

Web 기반 수학 모형 구현에 관한 연구

A Study for a Web-based Implementation of Mathematical modeling.

이용대(Yongdae Lee)*, 박영준(Youngjoon Park)*, 김승권(Sheung-Kown Kim)*

* 고려대학교 산업공학과

Abstract

본 연구에서는 Web 기반 사용자 인터페이스를 사용하여 일반적인 수학 모형을 만드는 Web기반 클라이언트/서버 시스템의 구현 방법을 제시하고, 수학모형을 위한 데이터 베이스 설계 방법을 소개한다.

수학 모형을 설계하고 구현하기 위해서는 방대한 양의 데이터 처리가 반드시 필요하며 이를 위하여 데이터베이스를 수학 모형에 맞게 설계함으로써 보다 효과적으로 수학 모형을 구현 할 수 있다.

한편 Web을 이용한 클라이언트/서버 구조는 사용자가 Web Browser 만으로 프로그램을 이용할 수 있다. 본 연구에서는 3계층 데이터베이스 접근 모델을 갖는 Web기반 클라이언트/서버 구조를 설계하였으며, Front-End를 Java Applet을 사용하고, Back-End로써 실제 수학 모형을 만들어주는 Java Application 서버를 구현하며 Front-End와 Back-End 사이의 통신을 담당하는 네트워크 아키텍처로 자바 소켓을 사용한다.

일반적인 수학모형을 데이터베이스를 이용하여 구현함으로써 수리계획, 일정계획, 물류관리 등 수학 모형을 이용하여 최적화 결과를 도출하고자 할 때, 입력 부분과 출력 부분의 사용자 인터페이스만 구현하여 붙여주면 쉽게 응용하여 확장할 수 있다.

1. 서론

본 연구는 수학 모형을 수립하기를 원하는 모델러가 모델과 데이터만 정의해 주면 수학모형을 만들고 최적 해를 구하는 Web기반 클라이언트/서버 시스템을 구현하는 것이다. 이를 위해서 일반적인 수학 모형을 위한 데이터베이스 설계와, Web 기반 사용자 인터페이스가 필요하다.

보통 최적화 패키지를 이용하여 대규모 수학 모형을 풀기 위해서는 최적화 패키지 입력 파일을 만들어 주어야 한다. 하지만 이것은 프로그래밍에 익숙하지 못한 사용자에게 쉽지 않다. 그리고 모형이 바뀔 때마다 프로그램을 다시 작성하거나 수정을 많이 해야하는 불편함이 있다. 또한 방대한 데이터를 갖는 복잡한 대규모 수학 모형은 처리해야 할 데이터가 매우 많다. 더구나 풀고자 하는 문제가 비슷한 유형의 문제라면 그와 유사한 문제를 풀기 위하여 독립적인 모형을 구현해야 하고, 그에 필요한 최적화 엔진을 따로 구비해야하는 단점이 있다. 그러나 인터넷이라는 정보 인프라 골격이 출현함에 따라 업무 수요에 따라 분산되어 나타나는 유사한 정보처리 업무를 한 곳에서 통합하여 관리할 수 있게 되었다. 이제 인터넷은 단순히 정보를 제공하고 나누는 공간을 넘어 정보시스템, 전자상거래, 응용프로그램 수행 등 그 영역이 무한하게 발전하고 있다. 이와 같은 환경에서 Web기반 사용

자 인터페이스를 이용하면, 사용자가 Web Browser만으로 운영체제에 상관없이 프로그램을 편리하게 수행하고 원하는 분석을 할 수 있을 것이다.

이와 같은 문제점을 인식하여 Krishnan(1990)은 생산, 분배, 재고 계획 분야에 대하여 비전문가들이 LP model을 사용할 수 있도록 PDM이라는 tool을 제시하면서 prolog에서 쓸 수 있는 언어를 소개하였다. 또한 Fourer(1997)는 대규모 수학 모형을 위한 데이터베이스 스키마를 제시하고 철강 산업 계획 문제를 대상으로 데이터베이스 설계, 구축 방안을 제안하였다. 한편 Chou(1998)는 일반화된 웹 정보 시스템 구조를 제안하고, 웹기반 의사결정 시스템을 구현하였다. 그리고 김태형(1999)은 장비 교체 문제에 관한 의사 결정 지원 시스템을 웹기반으로 설계하고, 웹기반 의사 결정 지원 시스템의 기본적인 구조를 제시하였다.

