

STEP을 이용한 유한요소해석 정보모델 구축

최 영*, 조성욱*, 권기억**

Information Modeling for Finite Element Analysis Using STEP

Young Choi*, Seong Wook Cho* and Ki Eak Kwon**

ABSTRACT

Finite element analysis is very important in the design and analysis of mechanical engineering. The process of FEA encompasses shape modeling, mesh generation, matrix solving and post-processing. Some of these processes can be tightly integrated with the current software architectures and data sharing mode. However, complete integration of all the FEA process itself and the integration to the manufacturing processes is almost impossible in the current practice. The barriers to this problem are inconsistent data format and the enterprise-wise software integration technology. In this research, the information model based on STEP AP209 was chosen for handling finite element analysis data. The international standard for the FEA data can bridge the gap between design, analysis and manufacturing processes. The STEP-based FEA system can be further tightly integrated to the distributed software and database environment using CORBA technology. The prototype FEA system DICESS is implemented to verify the proposed concepts.

Key words : Finite Element Analysis, STEP

1. 서 론

기계공학의 설계해석 분야에서 유한요소해석 방법의 사용은 매우 중요한 비중을 차지하고 있다. 따라서 대부분의 상용 CAD/CAM/CAE 통합 시스템에서는 유한요소해석 패키지를 시스템의 일부로 제공하고 있다. 이러한 시스템에서는 제품 모델링 기능, 자동 격자 생성을 포함하는 전처리 기능, 해석 기능, 후처리 기능 등이 유기적으로 통합되어 있어서 기능별로 특성이 있는 다른 소프트웨어를 사용하고자 할 때에는 소프트웨어 간에 데이터를 변환해야 하는 어려움이 있다. 이와는 달리 우수한 전처리 기능을 가진 소프트웨어는 전처리 기능만을 위해서 사용되기도 하는데 이러한 소프트웨어는 여러 가지 해석을 위한 데이터 생성기능을 가져야 한다. 이와 같이 유한요소해석에 필요한 여러 가지 소프트웨어 구성 요소들이 각각 다른 데이터 형식을 취하고 있으므로

소프트웨어간에 데이터 호환성이 문제가 되어 원하는 구성 요소들을 사용하지 못하게 되는 일이 발생할 수도 있다. 이러한 데이터 공유 문제는 모델링 기능, 전처리 기능, 해석 기능, 후처리 기능 등을 포함하는 유한요소해석의 전과정에서 공히 사용할 수 있는 유한요소해석의 표준 정보모델을 구축함으로써 해결할 수 있다. 따라서, 모든 유한요소해석 소프트웨어 구성 요소들이 이러한 표준 정보모델에 기준하여 만들어지거나 최소한 표준 정보모델을 위한 인터페이스만 제공하면 데이터의 호환문제를 완전히 제거할 수 있게 될 것이다.

본 연구에서는 국제 표준 기구인 ISO에서 추진하고 있는 국제 제품 모델링 데이터 교환 표준인 STEP의 유한요소해석을 위하여 표준으로 제정중인 통합자원과 이들을 포함하는 응용프로토크인 AP 209의 표준 스키마들을 이해함으로써 STEP 공통자원과 응용자원을 결합한 유한요소해석 통합정보모델을 구성하고, 이를 기반으로 하는 유한요소해석 시스템을 통하여 시스템 통합의 타당성을 연구하고자 한다^[1].

*종신회원, 중앙대학교 기계설계학과

**학생회원, 중앙대학교 대학원 기계설계학과

국제 표준인 STEP을 이용하여 통합 시스템을 구축하면 다음과 같은 장점이 있다.

(1) STEP에서 개발한 공통자원을 그대로 이용할 수 있다. 예를 들어, 비다양체 데이터모델을 구성하기 위하여 Part 42의 기하학적 정보를 활용할 수 있다.

(2) EXPRESS 컴파일러, 에디터, SDAI 등과 같은 STEP 응용프로그램을 이용하여 시스템 개발을 신속히 할 수 있다.

(3) STEP을 지원하는 다른 소프트웨어와 직접 연결하여 데이터를 교환할 수 있다. 이러한 경우, 시스템이 특정 업체의 하드웨어나 소프트웨어에 종속되지 않기 때문에 오픈 시스템이 되어 투자가 보호된다.

2. 유한요소해석을 위한 STEP 표준

2.1 유한요소해석의 표준화

연속체를 유한요소법(FEM)으로 해석하기 위해서는 요소에 연결된 연속체상의 절점들로 구성된 모델로 이산화 하여야 한다. 요소는 관련된 물리적 성질과 재료의 물성치를 가지고 있다. 또한 좌표계, 집합체(절점 집합 혹은 요소 집합)들이 필요하며, 이산화된 격자 모델의 제어를 위한 자료가 필요하다. 하중 및 경계 조건, 출력 자료 제어 조건, 해석방법의 선택 정보 등은 이산화된 격자 모델과 함께 해석을 위한 완전한 입력 자료를 구성한다. 해석이 수행되면 해석결과는 절점상, 혹은 요소 내의 특정한 점에서 출력으로 주어진다. 결과 모델과는 관계없는 고유치, 혹은 총 변형 에너지 등의 출력이 주어지는 경우도 있다. 이와 같은 해석의 기본 형상과 해석 제어 등을 총괄하는 표준을 결정하면 모델링 시스템과 해석 시스템간의 원활한 데이터의 흐름이 가능하며 해석기의 설계, 개발 시에 표준 기능으로서 활용이 가능하다. 현재 이와 같은 유한요소해석의 표준은 STEP의 응용자원으로서, ISO Technical Committee 184(Industrial Automation Systems)의 Sub-committee 4-Working Group 3 Project 9(Product Modeling/Finite Element Analysis) 즉 ISO TC 184/SC4/WG3/P9에 의해 준비되어 오고 있다. 이 표준은 현재 Committee-Draft의 단계이며 Industrial Automation Systems, Exchange of Product Model Data Part 104: Application Resources: Finite Element Analysis라는 제목으로 개선이 진행되고 있다. 이 규격은 위에서 기술한 자료들, 즉 격자 모델에 관련된 자료, 관련된 해석 제어, 경계 조건, 해석 결과에 관한 정보 등에 관하여 기술한다^[2].

