INTERNATIONAL DESIGN CONFERENCE - DESIGN 2002
Dubrovnik, May 14 - 17, 2002.

Real Experiences of Virtual Worlds

T.W. Maver, J. Petric, G. Ucelli and G. Conti

Virtual Reality, Collaborative Design and Distributed Environment

1. Introduction

The present use of 3D simulations or more effective virtual worlds has provided the designer with new media capable of storing several levels of information traditionally obtained only with the help of multiple media, usually more time and resource-consuming.
Virtual models in particular can store information about planning issues, geometric design, material choices or even furniture and lighting conditions. This level of representation provides the designer with all the necessary tools to represent an architectural environment and facilitate the research of potentially hidden errors.
The use of Virtual Reality (VR) within the design process has not only enabled the designer to store more information than with the use of the traditional media and to check the design solutions more efficiently but furthermore it has enhanced the level of simulation providing:

- **Immersion**: Users are completely surrounded by the environment.
- **Presence**: Being surrounded the participant has actually the sensation of being *in* the environment. The Virtual Environment becomes then a place on its own and its perception is similar to *ordinary* environments.
- **Interactivity**: This is surely the most important *feature* provided by VR: the environment allows the participant to be involved and the result of the actions done by the participant is visualized in the VE.
- **Autonomy**: Participants are neither constrained in paths nor in views preset by others but have the freedom and autonomy to explore any single part of the environment.
- **Collaboration**: Multiple users are allowed to take part and to interact in the same VE.

The use of VR can also broad the boundaries of traditional perception to experiences of worlds not necessarily real or material or it gives the freedom to safely simulate dangerous or expensive condition for training purposes. In fact some applications can simulate something completely different from anything we have ever directly experienced such as the visualisation of the ebb and flow of the world's financial markets or the information of a large corporate database. Other applications provide ways of viewing from an advantageous perspective not possible or too expensive in the real world, like scientific simulators, tele-presense systems and air traffic control systems.

The speed at which technology is evolving is making the application of VR within the design professions a feasible approach. AEC companies have already started to evaluate how time consuming the traditional presentation path can be where animations or walkthroughs are used to show designs solutions to their clients. In fact traditional CAD/CAAD systems are used as rendering tools more than design tools. Any change on design solutions is subject to the inevitable delay of having to step back to the CAD/CAAD systems and then the result must be rendered again to be eventually visualized. This approach is obviously not only unconvenient but time consuming and therefore costly.

Consequence of these issues is that some design and manufacturing companies have already started to investigate how VR can be used within the design process.

2. Background

Since the presentation of The CAVE at the SIGGRAPH conference in 1992 all the media have been presenting VR technologies as the new tools that designers were waiting for and in particular they have been erroneously assumed that it would have triggered the architects' interest for its power to communicate desing ideas.
Unfortunately VR is far from being used on a large scale base during the architectural design activities although it would provide a natural and easy-to-understand interface between practitioners and clients. Moreover VR would enable architects to test the design functionality and to see whether the design solutions reach the clients expectations; it would increase the possibilities to desing a better-built environment by:

- addressing sustainability through environmental simulations and appraisal
- engaging design creativity through immersive design

We have identified two important reasons why VR has not been widely used in the architectural context: the lack of interfaces designed for architects and the wrong positioning of the VR phase inside the design process.

If we observe the software available on the market from a user-centric point of view it is clear that none of the current packages used by architects provides an easy interface to generate virtual environments during the creation phase of the design process. CAAD packages are often complex rendering tools more than design tools and the funtionalities related to VR they provide are subordinated to the creation of 3D models and exportation. Consequently any change on design solutions is subject to the inevitable delay of going back to the CAAD systems and refine the 3D model therefore the creation of 3D models results so unpractical and time-consuming that becomes worth doing only when every design solution has been already taken. In these circumstances the use of VR would just increase costs.

Furthermore the lack of interfaces suitable for architects has as consequence the wrong positioning of VR technologies in the design process. In fact if the virtual environments created using CAAD packages are generated as refinements, adjustments and exportation from traditional 3D scenes, it is clear that practitioners will consider their use only as presentation tools. Keeping up to date 3D models is an expensive task and obviously even more expensive is to upgrade VEs generated from them.
Therefore we could say that VR is relegated to the end of the design process rather then being used to engage design creativity through immersive design.

