

E-Seminar

Jie Zhang
Department of Computer Science
Brown University

Submitted in partial fulfillment of the requirements for the Degree of Master of Science
in the Department of Computer Science at Brown University

Signature (Professor Steve Reiss, Project Advisor)

Date

Introduction

Imagine a web site that hosts all sorts of interesting seminars. New topics are posted everyday, covering virtually anything you can think of. When you see a topic that interests you, you simply mark your calendar and put aside one hour for it. When the time comes, you turn on your computer, go to that web site, and 'enter' the seminar room. The speaker navigates you through the slides as she explains the topic. Whenever you have a question, you type it into a text field. Then the question will be posted onto the question board along with questions from other audience. The speaker will answer the question when she sees it. More discussions are developed as the seminar goes on... When the seminar comes to its happy ending, you are pretty satisfied with what you have learned and decide to come back later. Meanwhile, you are weighing the idea of posting your own seminar on this site someday ...

What described above is an online seminar site – a website where people can post seminars, present slides, explain ideas and share knowledge. Anyone, from anywhere around the globe, can attend any seminar through this web site. Without leaving one's desk, the audience can listen to some speaker thousands of miles away, interacting with one another and learning about new and interesting topics.

This virtual seminar room simulates its real world counterpart in many aspects such as the following:

- ★ A speaker leads the session. That is to say, the speaker is responsible not only for the content of the topic, but also the control of the tempo of the seminar and answering questions from the audience.
- ★ The seminar is live (in real time). Like in real life, both the speaker and the audience are present at the seminar site at a pre-set time slot. Those who join late will see a seminar that is already in progress, but they will be synchronized to the same slide at their entry time.
- ★ Continuous interactions happen between the speaker and the audience. Those interactions include: questions from the audience, feedback from the speaker, and live discussions.

Despite their similarities, the virtual seminar room is far more advantageous than the one in the real world. The fact that it is constructed in the cyberspace grants instant accessibility to a global audience through the omnipresence of the Internet nowadays. Therefore:

- ★ The geographical restrictions that separate people no longer exist. Anyone who shares the same interest can come together, without the hassle of costly planning and traveling that are the trademarks of conventional seminars.
- ★ In the virtual seminar room, the audience member can interact with one another far more freely without disturbing the peace. He or she could type in questions and comments throughout the seminar. The rest of the audience can choose to read or ignore the information as the seminar progresses.

- ★ The virtual seminar site is flexible in size and could accommodate any number of audiences. Within the limit of the computer resource and network capacity, the site can theoretically handle various numbers of visitors.
- ★ It requires much less maintenance than its counterpart in the real world. Most operations, such as seminar schedule and email alert, can be handled automatically by intelligent computer programs. Human supervision is needed at a minimal level.

In order to realize the great potential of the virtual seminar described above, one needs to examine the current situation in the application of Internet as a teaching and learning tool. The conceptual aspect of e-learning, as well as the currently available web technology will be introduced in the following section.

Current E-Learning Applications

Distant learning has been around for quite a while. Along with the development of telecommunication technologies, various distant teaching programs have been broadcasted, first on radio, and then on TV. Most traditional distant learning is asynchronous: the program has to be pre-recorded before the broadcasting. Even when the broadcast is live, there is no easy way for the student at home to contact the instructor in the studio, even more difficult for the instructor to monitor the learning progress of the students. The lack of interaction is the major obstacle in achieving satisfactory teaching and learning experience.

It is no surprise that personal computer attracted much attention from the people in distant learning area as it is becoming more and more widely used at home and office, especially after the Internet take-off a decade ago. As a result, e-learning has joined the family of distant learning.

The advantage of e-learning comes from its capacity of achieving real-time learning experience. The interaction between the instructor and the students can be delivered via the Internet in the form of audio or video data at negligible time delay. In addition, computer resource sharing enables the instructor to monitor the class and adjust the teaching schedule accordingly. A virtual classroom is achievable in an e-learning environment.

A few e-learning applications, such as Centra Symposium and Mentergy LearnLinc, have been developed in recent years. To achieve the same goal of building a virtual classroom over the Internet, they have very similar features in their applications. In the next section, a case study of LearnLinc will present a more detailed picture of current e-learning status.

A Case Study of LearnLinc

Developed by Mentergy, Inc., LearnLinc is designed to be a live virtual classroom environment that enables corporations to deliver live e-learning courseware to employees or students via the Internet. Similar to other e-learning applications, LearnLinc targets at corporate training. The corporate provides the teaching content, and the team in Mentergy integrates the content with the above-mentioned tools and provides instructor-training and broadcasting services plus any technical support.

