산업기술연구(강원대학교 산업기술연구소 논문집), 제18집, 1998.
Research Report. The Institute of Industrial Technology, Kangwon Nat'l Univ., Korea, No. 18, 1998.

地理 情報 시스템을 위한 三段階 데이터베이스 設計

Three-Phase Database Design for a Geographic Information System

옥 한석[1], 김 갑열[2], 김 창환[3], 김 상욱[4], 신 재호[5], 양 재웅[6]
Ock Han-Suk Kim Gap-Youl Kim Chang-Hwan Kim Sang-Wook Shin Jae-Ho Yang Jae-Ung

ABSTRACT

Effective design of a database is essential for operating application systems efficiently. This paper discusses database design for a geographic information system. The goals of database design are multiple: to satisfy the information content requirements of the specified users and applications; to provide a natural and easy-to-understand structuring of the information; and to support processing requirements and any performance objectives such as response time, processing time, and storage space. Database design is a very complex process and is decomposed in three phases: conceptual, logical, and physical design. In this paper, we first collect and analyze the requirements for a geographic information system. We also perform database design for these requirements through the three design phases systematically. Our results would contribute to the effective construction of a database for a geographic information system.

1. 서론

효율적인 토지정보관리를 실현하기 위해서는 합리적인 시스템 개발이 필요하다. 토지정보관리 시스템의 원활한 기능 수행을 위해서는 각종 정보들을 효과적으로 데이타베이스화 하여야 한다. 본 논문에서는 토지정보관리 시스템을 위한 데이타베이스 설계(database design)에 관하여 논의하고자 한다.

데이타베이스 설계는 응용에서 관리해야 하는 데이타를 구조화하는 과정이며, 크게 개념적 설계(conceptual design), 논리적 설계(logical design), 물리적 설계(physical design)의 세 단계로 구분된다[2][6][7]. 개념적 설계는 응용의 요구 사항 분석을 통하여 파악한 데이타의 내용과 이들간의 연관성을 명확하게 제시한다. 논리적 설계는 개념적 설계 결과를 대상으로 특정 데이타베이스 관리 시스템(database management system: DBMS)이 제공하는 데이타 모델을 이용하여 데이타를 조직화한다. 물리적 설계는 논리적 설계 결과를 대상으로 응용에서 자주 사용되는 질의들의 특성에 적합한 저장 구조와 액세스 방법 등을 결정함으로써 시스템이 최적의 성능을 발휘할 수 있도록 한다[8][9][10].

본 논문에서는 첫째, 요구 사항 분석을 통하여 토지정보관리 시스템에서 사용되는 데이타 집합을 정의한다. 둘째, 이러한 데이타 집합을 대상으로 개체-관계 모델(entity-relationship model)[4][3]을 이용한 개념적 설계 과정을 보인다. 셋째, 개념적 설계의 결과를 대상으로 관계 모델(relational model)[5]을 이용한 논리적 설계 과정을 보인다. 물리적 설계는 논리적 설계 결과를 대상으로 응용에서 자주 사용되는 질의들의 특성에 적합한 저장 구조와 액세스 방법 등을 결정함으로써 시스템이 최적의 성능을 발휘할 수 있도록 한다.

본 연구의 목적은 도시토지정보관리 시스템을 위한 최적의 데이타베이스 구축을 위하여 전통적인

1) 옥한석, 강원대학교 사범대학 지리교육과 교수
2) 김갑열, 강원대학교 사회대학 부동산학과 교수
3) 김창환, 강원대학교 사범대학 지리교육과 교수
4) 김상욱, 강원대학교 공과대학 정보통신공학과 교수
5) 신재호, 강원대학교 공과대학 정보통신공학과 학부 졸업
6) 양재웅, 강원대학교 공과대학 정보통신공학과 대학원 석사과정
이 연구는 정보통신부의 정보통신 우수 대학원 지원사업과 성곡학술 문화재단의 연구비 지원사업에 의하여 수행되었습니다.

삼단계 데이타베이스 설계 방법론을 적용하여 설계함으로써 데이타의 의미적인 측면과 시스템의 성능적인 측면을 동시에 최적화 한다는 데 있다. 본 연구 결과를 이용하여 도시토지정보관리를 위하여 필요한 각종 데이타를 파악할 수 있으며, 이들을 효과적으로 데이타베이스화 할 수 있다.

2. 데이타베이스 설계 개요

도시토지정보관리용 최적 데이타베이스 구축을 위해서는 체계적인 데이타베이스 설계가 필요하다. 데이타베이스 설계(database design)는 특정 응용에서 사용되는 데이타들을 효과적으로 관리하기위하여 조직화하는 작업으로 정의된다[7][13][9]. 데이터베이스 설계는 요구 사항의 수집 및 분석에 의하여 파악된 데이타를 대상으로 개념적 설계(conceptual design) 단계, 논리적 설계(logical design) 단계, 그리고 물리적 설계(physical design) 단계의 세가지 절차를 걸쳐 진행된다. 본 장에서는 관련 연구로서 이러한 각각의 데이타베이스 설계 단계의 특징에 대하여 간략히 설명한다.

