참조한다. 그림 2는 네트워크 데이터 구조에서 사용하는 위상 관계를 나타낸 것이다. 순열은 스파게티의 Inv 클래스를 사용하여 선의 반대 방향을 연결하며, 순열은 끝점을 공유하는 선의 표현을 위해서 사용된다. 그림 c는 dart와 노드 사이의 관계를 표현한 것이다. 클래스 SimpArea는 dart들에 의해서 구성된 면을 표현한다.

map 데이터 구조는 앞의 두가지 구조를 바탕으로 면을 처리하는 구조이다. 이러한 처리를 위하여 map 데이터 구조에서는 topological map

이론을 사용한다. topological map 이론은 면을 구성하는 선을 검색하는 것으로서 앞의  $\alpha, \sigma$  순열의 합 순열  $\phi$ 를 사용한다. 이 구조의 프리미티브로는 클래스 DartF와 클래스 Face가 있다. DartF는 Dart 클래스가 가진 정보 이외에 dart가 속한 면을 가리키는 정보를 가진다. 클래스 Face는 dart로 구성된 면을 표현한 것으로 두개의 하위 클래스 Hole과 Zone을 갖는데, 클래스 Zone은 영역을 포함한 영역을 표현하며 Hole은 Zone이 아닌 영역을 표현하는데 사용된다. 그림 3은 지금까지의 세 가지 구조에 대한 상속 계층 구조를 표현한 것이다. 이 데이터 모델은 위상 관계를 적절히 표현하였으나 계층 구조가 너무 복잡하며, 비공간 속성의 표현 방법 및 공간 속성과 비공간 속성간의 연결 방법을 언급하지 않았다.

### 4.3 M. F. Worboy의 모델[10]

이 모델은 IFO[16]를 사용하여 지리 정보를 표기한다. IFO는 1984년 Abiteboul과 Hull이 제안한 모델로서 확장된 E-R 모델의 성질을 지원하는 객체지향 모델이다. IFO 모델은 printable, abstract, free의 세 가지 기본 데이터형을 가진다. printable(사각형으로 표시)형은 정수, 실수, 문자열 등과 같이 직접 입력과 출력으로 사용되는

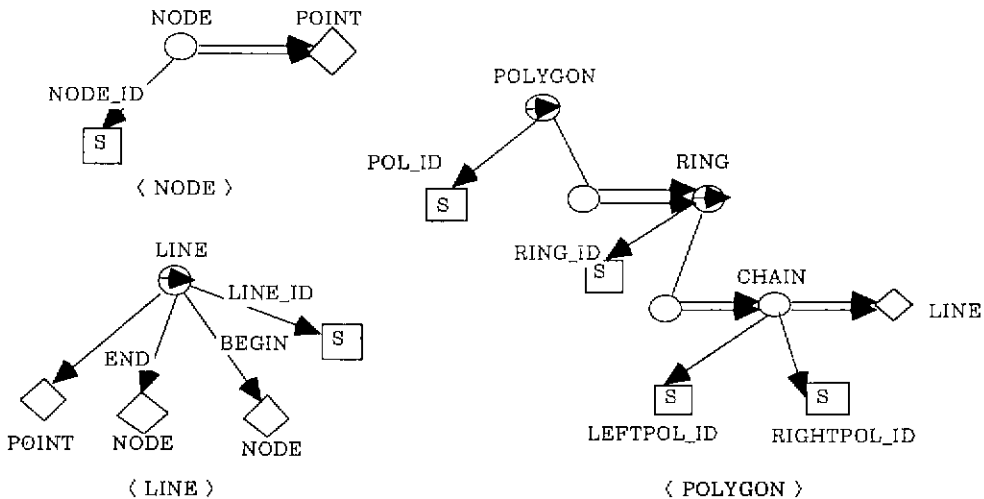


그림 4 IFO를 이용한 점, 선, 면의 모델.