Such approach in e-learning applications could be a good solution to corporate training. For example, if there is a major software update to the inventory control system in a multi-national corporation, a well-designed LearnLinc training session can quickly bring all concerned employees scattered in hundreds of branches to the same skill level. It will certainly be more cost-effective than sending hundreds of instructors to all the branches. Ideally, every employee can take part in the session from their desktop, which will fit better into their routine working schedule. The interactive learning tools offered by LearnLinc include:

- ★ Communicate with students via two-way audio or video conferencing.
- ★ Teach the class by synchronizing PowerPoint, multimedia, or web-based content on all student screens.
- ★ View hands raised and give students control to ask questions, run courseware, or take the entire class to a Web site.

- ★ Demonstrate or share individual software applications or an entire desktop for interactive software skills training.
- ★ Record entire LearnLinc classes and playback anytime as self-paced content.
- ★ Create breakout groups on-the-fly to facilitate team interaction.
- ★ Have an Assistant Instructor monitor the classroom and interact with students privately.
- ★ Launch streaming video to deliver live presentations or pre-recorded video clips.

However, because of its underlined corporate training orientation, LearnLinc doesn't appear to suit other e-learning need, such as the online seminar mentioned at the very beginning of this paper. Several drawbacks of LearnLinc include:

- ★ Lack of flexibility. The teaching session has to be created by the personnel from the corporate and the technical team at Mentergy together. Neither party has enough tools or content to create a standalone teaching session. Each session has to be customized.
- ★ Complicated technical issues. All the tools are nice and powerful, but those hotkeys and buttons are not as intuitive as the chalk and board. Each instructor, even the audience, has to be trained pre-session. Not to mention that a technician from Mentergy must stay on-call to take care of any technical emergencies during the training session.
- ★ Limited scalability. LearnLinc has a 25 seats (including instructor and assistant) limit for each session. This is quite a limitation compared to real life class that can vary from several participants to over a hundred.
- ★ A closed system. This kind of system actually works fine for corporate training. After all, it won't be necessary to make the training session open to people outside the company. However, there will be difficulties if we want to transplant this system into an academic setting. More often than not, the class is open to students from different groups. A closed system is unimaginable during the two-week class-shopping period.

A more desirable e-seminar site should be an open system emphasizing knowledge sharing. Virtually anyone should be able to attend any seminar that interests her without any complicated setup or technical assistance. The role of the participants should be interchangeable, today's audience could very well be the speaker tomorrow. In the next section, the author elaborates on the model and implementation details of such an online web-site.

The Seminar Web-site

➤ Internet is the ideal media.

In the Seminar project, the Internet is where the virtual seminar room resides (**Figure 1**). The Internet is the ideal media for the “Virtual” seminar room for the following reasons:

- 1) The Internet is the only media that has an open structure. People of different geological, linguistic, and cultural background could share information and ideas seamlessly. Old boundaries no longer exist in cyber space.
- 2) The Internet is the only media that is capable of reaching every remote corner of the world with minute delay at little cost.

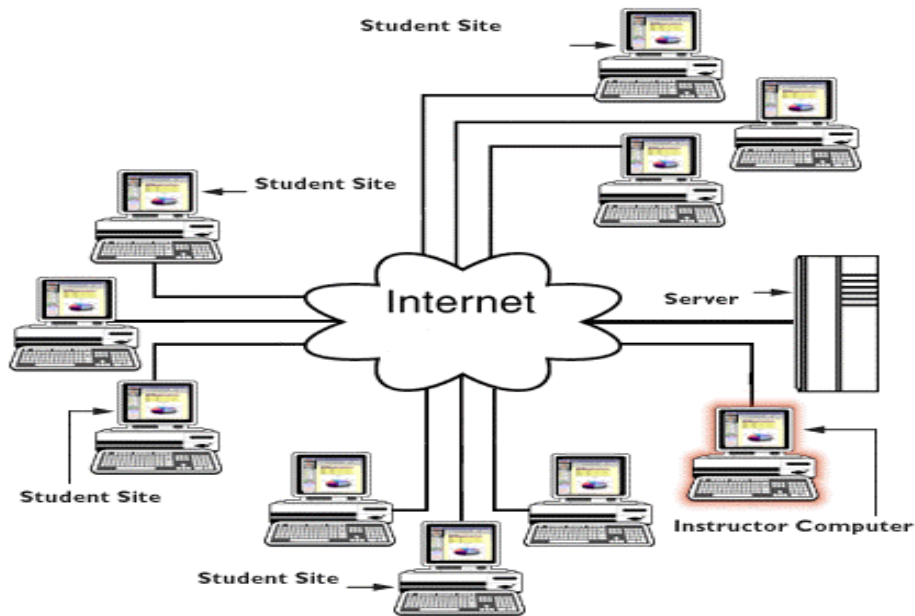


Figure 1 A virtual seminar room.

