

관계형 데이터베이스 설계를 위한 개체-관계 모델링 시스템 개발

유재건^{*}

한성대학교 산업및기계시스템공학부

Development of an Entity-Relationship Modeling System for Designing Relational Database

Jae-Gun Yoo

School of Industrial & Mechanical System Engineering, Hansung University, Seoul, 136-792

Entity-relationship modeling for designing relational database is a very complicated thinking process that requires extensive knowledge and experiences. It is very likely that designers make mistakes in this process. In order to minimize the mistakes, a systematic method to guide the thinking process is needed. In this research, an entity-relationship modeling system is developed, which resolves the whole process of information modeling, data modeling, and functional dependency relationship analysis into small and simple decision-making steps. Therefore, it can reduce the possibility of making decision errors and improve the efficiency of the modeling process. Its functionality and efficiency is verified through some modeling examples. It is expected that the modeling system can be commercialized, if some functions are added, such as detection, warning, and correction of decision errors, and educational help.

Keywords: information modeling, entity-relationship modeling, relational database, database design

1. 개체-관계 모델링 시스템 개발의 배경

데이터베이스 설계는 정보 시스템에 대한 사용자의 정보 요구 사항에 따라 데이터베이스 논리적 구조를 결정하는 과정이다. 사용자의 정보 요구사항으로부터 독립 항목들로 이루어진 데이터 항목의 집합이 추출되면, 각 항목은 정보 모델링 과정을 통하여 개체와 관계의 속성으로 분류되어 개체관계도로 표현된다. 이 개체관계도는 데이터 모델링 과정을 거쳐 관계형 테이블 구조가 생성된다.

데이터 모델링 과정은 매우 단순한 규칙에 의해 이루어지는데 비해 정보 모델링은 인간의 광범위한 지식과 경험이 요구되는 매우 복잡한 사고 과정이다(이춘열 *et. al.*, 2000). 그럼에도

불구하고 정보 모델링 과정의 결과를 개체관계도로 표현하고, 이를 테이블 구조로 변환하는 도구들은 상용화되어 있으나, 정보 모델링 과정을 지원하는 도구는 개발되어 있지 않다(Gomez, F. *et. al.*, 1999). 따라서 숙련된 설계자라고 하더라도 이 과정에서 많은 시행 착오를 거치게 되며, 결과물인 테이블 구조의 함수 종속 관계를 분석하는 추가 과정이 필요하다.

특히 정보 모델링 과정의 학습자들은 사고의 전개 방향을 찾지 못함으로써 상당한 혼란을 겪게 된다. 대부분 다양한 사례에 대한 모델링 과정을 경험한 후에야 나름대로의 사고방식을 습득하게 된다. 이러한 학습방법은 많은 시간과 시행착오가 요구된다. 특히 한정된 시간에 이루어지는 정보 모델링 교육에서 실질적인 모델링 능력의 배양은 기대하기 어렵다. 따

본 연구는 2002학년도 한성대학교 교내 특별연구비 지원과제임.

^{*}연락처 : 유재건, 136-792 서울 성북구 삼선동2가 389 한성대학교 산업및기계시스템공학부, Fax : 02-760-4490,

E-mail : jgyoo@hansung.ac.kr

라서 설계자나 학습자의 사고 전개 방향을 안내해 줄 수 있는 시스템의 개발이 필요하다.

본 연구에서는 데이터 항목 집합이 주어진 상태에서 보이스-코드 정규형을 만족하는 테이블 구조를 결정하기 위한 개체-관계 모델링 시스템을 개발하였다. 여기에서 테이블 구조란 테이블의 종류와 각 테이블의 열, 그리고 기본키, 외래키 및 대체키 열로 정의된다. 본 시스템은 정보 모델링과 데이터 모델링, 그리고 함수 종속 관계 분석의 기능을 동시에 수행한다. 복잡한 사고 과정을 단순하고 구체적인 판단 과정으로 분해하고 체계화함으로써 사용자는 시스템이 제시하는 몇 가지 질문에 대하여 답하는 반복적인 과정을 통하여 테이블 설계 작업을 완료하게 된다.

