Multi-perspective Collaborative Design in Persistent Networked Virtual Environments

Jason Leigh, Andrew E. Johnson, Christina A. Vasilakis, Thomas A. DeFanti
Electronic Visualization Laboratory (EVL), University of Illinois at Chicago
(jleigh@eecs.uic.edu, http://jaka.eecs.uic.edu/~spiff/calvin)

"Each vantage point, each mode of organization will create a new structure. And each new structure will enable you to see a different manner of meaning, acting as a new method of classification from which the whole can be grasped and understood."
– *Richard Saul Wurman,*
Information Anxiety, 1989.

Abstract

In this paper we present an approach to applying virtual reality in architectural design and collaborative visualization which emphasizes the use of multiple perspectives. These perspectives, including multiple mental models as well as multiple visual viewpoints, allow virtual reality to be applied in the earlier, more creative, phases of the design process, rather than just as a walkthrough of the final design. CALVIN, a prototype interface which implements these ideas, has been created using the CAVE^{tm} virtual reality theatre.

1. Introduction

Three dimensional walk-throughs employing an "inside-out" perspective have traditionally been the raison d'etre for virtual reality (VR) in architectural design[2]. This "inside-out" perspective allows a person to see the environment as if standing *inside* the environment and looking *out* at the surroundings. We believe this limits the usefulness of VR to only the final stages of the architectural design process, where a CAD model can be given to the VR environment to display. Our surveys of veteran architects and architectural students suggest that this final stage of design, involving the building of the CAD model, only occupies about 20% of the entire design time. The remaining time is spent iterating over many experimental designs on paper or through miniature models- a process which is largely, if not completely, unsupported by computers.

We present an elaboration of the traditional application of VR to architectural design which involves the use of multiple perspectives to support this process. These perspectives include those that:

1. are produced as a result of applying differing camera parameters to view a design.
2. are produced as a result of applying information filters that are designed specifically for the tasks performed by individual users.
3. occur when multiple collaborators offer their opinions on the architectural design.
4. occur when collaborators experiment with multiple designs.
5. occur from design ideas maturing over time.

In the following sections we will describe these ideas in greater depth, explaining the motivation for our approach, as well as describing a prototype implementation of a networked virtual design space that embodies some of these concepts. Finally we will briefly discuss how this multi-perspective approach may be generalizable to other visualization domains.

2. Surveys

The concept of applying multiple perspectives to a design environment was motivated by two informal surveys conducted on groups of veteran architects and architecture students. The first survey consisted of questions regarding the general architectural design process and the role of computers in this process. The second survey consisted of questions regarding the nature of collaboration in architectural design. For brevity, we will summarize our findings below (full details are available upon request).

The first survey showed that the architectural design process consisted mainly of:

1. Initial research.
2. Sketching ideas.
3. Building physical, and to a lesser extent, computer models or studies for quick evaluation.
4. Committing a final design to a CAD package.

In general, design problems included external architectural design, internal architectural design, and renovation/re-organization of an existing space. Steps 2 and 3 are typically repeated a number of times before step 4 is attempted. In some instances architects will return from step 4 to step 2, however the greatest amount of time was spent iterating over steps 2 and 3. Step 4 was considered the least creative and most tedious phase of design. It was also described as the most obvious phase to apply VR, and most of those surveyed indicated that it would be a good way to *impress* clients with VR walk-throughs of the architecture. It was noted however that visualization in general was considered valuable in explaining the design of the building to clients who could not interpret floor plans.

On the issue of collaboration in architecture, the second survey showed us that collaboration was a crucial part of the design process. The collaborations could consist of members from within an architectural firm or members from geographically distant locations (perhaps several thousand miles away). Collaborations were equally formal and informal. That is, they spent approximately equal amounts of time in informal meetings with colleagues to chat about problems, as they did in formal meetings that were scheduled with colleagues, clients, and engineers. However, it was also mentioned that more work was done in informal collaboration, where the emphasis was on the exploration of ideas, compared to formal collaboration, which mostly consisted of confirming designs that were brought to the meetings.

