A DISTANCE LEARNING NETWORK CONTROL SYSTEM

C.N. Mckinney, Q. Tran, V. Gallegos and P. Gercken
IBM Federal Systems Company
9500 Godwin Drive Manassas, Virginia 22110

ABSTRACT

In the "Extending Classrooms to the Military Workplace" paper submitted last year, we discussed the benefits of distance learning over conventional training programs. We focused on hardware and introduced multimedia and a modular, building block architecture that supports distance learning and Computer Based Training (CBT) on one platform.

This year, we focus on implementation of a new system and we detail the software architecture. We performed a comparative analysis of several distance learning systems currently in operation and designed a new system incorporating the best features discovered during this investigation. This analysis identified a need for distance learning systems to use existing Department of Defense multimedia and networking technologies; provide capabilities for transmitting lessons to individual workstations; and provide features that allow one person to control the entire distance learning network.

This paper describes a generic framework for a distance learning network control system that allows one instructor to control the entire network operation and allows communication to receive sites over satellite, terrestrial, and Local Area Network (LAN) interfaces. The proposed control system supports two-way video, audio, and data transmissions between the broadcast and receive locations, and provides system monitoring capabilities from one central console. We discuss interoperability, open systems, and the functional requirements needed to control a distance learning classroom session. We describe basic software components of the distance learning control system: user interface, LAN control system, satellite control system, terrestrial control system, and receive site control system. We also define interface specifications and performance requirements, and define the relationship between components and subcomponents within the distance learning architecture. Finally, we give examples of how the implementation of the proposed control system can lead to reduced costs in developing, maintaining, and enhancing distance learning systems.

BIOGRAPHIES

Clifton McKinney is an Advisory Systems Engineer with the IBM Federal Systems Company (FSC), Manassas, Virginia. He now works in the Instructional Technologies New Business organization as Chief Systems Engineer. He writes proposals, develops architectures, and delivers distance learning and networked CBT solutions.

Quang M. Tran is an Advisory Software Engineer with IBM FSC in Manassas, Virginia. He was Lead Software Engineer for several training system projects at IBM FSC. Currently, he is a Software Engineer on the Close Combat Tactical Trainer contract.

Venisa Gallegos is an Associate Systems Engineer with IBM FSC in Manassas, Virginia. Her experience covers positions as a Programmer and Design Engineer on several classroom-based multimedia systems, including distance learning and networked multimedia classroom. She is now working on an independent research and development (IRAD) project to investigate video teleconferencing over a LAN.

Pam Gercken is an Associate Software Engineer with IBM FSC in Manassas, Virginia. She has been involved in several classroom-based training programs, including the interactive multimedia classroom, networked CBT systems, and distance learning systems.

A DISTANCE LEARNING NETWORK CONTROL SYSTEM

C.N. Mckinney, Q. Tran, V. Gallegos and P. Gercken
IBM Federal Systems Company
9500 Godwin Drive Manassas, Virginia 22110

INTRODUCTION

Reductions in military budgets have forced the Department of Defense (DOD) and other federal agencies to develop cost effective methods of training and recertifying employees. The existing methods for training large groups of individuals is under attack due to instructor/student travelling costs and facility overhead expenses. In many cases the government has elected to train large groups of individuals using Distance Learning Systems (DLS). This paper describes an architecture that will allow the integration of DLSs into existing Local Area Network (LAN) based computer network environments. We discuss results obtained from surveys of existing DLSs and identify the primary features that should be provided by any DLS. We also explain why many of the standards defined to support an Open System computing environment are also applicable to DLSs. Finally, we discuss the cost benefits of implementing a DLS using our proposed architecture.

This paper focuses on the design requirement for a Distance Learning (DL) control system. We will not discuss the video and audio requirements of a DLS in detail. The selection of video and audio compression and transmission equipment is dependent on the user's picture and audio quality requirements. The proposed architecture allows users to select the appropriate video and audio equipment that meets their specific requirements without impacting the software control system.

