

논문논문번호 18-6-3

관계 데이터베이스 시스템에서의 한글 응용 생성기 개발[†]
 (Development of Hangul Application Generator
 in Relational Database Systems)

박 상 현* 용 환 승** 이 석 호***
 (Sanghyun Park) (Hwan-Seung Yong) (Sukho Lee)

요약

데이터베이스를 이용한 대화식 응용을 작성하기 위해서는 프로그래밍 언어와 데이터베이스 시스템을 동시에 잘 알고 있어야 한다는 단점이 있다. 이러한 단점은 응용의 개발에 많은 어려움을 준다.

본 논문에서는 대화식 데이터베이스 응용을 손쉽게 작성할 수 있도록 하는 한글 응용 생성기에 관한 설계와 구현 사항에 대해 기술한다. 절차적인 언어를 이용해 응용을 작성하는 기존의 방법과는 달리 본 한글 응용 생성기에서는 한글 메뉴 인터페이스를 통해 메뉴와 양식을 명세화함으로써 응용을 작성할 수 있다. 응용은 메뉴와 양식으로 구성된 계층을 이루게 되는데 응용의 신속한 실행을 위해 응용을 구성하는 음의의 메뉴나 양식으로도 직접적인 분기가 가능하다. 또한 관계 데이터베이스의 기본 연산인 검색, 변경, 삽입, 삭제 등을 수행하는 질의어도 시스템에 의해 자동적으로 작성되어 실행된다.

ABSTRACT

In designing interactive database applications, a designer should know well about database system and programming language. This requirement forces to take long time to development applications.

This paper describes a design and implementation of an application generator, HAG(Hangul Application Generator), which supports rapid application development. Using this application generator, applications can be developed not by writing procedural codes but by specifying menus and forms through Hangul menu interfaces.

As a consequence, user can give queries against database without requiring any knowledge of the query syntax. In this system, an application is represented as a tree-like hierarchy which consists of menus and forms. Within this hierarchy, the user can traverse to any form or menu directly for rapid execution.

[†] 본 연구는 과학기술처 국책연구과제의 지원에 의한 결과임

* 준희원. 대우통신 연구원

** 종신회원. 서울대학교 컴퓨터공학과 박사과정

*** 종신회원. 서울대학교 컴퓨터공학과 교수

접수일자: 1991년 1월 24일

1. 서론

컴퓨터를 이용한 많은 응용이 개발되고 있는데 그중의 하나가 대화식 정보 시스템(IIS, Interactive Informa-

tion System)이다[6]. IIS 환경 하에서는 데이터베이스 접근(access), 데이터 저장(storing), 데이터 조작(manipulation), 제어(control), 데이터 표현(presentation) 등의 작업이 필요한데 단지 데이터베이스 관리 시스템(DBMS)만으로는 이를 요구를 완전히 처리할 수 없다[10]. 따라서 DBMS를 후위(backend) 시스템으로 하여 이를 요구를 효율적으로 처리할 수 있는 전위(frontend) 시스템 구축에 대한 연구가 활발하다. 이러한 전위 시스템은 데이터 접근을 위한 시스템, 데이터 표현을 위한 시스템, 응용 생성을 위한 시스템으로 분류할 수 있다[10].

데이터 접근이란 정보를 데이터베이스에 저장하고 필요시에 검색하는 것으로 이를 처리하기 위해 질의어(Query Language), 메뉴와 양식에 의한 인터페이스(Menu-and-forms-driven interface), 자연 언어 인터페이스(Natural language interface) 등이 사용된다[10]. 데이터 표현이란 데이터베이스에 저장된 데이터를 사용자가 알아보기 쉽게 나타내는 것으로 단순 테이블 형태, 문서 보고서 형태, 그래픽 보고서 형태, 응성 출력 형태 등이 존재한다. 응용 생성이란 대화식 데이터베이스 응용을 비절차적으로 생성하는 것으로 기존의 고급 언어에 비해 한단계 진보된 언어로 본다. 즉 사용자는 자기가 무엇(what)을 할 것인가만 명세하면 되고 어떻게(how) 할 것인가는 시스템에 의해 결정되는 것으로서 이를 4세대 언어(4th Generation Language)[1, 2, 10, 12]라고도 한다. 즉 프로그래밍 언어를 잘 모르는 초보자라도 데이터베이스 응용을 손쉽게 작성할 수 있도록 하는 것이 응용 생성 시스템의 목적이라고 할 수 있다.

본 논문에서는 데이터베이스 시스템을 후위 시스템으로 하여 데이터베이스 응용을 손쉽게 작성할 수 있는 한글 응용 생성기를 설계하고 구현한다[11, 13, 14, 15]. 한글 응용 생성기는 데이터 접근, 데이터 표현, 응용 생성 등의 데이터베이스 전위 시스템 구축에 필요한 기능을 지원함으로써 4세대 시스템 구축을 목적으로 한다. 또한 데이터베이스의 무결성 유지와 응용의 신속한 실행을 위한 방법도 제시된다. 데이터 접근 방법으로는 메뉴와 양식을 이용한 인터페이스를 사용하여 질의어에 익숙치 않은 초보자라도 쉽게 데이터베이스 연산을 수행할 수 있다. 이러한 데이터 접근 방법은 INGRES의 QBF(Query By Forms)[10]와 유사하다. 데이터 표현 방법으로는 화면 위에 데이터베이스 테이블을 보기 쉽게

배치한 양식을 사용한다[3, 6, 7]. 매릴랜드 대학의 Formanager[3]도 이와 동일한 방법을 사용한다. 응용 생성을 위한 방법으로 시스템이 입력에 필요한 정보를 안내하고 사용자는 작성에 필요한 내용만을 입력함으로써 응용이 생성되는 방법을 이용한다.