2.1. 요구 사항 수집 및 분석

요구 사항 수집 및 분석(requirements collection and analysis)은 데이타베이스를 설계하기 위한 사전 작업으로서 해당 응용에서 사용되는 데이타를 수집하고, 이들이 해당 응용에서 어떠한 형태로 사용되는가를 구체적으로 분석하는 과정이다. 이 과정에서 나타나는 결과는 이후에 수행되는 데이타베이스 설계 단계의 초석이 되므로 정확한 데이타의 수집과 분석은 매우 중요하다.

2.2. 개념적 설계 단계

개념적 설계(conceptual design)는 요구 사항 수집 및 분석 단계에서 파악한 데이타를 기반으로 개념적 스키마(conceptual schema)를 구성하는 작업이다. 개념적 스키마는 데이타베이스 구조에 관한 상위 단계 기술 정보로서 응용에서 사용되는 데이타의 특성과 그들간의 연관성을 명확하게 제시한다. 따라서 개념적 설계는 응용에서 사용하고자 하는 DBMS가 지원하는 데이타 모델에 종속되지 않는다. 개념적 데이타 모델(conceptual data model)은 개념적 스키마를 기술하는 언어로서 개념적 설계를 지원한다. 현재 가장 널리 사용되는 것은 Chen에 의하여 제안된 개체-관계 모델(entity-relationship model)이다.

2.3. 논리적 설계 단계

논리적 설계(logical design)는 개념적 스키마를 기반으로 논리적 스키마(logical schema)를 구성하는 작업이다. 논리적 스키마는 특정 DBMS에 의하여 운영되는 데이타베이스의 구조적 특성을 기술한 것이다. 논리적 데이타 모델(logical data model)이란 논리적 스키마를 표현하기 위한 언어로서 논리적 설계를 지원한다. 지금까지 널리 사용되고 있는 논리적 데이타 모델로는 관계 모델(relational model), 네트워크 모델(network model), 계층 모델(hierarchical model), 객체지향 모델(object-oriented model) 등이 있다[7]. 같은 데이타베이스를 위한 논리적 스키마라도 데이타 모델에 따라 서로 달라질 수 있다.

2.4. 물리적 설계 단계

물리적 설계(physical design)는 논리적 스키마를 기반으로 물리적 스키마(physical schema)를 구성하는 작업이다. 물리적 스키마는 데이타베이스를 디스크내에서 구현하는 방법을 기술한 것이며, 물리적 설계를 통하여 응용의 특성에 맞는 저장 구조와 액세스 방법을 결정함으로써 전체 시스템이 최적의 성능을 발휘할 수 있다. 물리적 설계는 응용에서 사용하고자 하는 특정 DBMS가 제공하는 저장 구조 및 액세스 특성에 의하여 큰 영향을 받으므로 올바른 물리적 설계를 위해서는 해당 DBMS의 구조적 특성을 잘 파악해야 한다.

3. 요구 사항 수집 및 분석

본 장에서는 도시토지정보관리를 위한 요구 사항들을 수집하고 분석한 결과를 제시한다. 여기서는 도시토지정보관리에서 실제로 나타나는 데이타 객체(object)들의 특성을 요약한다.

- 권리자: 성명, 주민등록번호, 주소 등의 특성을 갖는다. 각 권리자는 다수의 토지에 대하

여 소유권을 가질 수 있으며, 각 소유권에 대한 현황을 알 수 있는 변동일자와 변동원인도 유지한다. 각 권리자는 다수의 토지에 대하여 저당권을 가질 수 있으며, 각 저당권에 대한 현황을 알 수 있는 등기원인도 유지한다. 또한, 각 권리자는 다수의 토지에 대하여 근저당권을 가질 수 있으며, 각 근저당권에 대한 현황을 알 수 있는 접수년월일, 채권금액, 채무자, 권리자도 유지한다.

- 토지: 식별자 등의 특성을 갖는다. 토지소재지로서 도, 시, 동, 지번들의 특성을 갖는다. 토지용도로서 용도지역(1), 용도지역(2), 용도구역, 용도지구, 변경일자들을 특성으로 가지며, 이러한 특성값들은 지속적으로 축적된다. 토지표시로서 면적, 지목, 축척들의 특성을 갖는다. 토지등급은 등급수정년월일, 공시지가들의 특성을 가지며, 이러한 특성값들은 지속적으로 축적된다. 각 토지에는 건축된 다수의 건축물이 존재한다.

- 건축물: 식별자, 용도, 구조, 도면 등의 특성을 갖는다. 건축물 현황으로서 층별용도, 대지면적, 높이, 건축면적들의 특성을 가지며, 건축면적은 건폐율, 용적율들의 특성을 갖는다. 건축자 성명으로는 설계자, 감리자, 시공자들의 특성을 갖는다. 시공일자로서 착공일자, 허가일자들의 특성을 갖는다. 주차장은 대수, 옥내, 옥외, 면적들의 특성을 가지며, 이러한 특성값들은 지속적으로 축적된다. 정화시설로서 형식, 용량들의 특성을 갖는다. 승강기로서 용도, 대수들의 특성을 가지며, 이러한 특성값들은 지속적으로 축적된다. 각 건축물에는 다수의 부속건축물이 존재한다. 각 건축물에는 건축물을 소유하는 다수의 권리자가 존재한다.