2. 이론적 배경

본 논문에서 관계는 관계 개체로 불리며 둘 이상의 단위 개체가 관련된다. 개체는 단위 개체와 관계 개체 모두를 지칭한다. 따라서 모든 데이터 항목(이하 데이터 항목은 항목으로 부른다)은 개체의 속성으로 분류되며, 곧 테이블의 열이 된다. 보이스-코드 정규형 테이블에서 얻을 수 있는 모든 정보는 f : 후보키후 \rightarrow 보기를 제외한 열의 함수 관계에 담겨 있다. 하나의 테이블에 또 다른 개체에 대한 정보가 포함되어서는 안 된다. 즉, 후보키를 제외한 열이 독립 변수인 함수 관계는 허용되지 않는다(Date, C. J., 1995).

이와 같이 개체의 속성은 개체 식별자와 함수 관계를 가지며, 역으로 개체 식별자와 함수 관계를 갖는 항목은 개체의 속성이 된다. 따라서 항목을 개체의 속성으로 분류할 때 개체의 식별자와의 함수 관계를 기준으로 판단하면 된다. 이와 같이 정량적이고 구체적인 규칙을 사용함으로써 사고 과정이 좀 더 명확하고 용이하게 이루어질 수 있다.

3. 시스템의 기본적인 논리 구조

본 시스템이 사용자에게 요구하는 판단은 크게 네 단계로 분류되며, 각각에 대한 시스템의 기본적인 논리에 대한 설명은 다음과 같다.

단계 1. 항목과 관련된 개체의 결정

항목과 관련된 개체를 결정한다. 관련된 개체의 수는 한 번에 하나로 제한한다. 이는 알고리즘의 일관성과 단순성을 유지하기 위한 것이다. 항목과 관련된 하나의 단위 개체를 결정한 후 항목이 그 개체의 속성으로 분류될 수 있는지 판단한다. 그렇지 않다면 항목과 관련된 또 다른 단위 개체를 결정하고 두 개의 단위 개체가 관련된 하나의 관계 개체를 정의한다. 그리고 항목이 이 관계 개체의 속성으로 분류될 수 있는지 판단

한다. 따라서 한번에 하나의 개체에 대하여 판단하게 된다. 이러한 과정을 반복하면 세 개 이상의 단위 개체가 관련된 관계도 하나의 관계 개체로 정의할 수 있다(Jones, T.H. et. al., 1996).

관계 개체를 정의할 때 단위 개체의 인스턴스가 하나씩만 포함된 명제를 하나의 관계 개체 인스턴스로 한다는 개념을 시스템 사용자가 명확하게 이해하는 것이 중요하다. 관계 개체를 이와 같이 정의함으로써 관계 개체와 단위 개체 간에는 항상 일대일 또는 일대다의 관계를 갖게 되고, 단위 개체의 기본키는 관계 개체의 속성으로 존재하게 된다.

단위 개체를 생각해내는 과정은 시스템 사용자의 지식과 경험에 의존된다. 이 과정에서 사용자는 항목과 관련 단위 개체들이 포함된 명제를 만들어보고, 명제에 포함된 정보가 데이터베이스에 저장되어야 할 정보를 담고 있는지 판단해야 한다.

단계 2. 개체의 기본키 속성의 결정

단위 개체의 기본키는 사용자가 판단한다. 그러나 관계 개체의 기본키는 시스템에 의해 단위 개체의 기본키 집합으로 결정된다. 사용자가 관계 개체의 기본키를 별도로 지정할 수도 있는데, 이 경우 단위 개체의 기본키 집합은 대체키가 된다. 물론 기본키가 아닌 개체의 식별자 속성들은 대체키가 된다.

