MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Building Multi-User Interactive Multimedia Environments at MERL

David B. Anderson, John W. Barrus, John H. Howard, Charles Rich, Chia Shen, Richard C. Waters

TR95-17 November 1995

Abstract

Building multi-user interactive multimedia environments at Mitsubishi Electric Research Laboratories (MERL) is a highly interdisciplinary activity, which involves the efforts of more than a dozen members of the laboratory. This report describes this research at three levels. At the bottom, supporting everything, are high-speed networks. On top of networks, we have built a piece of m̈iddlewarec̈alled Spline. Finally, Diamond Park is an experimental environment we are building using Spline.

IEEE Multimedia, Vol. 2, No. 4, Winter 1995, pp. 77-82

Publication History:–

1. First printing, TR-95-17, November 1995

Building multi-user interactive multimedia environments at Mitsubishi Electric Research Laboratories (MERL) is a highly interdisciplinary activity, which involves the efforts of more than a dozen members of the laboratory. As will be evident in the descriptions below, our prototype systems involve expertise in computer graphics, animation, networking, artificial intelligence, human-computer interaction, computer vision, spoken language understanding, graphic design, learning theory, and drama.

The relationship between the three areas of work at MERL described in this report is a simple three-layer tier. At the bottom, supporting everything, are high-speed networks. On top of networks, we have built a piece of "middleware" called Spline. Finally, Diamond Park is an experimental environment we are building using Spline.

Collaborative learning, work, and play

In today's society, learning, work, and play are not separate, solitary pursuits, but collaborative activities interwoven into each other and into the social life of communities. The goal of our research on multimedia environments is to use information technology to foster these kinds of activities.

For example, we envisage group learning environments where people learn from each other and teachers and by interacting with computer simulations. We envisage collaborative work environments where people at different locations interact with each other and shared, computer-simulated artifacts to design a machine, plan a large-scale disaster relief effort, or diagnose a complex equipment failure. We envisage online play environments where people participate in distributed games, historical dramas, or create virtual microcosms for each other's entertainment. Two major influences on our vision have been the popularity of so-called MUD's (Multi-User Dungeons) and the effectiveness of the US Department of Defense's Distributed Interactive Simulation (DIS) program.

Key technical features

The environments we are interested in building share a number of key technical features:

- *Multiple users.* At least several and possibly hundreds or thousands of geographically dispersed users interact simultaneously in real time.

