A Software System for Cost-Effective Internet Delivery of Synchronous Distance Education

J. Mark Pullen
George Mason University
Fairfax, VA 22030
mpullen@gmu.edu

keywords: synchronous distance education, multimedia networking

ABSTRACT

The Internet is widely used for asynchronous distance education. Synchronous distance education is harder to deliver; it poses challenges in technology, economy, and organization in addition to the challenge of distance. We have developed an Internet-based system for synchronous distance education, where the instructor's voice, graphics, dynamic annotations, and optional video are streamed to the student at home or office and also recorded for server-based delayed online delivery. Network EducationWare (NEW) will deliver a total of twenty full-length courses this year at George Mason University. NEW has been assembled from open source Internet multicasting software and is made available free to the academic community in the same spirit. We describe issues faced in scaling up NEW to meet the demands of multiple courses and large numbers of students.

INTRODUCTION

The Internet is widely used for asynchronous distance education, consisting mostly of remote access to Web-based course materials. Certainly, this use of remote access saves a lot of travel to the library and also saves many trees from being sacrificed for paper. In cases where it is used to provide a full course, rather than simply augmenting an instructor-taught course, it provides a faster means of delivery for that earlier form of distance education, the correspondence course. However, over many generations of schooling, academia has arrived at the collective conclusion that an instructor, serving as mentor and interpreter of course materials, can make more effective use of students' time for learning. The Internet does not invalidate this conclusion. On the contrary, just as the Internet has made the correspondence course more accessible and flexible, synchronous distance education creates an ability to deliver instructor-presented classes to students who find that distance makes classroom attendance too expensive or simply impractical.

Until recently, synchronous distance education was delivered primarily by television. Today the Internet, combined with the personal computer, offers a means of electronic delivery that can be more effective than television, both educationally and in terms of delivery cost. While many faculty members are conditioned by "television teaching" to assume that video plus audio is the most effective means of delivery, our experience shows that in many cases audiographics (the combination of audio, prepared graphics, and dynamic graphic annotation) is more effective than video [1]. Moreover, audiographics requires about one-fourth as much network capacity as video and therefore reduces the cost of synchronous course delivery [2].

While synchronous Internet delivery takes advantage of the near-ubiquity of inexpensive Internet connections, it faces the problem that the Internet today does not support a guaranteed quality of service that can ensure a quality delivery of audiographics. At George Mason University (GMU), we have a growing synchronous distance education program. We are located in a high-technology urban area with excellent Internet service and legendary levels of vehicle traffic congestion. We have found that network traffic congestion also reaches extreme levels in some cases. While many of our students are able to connect via their Internet Service Provider, we have found the availability of a campus modem bank essential to guarantee that every student can obtain a quality connection to class, thus avoiding both congested roads and congested networks.

Two other critical challenges to Internet distance education are economic: providing adequate computer hardware and software at low cost and operating an educational program with a distance component at an affordable cost. The remainder of this paper describes the solution we have developed at GMU to meet these challenges.

PARAMETERS OF THE SOLUTION

The author offered experimental synchronous Internet access to courses at GMU for several years, starting with various cobbled-together collections of multimedia network software and proceeding to an early commercial product that did not succeed in a business sense [3].

During this process, a formula for success was determined experimentally:

- Quality software is essential
- Software must function over low-capacity Internet connections
- The entire system, from teaching to online delivery, must be designed to be simple and robust, functioning in almost any Internet environment
- The system must make online teaching and learning easy

In 2001 our laboratory took on the challenge of creating a solution that meets this challenge. We have created Network EducationWare (NEW) primarily from open source software that is available with no license fee to all. The tools we started with were created for use with the Internet multicasting, where one station sends an identical message to many others. This approach is sometimes called peer-to-peer because all computers have identical ability to send to each other. It is offers a simple model for scaling to nearly unlimited numbers of participants and has attracted talented software authors who have made some important tools available free (see the next section for a description of these). However, the multicast model by itself is not an economic success; in fact, multicast service is not a common offering in the commercial Internet. We have made important adaptations to this simple model, both in network technology and in course management.

