

Web 기반 분산설계기법

Web Based Distributed Design Method



양 영 순*

*서울대학교 조선해양공학과 교수

최근 인터넷 및 네트워크 환경의 급격한 발전으로 인해 다양한 분야에서 웹 기반 응용프로그램의 개발을 위한 연구가 활발히 진행되고 있으며, 이에 따라 설계 혹은 생산 관련 분야에서도 웹 기반 분산설계를 위한 연구를 수행하려는 새로운 패러다임(Paradigm)의 변화를 보이고 있다. 즉, 기존의 순차적이고 독립적인 설계 방식의 한계점을 인식하여 인터넷과 웹을 통해 부서 간 혹은 기업간의 분산된 설계 환경을 통합하여 실시간 설계 정보 교환 및 상호작용을 통해 제품개발 기간의 단축 및 경쟁력 있는 제품을 개발하기 위한 하나의 설계 대안으로 대두 되고있다.

한편, 다분야 통합 최적설계(Multidisciplinary Design Optimization, MDO)기술은 분산 설계를 위한 기반 설계기술로서, 특히 협동 최적화 기법(Collaborative Optimization, CO)은 분산된 환경에 가장 적합하게 구성된 MDO 기법이며, 다양한 공학적 원리, 지식, 기술 등을 동시에 고려하여 균형 있고 유기적인 방법으로 최적의 설계를 결정하는 체계적인 설계자동화기술이다. 특히, 산업 현장의 설계 프로세스에 있어서, 세부적인 분야로의 편중된 연구와 각 분야만을 위한적정설계로 인한 분야간 상충현상이 발생하게 되는 것을 볼 수 있는데, 이러한 분야간 혹은 부서간의 설계 타협을 효율적으로 해결하며, 관련 분야간 설계 요소를 동시에 고려하는 설계의 통합화 및 자동화를 구현해 줄 수 있는 MDO 기술을 도입함으로써, 제품 설계에 있어서 고효율 및 균형 있는 설계를 가능케 해준다.

MDO는 1980년 이전까지만 해도 개별적이고 순차적인 최적화 알고리즘에 관한 연구가 주로 진행되다, 1980년대에 들어서 점차 복잡한 시스템의 효율적인 설계 방안으로 대두되어, 1990년대에는 MDO의 설계 방법론 및 프레임워크(Framework)를 구축하기 위한 연구가 진행되었다. 그러나 향후에는 웹 환경의 다양한 자원을 효율적으로 관리하고 통합하기 위한 웹 기반 MDO 프레임워크의 기술이 향후 필요할 것으로 기대된다(Fig. 1).

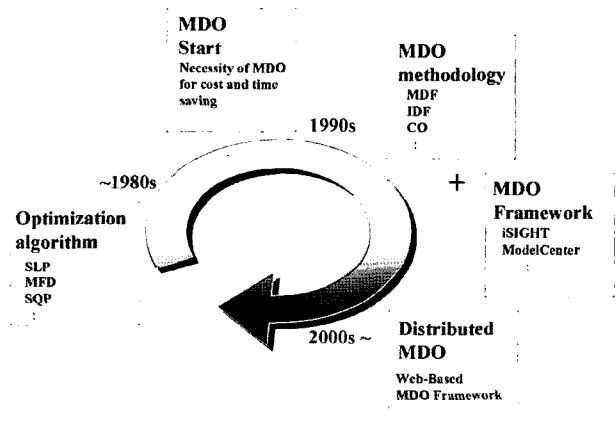


Fig. 1 Changing Paradigm in MDO

따라서 본 연구의 목적은 조선, 기계, 항공 분야와 같이 복잡한 시스템의 효율적인 설계 기법으로 대두되고 있는 MDO와 다양한 설계/해석 정보의 연계 즉, CAD/CAE

