

인터넷 기반 Collaborative Design 시스템 개발

한승헌(연세대 대학원)*, 차주헌(KIST), 최용제(연세대)

ABSTRACT

The configuration of design technology in 21st century must require collaborative design system, in which distributed design organizations and their related enterprises can mutually cooperate by overcoming time and regional restrictions, and by exactly exchanging and sharing design data. To achieve these objectives, it is necessary that the design system can properly deal with the dynamic change between regionally distributed modules. In this paper, we will discuss developing the dynamic virtual collaborative design system based on WWW by implementing recent web technologies such as Java and VRML.

Key Words: Collaborative Design (상호협동 설계), Distributed Environment (분산환경), WWW (World Wide Web), Internet-based Design (인터넷 기반 설계), Java, VRML

1. 서 론

설계에 있어서 그 조직간에 혹은 그와 관련된 기업간의 설계 데이터 혹은 설계 정보를 서로 교환할 수 있도록 하는 것은 필수적이다. 모든 설계정보가 컴퓨터에 저장되고 이를 컴퓨터 네트워크를 통하여 공유할 수 있는 것이 오늘날 과거에 비해 비약적으로 발전된 점이다. 또한 이제는 WWW의 보편화에 힘입어 전세계 모든 기업, 설계자, 개인들이 인터넷이라는 가상세계에 묶여, 서로간의 정보교환뿐만 아니라 소비자의 요구에 맞는 생산품을 얻기 위해 인터넷을 통하여 가상의 회사를 만들어내고, 전세계에 퍼져있는 설계자, 생산자에게서 그때그때 필요한 만큼의 수요만을 생산해내는 차세대 주문생산 개념도 차차 정착해 가는 추세이다. [1~3]

인터넷 기술에도 많은 변화가 있었다. 이미 언급한 바 있는 HTML(Hyper Text Management Language)을 기반으로 한 WWW는 인터넷을 일반인들에게 보편화 하는데 큰 공헌을 하였으며, 이 WWW에 기반하는 Java 언어는 단순한 문서의 표시에 불과한 WWW에 생명을 불어넣었다. 또한 VRML(Virtual Reality Modeling Language) 2.0은 인터넷에 3D 그래픽과 간단한 애니메이션을 추가시킴으로써 인터넷의 정보를 현실감 있게 표현할 수 있도록 해주었다.

이러한 기술들은 현재 STEP 데이터 가시화 연구 등의 설계분야[3], Java Collaborative Application over Internet의 연구에 사용되고 있다.

하지만, 앞에서 언급한 연구들은 3차원 형상 모델이 없는 데이터의 상호교환이나, 동적인 정보교환이 없는 데이터의 가시화에 중점을 맞추고 있다. 더 효율적인 설계 공유를 위해서는 3차원 형상 모델과 그 데이터의 상호교환 모두가 필요하다.

본 논문에서는 급속히 발전하는 인터넷 기술인 Java와 VRML 기술을 이용하여 3차원 형상과 설계 데이터의 동적인 상호교환이 가능한 새로운 개념의 Collaborative Design 시스템을 제안하고 그 구현 예를 보고하고자 한다.

2. Collaborative Design 시스템

협동적 설계 시스템(Collaborative Design System)은 지역적으로 떨어져 있는 여러 개의 설계 및 생산 조직간의 설계자 혹은 생산자, 소프트웨어적 모듈 등이 실시간에 공동으로 설계작업을 수행할 수 있는 환경을 제공함으로써 설계작업의 효율을 극대화시키고자 하는 시스템이다. 이러한 목적을 만족시키기 위해서는 다음과 같은 두 가지 요소를 고려할 필요가 있다.

1) 설계대상의 공유

이와 관련한 인터넷 시스템은 설계정보가 저장되어 있는 파일을 FTP 등을 사용하여 직접 교환하는 기존의 방법이 있으며, 또한 2차원 데이터의 상호공유를 위한 화이트 보드 시스템 등이 있다.

2) 설계자의 행위 표현.

실제로 공동 설계작업에 있어서, 여러 사람이 하나의 설계대상을 놓고 토론을 할 때, 그들이 나누고 있는 대화뿐만 아니라 그들의 몸짓이나 표정 등이 대화 이상의 효과를 나타낼 수 있다. 이러한 설계자의 행위를 표현하기 위해 화상회의, 채팅 시스템 등이 사용될 수 있다. [4]

본 논문에서는 첫번째 요소인 설계대상의 공유에 초점을 맞추어 다음과 같은 시스템 기능을 설정하였다.

