

◆특집◆ 웹기반 엔지니어링 협업 기술

분산 협업 환경 기반 엔지니어링 해석 서비스 시스템 개발

김기일*, 권기억**, 박지형***, 조성욱*

Web-based CAE Service System for Distributive Collaborative Engineering Environment

Ki Il Kim*, Ki Eak Kwon**, Ji Hyung Park*** and Seong Wook Cho*

Key Words : Finite Element Analysis(유한요소해석), Web Services(웹 서비스), .NET Framework(닷넷 프레임 워크), SOAP(soap 프로토콜), XML(확장 마크업 언어), Distributed(분산), Collaborative(협업)

1. 서론

인터넷과 웹 기술의 보급은 데이터의 공유와 글로벌 협력을 위한 새로운 소프트웨어 도구의 출현을 가능하게 하고 있으며 1990 년대 중반에 개념이 형성된 Web-enabled Engineering 은 다수의 개발자나 사용자가 단지 웹 브라우저와 같은 최소한의 클라이언트로 제품의 개발에 참여하거나 정보를 공유할 수 있다는 이점으로 인하여, 짧은 기간 동안 급속한 발전을 거듭해 왔다. 이러한 초고속 인터넷 환경은 기업간 또는 기업 내 부서간 설계 문서와 설계데이터 및 생산 정보 등의 관리를 네트워크 환경에서 통합하고 공유 함으로서 분산/협업을 가능하게 하고 있다. 또한 웹 브라우저와 멀티미디어를 이용하여 다중 사용자간에 정보를 공유함으로써 다양한 형태의 공동협업을 가능하게 하고 있다.

네트워크를 통한 분업/협업 환경은 각각의 부서나 기업에서 보다 전문화되고 세분화된 작업을 동시에 수행할 수 있도록 함으로써 새로운 제품에 대한 개발 비용과 시간을 절감할 수 있다. 그러나

이러한 환경에서는 각각의 부서나 기업에서 사용하는 시스템의 이질성이나 제품 데이터 표현 방법의 차이로 인해 이를 통합하기 위한 부가적인 개발 비용과 시간이 발생하기도 한다. 따라서 생산 과정에서의 상호협력 작업을 성공적으로 수행하기 위해서는 제품 개발 초기 단계에서부터 분산된 환경에서의 관련 시스템들을 유기적으로 통합 관리할 수 있어야 하며 각 사이트간 의사 교환이 원활하게 이루어져야 한다. 이를 통하여 분산된 다수의 제품 개발자들이 이질적이고 지역적으로 분산된 컴퓨팅 환경에서 상이한 업무를 독립적으로 수행하면서 협력을 통한 제품의 공동 개발이 가능해질 것이다.

본 연구에서는 이기종의 시스템들로 이루어진 인터넷 환경에서 서로 다른 CAD/CAE 시스템들간의 정보 공유 및 상호 작용 등의 공동 협력 작업을 수행하는 웹 기반의 분산/협업 엔지니어링 시스템에 관한 연구를 수행하고 해석과정의 분산 병렬화를 적용하여 분산/협업 시스템에서의 보다 효율적인 유한요소해석을 수행할 수 있는 시스템을 구현하였다.

2. 분산/협업 환경을 위한 기반 기술

2.1 웹 서비스를 위한 표준 기술

웹 서비스는 SOAP 을 이용하여 원격지에 있는

* 중앙대학교 기계공학부
Tel. 02-820-5313, Fax. 02-816-4972
Email scho@cau.ac.kr

** B2Gen

*** 한국과학기술연구원

메소드를 호출할 수 있는 새로운 방법으로서 하나의 사이트에서 모든 서비스나 내용을 받던 체계에서, 사이트별로 전문화된 단위기능을 받아서, 하나의 통합된 서비스를 제공한다.

이는 기존의 인터넷 기반을 활용하여 단순 데이터만을 제공하는 시대에서 데이터를 이용한 서비스 단위를 제공하는 체계로 바뀜을 의미한다. 이는 기존의 웹 시스템에서 제공되는 프레임워크와 플랫폼이 단일 사이트를 구축하기 위한 것이라고 한다면, 웹 서비스에서의 프레임워크와 플랫폼이 다수의 사이트를 서비스 단위로 하나로 묶기 위한 것이라 할 수 있다.