The adaptations for network use were to emulate the multicasting model with a system of servers, and to encapsulate the User Datagram Protocol (UDP) messages used in multicasting within the more common Transmission Control Protocol (TCP) for better acceptance. The server architecture was made necessary by the need to send the same program to a group of students (multicasting) while operating over Internet connections that do not provide such a service. Therefore, we created a software component called the Transport Layer Multicaster (TLM) that accepts messages from one user and sends them to all others in the same group. The TLM is programmed to accept the UDP messages that are sent by the multicast audiographic tools, but it also has a feature whereby a client program in the user's computer collects the multicast messages and passes them through a "tunnel" to the TLM. In other words, the multicast messages are encapsulated in a more common format (called unicast) that the network will support. By network orthodoxy this tunnel also should consist of UDP messages; however, we have found that many students connect from behind a corporate security "firewall" or through a cable provider's Network Address translation (NAT) system. As neither of these situations is conducive to using UDP, we encapsulated the multicast messages in a TCP stream. While only about eighty percent efficient, this provides a general solution to NATs and makes it more likely that a connection can be arranged from behind the firewall.

Managing courses that cater to synchronous Internet students has provided an entirely different challenge. The first realization faced was that simply providing a lecture to a distant student does not provide an entire educational solution. As set forth eloquently in [4], a complete distance education system involves many functions of the university that we often take for granted, such as registration, a bookstore, and advising. Our solution to this was to realize that, because we operate in a major urban area and have many working students, we can best serve regional students who come to campus rarely when they need services. This notion that students can and should partake of campus services is further reflected in our approach to distance teaching, which we call "simulteaching." Simply stated, we use the same software to teach to a group of students in the classroom and another group over the network at the same time. The students can switch back and forth from week to week. Many of the most frequent users of the recorded classes are in fact in-person students who have missed a class. Figure 1 shows the system arrangement we use with NEW. Having the classroom projector driven by the same software used by online students has proved to build confidence in that the whole simulteaching audience receive the same presentation.

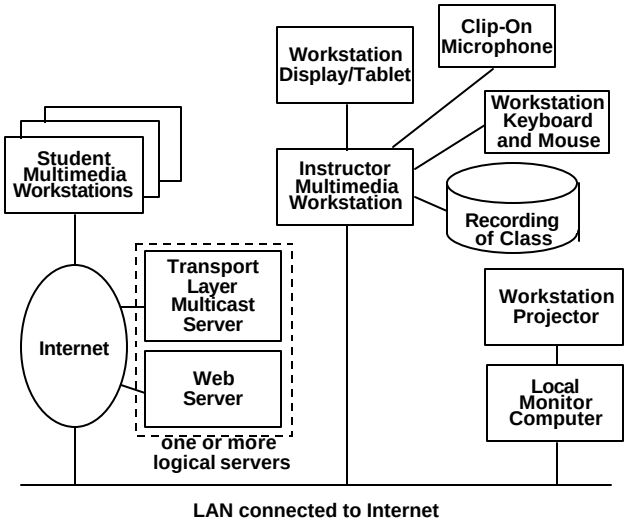

Figure 1. Simulteaching System

SYSTEM BUILDING BLOCKS

The NEW system is constructed from thirteen functional software components. These all have the common characteristic that they can run on multiple computing platforms (at least Windows, Unix, Linux, and Macintosh), although the first release is configured only for Windows. Beyond this, each component is

specialized to fill a particular function. Figure 1 shows how the blocks fit together. Each block communicates with the other via the Internet Protocol (IP), with the result that no two blocks need to be run on the same computer, although they generally are grouped on one server computer and many user computers as shown in Figure 2. We hasten to add here that, although GMU has developed over half of the components shown, our fraction of the total software development effort is much smaller; in fact our work is surely less that ten percent of the total effort.

The sources of software not created by GMU are available on our website http://netlab.gmu.edu/NEW. All components are available under open source license for academic purposes. License details vary, but in general any of them may be modified and redistributed free of license fees, so long as the original copyright is propagated and the application is academic.

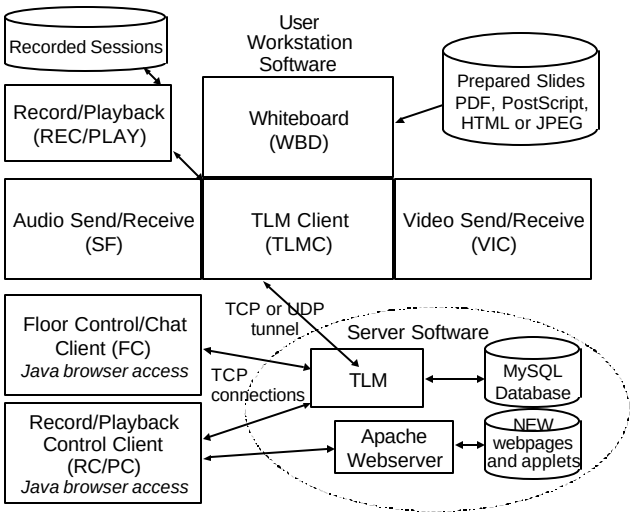

Figure 2. System Components and Interconnection

Audio (SF): This component is arguably the most important in the system, because it is essential to the students' experience and also because conveying voice with good quality over the Internet presents a big challenge. We chose open source software developed in C by John Walker, called Speak Freely, which is capable of passing good voice quality over the Internet, even under conditions of limited congestion. This is a multicasting, peer-to-peer application that both captures and reproduces digital audio, using a standard sound interface. We have added a new user interface to SF at GMU, as shown in figure 3.