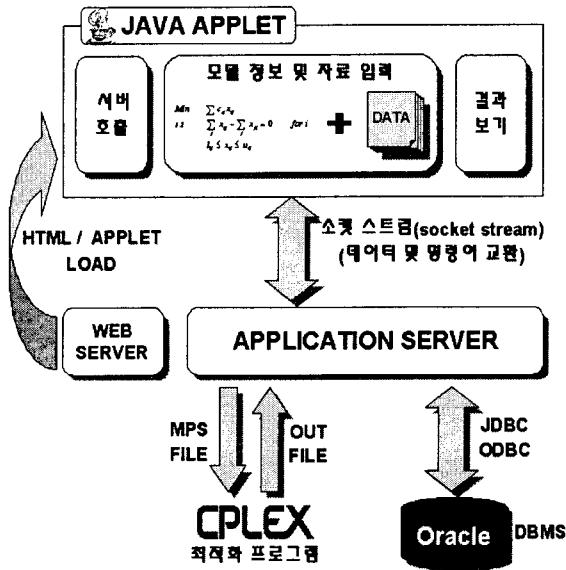
본 연구에서는 2계층 구조의 데이터베이스 접근 모델로 설계한 김태형(1999)의 연구와는 달리 3계층 구조로 설계하여 Client Applet이 빠르고 원활하게 작동 할 수 있도록 하는 Web기반 클라이언트/서버 시스템의 구현 방법을 제시한다. 그리고 Fourer(1997)보다 좀 더 일반적인 수학모형을 위한 데이터베이스 설계방법을 제안한다.

2. Web 기반 수학 모형 구현 시스템 소개

2.1 시스템 소개 및 개발 환경

이 시스템은 Web 기반 클라이언트/서버 컴퓨팅 환경에서, 수학 모형 모델러가 수학 모형과 데이터를 정의해 주면 수학 모형을 만들고 최적해를 구해 줄 수 있도록 하는 응용 프로그램이다.

이 시스템은 전체적으로 클라이언트 서버 구조로 되어있다. 먼저 클라이언트는 JAVA Applet을 Web Browser를 이용하여 실행시키며, 서버는 JAVA Application이 담당한다. [그림 1]

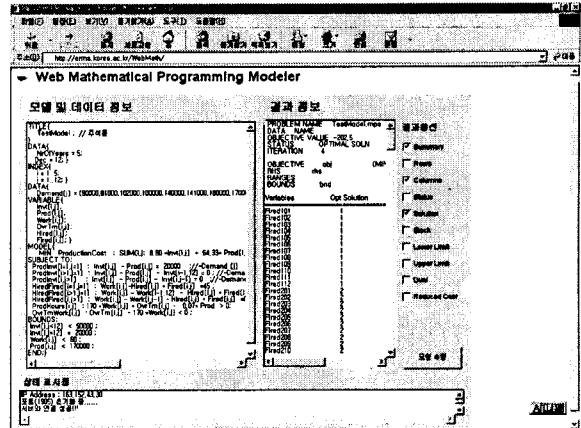


[그림 1] 시스템 아키텍처

시스템 작동은 사용자가 이 시스템 웹사이트에 접속한 후 Web Browser를 이용하여 클라이언트 JAVA Applet을 실행하면 Applet은 서버를 호출하여 서버와 클라이언트의 통신을 시작한다. Web Browser에는 [그림 2]에서 보는 바와 같이 모형 정보와 데이터를 입력하는 화면이 나타나고 사용자는 자신의 수학 모형과 데이터를 입력한 후 실행 버튼을 눌러 수행 명령을 내린다. 그러면 Applet은 모형 정보와 데이터를 Application Server로 보내주게 된다. Server는 모형 정보와 데이터를 읽어 데이터베이스에 입력하고, 그 자료를 이용하여 최적화 프로그램의 입력 파일인 MPS(Mathematical Programming System)파일을 만들어 준다. Server는 MPS 파일을 이용하여 최적화 프로그램을 수행하고 최적화 결과 파일을 받는다. 이 결과를 데이터베이스에 입력한 후 목적 함수 값 등 사용자가 원하는 결과 내용을 Applet에 보내준다. 사용자는 자신이 요구한 항목 결과를 확인 할 수 있고, 다른 결과를 추가로 받아 볼 수 있다. 그리고 모형을 수정하여 다시 위 과정을 수행 할 수 있다.