2.2 유한요소해석을 위한 STEP 데이터 표준

2.2.1 유한요소해석을 위한 STEP의 기본 스키마

(1) Structural response definition schema

이 스키마의 주제는 하나의 제품의 구조적 반응(structural response)을 기술하는 유한요소해석 모델에 대한 정의적 특징(definitional aspects)이다. 유한요소모델의 정의적 특징은 표현특징과 관계가 있고, 표현특징들은 structural_response_property와 연관된 property_definition_representation 에 의해 기술된다. 이 관계는 structural_response_representation_schema내의 structural_response_property_definition_representation 엔터티에 적용되는 법칙에 따른다.

(2) Structural response representation schema.

이 스키마의 주제는 한 제품에 대한 FEA 모델을 생성하기 위해 결합되는 일반화된 절점들, 요소들, 재료들과 요소 특성들이다. 기본 개념과 가정들은 유한요소해석 모델의 유일성, 제품과 유한요소모델의 관계, 2차원 및 3차원 유한요소모델에 대한 기하학적 구성(founding)과 정의, 식별자(identifiers), 좌표계, 요소 적분, 단위와 척도(measures), 재료 스키마와의 인터페이스, 모델에 대한 정의 표현간의 관계, 제어, 유한요소해석 결과들과 종속성(subtyping philosophy)과 확장성 등을 포함한다. 이 스키마에서는 제품 관계, FEA 모델 관계, 기하학적 설립과 해석 공간, 식별자, 좌표계, 요소 행렬 적분, 단위 와 척도, 재료 스키마와의 인터페이스, FEA 모델, 제어 및 결과 관계 등이 정의되어있다.

(3) FEA control and result schema

이 스키마의 주제는 유한요소해석을 수행하기 위해 필요한 제어 그리고 유한요소모델과 제어의 조합의 해석으로부터 나온 해석 결과이다. 이 스키마에서는 Control, Process, Result, State, Step, State definition, Units and measures 등이 정의되어있다.

(4) FEA scalar vector tensor schema

이 스키마의 주제는 유한요소해석의 입력과 결과를 나타내기 위한 scalars, vectors(1st order tensors), tensors에 대한 정의이다.

2.2.2 Application Protocol : AP209

이 응용 프로토콜은 컴퓨터를 이용한 해석이 가능한 형상과 이에 관계된 유한요소해석(FEA) 모델, 해석 결과, 물성치 등을 포함하여 정의된 복합(금속 구조물의 유한요소해석을 이용한 설계를 위해 필요한 정보와 범위를 정의한다. 다음 그림 1은 AP 209에서 지원하는 설계와 해석을 위한 공통 데이터 파일의 내용과 프로세스와의 관계를 나타낸다^[3].

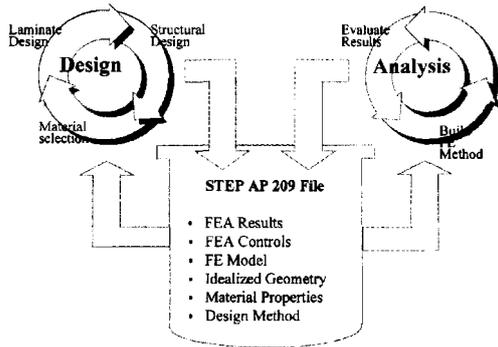


그림 1. AP 209 repository file.

본 연구에서는 유한요소해석을 위한 STEP 표준 스키마들에 기반을 둔 표준 정보모델을 구축하기 위하여 이들을 포함하고 있는 STEP의 응용 프로토콜인 AP 209를 이용하여 기본 정보모델을 구현하였다. 그러나 AP209의 설계 데이터를 위한 기본 스키마들에 대한 구현은 본 연구에서 다루지 않았다.

2.3 유한요소해석 시스템

유한요소해석은 다음 그림과 같이 구조물과 연속체를 해석하기 위한 수치적인 기법으로서 계산하기 어려운 복잡한 문제를 해석적으로 그 해를 근사적으로 구하기 위한 방법으로 오늘날에는 열 전달, 유체 유동, 전자기장 등 많은 영역의 문제를 다루고 있다^[4].

다음 그림 2는 유한요소해석을 위한 기본 입력 정보인 구조물에 작용하는 임의의 하중과 변위에 대한 경계조건 등과 절점과 요소에 대한 해석 결과와 해석을 위한 여러 조건을 나타내었다.