형이다. abstract(다이아몬드로 표시)형은 사람, 도시와 같이 개념적, 추상적인 객체를 표현하기 위한 형이다. free(원으로 표시)형은 generalization, specialization과 같은 관계를 표현하기 위한 링크로서 사용된다.

이 모델에서 사용하는 기본 지리 객체는 미국의 National Committee for Cartographic Data Standard에서 제안한 정의를 기반으로 면을 표현한 것으로 다음과 같다.

- 점(point) : 지도내에 좌표와 식별자를 갖는 0차원 공간 객체
- 선(line) : 선의 시작 노드와 끝 노드를 포함한 순서 있는 점들의 리스트
- chain : 하나 이상의 polygon에 속한 선(좌측과 우측 polygon의 식별자를 포함)
- 노드(node) : 하나 이상의 선이나 chain의 끝점이나 교점
- 링(ring) : 하나 이상의 chain들로 구성
- 면(polygon) : 하나의 외부 링과 0개 이상의 내부 링으로 구성

그림 4는 IFO를 이용하여 노드, 선, 면을 표현한 것이다. 노드는 점 객체의 성질을 상속받는다. 선은 시작 노드와 끝 노드 그리고 여러 개의 중간 점들로 구성된다. 면은 하나 이상의 외부 링과 0개 이상의 내부 노드로 구성되므로 복잡한 면을 표현할 수 있다. IFO를 이용하여 표기한 모델은 관계형 모델을 사용한 모델보다 자연스럽고 효율적이며 의미의 전달이 명확하다.

이 모델은 상속 관계를 사용하여 공간 속성과 비공간 속성을 결합한다. 그러나 이 모델은 비공간 객체간의 계층 구조를 표현하는 점이 부족하며, 공간 객체간의 연산을 제공하지 않는다.

4.4 L. Mohan의 모델[11]

이 모델은 앞의 세 가지 모델과는 달리 지리 데이터의 기본 객체들을 다루는 모델은 아니며, 행정 구역의 표현을 객체지향 개념을 이용하여 표현함으로써 객체지향 모델의 이점을 보여준다.

객체지향 개념을 사용한 표현은 공간 구조의 계층 구조를 쉽고 자연스럽게 표현한다. 영역

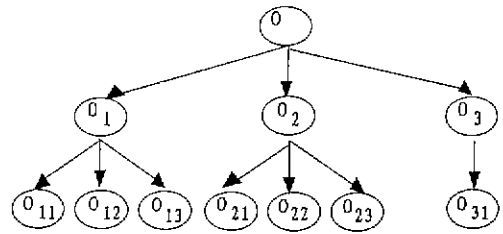


그림 5 공간 객체의 계층 구조.

Object Subclass C	Object Subclass C <sub>1</sub>
Name: 0	Name: O <sub>1</sub>
Component_of: NULL	Component_of: 0
Comprised_of: O <sub>1</sub> , O <sub>2</sub> , O <sub>3</sub>	Comprised_of: O <sub>11</sub> , O <sub>12</sub> , O <sub>13</sub>

그림 6 comprised\_of와 component\_of를 사용한 객체의 표현.

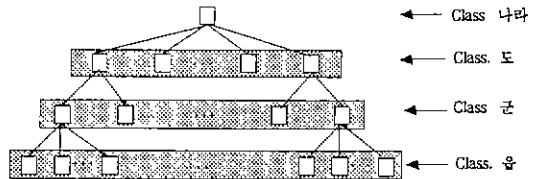


그림 7 공간 단계에 따른 그룹핑.

O가 세 개의 하위 영역 O<sub>1</sub>, O<sub>2</sub>, O<sub>3</sub>로 구성되고, 각 하위 영역들이 다시 하위 영역들로 구성된다면 그림 5와 같은 계층 구조로 표시된다. 그림 5는 공간 영역간의 계층 구조로서 각 단계는 서로 다른 클래스의 인스턴스들이다. 예를 들어서 O는 클래스 C의 인스턴스이고, O<sub>1</sub>, O<sub>2</sub>, O<sub>3</sub>은 클래스 C<sub>1</sub>의 인스턴스이며, O<sub>11</sub>, O<sub>12</sub>, ..., O<sub>32</sub>는 클래스 C<sub>2</sub>의 인스턴스이다. 객체들 사이의 관계는 component\_of와 comprised\_of 속성을 사용하여 그림 6과 같이 표현할 수 있다.