- 1) 인터넷을 통한 3차원 데이터의 공유.
- 2) 설계 데이터의 전달을 위한 데이터 구조 모델 개발
- 3) 자율 지능형 정보 전송 체계 모델 개발
- 4) 각 클라이언트들간의 지속성(Consistency) 유지를 위한 알고리즘 제시

3. 시스템의 개요

이러한 연구 방향에 맞추어 시스템은 기본적으로 다음과 같은 클라이언트/서버 (Client & Server) 구조를 갖는다. 여기에 덧붙여 클라이언트 매니저 (Client Manager)를 설정하여 클라이언트 정보의 지속성을 유지시키는 구조를 가진다. 이러한 구조는 Fig.1에 도시화하였다.

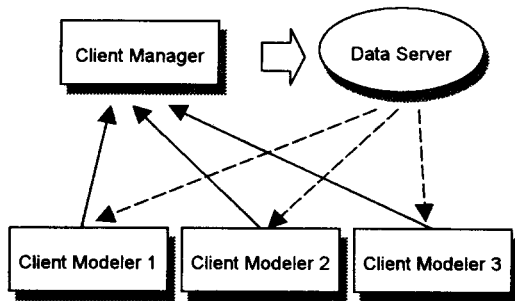


Fig.1 System Architecture

3.1 클라이언트 모델러 (Client Modeler)

클라이언트는 CAD 시스템의 기본적인 모든 것을 가지고 있다. 클라이언트 시스템의 기본 골격은 Java 애플릿 (Applet)에 의해 구현되며, 3차원 형상의 가시화 부분은 VRML에 의해 수행된다.

클라이언트가 처음 서버가 위치한 웹사이트에 접속을 하면 서버에게 현재 진행 중인 데이터를 요청

하게 된다. 그 다음 서버로부터 데이터 스트림 (Data Stream)을 받게 되며, 이를 기반으로 클라이언트는 현재 진행 중인 Scene의 VR 인스턴스(Instance)를 생성하게 된다. 설계자는 이러한 과정을 통하여 현재의 설계 진척 상황을 알 수 있게 되며, 이 Scene을 통하여 설계작업을 수행한다. 이러한 설계자의 작업은 작업에 의해 설계정보가 변할 때 마다 다시 서버에 설계 데이터를 전송하게 된다.

클라이언트 모델러의 주요 기능을 요약하면 다음과 같이 기술할 수 있다.

- 1) 진행 중인 작업의 가시화.
- 2) 작업에 참여하는 환경 제시.
- 3) 작업변화를 서버에 전달.
- 4) 서버로부터 작업변화 관찰.

3.2 데이터 서버 (Data Server)

데이터 서버는 Java 어플리케이션으로 작성된다. 서버의 주된 역할은 데이터의 분배 및 저장이다. 즉, 처음으로 클라이언트 모델러가 서버에 접속을 했을 때, 클라이언트의 요청에 따라 서버의 설계 모델 객체들을 스트림의 형태로 변환하여 클라이언트로 전송하고, 어떤 한 클라이언트의 설계 객체가 변화하였을 때, 클라이언트 매니저로부터의 요청에 따라, 다른 클라이언트에게 이 변화를 알리는 역할을 한다. 또한, 설계 객체를 파일의 형태로 저장하는 역할도 수행한다.

3.3 클라이언트 매니저 (Client Manager)

클라이언트 매니저는 각 클라이언트들간의 지속성을 유지시키기 위해 특별히 지정된 클라이언트이다. 설계가 진행되고 있는 동안, 각각의 클라이언트들은 같은 시간에 서로 같은 설계대상물을 다르게 변화시킬 수 있다. 이러한 문제를 해결하기 위해, 이 클라이언트 매니저는 각각의 클라이언트 모델러로부터 변화된 설계정보를 기억하고, 설계를 관찰할 수 있는 관리자에게 의뢰하여 어떠한 클라이언트로부터의 정보가 유용한지를 판단하고, 이를 데이터 서버에 알린다. 여기서 관리자란 인간일 수도 있고, 버전 컨트롤러 혹은 다른 해석 모듈일 수도 있다.

위에서의 3가지 기본 구성요소를 기본으로 인터넷을 통하여 지역적으로 분산되어 있는 설계자들이 동시에 하나의 설계 대상물을 가지고 설계작업을 수

행할 수 있는 시스템을 구현한다.

