

# 웹 기반 협동설계를 위한 솔리드 모델러에 관한 연구

정회원 김 응 곤\*, 윤 보 열\*\*

## A Study on a Solid Modeler for Web-based Collaborative Design

Eung-kon Kim\*, Bo-yul Yoon\*\* *Regular Members*

### 요 약

오늘날 컴퓨터와 통신 기술의 발달로 시간과 공간의 제약 없이 공유된 가상 공간에서 작업을 하는 컴퓨터 지원 협동 시스템(CSCW)이 등장하고 있다. 그러나 대부분의 시스템은 그룹웨어를 사용하고, 공유객체는 윈도우 탐색기 형태를 지니고 있고, CAD를 비롯하여 그래픽 분야의 시스템은 하드웨어와 응용 소프트웨어에 크게 의존하기 때문에 협동시스템이 많지 않다.

본 논문은 인터넷 웹 상에서 협동작업으로 이루어지는 CAD시스템의 3차원 솔리드 모델러를 제안한다. 이 시스템은 Java와 Java 3D API를 이용하여 개발함으로써 플랫폼에 구애받지 않고, 특정한 3차원 그래픽 소프트웨어 없이 인터넷망과 웹브라우저를 이용하여 공유된 가상 공간에서 협동작업이 이루어지도록 한다. 협동설계 참여자들은 인터넷을 통해 솔리드 모델러 서버에 접근하여 원하는 3D 도형을 생성하고 조작한다.

### ABSTRACT

As computer systems and communication technologies develop rapidly, CSCW(Computer Supported Collaborative Work) system appears nowadays, through which it is available to work on virtual space without any restriction of time and place. Most of CWCS systems depend on a special network and groupware. The systems of graphics and CAD are not so many because they are specified by hardware and application software.

We propose a Web-based collaborative CAD system which is independent from any platforms, and develop a 3D solid modeler in the system. This system can be worked in the environment of Client/Server architecture. Clients connect to the design server through Java applet on WWW. The server is implemented by Java application.

### 1. 서 론

컴퓨터를 이용한 설계가 보편화되면서 다양한 CAD 시스템이 등장하게 되었다. 하지만 상용 CAD 시스템은 가격이 비싸고, 플랫폼에 의존하여 디스플레이해야 하므로 프로그램을 포팅하기가 쉽지 않다. 멀티플랫폼 프로그램 개발자는 플랫폼마다 별도로 코딩해 주어야 하고 변화가 있을 때마다 각 플랫폼에 맞도록 수정하고 테스트하여야 한다<sup>1)</sup>.

자바언어를 이용하여 웹 상에서 수행되는 CAD 프로그램을 개발하면 이러한 문제들을 해결할 수 있다. 자바코드는 기계어로 컴파일되지 않고 플랫폼

에 독립적인 바이트코드로 컴파일되므로 SGI 워크스테이션이나 SPARC 스테이션에서 동작하는 것처럼 PC에서도 잘 동작한다.

인터넷 서비스 중에서 웹은 사용하기 편리한 사용자 인터페이스와 멀티미디어 환경을 제공하여 주기 때문에 웹을 기반으로 한 다양한 서비스가 등장하고 있다. 이러한 일련의 변화는 CAD 엔지니어링 환경에도 큰 변화를 불러 일으켰다. 일례로 인터넷을 통한 각종 설계 정보 공유가 가능해졌으며, 더불어 CAD 데이터를 공유함으로써 공동의 프로젝트를 전개해 나갈 수도 있다. 이러한 인터넷의 특징은 CAD 시스템 환경을 크게 바꾸었으며, 또한 인터넷

\* 순천대학교 컴퓨터과학과,  
논문번호: 020231-0514, 접수일자: 2002년 5월 14일

\*\* 순천대학교 컴퓨터그래픽스 연구실

도 이러한 엔지니어링 분야를 지원하기 위한 발전 을 거듭하고 있다<sup>[2]</sup>.

상업용 솔리드 모델링 시스템은 여러 해 동안 사 용되어 왔지만 아직까지 사용하기가 어려우며, 기능 이 상대적으로 제한되어 있으며 다른 응용 시스템들 과 통합하기가 어려웠다. 또한 웹 상에서 솔리드 모 델의 디스플레이와 변경을 용이하게 해주는 도구는 개발되어 있지 않다. 최근 VRML(Virtual Reality Modeling Language)언어를 이용하여 웹 상의 가상 세계에서 3차원 형상을 표현하는 도구가 개발되고 있으나 상호작용의 기능이 제한되어 있는 실정이다<sup>[3]</sup>.

Java 3D를 이용하여 솔리드 모델러를 개발하면 여러 가지 장점이 있다. 첫째, Java 3D는 Java 확장 API이므로 시스템의 제약 없이 언제나 동일한 결과 를 얻을 수 있어서 개발자들이 여러 플랫폼에 맞추 어 개발하는 시간과 노력을 줄일 수 있다. 둘째, Java 3D는 기존의 수많은 그래픽 라이브러리의 특징과 편리함, 그리고 다른 라이브러리에서 사용하 지 않는 새로운 기술들을 포함하고 있다. 셋째로 Java 3D는 웹 상에서 구현이 가능하며, Java와 완 벽하게 호환된다. 지금까지의 솔리드 모델러는 특정 플랫폼에서만 실행이 되거나, Java로 구현하였다라 도 Java 3D를 사용하지 않아 모든 기능들을 만들어 야 하므로 개발 시간과 기능 면에서 한계가 따랐다.