본 논문의 연구 내용을 요약하면 데이터베이스를 후위 시스템으로 하여 데이터베이스 응용을 손쉽게 작성할 수 있는 통합적인 개발 도구인 한글 응용 생성기를 설계하고 구현하는 것이 된다.

2. 한글 응용 생성기의 특징

한글 응용 생성기가 4세대 시스템 구축을 위해 제공하는 기능은 다음과 같다.

2.1 응용 작성의 간편성을 위한 기능

데이터베이스나 프로그래밍 언어에 대해 익숙하지 않은 초보자라도 데이터베이스 응용을 손쉽게 작성할 수 있도록 한글 응용 생성기가 지원하는 기능은 다음과 같다.

첫째, 데이터베이스의 데이터 사전 정보를 제공하는 기능을 제공한다. 양식상의 항목 정의시 이미 존재하는 데이터베이스의 데이터 사전 정보를 그대로 이용할 수 있어 별도로 항목을 정의할 필요가 없다[3].

둘째, 데이터 사전의 정보와 사용자의 항목 명세를 이용해 양식 위에 항목을 자동으로 배치하는 기능을 제공한다[3, 4, 5].

셋째, 테이블로부터 필요한 데이터를 검색하기 위한 양식 검색 조건을 자동적으로 작성하고 양식 상의 항목과 대응하는 사용자 변수도 시스템에 의해 자동적으로 지정되는 기능을 제공한다[3].

2.2 높은 수준의 응용을 작성하기 위한 기능

초보자를 위해 기존의 시스템이 제공하는 많은 시스템 디폴트(default)는 작성할 수 있는 응용의 한계를 가져왔다. 따라서 본 시스템에서는 프로그램 전문가의 요구를 충족시킬 수 있는 높은 수준의 응용을 손쉽게 작성할 수 있도록 다음과 같은 기능을 제공한다.

첫째, 데이터베이스의 데이터 정의를 변경하여 항목을 재정의할 수 있는 기능이다. 시스템이 제공하는 데이터 사전의 정보를 작성하고자 하는 응용에 맞게 변경하여

항목 길이와 항목명, 항목의 데이터 타입 등을 재정의할 수 있다[3].

둘째, 작성자는 시스템에 의해 자동적으로 구성된 양식 위에서 텍스트 에디팅을 통하여 원하는 형태로 양식의 배치를 변경할 수 있다[3, 4, 5, 8].

셋째, 시스템에 의해 작성된 양식 검색 조건을 작성자가 변경하고 새로운 조건을 추가로 삽입할 수 있는 기능을 제공하여 AND나 OR가 포함된 복잡한 검색 조건도 작성 가능하다.

2.3 신뢰성있는 응용을 생성할 수 있는 기능

시스템은 작성자에 의해 명세된 내용이 이전에 명세된 내용과 일관성을 유지하는지를 점검하며 데이터베이스의 데이터 사전에 정의된 내용에 위배되는지를 조사한다. 즉 응용의 수행시 에러를 검출하는 것이 아니라 응용 작성시 에러를 발견한다. 또한 범용 프로그래밍 언어와는 달리 응용의 작성이 완전히 끝나지 않은 상태에서도 부분적으로 응용을 수행시킬 수 있는 기능도 지원된다.

2.4 간편하고 효율적인 응용의 실행을 위한 기능

시스템이 응용의 전체 구조를 보여준 후 사용자가 응용을 구성하는 임의의 메뉴나 양식으로 직접 분기할 수 있도록 하는 기능을 제공하여 응용을 신속하게 수행할 수 있다

2.5 데이터베이스의 무결성을 유지하기 위한 기능

작성자는 항목을 통해 데이터베이스로 입력되는 값들에 대해 이들 값의 제약 조건을 구성할 수 있다. 제약 조건의 종류로는 입력될 값의 상한과 하한을 설정하는 범위 점검(Range Checking)과 입력 값의 가능 대상 목록을 나열하는 선택 목록(Selection List) 설정[5]의 두 가지가 제공된다.

2.6 간편한 데이터베이스 연산 수행을 위한 기능

질의어를 사용자가 작성할 필요없이 원하는 연산과 연산에 필요한 데이터 값만 명세하면 시스템에 의해 자동적으로 질의어가 생성되어 수행된다. 또한 데이터 값의 명세서에도 데이터 타입을 고려하지 않고 입력해도 시스템이 이를 적절하게 변환하는 기능이 제공된다.

2.7 한글 메뉴 지원 및 한글 데이터 처리 기능

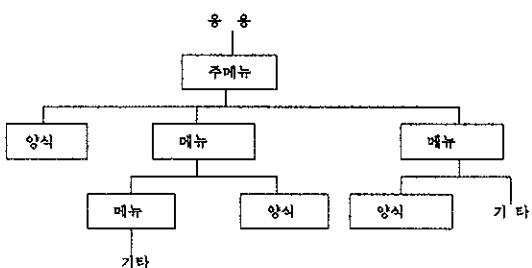
국내 실정에 맞도록 모든 메뉴 화면은 한글로 표현되며 데이터베이스에 테이블을 생성할 때 테이블명, 애트리뷰트 명, 데이터 값들도 모두 한글이 가능하다.