DISTANCE LEARNING OVERVIEW

Distance learning is defined as the delivery of curriculum to students beyond the four walls of a conventional classroom. Unlike video teleconferencing, which requires as a minimum two-way video and audio transmissions, distance learning normally requires one way high quality video and two way audio. Since the primary objective of distance learning is to extend the learning environment to students at remote locations, it is desirable to provide methods for instructors to test the comprehension level of students. This testing feature can be implemented as an ad-hoc question initiation and results display capability or as a full feature test capability with student results being collected and stored for future analysis purposes. In either case, the system allows instructors to measure the effectiveness of the curriculum and student comprehension. To support this requirement, DLSs must provide a method for students and instructors to interact and allow instructors to list total student responses at receive sites (ad-hoc questions) or identify individual student responses (testing). Another DL requirement is to provide methods for students to "raise their hands" and ask a question. Just like in a conventional classroom, the instructor determines when a student can talk during the lecture. Therefore, the instructor should have the capability of answering questions or disregarding/canceling help requests. Finally, the DLS should provide a method for controlling transmission equipment and alerting instructors and/or administrators when errors are detected in the transmission equipment.

The ideal DLS is a system that allows an instructor to utilize the methods and techniques currently used to teach students in the conventional classroom environment. Hence, the training effectiveness and student comprehension levels being obtained in the current environment could also be obtained in a distance learning environment. The only differences in the two environments would be the location of the students.

Comparative Analysis

Over the last three years we have been evaluating IBM and other vendor developed DLSs. Most of these DLSs were a subset of a video teleconferencing system. That is, they provided one way audio and video transmission (instructor). There was no method for students to interact with the instructor other than a voice grade telephone line. Some systems provided Student Response Units (SRU) to allow students to answer questions; these devices also had microphone systems so students could request help and talk to the instructors. Other systems offered two-way video for instructor and student return video transmissions. This system was usually a video teleconferencing system with a Multipoint Control Unit (MCU) which was used as a distance learning system. In most cases, *these systems did not provide methods for* instructors to ask questions and collect results. Although the technologies are available to support highly interactive distance learning systems, the design and implementation of these systems resulted in the following problems: too many people were needed to operate the DLSs; the DLSs were developed/integrated as stand-alone solutions; lack of modularity and growth capability; and limited transmission capabilities.

Many of the DLSs evaluated required at least three people to operate the system: instructor, broadcast site administrator and receive site administrator. In some cases another person was utilized to operate the multimedia equipment and monitor incoming calls from students. Our design will allow the entire distance learning network to be controlled by one instructor.

Advancements in LAN technologies and reductions in Personal Computer (PC) costs have led to a dramatic increase in the migration to LAN based computer systems. In addition, the migration from Host based systems to client/ server environments has been a major contributor to the increased demand for PC networks. The potential result of this new environment is that every user will have a workstation as opposed to a terminal. Many of today's workstations can support multimedia upgrades to display full motion video and receive audio by inserting additional multimedia cards. Furthermore, since these workstations have processing power, software emulated SRU devices can be developed. The conclusion is that DLSs must be capable of operating in these new LAN based environments by utilizing existing multimedia machines as opposed to being a separate solution. This new DLS system is now a subset of an existing LAN based computer/training system, which can also support simulation, CBT, procurement, logistics and other military applications.

Many of the DLSs evaluated did not provide easy methods for upgrading their product. In cases where upgrades were possible, the upgrade had to be provided by the vendor because the hardware and software were proprietary. DLS features must be provided as modular upgrades to the baseline platform. These upgrades are not proprietary, but, whenever possible, should comply with existing *and future hardware and software interface* standards. In effect, the DLS should be developed on an Open System platform. Since the DLS is now a subset of the existing computer system network, it should comply with Open System supported standards such as: Portable Operating System Interface for Computing Environment (POSIX), Structured Query Language (SQL), Government Open System Interconnect Profile (GOSIP); Transmit Control Program Internet Protocol (TCP/IP); Distributed Computing Environment (DCE); Computer-aided Acquisition and Logistic Support (CALS); and H.261/Px64 (video compression algorithm).