따라서 본 논문에서는 CAD시스템은 별도의 전용 시스템 없이 인터넷 망과 웹 브라우저를 통해 원격 공동설계시스템 서버에 접속하여 각 클라이언트에서 공동작업이 이루어지도록 한다. 이 시스템은 클라이 언트/서버 구조로 클라이언트는 자바 애플릿을 통해 웹 상에서 접근하고 서버는 자바 애플리케이션으로 접속을 통제하는 접속관리자, 작업 그룹의 동기화를 유지하며 공유된 작업 공간을 확보하는 작업관리자, 그리고 3차원 도형을 그릴 수 있는 솔리드 모델러 로 이루어져 있다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연 구를 살펴본다. 3장에서는 협동CAD 시스템의 전체 구조를 개략적으로 보여 준다. 4장에서는 협동CAD 시스템의 솔리드 모델러의 구성과 기능을 설명한다. 5장에서는 솔리드 모델러의 구현으로 각 모듈들을 설명한다. 6장은 기존 시스템과 비교하고 요약 및 결론을 맺는다.

## II. 관련 연구

전통적인 협동작업은 일정한 시간에 일정한 장소

에서 함께 만나 자료를 보고 서로 의견을 말하면서 진행되었다. 오늘날에는 컴퓨터와 통신 기술의 발달로 시간과 공간의 제약 없이 공유된 가상 공간에서 상호작용을 하면서 효율적인 작업을 하는 새로운 시스템이 대두되고 있다<sup>[5,6]</sup>.

컴퓨터 지원 협동 작업 (CSCW : Computer Supported Cooperative Work)은 컴퓨터를 기반으로 공동작업을 지원하고 공동작업 그룹 사이의 상호작용이 이루어지도록 하고 있다. CSCW의 기반이 되는 컴퓨터 기술로는 컴퓨터, 사용자 인터페이스, 네 트워크, 멀티미디어, 객체지향 개념, 가상현실, 인공 지능 등이다. CSCW의 궁극적인 목표는 사람들이 시간과 공간에 관계없이 협동작업이 효율적으로 이 루어지도록 하는 것이다<sup>[7]</sup>.

지금까지 개발되어 이용해진 CSCW는 전자우편 을 비롯하여 화상회의, 공동프로그래밍, 전자결재, 원격교육, 원격 진료 등이 있다. 하지만 특정한 시 스템 플랫폼을 요구하거나 특정한 네트워크와 그룹 소프트웨어를 사용하여 폐쇄적으로 공동작업이 이루 어지는 경우가 대부분이다.

웹 상에서 간단하게 설계할 수 있는 Web CAD<sup>[8]</sup> 는 밀링머신 등 NC가공 기계와 연계하여 사용할 수 있다. 그러나 여러 명이 동시에 상호작용을 하면 서 협동작업을 할 수는 없다.

VRML 형식으로 동적인 3차원 데이터를 공유하 는 공동설계/조립 평가 시스템이 있다. 부품 라이브 러리를 구축하고, 이를 이용하여 제품개발자와 부품 공급자가 각자의 부품형상을 가지고 서로 의견을 공유하면서 웹 상에서 가상조립을 하고 동영상으로 저장이 가능한 시스템을 구현하였다<sup>[9]</sup>.

가상 가구 조립 시스템 웹 상에서 가구 부품들을 선택하여 배열하고 크기를 조정할 수 있다. 협동작 업이 이루어지지는 않지만 가구 판의 사이즈와 각 도 등을 임의로 조정할 수 있으며 조립 완성 후 사 용부품 리스트가 작성된다<sup>[10]</sup>.

## III. 협동 CAD시스템의 구조

컴퓨터 지원 협동 작업이 이루어지려면 공간적으 로 떨어진 사용자들의 정보를 공유하기 위하여 컴 퓨터 통신을 이용하여 정보를 주고받는 네트워킹이 되어야 하고, 이 통신망을 통하여 서로 협동작업을 수행하는 대상 물체를 공유 공간에서 볼 수 있도록 효과적인 파일 포맷이 요구되고 뷰 유지가 되어야 한다.

시스템은 클라이언트/서버 구조로 클라이언트는 자바 애플릿을 통해 웹 상에서 접근하고 서버는 자바 애플리케이션으로 접속을 통제하는 접속관리자, 작업 그룹의 동기화를 유지하며 공유된 작업 공간을 확보하는 작업관리자, 그리고 3차원 도형을 그릴 수 있는 솔리드 모델러로 이루어져 있다. 다음 그림 1은 협동 CAD시스템 전체 구조를 보여주고 있다.

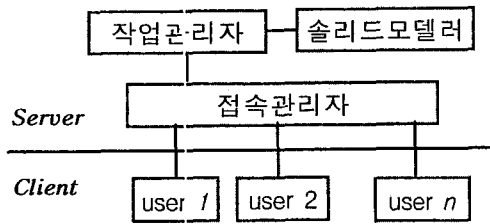


그림 1. 시스템의 구조.

