An Educational Site on the World Wide Web

L.J.M. Rothkrantz, R.J. van Vark, A.J.S.I.K. Bos, and D.J.A. Pauw

Delft University of Technology
Faculty of Technical Mathematics and Informatics
Zuidplantsoen 4, 2628 BZ Delft, The Netherlands
email: {L.J.M.Rothkrantz,R.J.vanVark,A.J.S.I.K.Bos,D.J.A.Pauw}@cs.tudelft.nl

Abstract

At the Delft University of Technology, a project has been started aimed at the creation of an educational site on the World Wide Web for remote learning. This paper describes a prototype of the site using an introductory course in Neural Networks which has recently become available. The described educational site does not only provide plain texts concerning the course, but also provides a number of additional features to improve interaction and communication between students and teachers. The site is developed using publicly available software like Perl and Java. The course has been tested during the last year. In this paper, design and implementation of the educational site are reported as well as the results of the evaluation.

1 Introduction

In recent years, the interest in the Internet and the World Wide Web (WWW) has been growing rapidly. This growth has not gone unnoticed within the educational environment as most universities were among the first institutes to use similar Internet-like facilities when they became available in the early seventies. Nowadays, it is possible for almost anyone to gain access to the Internet, of course including many universities and their students.

At this time, the facilities which make up the Internet are suitable to fulfill educational goals. Using the Internet results in better utilisation of computer facilities, as computers in student dorms become available within the educational environment. Besides better computer utilisation, it is also possible for students to do courses at other universities using educational websites. Of course, students can communicate with teachers from many different locations using the Internet.

The computer has been part of the educational environment for many years. At this moment, many traditional courses are presented to students applying computer assisted instruction and computer based learning. Creating an educational site on WWW has to attribute to existing courses and educational methods. The enhancement of courses with an educational site has to maximize the following two characteristics of the Internet:

- Internet is an excellent medium for distributing information. Most universities and research institutes and many private companies and governmental organisations provide public access to their libraries and Internet sites.
- Internet is an appropriate medium for communication. Besides email and WWW, there are discussion groups, so called usenet newsgroups, which offer information concerning many different topics.

Therefore, the Internet and especially WWW can be used in the educational domain to distribute lecture notes, manuals, software, etc. Course texts can be presented using HTML, i.e. hypertext mark-up language. This is a mark-up language which can be used to structure text and combine any number of pages into a framework which can be used to educate students. In these courses different media can be used: text, sound, vision, etc. Software used in the courses can be launched from different machines at remote locations.

However, the most intriguing part in the educational environment is based on the communicative aspects of the Internet. In a traditional educational environment, interaction between teachers and students is difficult to realise on a regular bases, caused by crowded classes and busy teachers. Using Internet facilities, communication between teachers and students can be easily improved. Developers of educational sites are challenged to create courses which are user friendly and which especially provoke communication between teachers and students. An attempt to create such an educational site will be described in the following sections.

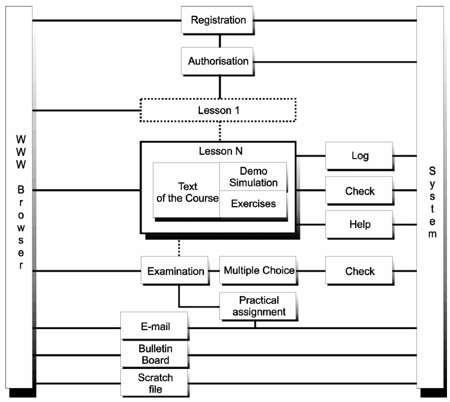

Figure 1: Structure of a course on the educational site

2 Educational goals

We believe that you can learn computer science by applying it in practice. Reading books and lecture notes gives some general understanding in theories underlying modern computer science. For deep level understanding and learning, some elaboration of the course material is necessary. For these reasons most courses include additional classes to make exercises or to work in computerlaboratories. We want to integrate these activities. But to be realistic, offering theory and exercises simultaneously does not imply that students integrate these activities spontaneously. So we need some motivating activities and control mechanism to realise that goal.

- We assumed that the WWW is a motivating environment and we used some additional software and simulation activities which proved to be motivating in laboratory rooms.
- We check the activities of the students. The login time is logged and students have to make exercises individually. Every week they get a report (automatically) about their study progress and the time that they have spent on the course. Of course we realize that we have less control of the activities of the students during the time that they are logged in. However, we can log which pages have been looked at.
- Students have to do some group work and they have to write a common report. This implies that they have to communicate to each other by email or by bulletin board.