Video (VIC): This component sends and receives digital video, using a standard video adapter. It performs several standard compression techniques and will handle a range of sending rates from less than one per second to full-motion video. We provide video as an option from our online classroom, usable by students with Internet connections at 200 kilobits a second and up. A typical delivery rate is two frames of 320 by 240 pixels per second, with transmit rate limited to 100 kilobits per second. VIC was developed in C++ and TCL/TK at Lawrence Berkeley Laboratory, California and enhanced for Windows use by University College, London, UK.

Figure 3: SF User Interface

Whiteboard (WBD): This component provides a shared graphic presentation medium. It will display a pre-composed graphic prepared in any of the open formats HTML, PostScript, JPEG, or Adobe Portable Document Format (PDF). The last of these is particularly valuable for creating lecture slides, as any document that can be printed on a Windows system can be transformed to PDF. The WBD user can annotate the shared space with lines, arrows, rectangles, ellipses, and text in any color, a very useful feature for maintaining the attention of the visual learner. It was developed in C and TCL/TK by Julian Highfield at Loughborough University, U.K., and is distributed by University College, London, U.K. We added HTML and JPEG capabilities to WBD at GMU. WBD may be scaled to cover a large or small screen area. A very small WBD is shown in figure 4.

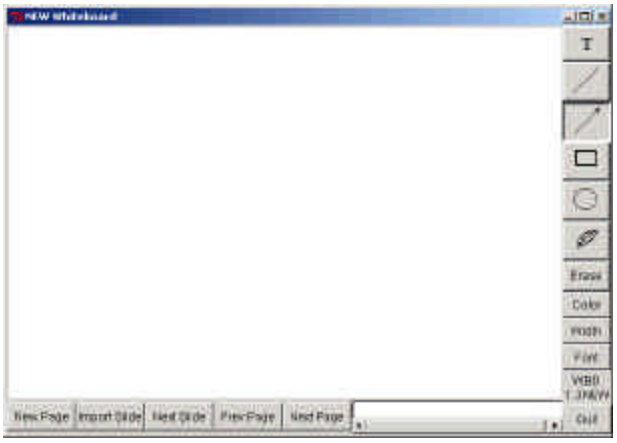

Figure 4. WBD User Interface

Floor Control (FC): This component shows the participants in the session, controls access to the virtual "floor," provides for text questions to the instructor and text chat among the participants, and accepts URLs from the floorholder for browser launch at all peer participants. Participants may be treated as peers, such that any requester obtains the floor; this is good for seminars. Alternately, a two-tier model of floor control is available, based on privileges coded in the database: instructors can take the floor at any time, change the floor grant status for students among "always," "ask me," and "never," and grant student floor requests made in the "ask me" status. FC was developed in Java at GMU and works with the TLM server. The FC user interface for an instructor is shown in figure 5.

Figure 5. FC User Interface

Transport Layer Multicaster Client (TLMC): This component collects audio, whiteboard, and optional video messages from the multimedia tools, encapsulates them into a UDP or TCP tunnel, and sends them to the TLM. It is packaged with the audio client, because no effective use of the system can be made without both of these critical modules. It also tests for adequate network capacity and launches the Floor Control client, Record or Playback control, Recorder or Player, and the multimedia peer-to-peer applications (SF, WBD, and VIC). It has the additional option to launch a Web browser or other Web-enabled software, upon receiving a request message entered through the Floor Control from any peer. TLMC was developed in C at GMU.

Transport Layer Multicaster (TLM): This component implements the multicast paradigm over the general Internet among a group of participating workstations. It provides access control using passwords and optionally using network addresses. Both TCP and UDP tunnels from TLMCs are supported. It implements floor control on the audio, whiteboard, and video streams, and login authentication via the MySQL database. It was developed in Java at GMU. We are able to support ten courses, with up to three simultaneous sessions, in a high-end workstation (Sun Ultra 60) that also provides all of our laboratory's Internet services, including the webserver and database for NEW (described below) as well a twenty-user playback server (also described below).

Record (REC): This component captures the timestamped stream of messages as seen at a particular user's workstation and records them to disk. They may then be played from that disk or transferred to any other computer for playback. REC was developed in Java at GMU.