4. 시스템 구현 방법

시스템의 작업환경은 웹브라우저인 Netscape 와 VRML Scene 을 보기 위한 Netscape 의 Plug-in 모듈인 Cosmo Player, 설계작업 Java 애플릿에 의해 구성된다.

데이터를 주고 받는 작업은 Java API 의 Socket 클래스로부터 상속을 받아 구현한 Socket 통신이며, VRML Scene 과 Java 애플릿과의 연결을 위해 Silicon-Graphics 사의 EAI(External Authoring Interface) [5]를 이용하여 구현한다.

Fig.2 는 실제의 서버와 클라이언트의 구성을 도시화한 것이다.

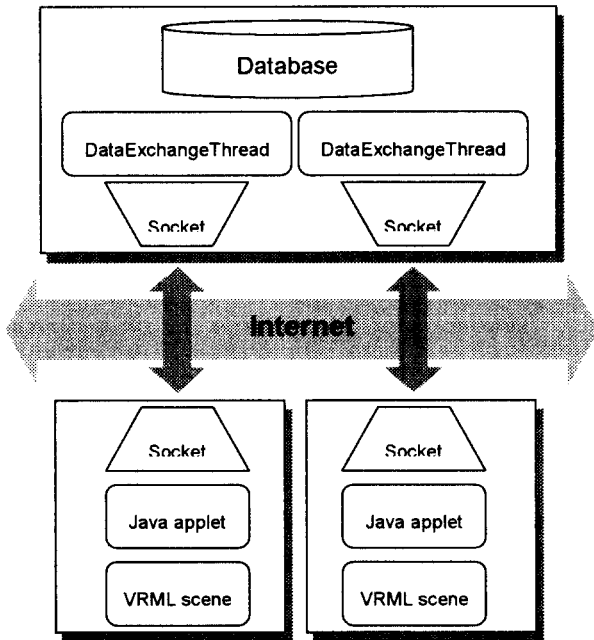


Fig.2 Configuration of Client & Server System

4.1 설계 객체의 직렬화 (Serialization)

설계 객체를 Socket 통신으로 전달하려면 객체를 스트림으로 표현하는 것이 필요하다. 설계 객체를 데이터 스트림으로 변환하여 통신을 구현함에 있어서 객체 생성, 객체 변환, 객체 소멸 즉, 'New', 'Update', 'Delete'와 같은 3가지의 기본 메시지 정의가 필요하다.

4.2 설계 클라이언트 초기화

설계 클라이언트가 처음 서버가 위치한 곳에 접속하면 지금까지의 설계상황을 볼 수 있어야 한다. 이는 접속 초기에 데이터 서버에서 데이터 교환 쓰레드 (Data Exchange Thread)를 생성하고 이 쓰레드가 전체의 설계 객체 리스트를 스트림으로 변환하여 클라이언트에 전송하고 설계 클라이언트에 이 스트림을 해석하여 객체를 순차적으로 생성해낸다.

4.3 설계 클라이언트의 변화 관찰

설계 서버는 각 데이터 교환 쓰레드의 관찰자 (observer)를 가지고 있다. 따라서, 클라이언트의 객체 생성 및 변환, 소멸의 메시지가 데이터 교환 쓰레드로 들어오게 되면, 이 쓰레드를 관찰하고 있는 관찰자가 이에 반응하고 변환 메시지를 다른 데이터 교환 쓰레드에 전달하여 다른 클라이언트들이 이를 받아 자신의 객체 데이터를 변환하게 된다. Fig.3 은 이러한 과정을 도시화한 것이다.

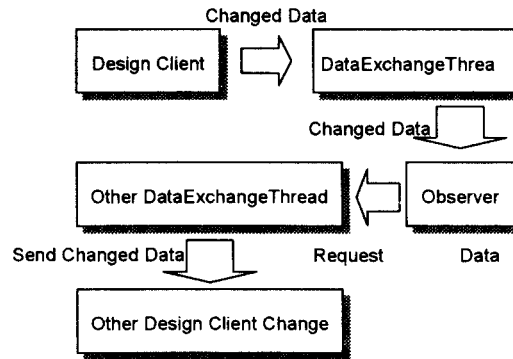


Fig.3 Flow of Changed Data

4.4 VRML 과 Java 애플릿의 연결

Silicon Graphic 사에서는 HTML 문서상에서 애플릿 과 VRML Scene 의 연결을 가능케 해주는 Java API 인 EAI (External Authoring Interface)를 제공한다. 이 인터페이스를 이용함으로써 웹에서의 통신기능은 애플릿에 의해 구현되며, 설계정보의 표현은 VRML 에 의해 가시화된다.

