Murray W. Goldberg, "CALOS: First Results From an Experiment in Computer-Aided Learning". Accepted for publication in the Proceedings of the ACM's 28th SIGCSE Technical Symposium on Computer Science Education (1997).

CALOS: First Results From an Experiment in Computer-Aided Learning

Murray W. Goldberg
Instructor, Senior Faculty Advisor
Department of Computer Science, University of British Columbia,
goldberg@cs.ubc.ca

Abstract

This paper describes the outcome of an experiment in Computer-Aided Learning (CAL) using the World-Wide Web (WWW). The purpose of the experiment is to evaluate WWW-based course delivery both in terms of academic performance and in terms of student acceptance. A complete WWW-based offering of an existing third-year university course in Computer Science was produced. Students enrolled in the course were divided into three groups:

1. those taking the course using only the WWW-based resource (no lectures),
2. those taking the course by attending lectures (no access to the WWW-based resource), and
3. those with access to both lectures and the WWW-based resource.

Academic results were evaluated and qualitative student impressions were solicited. In general the WWW-only method of course delivery was an equal success compared to the lecture-only delivery, both in terms of academic results and in terms of student acceptance. On average, students in the combined lecture and WWW-based delivery performed better academically than students in the other groups.

Keywords

WWW Education Computer-Aided Learning

1. INTRODUCTION

This paper describes the outcome of an experiment in Computer-Aided Learning (CAL) using the World-Wide Web (WWW). This paper will be of interest to educators for its evaluation of the educational effectiveness and student acceptance of the WWW as a teaching resource. More generally it will be of interest to anyone designing WWW-based courses or training material.

The WWW provides educators with an outstanding opportunity to deliver courses, or to supplement existing courses, using a networked computer. There are clear advantages to the use of the WWW for CAL:

- courses can be developed once, by experts, and delivered at any scale. This creates the opportunity for improved course quality. It is also possible that a greater number of students can be served at a reduced cost. Finally, this ensures uniform delivery of courses; the quality will not vary with that of the instructor delivering the course.
- the pace, path and depth of the learning can be adapted by the learner to suit his or her background, level of aptitude and schedule.
- interactive Web-based exercises offer more opportunity for repeated experimentation and practice than is available with a chalk-board or pen and paper.
- the networked nature of the Web allows traditionally isolated distance-education students to be brought into a community of communicating peers through the use of bulletin-boards, chat areas and shared workspaces.
- courses can be delivered at any location. All that is required is access to a computer with a network connection or modem. This is an advantage for all students but is a special advantage for physically challenged students or those students for whom travel to a common campus is otherwise difficult.
- instant student self-evaluation is available through the use of multiple-choice questions or intelligent exercise evaluators.
- the Web is a familiar resource and Web browsers are quickly mastered even by students new to computers.

These opportunities for educators have lead to the development of a large number of WWW-based courses and other training resources [1], [2], [3], [4], [5], [6], [7]. In the department of Computer Science at the University of British Columbia we, too, have taken advantage of the educational opportunities presented by the WWW by creating a WWW-based version of a third-year course dealing with the fundamentals of Operating System design (CALOS: Computer-Aided Learning for Operating Systems) [8]. This WWW-based course is a rich resource including: interactive exercises, interactive simulations and demonstrations, on-line notes, student - instructor communication mechanisms, student - student communication mechanisms, progress tracking, navigational aids, student evaluations, student annotations, a glossary and a bibliography. We make use of text, images, animations and voice. Implementation makes heavy use of CGI programs and dynamic GIF and HTML creation. Click here for an image of the startup page for the course.

It was our intention from the beginning not to simply create the course and use it, but also to measure the effectiveness of the WWW and its level of student acceptance as a teaching resource. It was also our intention to experiment with the use of the WWW resource in varying contexts both in combination with lectures and as a stand-alone course. We have done our first round of experimentation in this respect and this paper presents our first results.

The remainder of this paper is organized as follows: section 2 presents an overview of the composition and implementation of our course. Section 3 describes the structure of the experiment in course delivery. Section 4 presents both the quantitative and qualitative results of our experiment. Finally, sections 5 and 6 present conclusions and future directions.

2. COURSE OVERVIEW

This section gives an overview of the WWW-based course implementation and contents. A more detailed description of the course can be found in [8].