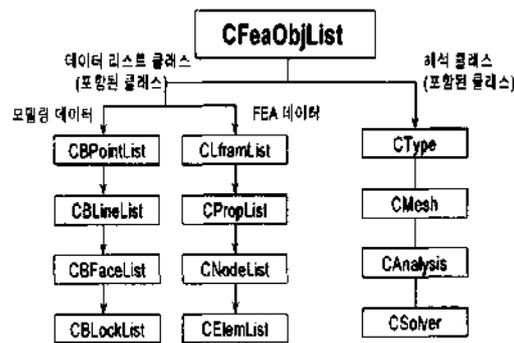


그림 2. 유한요소해석의 기본 정보 모델.

유한요소해석을 위한 기본 데이터 구조는 유한요소해석 표준을 위한 STEP의 기본 스키마를 참조하

였으며, 기본 데이터는 STEP의 객체 지향적인 구조를 위하여 C++로 정의하였다. 다음은 본 연구에서 정의된 기본 데이터 클래스를 나타낸다.

표 1. 정의된 기본 클래스

객체 이름	기능
CFeaObjList	데이터 교환 및 공유를 위한 객체
CType	요소 타입, 요소 형상 타입 재료 물성 타입, 요소 단면 타입 등을 정의할 위한 객체
CMesh	격자 생성을 위한 객체
CBPointList	모델링(임의의 점) 데이터를 위한 객체
CBLinelist	모델링(임의의 선) 데이터를 위한 객체
CBFacelist	모델링(임의의 면) 데이터를 위한 객체
CBLockList	모델링(임의의 블록) 데이터를 위한 객체
CPropList	재료 특성을 정의하기 위한 객체
CNodeList	
CElemList	유한요소해석을 위한 절점, 요소, 좌표계 객체
CLframlist	
CEntityList	STEP의 기본 엔티티를 정의하기 위한 기본 객체
CStepDesign	STEP 물리 파일과의 데이터 교환을 위한 객체
CAnalysis, CSolver	해석 및 해를 구하기 위한 객체

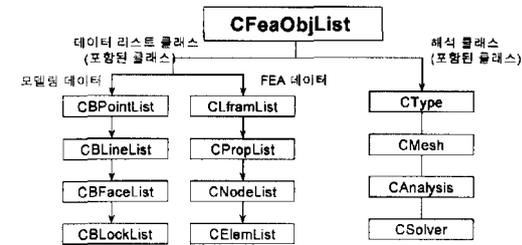


그림 3. 기본 데이터 클래스.

본 연구를 통해서 개발된 유한요소해석 시스템의 기본 데이터(모델링과 해석데이터)는 대부분 선형 리스트 데이터 구조로 이루어졌다.

1) 모델링 데이터

기본 모델링을 위한 데이터는 점, 선, 면, 2차원과 3차원 블록이 있다.

표 2. 모델링 데이터 클래스

Class Name	기본 속성(Attribute)
CBPoint	임의의 점의 X, Y, Z좌표
CBLinelist	선을 이루는 점(2개)의 포인터
CBFace	면을 이루는 선(4개)의 포인터
CBLock	블록을 이루는 면(2차원: 1면, 3차원: 6면)의 포인터

2) 자동 격자 생성

현재 여러 가지 이론에 기초하여 유한요소 격자를 생성하는 기술이 연구되고 있고, 각각의 이론과 방

법에는 장단점이 있다. 그러나 여러 가지 효율적인 측면을 고려하여 본 연구에서 사용된 격자 생성방법은 사상법에 기초한 방법이다. 이 방법은 먼저 해석 영역을 세 번이나 네 번으로 둘러싸인 여러 개의 영역으로 분할해야 한다. 각 영역에 대한 요소의 생성은 매개 변수 영역에서 생성된 요소를 사상함으로써 이루어진다. 본 연구에서는 위에서 정의한 기본 모델링 데이터로 정의된 2차원 혹은 3차원 블록들로 이루어진 임의의 구조물을 쿨스 패치를 위한 기본식을 이용하여 영역의 경계선을 기준으로 내부의 요소와 절점을 생성한다^[5].

표 3. 자동 격자 생성을 위한 클래스

Class Name	기본 속성(Attribute)
CMeshSize	Block의 격자 크기
CMesh	자동 격자 생성을 위한 Method가 정의

3) 해석 데이터

유한요소해석을 위한 기본 데이터는 앞에서 모든 이 절점과 절점으로 이루어진 요소와 이들 요소의 물성값과 각각의 절점과 요소에서의 결과값 등이 정의되었다.

표 4. 해석 데이터를 위한 클래스

Class Name	기본 속성(Attribute)
CAnalysis	문제의 해석을 위한 기본 정보 데이터를 정의
CSection	Thickness, Area, Moment of Inertia 등의 요소 상수를 정의
CMaterial	Young's modulus, Poisson's ratio 등의 요소 재료 특성을 정의
CLframe	절점 및 요소의 국부 좌표계를 정의
CNode	임의의 절점의 X, Y, Z 좌표와 절점의 좌표계 등이 정의
CElement	요소를 이루는 절점 리스트(요소 타입에 따라 그 개수가 결정)와 요소 특성(재료 불성 및 상수)값 등이 정의

4) 해

본 연구에서 각 절점에서의 변위를 계산하기 위해 사용된 방법은 Skyline Solver를 사용하였다^[4].

