



Video Streaming Solutions for Web-Based E-Learning Courses

Karl Kurbel

European University Viadrina Frankfurt (Oder), Business Informatics
POB 17 86, 15207 Frankfurt (Oder), Germany
Telephone: +49-335-5534-2320, Fax: +49-335-5534-2321
E-Mail: kurbel@euv-frankfurt-o.de

ABSTRACT

The background of this work is the development of multimedia courses for an Internet-based master program in Business Informatics. This program is provided by a virtual organization. Delivering multimedia-based courses over the Internet is a challenging task. Low-cost multimedia technologies, in particular technologies for video production and delivery over the Internet, have just become available. Established rules and recommendations how to use those technologies effectively are still missing. In this paper an approach is discussed in which course material is divided into separate parts - one silent video file containing visual demonstrations, and another video with sound, with the instructor's explanations. We discuss approaches to achieve synchronization among two separate videos streams. Based on Microsoft Windows Media technology this problem can be solved but the solution is not straightforward. Problems encountered and the final solution to those problems are outlined.

Keywords: Streaming video, video-based e-learning courses, Windows Media technology, synchronizing playback, Virtual Global University

1 BACKGROUND OF THE WORK

The work reported in this paper is based on experiences from planning and operating a virtual master program in Business Informatics. This program was developed by the Virtual Global University (VGU), an organization founded by 17 professors of Business Informatics from Germany, Austria and Switzerland [1]. It is offered worldwide, leading to the degree of an "International Master of Business Informatics" (MBI). VGU as a private virtual organization has a cooperation agreement with a "real" state university - the European University Viadrina (EUV) in Frankfurt (Oder), Germany. While VGU provides expertise and teaching for the program, EUV is responsible for ensuring that the academic and educational standards of the program are maintained at an appropriate level.

Many courses and programs in today's virtual education are still text and paper based. We will refer to those as "traditional" virtual courses and programs. In Europe, rather advanced multimedia technologies for higher education have been investigated in research projects and prototypical courses over the past 10 years (e.g. [2]). Advanced tools for cooperative work are also being tried [3].

For the master program underlying this paper, multimedia technologies above the text-based level were intended to play a major role in the instructional design. The technologies for the MBI courses were chosen depending on the type and on the content of a course. A mix of technologies is actually used, while video and audio-based courses are the most important ones.

Section 2 outlines the use of video technology for e-learning. Subsequently, the front-end of a video-based e-learning course as it appears to the user is discussed. Section 4 addresses the problems that have to be attacked to produce and deliver such a course over the Internet. In section 5, features and shortcomings of the Windows Media Player as observed in our work are described and the solution to the problems encountered is presented. Some conclusions from our experience with video-based courses are drawn in section 6.

2 VIDEO AS A BASIS FOR E-LEARNING

Web-based teaching and learning are becoming more and more popular as the numbers of virtual courses and virtual universities (like JIU [4], VGU [5], and VIROR [6]) are growing. While in the beginning static HTML pages, text, and graphics files were mainly used, animated and multimedia contents have significantly enhanced e-learning since then. Video clips are also found in web-based courses, but videos as the primary instructional medium are not very common yet. One reason for that is certainly that production and delivery of video-based courses are rather time consuming, requiring profound know-how about the technology.

On the other hand, video-based courses have one significant advantage. Many lecturers have years of experience giving "lectures", that is standing in a lecture hall or a classroom and explaining complicated matters to students, possibly using visual aids like overhead transparencies or Powerpoint slides. In computer-oriented subjects, online presentations on a video beamer may be another form of visualization. Using videos as the main instructional medium makes it possible to transfer this well-trying, though traditional method of teaching to the web age.

Common platforms to provide and distribute streaming video content over the Internet are Microsoft Windows Media [7], Real Media [8], and Apple QuickTime [9]. Windows Media technology was chosen for several reasons. First, the Windows Media Video V7 Screen codec (codec = coder/decoder for video and audio data) proved to be very good for online recordings from a computer screen. In our courses we need this feature because often things are explained by the lecturer online and real-time. Second, the Microsoft Media Server is included in the licence for Microsoft Internet Information Server (IIS) and that licence does not limit the number of client connections. The Real Media Server, for example, is free but only for 25 client connections at a time.