Most of the DLSs evaluated only provided video and audio transmission to receive sites over satellite interfaces. If interactivity was supported, it would be implemented over a terrestrial interface using 56kbps, Packet Switching Data networks (PSDN) and/or voice grade telephone lines. None of the systems evaluated provided transmissions to receive sites over LAN interfaces. With new Wide Area Network (WAN) technologies and services becoming available, such as Asynchronous Transfer Mode (ATM), Switched Multimegabit Data Service (SMDS), Broadband Integrated Service Data Network (BISDN), and Synchronous Optical Network (SONET), it will be possible in the near future to transmit compressed video and audio economically over terrestrial lines. The new DLSs must be independent of the transmission medium; thus, allowing user to take advantage of existing infrastructure networks and expand when new technologies become mature.

DLS Functional Specification

Extensive trade studies and surveys of the DLS users community were conducted to compile a list of functions that should be provided in a generic DLS. The DLS functions are categorized according to the three types of DLS participants (operators): student, instructor, and administrator. For each type of participant, there is an associated workstation that provides the required functions. Detailed task analysis were performed to ensure that the tasks to be performed by the student, instructor, and administrator were provided by the DLS.

Instructor

The instructor shall be provided with a multimedia station to deliver multimedia supported lectures and a workstation to control the DLS. The workstation shall be independent of the multimedia station; that is, the DLS shall operate independent of any multimedia authoring and delivery package. Furthermore, the DLS shall not be dependent on specific video/audio compression equipment. The instructor shall have the capability of utilizing laser discs, video cameras, text, computer graphics, digitized audio, VHS tapes, and animation to augment conventional stand-up lecture presentations. The DLS shall allow instructors to transmit high quality video, audio and network commands over satellite, terrestrial and/or LAN interfaces. Lecture material shall appear in a window on PC based student stations. Receive site return audio, video, and data shall be supported over terrestrial, satellite and/or LAN interfaces. The DLS shall provide six basic functions from the instructor workstation: system initialization, roster collection, question initiation, test initiation, student help processing, and system monitoring. Touch, mouse and keyboard entry capabilities shall be accepted by the instructor workstation.

The DLS shall provide a method of allowing instructors to assign receive sites to specific DL sessions. Configuration files shall be created in ASCII formats and shall be protected using passwords and encryption. These files shall be accessible from a data base stored on the instructor workstation or from a central data base over a LAN. The DLS receive sites shall always be able to receive an initialization command from the instructor; hence, any activities on the student workstation shall be interrupted and the DL software shall be started.

The DLS shall provide a function that allows instructors to either determine how many students are participating in a DL session or identify students on an individual basis. In the former case, the instructor shall allow students to send a signal from their input device and the system shall tabulate attendance from all participating sites. Attendance data shall be displayed in a Roster display window. In the latter case, students shall enter an identification number during roster generation. The DLS shall tabulate the attendance entries and correlate student names to ID numbers, which are retrieved from a central or local database. The DLS shall display the total number of students at receive sites, as well as the names of the students when requested.

The DLS shall provide a capability for instructors to determine how well students are grasping the material; this shall be accomplished using test and question generation functions. The question function shall allow an instructor to ask true/false or multiple choice questions at any point during the DL session (ad-hoc feature). The DLS shall provide real time display of results to the instructor and receive sites. The test function shall provide an encrypted, password protected text editor for creating test questions. Test questions and answers shall be capable of being accessed from a central or remote database. At the conclusion of a test, the instructor shall have the ability to view student responses on an individual basis and perform statistical analysis on the results.

Student interaction with instructors shall be supported through the use of a hardware or software emulated SRUs. The emulated unit, which is used to emulate an input device on a workstation, shall support all the functions provided by the hardware device. Keypad types used at receive sites shall be transparent to the instructor. SRUs shall provide help keys to allow students to signal the instructor when help is needed. The DLS shall notify the instructor of the requested help by displaying a message box on the instructor station identifying the receive site location. When the instructor acknowledges the help request, the site name shall be inserted in the Help request list box. Next, the instructor shall have the option of canceling the help request or opening up a dialogue by enabling the receive site voice equipment. While a receive site is pending a response to its help requests, the DLS shall disable all other SRU help request keys at that receive site.