In summary, the surveys showed that architectural design is an iterative process that relies heavily on forming collaborations with colleagues, clients and engineers. In the next section we will discuss how our approach of providing multiple perspectives can be instrumental in supporting this collaborative design process.

3. Multiple Perspectives in Design

One of the obvious affordances of VR is its ability to depict environments from an "inside-out" perspective. That is, the viewers are placed in a position where they are physically immersed in the environment. This has been leveraged by many researchers[2, 5] to produce three-dimensional walk-throughs of architectural spaces. These implementations have been very successful because they offer clients the ability to tour a building design before it is actually built.

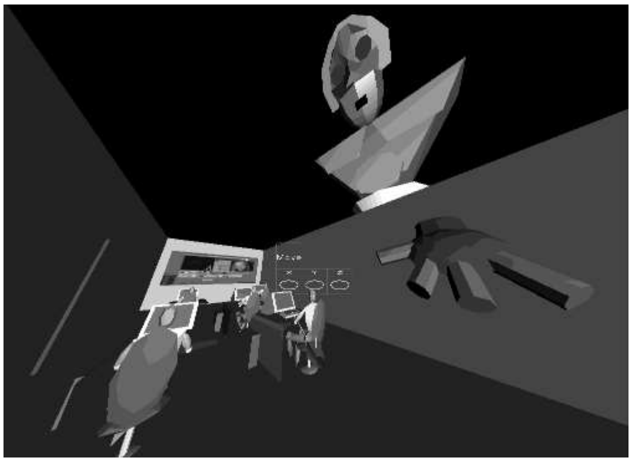

Figure 1. A mortal's view of the world. High above one can see the avatar of a deity.

3.1. Multiple Camera Parameters

Although the single "inside-out" perspective is useful in the evaluation of a pre-designed space, it is not necessarily the most appropriate perspective for the actual design process. For example, in the organization of office cubicles, it would be difficult, from an "inside-out" perspective to be able to position cubicles relative to one another and still maintain a global sense of their relative position in a room. Such a task is inherently easier if it were performed on a miniature model of the room with pluggable cubicle components. This miniaturized view can be referred to as an "outside-in" view. This alternative perspective has already been applied by a number of researchers[4, 13] with considerable success. We call this notion of providing two viewpoints for perspective viewing, "mortals and deities." In the most trivial scenario mortals view the world from an "inside-out" perspective (figure 1) while deities view the world from an "outside-in" perspective (figure 2). In a collaborative environment however, deities may assume more influential roles over mortals.

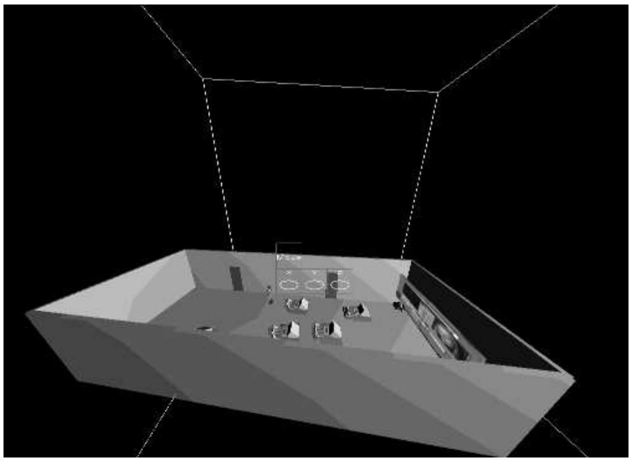

Figure 2. A deity's miniaturized view of the world. The mortal is miniaturized in the world.

