MUDs Grow Up: Social Virtual Reality in the Real World

Pavel Curtis and David A. Nichols

Xerox PARC

January 19, 1993

1 Introduction

MUDs, or "Multi-User Dungeons," are programs that accept network connections from multiple simultaneous users and provide access to a shared database of "rooms", "exits", and other objects. Users browse and manipulate the database from "inside" the rooms, seeing only those objects that are in the same room and moving between rooms mostly via the exits that connect them. MUDs are thus a kind of *virtual reality*, an electronically-represented "place" that users can visit.

MUDs are, however, unlike the kind of virtual realities that one usually hears about in three important ways:

- MUDs do not employ fancy graphics or special position-sensing hardware to immerse the user in a sensually vivid virtual environment; rather, they rely entirely on plain, unformatted text to communicate with the users. For this reason, MUDs are frequently referred to as *text-based virtual realities*. [1]
- MUDs are extensible from within; MUD users can add new rooms and other objects to the database and give those objects unique virtual behavior, using an embedded programming language.
- MUDs generally have many users connected at the same time. All of those users are browsing and manipulating the same database and can encounter both the other users and their newly-created objects. MUD users can also communicate with each other directly, in real time, usually by typing messages that are seen by all other users in the same room.

We believe that this last difference provides the most interesting basis for generalization, so we refer to MUDs as the first examples of *social virtual realities*.

MUDs have existed for about ten years, becoming particularly prominent on the global Internet in the past five years or so. Throughout that time, they have been used almost exclusively for recreational purposes. Many MUDs are specialized for playing a game rather like "Dungeons and Dragons," in which the players are assigned numerical measures of various physical and mental characteristics and then have fantasy adventures in a role-playing style. Nearly all other MUDs are used almost exclusively for leisure-time social activity, with the participants spending their connected periods talking with each other and building new areas or objects for general enjoyment.

A recent list of Internet-accessible MUDs showed well over 200 advertised, running at sites all over the world. The busiest of these frequently host 50 to 100 simultaneous users. Clearly, these recreational MUDs are very popular systems.

It seems clear to us that the simple technology of MUDs should also be useful in other, non-recreational applications. This paper presents our plans for exploring the implementation, applications, and implications of MUDs in work-oriented contexts. In the remainder of this introduction, we describe the capabilities of our own MUD server. We then describe the two major systems we are building as foci for our research.

The LambdaMOO Server

We use a MUD server developed here at PARC, called "LambdaMOO." The facilities it provides are generally typical of those available on other MUD servers, though in a form that we find particularly convenient.

The server has only three jobs: accepting and managing network connections from the users, maintaining the database in a robust and persistent way, and executing programs written in the server's embedded programming language, called "MOO." Users invoke MOO programs each time they enter a command and the server interprets a simple compiled version of the MOO code.

MOO code has access to all of information about the objects stored in the database (including the code for all of the stored MOO programs) and produces its effects either by changing those objects or by reading or writing simple text strings from or to the various users' network connections. To make the programming model as simple as possible for naive users, each user command's associated program is run to completion before any other command is begun; thus, no issues of concurrency or locking arise.

As a consequence of this design, MOO programs do not run so quickly as to allow them to be used to handle any high-frequency events, such as character-by-character typing, mouse motions, or real-time network processing. On the other hand, the interpreter is efficient enough for use in most human-speed command-response interfaces.

2 The Astro-VR System

In collaboration with Dave Van Buren, an astronomer at the NASA/JPL Infrared Processing and Analysis Center, we are building "Astro-VR," a social virtual reality intended for use by the international astronomy community. We expect to have our first serious users, a major research project with principal investigators in both the United States and Italy, in March of 1993.

The system, built on top of the LambdaMOO server, is intended to provide a place for working astronomers to talk with one another, give short presentations, and otherwise collaborate on astronomical research. In most cases, this system will provide the only available means for active collaboration at a level beyond simple electronic mail.

Initially, Astro-VR will provide the following facilities of interest to our user community:

- real-time multi-user communication,
- a self-contained electronic mail and bulletin board system,
- shared, user-supplied *links* to online astronomical images,
- an editor/viewer for short presentations of text and images, and
- collaborative access to standard programs used by astronomers.

