Open Research Online

A new navigation paradigm for virtual reality: the guided visit through a virtual world

Conference or Workshop Item

oro.open.ac.uk

A new navigation paradigm for virtual reality: the guided visit through a virtual world

J. Azpiazu*+, C. Pedrinaci*+, J. Aguado*+, A. García-Alonso+, A. Bernaras*

*San Sebastian Technology Park, Spain, +Euskal Herriko Unibertsitatea, San Sebastian, Spain
jon@miramon.net, abernaras@idom.es, agalonso@si.ehu.es

Abstract

The three main navigation paradigms for virtual worlds, i.e., free navigation, automatic tours, and multiuser navigation show important limitations when dealing with guided visits that involve interactive cooperation among several users in 3D virtual worlds over the Internet. In this paper, we present our research into this issue and some important results. We propose a new navigation paradigm denominated guided visit through a virtual world, where the capacity of a user guiding several remote users through the virtual world is enriched with the capacity to dynamically interchange the role of guiding between the connected users. The user that acts as a guide moves freely through the virtual world, and his/her movements are reproduced by the browsers of the other guided users. We also present the architecture and the system we developed that implements this paradigm, as well as its integration in a working real-world application that demonstrates its use.

1. Introduction

Virtual 3D worlds are being used nowadays in multitude of different areas, such as e-commerce, culture, games, education... Nevertheless, once surpassed the barriers as far as development of the hardware refers, so that the applications that make use of virtual reality extend more, the most important barrier users find is the usability. When entering virtual worlds for the first time, users must familiarize with the new methods of user-machine interaction, the metaphors used in virtual reality, the visualization and navigation, etc. This new aspect is being object of numerous investigations, that mainly approach to it from two different although non-excluding points of view: from the front of software, the development of new methods and/or mechanisms to facilitate the understanding and simplicity of use by the user [5][6],

and construction of new hardware elements that make more intuitive the use of virtual reality applications [9].

This document begins exposing the limitations showed by the current navigation paradigms in virtual worlds. In order to go beyond these limitations, we propose a new navigation paradigm denominated guided visit through a virtual world, that combines characteristics of the previous paradigms, and incorporates other novel ones. We also present the developed system that implements this new paradigm. The system allows making a guided visit through a virtual world, among some other features that will be exposed later in this paper.

2. Background

The classic navigation paradigms in virtual worlds are three: free navigation, guided tour or automatic tour, and multiuser navigation. If the main interest of the creator of the world is to show it to the visitors, and not just to be used as a background scene, both paradigms of navigation (free navigation versus automatic tour) show several limitations. The development of the multiuser systems is still partial, and most of the existing multiuser browsers, center their application in chat systems, in which users ramble through the virtual worlds, while being able to maintain conversations with those near users or with those others that share the same common space, like a lounge or a room. Another scope of application of multiuser systems is the cooperative interaction [7][10], an area under development.

Using the paradigm of free navigation, the visitors can freely move through the world, visiting those areas that are more of their interest. However, this total freedom can cause that the user gets lost in the virtual world, or does not observe certain details that the creator considers important. This is just like it happens in the real life, when a traveller arrives at a city new and strange for him.

In contrast, automatic tours allow the designer of the world to guide the visitor, and to make greater

emphasis in the areas that he decides. Nevertheless, this paradigm of navigation shows two important limitations. The first one is the lack of interactivity of the user, since throughout all the passage, the user is limited to observe what is showed to him, as if it was a prerecorded video. On the other hand, the path will have been defined at the moment of designing the world, reason why it will be the same one for all the users who visit the world, that is to say, is not customizable or adaptable to the necessities of the visitor.

The developed works to improve the use of these two paradigms of navigation, are based on tools that facilitate the creation of automatic tours [4][11], or more innovating methods like the "navigation by query" [2] in which a list of predetermined viewpoints is offered to the users, giving them the possibility of performing a free navigation, or being taken to one of these viewpoints.

It is worth noting that although in the bibliography the term guided tour is used some times to talk about automatic tours, this document does not deal with this navigation paradigm, but uses the same expression to denominate a new one.

3. New navigation paradigm: the guided visit through a virtual world

The new navigation paradigm that we propose tries on the one hand to correct the lack of usability of the free navigation, at the same time that increases the level of interactivity of automatic tours. To achieve this objective, we have included some of the characteristics of the multiuser navigation, combined with characteristics of the free navigation and automatic tours.

