Sewing Worlds Together With SEAMS: A Mechanism To Construct Complex Virtual Environments

Dieter Schmalstieg[1] and Gernot Schaufler[2]

1. Introduction

The idea of using virtual environments as a communication medium or as a work environment is found intriguing by many supporters, and it has already been demonstrated that such technology can be successfully used for large audiences (Pausch et al., 1996). However, widespread commercial success of virtual environments is still not evident.

Why is it that the idea of *Cyberspace* as first presented over a decade ago in William Gibson's novel "Neuromancer" (Gibson, 1984) is so enormously appealing, yet we see little manifestation of these ideas in our everyday technological environment? At least a partial answer may be found by comparing a future Cyberspace to the most successful networked application of today, the World Wide Web (WWW).

Distributed content development: One important property of the WWW that sets it apart from all previous media is the possibility of relatively simple participation by every user. The possibility to communicate with everyone else instead of just consuming the presented content (as with, for example, TV) fuels both individual and commercial interest and leads to a continuing fast growth of the Internet, the largest computer network to date. Distributed content development has the tremendous organizational advantage that no central authority is required for coordination, a prerequisite known to slow down development efforts. This is a key factor in building truly scaleable virtual environments. For virtual environments, this means that users should be able to create 3D environments and bring them online to allow others to interactively explore their creations and use them as a meeting place.

Nature of the medium: The document oriented nature of the WWW makes the use of hyperlinks a natural choice for navigation. We are used to the finite extent of documents, and to cross-references within them that help us to navigate through larger collections of data. In contrast, a virtual environment mimics 3D space, which is not per se decomposed into individual regions, but continuous. Unfortunately, distributed development requires that individuals concurrently and independently work on different regions of the virtual environment. These regions can be assembled into a continuous whole, but this approach requires a central coordination authority to answer the question who gets to develop which region, which is difficult if not impossible to achieve on a global scale - the WWW of today is already troubled by the comparatively simple issue of domain names.

Alternatively, a virtual environment may be broken down into parts without continuity among them. This has the advantage that unlimited space is available for every part, and is the approach taken by today's virtual environments on the Internet, based on the Virtual

[1] Vienna University of Technology, Vienna, Austria; email: dieter@cg.tuwien.ac.at
[2] Johannes Kepler University, Linz, Austria; email: gs@gup.uni-linz.ac.at

Reality Modeling Language VRML (Hartman & Wernecke, 1996). Traveling from one worlds to another is done by teleportation - instant transport to another position, which in VMRL is coupled to so-called *anchor* objects, which allow the user to trigger the teleportation. In our experience, the use of teleportation is often confusing as one often triggers it accidentally and it does not have any correspondence to real-world travel. In (Bowman et al., 1997), the only formal evaluation documented in the literature is made on the issue, and our postulation that teleportation leads to disorientation is confirmed. However, teleportation has the advantage over continuous virtual environments that large distances can be covered in an instant, a property which we would like to retain.

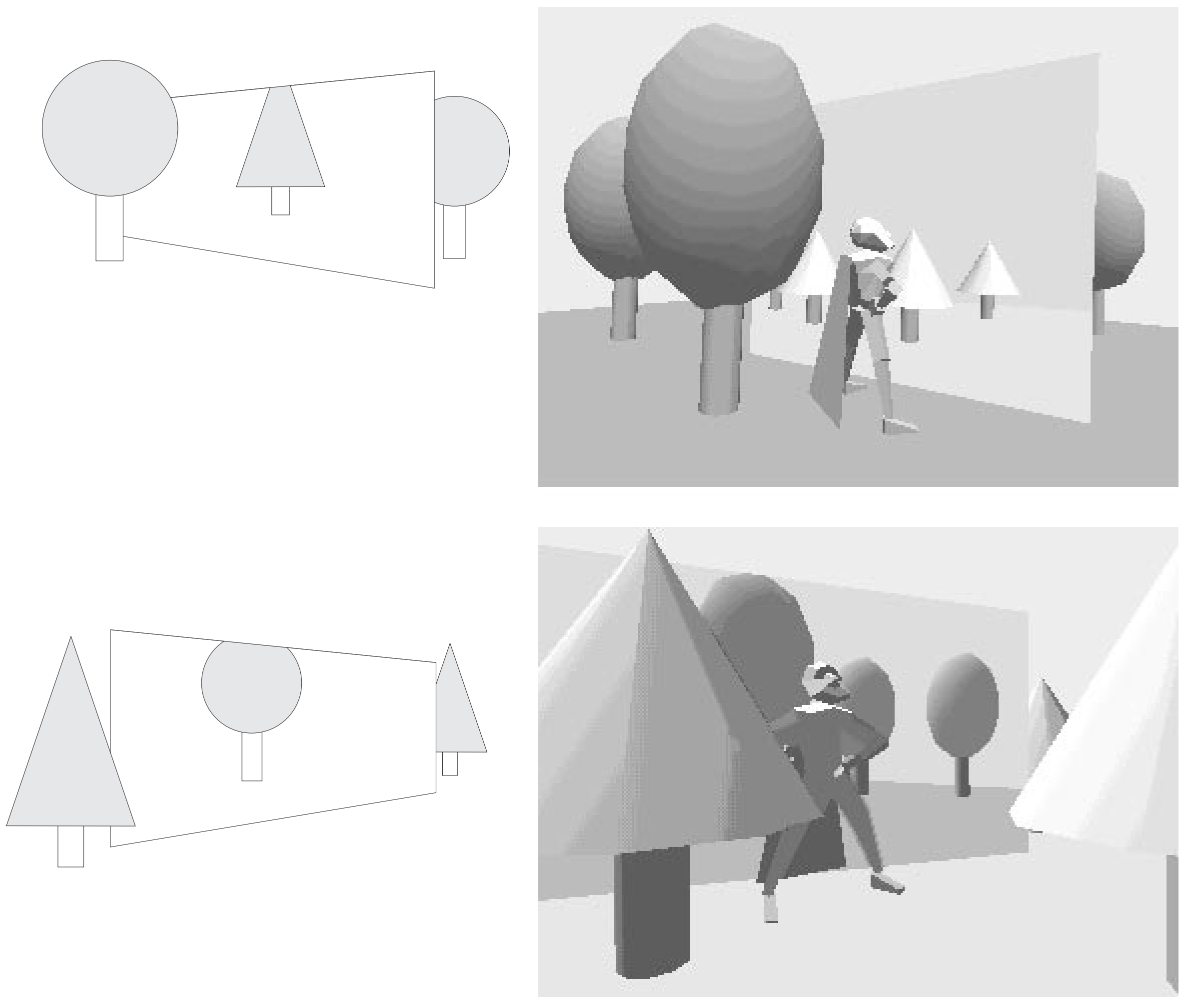


Figure 1: SEAMS allow to see and go from one world (round trees) into another (cone trees) and vice versa. Shown is a schematic view (left) and a screen shot (right)

In this paper, we will present a new mechanism for the construction, organization and navigation of complex virtual environments. The *Spatially Extended Anchor Mechanism* (SEAM) is a tool for the distributed development of continuous virtual environments without the need for a central authority. It allows instant travel over large distances without a disruption of continuity (Figure 1). We will point out how it can be used in distributed virtual environments of different origin and intent, from simple current VRML browsers to environments for large user communities in the sense of NPSNET (Macedonia, 1995). We will also demonstrate how SEAMS are a superior mechanism for building immersive user interfaces based on 3D Magic Lenses (similar in spirit to Viega et al., 1996), which can be used to create another type of complex virtual environment.