Because Astro-VR is built on top of LambdaMOO, all of these facilities can be extended and customized by individual users, using the embedded programming language built into the server.

Real-time Communication

The standard MUD means of textual self-expression and communication (e.g., speech, gestures, paging, whispering, self-description, etc.) are available to our users as well, because Astro-VR is, at heart, a MUD. While such communications channels are obviously not optimal, they are, at present, the best kind of conversational tools widely usable in today's global Internet. The wealth of experience already gained with recreational MUDs leads us to believe that this level of communications technology will prove useful and even sufficient until such time as something better (such as networked digitized voice) becomes feasible on a large scale.

Self-Contained Electronic Mail

The international astronomical community, like most modern scientific communities, already makes a great deal of use of electronic mail. However, there remains a need for email forums both concerned with astronomical research and restricted to use by working astronomers. Bulletin boards and mailing lists such as the newsgroups of USENET have the problem (and virtue, perhaps) that access is unlimited; every user on USENET has the option of reading and posting to any newsgroup. From the perspective of a serious practitioner in some field, this communication channel is very "noisy." The sci.space newsgroup, for example, is posted to by people with all levels of astronomical background and expertise, from utter novices to working professionals. The general level of discouse is thus driven toward the middle ground, the knowledgeable hobbyists. Clearly, such a forum is inappropriate and ineffective as an outlet for serious discussion between experts.

The standard LambdaMOO database includes an electronic mail and bulletin board system completely contained in the virtual reality; email sent on a MUD does not, in general, leave it and email from outside the MUD cannot come in. In effect, the MUD provides a self-contained electronic mail community; the users can send email and maintain bulletin boards that are open only to that community, even though the participants are geographically scattered. We expect the users to find that a much higher level of bulletin board discourse is possible inside Astro-VR than outside.

Active Links to Images

Astronomers, perhaps to a larger extent than many other professionals, rely heavily on detailed photographs and other images to convey their ideas and discoveries to one another. At present, however, it is difficult for them to make their images easily available to one another. For example, it is difficult to discover if any (local or remote) colleague has online an image of any particular celestial object, let alone to get access to that image and display it on a given computer screen. In Astro-VR, it is easy for users to "register" the images in their collections, creating an object in the virtual reality to represent it. When another user "opens" such an object, the corresponding image is automatically fetched across the network and displayed in a window on the requestor's screen. These active *links* to images will be indexed in a number of ways on Astro-VR (e.g., by name, location, type of image, type of celestial object, etc.) for searching and browsing by other users.

Presentation Editor and Viewer

An important part of the scientific work practice is the periodic meeting of relatively small working groups, usually having 5 to 20 members. At such meetings, individual project members frequently give short, 10 to 20 minute presentations on recent efforts and/or results. To facilitate this kind of activity within Astro-VR, we are providing a set of *conference rooms*, which have extra commands for preparing and presenting such short talks.

A *talk* is structured as a sequence of *segments*, each containing a paragraph or two of text and one image, accessed using the active links discussed above. When each segment is presented, its text is printed out to all participants and the associated image is displayed on their screens. A distinguished user, the *presenter*, controls the "pace" of the talk by deciding when to move from one segment to the next.

Any discussions that take place during the presentation are recorded and stored with the talk. These discussion may be displayed during subsequent presentations of the talk. This allows project members who were not present at the initial presentation to catch up on what happened there and even to add their own annotations to the talk for other listeners. All talks are potentially archived for viewing by interested parties at any later date.

Collaborative Access to Standard Tools

There are a number of computational tools frequently used by most working astronomers. For example, tools like the interactive plotting system mongo, the symbolic algebra system Mathematica, a variety of astronomical database systems, and even simple calculators all qualify as major tools in their day-to-day work. Through Astro-VR, we intend to allow for astronomers to use these familiar tools collaboratively. Using these tools will be like crowding around the screen of a shared workstation, handing the keyboard and/or mouse back and forth and discussing the results produced. In the case of Astro-VR, though, the shared workstation is a virtual construct and the users can "crowd around" from the comfort of their individual offices around the world.

It is important to note that we are not, in this case, attempting to provide any new tools for our user community; there are many more knowledgeable people in astronomy doing that already. Rather, we see our task as providing the "glue" that allows collaborative use of those tools the astronomers are already using.

