Intelligent Web Agents for a 3D Virtual Community

T.M. Dave, Yan-Qing Zhang, G.S. Scott Owen and R. Sunderraman

Department of Computer Science, Georgia State University, Atlanta, GA 30303, U.S.A.

ABSTRACT

In this paper, we propose an Avatar-based intelligent agent technique for 3D Web based Virtual Communities based on distributed artificial intelligence, intelligent agent techniques, and databases and knowledge bases in a digital library. One of the goals of this joint NSF (IIS-9980130) and ACM SIGGRAPH Education Committee (ASEC) project is to create a virtual community of educators and students who have a common interest in computer graphics, visualization, and interactive techniques. In this virtual community (ASEC World) Avatars will represent the educators, students, and other visitors to the world. Intelligent agents represented as specially dressed Avatars will be available to assist the visitors to ASEC World. The basic Web client-server architecture of the intelligent knowledge-based avatars is given. Importantly, the intelligent Web agent software system for the 3D virtual community is implemented successfully.

Keywords: Intelligent Agents, Digital Library, Graphics and Visualization, Electronic Education, Virtual Reality, Avatars.

1. INTRODUCTION

In recent years, intelligent agent techniques and other advanced methods have been widely used in Web-based applications such as E-Commerce, Web search engines, Web mining, 3D Web applications, etc [1][3-6][11][12][19-22][31][33][34]. Web based virtual communities; including educationally oriented ones is another potentially huge market that can combine intelligent agent technology, distributed systems, and digital libraries. Intelligent 3D agents are a new category of information society tools. We are studying intelligent agents applied to educational virtual communities. A natural extension of this is to apply intelligent agents technology to virtual classes, i.e., in the university environment for aiding online teaching and as teaching assistants for students. This paper focuses on using Avatar intelligent agents in a virtual online community. The major goal is to create a 3D-virtual community (ASEC World), using VRML based technology that provides access to the graphics and visualization educational materials in the ASEC Digital Library (ASECDL) and common spaces for visitors to congregate and discuss their interests. There will be sub-worlds in ASEC World that are devoted to specific topics, e.g. computer animation. ASEC World will also have special "student volunteer" (SV) avatars (smart agents) that will be able to answer questions and facilitate communication between visitors. Each sub-world of ASEC World has its own agents that have knowledge related to the sub world's domain.

Developing a sense of community among Digital Library users and developers is one of the important issues supported by a NSF grant (IIS-9980130). A real time chat room engenders some sense of community, but is text based and not visual. Since the ASECDL contains materials for instruction in computer graphics, visualization, and interactive techniques, it primarily appeals to a visually oriented audience. Therefore, a visual 3D virtual world is more appropriate and effective for this community. As more computer users are exposed to 3D environments, such as games, online 3D virtual worlds are going to become more common and expected.

There are various different intelligent agents such as mobile agents, interface agents, information agents, knowledge agents, collaborative agents, Internet agents, and hybrid agents [20]. In this paper, a hybrid intelligent agent is developed based on 3D–graphics-based interface agents, information/knowledge agents and Internet agents. The objective of designing ASEC World with the hybrid intelligent agents is to generate a 3D world that will represent the virtual community and space corresponding to the ASECDL. Each area or sub-world of ASEC World will have SV agents. These 3D avatar agents with knowledge bases containing both domain-specific questions and relevant answers can move if they want to contact someone, talk when they need and disappear if necessary.

Section 2 shows the architecture of a client-server-based ASEC World agents. Section 3 describes how to actually implement the intelligent Web agents in the 3D ASEC World. Section 4 indicates various possible applications. Finally, section 5 gives major conclusions.

Intelligent Computing: Theory and Applications, Kevin L. Priddy, Peter J. Angeline, Editors, Proceedings of SPIE Vol. 5103 (2003) © 2003 SPIE · 0277-786X/03/$15.00

2. ARCHITECTURE OF INTELLIGENT-AGENTS-BASED VIRTUAL WORLD

Our interest in distributed multi-user environment was stimulated by a general interest in VRML and then by its practical application in education. We outlined our VRML-based multi-user system [34] and finally implemented a prototype. Our approach is guided by the following requirements:

Openness: The system should be open and universal for different hardware and software environment.

