Development and Application of Virtual Avatar Interactive Technology

Duan Xinyu, Jiang Pin

Media Research Institute, Anyang Normal University
Anyang, Henan Province, China
dxy@aynu.edu.cn

Abstract

Virtual avatar interactive technology is the foundation for cooperative multi-user to apperceive current colony's working status. When running VRML program, an invisible explorer avatar can be assumed in three-dimensional space. Based on VRML, described its avatar application, extension technology, browser function and explore navigation. Mainly studied the NavigationInfo node's grammatical pattern and its avatarSize field. These determine the collision detection and user perspective scope with the undulating terrain.

Key Words

3D Avatar, Java, Script, Interaction, Browser

1. Introduction

In shared virtual technology, users interact with surrounding environment by adopting avatar Interactive Technology. The foundation of this kind of interaction is the sensing ability of virtual avatar to its environment, including existence sensing, position sensing, avatar identification, action sensing and identity sensing. If be able to sense the environment variation, then virtual avatar can manipulate objects in the scene, interact with other user's avatar, or formulate action plan according to environment variation [1].

2. VRML and Avatar

VRML is a kind of modeling language for constructing virtual environment. It's a 3D interchange format, mainly used to describe three dimensional object and action. Some usually used elements for 3D application system have been defined, such as hierarchy transition, light source, viewpoint, geometrical shape, animation, fog, material characteristic and texture mapping [2]. The basic target of VRML is to establish three dimensional and interactive scenes on Internet. Its foundation characteristics includes distribution, three dimension, interaction, multimedia integration and scene verisimilitude.

In a cooperative multi-user environment, we can represent a user with a 3D avatar, in the name of this user, in VRML's virtual scene. By programming PROTO sentence, VRML can define multi client object. Users can choose system provided avatar object, or alternatively employ object defined by him selves. In this way, multi-user will be able to apperceive current colony's cooperative working status. In order to realize the effect of what you see is what I see (WYSIWIS), we should bring all users' feeedback information into correspondence. A 3D avatar is a special object in VRML scene. It is the concrete expression of user's figure and behavior in virtual environment [3].

By employing VRML's ProximitySensor node, system can trace local user viewpoint's shift and whirl status, simultaneously obtaining the Position_changed and Orientation_changed value. Then this two fields' value can be used to change 3D avatar's position and orientation for other users, who are in the same virtual scene. Avatars can exchange information among themselves, through text or animation mode. By the control of Script node's TimeSensor sensor, further control OrientationInterpolator and PositionInterpolator interpolator, we can dynamically change avatar object's orientation to realize avatar animation. In this way, other users will be able to apperceive this user's shifting and whirling to which object, in which region, and what is the content who is paying close attention to. When a user goes into or gets out from the scene, its 3D avatar should be increased or decreased from virtual cooperative scene.

978-0-7695-3489-3/08 $25.00 © 2008 IEEE
DOI 10.1109/ICACTE.2008.19

3. VRML Extension

There are mainly used four kinds of techniques to extend VRML function. They are described as follows.

3.1 Embedded VRML into HTML

The accessing manner of VRML is based on Client / Server mode. Users download necessary files, and then interactively access its described virtual environment, in the manner of installed VRML browser plugging in local platform. By using <Embed> and <Object> et cetera tag of HTML file, it is simple to embody VRML files into HTML.

3.2 Associating Script Node with Java Class

Because VRML is a modeling language, not a programming language, it is impossible to construct a complete system by merely employing VRML's itself object interactive mechanism. VRML does not support some elementary characteristics of common program designing language, for example condition branch, jump branch and loop sentence etc. So, in practical virtual designing, some complicate functions can be realized by Java class associating with relative VRML's Script node. Specified Java class will produce a series of actions when the corresponding Script node receives event. In the same time, it also can modify nodes' field value, or let node generate event. Relative VRML's event architecture is shown below [4].