표 5. 해를 계산하기 위한 클래스

Class Name	기본 속성(Attribute)
CStatic	Global stiffness matrix, global force vector 등을 정의
CSolver	절점의 자유도에 대한 변위(displacement)

2.4 STEP 데이터 표준을 이용한 유한요소해석 시스템 개발

STEP 정보 모델링 데이터를 기반으로 하는 응용 프로그램의 구현을 위해 요구되는 기능은, EXPRESS 언어로 기술된 스키마를 해석하여 STEP 물리 파일에서 서술하고 있는 엔티티들의 데이터베이스를 구축/질의하는 기능과 정의된 유한요소해석 데이터를 이용하여 해석하는 기능으로 나눌 수 있다. 첫 번째 기능을 구현하기 위해서는 STEP Tools Inc.의 ST-Developer 1.5에 포함되어 있는 ROSE 컴파일러를 이용하여 선행 컴파일된 스키마와 AP209(복합 금속구조물의 유한요소해석을 이용한 설계) 스키마의 엔티티들을 C++ 클래스로 정의한 헤더 파일을 생성한 후 ROSE library를 이용하여 엔티티 데이터베이스에 접근한다. 두 번째 기능을 위해서 유한요소해석을 위한 기본 데이터를 설계하여 STEP데이터와의 인터페이스를 위한 모듈을 구현하고, 이를 통해 변환된 데이터를 이용한 유한요소해석을 수행하는 해석 모듈을 구현한다^[6, 7, 8]. 다음 그림 4는 STEP 데이터를 이용한 해석 과정을 나타낸다.

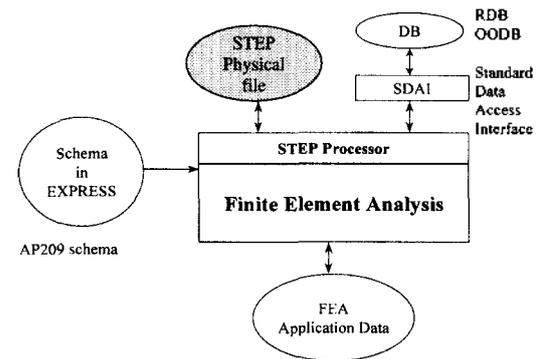


그림 4. STEP 데이터를 이용한 해석 과정.

다음 그림 5는 STEP을 이용한 유한요소해석 시스템의 개발 과정을 나타낸다. 해석을 위한 기본 데이터 설계와 유한요소해석 계산 모듈은 C++을 이용하여 개발하였고, 사용자와의 인터페이스와 해석된 결과의 디스플레이를 위해 Visual C++(MFC library)을 이용하여 최종적인 유한요소해석 시스템을 윈도우용 프로그램으로 개발하였다.

입력된 임의의 유한요소해석을 위한 STEP 물리파일을 실제 해석 모듈에서 사용하기 위해 앞에서 정의한 기본 데이터로 변환하여 해석을 수행한다. 다

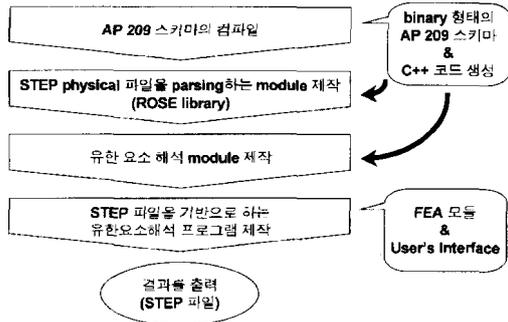


그림 5. STEP을 이용한 FEA 프로그램 개발 과정.

그림 6은 임의의 유한요소해석을 위한 STEP 데이터 파일을 나타낸다.

```

ISO10303-21
HEADER
  * Exchange file generated using STEP DEVELOPER v1.5.7
  FILE DESCRIPTION: "description of the implementation level 21.01"
  FILE NAME: "name of the design" (use string "198744-25T414 10.0101")
  * author: "CU" organization: "CU" processor: version: "STEP DEVELOPER v1.5.7" originating system:
  ...
  * Substation: ""
  * FEA SCHEMATIC STRUCTURAL ANALYSIS DEFINITION
  ENDSEC
DATA
  #10 = [FEA MODEL, 2D plane element test, 2.5, 1, 1, plane example, 1, static analysis, PLANAR]
  #20 = [CARTESIAN, POINT] (origin point) (0, 0, 0)
  #30 = [FEA AXIS PLACEMENT DEFINITION] #20 & CARTESIAN (global coordinate system)
  #40 = [CARTESIAN, POINT] coordinate set model (set 0)
  ...
  #2nd = [UNIT, model] #20, #25, #105
  ...
  #40 = [PLANE, VOLUME, 2D ELEMENT DEFINITION] LINEAR
  #50 = [VOLUME, 3D ELEMENT DEFINITION] TETRA, PLANE, STRESS
  #60 = [VOLUME, 3D ELEMENT DEFINITION] LINEAR, LAGRANGIAN, 3
  #70 = [FEA PARAMETRIC POINT] (2, 0.572, 0.572)
  #80 = [VOLUME, ELEMENT DEFINITION] #40, #60
  #90 = [VOLUME, POSITION, WEIGHT] #10, 1
  #100 = [FEA PARAMETRIC POINT] (0.572, 0.572)
  ENDSEC
END=ISO10303-21
    
```

그림 6. STEP 블리 파일의 예.