3. 응용의 자료 구조

작성되는 응용은 메뉴와 양식들의 집합으로 구성된 나무 형태의 구조가 되는데 이를 응용 트리(Application Tree)[11, 13, 14, 15]라고 한다. 응용 트리의 최상위 노드는 항상 메뉴 노드인데 이를 주메뉴 노드라고 한다. 즉 응용의 실행은 항상 주메뉴로부터 시작한다.

메뉴 노드는 임의 수의 자식 노드를 가질 수 있는데 자식 노드의 형태로는 메뉴 노드와 양식 노드가 모두 가능하다. 메뉴 노드에서 분기의 방향은 사용자의 선택에 의해 결정된다. 양식 노드에서는 실제 데이터베이스와의 모든 작업, 즉 데이터의 검색, 새로운 데이터의 입력, 데이터 값의 변경, 불필요한 데이터의 삭제 등의 작업이 일어난다.

응용 트리의 구조는 (그림 1)에 나타나 있다. (그림 1)에서는 주메뉴가 하나의 양식 노드와 두 개의 메뉴 노드를 가진 자식 노드로 갖는다. 메뉴 노드는 또 다른 메뉴 노드와 양식 노드를 자식 노드로 가져 응용은 단계의 메뉴 시스템으로 구성된다. 응용 트리는 레벨에 제약이 없으며 트리가 균형화될 필요도 없다.



(그림 1) 응용 트리의 구조

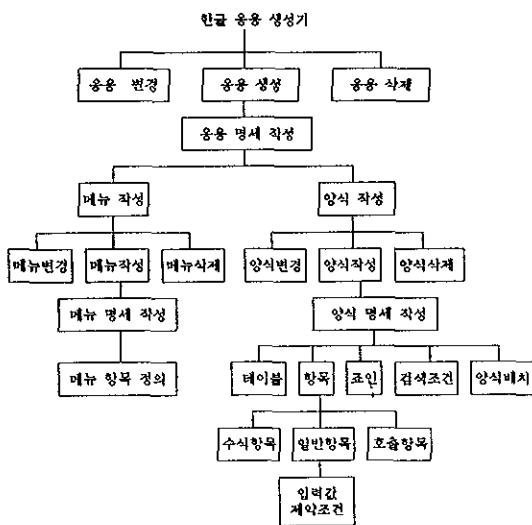
4. 한글 응용 생성기의 사용자 인터페이스

한글 응용 생성기의 사용자 인터페이스는 응용을 개발하기 위한 인터페이스와 작성된 응용을 실행하기 위

한 사용 인터페이스로 크게 구분할 수 있다. 이 장에서는 한글 응용 생성기를 이용하여 응용을 생성하고 실행하는 과정에 대해 살펴본다.

4.1 응용의 생성

응용을 생성하기 위해서는 응용 트리를 구성하는 메뉴 노드와 양식 노드를 작성해야 한다. 응용을 생성하기 위한 명세 과정을 그림으로 나타내면 (그림 2)와 같은데 이를 명세 트리(Specification Tree)[11, 13, 14, 15]라고 한다. 작성자는 명세 트리를 따라가면서 응용 작서에 필요한 사항을 명세함으로써 원하는 응용을 작성할 수 있다



4.1.1 메뉴의 작성

메뉴를 설계하기 위해서는 먼저 메뉴의 이름과 상단 제목을 명세해야 한다. 그리고 메뉴를 구성하는 각 항목에 대해 항목의 이름과 그 항목이 선택되었을 때 호출되는 자식 노드의 타입과 이름을 명세해야 한다. 호출되는 자식 노드의 타입으로는 메뉴와 양식이 있다. 메뉴의 역할은 범용 프로그램 언어의 조건부 분기에 해당하는 것으로 분기의 방향은 대화를 통해 사용자에 의해 결정된다.

4.1.2 양식의 지정

양식을 설계하기 위해서는 먼저 양식의 이름과 상단 제목을 명세해야 한다. 그리고 양식이 대상으로 하는 테

이블, 항목, 표인 조건, 검색 조건, 양식 배치를 세부적으로 작성해야 한다.

1) 테이블의 정의

양식이 대상으로 하고 있는 테이블의 이름과 테이블의 처리 단위를 명세해야 한다. 테이블의 이름이란 양식이 대상으로 하는 데이터베이스의 테이블 이름을 말한다. 테이블의 처리 단위는 양식에서 그 테이블을 어떤 단위로 처리하는가를 나타내는데 테이블 단위(Table Unit)와 투플 단위(Tuple Unit)[3]가 있다. 테이블 단위는 테이블의 여러 투플이 함께 양식에 나타나게 되는 것이고 투플 단위는 오직 한 투플의 내용만이 양식에 출력됨을 나타낸다.

2) 항목의 정의

항목을 분류하면 일반 항목, 계산 항목 등이 있다.

일반 항목은 데이터베이스 필드와 일대일로 대응하여 데이터베이스와의 입출력 작업이 일어난다. 일반 항목을 정의하기 위해서는 항목이 속해 있는 테이블명, 항목의 이름, 항목의 설명자, 항목의 타입, 항목의 길이, 항목의 사용 형식 등을 명세해야 한다. 항목의 사용 형식[3]에는 검색(Search) 형식, 입력(Entry) 형식, 출력(Output) 형식 등이 존재한다

시스템은 설계자가 테이블 정의 부분에서 명세한 테이블 이름을 이용하여 그 테이블에 관한 정보를 데이터베이스 내의 데이터 사전으로부터 추출하여 화면 위에 보여준다. 추출된 정보에는 테이블명, 항목의 이름, 항목의 타입, 항목의 길이가 포함되어 있다. 설계자는 그 화면 위에서 양식에 필요한 항목만을 선택하여 항목의 설명자와 항목의 사용 형식을 지정할 수 있고 이미 정의된 내용도 원하는 형태로 변경할 수 있다. 이러한 항목의 정의 방법은 기존 시스템에 비해 작업의 간편성과 원하는 대로 양식의 항목을 재정의할 수 있다는 장점을 가지고 있다.