3.2. Multiple Information Filters

Although multiple camera perspectives have already been applied to VR in architecture, little has been done to generalize this notion in collaborative environments. Bier's[1] Magic-Lens concept provides a two-handed interface to allow a user to manipulate and activate a set of filters to gain multiple perspectives over what is being visualized but does not extend this to consider the dynamics of working amongst remote collaborators. Olson[11] asserts (and our surveys also suggest) that multiple representations are important in a collaborative work environment. In this case, the collaborators may be working on the same aggregate information but they are exploring decidedly different views. For example an architect would want to see the information as three-dimensional structures whereas a mechanical engineer would want to see the same structure in terms of stress and strain. Providing multiple perspectives over the same general architectural model allows each participant to apply his or her expertise to the problem at hand, by supplying them with the visual representations they are accustomed to interpreting.

3.3. Multiple Opinions via Collaboration

An important part of collaboration for architects is in eliciting feedback from their colleagues, clients and engineers. In the context of mortals and deities, the roles collaborators assume can dictate the actions they are capable of exercising in a collaborative virtual environment. For example, mortals view the world from the "inside-out" and therefore can more easily perform fine manipulation of the environment such as finely moving, scaling, or rotating a sub-part of an object in a scene. On the other hand, deities view the world from the "outside-in" and therefore can more easily perform gross manipulation on objects in the environment.

Mortals and deities can assume the roles of apprentices and teachers respectively[8], or even clients and demonstrators. In such roles, the apprentice/client is in a position of learning/receiving a tour from the deity. Hence the deity can selectively turn certain mortal "powers" on or off as necessary. A deity may also have the ability to literally pick up mortals and bring them to the deity's perspective or even reduce their own perspective to that of the mortal's so that they may reside in the same space with the mortal.

3.4. Experimenting with Multiple Designs

The application of VR in architecture has largely been dubbed a "rapid prototyping" tool. Although this is useful in finding design problems before a great deal of money is spent in actually building the structure, the use of virtual reality in architecture still does not support a great deal of the early creative design processes; it only supports the end product. In order to walk through an architectural space, a CAD model must first be constructed. Our study of architects have shown that these CAD models only emerge after the architect has, to a great extent, committed to the design. Changes made at this late a stage will be costly in redesign time. Hence VR is only being applied after the tedium of drawing a CAD model is complete; it is not being used to support the creative task of design and problem solving. As VR seems so well suited to solving problems in architecture, it seems ironic that it is being used to support the least creative phase of the process.

Our intent is to provide an environment that introduces VR early in the design process. This can be achieved through two means. Firstly, a large database of interior objects (desks, counters, chairs, toilets, windows, etc) can be provided that will allow users to plug objects into the environment. This is useful to architects who are re-organizing or renovating an existing floor plan. The database may also contain the cost of using particular objects or materials and can compute the cost of taking out a wall, and so on.

Secondly, a three dimensional sketching interface can allow designers to quickly turn their hand-drawn sketches into rough three dimensional studies. The user builds models by sketching the rough shape of the object in space. The computer then fills in the space with polygonal surfaces to create a rough solid representation. These rough studies can then become additional manipulable objects in the database of pre-defined objects.

3.5. Maturing Design Ideas over Time

Finally we wish to incorporate the notion of time in the design environment. That is, the virtual environment still persists after the participants leave. At a later time, any participant may re-enter the space to perform more design work. This encourages informal collaborations to take place. It also affords the use of autonomous agents that can continue to perform tasks even when users have left the virtual environment. Since creativity does not follow a schedule we believe that a collaborative environment that requires a priori scheduling of its participants would be too limiting. By providing the designers with such a persistent virtual world, they may enter the world any time they have new inspirations for possible design solutions. The notion of persistent virtual spaces is greatly influenced by MUDs (multi-user dungeons), text based multi-participant virtual environments that allows participants to enter, interact and leave at any time.

4. CALVIN

CALVIN (Collaborative Architectural Layout Via Immersive Navigation) is a prototype system that applies our ideas of providing multiple perspectives for collaborative design. At the present, CALVIN implements only a subset of these concepts. Specifically, CALVIN implements multiple camera perspectives and allows multiple participants to collaboratively design in a shared architectural space.

