Bringing Real Applications to the Virtual Environment

David Frerichs - Future Vision Technologies, Inc.
Virtual Reality World, July/August 1994 pp. 50-54

If virtual environment (VE) technology is such a quantum leap in the evolution of the human/computer interface, then why haven't more developers worked to incorporate it into their applications? The most common answer to this question is the technology. It is too cumbersome, complex or exotic for most uses. While this is true of the full immersion systems, it is not true of a through-the-window interface where a standard mouse and monitor allows the user to access the virtual world. So the claim that synthetic environments are not practical falls through.

What is the real problem? The problem is that virtual environment technology presents a completely different model for building applications. The model is so different that there is no clearly defined paradigm for even defining what the application should be like, let alone what types of problems the application can solve.

Most developers today who are bold enough to use the technology are heading off into the wilderness of VE development using maps designed for the two-dimensional desktop, or even worse, maps designed for the text interface. The end result is that current virtual environment applications end up being exactly the opposite of what was intended; they are non-intuitive.

A new paradigm for developing applications in the 3D environment must be defined in order to open the VE wilderness for development. Clay Graham of PowerSpace in San Francisco, CA has come up with such a paradigm called Virtual Architecture. Virtual Architecture uses the principles of physical architecture to define virtual worlds. The Virtual Architect uses architecture's understanding of basic human interaction of structures and their conveyance of purpose to craft worlds which are not only the ultimate in functionality but are also comfortable to inhabit.

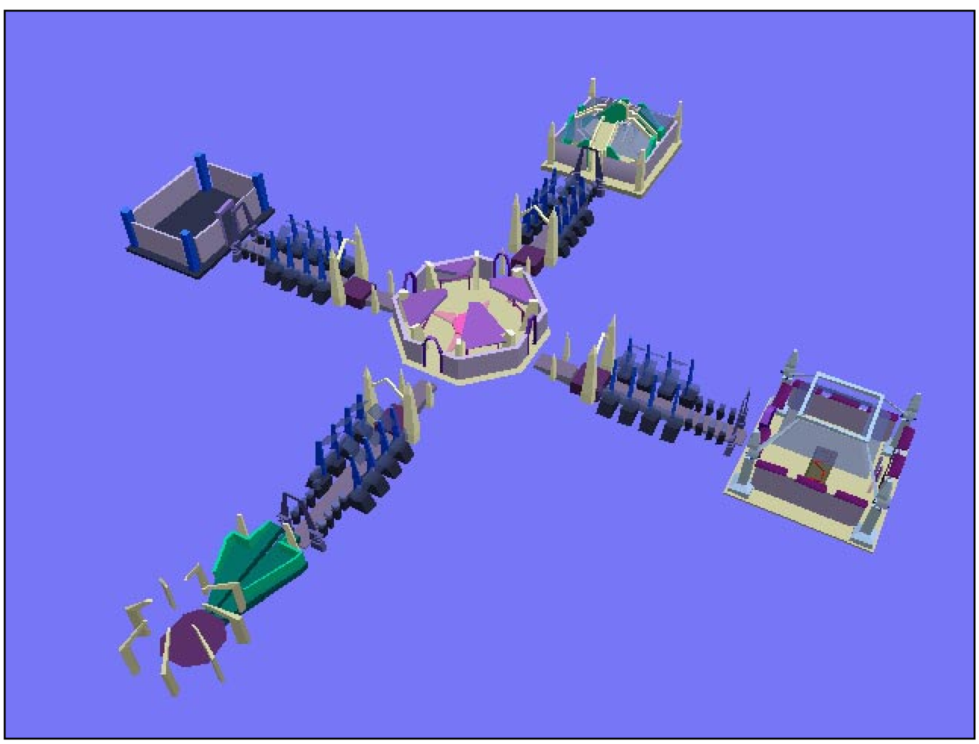

Figure 1: This is an aerial view of the application spaces as they are spatially arranged. Only one of the geometry constructs is visible at any time, this file is provided as a reference of the relationship between the spaces.