웹 서비스는 다음과 같은 3 가지 구성요소로 이루어진다.

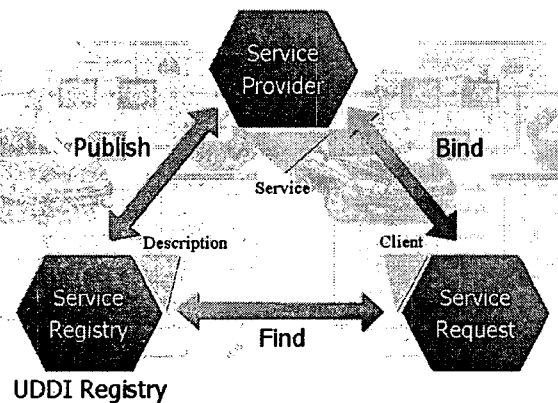


Fig. 1 Web Services Roles, Operations, Artifacts

Service Requester: 원하는 서비스를 서비스 등록자에서 검색을 하여 찾은 후, 그 정보를 토대로 해당 서비스 제공자에 접속하여 서비스를 받는 체계, 서비스 제공자를 알고 있는 경우는 서비스 등록자를 통하지 않고 직접 서비스를 받을 수도 있다.

Service Registry: 서비스 등록자는 서비스 제공자가 등록한 서비스 내용 및 서비스 제공자 소개를 분류하여 서비스 요청자가 요청 시 필요한 정보를 제공한다.

Service Provider: 자신의 서비스 내용을 서비스 등록자에게 등록한다

2.1.1 XML

기존의 HTML 태그의 한계점과 SGML 문법의 복잡성 등을 개선하여 W3C 에서 제안한 표준 언

어이다. XML 문서는 다양한 태그를 정의하여 구조적인 데이터를 나타낼 수 있기 때문에, 데이터 간의 복잡한 관계를 나타내기에 적합하다.

HTML 은 데이터와 뷰(view)를 하나로 표현하지만, XML 은 데이터와 뷰가 분리되어 있다. XML 의 데이터를 웹으로 보여주기 위해서는 스타일시트(Style Sheet)가 필요하다. XSL(XML Style Sheet Language) 은 XML 의 스타일시트 언어로 크게 두 가지 역할을 한다. 첫째는 XSLT(XSL Transformations) 라고 부르는 것으로 XML 문서를 디자이너가 고안한 다른 문서로 변환하는 일을 처리한다. 둘째는 XSL Formatting 으로 첫 번째의 변환(transformation) 결과로 나온 문서를 음성, 동영상, 프린트등과 같은 각종 미디어 포맷으로 바꿔 주는 역할을 한다.

2.1.2 SOAP

SOAP 은 분산 환경에서 XML 데이터의 메시지를 위한 가볍고 간단한 프로토콜이다. SOAP 은 아래와 같이 크게 세 부분으로 나뉜다.

SOAP envelope: 메시지에 무엇이 있는지, 어떻게 처리할지 대해 정의하는 프레임워크

SOAP encoding rules: 어플리케이션에서 정의된 데이터 타입들을 표현

SOAP RPC: SOAP 메시지의 RPC(Remote Procedure Call)

SOAP 은 XML 형태의 메시지를 HTTP 프로토콜이나 다른 프로토콜(FTP, SMTP 등)을 사용하여 전달할 수 있는 인터넷 기반의 프로토콜이다. 특히 HTTP 와 같은 인터넷 기반의 프로토콜을 이용하여 RPC 기능까지 제공하기 때문에 특별히 ORB 객체를 요구하는 CORBA 의 RPC 에 비해 융통성이 있다. 인터넷 기반의 프로토콜을 기반으로 특별한 제약 없이 메시지를 보낼 수 있다는 장점을 이용하여, XML 형태인 SOAP RPC 는 메시지를 분석 할 수 있는 파서만 제공 된다면 개발 언어와 오브젝트 모델, 운영 체제, 플랫폼에 상관없이 메시징이나 RPC 가 사용 가능하다.