Record Control (RC): This component provides a VCR-like start/stop/pause interface for REC with a display of status and a record counter. It can run on the same computer as REC or on another Internet-connected computer. RC was developed in Java at GMU.

Playback (PLAY): This component regenerates the original stream of messages from a timestamped recording. It can function either as a standalone player that allows a workstation to play the recording through the multimedia peers (TLMC, SF, WBD, VIC) or as a multi-user server that streams the playback to the TLMC over the Internet. PLAY was developed in Java at GMU.

Playback Control (PC): This component provides a VCR-like start/stop/pause/prev/next interface for PLAY with a record counter. The ability to skip to the next slide backward or forward has proved very popular with users. PC can run on the same computer as PLAY or on another Internet-connected computer. It was developed in Java at GMU. The user interface of the applet version of PC is shown in figure 6. There is also a Java application for stand-alone use.

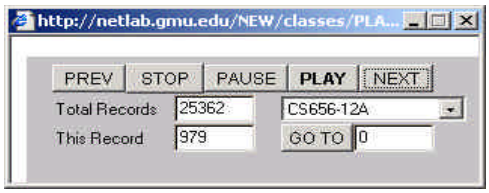

Figure 6. PC User Interface

NEW Course Management System Webpages: NEW was designed for the Web, with student access via webpage and Java applets for FC and PC plus access to lecture slides. The web interface is fully integrated with the MySQL database. Student login is accomplished through the browser. Instructors have additional access to the database for status of slide files, recordings, system usage, and adding/modifying student accounts. The course management system was developed in HTML and PHP at GMU.

Database (MySQL): This is a high quality, multiplatform, open source, network-accessible relational database that implements the Structured Query Language (SQL). It was developed in C and C++ by the Swedish company MySQL AB. It is used for authentication and usage logging, and runs just as it came in binary from the MySQL website.

Webserver (Apache): This is a high quality, multiplatform, open source webserver that is widely used in the Internet. It was developed in C by Apache Digital Corporation. Because the student interface is accessed through the NEW course management webpages with FC and PC in the form of Java applets, a webserver is a necessary component of NEW. It runs just as it came in binary from the Apache website.

SCALING ISSUES

One of the lessons of the "dot com fiasco" of recent years has been that promising technologies, however well implemented, cannot by themselves generate a successful enterprise. The technologies must be designed to fit the users' needs very well or they will not be accepted. Further, the system must be designed for effective scale-up or it will not be able to grow. The number of synchronous distance education courses at GMU has doubled each year in the past three. In academic year 2000/2001, the author taught one course per semester in this mode; in academic year 2002/2003 there were a total of twenty courses. While this is only the beginning of what we believe will be a continued growth curve, it has taught us several important lessons about running synchronous distance education which we will relate here.

Simplicity is essential in teaching and learning software. While early adopters will endure almost anything, in production use neither faculty nor students respond well to complex user interfaces. We have learned that each tool must have clear functions, implemented by intuitive graphic elements. For example, use of Internet audio is effectively defeated if the software volume control provided by Windows is hidden inside the control panel's "audio properties" interface. We found it necessary to add both microphone and speaker controls to the SF audio interface, and then to include an automatic gain control on the microphone. Similarly, a closed-loop test of the audio system had to be built into the interface so the user can confirm that the full audio system is working.

Online is not for everyone (but playback is). In a typical course, at most 25 percent of the students truly want Internet delivery. More may enroll if it is the only way to take a highly sought course, but those who are not committed to participating at a distance soon will squeeze into the classroom if permitted to do so. Online delivery appeals to students who have particular schedule or commuting difficulties, and see the time saved as a good exchange for whatever sense of presence is lost. In our experience, among the network students there is an even smaller fraction of students who truly prefer Internet delivery because they find it avoids the distractions of the classroom environment. There also are students who enroll for network delivery but do not connect regularly during class; instead they time-shift the course by playing the recording. However, the synchronous users have turned out to be at most twenty percent of the users of the playback system. Typically between 60 and 100 percent of students will use the playback system at some time during the semester to make up a missed class or review a lecture. We conclude that, in our environment, simulteaching with recording can be expected to attract up to 25 percent additional students and also to make the in-classroom portion of the course more attractive (most of our simultaught sections are at maxim enrollment).

Asynchronous courses are not for everyone (but, again, playback is). The availability of recorded lectures has prompted us to grow in a different direction, by offering professional education classes that use the recordings. We find that less than half of the students who enroll in these courses complete them. Follow-up generally yields the response "I just got too busy to finish it." However, we find that almost all of our part-time graduate students, online or in-classroom, are able to find time to complete their synchronous courses. The big difference seems to be the existence of a weekly lecture and assignment schedule to motivate completion. Online students who defer a lecture generally play it back within a week in order to prepare for homework or project submission. This again underscores the value to working students of simulteaching with recording.

