Design Issues for a 3D Interface to TechnoSphere

G. Selley (1), A. Huxor (2), & R. Hawkes

(1). London College of Printing & Distributive Trades 10 Back Hill London EC1R5EN. Tel: *+44 171 514 6868* Fax: *+44 171 514 6868*

(2). Centre for Electronic Arts Middlesex University Cat Hill Barnet EN48HT. Tel: *+44 181 362 6712*

e-mail: *g.selley@lcpdt.linst.ac.uk a.huxor@mdx.ac.uk rjh@hplb.hpl.hp.com*

TechnoSphere (www.technosphere.org.uk) is an Internet-based artificial life simulator allowing people to create their own creatures, and receive communications from them as they grow, evolve, and die in a virtual 3D environment. Some of the users have indicated an interest in having a real-time 3D interface to this world, in addition to the current asynchronous email messaging system and existing 2D interface tools, and this is being currently investigated. This paper addresses two classes of problem that have been identified in the consideration of a 3D interface: technical feasibility and user based issues.

Keywords: TechnoSphere, artificial life, VRML, virtual environments.

1. Introduction

TechnoSphere (www.technosphere.org.uk) is an Internet-based artificial life simulator, allowing people to create their own creatures, and receive communications from them as they grow, evolve, and die in a virtual 3D environment. Users create artificial lifeforms and follow their interactions in the virtual environment, via email and a suite of world wide web based interface tools. In response to comments from some of the users of TechnoSphere, the authors have been assessing the potential of integrating 3D interface components into the system: an interface that would allow the user a more 'immersive' experience of the environment. If undertaken, such an interface would move TechnoSphere into the mainstream of virtual environment simulation which would have implications for the users of TechnoSphere. This paper looks at contradictory responses to a 3D interface from both technical and user based perspectives, and explores the issues that arise in these areas.

2. An overview of TechnoSphere

TechnoSphere consists of a WWW server, a 3D artificial life environment, a rendering engine, and an email server. It uses a WWW toolset to support user interaction, allowing users to create lifeforms, register them with the system, and retrieve information about their creatures. The system currently supports upwards of 300,000 life-forms, of which 50,000 are still 'alive,' and has many thousands of users. Paradoxically, despite the limited interaction, the users' emails to the developers indicate that they experience a rich form of engagement with their creatures, which extends to a wider community of users. It is this current successful quality of TechnoSphere that we want to maintain through any changes to the system.

The main TechnoSphere program includes a simulated environment consisting of a 16Kms square mountainous terrain with substantial plains, creatures, and a single form of vegetation. The artificial lifeforms - which can be either herbivore or carnivore - grow, seek food and mates, chase and evade each other, and produce offspring that are variants on their parents. The artificial life environment is a program that stores the artificial lifeforms and calculates their interactions with each other and the environment, and supports their reproduction and evolution.

Herbivores spend their lives wandering the plains, searching for food and companionship, whilst avoiding the insatiable appetites of carnivores. Companionship is sought in the form of other herbivores who, for the most part, share a similar appearance. A carnivore's existence is similar except that they prey on the herbivores for food rather than vegetation.

Users create an artificial lifeform by selecting components from a menu, they give the creature a name, and attach their email address so that the creature can send email to them. There are four main types of components: mouths, bodies, eyes, and wheels. Each component type has a set of values which influence how that component functions within TechnoSphere. For example, a mouth is described by a set of values that define, the size of a bite, the rate at which the mouth can bite, and the amount of food consumed with each bite, amongst others. Similarly, the body parameters describe the amount of food (energy) the body can store, and the rate at which it can be digested and made available to the other body parts. Eyes have a viewing range and angle of view, and wheels have a speed, radius and rate at which they consume energy. The creature functions by using an energy model that dissipates available energy to the body parts. When a user selects a particular component they are, in effect, choosing pre-set values for the associated parameters, although the current interface does not allow users to see the value forcing them to make a blind selection.