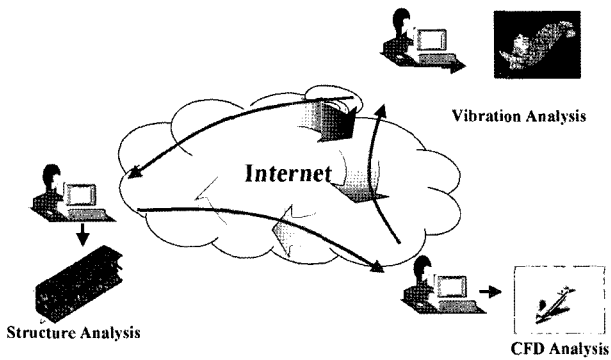


Fig. 2 Web-based Distributed Design

연계 및 웹 기반 분산/협업 설계의 통합화된 프레임워크를 구축하고 활용하기 위한 연구를 수행하고자 한다(Fig. 2).

1. 관련 기술 현황

기존의 분산설계를 위한 연구는 CORBA(Common Object Request Broker Architecture), DCOM (Distributed Component Object Model), 자바의 RMI(Remote Method Invocation)를 이용한 통신방법이 연구되어 왔다.

그러나 이러한 방법들은 상호 운영성의 제약뿐만 아니라 방화벽(Firewall)으로 인해 인트라넷(Intranet) 환경에 적합하며(Fig. 3), 분산된 다양한 플랫폼 및 다른 언어로 개발된 프로그램들을 통합하고, 기업 내의 부서 간 또는 기업간의 협업설계를 위해서는 인터넷상의 표준 프로토콜인 HTTP(Hyper Text Transfer Protocol) 및 XML (eXtensible Mark-up Language)을 기반으로 한 XML-RPC (eXtensible Mark-up Language - Remote Procedure Call), SOAP (Simple Object Access Protocol)과 같은 프로토콜을 사용해야 방화벽의 문제점 및 상호 운영성의 제약 없이 웹 기반 분산설계가 가능하다.

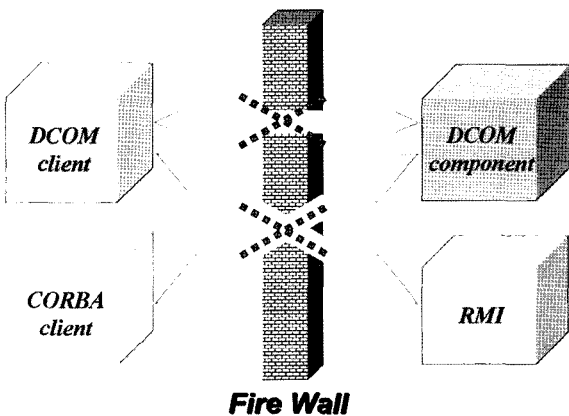


Fig. 3 기존의 분산설계의 문제점

1.1 RPC(Remote Procedure Call)

RPC는 웹보다 훨씬 오래된 기술이며, 한 프로그램이 네트워크 상의 다른 컴퓨터에 위치하고 있는 프로그램에 서비스를 요청하는데 사용되는 프로토콜로서, 마치 같은 프로그램 내의 함수를 호출하는 것과 같이 원격의 프로그램 내의 함수를 호출해서 결과를 전달받는 형식으로 이루어진다. 한편, 요청하는 프로그램이 원격 절차의 처리 결과가 반환될 때까지 일시 정지되어야 하는 동기식(Synchronous)으로 응답을 처리하게 된다.

1.2 XML-RPC

XML-RPC는 1998년 W3C(World Wide Web Consosium)에서 차세대 웹 표준으로 제정한 XML (eXtensible Mark-up Language)기반 분산 시스템 통신 방법으로 XML의 확장성 및 융통성으로 인해 통신의 내용을 포함하게 되면서, RPC를 기술하는 어휘를 제공하며, HTTP를 통신의 매커니즘으로 사용하게 된다. 따라서 XML-RPC는 전통적인 RPC의 문제점이었던 인코딩(Encoding) 및 객체의 직렬화(Serialization)를 XML을 이용해 간단한 텍스트(Text)로 데이터를 표현할 수 있게 된다.