The growing Internet culture makes synchronous teaching seems more natural. The latest generation of students has grown up with a keyboard in hand and Internet connection the norm. Another factor that lends itself to acceptance of online teaching is the pervasive use of electronic mail (email) for student assistance and mentoring. Over the last decade, email has become the most common means of communication between students and faculty at GMU, probably because its asynchronous form avoids the need to synchronize schedules between student and instructor.

However, even progressive faculty members resist new media. In the GMU School of Information Technology and Engineering we pride ourselves on progressive approaches. We are part of Virginia's newest state university and are located in a leading high-technology area, the "home of the Internet." Even so, the idea of

teaching over the Internet has caught on slowly. Simulteaching with NEW is succeeding because it adopts a paradigm the faculty already know, in that the NEW WBD interface is a great deal like an overhead projector and the other tools do not demand much attention. To get a new instructor started mostly requires heavy reminders that:

- The online student does not benefit when you point your hand at the screen- use the WBD arrow!
- The online student can't hear the questions from the back row- repeat the question!

Given such a simple environment and good support (see below), many of our faculty have become converts to simulteaching with NEW.

Online teaching requires more institutional support. The university saves money on classroom facilities for online students, but it must invest at least part of that savings in supporting the process. Reliability of the online teaching system must be very high; students who miss classes because something breaks will have little patience with the system, and faculty who lose class time dealing with technology problems will have even less. Systems must be tested thoroughly. Support personnel must be trained thoroughly, and imbued with the attitude to "fix the problem, quickly!" Most GMU faculty today do have their teaching materials computerized (usually in Microsoft PowerPoint), but the slides are likely to need some revision for the first online use; for example they may entail extensive scanned-in bitmaps that transfer very slowly over student modem lines, or they may involve fonts that are hard to read in on a projected whiteboard in the classroom. We have settled on an arrangement where graduate teaching assistants edit the slides for readability and process them from the original format to PDF files ready for the WBD. They also post the slides for students to use in preparing for class. These same teaching assistants then monitor the class and stand by the phone to answer student questions. Our experience is that a teaching assistant can handle about six courses this way and can simultaneously monitor two or three classes. The average course has an enrollment of about eight synchronous students in addition to forty in-classroom students. Thus we anticipate a need for one extra teaching assistant for every fifty synchronous students.

CONCLUSION

The payoff for all of this effort is better support for students. Even during its growing pains, our students give simulteaching using NEW a strong endorsement (average better than four out of five on semester-end evaluations). Students now have the option to attend class from home or office or delay class to a more convenient time. The university pays less in added support than these students bring in tuition and saves in classroom facilities as well. Little extra effort is required to teach the online student, although their added numbers are reflected in grading and mentoring efforts just as much as if they were physically present in class. The most telling point is that synchronous Internet course delivery opens up availability of higher education to a whole new sector of our society and correspondingly opens up a whole new market to the university. Furthermore, the student population reached, and the corresponding market, becomes even larger when asynchronous offering of recorded synchronous courses is considered. Based on these observations, and on our growing experience with synchronous simulteaching using NEW, we conclude that there is a very large potential for this mode of distance education.

REFERENCES

[1] Pullen, J.M., Synchronous Distance Education and the Internet, Internet Society Annual Conference 1998, Geneva, Switzerland, July 1998

[2] Applicability Of Internet Video In Distance Education For Engineering," *Proceedings of the IEEE Frontiers in Education Conference*, Reno, NV, October 2001

[3] Pullen, J.M. and M. Benson, "ClassWise: Synchronous Internet Desktop Education," November 1999 special multimedia CDROM issue of *IEEE Transactions on Education* Vol 42 No 4; printed abstract p 370

[4] Carswell, L., "The 'Virtual University': Toward an Internet Paradigm?" ACM SIGCSE Bulletin, September 1998

AUTHOR BIOGRAPHY

J. Mark Pullen is Professor of Computer Science at George Mason University, where he heads the Networking and Simulation Laboratory in the C^3I Center. He holds BSEE and MSEE degrees from West Virginia University, and the Doctor of Science in Computer Science from the George Washington University. He is a licensed Professional Engineer, Fellow of the IEEE, and Fellow of the ACM. Dr. Pullen teaches courses in computer networking and has active research in networking for distributed virtual simulation and networked multimedia tools for distance education. He is responsible for the Network EducationWare (NEW) open-source suite of tools for synchronous Internet distance education.