We will begin first by describing the individual components of CALVIN. This will be followed by a description of the application of CALVIN to the design layout of a computer room at the National Center for Supercomputing Applications (NCSA).

CALVIN is an outgrowth of CASA (Computer Augmentation for Smart Architectonics)[14], a networked collaborative environment designed to allow the prototyping of "smart" homes and environments in VR. CASA, and consequently CALVIN, was designed to run in the CAVE virtual environment. The CAVE is a 10 foot by 10 foot by 10 foot room constructed of translucent walls that are rear-projected with stereoscopic images. A participant using the CAVE dons a pair of LCD shutter glasses to mediate the stereographics imagery. A magnetic tracker, attached to the glasses, relays the position and orientation of the user's head to the computer. A 3-button wand, also equipped with a magnetic tracker, is provided to allow interaction with the virtual environment. The graphics for the CAVE are driven by an SGI Onyx.

The core of CALVIN is the CAVE library. On top of this, CALVIN uses OpenInventor^{tm} to render the virtual environment. CALVIN provides two main interfaces for interaction: the virtual visor and a speech recognition module. Multiple distributed CALVINs running on separate VR systems are connected via a centralized database server that guarantees consistency across all the separate environments.

Although CALVIN was originally conceived for the CAVE, the CAVE library itself is capable of supporting a number of VR platforms, including the ImmersaDesk^{tm}, BOOM, fish-tank VR systems, and simple graphics workstations. The Immersadesk is a scaled-down version of the CAVE with only a single projection screen angled to resemble a large drafting table.

4.1. Graphics Support for CALVIN

CALVIN is written in C++ using OpenInventor as the underlying graphics library. This provides a number of benefits: 1. OpenInventor provides a convenient means to organize three-dimensional scene hierarchies while at the same time offering scene culling and level-of-detail management. 2. The entire scene hierarchy can be saved at any time and trivially converted to VRML (Virtual Reality Modeling Language) for general distribution over the World Wide Web or to collaborators. 3. OpenInventor and the CAVE library are portable to other platforms. 4. There is a large collection of VRML architectural models gradually appearing on the Web. These can directly be used as models to be explored by CALVIN as test cases.

Mortals and deities are realized in CALVIN as user-designable avatars. An avatar is a persona that each participant occupies to establish a representation in the virtual environment. Users do not see themselves in these avatar forms- only the other participants of the environment see the user in this way.

In CALVIN, avatars consist of a separate head, body and hand. This is motivated by the fact that we currently have two trackers attached to a participant: one tracker for the head and one for the hand. Tracking of the head and hand allows the environment to transmit gestures such as the nodding of a user's head or waving of their hand to the other networked participants.

Avatars can be designed using any commercially available modeling package such as Alias or SoftImage and then converted to the OpenInventor format to be read by CALVIN. Models may be as simple as a

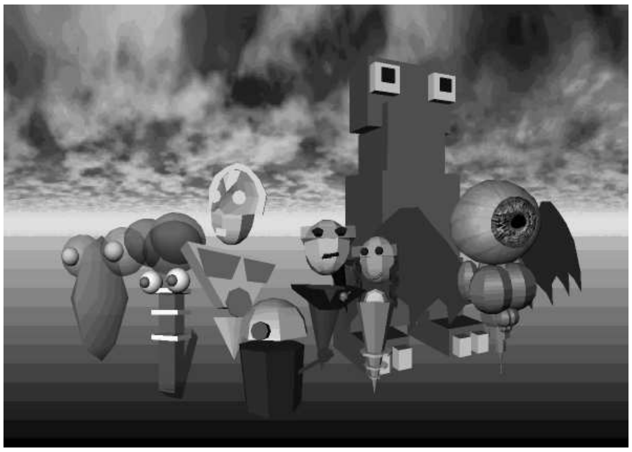

Figure 3. A selection of CALVIN's avatars.

cube, as is often used to arbitrarily represent avatars, or as elaborate as anything the user chooses to design. Since our laboratory has had a long history of artists working with computer scientists, we leverage the talent of the artists to design more creative avatars (figure 3). This may heighten the sense of *drama* in a virtual experience[9] but also it has been argued to improve the sense of *presence* in a virtual environment[6].