Figure 1. Traditional Schema.

Having identified these problems behind the difficulty of using VR from the very beginning of the design phase, the research group has thought to engage itself in the development of a VR system that provides a flexible user-friendly immersive environment to support collaborative design on a synchronous co-editing base, being called JCAD-VR.
This paper will report the present state of the JCAD-VR system and will highlight its future development.

3. The JCAD-VR schema

The idea upon which the JCAD-VR framework is founded is to anticipate the use of VR within the creation phase thus taking full advantage of VR technology. The system in fact allows to create simple virtual environments through a user-friendly interface without forcing the user to model it with traditional CAAD packages. The use of CAAD packages is therefore left to the final stage of the project, where further refinements are needed.
It creates simple parametric 3D-shapes directly in a co-edit VR environment, thus allowing the design to be shared as it evolves.

Figure 2. JCAD-VR Schema.

To allow constant collaboration between several users the entire project is based on client-server architecture where every user accesses the virtual world, interacts with the VE and shares design tasks. The whole framework is organised in an object-oriented fashion, where each module fulfils a certain task and it is independently coded. This approach has allowed the delivery of an initial functioning core of the system, whose capabilities will be expanded in the near future.

Figure 3. The JCAD-VR framework schema

From the implementation point of view JCAD-VR handles the VE through two closely connected sections: a *3D engine* and a *services unit* each made of several *modules.*

3.1 3D engine unit

The 3D engine handles all the information regarding the visual aspects of the VE. It includes the code necessary to create and modify geometric entities (**geometry core**), to run the 3D-interface (**interface core**) and to deal with several different output devices (**visual core**).

The first module of the **geometry core** handles the creation of 3D objects: both geometric primitives (cones, boxes, spheres etc.) and architectural entities (walls, slabs etc.). To the architectural entities some extra properties were provided such as: information on internal and external faces or windows and doors attached to them.
The *geometry module* will also provide the means for attaching materials to objects and add lights and objects from a library to the virtual world through the *database module*.

The **interface core** does not implements a traditional graphic user interface (GUI): JCAD-VR has been provided with a 3D interface that is an integral part of the virtual world itself. The idea behind it is that instead of the traditional menus and toolbars the UI is immersed in VE providing the means for the interaction: 3D menus pop up showing 3D icons and the 3D menus themselves can be moved for the convenience of the user. Visual feedback is provided in the form of rulers showing the size of objects or 3D icons helping the user in the operations to be done on the objects.

The **visual core** is the part of the framework that allows the interfacing with the visualization devices. The client application has been implemented in order to be used on PCs as well as on Sgi supercomputers. The former are normal PCs whose video-card is displaying the virtual world only on a traditional window at full screen, the latter is a 12-processors 6Gb Ram Sgi Onyx2 system running a Reality Centre. When JCAD-VR is launched on the system running the Reality Centre it can take advantage of the increased computational power stretching its visual output on a 5 metre wide 2 metre high tassellated screen where 3 projectors create a 160 degree panoramic image.
For the sake of flexibility the entire system is coded in Java□. The choice, even if less efficient in terms of performances if compared with some other languages, offered indeed great flexibility, true scalability and last but not least fully multi-platform support. Moreover the use of Java□ programming language became a natural choice when its 3D suite was released (Java3D□).
This choice has provided the flexibility neceessary to deliver images for a range of viewing devices and the internal architecture of the visual core is such that modules might be easily adapted to allow use of different VR devices such as CAVEs or HMDs.

3.2 Services unit

The *services unit* handles all the circulation of data within the system.
It is the backbone of the interconnection between users: it manages network connections, it exchanges data between users (**network core**) and it keeps track of the state of the virtual world through a database from which it also retrieves objects information (**database core**).
The *services unit* is based on a client/server architecture therefore it is implemented across two independent packages of the framework the *client* and the *server* and the **network core** is allowing the transmission of data between them.