개발 환경은 Application Server가 JDK 1.2 기반으로 만들어 졌고, 클라이언트 Applet은 JDK 1.1.x 기반으로 작성하여 Web Browser(넷스케이프 4.5 이상, 익스플로러 4.0 이상)에서 실행 될 수 있다. 자바 개발 툴은 JBuilder3.0을 사용하였다. 그리고 DBMS는 오라클8을 사용하였지만 JDBC/ODBC

를 이용하여 DBMS에 독립적이다. 최적화 엔진은 CPLEX를 이용하였다.



[그림 2] Web 기반 수학 모형 구현 시스템 데모

2.2. Web 분산 컴퓨팅 구조

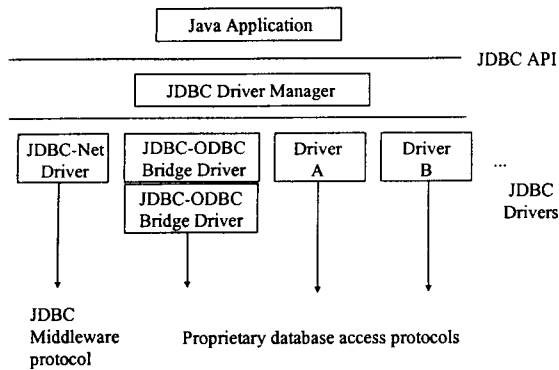
Web 기반 분산 컴퓨팅 기술은 HTTP/CGI, 자바 소켓, 서블릿, RMI, CORBA 등이 있다.[4] 이 중 본 연구에서는 자바 소켓을 이용하여 Web과 서버의 통신을 구현하였다.

소켓은 통신을 하기 위하여 이름과 네트워크 주소를 갖는다. TCP/IP 상의 소켓 주소는 인터넷 프로토콜(IP: Internet Protocol) 주소와 포트 주소 두 부분으로 구성된다. 포트는 같은 호스트에서 여러 응용 프로그램이 실행 될 수 있으므로 클라이언트와 서버 사이의 약속된 번호이다. 포트는 0~65,536번까지 사용할 수 있다.[5]

클라이언트/서버 네트워크 연결되는 절차는 먼저 서버에서 포트 번호를 갖고 서버 소켓을 연후 클라이언트의 접속을 기다린다. 클라이언트는 서버의 정해진 IP 주소와 포트 번호를 갖고 소켓을 열어 서버와의 통신을 시도한다. 통신이 성공하면 클라이언트와 서버는 서로 문자열을 주고받을 수 있다. 이 문자열을 이용하여 명령을 전달 할 수 있으며, 통신을 닫을 때에도 서로 약속된 문자열을 주고받은 후 통신을 마친다.

2.3. 데이터베이스 접근 모델

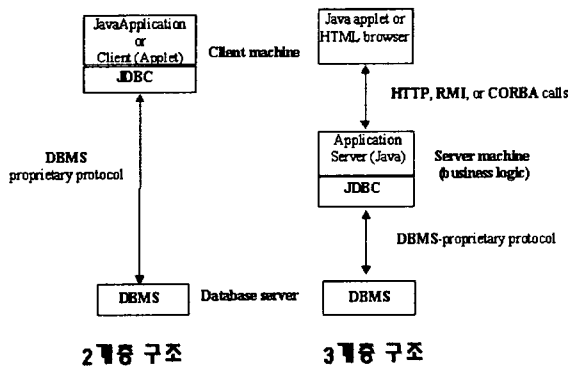
자바는 JDBC(Java DataBase Connectivity)라는 자바 어플리케이션 표준 SQL API(Application Programming Interface)를 지원한다. JDBC를 이용하면 자바가 실행 될 수 있는 모든 운영 체제에서 DBMS에 독립적으로 데이터베이스에 연결 할 수 있다. JDBC는 [그림 3]과 같은 구조를 갖는다.[6]



[그림 3] JDBC 구조도

JDBC를 이용하여 데이터베이스에 접근하기 위해서는 JDBC 드라이버가 필요하다. JDBC 드라이버는 JDBC-ODBC 브리지 드라이버, 데이터베이스 API 드라이버, 네트워크 프로토콜 드라이버, 데이터베이스 프로토콜 드라이버 등이 있다. 특히 JDBC-ODBC 브리지 드라이버는 3계층 구조에 자주 쓰인다.