Although the avatars shown in figure 3 may seem more appropriate in a child's learning environment rather than a professional design tool, we believe nevertheless that it is important to provide significantly contrasting avatar representations, and to provide avatars that give sufficient cues for viewers to discern the direction the avatar is facing. In real life we identify people at a distance by their gait, their height, their complexion, or the color of their hair. In a virtual world gait is difficult to distinguish since avatars tend to glide across the landscape. If every avatar were uniformly designed (e.g. a cube with a texture-mapped face on all sides) they would be difficult to distinguish from one another at greater distances. Instead, by providing avatars with obvious fronts and backs, participants may communicate notions of relative position to one another with phrases such as "it is behind you," or "turn to your left."

4.2. Persistence and Network Support for CALVIN

To generate the degree of persistence that will allow collaborators to work both synchronously and asynchronously, CALVIN manages a central repository of information that maintains the states of the various on-going design environments. This repository contains a collection of objects (including light sources) to be placed in the scenes, a collection of avatars, and a collection of scene description files.

Each object and avatar part is stored in its own inventor format file, allowing each object to be modified independently, and shared independently. Each scene description file stores information on the position, orientation, and scale of every object within the scene. Each user maintains a user definition file for each scene they are involved with. This file maintains information on which avatar is portraying the user, where the user is located in the scene, and whether the user is a mortal or a deity. This information is loaded into CALVIN's central database server prior to a design session.

CALVIN's network component allows multiple networked participants to work in the same virtual space (figure 4). Multiple distributed CALVINs running on separate VR systems are connected via the centralized database that guarantees consistency across the various environments. The communications library supporting CALVIN's networking is built on a client-server model with the number of remote clients limited only by bandwidth and latency. Many similar approaches have been implemented in the past[3, 12, 17]. Although a centralized database approach can be a bottleneck for networks with low bandwidth and high latency, CALVIN uses this simple approach because it allows us to build distributed virtual environments to concentrate on the human-factors issues of collaborative interaction in persistent virtual environments. We believe that network bandwidth and latency will always pose a bottleneck in any complex distributed environment that involves a large number of participants, hence it is important to devise interaction techniques to minimize this problem at a perceptual level.

Currently CALVIN is started at all participating sites simultaneously, so a new user can not join an existing design session, however a user can leave CALVIN without disturbing the others. The central database stores the current environment as well as all previously saved versions of the environment in the central repository. A user starting up CALVIN is automatically taken to the current version of the environment, but can revert back to a previous version if necessary.

4.3. Virtual Visor and InYerFace

The Virtual Visor is a virtual display device for the CAVE. It simulates a head-up display (HUD) on which status information can be displayed. This is motivated by HUDs that have been researched for some time by the military[15, 16]. In general they have been shown to allow faster transition from instrumentation to vehicle guidance when conformal symbology (symbology that has some visual correlation with the entities in the

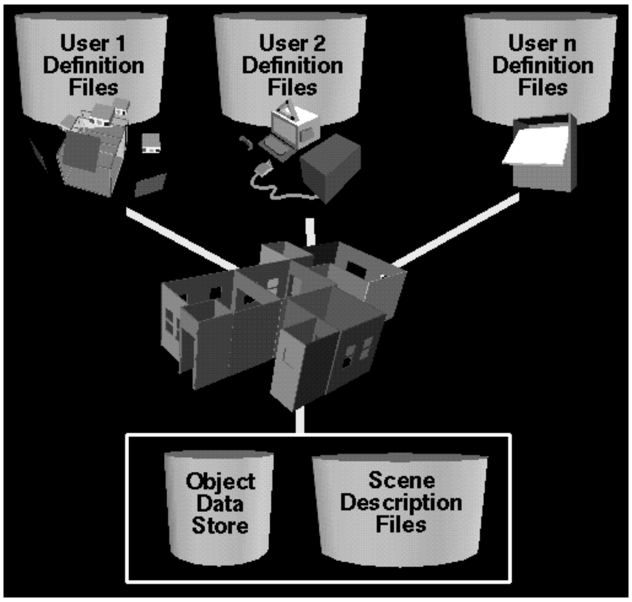

Figure 4. CALVIN connects its remote participants via a centralized database.