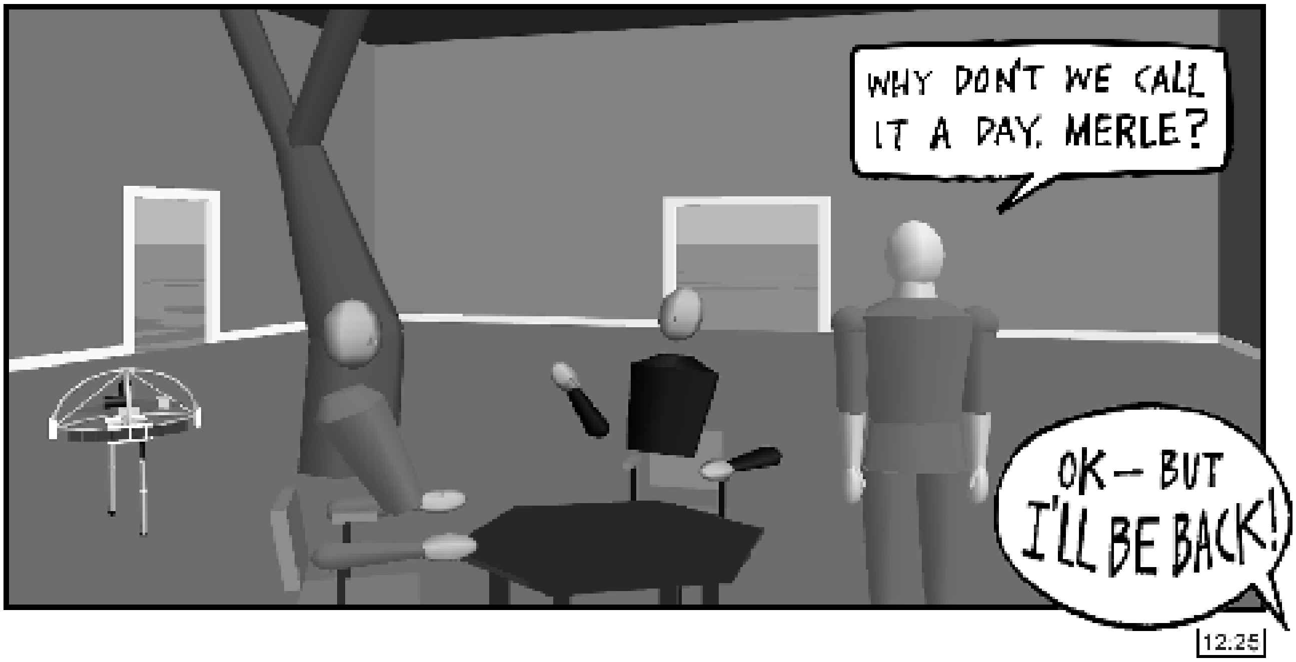

Figure 1: Illustration from the first environment we built.

- *Spoken interaction.* Users communicate with each other using natural spoken language.
- *Computer simulations.* In addition to interacting with each other, users interact with computer simulations which range from the very simple (e.g., a revolving door) to the very complex (e.g., a human-like robot).
- *Immersion.* Users experience a 3D graphical world with localized stereo sound and physical feedback.
- *Run-time modifiability.* Users can make temporary and permanent modifications and extensions to the environment while it is running, so that the content of an environment can grow in proportion to the talent of the user community (not just the system implementors),

The first environment we built with these features is described in companion articles in IEEE MultiMedia and Computer (see Figure 1). From this experience we learned that in order to experiment with different kinds of applications for these kinds of environments and with real users, we needed a robust and efficient platform upon which to build. Richard Waters and David Anderson, assisted by John Barrus, Joe Marks, and Michael Casey, have therefore developed a system called Spline (Scalable PLatform for INteractive Environments) that supports all of the key technical features listed above.

Scalable platform for interactive environments

From the point of view of an application programmer, Spline presents a simple conceptual model. Individual Spline processes implementing, for example, a user interface or a computer simulation, interact with each other by making changes in a shared, object-oriented world model. The world model contains information about everything in a virtual world—where objects are, what they look like, what sounds they are making, and so on.

The internal structure of a Spline process is shown in Figure 2. For efficiency and scalability, Spline replicates the world model, maintaining approximately equal local copies for each process. This is the same basic approach taken by DIS. Spline improves over DIS, however, in three important ways. First, the world model is broken into separately managed "locales". Second, new kinds of objects can be introduced into the world model at run time. Third, ownership of objects can be transferred between processes.

The inter-process communication module in Figure 2 uses multicast messages to describe to remote processes the world model changes made by the local process and to update the local world model based on messages from remote processes. The application support module contains tools to facilitate writing applications, such as routines for making one object follow another and facilities for playing back recorded sounds and motions.

Figure 3 shows a Spline configuration that could be used to support a human user. The figure shows three Spline processes. (It is likely, but not necessary, that all three of these processes run on a single machine. If they do, there is the potential for considerable sharing as indicated by the dashed box in the figure.)

The visual rendering process on the left creates images for the user to see from the appropriate eye point. The audio rendering process on the right creates the appropriate stereo sound for the user to hear and transfers the user's digitized voice signal into the world model. Both the visual and audio rendering modules are provided as part of Spline.

The user interface process in the middle of the figure will depend on the particular environment being built. For example, this process might receive information from a camera looking at the user or provide force feedback to the user through a hand-held device.

Spline currently runs on SGI and (without graphics) HP workstations

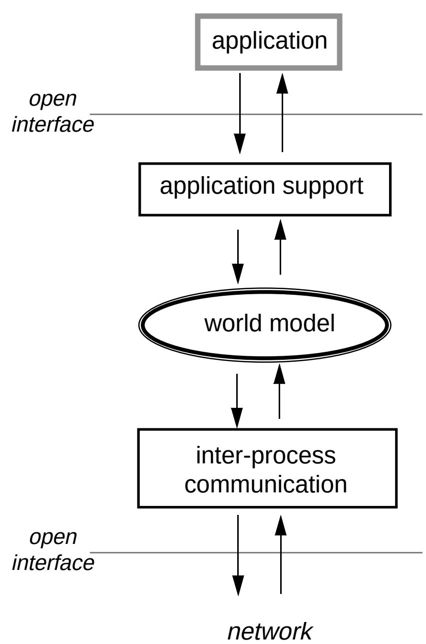

Figure 2: Internal structure of a Spline process.

and is being used in-house for the construction of the Diamond Park environment described below. We plan to port Spline to PC's in the coming year. Notice in Figure 2 that the application and network interfaces have both been defined as open interfaces to encourage the growth of a community of users.

