

□ 기술해설 □

## 객체 지향 규칙 기반 지형 질의어의 설계 및 구현

부산대학교 김진덕\* · 선종복\* · 문상호\* · 홍봉희\*\*

● 목	차 ●
1. 서 론	5.3 기능적인 측면의 요건
2. 관련연구	6. 지형 질의어(GeoQL)의 설계
3. 객체 지향 규칙 기반 GIS 구조	6.1 공간 연산자와 공간 관련성 표현자의 메소드 정의
4. 지형정보의 모델링	6.2 공간의사결정을 위한 규칙 정의
4.1 관계 데이터 모델과 객체 지향 데이터 모델의 비교	6.3 GeoQL의 설계
4.2 객체 지향 기법을 이용한 계층구조의 표현	7. 질의 처리기의 구현
4.3 객체 지향 공간 데이터 표현 방법	7.1 GeoQL 질의 처리기의 구조
5. 지형 질의어의 요구조건	7.2 분할 기법에 의한 질의 처리
5.1 질의 유형에 따른 요건	8. 지형질의어 사용을 위한 그래픽 사용자 인터페이스의 설계 및 구현
5.2 사용상의 편의를 위한 요건	9. 결 론

### 1. 서 론

지형 정보 시스템(GIS : Geographic Information System)은 대용량의 지형정보를 입력 저장한 뒤 관련된 공간 데이터와 속성 데이터를 분석하여 지형정보를 검색하는 시스템이다[EGEN 88, ESRI92, WANG92]. 관계 데이터 모델을 이용하는 지형 정보 시스템은 벡터 데이터의 입력이 간단하고 테이블 조작이 간단하지만, 지형정보의 계층 구조, 다양한 관련성 등을 인위적인 테이블로 표현하기 때문에 계층 구조의 지형 정보 모델링이 어렵고[MOHA88, WANG92], 이로 인해 복합 객체의 표현이 어려운 문제가 발생한다. 그리고 테이블에 표현된 지형 정보나 관련성을 검색하기 위하여 사용자가 일일이 테이블 간의 관련성을 토대로 조인과 같은 SQL 프로그래밍을 해야 하는 문제가 있다.

기존 GIS의 또다른 문제로서는, 지금까지의 GIS는 공간 연산, 공간 데이터간의 공간 관련성을 표현하는 기능 및 단순 속성 검색 기능은 뛰어나지만 많은 인력과 비용을 투자하여 입력한 속성 및 공간 데이터를 이용한 분석기능은 미흡하다는 점이다. 그래서 현재 사용되고 있는 GIS는 구조화된 문제(structured problem)만을 해결할 수 있는 “정보위주의 GIS(information-oriented GIS)”라 불린다[김은형93]. 반면 현재 GIS 응용 분야는 단순한 공간적 문제나 속성정보의 검색에 국한되지 않는다. 앞으로 GIS가 나아가야 할 방향은 계획이나 설계와 같은 비정형 문제(non-routine or ill-structured problems)의 해결을 지원할 수 있는 “개념 위주의 GIS(conceptualization-oriented GIS)”라 할 수 있다. 계획이나 설계와 같은 공간 의사 결정(Spatial Decision Making)[DENS91, ZHU92] 문제는 다양한 공간 데이터의 관련성 및 속성 데이터간의 관련된 정보를 이용할 뿐만아니라 전문가의 노하우

\*준회원  
\*\*정회원

(knowhow)를 이용하여 해결해야 할 경우가 많다. 이를 위해서는 GIS의 기능에 전문가 시스템의 기능을 추가하는 것이 필요하다.

이 논문에서는 이러한 기존 GIS S/W의 문제점을 해결하기 위한 객체 지향 규칙 기반 GIS (Object-Oriented Rule-Based GIS)를 제시한다. 객체 지향 규칙 기반 GIS는 기존의 관계 데이터 모델의 한계를 극복하기 위하여 객체 지향 기법을 이용하여 지형 정보를 모델링하며, 공간 의사 결정을 지원하기 위하여 지형 질의어에서 규칙을 호출할 수 있는 방법을 제시한다. 이 논문에서는 공간 의사 결정을 효율적으로 지원하기 위해 특정 응용의 전문가 지식을 규칙으로 표현한다. 이러한 규칙 추론의 결과를 계획이나 설계와 같은 비정형문제의 해결에 이용하기 위하여 추론의 결과를 지형질의어의 결과로서 보여주는 방식을 채택하였다. 규칙의 추론은 주어진 문제에 따라 전향 체인 방식과 후향 체인 방식을 이용하였으며, 이와 같은 방식으로 최적의 의사 결정을 내릴 수 있도록 하였다.

이 논문에서 제시하는 객체 지향 규칙 기반 지형 질의어인 GeoQL(Geographical Object-oriented Query Language)은 지형 정보 객체의 메소드로 표현된 공간연산자를 질의어의 Select나 Where구문내에서 직접 호출할 수 있도록 하였다. 그리고 지금까지는 규칙 추론을 위해 별도의 구문을 사용하였지만 이 논문에서는 SQL형식의 GeoQL 구문에 규칙을 호출하는 기능을 내장하였다. 그러므로 사용자는 속성 데이터 검색, 공간 데이터 연산, 규칙 추론 등을 동일한 인터페이스(Uniform Interface)로 수행할 수 있다. 그리고 규칙 추론의 결과를 사용자가 질의어에서 사용할 수 있도록 하였다. 규칙 추론에서 발생하는 처리결과와 불일치 문제(질의처리는 객체의 집합을 다루지만 규칙 추론 결과는 사실(Fact)의 집합)를 해결하는 방법을 제시한다. 이러한 규칙 실행과 질의어 실행간의 불일치 문제는 규칙 추론 유형(전향 체인 방식, 후향 체인 방식)에 따라 다르게 해결되어야 한다. 따라서 GeoQL 질의 처리는 속성 데이터를 처리하는 텍스트 검색 모듈, 공간 데이터를 다루는 메소드 처리 모듈, 규칙 처리 모듈로 분리되어 실행된다.

그래서 입력된 질의어를 분할하여 해당 모듈로 보낸다음 처리 결과를 다시 병합하는 분할 기법을 이용하였다.

이 논문의 구성은 다음과 같다. 2장에서는 관련연구에 대해 기술하고, 3장에서는 이 논문에서 설계, 구현하고 있는 객체 지향 규칙 기반 지형 정보시스템의 전체구조에 대해 설명을 하고, 4장에서는 질의처리기와 밀접한 관련이 있는 지형정보의 모델링에 대해서 설명을 한다. 그리고 5장에서는 지형 질의어가 갖추어야 할 기본조건에 대해, 6장에서는 지형 질의어의 설계 내용을 기술한다. 특히 공간 의사 결정의 정의 및 GeoQL내에서의 규칙 추론 호출 방법을 기술한다. 그리고 7장에서는 질의처리의 구현 방법에 대해 기술하고, 구현과정을 예를 들어 설명한다. 8장에서는 지형 질의를 위한 GUI에 대해 설명하고, 9장에서 결론을 맺는다.

## 2. 관련연구

GIS와 전문가 시스템을 통합하여 공간 의사 결정을 지원하는 방법은 사용자가 서로 다른 두 개의 시스템을 배워야 하는 문제점이 있다. [ZHU92]는 공간 문제에 대한 의사 결정을 지원하기 위하여 기존의 GIS 시스템과 전문가 시스템의 통합에 관한 연구로서 ARC/INFO와 전문가 시스템인 CLIPS를 통합하여 구현하였다. 공간 의사 결정을 위해 ARC/INFO에 구축된 지형 정보를 데이터 파일로 CLIPS에게 넘겨주면 관련된 규칙을 추론한다. 그리고 결과를 파일로 다시 ARC/INFO에 넘겨 출력하게 한다. 그러나 이 방법은 느슨한 통합(loosely coupled)방법으로서 GIS와 전문가 시스템간의 데이터 교환이 자동화되지 않았고 사용자의 개입이 필요로 하는 불편한 점이 있다. 그러므로 사용자는 속성 데이터 검색 및 공간 데이터 연산을 위한 ARC/INFO의 명령어를 숙지함과 동시에 CLIPS의 규칙 추론 명령어를 숙지해야 하는 문제점이 있다. 그리고 CLIPS의 규칙 추론 결과가 단순 텍스트형식의 파일로 전송이 되므로 ARC/INFO에서 그 결과를 이용한 질의가 불가능하다. 따라서 공간 의사 결정 기능과 GIS기능이 강결합된 시

스택이 바람직하다.

객체 지향 SQL을 GIS에 적합하게 확장한 [SCHO91]은 OODBMS인 O<sub>2</sub>을 이용하여 점, 라인, 다각형과 같은 기본적인 공간데이터를 클래스로 선언하고 각 클래스에 적용가능한 공간 연산자를 메소드로 구현하였다. 그리고 상위 클래스를 두어 공통적으로 적용되는 공간 연산자를 메소드로 선언하고 하위 클래스에서 상속받아 사용할 수 있도록 하였다. 객체 지향 시스템을 기반으로 한 연구에서는 공간 연산자를 클래스의 메소드로 정의하는 것은 일반적인 현상이다. 새로운 공간 연산자의 추가는 클래스의 메소드로만 정의하면 되므로 질의 처리기의 수정없이 쉽게 확장이 된다. 이 연구는 객체 지향 시스템을 기반으로 하고 공간 연산자를 클래스의 메소드로 정의하여 질의어에서 사용할 수 있다는 점에서 GeoQL과 유사하다. 그렇지만 이 연구에서는 속성 데이터와 공간 데이터로 구성된 사용자 정의 추상화 객체의 정의기능과 공간의사 결정을 위한 규칙 추론의 기능이 포함되어 있지 않다.

