

# STARS

University of Central Florida  
**STARS**

---

Institute for Simulation and Training

Digital Collections

---

1-1-1998

## Web-based Distributed Simulation

Kuo Chi Lin

Find similar works at: <https://stars.library.ucf.edu/istlibrary>  
University of Central Florida Libraries <http://library.ucf.edu>

This Research Report is brought to you for free and open access by the Digital Collections at STARS. It has been accepted for inclusion in Institute for Simulation and Training by an authorized administrator of STARS. For more information, please contact [STARS@ucf.edu](mailto:STARS@ucf.edu).

---

### Recommended Citation

Lin, Kuo Chi, "Web-based Distributed Simulation" (1998). *Institute for Simulation and Training*. 210.  
<https://stars.library.ucf.edu/istlibrary/210>



INSTITUTE FOR SIMULATION & TRAINING

*Web-Based Distributed Simulation*

Sponsor: Air Force Office of  
Scientific Research

Contract Number:  
98-0822

P.I.: Dr. Kuo-Chi Lin

Prepared by:  
Institute for  
Simulation & Training  
University of Central Florida  
Orlando, Florida

Prepared for:  
Air Force Office of  
Scientific Research



IST-TR-98-14

B329

# WEB-BASED DISTRIBUTED SIMULATION

Kuo-Chi Lin  
Institute for Simulation and Training  
University of Central Florida  
Orlando, FL 32826

## Abstract

The multi-user virtual environment is a computer-generated synthetic world on the network where users can navigate and interact with each other in the same virtual world. The World Wide Web (WWW) environment is a client/server architecture composed of various server sites to form a global database web over the Internet. Since the existing WWW browsers have difficulty to create an interactive multi-user virtual world, the SharedWeb system, a three-dimensional web browsing system, is designed to support the interactions among the clients over the WWW environment. This report describes the design and implementation of SharedWeb.

The SharedWeb system has the following features that are necessary for the seamless integration between a multi-user interactive system and the WWW environment. 1. The user can download a scene file from any WWW server with the HyperText Transfer Protocol (HTTP). 2. The SharedWeb system provides the hyperlink feature to retrieve various media resources that are supported by the WWW environment. 3. The SharedWeb system can handle information exchange between HTML documents and the objects inside a virtual world. 4. The SharedWeb system server itself is an add-on of an existing WWW server. 5. The user can directly enter a virtual world through the WWW server from which the scene file is downloaded.

The SharedWeb system has been successfully tested in both the Local Area Network (LAN) and Wide Area Network (WAN) environment. The SharedWeb system uses the Dead Reckoning technique to reduce the network traffic created by the interactions among the users, and to maintain the interactivity even when the

congested Internet delays the PDUs. After receiving a delayed PDU, proper delay compensation and smoothing are needed to make the interaction realistic.

## 1. Introduction

Virtual environment is a technique to immerse the user into a synthetic world built by the computer or a remote environment electronically transmitted to the computer for the purpose of training, simulation, prototyping, design, or data acquisition. Due to the rapid development of the virtual environment technique, the multiple-user virtual environment has become a popular topic over the years. The multi-user virtual environment is a computer-generated synthetic world on the network where users can navigate and interact with each other in the same virtual world. The gaming industry has developed many multi-user systems over the Internet. An extensive survey of these existing systems can be found in Section 2.

The World Wide Web (WWW) is a project that brings a global information universe to the existence. [1] The Virtual Reality Modeling Language (VRML) project is an attempt to deliver a virtual world into the WWW environment. [2] The original goal of designing VRML language is to allow multiple participants to interact over the Internet. After the introduction of the VRML language, many VRML-supported browsers were developed, which include the WebSpace from SGI, WebOOGL from the Geometry Center, AmberGL from DIVE Lab., WorldView for Intervista Software, Inc. With these browsers, you can view and walk through a virtual world described by the retrieved VRML file. However, none of these browsers has fulfilled the original goal of multiple-user interactivity, even for those browsers that have implemented VRML version 2.0 released in August 1996.

In this report, the author will describe the design and implementation of

SharedWeb, a web browsing system that can support the interaction among clients over the existent WWW environment.

## 2. Multi-user Browser System

Utilizing the existing WWW environment to build a multi-user interactive system, new communication protocols need to be added to provide the interactions among distributed web browsers. Instead of modifying the existing HTTP protocols WWW is using, the goal is to design a new multi-user virtual system and seamlessly integrate it with the WWW architecture. For a distributed multi-user system to be seamlessly integrated with the WWW environment, it must have the following characteristics: [3]

1. The user can download a scene file from any WWW server with the HyperText Transfer Protocol (HTTP).
2. The multi-user system must provide the hyperlink feature to retrieve various media resources that are supported by the WWW environment.
3. The multi-user system must be able to handle information exchange between HTML documents and the objects inside a virtual world.
4. The multi-user system server itself is an add-on of an existing WWW server.
5. The user can directly enter a virtual world through the WWW server from which the scene file is downloaded.

In summary, the seamless integration implies that the user can simply download a virtual scene file from the supporting WWW server, navigate through the virtual worlds managed by the WWW servers, and fully utilize the services provided by the

WWW environment without the user's awareness.

The researches on the multi-user virtual environment can be classified into two categories: network-based and web-based virtual reality systems. The network-based system researchers generally focus on providing a large-scale environment with every available network techniques. On the other hand, the web-base system researchers emphasize on providing multi-user virtual environment on top of the existent WWW environment. Although our work is concentrated on designing a multi-user virtual reality system over the existent WWW environment, the techniques from the network-base multi-user system also provide partial solutions to our work.

The WWW environment is a client/server architecture using HTTP protocol. Due to the intrinsic properties of HTTP, the WWW system is a one way street that can not achieve interactivity among users. Hence, new mechanisms and protocols have to be designed to support the multi-user interaction over the WWW environment. The VRML standard is an attempt to deliver the virtual world on the WWW environment, and its original goal is to support the interaction of multiple participants. However, even the VRML version 2.0 [2] failed to define the standard of the multiple user interactions. Therefore, several designers of the web-based multi-user systems have developed their own proprietary methods to support the multi-user interaction. [4~5]

Virtual Society (VS) is a research project of Sony Computer Science Laboratory, Inc. in attempt to define a global architecture and a set of protocols to realize a multi-user interactive 3D environment in a WWW setting. [4] It was originally based on the DIVE platform developed by the Swedish Institute of Computer Science. [6] The VS project has developed a VRML-standard virtual reality system over the WWW environment, called Community Place Browser. [7] The VS architecture, however, is

not seamlessly integrated with the WWW environment. For example, the VS client can not enter a virtual world from the WWW server from which the scene file was downloaded. After a scene file is downloaded from a WWW server, the VS client must send a URL to that WWW server to query the necessary information, such as IP address and port number, of the VS server. In addition, the VS server does not support information exchange between the HTML document and objects inside a virtual world. Hence, VS is an extension of the networked virtual reality system to the WWW environment rather than a merge of the two.