2.2 웹 서비스 구현 환경

본 연구에서 구현된 웹 서비스는 .Net Framework 환경에서 C#을 이용하여 구현하였으며

쳐지는 경우에만 데이터 교환이 국부적으로 일어나도록 함으로써 병렬성을 다른 직접적 해법에 비해서 월등히 향상시킬 수 있다.⁹

프론트 해법은 요소 강성행렬을 조립해 나가는 과정에서 조립이 완료된 자유도를 그 즉시 소거해 나가는 방법이다. 특히 이 방법은 유한요소 해석 과정에서 나타나는 행렬의 구조적 특성을 충분히 살려 계산량을 크게 줄일 수 있을 뿐만 아니라 전역 강성행렬을 조립하지 않고 소거된 자유도를 보조 기억장치에 저장함으로써 필요로 하는 주 기억장치의 용량을 크게 줄일 수도 있다.

다음 Fig. 3 는 유한요소들을 조립해 나가는 과정에서 이후에 조립될 요소들의 자유도에 전혀 영향을 미치지 않는 자유도를 소거해가는 과정을 보여주고 있다.

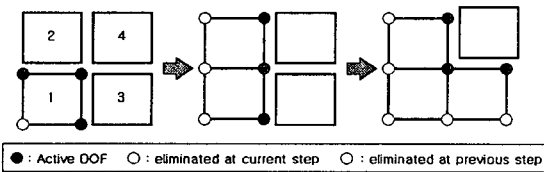


Fig. 3 Successive elimination in the frontal method

다중 프론트 해법은 분할된 영역에서의 각각의 프론트를 다른 영역에서 퍼져나가도록 해서 이들 프론트를 다시 합쳐나가는 방법을 사용한다. 이 과정에서 각각의 영역에서의 소거과정은 각각 다른 프로세서를 이용하여 계산하게 된다. 다음 Fig. 4 는 이러한 해석과정을 보여주고 있다.

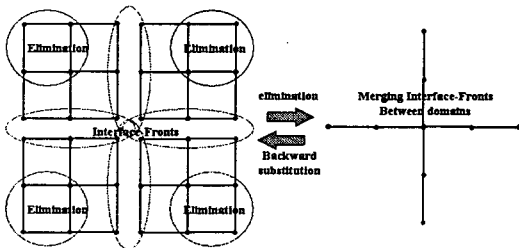


Fig. 4 Analysis procedure in Multi Frontal Method

이러한 병렬 해석시스템에서 각각의 프로세스들 간에 계산량의 균등한 분배를 위해 해석할 영역(유한요소)에 대한 적절한 분할이 필요하고, 또한 각각의 프로세서들 간의 통신을 최소화하기 위한 방법으로서 각 영역 사이에 공유되는 인접한

요소(절점)들을 최소화하는 것이 필요하다.

본 연구에서 보다 효과적인 영역분할을 위하여 Metis¹⁰ 를 이용한 다단계 그래프 분할 알고리즘 기법¹¹을 이용하여 다중프론트 해법을 위한 프론트 분할을 구현 하였다.¹² 다음 Fig. 5 에 다단계 그래프 분할기법을 이용한 영역분할의 예를 나타내고 있다.

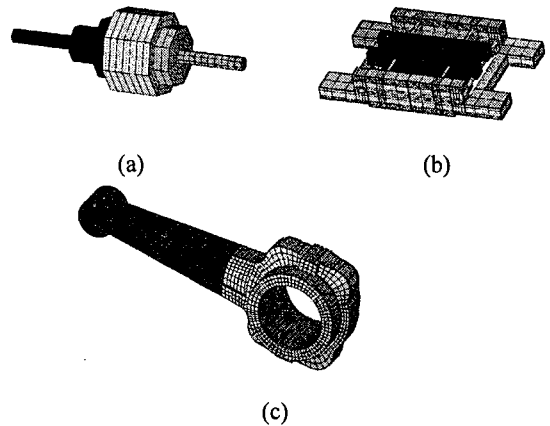


Fig. 5 Domain decomposition by graph theory (a) 3564 DOF, (b) 19968 DOF, and (c) 38331 DOF

위 그림은 각각의 모델을 3 개의 영역으로 분할한 상태를 보여주고 있다. 분할된 각각의 영역은 서로 다른 프로세서를 통해서 계산 하게 된다.