The central idea of the new paradigm is based on the operation of the museums in the real life: the fastest and most effective method to visit a museum is to make that an expert (a person who knows the environment) takes the user through the different points of interest, at the same time that explains to him the different aspects that are being showed to him at every moment. This concept applied to navigation in virtual worlds, implies that a user who enters the virtual world is not navigating alone, but that is connected in some way (direct connection, or through an intermediate server) to an expert user, that knows the virtual world, and that is going to guide him through this one. There are several different methods to guide a user through a virtual world (virtual agents, indications within the virtual world itself ...). Nevertheless these methods require the participation of the user, and therefore certain knowledge by him so that they are successful. The solution proposed is the simplest one from the point of view of ease of use: with no need of any type of intervention on the side of the user who is being guided, his avatar will move through the virtual world, reproducing the movements that the expert user (from now on, the guide), makes remotely in his navigator. Both users, will maintain some type of communication (telephone, voice chat, text chat), so that through this communication, navigation complies to the exigencies and necessities of the guided user.

Therefore, in order to perform a guided virtual visit, there must be at least two implied users. One of them will take the guide role, whereas the other user happens to be denominated guided. The guide will have free navigation through the virtual world, whereas the avatar of the guided user is permanently bound to the one of the guide, reproducing the movements of this one, so that at any moment, the guided user will have the same perspective of the world (position and orientation) as the guide. Nevertheless, and following with the initial idea of the guided visit in a real museum, and considering the efficiency, a guide must be able to take control over more than just one user simultaneously. Extrapolating this detail to the paradigm we propose, an interconnected user group is established, where each one of them can be in different geographical places, but that at any moment shares the same position, and therefore the same view of the virtual world by which they navigate. Although several users are simultaneously sharing the role of being guided, at any moment there must be just one guide, who will be the only one that has freedom of movements through the virtual world.

The new paradigm proposed up to now is perfectly applicable in those cases where it is intended to show the virtual world, or to help the users to navigate through it. However, from the point of view of the users that are being guided, it shows a lack of interactivity very similar to the case of the automatic tours. The only way a guided user can influence over the navigation is by using the communication and asking the guide to change the path he is following. In order to avoid this, another new feature is added to the paradigm: users must be able to change the role they are playing during the visit, so that a user that is being guided can get to guide the rest of the users. Of course, if a user that is being guided becomes the new guide, the user that was guiding the visit up to this moment must change his role and become a guided user.

We have developed a system that allows a group of users to perform a guided visit through a virtual world. The system is intended to be ran over the Internet, so the language chosen to describe the virtual world is VRML, the most used and ISO standard to display 3D in the web. The control of the navigation can be transferred between the different users that are taking

part of the visit, by using the concepts of collaborative session and client/server architecture.
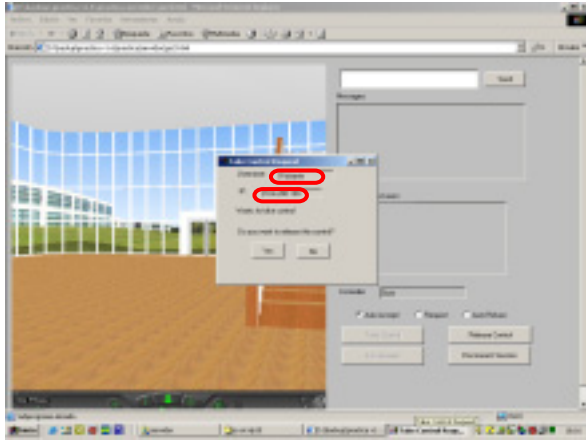

Figure 1. Guided visit system

In order to analyze the developed system, we can distinguish two different aspects: the communication architecture that supports the system, and the architecture of the system itself.

4. Communication architecture

The approach used to design the communication architecture is based on the client/server architecture. The clients, make connections to a server, which serves data to the clients. These, process the data according to the running application. Applying this concept to the guided visit paradigm, the server is the node who is playing the guide role, and transmits the position and orientation of its avatar to the connected clients. The clients render the scene according to the coordinates received from the guide, and draw it, so that if there is not an excessive delay, and the refresh rate is correct, the clients will have the feeling of being guided through the virtual world. The main difference between the classic client/server architecture and the one we used is that the navigation paradigm requires the ability to change the node that is playing the guide role, that is, to dynamically change which node is being the server. Due to this capability of changing the role, this architecture is denominated dynamic client/server architecture.

