

설계정보 데이터베이스를 기반으로 하는 구조설계 S/W Platform의 개발

Development of S/W Platform for the Structural Design System Based on Design Database

이 대 희* · 이 정 재**
Lee, Dae Hee · Lee, Jeong Jae
김 한 중*** · 윤 성 수*
Kim, Han Joong · Yoon, Seong Soo

Abstract

Agricultural facilities can be designed by conceptual, preliminary, primary, and detailed design stages. Because each stage has a different objective, it must be applied to dissimilar design and analysis level. Engineers have to judge from evaluation that is estimated agricultural resources in the conceptual design stage. Methods of computer aided design are achieved to individual functions, however it is inefficient to perform entire processes of design and hard to systematically accumulate results of design. Study on the integrated structural design system has been continued, but those system have adopted "closed architecture" which is known to inflexible method for developing applications.

In this study, the design platform is an environment that can support to integrated design system independently, and an design platform is proposed by analyzing design processes using object oriented method. The concepts of software platform have resulted from several practical ideas, OOA/OOP and related some points. This paper aims at developing the software platform of a software environment to support the design phases of the agricultural facilities.

I. 서 론

구조물의 설계는 구조계획, 예비설계, 구조해석, 부재설계 및 검토와 같은 과정을 반복하면서 최적

의 결과에 접근해야 한다. 지금까지의 설계 과정에 관한 전산화는 반복적인 단위 업무를 지원할 수 있는 S/W의 개발이 진행되어 왔다. 특히 컴퓨터의 빠른 연산능력을 이용해야 하는 구조물 해석과

*서울대학교 대학원

**서울대학교 농업생명과학대학

*** (재) 생산공학연구소

키워드 : 소프트웨어 플랫폼, 데이터베이스,
통합설계, 구조설계

같이 문제가 정형화된 영역에서는 많은 발전이 있었으며, 이전 단계의 설계결과와 참조나 재사용에 의한 설계 경험의 활용 및 설계지식의 축적에 관한 요구를 충분히 충족시키지는 못하지만 비교적 규격화와 표준화가 용이한 건축구조설계와 기계설계 등에서는 설계지원 S/W가 계획, 설계, 해석 및 부재 설계 과정을 한 시스템에서 모두 포함하고자 하는 통합 설계 시스템으로 발전하고 있다.

기존의 개발된 통합설계 시스템은 설계 순서에 따라 모듈을 제작하고, 이 모듈을 연결하는 시스템을 구성하는 방법을 채택하고 있기 때문에 개발환경 자체(OS) 또는 사용자지원 P/G(GUI) 등이 더 이상 환경의 변화에 적응이 불가능한 문제점이 제기되었으며, 기존의 개발된 프로그램에서 재사용이 불가능한 경우가 발생하면서 자원활용의 측면에서 이에 대한 대책이 요구되었다 [1,3,12]. 뿐만 아니라 기존의 S/W 개발 방법론에 의해서 구축되었기 때문에 시스템의 확장과 유지관리를 위해 너무 많은 비용과 시간을 요하게 되는 문제점을 안고 있었다. S/W 개발 측면에서는 근래에 많이 연구된 객체지향 기법(OOA/OOP)이 도입되면서 규모가 큰 통합 설계 지원 시스템의 개발이 가능하게 되었으며 [10], 파일 단위로 이루어지던 자료의 가공이 데이터베이스로부터 제공되는 방식으로 전환되고 있다 [3,7,8]. 또한 데이터베이스를 기반으로 하는 경우에는 데이터베이스의 규격에 따른 자료의 규격화가 가능하며, 서로 다른 시스템간에 자료를 공유할 수 있어 보다 나은 정보의 공유방법으로 이용되고 있다. 그러나 현재까지 개발된 데이터베이스 운영 프로그램과 설계지원 시스템들은 각각의 개발환경과 시스템 의존성이 강하여 이와 같은 장점을 잘 이용하지 못하고 있다.

따라서, 본 연구에서는 토목 구조물의 통합설계 시스템 구현을 위하여, 설계와 관련된 정보들이 데이터베이스의 형태로 저장되도록 하여 공유와 재사용과 설계에 활용할 수 있는 하나의 사용 환경으로서 구조설계 플랫폼을 개발하고, 서로 다른 목적으로 개발된 시스템에서도 적용이 가능한 정보 공유 방안을 제시하고자 한다.

II. 설계정보의 분석

1. 설계정보를 위한 데이터 모델

복합적인 설계과정을 효과적으로 지원해 줄 수 있는 설계시스템을 구성하기 위해서는 먼저 시스템에 구현될 설계정보와 작업 과정들을 개념적 모델로 형상화하여야 한다. 정형화된 개념적 데이터 모델은 시스템의 유연성을 보장해줄 뿐 만 아니라, 개발자나 사용자 모두에 있어 시스템을 쉽게 이해할 수 있도록 도와주고, 시스템의 수명기간을 연장할 수 있는 것으로 알려져 있다 [16]. 설계분야에서는 데이터 모델을 도입하여 설계 대상에 대한 정보를 형상화하는 모델(product model)과 설계과정(process model)을 형상화하는 방법으로 구분하여 정의한 바 있다 [12,15]. 이들은 개체-관계형(E-R) 모델에 바탕을 두고 있으며, 객체지향 모델로부터 상속, 연관 등의 개념을 도입하여 구조물간의 관계를 표현하고 있다.