관계 데이터 모델을 기반으로 한 SQL 확장형 지형 질의어에 관한 연구로는 Spatial SQL [EGEN94]와 PSQL[ROUS88]이 있다. 첫째, 연산자의 확장성을 고려하면 Spatial SQL은 관계 데이터베이스를 기반으로 공간 연산자를 관계 연산자로 선언하였으므로 새로운 공간 연산자 추가시 관계 DBMS내에 연산자를 정의해야하므로 DBMS 커널을 수정해야 하는 문제점이 있다. 반면 PSQL은 ADT를 이용하여 공간 데이터 타입을 정의하고 공간 연산자를 ADT내의 Operator로 정의하였다. 그래서 새로운 공간 연산자 추가시 해당 데이터 타입에 Operator를 추가함으로써 쉽게 확장이 된다. 둘째, 공간 연산자 선택의 편의성을 고려하면 Spatial SQL과 PSQL 모두 공간 연산자의 과부하(Overloading)를 허용하지 않으므로 연산자 선택의 어려움이 있다. 예를 들어 두 객체 간의 거리를 구하는 "Distance"라는 공간 연산자를 생각해 보자. 다음과 같은 두개의 연산자가 필요하다고 가정한다. 점 객체(Point)와 다각형 객체(Polygon)사이의 거리를 구하는 연산자 P\_Distance(Point, Polygon)가 있고, 점 객체와 선으로 구성된 객체(Arc)사

이의 거리를 구하는 연산자 A\_Distance(Point, Arc)가 있다고 가정하자. 절차 언어(Procedural Language)에서는 연산자의 과부하를 지원하지 않는다. 그러므로 이와 같은 경우 두 객체사이의 거리를 구하는 동일한 역할을 수행하는 연산자 일지라도 적용되는 인수가 다르기 때문에 서로 다른 이름의 함수로 구현해야 한다. 따라서 사용자는 적절한 공간 연산자의 선택에 어려움을 겪게 된다.

### 3. 객체 지향 규칙 기반 GIS구조

객체 지향 규칙 기반 지형 정보 시스템의 전체적인 구조는 그림 1과 같다. 먼저 지형 정보 객체의 생성에 필요한 공간 데이터의 입력은 기존의 상용 디지털입력 모듈이나 벡터링 모듈을 이용하고, 입력된 공간 데이터는 화일에 저장되는 것을 가정하고 지형 객체 생성 모듈에서 이용한다. 입력된 공간 데이터를 이용하여 Point, Arc, Polygon과 같은 도형 요소 객체를 생성한다. 실제로 이러한 기능을 담당하는 것은 지형 객체 생성 모듈의 세부 모듈인 매핑 객체 모듈[문상호94a]이 담당한다.

지형 객체 생성 모듈[김진덕94a]은 지형 정보 객체를 생성하는 모듈로서 객체 지향 모델링 구조에 따라 객체를 생성하고 저장한다. 이 모듈은 공간 데이터를 기반으로 지형 정보 객체를 상향

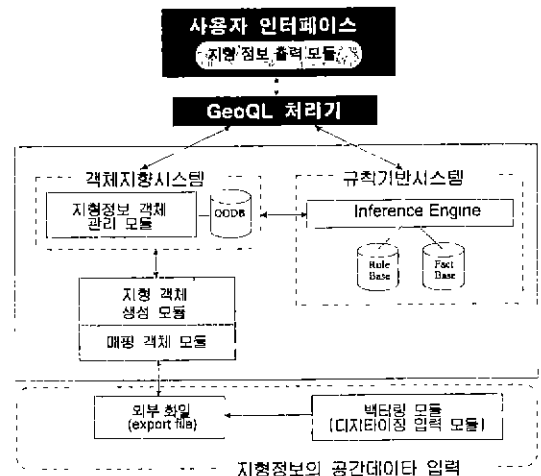


그림 1 객체 지향 규칙 기반 GIS 전체 구조

식으로 생성하는 방법을 이용하고 있으며, 생성된 지형 정보 객체는 Object Base에 저장된다.

규칙기반 시스템은 GeoQL에서 정의한 각종 관련성 규칙추론과 공간 의사 결정을 담당한다. 이는 규칙 및 fact들을 저장하는 Knowledge Base 부분, 규칙을 추론하는 추론 엔진(Inference Engine) 부분으로 구성되어 있다.

GeoQL 처리기[김진덕94b]는 지형 정보 검색을 위하여 사용자가 입력한 GeoQL 문장을 실행한다. 이 질의 처리기는 지형 정보 검색과 공간 의사 결정을 위하여 Object Base와 Rule Base와 연결되어 있으며, 검색된 지형 정보의 출력을 위하여 그래픽 사용자 인터페이스내의 출력 모듈과 결합된다.

그래픽 사용자 인터페이스는 GeoQL에 의해 검색된 지형 정보를 화면에 출력하는 것이다. 지형 질의어를 위해 GUI가 필요한 이유는 지형 질의어만으로 출력에 필요한 환경 설정 및 지원과 같은 다양한 기능을 제공할 수 없기 때문이다. 그래픽 사용자 인터페이스는 1) 사용자의 질의에 의해 검색된 지형 정보를 화면에 출력하는 출력 모듈, 2) 사용자의 질의를 받아들여 질의 처리기로 보내는 기능을 하는 질의 입력부, 3) 다양한 형태의 지형 정보 출력을 위한 환경 설정부, 4) 객체 브라우저 등으로 구성된다.

지형 객체 생성 모듈, 지형 정보 관리 모듈등에 대한 구체적인 기술은 생략하기로 하고 여기서는 지형 정보의 객체 지향 모델링, GeoQL 처리기와 그래픽 사용자 인터페이스를 중심으로 자세히 설명하고자 한다.

#### 4. 지형정보의 모델링

데이터 모델링의 기준은 응용 프로그램에 적용될 데이터의 유형을 표현할 수 있는지 없는지를 나타내는 모델링 파워, 보다 쉽게 표현이 가능한가를 나타내는 모델링 효율, 응용에 따른 질의를 잘 수용하는지를 나타내는 질의의 유용성 등이 있다. 그러므로 GIS 구축시 이런 기준들을 고려하여 데이터 모델을 선정해야 한다. GIS용 데이터는 정형적인 속성 데이터와 비정형적인 공간 데이터로 구성이 된다. 이들 데이터는 서로

연결되어 있어야 한다. GIS를 위한 데이터 모델링은 크게 관계 데이터 모델을 기반으로하여 공간 데이터를 처리할 수 있도록 확장한 경우와 객체 지향 데이터 모델을 이용하는 경우로 구분할 수 있다. 4.1절에서는 각 데이터 모델의 장, 단점을 자세히 살펴보기로 한다. 4.2절에서는 이 논문에서 제시하는 객체 지향 지형 정보 모델링에 대해 구체적으로 기술한다.

#### 4.1 관계 데이터 모델과 객체 지향 데이터 모델의 비교

지금까지 Arc/Info와 같은 상용 GIS에서는 주로 관계 데이터 모델을 이용하여 왔으며 최근 객체 지향 데이터 모델을 이용한 시스템이 등장하고 있다. 여기서는 관계 데이터 모델과 객체 지향 데이터 모델의 장단점[WANG92]을 살펴보기로 하자.

관계 데이터 모델의 장점은 ① 기본 속성데이터를 테이블형태로 쉽게 구축할 수 있고, ② RD-BMS의 데이터 보안, 복구, 트랜잭션 로그 기능 등을 이용할 수 있으며, ③ 표준화된 질의어인 SQL을 제공하여 검색의 편의 및 유연성을 제공한다는 점이다.

반면 이 모델의 단점으로는 RDBMS가 고정된 길이의 레코드만을 지원하므로 가변형의 공간 데이터 타입을 다루기 어렵다. 그래서 보통 공간 데이터를 위한 처리기를 따로 두는 확장형 RDBMS를 이용하게 된다. 관계 데이터 모델은 복합 지형정보 객체를 표현하기 어렵다.

객체 지향 데이터 모델의 장점은 ① 복합 계층 구조 및 상속 계층 구조를 자연스럽게 지원하여 복합적인 지형 정보 객체를 표현하기 쉽고 클래스 간의 상속이 가능하다. 그러므로 자연 세계를 보다 알기 쉽게 구조화할 수 있어 사용자의 질의도 보다 편리해진다. ② 네트워크 데이터 모델과 유사하여 데이터 검색이 보다 효율적이다. 반면 관계 데이터 모델은 조인 오버헤드가 있다. ③ 가변 길이의 데이터 저장을 허용하므로 공간 및 속성 데이터를 하나의 데이터베이스에 저장, 관리할 수 있으며, ④ 공간 의사 결정을 위한 전문가 시스템과의 연계가 쉽다는 장점이 있다.

표 1 관계 데이터 모델과 객체 지향 데이터 모델의 비교 [WANG92]

데이터 모델	적당한 응용	데이터베이스 구축
관계 데이터 모델	기본 속성 데이터의 Ad-hoc Query가 잦은 응용 프로그램	쉬움
객체 지향 데이터 모델	복합 지형 정보 객체에 대한 빠른 검색을 요구하는 응용 프로그램	어려움

그 이유는 객체 지향 시스템은 전문가 시스템을 위한 추론 엔진을 클래스로 캡슐화(Encapsulation)하기 때문이다. 따라서 추론은 객체의 메소드를 호출로서 처리된다.

반면 객체 지향 데이터 모델의 단점은 단순히 테이블만을 생성하여 입력하는 관계 데이터 모델에 비해 데이터 구조의 생성 및 데이터 베이스의 구축이 어렵다는 점이다.

두 모델의 장단점을 응용과 데이터베이스 구축 측면에서 간단히 요약하면 표 1과 같다.

#### 4.2 객체 지향 기법을 이용한 계층구조의 표현

지형 정보 모델링에 있어서 지형 정보간의 복잡하고 다양한 계층 구조를 적절하게 표현하고 필요에 따른 검색이 가능해야 한다는 점에서 계층 구조의 표현은 매우 중요하다. GIS는 응용 분야에 따라 지형 정보간의 다양한 계층 구조가 있다. 계층 구조에는 시, 도 등과 같은 행정 구역 단위 등을 포함하는 물리적인 구조와 유사성 또는 종속성을 갖는 지형 정보를 효율적으로 관리하기 위하여 정의하는 논리적 구조가 있다. 그리고 실제 GIS에서 분석, 통계 등 많은 업무가 계층 구조를 기본으로 한다. 따라서 이러한 지형 정보간의 계층 구조를 쉽게 표현할 수 있고, 효율적인 검색이 가능한 모델링 방법이 필요하다 [WANG92].