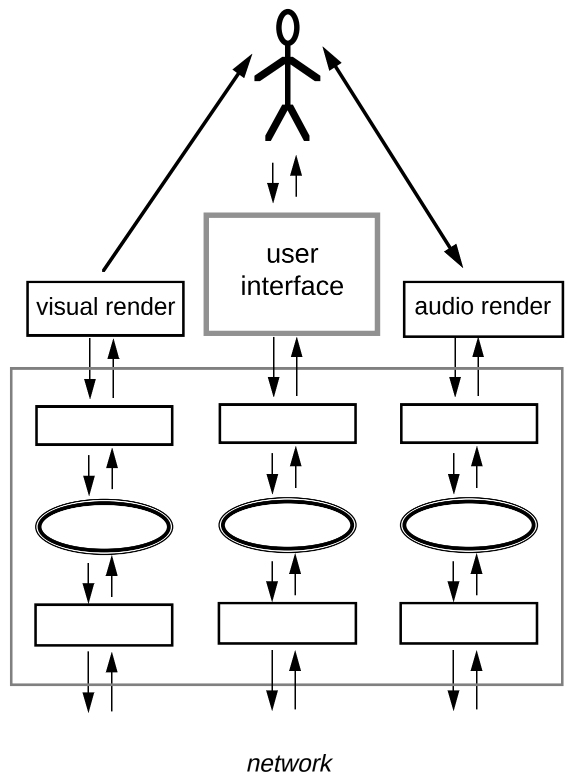

Figure 3: Spline configuration to support a human user.

Network requirements

The current version of Spline runs on Ethernet using Internet Protocol (IP), which does not provide all of the network functionality we will ultimately require. Our long-term requirements for sophisticated, high-speed network technology include:

- *Bandwidth.* The network connecting Spline processes needs to support continuous transmission of audio (32 kbps and up) and video (1 Mbps and up), as well as burst transfers of large amounts of graphical data.
- *Scalability.* The network must be scalable and expandable to accomodate hundreds or thousands of users.
- *Heterogeneity.* Communication between Spline processes includes data of greatly varying sizes and time criticality including: high-bandwith real-time data streams (e.g., live speech), frequent small data packets (e.g., object position updates), and occasional very large data transfers (e.g., graphical models).
- *Multicast.* In Spline's use of multicast, the number of multicast addresses can be much larger than the number of users and a particular node typically selects only a small subset of incoming messages.
- *Low-overhead demultiplexing.* Since a Spline process typically receives large numbers of short messages with only a small amount of computation to be done for each, the network interface overhead for message demultiplexing needs to be very low.

ATM research

At MERL, we are investigating network issues in Asynchronous Transfer Mode (ATM) as the basis for supporting Spline-like systems. While continuing to support Spline on current networks, we also plan to port it to ATM as soon as feasible. Our current ATM research includes work on the network interface, traffic management and control, and switch design.

Randy Osborne is studying changes in network interfaces to support low-latency communication, including how to avoid overhead in demultiplexing messages and how to give user processes more control over message buffering. His "hybrid deposit" architecture uses both sender and destination information to demultiplex messages directly to where they are needed. Some of these ideas are incorporated in a novel ATM interface chip under design at MERL. The design gives applications direct, protected access to the network to achieve both high bandwidth and low latency, while containing hooks for achieving hybrid deposit functions.

Qin Zheng and Chia Shen are studying the real-time channel approach to congestion control in ATM networks and have proposed a new message transmission scheme based on the concept of earliest-deadline-first scheduling. The achievable quality of service with this scheme would make ATM networks capable of supporting real-time services which have been promised but not yet successfully realized. In collaboration with Harrick Vin and Pawan Goyal at the University of Texas at Austin, Shen is also investigating using credit-based flow control for video applications.

John Howard has been involved in the specification of a flow control scheme called Quantum Flow Control (QFC), which we believe is necessary and effective in supporting bursty ATM LAN traffic where application bandwidth requirements are difficult to predict in advance. QFC supports instantaneous access to unused network bandwidth with cell loss rates no greater than the error rates in the network links.