설계 작업에 필요한 정보 및 중간 과정에서 만들어지는 설계정보와 결과는 데이터베이스에 집적됨으로써 전산화된 설계과정에서 재활용될 수 있다. 따라서 설계 정보 데이터베이스는 통합 설계 시스템 구축의 핵심적 부분이라 할 수 있다. 천 [11]은 객체지향 분석에 의해 설계과정을 분석하고 객체지향 데이터베이스를 사용해 구축한 바 있으며, 객체지향모델과 관계형 모델의 변환 가능성을 보여 주는 연구가 시도되었다 [9]. 이들의 연구는 구조물을 적절한 클래스로 표현하고 이를 객체지향 데이터베이스에 구축하는데 관심을 집중하고 있다. 그러나 이 방법은 데이터베이스를 활용하는 측면에서 설계 통합시스템 구축에는 적합한 것으로 판단되나 객체지향데이터베이스를 기초로 한 통합 데이터베이스 설계 및 운영프로그램의 개발 분야에 적용하기 위해서는 보다 많은 검토가 필요한 것으로 생각된다.

구조 설계 소프트웨어의 현안으로서 적응적인 소프트웨어(adaptive S/W)의 개발, 개발 시간의 절약 문제와 복잡한 공학적 개체(entity)의 표현

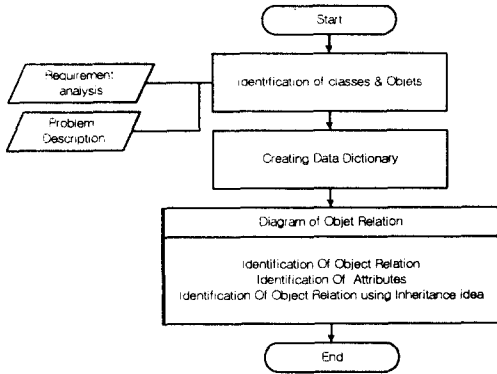


Fig. 1 Procedure of object modeling

등이 있다. 이를 위하여 본 연구에서는 설계와 연관된 상이한 작업들은 먼저 객체지향설계법(Object-Oriented Analysis/Design)을 이용하여 시스템의 운영 및 설계과정을 분석하고 시스템을 개발하였으며, 데이터베이스의 변화요구에 대응하기 위해 요구되는 통합구조설계 시스템의 데이터 관리를 위하여 관계형 데이터베이스를 도입하였다.

2. 구조물 설계 정보의 객체지향분석

설계는 구조물 모델링 과정, 수치 해석, 설계의 지침 및 표준 등 복잡한 기준의 해석과정을 포함한다. 이를 효과적으로 전산화하기 위해서는 각 단계별로 발생하는 객체와 설계정보의 흐름을 묘사하여 객체를 모델링 할 수 있는 방법으로 공학 분야에 있어서 널리 사용되고 있는 Rumbaugh의 객체 모델링 기법(OMT : Object-Modeling Technique)을 채택하였다. 객체 모델링의 절차는 Fig. 1의 절차를 따른다.

객체의 설계는 객체 및 클래스의 식별과 데이터 사전의 작성, 객체 관계의 작성, 객체 속성의 정의 등으로 진행된다. 객체 및 클래스를 식별하기 위해서 현행 시스템의 사용자, 관리자, 개발자로부터 현 시스템을 기술하는 요구 분석서를 작성하고 이를 통해 현행 시스템의 문제점 파악과 새로이 개발될 시스템을 서술하는 문제 설명서를 작성할 수 있다. 본 연구에서는 문제 설명서로부터 명사를

공사, 프로젝트, 부지, 시행자, 시공자, 감리자, 설계자, 구조물, 2차원 구조물, 3차원 구조물, 해석, 뼈대 구조물, 슬리트 구조물, 1차원 요소, 2차원 요소, 3차원 요소, 요소, 노드, 재료, 단면, 단면적, 단면 2차 모멘트, 자유도, 구속조건, 하중, 정적하중, 분포하중, 삼각형 분포, 사각형 분포, 해석기, 변위, 응력, 요구사항, 허용 범위, 치수, 해석 결과, 설계 공간, 최적화, 구조 계산서, 물량, 단가

Fig. 2 Identified classes

식별하여 후보 클래스로 선정한 다음에 이들 중에서 동작만을 의미하거나 의미가 모호한 부적절한 단어를 제거하여 시스템을 표현할 수 있는 클래스들을 정의하는데 사용하였다. Fig. 2는 식별된 클래스이다. 설계된 객체는 상호간의 위상관계(topology) 형성과정을 통하여 객체간의 관계성 정의, 객체의 속성 정의 및 상속관계를 정의하였다. Fig. 3은 설계정보 구성에 사용된 객체-관계도 이다.