➤ Client/server structure is the workhorse.

A client/server structure is utilized to synchronize several events, including:

- 1) Keep everyone on the same slide. Anyone that is late will have her screen synchronized to the same status as the rest of the audience.
- 2) Post a question onto the shared question board.
- 3) Deliver the voice communication in a chat room fashion.

A participant of a seminar can take on one of the two roles: the speaker or a member of the audience. The flow chart displays the cycle of a seminar and the tasks for each role (**Figure 2**).

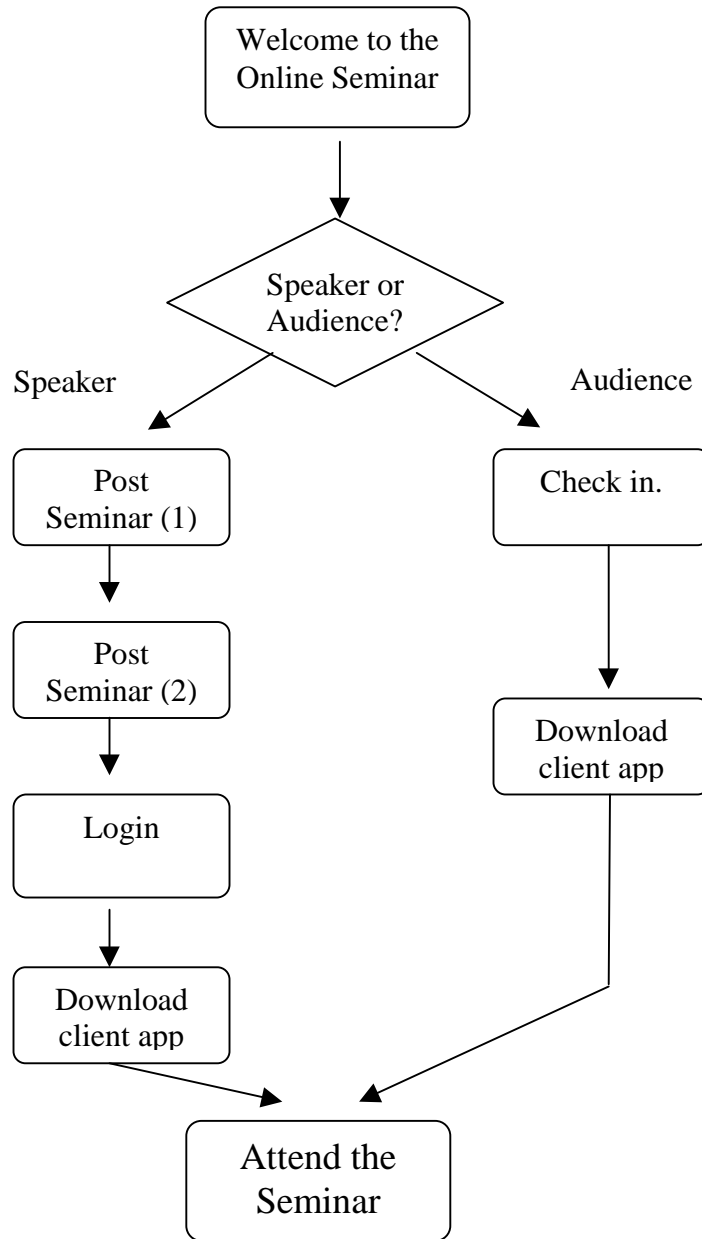


Figure 2 The flowchart of a seminar.

➤ **The speaker**

The speaker of a seminar is the person who initiates and gives the seminar, her tasks include:

- 1) Post seminar information.
In the step of “Post Seminar (1)”, the speaker fills in a form to introduce herself, the seminar topic and abstract and desired time, etc.
- 2) Upload slides onto the seminar server.
This is accomplished in “Post Seminar (2)” step.
- 3) Download the self-extracted client application and fire it up.
- 4) Give the presentation.
- 5) Control the slides.
- 6) Control the discussion.