접속관리자는 웹서버를 통하여 들어온 클라이언트의 서비스 요청을 받아 분석하여 메시지로 세션 관리자에게 보내고, 처리된 결과를 웹 문서의 형태로 클라이언트에 전송한다. 사용자의 ID에 따라 접속을 설정하거나 해제할 수 있고, 작업하는 동안 클라이언트의 접속을 계속 유지시키는 역할을 한다. 포트 번호를 이용해서 서버 소켓을 생성하고 포트에 접근하는 클라이언트를 기다린다. 공동작업을 위해 각 노드들의 세션 연결에 관한 정보를 가지고 있으면서 공동작업시 작업 그룹을 생성하거나 삭제하는 기능을 담당한다. 작업 그룹에 클라이언트가 입장하거나 퇴장할 때 세션 정보를 관리하며 동기화 되도록 세션을 유지시킨다. 그림 2는 접속관리자의 구조를 보여준다.

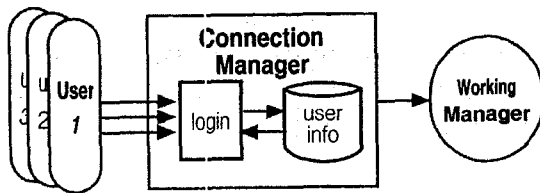


그림 2. 접속관리자 구조

작업관리자는 클라이언트의 작업 요청에 따라 실제적인 작업을 처리한다. 개인작업공간과 공유작업 공간을 확보하도록 하고 도형 객체를 생성하거나 저장된 파일을 불러와 변형시킨다. 공유작업공간에서는 공동작업의 일관된 상태를 유지하면서, 그룹의 클라이언트들이 WYSIWIS 상태에서 작업을 수행할 수 있도록 뷰 동기화(View Synchronization)한다.

그림 3은 작업관리자의 구조를 보여준다.

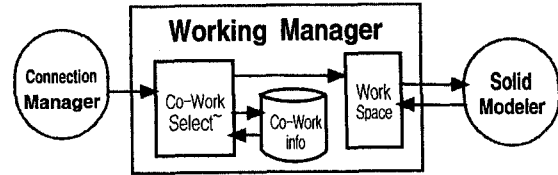


그림 3. 작업관리자 구조

솔리드 모델러의 시스템 라이브러리는 여러 가지 Java 클래스들로 구성된다. 워크스페이스는 워크스페이스에 솔리드를 추가하거나 삭제하는 일을 수행하며 뷰는 워크스페이스에 솔리드 객체들을 디스플레이하는 추상클래스로서 평행투영 뷰 클래스와 원근투영 뷰 클래스는 뷰 클래스로부터 상속받으며, 워크스페이스에 여러 방향의 뷰를 나타낸다. 그림 4는 솔리드 모델러의 개략적인 구조를 나타낸다.

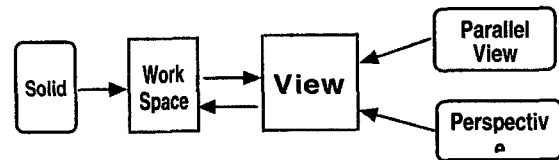


그림 4. 솔리드 모델러의 구조

클라이언트에서는 일반적인 HTML문서와 Java 애플릿을 사용한다. 기본적인 통신프로토콜은 HTTP를 사용하며 설계데이터의 교환은 소켓을 이용한다. 솔리드 모델의 X-Y평면, Y-Z평면, X-Z평면에 대한 투영도, 3차원 모델에 대한 평행투영도와 원근투영도를 4 개의 창에 표시한다. 툴박스를 통해 기본 도형을 만들 수 있고, 정밀한 객체를 생성하기 위해 수치입력으로 객체를 만들 수 있다. 원만한 상호작용을 위해 채팅 창을 별도로 두고 대화를 나눌 수 있다. 그림 5는 본 시스템의 공유작업 공간의 클라이언트 인터페이스를 보여주고 있다.

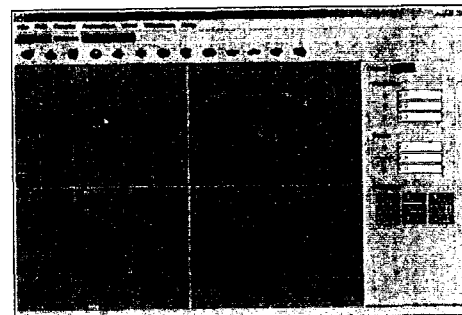


그림 5. 클라이언트의 인터페이스

#### IV. 협동설계를 위한 솔리드 모델러

##### 1. 솔리드의 데이터 구조

솔리드 모델은 솔리드(Solid), 면(Face), 루프(Loop), 반모서리(HalfEdge)와 꼭지점(Vertex)으로 구성되며, Mantyla가 제안한 반모서리(Half Edge) 데이터구조를 이용한다<sup>[11]</sup>. 솔리드의 각 면은 모서리 루프들의 꼭지점의 좌표로 결정되는 다각형이다. 각 모서리는 두 개의 모서리 루프에 속하며, 모서리는 두 개의 반모서리로 이루어진다. 즉, 하나의 모서리를 두 개로 쪼개서 쪼개진 각각의 모서리가 원래 모서리를 공유하는 두 개의 면에 대해 따로따로 사용하며, 각 모서리를 서로 방향이 반대인 두 개의 반모서리로 쪼개서 각각의 면에 대해 일관성있는 방향을 갖는 반모서리의 연결리스트의 형태로 저장한다.