III. 구조설계 S/W 플랫폼의 개발

1. S/W 플랫폼 계층의 정의

구조설계용 S/W의 개발은 많은 단계의 설계 과정과 관련된 단위 기능들의 복합적인 결합이 요구된다. 설계 과정은 Fig. 4와 같이 단위 기능인 해석, 전처리, 후처리, 최적화, 부재설계 등으로 세분화 할 수 있다. 따라서 각 단위 기능별로 개발된 시스템은 설계를 수행하는 전과정에서 일관되게 입출력을 처리하도록 하여야 한다. 본 연구에서는 구조설계 S/W 플랫폼의 개발을 위하여 Fig. 5 a)와 같이 사전에 약속된 파일기반의 연동 구조 대신에 Fig. 5 b)처럼 독립적으로 개발된 각 응용프로그램이 연결될 수 있도록 하여 자료의 축적과 새로운 응용프로그램에 대한 적응성을 높일 수 있도록 하였다. 이는 정형화된 데이터 관리 방식(DBMS, Database Management System)과 SQL이라는 표준화된 접근 방식을 지원하여 해결하였다. 또한 비교적 사용 경험이 적은 데이터베이스

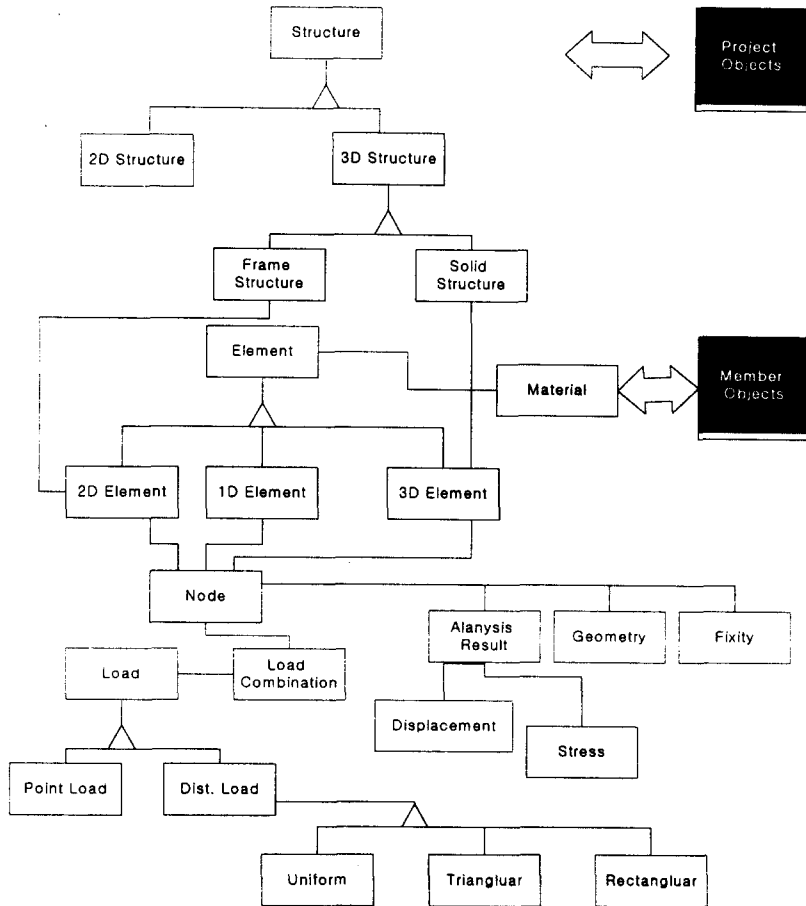


Fig. 3 Class diagram using hierarchical concept

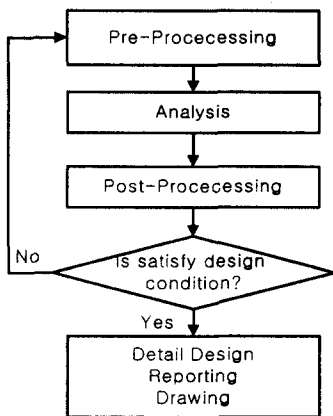
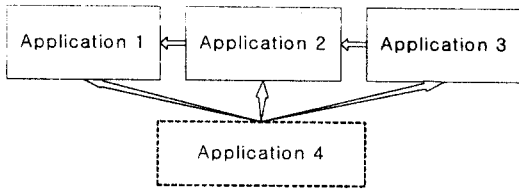


Fig. 4 Procedure of structural design

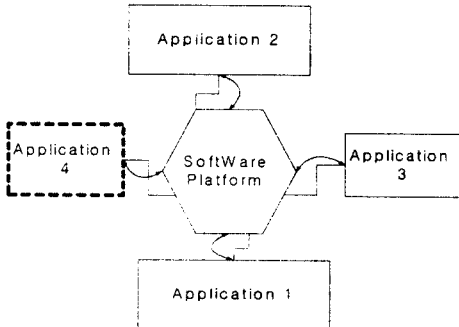
에 대한 생소함을 해소하기 위하여 데이터베이스에 대한 인터페이스를 두어 응용프로그램이 표준화된 데이터베이스의 조작방법을 지원하도록 하였다.