3 Model

A student receives a user name and a password for authentication purposes at the start of a course which he receives by email as soon as he registers by filling out a form. The backbone of the course consists of the HTML pages with the text of the lessons, basically resulting in an electronic textbook.

After authentication, the student can navigate freely through the course. All the students' actions are logged, so the teacher can check progress of the students. In figure 1 a scheme of the course is presented.

While reading the text the student is requested to launch demo programs and simulations which are short demonstrations of the theory. This provides for a natural integration of theory and application. Every lesson has a number of exercises during which the student can ask for help and can ask for correction of his solutions. If needed he can communicate with other students or the teacher by using the bulletin board or by using email.

After finishing the course the student can request for examination. This examination can be done by filling out an electronic multiple choice exam. After completing the exam the student can ask for correction and results of the exam will be presented immediately. Another option is that the student gets a practical assignment accompanied with additional text and software. After completing the assignment the report has to be sent to the teacher by email.

4 Design Aspects

The development of the educational site is based on several design aspects which will be described in this section. Internet can be used as an independent machine platform. By using WWW browsers, one is able to use the same kind of user interface on different operating systems. The WWW browsers that can be used for following our cybercourse have got two restrictions. First they should be capable of handling Java applets and second they should be able to handle frames. An important aspect is the choice between remote versus local processing. Of course, the modular organisation of the pages and navigating within a course is also considered important.

4.1 Remote versus Local Processing

When the course development was started, the course was intended to be as platform independent as was reasonably possible. The easiest way to achieve platform independency is to write applications which run on the web server, making them independent of the student's browser and its platform. This way every browser, regardless of platform, would be able to run these applications on the server.

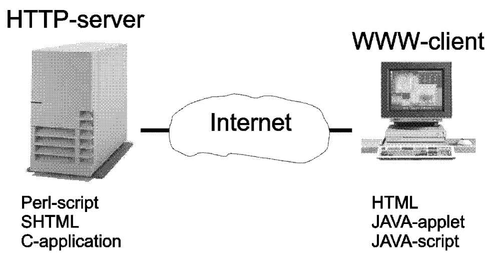

Figure 2: Load balance between server and browser

The disadvantage of the above method is that it places a rather computational intensive load on the server. With Java it is possible to build applications, better known as applets, which run inside each Java-capable browser at the remote WWW client. In this way the server load will be significantly less compared to the use of server side applications (see figure 2).

4.2 Modular Structure and Routing

The HTML pages of the course are structured modular and are organised in a hypertext structure. To realise an optimal flexibility between modules a navigation tool is designed. Using this tool it is possible to reach all other modules from within a certain page. At every page a menu displays information of the current location in the course and how to reach other modules. This information is shown to the student as a horizontal and a vertical icon bar (see figure 3).

5 Security Measurements

Security measurements can be divided into the technical measurements (points one, two and three), and the formal measurements (points 4 and 5). For the technical measurements it is important to say that the WWW server runs on the Unix operating system.

1. Standard security measures available in the Unix operating system have been applied. One of these security measures is the use of groups labeled with 'owner', 'group', and the 'rest of the world'. This prevents access at the basic level.
2. Information is stored by using email, instead of direct writing to files by the HTTP-demon.
3. Used the already functioning NCSA HTTPD. This means for security the use of the 'htaccess' and the 'htpasswd' methods. This can be used to block out persons who have not the permission to access the site.
4. There is the formal procedure which implies that only students who have enlisted for the course can enlist for the WWW-course. The cybercourse appliance has to be done through the 'login' screen of the cybercourse. Several information items have to be given. The teacher can see the list of students who have sent their appliance. The teacher finally has to grant permission to the students. An email is sent automatically to the student when the permission is granted.
5. To be sure that students make the work individually there are some control mechanism. To do an exam on a distant place or time there has got to be a supervisor i.e. a teacher from another department or university who is willing to supervise.