When a new creature is created using the interface, the details are posted to the TechnoSphere server where the creature is constructed and entered into the simulation, where it starts to interact with the other creatures. The artificial life environment also allows creatures to mate, the offspring taking a mixture of physical and functional attributes from both of its parents. The creatures that are the result of this mating begin their existence in TechnoSphere as a foetus, which gestates over a number of days, until the creature carrying them gives birth. Creatures continue life as children, progress through adulthood, old age and eventually die, becoming another food supply for the carnivores.

At key moments in the creatures existence, such as deaths, births, and mating, the creatures send email to their makers with details of the event and the other creatures involved with the event.

All children born of a creature will inherit that email address, as will their children, and their children's children, and so on. This means that users are kept in contact with their immediate family without the need for further action allowing the user to track an entire lineage. The following are typical email messages sent by creatures to their creators.

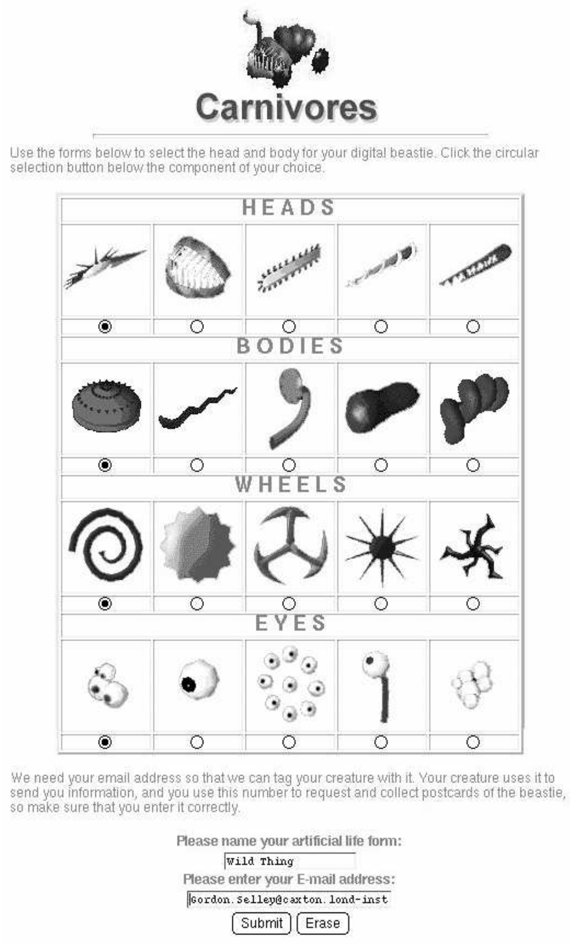

Fig 1. Creature Designer tool

Egg_Head (ID 211536) has sadly died of starvation.
Wiggy (ID 74788) died trying to defend against the attack by jellied_eelI (ID 43766).
Hasta_la_vista_baby (ID 37552) mated with P_130I (ID 31756) and became pregnant.
Hasta_la_vista_baby (ID 37552) has given birth to one of your creatures offspring, baby P_130II (ID 79098).

Users of the system can receive email from the their creatures at any time of the day along with email from their colleagues and friends. In addition, users have access to a range of tools that can make inquiries into the simulation and present the results. They can get information on the current state of their creature, the location of the creature in a graphical map of the terrain, a creatures family tree (parents and offspring), and statistics of the whole simulation (numbers of creatures alive and the most effective creatures from a range of categories).

Users often email requests for new functions and improvements to the system. It is interesting that most of these requests concern features which allow them to personalise their creatures or to enhance the social aspects of their experience. For example, we have developed functions, at the request of the users, which allow the user to rename their creatures and to add obituary remarks for them when they die. In addition, we have integrated the online rendering system which allows users to view a rendered image of their creature in the terrain surrounded by the other creatures. Used in conjunction with the other tools, such as the creatures attribute lists (for health, current action, etc.), and location maps, the user is able to build up a picture of what their creature is doing and to track its development over time.