The DLS shall monitor the status of all receive sites configured into the DL session. In most instances, system monitoring shall be automatic with message transmissions to the instructor when receive sites become inactive or when equipment failures are detected. The DLS shall read a system initialization table to determine the interval cycle for monitoring receive sites. This table shall be encrypted and password protected, and shall be accessed from a local or remote database. When a receive site is identified as inactive, the instructor shall be notified and the Roster window shall be updated to identify the site as inactive. The DLS shall notify the instructor when the site becomes active and update the Roster window to indicate the site is on-line.

Student

The DLS shall provide methods for the student to participate in a DL session using hardware keypads and a large screen projection system or using a PC with emulated keypads. Students shall have the ability to register into classes, answer questions, request assistance, speak to the instructor and take tests. Microphone and speaker systems shall be enabled by the DLS when the instructor opens communication with a student and be disabled when the instructor terminates the conversation.

Administrator

Although the DLS shall be designed so one person could operate the entire network, the system shall provide system console support for maintaining LAN, terrestrial and satellite equipment. Administrators shall have the ability to run diagnostics tests on all DL equipment. Administrators shall also have authority to override the existing communication equipment and enter new parameters. All errors detected shall be displayed on the administrator console and, depending on the error, shall halt execution of the DL control software. Access to equipment parameters shall be secured using a password and encryption. The administrator shall be provided an edit function to enter and change equipment parameters. The edit function shall also be used to enter receive site configurations. Equipment and configuration information shall be accessed from a local or remote database.

SYSTEM ARCHITECTURE

The DL software architecture is shown in Figure 1. The architecture is partitioned into four system components: Instructor Interface Control System (IICS), Student Response Control System (SRCS), Receive Site Control System (RSCS), and Network Control System (NCS). System components communicate with each other over LAN, satellite and terrestrial interfaces using TCP/IP, a De-Facto communication protocol. Referring to Figure 2, data flow within the DLS starts with instructor commands and status transmissions between the IICS and NCS. The IICS component is responsible for accepting commands from the instructor and displaying network status. Instructors will also use the IICS component to define classroom receive site configuration; initialize receive sites, display test and question results; enable receive site dialogue; and access classroom rosters. Rosters and tests can be accessed from a central repository over the LAN or WAN interfaces.

The NCS component will control terrestrial and satellite equipment and will be the focal point for information passed between the instructor and receive sites. The NCS sends commands and receives status from all receive sites participating in the classroom lecture. The RSCS is responsible for communication with the broadcast site NCS and controlling data flows to and from the SRUs. If the SRUs are hardware devices, the RSCS will have a direct connection to the SRUs. However, if the SRUs are emulated on a workstation, then the RSCS will communicate with a SRCS to control data flow to and from the student. The RSCS and SRCS will be connected on a LAN at the receive site and will utilize TCP/IP as the LAN communication protocol.

Specific features within the four system components will be offered as software modules. Notice that software modules do not have direct communication links to each other (Figure 1). Modules communicate with the executive program using a standardized inter-module communication interface. It is this standard interface that will allow software upgrades to existing platforms with minimal replacement of existing software or hardware.

