

# ONLINE COLLABORATIVE DRAWING BOARD FOR REAL-TIME STUDENT-INSTRUCTOR INTERACTION AND LECTURE CREATION

Mohammad AlRamahi and Kurt Gramoll  
School of Aerospace and Mechanical Engineering  
The University of Oklahoma

## Abstract

Even though distance learning has become a significant aspect of engineering education, improvements are still needed. One area that needs more research and tool development is web-based collaboration. There is a strong need for collaboration tools to assist both instructors and students in working together during discussion groups, office hours and lectures to solve problems from a distance.

This research involved developing an online tool to allow real-time voice and vector-based drawing communication between the instructor and students. The tool, LectureBoard, was developed to provide online, collaborative and spontaneous lecture creation. The LectureBoard is a vector-based, Flash drawing program that is used through a browser. It provides both voice and text chat in addition to sharing any drawn object on the screen. The tool is completely encapsulated in a Flash Shockwave file so neither program downloads nor special software is required. The tool uses Flash Communication Server MX to coordinate the real-time communications between all session users. The LectureBoard is capable of recording a session and saving it in Flash shockwave format for later playback. This provides other students the opportunity to access the session later for review. The LectureBoard is integrated with the [www.eCourses.ou.edu](http://www.eCourses.ou.edu) engineering course management system at the University of Oklahoma. Use of the system is open to any instructor without cost. This paper gives details on how the system was developed and used to facilitate collaboration for distance learning classes.

## Introduction

Instructors utilize a variety of education techniques to deliver information to their students. Lately, there is an increasing interest in online distance learning due to the capabilities of the Internet [1]. It provides versatile, cost and time effective solutions for distance learning through online lectures. The issue of what is the best method to implement online lectures in the classroom still remains. Building static web pages with hyperlinks, bitmap images, diagrams, and videos have been used for a number of years. New methods to enhance online learning have been introduced, such as collaboration.

The interest in collaboration is not new to the Internet world. It was commercially introduced in 1996 [2]. A major factor that has contributed to its use is that 87% of U.S. workers have broadband access [2]. The collaboration horizon has expanded from companies to include the education sector. Many people are investigating possible uses for collaboration to enhance the online learning methods they are using [3].

One of the most time consuming task for an instructor is to create a lecture in a web-ready format. Creating a lecture is in fact a result of the collaboration between students and their instructor. Tools are needed which enable the instructors and the students to interact with each other, collaboratively work on a lecture, and then allow the option to record it for later use.

Several software packages have been developed to tackle lecture creation for online usage, and the collaboration issues. A well known widely used software available on the

market is Symposium from Centra®[4]. This application has many useful features such as audio/video conferencing, whiteboard, peer-to-peer interaction and more. Symposium has two parts, the client side which the user must install on the local machine and the server side. Symposium is platform dependent, which reduces the audience to the users of that platform. The whiteboard used in Symposium is not vector-based, making it difficult to edit or move a drawing around. The high costs of purchasing, running, and maintaining this application is another problem. Finally, Symposium is a tool developed for general use, and not optimized to be used for technical topics like engineering.

Net Meeting from Microsoft is another real-time collaboration and conferencing, Windows-based client [5]. It contains many features such as audio/video conferencing, collaborative whiteboard, file transfer, and program and remote desktop sharing. While Net Meeting can be used for online collaborative lectures, it was not designed for that purpose. It also must be installed on the user's local machine running Windows® as its operating system. Although the provided whiteboard is able to save pages of drawings, the pages are not saved in a web-ready format, the drawings are not vector-based, and the voices are not recorded.

At North Carolina State University, a new program has been developed called Belshazzar [6]. Belshazzar is a client-side application that helps create lectures on a local computer by using a tablet with pen, which is connected to the serial port, as the input device and saving the drawings and sounds as a Flash movie. The use of the Flash file format is the optimal choice because the Flash player is installed with Internet Explorer, and almost every Internet user has the latest version. Belshazzar is a good tool for lecture creation, and since it reads the input from the serial port, the application response is smooth. However, the software suffers from drawbacks. A major drawback of this software is the lack of collaboration. Belshazzar is client-side software that does not engage with any

other online user. Only the instructor constructs the lecture without student interaction. Since specific brand of tablet must be used, the accessibility of the software is restricted to computers with that tablet.

## LectureBoard

The LectureBoard was developed to help instructors and students hold a collaborative online lecture, and overcome the disadvantages of other tools. LectureBoard was designed with Macromedia Flash MX [7], which interacts with server-side applications and services, to enable instructors and students to hold an online lecture. Both instructors and students can voice chat with each other, draw on the whiteboard, and record the session for later playback. See figure 1 for a snapshot of the LectureBoard.