또한, XML-RPC의 가장 큰 특징은 플랫폼(Platform) 및 개발 언어에 관계없이 분산된 시스템 통신이 가능하며, HTTP포트를 통한 통신수단을 사용함으로써 방화벽의 수정 없이 통신이 가능하다는 것이다. 즉, 대부분의 웹 서버와 네트워크는 컴퓨터의 소수의 포트(Port)로만 접근을 허용하는 방화벽으로 보호되는데 웹 기반 HTTP 포트인 80번 포트는 방화벽에 개방되어 있기 때문이다.

XML-RPC요청(Call)은 클라이언트(Client)와 서버(Server) 사이에서 이루어지게 되는데 서버와 클라이언트간의 함수 호출을 캡슐화 한 후 XML 문서를 통해 전달하고 되돌려 받게 된다(Fig. 4).

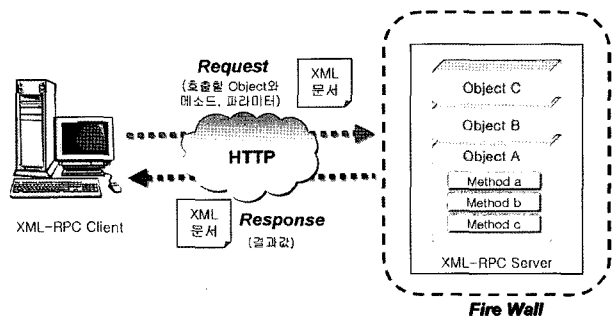


Fig. 4 XML-RPC Communication Mechanism

1.3 SOAP

SOAP(Simple Object Access Protocol)은 마이크로 소프트(Microsoft)와 IBM에 의해 주도적으로 개발되고, W3C를 중심으로 표준화가 진행되고 있는 XML 프로토콜이다. SOAP은 XML로 인코딩된 정보를 교환하는데 쓰이는 상대적으로 가벼운 프로토콜이며, XML-RPC가 전통적인 프로시저 방식을 따르는데 반해 SOAP은 객체를 해당 객체의 상태 정보와 함께 네트워크를 통해 전달하는 객체 지향 패러다임에 그 근원을 두고 있다. SOAP은 아래와 같이 크게 세부분으로 나뉜다.

- SOAP envelope : 메시지에 무엇이 있는지, 어떻게 처리할지에 대해 정의하는 프레임 워크
- SOAP encoding rules : 어플리케이션에서 정의된 데이터 타입들을 표현
- SOAP RPC : SOAP 메시지의 RPC

SOAP은 XML과 HTTP를 기반하므로 XML-RPC와 같이 플랫폼 및 개발언어에 독립적이고, 웹을 통해 데이터를 전송하며, 방화벽에서도 데이터 전송이 가능한 분산 시스템의 핵심 구성요소들 중의 하나로서 중요한 역할을 하고 있다. SOAP 1.0버전은 [SOAP spec] XML-RPC의 초기 개발사인 Userland사의 David Weiner, 마이크로 소프트사에 의해 개발되었고, 현재 SOAP 1.2[SOAP 1.2 spec] 버전까지 발표되었다.

2 XML 기반 분산 설계 프레임워크

지금까지 XML 기반 분산설계 프레임워크의 구축 및 활용을 위해 다양한 분산컴퓨팅 프로토콜의 개념을 알아 보았다. 이러한 개념을 토대로, 실제로 XML 기반 분산설계 프레임워크의 구축 시 필요한 조건이 무엇인지, 향후 연구하게 될 분산설계의 응용의 예를 알아보자.

2.1 프레임워크의 구성조건

XML 기반 분산 설계 프레임워크의 구성조건은 다음과 같은 특성을 갖고 있어야 한다.

- XML과 인터넷 표준을 지원해야 한다.
XML, HTTP와 같은 인터넷 표준을 사용하여, 인터넷을 활용하여 방화벽을 통해 데이터 및 설계 정보를 효과적으로 교환할 수 있어야 한다.

- 프로그램간의 상호 연동성(Interoperability)이 있어야 한다.

