MuDiS - A Virtual Learning Environment

by

Jeetinder Singh, Sivaswamy J, Krishnarajulu Naidu

in


Report No: IIIT/TR/2008/139

Centre for IT in Education
International Institute of Information Technology
Hyderabad - 500 032, INDIA
November 2008
Abstract

This paper presents a multi-component; distributed system (MuDiS) based solution for building a virtual learning environment which combines a wide range of technology, tools and digital gaming concepts to create an interactive tool for science education. MuDiS has been designed to be an extensible and easy-to-use system. The proposed environment is intended for designing and monitoring of educational content as well as creating a platform for exploring ideas. The system allows exchange of educational content and integrate different pedagogical approaches to learning and teaching under the same environment. We have developed a virtual physics lab that serves as an exemplar for the proposed virtual environment. We present the design details of the physics lab and discuss its performance.

Keywords: Simulation Space (SS), Virtual Environment (VE), Virtual Physics Lab, MuDiS network

1. Introduction

With advances in technology and proliferation of computers in schools and homes, there is now an opportunity to create a new culture of learning. In such a culture, curiosity and innovative thinking is encouraged by the provision of a novel means for testing ideas. A flexible simulation space (SS) for experimentation is an example of a novel solution that enables the testing process.

VEs are interactive systems which not only create curiosity but also drive one to test ideas in different conditions. Various Java applets or Flash-based animations available on the Internet are examples of VEs where the environments are highly constrained. An alternative is to create open environments and provide some basic tools with which users can construct experiments or educational content with minimal need for knowledge computer programming. Examples of this approach can be found in the 2D tool Squeak E-toys [1], 3D tools for games JiVE [2], programming of animated character [3], and science video games [4].

In this paper, we present an approach to create a MuDiS that can be used in science education at primarily the school level, though it is also usable in tertiary education. At the core of the proposed MuDiS is a virtual simulation space consisting of objects that respond to user inputs and interact with each other, mimicking real world behaviors. MuDiS is a shared, scalable system that leverages multimedia, networking and the WWW to simulate different virtual worlds, analyse data, and allow multiple users to interact. The intent in designing such a MuDiS is to permit dual uses, namely, as an instructional tool and as a tool for exploration of ideas.

The following sections first describe the concept and design behind the development of MuDiS for science learning and then present an exemplar in the domain of physics education.

2. MuDiS - A virtual environment for science learning

The basic components of the proposed MuDiS are as follows: Learning resource library, Authoring tool, MuDiS network and Kernel. The modular architecture is aimed at making MuDiS versatile, scaleable and flexible. We will discuss each module in turn next.

A Learning resource library (LL) is a distributed database of educational content to help perform experiments (in the form of applets, animations, etc) created within the MuDiS or added from external sources. The LL can be broken into a personal versus shared network parts. The personal part is created by a user on a specific machine whereas the network part is created by sharing personal content.

An authoring tool is a means to create experiments, and test ideas. The authoring tool allows any lay user to create and simulate an environment merely by using available objects, defining relationships between them and sequencing their interactions in an appropriate order without the requirement of technical knowledge. It should help set up an intelligent environment for a required
domain with seamless integration of domain-specific simulation sub-engines defining the interaction between objects and objects with the environment. This includes gaming type of environments. The authoring tool can be used to integrate new technologies for exploring ideas using a graphical rendering of experimental objects, perform scientific calculations, and interact with users to provide flexibility to the user in their exploration. Specifically, it provides a visually rich environment where information manipulation and display is possible by variety modes of interaction to support different views and levels of comfort for a user.