2.1 Implementation Summary

The WWW version of CPSC 315 is highly interactive. In order to achieve its interactive nature, HTML and images for the pages are generated dynamically. The pages are composed in response to the course notes, student information, system navigation information and student input to exercises. The course contents are written in an augmented version of HTML which is passed through PERL scripts. These scripts generate HTML and GIF images for display. Operating system simulations and interactive exercises are written in C and PERL. These generate HTML and GIF images dynamically.

This mechanism allows for significant flexibility and interactivity, yet requires no special software or tools on the client (browser) side of the connection. The advantage is that our tools have to run only on our server; no complex porting or installation of software to client machines is necessary.

2.2 Course Contents

2.2.1 Page Layout

Every course page has a button-bar at the top and bottom allowing students access to course features (navigation, glossary, exercises, bibliography, chat, bulletin-board, etc.). These button bars are generated dynamically according to

1. the information in the external navigation file (discussed below),
2. the student's log and preferences files, and
3. the HTML and supplementary (non-HTML) information contained in the source pages.

For example, if the student's preferences file indicates the use of a black-and-white display, then the button bars are generated from black-and-white buttons (color buttons do not show up well on black and white displays). Likewise, if there is a set of multiple-choice questions embedded in the source file, these questions are stripped before being sent to the browser, and a question icon is generated for the button bar on that page. Clicking the question icon produces a form displaying the multiple-choice questions and prompting for their answers. Click here for a sample page of notes.

2.2.2 Navigation

Our navigation goals are four-fold:

- make the usual path through the course material obvious and direct
- facilitate unusual paths through the course material to accommodate students already familiar with certain aspects of the course material
- encourage excursions off the usual course path by allowing the student to instantly return to the point at which the path was left. A student should be able to do this using a single mouse click no matter how far off the main path they veer.
- when starting a new session a student should be placed in the same navigation context that they were in when they terminated their previous visit to the course.

In order to accomplish the first two goals, we have provided both a hierarchical view of the course, and a linear view. Using the hierarchical view a student can view the course at any level of granularity and skip directly to sections of interest. Using the linear view a student can proceed down the usual path of

course material, from one page to the next. No navigation information is recorded in the course content pages. Instead, there is a single file containing the sequence of pages on the usual path. This file is referenced by the PERL scripts in order to generate forward and backward arrows on the course pages. This makes reordering and insertion or deletion of pages trivial.

In order to accomplish the latter two goals, we need to know the identity of the student making each request. Access controls, which are used to restrict course access, allow us to know the student's identity. Once the identity of the student is known, it is a simple matter to log that student's accesses to the course pages. Having a recorded history of each student's references allows us to place that student either back on the usual path (after an excursion off that path) or back at the same location from which they ended their previous session. Such a log also allows us to make inferences about the student and various course sections based on the number of accesses.

2.2.3 Communication

Communication among students, and among students and the instructor is accomplished using a chat service and bulletin-board.

The chat service allows real-time communication among course participants. It presents a virtual set of rooms that can be entered. The names of the occupants of each room are displayed for all to see. The door to each room can be locked to facilitate private conversations. Each room can be labeled by its occupants in order to advertise the nature of the conversation taking place.

The bulletin board allows the posting of messages in a public forum. It can be accessed in either of two ways: either from the central bulletin-board page, or from any of the course-notes pages. When accessed from the central bulletin-board page, messages are displayed for the student to peruse. Knowing the identity of the reader allows us to refrain from presenting messages that have already been read by that student (unless he or she so desires). The usual range of bulletin-board features exist including threading and responding to messages with an embedded quote of a previous article.

When the bulletin-board is accessed from a page of course-notes, only messages pertaining to that page are presented. A message pertains to a page if it was posted from that page, or if it is a response to a message originally posted from that page. When viewed from a course-notes page, all messages, including ones previously read by the student, are presented. This essentially creates a global annotation mechanism where discussions about that page are recorded for others to see. The expectation is that questions, answers and other discussions about each page become part of the permanent record for that page. They supplement the learning experience, actually forming part of the content of the course. Shortcomings in the course-pages will usually be addressed in the related bulletin-board articles. This adds a dynamic, self-correcting nature to the course notes.