##### 2. 시스템 라이브러리

솔리드 모델러는 Java 3D API를 이용하여 개발하며, 시스템 라이브러리는 여러 가지 Java 클래스들로 구성된다. 시스템 레벨의 클래스는 워크스페이스 클래스(Workspace class), 뷰 클래스(View class), 평행투영 뷰 클래스(ParallelView class), 원근투영 뷰 클래스(PerspectView class), 솔리드 클래스(Solid class)로 구성하였다<sup>[12]</sup>.

워크스페이스 클래스는 워크스페이스에 솔리드를 추가하거나 삭제하는 일을 수행하는 클래스이다. 뷰 클래스는 워크스페이스에 솔리드 객체들을 디스플레이하는 추상클래스로서 평행투영 뷰 클래스와 원근투영 뷰 클래스는 뷰 클래스로부터 상속받으며, 워크스페이스에 여러 방향의 뷰를 나타낸다.

솔리드의 기본입체로는 육면체, 원기둥, 원뿔, 토러스, 피라미드, 구 등이 있다. 이들 기본입체에 대하여 INTERSECT, DIFFERENCE, UNION과 같은 부울리안 연산을 수행하여 솔리드 모델을 만들 수 있게 된다. 솔리드 객체에 대하여 TRANSLATION, SCALING, REFLECTION, ROTATION과 같은 변환을 수행할 수 있다. 뷰 제어와 렌더링 및 애니메이션은 Java 3D에서 제공하는 풍부한 3차원 그래픽 라이브러리를 이용한다.

##### 3. 오브젝트의 생성

장면그래프(Scene Graph)를 사용하여 오브젝트를 생성함으로써 부분적인 오브젝트를 만들어 쉽게 결합할 수 있으며, 오브젝트를 계층적으로 제어하여 조

작할 수 있는 장점을 지니고 있다. 그림 6은 오브젝트의 계층 구조를 보여주고 있다<sup>[13]</sup>.

Java 3D API의 Scene Graph를 이용하여 visual object는 도형의 요소 Geometry와 색깔, 재질 등의 외형 속성을 나타내는 Appearance로 나타낼 수 있고 여기에 카메라, 조명 등 view branch group과 결합시킬 수 있다.

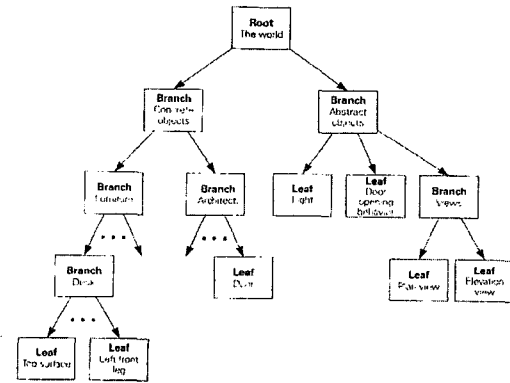


그림 6. 오브젝트의 계층구조

##### 4. 오브젝트의 변형

Scene Graph 내에서의 오브젝트 변형은 TransformGroup 내에 기하도형을 생성한 후 이동, 확대, 축소를 위한 Transform3D 클래스의 추가로 이루어진다.

오브젝트의 이동은 기본적으로 평행이동으로 이루어진다. Transform3D의 메소드 내에서 부동소수점 형식의 x, y, z 좌표의 입력으로 수행된다.

오브젝트의 확대와 축소는 Scale 메소드에 의해 이루어지는데, 주어진 벡터의 값만큼 x, y, z 축으로 균일하지 않은 크기로 수행된다.

오브젝트의 회전변환은 회전축의 명시에 의해 원점에 대해 z축을 기준으로  $\theta$  만큼 회전변환을 수행하게 된다.

##### 5. Picking

3D 공간에서의 picking은 Z축으로 가장 앞에 놓인 오브젝트를 picking하도록 한다. 그림 7은 마우스에서 ray를 쏘아 첫 번째 비행기의 오브젝트를 picking하는 것을 보여 준다<sup>[13]</sup>.

마우스포인터에서의 광선이 오브젝트와 겹치는 것을 찾는 방법은 컴퓨터 내부적으로 많은 산술연산을 요구하고 있다. Java 3D에서는 scene graph를 사용하므로 어떤 노드에 picking을 할 수 있다. 노드를 picking하는 경우에는 많은 계산을 필요로 하

지 않으며 그림 8처럼 하위 오브젝트도 따라서 picking되는 효과를 가져오게 된다.

