

WebSiteGen : 웹 기반 데이터베이스 응용의 자동 생성 도구*

°유은자, 음두현

덕성여자대학교 전산학과

¹bitty@namhae.duksung.ac.kr, dheum@namhae.duksung.ac.kr

WebSiteGen : a Tool for Automatic Generation of Web-based Database Applications

°Eunja Yoo, Doohun Eum

Duksung Women's Univ.

요약

오늘날 우리는 HTML 폼을 통해 지역 데이터베이스뿐 아니라 원격 데이터베이스도 쉽게 접근할 수 있다. HTML 폼으로 데이터베이스 응용을 구현하는 것이 쉽기는 하나 아직 HTML 폼과 CGI 프로그램은 코딩되어야만 한다. 본 논문에서는, 응용 데이터베이스 구축을 위한 SQL 문장들, 사용자 인터페이스로 사용되는 폼들, 그리고 그 폼들을 통해 이루어진 질의를 처리하는 자바 서블릿을 자동으로 생성하는 소프트웨어 도구를 소개한다. 생성될 테이블들과 폼들은 응용을 위한 클래스 다이어그램으로부터 결정된다. 따라서 WebSiteGen으로 명명한 우리의 소프트웨어 도구는 웹 기반 데이터베이스 응용의 구현을 자동화 한다.

1. 서론

전자상거래(Electronic Commerce)는 인터넷상에서 이루어지는 활발한 응용 분야이다. 전자상거래 사이트들은 매년 큰 폭으로 증가하여, 책 구입에서 주식 투자까지의 다양한 형태로 우리들의 일상 생활을 변화시키고 있다. 이러한 전자상거래 응용의 주된 요소는 웹 기반의 데이터베이스 응용이다.

본 논문에서는 웹 기반 데이터베이스 응용의 구현을 자동화하는 소프트웨어 도구, WebSiteGen을 소개한다. WebSiteGen은 응용 데이터베이스 구축을 위한 SQL 문장들, 사용자 인터페이스로 사용되는 폼들, 그 폼들을 통해 질의된 데이터를 검색하는 자바 서블릿을 자동 생성한다. 생성될 SQL 문장들과 폼들은 데이터베이스 응용을 위한 클래스 다이어그램에 의해 결정된다.

UML(Unified Modeling Language)의 클래스 다이어그램은 응용에 필요한 클래스들과 그들간의 관계 타입들을 직관적으로 나타낸다[1]. 클래스 다이어그램으로부터, 필요한 데이터베이스 구축을 위한 SQL 문장들을 생성해내는 규칙들은 이미 잘 알려져 있고 Oracle Design 2000과 같은 소프트웨어들은 이러한 작업을 수행하는 도구들이다[2].

데이터베이스에 저장된 데이터는 주로 폼을 통해 검색되고 갱신된다. 폼은 사용하기 쉬운 인터페이스를 제공하므로 널리 사용되고 있다. 예를 들어, Microsoft Access 2000이 제공하는 폼은 네트워크상에 분산되어 있는 데이터베이스도 접근할 수 있다[3]. 그러나 클래스 다이어그램으로부터 폼을 자동으로 생성하지는 못한다. 본 논문에서 제안한 방법은 클래스 다이어그램

으로부터 create table SQL 문장들뿐 아니라 필요한 폼들과 자바 서블릿을 자동으로 생성할 수 있다.

각 폼은 하나의 주 영역(main area)과 내포된 테이블(nested table)들로 구성된다. 주 영역에는 닷 엔티티(anchor entity)라 불리는 현재 관심의 대상인 엔티티에 대한 정보와 닷 엔티티와 일대일 또는 다대일 관계 타입으로 연관된 엔티티들에 대한 정보를 출력한다. 내포된 테이블에는 닷 엔티티와 일대다 또는 다대다 관계 타입으로 연관된 엔티티들에 대한 정보를 출력한다. 폼은 여러 개의 내포된 테이블들을 포함 할 수 있으나 내포는 한 단계만 허용한다. 즉, 내포된 테이블이 다른 테이블을 내포할 수 없다.

2절에서 WebSiteGen의 개요를 설명하고 3절에서는 WebSiteGen의 대략적인 구현을 설명한다. 마지막 절에서는 본 논문의 결론과 앞으로의 연구 방향을 정리한다.