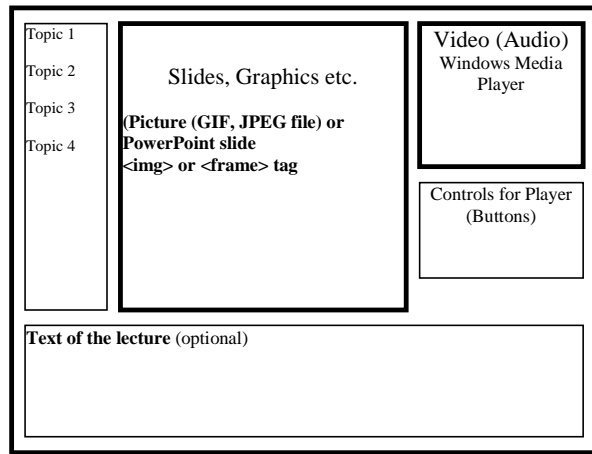
3 E-LEARNING USER INTERFACE WITH TWO VIDEO STREAMS

Following the paradigm of lecturing and visualizing, the user interface of VGU's video-based courses as it is presented to the student is outlined in figure 1. The crucial part of the web page is the Media Player in the upper right corner. In this corner a video of the lecture is running in a streaming compatible format.

The area left of the player is used to show visual materials accompanying the lecturer's explanations. In the simplest case these are text slides, tables, diagrams, charts, etc., similar to information that would be written on a blackboard or an overhead projector in a conventional lecture. In addition there are controls for user interaction with the course page, for example starting the lecture, advancing or going back to a certain topic, etc.

Basically there are two approaches to provide visual material plus video lecturing for such a course. The first one is fairly easy to achieve: Split up the lecture in separate topics and store the respective video clips in separate files. If the topics are associated with slides, then each topic, including the video clip, will be started separately.

Fig. 1: Functional diagram of a course page



This solution is easy from a technical point of view, but it is not always feasible to split up a lecture in such a way. Moreover, only static images can be used as accompanying visual information. Displaying online recordings of program runs, for example, or videos shown by the lecturer is not possible in this way; nor can recordings from two or more video sources be displayed at the same time. For example, it is not possible to do screen capturing on two or more computers and record the pictures into one file. One more problem is that screen recordings cannot be merged with recordings from a video camera without serious loss of quality, or only at the cost of very large files.

As a possible solution for those problems one might think of using a pop-up window in which the second video runs. However, this is quite uncomfortable, requiring time for connecting to the media server and for preloading/buffering some video data.

In some situations it is better, or even the only possible solution, to use two synchronized video clips. One example is the production of multilingual content. Here it is advisable to separate a video file with visual material from sound files for different languages. This solution saves space on the hard disk, proportional to the number of languages and to the size of the video file.

In the general outline of a course page as shown in figure 1 the area for visual material can be used for displaying a video stream as well. How this is achieved will be discussed in section 5. For the student there is no difference in the graphical user interface no matter whether static files or videos are displayed. Technology dependent peculiarities are hidden below the user interface.

4 INTEGRATING SEPARATE VIDEO STREAMS: FUNDAMENTAL PROBLEMES

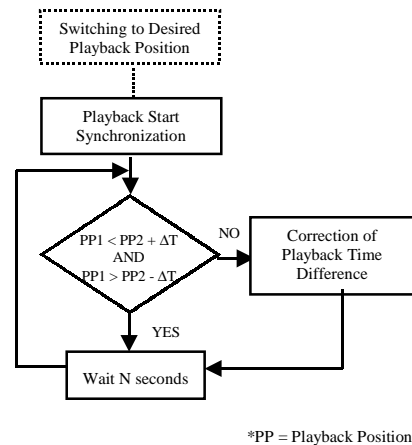
Major problems in creating course pages like the ones in figures 1 and 2 result from limitations of the Windows Media Player object model. The available methods and properties do not provide mechanisms to control all stages of client-server interaction and to redistribute connection bandwidth among two players adequately. In other words, even if more connection bandwidth is available than required by the sum of bit rates of the two video streams, one cannot expect that the bandwidth will be divided in proportion to the bit rates of the streams, nor even equally.