➤ **The audience**

Anyone interested in listening to a seminar logs into that seminar and becomes a member of the audience. Her tasks include:

- 1) Download the self-extracted client application and fire it up.
- 2) Listen to the speaker.
- 3) Ask questions in text format.
- 4) Join voice discussion under the control of the speaker.

The Implementation of the E-Seminar Project

In order for the speaker and the audience to accomplish their tasks and obtain satisfying knowledge sharing experience, this project consists of three components:

1. A user-friendly web interface.

The top section displays the topic, speaker information, abstract and time of the immediate next seminar. Speaker and audience can login using the links underneath.

The middle section displays the link to the archives of past and future seminars. Under the discretion of the speaker, past seminar slides can be saved on the seminar server for future browsing. Information about future seminar, such as speaker, topic and abstract, can be obtained by clicking on the title link. The information on this page is updated by server side included commands and Perl scripts which we will discuss later (**Figure 3**).

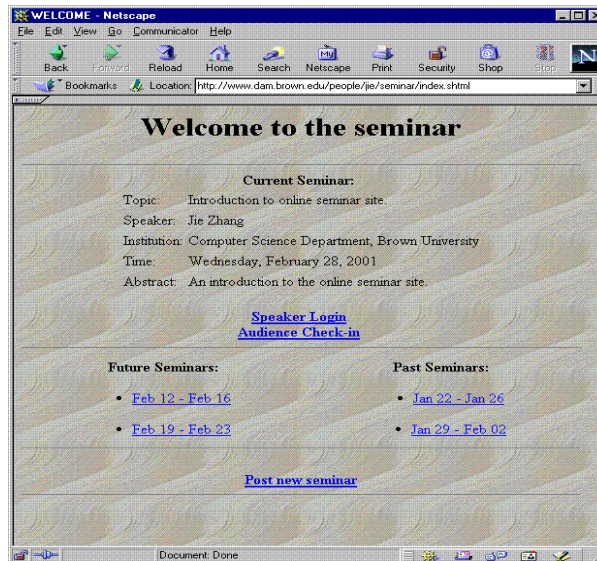


Figure 3 Welcome to the Seminar.

The link at the bottom of that page is for potential speaker posting new seminars. From there, a person who wants to give an online seminar can follow the instructions to easily set up a seminar (figure 4).

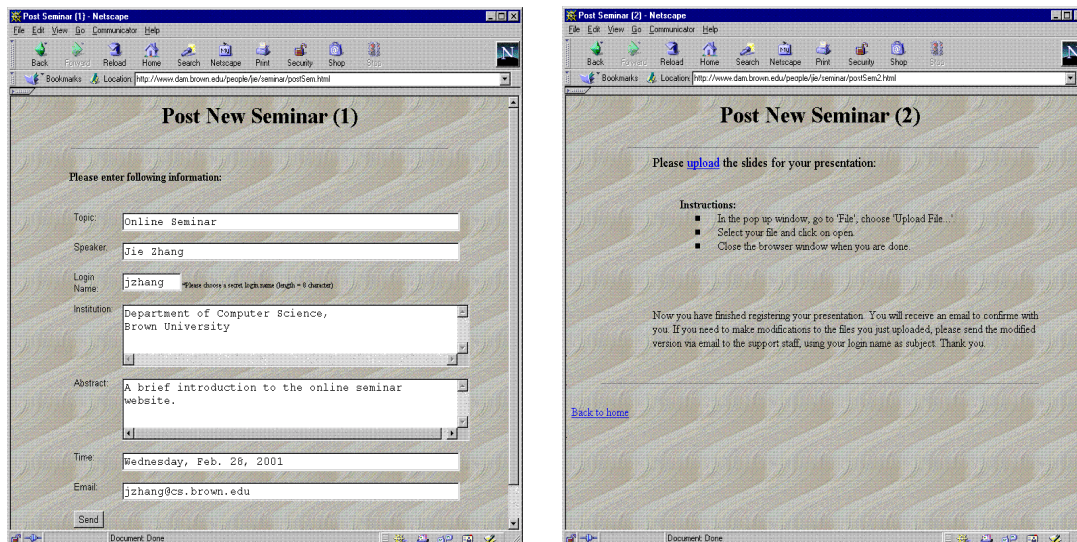


Figure 4 The speaker posts a seminar.

The left page shows the form that the speaker is required to fill in. In that form, the speaker provides information about the topic, the speaker name, the secret login name for authentication, seminar abstract, desired time and her e-mail address.