응용프로그램에 입력된 STEP 데이터는 메모리 상에 인스턴싱되고, ROSE 라이브러리에 의해서 Rose-Object의 인스턴스를 하나의 노드로 하는 그래프구조를 생성하게 된다. 데이터의 검색은 이러한 그래프를 순회함으로써 구현된다. 따라서 데이터의 검색을 통

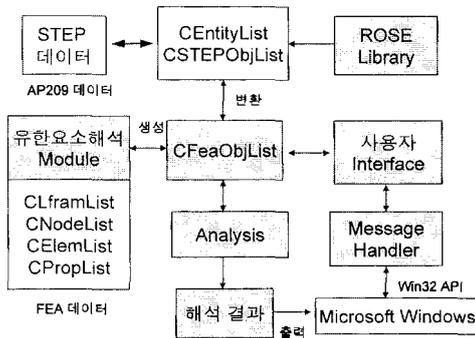


그림 7. 프로그램 모듈간의 상호작용.

해 필요한 정보모델 데이터를 얻고, 이를 해석을 위한 기본 데이터로 변환한 후 해석한다. 다음 그림 7은 프로그램의 실제 구현과정을 나타낸다¹⁾.

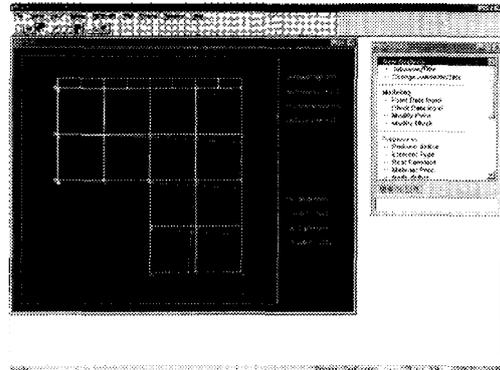


그림 8. 해석 예.

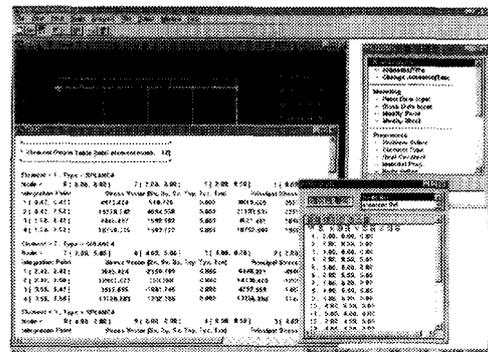


그림 9. 결과 출력 리스트 예.

다음의 그림 8, 9는 STEP 기반의 유한요소해석을 위한 프로그램의 실행화면을 나타낸다. 입력된 STEP파일을 통해서 데이터를 얻고, 얻어진 데이터를 이용하여 계산한 결과를 나타낸다.

2.5 STEP 데이터 표준을 이용한 유한요소해석 시스템의 응용

현재의 네트워크 환경은 여러 공급자들에 의해서 생산된 이기종의 시스템들로 이루어져 있다. 이러한 네트워크 시스템에서 정보를 공유하는 분산 시스템을 구축하기 위해서는 어플리케이션의 통합과 분산 프로세싱이 동시에 고려되어야 한다. 즉 응용프로그램에 네트워크 기능을 추가 시켜야 하고 네트워크 시스템간의 상호작용을 고려해야 한다. 그리고 제작된 응용프로그램은 앞서 밝힌 바와 같이 하드웨어

종류, 운영 체제, 네트워크 프로토콜과 어플리케이션 포맷에 무관하여야 한다.

CAD/CAE 시스템을 분산 객체 환경으로 구현하는 경우에도 마찬가지로 위와 같은 요구 사항이 적용되며 특히 설계부터 해석까지 일괄적인 작업이 가능하고, 작업자들이 효율적으로 업무를 수행할 수 있도록 프로젝트를 관리할 수 있어야 한다. 또한 분산 CAD/CAE 시스템을 구축하기 위해서는 네트워크 서비스, 관련 개발 도구와 기반 기술의 통합, 시스템 환경 구축을 위한 투자, 네트워크 어플리케이션 등이 필요하다. 이러한 요소들 중에서 네트워크 어플리케이션은 DB 서버와 각각의 응용 서버들로 이루어져 있으며 사용자가 STEP DB를 완전하게 제어하거나 사용하기 위해서는 DB 서버와 사용자간에, DB 서버를 관리하기 위한 작업들을 IDL로 정의하는 관리 프로토콜, STEP DB를 파트 모델들로 분할하는 추상 프로토콜, 클라이언트들이(SDAI를 이용하여) 네트워크를 통해 STEP DB를 접근할 수 있게 해주는 분산 프로토콜 등이 필요하다^[9, 10, 11, 12].

본 연구를 통하여 개발된 STEP 기반의 유한요소 해석 시스템을 분산 환경에 응용(DICCES: Distributed CAD/CAE Environment using STEP)함으로써 앞서 서론에서 기술한 여러 가지 기대 방안에 대한 타당성을 살펴보았다.

본 연구에서는 분산 프로토콜에서 SDAI를 사용하는 대신에 STEP Tools사의 ST-Developer에서 제공하는 ROSE를 사용하였다. 한편 부품 정보를 위해 STEP DB를 파트 모델들로 분할하는 추상 프로토콜은 본 연구에서 아직 구현되어 있지 않다^[13, 14, 15].