4. 개념적 설계

본 장에서는 제 3장에서 제시한 제반 요구 사항들을 기반으로 도시토지정보관리를 위한 개념적 설계 결과를 제시한다. 본 연구에서는 개념적 데이타베이스 설계를 위하여 가장 널리 사용되는 개체-관계 모델(entity-relationship model)을 이용하였으며, 이 모델에서 사용되는 각 용어 및 기호는 참고 문헌[7]에서 제시한 것을 참조하였다.

4.1. 개체 타입의 결정

요구 사항 분석을 통하여 도시토지정보관리에서 다루어지는 주요 객체들은 권리자, 토지, 건축물 등으로 파악되었으므로 이들을 각각의 이름을 갖는 개체 타입(entity type)으로 정의한다.

- 권리자: 성명, 주민등록번호, 주소 등은 권리자의 특성에 해당되므로 각각 해당 이름을 갖는 애트리뷰트(attribute)로 유지한다. 주민등록번호는 권리자를 유일하게 식별할 수 있으므로 키(key)로 지정한다. 각 권리자에 대응되는 다수의 소유권현황, 저당권, 근저당권에 관한 정보는 토지라는 다른 개체에서 참조가 가능하므로 이를 별도의 애트리뷰트로 지정하지 않고, 권리자와 토지간의 관계(relationship)로써 해결한다. 그림 4.1은 이것을 개체-관계 다이어그램(entity-relationship diagram)으로 표현한 것이다.

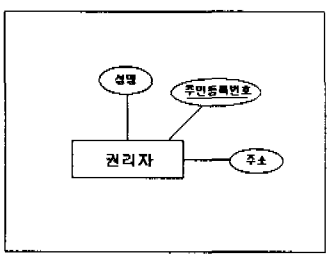

그림 4.1 권리자 개체

- 토지: 식별자 등은토지의 특성에 해당되므로 해당 이름을 갖는 애트리뷰트로 유지한다. 식별자는 토지를 유일하게 식별할 수 있으므로 키로 지정한다. 토지소재지 정보는 도, 시, 동, 지번 등으로 구분이 가능하므로 복합 애트리뷰트(composite attribute)로 지정한다. 토지용도정보는 용도지역(1), 용도지역(2), 용도구역, 용도지구, 변경일자 등으로 구분이

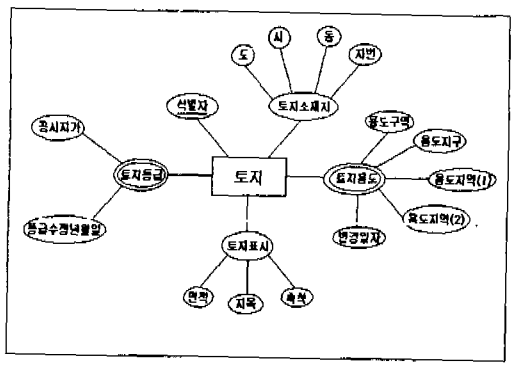

그림 4.2 토지 개체

가능하므로 복합 애트리뷰트로 지정한다. 또한, 이러한 특성값들은 지속적으로 축적되는 특성을 가지므로 집합 애트리뷰트(set attribute)로 지정한다. 토지표시정보는 면적, 지목, 축척 등으로 구분이 가능하므로 복합 애트리뷰트로 지정한다. 토지등급정보는 등급수정년월일과 공시지가 등으로 구분이 가능하므로 복합 애트리뷰트로 지정한다. 또한, 이러한 특성값들은 지속적으로 축적되는 특성을 가지므로 집합 애트리뷰트로 지정한다. 각 토지에 건축되는 다수의 건축물들에 관한 정보는 건축물이라는 다른 개체에서 참조가 가능하므로 이를 별도의 애트리뷰트로 지정하지않고, 토지와 건축물간의 관계로 해결한다. 그림 4.2는 이것을 개체-관계 다이어그램으로 표현한 것이다.

- 건축물: 식별자, 용도, 구조, 도면 등은 건축물의 특성에 해당되므로 각각 해당 이름을 갖는 애트리뷰트로 유지한다. 식별자는 건축물을 유일하게 식별할 수 있으므로 키로 지정한다. 건출물 현황정보는 층별용도, 대지면적, 높이, 건축면적 등으로 구분이 가능하므로 복합 애트리뷰트로 지정한다. 또한, 건축면적정보는 건폐율과 용적율 등으로 구분이 가능하므로 복합 애트리뷰트로 지정한다. 건축자 성명정보는 설계자, 감리자, 시공자 등으로 구분이 가능하므로 복합 애트리뷰트로 지정한다. 시공일자정보는 착공일자와 허가일자 등으로 구분이 가능하므로 복합 애트리뷰트로 지정한다. 주차장정보는 대수, 옥내, 옥외, 면적등으로 구분이 가능하므로 복