When the speaker clicks on the “Send” button, the information contained in this filled form is sent to the seminar site administrator automatically. FormMail is used here to accomplish this task. For this form, FormMail is also configured to check if each field is

filled properly. For example, the secret login password field must be filled and the e-mail address should be in the right format. The speaker will be shown a confirmation page to verify the information.

The page on the right helps the speaker to upload her slides onto the seminar server. There are detailed instructions in the middle section. This uploading is done by 'ftp' the slides to designated directory on an ftp server. File type checking is employed for security concern.

By now, the speaker has finished her preparation for the seminar. All the tasks are straightforward and easy to accomplish with the clear instructions.

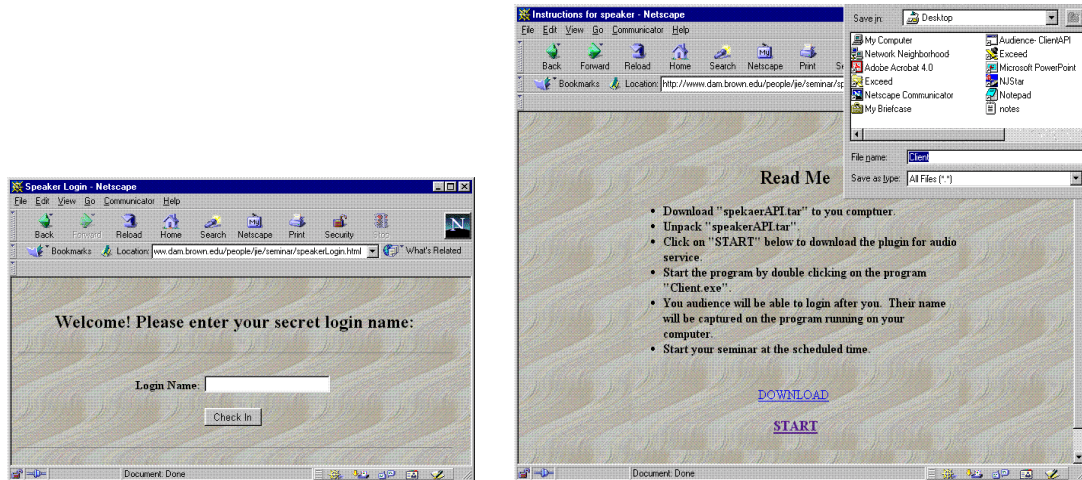


Figure 5 Speaker login and client application download.

Figure 5 shows the login window for the speaker and the client application download page for the speaker and the audience. On the left, the speaker is required to authenticate herself by the secret login password she chose in the post seminar step. The audience will be led to the page on the right when they check in. Again, clear instructions about how to download and fire up the application are given. Essentially, the participants need to click on the two links, first on "DOWNLOAD", then on "START".

2. A Java application that implements the client/server structure.

The Java application is the workhorse of this project. Client applications are running at the speaker and the audience side, while a server application is running on the seminar server. Sockets are established to connect the clients and the server (**Figure 6**).

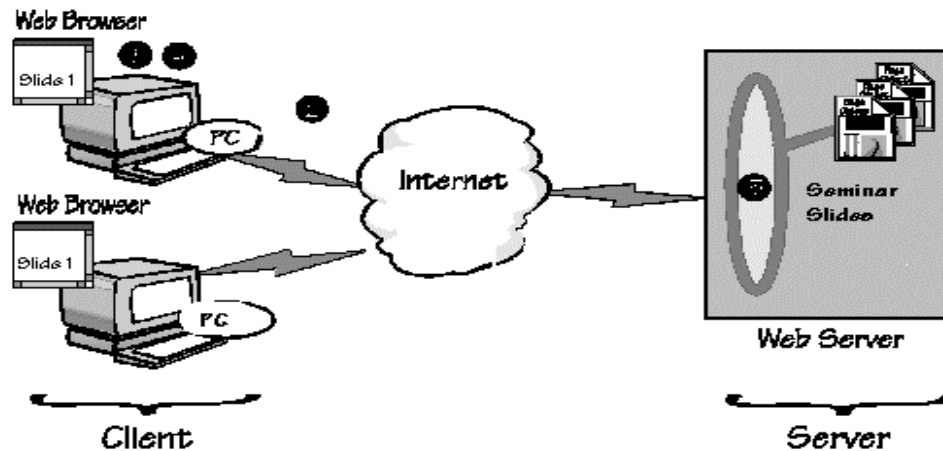


Figure 6 The client/server structure.

