

A Java3Dä Tool for Real-Time Collaboration in a Virtual Reality CAAD Environment

G. Conti and G. Ucelli
University of Strathclyde
Department of Architecture and Building Science
ABACUS Unit
Glasgow
UK

ABSTRACT

Today the development of network-based virtual communities and the use of avatars have brought a new level of complexity to the meaning of virtuality, providing the technology for remote presence and collaborative experiences. In this project the intention was to pursue this articulated vision of VR in order to assist the design profession during the early stages of the design process. The objective was to provide a tool that is capable of creating 3D shapes in a shared VR environment, thus allowing the design and its evolution to be shared. The use of the Java programming language was a natural choice for this project. Because of Java's performance scalability and hardware independence the concept of CAAD has been extended, making it possible to create a VR environment that can co-exist between high-end supercomputers and standard PCs. The project is currently being tested using PCs and an SGI system running a Reality Centre. The research reported in this paper describes the architecture and application of software that aims to increase the opportunity for collaboration within virtual worlds and enable effective and transparent information exchange.

1 BACKGROUND

Design methodologists in the past agreed on the need of iterative cycles between several phases of the design process. From studies concerning designers' behaviour many authors observed an indefinite number of return loops from the moment when gathering of information and structuring of the design problem take place (known as *analysis*) to the one when design solutions are generated (known as *synthesis*) (Lawson 1990).

The use of Virtual Reality (VR) within the design process could give to the designer an appropriate quick and practical response to his/her need of iteration and search for design solutions. Moreover it enables the capture of more information than with the use of the traditional media and the checking of the design solutions more efficiently, thereby enhancing simulation capabilities. Furthermore VR broadens the boundaries of traditional perception providing experiences of worlds not necessarily real or material. In short it is the perfect simulation medium when in search of design solutions.

It is then highly predictable that in the near future VR will become the interface for the next generation of computer aided drawing (CAD) applications and we can anticipate the change of its use from a mere presentation medium to a more

powerful and effective design tool.

CAD/CAAD packages are very powerful but often complex rendering tools. They were not meant to be investigation tools so generating 3D models is often impractical and time-consuming. Therefore they are usually employed when every design decision has been already taken.

Virtual environments (VE) are often created using CAAD packages as refinements, adjustments and exportation from traditional 3D scenes. Keeping up-to-date 3D models is an expensive task and obviously even more expensive is the upgrading of the VEs generated from them.

In these circumstances the use of VR would just increase costs. As result VR is relegated to the end of the design process rather than being used to engage design creativity through immersive design. Figure 1 below shows the current position of VR within the design process.

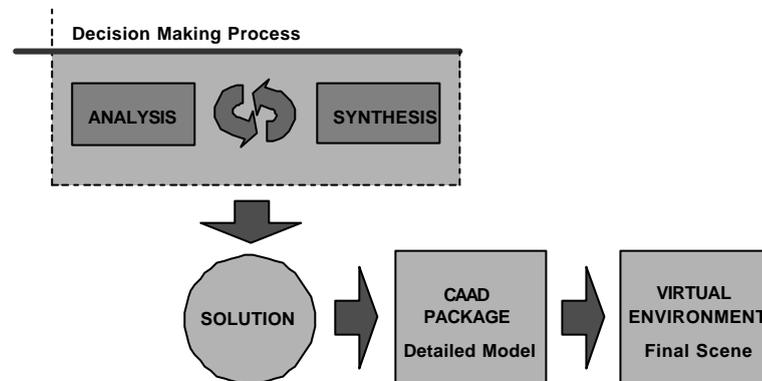


Figure 1: Use of VR in the traditional scenario

In the traditional scenario the decision making process does not take advantage of the technology but relegates the use of VR to the end of the process as a more convincing tool to impress contractors and clients. Only once the final solution has been achieved is it worth investing time in more powerful visualisation media.

Being aware of this background the research group engaged itself in the development of a VR system, named JCAD-VR, to help designers in the first stage of the design process.