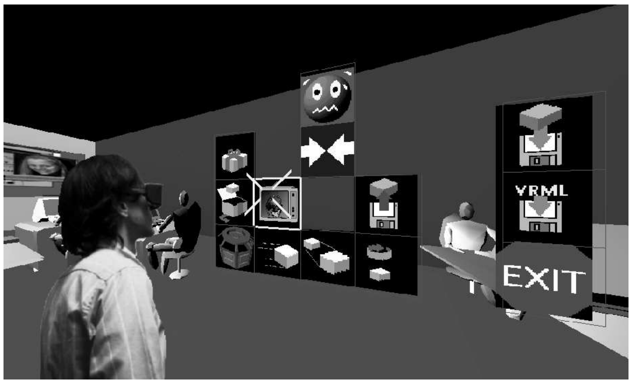

Figure 5. The Virtual Visor and its InYerFace.

scene) is used.

In our design applications where the representations do not have obvious conformal symbology, we have chosen to move the non-conformal displays to the perimeter of the HUD so that they provide the convenience of having the instrumentation nearby, while reducing the interference with the main elements in the scene.

One extension to the application of a HUD in VR, is the InYerFace. The InYerFace augments the Virtual Visor by adding the functionality of being an input device controlled using the user's head orientation. When the InYerFace is activated, the visor freezes in space to display selectable instrumentation (figure 5). A targeting cursor appears in front of the user. When the user's head is moved, the targeting cursor instantaneously follows. To make a selection the user aims the cursor at a selectable visor item (simply by looking at it) and presses a button on the wand.

The motivation for this paradigm is that in contrast to everyday life where tactile feedback allows the operation of devices without necessarily looking at them, images in virtual environments have no tactile properties. Hence when one makes selections from a typical virtual menu[7] one has to look at the menu item first and then use one's wand or glove to make the selection. The task involves three steps: 1. seeing the menu item, 2. aiming the wand or glove at the menu item and 3. making a selection. With the InYerFace, step 2 is eliminated because once the user acquires the target visually, the user is already aligned to make the selection.

The InYerFace may provide an additional benefit when it is used in the Head-Mounted Display, Fish-Tank VR systems, and the ImmersaDesk. In such systems there is a common problem of fatigue that occurs due to prolonged raising of the user's arm to make menu selections. The InYerFace can be used to greatly reduce this problem by reducing the number of operations requiring arm movements.

CALVIN also provides a less visually intrusive alternative to the InYerFace using speech recognition to bypass the InYerFace entirely with voice commands. Speech recognition in the CAVE is currently provided by a commercially available, speaker-dependent speech recognition software package running on a PC-compatible computer.

4.4. Application of CALVIN to design

In the first experimental application of CALVIN, a scale model of a computer room at NCSA was reproduced in VR. The goal was to take the basic components of a Powerwall (a two-screen passive stereoscopic projection environment and a number of workstations) and organize them to fit a room under the following conditions:

1. ensure each workstation has an unoccluded view of the screens.

2. ensure there is enough projection distance between the projectors and the screens.

3. ensure there is enough room between the workstations for floor traffic.

CALVIN was used for this task because it allowed us to check those things (unoccluded views, room for floor traffic) that otherwise would have required moving actual equipment into the actual room. It also allowed many prospective users to experience a room 3 hours away by car (between EVL and NCSA), and several months away from construction, and share their comments.