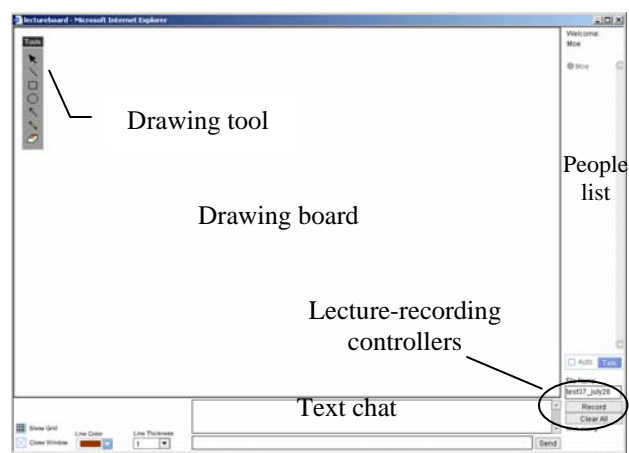


Figure 1: Snapshot of the LectureBoard labeling the main elements

There are several advantages to this tool. First and most importantly, the LectureBoard solves the issue of collaboration for lecture creation. As many users as the bandwidth allows can use the tool and collaboratively work to build a lecture. The bandwidth limits depend on two factors: the edition and license of the Flash Communication Server MX [8]; and the local LAN bandwidth limitations. The basic edition of Flash Communication Server, which is the edition used for the purpose of this research, has a ceiling capacity of 1 Mb/s or 50 simultaneous connections, whichever limit is reached first.

The professional edition has a ceiling capacity of 10 Mb/s or 2,500 simultaneous connections that can be expanded up to 30 Mb/s or 7,500 simultaneous connections, whichever limit is reached first.

Second, the shockwave flash files are automatically generated, stored and archived on the server. This means that no files are saved on the client machine and nothing is processed there. Third, since the LectureBoard was developed with Flash, it is platform independent, with no need for any special programs to be installed on the client machine, and no need to worry about lowering the security level of the web browser. Fourth, this tool accepts input from any peripheral connected to a PC or directly from a TabletPC. And finally, the cost of running and maintaining the LectureBoard is low compared to other tools such as Centra Symposium.

### *Interface*

The LectureBoard was designed to maximize the space of a common XGA resolution screen at 1024x768 pixels because it is common and gives the instructor the widest drawing board area possible. However, since the application is a Flash file, it can be dynamically resized to any size. As shown in figure 1, the LectureBoard contains a white drawing area, list of logged-in students and instructors, text chat window, and two drop-down menus to choose the line color and thickness. The instructor's window contains the same elements plus lecture-recording controllers which are not available for students.

The drawing tool menu is floating so it does not occupy space on the drawing board. The user can move it to any place in the screen at any point of time. Drawing shapes are provided within this menu bar such as line, rectangle, circle, arrow, free-hand, eraser, and object selector.

Users of LectureBoard can communicate with each other in three different ways. First, all drawings can be shared between all users.

Second, users can speak in real-time between each other as they are drawing. Finally users can communicate through the text chat component located at the bottom of the drawing board. The text chat is particularly useful if a user does not have a microphone or there is some technical difficulty with voice.

Colors and line thicknesses were added in order to make the drawings clearer. Each drawn object can have its own line color and/or thickness (figure 2a). Moreover, for engineering users who like having accurate drawings, grids are provided (figure 2b).