The Client

```
public class Client extends Thread {
    //member variables defined here
    //constructor
    public void run() {
        try {
            sock = new Socket("LOCALHOST", 8081); //server application runs on the localhost during test
            System.out.println("connection made");
            in = sock.getInputStream(); //initialize data streams
            out = sock.getOutputStream();
            pout = new PrintStream(out);
            din = new DataInputStream(in);
            dout = new DataOutputStream(out);
            while(true) {
                String request = din.readLine();
            }
        }
        catch (UnknownHostException e) {System.out.println("can't find host"); }
        catch ( IOException e) {System.out.println("Error connecting to host");}
    }
    //more functions
} // end of client class
```

The Server

```
public class SeminarServer implements Runnable {
    ThreadGroup dealGroup;
    ServerSocket serverSocket;
    //more member variables here
    //constructors here
    public void start(){
        dealGroup=new ThreadGroup("DealThreads");
        kicker=new Thread(this);
        kicker.start();
    }
}
```

```

}
public void run(){
    try{
        serverSocket=new ServerSocket(port);
        info_print("\nSeminar Server running on " + hostname + ":" + port);
        while(true){
            Socket clientSocket=serverSocket.accept();
            DealThread dt = new DealThread(dealGroup, this,clientSocket, ++number);
            dt.start();
            listeners.addElement(dt);
        }//end while
    } catch(IOException e){
        e.printStackTrace();
        info_print("Errors in opening ServerSocket\n");
        System.exit(1);
    }//end catch
} //end run
//take client input and output to every client who is online
public void broadcast(String s) {
    Enumeration elements = listeners.elements();
    while (elements.hasMoreElements())
        ((DealThread) elements.nextElement()).sendData(s);
}
public synchronized void addUser(String s){
    info_print("add new user: " + s);
}
public synchronized void delUser(String s){
    info_print("remove user: " + s);
}
}
public static void main(String[] args) {
    SeminarServer SeminarServer=new SeminarServer();
    if ( args.length > 0 ) SeminarServer.port = Integer.parseInt ( args[0] );
    else SeminarServer.port =DEFAULT_PORT;
    SeminarServer.init();
    SeminarServer.start();
} //end main
} //end SeminarServer

```

The Client is a subclass of the generic thread class. Every client thread instance is added into an instance of ThreadGroup class after the socket connection between the server and the client is established. The ThreadGroup instance relays the broadcast communication to each group member from the server.

The ClientAPI

A modal dialogue box, Username window, is popped up before the ClientAPI can be accessed (**Figure 7**). The username is used more for naming convenience than authentication.

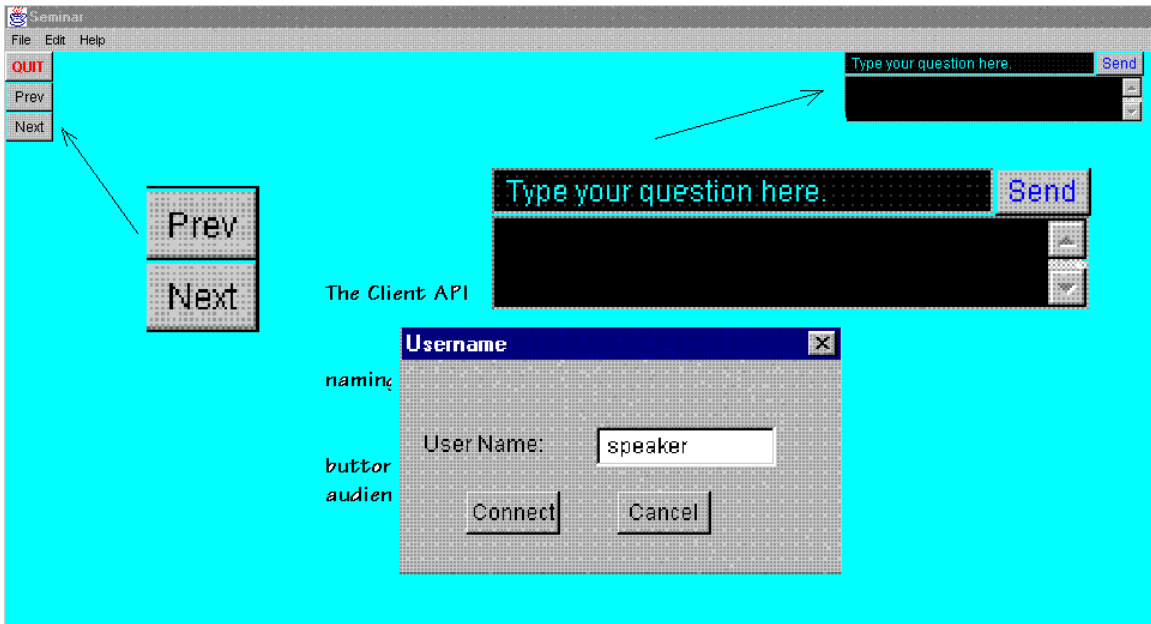


Figure 7 ClientAPI.