Generic models of workstations from a library of 3D office interiors were imported into CALVIN. For testing purposes two VR systems (the CAVE, and a workstation powered by a deskside Onyx) were loaded with CALVIN and connected via our lab's local network. In situations where the participants may be situated at geographically distant locations (for example between EVL and NCSA), a conference phone system can be used to relay voice between the collaborators.

Designing in the space first involved the deity organizing the workstations and the projection screens in a preliminary configuration (figures 1, 2). The Virtual Visor was used to display the current transformation being applied to a workstation, and the InYerFace and speech recognition system were used to choose between the various transformation modes (translate, scale, rotate, save model, etc). While the deity oriented the workstations, the mortal (in the CAVE) walked from workstation to workstation evaluating the visibility of the projection screen, performing fine adjustments on the position of the workstations, and ensuring there was sufficient room between them for floor traffic. If there were problems, the mortal and the deity would try various possible alterations to the scene. One of the most obvious areas of cooperation was when the deity was placing one of the projection screens against the wall, the mortal would stand by the wall and guide the deity verbally into moving the screen to meet the wall. Typically during the design process several people stood in the CAVE giving advice and suggestions to the mortal.

CALVIN allowed us to try out many different possible configurations, and quickly modify them based on the mortal's feedback, converging to the final design of the room. The people who participated in this design session felt that CALVIN was a valuable tool, not only in its ability to construct the space but also in the way it encouraged several users to actively participate in the designing of the space.

This was only the first application of CALVIN to architectural design. We are currently working with ARPA to apply CALVIN to the design of reusable workstation pods. The results of this work were demonstrated as part of the GII (Global Information Infrastructure) Testbed at the SuperComputing'95 conference in San Diego.

5. Future Work

From our initial experience with CALVIN we were able to compile a list of improvements:
These include the following:

1. We believe the approach of applying multiple perspectives to the problem of architectural design can be generalized to other disciplines such as collaborative engineering and scientific visualization. In scientific visualization, the data being viewed is typically multi-dimensional. By applying the notion of multiple perspectives we may be able to partition the number of dimensions into smaller, more manageable pieces which multiple participants can then explore simultaneously.

   It is important to realize however, that having multiple participants operate on different views may cause more confusion than insight- especially when participants attempt to communicate what they see to each other. Hence an appropriate interface must be provided to allow participants to share views and more importantly, mental models[10], to maintain proper coordination between efforts.

2. Currently user interaction with objects is limited in CALVIN. We plan to add a 'duplicate' option to allow a user to make a copy of an existing object, and allow the user to place a new object into the scene from a palette of available objects. We are beginning to explore virtual sketching tools that will allow easy translation from 3D sketches to 3D polygonal models. We plan to allow groupings of objects, and the ability to modify the objects themselves (e.g. the colour, the texture.)

3. Currently CALVIN has a very limited form of persistence as users can not join a CALVIN session in progress. We plan to move to a more flexible networking architecture in future versions.

4. We are in the process of replacing the current speech recognition system with a continuous speech, speaker-independent system.

We believe that with the core components of CALVIN in place we can begin to ask two groups of interesting questions.
With respect to multi-perspective collaboration:

1. What is the most appropriate interface for coordinating multiple perspectives for design, and for visualization in general.

2. Is there an increase in performance and an increase in the quality of the design that results from a multi-perspective approach, compared to a single perspective approach?

3. Does multi-perspective information filtering instead, cause confusion amongst remote collaborators? If so how can this be resolved?

With respect to persistent virtual worlds:

1. Is there a benefit to working in a persistent virtual world, rather than a temporally finite virtual world?

2. What is the most appropriate venue for a persistent virtual world (for example a virtual landscape, a confined virtual laboratory, or a conference room)?

3. What is the role of time in a persistent virtual world? For example, can time be reversed arbitrarily so that previous changes of the world can be re-examined? What is the most appropriate interface for effecting this change?

4. How do users who change the virtual world inform participants who enter the world at a later time, what changes have occurred? For example, should there be a virtual newspaper with headlines or perhaps virtual post-it notes?