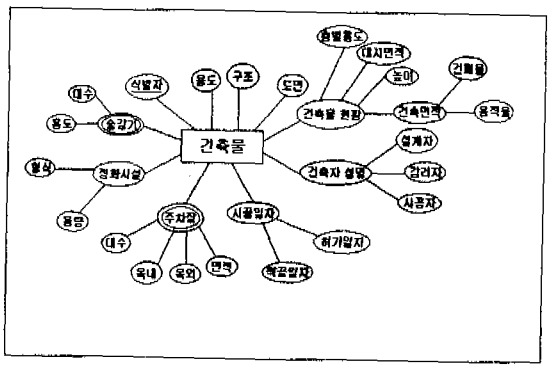

그림 4.3 건축물 개체

합 애트리뷰트로 지정한다. 또한, 이러한 특성값들은 지속적으로 축적되는 특성을 가지므로 집합 애트리뷰트로 지정한다. 정화시설 정보는 형식과 용량 등으로 구분이 가능하므로 복합 애트리뷰트로 지정한다. 승강기정보는 용도, 대수 등으로 구분이 가능하므로 복합 애트리뷰트로 지정한다. 또한, 이러한 특성값들은 지속적으로 축적되는 특성을 가지므로 집합 애트리뷰트로 지정한다. 각 건축물을 소유하는 다수의 권리자에 대한 정보는 권리자라는 다른 개체에서 참조가 가능하므로 이를 별도의 애트리뷰트로 지정하지 않고, 건축물과 권리자간의 관계로서 해결한다. 그림 4.3은 이것을 개체-관계 다이어그램으로 표현한 것이다.

4.2. 관계 타입의 결정

요구 분석의 결과 , 세 개의 개체 타입 권리자,토지, 건축물에는 다음과 같은 여섯 개의 관계 타입인 소유권현황, 저당권, 근저당권, 건축, 부속건축, 소유가 존재함을 발견할 수 있다.

- 소유권현황: 권리자와 토지간의 관계를 표현한다. 변동원인과 변동일자 등은 소유권현황의 특성에 해당하므로 해당이름을 갖는 애트리뷰트로 유지한다. 각 권리자는 다수의 토지를 소유할 수 있으며, 각 토지는 다수의 권리자에 소유될 수 있으므로 권리자와 토지로는 M:N의 사상 카디날리티(mapping cardinality)를 가진다. 권리자는 토지를 소유하지 않는 경우도 있으므로 선택적으로 관

계에 참여(partial participation)한다. 반면, 토지는 반드시 권리자에 소유되어야 하므로 필수적으로 관계에 참여(total participation)한다.

- 저당권: 권리자와 토지간의 관계를 표현한다. 등기원인 등은 저당권의 특성에 해당하므로 해당이름을 갖는 애트리뷰트로 유지한다. 각 권리자는 다수의 토지에 대한 저당권을 가지고 있으며, 각 토지는 다수의 권리자에 의해 저당권을 잡힐 수 있으므로 권리자와 토지로는 M:N의 사상 카디날리티를 가진다. 일부 권리자만이 토지의 저당권을 가지므로 선택적으로 관계에 참여한다. 또한, 일부 토지만이 권리자에의해 저당권을 잡힐 수 있으므로 선택적으로 관계에 참여한다.

- 근저당권: 권리자와 토지간의 관계를 표현한다. 접수년월일, 채권금액, 채무자, 권리자 등은 근저당권의 특성에 해당하므로 각각 해당이름을 갖는 애트리뷰트로 유지한다. 각 권리자는 다수의 토지에 대한 근저당권을 가지고 있으며, 각 토지는 다수의 권리자에 의해 근저당권을 잡힐 수 있으므로 권리자와 토지로는 M:N의 사상 카디날리티를 가진다. 일부 권리자만이 토지의 근저당권을 가지므로 선택적으로 관계에 참여한다. 또한, 일부 토지만이 권리자에의해 근저당권을 잡힐 수 있으므로 선택적으로 관계에 참여한다.

- 건축: 토지와 건축물간의 관계를 표현한다. 각 토지에는 여러개의 건축물이 건축되며, 각 축물은 다수의 토지에 건축될 수 있으므로 토지와 건축물로는 M:N의 사상 카디날리티를 가진다. 모든 토지에 반드시 건축물이 건축되는 것이 아니므로 선택적으로 관계에 참여한다. 반면, 모든 건축물은 반드시 토지에 건축되므로 필수적으로 관계에 참여한다.

- 부속건축: 건축물과 부속건축물간의 관계를 표현한다. 각 건축물에는 다수의 부속건축물이 있을 수 있으며, 각 부속건출물은 하나의 건축물에 있으므로 건축물과 부속건축물로는 1:N의 사상 카디날리티를 가진다. 모든 건축물에 부속건축물이 존재하는 것은 아니므로 선택적으로 관계에 참여한다. 또한, 모든 건축물이 반드시 어떤 건축물에 부속되는 것은 아니므로 선택적으로 관계에 참여한다.

- 소유: 건축물과 권리자간의 관계를 표현한다. 각 건축물은 다수의 권리자에 의해 소유될 수 있으며, 각 권리자는 다수의 건축물을 소유할 수 있으므로 건축물과 권리자로는 M:N의 사상 카디날리티를 가진다. 모든 건축물은 반드시 권리자에의해 소유되어야 하므로 필수적으로 관계에 참여한다. 반면, 권리자가 반드시 건축물을 소유하지 않는 경우도 있으므로 선택적으로 관계에 참여한다.

그림 4.4는 도시토지정보관리에서 나타나는 이러한 개체들간의 관계들을 개체-관계 다이어그램의 형태로 표현한 것이다.