분산 설계 시 다양한 응용 프로그램의 이질적인 환경을 통합하기 위해서는 프로그램간의 상호 연동성이 필수적이라 할 수 있다.

- 해석 모듈간의 데이터 교환 표준화가 되어야 한다.
상용 해석 프로그램간의 최적화 혹은 MDO와의 연계 시 해석 모듈과 데이터 표준화가 구축 되어야 한다.

2.2 분산 설계 구현의 활용 방안

분산 환경과 협업설계와 관련하여 일반적인 컴퓨터 네트워크 측면에서 보았을 때, 초기 대부분의 시스템은 중앙에 있는 서버 컴퓨터에 직접 접속하여 모든 작업을 수행하는 호스트 중심의 중앙 집중식 방법이었다. 이 후 컴퓨터의 발달과 더불어 서버의 처리 능력을 가진 PC가 대중화되면서, 클라이언트에서 상당 부분의 기능을 처리하게 하고, 서버에서는 데이터를 다루는 형태의 클라이언트/서버 기술을 이용한 분산 컴퓨터 환경이 제시되었다. 이러한 구조는 현재 많은 기업들이 사용하고 있으며, 처리 응답 속도가 빠르고 컴퓨터의 많은 부분을 직접 제어할 수 있으므로 사용자들의 호응이 높지만 관리자의 입장에서는 유지보수가 힘들고, 애플리케이션과 데이터 형식에 대한 중앙 집중적 관리가 어려운 면이 있다. 특히 최근 인터넷 및 네트워크 환경의 급격한 발전으로 인해 웹 서비스를 이용한 분산설계 기법이 새롭게 대두 되고 있다.

분산설계 시스템은 컴퓨터네트워크로 연결되었으나 공간적, 논리적으로 분산되어 있는 일련의 하드웨어, 소프트웨어, 데이터, 사용자가 특정 목적을 달성하기 위하여 모인 집합체를 말한다. 하나의 분산시스템은 물리적 네트워킹, 시스템서비스, 응용소프트웨어 등의시스템 구성요소에 따른 3가지 수준의 특성을 가지며, 이를 통틀어 분산 설계 시스템 이라고 한다.

1) 분산설계 시스템의 종류

자료의 교환 등을 목적으로 다수의 데이터베이스시스템을 통신네트워크로 연결하여 구성된 분산시스템은 자치성의 정도에 따라 다음의 세 가지로 분류할 수 있다.

- 분산시스템(distributed system) : 물리적, 공간적으로 나뉘어져 있으나 논리적으로는 하나인 시스템. 이를 구성하는 하위DB들은 전혀 자치성은 없지만 완벽한 호환성을 가지고 있다. 단일 기관 내부에서 형편상

DB를 분산시켜야 할 필요가 있을 때 많이 사용하는 방법이다.

- 다중시스템(multiple system) : 완벽한 자치성을 가지고 독립적으로 구축된 DB들로 구성된 시스템. 자료와 프로세스를 공유하려면 각 DB간 1:1 질의 언어(query language)를 개발하여야 하므로 많은 노력이 필요하다.
- 연합시스템(federated system) : 분산시스템과 다중시스템을 혼합한 형태로 자치성과 호환성을 보장한다. 그러나 공유할 부분의 데이터에 대한 송출스키마(export schema)와 연합스키마(federated schema)의 개발이 필요하다.

2) 클라이언트/서버 모델

분산컴퓨팅환경의 구현에 있어 가장 기본적인 모델은 클라이언트/서버모델이다. 클라이언트/서버모델에 있어 클라이언트는 서버에게 필요한 자료나 프로세스를 요청하고, 서버는 이에 응답하여 각종 서비스를 클라이언트에게 제공하는 단순한 구조를 갖고 있다. 일반적으로 클라이언트/서버라 하면 이런 클라이언트와 서버의 역할이 네트워크 상에서 서로 다른 컴퓨터에서 이루어지는 것을 말한다. 클라이언트/서버모델에서 중요한 개념을 보면 다음과 같다.