3.2 해석 시스템의 분산 병렬화

본 연구에서는 다중 프론트 해법과 다단계 그래프 분할 기법을 이용한 유한요소법의 분산 병렬화를 구현하였다. 개발된 분산 해석시스템에 대해서 Fig. 5 의 모델을 대상으로 서로 다른 사양의 컴퓨터를 이용하여 성능비교를 함으로써 타당성을 살펴보았다. 3 개의 Frontal Server 로 구성된 분산 환경하에서 예제 해석을 수행하였다

성능 비교 결과, Fig. 6 에 보인 바와 같이 하나의 컴퓨터 상에서 해석을 수행하는 일반적인 단일 프로세스에 의한 해석시간보다 분산 병렬 시스템에 의한 해석시간이 단축되는 것을 확인할 수 있다. 분산 병렬 시스템을 이용할 경우, 영역분할을 통해 생성된 데이터들을 네트워크를 통해 각각의 Frontal 서버로 전송하는 과정의 시간을 포함하는 전체 시간도 역시 단일 프로세스에 의한 해석 시간보다 현저히 단축되는 것을 확인할 수 있다.

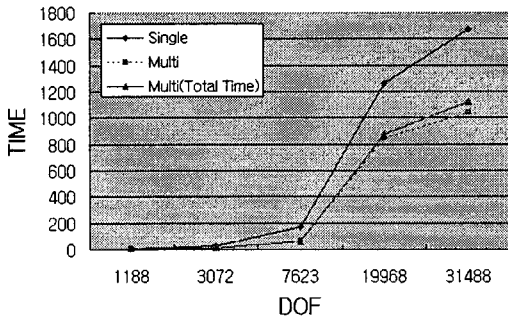


Fig. 6 Performance comparison for DOF increment

3.3 해석 가시화 시스템

본 연구에서 구현된 뷰어는 와이어 프레임을 보여 주거나 삼각형 데이터를 받아 셰이딩 해서 보여주는 기능을 한다. 또한 마우스 메시지에 반응하여 화면을 확대, 축소하거나 회전, 이동이 가능하며 이러한 뷰 변경 메시지를 서버에 전달하고, 전달 받는 기능을 가지는데, 뷰 변경시 화면의 크기나 중형비가 다른 사용자들 간에도 일관된 뷰를 공유할 수 있어야 하므로 정규화된 뷰포트 데이터와 회전 변환 행렬을 전송하거나 받게 된다. 사용자는 모델이 나타나 있는 창에서 마우스를 드래그함으로써 모델의 확대/축소, 회전, 이동 등 뷰 변경을 할 수 있다. 클라이언트 모델 뷰어는 서버에 삼각형 페이스트 데이터를 요구하게 되고 서버는 솔리드 모델의 가시화 과정을 거쳐 삼각형 데이터로 만들어 보내주게 된다. 삼각형의 생성과 전송 시에도 컨퍼런스는 지장 없이 진행 되어야 하므로 서버는 이 일을 멀티스레드를 통해 해결 한다.

3.4 해석 컨퍼런스 시스템

본 논문에서 구성된 시스템 환경은 서버, 클라이언트, 데이터 베이스의 3 개의 층 구조를 가지고 있다. 시스템의 클라이언트는 C#으로 구현되어 웹상의 컴포넌트로 사용자에게 제공되며 각각의 서버는 필요한 데이터를 STEP 으로 기술된 제품 데이터(DB 서버)에 접근한다.

사용자들이 발생시키는 메시지와 이벤트는 .Net Remitting 을 통하여 서버에 전달되고, 서버는 이러한 메시지를 또 다른 사용자들에게 전달함으로써 컨퍼런스를 가능하게 한다. 이때 사용자들은 컨퍼런스의 대상이 되는 데이터의 3 차원 형상 및 해석 결과들을 공유하며 채팅 기능을 이용하여

서로의 의견을 공유할 수 있다. 또한 각각의 응용 서버들은 데이터베이스나 기타 설계 및 해석을 위한 모듈 또는 객체들과 연결되어 사용자들의 요구에 응답하고, 때로는 능동적으로 사용자에게 데이터와 메시지를 전달하게 된다.