단계 3. 개체의 기본키와 항목 간 함수 관계의 판단

개체 인스턴스들의 기본키 값의 집합을 E라 하고, 항목값의 집합을 I라고 할 때, E의 원소에 대응되는 I의 원소의 개수에 따라 다음과 같이 차수(cardinality)가 결정된다.

- 0-1: 대응 원소가 없거나 하나인 경우
- 1-1: 대응 원소가 항상 하나인 경우
- 0-n: 대응 원소가 없거나 하나 이상인 경우
- 1-n: 대응 원소가 하나 이상인 경우

차수에 따라 다음과 같이 개체 및 속성을 결정한다.

- 차수가 0-1 또는 1-1인 경우, 개체의 기본키를 독립 변수, 항목을 종속 변수로 하는 함수 관계가 존재하며, 항목은 개체의 속성이 된다.
- 차수가 0-n 또는 1-n인 경우, 항목을 어떤 기준에 의해 충분히 적은 수의 세부 항목으로 분해할 수 있고 개체의 기본키를 독립 변수, 세부 항목을 종속 변수로 하는 함수 관계가 성립된다면 각 세부 항목은 개체의 속성이 된다.
- 그렇지 않다면 이 항목은 또 다른 단위 개체와도 관련되어야 하며, 이는 새로운 관계 개체가 정의되어야 함을 의미한다.

단계 4. 개체 기본키 간 함수 관계의 판단

항목과 무관하게 두 개의 단위 개체가 관련된 새로운 명제가 데이터베이스에 추가로 저장되어야 할 정보를 나타낼 수 있다. 위의 세 단계를 통하여 추출된 개체의 집합에서 두 개의 개체를 선택하여 개체 1 기본키에 대한 개체 2 기본키의 차수

와 개체 2 기본키에 대한 개체 1 기본키의 차수를 결정한다. 차수에 따라 다음과 같이 개체 및 속성을 결정한다.

- 두 차수 중 하나가 0-1 또는 1-1인 경우, 두 개체 기본키 간에 함수 관계가 존재하며, 종속 변수에 해당하는 개체의 기본키는 독립 변수에 해당하는 개체의 속성이 된다.
- 두 차수가 모두 0-n 또는 1-n인 경우, 두 개체가 관련된 새로운 관계 개체가 정의되어야 한다. 두 개체의 기본키는 새로운 관계 개체의 속성이 된다. 두 개체의 동일한 인스턴스들이 포함된 관계 개체 인스턴스가 복수 개 존재할 수 있다면, 이 복수 개의 관계 개체 인스턴스들을 구별하는 기준을 또 다른 관련 개체로 추가하여야 한다.

4. 시스템 내부 알고리즘

시스템의 입력 데이터는 입력 테이블에 출력 데이터는 출력 테이블에 저장된다. 입력 테이블의 열 집합은 {항목, 처리}이다. {항목} 열은 데이터 항목을, {처리} 열은 알고리즘이 적용된 횟수를 나타낸다. {항목} 열은 기본키 제약조건을 갖는다. 즉 항목 값은 모두 서로 다른 값을 갖는다. 출력 테이블의 열 집합은 {개체, 속성, 기본키, 외래키, 외래키 개체, 대체키, 대체키 구분}이다. {개체} 열은 테이블명, {속성} 열은 열 이름, {기본키}/{외래키}/{대체키} 열은 각각 기본키/외래키/대체키 여부를 나타낸다. {외래키 개체} 열은 외래키에 대한 참조 개체를 나타내며, {대체키 구분} 열은 두 개 이상의 대체키가 하나의 개체에 있을 경우 이를 구분하기 위한 것이다.

다음은 시스템 내부 알고리즘에 대한 설명이다. 시스템 사용자의 판단이 필요한 부분은 밑줄로 표시하였다. 스텝 번호가 일련번호가 아닌 것은 간략한 설명을 위해서 일부 스텝들이 다른 스텝의 설명에 포함되었기 때문이다.