The German National Research Center for Information Technology (GMD) has proposed another type of web-based multi-user system by defining new nodes for VRML standard. [5] Different from the VS project, the multi-user system by GMD does not require a multi-user server to maintain the consistency among distributed objects. To achieve the consistent control of shared objects, GMD proposed a so-called Active Lock mechanism and two extension nodes, Interaction node and Behavior node, for the VRML standard. The research of GMD has been concentrating on extending the VRML standard for the multi-user interaction. Issues of seamless integration discussed in the previous section still require further study.

Scalable Platform for Large Interactive Networked Environment (Spline) by Mitsubishi Electric Research Lab. (MERL) uses yet another approach to integrate the multi-user virtual environment with the WWW environment. [8] Spline is essentially a networked-based multi-user virtual reality system that uses Interactive Sharing Transfer Protocol (ISTP) as its communication protocol. [9] ISTP incorporated several simple HTTP protocols to enable the HTTP server to participate in multi-user interactions. With the help of ISTP, the Spline system provides a WWW-like virtual environment. However, it is not a web-based multi-user system.

Other web-based multi-user systems that are commercially available include Ccpro [10] from Blaxxun interactive Corp., OnLive! Community browser [11] from OnLive! Technologies Inc., OZ Virtual [12] from OZ Interactive Inc. and V\*Realm Multi-user browser [13] from Integrated Data Systems, Inc. All of the above browsers adopt the VRML standard as their virtual scene file format. Each of the above multi-user systems uses its own technique to provide interactivity among distributed users. None of the above browsers is fully integrated with the traditional HTML browser. Active World browser [14] from Worlds Inc. is another kind of multi-user browser that is not a VRML browser, whereas it merges the traditional HTML browser into its own browser. However, it can not handle information exchange between HTML documents and the objects inside a virtual world.

### **3. Design Considerations**

Since the existing WWW browsers have difficulties in creating an interactive multi-user virtual world, the SharedWeb system, a three-dimensional web browsing system, is designed to support the interactions among the clients over the WWW environment. [15]

#### **3.1 Mechanisms to Support the Multi-user Interactivity over WWW**

There are four intrinsic problems that must be solved to support the multi-user interaction over the WWW environment. They are Client-Information Recording, Server-to-Client Callback, Excessive Network Load, and Virtual World Entrance.

##### **3.1.1 Client-Information Recording**

The WWW server uses a request-and-response technique to serve its client's



request. The link between the server and client is established only when a client issues a request to the server. After the requested information is sent, the link is broken and the client is "forgotten" by the server immediately. However, in a multi-user interaction system, the server needs to keep track of all the users. Since the link is broken immediately after the requested information is sent to the client, the WWW server does not keep the records of the clients who have requested information. We call this the Client Information Recording problem.

To solve this problem, each browser must be able to provide a sequence number, called Unique Identification (UID), to uniquely identify itself to the server. Modified from Chin's proposal [16~17], the SharedWeb browser uses the combination of Login name, Time stamp and listen Port number (LTP) as the UID to identify itself. The login name is the nickname of the user to login to a virtual world, the value of the time stamp is the time when the browser was first activated and the listen port number is the port number that the browser used to receive messages. With this UID number, the SharedWeb server can easily identify a new registered user and assigns a User ID to that browser. Moreover, the SharedWeb server will record this UID number for future reference.

### **3.1.2 Server-to-Client Callback**

The link between the server and the client is established only when a client issues a specific request. The WWW server does not actively set up a connection to its clients; therefore, it does not have the ability to forward information from one client to other clients. This is referred as the Server-to-Client Callback problem.

To solve this problem, the SharedWeb server is designed to have the ability of actively set up connection the clients and forwarding information from one user to

other users.

Another technique being widely discussed recently is "Cookies". [18~19] The Cookie technique is based on a two-stage process. First, a "cookie", or a small piece of information, is sent by the Web server and stored in the user's computer without its consent or knowledge. Second, the cookie clandestinely and automatically establishes connection to the Web server. In its current form, Cookies can not be used as the Callback mechanism from WWW server.

### 3.1.3 Excessive Network Load

Since the link between WWW server and client is established only when a request is issued, in addition to the intrinsic network latency, there is a delay associated with establishing the link. A central server is usually used in a web-based multiuser virtual system to mediate the communication among the clients. [5] The network bandwidth will be overloaded very quickly as the number of users increases, and the server will rapidly become an input/output bottleneck. We describe this situation as an Excessive Network Load problem.

Distributed Interactive Simulation (DIS) [20] is a technique developed by the Department of Defense for military training. It is a fully distributed simulation environment with no centralized control. In DIS, a concept called Dead Reckoning is employed to reduce the network traffic problem. [21~22] The Dead Reckoning model uses the current position and velocity/acceleration to predict the future positions. All entities use the same dead reckoning model to update the position of one particular entity to achieve consistency. Each entity is responsible for comparing its own dead reckoning model with the true model. When the error exceeds a predefined threshold value, it needs to inform all entities its new position. SharedWeb can adopt the same

concept. The states of all avatars are updated based on the respective dead reckoning models until the new status change information of one particular avatar is received. The information is sent in the form of a self-contained data package, called Protocol Data Unit (PDU), and forwarded to each participating browser through the SharedWeb server. The PDU is sent to the SharedWeb server only when the deviation of the dead reckoning model from the true model exceeds a predefined threshold. Hence, the bandwidth requirement can be reduced.

### **3.1.4 Virtual World Entrance**

For a communication protocol to support the multi-user interaction over the WWW environment, the seamless integration is important for an adopted virtual environment to take full advantage of the WWW environment. The user has to be able to set up connection to the server without hassle. The problem is referred as Virtual World Entrance.