In order to implement this new architecture, we made use of the Collaborative API [3][8] developed by Pedrinaci. The API's origin was a 3D collaborative design application, but it is modularly designed so that can be used in multitude of different contexts, allowing the programmer to define the message types that will be used. The Collaborative API implements and
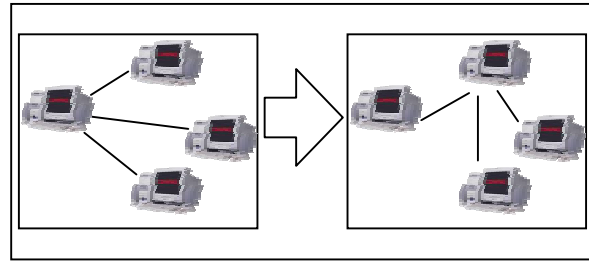

Figure 2. Dynamic client/server architecture

supports the architecture, abstracting the API's user (that is, the application developer that uses the API) from the lower-level tasks that communications require. When performing a change of control (changing the node that is acting as the server) the API will take care of closing all unnecessary connections and opening the new ones transparently for the users. It also provides two interesting characteristics: first of all, and contrary to many similar applications, it eliminates the need of a specialized server. One of the nodes involved in the session (concretely the one who has the guide role at a given moment) will be acting as the server, without needing to have any additional software or hardware. The second feature is that the Collaborative API provides multi-destiny communication without using any type of additional hardware, as could be a multicast router. It manages internally the multi-destiny communication via software, in a transparent way for users and designers who use the API.

5. System architecture

The development of the system was made having four main factors in mind: obtaining a modular system, the ease of use for all possible kind of users, reaching a system usable over the Internet, and making the most of the users resources.
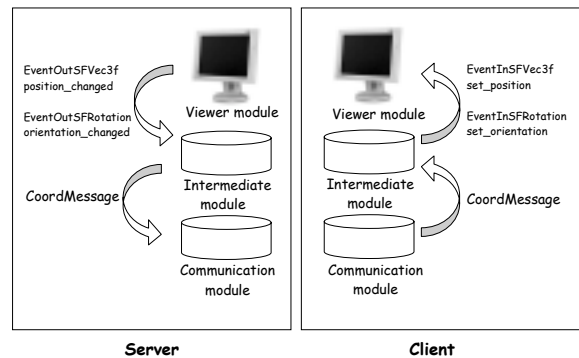

Figure 3. System architecture

The system has been divided in three modules: the communication module, the intermediate module and the viewer module. In addition, each module must be able to change its behavior depending on the role it takes in every moment, client or server. Next we explain briefly the functions and characteristics of the modules.

Communication module. The module that supports the dynamic client/server architecture is the Collaborative API, described in the previous section.

Intermediate module. The core of the guided visit system, keeps in every moment consistent information based on the actual state of the collaborative session, and manages the communications with the communication module and the viewer module. Java was chosen as the programming language due to a double motivation: keeping the system independent of the architecture and operating system of the computer where it is being used and the possibility of packing the application as an applet. When packing the application as an applet, the user does not need to download, install and finally run the application, but just by entering the URL that contains the applet embedded, it shows ready to be used to the user. In order to avoid peaks in the data transmission that can saturate the network or the client's machines, it implements a timer that keeps constant the data flow. By adjusting the timer's delay, the level of consumed resources can be dynamically changed, in order to get adapted to the users resources.

Viewer module. Following the ease of use philosophy, the system makes use of the VRML browser that the user has currently installed in his computer, and that already knows how to use. The communication between the VRML browser and the intermediate module is achieved by using the EAI (External Authoring Interface), so that any standard browser that implements these language independent specifications shall be used. The behavior of this module, controlled by the intermediate module, implies different tasks depending on the role the node is playing: when dealing with a guide node, it has to obtain the coordinates where its avatar is by means of the captured events (*position_changed, orientation_changed*); in a guided node, the intermediate module will provide the appropriate coordinates, that the viewer must refresh and display.