2. The Spatially Extended Anchor Mechanism (SEAM)

The SEAM is a new modeling primitive to define the relationship between virtual worlds. A SEAM may be defined as a “door into another world”. An example is shown in Figure 1. The name was chosen to indicate that it is an extension of the well-known WWW anchors of VRML and because it expresses the function of geometrically connecting different spaces together. The extent of the SEAM is defined by a single polygon, the SEAM polygon, which can be arbitrarily positioned. A complete SEAM definition is composed of a scene polygon, a reference (URL) to the world behind the SEAM and a transformation matrix specifying the geometric relationship between the two worlds.

A user may not only look, but also walk through the SEAM to enter another world or reach through the SEAM to manipulate another world. Hence SEAMS are useful in a number of ways: as a mechanism to visually and spatially concatenate worlds, as a navigation metaphor, and as a 3D user interface element.

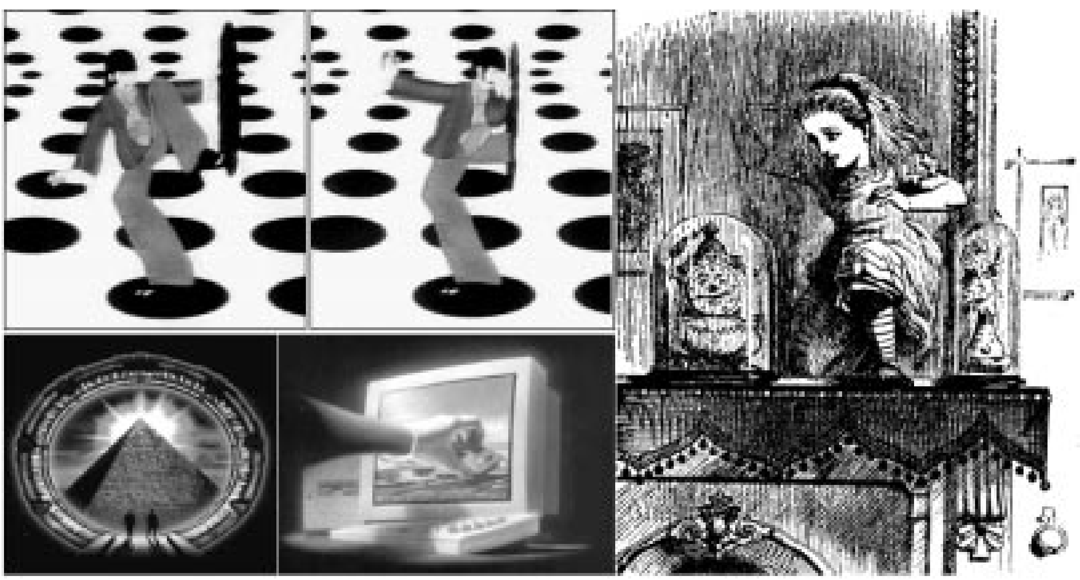


Figure 2: Examples for connections of different worlds - Yellow Submarine (top left), Stargate (bottom left), advertisement by Superscape Inc. (bottom middle), Through the Looking Glass (right)

Using SEAMS to connect virtual environments gives the authors of virtual worlds and the administrators of Internet-based distributed 3D applications great freedom in design and organization of their content. It also solves the problem of disorienting navigation via teleportation, because it is possible to look into the adjacent world before entering it and because there are no discontinuities in movement. Yet SEAMS resemble teleportation in that they allow efficient coverage of large distances. SEAMS allow conventional Euclidean relationships among worlds to be established, but also to connect worlds in ways impossible with Euclidean geometry.

It may appear that navigating via SEAMS is counter-intuitive, because it does not resemble any type of navigation known from reality. However, the idea of "magic mirrors" or "wormholes" is very popular and can easily be grasped even by people not familiar with science fiction or fantasy. Almost everyone has come across movies like "Mary Poppins", "Yellow Submarine" or "Star Gate", books like "Through the Looking Glass" (Carrol, 1876), or TV shows like "Star Trek: Deep Space 9" or "Babylon 5" (Figure 2), and consequently even novice users have no difficulties in understanding the concept.

3. Related work

Previously proposed virtual environment research systems have aimed at the subdivision rather than concatenation of virtual worlds: very large virtual environment are often subdivided into more manageable chunks (called "regions", "cells", "locales") to exploit spatial coherence. This subdivision is used for visibility determination in real-time rendering and for reduction of network communication in distributed virtual environments. To our knowledge, little attention has been paid to using subdivision for modeling and organization of virtual environments.

Closest to our intentions is Diamond Park/Spline, developed at MERL (Barrus et al., 1996). It decomposes the virtual universe into "locales" which are associated with separate coordinate systems. Special attention was paid to precision in modeling and simulation. The work also introduced some considerations for world modeling and arrangement of locales that are similar to SEAMS, but avoids the problems of visibility and overlapping worlds by using bent corridors and anterooms.

Visibility preprocessing for real-time rendering was first advocated in (Airey et al., 1990). A virtual environment is decomposed into cells for which mutual visibility is precomputed. Several approaches for exploiting visibility to accelerate rendering were proposed, for example in the work of (Teller & Sequin, 1991) and (Luebke & Georges, 1995).

A large body of work has been dedicated to the optimization of network communication, which is crucial for large-scale virtual environments with a large number of participants. Virtual environments are subdivided either regularly, based on visibility or into arbitrary regions. To reduce the amount of communication as much as possible, network protocols have been developed based on a multicasting topology such as NPSNET (Macedonia et al., 1995) or DIVE (Carlsson and Hagsand, 1993), a client-server topology such as NetEffect (Das et al., 1997) or hybrids of both such as RING (Funkhouser, 1996) or Community Place (Lea et al., 1997). Multicasting allows efficient delivery of messages to many participants. However, it looses much of its efficiency when transported over a wide area network, and is not generally supported on today's Internet. Furthermore, maintenance of multicast group membership is costly in terms of performance and only tractable for special types of application (Macedonia, 1995). Client-server architectures perform efficient message filtering, but can suffer from bottlenecks if the server is overloaded. Hybrid topologies can merge the advantages of both, but no single approach has yet been adopted by Internet users as a standard. In section 6 we will discuss how SEAMS can be a useful tool for implementing distributed virtual environments independent of the networking approach.