To solve this problem, we can either embed the multi-user server information into the downloaded scene file or allow the client to issue a query message to the WWW server for the information of the multi-user server. Although the first option is simpler, this method will cause trouble when a multi-user server is crashed and a replacement server is activated. As to the second option, the client has to wait until the replied HTML document is received before it can be connected to the virtual world as Virtual Society from Sony SCL did. [23] Both methods fail to satisfy the criteria of the seamless integration as the outline in Section 2.

The WWW server has to treat the multi-user server as an add-on application in order to integrate the network virtual system into the WWW environment seamlessly. In the WWW architecture, the WWW server uses the Common Gateway Interface

(CGI) to communicate to its add-on application. [24] Hence, a CGI program can be implemented to forward client information from the WWW server to the multi-user server. Using this approach, whether the multi-user server and the WWW server reside in the same host does not matter to the client. In addition, a hierarchy of the multi-user servers can be accordingly implemented. The WWW server passes the login information of a client to a load-monitor program, which will in turn forward the client to one of the multi-user servers without the awareness of the client. In this case, the multi-user server system can have a better load balance and fault tolerance.

### 3.2 Time Synchronization

Time synchronization is another important issue in the multi-user virtual system. The multi-user virtual environment is a synthetic world on the network where users can navigate and interact with each other. In our real world, all people use the same standard time to synchronize their local clocks and interact with each other. In the virtual world, however, the users join the scene from different computers, and the time in each user's virtual world is decided by the respective computer's system time. The time needs to be concordant for all users in the same virtual world so that the event actions are referring to the same clock. It is the Time Synchronization problem.

The Network Time Protocol (NTP) has been used to synchronize the time of a computer client or server to another server or reference time source, such as a radio or satellite receiver or modem. It provides client accuracy typically within a millisecond on LANs and up to a few tens of milliseconds on WANs relative to a primary server synchronized to Coordinated Universal Time (UTC) via a Global Positioning Service (GPS) receiver, for example. Typical NTP configurations utilize multiple redundant servers and diverse network paths, in order to achieve high accuracy and reliability.

Some configurations include cryptographic authentication to prevent accidental or malicious protocol attacks. [25]

The NTP architecture, protocol and algorithms have evolved over almost two decades, with the latest NTP Version 3 and 4 designated an Internet standard. Among the goals of this design are: [26]

1. Optimize the computer clock accuracy and stability, subject to constraints of network overheads and/or telephone toll charges, relative to local and/or remote sources of time.
2. Enhance the reliability by detecting and discarding misbehaving local and/or remote sources and reconfiguring network paths as necessary.
3. Automatically adjust algorithm parameters in response to prevailing network delay/jitter conditions and the measured stability of the computer clock oscillator.

The NTP technique can help the SharedWeb system to achieve time synchronization. All users synchronize their time with SharedWeb server using the NTP technique before they enter the global virtual world. It can ensure that one event happens at the same time for all users in the same virtual world.

#### **4. The SharedWeb System**

With the mechanisms presented in the previous section, the SharedWeb system provides a natural extension of the WWW environment to a multi-user virtual environment. The infrastructure of the SharedWeb system is shown in Figure 1. The nodes of the SharedWeb system can be classified into two types: the server site and

the browser site. The browser site is composed of four modules: Multiple Participants Interface, 3D Render Engine, Chat Phase, and WWW Homepage Viewer modules. The server site is built on top of the existent WWW server with two modules: a CGI program and the SharedWeb server. The server and the browser are communicated by the SharedWeb Communication Protocol (SWCP), which will be fully explored in the next subsection. Each of these modules is responsible for the tasks described as below:

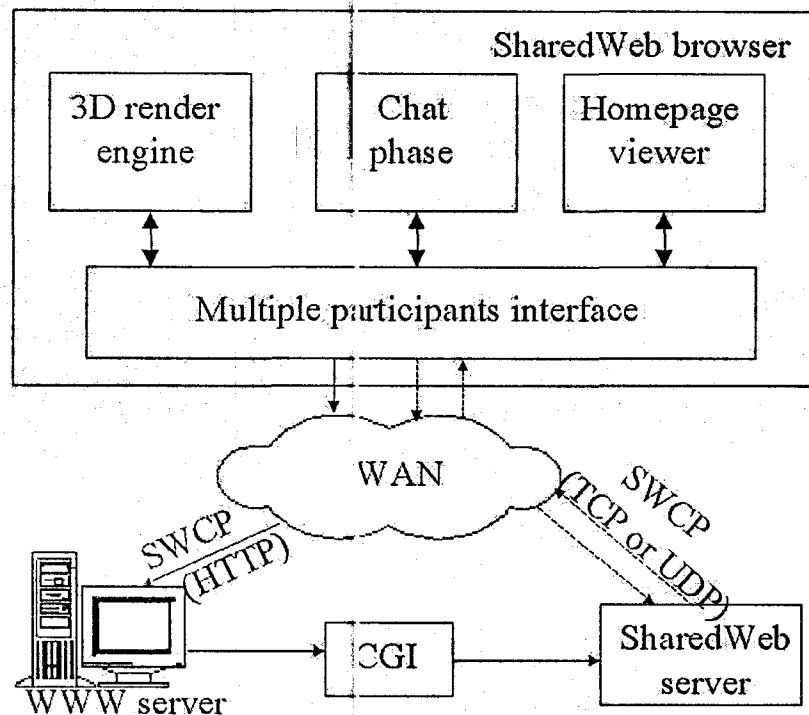


Figure 1. Infrastructure of the SharedWeb system.

#### 4.1 The Browser Site

The Multiple Participant Interface module runs on top of Winsock to manipulate the interaction between the browser and other users. This module is the kernel of a SharedWeb browser. The main functions of this module are to calculate browser's

UID, encapsulate and de-capsulate the PDUs, transmit/receive data to/from the network, and coordinate messages among 3D Render Engine, Chat Phase, and WWW Homepage Viewer modules.

The 3D Render Engine module provides a visual window for the user to explore the virtual world. When a browser retrieves a scene file from the WWW server, it is an ASCII file. The scene file will then be translated into a virtual world database recognized by the 3D Render Engine module.

The Chat Phase module is designed to support text communication among distributed users. This module enhances the interactions among distributed participants and, hence, broadens the applicability of the SharedWeb system. Without this module, the shared virtual world will be a wordless world that makes the user bored easily when playing with the SharedWeb browser.

The WWW Homepage Viewer module is a standard HTML browser. By seamlessly integrating this module into the SharedWeb browser, a user can enter another virtual world by double clicking a hyperlink provided on a Homepage or inside the virtual world. Further, the user can exchange information with each other by using the Homepage and the HTTP protocol.