4. 멀티 프론탈 유한요소법을 이용한 엔지니어링 해석 서비스 시스템 설계

4.1 구현 환경 및 개발도구

네트워크상에서 작동하는 어플리케이션이 아닌, 독립적으로 실행되는 어플리케이션을 개발한다 할지라도 각각의 모듈간 인터페이스를 정의하는 것이 일반적이다. 모듈정의를 명확하게 이루어지면 이를 IDL 로 기술하기가 용이해 진다. 본 시스템의 IDL 은 해석을 위한 유한요소모델 및 결과 데이터를 위한 인터페이스를 중심으로 클라이언트와 서버가 상호 작용하게 된다.

작성된 IDL 은 서버 스켈리턴과 클라이언트 스텝으로 컴파일 되는데 이들은 각각의 구현을 위한 언어로 표현된다. 본 시스템에서 C++ 서버 스켈리턴을 생성하기 위해서 Iona Technology 사의 Orbix 3.0.1 의 IDL 컴파일러를 사용하고, 자바 클라이언트 스텝을 생성하기 위해서 같은 회사의 OrbixWeb 2.0.1 의 IDL 컴파일러를 사용하였다. 웹 서비스는 .NET Framework 환경에서 Visual C#을 이용하여 구현하였다. 내부 모델링 커널 및 해석을 위한 모듈은 Visual C++을 이용하여 구현하여 웹 서비스 구현 모듈에서 import 하였다.

4.2 시스템 구조

본 논문에서는 DB 서버와 응용 프로그램 서버를 사용자들이 사용하는 클라이언트 부분과 완전히 분리하고 좀더 유연하고 확장 가능한 시스템을 구현하였다. 본 시스템은 STEP 을 기반으로 한 DB 서버를 기점으로 하여 여러 종류의 응용 서버들을 네트워크로 연결하여 데이터와 응용 시스템을 공유하게 된다. 기본적으로 클라이언트는 모든 서버와 연결하여 작업할 수 있으며, 각각의 응용 서버들은 DB 에 데이터를 요청할 수 있다. 또한 해석 서버에서는 계산 과정의 분산 병렬화를 위한 분산 병렬 Solver 을 포함한 다양한 해석을 위한 해석시스템들이 구성되어 있다. 다음 Fig. 11 은 본 논문에서 구현된 분산 시스템을 보여주고 있다.

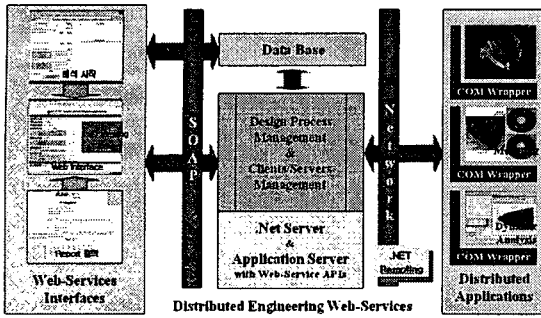


Fig. 7 Distributed FEA System Architecture

4.3 세션과 콜백 메커니즘

클라이언트는 세션에 참여함으로써 서버 객체에 오퍼레이션을 요구할 수 있으며 서버가 제공하는 대부분의 오퍼레이션은 세션을 통하여 제공된다. 또한 서버 측에서 클라이언트에 오퍼레이션을 요구할 경우 클라이언트가 그러한 요구에 대한 응답을 할 수 있어야 하는데, 이것을 가능하게 하는 것이 콜백(call-back) 메커니즘이다.

Fig. 12는 콜백 객체와 세션 오퍼레이션 및 데이터의 상호 작용을 나타낸 그림이다. 세션은 세션에 참여한 클라이언트의 콜백 객체를 참조하고 있으며 클라이언트는 자신이 참여한 세션 객체를 참조하고 있다. 그림에 나타난 proxy 객체들은 실제의 구현은 네트워크 건너편에 있는 객체이지만 로컬 플랫폼에 있는 것처럼 사용되며 실제 객체와의 인보케이션(invocation)은 사용자에 투명하게 일어난다.