Consistency: Distributed users in the same world are supposed to have the same view.

Dynamics: Users can freely play in the virtual environment at any moment.

Database-Interface: The system should have an interface to a database of objects, avatars and behaviors.

The client/server [15] software architecture is a modular infrastructure to improve usability, flexibility, interoperability, and scalability.

Since there is direct communication only between clients and the server, no modifications can be missed by the server. Our MUSE employs centralized data management with data replication on demand. A client knows at most one world at any moment. When a client switches from one world to another, it receives necessary information of the new world from the server.

The MUSE's requires allowing multiple users to connect to a centralized system as shown in Figure 1. The centralized system can co-ordinate the activities of users with similar requests. The agent layer controls the communication between the user and centralized server. Thus the system can be divided into three architectures: (1) Client Architecture, (2) Server Architecture, and (3) Agent Architecture.

The architecture of the 3D world multi-user server environment is client-server based. The client-server is web based with the server being hosted on the web server and the client connects to the server on the Internet.

An agent can be assumed as a black box that receives a range of inputs that it accepts using a perception component. It uses its intelligence to process these inputs and produces an output.

The major task of an agent is to use interaction modules that specifically match to the capabilities and particularities of the corresponding interaction partner. The agent uses interaction modules to obtain information and changes within its environment and also to initiate its own actions. The central task of agents is not to interact with the environment but rather process and interpret the perceived information and to achieve its own goals. The incoming information must be first integrated in an appropriate manner and accepted in agent's knowledge base. This process is called information fusion. The next step is process this external information. This is the central component of the agent because it reflects the agent's true functionality. The aim of information processing is to interpret the available data and to form a specific plan. Final step of the agent is to use in the form of output or actions.

In the distributed environment we use an agent to assert the users in the 3D world. There is an agent in each room or world of the 3-D environment. The agent has a knowledge base, which it uses for information processing. The users input are used in the information fusion task. The knowledge base resides on a relational database management system (RDBMS). The access to this RDBMS is through Java's open database connectivity JDBC. The agent stores and retrieves information for processing the information using this knowledge base.

3. REAL IMPLEMENTATION

This virtual community is implemented on the client/server architecture on the web (see Figure 3). The multi-user environment is hosted on a HTTP server. The client accesses the server through the Internet through a Java applet. The server is managing the connections to the multi-user environment.

On the client when the user logs to the server he has the option to choose a world that he wants to join. Based on the world he enters he is assigned an agent to him if the world is empty If there is already somebody in the world the name of the new person is added in the agent's list. It is the job of this agent to take care of this visitor. When the knowledge base is inadequate to answer the visitor's questions, the agent will contact other agents, or a member of the ASEC who can answer it and the answer is passed on to the visitor.

If the visitor wants to go to other worlds she can ask the agent itself and it would communicate with other agents with the central moderator and let the person know about which other worlds would be good for her. The person can just navigate through the virtual worlds and find a suitable area for herself.

The agent works on its local knowledge base and in order to access other agents' knowledge it has to go through the central moderator that keeps track of the other agents' specific interests [13]. Thus the major functions of this agent would be Reactivity, Goal Orientation, Communication/co-operation, Reasoning, Autonomy, and Character.

This project uses Java, XML, VRML technologies for development [36][17][26][1] and artificial intelligence techniques (such as rule-based expert systems).

In order to allow multiple users to in the single environment [17], information has to be shared between clients. Since these clients may be distributed over a heterogeneous network, any implementation has to be platform independent.

The virtual environments try to mimic the real world. In the virtual environment you create visual representation of a synthetic environment, acoustic and other sensual simulation is equally important. A proper multimedia-rendering engine has to be platform independent as well, and is therefore essential to guarantee smooth visualization of the environment.