Some minor modifications must be made to the virtual world to make it usable for a virtual visit. More concretely, three new nodes must be added to the VRML file. These nodes allow the intermediate module to access the virtual world through the viewer module, extracting information from it and/or performing changes according to the data received from the communication module.

```
DEF Navegacion NavigationInfo {
  type ["WALK","NONE"]
  }
DEF Get_Position_Sensor ProximitySensor{
            center 0.0 0.0 0.0
            size 10E10 10E10 10E10
            enabled TRUE
  }
DEF Camara Viewpoint {
      fieldOfView 0.75
      orientation 0 -1 0 1.77594
      position -156.936 -0.960924 -29.9995
      jump TRUE
  }
```

Figure 4. Additional VRML code

The first added node is a *NavigationInfo* node. This node permits to block the navigation control console. By blocking the navigation controls when a user is playing the guided role, we avoid possible desynchronizations if any guided user performs some movement by himself, escaping from the control of the guide. When a role changing occurs (either a *take control* or a *release control* operation), the navigation control console of the VRML browser in the new guide's side is unblocked, allowing this user to perform a free navigation whose movements will be reproduced by the other participants browsers.

In order to detect the movements of the guide's avatar in the virtual world, the intermediate module needs a *ProximitySensor* node to be defined in the VRML file. This sensor's size is set to cover all the world, and is the responsible of generating the *position_changed* and *orientation_changed* events named before.

The last node defines the viewpoint of the virtual world for the guided users. Changing the values of this node, the intermediate module achieves to produce the effect of being guided.

6. Protocol

From the beginning of the development of the system, one of the factors more in mind was to make it usable over the Internet. For this reason, the protocol used for the communications between the different users should be one of the dominant protocols in the Internet: TCP or UDP. The one chosen was TCP, due to its retransmission mechanisms that assures the reliability when delivering messages over the network. This feature avoids possible inconsistencies in the state of the session if any session join or session quit messages does not arrive to any of the participants.

The mechanism implemented to allow new participants to join the session consists in a four steps message sequence. This way, users do not need to synchronize at the beginning of the session, but they can join the

session at any moment. When joining an existing session, the fourth message in the message contains the coordinates of the current position of the avatars in the virtual world, so that the new user can update his position in order to match the one of the other users.

Considering that the node that is acting as the server may change along the session, new users that want to join the session may not know which node they should ask permission to. The solution implemented in the developed system consists in redirecting the petitions from the actually participants to the node which is the server in that current moment. By means of this mechanism, new users that want to become participants of the session, just need to know one of the participants that already belongs to the session. If one of the participants receive a join session message, he automatically redirects the message to the server, who will process it.

When making an application available to be used in the Internet, a wide range of users may use it. It is reasonable to assume that different users may have different availability of resources. When dealing with Internet applications it is also good to take in consideration that there could be moments with heavy traffic load, that can make difficult to keep a good bandwidth. The way the guided visit system adapts to the available resources is by means of the timer, already mentioned above in this paper. The delay of the timer is adjustable by the server with three different fixed values (high, medium, low), that allow to increase or reduce the data flow generated.

7. Real case application

The system has been proved by applying it to a real case, in the Technology Park of San Sebastian (Spain).

In the San Sebastian Technology Park virtual 3D world on the Internet, called "Virtual Miramon" [1], visitors can move freely through corridors and stairs, visit an exhibition hall and several conference and meeting rooms using different configurations for different types of events. The 3D world is integrated in the web pages of the San Sebastian Technology Park, thus being possible, for example, to make an online booking of the premises visited in the virtual world. Among other possibilities, visitors can start an automatic tour that takes them in a pre-defined path through the main building. This is a comfortable and easy way of 3D navigation: just rest and watch.

However, the experience of the last few years demonstrates that visitors and/or conference organizers are attracted by a much higher degree of interactivity with personnel of the Park and other visitors sharing the virtual 3D world, when visiting the premises (both in-situ and virtually), and by having partially guided visits instead of free visits or automatic tours. This motivated a move to investigate the possibilities of introducing further interactivity capabilities in Virtual Miramon, and in particular guided visits.

For example, a guide and one or more visitors, physically located anywhere in the world, can start a guided visit to Virtual Miramon through Internet. The guide, usually a member of staff of the Technology Park of San Sebastian, moves around the 3D world showing and explaining the premises, while the visitors follow the guide seeing the same view. At any time during the visit, any of the visitors can ask the guide for the control of the navigation, in order to look at those areas that have attracted his/her attention, or make questions about concrete aspects of the world.