그림 5에서 표현한 것과 같이 객체 O<sub>1</sub>은 클래스 C<sub>1</sub>의 인스턴스이고, 상위 객체 O의 일부분이 되며, 객체 O<sub>11</sub>, O<sub>12</sub>, O<sub>13</sub>의 하위 객체들로 이루어져 있음을 나타낸다. 이러한 표현은 객체지향의 컬렉션 타입(집합, 배열, 리스트 등)과 내포 타입에 의해서 자연스럽게 표현되며, 이러한 구조를 사용하여 새로운 클래스의 생성과 다중 상

속이 가능하다. 그림 5의 예에서 객체  $O_{11}$ ,  $O_{12}$ ,  $O_{21}$ ,  $O_{23}$ 이 공통 특성  $P_A$ 를 가지고, 객체  $O_{21}$ ,  $O_{23}$ ,  $O_{31}$ 이 공통 특성  $P_B$ 를 가지면, 각각의 공통 특성을 가지는 새로운 클래스  $C_{PA}$ ,  $C_{PB}$ 를 생성할 수 있다. 앞의 구조를 이용하여 행정 구역을 나타내고자 할 때, 나라, 도, 군, 읍 등의 네 개의 기본 클래스가 필요하다. 그림 7은 이 모델을 사용하여 행정구역을 트리 구조로 나타낸 것이다.

### 5. OMT를 이용한 객체지향 지리 정보 모델링

지리 정보는 크게 공간 데이터와 비공간 데이터로 구분된다. 비공간 데이터는 기존의 DBMS로 처리할 수 있는 문자열, 숫자 등과 같은 형태이지만, 공간 데이터는 2차원 공간에서 표현되므로 공간 데이터를 표현하기 위한 특별한 자료 구조가 필요하다. 본 연구에서 제안하는 모델은 객체지향 소프트웨어 방법론인 OMT [13]를 사용하여 객체지향 개념을 기반으로 비공간 데이터와 공간 데이터를 표현하였다.

#### 5.1 OMT 기법[13]

OMT는 소프트웨어 시스템을 개발하는 소프트웨어 생명 주기의 전 과정에 객체지향 개념을 적용한 방법론이다. OMT는 기존의 여러 방법론들의 장점들을 객체지향 개념으로 통합한 형태로써, 객체 모델(object model), 동적 모델(dynamic model), 기능 모델(functional model) 등의 세 모델들을 기초로 하고 있다. 주로 객체 모델에 중점을 두고 있으나 어떤 모델에 비중을 두느냐에 따라 다른 방법론이 되기도 한다. OMT 기법은 여러 객체지향 방법론 중에서 좋은 기법으로 평가받고 있다[17,18]. 객체 모델링시에 EER (Exhended Entity Relationship) 모델링 기능을 수용할 수 있도록 고안되었기 때문에 목적 시스템의 분석 기능이 뛰어나다. 또한 자동화하기에 편하도록 표기법이 정의된 점도 장점이다.

OMT 기법의 객체모델 표기법을 간단하게 설명하면 다음과 같다. 클래스는 그림 8에서 보는

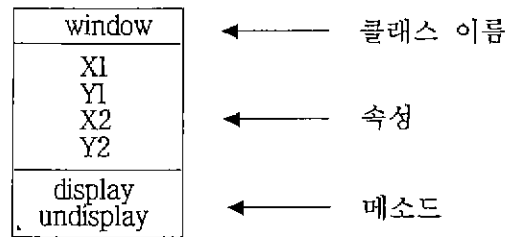


그림 8 OMT 객체모델에서의 클래스 표현.