#### **4.2 The Server Site**

The WWW server is a gateway for a SharedWeb user to enter a virtual world. For the user, he only needs to download a scene file from this WWW server and to login his nickname to enter a virtual world. After the WWW server receives the login information from a browser, the WWW server will pass the received user information to the SharedWeb server through the CGI program mentioned in the previous section.

The SharedWeb server module is the kernel of the entire SharedWeb system.

This program is responsible for mediating the information exchange among the browsers in the same virtual world. That is, all the client information is sent to the server and the server is responsible for routing the received information to the browsers in the same virtual world, with the exception of the chat (see Chat PDU below) among users.

#### 4.3 SharedWeb Communication Protocol (SWCP)

There is no standardized communication protocol for the web-based multi-user virtual system. Different protocols have been proposed by various systems over the years. [4~5][9] The communication protocol includes steps to join a virtual world, to forward messages among distributed users, and to leave the virtual world. On the other hand, the DIS standard is by far the most successful technique that enables a fully distributed virtual environment to be built on a network. The DIS standard [20] defines the data format, PDU, and the communication protocol that employs various PDUs to achieve the spatial consistency among distributed users.

Although the DIS standard has been proven to be an effective method to support the networked multi-user virtual environment on the Internet, it is not designed for the web-based multi-user virtual environment. Inspired by the DIS technique, the SharedWeb system adopts the concept of PDUs to provide the multi-user interaction over the WWW environment. The PDU design focuses fully on the integration of the virtual system into the WWW architecture. Seven PDUs are defined in the SharedWeb system: [27]

1. Initial PDU: This PDU is used to submit the information required to log into a virtual world, and to inform other users that a new user just joined.



2. Acknowledge PDU: When the SharedWeb server receives an Initial PDU from a browser, it will respond with an Acknowledge PDU to the browser.
3. End PDU: When a user wants to exit from the virtual world, it will send an End PDU to notify the SharedWeb server and all other users.
4. Object State PDU: Browser uses this PDU to send the status-change information to the SharedWeb server and other browsers. To reduce network traffic, this PDU is issued only when the dead-reckoned position of the avatar deviates from its actual position by a predefined threshold value.
5. Chat PDU: SharedWeb system supports the peer-to-peer chatting protocol. This is the only PDU that bypasses the SharedWeb server. It can prevent the bottleneck on the SharedWeb server site as the number of user increases.
6. HTTP PDU: This PDU has two functions in this system. First, this PDU allows the user to send a URL address of a network resource, such as an HTML document or a audio clip, to other users. Second, it lets the users control the objects in the virtual world from an HTML document. Whenever data are filled into an HTML document provided by the SharedWeb browser, this information is sent to its own Web server, as shown in Figure 1. The CGI decodes the document and forward the information to the SharedWeb server. The SharedWeb server then packs the information into an HTTP PDU and forwards to other browsers.
7. Error PDU: When an error occurs between the clients and server, users involved are informed using this PDU.

#### 4.4 User Interaction with the SharedWeb System

The user interactions in the SharedWeb system include six steps: Initialization, Login, Update, Chat, HTTP, and End. [28] The SharedWeb browser's start-up page is

shown in Figure 2.

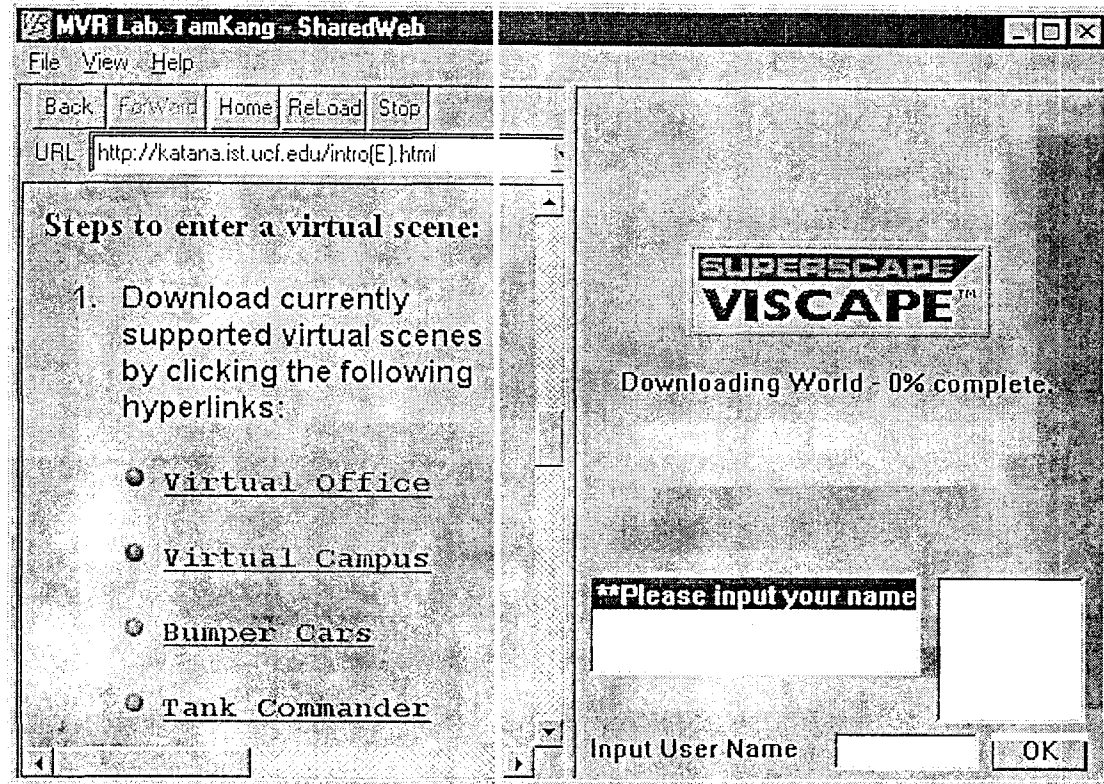


Figure 2. Start-up page of the SharedWeb browser.

#### 4.4.1 Initialization step

This is the first step a user joins the multi-user system. Once launched, the SharedWeb browser (client) shows a homepage with a list of exercises available, as shown in the left window in Figure 2. The user selects the exercise and downloads the virtual world scene file and the related information document.

#### 4.4.2 Login step

The user keys in a nickname of his/her choice in the lower-right "Input User Name" window in Figure 2. After the SharedWeb server receives the Initial PDU, it will register the user and send an Acknowledge PDU back to the SharedWeb browser.