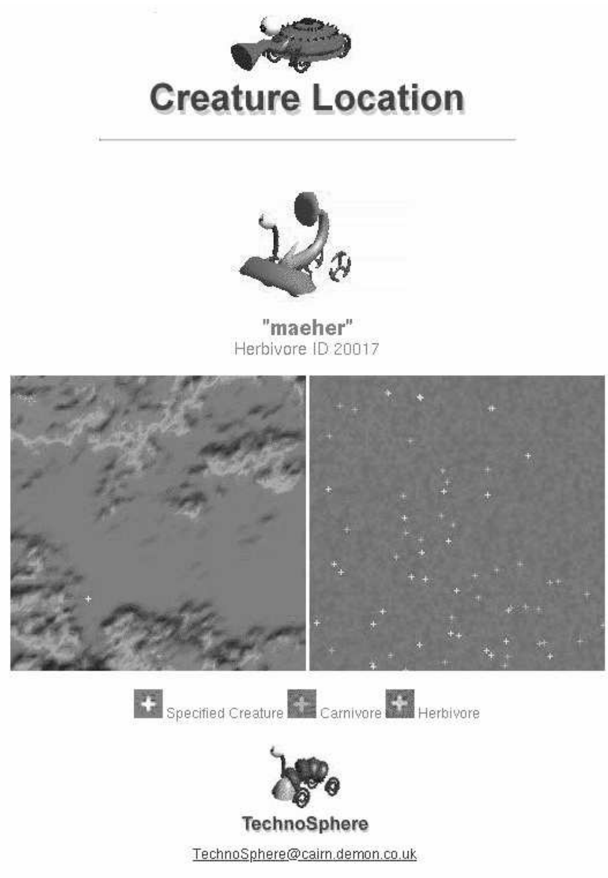

Fig 2. Output from the creature location tool

One of the interesting characteristics of TechnoSphere that was not expected in the original design, has been its social nature. For example, the users of TechnoSphere include groups of people who work together, and who each have made creatures. They exchange news of their creatures with each other, often keeping information (ID's, children, pictures, deaths etc.) on notice boards. One function that was added after user requests was the ability to retrieve the email of the user whose creature had attacked, eaten or mated with ones own. The users then appear to have exchanges with these other users who they may never have met.

Essentially TechnoSphere is providing very little in terms of traditional interactivity: you cannot influence your creatures. You receive a few emails, and can ask basic questions about the whereabouts and condition of your creature, but, the community of users has a rich experience. This is in contrast to our initial thoughts in which we assumed that a more immersive approach could support rich forms of interaction.

We have received a few requests for a 3D interface which would allow the user to roam around the terrain in real time and interact, or guide their creature to some end, usually in finding food or to kill a particular creature. We are now in a situation where we have to consider if a 3D interface is technically viable, and also if desirable from the users' perspective.

3. Design Issues for a 3D Interface to TechnoSphere

3.1 Technical issues

TechnoSphere could be experienced in 3D in different ways. The simplest form is merely a virtual window on the environment through which the user can passively monitor the progress of the simulation. This could be from a fixed viewpoint or, alternatively, the user may be given control of the viewpoint and permitted to move around the environment unhindered. The ability to interact with TechnoSphere and influence it in some way would be the most technologically demanding and, at this moment in time, requires extending the current simulation engine. These alternatives will be discussed in this section and their prospects considered, but before we continue, here are some relevant facts:

3.1.1 Simulation facts

The population of the environment varies but the average number of creatures alive at one time is around 50,000. Creatures currently only inhabit the plains and do not proceed into the mountainous areas. The terrain is 16Km by 16Km in size and is segmented by the simulation engine into 64 x 64 cells which measure 256m by 256m. Only approximately one third of the terrain is flat, and therefore the population is distributed over around 1351 of these areas. Given a uniform distribution we can expect around 37 creatures per area. However, the distribution is far from uniform due to feeding patterns. Herbivores eat grass which is only present on about 25% of the cells, and a large percentage of carnivores can be found on these feeding grounds surrounding the herbivores. Thus we find, on average, upwards of 140 creatures in these feeding cells.