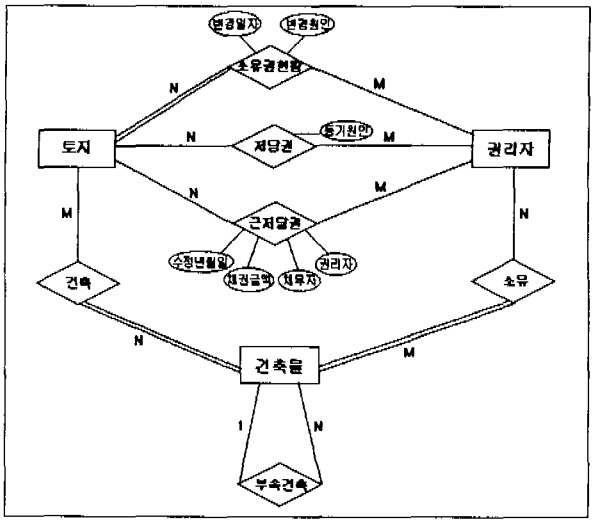

그림 87.4 개체들 간의 관계

5. 논리적 데이타베이스 설계

본 장에서는 제 4장에서 제시한 개념적 설계 결과를 기반으로 도시토지정보관리용 데이타베이스를 위한 논리적 설계를 수행한다. 본 연구에서는 논리적 설계를 위하여 현재 가장 많은 상용 DBMS에서 제공하는 관계 모델(relational model)을 이용하였다.

5.1. 개체 타입의 변환

- 권리자: 단순 애트리뷰트 성명, 주민등록번호, 주소 등을 애트리뷰트들로 가지는 릴레이션으로 변환한다. 주민등록번호를 기본 키(primary key)로 지정한다.
- 토지: 단순 애트리뷰트 식별자 등을 애트리뷰트들로 가지는 릴레이션으로 변환한다. 식별자를 기본 키로 지정한다.
- 건축물: 단순 애트리뷰트 식별자, 용도, 구조, 도면 등을 애트리뷰트들로 가지는 릴레이션으로 변환한다. 식별자를 기본 키로 지정한다.

5.2. 복합 애트리뷰트의 변환

- 토지 개체내의 토지소재지정보: 복합 애트리뷰트이지만 집합 애트리뷰트는 아니므로 도, 시, 동, 지번 등을 토지 릴레이션의 애트리뷰트로 지정한다.
- 토지 개체내의 토지표시정보: 복합 애트리뷰트이지만 집합 애트리뷰트는 아니므로 면적, 지목, 축척 등을 토지 릴레이션의 애트리뷰트로 지정한다.
- 건축물 개체내의 건축물 현황정보: 복합 애트리뷰트이지만 집합 애트리뷰트는 아니므로 층별용도, 대지면적, 높이, 건폐율, 용적율 등을 건축물 릴레이션의 애트리뷰트로 지정한다.
- 건축물 개체내의 건축자 성명정보: 복합 애트리뷰트이지만 집합 애트리뷰트는 아니므로 설계자, 감리자, 시공자 등을 건축물 릴레이션의 애트리뷰트로 지정한다.
- 건축물 개체내의 시공일자정보: 복합 애토리뷰트이지만 집합 애트리뷰트는 아니므로 착공일자와 허가일자들을 건축물 릴레이션의 애트리뷰트로 지정한다.
- 건축물 개체내의 정화시설정보: 복합 애트리뷰트이지만 집합 애트리뷰트는 아니므로 정화시설형식과 정화용량들을 건축물 릴레이션의 애트리뷰트로 지정한다.

5.3. 집합 애트리뷰트의 변환

- 토지 개체내의 토지용도: 두가지 용도지역과 용도구역, 용도지구, 토지용도, 변경일자와 토지 릴레이션의 기본 키인 토지식별자 등을 애트리뷰트로 갖는 토지용도라는 릴레이션으로 지정한다. 이때, 변경일시와 토지식별자를 묶어 기본 키로 지정한다.
- 토지 개체내의 토지등급: 공시자가와 등급수정년월일, 그리고 토지 릴레이션의 기본 키인 토지식별자 등을 애트리뷰트로 갖는 토지등급이라는 릴레이션으로 지정한다. 이때, 등급수정년월일과 토지식별자를 묶어 기본 키로 지정한다.
- 건축물 개체내의 주차장: 대수, 옥내, 옥외, 면적, 그리고 건축물 릴레이션의 기본 키인 건축물식별자 등을 애트리뷰트로 갖는 건축물주차장이라는 릴레이션으로 지정한다. 이때, 대수와 건축물식별자를 묶어 기본 키로 지정한다.
- 건축물 개체내의 승강기: 용도, 대수, 그리고 건축물 릴레이션의 기본 키인 건축물식별자 등을 애트리뷰트로 갖는 건축물주차장이라는 릴레이션으로 지정한다. 이때, 승강기대수와 건축물식별자를 묶어 기본 키로 지정한다.

5.4. 관계 타입의 변환

- 소유권현황: 권리자와 토지간의 M:N의 관계이므로 새로운 릴레이션으로 사상시킨다. 이 릴레이션은 변동일자, 변동원인, 그리고 관계에 참여하던 두 개체의 키 권리자주민등록번호와 토지식별자들을 자신의 애트리뷰트로 가진다. 이때, 권리자주민등록번호와 토지식별자를 묶어 기본 키로 지정한다.