Figure 3 is an example of the browser after the registration is completed.

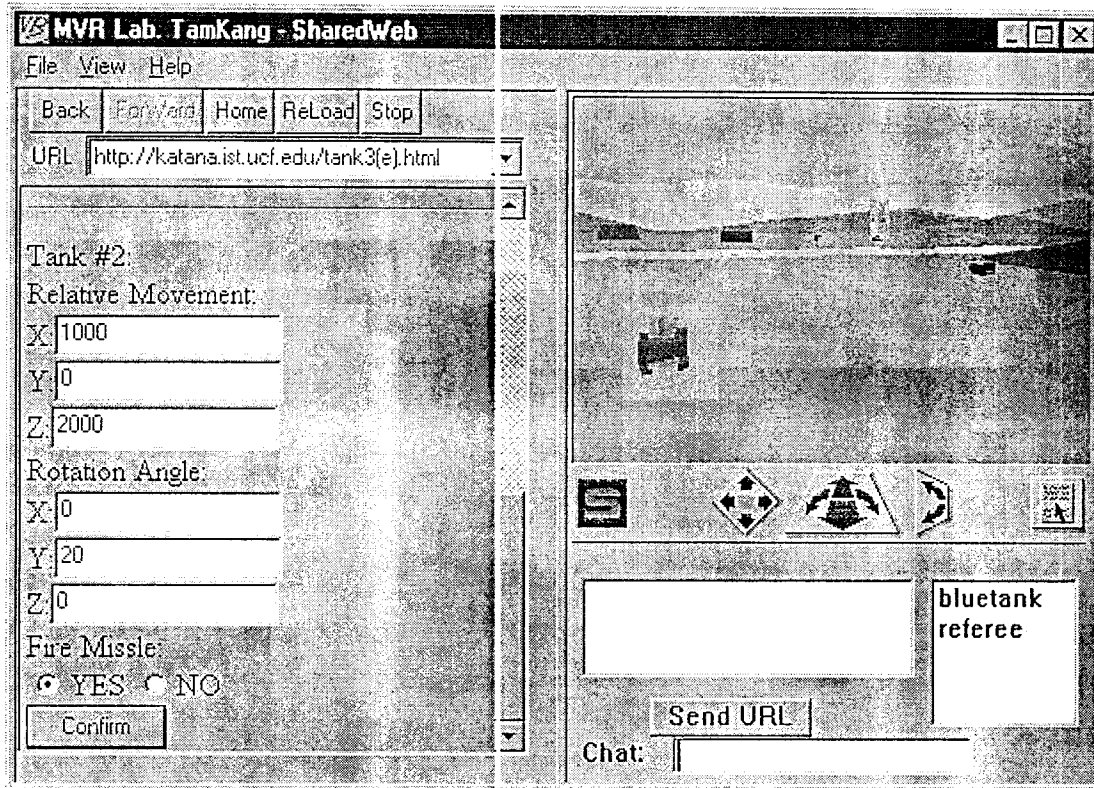


Figure 3. Viewpoint of the red tank commander after registration.

#### 4.4.3 Update step

The user now can start to navigate the avatar in the virtual world. The SharedWeb browser provides a control bar and an HTML form, as shown in Figure 3. The user can use either one to control the avatar. When the dead-reckoned position of the avatar deviates from its actual position by a predefined threshold value, the browser automatically sends an Object State PDU to the server to inform other users in the same virtual world its update position.

#### 4.4.4 Chat step

The user uses a chat window, as shown in Figure 3, to send text messages to any

or all users in the same virtual world. The SharedWeb has adopted the Internet Relay Protocol [29] in its Chat PDU design.

#### **4.4.5 HTTP step**

The SharedWeb system can integrate Web resources into the virtual world. This is achieved by implementing the HTTP PDU. The user can send a URL address to other users in the same virtual world. The SharedWeb browser receiving the HTTP PDU will auto-download this HTML document from the addressed WWW server.

#### **4.4.6 End step**

The user can exit from the virtual world by select "Exit World" under the "File" menu. The browser sends an End PDU to the SharedWeb server.

### **5. SharedWeb System Implementation**

The SharedWeb system is implemented using Microsoft Visual C++ version 5.0. The SharedWeb server runs on Windows NT Server version 4.0 with an embedded WWW server. The SharedWeb browser is currently available as a standalone browser. In the future, it will be implemented as a plug-in application for the Netscape Communicator and Internet Explorer. The user interface of the SharedWeb browser is shown in Figure 2 and 3.

As displayed on the upper-right window of Figure 2, Viscape from Superscape Inc. [30] is used as the 3D Render Engine. When the browser is initiated, the user will get SharedWeb system information in the left HTML window. After the user has successfully logged in a virtual world, the Chat Input Box will be activated automatically and all of the users' names inside the virtual world will be listed in the

User List Window. Further more, a “Send URL” button will emerge at the same time, as shown in Figure 4. The “Send URL.” button is a special function for the user to send a hyperlink resource, such as an HTML document or audio clip, to other users. This function is another feature for the SharedWeb system to be seamlessly integrated into the WWW environment.