일반 항목을 이용해 데이터베이스로의 입출력 작업을 하고자 할 때는 ‘입력값 조사’ 화면을 불러 사용할 수 있다. 기존의 시스템에서는 단지 입력되는 값이 데이터 사전에 정의된 내용과 다른 경우만 점검하는데 비하여 본 시스템에서는 좀 더 광범위한 설계자의 요구를 반영하여 데이터베이스의 무결성을 유지할 수 있도록 한다. 또한 입력값의 제약 조건을 정의하는 과정도 따로 프로

그래밍 언어를 이용하여 프로시저를 작성할 필요없이 변수의 개념이나 프로그래밍 언어의 문법을 모르는 사람이라도 누구나 시스템의 안내에 따라 쉽게 제약 조건을 정의할 수 있도록 한다.

입력값 조사로 범위 제약 조건과 선택 목록의 두 가지를 이용한다. 범위 제약 조건은 데이터가 만족되어야 할 범위를 명세하는 것으로 입력되는 테이터 값의 하한과 상한, 위배시 출력되는 에러 메시지를 명세한다. 선택 목록[5]은 데이터가 포함되는 목록(List)을 명세하는 것으로 목록의 내용은 사용자가 데이터 값을 입력하고자 할 때 화면에 출력되어 그 중의 하나를 사용자가 선택하게 한다.

수식 항목은 데이터베이스 필드를 이용한 수식을 정의하기 위해 사용된다. 수식에는 일반 산술 연산자와 모임 함수(Aggregate Function) 등이 사용될 수 있다.

3) 조인 항목의 정의

조인은 두개 이상의 테이블을 연결시키는 것으로 관계 데이터베이스 시스템에서는 필수적인 기능이다. 양식이 두개 이상의 테이블을 이용해 정의되는 경우는 테이블 간의 조인을 정의해야 한다. 조인을 정의하기 위해서 사용자는 조인에 사용되는 테이블명과 항목명을 명세해야 한다.

4) 검색 조건의 정의

양식 상에 필요한 데이터를 가져오기 위한 검색 조건을 정의한다. 본 시스템에서는 설계자에 의한 항목 정의와 조인 항목의 내용을 기반으로 하여 기본적인 검색 조건을 구성한 후 그 내용을 설계자가 원하는 형태로 변경하고 새로운 조건을 추가할 수 있는 것을 허용한다. 그러므로 AND와 OR가 섞인 복잡한 검색 조건도 손쉽게 작성할 수 있다.

5) 양식 배치

시스템이 자동적으로 상단 제목, 항목의 배치, 반복 항목의 배치 등을 결정하고 설계자는 작성된 양식 위에서 전화면 편집기를 통하여 양식 배치를 변경할 수 있다.

4.2 응용의 실행

사용자가 응용을 실행하고자 하면 시스템은 응용 사

전의 정보를 이용해 응용의 목록을 보여준다. 응용 목록 중에서 원하는 응용을 선택가하여 응용을 실행시킨다. 응용은 주메뉴로부터 시작되므로 주메뉴가 가장 먼저 실행된다. 사용자는 응용 트리를 순회하면서 또한 메뉴나 양식을 실행 도중 특정 키를 누름으로써 응용의 전체적인 구조를 볼 수 있으며 응용을 이루는 임의의 메뉴나 양식으로 분기가 가능하다.

4.2.1 메뉴의 실행

메뉴가 실행되면 화면에는 메뉴의 상단 제목과 함께 메뉴를 구성하는 항목들이 배열된다. 메뉴 항목 중 원하는 항목을 사용자가 선택하여 응용 트리의 분기 방향을 결정한다. 메뉴 수행이 끝나고 하위 메뉴가 수행되거나 양식이 수행되며, 상위 메뉴로 복귀할 수도 있다.

4.2.2 양식의 실행

양식 노드 위에서 데이터베이스와의 입출력 작업 및 검색 작업을 한다. 양식 화면은 연산 선택 원도우, 시스템 메시지 원도우, 양식 원도우의 세 개의 원도우로 구성되어 있다.

연산 선택 원도우는 양식 위에서 수행할 수 있는 연산 목록을 보여주며 시스템 메시지 원도우는 양식을 사용자가 손쉽게 사용할 수 있도록 시스템 정보를 전송하며 양식 원도우에는 작성자에 의해 선택된 데이터베이스 항목들이 배열된다. 연산 선택 화면에서 먼저 실행할 연산을 선택하면 시스템은 연산 실행에 필요한 항목이 무엇인가에 대한 정보를 시스템 메시지 화면에 보여준다. 그 내용을 보고 필요한 데이터를 해당 원도우에 채움으로써 원하는 연산을 수행시킬 수 있으며 처리된 결과는 양식 원도우에 출력된다.

사용자는 원하는 연산 종류의 선택과 연산 수행에 필요한 항목을 채우는 과정을 반복함으로써 계속적인 데이터베이스 연산을 실행할 수 있으며 양식 상에서 원하는 연산을 모두 끝낸 후에는 응용 수행을 완전히 완료할 수도 있고 상위 메뉴로 복귀하여 다른 메뉴로 분기하거나 또 다른 양식을 수행할 수 있다.