This shortcoming implies that two video files will rarely start playing simultaneously even if they have exactly the same parameters (e.g. resolution, codec, file size). The synchronization task is thus left to the course creator.

To guarantee proper appearance and behavior of video-based courses we developed and implemented an algorithm for synchronizing and monitoring two video streams. It has the following components as shown in figure 2:

- Synchronizing playback start,
- Monitoring synchronized playback of two videos,
- Corrections if synchronization is off,
- Setting two players to specific positions on demand.

Fig. 2: Synchronization algorithm



The problem that two players do not start at the same time is actually caused by the different data compression algorithms in the codecs used. The video file from screen capturing (with Matchware Screen Corder 2.0 [11]) is translated into streamable format with the Windows Media Video V7 Screen codec. The conventional camera recording is compressed with the Windows Media Video V8 codec.

Because of the particularities of these codecs a procedure is needed which checks the states of the players. When the first one is ready to play a video, the procedure checks the state of the second player and waits until the second one is also ready to start.

Once playback has started it is necessary to check in short intervals if the two videos are still in the correct positions. The playing times can be measured in seconds or frames from the start of the files. If a difference is detected and if that difference exceeds a certain limit a procedure is started that eliminates the time lag.

The procedure for correcting the differences in synchronization pauses the player that is running ahead and checks playback of the late player. This is done until the slower player has approximately caught up with the other one. Then the first player is released to continue playing.

One serious problem, however, needs to be observed. Due to various reasons the players can change their states (like playing, pausing, buffering, etc.) between individual calls of the procedure and even during the execution of the procedure. Reasons for this may be, for example, actions by the user, reaching a marker in the video file, or a state change in the connection with the Media Server. Therefore additional monitoring of the states of the two players not only at the beginning but also during playback is very important. Otherwise the procedure can crash or the user's web browser can get frozen.

As indicated in figure 2 the user may jump directly to a specific topic of the lecture forwards or backwards by clicking on a control on the left-hand side. In such a case both video files have to be advanced or set back to the desired positions in a synchronized way. Once this is achieved both players have to continue or start playing at the same time. This is basically the same problem as ensuring a synchronized start of two players in the first procedure, so that procedure can be adopted and reused.

5 INTEGRATING VIDEO STREAMS WITH WINDOWS MEDIA TECHNOLOGY

Providing two videos on the course page made it necessary to find a way to include two instances of the Windows Media Player (WMP) into one page. As stand-alone applications, two or more players cannot be started. Microsoft's technical documentation, on the other hand, does not specify whether it is allowed to use two or more WMP ActiveX controls in the same HTML page or not.

Therefore tests were performed to find out whether this is possible or not. Two WMP ActiveX controls were embedded in one page. In fact, both controls established a connection to the Media Server and started playback whenever the page was loaded. At the same time we found that a synchroniza-

tion procedure is absolutely necessary because the playbacks never started simultaneously.

From the WMP versions available at the time the tests were performed, 6.4 and 7.1 [10], the latter one was chosen because more arguments seemed to be in favor of that version. Later when 9.0 became accessible we tested a prerelease of that version as well and found that its behavior was unsatisfactory.

In our initial work with WMP 7.1, two ActiveX controls were embedded into the HTML page with the help of the <object> tag. Synchronization procedures for playback as outlined above were developed in VBScript.

Testing the initial pages on six computers with different processors and connection speeds showed that program crashes occurred sometimes. The number of failures was inversely related with actual system load. Since the reasons for those system failures were neither obvious nor found in the technical documentation, a testing procedure had to be developed to allow for step-by-step execution of the synchronization procedures.