관계 데이터 모델에서 계층 구조를 표현하기 위해 해당 지형 객체를 나타내는 테이블에 상위 지형 정보를 나타내는 애트리뷰트를 사용한다 [WANG92]. 그리고 이 애트리뷰트를 검색함

으로써 계층 구조의 검색이 가능하다. 그러나 이 방법의 문제점은 계층 구조를 나타내기 위하여 테이블내에 인위적인 애트리뷰트로 표현하기 때문에 자연스럽게 못할 뿐만아니라 테이블간의 관계가 매우 복잡해 진다[WANG92]. 또한 사용자가 계층 구조의 각 지형 정보를 검색하려면 계층 구조를 표현하는 테이블들간의 애트리뷰트를 완전히 이해해야 할 뿐만아니라 계층 구조 검색을 위해서는 사용자가 직접 연관된 테이블들간에 join 연산을 해야 하는 문제점이 있다.

그러므로 계층 구조 표현에 있어서 이러한 관계 데이터 모델의 문제점을 해결하기 위하여 객체 지향 지형 정보 모델링을 이용하는데, 다양한 지형 정보와 계층 구조를 표현하기 위하여 제시된 복합 계층 구조는 그림 2와 같다.

도면의 관리를 위하여 mesh로 분할된 각 mesh 도면을 그림 2와 같이 도면 객체로 표현한다[田靖彦90]. 그리고 각 도면 객체는 레이어 객체로 구성된다. 레이어 객체는 세그먼트 객체로 구성되고 세그먼트 객체는 Point, Arc, Polygon의 기본 도형 요소 객체로 구성되는 복합 계층 구조로 표현된다. 세그먼트 객체[田靖彦90]는 하나의 의미있는 단위이며 속성 값을 부여할 수 있는 기본단위가 된다. 따라서 ARC로 된 도형 요소 객체가 배수관이라는 세그먼트 객체가 될 때 사용자는 더이상 선의 연결인 ARC로 보지 않고 매설년도, 관경, 유량을 가진 배수관으로 인식하게 된다. 세그먼트 그룹 객체는 여러가지 의미있는 세그먼트가 모여 또 다른 의미를 가지는 집합체를 일컫는다. 그리고 세그먼트 그룹 객체는 몇 계층이라도 확장이 가능하다. 세그먼트 그룹 객체를 지원함으로써 지형 정보의 계층

적 표현이 가능해진다. 레이어 객체는 일정 지형 범위 이내에서 동일한 의미를 가지는 세그먼트 객체들의 집합이며 도면을 일정한 크기의 사각형으로 분할하여 관리할 경우 각 사각형을 도면 객체라고 볼 때 각 도면 객체는 여러개의 레이어 객체로 구성이 된다[田靖彦90].

그림 2에서 도형요소 객체는 도면상에서 표시되는 공간 데이터에 해당하는 객체로서 벡터 정보와 위상 값을 가지고 있다. 이 도형 요소 객체를 바탕으로 사용자 정의 추상화 지형 객체인 세그먼트와 세그먼트 그룹 객체를 정의함으로써 사용자는 다양한 상위 지형 정보를 정의할

수 있다. 상수도 도면의 경우 배수관, 급수관, 제수변 등은 세그먼트 객체로 표현되고 소블럭, 중블럭, 대블럭은 세그먼트 그룹 객체로 표현된다.

그림 2에서 복합 계층 구조를 구성하는 각각의 지형정보 클래스는 GIS 응용에 따라 여러가지 서브 클래스(subclass)로 정의할 수 있으며, 서브 클래스는 상위 클래스(superclass)로부터 공통적인 속성과 메소드를 상속받는다. 따라서 사용자는 지형 정보를 객체로 표현하기 위한 클래스를 쉽게 정의할 수 있고 확장이 용이하다. 그림 3은 상수도 관리에서 지형 정보를 표현하기 위해 정의된 상속 계층 구조이다. 그림 3에서 최상위 클래스로 정의된 지형정보 클래스에는 각 하위 클래스들에 공통적으로 적용되는 슬롯 및 메소드를 선언하였다. 그래서 각 하위 클래스는 지형정보 클래스에 정의된 슬롯 및 메소드를 그대로 사용할 수 있다.

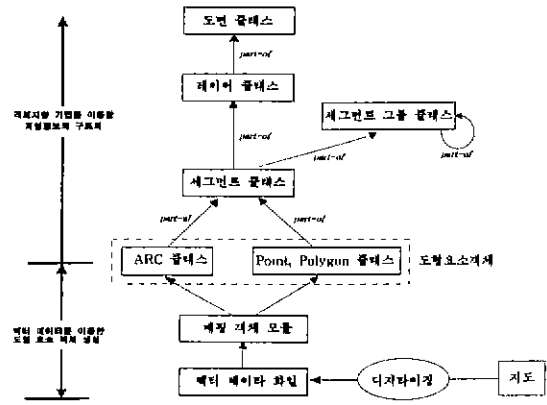


그림 2 객체 지향 기법을 이용한 지형 정보의 계층 구조

### 4.3 객체 지향 공간 데이터 표현 방법

레이어 개념을 기본으로 한 기존의 상용 GIS인 ARC/INFO[ESRI92]에서의 공간 데이터 저장은 다음과 같은 방식을 사용하고 있다. 먼저 원도(Source Map)를 일정간격의 정방형으로 구분한

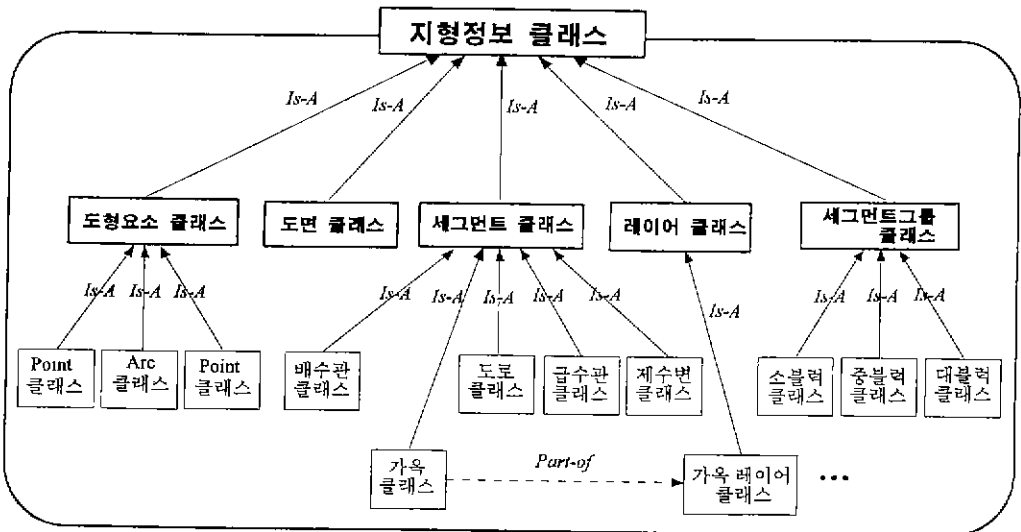


그림 3 지형 정보의 상속 계층 구조

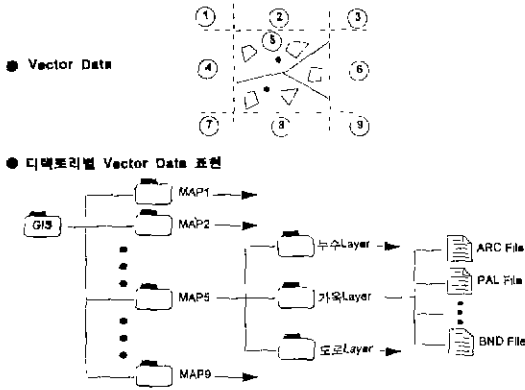


그림 4 ARC/INFO 공간데이터 표현

다음 각 정방형별로 하나의 디렉토리를 할당한다. 그리고 각 정방형은 다시 여러개의 레이어로 구분이 된다. 그래서 각 디렉토리에 각각의 레이어에 해당하는 서브디렉토리를 두어 해당하는 공간 데이터 및 속성 데이터를 화일 형식으로 저장한다. 그림 4에서는 원도를 9개의 정방형으로 분리한 다음 각 정방형에 해당하는 디렉토리를 할당하며(MAP1, MAP2, ..., MAP9), 각 디렉토리는 또다시 서브디렉토리를 갖는다. 그림에서는 "MAP5" 디렉토리가 "누수Layer", "가옥Layer", "도로Layer"를 서브디렉토리로 갖는다. 이것은 MAP5가 누수지점에 관한 정보를 다루는 누수Layer, 가옥의 정보를 포함하는 가옥Layer, 도로정보를 가지는 도로Layer로 구성됨을 의미한다. 그리고 각 Layer 디렉토리는 공간데이터를 저장한 ARC 화일, 속성 데이터를 저장한 PAT 또는 AAT 화일, 경계정보를 저장한 BND화일 등등으로 구성된다.