4. Rendering SEAMS

In this section, we explain how SEAMS can efficiently be rendered using standard hardware acceleration. Technically, a SEAM is a polygon through which a live image of another world is visible. Currently we employ one polygon per SEAM, but there is no conceptual reason why a SEAM could not be a polygonal mesh or a non-planar primitive. The polygonal nature of the SEAM, however, allows us to exploit the capabilities of hardware-assisted polygon renderers (Neider et al., 1993).

4.1 Terminology

To aid in the explanation of the rendering method, we will introduce a few terms (see also Figure 3). The world that currently contains the user and the SEAM is called *primary world*, the world behind the SEAM is called *secondary world*. Any world that can only indirectly be accessed via multiple SEAMS is called a *higher order world*. The union of all worlds is called the *virtual universe*. SEAMS can be *unidirectional* (there is only a connection from the primary to the secondary world), or *bidirectional* (the connection is two-way). A *one-sided* SEAM can only be observed from the front, and is invisible from the back (in the primary world!), whereas a *two-sided* SEAM is visible from the back - it may show the secondary world, but it may also be rendered opaque, e. g., as a grey polygon.

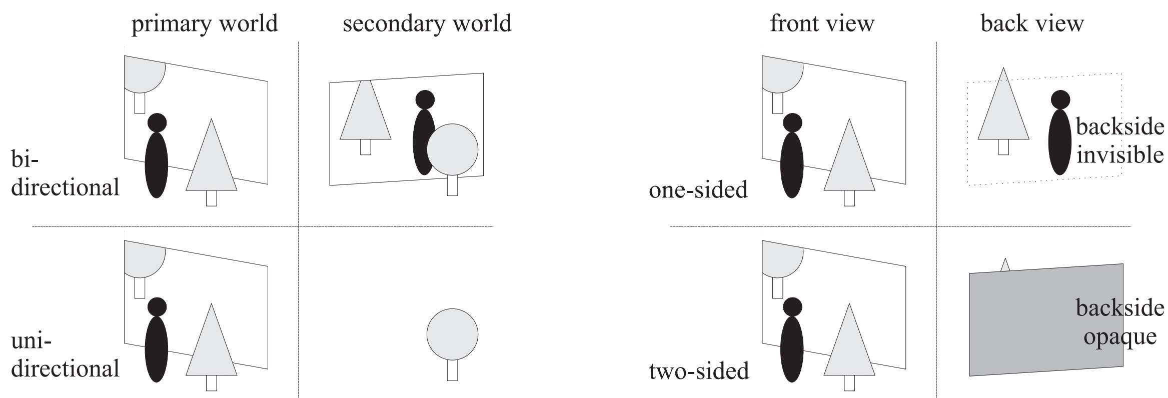


Figure 3: Different types of SEAMS

4.2 Rendering a single SEAM

The standard approach to rendering three-dimensional scenes (the primary world) is to render each object in turn, solving the problem of occlusion with a Z-buffer. A SEAM object requires us to render a picture of yet another scene (the secondary world) onto the SEAM's surface in a view dependent manner.

Our approach for rendering a SEAM generates the image of the secondary world at the right position in the image of the primary world. For correct visibility, rendering of the image of the secondary world must be restricted to the area covered by the visible portion of the SEAM polygon in the image of the primary world (Figure 4). This approach is similar to a method suggested in (McReynolds and Blythe, 1997).

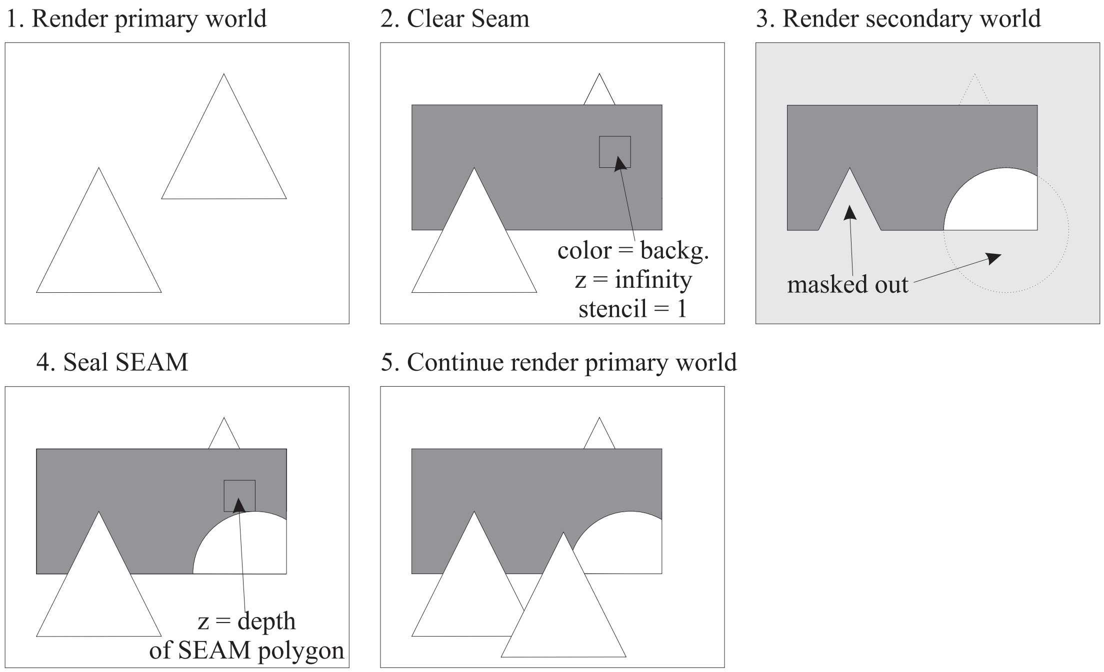


Figure 4: Process of rendering a SEAM

The mentioned confinement of the secondary world's image to the area of the SEAM polygon is realized using a mask in the so-called stencil buffer (Neider et al., 1993), which can allow or disallow graphics output on a per pixel base. The details of the rendering are as follows (Figure 4):

The geometry is given as a directed acyclic[3] graph (scene graph). The scene graph of the primary world is traversed and rendered. When a SEAM is encountered, the associated polygon is passed to the rendering hardware for scan conversion. For all pixels of the SEAM polygon found to be visible: (1) the mask (stencil buffer) is set to 1, (2) The frame buffer is set to the background color of the secondary world (clear screen), (3) the Z-buffer is set to infinity (clear Z-buffer). Note that these image modifications are only carried out for the visible portion of the SEAM surface and that steps (2) and (3) can be carried out in a single step.