- 서버(server) : 클라이언트의 요청에 응하여 서비스를 수행하는 부분
- 클라이언트(client) : 서버에 필요한 서비스를 수행하라는 요청을 보내고, 이에 대하여 서버가 보내준 결과를 받는 기능을 수행하는 부분
- 서비스(service) : 클라이언트의 요청에 따라 서버가 클라이언트에게 자료를 제공하거나 특정기능을 수행하는 행위를 말한다.

클라이언트/서버모델의 중요한 특징 가운데 하나는 클라이언트와 서버가 항상 고정되어 있는 것이 아니라 어떠한 경우에는 역할이 서로 바뀔 수도 있다는 것이다. 이런 서로 대칭적으로 대등한 클라이언트/서버 모델을 P2P(peer-to-peer)라고 한다.

클라이언트/서버모델 구현에 있어 가장 어려운 점은 서로 다른 하드웨어모델과 운영체제, 응용소프트웨어로 구성된 컴퓨터시스템들을 연결하기 위하여 네트워크케이블부터 하드웨어, 소프트웨어, 각 레벨의 통신프로토콜 등에 대한 수많은 표준의 적용이 필요하다는 것이다. 네트워크

로 연결된 컴퓨터시스템 사이에 존재하는 이러한 이질성(heterogeneity)문제의 해결은 공간정보시스템의 상호 운용성(interoperability)을 높이는데 결정적인 부분을 차지하고 있다.

3) 컴포넌트 기술

클라이언트/서버 모델에 대한 혁명이 일어날 즈음에 컴포넌트 기술에 대한 관심이 고조되기 시작했다. 처음에 개발자들은 간단한 코드만 재사용하고 있었다. 이것은 일반적으로 사용하는 함수들과 서브루틴(subroutine)으로 구성되어 있었으며 많은 문제들도 갖고 있었는데, 만약 다른 프로젝트에서도 사용될 수 있도록 주의 깊게 만들어지지 않았다면 별로 쓸모없게 되고 다시 만들거나 재구현해야만 했다.

프로그램 기술의 추세가 객체지향(object-oriented) 프로그래밍 모델 쪽으로 흐르면서 이러한 문제에 대한 해결책으로 클래스를 사용하는 것이 제시되었다. 객체지향 프로그래밍은 개개의 메소드에서 클래스의 구현을 감추고 클라이언트가 헤더에 접근하거나 파일을 정의하는 것을 가능하게 하는 기술이다. 이것은 개발자들에게 새로운 클래스를 만들어야 하는 등의 문제들을 안겨다 주었으며, 특정 프로그래밍 언어에 의존하게 되는 등 더욱 더 해결하기 어려운 문제에 직면하게 되었다.

컴포넌트에서의 또 다른 솔루션은 인터페이스 기반의 프로그래밍이다. 이 기술로 인하여 컴포넌트는 자신과 잘 맞는 인터페이스를 가지게 되었고, 이 인터페이스를 규약으로 삼아서 다른 개발자들은 인터페이스가 변하여도 코드를 변경하지 않도록 개발할 수 있게 되었다.

컴포넌트 인터페이스는 바이너리(binary)로 만들어져 개발자들이 여러 다른 언어로 개발을 해도 무관하도록 되어 있다. 컴포넌트 기반 분산설계는 CORBA(Common Object Request Broker Architecture), DCOM(Distributed Component Object Model), 자바의 RMI(Remote Method Invocation) 및 가장최근 개발된 닷넷(.NET)을 이용한 닷넷 리모팅(.NET Remoting)과 닷넷 웹서비스(.NET Web Service)의 통신방법이 있다.

4) 웹 서비스

웹 서비스는 인터페이스를 통한 기능의 하나이다. 이 기능은 HTTP 같은 표준 인터넷 프로토콜을 통해서 가능하다. 즉, 어떤 클라이언트도 인터넷을 통해서 인터넷에서 서버에 접속하는 RPC(Remote Procedure Call) 같은 것을 사용하고 XML로 응답을 받을 수 있는 기술이다. 웹

서비스는 메시지가 클라이언트와 서버에 전송될 때 SOAP (Simple Object Access Protocol)이라고 불리는 특별한 XML로 인코딩 된다. 이 프로토콜은 원거리의 장비에 접속할 때 제공되는 표준이다.