8. Conclusions

Once seen the limitations that, from the point of view of the usability show virtual worlds that offer free navigation, we go further over the automatic or guided tours to propose a new navigation paradigm: the guided visit through a virtual world. Using this paradigm, a user who knows the virtual world, guides another user or a group of users through the virtual world, showing them its most interesting aspects while he can give them explanations about the environment they are visiting. The method for guiding the users is by "taking them by the hand", that means, that the movements performed by the guide in his browser are transmitted and reproduced in the browsers of the guided users. In order to increase the level of interaction of the guided users (they would be limited to watch what is being shown to them) it is allowed to change the role of guide between the users involved in the visit.

As a demonstration for this new paradigm, we have developed a system that allows to perform a guided visit through a virtual world. Taking in consideration that the system is designed to be used through the Internet, we chose VRML as the language to describe the virtual world. There is no need of a specialized server to use the system, as the guide node directly transmits the coordinates to the guided ones using TCP/IP, and the rendering and visualization of the views take place at the users machines.

The system developed implements the guided visit paradigm, also implementing the change of control between the users involved, for which we used a dynamic client/server architecture. The system was probed in several demos during the 2003 year, and some of its characteristics are:
- implementation of the guided visit through a virtual world paradigm, including the change of control between users

- use of the concept of collaborative by means of the utilization of the dynamic client/server architecture
- control of the treatment to give to the new users that want to join the session (automatic accept or refuse, or ask to the user who has the guide role)
- log file containing all the network messages generated by the system
- peer-processing approach: the new views of the world are not transmitted through the network, but the rendering and visualization are fulfilled at the users machines
- a specialized server is not needed, once the Java applet and the VRML world have been downloaded a connection is established between the guided users and the guide, so that the communications happen directly from one user to another
- dynamic adjusting of the consumed resources according to the available resources of the users (network bandwidth and rendering capability of the machines)
- text chat that communicates all the users involved in the session
- capacity to join or quit the session at any moment during the guided visit, being automatically transported to the point it is taking place at this moment

9. References

[1] Azpiazu J., Pedrinaci C., Aguado J., Bernaras A., "Virtual Miramon", Home page: http://www.miramon.net/miramon_virtual.html, (Last visited: April 2004)

[2] Ballegooij A, Eliëns A., "Navigation by Query in Virtual Worlds", *Proceedings of the sixth international conference on 3D Web technology*, Paderbon, Germany, 2001, pp 77- 83

[3] Borro D., Recio I., Pedrinaci C., Sánchez H., Garcia-Alonso A., "Peer-to-peer techniques applied to the cooperative visualization of CAD models", *22nd International Exhibition of CAD/CAM Tools and Methodologies for Product Design, Manufacturing and Lifecycle Management*, Paris, France, 2003

[4] Chittaro L., Ranon R., Ieronutti L., "Guiding Visitors of Web3D Worlds through Automatically Generated Tours", *Proceeding of the eighth international conference on 3D web technology*, pp 27-38, Saint Malo, France, 2003

[5] Darken R.P., Sibert J.L., "A Toolset for Navigation in Virtual Environments", *Proceedings of ACM User Interface Software & Technology*, 1993 pp. 157-165

[6] Herndon K. P., Van Dam A., Gleicher M. "Workshop on the Challenges of 3D Interaction", *SIGCHI Bulletin,* October 1994, Volume 26, Number 4

[7] Jugel M.L., "Enhancing MUVEs: Connecting Virtual Objects with Environmental Simulation", *The 2001 International Conference on Virtual Worlds and Simulation*, Phoenix, United States, 2001

[8] Pedrinaci C., Aguado J., Azpiazu J., Bernaras A., García-Alonso A., Sánchez H., "A Generic Communications Module for Cooperative 3D Visualization and Modelling over the Internet: the Collaborative API", to be published in the *Proceedings of the IEEE-VRIC 2004, Laval-Virtual*, Laval, France, 2004.

[9] Robinson J., Dumoulin S., Stewart J., "MVIP-II: a protocol for enabling communication in collaborative virtual environments", *Proceeding of the eighth international conference on 3D web technology*, Saint Malo, France, 2003, pp 155 – 160

[10] Ruddle A.R., Savage J., Jones D. M. "Symmetric and Asymmetric Action Integration During Cooperative Object Manipulation in Virtual Environments", *ACM Transactions on Computer-Human Interaction*, vol. 9, pp 285-308, 2002

[11] Schaerf M., Tessicini A., "JubilEasy: Build a personalized 3D visit of Rome", *Proceedings VRML 99 of the Fourth Symposium on The Virtual Reality Modeling Language*, Paderborn, Germany,1999