- 저당권: 권리자와 토지간의 M:N의 관계이므로 릴레이션으로 사상시킨다. 이 릴레이션은 등기원인, 그리고 관계에 참여하던 두 개체의 키 권리자주민등록번호와 토지식별자들을 자신의 애트리뷰트로 가진다. 이때, 권리자주민등록번호와 토지식별자를 묶어 기본 키로 지정한다.

- 근저당권: 권리자와 토지간의 M:N의 관계이므로 릴레이션으로 사상시킨다. 이 릴레이션은 접수년월일, 채권금액, 채무자, 권리자, 그리고 관계에 참여하던 두 개체의 키 권리자주민등록번호와 토지식별자들을 자신의 애트리뷰트로 가진다. 이때, 권리자주민등록번호와 토지식별자를 묶어 기본 키로 지정한다.

- 건축: 토지와 건축물간의 M:N의 관계이므로 릴레이션으로 사상시킨다. 이 릴레이션은 관계에 참여하던 두 개체의 키 토지식별자와 건축물식별자들을 자신의 애트리뷰트로 가진다. 이때, 토지식별자와 건축물식별자를 묶어 기본 키로 지정한다.

- 부속건축: 건축물과 부속건축물간의 1:N의 관계이므로 모건축물을 건축물 릴레이션내에 외래 키(foreign key) 애트리뷰트로 추가시킨다.

- 소유: 건축물과 권리자간의 M:N의 관계이므로 릴레이션으로 사상시킨다. 이 릴레이션은 관계에 참여하던 두 개체의 키 건축물식별자와 권리자주민등록번호들을 자신의 애트리뷰트로 가진다. 이때, 건축물식별자와 권리자주민등로번호를 묶어 기본 키로 지정한다.

그림 5.1은 논리적 데이터베이스 설계를 위하여 위와 같은 변환 과정을 통하여 생성한 결과인 릴레이션들을 나타낸 것이다. 밑줄친 애트리뷰트 각 릴레이션의 기본 키를 구성하는 애트리뷰트임을 의미하며, 굵은 글자체의 속성은 외래 키에 해당되는 애트리뷰트임을 의미한다.

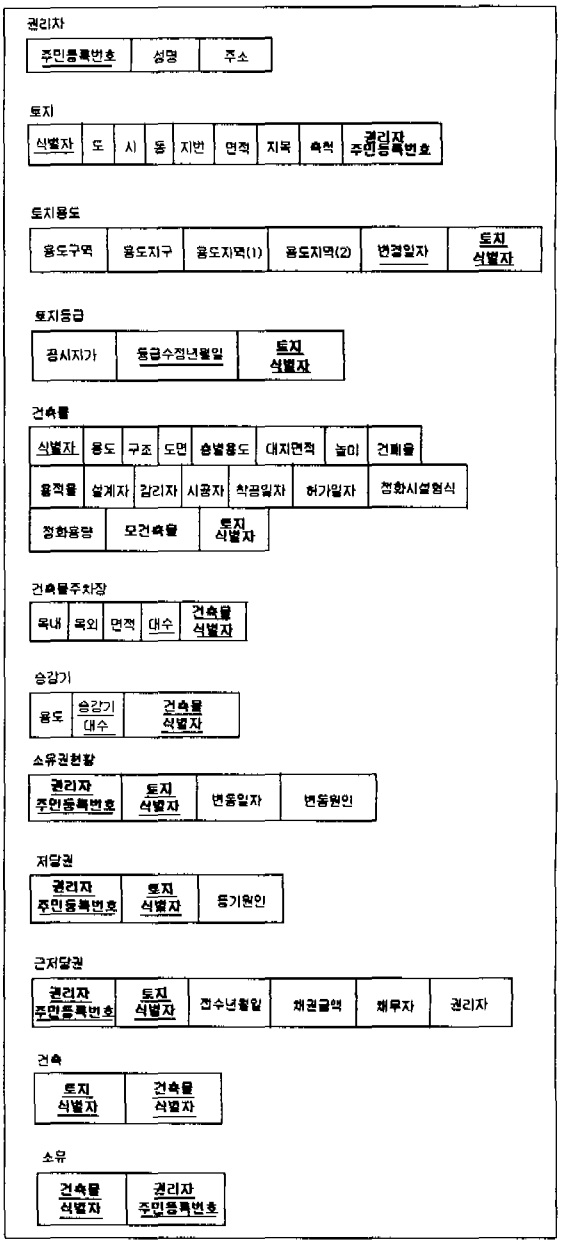

그림 5.1 논리적 데이터베이스 설계 결과

6. 물리적 데이타베이스 설계

본 장에서는 제 5장에서 제시한 논리적 설계 결과를 기반으로 도시토지정보관리를 위한 물리적 설계를 수행한다. 본 연구에서는 현재 대부분의 상용 DBMS에서 제공하는 클러스터링 인덱스(clustering index)와 이차 인덱스(secondary index)를 어떻게 제공할 것인가를 제시한다.