이러한 저장방식은 정방형으로 나눈 지역별로 데이터를 관리하므로 특정 지역만을 대상으로 검색하거나 질의할 때는 효율이 뛰어나다. 그러나 실제 원도에서는 이웃한 정방형의 공간데이터가 저장구조상으로는 서로 다른 디렉토리에 존재한다. 그러므로 인근 지역을 검색할 때는 사용자가 디렉토리 이동 작업을 해야 하고 검색 효율도 떨어진다. 전체 지역의 특정 지형 정보, 예를 들어 도로망을 검색하고자 할 때도 모든 도로 레이어 디렉토리를 검색하여 화면상에 출력해야 한다. 그리고 정방형의 분할선상에 존재

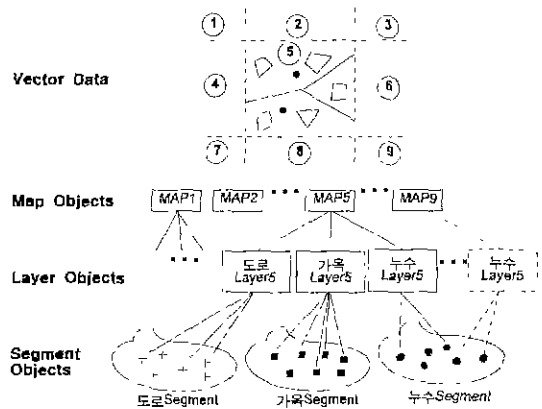


그림 5 객체 지향 공간데이터 표현

하는 공간 데이터는 실제로 하나의 지형정보를 나타내지만 저장 구조상에는 분할되어 2개 이상의 지형정보로 저장된다.

한편 이 논문에서는 그림 2에서 제시한 클래스 계층 구조를 이용하여 지형정보를 표현하고자 한다. 그림 5에 나타난 바와 같이 원도를 구분하여 나눈 정방형을 MAP 클래스의 인스턴스(MAP1, MAP2, ..., MAP9)로 표현하였고, 각 인스턴스는 레이어 클래스의 인스턴스들(도로Layer5, 가옥Layer5, 누수Layer5)을 구성요소로 갖게 된다. 그리고 이런 레이어 객체(Layer Object)들은 레이어를 구성하고 있는 세그먼트 객체(Segment Object)들을 구성요소로 갖게 된다. 세그먼트 객체는 속성 데이터를 슬롯으로 저장하며 공간데이터를 지정하는 포인터(Pointer)를 가진다. 이러한 세그먼트 객체는 기존 ARC/INFO의 저장 방식과는 달리 공간데이터 및 속성 데이터를 저장한 세그먼트 객체가 분산되어 저장하지 않고 하나의 객체 베이스(Object Base)에 저장된다. 각 레이어 객체는 구성요소인 세그먼트 객체를 지정하는 포인터를 슬롯값으로 갖는다. 그러므로 특정 레이어에 포함되는 모든 객체를 검색하고자 할 경우 이 슬롯값을 이용하여 구성요소인 세그먼트 객체를 액세스하는 것이 용이하다.

이러한 저장방식은 레이어 클래스의 객체가 레이어를 구성하는 모든 세그먼트 객체의 포인터를 가지고 있다. 그러므로 특정 레이어의 구

성요소를 검색하고자 할 때는 세그먼트 객체의 포인터를 따라 검색할 수 있다. 이 레이어 클래스의 객체가 그림 4의 각 레이어 디렉토리 역할을 한다. 따라서 이 방식은 레이어 개념을 기본으로 하는 저장방식의 장점인 데이터의 지역별 관리를 지원한다. 또한 레이어 개념 저장방식의 단점인 전체 지역 검색, 인근 지역 검색을 쉽게 할 수 있다. 인근 지역 검색은 인근 지역의 정방형을 나타내는 객체의 이름으로 검색이 되고, 전체 지역 검색은 해당하는 세그먼트 클래스의 인스턴스를 모두 출력으로 쉽게 검색이 된다. 예를 들어 전체 지역의 도로망을 검색하고자 한다면 도로 세그먼트 클래스의 인스턴스를 검색함으로써 전체 도로망을 화면상에 출력시킬 수 있다. 그리고 정방형의 분할선으로 인한 지형정보(세그먼트 객체) 자체의 분할이 되지 않는다. 왜냐하면 분할선상의 객체(세그먼트 객체)는 그대로 두고 레이어 객체에서 레이어를 구성하는 세그먼트에 대한 포인터를 포함하기 때문이다. 이 때 분할 선상의 객체를 양쪽에서 공유할 수도 있고, 많이 치우친 한쪽 레이어 객체에만 표현할 수도 있다.

## 5. 지형 질의어의 요구조건

지형 정보 시스템에 적합한 지형 질의어 [EGEN88, EGEN94, ROUS88, SCHO91, TANA88, WEBS90]의 기본 요구조건은 크게 질의 유형에 따른 조건, 사용상의 편의를 위한 조건, 기능적인 측면에서의 지형 질의어가 갖추어야 할 요구조건으로 분류할 수 있다.

### 5.1 질의 유형에 따른 요건

지형 정보 시스템의 질의는 1)속성 데이터만을 검색하여 텍스트 형식의 속성 데이터만을 출력하는 경우, 2)공간 데이터만을 검색하여 공간 데이터를 화면상에 출력하는 경우, 3)공간 데이터와 속성 데이터 모두를 이용하여 검색한 뒤 검색 결과로 공간 데이터 또는 속성 데이터를 출력하는 경우의 3가지로 분류된다. 이와 같은 3가지의 질의 유형을 모두 만족하는 질의 처리

기는 검색 기능으로서 텍스트 검색을 위한 SQL의 기능과 공간 데이터 검색 및 공간 연산 기능이 필요하며, 출력 기능으로서 테이블 형식의 텍스트 출력 기능 및 화면출력을 위한 제어 기능이 갖추어져야 한다.

### 5.2 사용상의 편의를 위한 요건

사용상의 편의(User Friendly) 측면에서 지형 질의어가 갖추어야 할 요구조건으로 1) 지형 질의어의 Syntax 형식은 가능한 한 SQL 형태를 그대로 활용되도록 공간 연산 및 화면 출력 제어 기능의 확장에 따른 사용자의 혼란을 최대한 줄여야 한다. 즉, 속성 데이터와 공간 데이터를 동일한 형식(Uniform Interface)으로 검색할 수 있어야 한다. 2) 화면상에 출력된 지형 정보를 마우스로 클릭하면서 질의하는 대화식 질의를 지원해야 한다. 지형 정보 시스템에서는 화면상에 출력된 지도 정보를 사용자가 눈으로 확인해가며 원하는 지형 정보 객체를 마우스로 클릭하여 질의하는 것이 효율적이기 때문이다.

### 5.3 기능적인 측면의 요건

마지막으로 기능적인 측면에서 지형 질의어가 갖추어야 할 요구조건[EGEN94]은 질의어의 자체의 기본 기능과 질의를 위한 그래픽 사용자 인터페이스의 기능으로 나누어진다. 질의어 자체의 기능과 그래픽 사용자 인터페이스의 기능은 다음과 같다.

#### 5.3.1 질의어 자체 기능

질의어 자체가 갖추어야 할 기본 기능을 살펴보면 1) 질의의 검색 결과를 화면상에 그래픽 형태로 출력하여 자연적인 지형 정보의 인식 및 분석이 가능해야 하며, 2) 검색된 지형 정보 객체들에 대해 또다른 질의를 수행하는 중복질의를 허용하며, 3) 다양한 공간 데이터에 대한 공간 연산 및 공간 관련성 표현을 지원해야 한다. 그리고 이러한 공간 연산자 및 공간 관련성 표현자를 질의어에서 사용할 수 있어야 한다. 4) 기존의 상용 GIS S/W나 지금까지의 연구는 속성



데이터의 단순 검색만을 지원한다. 앞으로의 GIS가 특정 응용 분야의 비정형 업무처리의 이용에 확대될 것을 감안하면 지형 정보를 기반으로 한 계획, 판단 등을 위한 공간 의사 결정을 지원해야 한다. 이 논문에서 새로 추가한 지형 질의어 요건은 지형 질의어 내에서 공간 의사 결정을 위해 규칙을 호출할 수 있고 규칙 처리 결과를 지형 질의어 처리 결과로서 보여주는 기능이다. 공간 의사 결정에 대해서는 6.2절에서 자세히 설명한다.

### 5.3.2 그래픽 사용자 인터페이스의 기능

질의와 연관된 그래픽 사용자 인터페이스의 기능을 살펴보면 1) GIS는 화면상에 출력된 공간 데이터를 마우스로 클릭하여 그 지형 객체의 속성 정보를 얻거나 마우스로 질의 영역을 설정하는 대화식 질의 기능의 추가가 필요하며, 2) 다양한 형태의 출력을 위해 색깔, 패턴, 심볼 등을 임의로 지정할 수 있어야 하며, 3) 현재 출력된 공간 데이터의 인식을 위해 범례(Legend)를 표시해 주어야 하며, 4) 출력된 지형 정보에 대한 레이블 출력 기능으로 사용자 인식의 편의를 제공해야 하고, 5) 광범위한 데이터를 하나의 화면상에 출력할 수 없는 경우가 많으므로 축소/확대 및 Scroll/Pan 기능이 포함되어야 한다.

## 6. 지형 질의어(GeoQL)의 설계

이 장에서는 지형 정보 객체의 공간 데이터 검색, 공간 데이터 연산, 비정형 문제를 처리하는 공간의사 결정 기능을 포함하는 질의어인 GeoQL의 설계 내용을 자세히 기술한다.

### 6.1 공간 연산자와 공간 관련성 표현자의 메소드 정의

이 논문에서 제시하는 지형 질의어인 GeoQL은 SQL 확장형이다. GeoQL에서 사용되는 연산자는 공간 연산자와 공간 관련성 표현자로 구분한다. 공간 연산자는 일정한 값(value)을 반환하는 반면 공간 관련성 표현자는 참 또는 거짓을 반환한다. 예를 들어 두 객체사이의 거리를 구

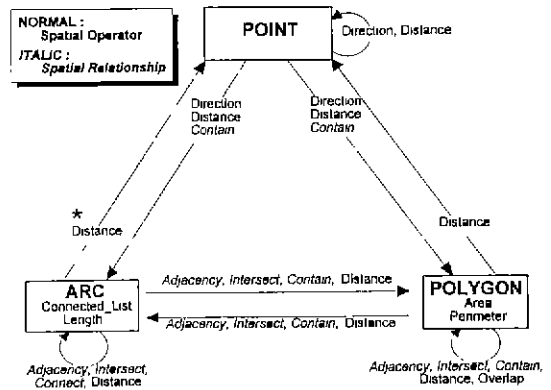


그림 6 공간 연산자와 공간 관련성 표현자

하는 Distance, 특정 객체의 면적을 구하는 Area 등과 같은 연산자는 일정한 값을 반환하므로 공간 연산자이다. 반면 두 지형정보 객체가 연결이 되었는지를 판명하는 연산자인 Connect, 두 객체가 교차하는지를 나타내는 연산자인 Intersect 등은 참, 거짓을 반환한다. 이런 연산자는 공간 관련성 표현자라 명명한다. 공간 연산자 및 공간 관련성 표현자를 모두 메소드로 정의하였다. 이러한 연산자는 실행 시간(Run Time)에 공간 데이터를 이용하여 계산을 수행함으로써 실행 결과를 반환한다. 넓은 의미에서의 공간 연산자는 공간 연산자와 공간 관련성 표현자를 포함한다. 연산자를 공간 데이터 클래스에 따라 분류하면 그림 6과 같다. 그림 6에서 각각의 화살표 선상에 있는 연산자는 화살표가 시작하는 클래스 내에 정의되어 있고 화살표가 가리키는 쪽의 클래스는 그 연산자에 적용될 인수를 나타낸다.