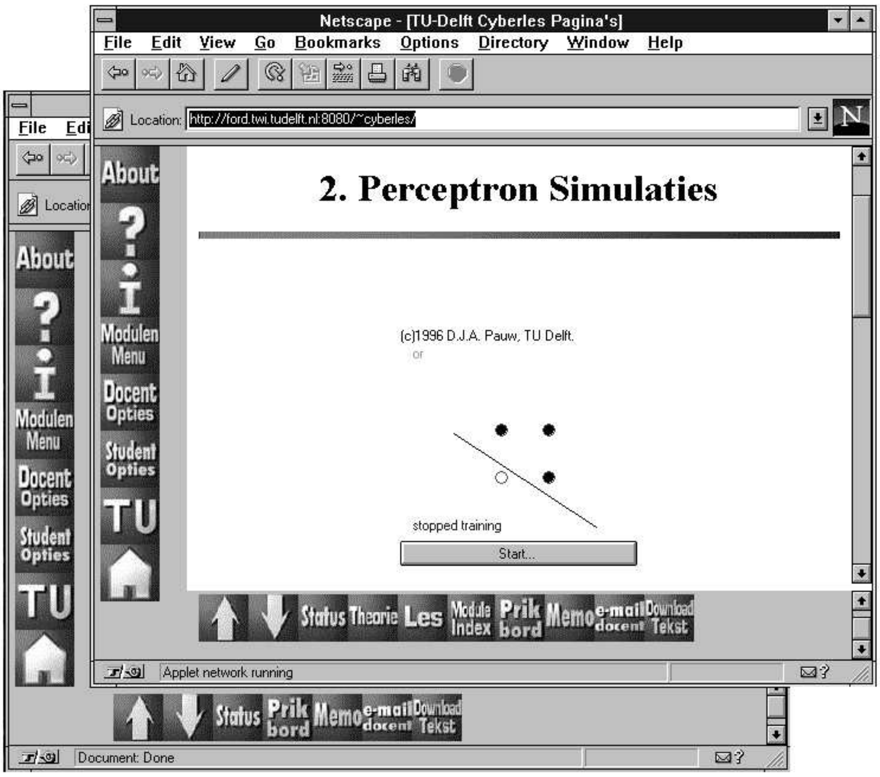

Figure 3: Navigating in a course

6 Implementation

This section describes tools used during the development of the educational site. Next the general setup is presented followed by the Train application to describe the techniques used. Finally some options available to teachers and students are given.

6.1 Software Tools

For the development of the course a number of software packages have been used.

- Graphical programs to create icons and pictures.
- Text editors to create and edit HTML pages. Various dedicated HTML editors were tested, but none contained all features to be included in the course pages. Therefore, pages were created using standard text editors.
- Programming languages like C, C++, Java, Perl, and Java-script to create scripts and demonstrations.

6.2 General Setup

The appearance of HTML pages is controlled by various CGI scripts, which have been implemented using Perl. Scripts exist for navigation, bulletin board, scratch file, progress monitoring, student-teacher options, authentication, logging and application software. Databases containing the information about students, student progress, and module information are plain text files. These files can be easily manipulated using Perl. Applying simple conversions, these files can be used to build HTML pages.

The educational site is installed on a NCSA HTTP server. Certain access control methods of this NCSA HTTP-server were used to control admission and authentication of students. An access control method exists for the complete directory structure and for some subdirectories. The access of students can be controlled by the teacher. In some applications students have to set certain parameters. For example, in the neural network module students have to choose the number of hidden layers and the number of neurons per layer. To prevent overloading of the web server, special Java scripts have been developed to keep the parameters within certain limits. The application invoked by these scripts is written in C. Most of the other applications and demonstrations have been written in Java.

6.3 Train application

The Train application was already available as a DOS based program. This program was rebuilt to run on the HTTP-demon. In order to achieve this we had to remove the user interface and rewrite the source into ANSI-C, which could be compiled on the server. Software which is highly interactive, and which does not have a separate "input" and "output" phase within the flow of the program, is not good for a try to rewrite the application to a server-side application.

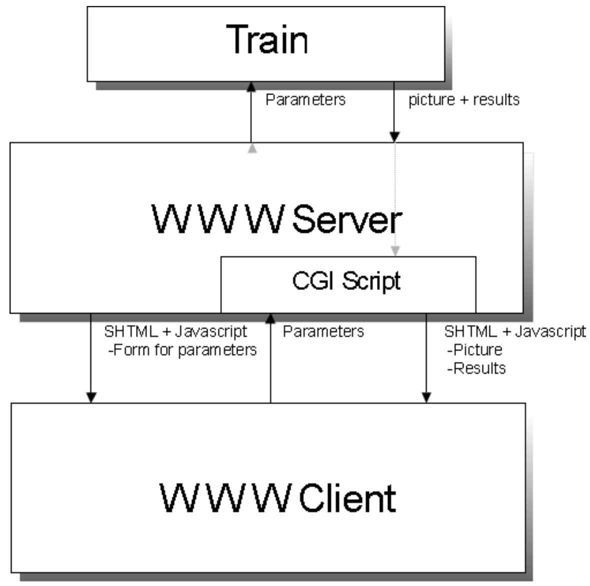

Figure 4: Schematic overview of Unix Train example

The input for the server-side "Train" application (see figure 4), comes from the WWW-client. This goes into a CGI script, and functions as input for "Train". Train runs with the given parameters on the WWW-server. The results are given to another CGI script which gives the data to a JavaScript. And finally the results are given in an SHTML file to the browser, which is able to display the results on screen.