In the shared environment the users has to able to interact with the environment [36], other users and the objects in the environment. Thus the application has to provide the functionality that permits the manipulation of the multimedia information [1] and the way it is rendered.

Open technologies are optimal for satisfying the above requirements. The technologies include Java, VRML, EAI of the VRML, RDBMS. The implementation and authoring of multi-user environments requires thorough knowledge of VRML, Java, JavaScript, and the EAI.

A brief description about these technologies is given below.

Java is an object–oriented programming language developed by the Sun Microsystems. Java is a web programming language, which supports platform independence. Java code can be executed in HTML browsers and used as the scripting language in VRML as well. Java applets can open network connections to servers as well as present graphical user interface. Furthermore with the help of External Authoring Interface, it can access the VRML plugin functionality [4] to display and control visual simulation.

In traditional programming languages a compiler or a runtime interpreter is used to convert program source code into system specific binary code Java adopts another route here. The Java compiler doe not directly translate the Java source code into binary code but so called Java byte code. This byte code is platform independent and can be executed without modification on all platforms that support Java. A Java interpreter developed for a particular platform is used to execute the byte code on the target platform. Thus Java uses both compiler (to create the byte code) and an interpreter (to execute the byte code). A so-called virtual machine is used to execute the Java byte-code on the target platform. A virtual machine is added to the existing operating system of the target computer and provides a simulated consistent runtime environment. Irrespective of the actual system platform Java virtual machine always provides a Java program with standardized runtime environment.

The following are the features, which makes it attractive for multi-user server environment.

The Virtual Reality Modeling Language (VRML) is a file format for the description of dynamic scene graphs containing 3D objects with their visual appearance, multimedia content, an event model, and scripting capabilities. VRML is designed to be used on the Internet and local client systems and to be used as an exchange file format. VRML is the universal language for integrated 3D graphics and multimedia [13]. The multi-user distributed world use VRML [24] as the rendering and interaction vehicle.

VRML is designed based on following criteria

Author ability: Enable the development of computer programs of creating, editing and maintaining VRML files as well as automatic translation programs for converting other commonly used 3D file formats into VRML files

Composability: Provide the ability to use and combine dynamic 3D objects in the VRML world and thus allow re-usability

Extensibility: Provide ability to add new object types not explicitly defined in VRML

Be capable of implementation: Capable of implementation on a wide range of systems

Performance: Emphasize scalable, interactive performance on wide range of computing platforms

Scalability: Enable arbitrarily large dynamic VRML worlds.

VRML is capable of representing static and animated dynamic 3D and multimedia objects with hyperlinks to other media such as text sounds movies and images. VRML browsers as well as authoring tools for the creation of VRML files are widely available for many platforms. VRML supports extensibility model that allows new dynamic 3D-object [18] to be defined allowing application communities to develop interoperable extensions.

EAI defines the interface that applications external to the VRML browser in our case the applet to access and manipulate the objects in the VRML scene. The External Authoring Interface connects the Java Virtual Machine running in the web browser to execute applets and plug in used to display VRML content as shown in figure. It is accessed with a set of Java classes defined in the EAI Specification [29]. The client uses the EAI to control the visual simulation in the VRML plug in. Thus EAI allows the external environment to access the nodes in the VRML scene using the existing

VRML event model. In this model an eventOut of a given node can be routed to an eventIn of another node. When the eventOut generates an event the eventIn is notified and its node processes that event. Additionally, if the Script node has a reference to a given node it can send events directly to any eventIn of that node and it can read the last value sent from any of its eventOuts.

This interface allows a two-way communication between the application and the VRML browser. The applet loads VRML content into the plug-in and adds the avatar geometry to the existing world representation. Navigation is possible by using plug in controls therefore the plug-ins have to update the applet about the users current position and orientation in the world an pass events to from the VRML scene.