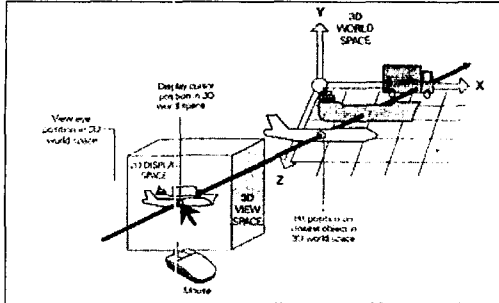


그림 7. 3D World의 object picking

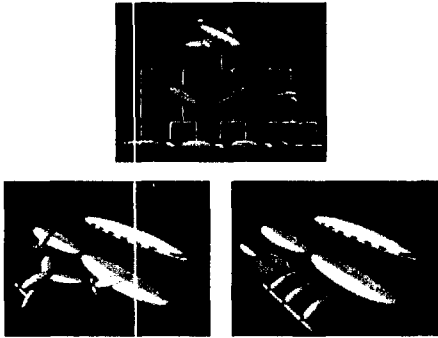


그림 8. 하위 오브젝트의 picking

Java 3D를 이용하여 3차원 가상 공간에서 효과적으로 오브젝트를 picking하기 위해 다음과 같은 방법을 사용하였다<sup>14)</sup>.

첫째 방법은 scene graph node의 attribute와 capability를 이용하는 것으로, 노드에 setPickable() 메소드를 사용하여 true와 false값에 따라 선택한 노드의 하위구조까지도 intersection연산을 하지 않도록 한다.

```

Node Method
extends: SceneGraphObject
subclasses: Group, Leaf
void setBounds(Bounds bounds)
// 노드의 geometric bounds를 설정한다
void setBoundsAutoCompute(boolean autoCompute)
// 노드 on/off의 geometric bounds를 자동계산하게 한다
setPickable(boolean pickable)
// true로 설정하면 이 노드는 picking이 되고, false면
// 이 노드와 하위 노드는 모두 picking되지 않는다
    
```

둘째는 bounds를 설정하는 것으로 복잡한 도형에 대해 일정한 영역을 설정하여 연산을 빠르고 간단하게 하도록 한다.

```

PickBounds
PickBounds()
// PickBounds 생성
PickBounds(Bounds boundsObject)
// 특정한 파라미터를 주어 PickBounds 생성
Bounds get()
// PickBounds로부터 boundsObject를 얻는다
void set(Bounds boundsObject)
A // boundsObject를 이 PickBounds 안으로 설정한다
    
```

셋째는 pick testing의 scope를 한정하여 intersection을 찾기 위해 모든 부분에 대해 연산하는 것을 줄일 수 있다.

### 6. Concurrency

공유작업공간에서는 협동작업의 일관된 상태를 유지하면서 그룹의 참여자들이 작업을 수행할 수 있도록 지원해야 한다. 이런 협동 작업의 일관성을 유지하기 위해서는 동기화기법이 필요하다. 멀티 뷰를 통한 협력시스템에서의 뷰 동기화(View Synchronization) 유지를 위해 참여자의 조작은 서버로 전달되고 서버는 그룹의 각 참여자들에게 브로드 캐스팅하므로 WYSIWIS (What You See Is What I See)를 유지하도록 한다.

협동 작업의 동시성 제어를 위해 ordering과 locking을 이용하는 방법이 일반적이다. ordering방법은 여러 참여자의 작업을 일정한 순서대로 계속 적용시켜 나간다. locking방법은 공유객체의 작업 권한을 한 참여자에게 줌으로써 일관성을 유지하는 방법이다. 본 시스템에서는 협동 작업의 객체를 순서대로 제어하는 것이 아니라 동시에 제어해야 하므로 후자인 locking방법을 사용한다.

locking은 locked와 unlocked의 두 가지의 상태로 1과 0의 값 중 어느 하나를 가진다. 하나의 lock은 하나의 오브젝트와 연관된다. 오브젝트 X에 관해 lock(X)=1이면 이 오브젝트를 요청한 action은 오브젝트 X에 접근 할 수 없다. lock(X)=0이면 오브젝트 X의 접근이 가능하다.

두 개의 연산 lock과 unlock은 오브젝트에 어떤 action을 취할 때 반드시 포함되어야 한다. 참여자의 어떤 action이 오브젝트 X에 접근하려 할 때에는 먼저 lock(X)연산을 실행한다. 이 때 lock(X)=1이면 이 action은 기다리고, lock(X)=0이면 이 action은 lock(X)의 값을 1로 고정시키고 오브젝트 X에 접근한다. 이 action이 오브젝트 X의 사용이 끝나면 unlock(X)연산을 실행하여 lock(X)의 값을 0으로 고정시킨다. 이 때 다른 참여자나 다른

action이 오브젝트X에 다시 접근할 수 있게 된다<sup>[15]</sup>.

하나의 action만이 특정 오브젝트에 대해 lock을 걸고 사용하기 때문에 협동작업에 있어서는 너무 제약적이다. 어떤 action이 오직 읽기 위해 한 오브젝트에 접근한다면 여러 action이 이 오브젝트에 병행적으로 접근하더라도 문제가 없을 것이다. 그러나 어떤 action이 오브젝트에 대해 변형을 가하거나 저장하려고 한다면 그 action이 오브젝트X에 대해 독점적 접근을 가져야 한다. 읽기만을 위해 병행접근을 허용한다면 두 가지 형태의 lock연산이 필요하다.