2. WebSiteGen의 개요

본 절에서는 웹 기반 데이터베이스 응용을 자동으로 생성하는 도구인 WebSiteGen의 개요를 소개한다.

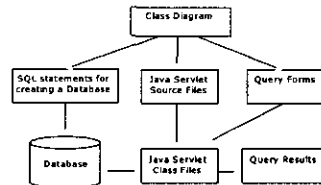


그림 1. WebSiteGen의 구조

* 본 논문은 KISTEP 2001년도 여자 대학교 연구기반 확충 사업 연구비 지원으로 수행되었음. (과제번호:00-N6-05-01-A-03)

그림 1은 WebSiteGen의 구조이다. WebSiteGen은 데이터베이스 응용을 위한 클래스 다이어그램으로부터 응용 프로그램에서 필요한 데이터를 저장하는 관계형 테이블, 사용자 인터페이스로 사용될 HTML 폼, HTML 폼을 통한 질의를 처리할 자바 서블릿을 자동 생성한다.

WebSiteGen의 기능을 설명하기 위해 본 논문에서는 간단한 전자상거래 응용(e-order company)을 예로 들어 그 클래스 다이어그램을 먼저 보이고 이로부터 생성될 SQL 문장들, 폼들, 자바 서블릿을 보인다.

2.1 클래스 다이어그램

클래스 다이어그램은 클래스들과 그들간의 관계를 도식적으로 보여준다. 그림 2는 e-order company 응용의 클래스 다이어그램이다.

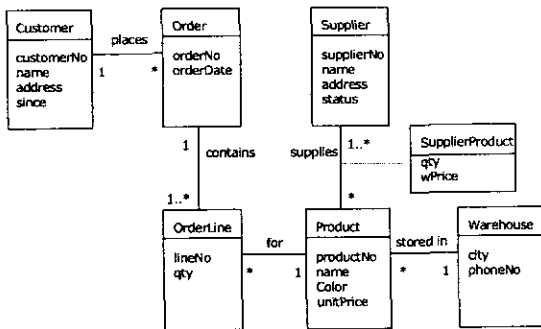


그림 2. e-order company 응용의 클래스 다이어그램

2.2 무결성 제약조건을 고려한 테이블의 생성

클래스 다이어그램에 표현된 정보를 이용해 응용 데이터베이스를 위해 필요한 SQL 문장들을 생성할 수 있다. WebSiteGen은 클래스 다이어그램 내의 각 엔티티 타입에 대해 테이블을 생성한다. 이 때, 두 엔티티 타입 A와 B간의 관계는 다음과 같이 처리된다. 관계가 일대일 또는 일대다일 경우, 엔티티 타입 B에 대한 테이블은 엔티티 타입 A에 대한 테이블의 기본키를 지정하는 외래키를 포함하고, 관계 타입에 대한 테이블은 필요하지 않다. 관계가 다대다일 경우는 관계 타입을 테이블 AB로 구현하고 테이블 AB는 엔티티 타입 A, B에 대한 테이블들의 기본키들을 지정하는 두 개의 외래키를 포함한다.

매치되지 않는 외래키 값을 방지하기 위해 restricted, cascade, set null의 세 가지 외래키 규칙이 사용된다. 참조 무결성 규칙은 테이블의 레코드를 갱신·삭제 연산 후에 매치되지 않는 외래키 값이 발생하는 것을 방지할 수 있다. 무결성 제약조건은 서로 다른 테이블에 존재하는 열들간의 합법적인 관계를 명시하며 관계 타입의 카디널리티와 응용의 의미로부터 유도해 낼 수 있다.

e-order company 응용에서, 엔티티 타입 Customer는 엔티티 타입 Order와 일대다 관계 타입으로 연관된다. WebSiteGen에 의해 생성되는 다음의 SQL 문장은 엔티티 타입 Order에 대한 테이블 Order를 생성한다.

```

Create table Order (
  orderNo    char(20),
  orderDate  char(20),
  customerNo char(20),
  primary   key(orderNo, customerNo),
  foreign key customerNo references Customer(customerNo)
  on delete cascade
  on update cascade
);
    
```

2.3 폼의 생성

WebSiteGen이 생성하는 폼은 그림 3과 같이 구성된다. 주 영역(main area)은 현재 관심 있는 엔티티(및 엔티티, anchor entity)에 대한 정보와 닷 엔티티와 직·간접적으로 일대일 또는 다대일 관계 타입으로 연관되는 엔티티들에 대한 정보를 보여준다. 내포된 테이블(nested table)은 닷 엔티티와 일대다 또는 다대다의 관계 타입으로 연관되는 엔티티들을 보여준다.