먼저, 물리적 데이타베이스 설계를 위하여 도시토지정보관리 시스템에서 사용이 예상되는 가상의 질의 집합을 다음과 같은 표준화된 SQL 문(structured query language statement)[11] 형태로 정의한다. 여기서 나타나는 X와 Y는 질의내의 조건

에서 사용되는 입의의 값을 의미한다. 편의상 각 질의가 시스템내에서 발생하는 빈도는 같다고 가정한다.

질의 1:

SELECT 권리자.성명, 권리자.주소, 토지.지번

FROM 권리자, 소유권현황, 토지

WHERE 권리자.주민등록번호=소유권현황.권리자주민등록번호 AND 소유권현황.토지식별자=토지.식별자

질의 2:

SELECT 권리자.주민등록번호, 권리자.성명, 저당권.등기원인

FROM 권리자, 저당권

WHERE 권리자.주민등록번호=저당권.권리자주민등록번호

질의 3:

SELECT 근저당권.채무자, 근저당권.권리자, 토지.축척

FROM 권리자, 근저당권, 토지

WHERE 권리자.주민등록번호=근저당권.권리자주민등록번호 AND 근저당권.토지식별자=토지식별자 AND 권리자.성명=X

질의 4:

SELECT *

FROM 권리자

WHERE 권리자.성명=X

질의 5:

SELECT 토지.도, 토지.시, 토지.동, 토지용도.용도구역

FROM 토지, 용도구역

WHERE 토지.식별자=토지용도.토지식별자

질의 6:

SELECT 토지.축척, 토지등급.공시지가

FROM 토지등급, 토지

WHERE 토지등급.등급수정년월일=Y AND 토지.식별자=토지등급.토지식별자

질의 7:

SELECT *

FROM 토지용도

WHERE 토지용도.변경일자=Y

질의 8:

SELECT 토지.지번, 건축물.용도, 건축물.구조, 건축물.높이

FROM 토지, 건축, 건축물

WHERE 토지.식별자=건축.토지식별자 AND 건축.건축물식별자=건축물.식별자

질의 9:

SELECT *

FROM 토지

WHERE 토지.지번=Y

질의 10:

SELECT *

FROM 건축물

WHERE 건축물.착공일자=Y

질의 11:

SELECT 건축물.층별용도, 건축물.정화용량, 건축물주차장면적

FROM 건축물, 건축물주차장

WHERE 건축물.식별자=건축물주차장.건축물식별자

질의 12:

SELECT 건축물.건폐율, 건축물.용적율

FROM 건축물, 승강기

WHERE 승강기.승강기대수=X AND 건축물.식별자=승강기.건축물식별자

질의 13:

SELECT 옥내, 옥외, 면적

FROM 건축물주차장

WHERE 건축물주차장.대수=Y

질의 14:

SELECT 토지.면적, 건축물.높이, 권리자.성명

FROM 토지, 건축, 건축물, 소유, 권리자

WHERE 토지.식별자=건축.토지식별자 AND

건축.건축물식별자=건축물.식별자 AND

소유.권리자주민등록번호=권리자.주민등록번호

질의 15:

SELECT 건축물.구조, 권리자.주민등록번호

FROM 건축물, 소유, 권리자

WHERE 권리자.성명=X AND

건축물.식별자=소유.건축물식별자 AND

소유.권리자주민등록번호=권리자.주민등록번호

다음은 위의 질의 집합이 사용된다는 가정하에 앞부분에서 언급한 지침을 이용하여 물리적 데이타베이스 설계를 수행한 최종 결과를 나타낸다.

- 권리자: 질의 1, 2, 3, 14, 15에서 주민등록번호가 조인 조건에서 사용된다. 같은 주민등록번호를 가지는 권리자는 단 하나이므로 주민등록번호에 이차 인덱스를 생성한다. 질의 3, 4, 15에서 성명이 조건에 사용되므로 성명에 이차 인덱스를 생성한다.

- 토지: 질의 1, 5 ,6, 8, 14에서 식별자가 조인 조건에서 사용된다. 같은 식별자를 가지는 토지는 단 하나이므로 식별자에 이차 인덱스를 생성한다. 질의 9에서 지번이 조건에 사용되므로 지번에 이차 인덱스를 생성한다.

- 토지용도: 질의 5에서 토지식별자가 조인 조건에서 사용된다. 같은 토지식별자를 가지는 다수의 토지용도가 존재하므로 토지식별자에 클러스터링 인덱스를 생성한다. 질의 7에서 변경일자가 조건에 사용되므로 변경일자에 이차 인덱스를 생성한다.

- 토지등급: 질의 6에서 토지식별자가 조인 조건에서 사용된다. 같은 토지식별자를 가지는 다수의 토지등급이 존재하므로 토지식별자에 클러스터링 인덱스를 생성한다. 질의 6에서 등급수정년월일이 조건에 사용되므로 등급수정년월일에 이차 인덱스를 생성한다.

- 건축물: 질의 8, 11, 12, 14, 15에서 식별자가 조인 조건에서 사용된다. 같은 식별자를 가지는 건축물은 단 하나이므로 식별자에 이차 인덱스를 생성한다. 질의 10에서 착공일자가 조건에 사용되므로 착공일자에 이차 인덱스를 생성한다.