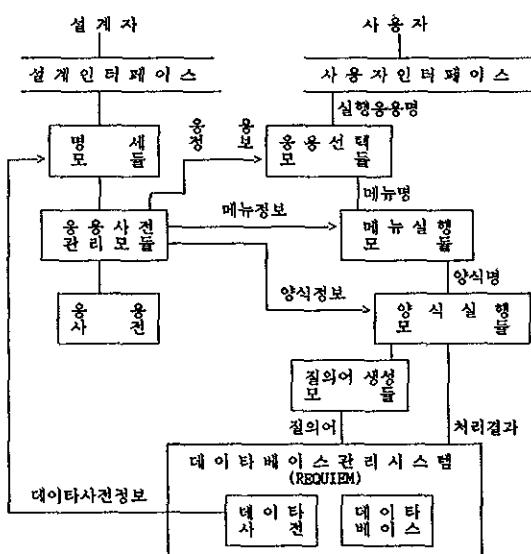
5. 한글 응용 생성기의 시스템 구성

이 장에서는 한글 응용 생성기의 전체적인 시스템 구성과 시스템의 각 구성 모듈에 대해 기술한다. 이에 대

한 자세한 기술은 [13]과 [14]에 잘 나타나 있다.

5.1 시스템 구성

한글 응용 생성기는 설계 인터페이스, 사용자 인터페이스 등의 인터페이스 부분과 명세 모듈, 응용 사전 관리 모듈, 응용 선택 모듈, 메뉴 실행 모듈, 양식 실행 모듈, 질의어 생성 모듈 등의 프로그램 모듈 부분, 응용 사전, 관계 데이터베이스 관리 시스템인 REQUIEM[9] 등으로 구성되어 있다. 시스템 구성도는 (그림 3)과 같다.



(그림 3) 한글 응용 생성기의 시스템 구성도

시스템의 작동 과정을 설명하면 다음과 같다. 먼저 설계자는 설계 인터페이스를 통해 명세 모듈과 대화식으로 통화하면서 응용 사전에 필요한 세부 사항을 정의한다. 정의된 내용은 응용 사전 관리 모듈을 통해 응용 사전에 저장된다.

사용자는 사용자 인터페이스를 통해 실행할 응용 이름을 선택한다. 응용 선택 모듈은 선택된 응용에 대한 정보를 응용 사전의 헤더 파일로부터 추출하여 주메뉴 이름을 메뉴 실행 모듈에 전달하고 제어권을 메뉴 실행 모듈에 전송한다.

메뉴 이름을 전달받은 메뉴 실행 모듈은 해당되는 메뉴의 정보를 이용해 메뉴를 실행시키면서 사용자로 하여금 원하는 메뉴 항목을 선택하게 한다. 선택된 메뉴 항목이 '양식'일 때까지 메뉴 실행 모듈은 작동을 계속

한다. 만약 '양식'의 실행이 사용자에 의해 선택되어지면 메뉴 실행 모듈은 실행될 양식의 이름을 양식 실행 모듈에 전달하고 제어권을 양식 실행 모듈에 전송한다.

양식 이름을 전달받은 양식 실행 모듈은 해당하는 양식의 정보를 이용해 양식을 실행시킨다. 사용자는 양식을 이용해 원하는 연산을 명령하고 그 명령은 질의어 생성 모듈을 통해 질의어로 변환되어 데이터베이스 시스템에 전달된다. 전달된 질의어는 데이터베이스 관리 시스템에서 실행되고 그 결과는 다시 양식 실행 모듈에 전송되어 양식 위에 출력된다. 사용자가 양식에서의 작업을 끝내면 시스템의 제어권은 다시 메뉴 실행 모듈로 전송되어 양식이 실행되기 바로 전의 메뉴로 되돌아간다.

5.2 모듈별 기능

5.2.1 설계 인터페이스

설계자가 응용을 용이하게 작성할 수 있도록 시스템이 제공하는 원도우 형태의 인터페이스이다. 본 시스템에서는 스택화된 원도우(stacked window)를 인터페이스의 기본 형태로 사용하는데 이 방식은 원도우 간의 중첩을 허용하지 않는 대신 처리 속도가 빠르다는 장점이 있다.

5.2.2 명세 모듈

명세 트리(Specification Tree)를 따라 가면서 응용의 작성에 필요한 내용을 설계자로부터 전달받아 응용 사전 관리 모듈에 전달한다. 명세 모듈을 이용해 설계자는 응용을 구성하는 메뉴와 양식을 작성한다. 시스템은 데이터베이스의 데이터 사전 정보를 최대한으로 이용하고 시스템에 의한 자동 생성 기능을 지원함으로써 복잡하고 어려운 응용이라도 쉽게 작성할 수 있도록 한다.

5.2.3 응용 사전 관리 모듈

명세 모듈로부터 응용에 관한 정보를 전달받아 응용 사전에 저장한다. 저장된 내용은 응용의 실행에 사용된다.

5.2.4 응용 사전

응용에 관한 정보를 실제 저장하는 부분으로 하나의 헤더 파일과 다수의 응용 파일로 구성되어 있다. 헤더 파일에는 각각의 응용에 대해 응용의 이름과 응용의 주

메뉴 이름 등 응용에 대한 대표 정보가 저장되어 있다. 응용 파일에는 응용을 구성하는 메뉴와 양식에 대한 세부 정보가 저장된다.

5.2.5 사용자 인터페이스

사용자가 응용을 실행시킬 수 있는 윈도우 형태의 메뉴 인터페이스를 제공한다. 설계 인터페이스와 마찬가지로 스펙화된 윈도우를 사용한다.