한편 JDBC 데이터베이스 접근 모델은 [그림 4]와 같이 클라이언트, 서버, 데이터베이스 사이의 구조에 따라 2계층 구조와 3계층 구조로 나눌 수 있다.[6] 2계층 구조는 자바 애플릿과 자바 어플리케이션이 JDBC 드라이버를 통하여 직접 접근하는 형식이다. 3계층 구조는 2계층 구조와 달리 미들웨어 계층이 첨가되어 데이터베이스에 직접 질의를 하는 구조이다. 3계층 구조는 사용자 인터페이스와 데이터베이스 질의 코드가 분리되어 프로그램을 확장하거나 수정하는 것이 편리하다. 하지만 2계층 구조 보다 속도가 약간 느려 진다는 단점이 있다.



[그림 4] 데이터베이스 접근 모델

본 연구는 [그림 1]에 나타난 것처럼 ODBC-JDBC 브리지를 이용하여 3계층 구조의 JDBC 접근 모델을 사용하였다. 3계층 구조를 사용하면 클라이언트 Applet의 크기와 역할이 줄어들어 Applet이 Web Browser에서 더 빠르게 수행 될 수 있다. 또한 ODBC-JDBC 브리지 드라이버는 클라이언트에 드라이버를 설치하지 않고 Application 서

버를 통해 데이터베이스에 접근 할 수 있다.

2.4. 일반적 수학 모형을 위한 데이터베이스 설계

일반적으로 수학 모형은 목적함수와 제약식, 변수, 계수 등으로 구성된다. 이와 같은 수학 모형의 요소들을 바탕으로 [그림 5]와 같이 일반적 수학 모형을 위한 데이터베이스를 설계하였다.

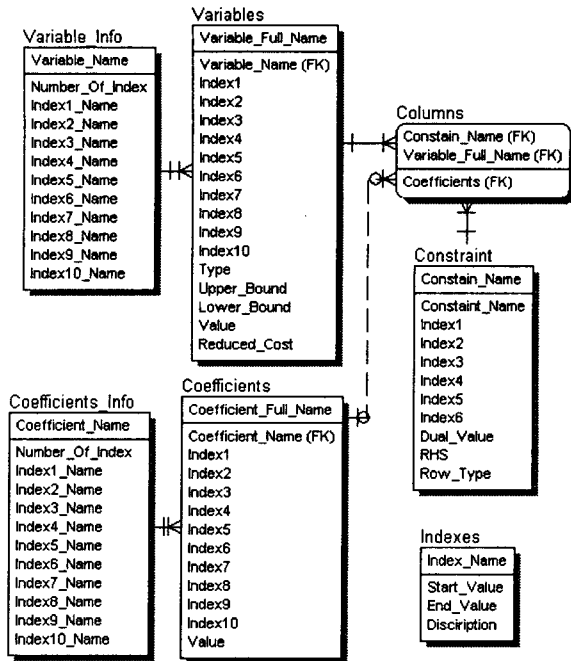
먼저 변수에 관한 항목들로 Variables_Info 테이블과 Variables 테이블이 있다. Variables_Info 테이블은 변수에 대한 특성 정보를 담고 있다. 즉 변수의 이름과 인덱스의 수, 그리고 각 인덱스 이름 정보를 갖고 있다. 예를 들면 X_{ij} 라는 변수는 변수 이름이 X가 되고 인덱스의 수는 2, 인덱스 이름은 i, j가 된다. 한편 Variables 테이블은 Variables_Info 테이블을 바탕으로 수학 모형에 쓰이는 모든 변수 정보를 갖고 있다. 즉 X_{ij} 대신

X_{11}, X_{12}, X_{13} 과 같이 실제 수학 모형에 사용되는 정보를 담게된다. 그리고 각 변수의 상한 값, 하한 값, 정수 변수에 관한 정보 갖는다. 그리고 최적 결과 값과 Reduced Cost 정보를 모형 수행 후 얻게 된다.