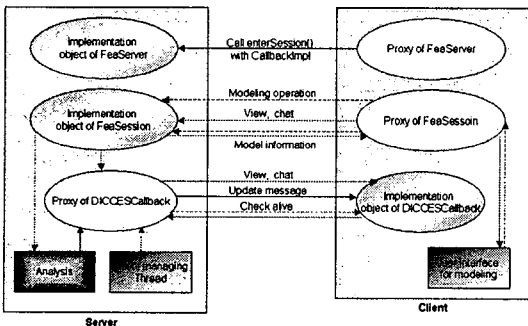


Fig. 8 Operations between session and callbacks

클라이언트는 세션 객체에 데이터나 질의 등의 서비스를 요구할 수 있고, 세션은 콜백 메커니즘을 통하여 클라이언트에 데이터를 전달한다.

4.4 해석 시스템의 분산/협업 작업 시나리오

본 논문에서 구현된 분산 해석시스템의 작업 순서는 다음과 같다.

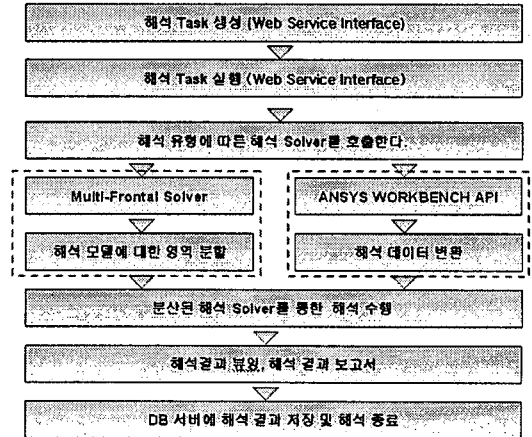


Fig. 9 Scenario of Distributed/Collaborative Analysis Process

- (1) 사용자는 웹서버를 통해 클라이언트 요소를 구성하는 컴포넌트를 다운을 받고 실행할 수 있다.
- (2) 사용자는 해석 서버에 접속하여 새로운 태스크를 시작하거나, 기존의 해석 태스크를 선택하여 작업을 수행한다.
- (3) 사용자는 원하는 해석유형에 따른 해석 서버를 선택한다.
- (4) Multi-Frontal Solver 에 의한 해석은 해석 모델에 대한 영역분할을 하고 이를 각각의 분산된 Frontal 서버를 통해 해석을 수행한다.
- (5) 사용자는 각각의 서버를 통해 해석 결과를 받게 되고, 이를 DB 서버에 저장한다.
- (6) 각각의 사용자는 해석 결과 및 모델 데이터를 DB 서버로부터 얻어, 이를 공유하면서 서로의 의견을 교환하면서 설계 및 해석 과정을 통해 발견된 문제를 해결할 수 있도록 한다.

5. 시스템 구현 및 실행예제

본 연구에서 구현된 웹 서비스는 전체 통합 시스템의 웹 페이지 상에 CAE 태스크 컴포넌트로 추가되어 동작한다. 다음은 실행 예를 보여준다.

1. 사용자가 CAE 태스크를 생성하거나 이미

생성되어진 CAE 태스크를 시작한다.

2. 해석을 수행할 해석 모델을 읽어 들인다.
3. 해석에 필요한 조건들을 입력하고 해석을 수행한다.
4. 해석 결과를 리스트 및 뷰어를 통하여 검토하고 저장한다.

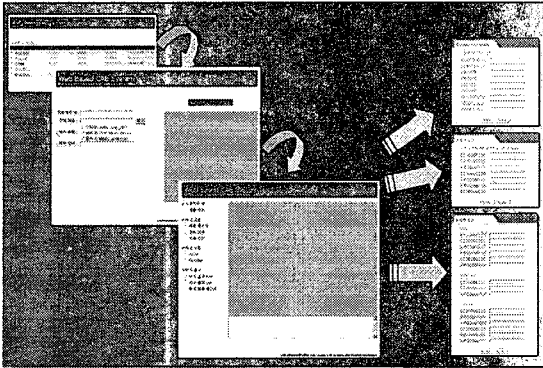


Fig. 10 CAE Service task scenario

5.1 태스크 목록 및 태스크 생성

전체 프로젝트가 생성되고 해석 태스크가 생성되면 사용자는 해석을 위한 태스크의 선택 및 생성이 가능해진다. 다음은 프로젝트에서 생성되어진 해석 태스크의 목록을 보여준다. 사용자는 새로운 해석 태스크를 생성 및 기존 태스크를 삭제 할 수 있으며 기존해석을 계속할 수 있다. 태스크 목록에서는 간단한 생성 정보 및 해석의 종류 및 사용자 등이 표시된다.