① 공유로크 (Shared Lock (lock-S)): 어떤 action이 오브젝트X에 대해 공유로크(Lock-S)를 걸면 이 오브젝트에 대해 읽어 올 수는 있으나 기록할 수는 없다. 이 때 이 오브젝트X에 대해 다른 action도 공유 lock를 걸 수가 있다.

② 전용로크 (Exclusive Lock(lock-X)) : 어떤 action이 오브젝트X에 대해 전용로크(Lock-X)를 걸면 참여자는 이 오브젝트에 대해 읽고 기록할 수 있다. 이 때 이 오브젝트X에 대해 다른 action은 어떤 lock도 걸 수가 없다.

오브젝트를 생성한 사람만이 삭제하거나 저장할 수 있도록 한다. 오브젝트의 action에 따른 lock table은 표 1과 같다.

표 1. Object의 Action에 따른 Lock Table

ACTION	LOCK	
	lock-S	lock-X
create	○	×
open	○	×
delete	×	○
translation	○	×
zooming	○	×
rotation	○	×
save	×	○

오브젝트 X에 대해 공유lock이 필요한 action은 lock-S(X)연산을 실행해야 하고, 전용lock이 필요한 action은 lock-X(X)연산을 실행해야 한다. 오브젝트 X에 대한 잠금을 풀기 위해서는 unlock(X)연산을 수행하면 된다. 참여자가 오브젝트를 사용할 때 다음과 같은 locking규정을 따르도록 한다.

1. 참여자가 오브젝트 X에 대해 읽거나 조작하고자 할 때는 lock-S(X)나 lock-X(X)연산을 먼저 실행

한다.

2. 참여자가 오브젝트 X에 대해 저장하고자 할 때는 lock-X(X)연산을 먼저 실행한다.
3. 참여자가 lock-S(X)나 lock-X(X)연산을 하려고 할 때, 오브젝트 X가 이미 양립될 수 없는 lock이 걸려 있으면 그것이 unlock이 될 때까지 기다려야 한다.
4. 참여자가 모든 action의 실행을 종료하기 전에 lock을 건 오브젝트에 대해 반드시 unlock(X)를 실행해야 한다.
5. 참여자가 lock을 걸지 않은 오브젝트 X에 대해 unlock(X)를 실행할 수 없다.

## V. 솔리드 모델러의 구현

### 1. 인터페이스

사용자 인터페이스는 일반적인 HTML문서와 Java 애플릿을 사용한다. 기본적인 통신프로토콜은 HTTP를 사용하며 설계데이터의 교환은 소켓을 이용한다. 솔리드 모델의 X-Y평면, Y-Z평면, X-Z평면에 대한 투영도, 3차원 모델에 대한 평행투영도와 원근투영도를 4 개의 창에 표시한다. 툴박스를 통해 기본 도형을 만들 수 있고, 정밀한 객체를 생성하기 위해 수치입력으로 객체를 만들 수 있다. 원만한 상호작용을 위해 채팅 창을 별도로 두고 대화를 나눌 수 있다. 3장의 그림 5는 본 시스템의 공유작업 공간의 인터페이스를 보여주고 있다.

### 2. View Port

사용자가 오브젝트들을 화면상에서 둘러보면서 작업할 수 있게 해주는 뷰포트를 구성하고 있는 파일들은 다음과 같다.

#### ● ViewPort.java

화면을 front, top, left, right 의 방향에서 표현하도록 눈의 위치를 이동시킨다.

#### ● MakeUniverse.java

- Virtual Universe를 생성한다.
- 기본 Light를 설치한다.
- Object를 4개 방향의 Canvas3D에 표현하고, 마우스를 통한Transform, Appearance의 변형이 가능하도록 설정한다.
- 마우스를 통한 Object 클릭시 MouseBehavior Call, MouseEventBehavior의 내용이 실행되도록 설정한다.

### 3. 마우스를 통한 Object의 handling

본 시스템에서는 마우스의 조작을 통해 오브젝트를 변환시킬 수 있다. 이때 마우스의 이벤트를 받아들이는 코드가 필요하고 받아들여진 마우스의 동작에 의해 화면의 셋팅박스에 변화된 데이터의 값이 표시되게 된다.

- MouseBehaviorCall.java
  - 마우스를 통해 Object의 Transform을 변형할 때마다 수행된다.
  - Transform이 변경될 때마다 화면 오른쪽의 SettingBox의 Position, Scale, Rotation의 정보를 수정
- MouseEventBehavior.java
  - 마우스로 Object를 선택하였을 때 마다 수행된다.
  - 오른쪽 SettingBox의 내용을 초기화한다.
  - 선택된 Object의 Appearance 정보를 얻어내어 SettingBox의 각종 Color 값을 초기화한다.
- SetTransform.java
  - SettingBox의 Position, Scale 값이 사용자에게 의해 변경될때, 해당 Object의 Transform을 변경하는 부분이다.

### 4. Primitive Object의 생성

오브젝트를 표현하는 방법에는 여러 가지가 있지만, 본 시스템에서는 Box, Sphere, Cylinder, Cone, Tetrahedron, Octahedron 등의 라이브러리화된 여러 도형들을 변형시켜 구성하는 방법을 채택하고 있다.