2. 응용프로그램 통합을 위한 S/W 플랫폼의 구성

단위업무 중심으로 개발된 응용프로그램 중에서 별도의 사용자 지원 인터페이스가 없이 파일 기반의 입·출력을 사용하는 프로그램이 S/W 플랫폼에서 통합될 수 있도록 인터페이스 계층과 응용프로그램 계층, DB 관리자를 위한 데이터베이스 인터페이스계층과 전·후처리기 계층을 개발하였다.



a) Previous pattern for developing each unit application



b) Ideal pattern for developing applications based on the S/W Platform

Fig. 5 Platform with applications

가. 인터페이스 계층

설계정보의 이용 과정에서 데이터베이스와 관련된 응용프로그램이 데이터베이스에 안전하게 접근하여 필요한 정보를 추출하고 변경하며 추가, 삭제하는 등의 조작이 가능하도록 하기 위해서는 이를 위한 인터페이스가 필요하다. 통합데이터베이스의 경우에는 네트워크상의 여러 기종의 데이터베이스 관리시스템에 대한 접속방법과 데이터 조작, 비교적 다양한 응용프로그램을 위한 데이터 제공 방법 등이 모두 고려되어야 한다. 여러 가지 기종의 데이터베이스에 대한 인터페이스 설계를 위해서는 필요한 데이터 베이스에 대한 표준적인 접속 드라이버(driver)를 필요로 한다. 따라서 ODBC 환경과 Java를 이용하는 경우에는 JDBC 또는 JDBC-ODBC드라이버를 이용하여 데이터베이스를 필요로 하는 객체가 여러 응용 프로그램에서 공통된 형식으로 설계된 SQLProcessor를 상속하여 적절한 작업의 Processor에 의해 SQL로 변환후에 처리되도록 개발하였다.

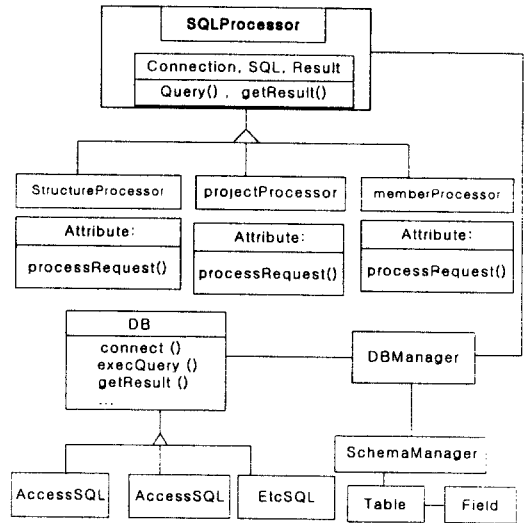


Fig. 6 Database interface structure

나. 응용프로그램 계층

응용프로그램 계층은 실제 작업이 수행되는 공간으로 인터페이스를 통해 데이터베이스로부터 가져온 정보를 가공하고, 인터페이스를 거쳐 데이터베이스로 정보를 저장하는 과정을 지원한다. 설계 과정은 하나의 단위 과정으로 처리되지 않는 복합 작업이므로 각 작업간의 연관 관계, 즉 작업의 순서나 상호 필요성 등을 관리할 필요가 있다. 따라서 응용프로그램 계층에서는 응용프로그램을 등록하고 연관 관계에 따라 수행시키는 장치가 필요하게 된다. 하나의 응용프로그램은 입출력 과정이 데이터베이스와의 연관성만으로 처리되도록 하여 다른 응용프로그램에 대해 독립적으로 작동할 수 있도록 하였다. 이러한 독립적인 작동 특성은 설계 과정을 이루는 여러 작업 중에도 보완되거나 교체가 요구되는 경우 프로그램의 유지 보수를 편리하게 할 것으로 기대된다.