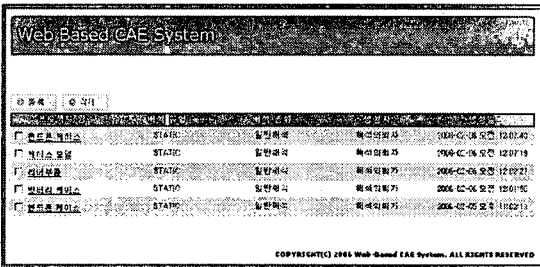


Fig. 11 CAE task service list

5.2 해석 수행 모델의 입력

해석 태스크가 생성되고 선택이 되면 사용자는 해석모델을 입력할 수 있게 된다. Fig.12 는 그 예를 보여준다.

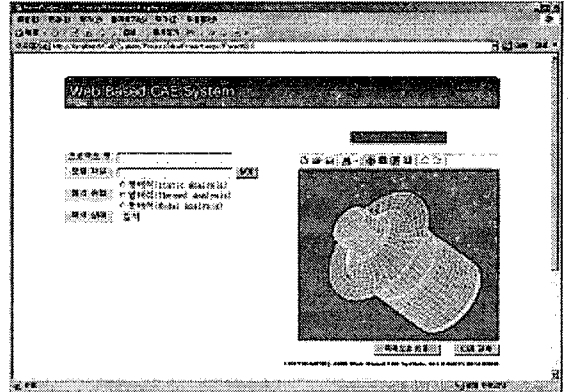


Fig. 12 Make CAE service task

5.3 해석 수행 및 결과 확인

해석 모델이 읽어지고 다음 단계로 넘어가면 다음과 같은 화면이 나오게 된다. 여기서 간단한 해석 태스크의 정보를 확인 할 수 있으며 뷰어를 통하여 모델의 확인 및 경계조건의 입력이 가능하다. 사용자는 왼쪽의 메뉴를 사용하여 경계조건 입력, 재료 물성치의 입력 및 결과를 확인 할 수 있으며 해석을 수행할 수 있다. 해석이 완료 되면 완료 메시지와 함께 해석결과를 뷰어 및 리스트창으로 확인이 가능하다. 해석 결과가 만족스러우면 결과 리포트를 출력하고 모델 데이터 및 해석 결과들은 DB 내에 저장이 되고 해석을 종료한다.

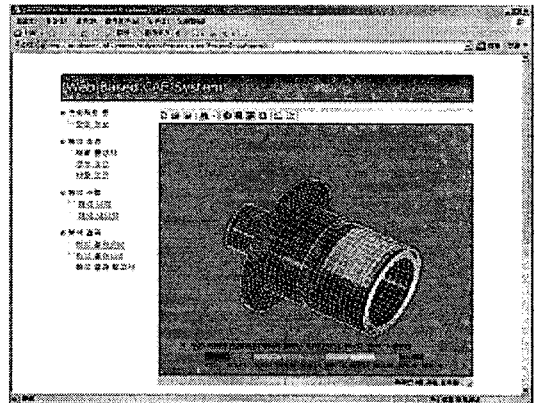


Fig. 13 CAE service main Layout

6. 결론

본 논문에서는 분산/협업 엔지니어링 환경에서 설계와 해석의 통합 연계 작업을 수행할 수 있는

시스템 구현에 대해서 논의하였다. 그 결과로 효과적인 분산 병렬 해석 서버를 포함한 분산/협업 환경에서의 설계 및 해석 시스템을 구현하였다.