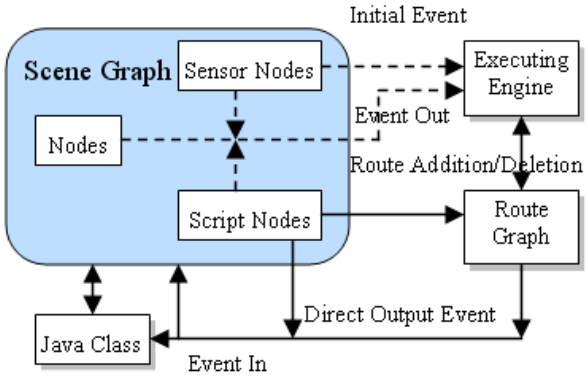

Fig.1 Event architecture and Java class

3.3 Quoting JavaScript Code

In VRML, by designating Script node's URL field value, a programmer can directly point this hyperlink to designated JavaScript's URL. Or, we can set script in the manner of VRML's Inline sentence.

3.4 Program through EAI

In the way of employing VRML's external authoring interface (EAI), users can design external application program to realize communications between virtual scene and Java Applet. In VRML 97, every node has a TimeStamp to record node event's occurring sequence. When receiving an EventIn, Script node will treat it as a standard Java platform object event. The event object has three fields to store relative information. They are Name, Value and TimeStamp. Theirs values are transmitted in from EventIn [5].

4. VRML Browser

VRML defines a sort of file format for integrating three dimension objects with multimedia elements. From the perspective of grammar, VRML is a congregation of 3D multimedia objects defined and organized explicitly. In the angle of semantics, VRML file describes the abstract function activity for interactive 3D multimedia information, based on time.

4. 1 Function of VRML Browser [6]

The explanation, execution and presentation of VRML files are realized through VRML browser. The browser is responsible for presenting sight and sound experience of scene graphs. We called this presentation virtual world. Person or mechanical entity, that is users, can manipulate the browser for navigation in virtual world. The presented realm will be experienced as an effect from a particular direction and position of user. This kind of direction and position choice in the scene is called viewfinder. VRML browser provides some predefined navigation paradigms. By applying them, users move viewfinder when navigating in the world. By program VRML detecting nodes, browser can respond to user's interaction with geometric solid. Figure 2 describes a VRML browser concept model. The browser is showed as presentation of application program, and accepts user input in the mode of VRML file selection or users intervene. Grammar parser, visual presentation and sound presentation are three major parts of VRML browser.

There are two ways of switching from one viewpoint to another, fast skip and gradual progress. Generally speaking, viewpoint's fast skipping can be used to definite those important and exciting observation point in the virtual world. Its advantage is to let viewers ignore those ordinary and common scenes. Rather than, the progressive mechanism can be

used to build smooth transition among viewpoints. This can also be seen as the explore animation here.

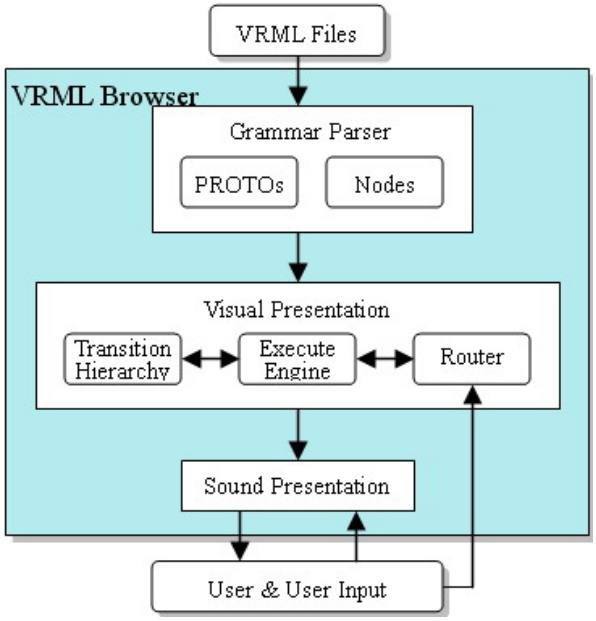

Fig.2 VRML browser concept model

4.2 Browser Navigation [7]

Through the use of 3D modeling as a viewer avatar in virtual world, navigation is the progress of avatar's walk, observation and 3D interaction in VRML scene. In virtual world, we can use NavigationInfo node to provide explorer's avatar information and set system navigation mode concerning avatar's current viewpoint.

The NavigationInfo node's grammatical pattern is shortened below.

```
NavigationInfo {
   type ["WALK" "ANY"]  #exposedField MFString
   avatarSize  [0.25  1.6  0.75]  #exposedField
MFFloat}
```

The type field's value is used to install explorer avatar's navigation type. There are five navigation types to be chosen. They are WALK, FLY, EXAMINE, ANY and NONE. Different navigation types apply to different situations.