다. 데이터베이스 관리자(DBManager) 계층

DBManager는 네트워크에서 데이터베이스의 구조와 자료를 조작하는 것을 지원한다. 기존 응용프로그램을 사용하기 위해서는 우선 입출력 자료의 내용을 시스템 설계자가 분석하여 현재 사용

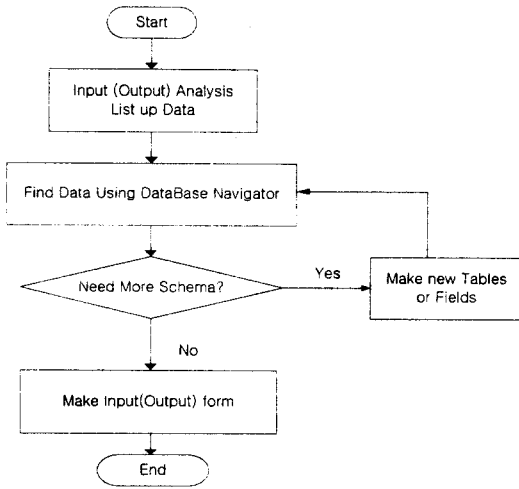


Fig. 7 Procedure of I/O form generation based on the database connection module

가능한 데이터베이스의 스키마와 내용으로 들어갈 수 있는지를 판단해야한다. 만약 새로운 스키마가 필요한 경우 Fig. 7과 같이 DBManager의 해당 기능을 사용하여 처리할 수 있으며, 이때 데이터베이스 관리시스템은 네트워크 상에서 동시에 여러 개를 접속할 수 있도록 한다. 또한 동시에 여러 개의 스키마를 관리할 수 있다.

입력자료가 복잡하고 반복적인 과정을 갖는 경우에 이 과정은 더욱 어려워진다. 따라서 본 연구에서는 입력자료의 구성과 데이터베이스 입력, 출력자료의 데이터베이스 입력과정을 데이터베이스의 표준 인터페이스 언어 중에서 자료의 입·출력과 반복작업의 수행, 필요한 항목의 생성 등을 자동적으로 처리할 수 있는 언어명세를 갖는 스크립트를 정의하여 사용하였다. 이 스크립트 언어의 작성 및 해석과정은 Fig. 8에서와 같이 진행된다. 플랫폼 스크립트는 데이터베이스 관련 기능, 변수 관리 관련 기능, 반복처리 관련 기능, 토큰(token) 처리 관련 기능을 수행할 수 있으며, Table 1과 같이 정의하였다.

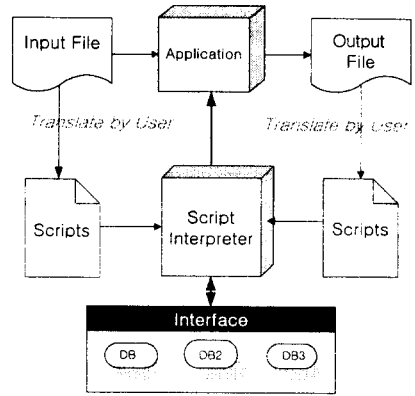


Fig. 8 Procedure of script interpretation

Table 1 Platform script commands

Category	Command	Function
Data Base	Add_DB	Add database to database list
	SET_DB	Set current database
	SQL	Query SQL
FILE MANIPULATION	READ	Read a token
	WRITE	Write a token
	SKIP_LINE	Skip a line
	SKIP_TOKEN	Skip a token
VARIABLE	SET_INT_VAR	Set variable & its value
	SET_FLOAT_VAR	Set variable & its value
	SET_DOUBLE_VAR	Set variable & its value
	SET_CHAR_VAR	Set variable & its value
	VAR	Replace with variable value
LOOP	FOR	For loop
	WHILE	While loop

라. 사용자지원 환경(GUI)과 전후처리 (Pre/post processor) 계층

공통적인 해석 및 설계지원 환경으로써 설계 플랫폼의 두 가지 개발 전략은 기존 프로그램의 작동 환경 제공, 순수하게 설계 플랫폼 기반으로도 프로그램의 개발이 가능하도록 하는 것이다. 본 연구에서 개발한 GUI와 Pre/post processor 계층은 객체지향 언어인 Java™를 사용하여 개발하였다. 유한 요소 해석이나 경계요소법 등 수치 해석적인 방법으로 구조물을 해석하기 위해서는 구조

물을 요소로 분해하는 과정인 전처리 과정이 필요한데, 이 과정은 매우 간단한 구조물 이외의 현실적인 구조물에서는 방대한 양의 계산을 요하므로 수 작업으로 처리하기가 곤란하기 때문에 구조물의 형상, 하중의 정의 및 하중조합, 부재의 단면 정의 등을 수행하고 이차적으로 노드와 요소를 생성하여 해석기의 입력자료를 제공한다. 본 연구에서는 전처리기에서 생성된 파일을 입력자료로 사용하는 해석기에 제공하는 과정에서 입력자료가 먼저 데이터베이스에 저장되도록 하였다. 따라서 이후에는 전·후처리를 사용하지 않고도 데이터베이스를 통하여 부재의 재원과 물성치를 변경할 수 있게 된다. 또한 데이터의 저장과 관리, 표현에 있어서 확장성이 뛰어난 GUI의 개발을 위하여 Model-View-Control (MVC) 개념을 채택하였다. 이 방법은 정보를 저장하는 Model, 실제 사용자가 볼 수 있는 View, 사용자와의 상호 관계를 처리하는 Control로 GUI가 구성된다.