### 6.2 공간의사결정을 위한 규칙 정의

지금까지의 GIS에 관한 연구[ESRI92]에서는 공간 연산, 공간 데이터간의 공간 관련성을 표현하는 기능 및 단순 속성 검색 기능은 뛰어나다. 반면 현재 GIS 응용 분야는 단순한 공간적 문제나 속성정보의 검색에 국한되지 않는다. 앞으로 GIS가 나아가야 할 방향은 계획이나 설계와 같은 비정형문제(non-routine or ill-structured problems)의 해결을 지원할 수 있는 "개념 위주의 GIS(conceptualization-oriented GIS)"라 할

수 있다. 계획이나 설계와 같은 비정형 문제는 다양한 공간 데이터의 관련성 및 속성 데이터간의 관련된 정보를 이용할 뿐만아니라 전문가의 노하우(knowhow)를 이용하여 해결해야 할 경우가 많다. 이런 업무를 수행하기 위해 “공간 의사 결정(Spatial Decision Making)”[DENS91, ZHU 92, 문상호94b]을 도입한다.

최근 IJGIS(International Journal of Geographical Information System), GIS/LIS(Land Information System), URISA(Urban and Regional Information Systems Association), NCGIA(National Center for Geographic Information and Analysis) 등 GIS학술지 및 연구기관에서는 “공간 의사 결정”에 대한 많은 연구가 진행되고 있다. [DENS91]에서는 “공간 의사 결정 시스템”을, ① 특정 응용에서의 비정형 업무의 해결을 위해 정의되고, ② 공간적, 비공간적 자료들의 관련성을 이용하며, ③ 분석 및 통계 모델의 사용을 용이하게 하고, ④ 그래픽 사용자 인터페이스의 사용으로 필요한 정보를 의사 결정자가 쉽게 얻을 수 있고, ⑤ 새로운 기능을 쉽게 확장할 수 있는 시스템이라고 정의하고 있다.

앞에서 언급한 바와 같이 지금까지의 GIS는 단순 문제만을 해결할 수 있는 “정보위주의 GIS”이다. 그동안의 GIS는 아름답고 화려한 도면과 지도를 생성하여 많은 사용자에게 관심, 흥미를 제공하였지만 각 정보간의 관계를 이해하는 데 실질적인 도움이 안되어 최종 결정을 위한 작업 및 책임은 전적으로 그 사용자의 지각 능력에 의존하게 된다[TURK90]. 이는 GIS가 비정형 문제를 위한 명쾌한 해결책을 제시하지는 않는다는 결론을 내릴 수 있다. 반면 GIS가 발달하면 할수록 기존의 단순문제보다는 해결 과정이 뚜렷하게 나타나지 않는 비정형문제가 증가할 것이다. 이를 감안할 때 GIS에서는 공간 의사 결정의 기능이 필수적으로 추가되어야 한다.

GeoQL은 공간의사결정을 지원하기 위해 비정형 문제를 규칙으로 정의하고 이러한 규칙들을 추론할 수 있는 메소드를 해당 클래스에 정의하였다. 그리고 사용자는 GoeQL의 Select나 Where구문에서 메소드를 호출함으로써 계획, 설계와 같은 비정형 문제를 해결할 수 있다. 규

칙을 이용한 이유는 다음과 같다. 전문가의 지식을 규칙으로 표현하기가 용이하며 규칙을 동적으로 추가하기가 쉽게 때문이다. 이때 규칙 추론에서는 프레디키트가 관련 정보를 포함하게 된다. 아울러 복잡한 추론 과정과 같은 프로그래밍이 불필요하다는 장점이 있다. 반면 비정형 문제를 프로그래밍으로 해결하고자 할 경우 새로운 전문가 지식이 추가되면 관련된 함수들을 수정해야 하는 불편함이 있다.

### 6.3 GeoQL의 설계

기존의 관계 연산자에 공간 연산자를 추가하는 DBMS 확장 방법은 공간 DBMS내에 적용될 공간 연산자를 미리 정의해야 한다. 그리고 새로운 공간 연산자를 추가하고자 할 경우 커널을 재수정해야 한다. 반면 이 논문에서는 공간 연산자 및 규칙을 SQL의 함수로서 내장(Embedded)시킴으로써 질의처리기 모듈을 변경하지 않고 클래스에 공간 연산자와 규칙을 정의하는 메소드만을 추가하면 된다. GeoQL에서의 속성 데이터는 객체에 정의된 슬롯값을 점 표기법(Dot Notation)을 이용하여 검색가능하다. 그리고 객체 지향 시스템의 특징인 OID를 이용한 경로 탐색 항해(Path Navigation)가 가능하다. 아래는 “이름이 MP256인 배수관의 속성 데이터를 검색하라”라는 단순 속성 검색 질의 예이다.

```
SELECT p.name, p.capacity, p.press
FROM p. IN main_pipe
WHERE p.name="MP256";
```

#### 6.3.1 메소드 형식의 공간 연산자 및 공간 관련성 표현자

이 논문에서는 공간 연산자를 상속 계층 구조의 메소드로 정의하여 공통적인 공간 연산자는 추상화된 객체의 메소드로 모델링하고 서브클래스에서 재정의(undefine)하는 방법을 사용한다. 그리고 공간 연산자를 과부하연산자(Overloading Operator)로 정의하여 사용상의 편의를 도모하였다. 그리고 공간 연산자를 추가해야 할 경우 필요한 클래스내에 재정의하거나 추가하면

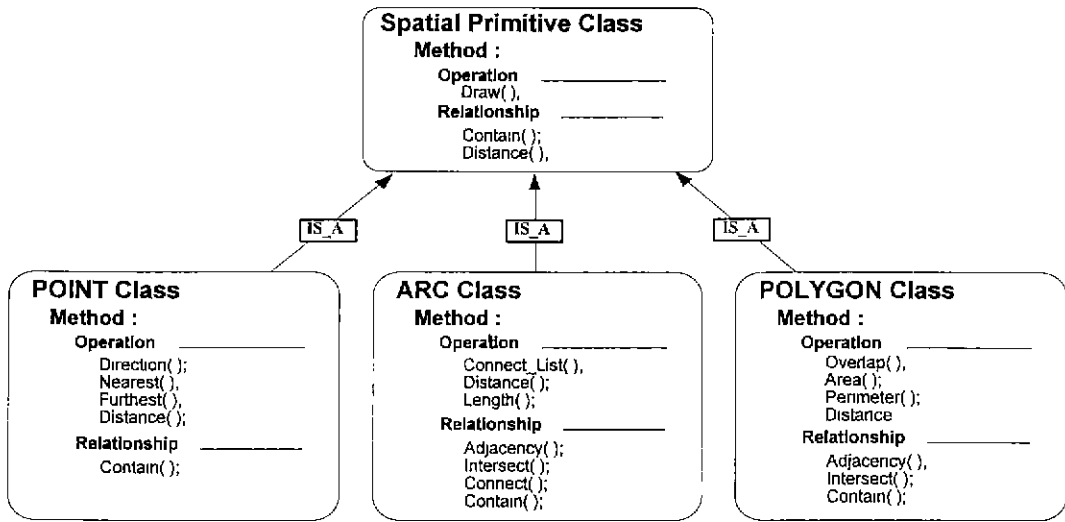


그림 7 클래스별 공간 연산자

된다. 따라서 확장성이 높아진다. 예를 들어 그림 6에서 ARC 클래스의 Distance라는 공간 연산자는 공간 요소 클래스내에 정의된 것을 상속 및 재정의하였다. 그리고 이 연산자는 인수로서 POINT, ARC, POLYGON을 갖는 3가지 종류의 연산자가 필요하지만 과부하연산자로 정의하기 때문에 하나의 연산자로 표현하는 것이 가능하다. 그림 6에서 정의된 공간 연산자와 공간 관련성 표현자를 각 클래스별로 표현하면 그림 7과 같다.

공간 연산자를 이용한 질의예로서 “머리재 빌딩으로부터 100 M이내의 빌딩을 화면상에 출력하고 이름과 주소를 텍스트로 출력하라”를 수행하는 질의는 아래와 같다. bldg는 다각형으로 구성된 레이어이고 Distance 공간 연산자는 다각형 클래스에 정의된 메소드이다.

```

SELECT BD1.draw(), BD1.name,
BD1.address
FROM BD1 IN bldg, BD2 IN bldg
WHERE BD2.distance(BD1)<100
and BD2.name=머리재;
    
```

**6.3.2 규칙을 이용한 공간 의사 결정**

GeoQL은 Select나 Where절에서 규칙을 호출할 수 있는 기능이 포함되어 있다. 공간 의사

결정은 지형 정보의 공간 데이터 및 속성데이터 관련성에 밀접한 관련이 있다. 특히 의사 결정을 위해서는 주로 속성 데이터간의 관련성을 근거로 규칙 추론을 하게 된다. 이들 공간 의사 결정 문제는 GIS 응용 분야와 지형 정보들에 따라 매우 다양하다. 이러한 공간 의사 결정은 제한된 전문영역을 위해 설정이 된다. 그러므로 전문가의 지식, 노하우(Knowhow)가 반영이 되어야 한다. 따라서 이러한 여건을 고려하여 불패, 메소드나 슬롯을 이용한 관련성 표현은 부적합하며, 규칙을 이용하는 것이 바람직하다.