- myQuadGrid.java, myTriGrid.java
  - QuadArray, TriangleStripArray를 이용하여 거리가 일정치 않은 MxN 벡터스 매트릭스 표현하며, 다른 Object 생성시 사용한다.
- myBox.java, mySphere.java  
myCylinder.java, myCone.java  
myTetrahedron.java, myOctahedron.java
  - Shape3D를 상속받아 Box, Sphere, Cylinder, Cone, Tetrahedron, Octahedron 모양의 Object를 생성한다.

### 5. 파일의 loading과 saving

클라이언트의 공동작업에 의해 구성되어진 오브젝트들은 서버에 저장되게 된다. 그뿐만 아니라 다른 어플리케이션에 의해 만들어진 파일을 Load하는 기능을 가지고 있다. 이때 Load할 수 있는 파일들

로는 obj, wrl 파일형식이 있다. 그림 9는 다른 파일 포맷에서 로딩한 예이다.

- wwwCADServer.java
  - 클라이언트(wwwCAD)가 요청한 파일의 내용을 전송하거나 클라이언트가 보내온 작업들을 저장하는 부분이다.
- FileLoad.java
  - 서버에 저장된 파일을 Loading한다.
- FileSave.java
  - 작업한 내용을 서버에 Saving한다.

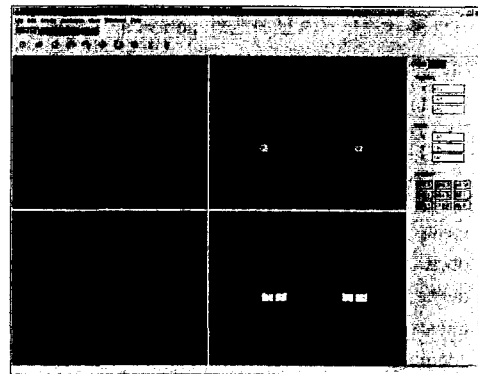
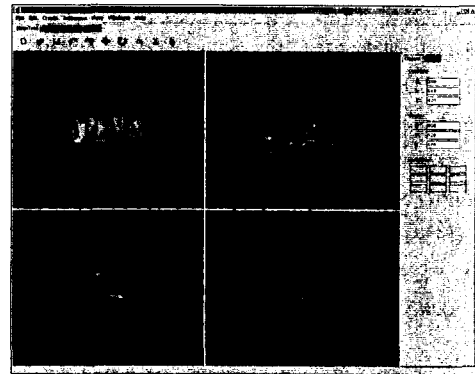


그림 9. 파일 로딩의 예 (\*.obj, \*.wrl)

### 6. 채팅 창

공동작업을 하는 사용자들의 의사소통을 돕기 위해 채팅기능을 제공한다. 이때는 소켓클래스를 사용하여 채팅서버와 클라이언트를 동작하게 한다.

- ChatApplet.java
  - 채팅 클라이언트를 만들어주는 애플릿이다.
- ChatServer.java
  - 채팅 서버를 만들어 준다.
- ChatHandler.java
  - 서버 부분에서 메시지를 각 클라이언트

에 분배해주는 부분을 담당한다.

## VI. 요약 및 결론

본 논문에서는 플랫폼에 구애받지 않으며 특정한 3차원 그래픽 소프트웨어 없이 웹 상에서 설계할 수 있는 웹 기반 협동설계를 위한 솔리드 모델러를 제안하였다. Java 3D를 이용하여 개발함으로써 많은 그래픽 라이브러리들의 특징과 편리함, 그리고 다른 라이브러리에서 사용하지 않는 뷰, 렌더링, 애니메이션 등과 같은 새로운 기술들을 포함하고 있다.

Java언어와 인터넷 웹을 이용하게 때문에 특정 하드웨어나 네트워크를 요구하지 않으며, 특정 OS나 응용소프트웨어에 구애받지 않고 웹 상에서 원격 공동설계시스템 서버에 접속하여 공유된 가상공간에서 공동작업이 이루어지도록 한다.

클라이언트는 자바 애플릿을 통해 웹 상에서 접근하고 서버는 자바 애플리케이션으로 접속을 통제하는 접속관리자, 작업 그룹의 동기화를 유지하며 공유작업공간을 확보하는 작업관리자, 그리고 3차원 도형 객체를 만드는 솔리드 모델러로 이루어져 있다. VRML과 Java3D를 사용해 웹 상에서 쉽게 3D 도형 객체를 생성하고 수정하고 전송하거나 저장할 수 있으며, 채팅을 통해 인터랙티브한 작업을 가능하게 하고 있다.

Java 3D를 이용해 오브젝트의 picking을 scene graph의 노드에 picking 속성을 주는 방법, bounds를 설정하는 방법, picking test의 범위를 한정하는 방법을 사용하여 computation의 부담을 줄이고 효과적인 picking이 이루어지도록 했다. 이는 계층구조를 가진 3D 도형에서 하위 계층의 모든 노드를 제어할 수 있는 큰 장점을 보여준다. 그리고 picking된 오브젝트에 대해 어떤 조사를 가할 때 협동작업에 참여한 사람의 action에 따라 오브젝트에 Shared lock과 Exclusive lock을 적절하게 사용하도록 하였다. 이는 참여자들이 여러 오브젝트를 서로 공유하면서 방해받지 않고 효과적인 협동설계가 가능하게 해준다.