Software which is highly interactive, has been written in Java from scratch. Java uses the processor of the remote WWW client. This implies a reduction of the WWW server workload.

6.4 Options

Several options for both teachers and students can be found in the prototype. For the teacher there are two main options:

- Adding a new module
  A tool has been created which can add the complete framework of a new module.

- Checking progress of students
  A progress indicator to monitor the progress of students has been built. The teacher can get information about which pages have been visited, which demonstrations have been launched and results of the exercises and exams for each student individually.

For the students the following options are available at the educational site:

- Scratch file
  This can be used by a student to make personal notes about the course he is following. Making a note is done by pushing a button, which activates a page with an editor. When the student is finished making notes he can store these notes in a personal directory on the server.

- Bulletin board
  On the bulletin board, students can pose questions, contact other students and make public notes which are shared with other students. This makes it possible to share reports of common practical assignments. It can also be used by the teacher to inform students.

- Change password
  When a student applies for a course, he is given a password of his own choice. For security reasons he can change this password himself.

7 Course Material

The first course which is developed as a course on the educational site is an introductory course in Artificial Neural Networks. This course was chosen because a lot of application software and simulations were publicly available. There was also a database of more than 100 multiple choice questions for examination.

In one of the applications users have to investigate how a simple neural network with one hidden layer can represent a function (see figure 5). The user starts with a given function, $y = f(x)$ and a set of N generated sample points $(x_i, f(x_i))$. A feed forward neural network is trained on this set of N points, using a back propagation algorithm. Some test points are chosen to compute the error between the value of the function and the value generated by the network.

8 Testing

We have tested cybercourse on four kinds of operating systems, with different kinds of browsers (see table 1).

The problems we encountered with browsers were:

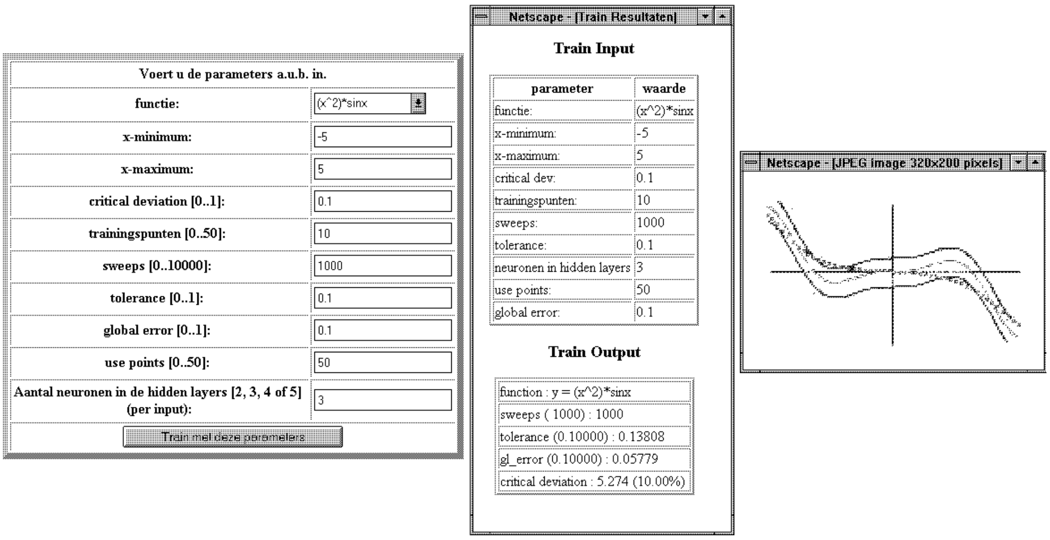

Figure 5: Sample output of the neural network course

- Microsoft Internet Explorer v 3.1 beta, in the Windows 3.11 environment did not work correctly all the time. For example showing Java applets did not work at all.
- The Netscape 2.0.1 in the MacOS environment crashed quite often and used a high amount of computermemory.
- Netscape 3.0 and Netscape Gold 3.0 in the Windows 3.11 environment had a problem how to show the colors of a Java applet and pictures in the right way. However this problem could be solved by opening a second browser window.
- Netscape 3.0 and Netscape Gold 3.0 in the Windows NT environment had the same problem as described for the Windows 3.11 environment.