규칙을 이용하는 경우 전문가의 지식을 자연스럽게 규칙으로 정의할 수 있다. 또한 전문가의 새로운 지식이 추가되거나 변경시 해당 규칙만을 변경하여 수정이나 확장을 쉽게 할 수 있다. 그리고 규칙을 추론한 결과가 프레디키트 형식으로 Fact Base에 저장이 되므로, 추후 같은 관련성을 검색할 경우 추론 과정없이 프레디키트를 통하여 빨리 검색할 수 있다. 이 논문에서는 사용상의 편의를 위해 관련성을 표현한 규칙을 직접 실행하는 명령을 사용하지 않고 메소드를 이용한다. 즉 객체가 규칙을 호출하는 메소드를 가진다. 따라서 규칙을 추론하고자 할 경우 GeoQL을 이용하여 해당 객체의 메소드를 호출하면 된다. 상수도 관리 업무에서의 노후배수관 교체에

관한 공간 의사 결정 문제를 예로 설명하고자 한다. 배수관의 노후나 약화 등으로 인하여 사용자에게 양질의 물을 공급하기가 어려울 때 관리자 혹은 전문가는 노후관 교체에 대한 의사 결정을 내려야 한다. 즉, 어느 배수관을 먼저 교체할 것인가를 결정해야 한다. 지금까지는 계획, 설계와 같은 비정형 문제를 담당자가 단순히 화면상에 출력된 지형 정보와 단순 속성 정보를 근거로 상황에 따라 적절히 판단, 의사 결정을 내렸다. 이 결정은 전적으로 담당자의 주관적인 판단에 의해 좌우가 된다. 이는 기존 GIS가 단순 정보를 제공할 뿐 의사 결정에 실질적으로 도움이 되지못함을 의미한다.

이 논문에서는 이러한 공간 의사 결정을 지원하기 위해 전문가의 각종 지식을 규칙으로 표현하고 가중치 등과 같은 각종 부가정보를 규칙 내에 포함시켰다. 따라서 사용자가 명백한 근거에 의한 객관적인 판단을 내릴 수 있도록 하였다. 노후관 교체와 같은 공간 의사 결정에서는 교체 시기 등급이 “위험” 단계인 배수관을 결정하는 작업이다. 교체 시기 등급을 결정하기 위해서는 배수관의 노후등급, 배수관의 재질 등급, 해당 등급의 가중치 등 각종 정보가 필요하다. 이 때 필요한 정보는 전문가의 지식을 기준으로 선정한다. 전문가의 지식이 변할 때는 해당 규칙만을 수정하면 된다.

“현재 상태가 위험하여 교체해야할 상수도관을 화면상에 출력하라”는 질의는 아래와 같다. 질의에서 상수도관(main\_pipe) 클래스내에 정의된 메소드 exchange\_factor를 수행한다. 이 메소드는 프레디키트 Exchange (<?pipe\_name, ?factor>)를 사용하여 규칙추론을 한다. 메소드 exchange-factor에서 호출하는 규칙은 아래와 같다.

```
SELECT p.draw()
FROM p IN main_pipe
WHERE p.exchange_factor()="danger";
```

```
if : (Grade_Old(?Pipe_Name, ?old_grade))
      (Grade_Weak(?Pipe_Name, ?weak_grade))
      ?factor:=Computation
```

```
(노후등급 계산: old_grade, weak_grade 이
용)
then : Exchange(?Pipe_Name, ?factor)
if : (?Pipe_Name.매설년도<기준년도)
      (?old_grade:=기준년도-매설년도)
then : Grade_Old(?Pipe_Name, ?old_grade)
if : ?weak_grade=Computation
      (Pipe_Name의 재질에 따라 weak_grade계산)
then : Grade_Weak(?Pipe_Name, ?weak_grade)
```

### 6.3.3 대화식 질의

지형 질의어가 반드시 제공해야 할 기능은 마우스를 이용한 대화식 질의를 지원해야 한다는 점이다. GeoQL의 Where구문내에 PICK, ZONE 키워드를 두어 화면상에 출력된 객체를 선택(PICK)하거나 질의 영역(ZONE)을 질의 수행중에 직접 지정할 수 있도록 하였다. 아래는 사용자가 마우스로 지정한 영역내에 있는 빌딩의 속성 데이터 검색 질의이다.

```
SELECT p.address, p.owner, p.population,
p.floor
FROM p IN bldg
WHERE p within ZONE
```

### 6.3.4 복합질의

아래의 질의는 속성 데이터의 검색, 공간 연산의 수행, 관련성 표현을 위한 규칙 추론, 영역 지정에 의한 대화식 질의 등을 포함한 복합질의이다. 이러한 복합 질의의 처리 메카니즘을 7.2 절에서 자세히 기술한다.

```
SELECT p.draw()
FROM p IN sub_pipe
WHERE p within ZONE
and p.capacity >=50 and p.oldpipe_factor()
="danger";
```

## 7. 질의 처리기의 구현

### 7.1 GeoQL 질의 처리기의 구조

GeoQL 질의 처리기의 전체 구조는 그림 8과

같다. 그림 8의 질의 수행기는 공간 데이터의 검색 및 출력과 같은 GIS 기능을 요구하는 질의일 경우에는 객체 지향 시스템을 검색하여 질의 결과를 출력한다. 반면 분석된 구문이 규칙 추론과 같은 공간 의사 결정 기능을 요구하는 질의일 경우에는 질의 수행기는 Rule Base에서 필요한 규칙과 fact를 검색하여 추론한다. 규칙 내에서 객체의 슬롯값을 읽어야 하는 경우에는 규칙 기반 시스템은 객체지향 시스템의 객체 배이스를 접근한다. 사용자의 질의 입력이 구문상의 오류가 있을 경우에는 질의 입력기의 오류 메시지 출력 윈도우에 오류 메시지가 전송된다.

그리고 사용자 인터페이스에 객체 브라우저를 두어 사용자가 GeoQL을 사용할 때 필요한 클래스 계층 구조와 각 클래스의 슬롯과 메소드들을 브라우징하는 기능을 제공한다. 이는 관계형 데이터베이스에서는 정의된 테이블 및 애트리뷰트에 관한 정보가 Catalog에 정의되어 있으므로 질의시 이를 참조하면 되지만 객체 지향 시스템은 이를 대신할 만한 정보가 따로 존재하지 않기 때문이다.

### 7.2 분할 기법에 의한 질의 처리

GeoQL에서 다루는 질의는 크게 텍스트 검색

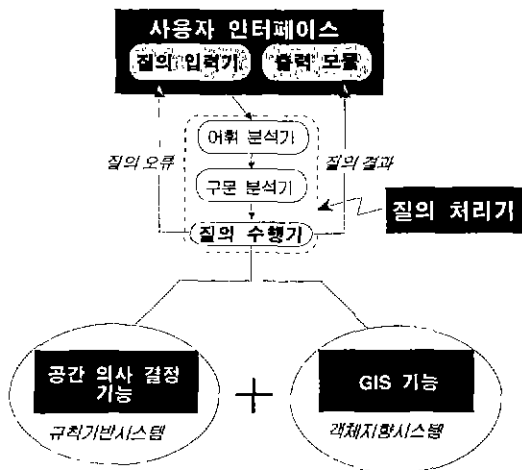


그림 8 질의 처리기의 구조

질의, 규칙 추론 질의, 공간 연산을 위한 메소드 호출 질의와 같은 3부분으로 나눈다. 이 논문에서는 사용자로부터 입력 GeoQL 구문을 위와 같은 3개의 부모들로 분할하여 처리한 다음 부모들에서 반환한 값을 다시 질의 처리기에서 재구성하는 방법을 택했다. 사용자가 질의어를 입력하면 부모 모듈인 질의 처리기는 질의를 유형별로 분류한다. 분류한 질의가 속성 데이터 검색과 같은 텍스트 검색이면 "Text Processor"로 보내지고, 공간 데이터에 대한 공간 연산이면 "Method Processor"로 보내진다. 그리고 공간 의사 결정과 같은 규칙 추론 질의이면 "Rule Processor"로 보내진다. 각 부모들에서 연산이 완료되면 부모들로 값을 반환한다. 그러면 부모 모듈인 질의 처리기에서는 이들 값을 취합하여 결과를 생성한다. 이때 결과가 텍스트 출력이면 그래픽 사용자 인터페이스의 텍스트 출력 모듈로 출력되고, 결과가 그래픽 출력이면 그래픽 사용자 인터페이스의 그래픽 출력 모듈로 출력된다. 그림 9는 이를 도식화한 것이다.