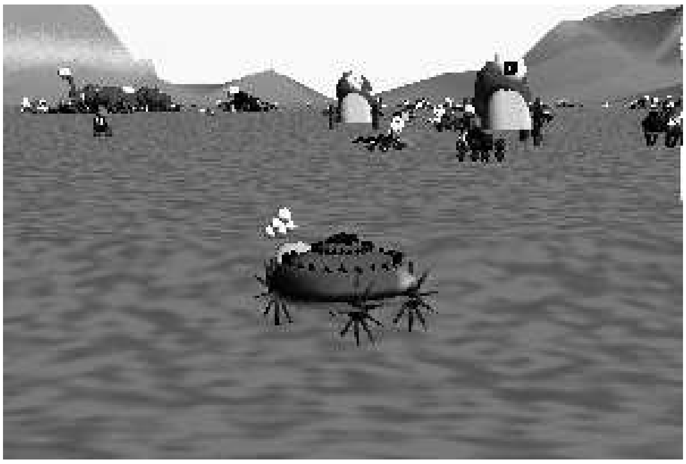

Fig 3. Example of image output from the snapshot tool

Thousands of the creatures present in the environment at any one time have been introduced into TechnoSphere by the users. There are a reasonable number of computer-generated creatures (to maintain population stability) but the majority of creatures are the offspring of these user-created creatures. All of these creatures have different behaviours influenced by their attributes, their environment, and some random element. Although there are often set patterns of behaviour, it is very difficult to predict what they will do from one moment to the next.

3.1.2 3D model facts

The geometrical complexity of each area of the terrain varies quite dramatically, but all 4096 cells can be described by a 400Kbyte VRML file. There are only 25 different geometrical body parts that are used to construct the 1250 different types of creature. Again geometrical complexity varies, but a typical body part requires about 18Kbytes of VRML, approximately 460Kbytes for all possible body parts. Generating different level of detail (LOD) models for the terrain and body parts would substantially increase the amount of geometrical data.

3.1.3 Bandwidth

The TechnoSphere server is one hop on a dedicated 10Mbps Ethernet network to a 34Mbps SuperJANET connection. However, this is about as far as we could go when making any predictions on network bandwidth and latency concerning TechnoSphere. All users connect to this server from the Internet from access points all over the world. The number of networks composing the Internet present such a varied array of bandwidth and latency that any predictions of these quantities between two endpoints is problematic; round-trip times in the order of 1000-2000ms are not uncommon.

For example, many users of TechnoSphere connect to the Internet from home using dial-up accounts. Assuming a 28.8Kbps modem this gives us a very optimistic data transfer rate of 2.5Kbytes per second (unlikely to be met using TCP/IP). At this speed it would take around 6 minutes to download all the basic geometrical data for TechnoSphere. Realistically, it will almost certainly take a lot longer due to the traffic on the Internet between the server and the users Internet Service Provider.

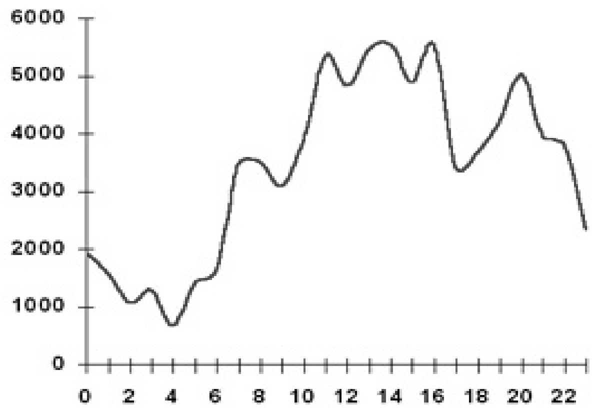

Fig 4. Typical daily access to TechnoSphere web server (hits per hour, 24 hour clock).

