SOFTWARE DEMONSTRATIONS AT A DISTANCE

Paul L. Schlieve, PhD*
Terry L. Holcomb, PhD*

Transferring instructional strategies from a face-to-face classroom setting into a distance-learning setting is an ongoing challenge. When the instruction includes the operation of software, the challenges increase dramatically.

The landscape is littered with failed "distance" software demonstration technologies: Scan conversion of computer screens into NTSC format, remote control of remotely-located computer systems, computer conferencing, and static graphics displays all suffer from various limitations that can prevent successful communication of ideas to students.

At a fundamental level, computer displays and television systems don't mix very well. The relatively low horizontal frequency and the use of interlacing are poorly suited to the display of computer software screens.

Our successes with streaming lecture video material, both live and prerecorded, with RealNetworks technology has led us to rely extensively on that technology for the delivery of instruction. While the video quality of lecture materials has been highly acceptable to students, initial attempts to include software operation within a course were less than satisfactory.

Before projection video systems that accepted direct computer input became common and affordable for in-classroom presentation of computer software, we used scan conversion technology to convert the computer's output into an NTSC television signal for projection. While the technology limited us to 640x480 resolutions and was certainly a compromise in quality, it was a successful strategy for in-class software demonstrations. The extension of scan-conversion technology to Internet-based streaming video, however, was disappointing. We determined that it was, in fact, possible to generate a video signal at the receiving end that resembled the original source material. However, the combined degradations of scan conversion and the lossy compression of streaming conspired to provide a technology that only provided readability of the largest fonts – such as might be used for the main titles in a PowerPoint™ presentation. It quickly became clear that streaming scan-converted images of computer screens had little potential as an instructional tool.

Somewhat more successful was the technique of capturing still images of key screens and transmitting them to the user as a series of still graphics. In such a scenario, the graphics can be delivered as Web images, or delivered as part of a

*Department of Technology and Cognition, University of North Texas

streaming multimedia presentation using SMIL (Synchronized Multimedia Integration Language) from a RealVideo™ server. While this technique works great for computer presentation slides and many illustrative graphic materials, there are, however, three factors that work against this strategy in the context of software demonstrations..

The first issue with the use of still graphics to demonstrate software is the inherent loss of "flow" when moving from a continuous-motion media to a still media. When demonstrating software, the user's eye is naturally drawn to the area of the screen that is in motion. When the motion is removed, the user faces an additional challenge in determining which portion of the screen image is important at a particular instant.

The second issue with still graphics is closely related to the first – the materials are time-consuming to produce. Capturing, arranging and presenting still graphics is time-consuming. The inherent loss of focus in the transition from a motion-based demonstration to a series of still graphics can be mitigated somewhat by adding boxes, lines, arrows, and key text to the graphics. However, each of these manipulations adds tedious graphics production time to the process.

The final issue with presenting software demonstrations as a series of still graphics is that the demonstration must be "canned." It is not possible to generate a live demonstration in real-time using still graphics.

The most recent technique, the use of video streaming technology to deliver software demonstrations across the Internet overcomes the limitations identified to date and now makes it practical to conduct remote software demonstrations using standards-based technology. The breakthrough technology comes from OPTX International of Chico, California and is marketed under the name ScreenWatch. The system starts with a Windows NT computer system running the ScreenWatch recording software, which hooks into the Windows NT display driver, and saves the information as a file that can be streamed from a RealVideo server.

ScreenWatch isn't the first software to be able to record screen activity. However, the compression achieved by the ScreenWatch system is an order of magnitude better than alternative solutions. Allowing bandwidths as low as 28kb/sec with corresponding file sizes of less than ten megabytes of disk storage per hour, makes it realistic to deliver ScreenWatch software demonstrations across dial-up connections and to archive many hours of software demonstrations on servers with modest hardware specifications.

The ScreenWatch production cycle is surprisingly simple. It is important to note that the ScreenWatch media type is a visual media only. It does not address the need for a corresponding audio track. Audio materials are typically captured at the same time as the screen recording. It isn't reasonable to record the audio in the background of the computer doing the screen recording. It is reasonable, however, to record the audio onto tape for later digitization while the screen recording is being made. It is also reasonable to capture the audio directly into an audio recording application running on a different computer system. One trick that makes the production process run more easily is to open a text editor (i.e. Windows *Notepad*) immediately after starting the screen recording process and type a series of numbers, reading the numbers aloud while typing them. Later, this can serve as a synchronization reference point for synchronizing the audio and ScreenWatch data streams. When the audio and video are put together with SMIL, simply start each data stream with the appropriate offset to achieve perfect synchronization, skipping past the actual synchronization content. The generated files are so small that the small bit of disk space occupied by the synchronization content is not worth worrying about.

The technology is not without limitations. Although the image delivered to the student is an absolutely beautiful 800x600 display, one of the keys to the high compression provided by the system is a 16-color limitation. So while the system does a fabulous job of demonstrating the operation of office productivity software, network management software, programming systems, and the like, teaching concepts in image manipulation using software such as Photoshop™ would suffer from the limited color palette.

The most recent capability to be added to the software is the ability to do live streaming of software demonstrations by using the live-streaming ability of the RealVideo server. Doing software demonstrations on demand in response to student questions is an important instructional component in a computer applications course.

Internet-based distance learning is an exciting new instructional frontier. The capacity to demonstrate software procedures across the Internet to students in multiple locations is an important component the Internet-based distance learning puzzle. Many of the frustrations and limitations scan conversion and still-graphic simulations are addressed by ScreenWatch streaming screen displays in combination with RealVideo technology.