The External Authoring Interface allows four types of access into the VRML scene: (1) Accessing the functionality of the Browser script interface, (2) Sending events to eventIns of nodes inside the scene, (3) Reading the last value sent from event outs of nodes inside the scene, and (4) Getting notified when events change values of node fields inside the scene.

There are three main items in the VRML browser that can be accessed from an external application: The browser, nodes within the scene graph and fields within nodes. The VRML browser exposes a set of services, which allow the external application the JVM of applet to interact with it.

When multiple applications make request of the browser the request can be serviced in the order of arrival time at the browser. The arrival time is used to sort out conflicting request from the multiple applications to ensure consistent results in the application in the correct order.

4. CONCLUSIONS

ASEC World provides a prototype of the 3D virtual communities of tomorrow. It can simulate a virtual meeting space with avatars representing both visitors and intelligent agents. Students, faculty, and other visitors are able to communicate more comfortably and more efficiently and can use the agents for quick access to information. This system is a multi-user VRML application. It enables several users to share a single VRML world and interact with it as it is possible with an ordinary VRML world. VRML events are distributed between the several instances of the VRML world on the different user computers. The system is split into a server program running on the web server where the VRML worlds and supporting HTML files are stored. The client is realized as a Java applet, which uses the EAI to communicate with a VRML browser plugin such as Blaxxun Contact or ParallelGraphics Cortona. The agent is the virtual helper who provides information for the users such as students and instructors.

ACKNOWLEDGMENTS

The authors would like to appreciate the support by NSF under Grant IIS-9980130 and the ACM SIGGRAPH Education Committee.

REFERENCES

1. Ana Carolina Hermann, Luciano Paschoal Gaspary and Janilce B. Almeida; "Design and execution of adaptive multimedia applications in the Internet;" Proceedings of the 1999 ACM symposium on Applied computing, 1999, Pages 624 - 629.
2. Andrew Yang and Yael Bachar; Using Java and the socket interface in teaching client/server programming Proceedings of the 4th annual SIGCSE/SIGCUE on Innovation and technology in computer science education, 1999, Page 206.
3. Beth Adelson, Evocative agents and multi-media interface design,Conference proceedings on Human factors in computing systems, 1992, Pages 351 - 356 .
4. Bernhard Jung and Jan-Torsten Milde; An open virtual environment for autonomous agents using VRML and Java Proceedings of the fourth symposium on The virtual reality modeling language, 1999, Pages 7 - 11.
5. Blaxxun Corporation http://www.blaxxun.com.
6. Carlos Martinho and Ana Paiva, Pathematic Agents: Rapid Development of Believable Emotional Agents in Intelligent Virtual Environments, Proceedings of the third annual conference on Autonomous Agents, 1999, Pages 1 - 8.
7. Dan Ancona, "*3DXML* Authoring, Publishing and Viewing Structured Information," http://www.vrml.org/WorkingGroups/dbwork/ancona/home.html, August 17, 1998.
8. Daniel Lipkin "Integrating XML and VRML: A Technical Discussion", http://www.vrml.org/WorkingGroups/dbwork/vrmlxml.html July 30, 1998.