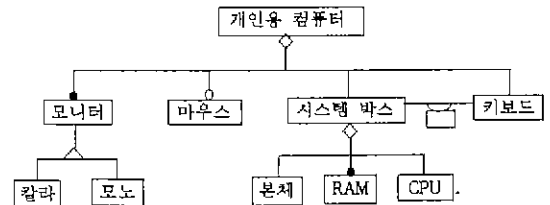


그림 9 OMT 객체 모델에서의 is\_a, part\_of 관계 표현.

것과 같이 박스가 세 등분되어 있고, 위에서 부터 차례로 클래스 이름, 속성, 메소드들을 표현한다. 그림 8은 클래스 이름이 window이고 x1, y1, x2, y2의 속성을 가지고 있으며 메소드로 'display', 'undisplay'를 가지고 있음을 보여준다.

part\_of 관계는 '∩'으로 표기하고 multiplicity는 '∩', '∪'로 표기한다. '∩'는 part가 0개 이상 있어야 함을, '∪'은 0 혹은 한개의 part가 있음을 의미한다. is\_a 관계는 '⊃'로 표기하고, 삼각형 모양의 윗쪽에 존재하는 클래스를 슈퍼 클래스, 아래쪽에 존재하는 클래스를 서브클래스라 한다. '⊃'는 클래스간의 관계를 표현하기 위한 클래스를 표현한다. 그림 9는 개인용 컴퓨터의 구성 요소를 is\_a, part\_of 관계로 표기한 예이다.

개인용 컴퓨터는 모니터, 마우스, 시스템과 키보드가 있어야 함을 나타낸다. 모니터는 컬러 모니터와 흑백 모니터가 있다. 시스템과 키보드는 관계 클래스에 의해서 관계를 표현한다.

#### 5.2 공간 데이터 모델링

공간 데이터의 가장 기본이 되는 객체는 점, 선, 면이며 세 가지 기본 객체의 위상 관계 및





래스로 가진다. POLYLINE 클래스는 위상 관계를 표현하기 위해 선의 왼쪽 영역과 오른쪽 영역을 표현하는 속성을 가진다. POLYGON 클래스는 영역을 표현하며, POLYLINE 클래스의 인스턴스들로 구성된다. POLYGON 클래스는 COMPLEX\_POLYGON 클래스를 서브클래스로 가진다. COMPLEX\_POLYGON 클래스는 복잡한 영역, 즉 영역내에 속한 영역을 표현하며, 포함된 영역을 속성으로 가진다. 클래스 내에 길이, 면적, 둘레 등을 속성으로 표현한 이유는 객체에 대한 사용자의 입력 시간이 컴퓨터 처리 속도에 비하여 현저히 느리므로 입력시에 속성의 값을 계산하여 저장하는 것이 효율적이다.

GEOS 클래스는 GEO 클래스의 점, 선, 면의 집합이 갖는 성질을 표현한다. 예를 들어, 여러 개의 선이 연결되어 하나의 큰 선을 구성하는 경우, 여러 개의 점, 선들이 공통적인 성질을 갖게 되어 그룹화되는 경우이다. 이렇게 동형 객체들의 집합을 하나의 인스턴스로 처리함으로써 각각의 객체가 갖고 있는 연산을 일일이 수행할 필요가 없으므로 연산의 투명성을 제공할 수 있다. 동형이 아닌 객체 사이의 연산은 점, 선, 면(점들, 선들, 면들) 사이의 관계 클래스에서 정의된다.

그림 10의 클래스에서 정의된 메소드들은 속성으로 저장하기에는 부적당한 성질을 메소드로 정의하였다. 공간 메소드에 대한 정의를 간략히 설명하면 다음과 같다.