2.2.4 Interactive Exercises and Simulations

Interactive, graphical computer simulations allow students to explore operating system concepts. They are both student-driven and computer driven. Student-driven interactive examples simulate operating system responses to student-generated input conditions. Computer-driven simulations exercise the student's knowledge by presenting a set of conditions and asking the student to indicate outcomes. The student's response is checked for correctness by the simulation. The computer can generate an almost infinite set of starting conditions; no two exercises are the same.

The simulations include six disk scheduling algorithms, three memory allocation algorithms, five page replacement algorithms, four process scheduling algorithms (including exponential averaging), process creation constructs, and others. Click here for an image of a disk-scheduling simulation.

These interactive exercises and simulations form an integral part of the course, adding an experience that cannot be matched in the classroom. In a typical class only one or two such examples can be presented due to the time required to set up and evaluate the problem. The WWW-based interactive examples relieve the student of having to set up the problem. As a result it is often the case that five or ten such exercises can be performed in the time required to do one corresponding pen-and-paper exercise. Click here for an image of an interactive exercise.

2.2.5 Student Evaluation

Multiple-choice questions are provided that enable the student to perform self-evaluation at any time. Questions related to the current pages are available by clicking an icon on the button bar. For each question answered, a result is displayed (correct or incorrect) and a textual or graphic explanation of why the answer was correct or incorrect is given. Click here for an image of a multiple-choice question page.

There is also a mechanism through which timed quizzes can be administered. A form with questions and text-entry boxes for the student's responses are presented. A timer is used to record the interval between when the student is presented with the quiz and when the response is given.

2.2.6 Other Course Features

Other features include the following:

- a complete, search-able course glossary containing text, images, and references to course notes
- each notes-page has an icon which, when clicked, presents the list of page and chapter references to the course texts for that page of notes
- an annotation mechanism that allows students to make private annotations of individual pages
- each page of notes includes an icon which, when clicked, presents a page containing learning goals for that page. This provides the students with topic emphasis that would otherwise be provided in lectures.

3. THE EXPERIMENT

This section presents an overview of the structure of the experiment that evaluates the use of the WWW-based course material both alone and in combination with lectures.

3.1 Course Background

CPSC 315 is a popular third-year Computer Science course dealing with the fundamentals of Operating System design. Traditionally it has been offered in a regular lecture format (three hours per week) two to three times per year to roughly 80 students per offering. Student evaluation consists of four programming assignments, one midterm exam, and one final exam. The author has been teaching this course for the last five years, and has consistently enjoyed top or near-top student ratings. The popularity of the course makes it particularly suitable for a comparison with a WWW-based offering. Here we can

be sure that it is not the case that students over-rate the WWW-based offering because they consider anything better than the current, lecture-based, offering.

3.2 Experiment Setup

In the fall of 1995 students of CPSC 315 were asked to volunteer to be part of an experiment comparing various offerings of the course. The understanding was that a volunteer could opt-out of the experiment at any time and return to the traditional lecture-only course format. Sixty-four of the sixty-eight students agreed to volunteer.

Those that volunteered were split into three groups. The placement was random except that an attempt was made to balance the groups so that all levels of aptitude were represented in each group. Aptitude level was measured from the student's performance in the main prerequisite course: CPSC 216; Data Structures.

The three groups were defined as follows:

- Group 1 was the WWW-only group. This group of students was not allowed to attend lectures. They took the course entirely on-line. These students were expected to keep pace with the regular offering, though initially no attempt was made to be sure they did so. This group met with the instructor once per week. This meeting was not a lecture, but was used as a time to discuss the student's perceptions and problems with the course.
- Group 2 attended lectures and had full access to the WWW-based material.
- Group 3 did not have access to the WWW-based material but received the course in the traditional format: lectures only. This group was the control group.

All three groups of students wrote the same final exam and did the same assignments. Scheduling problems prevented the writing of the same midterm; the WWW-only group wrote a different midterm than the remainder of the students. The two midterms, however, tested exactly the same material, and an attempt was made to try and present questions of equal difficulty on both exams.

Two forms of results were expected. First, quantitative results could be measured by comparing the academic performance of each group. Second, qualitative results were sought through the administration of a questionnaire. The questionnaire was created by the UBC Center for Applied Studies in Evaluation. To ensure honest answers, students were assured that the questionnaire would be treated as anonymous, and that it would not be viewed until after course grades were submitted. The questionnaire was given out during the final exam and returned over the following week. Both sets of results are presented in the section that follows.