5. Should participants only share one world or can they hold private spaces that they can maintain and permit others to see? For example, can users build sub-designs that may be later integrated with the main design?

Acknowledgements

We would like to thank all the students and architects who generously shared their views; in particular Larry M. Silva. We would especially like to thank our collaborators, Michael Kelley, Bruce Gibeson and Steve Grinavic at University of Southern California and ARPA.

This research is partially supported by NSF grant CDA-9303433 which includes support from ARPA.

References

[1] E. A. Bier, M. C. Stone, K. Pier, W. Buxton, and T. DeRose. Toolglass and Magic Lenses: The see-through interface. In J. T. Kajiya, editor, *Computer Graphics (SIGGRAPH '93 Proceedings)*, volume 27, pages 73–80, Aug. 1993.

[2] F. P. Brooks, Jr. Walkthrough — A dynamic graphics system for simulating virtual buildings. In F. Crow and S. M. Pizer, editors, *Proceedings of 1986 Workshop on Interactive 3D Graphics*, pages 9–21, Oct. 1986.

[3] C. Carlsson and O. Hagsand. DIVE - a multi-user virtual reality system. In *Proceedings of the IEEE Virtual Reality Annual International Symposium*, 1993.

[4] S. Fisher. The ames virtual environment workstation (view). In *Course Notes 29*. 1989.

[5] T. A. Funkhouser, C. H. Sequin, and S. J. Teller. Management of large amounts of data in interactive building walkthroughs. In D. Zeltzer, editor, *Computer Graphics (1992 Symposium on Interactive 3D Graphics)*, volume 25, pages 11–20, Mar. 1992.

[6] C. Heeter. Being there: The subjective experience of presence. *Presence: Teleoperators and Virtual Environments*, 1(2):262–271, 1992.

[7] R. H. Jacoby and S. R. Ellis. Course notes 9. In *Using Virtual Menus in a Virtual Environment, in Implementation of Virtual Environments*, pages 12.1–12.9. SIGGRAPH, July 1992.

[8] L. Kjelldahl and J. Lundequist. Computer aided architectural design work. In H. J. Bullinger and B. Shackel, editors, *Human-Computer Interaction - INTERACT'87*, pages 1097–1100, 1987.

[9] B. Laurel and R. Strickland. Placeholder: Landscape and narrative in virtual environments. In *Multimedia'94*, pages 121–132, Oct. 1994.

[10] D. A. Norman. *The Design of Everyday Things*. Doubleday, 1988.

[11] J. S. Olson, L. A. Mack, and P. Wellner. Concurrent editing: The groups interface. In D. D. et al., editor, *Human-Computer Interaction - INTERACT'90*, pages 835–840, 1990.

[12] G. Singh, L. Serra, W. Png, and H. Ng. BrickNet: A software toolkit for network-based virtual worlds. *Presence: Teleoperators and Virtual Environments*, 3(1):19–34, 1994.

[13] R. Stoakley, M. J. Conway, and R. Pausch. Virtual reality on a WIM: Interactive worlds in miniature. In *SIGCHI '95 Proceedings*, 1995.

[14] C. Vasilakis. CASA- Computer Augmentation for Smart Architectonics: Masters of Fine Arts thesis and performance art piece at Electronic Visualization Event 4, University of Illinois at Chicago, 1995.

[15] D. J. Weintraub, R. F. Haines, and R. J. Randle. Head-up display (HUD) utility, ii: Runway to HUD transitions monitoring eye focus and decision times. In *Proceedings of the Human Factors Society*, pages 615–619, 1985.

[16] C. D. Wickens and L. J. Long. Conformal symbology, attention shifts, and the head-up display. In *Proceedings of the Human Factors and Ergonomics Society*, pages 6–10, Oct. 1994.

[17] M. J. Zyda, D. R. Pratt, J. G. Monahan, and K. P. Wilson. NPSNET: Constructing a 3D virtual world. In *Proceedings of the 1992 Symposium on Interative 3D Graphics*, pages 147–156, 1992.