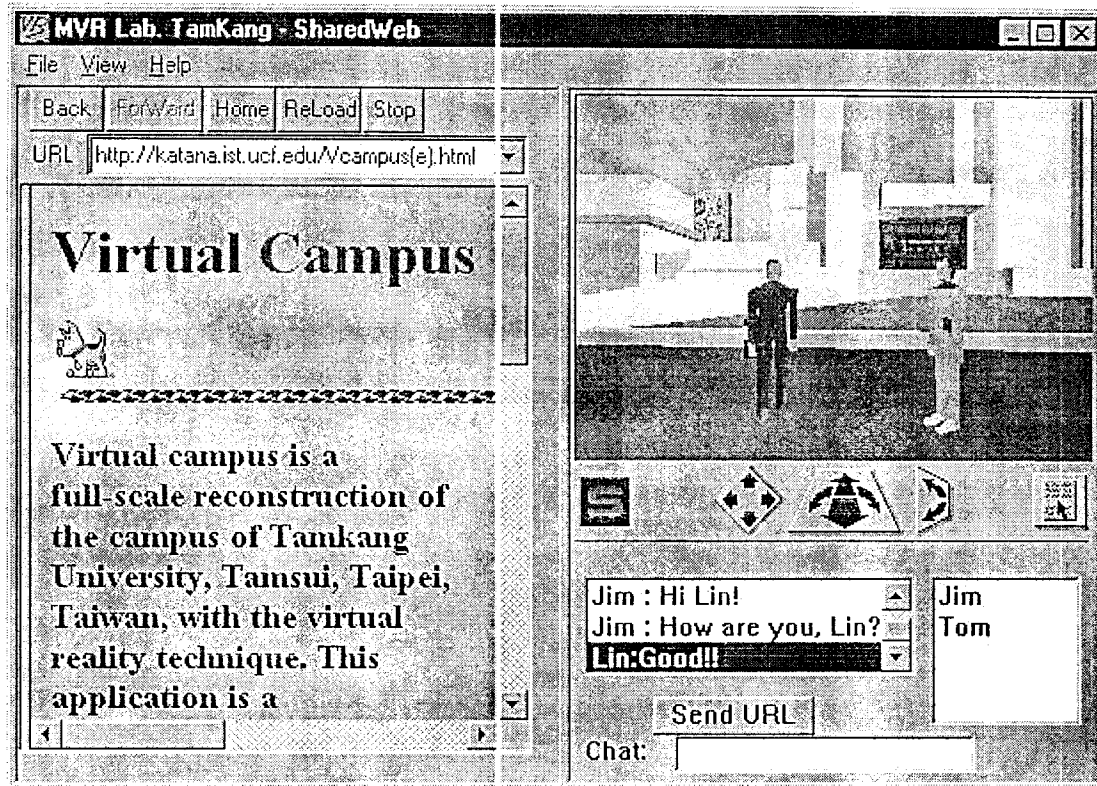


Figure 4. A scene in the virtual campus.

After the “Send URL” button is clicked, it will disappear and a highlighted message will be displayed in Chat Output Window to notify the user. In the meantime, the label of the Chat Input Box is changed to “URL:” for the user to input the URL address of a network resource, such as an HTML document or audio clip. Moreover, the sender selects the receiver(s) by clicking one or more name(s) from the

User List Window. By clicking the O.K button, the specified network resource will then be automatically forwarded to the receiver(s).

To fully explore the features provided by the SharedWeb system, four virtual worlds have been constructed. The following scenarios are the virtual worlds currently supported in the SharedWeb environment. The user can move from one scenario to another without logging out.

### **5.1 Virtual Office**

Virtual Office is an application of the SharedWeb system to demonstrate the benefits of integrating an HTML browser with a 3D browser. The scene of this virtual world is shown in Figure 5. The scenario of this demonstration is the Office of Administration. The user can request an application form in the HTML format. The form requested will appear on the left window in Figure 5. After filling the form, the user can submit it by clicking the "Confirm" button. The chatting capability in this demonstration provides a means for the student to ask questions in case he/she has any questions about the procedures.

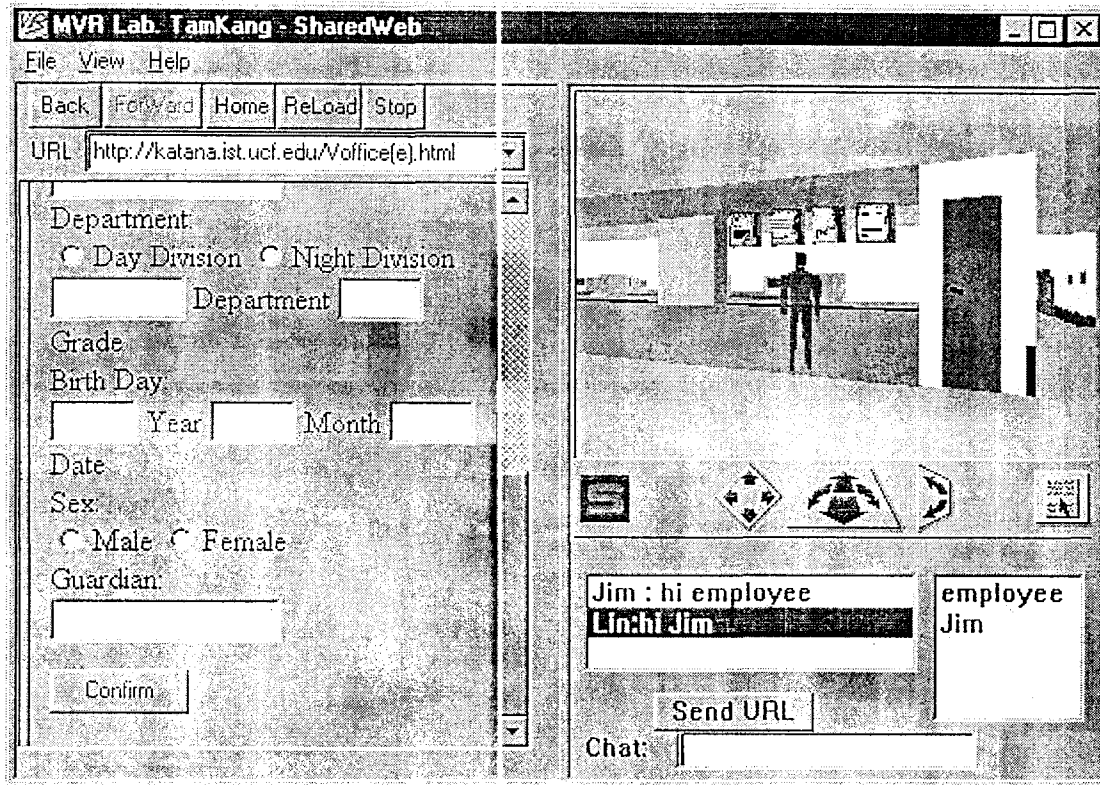


Figure 5. A scene in the Virtual Office and the form requested in HTML form.

Three methods are provided by the SharedWeb system for students to access the application form. The first method is to ask a virtual employee, who also logs in via a SharedWeb Browser, in the virtual office to send him/her the application form. The second way to retrieve the application form is to click a virtual photo on the wall that represents a particular application form. The third way for the student to access any application form is to click the HTML hyperlinks.

## 5.2 Virtual Campus

The virtual campus is modeled with the campus of Tamkang University in Taiwan. The user can walk through the campus and access to different virtual buildings and virtual offices. Figure 4 in Section 5 shows an example of the scenes the user sees in the virtual campus. The Office of Administration in scenario 5.1 is

one of the virtual offices the user can enter.