3.1.4 Offline viewing of the environment

If all users crave is eye candy then they might be easily satisfied by a three-dimensional model of the terrain, a varied collection of creatures and an animation bustling with activity. If the user was only interested in a fixed portion of the terrain then only this subset of areas would have to be downloaded. The effectiveness of this approach largely depends on the camera angle and the amount of detail the user requires. The nature of the TechnoSphere terrain (and weather) is that from any point of the plains, the hills are always in view. There are many ways in which this problem may be relieved, e.g. level of detail, depth cueing, 2D backdrops, etc., but we shall not go into them here.

On the whole, this offline approach is the simplest to achieve. Apart from the initial download of models, behavioural scripts, etc., there is no further communication with the server running the simulation. If the viewpoint was controlled by the user then the speed of movement through the environment is limited only by the local machine.

The creature behaviour could be made comparable with the actual simulation, however, it could not replicate the exact behaviours of the 'real' TechnoSphere creatures without completely replicating the entire simulation locally. This is not practical unless the user has a very high specification machine and Internet connection. From a user perspective, a approximation of the TechnoSphere simulation, may be less engaging.

3.1.5 Online viewing of the environment

Of much more interest to some users is the ability to monitor and possibly interact with the creatures and the environment as the simulation progresses. This requires that creatures appear in the right place at the right time, and requires information to that effect to be sent by the server to each client monitoring TechnoSphere. There is no need to receive information about every creature, just those within the current viewpoint. The simplest information to send would be a position and orientation update for each creature. Given that both of these would require a total of 6 x 32 bit floating-point numbers and, assuming a home user with a 28.8Kbps modems, this would enable around 100 updates a second. If we were monitoring one of the feeding areas with around 140 creatures then we could not even manage one scene update per second.

The best way of improving this situation is to send higher-level information. In this case we might consider the transmission of position and orientation updates only if there has been a significant deviation from the last transmission. Such a model, dead- reckoning, is very effective in reducing the number of transmissions required to approximate an objects, behaviour. An approximation algorithm is applied to each creature of interest at the server and at the client to predict the creatures movement. When the server detects that the actual position and orientation has deviated from the predicted values by a given threshold, the current values are sent to the client. The client, in turn, takes the received values and updates its prediction algorithm whose output is used to move the creatures around the environment. Other discrete events that require graphical representation would have to be sent separately, e.g. fighting, giving birth, etc.