In a sequence of tests we found that all page crashes and browser hangs are occurring during or immediately after the first access to the properties or methods of the Media Player ActiveX control. However, when that control was accessed again after some time crashes rarely occurred. This behavior of the WMP 7.1 control gave rise to the assumption that for some reason the indicated state of a control (i.e. the value of the "PlayState" property) changes faster than the actual state of that control. In other words, the value of the properties is changed first and only afterwards all operations belonging to that state change are executed. This observation explains to some extent why the scripting procedures behave differently when executed on different computers.

A similar phenomenon can be observed on a conventional video tape recorder: When the "Play" button on the remote control is pressed the "Play" sign appears immediately on the recorder's display. However, before a picture from the tape appears on the screen the tape has to be stretched around the video head, and appropriate speeds of both revolving of the head and transporting the tape have to be established. The process may take up to several seconds. During that time the recorder displays the "Play" sign but no picture is visible on the screen. This problem has been known in the field of professional video editing for a long time. Today's video equipment has special functions that allow winding the tape back for several seconds to compensate for the difference between the indicated and the actual playback start.

However, solving the problem of delayed action in the underlying problem in an analogous way - by placing pauses after one player's actions - was not possible. Our findings from a long phase of testing and debugging efforts were that the Windows Media Player 7.1 version requires powerful computer systems to avoid the synchronization problems mentioned above. On many PCs the system performance demanded to play synchronized videos in a reliable and stable way is not available. Therefore the decision was made to go back to the Windows Media Player version 6.4 in order to obtain reasonable performance of the video material on average personal computers.

Adapting the synchronization procedures for WMP 6.4 required some deep changes because the object models underlying the versions 7.1 and 6.4 exhibit significant differences [10]. For example, the synchronization procedures responsible for playback start use the "BufferingProgress" property of the 7.1 version. This property helps to detect if a player is ready for playback or not. Once the value reaches 100 (%) then it is possible to start the playback. In the 6.4 version, however, this property always gets the value 100 when the specified time for buffering is over, no matter whether the buffering has really been accomplished or not. Therefore the procedure that checks the buffering progress had to be substituted by a procedure that watches the "BufferingStart" event. This event occurs at the start and the end of buffering.

Tests with the WMP 6.4 version were performed on the same computers as the previous tests with WMP 7.1. Results were excellent with respect to system stability, independently of actual system workload and bandwidth of network connection.

6 CONCLUSIONS

As a result of our work we found that video streams for e-learning can be synchronized using components of the Windows Media platform although this

is not specified in the technical documentation. One solution to the synchronization problem is to use an HTML page with two ActiveX controls representing Windows Media Players. The 7.1 version of the Windows Media Player with many new features is only appropriate if it can be run on a sufficiently powerful computer to avoid the problems and program crashes encountered in our tests. WMP 9.0 in the prerelease version we tested did not satisfy the requirements with regard to playing two synchronized videos.

WMP 6.4 is compatible with all Windows operating systems including Windows NT, and with the Netscape Navigator web browser. This version plays two synchronized video streams correctly, practically on every PC that fulfills the minimum system requirements imposed by modern video codecs such as Windows Media Video V8 and Microsoft MPEG 4 V 3.

The synchronization procedures written in VBScript work correctly and robustly on any PC with that capacity. They provide satisfactory playback synchronization even if the available network bandwidth is too low to deliver both video streams simultaneously in real-time. In such a case some pauses will inevitably occur, due to additional buffering, but the videos will still run correctly. If it is known that the network bandwidth is smaller than the sum of the bit rates of the two video streams, then it might be considered to preload one of the files, store it, and synchronize playback with the video stream delivered real-time. The synchronization procedure does not require any changes in this case.

Taking into account that more and more ADSL connections with stable bandwidth of 400 - 700 Kbit/s are becoming available, this bandwidth is sufficient for uninterrupted delivery of a lecture video with 320 x 240 pixel resolution and 25 fps, compressed with Windows Media Video V8 codec, together with another video containing visual demonstration material with 800 x 600 pixel resolution and 10 fps, compressed with Windows Media Video 7 Screen codec. From our experience it can be concluded that the technical conditions for producing and delivering e-learning courses based on synchronized video streams already exist. In the project underlying our work - developing multimedia courses of the Virtual Global University (VGU) [5] - this approach has been successfully applied.