### 5.3 Bumper Cars

This is a game of bumper cars. Each user controls his/her own car and bumps with other cars, as shown in Figure 6.

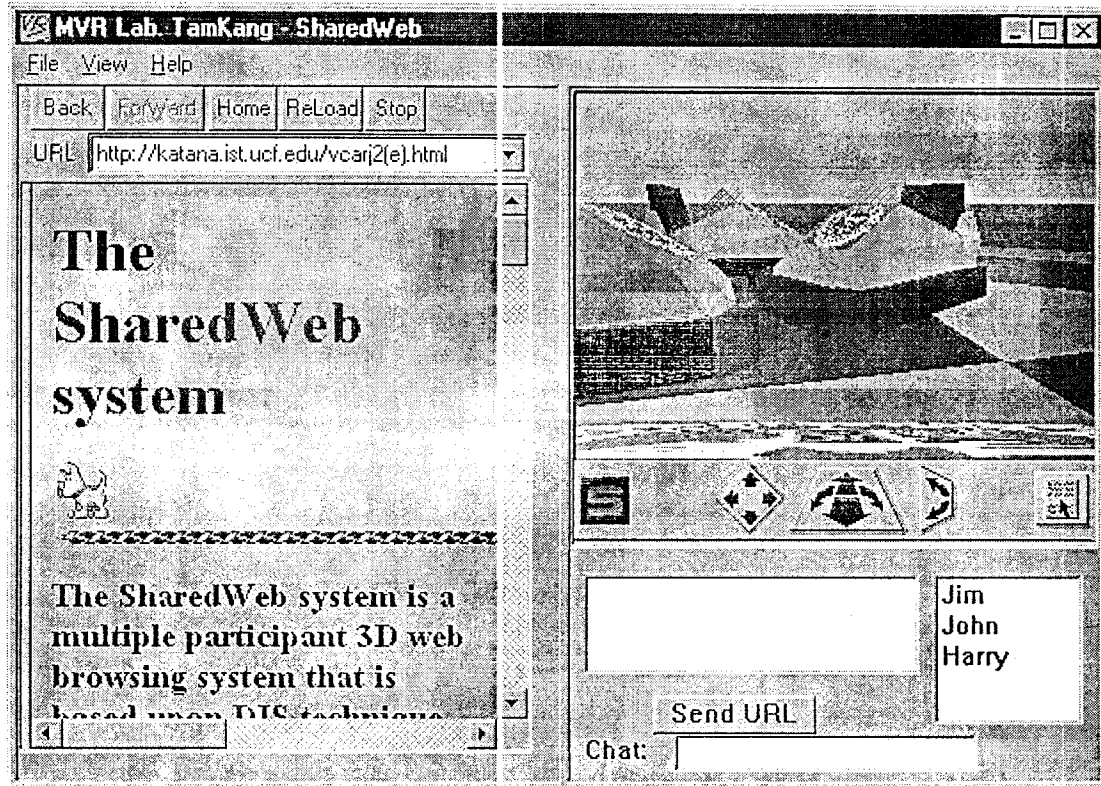


Figure 6. A scene in the Bumper Cars.

### 5.4 Tank Commander

This interactive tank combat game is a prototype a system integrating the 3D technique and WWW environment. This application is a 3D version of the role-playing game (RPG) simulation.

Traditionally, RPG is a two-dimensional graphical display environment. It has an interactive dialog box, or command sheet, for the user to control his character



inside the game. Since the SharedWeb system fully integrates the HTML browser with the virtual reality navigator, the command sheet can be easily implemented by the HTML document so that the content of the command sheet can be dynamically modified. Also, a game master is required for RPG to control the game rules. For the SharedWeb system, the game master was implemented in the SharedWeb server.

The prototype system allows three users to enter virtual battlefield, the "bluetank," the "redtank," and the "referee". Bluetank and redtank can use the HTML document to control two tanks each in the virtual battlefield. Figure 3 in Section 4.4.2 shows the viewpoint of the red tank commander. The user command sheet has automatically downloaded into the HTML browser. The user can input data to control tank movements. In addition, check buttons allow the user to choose whether or not to fire tank missiles. After the user inputs the commands and presses the "Confirm" button, an HTTP PDU is sent to the Sharedweb server. The server then forwards it to all participants and the status of the tank is modified accordingly.

## 6. Conclusion and Future Works

The SharedWeb system has been successfully tested in both the Local Area Network (LAN) and Wide Area Network (WAN) environment. The major concern is the time delay in the information exchange among the SharedWeb browsers. The time delays in the LAN environment are very small. In WAN, however, the network congestion is always a concern. The SharedWeb system uses the Dead Reckoning technique to reduce the network traffic created by the interactions among the users. However, there is not much can be done to deal with the congestion in the Internet itself. After all, unlike DIS--which is running in an isolated network-- all the users in the world share the Internet. We have no control over when other users would create

a heavy traffic in the Internet. In any case, the Dead Reckoning technique can maintain the interactivity even when the congested network delays the PDUs. After receiving a delayed PDU, proper delay compensation and smoothing are needed to make the interaction realistic. [31]

This report has presented the concept of seamless integration of the multi-user system into the WWW environment. With this seamless integration, the existing WWW server can easily support multi-user interactivity, and a shared virtual world can thus be easily constructed. The infrastructure and the communication protocol of this concept have been fully elaborated in this report. In addition, the architecture of a web-based multi-user system, called the SharedWeb system, has also been presented. Under the architecture of the SharedWeb system, the existing WWW server remains intact while supporting the multiple participant interactions. More significantly, the SharedWeb environment is transparent to the users. That is, each user only needs to download a scene file from a WWW server and logs in with his nickname, and the WWW server will then gracefully transfer the user to an appropriate SharedWeb server.

Although the SharedWeb system successfully demonstrates the concept of seamless integration, more researches are still required to improve the multi-user interaction. For example, the consistency control of the shared objects is an important function for computer-supported cooperative work (CSCW) type of applications. [32] Moreover, in a virtual environment with  $N$  avatars, the current SharedWeb server must "broadcast" a change in any avatar's state to  $N - 1$  browsers. Thus, in the worst case, the server must send  $N^2 - N$  PDUs to the network. This approach yields  $O(N^2)$  complexity for updating messages and could exhaust the computation power of a host even before it saturates the network. To overcome this problem, the concept of

spatial culling [33] is currently under investigation for the SharedWeb system.

Multi-server architecture of SharedWeb system is another important future work.