3 The Jupiter System

To a large extent, Astro-VR represents the level of interface functionality that we can easily export to a large number of users, given the current state of the global Internet and the kinds of workstations typically available to our users. In an environment where we have much more control over the network and the workstations in use by the participants, though, we can explore the use of more advanced technologies. Then, as the Internet and users' workstations improve, we can use the lessons learned in our more controlled environment to improve the general lot.

Our other major effort, then, is an extended MUD that supports a richer set of communication media than plain text. This system, called "Jupiter," will be used by researchers both here at PARC and at our companion laboratory, EuroPARC, in England. In addition, it will support users both within the laboratory buildings and at their homes.

Enhancements to standard MUDs

Current MUDs use text as the sole communication medium, both between users and between the server and user. Text has the advantage of being universally available, allowing a wide range of users to participate. For recreational use, text has additional advantages similar to those of radio over television, as words can often paint a more compelling mental image than can a picture produced with MacDraw.

Unfortunately, text has significant drawbacks in a work environment. First, typing is much slower than speech for real-time communications. While most people can read faster than they can listen, they can speak much faster than they can type. In a world where telephones and face-to-face communications are readily available, real-time typing will not be a popular alternative.

Secondly, the aesthetic properties of text descriptions are largely wasted in a work setting. A worker's use of the computer is generally secondary to accomplishing some other task, and a dramatic description of the successful printing of a document, after defeating the evil forces of paper jams and toner spills, gets old after a while.

In addition to text-based interaction, therefore, Jupiter provides audio, video, and window-based graphical user interfaces. While typical MUDs assume that users have only a dumb terminal, Jupiter assumes that they have digital audio and a graphics display, with optional video input. This is an increasingly realistic assumption, as almost all computers sold today (PCs, Macs, Unix workstations) have graphics displays, and most have telephone-quality audio or can have it added cheaply (for about $100).

Jupiter embodies more assumptions about the capabilities of the underlying network as well. In addition to demanding more bandwidth than the Internet can currently provide, we assume that a "multicast" capability [2] is available, in which individual packets can be received by many hosts without duplicating the packets on any link and without the sender knowing who any of the recipients are; we use this, for example, to handle real-time audio and video traffic efficiently.

Audio

Each room in Jupiter has a distinct audio channel associated with it. Users hear all the sounds from the room they're in and contribute to these sounds with their speech. When they move to a new room, their audio is automatically switched to the channel for the new room. Because the channel management is performed by MOO code inside the database, we can even arrange for the sounds from "nearby" rooms to filter through at a lower volume, allowing users to notice when possibly-interesting interactions are happening "right outside their door."

The audio is roughly telephone-quality and is transmitted over the local network. We are currently using the audio devices built into our Sun SPARCstations, but any equivalent audio hardware (such as the current flock of PC sound boards) would do.

Window-Based User Interfaces

The next addition to the standard MUD model is support for graphical user interfaces under the control of MOO programs. This allows MUD objects to interact with users via windows that appear on their screens. For example, opening a map object might display a diagram of the layout of the local rooms, allowing the user to move between them. Window-based interfaces can also be used for various *shared* applications, such as text or graphics editors that appear on more than one user's screen at once.

Our protocol for communicating between the server and client programs uses high-level descriptions of the interface elements. This allows us to move as much code as possible into the clients from the server, which reduces load and makes it easier to write new window-based MUD objects. It also reduces network bandwidth since the communication is in terms of high-level interface element events instead of low-level mouse and keyboard events. Finally, it allows for a great deal of interoperability, since the details of each user's local window system are hidden from the MOO program controlling the windows; thus, for example, one user of a shared window application might be using a PC running Windows while another uses a Unix workstation running X.

Just as current MUDs can be extended with new text-based objects, Jupiter can be extended to support new window-based objects. This flexibility is important; it allows for easy enhancement of the system, either by the original implementors or, indeed, by the individual users.

Video

While the addition of audio to MUDs provides a dramatic and obvious improvement in communication, the effect of adding real-time video is more subtle. It can enrich the perceived quality of computer-mediated conversation, as gestures, facial expressions, and physical surroundings are transmitted. Also, it makes it possible to monitor remote locations; for example, one can have a sense of whether or not someone else has been "around" recently, or if someone is currently using the telephone or holding a face-to-face meeting. Finally, video can allow users to participate in remote meetings where overhead slides or other visual props are used in presentations.