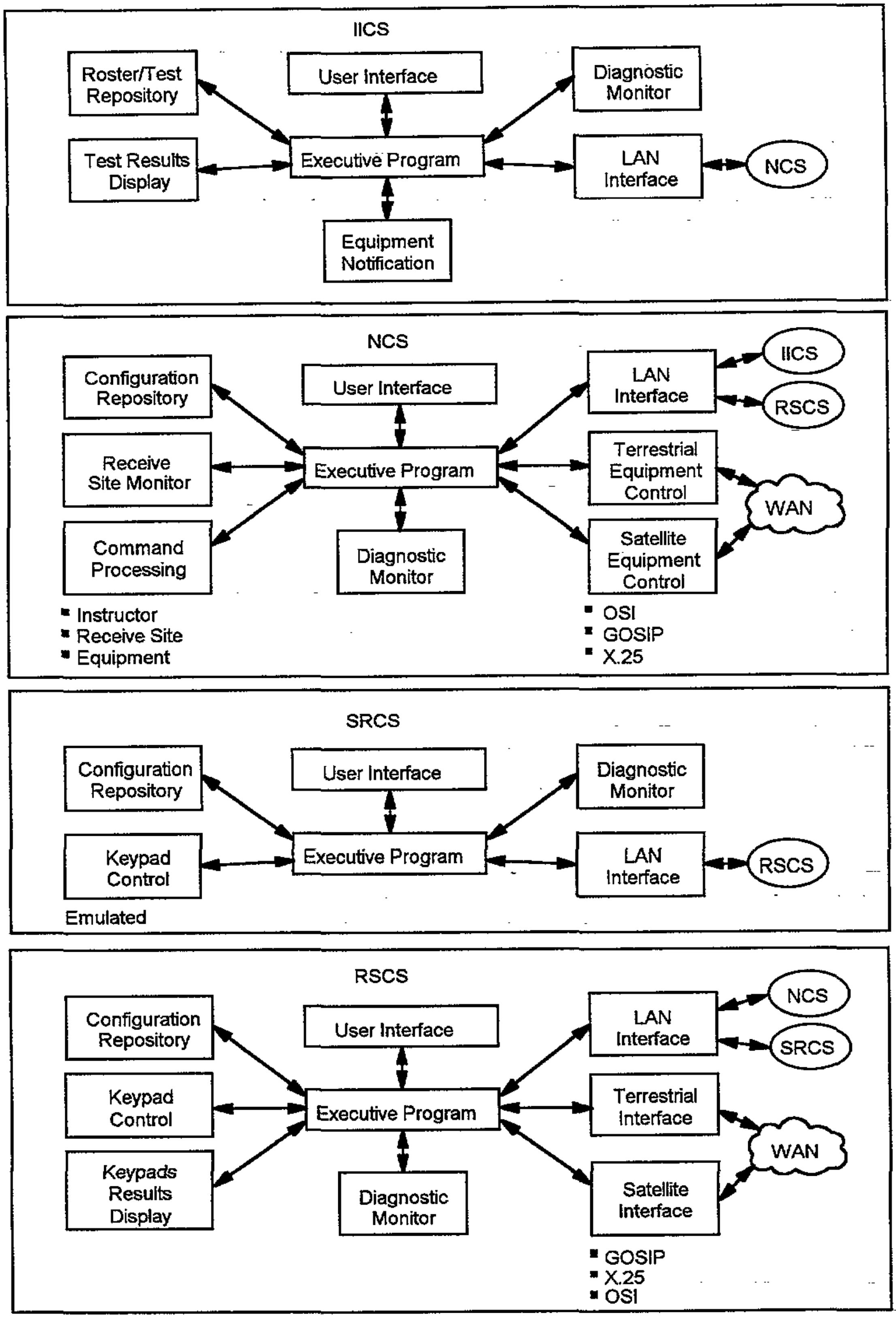

Figure 1. DLS Control System

06/93000596/F158

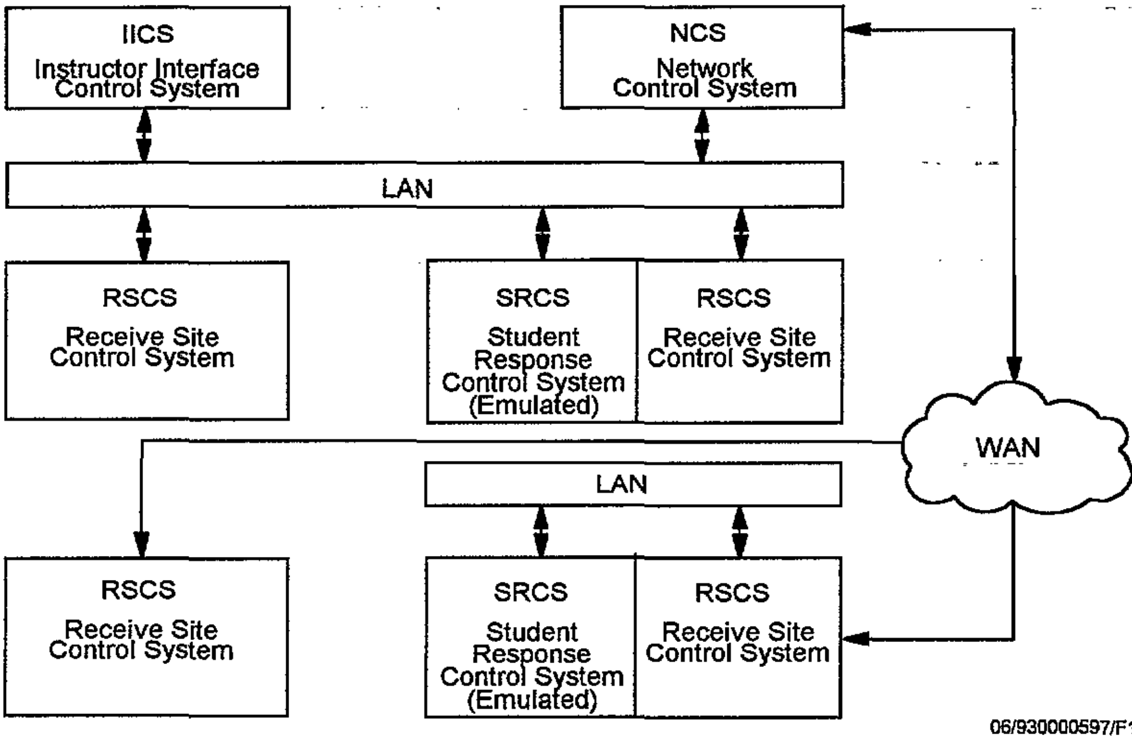

Figure 2. DLS Data Flow

The DLS will utilize the SQL interface, along with the DCE Remote Procedure Calls (RPC), to access information from file/database servers. This architecture can also support the CALS initiative by adding software modules that will transform information into CALS compliant formats before data is transferred to the central repository.

Notice that separate modules are used to control communication over WAN and LAN interfaces. By developing communication interface software as functional modules, the DLS will be able to utilize software that complies with the full GOSIP/OSI stack; portions of the GOSIP/OSI stack; or utilize an internal military communication standard.

The Executive program will control the flow of information throughout a DLS component. Specifically, the Executive program will: read a configuration table; select the appropriate interface drivers; control data transmissions between the DLS components; and control information transmissions between internal modules. The power and flexibility of the executive program is dependent on the type of operating system selected. Although the best choice for an operating system is a POSIX compliant system, current POSIX compliant systems are limited in multimedia support compared to DOS, DOS/Windows and OS/2 operating systems. The NCS will require a multitasking operating system due to timing constraints and the amount of events that have to be managed. Although it is recommended to use a multitasking operating system in the RSCS, IICS and SRCS components, the components can provide all the functional capabilities described in this paper using a single tasking operating system.

DLS Control System

The control system will encompass all the software and hardware that manages the flow of data between the Instructor and the student. As shown in Figure 1, within each component are modules that perform specific functions. Because many of the component modules perform the same functions, only the NCS and RSCS components will be discussed in detail.

The NCS component is responsible for the operation of the DLS. Specifically, the NCS Executive program will be responsible for initializing the DLS network; monitoring receive sites; and processing instructor and administrator commands. The NCS Executive program will utilize a round-robin method to perform all of its tasks. Instructor commands, incoming messages, outgoing status and hardware errors are constantly checked. There will be instances when the round-robin cycle is interrupted, such as when an catastrophic hardware error occurs or when an instructor command is detected. By using this round-robin and prioritization technique, the NCS can guarantee system performance in executing instructor commands and processing receive site responses.