- LOCATE: 객체의 좌표값을 출력
- SPLINE: 주어진 좌표쌍을 이용하여 곡선을 생성
- DISJOINT: 객체를 일정 기준에 의해서 분리
- ADJACENT: 인접한 객체의 검색
- NEAR: 일정 거리내에 있는 객체의 검색
- CLOSEST: 인접하지 않은 가장 가까운 객체의 검색
- DIRECTION: 객체의 일정 방향에 위치한 객체의 검색
- INCLUDED: 객체내에 포함된 객체의 검색
- INCLUDING: 객체를 포함한 객체의 검색
- OR, XOR, NOT: 면 객체 사이의 연산

- CENTER: 면 객체의 중심점 생성
- CONNECTED: 선 객체 집합의 연결성 검사
- SHORTEST\_PATH: 최단 경로 탐색

### 5.3 지리 정보 시스템의 모델링

공간 데이터가 앞의 그림 10과 같은 계층 구조를 갖는 것과 마찬가지로 비공간 데이터도 계층 구조를 가질 수 있다. 그러나 공간 데이터의 구조가 지리 정보 시스템이 사용하는 모든 응용에 의해서 공유되는데 비하여 비공간 데이터의 구조는 사용자의 처리 대상 데이터와 응용의 용도, 대상에 대한 사용자의 관점에 따라 다르게 정의될 수 있다. 따라서 사용자가 정의하는 비공간 데이터의 구조는 정확히 고정될 수 없는 것이다. 그림 11은 SPATIAL 구조를 바탕으로 지리 정보 시스템을 모델링한 것이다. 그림 11의 최하위 단계에 있는 GEOGRAPHIC-ELEMENT 클래스는 비공간 속성을 가지며 공간 속성을 가진 SPATIAL 클래스를 상속받는다. 이 GEOGRAPHIC-ELEMENT 클래스의 객체가 지도상에서 다루어지는 의미 있는 최소 단위이다. 이 모델에서는 객체지향의 상속 개념을 사용하여 공간 데이터와 비공간 데이터를 연결시키므로, 관계형 모

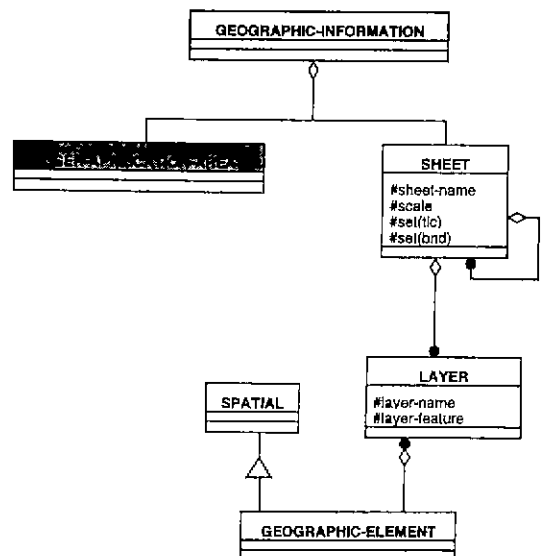


그림 11 지리 정보 시스템의 모델링.

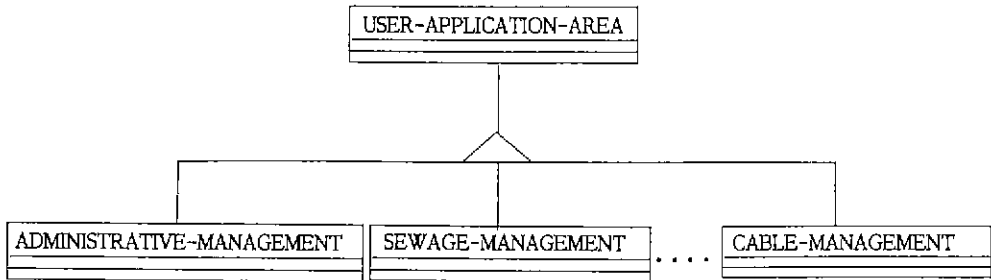


그림 12 User-Application-Area의 계층 구조.