The ClientAPI shown above is for the speaker. There are two slide control buttons on the left upper corner. They are disabled in the ClientAPI for the audience for obvious reason.

On the right upper corner, there is a text field and a multilane text area (the question board). Any attendant of the seminar can type in their comments or question into the text field. The text then gets posted onto the question board and displayed on everyone's screen.

The Audio Communication

The speaker's voice and any voice discussion is accomplished via a free chat room service provided by Telcopoint/hearme (<http://www.telcopoint.com>). When the speaker and the audience click on the "START" button on the "Read Me" page (**Figure 8**) right after they login, they "enter" the seminar room. Any of them can hold down the 'ctrl' key and start talking to the rest of the people in the chat room. After he/she finishes and releases the 'ctrl' key, the next person can start talking. Usually, the speaker will control the discussion by naming the next person to speak. If someone has a question, sending it to the question board will be the surest way to get 'heard'.

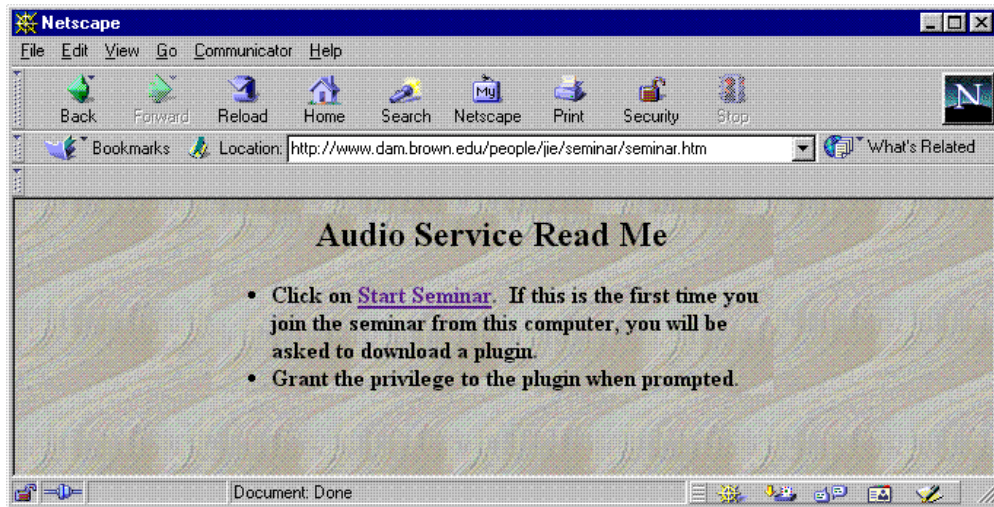


Figure 8 Instructions for the audio service set up.

3. A set of Perl scripts automates the web-site management.

In order to keep the seminar web-site functioning, there are some routine tasks that should be done throughout each seminar cycle:

1. Process all the e-mails requesting to post a seminar.
2. Create the seminar introduction page.
3. Create the application initialization file.
4. Put the image files into the right web directory.
5. Update the site content after a seminar.
6. If necessary, negotiate with a speaker about the seminar content and schedule.

Although what listed above seems to be a lot of tasks, these tasks can actually be broken down into two categories: those involved in adding a seminar and those involved in updating the seminar site. Two Perl scripts, SemAdd and SemUpdate, were written to relieve the web master from those tedious maintenance routines.