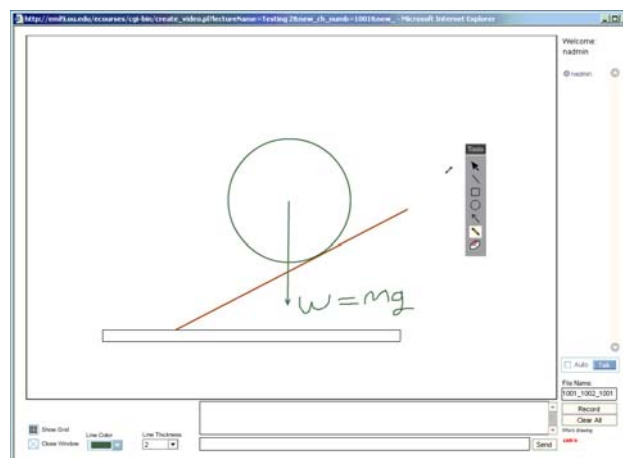


Figure 2a: Drawing objects with different line colors and thicknesses

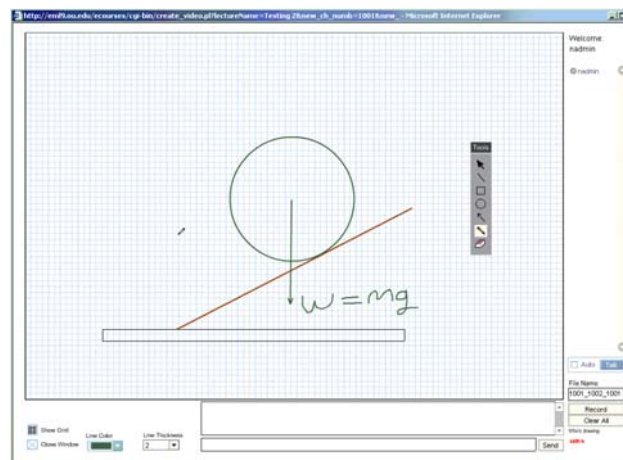


Figure 2b: Illustration of grid feature

### *Privilege Levels*

LectureBoard has been designed with two access levels: students and instructors. A student user can freely use the drawing board to share drawings with other logged-in users, and may communicate with them through the use of the voice and text chat. Students do not have the privilege to record the session, thus the session recording controls are not visible to students. In the instructor mode, the instructor can do everything the student user can do, plus they have the ability to clear the board and record the session.

Since it is only the privilege of the instructor to record a lecture, or a session, the recording controls are shown on his/her window only. The controls include an input text field for the lecture file name; record/stop button, clear drawing button, and finally a dynamic text field that shows the name of the user who created the last drawing object on the screen.

### *Usage of LectureBoard*

LectureBoard was integrated with the eCourses ([www.eCourses.ou.edu](http://www.eCourses.ou.edu)) online course system developed and implemented at the Engineering Media Lab (EML) to be used as an online lecture creation tool. LectureBoard enables the instructor to record an online lecture, while working in collaboration with the students to build it. Since it is integrated with the eCourses system, the created lecture is archived by its associated chapter and section.

### **Design Architecture of LectureBoard**

The LectureBoard consists of both client and server sides. The client side is compiled as a Flash shockwave file and runs inside a web browser. It is able to communicate with the server through the server APIs supported by ActionScript, the scripting language of Flash. These APIs are able to open a two-way connection channel with the server, call ASP pages and server-side ActionScript APIs, and execute a server request to the client. The client

side can be looked at as a fully functional interface that sends commands to the server.

The server side application handles all commands and requests received from the client. During the non-recording mode, called the student mode, the client Flash movie opens a two-way communication channel with the Flash Communication Server MX installed on the server. When the connection is successful, the client movie sends the drawing commands to the communication server, which in turn broadcasts it to all connected clients.

When a user logs in after the start of a session, and drawings are already on the board, they will lose any voice or text chatting that has been made in his/her absence, but the communication server updates the new client's Flash movie with whatever has been drawn on the whiteboard, exactly as a real classroom.

The voice chat is a pre-made component, developed by Macromedia that can be downloaded from the Macromedia website. This component uses the open communication channel to transfer or stream the voice over the network. For every connected user, Flash Communication Server MX creates a streaming file on the server for the streamed voice from that user. The file extension is FLV.

Recording a lecture is a complex process involving other technologies in addition to what has been previously presented. Since the voice is handled through the Flash Communication Server, the recording of the voice must also go through it. The Flash Communication Server has the ability to record the voice stream and save it in the FLV file format.

When the instructor issues the start recording command, the client movie sends all drawing commands issued by the instructor, or received from any of the users, to an ASP file. The main purpose of this ASP file is to accept a string containing drawing command details, open a text file on the server and store the received command. The commands received by the ASP

file are written in a special format called Media Markup Language (MML) developed by Blue\*Pacific [9]. These drawing commands are stored in a text file that is given the EMT extension.

Once the recording session is done, the Flash movie issues a command to another ASP file and starts the process of creating a Flash movie for playback. It ignites the ASP Turbine [9] component to generate the required SWF file.

The ASP Turbine reads a Flash template and the created EMT text file. It then parses and executes the drawing commands, and attaches them with the template to generate a compiled SWF file. Figure 3 shows the described system architecture. The ASP Turbine is a server side component that generates dynamic Flash movies through ASP code, using Media Markup Language commands, an XML like markup language.