5.2.6 응용 선택 모듈

응용 사전에 저장되어 있는 헤더 파일의 정보를 이용해 응용의 목록을 보여주고 사용자로 하여금 실행할 응용을 선택하게 한다. 선택된 응용에 대한 헤더 파일의 정보를 이용해 해당되는 데이터베이스를 열고(open)하고 응용 트리를 구성하며 가장 먼저 수행될 메뉴 이름을 메뉴 실행 모듈에 전송한다.

5.2.7 메뉴 실행 모듈

응용 선택 모듈로부터 전송되어 온 메뉴의 이름을 이용해 응용 사전 관리 모듈로부터 해당되는 메뉴 정보를 획득한다. 다수의 메뉴 항목 중 하나를 사용자가 선택하게 함으로써 다음에 수행할 대상을 결정한다. 선택된 대상이 메뉴이면 다시 그 메뉴에 대한 정보를 가져와서 작업을 계속한다. 그러나 선택된 대상이 양식이면 양식 실행 모듈에 양식의 이름을 전달하고 계어권을 양식 실행 모듈로 전송한다.

5.2.8 양식 실행 모듈

메뉴 실행 모듈로부터 전송되어 온 양식의 이름을 이용해 응용 사전 관리 모듈로부터 해당되는 양식 정보를 획득하여 양식을 실행시킨다. 사용자가 선택한 연산을 질의어 생성 모듈에 전송하고 데이터베이스 관리 시스템에서 실행된 연산 결과를 다시 양식에 보여주는 작업을 한다.

5.2.9 질의어 생성 모듈

사용자가 선택한 연산에 대해 그것을 데이터베이스 관리 시스템이 수행할 수 있는 질의어로 변환시켜 준다. 또한 사용자에 의한 입력 데이터 값은 데이터 정의에 맞게 변환하는 작업도 한다.

5.2.10 관계 데이터베이스 관리 시스템[9]

본 시스템은 하부 구조로 관계 데이터베이스 시스템인 REQUIEM을 사용한다. REQUIEM은 SQL과 유사한 질의어를 지원한다. 그러나 정신 질의어와 삽입 질의어인 경우는 본 한글 응용 생성기에 적합하지 않으므로 새로이 질의어 구문을 정의하였다. 변경된 질의어에 대한 내용은 참고문헌[13, 14]에 자세히 나타나 있다.

6. 양식 연산 처리

6.1 양식 연산 수행

데이터베이스 연산을 처리하기 위해 시스템이 수행하는 작업은 이전 작업(Before Action), 이후 작업(After Action)으로 크게 분류할 수 있다.

이전 작업은 양식을 화면에 출력하기 전에 데이터베이스 연산을 수행하기 위한 준비 작업을 하는 것으로서 양식이 대상으로 하는 테이블에 대한 데이터 사전 정보를 추출한다.

이후 작업은 양식을 화면에 출력한 후 사용자의 데이터베이스 처리 요구를 분석하고 이를 처리하는 것을 말한다. 사용자가 입력한 데이터가 입력값 제약 조건을 만족하는지를 점검하며 위배시에는 오류 메시지를 전송한다. 입력 데이터가 입력값 제약 조건을 만족할 때는 이를 데이터 사전의 데이터 정의에 맞게 변환한 후 데이터베이스 질의어를 생성한다. 생성된 질의어는 관계 데이터베이스 시스템에 의해 실행되고 실행 결과는 화면에 출력된다. 만약 검색된 결과 데이터를 이용한 계산이 정의되어 있는 경우는 이를 수행하고 계산 결과를 계산 항목 위에 출력한다.

6.2 양식 연산 처리 과정

양식 위에서 수행할 수 있는 데이터베이스 연산으로는 검색, 변경, 삽입, 삭제 등이 있다. 사용자는 원하는 연산의 종류를 선택하고 연산에 필요한 데이터를 채움으로써 데이터베이스 연산을 수행할 수 있는데 각각의 연산은 하나의 트랜잭션(TRANSACTION)으로 볼 수 있다. 즉 시스템은 데이터베이스에 어떤 연산을 수행하기 전에 그 연산이 실패할 경우를 미리 점검하여 배제하므로 데이터베이스 연산이 수행 도중 취소되는 경우는 없다.

데이터베이스 연산 수행 과정을 설명하기 위해 (그림 4)의 학생 정보 양식을 이용한다. 학생 정보 양식은 학생(STUDENT) 테이블과 과목(COURSE) 테이블, 수강(ENROLL) 테이블을 이용하여 학생에 대한 신상 정보와 성적 정보를 검색하기 위해 사용된다. 밑줄로 표현된 부분이 사용자에 의해 입력된 값을 나타낸다.

(그림 4) 양식의 구성 예: 학생 정보 양식

6.2.1 검색 연산 수행

양식을 화면에 출력하기 전에 이전 작업으로서 학생 테이블, 과목 테이블, 수강 테이블에 대한 데이터 사진 정보를 추출한다. 이전 작업이 끝난 후 양식이 화면에 출력된다. 출력된 화면 위에서 입력 연산자와 입력 데이터 값을 명세함으로써 검색 연산을 수행시킬 수 있다. 학생 정보 양식은 세 개의 테이블로 구성되어 있으므로 이를 간의 조인 작업이 필요하다. 학번이 "s003"인 학생의 신상 정보와 성적을 검색하는 연산을 질의어[9]로 표현하면 다음과 같다

```

SELECT : 학번, 이름, 학과, 나이, 주소, 전화번호,
        과목명, 학점단위수, 평점
FROM   : 학생, 과목, 수강
WHERE  : 학생.학번=수강.학번 and
        과목.과목번호=수강.과목번호 and
        학번="s003"
    
```

위의 질의어의 처리 결과가 (그림 4)에 출력되어 있다. 만약 처리 결과가 한 화면을 초과할 때는 아래 방향화살표나 윗 방향화살표를 이용하여 화면을 스크롤(SCROLL)시킬 수 있으며 연산 선택 원도우에서 '2. 다음 화면'이나 '3. 이전 화면'을 선택함으로써 화면을 전환할 수도 있다. '다음 화면'은 테이블 단위일 때는 다음

페이지를 의미하고 투플 단위일 때는 다음 레코드를 의미한다.