- 건축물주차장: 질의 11에서 건축물식별자가 조인 조건에서 사용된다. 같은 건축물식별자를 가지는 다수의 건축물주차장이 존재하므로 건축물식별자에 클러스터링 인덱스를 생성한다. 질의 13에서 대수가 조건에서 사용되므로 대수에 이차 인덱스를 생성한다.

- 승강기: 질의 12에서 건축물식별자가 조인 조건에서 사용된다. 같은 건축물식별자를 가지는 다수의 승강기가 존재하므로 건축물식별자에 클러스터링 인덱스를 생성한다. 질의 12에서 승강기대수가 조건에서 사용되므로 대수에 이차 인덱스를 생성한다.

- 소유권현황: 질의 1에서 권리자주민등록번호와 토지식별자가 조인 조건에서 사용된다. 같은 권리자주민등록번호와 토지식별자를 가지는 다수의 소유권현황이 존재하므로 각각의 식별자에 클러스터링 인덱스를 생성한다.

- 저당권: 질의 2에서 권리자주민등록번호가 죠인 조건에서 사용된다. 같은 권리자주민등록번를 가지는 다수의 저당권이 존재하므로 권리자주민등록번호에 클러스터링 인덱스를 생성한다.

- 근저당권: 질의 3에서 권리자주민등록번호와 토지식별자가 죠인 조건에서 사용된다. 같은 권리자주민등록번호와 토지식별자를 가지는 다수의 근저당권이 존재하므로 각각의 식별자에 클러스터링 인덱스를 생성한다.

- 건축: 질의 8, 14에서 토지식별자와 건축물식별자가 죠인 조건에서 사용된다. 같은 토지식별자와 건축물식별자를 가지는 다수의 건축이 존재하므로 각각의 식별자에 클러스터링 인덱스를 생성한다.

- 소유: 질의 14, 15에서 건축물식별자와 권리자주민등록번호가 죠인 조건에서 사용된다. 같은 건축물식별자와 권리자주민등록번호를 가지는 다수의 소유가 존재하므로 각각의 식별자에 클러스터링 인덱스를 생성한다.

7. 결 론

도시토지정보관리 시스템을 원활히 수행하기 위하여 효율적인 데이타의 관리가 요구된다. 효율적인 데이타 관리를 위해서는 이들 데이타를 위한 최적의 데이타베이스 구축이 선행되어야 한다. 본 논문에서는 도시토지정보관리를 위한 최적의 데이타베이스 구축을 위한 데이타베이스 설계에 관하여 논의하였다.

먼저, 요구 사항 분석을 통하여 도시토지정보관리에서 사용되는 데이타 집합을 정의하고, 이 데이타 집합을 대상으로 개체-관계 모델을 이용한 개념적 설계 과정을 제시하였다. 또한, 개념적 설계의 결과를 대상으로 관계 모델을 이용한 논리적 데이타베이스 설계 과정을 제시하였으며, 논리적 데이타베이스 설계를 제시하였으며, 논리적 데이타베이스 설계의 결과와 가정하는 질의 집합을 이용하여 물리적 데이타베이스 설계 과정을 제시하였다.

이러한 연구 결과는 실제 도시토지정보관리 시스템에 유용하게 활용될 수 있을 것이다. 본 연구 결과를 이용하여 도시토지정보관리를 위하여 필요한 각종 데이타를 파악할 수 있으며, 이들을 효과적으로 데이타베이스화 할 수 있다.

참고 문헌

[1] 이석호, *데이타베이스 개론*, 정익사, 1996.

[2] 조규익, *데이타베이스 설계: 설계 실무 지침서*, 홍릉과학출판사, 1994.

[3] Batini, C., Ceri, S., and Navathe, S. B., *Conceptual Database Design: An Entity-Relationship Approach*, Benjamin/Cummings Publishing Company, 1992.

[4] P. Chen, "The Entity Relationship Model: Toward a Unified View of Data," *ACM Trans. on Database Systems*, Vol. 1, No. 1, 1976.

[5] Codd, E., "A Relational Model for Large Shared Banks," *Communications of the ACM*, Vol. 13, No. 6, 1970.

[6] Connolly, T., Begg, C., and Strachan, A., *Database Systems: A Practical Approach to Design, Implementation, and Management*, Addison-Wesley Publishing Company, 1996.

[7] Elmasri, R. and Navathe, S. B., *Fundamentals of Database Systems, Second Edition*, Benjamin/Cummings Publishing Company, 1994.

[8] Hansen, G. W. and Hansen, J. V., *Database Management and Design*, Prentice Hall International, 1996.

[9] Korth, F. H. and Silberschatz, A., *Database System Concepts, Second Edition*, McGraw-Hill Publishing Company, 1991.

[10] Kroenke, D. M., *Database Processing:*

Fundamentals, Design, and Implementation, Prentice Hall International, 1995.

11] Melton, J. and Simon, A. R., *Understanding the New SQL: A Complete Guide,* Morgan Kaufmann Publishers, 1993.

12] Ozsu, M. T. and Valduriez, P., *Principles of Distributed Database Systems,* Prentice Hall International, 1991.

13] Ullman, J. D., *Principles of Database and Knowledge-Base Systems,* Vol. 1, Computer Science Press, 1988.