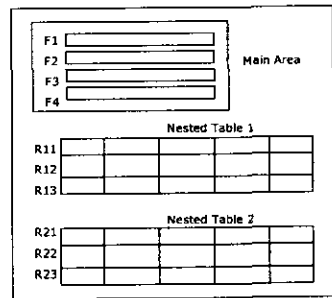


그림 3. 폼의 구조

그림 4는 e-order company 예제의 주문 폼이다. 주문 폼의 닷 엔티티는 한 개의 주문이다. 클래스 다이어그램에서, 엔티티 타입 Order는 엔티티 타입 Customer와 다대일 관계 타입으로 연관되므로 주문과 고객에 관한 정보는 주 영역에 디스플레이 된다. 각 주문은 한 명의 고객만이 연관된다. OrderLine은 Order와 일대다 관계 타입으로 연관되므로 여러 개의 주문 항목들이 각 주문에 연관된다. 따라서 주문 항목들은 내포된 테이블에 디스플레이된다. 또한, 하나의 주문 항목은 제품(Product)과 다대일 관계 타입으로 연관되므로 각 주문 항목에 의해 주문된 제품 정보도 디스플레이 될 수 있다. 마지막으로, 제품과 창고(Warehouse)는 다대일 관계 타입으로 연관되므로 창고에 대한 정보도 제품 정보와 함께 디스플레이 될 수 있다.

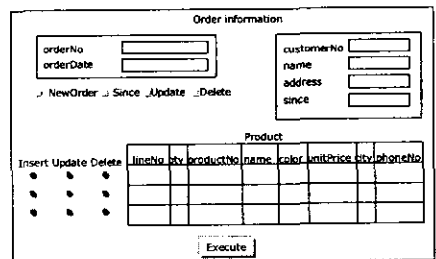


그림 4. e-order company의 주문 폼

그림 5는 e-order company 데이터베이스 응용의 클래스 다이어그램으로부터 생성될 폼들이 커버하는 엔티티 그룹들을 보인 것이다. 생성된 폼들로 응용의 모든 엔티티들과 관계들을 조작

할(insert, select, update, delete) 수 있으므로 생성된 폼들은 될 수 있다. 연결하고 모든 엔티티들을 커버 할 수 있다.

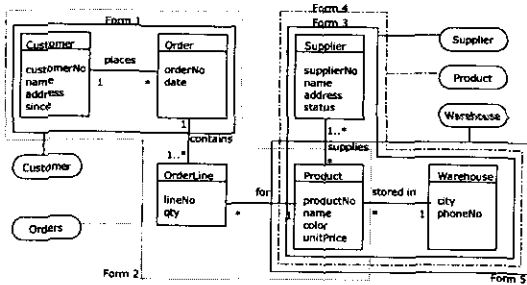


그림 5. 생성될 폼들이 커버할 엔티티 그룹들

2.4 자바 서블릿의 생성

자바 서블릿은 CGI 프로그램과 유사하며 웹 페이지를 동적으로 생성할 수 있다. 그러나 자바 서블릿은 CGI 프로그램과는 달리 서버 측의 확장이므로 서버의 일부분으로 수행된다. 따라서 자바 서블릿은 CGI 프로그램보다 훨씬 효율적이다. WebSiteGen이 생성하는 자바 서블릿은 Java Database Connectivity(JDBC)를 통해 데이터베이스와 연결된다. 즉, 서블릿은 HTML 폼으로부터 데이터를 받아 데이터베이스를 접근하며 HTML 폼들을 동적으로 생성한다.

3. WebSiteGen의 구현

WebSiteGen의 프로토타입은 Java로 구현되었다. WebSiteGen은 데이터베이스 구축을 위한 SQL 문장들, HTML 폼들, 그리고 자바 서블릿을 생성한다. 자바 서블릿은 Java Web Server(JWS)에 의해 실행되며 JDBC를 통해 Oracle 데이터베이스와 연동하여 HTML 폼들을 동적으로 생성한다.