(I=입력 테이블, M=중간 테이블, O=출력 테이블, PK=기본키, FK=외래키, AK=대체키, /=또는)

- (P1) I에 데이터 항목을 일괄 입력.
- (P2) I에서 항목을 선택하거나 새로 생성. D1.
- (D1) I 처리가 완료되었으면 M을 생성하고 D24, 아니면 D=선택/생성된 항목 P6.
- (P6) E1 = O에서 선택/생성된 개체. D4.
- (D4) D가 E1의 식별자이면 O에 D를 E1의 PK/AK로 추가하고 P2. 아닌 경우 선택된 E1이면 D9, 아니면 E=E1 후 P32.
- (D9) 'f: E1의 PKD'이면 P16, 아니면 D10.
- (P16) D에 FK 제약이 필요하다면 O에 D를 E1의 FK 속성으로 추가. 아니면 O에 D를 E1의 속성으로 추가. P2.
- (D10) 분해 가능하고 함수 관계가 성립하면 P20, 아니면 P21.
- (P20) D를 분해한 세부 항목들을 입력. P22.
- (P21) E2=O에서 선택/생성된 D 관련 추가 개체. 선택된 E2

이면 P25. 아닌 경우 D가 E2의 PK이면 O에 D를 PK로 추가하고 P25, 아니면 E=E2 후 P32.

- (P22) D에 FK 제약이 필요하다면 O에 세부 항목들을 E1의 FK 속성으로 추가. 아니면 O에 세부 항목들을 E1의 속성으로 추가. P2.
- (P25) R=O에서 선택/생성된 E1, E2 간의 관계 개체. 생성된 R이면 E1, E2의 PK를 R의 PK 및 FK 속성으로 추가. 아닌 경우 D가 E2의 식별자이면 D를 E2의 PK/AK로 추가하고 P2, 아니면 E1 = R 후 P6.
- (P32) E의 PK를 O에서 선택/생성 후 E의 기본키로 O에 추가. E=E1이면 D9, E = E2이면 P25.
- (D24) M 처리 완료이면 알고리즘 종료.
- (D25) E1, E2가 관련된 의미있고 새로운 명제가 존재하면 P40, 아니면 D24.
- (P40) E1, E2 간의 차수 C1, C2과 차수의 평균 m1, m2의 비교 우위를 결정. 결정된 값에 따라 P42~P45, P47중 하나를 수행한 후 D24.
- (P42) E2의 PK를 E1의 FK 속성으로 추가.
- (P43) E1의 PK에 FK 속성을 추가.
- (P44) E1의 PK를 E2의 FK 속성으로 추가.
- (P45) E2의 PK에 FK 속성을 추가.
- (P47) R=생성된 E1, E2 간의 관계 개체. E1, E2의 PK들을 R의 PK 및 FK 속성으로 추가. D34.
- (D34) 동일한 E1, E2 인스턴스들이 참여한 R 인스턴스가 하나이면 D24, 아니면 P52.
- (P52) E2=O에서 선택/생성된 개체(E2는 복수의 R 인스턴스를 구분하는 기준이 되는 개체). 생성된 E2이면 E2의 PK를 O에 추가. E1=R 후 D24.

5. 사례를 통한 시스템 검증

학사 관리, 대여 관리, 여객기 예약 등의 사례에 대하여 시스템을 적용하였다. 각각에 대하여 정보 요구사항에 변화를 주어 다양한 측면에서 시스템의 기능을 검증하였다.

학사 관리에 대한 하나의 적용 사례를 예로 시스템의 기능을 설명한다. 이 사례에서 입력 테이블의 항목들은 다음과 같다. 굵은 글꼴로 표시된 항목은 시스템 실행중 사용자에게 의해 추가된 것이다.