The NCS Executive program utilizes the other modules to communicate with receive sites, equipment and data bases. The Configuration Repository module will be responsible for communicating with data bases and allowing instructors to build configuration tables. At initialization, the Executive program will request initialization information from the Repository module and will initialize the system. At this point, the Receive Site Monitor module will begin sending commands to the receive sites to determine their status (active or inactive). The Executive program will use the configuration data obtained from the Configuration Repository module to determine what interface module to use (LAN, Satellite and/or terrestrial). Receive site responses will be handled by the appropriate interface module (LAN, Satellite or terrestrial) and will be forwarded to the Receive Site Monitor module. A list of active and inactive sites will then be generated and maintained. Throughout the distance learning session the Receive Site Monitor module will monitor the status of receive sites by issuing polls on a constant cycle. Receive sites will be considered inactive if they do not respond to a predetermined amount of poll requests. When receive sites move from inactive to active, the Receive Site Monitor module will update its status list and notify the IICS and NCS components.

Incoming commands from the NCS and IICS components are processed by the Command Processing module. When commands are received, they will be prioritized and inserted on a queue. High priority commands, such as equipments errors, will be processed immediately. The Command processing module will terminate a DLS session if a severe equipment error is detected. The Command processing module also formats and packetizes instructor commands for transmission over the three interfaces.

The Diagnostic Monitor module will monitor internal module communication and detect software errors. Errors detected will be logged in a central or remote database for analysis purposes. When software errors are detected, the Diagnostic module will notify the IICS or NCS components.

Communication equipment will be controlled by the LAN, Satellite and Terrestrial communication modules. These modules also packetize and format data to the standard specified in the configuration table (i.e., TCP/IP and GOSIP).

The RSCS will control all data flows between the NCS component and the SRUs. Poll and status commands are processed by the RSCS and, depending on the receive site return interface, will be transmitted back to the broadcast site upon receipt of a request for status poll or will be transmitted immediately. Keypad commands such as enable/disable keypad and start/stop help request polling will be processed by the Keypad Control module. The Keypad Control module will poll keypads at least twice the speed of the NCS receive site poll to ensure help requests are detected within one NCS poll cycle.

The RSCS will operate as a broadcast site processor or a remote site processor. In remote site mode, the RSCS will control hardware SRU devices or communicate over the LAN with the SRCSs to control the display of software emulated SRUs. In broadcast mode, the RSCS has the above capabilities and will also collect keypad results from the NCS and display the results on its monitor using the Keypad Results Display module. The RSCS will only operate in this mode if it is attached to the NCS over a LAN interface. This capability will allow the instructor to broadcast from any LAN attached classroom.

All of the remaining modules in the RSCS provide the same functions as described in the NCS section above.

IBM has performed tests on the performance of a DLS that has been delivered to a customer. The system is very similar to the DLS being proposed in this paper. The NCS component was developed on a 386 machine running at 25Mhz. We were able to monitor 10 receive sites within a three second poll interval and 40 receive sites within a ten second poll interval. When an instructor requested collection of student keypad data from 40 sites, the results were displayed within 10 seconds. Of course, the performance is dependent on WAN speeds, as well as processor speeds and return communication interfaces. However, this test showed that the round-robin technique can provide sufficient performance when large numbers of receive sites are configured into a DL network.

User Interface Design

Targeted users of the DLS are not necessarily computer experts or even computer-literate. Therefore, our interface design focuses on specific human factors such as: consistent button layouts; consistent message formats; system guided user selections; Common User Access (CUA)/ Graphical User Interface (GUI); touch, mouse, keyboard input capabilities; and an overall friendly user interface. Three interfaces are discussed in the following paragraphs: instructor, student and administrator.

Instructor

The IICS will be the instructor's main interface to the DLS. A windowed environment will be developed to allow the instructor to easily toggle between a DL session and other activities being performed on the workstation. The graphical buttons provided to allow the instructor to have complete control of a DL session are Receive Site Setup, Start/Stop Roster Generation, Ask/End Question, Start/Stop Test, Enable/Disable Audio Dialogue, Disregard Help Requests, Classroom Status and End Class. Because many functions must precede others, the buttons will gray in and out as each function is performed. The user interface will guide the instructors through the DL control features. Therefore, an instructor can not inadvertently cause a system crash during a DL session. Another capability that will be provided is an ability to toggle functions using one button. This capability helps to logically group functions and decreases confusion due to a cluttered screen. For example, the Ask Question button toggles to End Question, and vice versa. Message boxes will be used to notify the instructor when a student has a question; when errors are detected in interface equipment; or when a receive site becomes inactive due to transmission problems. List boxes will be used to store help requests and receive site roster data. This information is visible on the screen at all times. List and message boxes will have a consistent format for all types of status and error information being displayed to the instructor.