이러한 웹을 기반으로 하는 분산/협업 엔지니어링 시스템은 빠른 속도로 진행되는 개인용 컴퓨터 발전 및 초고속 네트워크 기술 등의 컴퓨터 기술과 함께 사용자들이 공학분야의 설계 및 해석을 위한 다양한 도구와 방법 등을 필요로 할 때 실시간 연결방식을 통해서 협력적인 서비스를 보다 빠르고 원활하게 제공할 수 있을 것이다. 또한, 설계 및 해석 분야에서 다양하게 사용되는 범용 도구들을 이용하여 이러한 시스템을 확장한다면 인터넷이나 웹을 통해 분산환경 하에서 동시 병행적으로 작업을 수행하는 다분야 통합 최적설계 분야 등에도 활용될 수 있을 것이다. 이러한 기술들은 현재에도 많은 분야에서 연구되고 있는 반드시 필요하고 실용 가능한 기술이라고 사료된다.

본 논문에서 구현된 분산/협업 환경에서의 시스템 및 개발 기술들은 기업내의 설계 및 제조 파트를 구축하는데 이용할 수 있음은 물론 이러한 시스템을 기업내의 전 영역으로 확장 시킨다면 네트워크를 통하여 정보를 공유하며, 생산활동에 참여하는 기업들의 모임인 가상 기업(virtual enterprise)을 구현하는 데에 매우 중요한 핵심기술이 될 것이다.

후기

본 연구는 산업자원부에서 추진하는 차세대 신기술개발사업의 하나로 수행되고 있는 '지식 프로세스 기반의 제품 개발 협업 프레임워크 개발' 과제의 지원을 받아 수행 되었습니다.

참고문헌

- Regli, W. C., "Internet-Enabled Computer-Aided Design," IEEE Internet Computing, Vol. 1, No. 1, pp. 39-50, 1997.
- Potter, C., "Web-Enabled Engineering: step-by-step," Computer Graphics World, pp. 64-69, 1997.
- Siegel, J., "CORBA fundamentals and programming," John Wiley & Sons, Inc., 1996.
- IONA Technologies, "OrbixWeb programming & reference guide," 1996.
- Korean STEP Center, "International Standard for the Exchange of Product Model Data (ISO 10303) STEP," Sungandang, pp. 11-105, 1996.
- "STEP Programmer's toolkit - ROSE library reference manual," STEP Tools Inc., 1996.
- "ISO 10303-209 - Part 209: Application Protocol: Composite and metallic structural analysis and related design," ISO, 1996.
- Kwon, K. E., "Development of Finite Element Analysis System using STEP Data Standard," A Master's Thesis, Department of Mechanical Engineering, Chungang University, pp. 19-55, 1997.
- Jennifer, A. S., "Parallel frontal solvers for large sparse linear systems," Technical Report TR-2002-012(<http://www.numerical.rl.ac.uk/reports/reports.html>), 2002.
- Karypis, G. and Kumar, V., "Metis ver. 4.0 : A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices," Department of Computer Science, University of Minnesota, 1995.
- Karypis, G. and Kumar, V., "A fast and high quality multilevel scheme for partitioning irregular graphs," Technical Report TR 95-035, Department of Computer Science, University of Minnesota, 1995.
- Kim, J. H., "Finite Element Analysis of Large-scale Structures Using Distributed Parallel Computing Technique," A Thesis for a Doctorate, Seoul National University, Aerospace Engineering, pp. 96-111, 1998.
- Kwon, K. E., Park, M. J. and Yang, S. W., "Distributed CAD/CAE Environment Based on STEP and CORBA," Fourth World Congress on Computational Mechanics, 1998.
- Cho, S. W., Choi, Y., Kwon, K. E., Park, M. J. and Yang, S. W., "Development of a Design Information Sharing System Using Network and STEP," Journal of the KSPE, Vol. 15, No. 9, pp. 82-92, 1998.
- W3C, "Extensible Markup Language (XML) 1.0," W3C Recommendation, 2000.
- W3C, "XML Schema: Part 0,1,2," W3C Recommendation, 2001.
- W3C, "SOAP Version 1.2," W3C Working Draft, 2001.

18. Kwon, K. E., Cho, S. W., Choi, Y. and Lee, G. B.,
“Application of Multi-Frontal Method in
Collaborative Engineering Environment,”
International Journal of CAD/CAM, Vol. 3, No. 2, pp.
51-60, 2003.