그림 10은 6.3.4에서 제시한 복합질의에 대한 처리과정의 예를 보여주고 있다. 그림 10에서 입력 질의는 "현재 출력된 지도에서 사용자가 마우스로 지정한 질의 영역 내에 속하고 배수관 용량이 50 이상이며, 교체 등급이 위험 등급이라 교체해야 할 배수관을 화면상에 출력하라"라는 기능을 수행한다. 입력 질의의 Where 절 중에 (2)가 텍스트 검색 질의이고, (1)이 공간 연산 질의이며, (3)이 공간 의사 결정을 위한 규칙 추론

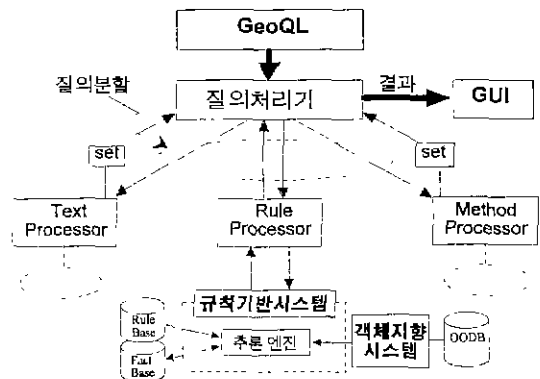


그림 9 분할 기법에 의한 질의 처리 과정

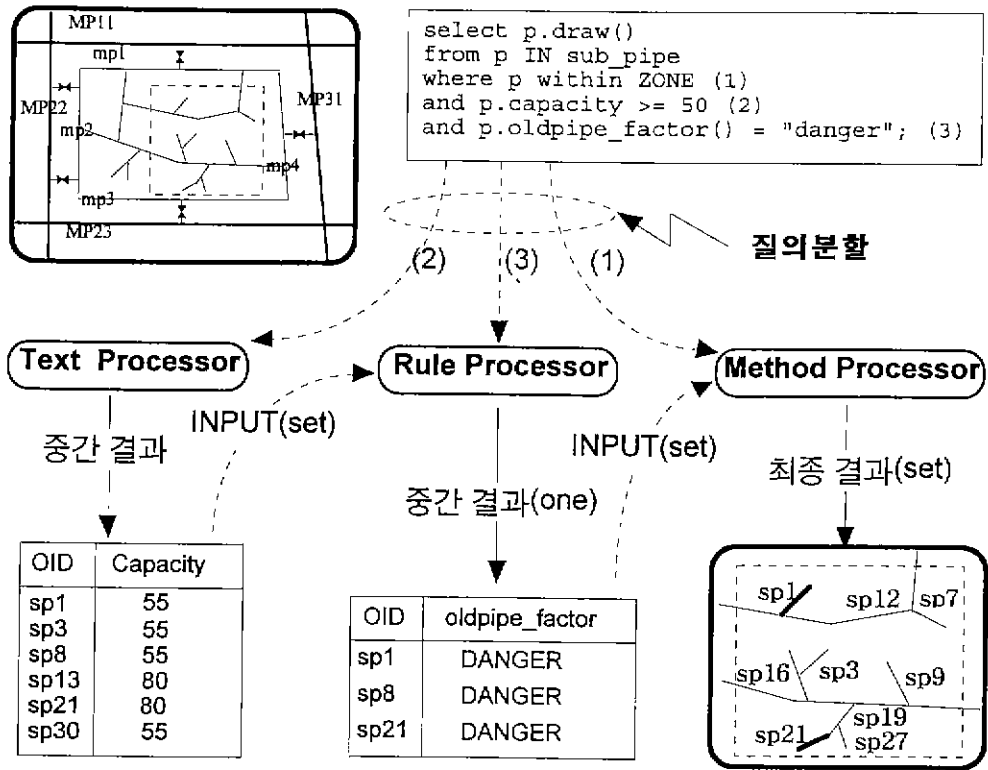


그림 10 분할 기법에 의한 실행 예

질의이다. 그림 10에서 제시된 질의는 복합 질의이므로 질의중 (1)의 수행을 위해 Method Processor로, (2)의 수행을 위해 Text Processor로 보내진다. (3)의 수행을 위해서는 Rule Processor로 보내지게 된다. 그리고 최종 결과가 화면상에 출력되어야 하므로 GUI의 그래픽 출력 모듈로 넘겨진다.

그런데 그림 10의 질의 수행과정은 텍스트 검색 질의인 (2)가 제일 먼저 수행되었다. 그리고 나머지 질의인 (1)과 (3)이 수행된다. 이것은 질의 수행 성능 향상을 위한 순서이다. 규칙 추론이나 공간 연산 등과 같은 질의는 텍스트 검색 질의에 비해 많은 연산이 요구된다. 그러므로 텍스트 검색 연산을 먼저 수행하여 결과 후보 객체를 여과(Filtering)한 뒤 보다 적은 수의 객체에 대해 규칙 추론이나 공간 연산을 하기 위해서이다. 그러나 질의 최적화 문제는 향후 연구 과제로 남겨둔다.

### 8. 지형질의어 사용을 위한 그래픽 사용자 인터페이스의 설계 및 구현

그래픽 사용자 인터페이스는 앞에서 언급한 바와 같이 크게 사용자의 질의에 의해 검색된 지형정보를 화면상에 출력하는 출력 및 기본 검색모듈, 사용자의 질의를 받아들여 질의 처리기로 보내는 기능을 하는 질의어 입력부, 다양한 형태의 지형 정보 출력을 위한 출력 환경 설정부, 객체 브라우저등으로 구성된다. 그림 11은 사용자 인터페이스의 구조를 나타낸다.

그림 11에서 A는 입력한 질의어가 질의처리기로 전송되는 과정이며, B는 질의처리 과정중의 구문 에러를 질의어 입력부로 반환되는 과정이다. C는 사용자가 질의어 입력시 객체에 정의된 슬롯이나 메소드를 확인하기 위해 객체 브라우저를 참조하는 과정이며, D는 환경설정 모듈에서 지정한 내용이 출력 모듈에 반영되어야 함을 나

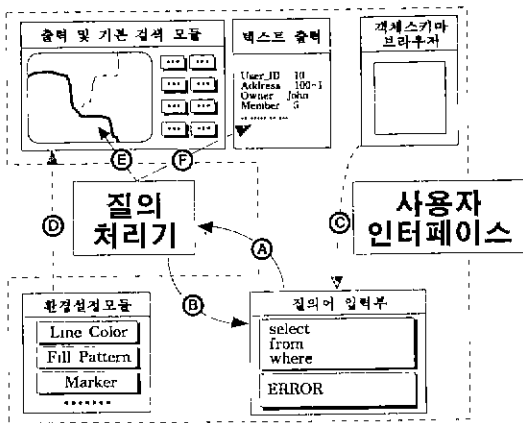


그림 11 사용자 인터페이스 구조

타내고, E와 F는 사용자가 입력한 질의어의 결과가 공간 데이터의 출력일 경우는 그래픽 출력 모듈로, 질의가 속성 데이터 검색일 경우에는 텍스트 출력 모듈로 출력됨을 보여준다.

출력 및 기본 검색 모듈은 질의의 결과에 대한 그래픽 출력 및 텍스트 출력을 담당한다. 환경 설정 모듈은 다양한 출력 양식 지정을 위한 모듈로서 출력될 지형정보를 사용자의 선택으로 원하는 출력 양식을 설정할 수 있는 환경을 제공한다. 질의어 입력부는 사용자로부터 질의어를 입력받아 질의처리기로 보내는 역할과 질의에 오류가 있을 경우 질의처리기로부터 넘어온 오류 메시지를 사용자가 볼 수 있도록 하기 위해 하나의 윈도우가 제공된다. 객체 브라우저는 객체의 슬롯과 공간연산자인 메소드 정보를 사용자

에게 제공하여 질의시 참조할 수 있도록 하였다.

### 8.1 객체 브라우저

질의를 하고자하는 사용자는 각 객체에 정의된 애트리뷰트나 공간 연산자인 메소드를 알고서 질의를 해야한다. 기존의 관계형 데이터베이스에서는 정의된 테이블이나 애트리뷰트에 관한 정보가 Catalog에 정의되어 있으므로 이를 참조하면 되지만 객체 지향 시스템은 이를 대신할 만한 정보가 따로 존재하지 않는다. 그래서 이 논문에서는 객체 브라우저를 두어 현재 정의된 클래스의 슬롯 정보와 메소드 정보를 한눈에 확인 가능하도록 하였다. 그림 12는 객체 브라우저에 관한 그림이다.

그림 12에서 좌측 윈도우는 현재 정의된 클래스의 계층구조를 트리 형식으로 나타내 주고 있고 우측 윈도우는 특정 클래스의 마우스로 클릭하였을 경우에 그 클래스에 정의된 슬롯과 메소드를 나타내 주고 있다. 이때 각 클래스의 슬롯과 메소드가 상위 클래스로 부터 상속되었는지, 상속되어 재정의 하였는지, 현재 클래스 자체에서 새로 정의한 것인지를 표현하는 플래그를 두게 된다. 아울러 이 객체 브라우저는 클래스를 정의하고 삭제하고 인스턴스를 생성하는 기능 등을 포함하게 된다. 그리고 사용자의 질의시 마우스로 선택한 객체에 대한 자세한 정보가 동시에 브라우징되는 동기화 브라우징(Synchronized Browsing)을 지원한다. 만약 사용자가 질의

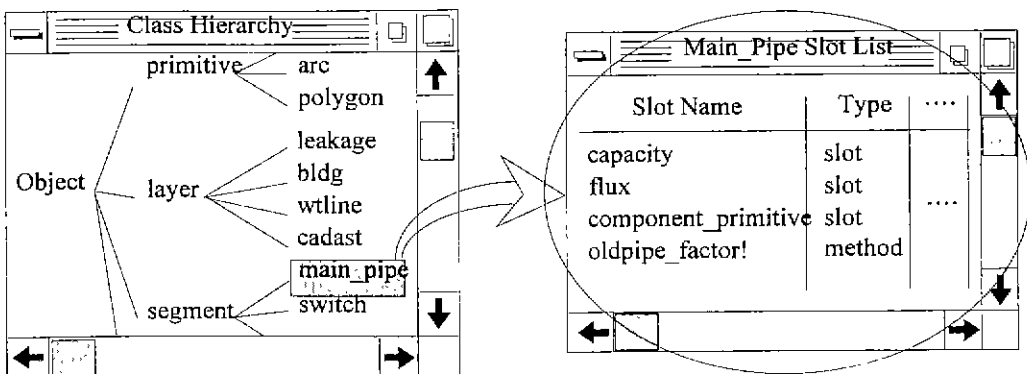


그림 12 객체 브라우저

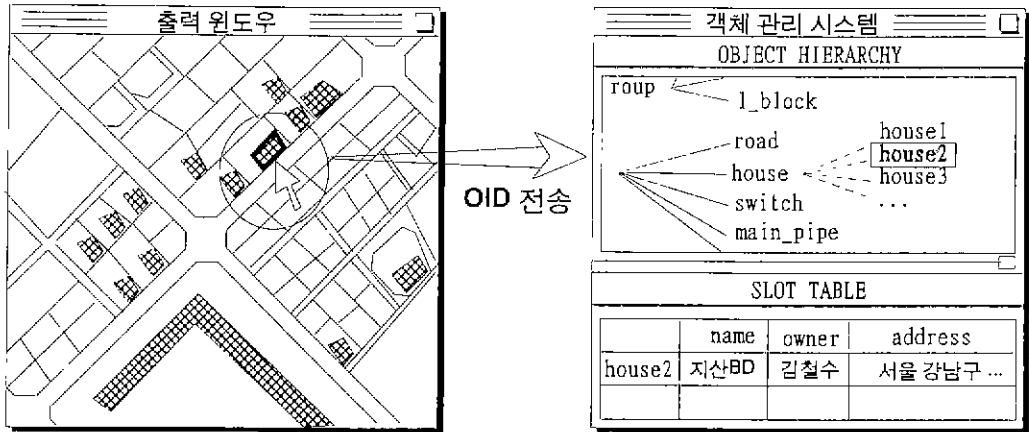


그림 13 동기화 브라우저

하고자 하는 객체에 정의된 슬롯이나 메소드를 알아보하고자할 경우 그림 13에 나타난 바와 같이 화면상에 출력된 지형 정보 객체를 마우스로 클릭하기만 하면 된다. 출력 윈도우에서 객체를 마우스로 선택했을 때 그 객체의 OID값이 객체 브라우저에 전송이 되어 해당되는 객체의 슬롯 및 메소드 정보를 화면상에 출력한다.