위와 같은 검색 질의어를 생성하기 위해서는 질의가 대상으로 하는 테이블명과 항목명, 검색 조건을 구성해야 한다. 테이블명은 검색할 항목을 포함하고 있는 테이블을 나타내며 항목명은 양식의 작성 시 출력 항목으로 지정된 항목을 말한다. 검색 조건의 변수와 연산자는 사용자의 입력 값과 입력 연산자로 대체된다. 입력 값이 없는 조건은 검색 조건 구성에서 삭제되어 입력 연산자가 없으면 기존의 연산자가 변경없이 그대로 사용된다. (그림 5)에 검색 질의어를 생성하는 알고리즘이 제시된다.

Algorithm SEARCH_query

Begin

1. SELECT문 구성

- 출력 항목으로 지정된 모든 데이터베이스 항목명을 출력한다.

2. FROM문 구성

- 출력 항목이 포함된 데이터베이스 테이블명을 출력한다.

3. WHERE문 구성

a) 조인 조건 구성

- 만약 대상 테이블이 두 개 이상이면, '테이블.항목=테이블.항목'과 같은 조인 조건을 구성한다.

b) 검색 조건 구성

- (테이블 항목 op & Var)와 같은 형태를 갖고 있는 검색 조건에 대해,

- & Var를 사용자의 입력 값으로 치환하고,
- op를 사용자의 입력 연산자로 치환한다.
- 만약 입력 연산자가 없으면 op는 변경되지 않는다.
- 입력 값이 없으면 그 검색 조건은 검색 조건 구성에서 삭제된다.

EndofAlgorithm SEARCH_query

(그림 5) 검색 질의어 생성 알고리즘

6.2.2 변경 연산 수행

먼저 필요한 데이터를 검색 연산을 통해 찾은 후 개신 연산을 선택함으로써 개신 작업을 할 수 있다. 학생

정보 양식 위에서 ‘박상현’인 학생의 나이를 26살로 변경하고자 할 때는, 이름이 ‘박상현’인 학생의 정보를 검색 연산을 통해 찾은 후 출력된 결과 화면 위에서 ‘나이’ 항목 위에 놓고 나이를 ‘26’으로 변경하면 된다.

이 때 시스템에 의해 생성되는 질의어는 다음과 같다.

```
UPDATE 나이=26
```

```
FROM 학생
```

```
WHERE 이름="박상현";
```

(그림 6)에 간단 질의어를 생성하는 알고리즘이 제시된다.

Algorithm UPDATE_query

Begin

UPDATE하려고 하는 각각의 테이블에 대해서

1. UPDATE문 구성

- 각각의 갱신된 항목에 대해

- ‘테이블.항목=갱신된-값’의 할당문을 추가한다.

2. FROM문 구성

- 갱신되는 테이블명을 출력한다.

3. WHERE문 구성

- SEARCH_query 작성시에 생성된 WHERE문을 그대로 사용한다.

EndofAlgorithm UPDATE_query

(그림 6) 간단 질의어 생성 알고리즘

6.2.3 삽입 연산 수행

삽입 연산을 선택하고 데이터 값을 입력함으로써 새로운 투플을 삽입시킬 수 있다. 학생 정보 양식 위에서 학번이 ‘s100’이고 이름이 ‘홍길동’인 학생을 삽입하고자 할 때 생성되는 질의어는 다음과 같다.

```
INSERT (학번="s100", 이름="홍길동", 학과="컴  
퓨터공학과", 나이=23)
```

```
FROM 학생;
```

(그림 7)에 삽입 질의어 생성 알고리즘이 제시된다.

Algorithm INSERT_query

Begin

INPUT하려고 하는 각각의 테이블에 대해

1. INSERT문 구성

- 새로 입력된 값을 갖고 있는 각 항목에 대해, ‘테이블.항목=입력값’과 같은 할당문을 추가한다

2. FROM문 구성

- 입력하려고 하는 테이블명을 출력한다.

EndofAlgorithm INSERT_query

(그림 7) 삽입 질의어 생성 알고리즘

6.2.4 삭제 연산 수행

먼저 삭제할 데이터를 검색 연산을 통해 찾은 후 삭제하고자 하는 코드 위에서 커서를 옮려놓고 삭제 연산을 선택하면 된다. 학번이 ‘s100’인 학생을 검색한 후 삭제하고자 할 때 생성되는 질의어는 다음과 같다.

```
DELETE STUDENT
```

```
WHERE 학번="s100";
```

(그림 8)에 삭제 질의어 생성 알고리즘이 제시된다.

Algorithm DELETE_query

Begin

DELETE하려고 하는 각각의 테이블에 대해

1. DELETE FROM문 구성

- 삭제할 테이블명을 출력한다.

2. WHERE문 구성

- SEARCH_query시에 작성한 WHERE문을 그대로 이용한다.