그리고 계수 값에 관한 정보는 Coefficients_Info 와 Coefficients 테이블에서 저장한다. Coefficients_Info 는 Variables_Info 테이블과 같은 형태로 계수에 대한 특성들에 관한 정보를 담고 있다. 또 Coefficients도 Variables Table 처럼 Coefficients_Info 테이블을 바탕으로 수학 모형에 쓰이는 계수 정보를 갖고 있다.

한편 Constraints 테이블은 모형의 제약식에 관련된 정보를 저장한다. 제약식 이름과 인덱스 정보, RHS, 등식에 관한 정보 등을 저장하고 Index Table은 변수, 계수, 제약식에서 사용하는 인덱스에 대한 정보를 갖고 있다. 인덱스의 시작 값과 마지막 값 등에 관한 정보를 갖는다.

그리고 다른 각 테이블의 정보들을 모아 Columns 테이블을 구성한다. Columns 테이블은 최종적으로 MPS 파일의 Columns 정보를 만들 때 필요한 값을 담게 된다.



[그림 5] 일반적 수학 모형을 위한 데이터베이스 스키마

본 연구에서 설계한 일반적 수학 모형을 위한 데이터베이스는 수학적 모형에 필요한 정보를 최대한 일반화하여 담고 있다. 즉 거의 모든 형태의 LP, IP, MIP 등 수학 모형에 관한 정보를 갖게 된다. 하지만 생산 계획 문제, 물류관리 문제, 수자원 운영 계획 문제 등 특정 형태의 수학 모델에 맞게 만든 데이터베이스보다는 효율이 떨어진다는 단점이 있다. 특정 수학 모델에 맞는 데이터베이스를 설계하려면 인덱스의 수 등을 조정할 필요가 있다.

3. 결론 및 향후 연구 과제

본 연구에서는 Web을 통해 수학 모형 모델러가 수학 모형과 데이터만 정의해 주면 수학적 모형을 만들고 최적 해를 구하는 Web기반 클라이언트/서버 시스템을 구현했고, 일반적 수학 모형에 맞는 데이터베이스를 설계하였다. 그리고 Applet의 크기와 역할을 줄이기 위해 3계층 구조를 갖는 Web기반 클라이언트/서버 시스템을 JDBC/ODBC 브리지를 사용하여 설계했고, 자바 소켓을 사용하여 서버와 Applet 사이의 통신을 구현하였다.

Web기반 사용자 인터페이스를 제공하는 Application 프로그램은 보안문제에 대한 연구가 더욱 요구되며, 분산 컴퓨팅 기술과 멀티 쓰레딩 기법을 이용하여 하나의 서버에 여러 사용자가 동시에 작업할 수 있는 방법 등에 관한 연구가 필요하다.

일반적 수학 모형을 위한 데이터베이스가 가질 수밖에 없는 비효율성 문제를 해결하기 위해 객체 지향 데이터베이스를 이용한 연구가 필요할 것이다. 또한 객체 지향 모델링을 통해 좀더 쉽고 효율적으로 일반적 수학 모형을 구현할 수 있을 것이다.

참고 문헌

- [1] Chou, Seng-cho T., Migrating to the Web: a Web financial information system server, Decision Support Systems, Vol. 23, pp29~40, 1998
- [2] Fourer, R. (1997), Database structures for mathematical programming models, Decision Support Systems, 20, 317-344
- [3] Krishnan, R. (1990), A logic modeling language for automated model construction, Decision Support Systems, 6, 123-152
- [4] Orfali, R., and Harkey, D., Client/Server Programming with JAVA and CORBA Second Edition, John Wiley & Sons, Inc., 1998.
- [5] Naba Barkati, Java Annotated Archives, McGraw-Hill, Inc., 2000.
- [6] 김태형, 수리적 분석을 위한 웹 기반 의사결정 지원 시스템의 설계와 구현, 고려대학교 석사학위 논문, 1999.