Hugh Lauer and Chia Shen have developed a general-purpose queuing architecture for an ATM switch, which supports multiple classes of traffic in the same network, each with its own scheduling and dispatching policies and algorithms, including hard real-time guarantees, continuous media, and very rapid response.

Finally, due in part to our experience with Spline, we have recently come to an increased appreciation of the importance of multicast facilities for distributed multimedia applications. Unfortunately, since ATM is by definition connection based, and since multicast has traditionally been supported using connectionless IP routers, how best to support multicast over ATM is still an open issue. We are just beginning research in this area, looking both at methods that enable the ATM network layer to directly support multicast and multicast implementation schemes in the network interface and/or switches.

Diamond Park

Our main purpose in constructing Spline is to facilitate experimentation by a variety of researchers at MERL and elsewhere into different aspects of our broad vision of multi-user interactive multimedia environments. At MERL, a team of artists and computer scientists led by John Barrus has developed

Figure 4: A panorama of Diamond Park showing its major buildings and sites connected by a network of bicycling paths. The graphic design and 3D modelling of Diamond Park was done by Ilene Sterns and Stephan McKeown.

an environment called Diamond Park, which incorporates several different experiments in multi-user interaction.

Most current 3D animated environments are small, single-use spaces with little opportunity for exploration. Diamond Park, on the other hand, is an expansive, multi-faceted environment (see Figures 4 through 6). The buildings in the park are reminiscent of a World's Fair, with architectures evocative of various historical periods and cultures, and are spread out over

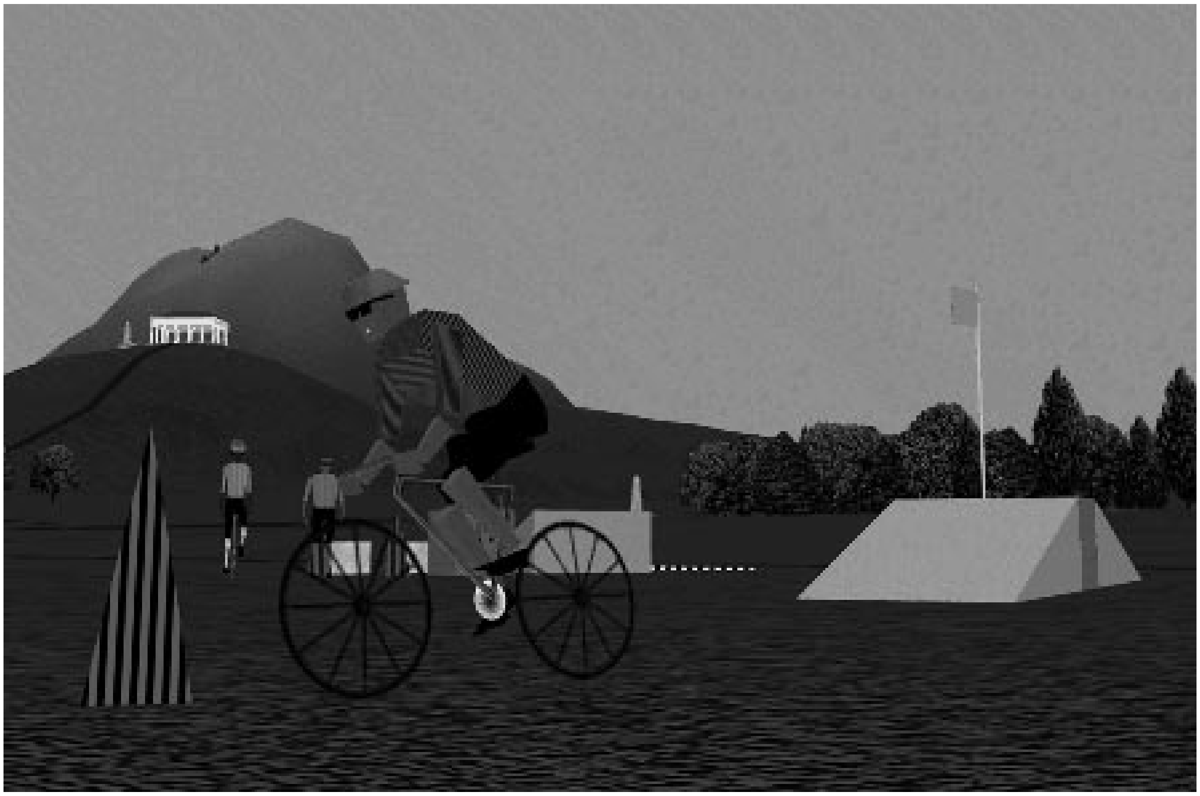

Figure 5: Diamond Park visitors take turns setting up pylons and bicycling through an obstacle course set up on one of the lawns in the park.

the grounds to encourage exploration. The buildings may be used as exhibit spaces or as venues for specific activities. For instance, the Greenhouse will provide a space for collaborative landscape design and the Museum was designed to house rotating exhibits of art work and kinetic sculpture from real museums.