본 연구에서 개발된 DICCES의 개발 환경은 다음의 표 6에 나타나 있다.

표 6. DICCES의 개발 환경

H/W	Pentium PC
OS	MS Windows NT 4.0
STEP tool	ST-Developer 1.5
CORBA tool	IONA ORBIX 2.02
C++ compiler	MS Visual C++ 4.2
Network protocol	TCP/IP
Graphic library	OpenGL Library

2.5.1 DICCES의 구조

DICCES은 DB 서버와 응용 프로그램 서버를 사용자들이 사용하는 클라이언트 부분과 완전히 분리하고 이들 사이에 연결 기능을 제공하는 미들웨어, 즉 ORB를 위치시킴으로써 좀더 유연하고 확장 가능한

시스템을 구현하였다. DICCES는 STEP을 기반으로 한 DB 서버를 기점으로 하여 여러 종류의 응용 서버들(CAD 서버, CAE 서버 등)을 네트워크로 연결하여 데이터와 응용 시스템들을 공유하게 된다. 기본적으로 클라이언트는 모든 서버와 연결하여 작업할 수 있으며, 각각의 응용 서버들은 DB에 데이터를 요청할 수 있다. DICCES의 구조는 다음 그림 10에 표현되어 있다.

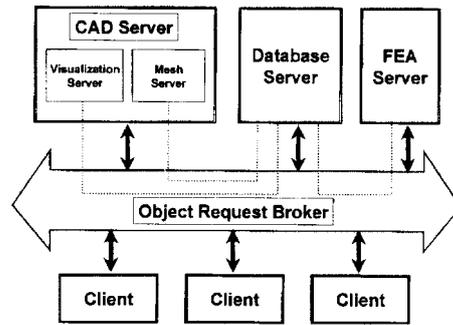


그림 10. DICCES의 구조.

2.5.2 개발 과정

전체적인 프로그램의 개발과정은 다음 그림 11에 설명되어 있다.

프로그램에서 사용된 각각의 서버들과 클라이언트의 구현과정은 다음과 같다^[16].

- (1) 클라이언트와 서버간의 인터페이스를 IDL로 작성한 뒤 컴파일하면 헤더 파일, 서버 와 클라이언트쪽 C++ 소스(IDL2C++) 파일이 생성된다.
- (2) 클라이언트 IDL2C++ 파일에 GUI를 위한 라이브러리를 추가하여 클라이언트 프로그램을 완성한다.
- (3) 각각의 서버를 위한 IDL2C++ 구현파일을 작성한다.
- (4) AP203과 AP209 EXPRESS 파일을 컴파일하여

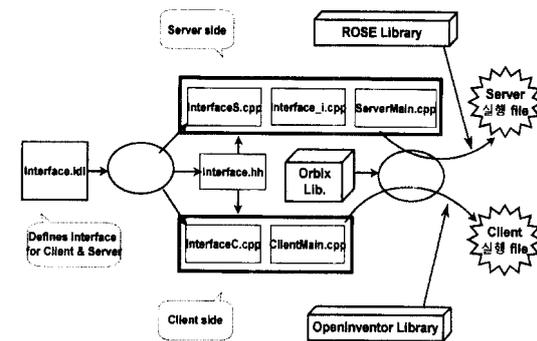


그림 11. DICCES 개발과정.

C++ 소스파일(EXPRESS2C++)을 생성한다.

(5) 서버쪽의 IDL2C++ 파일, IDL2C++ 구현파일, EXPRESS2C++ 파일에 ROSE라이브러리, Orbix 라이브러리를 추가하여 DB 서버를 완성한다.

(6) 응용 서버의 기능을 구현하기 위한 코드에 서버쪽의 IDL2C++ 파일, IDL2C++ 구현파일, Orbix 라이브러리를 추가하여 각각의 응용 서버를 완성한다.

(7) 완성된 응용 서버를 Orbix daemon에 등록시킨다.

2.5.3 서버와 클라이언트의 구현

본 연구에서 구현된 프로토타입 시스템은 DB 서버, 가시화(visualization) 서버, 메쉬(mesh) 서버, 해석(analysis) 서버 등으로 구성되어 있다.

(1) 데이터베이스 서버

STEP물리 파일에 기반을 둔 DB 서버는 프로젝트를 관리하고, 프로젝트 목록을 클라이언트에 전송하며, 작업 대상 파일의 위치(로컬 머신 + 로컬 경로)를 각 응용 서버에 전달하는 역할을 한다.

(2) 가시화 서버

가시화 서버는 DB 서버로부터 모델링 데이터를 얻은 후 이 데이터를 클라이언트에서 가시화하기 위한 데이터로 변환시킨 후 클라이언트에 전송하는 기능을 한다. 경계 표현법으로 정의된 솔리드 데이터중의 페이스와 그 경계를 검색해서 페이스는 NURBS 곡면으로 변환하고, 페이스의 경계를 표현하는 형상은 NURBS 곡면상의 파라메트릭 도메인에서 표현되는 트리밍 곡선으로 변환하는 과정을 거쳐 페이스 단위의 중립적 데이터를 얻어낸다. 결과적으로 가시화 서버는 STEP물리 파일의 형상 데이터가 표현하고 있는 여러 종류의 페이스들이 클라이언트에서 가시화 될 수 있도록 IDL로 정의한 인터페이스의 정의에 의한 트림 NURBS 곡면의 리스트로 변환시킨 데이터를 클라이언트에 넘겨주는 서버이다.