IV. 적용 및 고찰

1. S/W 플랫폼 기반의 자료처리 기법

설계업무의 전산화 과정에서 많은 단위 기능의 응용프로그램들이 만들어져 왔으며, 실무에 이용되고 있다. 이러한 응용프로그램 중 상당수는 장기간의 적용을 통해 효용성을 검증 받았을 뿐만 아니라 많은 사용자를 확보하고 있으며, 또한 UNIX와 DOS 외에도 다양한 운영체제에서 개발되었을 뿐만 아니라 Fortran, Cobol, C 언어 등으로 개발되었다. 따라서 이들을 짧은 기간에 통합적으로 사용할 수 있도록 개선하는데는 많은 어려움이 존재한다. 그러나 설계 플랫폼은 응용프로그램과 데이터베이스 인터페이스 계층을 이용하여 입출력이 단일한 과정으로 처리될 수 있기 때문에 자료처리 단계에서 통합을 이룰 수 있으며, 또 대상 응용프로그램들에게 별도의 개선작업, 즉, 사용자 지원환경(GUI) 등의 개발을 요구하지 않는 장점을 갖는다. 따라서 통합 설계에의 수행에 있어서

단위 기능을 담당하는데 적합할 것으로 판단된다.

구조설계 S/W 플랫폼 기반의 구조설계 통합시스템을 구현하기 위하여 단위 응용프로그램과 모델링, 설계 제원에 대한 데이터베이스 구축이 비교적 용이한 뼈대구조물의 설계문제를 대상으로 하였다. Table 2은 구조설계 플랫폼에서 사용하는 자료구조 스키마(Schema)이며, Table 3는 자료요구가 뼈대구조물의 해석과정에서 임의의 5번 절점에서 x축 방향의 변위 u를 가져오기 위해서는 Interface API의 StructureSQL의 getDisplacement 함수를 이용하는 과정을 보이고 있다. 함수에서는 인수를 받아들여 SQL문장을 생성하여 데이터베이스로부터 원하는 결과를 얻게 된다. 따라서 Fig. 6에서와 같이 설계 플랫폼 기반의 프로그램은 모두 SQLProcessor 객체를 상속하여 개발된 Interface API를 사용함으로써 쉽게 플랫폼에 통합될 수 있다.

Table 2의 명령어를 이용하면 Fig. 9와 같은 유한요소해석 입력자료를 데이터베이스에 첨가하고 이를 검색하여 추후에도 이용할 수 있다. 본 연구에서 이용한 FEA는 C언어를 사용해서 개발된 "3차원 뼈대구조물의 유한요소해석" 프로그램이며 Fig. 10은 입력자료의 일부 형식이다. 입력자료를

Table 2 Sample schema of framed structural analysis

Table Name	Field Name	Type	Description
Displacement	id	int4	Identification number
	u	float8	Displacement of X direction
	v	float8	Displacement of Y direction
	w	float8	Displacement of Z direction
	rx	float8	Rotation angle of XY plane
	ry	float8	Rotation angle of YZ plane
	rz	float8	Rotation angle of ZX plane

Table 3 Relation among request, API and SQL

Request	API	SQL
Need 5th nodes Displacement of X direction	StructureSQL.getDisplacement(5, StructureSQL.XX)	Select * from Displacement where Displacement.id = 5;

```

%%%TOTAL-ELEMENT, TOTAL-NODE,
ELE-NODE, MATERIAL-NO,
%%% LOAD-NO, DEGREE-OF-FREEDOM
NO. OF LOAD_COMB
CCC CONTROL DATA
7 8 2 1 2 6 2
%%%NO, ELTYPE, NODE1, NODE2,
MATERIAL, DIS-LOAD
0 1 0 1 0 -1 -1
1 1 1 3 1 1 -1
2 1 1 4 0 -1 -1
.....
%%%NO, A, E, G, Ix, Iy, Iz, J
0 8.36 29000000.0 11200000.0 0.0 27.0 27.0 46.8
%%%NO TYPE AXIS MAGNITUDE
LLL LOAD DATA
0 2 2 -10000.0
1 5 2 -416.7
%%% NO COMBINATION
SSS LOAD COMBINATION
0 0 -1
1 1 -1
    
```

Fig. 9 Required input file for FEA

```

FileName
ADD_DB 파닉스
jdbc:postgresql://IPaddress:5432/DatabaseName
userName s_wild postgresql.Driver what
SET_INT_VAR STR_ID 999
FOR 012
NEXT_TOKEN
READ_INTEGER T_ELEMENT
READ_INTEGER T_NODE
.....
NEXT_TOKEN
FOR 0 T_ELEMENT
SQL INSERT INTO ELEMENT_ID
VALUES ( READ , VAR STR_ID , READ ,
READ , READ ,
READ NEXT_TOKEN NEXT_TOKEN ) :
.....
    
```