{학번, 성명, 학과명, 과목명, 성적, 주민등록번호, 연구실, 직장, **한글 성명, 한자 성명, 교수 번호, 학과 코드, 과목 코드, 학기 코드**}

알고리즘에서 얻어진 결과 집합으로부터 다음의 보이스코드 정규형 테이블 구조를 얻을 수 있다. 굵은 글꼴은 기본키 속성, 기울임 글꼴은 외래키 속성, 밑줄은 대체키 속성을 나타낸다.

학생 : {**학번**, 한글 성명, 한자 성명, 주민등록번호, 교수 번호}
 교수 : {**교수 번호**, 한글 성명, 한자 성명, 주민등록번호}
 과목 : {**과목 코드**, 과목명}
 학과 : {**학과 코드**, 학과명}
 수강 : {**학번**, **과목 코드**}
 학기 : {**학기 코드**}
 학기별 수강 : {**학번**, **과목 코드**, **학기 코드**, 성적}
 전임 교수 : {**교수 번호**, 연구실}
 겸임 교수 : {**교수 번호**, 직장}

학생 테이블의 교수 번호는 학생과 교수 간의 일대다 지도 관계를 나타낸다. 또한 한글과 한자 성명은 성명 복수값 속성이 세부 항목으로 분해되었음을 나타낸다. 학기별 수강은 학생, 과목, 학기 단위 개체들 간의 관계를 나타낸다. 전임 교수와 겸임 교수의 교수 번호의 외래키 제약 조건은 교수 개체와 두 개체 간의 슈퍼-서브 클래스 관계를 나타낸다. 이와 같이 기본키, 외래키, 대체키 속성의 정의, 복수값 속성의 분해, 관계 개체의 정의, 슈퍼-서브 클래스 관계의 정의 등의 기능이 사례를 통하여 검증되었다.

시스템 관련 자료들은 저자의 홈페이지(webmail.hansung.ac.kr/~jgyoo)에서 찾을 수 있다.

6. 결론 및 추후 연구 방향

정보 모델링 지원을 위한 상용화된 도구뿐 아니라 이와 관련된 연구 문헌도 찾을 수 없었다. 본 연구를 통하여 개발된 개체

-관계 모델링 시스템은 다양한 사례를 통하여 기능이 검증되었으며, 개념적이고 추상적인 정보 모델링도 컴퓨터 소프트웨어에 의해 지원될 수 있음을 보였다. 복잡한 모델링 과정을 구체적이고 체계적인 과정으로 변환함으로써 설계자가 작고 구체적인 일련의 문제에 집중할 수 있게 되고, 설계자의 판단 오류를 최소화할 수 있음이 입증되었다.

그러나 설계자의 판단이 필요한 부분에서 판단 오류가 발생할 수 있으므로, 오류의 발생을 시스템이 감지하고 통보하는 기능이라든지 판단 오류를 최소화할 수 있도록 안내하는 기능, 판단 오류를 수정할 수 있는 기능 등이 시스템에 추가되어야 할 것이다. 예를 들어 사용자의 판단 오류에 의하여 발생된 부분 종속 함수 관계 또는 이행 함수 종속 관계를 시스템이 사전에 방지하거나 감지하고, 오류의 자동 수정 기능이 추가되어야 한다. 이러한 과정을 통하여 설계자를 위한 실무용 또는 모델링 학습자를 위한 교육용으로 사용될 수 있도록 개선되고 나아가 상용화 시스템으로 발전될 수 있을 것으로 기대된다.

참고문헌

- 이춘열;이종옥(2000), *관계형 데이터베이스 관리론*, 홍릉과학출판사.
 Gomez, F.; Segami C.; Delaune, C.(1999), A system for the semiautomatic generation of E-R models from natural language specifications, *Data & Knowledge Engineering*, 29, 57~81.
 Date, C. J.(1995), *An Introduction to Database Systems*(6thEd), Addison- Wesley.
 Jones, T.H.; Il-Yeol, S.(1996), Analysis of binary/ternary cardinality combinations in entity-relationship modeling, *Data & Knowledge Engineering*, 19(1), 39~64.