기존의 상용 CAD시스템은 웹에서의 사용이 불가능하며 협동작업이 이루어질 수 없는 반면에 본 솔리드 모델러는 모두 가능하다. 협동작업시스템은 대개 공유객체가 문자 중심이나 본 솔리드 모델러는 3D오브젝트를 생성하고 변형시킬 수 있다. 그리고 가격이나 업그레이드 면에서도 다른 시스템과 비교해서 매우 우수한 것임을 알 수 있다.

앞으로 네트워크의 속도와 안정성이 확보된다면 본 웹기반 협동CAD시스템은 CAD분야에 있어서 획기적인 변화를 가져오게 할 것이다. 여기에 렌더링이나 애니메이션 기능을 추가한다면 더 큰 진가를 발휘하게 되리라고 본다.

## 감사의 글

본 연구는 과학기술부, 한국과학재단지정 여수대학교 "설비자동화 및 정보시스템연구개발센터"의 지원에 의해 연구되었으며 이에 감사드립니다.

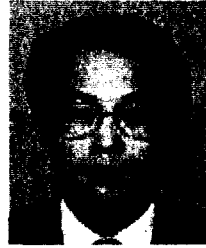
## 참고 문헌

- [1] 윤보열, 김응곤, "웹기반 솔리드 모델러의 설계", 한국정보처리학회 추계학술발표논문집, 1999.10.
- [2] 김관연, "WWW의 확장과 엔지니어링 환경의 변화", CAD&그래픽스, '1997. 4., pp.114.
- [3] Stepen C.F. Chan, Vincent T.Y. Ng, Albert S.F. Au, "A solid modeling library for the World Wide Web", Computer Networks and ISDN Systems 30, 1998 pp. 1853-1863
- [4] Subra Mohan, "The Fourth Generation of 3D Graphics APIs has arrived", [http://java.sun.com/products/java-media/3D/collateral/wp\\_mktg/j3d\\_wp.pdf](http://java.sun.com/products/java-media/3D/collateral/wp_mktg/j3d_wp.pdf), 1998
- [5] F. Faure, C.Faistnauer, G.Hesina, "Collaborative animation over the net", IEEE p107-116, 1999
- [6] Icgiri Hagiwara and Shinsuke Noda, "Homotopical Modeling as the Basis of New CAD Standard Homotopy CAD for Collaboration Engineering", IEEE p231-237
- [7] 신승현, "컴퓨터지원 협동작업 연구", <http://www.mococo.co.kr/press/press02.html>
- [8] WebCAD WWW, "WebCAD", <http://cad.berkeley.edu/>
- [9] 서울대 공학연구소, "네트워크와 가상현실을 이용한 공동설계/조립평가 시스템 개발", 최종 연구개발결과 보고서, 정보통신부, 1998.
- [10] Mathias Nusch, Bernhard Jung, "CAD on the World Wide Web : Virtual Assembly of Furniture with BEAVER", Faculty of Technplogy, University of Bielefeld, 1998
- [11] M. Mantyla, "An Introduction to Solid Model-



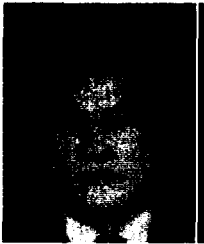
- ing”, Computer Science Press, Rockville, MD, 1988.
- [12] Stephan C.F. Chan, Vicent T.Y. Ng, Albert S.F. Au, “A Solid modeling library for the World Wide Web”, Computer Networks and ISDN Systems 30, 1998.
- [13] Jon Barrilleaux, “3D User Interfaces with Java 3D”, Manning Publications Co., 2001
- [14] Bouvier(00) Dennis J. Bouvier, “Getting started with the Java 3D API”, Sun microsystems, 2000, <http://java.sun.com/products/java-media/3D/collateral>
- [15] 이석호, “데이터베이스 시스템”, 정익사, 1995.

윤 보 열(Bo-yul Yoon)



1984 : 전남대학교 문학사  
1988 : 조선대학교 공학석사  
2002 : 순천대학교 이학박사  
1984 ~ 현재 : 순천매산고등학교  
교사  
<주관심 분야> 컴퓨터그래픽스, 멀티미디어, WBI

김 응 곤(Eung-kon Kim)



1980년 2월 : 조선대학교  
공학사  
1987년 2월 : 한양대학교  
공학석사  
1992년 2월 : 조선대학교  
공학박사

1984년 8월~1986년 8월 : 금성반도체(주) 연구원  
1987년 3월~1991년 2월 : 국방과학연구소 선임연구원  
1991년 3월~1993년 2월 : 여수수산대학교 컴퓨터공학과 전임강사  
1997년 3월~1998년 2월 : 미국 University of California, Santa Cruz Post Doc.  
1993년 3월~현재 : 순천대학교 컴퓨터과학과 부교수  
<주관심 분야> 컴퓨터그래픽스, 영상처리, 가상현실