Fig. 10 Platform script saves input file to DB

데이터베이스에 저장하기 위하여 번역한 결과의 일부를 보면 Fig. 11과 같다. 문제별로 입력자료를 작성하지만 공통적인 자료구조에 문제의 정보를

저장할 수 있기 때문에 이 방법은 한번 입력된 자료는 해석 뿐만이 아니라 설계과정에서도 이용될 수 있을 것으로 판단된다. 즉, 해석결과를 Fig. 12와 같이 시각적으로 볼 수 없는 환경에서는 데이터베이스 질의문을 통하여 원하는 부재의 응력과 변위를 검색 할 수도 있으며, 설계의 적합성을 판단하는 응용프로그램에서 데이터베이스가 새로운 입력자료 생성방법으로 활용될 수 있을 것으로 판단된다.

2. 설계 S/W 플랫폼 기반의 응용프로그램 개발 기법

새로 개발되는 프로그램이 설계 플랫폼을 기반으로 하는 경우에는 기존 프로그램과의 연계, 풍부한 자료의 이용 가능성, 자료 처리에 대한 부담 경감, 중복된 기능의 구현 노력 절감 등의 측면에 있어서 장점을 가질 것으로 기대된다. 프로그램의 개발 편의를 위해 플랫폼의 인터페이스는 API의 형태를 갖도록 하였다. 일반 프로그래머가 이 '설계 S/W 플랫폼'을 기반으로 하는 응용 프로그램을 개발하기 위해서는 개발 단계에서부터 자바(Java)를 사용하고 설계 플랫폼의 Interface API와 데이터베이스를 사용하는 개념을 채택하면 된다. 그러나 설계 플랫폼을 기반으로 작성된 전·후처리를 사용하는 경우에는 플랫폼 스크립트와 같은 부가적인 장치를 필요로 하지 않기 때문에 모델의 변경과 해석 결과는 동시에 설계 플랫폼의 인터페이스를 통해 데이터베이스에서 처리될 수 있다. Fig. 11과 Fig. 12는 해석모델의 전처리와 후처리 과정의 일부이다. Fig. 13에서와 같이 Application Manager가 사용 가능한 단위 프로그램을 나열하고 이들의 작동 순서 및 연관 관계를 맺어주는 역할을 하기 때문에 기존 응용프로그램인 해석 엔진과 플랫폼 기반 응용프로그램을 연동 하여 설계과정에서 해석 부분의 전후처리와 해석 결과의 시각화 과정을 지원하는 시스템을 손쉽게 개발할 수 있었다.

Fig. 11에서 작성된 구조물의 모델링 결과는 해

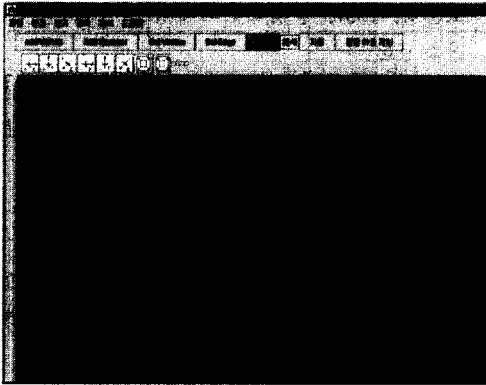


Fig. 11 Model structure created by platform

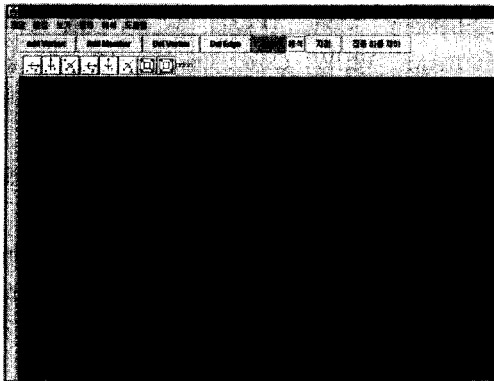


Fig. 12 Structure deformation diagram

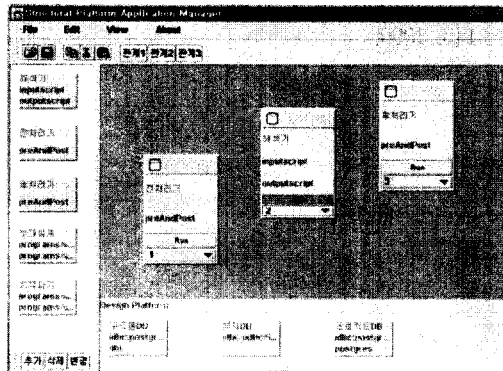


Fig. 13 Connected several procedures made by Application Manager

석기인 FEA에서 사용할 수 있도록 DB를 검색하여 입력자료를 파일로 만들어 주고, 또한 FEA의 해석결과는 다시 데이터베이스에 저장된다. 만약 설계자가 해석 결과를 보고 싶다면 DB로부터 검색하여 그 결과를 Fig 12와 같이 도출할 수 있다. Fig. 13에서 제시한 Application Manager는 몇 개의 응용프로그램이 추가적으로 요구되는 경우에 좌측에 등록된 응용프로그램 리스트로부터 선택하여 이전 단계의 결과와 DB 수준에서의 연관 관계를 정의하는 것으로 GUI와 자료의 전·후처리과정이 제공되는 시스템의 개발과정을 완료할 수 있다.