The avatarSize field's value is used to install explorer avatar's physique parameter. These parameters determine the collision detection and user perspective scope with the undulating terrain. When running VRML program, an invisible explorer avatar can be assumed in three-dimensional space. Usually this avatar also can be used to realize collision detection.

5. User Avatar and Collision Detection [8]

By default, VRML executes collision detection for all the objects in the scene. By using this default detection, we can prevent some phenomenon from happening, such as user going right through a wall.

Following is a shortened demonstration program. Its purpose is to set default user avatar's size dimension by using avatarSize field's value of NavigationInfo node. Then we can allow explorer's avatar come across the left free position of foreground, and obstruct by the right free position and wall.

```
#VRML V2.0 utf8
   NavigationInfo {
        avatarSize [1.1 5.0 2.0]}
   Transform {
        translation 0.0 0.0 -20.0
        # background wall and middle wall}
   Transform {
        translation -5.5 0.0 0.0
        # left wall}
   Transform {
        translation 5.0 0.0 0.0
        # right wall}
```

The foreground of the scene is three complete opaque walls. The width of the left free position is 3 VRML units, yet the width of right is 2 VRML units. The background of the scene is an intact wall, which rendered by a picture texture and apart from the foreground wall 20 VRML units.

In program code, the field value of avatarSize is set as [1.1 5.0 2.0]. As the component of avatarSize [0] is 1.1, so the avatar's diameter is 2.2. This avatar will be obstruct by the right free position, contraryly can come across the left free position. In the mode of browser "WALK", avatar's movement will be influenced by environment and terrain.

The initial scene molding circumstances is shown in figure 3 (a).

By installing Cosmo Player plugging, users can try to push his avatar forward. When aiming to the left or right free position of foreground, we can operate  Go button or  Slide button in Cosmo control panel, to move or rotate scene environment. In the process of progressing through the right free position, gradually the user will be obstruct firmly, no matter whatever effort to carry out, however using his talents to fullest. The obstruct scene effect is shown in figure 3 (b). Figure 3 (c) demonstrates the situation

when avatar coming across from the left free position and then looking back to the foreground walls.

(a) Initial scene

(b) Obstruct by the right free position

(c) Come across the left free position and look back

Fig.3 Invisible explorer avatar and collision detection

6. Conclusion

As VRML and JAVA has the characteristics of nothing to do with system platform, the virtual avatar interactive technology is platform-independent and transplant-strong characteristics. To avoid the graphics file's direct transmission on the network, VRML uses text messages to communicate, transferring the text describing information of graphics files. By shifting complex processing tasks to the local machine, VRML system greatly reduced the network load.

The implementation mechanism of VRML dynamic interactive model is event. Events are used to transmit transformation between scenes and external language, so separate basic elements of the implementation modalities from each other. The dynamic behavior of driving scene change is executed in VRML external. Its effectiveness is to implement interactive solution simplicity and flexibility.

7. Reference

[1] Ding Dawei, Zhu Miaoliang, A Dynamic Correlation Analysis based Awareness Model in MVE, Special issue, Computer Science [J], Vol.29 (11), 2002

[2] Manninen T. Rich, Interaction in Networked Virtual Environments, Proc. of the 8th ACM Conf. on Multimedia, 517-518, 2000

[3] Zhou Yingbiao, Feng Yucai, Multi-user VRML Virtual Environment Design [J], Computer System Application, 2000 (5), 17-19

[4] Adrian Scott, The marriage of Java and VRML, [EB/OL], http://www.vrmlsite.com

[5] Chen Yi, Chen Zelin, Research of the Virtual CSCW Systems Based on VRML [J], Application Research of Computers, 2003 (4), 49-51

[6] Xu Donghui, Liu Yisong, Research and Application of 3D Chat Room based on Distributed VRML Environment [J], Computer Engineering and Application, 2004.17, 141-143

[7] Duan Xinyu, Virtual Reality Foundation and VRML Program [M], Beijing: Higher Education Press, 2004

[8] Huang Tiejun, Liu Jian, VRML International Standard and Application guide [M], Beijing: Electronic Engineering Publishing Company, 1999