The **network core** is thus based on a multi-client server, several clients and the network allowing the communication. The server is the data-delivering unit that looks after the information to be broadcast. The clients are the users themselves who perform actions and queries, when active, and when passive,

rely on the server for receiving data update. The intrinsic multiplatform nature of JCAD-VR, inherited from the language used, allows the server to transmit data to a broad range of machines across several operating systems. The communication channel ensures the link between server and clients through a TCP/IP network.
As an independent part of the framework the server has a simple and autonomous interface that provides primarily information about the network system.
At the present stage the *network module* supports:

- Broadcasting of new geometries in the VE
- Notification of creation of new geometries in every user internal database and broadcasting of their numerical information
- Broadcasting of modifications applied on geometries in the VR scene
- Notification of changes on geometries in every user internal database
- Checking for user priority on the objects through a distributed locking mechanism
- *Avatars* representing multiple clients in the VE
- Interaction between users through a chat system and a whiteboard for freehand sketching in 2D.

It will be soon expanded to include new functionalities such as the transfer of voice and video accross users.

Figure 4. The client/server architecture of JCAD-VR where the server broadcasts to several clients including the Reality Centre

The **database core** includes an internal database, that keeps track of the numerical parameters of the geometries created or modified within the virtual scene, and an external database through which users will be able to retrieve more complex 3D shapes, AEC objects, materials, lights etc.
The internal database is closely coupled with the **network core**. Not only it keeps track of what it is happening in the user's virtual world but also, most importantly, it receives, through the network, information sent by other users' internal databases. If a new object is created or its geometric parameters are changed the system will upgrade the internal database of each user no matter who is doing the action.
For the convenience of the user an *I/O module* supports loading of external files thus allowing import from traditional CAAD packages.
Figure 5. Avatars and communication tools in JCAD-VR. Figure 6. JCAD-VR running on a PC, Sgi and in the Reality Centre.

4. Further Developments

Further developments and completion of several modules of the framework are planned to be carried out in the next future:

- A voice driven interface enhancing user friendliness of the UI
- Implementation of the Audio/Video module and introduction of Video Conferencing facilities.
- Support for driving devices such as 6-degree of freedom/virtual glove
- Implementation of a multi-environments server capable of dealing with several VR environments simultaneously.

5. Conclusions

The JCAD-VR framework has confirmed the possibility of anticipating the use of VR within the creation phase thus taking full advantage of VR technology. It allows access to VR in an easy and direct way not obliging users to model in traditional CAAD environments.
Moreover it creates simple parametric 3D-shapes directly in a co-edit VR environment, thus allowing the design to be shared as it evolves.

As already pointed out, JCAD-VR is an ongoing project and several enhancements are planned for the next releases aiming to get that feeling of intuitiveness and that control over the design that should be the goal of every application using Virtual Reality.

References

Barrilleaux, J.,"3D User Interfaces with Java 3D-A guide to computer-human interaction in three dimensions", Manning, Greenwich, 2000.

Bertol, D., "Designing Digital Spaces", John Wiley & Sons, 1997.

Conti, G., Ucelli, G. and Maver, T., "JCAD-VR - A Collaborative Design Tool in Virtual Reality", Proceedings of ECAADE 19. Helsinki, 2001, pp 454 – 459.

Dorta, T. and LaLande, P., "The Impact of VR on the Design Process", Proceedings of ACADIA'98, Quebec, 1998, pp 139 – 161.

Hughes, M., Hughes, C., Shoffner, M. and Winslow, M., " Java Network Programming", Manning, Greenwich, 1997.

Maver, T., Harrison, C. and Grant, M., "Virtual Environments for Special needs", Proceedings of CAAD Futures 2001. Eindhoven, 2001, pp ???

Petric, J., Ucelli, G. and Conti, G., "Educating the Virtual Architect", Proceedings of ECAADE 19. Helsinki, 2001, pp 388 – 393.

Professor T.W. Maver
ABACUS Unit,
Architecture and Building Science,
University of Strathclyde,
131 Rottenrow
G4 0NG Glasgow, UK
Tel. +44 141 5524400
Fax. +44 141 5523996
E-mail. jean.dick@strath.ac.uk