This architecture can further scale up the number of participants as well as provide the fault tolerance feature on the server site.

## References

- [1] T. Berners-Lee, et al., "The World-Wide Web", *Communication of the ACM*, Vol. 37, No. 8, August 1994, pp. 76-82.
- [2] "The Virtual Reality Modeling Language Specification Ver. 2.0", August 1996, available on line at <<http://vrmml.sgi.com/moving-worlds/spec>>
- [3] J. Y. Huang, J. L. Chang, C. W. Li and K. Lin, "Design of a Multiple Participant 3D War Game Environment over WWW", *Part of the SPIE Conference on Enabling Technology for Simulation Science II*, Orlando, Florida, April 1998, pp. 303-312.
- [4] Y. Honda, K. Matsuda, J. Rekimoto, and R. Lea, "Virtual Society: Extending the WWW to Support a Multiuser Interactive shared 3D environment", *Procs. VRML'95*, San Diego, Calif., Aug. 1995. available on line at <<http://www.csl.sony.co.jp/person/roder.html>>
- [5] W. Broll, "VRML : From the Web to Interactive Multi-user Virtual Reality", *Proc. Int'l Workshop MVD'95*, San Diego, Calif., Aug. 1995, available on line at <<http://orgwis.gmd.de/~broll/papers/MVD95.ps.gz>>
- [6] O. Hagsand, "Interactive Multi-user Ves in the DIVE System", *IEEE Multimedia*, Vol. 3, No. 1, Spring 1996, pp. 30-39.
- [7] "Community Place Browser", available on line at <<http://www.sonypic.com/vs>>
- [8] R. C. Waters, et al., "Diamond Part and Spline: Social Virtual Reality with 3D Animation, Spoken Interaction, and Runtime Extendability", *Presence*, Vol. 6, No. 4, August 1997, pp. 461-481.
- [9] R. C. Waters, D. B. Anderson, D. L. Schwenke, "The Interactive Sharing Transfer Protocol Version 1.0", *Tech. Report*, Mitsubishi Electric Research Lab, Cambridge, Mass., 1997, available on line at <<http://www.merl.com/reports/index.html/TR97-10>>
- [10] "Blaxxun Community Client", available on line at <<http://ww3.blacksun.com/>>
- [11] "OnLive! Community Browser", available on line at <<http://www.onlive.com>>
- [12] "OZ Virtual", available on line at <<http://www.oz-inc.com/ov>>
- [13] "V\*Realm Multi-User Browser", available on line at <<http://www.ids-net.com/ids/vrealm>>
- [14] "Active Worlds", available on line at <<http://www.activeworlds.com/>>
- [15] J. Y. Huang, C. T. Fang-Tson, J. C. Chen, and F. B. Wang, "SharedWeb -- A Shared Virtual Environment Over World Wide Web", *Pacific Graphics'97*, Korea, October 1997, pp. 178-189.
- [16] J.Y. Huang, F.B. Wang, W.H. Hsu, and J.F. Chen, "Usage of DIS Technique to Create an Interactive WWW Environment", *14 th DIS Workshop*, Orlando, Florida, March 1996, pp. 201-210.
- [17] C. Chee-Kai, "MOLTIP-UID and R-HTTP Specification", available on line at

- <<http://www.ontc.ncb.gov.sg/staff/cheekai/html/moltip-techspe.html>>
- [18] R. Sikorski and R. Peters, "A privacy Primer for the Web: Spam, Bread Crumbs, and Cookise", *Jama. The journal of the American medical association*, APR 1998 v279 n15 pp. 1219.
  - [19] R. Schwarz, "Manage Your Cookies With VB6 WebClasses", *Visual basic programmer's journal*, OCT 1998 v8 n12 pp. 88.
  - [20] IST, "Standard for Distributed Interactive Simulation--Application Protocols, Version 2.0", *IST Report No IST-CR-94-50*, March, 1994.
  - [21] K. C. Lin, "Dead reckoning and Distributed Interactive Simulation", *Proc. of SPIE's International Symposium on Aerospace, CR58*, Orlando, FL, April 1995, pp. 16-36.
  - [22] K. C. Lin and D. Schab, "Study on the Network Load in Distributed Interactive Simulation", *AIAA Journal of Aircraft*, Vol. 32, No. 6, 1995, pp. 1392-1394.
  - [23] R. Lea, Y. Honda, and K. Matsuda, "Virtual Society : Collaboration in 3D spaces on the Internet", available on line at  
<<http://www.csl.sony.co.jp/project/Vs/index.html>>
  - [24] "The Common Gateway Interface", available on line at  
<<http://hoohoo.ncsa.uiuc.edu/cgi/>>
  - [25] "The Network Time Protocol (NTP) Distribution", available on line at  
<[http://www.eecis.udel.edu/~ntp/ntp\\_spool/html/](http://www.eecis.udel.edu/~ntp/ntp_spool/html/)>
  - [26] D. L. Mills, "Clock Discipline Algorithms for the Network Time Protocol Version 4", *Proc. DIMACS Workshop on Networks in Distributed Computing*, March 1997, available on line at <<http://www.eecis.udel.edu/~mills>>
  - [27] J. Y. Huang, C. T. Fang-Tsou and J. L. Chang, "A Multiuser 3D Web Browsing System", *IEEE Internet Computing*, SEP. & OCT. 1998, pp. 70-80.
  - [28] J. Y. Huang, C. T. Fang-Tson, J. L. Chang and A. J. Lee, "Modeling of Multiple User Virtual Reality System with Petri Net Technique", *National Computer Symposium 1997*. In Chinese, December 1997, pp. F-81-F-88.
  - [29] J. Oikarinen and D. Reed, "Internet Relay Chat Protocol", *Internet RFC*, No. 1459, May 1993.
  - [30] "VRT for Windows - User Guide", Superscape Inc., UK, 1996.
  - [31] K. C. Lin, M. Wang, J. Wang, D. Schab, "The Smoothing of Dead Reckoning Image in Distributed Interactive Simulation", *AIAA Journal of Aircraft*, Vol. 33, No. 2, 1996, pp. 450-452.
  - [32] "Guide to Computer Supported Co-operative Working CSCW", available on line at <<http://dougal.derby.ac.uk/andree/cscw6.html>>
  - [33] C. Greenhalgh and S. Benford, "Massive, A Collaborative Virtual Environment for Teleconference", *ACM Trans. On Computer-Human Interaction*, Vol. 2, No. 3, 1995, pp. 239-261.