WebSiteGen에 의해 생성된 e-order company 응용의 클래스 다이어그램은 그림 6과 같다. 클래스 다이어그램으로부터 생성된 SQL 문장들은 Customer, Order, OrderLine, Product, Supplier, SupplierProduct, 그리고 Warehouse의 테이블들을 생성한다. 2.2절에서 설명한 폼 생성 방법에 따라, 각 폼이 커버할 엔티티 그룹들도 표시된다. Customer 폼의 닷 엔티티 타입인 Customer는 적색 이중 사각형에 의해 둘러싸이고, Customer를 확장하여 얻을 수 있는 엔티티 타입인 Order는 적색 사각형에 의해 둘러싸인다. Order 폼의 닷 엔티티 타입인 Order이며 회색 이중 사각형에 의해 둘러싸이고, 연관된 엔티티 타입들인 Customer, OrderLine, Product, 그리고 Warehouse는 모두 회색 사각형에 의해 둘러싸인다. Product 폼, Supplier 폼, Warehouse 폼을 위한 엔티티 그룹들도 다른 색으로 비슷하게 표시된다. 생성될 폼들 중, Customer 폼으로 가능한 연산들은 디스플레이된 다이얼로그 박스에 명시된다. insert, select, update, delete 연산들이 Customer 테이블에 허용되고, insert, update, delete 연산들이 Order 테이블에 허용된다.

클래스 다이어그램으로부터 create table SQL 문장들이 생성될 때, 외래키에 대한 규칙 또한 다이얼로그 박스에 의해 명시

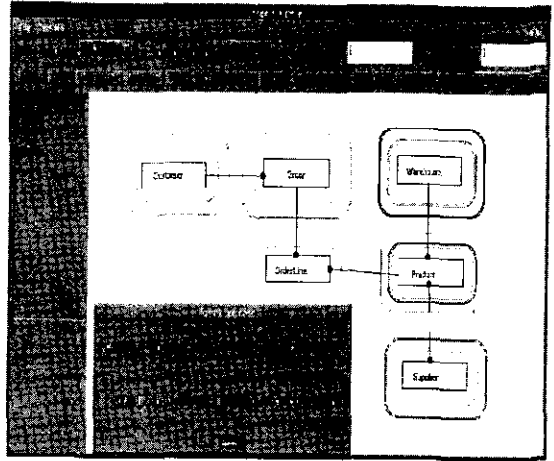


그림 6. WebSiteGen에 의해 생성된 클래스 다이어그램

4. 결론 및 향후 연구

클래스 다이어그램으로부터 웹 기반의 응용을 자동으로 생성하는 방법을 제안하고 이 작업을 수행하는 프로토타입, WebSiteGen을 구현하였다. WebSiteGen은 응용 데이터베이스 구축을 위한 SQL 문장들을 생성하고 사용자 인터페이스로 사용되는 폼들을 생성하며 데이터베이스와 폼들을 연결하는 자바 서블릿을 생성한다. 따라서 웹 기반 데이터베이스 응용의 구현을 자동화한다.

WebSiteGen은 웹 기반의 데이터베이스 응용을 자동 생성할 수 있는 소프트웨어 도구의 근간이 될 수 있다. 향후, ASP, PHP, 그리고 Cold Fusion과 같은 서버 측의 컴퓨팅 환경을 위해 동적으로 HTML 페이지를 생성할 수 있도록 할 것이다.

5. 참고문헌

- [1] Grady Booch, James Rumbaugh, and Lvar Jacobson, The Unified Modeling Language User Guide, Addison-Wesley, 1998.
- [2] Michael W. Stowe, Oracle Developer 2000 Handbook, 2nd Ed., PH, 1999.
- [3] Ken Getz and Miike Gilbert, Access 2000 Developer's Handbook, Vol. 1, Sybex, 1999.
- [4] Angel R. Puerta, Henrik Eriksson, John H. Gennar, and Mark A. Musen, Model-Based Automated Generation of User Interfaces, In Proceedings of the 12th National Conference of Artificial Intelligence, Vol. 1, 1994, 471-477.
- [5] Yongzhen Ou, On Mapping Between UML and Entity-Relationship Model, In Proceedings of the UML Workshop, 1977, 45-57.