EndofAlgorithm DELETE_query

7. 결 론

컴퓨터를 이용한 많은 응용이 개발되고 있는데 그중의 하나가 데이터베이스를 이용한 대화식 정보 처리이다[6]. 그러나 대화식 데이터베이스 응용의 개발을 위해서는 프로그래밍 언어와 데이터베이스를 동시에 잘 알고 있어야 하는 어려움이 존재한다[1, 2, 12]. 본 논문에서는 초보자라도 응용을 손쉽게 작성할 수 있도록 하는 한글 응용 생성기를 설계하고 구현하였다[11, 13, 14, 15].

한글 응용 생성기는 작성자가 명세한 내용과 데이터

베이스의 데이터 사전 정보를 이용하여 응용을 자동적으로 생성할 뿐만 아니라 자동적인 양식 배치 기능, 양식 검색 조건의 자동 생성 기능 등의 많은 시스템 디플트를 제공한다. 또한 높은 수준의 응용을 개발하기 위한 항목의 재정의 기능, 양식 검색 조건의 변경, 양식 배치 변경 기능 등의 많은 툴(tool)을 제공하며 생성된 응용의 신속한 실행을 위해 직접적인 실행 대상 선택 기능을 제공한다. 양식상에서 데이터베이스 연산을 손쉽게 하기 위해 질의어의 자동 생성 기능이 제공되고 데이터베이스의 무결성을 유지하기 위한 입력값 제약 조건 정의 기능이 제공된다. 또한 국내 실정에 적합하도록 한글 데이터 처리 및 한글 메뉴 기능이 제공된다. 따라서 본 응용 생성기를 이용함으로써 신뢰성이 있으면서도 수준이 높은 고급 응용을 초보자라도 손쉽게 작성할 수 있으며 생성된 응용은 효율적이면서도 신속하게 처리된다.

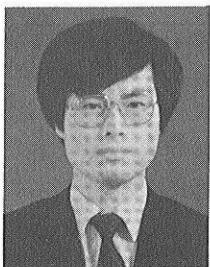
참 고 문 헌

1. Ellis, H., Alfons, K., and Balaji, N., "A Survey of Application Generators," IEEE Software, 1985.
2. Martin, J., *Fourth-Generation Language*, Prentice-Hall, 1985.
3. Yao, S. B., Hevner, A. R., SHI, Z., and LUO, D., "FORMANGER: An Office Forms Management System," ACM Trans. on Office Information Systems, Vol. 2, No. 3, July 1984.
4. *dBASE IV User's Guide*, Ashton-Tate, 1989.
5. Laurent, R., *4th Dimension User's Guide*, 1989.
6. Lawrence, A. R., and Knot, A. S., "A Form Application Development System," ACM Proc. of SIMOD, 1982.
7. Lawrence, A. R., "Fill-in-the-Form Programming," Proceedings of VLDB, 1985.
8. IBM, *Database Manager User's Guide*, 1990.
9. Papazoglou, M., and Valder, W., *Relational Database Management, a Systems Programming Approach*, Prentice-Hall, 1989.
10. Date, C. J., *An Introduction to Database Systems*, 4th ed., Vol. I, Addison-Wesely, 1986.
11. 서울대학교, "4세대 한글 관계 데이터베이스 시스템 개발에 관한 연구", 90 국책 과제 제2차 최종연구 보고서, 1990.
12. 경영과 컴퓨터, "4세대 언어, 그 현주소를 파악한다", 1989.
13. 박상현, "관계 데이터베이스 시스템에서의 응용 생성기에 관한 연구", 서울대학교 컴퓨터공학과 석사학위논문, 1991. 2.
14. 박상현, 용환승, 이석호, "관계 데이터베이스 시스템에서의 한글 응용 생성기에 관한 연구", 서울대학교 컴퓨터공학과 데이터베이스 연구실 Technical Report, DBTR-11-1, 1991. 1.
15. Sanghyun Park, Sukho Lee, "Hangul Application Generator for Relational Database Systems," Proceedings of NEWDB, 1991, Daejeon, Korea, pp. 85-108.



박 상 현

1989년 2월 서울대학교 공과대학 컴퓨터공학과 졸업
 1991년 2월 서울대학교 공과대학원 컴퓨터공학과 졸업(석사)
 1991년 10월 ~ 현재 대우통신 종합연구소 소프트웨어 연구
 실무 근무
 관심분야 : 데이터베이스 시스템, 4세대 언어



용 환 승

1983년 2월 서울대학교 컴퓨터공학과 졸업
1985년 2월 서울대학교 대학원 컴퓨터공학과 공학석사
1985년 ~ 1989년 한국전자통신연구소 연구원
1991년 2월 서울대학교 대학원 컴퓨터공학과 박사과정 수료
1991년 ~ 현재 서울대학교 중앙교육연구전산원 연구조교
관심분야 : Nested Relational Database, 4th Generation Language, Object-Oriented Database



이 석 호

1964년 연세대학교 정치외교학과 졸업
1964년에서 1969년까지 해병장교로 근무
1975년과 1979년에 미국 텍사스대학교 전신학 석사와 박사위 취득

1979년부터 1982년까지 한국과학원 전산학과 조교수
1982~1986년 한국정보과학회 논문편집위원장
1986~1988년 한국정보과학회 부회장
1988~1989년 IBM Watson 연구소 객원교수
1988~1990년 데이터베이스연구회 운영위원장
1989~현재 서울대학교 중앙교육연구전산원 원장
1982년부터 현재까지 서울대학교 컴퓨터공학과 교수로 재직중이며 데이터베이스, 파일처리, 자료구조 등을 강의.