### 9. 결 론

이 논문에서 제시하는 객체 지향 규칙 기반 지형 질의어인 GeoQL은 공간 연산자를 IS-A 계층 구조의 객체 메소드로 정의하였고, 공간 의사 결정을 위해 규칙을 정의하여 질의어에서 규칙을 추론할 수 있도록 하였다. 공간 연산자를 객체의 메소드로 정의함으로써 SQL에 내장된 공간 메소드를 DBMS나 지형 질의어의 커널수정없이 추가 및 변경이 가능한 이점이 있다. 그리고 서브 클래스의 메소드로 재정의하여 사용 가능하고 같은 의미를 가지는 공간 연산자는 과부하(Overloading)연산자로 정의하여 사용상의 편의를 도모했다. 그리고 계획 설계와 같은 비단순 문제를 처리하는 공간 의사 결정을 위해 규칙을 정의함으로써 확장성이 높아지고 전문가의 노하우를 규칙으로 자연스럽게 표현할 수 있다는 이점이 있다. 주어진 문제에 따라 전향 체인 방식과 후향 체인 방식으로 규칙의 추론이 가능하도록하여 최적의 의사 결정을 내릴 수 있도록

하였다.

메소드로 정의된 공간 연산자와 공간 의사 결정을 위한 규칙을 GeoQL내에 내장(Embedding)하여 질의어내에서 직접 호출할 수 있다. 특히 SQL 확장 형태의 구문에 규칙 호출 기능을 내장하는 방법을 새로 제시하여 사용자가 구체적인 규칙 및 프레디키트를 몰라도 쉽게 규칙 추론할 수 있도록 하였다. 그리고 규칙 추론의 결과를 사용자가 질의에 사용할 수 있도록 하였다. 질의 처리기의 구현 방법으로는 텍스트, 메소드, 규칙처리모듈로 분리하여 수행하는 질의 분할 기법을 택했으며 분할 처리 메카니즘을 제시하였다.

GIS의 그래픽 사용자 인터페이스는 지형 정보의 출력 기능과 환경 설정 및 기타 기능을 지원한다. 이 논문에서는 이러한 GUI와 질의처리를 통합하여 대화식 질의등을 쉽게 지원한다. 그리고 다양한 출력 기능 및 환경설정 등의 기능으로 사용자가 보다 쉽게 질의를 할 수 있도록 하였다.

앞으로의 연구과제는 성능 향상을 위한 GeoQL의 질의 최적화 문제와 공간 데이터 저장 구조에 관한 연구가 필요하다. 특히 기존 공간 인덱싱 기법에 의한 공간 데이터 저장 및 검색에서의 효율에 대한 전반적인 검토가 필요하다.

### 참고문헌

[DENS91] P. J. Densham, "Geographical Informa-

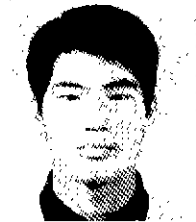


- tion Systems", Longman Scientific & Technical, Vol. 1, pp. 403-412, 1991.
- [EGEN88] Max J. Egenhofer and A. Frank, "Towards a spatial query language" User interface consideration", in Proc. 14th Int. Conf. VLDB, pp. 124-133, 1988.
- [EGEN94] Max J. Egenhofer, "Spatial Query: A Query and Presentation Language", IEEE Transaction On Knowledge and Data Engineering, Vol. 6, No. 1, pp. 86-95, 1994.
- [ESRI92] Environmental Systems Research Institute, Inc. "Understanding GIS : The Arc/Info method", 1992.
- [MOHA88] L.Mohan and R.L.Kahyap "An Object-Oriented Knowledge Representation for Spatial Information", IEEE TSE, 1988, Vol. 14, No. 5, pp. 675-681, May, 1988.
- [ROUS88] Nick Roussopoulos, Christor Faloutsos, Timos Sellis, "An Efficient Pictorial Database System for PSQL", IEEE Trans. on Soft. Eng., Vol. 14, No. 5, pp. 639-650, May, 1988.
- [SCHO91] Michel Scholl, Agnes Voisard, "Object-Oriented Database Systems for Geographic Applications: an Experiment with O2", Geographic Management Systems Workshop Proceedings, pp. 103-137, May, 1991.
- [TANA88] Tanaka. M. and Ichikawa T., "A Visual User Interface for Map Information Retrieval Based on Semantic Significance.", IEEE Trans. On Soft. Eng., Vol. 14, No. 5, pp. 666-670, May, 1988.
- [TURK90] Turk, A.G., "Towards an Understanding of Human-Computer Interaction Aspects of Geographic Information Systems", Cartography, Vol. 19, No. 1, pp.31-60, 1990.
- [WANG92] Wang, Joyce C. C., "GIS Major Data Structures VS. Applications", Urban & Regional Information Systems Associations, 1992, Annual Conf. Prof., Vol. 2, pp. 13-24.
- [WEBS90] Chris Webster, "Rule-base spatial search", INT. J. GIS, Vol. 4, No. 3, pp. 241-259, 1990.
- [WORB90] M. F. Worboys, H. M. Hearnshaw, D. J. Maguire, "Object-oriented modelling for spatial databases", INT. J. GIS, Vol. 4, No. 4, pp. 369-384, 1990.
- [ZHU92] Zhu, X. R. Healey, "Toward Intelligent Spatial Decision Support : Integrating Geographical Information Systems and Expert Systems", GIS/LIS '92 Annual Conference, Nov. Vol. 2, pp. 877-886, 1992.
- [김진덕94a] 김진덕, 문상호, 홍봉희, "객체 지향 지형 정보의 입력 방법에 관한 연구", '94 동계 데이터베이스 학술 대회, 10권, 1호, pp. 191-198, 1994.
- [김진덕94b] 김진덕, 문상호, 선종복, 홍봉희, "객체 지향 구축 기반 지형 질의어의 설계 및 구현", 한국 정보과학회, '94 가을 학술발표논문집, 21권 2호, pp. 105-108, 1994.
- [김은형93] 김은형, "비단순문제 해결을 위한 GIS 향상방안", 한국GIS학회지, 1권 1호, pp. 63-72, 1993.
- [문상호94a] 문상호, "객체 지향 지식 기반 지형정보 시스템의 설계 및 구현", 부산대학교 컴퓨터공학과 대학원 석사학위논문, 1994.
- [문상호94b] 문상호, 김진덕, 홍봉희, "상수도 공간 의사 결정을 위한 GIS와 전문가 시스템의 통합", 한국 정보과학회 '94 가을 학술발표논문집, 21권 1호, pp. 109-112, 1994.
- [홍봉희93] 홍봉희, "상수도 노후관 교체 의사 결정을 지원하기 위한 ARC/INFO와 ProKappa의 통합 처리 방법에 관한 연구", 서울시정 개발연구원 위촉연구 최종연구보고서, 1993, 11.
- [田靖彦90] 田靖彦, "地圖情報시스템(Mapping System)", 日本 日刊工業新聞社, 1990



**김진덕**

1987 ~1993 부산대학교 컴퓨터공학과, B.S.  
1993 ~1995 부산대학교 컴퓨터공학과, M.S.  
1995 ~현재 부산대학교 컴퓨터공학과 박사과정 재학  
관심 분야: 지형질의어, GUI, 공간 의사 결정



**선종복**

1990 ~1994 부산대학교 컴퓨터공학과, B.S.  
1994 ~현재 부산대학교 컴퓨터공학과 석사과정 재학  
관심 분야: 공간 질의어 최적화, 공간질의처리



**문상호**

1987 ~1991 부산대학교 컴퓨터공학과, B.S.  
1991 ~1993 한국기계연구원 전산시스템실 상근위촉연구원  
1992 ~1994 부산대학교 컴퓨터공학과, M.S.  
1994 ~현재 부산대학교 컴퓨터공학과 박사과정 재학  
관심 분야: 공간 데이터 모델링, 공간의사결정, 위상 데이터 모델링



**홍봉희**

1978 ~1982 서울대학교 컴퓨터공학과, B.S.  
1982 ~1984 서울대학교 컴퓨터공학과, M.S.  
1984 ~1988 서울대학교 컴퓨터공학과, Ph.D.  
1987 ~현재 부산대학교 컴퓨터공학과, 부교수  
관심 분야: GIS, Engineering DB, CIM DB

● 제 22회 임시총회 · 춘계학술발표회 ●

- 일 자 : 1995년 4월 28일(금)~29일(토)
- 장 소 : 조선대학교
- 주 최 : 한국정보과학회
- 문 의 : 학회사무국  
T. 02-588-9246~7  
F. 02-521-1352