After this preparation step, rendering of the secondary world is performed inside the stencil mask created in the previous step (so that the secondary world is not drawn outside the SEAM area), and with a clipping plane coincident with the SEAM polygon (so that the secondary world does not protrude from the SEAM).

Finally - before rendering of the primary world proceeds - the SEAM polygon is rendered again, but only the computed depth values are written into the Z-buffer. Thereby the SEAM is "sealed". The resulting Z-values are all smaller than any Z-value of the secondary world. Note that no geometric primitive of the primary world located behind the SEAM will overwrite a visible pixel from the secondary world rendering.

---

[3] May be a directed *cyclic* graph if recursive references are allowed (secondary or higher order world is the same as primary world)

4.3 Rendering nested SEAMS

In a more complex case a number of virtual worlds is connected using multiple SEAMS (Figure 5). From certain viewpoints, users may be able to see not only a secondary world, but another SEAM leading to a third world, and so forth. Evidently, we need to modify the rendering algorithm to support recursive traversal of worlds during rendering. This is relatively straightforward. Traversal of multiple worlds connected by SEAMS is done in depth first order, starting from the primary world. For every world, the algorithm proceeds as outlined above, except that the mask for a SEAM is created by incrementing the stencil mask value by one in the process of getting to the SEAM under consideration, thereby progressively reducing the set of pixels that may be effected. The visible area of the SEAM consists of the pixels where the mask value is equal to the recursion depth necessary to reach the SEAM.

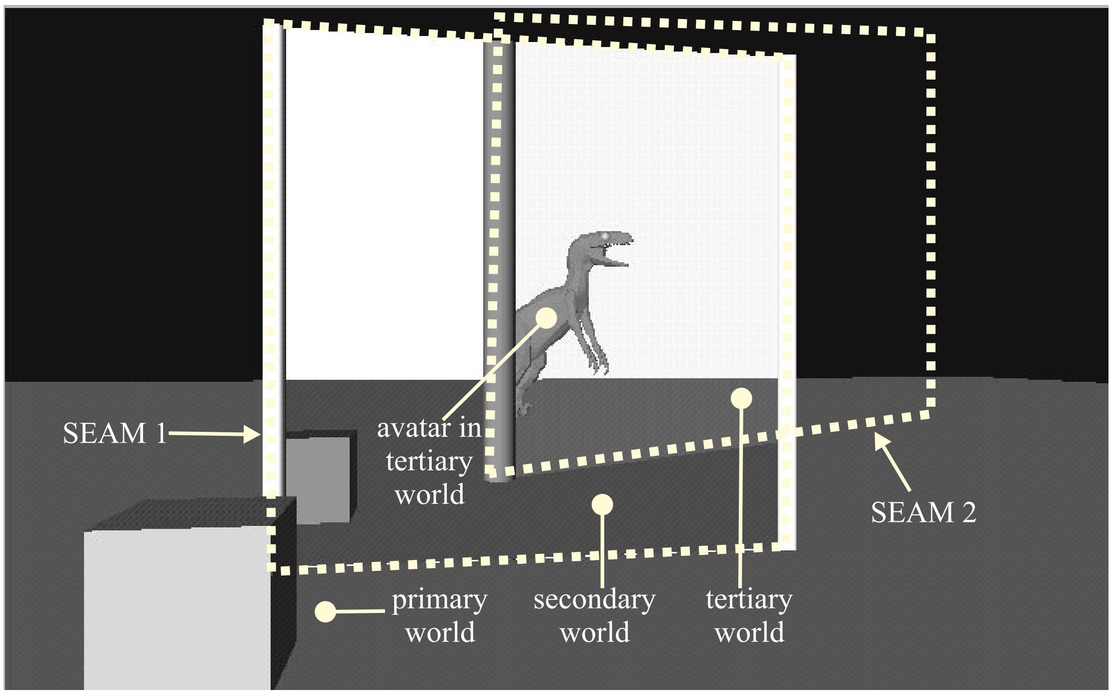


Figure 5: Nested SEAMS

An efficient implementation requires that only those worlds of which at least a small portion is visible are processed by the rendering algorithm. Otherwise a large universe of concatenated worlds would require a potentially infinite number of primitives to be rendered, most of which are not visible at all. The problem is equivalent to the visibility problem for occluded interiors. Potentially visible set (PVS) methods (Airey et al., 1990) decompose a large geometric database into regions with visibility coherence (most of the geometry contained in such a region is visible from most viewpoints inside the region). Fortunately, the difficult part of PVS algorithms - identifying reasonable regions and the portals that connect them - is trivially solved for a virtual universe connected by SEAMS, because regions correspond to worlds and portals correspond to SEAMS, which are both included in the world description.

The simple and efficient algorithm proposed in (Luebke & Georges, 1995) solves the visibility problem in real time. It projects the portals (SEAMS) into screen space and intersects the screen-aligned bounding boxes (bbox). Typically only a few SEAMS are visible in sequence, and so the aggregated bbox quickly decreases in size (Figure 6). An empty aggregated bbox indicates that no object can be visible through the considered sequence of SEAMS from the given viewpoint. Further optimization is done by determining the visibility of individual objects against the aggregated bbox. This is useful both for rendering and networking as scene graph traversal and rendering and network communication are only necessary for the visible portions of the universe.

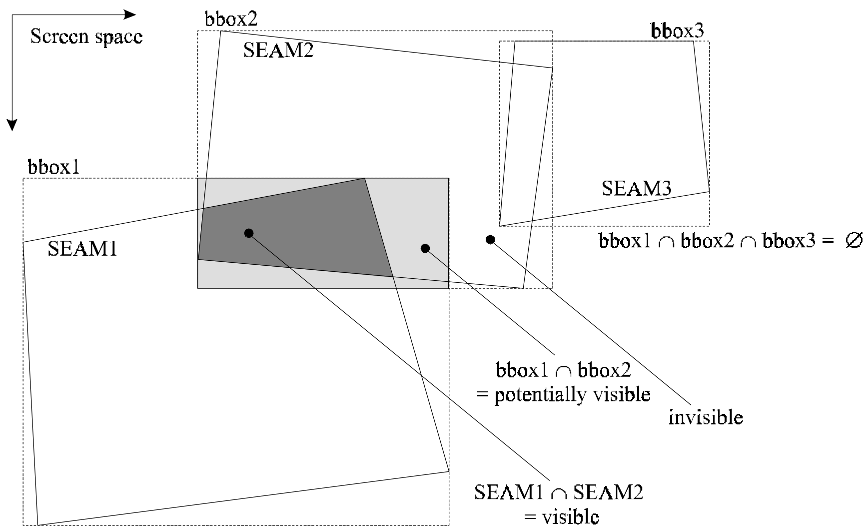


Figure 6: Visibility algorithm

5. SEAMS in distributed and multi-user virtual environments

SEAMS were conceived to support a virtual universe inhabited by a very large community of users. Large-scale multi user virtual environments require a careful design of the underlying network architecture. Using today's Internet technology, multi-user virtual environments are mostly being built in client-server technology. A server may continue to simulate a virtual world, even if no participants are present in it. Each server provides via the Internet a part of the virtual universe (a world). It is responsible for the simulation of this world and can decide on the development of the simulation within this world.