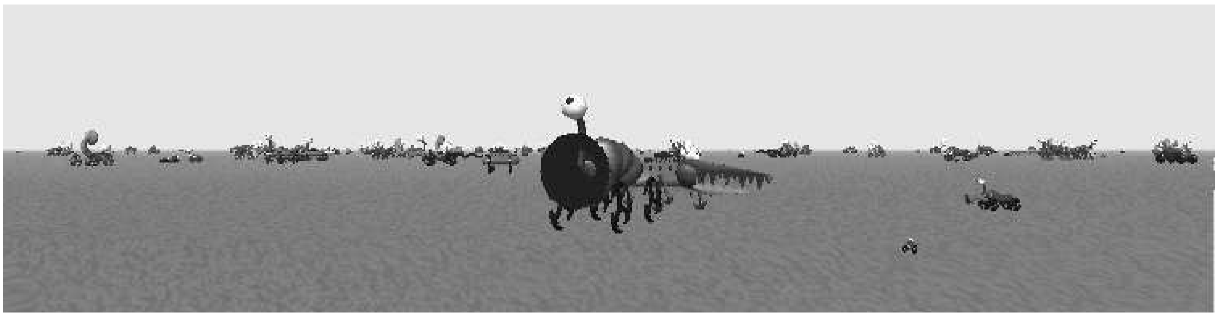

Fig 5. Scene from inside TechnoSphere

One major side effect of such an approach is visual discontinuity. Jumps in a creature's position and orientation can be expected as the dead-reckoning algorithm adjusts its predicted values to the actual values periodically sent to it by the server. (It is possible to refine this approach further by sending higher and higher level information, but this inevitably means that the clients must perform more and more work.) Assuming a retransmission of position and orientation once every 5 seconds and 140 creatures requiring updating, this will result in around 3.3Kbytes over 5 seconds, or approximately 670 bytes per second. This is well within client limitations, but what about the server? As stated earlier, the TechnoSphere is connected via a 10Mbps link, however it is typical to achieve maximum data throughput of around only 70% of this value using TCP/IP. Even then, utilisation of the available bandwidth is affected by the size of the messages sent across the network. Assuming the best case this means that a little under 1Mbyte of data can be transmitted per second.

Since multicast is not widely available, the server would have to maintain dedicated links with each client and if each client requires 670 bytes of data per second, this means that the server could handle around 1370 clients. This is an unrealistic expectation due to many factors, not least that the same machine has to actually run the simulation, a web server, serve many web pages, send email, and so on. The authors believe that the actual number of clients that may be supported in this manner would be at least 10 times less.

These figures belie one other important factor in a networked simulation: latency. If the user is just to passively monitor the progress of TechnoSphere then latency is of little consequence. The user will see a good approximation of the environments state, albeit a tardy one. If, however, users wish to exact changes on the environment and/or creatures then there must be much tighter consistency between what each user sees and the actual state of the simulation. Moderating changes to the simulation, especially when the focus of that change is the same for more than one user, opens up another can of worms. The techniques, issues and examples presented here have been grossly simplified for the space available and a more complete treatise of the problem of distributing, interactive simulations can be found in Hawkes (1996).

3.2 User based issues

As mentioned above, one of the motivations for consideration of a 3D interface to TechnoSphere came from email requests from users to the systems developers, in which a number of users asked for a real-time, 3D interface, to enable them to enter the world at anytime - to move through it at will, and to observe the activities of their own and others creatures. In addition to the technical problems, there is also a question as to whether a fully 3D-interface, if it could be built, might have a negative effect on many of the users' experience. These concerns relate to the users’ perception of their relationship to the TechnoSphere world, and its role as a landscape. Yuill (1997) argues that there are two alternative conceptions of landscape: the landscape as 'image', and the landscape as 'place'. In the former, the user is not in the space, but his or her sense of the 'here' is the landscape which is being looked at. It is that which is within the range of vision but unavailable for real interaction, such as one might find in a painting or on the cinema screen. Alternatively, in a landscape of 'place', it is human intervention which comes to the fore.

For Yuill, TechnoSphere stands firmly as a landscape of image. This situation creates an alternative vision of what constitutes 'immersion,' which TechnoSphere seems to exploit. "Due to the current bandwidth limitations, a live view of TechnoSphere is not possible, ...this dislocated vision of the project suggests the interpretative possibilities of TechnoSphere. A fully immersive version of TechnoSphere in which we could walk and feed the creatures would, from a critical perspective, be too easy, too seductive. Like the relatives of an immigrant family left back in the 'old country', we wait upon the latest news, letters, and postcards from our émigré children. Beyond the algorithmic progress of plant growth and creature procreation lies a community of parents held together by the thread of communication. The frustrations of distance maintain our awareness of the landscape of cyberspace (Yuill 1997: 75)."