(3) 메쉬 서버

메쉬 서버는 DB 서버로부터 유한 요소로 분할하기 위한 모델링 데이터를 얻은 후, 클라이언트로부터 입력된 메쉬 사이즈 정보를 통하여 요소를 생성한다. 본 연구에서는 사상법을 사용하여 임의의 네 변으로 이루어진 해석 영역을 매개 변수 영역으로 사상함으로써, 사변형의 유한 요소를 생성하는 방법을 사용하였다.

(4) 해석 서버

유한 요소 해석을 위한 해석 서버는 DB 서버로부터 분할된 유한 요소 데이터를 얻고 클라이언트로부터 입력된 하중 및 변위 등의 경계 조건에 대한 정보

를 사용하여 해를 구한다. 계산된 결과는 클라이언트의 요구에 의하여 DB 서버에 저장되거나 클라이언트에 다시 전송되어 보여진다.

(5) 클라이언트

서버와 클라이언트와의 통신은 IDL로 정의된 작업들로 이루어지며 클라이언트 프로그램은 각 인터페이스의 작업들에 해당하는 사용자 인터페이스를 제공한다. 클라이언트 프로그램은 DB 관리, 가시화, 메쉬, 해석을 위한 기능을 모두 통합할 수도 있고 일부 기능만을 제공하는 형태로도 제공할 수 있다. 본 연구에서 클라이언트 프로그램은 작업 파일을 선택할 수 있도록 프로젝트 목록 윈도우를 제공하며, 선택한 파일에 대한 가시화를 위하여 Open Inventor를 이용한 디스플레이 윈도우를 제공하며, 유한 요소 생성 및 해석을 수행하기 위한 요소 생성 조건과 해석 조건 등을 주기 위한 사용자 입력 윈도우가 제공된다. 또한 해석 결과를 출력하기 위한 텍스트 기반의 해석 결과 윈도우도 제공한다. 클라이언트는 각 응용 서버와 연결하여 사용자의 입력 정보를 연결된 응용 서버에 전달하는 역할과 응용 서버로부터의 메시지나 결과를 사용자에게 보여주는 역할을 한다. 이를 위해 보다 편리하고 명확한 사용자 인터페이스를 제공하여 이를 통해 입력된 정보를 응용 서버에서 사용되는 데이터로 변환시키는 기능과 연결된 서버로부터 전송된 데이터를 해석하여 디스플레이 해주는 기능을 갖는다.

2.5.4 클라이언트의 작업 시나리오

클라이언트가 서버에 연결하여 작업을 하는 순서는 다음과 같다.

(1) DB 서버에 연결하면 DB 서버가 작업 대상 STEP 물리 파일(AP203, AP209, ...) 리스트를 클라이언트에 보내준다. 이 리스트에서 작업을 해야하는 STEP 물리 파일을 선택한다.

(2) 응용 서버에 연결하여 선택한 STEP물리 파일의 이름을 넘겨주면, 응용 서버는 DB 서버에 연결하여 DB 서버의 메모리에 선택된 STEP물리 파일을 인스턴스한다.

(3) 클라이언트가 응용 서버에 가시화, 메쉬 생성, 해석 등의 작업을 요구하면 응용 서버는 DB서버에서 작업에 필요한 엔티티의 인스턴스 리스트를 요청하여 클라이언트가 요구한 작업을 수행한다.

(4) 응용 서버의 계산이 끝나면, 클라이언트는 응용 서버에서 결과를 받을 수 있다.

(5) 모든 작업이 성공적으로 수행되어 응용 서버에 저장 명령을 내리게 되면 응용 서버는 변경된 내용

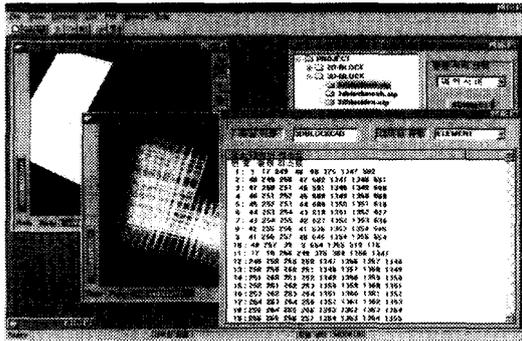


그림 12. DICCES의 클라이언트.

을 DB 서버에 넘겨준다. DB 서버에서는 작업 대상인 STEP 물리 파일에 변경된 내용을 갱신시켜서 저장하고, 파일 리스트를 업데이트시킨다.

여기서 주목할 점은 서버와 클라이언트간의 연결이 직접 이루어지지 않고 ORB를 통하여 이루어진다는 것이다. 즉, 클라이언트는 서버의 위치나 종류에 상관없이 ORB에 서비스를 요청하기만 하면 된다.

다음 그림 12은 DICCES환경에서 클라이언트가 구현된 에이머 우측 상단에 위치한 다이얼로그 박스 윈도우는 DB 서버에 연결하여 사용자가 원하는 STEP 물리 파일을 선택할 수 있도록 한 것이다. 그림 12의 좌측 상단에 위치한 윈도우로부터 반시계 방향으로 각각 가지화 서버에서 전송된 모델링 데이터를 보여주는 윈도우, 매쉬 서버에서 전송된 매쉬된 모델링 데이터를 보여주는 윈도우, 해석 서버에서 전송된 해석 결과를 텍스트로 보여주는 윈도우를 보여 주고 있다.