웹 서비스에 대한 기본적인 개념은 클라이언트와 서버가 어떤 기술, 어떤 언어, 어떤 장비이든 간에 상관없이 분산설계가 가능하다는 것이다.

3. 웹 서비스를 이용한 분산설계 구축 예

분산설계는 분산된 자원(Resource)의 통합화가 필수적인데, 여기에는 다음과 같은 이질적인 요소의 문제점이 발생한다.

- 지리적으로 분산된 설계자(Designer)
- 물리적인 분산(데이터, 프로그램 언어 및 플랫폼의 이질성)
- 통신수단의 이질성

따라서 엔지니어링분야에서 발생하는 이러한 본질적인 이질적인 요소들의 통합화를 수행하기 위해 다음과 같은 기본구성조건이 필수적이다.

- 표준화된 데이터 구조 : XML
- 분산된 정보교환을 위한 표준화된 통신 프로토콜 : HTTP
- 플랫폼 독립성(Platform independence)
- 개발언어간 상호 운용성(interoperability)
- 방화벽(Firewall)의 제한이 없음

웹 서비스(Web Service)는 위에서 제시한 분산된 자원의 통합화를 위해 발생하는 분산설계의 문제점을 해결하기 위한 차세대 설계 기법이다. 웹 서비스는 W3C(World Wide Web Consortium)에서 제안한 분산설계 기법으로 HTTP (Hyper Text Transfer Protocol), XML(eXtensible Markup Language), SOAP(Simple Object Access Protocol) 및

WSDL(Web Service Description Language)등의 산업 표준기술을 사용하여 분산서비스를 제공함으로써, 플랫폼 독립성, 개발 언어간 상호 운용성 및 방화벽의 제한 없이 분산설계를 수행할 수 있다.

따라서 분산설계를 수행하는데 있어서 웹 서비스를 이용하여 Fig. 5과 Fig. 6에 나타난 바와 같이 2가지 설계를 통해 그 활용방안의 가능성을 모색하고 있다.

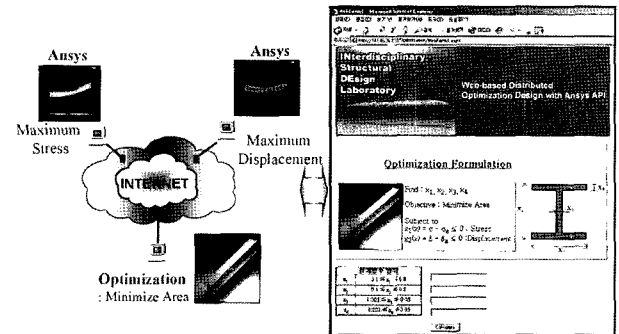


Fig. 5 웹 서비스를 이용한 분산설계 구축 예 (I-beam 설계)

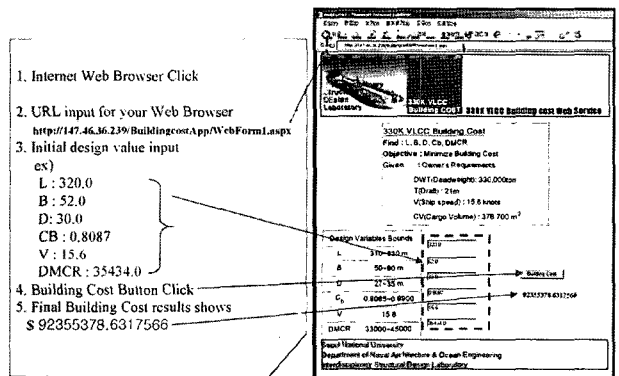


Fig. 6 웹 서비스를 이용한 분산설계 구축 예(선박초기설계)

감사의 글

이 연구는 2005년 과학재단 지정, 첨단조선공학 연구센터(ASERC)의 연구비 지원에 의한 결과의 일부임.