Network communication via text messaging and voice communication will improve education-aid tools which can help students to exchange, clarify their views and doubts and sharing the resources. Next we will discuss the design of mobile server centralized-distributive architecture which can be adopted to create a robust network system. The MuDiS network can be divided into local centralized network (LCNet) and global distributed network (GDNet). The LCNet consists of a network where one machine will act as a local centralized server, which will update all other machines in the network as and when users will sign-in and sign-out from the network. The GDNet consists of individual connected LCNet communicating with each other via local centralized servers. Each LCNet will act as node of contact for local network (LCNet) and for other global distributed local centralized server (GDNet). When a MuDiS user machine is connected to the network, it will search for the LCNet in the network. If the local MuDiS server is found then it will provide required information to the local centralized server to get registered to MuDiS network or else if no local centralized server is found in the network it will opt for serving as the local centralized server in the MuDiS network. Once the user is connected to MuDiS network, communication between users is carried out directly from the source to the destination. Each user is then an autonomous node in the MuDiS network whereas local MuDiS server will act as a node that will update participant user directory list if and when user will get connected or signed out from MuDiS network. In the event of the breakdown of a server node, each machine in the LCNet will opt for upgrading to be a centralized server node and the machine satisfying the required criterion will be permitted to do so. The primary drawback of this architecture is that it does not have a shared database. As a result, users will have to maintain their own full copy of resources.

The kernel coordinates the processes within the MuDiS. Based on the above proposed design, the specifications and role of the modules will be crisply defined, leading to independent and focused design of each module.

We next present a virtual physics lab that we have implemented as an exemplar of the proposed MuDiS designed.

3. Virtual Physics Lab- ViPLab

A virtual physics lab (ViPLab) was implemented using the above design to support primary to high school physics education. The LL has stand-alone experiments that can be used to test specific concepts in physics. These stand-alone experiments use interactive simulations which are annotated and enriched with real-time graphs, audio, video and text.

The authoring tool provides a virtual space, toolkits with a GUI to create new experiments. A collection of basic 2D geometric shapes (to model square block, round object, spring, rope etc.,) represent a set of objects which can be used as building blocks to create complex structures in the virtual space. The lab has different modes which are states of the environment at any give instant of time viz. set, play, pause, reset experiment. These are for creating, conducting, pausing and restarting with the last configuration, respectively, of the experiment. An experiment structure can be saved at any point of time into a database for later use and modifications. The environment properties can be changed by a user to perform experiments.
The physics engine module helps in simulating the physical environment once a required structure has been constructed. The physics engine tracks the objects and updates their status at regular intervals.

Input devices currently supported are mouse, keyboard in view of the target users. The lab space allows recording statistical observations or data for the current experiment and can record data of interest. The user interface supports dynamic graph generation to visualise the relationship between the experimental variables.

Currently teachers and students can connect to other machines with client-server approach. This feature helps in opening a discussion room on the network and enables viewing of, or explaining an experiment from a remote machine.

The basic ViPLab prototype has been viewed and used by about 200 school science teachers - though all of them were not necessarily Physics teachers - and by a number of undergraduate engineering students. The novelty of the approach and the ease-of-use of the tool were very much appreciated by all.

However, based on our experience of the initial development of the ViPLab and from peer review we have observed several constraints in simulation of a “real” world with available resources. A few are listed below.

1. Due to accumulation of mathematical calculation error over time the simulation results deviate from accepted actual behavior.
2. To simulate a wide range of concepts related to physics, different sub-physics engines are needed which work in synchronisation with each other and with a common object data structure.
3. When the number of objects in the environment is increased, the performance of the lab degrades depending on the PC configuration.
4. With the advancement in technology, GUI design needs to keep a tab on current technology to provide different solutions. This impacts regular changes to GUI design.
5. Simulation of a multi-user system that consists of multiple computers geographically distributed at various places requires additional components.

4. Conclusions

Advances in computing, multimedia and communication technology provides an opportunity to build a self growing, unit sharing virtual environment for teaching and learning. Our exemplar, namely, the ViPLab while validating the MuDiS proposal presented in this paper, points to the problems that need to be solved for the full realization of the MuDiS as an educational tool.

Acknowledgement

We are grateful to Media Labs Asia, India for partial support for this work through a Project Grant No. MLA/IIIT-HYD/04/DITRS3/01.

References