여기서 중요한 점은 STEP을 기반으로 하는 새로운 응용 서버를 구현하여 기능을 확장하는 것이 가능하다는 것이다. 즉, STEP에 등록되어 있는 모든 AP에 대한 응용 서버를 개발하여 DICCES에 추가하는 것이 가능하므로 CAD/CAE 분야 이외의 응용 분야와의 인터페이스가 가능하게 될 것이다.

3. 결 과

본 연구에서는 유한요소해석을 위한 응용프로토콜 개발을 위하여 STEP 공통자원과 응용자원을 결합한 유한요소해석 통합 정보모델을 구성하고 이를 이용하여 유한요소해석 시스템 개발을 통해 구축된 표준 정보모델의 타당성을 검증하였다. 이를 위하여 유한요소해석을 위한 STEP의 표준 스키마들을 이해하고 각각의 스키마에서 정의된 엔티티를 이용한 기본 정

정보모델의 구성과 이를 기반으로 하는 유한요소해석 시스템을 개발하였고 이를 통하여 구축된 표준 정보 모델에 대한 타당성을 연구하였다. 또한 개발된 유한요소해석 시스템을 이용하여 STEP 기반의 분산 CAD/CAE 시스템(DICCES)을 구현하여 STEP 기반의 통합 시스템의 활용 가능성을 모색해 보았다. 이러한 통합 시스템들로부터 생성된 STEP 파일 또는 데이터 베이스 시스템에 저장된 통합 유한요소해석 정보는 데이터의 변환 없이 여러 종류의 서로 다른 응용프로그램에 의하여 이용될 수 있다. STEP에 기반을 둔 통합 정보시스템 구축은 향후 STEP의 개발이 완료되고 주요 CAD/CAM/CAE 또는 소프트웨어 벤더들이 이를 지원할 때 그 효과가 극대화 될 것이다.

후 기

본 연구는 1996년도 교육부 학술 연구 조성비(기 계공학 ME96-E-30)에 의하여 지원되었습니다.

참고문헌

1. STEP 연구회, 제품 모델 정보 교환을 위한 국제 표준(ISO 10303) 스텝, 성안당, 1996.
2. ISO 10303-104-Part 104: Integrated Resources: FEA Reference Model, ISO, 1991.
3. ISO 10303-209-Part 209: Application Protocol: Composite and metallic structural analysis and related design, ISO, 1996.
4. Bathe, K.J., *Finite element procedures*, Prentice Hall, 1996.
5. 이원양, CAE 시스템을 위한 격자 생성기 및 열해석 모듈의 개발, 석사 학위 논문, 중앙대학교, 1997.
6. ISO 10303-11-Part 11: Description Method: EXPRESS language reference manual, ISO, 1994.
7. ISO 10303-22-Part 22: Implementation Method: Product Data Representation and Exchange, NIST, 1994.
8. *STEP Programmer's toolkit-ROSE library reference manual*, STEP Tools Inc., 1996.
9. Marache, M. et al., "A CORBA-based infrastructure managing STEP distributed models for virtual reality applications", *Proceedings of European Conference on Product Data Technology Days 1997*, pp. 119-128, 1997.
10. *The Common Object Request Broker: Architecture and Specification 2.0*, OMG, 1995.
11. Siegel, J., *CORBA fundamentals and programming*, John Wiley & Sons, Inc., 1996.
12. 박재현, "분산객체 기술을 꿈꾸며, CORBA", 마이 크로 소프트웨어, 1996년 3월호~1996년 6월호.

13. Hardwick, M. *et al.*, "Data protocols for the industrial virtual enterprise", *Internet Computing*, pp. 20-29, January-February 1997.
14. Willian, R., "Internet enabled computer aided design", *Internet Computing*, pp. 39-50, January-February 1997.
15. Han, J. and Requicha, A.A.G., "Modeler independent feature recognition in a distributed environment", *Network-centric CAD, special issue of Computer Aided Design*.
16. *Orbix 2 programming & reference guide*, IONA Ltd. 1995.



최 영

1979년 서울대학교 기계설계학과 학사
 1981년 한국과학기술원 생산공학과 석사
 1989년 Carnegie Mellon University 기계공학과 박사
 1981년 ~ 1984년 대우중공업 중앙연구소 연구원
 1989년 ~ 1990년 Engineering Design Research Center 연구원
 1992년 ~ 현재 중앙대학교 기계설계학과 교수
 관심분야 : 네트워크 CAD, Non-manifold modeling, 솔리드모델링, 곡선 및 곡면 모델링



조 성 욱

1979년 서울대학교 기계공학과 학사
 1981년 한국과학기술원 기계공학과 석사
 1990년 M.I.T. 기계공학과 박사
 1981년 ~ 1984년 대우중공업 중앙연구소 연구원
 1991년 ~ 1993년 포항공과대학교 기계공학과 교수
 1993년 ~ 현재 중앙대학교 기계설계학과 교수
 관심분야 : 전산고체역학, 유한요소해석법



권 기 억

1996년 중앙대학교 기계설계학과 학사
 1998년 중앙대학교 기계설계학과 석사
 관심분야 : 전산고체역학, 유한요소해석법