In short JCAD-VR provides a flexible user-friendly immersive environment to support collaborative design on a synchronous base.

It can be thought of as an investigation tool that allows the designer to sketch freely 3D shapes within the virtual context. Moreover design solutions are shared in a synchronous fashion with other participants through the system's network-based architecture. Figure 2 shows the proposed scenario using JCAD-VR within the decision making process:

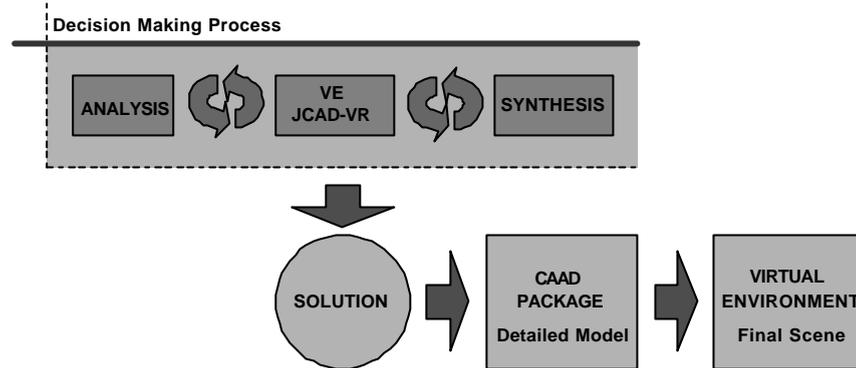


Figure 2: **Proposed scenario**

Here JCAD-VR provides the means for a more effective use of VR bridging between the phases of *analysis* and *synthesis*. VR is now employed at the very beginning of the decision making process when it is most likely to help in finding better design solutions. Once a convincing solution is achieved the task of the creation of a very detailed 3D model and the final VR scene is given to appropriate CAAD packages.

This paper reports the present state of the JCAD-VR system and will highlight its future development and testing.

2 SYSTEM CAPABILITIES

The system has been entirely developed around a client-server architecture to allow constant synchronous collaboration between several users. Every user accesses the virtual world, interacts with the VE and shares design tasks.

To better communicate the capabilities of JCAD-VR, the explanation of the client side is presented separately from the explanation of the server. As if in a software demo, all the possible actions/options allowed by the system will be identified.

2.1 Client package

1. On initiating JCAD-VR, the user is asked for a login name for the session and through an options panel he/she can:
 - Decide which server to connect to and through which server port (in case the software is running on a local server).
 - Decide if running in single or multiple screen mode. Single mode is set in case the display device is a normal screen; the multiple screen option has been included to allow devices such as the multi-projector display system processing the visual output of a Reality Centre.

- Load a VRML 97 file as a VR scene, if any.
- Activate or de-activate video conferencing facilities for the session (in case video conferencing facilities have been activated, support for video capturing device recognition and checking are provided).

Providing the information required for this initial window is quite straight forward. Ease of use has been taken seriously in consideration of those not accustomed to the system. JCAD-VR provides also a single user option in case collaboration is not required.

2. Every window disappears freeing the space to the 3D graphic user interface (GUI) of the system. 3D icons appear on the screen showing, as intuitively as possible, the functions that they allow; the chosen VRML 97 file is rendered and is shown on the screen as in a traditional VRML browser. Each user can now:

- Navigate through the VE: walking, tilting and panning movements are allowed.
- Observe other participants' movements through their 3D avatars in the VE. Avatars are made distinguishable from each other by having, nearby, a 3D text of the users' login name.
- Listen to the other participants' voices through the loudspeakers.
- Watch, in the virtual scene, the 3D panels showing the video captured and sent by other participants. Every 3D panel shows the participants' login name to make them clearly distinguishable. These 3D panels are intrinsic parts of the GUI of JCAD-VR. Instead of being conventional windows they are 3D entities within the VE. These panels, as well as all the elements of the 3D GUI, are moveable for the convenience of the user.
- Check in a monitor the local user's captured video that is streamed out to the server.
- Check the list of users.
- Chat with other participants. This option is provided to assure a certain degree of communication between the participants in case the video conferencing facility is de-activated; this is available even when the video conferencing has been set as active.
- Freehand sketch in 2D on a shared electronic whiteboard, the possibility to set colours is provided in order to ease distinction between participants' contributions.
- Create 3D shapes and 3D AEC objects: both geometric primitives (cones, boxes, spheres etc.) and architectural entities (walls, doors, windows etc.) are available. A "3D ruler" is provided to help the user in constructing objects.

The choice of a 3D GUI was made in response to the possibility that modules supporting several display devices such as CAVEs and head-mounted displays would be included. Not having a traditional windows/menus user interface JCAD-VR can be

used freely in every display situation minimising the effort to customise it for each device.

3. The 3D engine of the system renders all the possible changes of the VE: movements of avatars, video conferencing streams on the 3D panels and, most importantly, creation and modification of objects created within JCAD-VR.

Each user, concerning 3D objects, can:

- Observe every object created in the VE both those created locally and by other participants. The system upgrades the VE with the objects created by all the participants in a synchronous fashion. An identification routine is provided in order to give each object a unique ID number to avoid interference in each users' local database of objects.
- Pick every *active* object in the virtual scene. *Active* objects here are all the objects created within JCAD-VR and not geometry imported with the VRML 97 file. Objects originally part of the VRML 97 file are considered to be *passive* and are not pickable. Once an object is selected:
 - User priority on selected objects is set by a distributed locking mechanism. Locked objects will be no longer pickable for other participants until unlocked and their locked status becomes apparent through change into a red colour.
 - 3D icons, 3D panels with general dimensions and x, y ,z arrows are set visible to help the user operate on the object.
 - Translation, rotation and scaling in every direction is allowed. These modifications are operated on the object by simple dragging the arrow representing the x, y or z axis. A 3D ruler and a 3D panel provides simple visual feedback to check the modified dimensions. Some routines to constrain modifications allowed on AEC objects are included, for example a door cannot be moved onto or too close to a second door and vice versa.
 - Change of material is supported. In the first instance objects are created with a default grey colour but a library of textures is provided.
 - Deletion of the object is allowed.
- Observe the visual feedback of the locking engine mechanism in case one of the participants has selected an object.
- Observe upgrading of the VE in case any modification on one or more objects has been carried out by any participant.
- Check for information on all *active* objects in the VE through a local database of objects. Objects are divided by type and general geometrical information is provided such as length, width, height, volume, radius, material etc... The system notifies to every users' internal database any creation or modification of geometric objects within the virtual scene and broadcasts their numerical

information.

Figure 3 shows, for every action performed by the user, the consequential visual feedback and data sent/received to/from the server through the network:

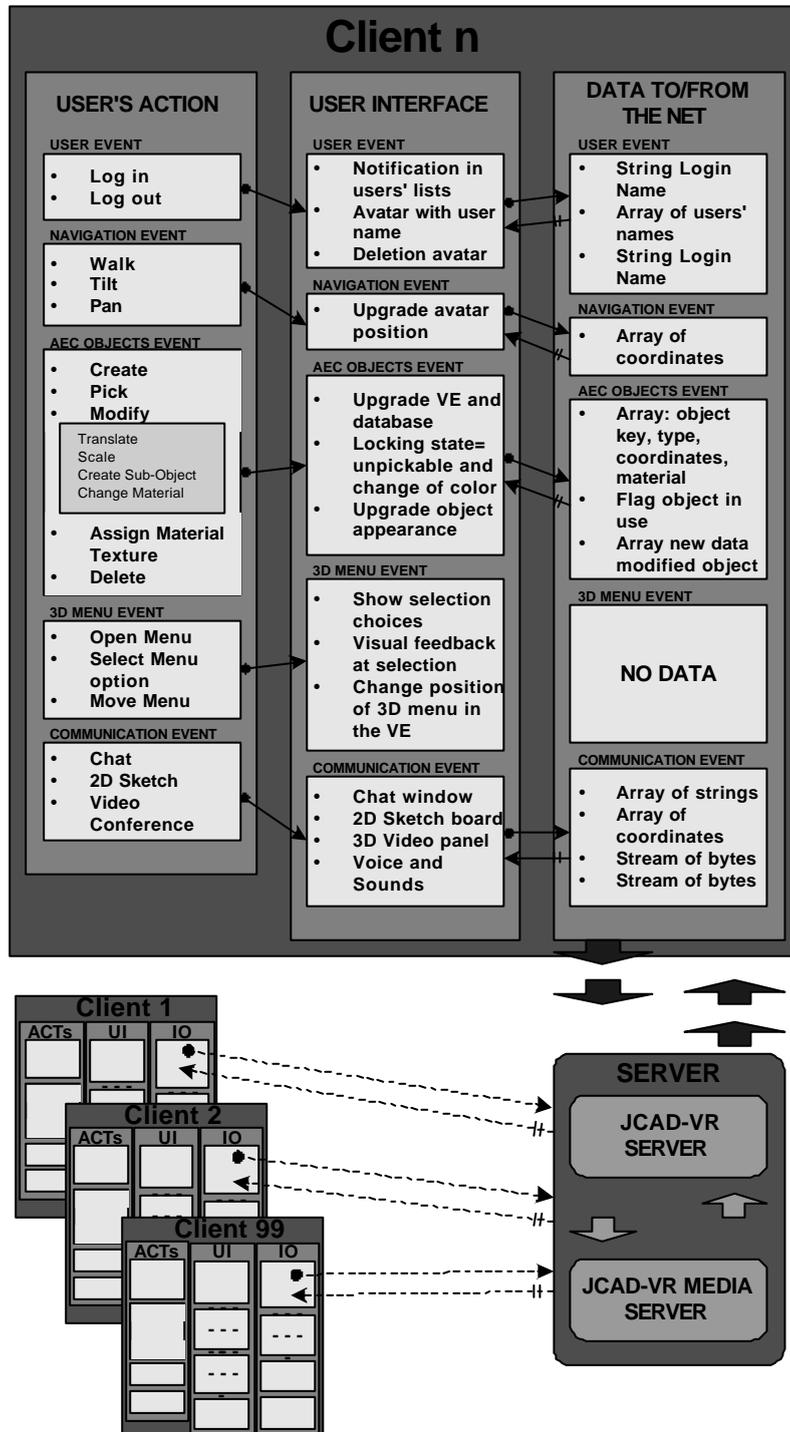


Figure 3: Client/Server Architecture of JCAD-VR



Figure 4: An image of JCAD-VR

2.2 Server package

The server is made of two parts: *JCAD-VR Server* which looks after the VE information to be broadcast, and *JCAD-VR Media server* which takes care of media streams and video conferencing tools. Both parts constitute the server system and they are closely linked to each other.

As an independent part of the framework the server has an autonomous and simpler interface that provides primarily information about the network status and transfer rate. A number of components are envisaged such as the communication status, the users on line, VR shared environments and settings and size of video conferencing windows. Since the clients are communicating through independent processes, a future enhancement will allow the server to deal with several VR environments simultaneously.

The intrinsic multiplatform nature of JCAD-VR, inherited from the language used to code it, allows the server to transmit data to a broad range of platforms, from normal PCs to the supercomputer running a Reality Centre, and leaves the research team the freedom to test the software with several operating systems. The communication channel ensures the link between server and clients through a TCP/IP network.

3 SYSTEM IMPLEMENTATION AND HARDWARE USED

The whole framework of JCAD-VR was organised to allow concurrent software development, in a modular fashion, by individual members of the R+D team (Conti et

al. 2001). To facilitate this, an object-oriented approach was identified as the most suitable one and the entire system was coded in Java™.