9.David A. Rabenhorst, Chu P. Wang, Alan Norton, Paula Sweeney and Greg Turk; "Interactive simulation in a multi-person virtual world" Conference proceedings on Human factors in computing systems, 1992, Pages 329 - 334.
10. David Waller and Jon Miller; A desktop virtual environment trainer provides superior retention of a spatial assembly skill, Proceedings of the conference on CHI 98 summary: human factors in computing systems, 1998, Page 339.
11. Dimitris N Chorafas, "Agent Technology Handbook," McGraw Hill, 1997.
12. Filippo Menczer and Richard K. Belew, "Adaptive information agents in distributed textual environments", Proceedings of the second international conference on Autonomous agents, 1998, Pages 157 - 164.
13. T. Goddard, V. S. Sunderam, "ToolSpace: Web Based 3D Collaboration",Proceedings of the fourth symposium on The virtual reality modeling language, 1999, Pages 161 – 166.
14.Grand S. et al. Creatures: Artificial Autonom-ous Software Agents for Home Entertainment. Proceedings of Autonomous Agents 97 Conference,ACM Press, 1997.
15. Hal Berghel, CYBER-SURFING:the State-of-the-Art in Client Server Browsing and Navigation, Proceedings of the 1995 ACM symposium on Applied computing, 1995, Pages 1 - 4.
16.John C. Doppke, Dennis Heimbigner and Alexander L. Wolf; Software process modeling and execution within virtual environments ACM Trans. Softw. Eng. Methodol. 7, 1 (Jan. 1998), Pages 1 - 40 .
17. Kurt Saar, VIRTUS: A Collaborative Multi-User Platform, VRML 99 Paderbom Gemany, 141-152.
18. Masaaki Taniguchi; Event processing for complicated routes in VRML 2.0 Proceedings of the third symposium on Virtual reality modeling language, 1998, Pages 83 - 88 .
19. Mattais Klusch ,Gerhard Weib, "Cop-operative Information agents II, learning, Mobility-and Electronic Commerce for Information Discovery on the Internet," Second International Workshop '98 Paris, France, 1998.
20. H. S. Nwana and N. Azarmi, " Software A gents a nd S oft C omputing: T owards E nhancing M achine I ntelligence," Springer-Verlag, 1997.
21. Patrick Doyle and Barbara Hayes-Roth; "Agents in annotated worlds", Proceedings of the second international conference on Autonomous agents, 1998, Pages 173 – 180.
22. Saacha Ossowski,"Co-ordination in Artificil Agent Societies," Springer, 1999.
23. G. Scott Owen, Raj Sunderraman and Y. Zhang, "The Development of a Digital Library to Support the Teaching of Computer Graphics and Visualization," *Proc. of GVE'99*, pp. 62-67, 1999.
24. Sungwoo Park, Taissok Han. Object-Oriented VRML for Multi-user Environments. Proceedings of the VRML '97 Symposium, pages 25-32, ACM, 1997.
25. Thomas A. Funkhouser; RING: a client-server system for multi-user virtual environments Proceedings of the 1995 symposium on Interactive 3D graphics, 1995, Page 85.
26. Vicki de Mey and Simon Gibbs;"A multimedia component kit: experiences with visual composition of applications" Proceedings of the first ACM international conference on Multimedia, 1993, Pages 291 - 300.
27. *VRML Standard Version 2.0, ISO/IEC CD 14772*, 1996, http://vrml.org/VRML2.0/
28. VRML Repository, http://www.web3d.org/vrml/
29. *VRML External Authoring Interface specifications*, 1997, http://vrml.org/WorkingGroups/vrml-eai/
30. *VRML Living Worlds Working Group proposal draft 2*, 1998, http://vrml.org/WorkingGroups/living-worlds/
31. Walter Brenner, Rudiger Zarnekow and hartmut Witting, "Intelligent Software Agents: foundations and applications," Springer, 1998.
32. Web3D Consortium http://www.vrml.org/fs_specifications.htm.
33. Winfried Lamersdorf, Michael Merz, "Trends in Distributed Systems for E-Commerce, International IFIP/GI Working Conference Trec'98," Hamburg Germany, 1998.
34. Wolfgang Broll and David England; Bringing worlds together: adding multi-user support to VRML; Proceedings of the 1995 symposium on Virtual reality modeling language, 1995, Pages 87 - 94.
35. Wolfgang Broll; Populating the Internet: supporting multiple users and shared applications with VRML Proceedings of the second symposium on Virtual reality modeling language, 1997, Page 33.
36. Yasuaki Honda, Kouichi Matsuda, Jun Rekimoto, Rodger Lea,"Virtual Society: Extending the WWW to Support a Multi-user Interactive Shared 3D Environment", VRML '95, San Diego CA USA, 109-116.