9 Evaluation

About 25 students were asked as alpha testers of the educational site. Bugs in the system were removed and some improvements were implemented. The course was beta tested on a group of 100 computer science undergraduate students.

The students who tested cybercourse were surveyed by a questionaire with 15 statements. The students could indicate their consent with these statements on a five point scale. Some open questions were added. In table 2 we give the mean score and standard deviation of the scores on some important items.

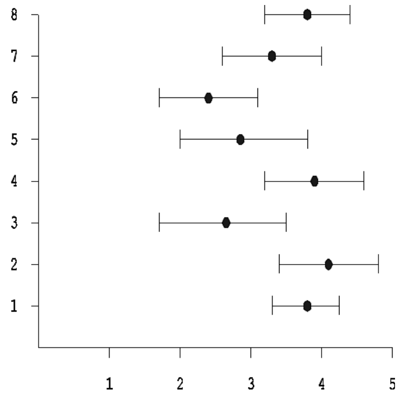

Figure 6: Evaluation Graph: 1=disagreement ... 3=neutral ... 5=agreement

And in figure 6 we can see a graph of these results.

From table 2 and the answers from the open questions we concluded:

- Most of the students were enthousiastic about remote learning. Students generally agreed that the web course was not suitable as a replacement of traditional teaching methods. However it was a valuable addition, which facilitated learning considerably. They intended to use the educational site for group works and practical assignments.
- A cybercourse has to be used in combination with lecture notes. A number of students printed

Operating System	*WWW browser*	*Version*
Apple Macintosh MacOS	Netscape	2.0.1
Microsoft Windows 3.11	Netscape	2.0.1
	Netscape	3.0
	Microsoft Internet Explorer	3.1 beta
Microsoft Windows NT 3.5	Netscape	2.0.1
	Netscape	3.0
	Netscape Gold	3.0
Sun Solaris	Netscape	2.2
	Netscape	3.0

Table 1: Operating System and tested browsers

Question	*Mean*	*Standard Deviation*
Because of the, cybercourse I spent more time to my study	3.8	.5
I was very satisfied with the cybercourse	4.1	.7
I was very satisfied with the integration of lectures, lecture notes and cybercourse	2.6	.9
I appreciated the facility to communicate with other students very much	3.9	.7
I like to work in groups	2.9	.9
I lost a lot of time using the cybercourse	2.4	.7
I have a better understanding of the coursematerial	3.3	.7
I appreciated the facility of distant examination	3.8	.6

Table 2: Evaluation results

out course texts, because they do not like reading text on a computer screen. They like to have printed text to read some pages simultaneously or to mark important parts and make notes.

- Most students were enthousiastic about doing exams on prefered time and place. Some students involved in project work at foreign universities did their exam from that university.
- The most positive point was the fact that the Internet can be used as a medium for distributing course material like manuals and software. It was easy to contact other students and teachers. About half of the students reported that they took part in communication by messages from group members or teachers by using the bulletin board or email.

10 Conclusions

A prototype for an educational web site has been developed and tested using a neural network module. The results of this test were promising. But certain problems remain to be solved.

Most teachers lack sufficient knowledge of Perl and HTML to develop their own pages and courses. An extended author tool is under development to overcome this problem.

The development of an educational site on the Internet is not possible without an experienced system manager or webmaster. This system manager has to have expert knowledge of Unix, HTML, Perl and the Internet.

Development of an educational site and additional courses take a lot of time and effort. The developers contributed colleagues from other technical universities in the Artificial Intelligence domain. To optimise investments and facilitate integration of web sites in education, they agreed to combine their efforts. All members create their own courses and make them available to other universities. The prototype described in this paper is the standard for all educational sites to be developed.

References

[1] S.A Cerri, J. Whiting, *Learning Technology in the European Communities*, Kluwer Academic Publishers, 1992.

[2] A.Mc. Dougall, C. Dowling (Eds.), *Computers in Education*, Elsevier Science Publishers B.V., 1990.

[3] D. Flanagan, *Java in a Nutshell*, O'Reilly and Associates, 1996.

[4] I.S. Graham, *HTML Sourcebook*, J. Wiley and Sons Inc., 1995.

[5] M. Helander (Ed.), *Handbook of Human–Computer Interaction*, Elsevier Science Publishers B.V., 1988.

[6] L. Wall, R.L. Schwartz, *Programming Perl* O'Reilly and Associates, 1991.