Virtual Architecture

Long before people had any idea about file systems and data structures, they had a solid grasp of concepts like sky, ground, room, and threshold. Capitalizing on what can be called the user's natural sense of "space" and "place," virtual architects can develop worlds which generate instant recognition of their purpose in the user. Virtual Architecture uses archetypes in order to convey the purpose of the space. Archetypes are a low-level symbolic language, understood at the base of the human psyche, which the virtual architect uses to encode meaning. Here are ten archetypes:

Axis: Infers a path to a goal and begs to be followed.
Court: Focus of action/inaction and cognitive awareness, a container of thresholds.
Relationship of path and place: The journey with a first threshold (initiation) and a final threshold (enlightenment).
Perimeter of domain: the limit of understanding, the limit of the defined world.
Portal: a breach in the perimeter, the mouth of an axis connecting two domains.
Openness and closure: defined by the perimeter, openness is the lack of understanding, closure is the comfortable area of knowledge.
Datum: Spatial datum define the scale and perimeter of the space, secondary datum may define items of interaction and are offset to show importance.
Column: Defines the perimeter and sets the vertical scale.
Wall: Primary method of defining the perimeter, also provides closure and clarifies breaches or portals.
Roof: Preserves the relationship of sky and ground and provides the elevation boundary for the area of interactivity.

In applications which use these building blocks, the task and meaning of the space is immediately obvious. In addition to archetypes, additional basics have been defined. The space must be aware of the user, assisting and reacting wherever possible and always preserving the integrity of location, movement and transition.

An obvious result of this is the elimination of instantaneous "teleportation" in virtual worlds. Jumps from place to place, while sounding wonderfully practical, end up confusing the user, especially in an immersive environment. Speedy, even magical, transportation is acceptable as long as care is taken to preserve the spatial integrity of the journey.

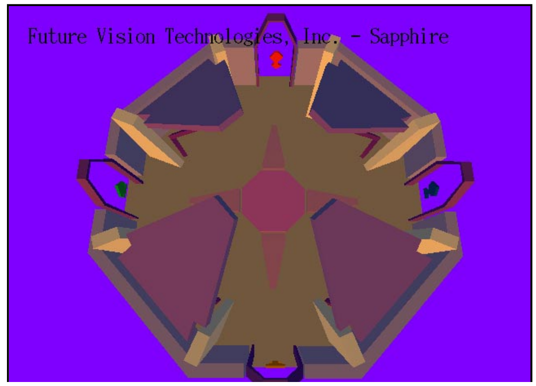

Figure 2: Entry room for the Toolset program. Each of the doors represents a separate application.

Cost analysis is also basic to Virtual Architecture. The virtual architect must follow a preset budget of different types of polygons, lights, textures and input devices. By following the new map of Virtual Architecture, developers can create applications which fully utilize the power of virtual environments.

Creating an Application Using Virtual Architecture

Creating an application using Virtual Architecture is closed-loop process between the developer and the virtual architect. The developer defines the application requirements and brings them to the virtual architect. The architect then defines the space including environmental geometry as well as the graphical and non-graphical items of interaction. The developer then uses this definition to implement an initial version. Each version is reviewed by the virtual architect and recommendations are made until the final goal is achieved.

In a joint project between Future Vision Technologies, Inc. and PowerSpace, two applications were developed using the Virtual Architecture paradigm: Toolset and No-Space. PowerSpace defined the paradigm and geometry in Autodesk 3D Studio™ and Future Vision Technologies implemented the application using Autodesk Cyberspace Developer Kit™ running on the Sapphire IME.

Cyberspace Developer Kit combined with Sapphire provides unique features which are crucial to Virtual Architecture. Items like object-oriented simulation capabilities, advanced surface definition support and interactive audio allowed the applications to go beyond the initial goals.

Figure 3: One of the transition corridors which links each application to the main room.

Toolset: Basic Computing in VE

Toolset was designed to bring real computing tasks in to the virtual environment and is the first application to be implemented using the Virtual Architecture paradigm. It is meant to open the mind of software developers to the real possibilities of interactive 3D environments for more than just CAD applications.