텔에서 포인터를 사용하여 연결할 때에 발생하는 문제점들을 해결할 수 있다. SPATIAL 클래스와 마찬가지로 GEOGRAPHIC-ELEMENT 클래스도 비공간 속성의 내용에 따라 클래스 계층을 형성할 수 있다. 이 GEOGRAPHIC-ELEMENT 클래스의 객체들 중 같은 성질을 가진 객체들의 집합이 layer 클래스를 이루며 레이어 객체들이 모두 결합되어 지도의 도면을 이룬다. 레이어는 지도상에서 개념적으로 동일한 객체의 그룹이며, 오버레이(overlay), 커버리지(coverage), 주제(theme)라고 부른다. SHEET 클래스의 객체인 도면(sheet)은 지도를 일정 간격으로 분할한 단위로 정의되며, 이 도면들이 결합되어 다시 도면을 생성할 수도 있다.

그러나 이러한 도면 자체가 완벽한 지리 정보 시스템의 처리 대상이라고 말할 수는 없다. 지리 정보는 지도 자체의 정보 이외에 각 응용마다 가지고 있는 응용의 성질을 표현해야 한다. 예를 들면, 행정 구역을 처리해야 하는 곳에서는 행정 구역에 관한 계층 지역 정보를 갖고, 전화선, 하수도 등의 경우에는 각각의 독특한 성질에 의하여 다른 계층 지역 정보를 가지고 있으므로 이러한 계층 지역 정보 등을 포함한 응용의 특성을 설명하여야 하며 특정 응용의 사용자는 자신의 응용에 맞는 정보만을 처리하면 된다.

그림 11의 USER-APPLICATION-AREA 클래스 안에서 각 응용에 관한 정보를 표현하며, USER-APPLICATION-AREA 클래스의 객체와 SHEET 클래스의 객체가 결합하여 완전한 지리 정보를 표현할 수 있다. USER-APPLICATION-AREA의 계층 예는 그림 12와 같다.

## 6. 결 론

지리 정보시스템을 비롯한 최근의 많은 응용 시스템들이 다루고 있는 공간 데이터는 이차원 상의 객체를 대상으로 하므로 기존의 데이터베이스 시스템이 데이터를 처리하는 것과는 다른 기술들을 필요로 한다. 본 논문에서는 객체지향 개념을 바탕으로 하여 점, 선, 면 등의 공간 데이터를 모델링하였다. 공간 데이터들 사이에는 많은 위상 관계들이 존재하며, 위상 관계들을 속성과 연산(메소드)중 어느 것으로 처리할 것인가가 중요한 요소가 되며 더욱 많은 연구가 필요하다. 또한 공간 데이터는 기존의 SQL과 같은 질의어로는 처리가 부적합하므로 새로운 질의어의 연구도 이루어져야 한다.

지리 정보 시스템이 처리하는 데이터는 공간 데이터 이외에 비공간 데이터도 포함하므로 두 가지 데이터를 효율적으로 연결하는 것이 시스템의 성능에 많은 영향을 미친다. 본 논문에서는 객체지향의 상속 관계를 이용하여 두 가지 데이터를 연결하였으며, 연결된 데이터를 기본으로 전체 지리 정보 시스템의 모델링을 제안하였다. 지리 정보 시스템의 경우 데이터의 입력이 시스템 구축의 상당 부분을 차지하므로 기존의 데이터를 활용하는 방법도 연구되어야 한다.

## 참고문헌

- [1] W. Kim, Modern Database Systems: The Object Model, Interoperability and Beyond. Addi-

son-Wesley, 1995.

[2] S. C. Wang and C. Lee, "Extensions to Object-Oriented Data Models for GIS Applications," Proc. Far East Workshop on GIS(FEGIS), Singapore, pp. 7-20, 1993.

[3] M. F. Worboy, "Object-Oriented Approaches to Geo-Referenced Information," Int'l J. of GIS, Vol. 8, No. 4, pp. 385-399, 1994.