A network architecture for the management of a distributed virtual universe faces a number of problems that may severely compromise performance and scalability. In the following we discuss how to share a virtual universe constructed using SEAMS with Internet users and support live interaction of potentially large audiences.

5.1 Building a continuous distributed universe

Consider the simple case of a single user traveling in a virtual universe composed of individual worlds connected via SEAMS. If multiple users are present in this universe, let us assume for the moment that they cannot perceive each other. Network communication for this configuration is only necessary to transmit the scene descriptions to the user. The user's client manages replicas of the primary world and of those higher order worlds that can be seen through the SEAMS. This approach leaves the client with the simple task of rendering the worlds for the user. Servers need to consider SEAMS only so far, as they have to instruct connected clients of the SEAMS' position and secondary worlds behind them.

Network communication can be reduced by transmitting only those parts of a world that are actually required for rendering (Schmalstieg & Gervautz, 1996). As the transmission of a world description may take some time, clients must anticipate their needs and prefetch data to have it available in time for rendering. The primary world must be available in any case, but download of higher order worlds can be delayed. Only potentially visible worlds need to be considered for downloading, similar to the strategy presented in (Funkhouser et al., 1992).

5.2 Compatibility with multi-user network architectures

In this section, we discuss how multi-user network architecutes work together with SEAMS. We first give a review of the approaches to virtual environment netwoking briefly mentioned in section 3, and show how they can be used together with SEAMS. We will also explain how multi-user applications can be constructed from these networking approaches.

For true multi-user support, a user must be able to perceive the activities of other users as well as the static world description. Every user is assigned a visual representation in the virtual environment, the *avatar*. For correct simulation, users must be aware of each other, which has been called the *players and ghosts* approach to simulation (Blau et al., 1992). User have to keep each other informed about their respective actions, in particular changes in position.

In the implementation of multi-user virtual environments, an important issue is scalability. A substantial body of related work has been dedicated to the optimization of network communication. In the following, we discuss one example for each of the relevant approaches.

- **Multicasting in NPSNET.** NPSNET (Macedonia, 1995) is a system for large-scale military simulation. It uses a hexagonal decomposition of the area. Each hexagonal region is associated with a network multicast group, so that simulation updates (e. g., avatar position) in a particular region are communicated to affected participants only.
- **Hybrid client-server/peer-to-peer topology in RING.** Funkhouser's RING system (Funkhouser, 1996) is aimed at the simulation of densely occluded environments (e. g., building interiors). It uses a hybrid client-server/peer-to-peer network topology.
- **Client-server architectures on the internet.** NetEffect (Das et al., 1997) and Internet game servers (Origin, 1997) are virtual environments specifically aimed at simultaneous participation of thousands of simultaneous users via the Internet. It is based on a client-server architecture, and uses sophisticated methods such as message filtering and dynamic load balancing in a network of servers to achieve a high degree of scalability.

One may not overlook that the choice for client-server architectures on the internet is also guided by commercial considerations such as access control and billing. However, currently operational solutions indicate the assumed scalability problem of servers (they may become a bottleneck) does not seem as pressing as some years ago.

We stress that SEAMS are compatible with all networking architectures outlined above as long as they subdivide a virtual universe into individual worlds. SEAMS do not resolve the scalability issue itself, but they provide an important building block in the design of large-scale multi-user virtual environments. Constructing a virtual universe with SEAMS also allows to exploit visibility constraints as outlined in section 4.3 to reduce the amount of necessary communication and hence improve scalability.

To summarize, SEAMS extend the general possibilities of a multi-user, multi-world implementation with the following options:

- Non-Euclidean connections between worlds can be created and allow the construction of worlds other than the typical building interiors or dungeons. Arbitrary SEAMS can be used in virtual worlds in the same way anchors are now used in VRML - in fact, a VRML browser may be extended to render anchors as SEAMS rather than static objects. TV screens, billboards and other artifacts can become doors to other worlds (compare color plate 1).
- Maintenance of the virtual environment is completely distributed. Unlike for example RING, there is no need for a global map of the overall environment. New servers can be added to and removed from the running systems, and SEAMS can be opened into existing worlds. Thus, there is a natural match with the structure of the Internet.

5.3 Applying SEAMS in multi-user worlds

SEAMS allow to connect virtual worlds in almost any desired fashion. However, the most useful applications do not necessarily result from exploiting this freedom to the fullest by building bizarre, geometrically impossible universes. A universe that is geometrically plausible may be easier to comprehend, and users will not easily loose orientation. Yet employing SEAMS for such a set-up allows independent modeling of the individual worlds by different parties, and facilitates load distribution onto different servers.

Good examples for such a natural decomposition into adjacent regions include the rooms of a house, with individual rooms modeled as worlds and with SEAMS as doors. A room is a relatively small unit of simulation and can be simulated with great accuracy. The same principle applies on a larger scale to a virtual city. A frequently mentioned application for virtual environments that can be constructed much more elegantly with SEAMS is a virtual shopping mall, where individual vendors offer commerce and clients are invited to browse and make a selection. This type of virtual universe is best presented as geometrically consistent, so that users are not surprised by what they see. Consequently, in such settings SEAMS should always be two-sided, so that users may switch back and forth between worlds as expected. SEAMS should be fitted into doorways or aligned with architectural openings, so that the user does not even notice the transition from one world to another. Free-standing SEAMS and other supernatural constructs are better avoided. This rule may sometimes intentionally be broken: In the shopping mall example, individual shops may be larger from the inside than from the outside, to provide space for shoppers. If users can only access the shop through one SEAM entrance, they may not even notice the implausibility (Figure 7, left). This idea has also been discussed in (Barrus et al., 1996), but their system cannot accommodate neighboring overlapping worlds, which can easily be achieved with SEAMS.