Visitors participate in park activities using a variety of input devices, including computer-controlled stationary bicycles and trackballs. The activities in the park take advantage of Spline's capabilities to allow geographically dispersed users to speak, hear, and interact with each other in real time. Diamond Park is unique among multi-user environments in placing high-quality sound at the core of the users' experience.

Diamond Park has a permanent population of computer-simulated robots

Figure 6: Inside the Diamond Park Velodrome, visitors can race on a 200m Olympic-specification bicycle track. A robot race official keeps track of lap times as the red and blue racers speed by.

designed by Charles Rich. For example, friendly robots greet visitors as soon they arrive in the Orientation Center. The Park's robotic staff appear as animated 3D figures that use simple speech and gestures to interact with human visitors.

A novel activity in Diamond Park is bicycling. Users seated on computer-controlled stationary bicycles located in different cities can participate in team riding, exercise, and racing. There are miles of paths in the park available for exploration. The 200 meter Velodrome track (see Figure 6), designed to Olympic specifications, is well-suited for both training and competition. Prof. Jessica Hodgins at the Georgia Institute of Technology is collaborating with MERL to develop real-time, physically-based dynamic models of bicycles and riders to provide Diamond Park with robotic bicyclists who can participate with human visitors in races and other activities.

Marilyn Walker is converting one of the vacant buildings in Diamond Park into a virtual theater, called VIVA (Virtual Interaction with Virtual Actors), where human actors can perform roles and interact with computer-simulated actors in a play. Walker's research goals are to support conversational interaction and improvisation by both human and simulated actors. In collaboration with Janet Cahn of the MIT Media Laboratory, Walker is learning how to control the intonation of computer-generated speech to get effects such as humor, irony, and so on.

Carol Strohecker is working on two projects, not yet implemented in Spline, that involve software "construction kits" to be used by individuals or collaborative groups.

One project focuses on the construction of virtual objects to explore concepts of movement, scale, and spatial relationships. Users can construct dinosaur skeletons and mobiles for experiments with balance, and polyhedra for explorations in topology. The mobiles in changed scale can become items of jewelry, and the polyhedra in changed scale can become landscapes on which the dinosaurs move and teeter.

A second project focuses on the construction of stories by drawing from the theatrical device of the Greek chorus. In ancient Greek plays, a chorus mediates between the audience and the action. Usually, chorus members are not directly part of the action, but reflect or modulate it as the play unfolds. Chorus members may address the characters, the audience, or each other as they clarify, magnify, subdue, transpose, re-tell, interpret, or give perspective to the narrative action. An initial prototype works with the story of the 1955 arrest of Rosa Parks and the first day of the Montgomery bus boycott, key events in the American Civil Rights Movement. In response to user queries, computer-simulated chorus members comment on this story from the perspectives of the past, the present (1955), and the future (1995). Users can add their own perspectives as chorus members of the future.

Diamond Park is an expandable environment for our explorations in distributed multimedia communities. We plan to continue to develop the Diamond Park community and activities for its visitors.

Further reading

- C. Rich et al. "An animated on-line community with artificial agents." *IEEE MultiMedia*, 1(4):32–42, Winter 1994.
- C. Rich et al. "Demonstration of an interactive multimedia environment." *IEEE Computer*, 27(12):15–22, December 1994.
- R. Osborne. "A hybrid deposit model for low overhead communication in high speed LANs." *1994 IFIP Fourth International Workshop on Protocols for High-Speed Networks*, Vancouver, Canada, August 1994.
- Q. Zheng, K.G. Shin, C. Shen. "Real-time communication in ATM network." *19th Annual Local Computer Network Conference*, Minneapolis, MN, October 1994.

The authors can be contacted by sending email to anderson@merl.com or visiting http://www.merl.com.