[4] C. M. Medeiros and F. Pires, "Databases for GIS," ACM SIGMOD RECORD, Vol. 23, No. 1, pp. 107-115, 1994.

[5] T. C. Waugh and R. G. Healey, "The GEOVIEW Design: A Relational Database Approach to Geographical Data Handling," Int'l J. of GIS, Vol. 1, No. 2, pp. 101-108, 1987.

[6] J. A. Orenstein, "PROBE Spatial Data Modeling and Query Processing in an Image Database Applications," IEEE Trans. on Software Engineering, Vol. 14, No. 5, pp. 611-629, 1988.

[7] W. G. Aref and H. Samet, "Extending a DBMS with Spatial Operations," Proc. 2nd Symp. on Spatial Databases, SSD91, pp. 299-318, 1991.

[8] M. Scholl and A. Voisard, "Object-Oriented Database Systems for Geographic Applications: an Experiment with O<sub>2</sub>," Geographic Management System Workshop Proceedings, pp. 103-137, 1991.

[9] B. David, et al., "GeO<sub>2</sub>: Why Objects in a Geographical DBMS?," Proc. 3rd Symp. on Spatial Databases, SSD93, pp. 264-276, 1993.

[10] M. F. Worboy, et al., "Object-Oriented Data Modeling for Spatial Databases," Int'l J. of GIS, Vol. 4, No. 4, pp. 369-383, 1990.

[11] L. Mohan, and R. L. Kahyap, "An Object-Oriented Knowledge Representation for Spatial Information," IEEE Trans. on Software Engineering, Vol. 14, No. 5, pp. 678-691, 1988.

[12] O. Gunter and W. F. Riekert, "The Design of GODOT: An Object-Oriented Geographic Information System," IEEE Data Engineering Bulletin, Vol. 16, No. 3, pp. 4-9, 1993.

[13] J. Rumbaugh, et al., Object Oriented Modeling and Design, Prentice Hall, 1991.

[14] S.-C. Wang and C. C. Joyce, "GIS Major Data Structures vs. Applications," Proc. Annual Conf. Vol. 2, Urban & Regional Information Sys-

tems Associations, 1992.

[15] M. Scholl and A. Voisard, "Thematic Map Modeling," Proc. 1st Symp. on Spatial Databases, SSD89, pp. 167-190, 1989.

[16] S. Abiteboul and R. Hull, "IFO: A Formal Semantic Database Model," ACM Trans. on Database systems, Vol. 9, No. 4, pp. 525-565, 1987.

[17] D. D. Champeaux and P. Faure, "A Comparative Study of Object-Oriented Analysis Methods," J. of Object-Oriented Programming, Vol. 5, No. 1, pp. 21-33, 1992.

[18] D. E. Monarchi and G. I. Puhr, "A Research Typology for Object-Oriented Analysis and Design," Communications of the ACM, Vol. 35, NO. 9, pp. 35-47, 1992.

[19] 이태식, 성효현 공역, GIS 입문, 미래건설연구소, 1994.

**최 응 세**



1994 서울시립대학교 전산통계학과 졸업 (B.S)  
 1994 ~ 현재 서울시립대학교 전산통계학과 석사과정  
 관심 분야: 객체지향 데이터베이스, 공간 데이터베이스

**장 영 권**



1994 서울시립대학교 전산통계학과 졸업 (B.S)  
 1994 ~ 현재 서울시립대학교 전산통계학과 석사과정  
 관심 분야: 객체지향 데이터베이스, 공간 데이터베이스

**홍 의 경**



1981 서울대학교 사범대학 수학교육과 졸업 (B.S)  
 1983 한국과학기술원 전산학과 졸업 (M.S.)  
 1991 한국과학기술원 전산학과 졸업 (Ph.D.)  
 1984 ~ 현재 서울시립대학교 전산통계학과 부교수  
 1994 ~ 현재 과기처 GIS 기술분과 위원회 위원  
 관심 분야: 분산 데이터베이스, 공간 데이터베이스, 컴퓨터 시스템 성능평가