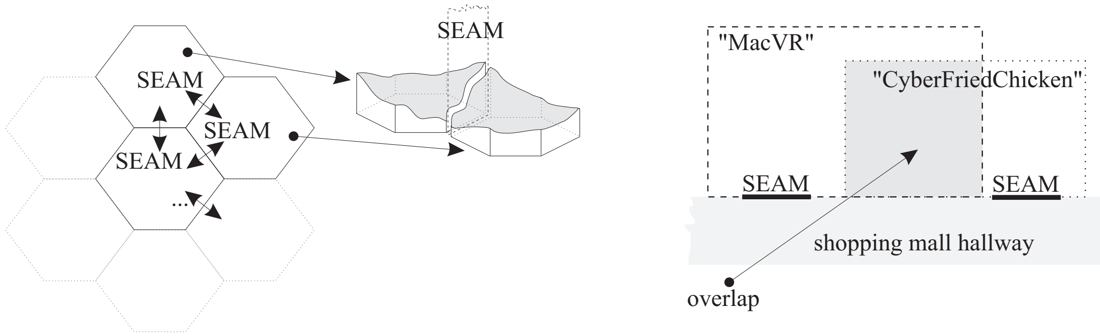


Figure 7: (left) using space in a virtual shopping mall; (right) Hexagonal world decomposition

Regular spatial setups like the hexagonal subdivision of NPSNET can be constructed with large bi-directional SEAMS. Every hexagon is a world with six SEAMS at the borders. Care must be taken that the playfield representation (e. g. terrain) is continuous at the borders (Figure 7, right). As users may theoretically see infinitely far over a flat landscape, there must be an artificial horizon, so that users can only see through a few SEAMS, possibly in combination with fog as an artificial limit.

If plausibility is not of importance, SEAMS can be used as improved 3D hyperlinks. One sided SEAMS can be used to reach any virtual world. In that way, “jump points” that allow to get from one world to another can easily be created, e. g. in the form of a room with multiple doors. The advantage over teleportation is that users are able to see into worlds before they enter them, avoiding confusion and allowing inspection of activities. E. g., a user may peek into multiple “chat worlds” to find out where his or her friends are staying. Users can construct their personal “world access room” linked to other worlds in analogy to a WWW bookmark file. Other applications of this type include a traveling brochure where holiday destinations can be experienced not only from pictures but in 3D (see color plate 1). If one sided SEAMS are used, the users already present in the secondary world should not be confronted with a new avatar that suddenly “pops” into the world, but a more appealing presentation (e. g. fade-in) would be preferable.

6. SEAMS as a user interface tool

Besides their application in virtual worlds, SEAMS can be used as a user interface primitive for immersive or augmented reality environments. They are the 3D equivalent to a 2D window interface, but with the unique property that a user may reach into them to manipulate live 3D applications. In our augmented reality environment “Studierstube” (Szalavári et al., 1996), we have implemented an application that allows to arrange SEAMS in the working range of a user wearing a head-mounted display and a 3D manipulation tool (Szalavári & Gervautz, 1997) (color plate 4).

Furthermore, SEAMS can be used as 3D magic lenses. Magic lenses are unconventional “see-though” user interface elements, that extend the metaphor of a magnifying glass to any sort of useful visual transformation of the data provided by the application. This paradigm was introduced in 2D by (Bier et al., 1993), and later extended to 3D magic lenses by (Viega et al., 1996). In this section, we show that SEAMS are an equivalent to 3D magic lenses, but without the shortcomings and limitations of the original implementation mentioned by Viega et al., as detailed below.

SEAMS are used to display a secondary world somewhere in a primary world. These worlds can have completely different content, but both worlds can also be different representations of the same content. In the latter case, a SEAM is perceived as a lens that modifies the content viewed through it.

For example, color plate 5 shows a flat X-ray magic lens revealing the skeleton underneath the skin of a virtual human. Use of a volumetric lens is shown in color plate 7, where a magic box is used for *focussing*: Streamlines of a complex dynamical system are shown at two different levels of density - a user selected focus defined by the extent of a magic box shows higher streamline density than the surrounding (Fuhrmann & Gröller, 1998). The magic box is composed of six SEAMS.

The rendering method for 3D magic lenses adopted in the original implementation of Viega et al. was based on hardware clipping planes alone and has several drawbacks that severely affect the generality of the approach:

- Trivial rendering is only possible for convex polygons up to six sides. A simple round lens such as depicted in color plate 5 is not possible.
- A separate rendering pass of the same scene is necessary for each side of the lens, which can be a considerable overhead for real-time applications.
- Concave lenses or lenses with more than six sides must be decomposed into convex pieces and require even more rendering passes. The authors report that this solution has not been implemented.

All the mentioned restrictions are easily overcome by using the stencil buffer along with the clipping planes, as our SEAMS implementation does. Only a single rendering pass is necessary, resulting in improved performance. Consequently, it is much easier to develop and use both flat and volumetric magic lenses, as our examples suggest. Ultimately, SEAMS can become the immersive equivalent to windows in conventional desktop systems, as the experiment depicted in color plate 4 suggests.

7. Implementation and Results

Our SEAMS implementation has been done in C++ and Open Inventor (Strauss & Carey, 1992). The SEAM primitive was embedded into what Open Inventor calls a node kit. Using Open Inventor software, our applications run on any platform with OpenGL support. As reasonably powerful OpenGL accelerator cards have recently become available at commodity prices to PC users, there is no obstacle to widespread use.

On top of the mentioned software setup, a simple multi-user virtual environment was constructed (color plate 6). Worlds are constructed according to the VRML or Open Inventor file format, but are linked with SEAMS rather than anchors (as mentioned before, anchors can be interpreted and displayed as SEAMS). The result from an informal evaluation conducted using this implementation is that SEAMS are indeed a useful navigation paradigm generally preferable to instantaneous teleportation.

Our experimental multi-user support was limited to moving through the virtual universe (walkthrough). We have implemented two variants of network support for multiple users: One based on multicasting, and one based on a client-server approach. As expected, multicasting is more efficient in delivering simulation data to the participants for a crowd of avatars that can see each other; message filtering performed by a server delivers better performance for more complex virtual universes with many worlds, where visibility culling on SEAMS can be employed. We set up a trial using a crowd of computer-controlled participants ("bots") together with two human users on two workstations connected via a non-dedicated 10mbps Ethernet and could successfully handle about 50 participants with several position updates per second.

8. Conclusions and future work

We have introduced a new mechanism for connecting virtual environments: SEAMS, essentially a *door* into another world. The key advantages are:

- A technically new, yet culturally familiar metaphor for navigating in virtual environments
- A principle that allows to build and organize complex virtual worlds, and link them to other worlds without a centralized organizing authority
- Application as 3D windows and magic lenses to construct user interfaces

Future work will involve creating more applications that make use of SEAMS and to experiment with different policies for the organization of a virtual universe.