In Jupiter, we intend to make it easy for users to view the output from any cameras associated either with other users in the same room or, indeed, with the room itself. The latter case comes up, for example, when Jupiter is used to attend meetings taking place in real rooms in the laboratory.

At present, we are using Sun VideoPix boards and small attached cameras. We have software-only video compression algorithms that allow any workstation to display the resulting video, even if it is not equipped with a camera. The resulting video is half-size grayscale at about 5 frames/second.

This video facility is not enough to support full video conferencing with audio/video lip-sync. However, it is sufficient to provide a sense of the activities of other users and for attending lectures via Jupiter; in the latter case, one only needs to see the speaker's slides and hear the audio to get most of the benefit of a talk.

Casual Interaction

One of our goals in building Jupiter is to support casual interactions between its participants and to allow them to participate in the casual interactions taking place in the real environment of the laboratory. The use of audio is clearly important for this, because casual interactions must be essentially effortless, and speaking is easier than typing. We plan to provide other kinds of support, as well.

Informal meeting areas are one example. In addition to the meeting rooms and offices, we envision areas that encourage casual interaction. The presence of a user in such a room would be an implicit invitation for conversation from others. One might keep casual use objects there, such as newspapers (automatically generated from wire feeds) and games. It is important that users be able to use these objects together, as this encourages conversation.

We also want to provide facilities to encourage chance encounters. One example is to use computer-generated sounds to notify users of the arrival or passage of other users. These electronic "footsteps" would remind a user to check to see who else was in the area. Another example is to mix in a small amount of audio from adjacent rooms, if their doors are open. These would provide the snippets of speech that often draw newcomers into a conversation.

Finally, we intend to equip the casual meeting areas of our laboratory building (such as coffee alcoves and lounges) with audio and video devices. These should allow Jupiter users to "happen upon" people in those areas and then to hold informal conversations with them.

Telecommuting

Many people currently spend some portion of their average week working from home, or *telecommuting*. In our experience, such people are effectively absent from the social milieu of the workplace, only “visible” in the form of occasional electronic mail messages, if that. We are interested in exploring that possibility that Jupiter can support a much more effective form of telecommuting, in which even remote workers can be active social participants.

Jupiter’s relatively modest network bandwidth requirements, coupled with recent advances in telephone connection quality, should allow workers to use Jupiter almost as effectively from home as from work.

ISDN telephone service is slowly becoming available from local phone companies thoughout the world, and provides two 64-kilobit/sec full-duplex channels per user at affordable prices. Jupiter would have to fit audio, windows, and varying amounts of video over such a connection. Audio can be compressed to as little as 13kb/sec, using techniques such as linear predictive coding, without major losses in quality. Our window-system protocol takes much less than half a channel because the interctions are at a high level of abstraction. This leaves one channel to use for video.

Our experimental video procotols can send 5 frames/sec of grayscale video using 128kb/sec of bandwidth. Using only 64kb/sec, the quality is lower but still tolerable, especially for largely static scenes such as workers in their offices. For a talk, we can send single frames of each slide, which easily fit into a single 64kb/sec channel.

The goal of telecommuting is to allow a worker to accomplish as much from home as they can by physically going to work. While Jupiter is not a full replacement for being there, it provides a good substitute for many structured activities, such as meetings. Beyond that, its value as a replacement for travelling to work will depend upon the extent to which the goals of casual interaction are met.

4 Project Status

Both Astro-VR and Jupiter are currently running, though neither is yet ready for new users. All of the technologies mentioned in the paper exist in fully-functional form; all that remains is to tie them together and then start learning from our users.

We expect that both systems will be in daily use by their intended communities by the beginning of the summer of 1993; we should be able to start saying something about our experience with the system at that time.

References

[1] Curtis, Pavel, “Mudding: Social Phenomena in Text-Based Virtual Realities,” in the Proceedings of the 1992 Conference on Directions and Implications of Advanced Computing, Berkeley, May 1992. Also available as Xerox PARC technical report CSL-92-4.

[2] Deering, Stephen E. and David R. Cheriton, “Multicast Routing in Datagram Internetworks and Extended LANs,” ACM Transactions on Computer Systems, vol. 8, no. 2, May 1990.