이 EAI 는 VRML 에서의 이벤트를 Java 애플릿 내 의 callback() 함수로 전달하는 기능을 가지며, 또한 Java 애플릿의 수행 결과를 VRML 에 전달함으로써

VRML Scene 을 바꾸어주는 기능을 포함하고 있다.

5. 시스템의 개발 사례

전절에서 설명한 방법으로 설계 클라이언트에 해당하는 간단한 기능을 가진 클라이언트 애플릿과 데이터 서버, 설계 매니저 역할을 하는 서버 어플리케이션 등을 실제로 구현해보았다.

5.1 클라이언트 구현

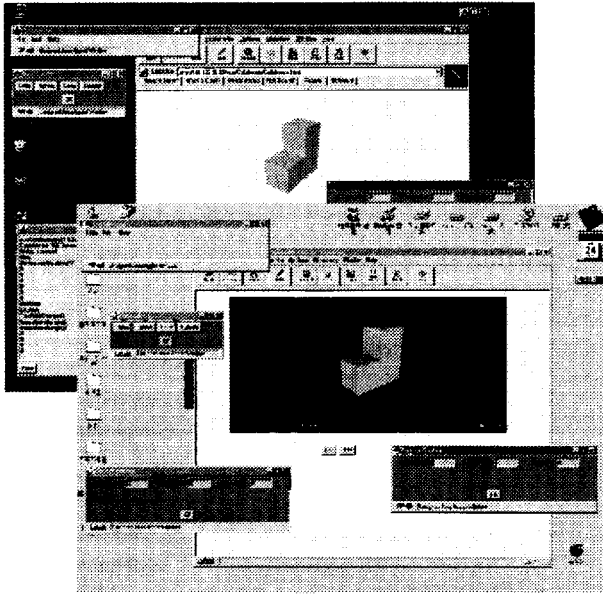


Fig 4. Implemented Clients (Two different machine view)

클라이언트들은 Fig.4 와 같이 지역적으로 다른 곳에서 Netscape 를 통하여 동일한 설계대상의 가시화 및 설계변경을 수행한다. 각 클라이언트는 기본적으로 간단한 편집 기능을 가진 프레임을 로드할 수 있고, 프레임은 객체를 생성하는 다이얼로그 (Dialog) 와 각 VRML Scene 위 에 존재하는 객체를 클릭하면 그 객체의 성질 (Property)을 조절할 수 있는 다이얼로그가 생성된다. 따라서 사용자는 이러한 GUI 를 사용하여 객체를 생성, 변화, 소멸할 수 있다.

5.2 서버 구현

서버는 Fig 5 에서처럼 간단한 콘솔 어플리케이션으로 제작하였다. 데이터의 흐름을 표현하고, 이를

요구 클라이언트에 전달한다.



Fig 5. Implemented Console Server

6. 결 론

지금까지 인터넷 상에서 상호 협동 설계를 위한 시스템을 개발하는 과정을 설명하였다. 본 시스템은 웹의 최신기술인 VRML2.0 과 Java 언어를 사용하여 지역적으로 서로 떨어져 있는 설계자사이에서 3 차원의 동적인 설계변화를 감지하여 이를 다른 설계자에게 알려주는 기능을 수행하면서, 데이터간의 지속성도 효과적으로 유지할 수 있었다. 따라서, 본 연구에서 제안한 Collaborative Design 시스템 구현을 통하여 설계작업을 보다 효율적으로 수행할 수 있는 신개념의 설계 환경이 제시될 수 있었다.

향후 연구로는 클라이언트 매니저의 에이전트화, 분산되어있는 전문가 시스템이나 해석 툴들과의 연결, 데이터의 표준화 등이 필요하다.

참고문헌

1. Inyong Ham and Sounder R.T.Kumara, " Global Collaborative Manufacturing and the World-Wide Web" 1996.
2. Richard Bentley, Thilo Horstmann, Klaas Sikkel, and Jonathan Trevor, "Supporting Collaborative Information Sharing with the World Wide Web: The BSCW Shared Workspace System" 1995
3. 김철영, 김남국, 김영호, 강석호, "웹과 STEP 을 이용한 제품 설계 정보 공유 시스템" 1996
4. John C. Tang "Finding from observational studies of collaborative work"
5. Chris Marrin " Proposal for a VRML 2.0 Informative Annex, External Authoring Interface Reference" Silicon Graphics, Inc. 1997