9. Acknowledgments

This work has been supported by the Austrian Science Foundation (*FWF*) under projects no. P-11392-MAT and P-12074-MAT. Special thanks to Anton Fuhrmann and Michael Gervautz for their help and assistance.

Further information and more visual results can be found at
`http://www.cg.tuwien.ac.at/research/vr/seams/`

10. References

(Airey et al., 1990) J. M. Airey, J. H. Rohlf, F. Brooks Jr.: Towards Image Realism with Interactive Update Rates in Complex Virtual Building Enviroments. Computer Graphics, Vol. 24, No. 2, pp. 41 (1990)

(Barrus, 1996) J. Barrus, R. Waters, D. Anderson: Locales and Beacons: Precise and Efficient Support for Large Multi-User Virtual Environments. Proceedings of VRAIS'96, Santa Clara CA, pp. 204-213 (1996)

(Bier et al, 1993) E. Bier, M. Stone, K. Pier, W. Buxton, and T. DeRose. Toolglass and Magic Lenses: The See-through Interface. Proceedings of SIGGRAPH'93, pp. 73-80 (1993)

(Blau et al., 1992) B. Blau, C. Hughes, J. Moshell, C. Lisle: Networked virtual environments. SIGGRAPH Symposium on Interactive 3D Graphics, 25(2), pp. 157-160 (1992)

(Bowman et al., 1997) D. Bowman, D. Koller, L. Hodges: Travel in Immersive Virtual Environments: An Evaluation of Viewpoint Motion Control Techniques. Proceedings of VRAIS'97, pp. 45-52 (1997)

(Carlsson & Hagsand, 1993) C. Carlsson, O. Hagsand: DIVE- A platform for multi-user virtual environments. Computers & Graphics, Vol. 17, No. 6, pp. 663-669 (1993)

(Carrol, 1872) L. Carrol: Through the Looking Glass. Novel, (1872)

(Das et al., 1997) T. K. Das, G. Singh, A. Mitchell, P. S. Kumar, and K. McGhee. NetEffect: A Network Architecture for Large-scale Multi-user Virtual World. Proceedings of ACM Symposium on Virtual Reality Software and Technology (VRST'97), pp. 157-163 (1997)

(Fuhrmann & Gröller, 1998) Real-Time Techniques for 3D Flow Visualization. To appear in: Proceedings of IEEE Visualization'98 (1998)

(Funkhouser et al., 1992) T. Funkhouser, C. Sequin, S. Teller: Management of Large Amounts of Data in Interactive Building Walkthroughs. SIGGRAPH Symposium on Interactive 3D Graphics, pp. 11-20 (1992)

(Funkhouser, 1996) T. Funkhouser: Network Topologies for Scaleable Multi-User Virtual Environments. Proceedings of VRAIS'96, Santa Clara CA, pp. 222-229 (1996)

(Gibson, 1984) W. Gibson: Neuromancer. New York: Berkley Communications Group (1984)

(Hartman & Wernecke, 1996) J. Hartman and J. Wernecke. The VRML 2.0 Handbook. Addison-Wesley (1996)

(Lea et al., 1997) R. Lea, Y. Honda, K. Matsuda, and S. Matsuda. Community Place: Architecture and Performance. Proceedings of ACM VRML'97, pages 41-50, 1997.

(Luebke & Georges, 1995) D. Luebke, C. Georges: Portals and Mirrors: Simple, Fast Evaluation of Potentially Visible Sets. Proceedings SIGGRAPH Symposium on Interactive 3D Graphics, pp. 105-106 (1995)

(Macedonia et al., 1995) M. Macedonia, M. Zyda, D. Pratt, D. Brutzman, P. Barham: Exploiting Reality with Multicast Groups: A Network Architecture for Large-scale Virtual Environments. Proceedings of VRAIS'95 (1995)

(McReynolds & Blythe, 1997) T. McReynolds, D. Blythe. Programming with OpenGL: Advanced Rendering. SIGGRAPH'97 course notes (1997)

(Neider et al., 1993) J. Neider, T. Davis, M. Woo: OpenGL – Programming Guide, The Official Guide to Learning OpenGL. Addison Wesley (1993)

(Origin, 1997) Origin: Ultima Online. Online computer game (1997)

(Pausch et al., 1996) R. Pausch, J. Snoddy, R. Taylor, S. Watson, E. Haseltine: Disney's Aladdin: First Steps Toward Storytelling in Virtual Reality. Proceedings of SIGGRAPH'96, pp. 193-204 (1996)

(Schmalstieg & Gervautz, 1996) D. Schmalstieg, M. Gervautz: Demand-Driven Geometry Transmission for Distributed Virtual Environments. Computer Graphics Forum (Proc. EUROGRAPHICS '96), Vol. 15, No. 3, pp. 421-433 (1996)

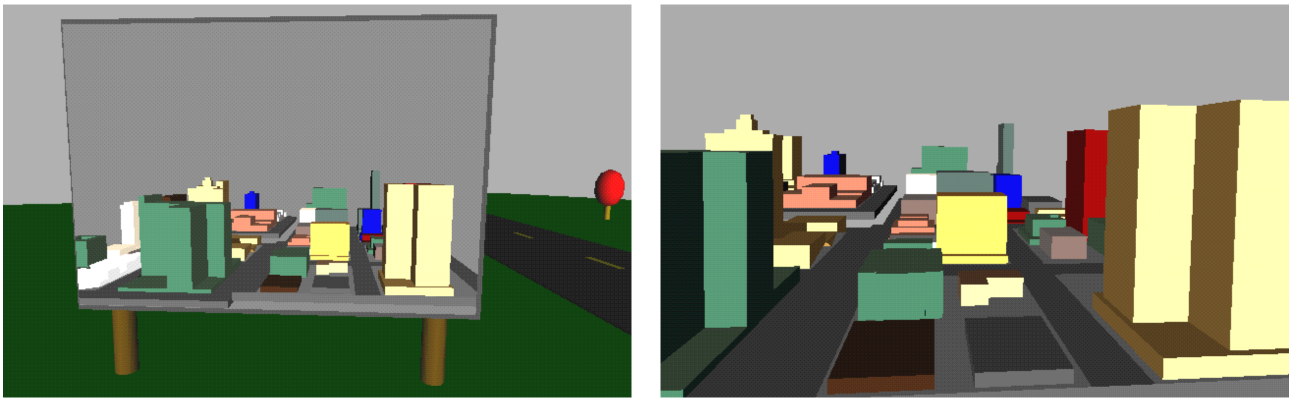
(Strauss & Carey, 1992) P. Strauss and R. Carey. An Object Oriented 3D Graphics Toolkit. Computer Graphics (Proceedings SIGGRAPH'92), pp. 341-347 (1992)