Toolset is a combination of related spaces each conveying a specific purpose. The user enters the application in Nexus. Nexus is a court containing four portals which lead to different task spaces. The portals are identified by a three-dimensional icon, allowing the user to differentiate them. This space is where the user first experiences the immersion and comfort aspects of Virtual Architecture.

Although the interface is through-the-window, a looping low-volume sound increases the users attention to and immersion in the application. The color and lighting model have also been chosen with care. Toolset uses a congruent, soothing color scheme to offset the color shock often experienced in other virtual environments and a sunset lighting model creates a familiar atmosphere.

The portal icons are proximity activated. By flying into them, the user begins the transition and is transported down a corridor to the task space. Rhythmic sound and geometry come together in the corridor to convey a sense of travel which possesses spatial integrity. On the other side, the user steps through a second portal into the target task space.

Toolset Task Spaces

Finder was the first task space created. Based on the belief that a two-dimensional interface is far from the optimal way to access files, Finder is an interactive, three-dimensional representation of the DOS file system. Individual files in the current directory are represented as cubes and sub-directories are represented as pyramids. File objects are color coded and spatially arranged (general X,Y area) according to extension, they spiral upward according to their age (higher is older), their volume is logarithmically scaled to their file size and they are opaque or translucent in accordance with their recognition.

Figure 4: A view of the file system structure inside of the "finder" application.

Upon activation, the files organize themselves with like file types congregating in clusters around the room. Unrecognized file types remain opaque and stay in the center of the room along with sub-directories. A 3D selector object allows users to grab files and move them around. The file objects have memory and know their place. If the homing option is enabled, the files will anxiously return to their proper location once released by the user. The 3D cursor, which floats in front of the user, is used in two other task spaces. It is a vast improvement over other ray-casting types of cursors used in other through-the-window applications and is the primary user interface tool in Toolset. Finder geometry provides a sense of airy closure through the use of a translucent roof combined with column-bound walls. The walls emphasize the portal icon and the path back to Nexus.

Figure 5: Inside the data visualization room. Each cube represents a piece of data.

The second task space uses 3D icons for data storage and retrieval. Unlike other storage rooms, this one is designed to catch the light in interesting ways and add pleasure to a possibly tedious task.

Translucent roof corners are offset by arching columns reaching overhead. Each set of columns partitions the space into smaller areas, organizing the data spatially. Data is represented by texture-mapped cubes which rotate to signify their importance and their readiness. In Toolset, these items represent sounds. If the user places the cursor inside the cube and selects it, the data is accessed and the sound is played.

This type of storage and retrieval can be applied to other types data like compressed geometries, video clips and text. It also lends itself well to visual representation of hypertext-like correlation of documents where logical links between sets of data can actually be seen. Once the task is

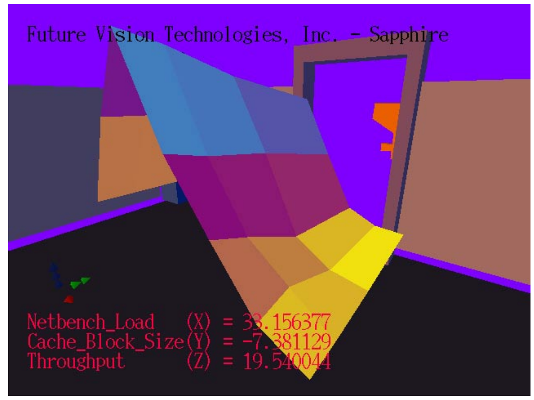

Figure 6: The spreadsheet room. The structure is a four dimensional graph of imported Excel spreadsheet data.

completed, the user simply returns to Nexus through the portal.

Data gathering technology has outpaced data interpretation technology leading to a backlog of potentially useful information. The third task space applies virtual environments toward a solution to this problem. This Data Visualizer imports spreadsheet elevation style data and displays it as a surface. Points on the surface are color-coded according to elevation and multiple sets of data can be loaded at once.

The axis origin is indicated by an icon which uses both color and shape to differentiate X,Y and Z. Axis labels and user location are displayed in a heads-up fashion in the corner of the screen. The user moves around the data and is able to inspect it from any

Figure 7: An aerial view of the art gallery before any exhibits have been experienced.

viewpoint. In addition, the data can be animated through multiple sets. These points allow the user to interpret the data faster and to see trends which would otherwise remain invisible.