As their names suggest, SemAdd handles the tasks involved in adding a seminar, namely, tasks 1 through 4:

```
#!/usr/local/bin/perl
use Getopt::Std;
getopts('t:n:i:a:d:l:e:');
#options
#t - topic
#n - speaker name
#i - institute
#a - abstract
#d - date
#l - login name
#e - email address
$SOURCEDIR='/sparc_soft/ftp.incoming/jie';
$TARGETDIR='/www.dam/htdocs/people/jie/seminar';
#transform input into text string
$input="<table><tr><td><b>Topic:</b></td><td>$opt_t</tr><tr><td><b>Speaker:</b></td><td>$opt_n</tr><tr><td><b>Institute:</b></td><td>$opt_i</tr><tr><td><b>Abstract:</b></td><td>$opt_a</tr><tr><td><b>Date:</b></td><td>$opt_d</tr><tr><td><b>Login:</b></td><td>$opt_l</tr><tr><td><b>Email:</b></td><td>$opt_e</tr></table>";
```

```

#make a new directory for file copying
$targetDir=`cat /www.dam/htdocs/people/jie/seminar/index_current` +1;
print "make new directory $targetDir\n";
system("mkdir /www.dam/htdocs/people/jie/seminar/$targetDir");
#create the index.html file in the target directory,
#which will be shown in the welcome page
system("touch /www.dam/htdocs/people/jie/seminar/$targetDir/index.html");
open(P, ">>/www.dam/htdocs/people/jie/seminar/$targetDir/index.html");
print P $input;
close(P);
#add to the list of seminars
open(P, ">>/www.dam/htdocs/people/jie/seminar/seminar.list");
print P $opt_t;
print P "\n";
close(P);
#add to the list of speaker login info
open(P, ">>/www.dam/htdocs/people/jie/seminar/login.list");
print P $opt_l;
print P "\n";
close(P);
#copy the image files into the target direcotry
system("cp /sparc_soft/ftp.incoming/jie/* /www.dam/htdocs/people/jie/seminar/$targetDir/");
#create IMAGE file (txt)
system("ls /sparc_soft/ftp.incoming/jie > IMAGE.txt");
#create the tar file for user to download.
system("tar cvf audienceAPI.tar IMAGE.txt client-a.exe");
system("tar cvf speakerAPI.tar IMAGE.txt Client.exe");

```

After each seminar, SemUpdate is run to update the content in the welcome page of the site. Both the current seminar introduction, the archives for future and past seminars and the speaker login page will be modified to reflect the most up-to-date status.

```

#!/usr/local/bin/perl
#update the index_current
$next_index=`cat /www.dam/htdocs/people/jie/seminar/index_current` +1;
open(P, ">>/www.dam/htdocs/people/jie/seminar/index_current");
print P $next_index;
close(P);
#update the speaker login info
$current_index = $next_index -1;
open(P, "<</www.dam/htdocs/people/jie/seminar/login.list");
$i = 0;
while(<P>){
    $i = $i + 1;
    if ($i == $current_index) {
        my($line) = $_;
        chomp($line);
        $oldLogin=$line;
    }
    if ($i == $current_index+1) {
        my($line) = $_;
        chomp($line);
        $newLogin=$line;
    }
}
close(P);
system("perl -pi.bak -e 's/$oldLogin/$newLogin/;' speakerLogin.html");
#update view_past
system("perl viewPast");
#update view_future
system("perl viewFuture");

```

Conclusions

In this fast-growing information age people rely on seminars to get informed and stay current. Seminar as a teaching and learning tool has never been more popular. The purpose of this project focuses on the design and implementation of a real-time seminar room on the web, which can greatly facilitate the sharing of knowledge online by bringing the seminar room into everyone's desktop.

The online seminar site provides the convenience as well as the content information to the masses. It has potential to transcend the way seminars are being conducted. Using Internet as the media for knowledge sharing has great potential in the future of e-learning. The far-reaching capacity of world wide web has brought people closer than ever before. In the open structure of the Internet seminar site, information flows effortlessly from one speaker to a dynamic group of listeners.

In order to take full advantage of such a concept, efforts have been made to make the site access easy and friendly to users. For instance, the speaker only needs to follow several easy instructions to prepare for the seminar. There is little burden on the audience side. The operations on the user interface are apparent and intuitive, therefore most people could use it with little or no training required. The system requires moderate computing resources that are available to most people with personal computers.

On the technical aspect, the major advantage of this project is the seamless incorporation of various techniques to serve a common purpose. In addition to the main programming and scripts design, this project also effectively uses various tools that are available and free to the public. Such approach cut the cost of development and therefore is more efficient than doing everything from scratch.

References:

- Anderson, Cushing; *ELearning in Practice, Blended Solutions in Action* (An IDC White Paper sponsored by Mentergy Inc.) 2000
- Anderson, Cushing *ELearning in Practice, Three Case Studies Sponsored by Mentergy Inc.* 2000
- Navickas, Leon; *Collaboration 2.0: Revolutionizing eBusiness Processes Through Live Collaboration over the Internet* Feb. 2000