Student (Emulated Input Device)

The main responsibility of the SRUs will be to provide a graphical user interface to students for inputting data, requesting help and viewing a lecture. This interface will also be in a windowed environment, allowing the user to toggle back and forth between a DL session, CBT application or any other type of computer application. A graphical representation of an input device will be provided, which will be presented in a separate window. The device will have a display area to allow the student to view the information being entered. Student input devices will support both numeric and text entries. The display area will scroll up and down when text information is entered. Student input devices will be activated by the instructor (ask questions and roster generation) or by the student (request help). When the input device is not active, it will appear as an icon on the screen. Students will view lectures in a separate, scalable window. Furthermore, students will have the ability to move the video window anywhere on the screen and change sizes from 1/4 screen to full screen. The interface will also allow the student to toggle between the input window and video window during a DL session. Similar to the input device window, the video window will be controlled by the instructor and student. Touch, keyboard and mouse inputs will be supported.

Administrator

The NCS will be responsible for controlling the DL interface equipment and providing a windowed user GUI to administrators for monitoring equipment, initializing equipment and displaying messages. Similar to the IISC interface, buttons will be provided to allow instructors to select specific options. Certain functions will be password protected. Message boxes will be provided to display parameter variables and error messages. List boxes will be used to allow the administrator to view and select from a range of equipment parameters.

SYSTEM IMPLEMENTATION COSTS

The cost associated with developing a DLS having all of the capabilities described above can be insignificant if the organization has an existing LAN based infrastructure. The DLS software could be loaded on the existing PCs and multimedia cards could be added to provide video and audio capabilities. A low end solution would be to provide distance learning across a building using an analog network to transfer modulated audio and video signals. There are technologies available that will allow the transmission of analog audio and video, along with baseband data, over existing LAN cables.

If receive sites are large distances from the broadcast location, video and audio transmission equipment will be needed. Cost trade-offs between terrestrial and satellite transmissions should be performed. As a simple rule of thumb, if the number of receive sites is more than four, it may be cheaper to use the satellite system. This formula may not be true in the near future with advancement being made in high speed digital communication services over land lines. Carrier charges will have to be added to life cycle costs. Also, charges for purchases of video compression, satellite and echo cancellation equipment will increase costs.

DLSs can only be justified if the system can effectively be used to train students and its cost is within the existing training budget. Cost justifications for DLSs involve an analysis of how conventional training is being conducted. Costs such as instructor travel, student travel, building costs, room and board, and lost time on the job should be included in cost estimates. Furthermore, the number of student being trained, number of instructors and future training requirements should also be factored into the cost estimate.

The proposed architecture will allow training organizations to better justify a DLS because the system operates in an existing client/server environment. Every user on the network can be trained without buying expensive communication and compression equipment. As the existing infrastructure grows to support more users, the DLS transmission capabilities grow also. For example, when high speed communication networks become available at cost effective prices, the DLS can be upgraded to support digital transmissions of video and audio over the network. This upgrade does not affect the DLS because the compression equipment is not tied to the software platform.

CONCLUSIONS

The development of a DLS requires a full understanding of standards and future technologies. The explosion in utilization of PCs in the home and work place has lead to an environment where PCs will be the primary device used for training and education. Therefore, it is imperative for training developers to provide solutions that will allow organizations to train people in their office and home environments. However, in developing DL training solutions, developers must provide systems that support a seamless integration into existing client/server environments and allows growth without major changes to existing hardware and software. The system must also allow instructors to use conventional stand-up lecture teaching methods so that student comprehension/retention is not compromised by the new system. The implementation of our proposed system satisfies the requirements described above because the system is nothing more than a set of client/server applications running in a networked environment. The software architecture is designed to adhere to standards and, therefore, supports an Open System environment. IBM has developed and delivered DL solutions that utilized portions of the proposed architecture.