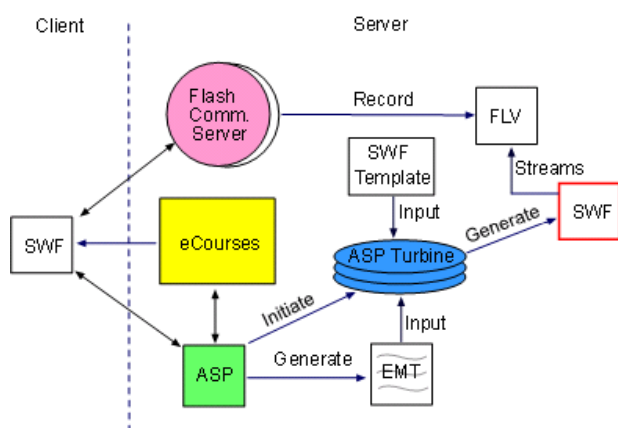


Figure 3: Architecture diagram of the LectureBoard application

The final result of the recording process is three files: an EMT file that is kept for future use, an FLV file that contains the voice recording, and the SWF containing the drawings recording and the necessary scripting code to stream the FLV file for playback. This process creates an SWF file without the need for Flash authoring program to build and publish Flash lectures through the use of ASP Turbine.

## Summary and Conclusion

An online lecture creation tool, called LectureBoard, was built and integrated with the eCourses system to allow online collaboration and lecture creation. The LectureBoard tool was developed using Macromedia Flash MX. It enables instructors and students to hold an online lecture that is provided with a drawing board, voice and text chat, and recording controls in order to record the session for later playback.

The LectureBoard has two parts: client side and server side. Communication between the client movie and the server application is possible through Flash Communication Server MX. It operates in two modes; student mode and instructor mode. In the student mode, all the functions are made available to the user except recording. With the instructor mode, all the functions are made available including session recording. The LectureBoard was integrated with the eCourses system, and can be accessed under the administrative menu list of the instructor log-in.

Although the LectureBoard was used for collaborative online lecture creation, there are two more possible uses for it: the first one is for online office hours meeting tool without recording. This tool can also be used by instructors and the students to meet online for an office hour session. In this case, both students and instructors can log in automatically in the student mode and no recording is required. Since no recording is involved, there is no archiving required. The second possibility is to record online office hour sessions. There is no difference in the interfacing and functionality than when used to create online lectures except in the archiving process.

## Acknowledgment

The authors gratefully acknowledge the support of this work from the National Science Foundation through Grant #0212224, "Transfer Facilitation for Engineering Students".

## Bibliography

1. Development of Web-based Distance Learning Environment Using Database, Master's thesis (2000), Tze-Leong Yew, The University of Oklahoma, p. xii.
2. Nicole Ridgway, A safer place to meet (2003), FORBES.
3. Internet-based Distributed Collaborative Environment for Engineering Education and Design, Master's thesis (2001), Quili Sun, The University of Oklahoma, p 5.
4. Centra® website: <http://www.centra.com/>
5. Net Meeting 3, <http://www.microsoft.com/windows/NetMeeting/default.ASP>
6. Belshazzar: A Simple Tool for Creating and Publishing Voice-Annotated Drawings on the Web, ASEE 2003 annual conference.
7. Macromedia Flash MX, <http://www.macromedia.com/software/flash/>
8. Flash Communication Server MX, <http://www.macromedia.com/software/flash.com/>
9. ASP Turbine Media Markup Language, Blue\*Pacific company, <http://www.aspturbine.com/products/aspturbine/docs/asp/mml.htm>

## Biographical Information

Mohammad AlRamahi received his M.S. degree in Mechanical Engineering from the University of Oklahoma. He completed his B.S. in Electrical/Computer Engineering in spring 2000 at J.U.S.T University, Jordan. He worked for 8 months as a software engineer for ONEWORLD Software Solutions in Jordan before starting his graduate program in U.S. His present work involves research on collaborative and online learning for engineering education.

Kurt Gramoll is the Hughes Centennial Professor of Engineering and Director of the Engineering Media Lab at the University of Oklahoma. He has developed and published CDs and web-based sites for engineering education, K-12 instruction, and industrial training. Dr. Gramoll received his B.S. degree in Civil engineering and M.S. degree in Mechanical Engineering, both from the University of Utah. He received his Ph.D. in Engineering Science and Mechanics from Virginia Tech. Previously, he has taught at Univ. of Memphis and Georgia Tech.