(Szalavári & Gervautz, 1997) Zs. Szalavári, M. Gervautz: The Personal Interaction Panel - A Two Handed Interface for Augmented Reality. Computer Graphics Forum (Proc. of EUROGRAPHICS'97, Budapest, Hungary) (1997)

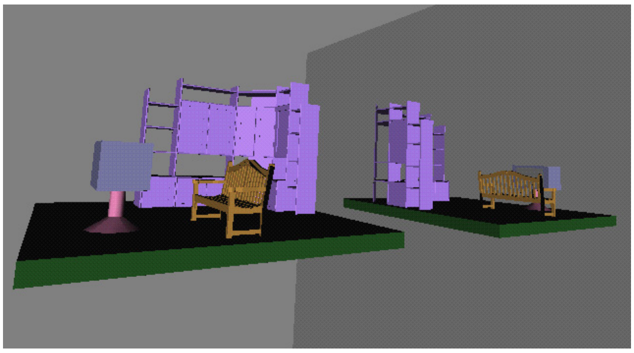
(Szalavári et al., 1996) Zs. Szalavári, D. Schmalstieg, A. Fuhrmann, M. Gervautz,: "Studierstube" - An Environment for Collaboration in Augmented Reality. To appear in: Virtual Reality - Systems, Development and Applications, Springer, 1998.

(Teller & Séquin, 1991) S. Teller, C. Séquin: Visibility Preprocessing For Interactive Walktroughs. Proceedings of SIGGRAPH'91, Vol. 25, No. 4, pp. 61-69 (1991)

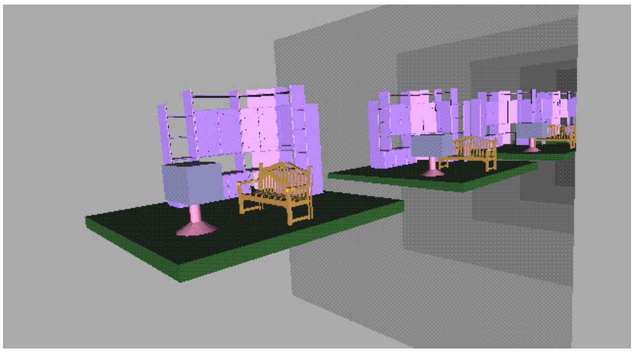
(Viega et al., 1996) J. Viega, M. Conway, G. Williams, R. Pausch: 3D Magic Lenses. Proceedings of the ACM Symposium on User Interface Software and Technology (UIST'96), Seattle Washington (1996)



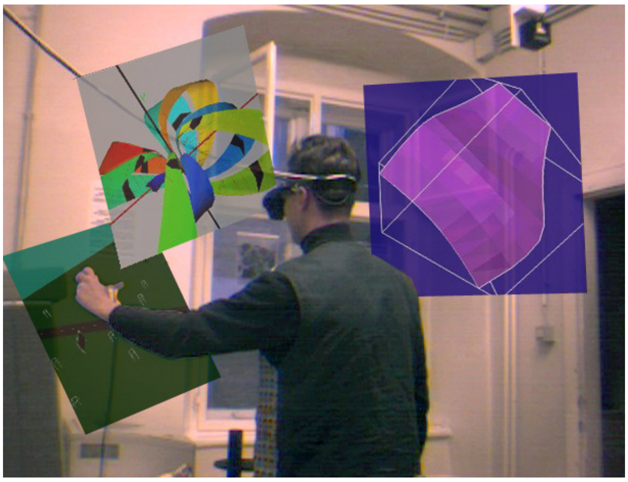
Color plate 1: Going through a billboard into another world



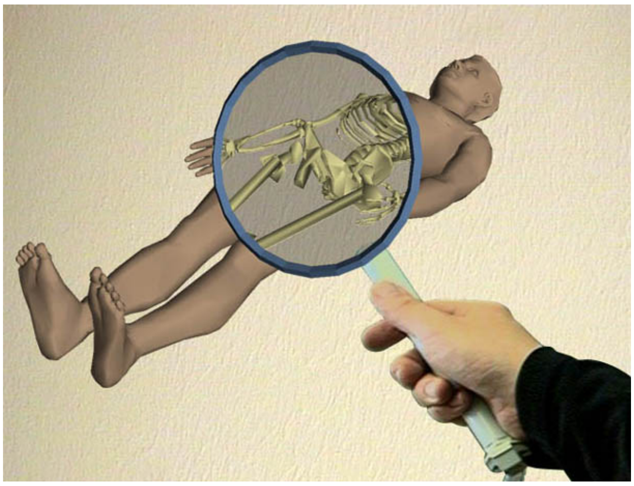
Color plate 2: A SEAM as a mirror



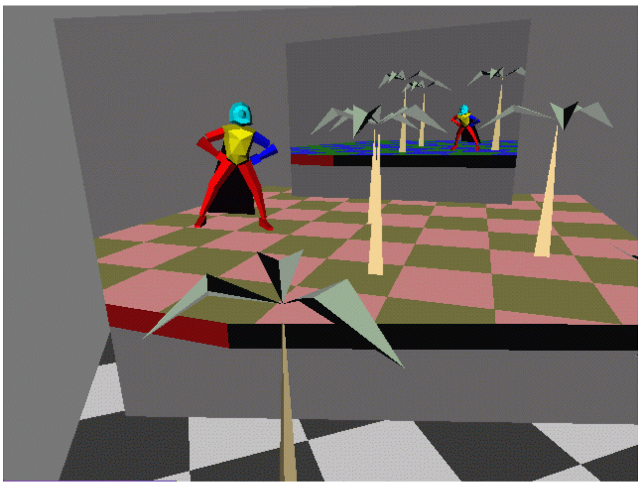
Color plate 3: Infinite worlds with recursive SEAMS



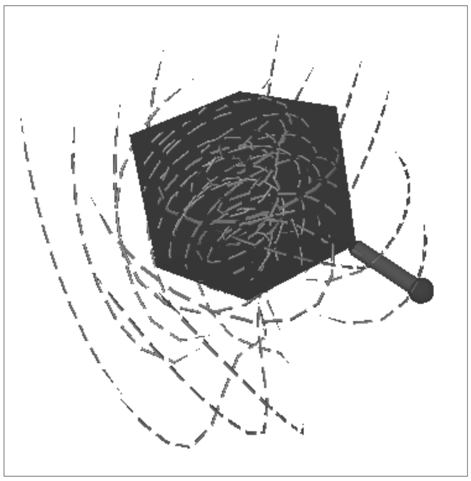
Color plate 4 :SEAMS to access 3D augmented reality



Color plate 5: A SEAM as a 3D magic lens for X-raying



Color plate 6 : Multi-user VRML browser with SEAMS



Color plate 7: 3D magic boxes for visualizazion focusing