4. EXPERIMENT RESULTS

This section deals with the results and experience gained from conducting the experiment as outlined in the previous section. The reader should be cautioned that given the small class size (under seventy students) no statistically significant outcome can be reported here. Regardless, we feel that the anecdotal results are still of interest.

4.1 Experience Gained During the Offering of the Course

The WWW-based version of CPSC 315 had never before been used to the extent it was during the experiment. As such, the students provided a great deal of valuable insight in terms of changes that would improve the WWW-based course. For example, the structure of the bulletin-board was redefined as a result of student input. Also, a *goals* section was added to each page of notes in order to indicate the relative importance of the course topics. This was something that the WWW-only students felt would be present in a lecture-based course, but was not initially present in the WWW-based course. As a third example, many students requested that we make printed copies of the WWW-based notes available for photocopying (which is easier than each student having to print each page). These students wanted to be able to read the notes at their leisure (for example, in bed).

Another interesting result is that the students in the WWW-only group were divided in terms of their degree of self-motivation. One subset of WWW-only students greatly appreciated the ability to self-pace this course in order to accommodate the changing demands of their other courses. The remainder felt that without external incentive, they were unable to maintain the motivation required to keep pace with the lecture-based offering. The problem did not seem to be one of boredom with the material or the delivery method (most students seemed to enjoy the novel experience and the ability to work on the course anywhere, and at anytime). Instead, the problem was that each of these students was faced with at least four other courses being taken concurrently. Each of these other courses was lecture-based and was therefore not self-paced. As a result, when the overall load on the student was high (as it is much of the term) it was the self-paced course (CPSC 315) that tended to suffer.

Competition with other lecture-based courses became the most serious obstacle to the WWW-only student's acceptance of the course delivery method. Of the four students that opted out of the WWW-only group (and thus out of the experiment), each cited this competition with lecture-based courses as the reason. During our weekly meetings, the author and the students decided that the way to combat this problem was to institute bi-weekly quizzes. This would require that the students keep pace in order to be prepared for the quizzes. Those students that were highly self-motivated declared themselves and were exempt from the quizzes. Once the quizzes were instituted, the problem greatly diminished and no further students opted out of the experiment. We found it interesting that many students lacked the motivation to keep up on their own and were happier when the course pace was forced upon them. This may not be surprising, though, given that the bulk of most students' academic experience is not self-paced. One might speculate that if the competition for student's time was from other WWW-based courses (and not from lecture-based courses) that this would not have been a problem.

4.2 Academic Results

It is not useful to simply compare the average academic performance of the three groups of students. While an attempt was made to have all levels of academic aptitude represented in each group, the groups were small enough that individual differences in background and ability could significantly skew the results. Also, simple averages do not differentiate the case where, for example, each member of the group scores a little better than expected, from the case where a couple members of the group score much better and the remainder of the group is unchanged.

Instead we adopted the following approach. We decided to count the number of students in each group that performed:

- better than expected

- as expected
- worse than expected

In order to determine expected performance, the students' grades in the main prerequisite course, CPSC 216, were used. This course is usually a good determiner of the grade in CPSC 315.

In order to determine whether a student performed as expected, we declared a value (call it MARGIN) to be used as follows. If the grade in CPSC 315 was within MARGIN (either way) of the grade in CPSC 216, the student was said to have performed as expected. If the grade was less than the 216 grade - MARGIN, then the student was said to have performed worse than expected. Finally, if the grade in 315 was higher than the 216 grade + MARGIN, the student was said to have performed better than expected.

Determining the value of MARGIN was somewhat arbitrary. Even so the choice did have an effect. If we chose the value too small, a small difference in performance could move the student from *better than expected* to *worse than expected* (or the other way). If we chose the value too large, then most of the class would appear to have performed as expected. We experimented with many values for MARGIN. As expected, changing the value of MARGIN moved students either into or out of the *performed as expected* group, but did not move them between the *better than expected* and *worse than expected* groups. Thus, all values for margin gave the same qualitative outcome. For the purposes of the following results, a MARGIN value of 5% was chosen. The results are as follows (each number represents the percentage of the students in that group that performed better than, worse than, or as expected):