The short messages sent by TechnoSphere to its users regarding their creatures and the state of the world, allows them to fill in, in their minds, what they believe is going on. Like a novel, the users create a complex world that over-specification might shatter. And these may differ from user to user. As Yuill further observes, through its veiled access, TechnoSphere is less a 'data-oriented' world to its users, but rather a 'constructivist' one "which sees the environment as the product of multiple perceptions" (Yuill, 1997: 73), each person model being dependent upon 'our state of mind, our interests, what we have been taught to experience, our personal expectations, and our familiarity with the medium' (Coyne 1994: 66). We often have a sense of what is happening to those with whom we communicate on the phone and by letter, a sense that is aided by having incomplete information, the 'scenery' and 'actors' in their lives from which we create a story.


Fig 6. TechnoSphere landscape

Furthermore, in addition to the user immersing in the TechnoSphere world through their own imagination, TechnoSphere also has the feature of immersing itself in the users' daily world. By appearing as messages amongst the email-box, it becomes part of the users' everyday life, not the escape typical of computer games. Within one session, users will receive emails from their colleagues, friends and their TechnoSphere creatures in the same way, and at the same time, and in a format that is used for 'real-world' communication. It thus inherits the qualities of the 'real' that come from these everyday computer-based tools.

In this sense, TechnoSphere has much in common with television soap opera. With a soap opera, the narrative can extend over days, weeks and years, and in a way that is part of the viewers everyday experience. They can cook, eat, chat and work while the programme in on. And they can discuss issues with other people who view the programme, who are experiencing the same timeline in the programme as themselves. Indeed, it could be argued that TechnoSphere is one of a new bread of Internet 'soaps' that draw upon the computational nature of the medium, but within an understandable genre. It is surely plausible that like soap operas, the success of TechnoSphere's narrative centres around its focus on themes of family lineage, parenthood, and competition between clans.

4. 3D interfaces to TechnoSphere

A 3D interface to TechnoSphere that would allow the user to view their creatures in real time, is problematic on three accounts. Technically it is not possible to support large scale interaction because of computational and network restrictions, with current Internet technology. From a user point of view it is not clear how such an interface would enhance the users experience, in fact some arguments suggest that such an interface might be detrimental to the user experience. And finally, TechnoSphere's underlying preferred model of interface development, which is essentially user driven, can be seen to be primarily motivated by the para-social concerns discussed above. Few asked for a real-time, fully rendered, interface. This latter view of interfaces to 3D worlds can be considered an extension of the 'data oriented' approach, an animated map, a perspective that is counter to TechnoSphere’s constructivist approach which supports a personal interaction and interpretation.

4.1 How can 3D enhance the TechnoSphere user experience ?

Given the problems identified above, the question remains, what can 3D offer TechnoSphere users? Users have requested the ability to gain information about the TechnoSphere world, therefor, it might be more appropriate to use 3D to visualise abstract characteristics of the world and activities within it. For example, the Sensorium (1997) project is a web site which offers the user an animated rendering of the earth’s seismic activity. Data is gathered from remote sensors, which have been placed around the earth surface, and made available on the WWW. The Sensorium project accesses this data and converts it into an animated 3D model from which a downloadable movie is rendered. The visualisation shows seismic activity over a two week period. A similar approach could be taken with TechnoSphere in viewing activity within the world over time. For example, visualisations could be produced of population trends and movements, growth of vegetation, areas of a high incidence of death, etc. Used in this way 3D could be used to support the current user interaction without interfering with the para-social constructivist approach inherent to TechnoSphere.

5. References

Hawkes (1996). References Hawkes R. (1996) A Software Architecture for Modeling and Distributing Virtual Environments. Ph.D. Thesis, University of Edinburgh. http://www.dcs.ed.ac.uk/~rjh/

Selley, G. (1996) On-Line Rendering for the World Wide Web. Proc. Eurographics UK Conference. Vol. 2 London, March 1996, pp.1-9.

Sensorium (1997) http://www.sensorium.org

Yuill, S. (1997) Jane Prophet: the Double Landscape. Transcript 2(3): 73-78.