V. 결 론

본 연구에서는 응용프로그램과 데이터베이스가 연동할 수 있는 통합된 구조 설계 S/W 플랫폼을 개발하였다. 기존의 응용 프로그램이 설계 플랫폼에서 데이터베이스를 기반으로 연동되도록 인터페이스를 설계하였으며, 구조설계 결과는 다시 데이터베이스에 축적되도록 하고, S/W 플랫폼을 기반으로 한 프로그램의 개발 예를 통하여 구조설계 S/W 플랫폼의 적용가능성을 제시하였다. 이 결과를 요약하면 다음과 같다.

1. 구조 설계와 관련된 응용프로그램을 수행하고 그 결과를 축적할 수 있는 S/W를 작성할 수 있는 구조 설계 S/W 플랫폼을 개발하였다. 또한 구조물 설계 과정을 객체지향방법을 통해 분석하였으며, 설계자료의 저장과 설계 프로그램의 관계는 관계형 데이터베이스를 이용하여 개발하였다.
2. 기존 프로그램이 설계 플랫폼과 연결될 수 있도록 하기 위한 장치로서 플랫폼 스크립트 언어와 그 해석기를 개발하였다. 이 기존의 데이터베이스에서도 사용이 가능하도록 하기 위하여 데이터베이스 인터페이스는 표준 SQL 구문을 대상으로 설계하였다.
3. 설계 플랫폼을 기반으로 사용하여 개발할 프로그램의 규약을 결정하였다.
4. 설계 플랫폼 기반에서 구조물의 모델링과 해석 과정에 이용된 단위 응용프로그램을 개발하여

해석자료와 설계에서 이용하기 위한 설계정보 통합 환경을 개발한 결과로부터 기존 응용프로그램 간의 통합 과정의 문제점을 극복할 수 있음을 보였으며 설계 플랫폼 기반 응용프로그램을 개발하고자 하는 경우에 통합 설계 시스템의 개발 방법론으로 제시할 만한 결과로 사료된다.

참 고 문 헌

1. 김수진, 김재준, 이병해, 1993, 프리캐스트 콘크리트 설계에서 CAD와 객체지향 데이터베이스의 인터페이스 연구, 대한건축학회지 13권 2호.
2. 김영민, 1998, 건축 구조물 기하 위상 및 설계정보의 3차원 모델링, 서울대학교 대학원.
3. 김억, 조문상, 1998, 객체지향 건축물 모델링을 위한 데이터베이스 설계와 도형정보 입력 방법에 관한 연구, 대한건축학회지 14권 2호.
4. 농업진흥공사, 1989, 기술업무 프로그램 매뉴얼.
5. 안은경, 천진호, 신동철, 이병해, 1998, 철근 콘크리트 구조의 초기 구조설계 시스템 개발, 대한건축학회지 14권 4호.
6. 이창호, 1999, 구조설계 CAD 시스템 개발을 위한 설계정보간의 상호 관계의 표현, 대한건축학회지 15권 2호.
7. 장주흠, 1998, 구조물 설계에 있어서 데이터베이스 관리 시스템의 응용에 관한 연구, 서울대학교 대학원.
8. 조문상, 1997, 통합데이터베이스 구축의 실제, 건축 9호.
9. 천진호, 정윤철, 이병해, 1999, 통합 구조설계 시스템 구축을 위한 중앙 데이터베이스 모델 제시와 구현, 대한건축학회지 15권 7호.
10. 최영근, 허계범, 1995, 객체지향 소프트웨어 공학, 한국 실리콘.
11. 천진호, 김홍구, 이병해, 1993, 객체지향 데이터베이스를 도입한 통합 구조설계 시스템의 구축, 대한건축학회지 9권 9호.
12. 홍성목, 고일두, 허명재, 김치경, 1992, 객체지향설계법에 의한 건축구조 통합시스템 개발에 관한 연구, 대한건축학회지 8권 11호.
13. Sheloney Moni, Donald W. White, 1996, FRAME VIEW: Object-Oriented Visualization System for Frame Analysis, Journal of Computing in Civil Engineering.
14. The PostgreSQL Development Team, 1998, PostgreSQL Administrators Guide.
15. Abdalla, J. A. Object-oriented principles and techniques for computer integrated design, Ph.D. Dissertation, University of California at Berkeley, Calif., 1989.
16. Richard Sause, A Design Process Model for Computer Integrated Structural Engineering, Engineering with Computers, Vol.6, pp 129-143, 1990.