REFERENCES

- [1] Kurbel, K.: A Completely Virtual Distance Education Program Based on the Internet - Case and Agenda of the International MBI Program; in: Hansen, H. R. et al. (Eds.), Proceedings of the 8th European Conference on Information Systems - ECIS 2000, A Cyberspace Odyssey, Vol. 2, Vienna, Austria, pp. 1363-1367.
- [2] Bodendorf, F., Grebner, R.: Towards Virtual Universities - ATM-Based Teleteaching in Germany; in: Forcht, K. (Ed.), Information Systems Beyond 2000 - 1998 IACIS Refereed Proceedings, Cancun, Mexico 1998, pp. 67-74
- [3] Kloeckner, K.: Preparing the Next Generation of Learning: Enhancing Learning Opportunities by WEB Based Cooperation; in: Proceedings of the Eden 10th Anniversary Conference, Learning Without Limits - Developing the Next Generation of Education; Stockholm, Sweden, 2001
- [4] Jones International University (JIU); <http://jiu-web-a.jonesinternational.edu/>
- [5] Virtual Global University (VGU); <http://www.vg-u.org>
- [6] Virtual University in the Upper Rhine Valley (VIROR); <http://www.viror.de>
- [7] Microsoft Windows Media; <http://www.microsoft.com/windows/windows-media/default.asp>
- [8] Real Media; <http://www.realnetworks.com/>
- [9] Apple Quick Time; <http://www.apple.com/quicktime/>
- [10] Windows Media Player; <http://www.microsoft.com/windows/windows-media/players.asp>
- [11] Matchware Screen Corder; <http://www.matchware.net/>
- [12] Feamster, Nicolas G., Adaptive Delivery of Real-Time streaming Video, M. Eng. Thesis, Massachusetts Institute of Technology, Cambridge 2001
- [13] Wee, Susie J.; Apostolopoulos, John G., Secure Scalable Video Streaming For Wireless Networks, IEEE International Conference on Acoustics, Speech, and Signal Processing, Salt Lake City, Utah 2001

0 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:
www.igi-global.com/proceeding-paper/video-streaming-solutions-web-based/32168

Related Content

Pricing Based on Real-Time Analysis of Forklift Utilization Using RFID in Warehouse Management

Numan Celebi, Kübra Savaand Ihsan Hakan Selvi (2018). *Encyclopedia of Information Science and Technology, Fourth Edition* (pp. 5490-5502).

www.irma-international.org/chapter/pricing-based-on-real-time-analysis-of-forklift-utilization-using-rfid-in-warehouse-management/184251

Factors Impacting Defect Density in Software Development Projects

Niharika Dayyala, Kent A. Walstrom, Kallol K. Bagchiand Godwin Udo (2022). *International Journal of Information Technologies and Systems Approach* (pp. 1-23).

www.irma-international.org/article/factors-impacting-defect-density-in-software-development-projects/304813

A Systemic Framework for Facilitating Better Client-Developer Collaboration in Complex Projects

Jeanette Wendy Wing, Doncho Petkovand Theo N. Andrew (2020). *International Journal of Information Technologies and Systems Approach* (pp. 46-60).

www.irma-international.org/article/a-systemic-framework-for-facilitating-better-client-developer-collaboration-in-complex-projects/240764

Incremental Learning Researches on Rough Set Theory: Status and Future

Dun Liuand Decui Liang (2014). *International Journal of Rough Sets and Data Analysis* (pp. 99-112).

www.irma-international.org/article/incremental-learning-researches-on-rough-set-theory/111315

A Human Rights-Based Approach to Bridge Gender Digital Divide: The Case Study of India

Ching Yuen Luk (2019). *Gender Gaps and the Social Inclusion Movement in ICT* (pp. 24-44).

www.irma-international.org/chapter/a-human-rights-based-approach-to-bridge-gender-digital-divide/218437