The goal is to move visualization from a comparison of points and numbers to a right-brain interpretation of shape. The space geometry in Data Visualizer is simple in order to emphasize the data objects. Plain walls and floor provide closure and scale reference and a portal allows the user to exit the task.

Figure 8: In this view, some of the exhibits have been experienced, as indicated by the partially assembled room.

A multi-media Gallery occupies the fourth task space. This space is an extension of the data storage paradigm and provides the user simultaneous access to related data items. A walkway extends from the entry portal to a central platform surrounded by arching columns. In between the columns are seven unique 3D icons which spin in sync to indicate their importance and correlation. The user places the selector into the icon and selects it. Once activated a piece of the floor and wall associated with that location rolls in and snaps into place. Then the visual portion of the exhibit comes up and its corresponding sound begins. The user selects the icon again to close the piece and then moves on to the next. As the user proceeds, the space assembles itself. Once the space is complete, the user knows all exhibits have been experienced and the task is over.

Toolset is a work in progress and will continue to be expanded and refined in order to become more functional. Logical next steps will include the accommodation of multiple users with the concepts of private and shared spaces. The basic framework of the application is extensible and forms a kind of high level VE operating system. It will continue to grow as more tasks are defined.

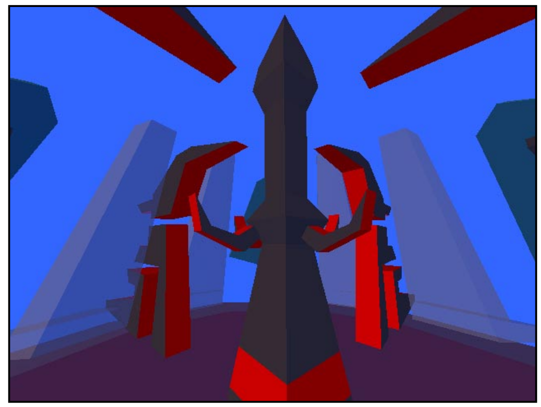

Figure 9: As the viewer approaches the sublime mechanism, it lays in wait - open like a carnivorous flower.

No-Space: Conveying an Abstract Idea

Virtual Architecture can also be applied to non-task-oriented environments. No-Space is included as an example of a space unto itself. A departure from the standard notion of "application", it is a purely meditative environment designed to convey an idea.

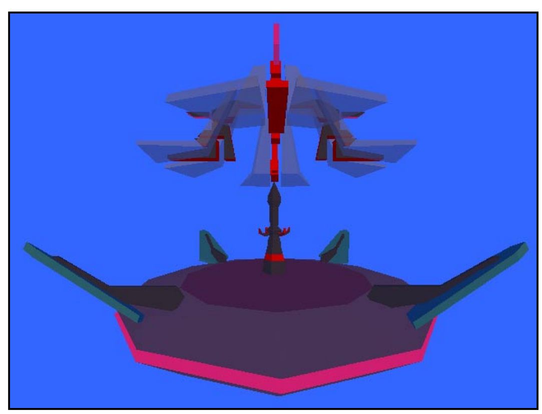

Figure 10: Inside the sublime mechanism after it has closed around the viewer.

The virtual environment equivalent to a poem. No-Space contains the sublime mechanism which is aware of the user. Initially it is open and inviting like a metallic flower. Upon approaching, a dim sound becomes apparent. As the user draws near, the familiar haunting sound becomes louder and its source is apparent, a spinning totem in the center of the flower. When the user enters the mechanism, the petals close, it ratchets shut and slams the user in. What was once open and inviting is now a tightly closed space filled with tension and sound. Finally, as the user backs away, the petals open again and allow the user to contemplate what has happened.

Toolset and No-Space are the first applications of their kind. They utilize the solid principles of Virtual Architecture to provide users with an intuitive, useful solutions to real computing problems. Virtual Architecture opens the way for developers to begin to stretch their minds and conceptualize how virtual environment technology can be applied to advance basic computing.