The choice, even if less efficient in term of performances if compared with some other languages, indeed offered great flexibility, true scalability and, last but not least, fully multi-platform support. Java3D™ was used to code the GUI and everything concerning the VE. Its network-centric nature, its multimedia integration together with the use of native hardware acceleration (OpenGL) and multi-processor support (in the case of Sgi workstation) make it the obvious choice for the development of a real-time multimedia collaborative system.

The client application, in response to the obvious hardware limits imposed by the use of different hardware, has been written to be easily customised to run on PCs as well as on a Sgi supercomputer. The former are normal PCs whose video-card displays the virtual world only on a traditional window or at full screen. The latter is a 12-processors Sgi Onyx2 system running the Reality Centre at ABACUS, University of Strathclyde, Glasgow. When JCAD-VR is launched on the Sgi it can take advantage of its computational power to stretch itself on a 5 metre wide 2 metre high tassellated screen where 3 Barco projectors create a 160 degree panoramic image. The internal architecture of JCAD-VR is such that modules might be easily adapted to allow use of different VR devices such as CAVEs or Headmounted Displays as well as several pointing devices such as a joystick, 3D mouse and VR Gloves.

From the collaborative point of view JCAD-VR is highly scalable and several communication media options are provided depending on the hardware limitations of the computer on which it is running.

The video conferencing facility has been coded using the Java™ Media Framework (JMF) which enables cross-platform capture, playback and streaming of audio and video at different transfer rate and resolutions. A great deal of effort has been expended by the research group to integrate closely the two sections of JCAD-VR: the 3D module with the multimedia module.

4 CONCLUSIONS AND FURTHER DEVELOPMENTS

The prototype JCAD-VR system makes some steps toward the change of VR usage from mere presentation medium to a more powerful and effective design tool, and establishes the feasibility of VR becoming the interface for the next generation of computer aided drawing (CAD) applications.

JCAD-VR is a wide and multidisciplinary project where several aspects concerning design have been investigated. The role of sketching within the design process, problems and interference in concurrent design, the importance of interfaces inside design packages and the use of VR in design are some of the theoretical issues raised and resolved.

During the implementation phase several technical issues became critical, making the project an occasion to test new technical solutions and technologies.

A series of experiments, involving other universities, are being undertaken. JCAD-VR will be tested on the 3D GUI and when it runs in collaborative mode with several users over the network. Outcomes of these experiments will be reported in future publications.

New modules and enhancements are planned for the next release, including:

- A voice driven interface enhancing user friendliness of the user interface
- 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 REFERENCES

- Barrilleaux, J. (2000) *3D User Interfaces with Java 3D, a guide to computer-human interaction in three dimensions*. Manning, Greenwich.
- Bertol, D. (1997) *Designing Digital Spaces, An architect's guide to Virtual Reality*. John Wiley & Sons, Inc.
- Conti, G., Ucelli, G. and Maver, T. (2001) JCAD-VR - A Collaborative Design Tool in Virtual Reality, *Proceedings of ECAADE 19*. Helsinki, 2001, pp 454 – 459.
- Dorta, T. and LaLande, P. (1998) The impact of Virtual Reality on the design process, *Proceedings from ACADIA Conference*, Quebec City, 1998, pp 139 – 161.
- Hughes, M., Hughes, C., Shoffner, M. and Winslow, M. (1997) *Java Network Programming*. Manning, Greenwich.
- Lawson, B. (1990) *How designers think: the design process demystified*. University Press, Cambridge.
- Sowizral, H. A. and Deering, M. F. (1999) The Java 3D API and Virtual Reality, *IEEE Computer Graphics and Applications*, May/June.
- Petric, J., Ucelli, G. and Conti, G. (2001) Educating the Virtual Architect, *Proceedings of ECAADE 19*. Helsinki, 2001, pp 388 – 393.