

1997-12-01

Creating and Implementing an Online Writing Course

Martin Holmes

Follow this and additional works at: <https://digital.car.chula.ac.th/pasaa>



Part of the [Reading and Language Commons](#)

Recommended Citation

Holmes, Martin (1997) "Creating and Implementing an Online Writing Course," *PASAA*: Vol. 27, Article 10.

DOI: 10.58837/CHULA.PASAA.27.1.10

Available at: <https://digital.car.chula.ac.th/pasaa/vol27/iss1/10>

This Original Article is brought to you for free and open access by the Chulalongkorn Journal Online (CUJO) at Chula Digital Collections. It has been accepted for inclusion in PASAA by an authorized editor of Chula Digital Collections. For more information, please contact ChulaDC@car.chula.ac.th.

Creating and Implementing an Online Writing Course

Martin Holmes

English Language Centre
University of Victoria, BC, Canada

Abstract

This discussion paper describes our experiences during the creation and deployment of an entirely Web-based intermediate English writing course at the University of Victoria. It is hoped that other institutions, materials-writers and instructors will find it useful when contemplating, planning or implementing similar initiatives.

Introduction

The number of educational courses delivered wholly or partially through Internet technology has increased dramatically in the last year. Among the reasons for this are cost-cutting (online materials are cheaper than books; providing computer-based materials sometimes appears to be cheaper than hiring a teacher), the increase in distributed education and distance learning, and the simple desire to be involved with a "happening" technology. Online materials have developed rapidly from the early days of static reading texts and course notes posted on Web sites; scripting and programming languages such as JavaScript, Java and VBScript allow complex interactive exercises to

be created, while other advances enable us to deliver real-time audio and video, or build 3D virtual worlds in which our students can interact with each other.

While many educators and educationalists are doing exciting things, however, much of the material being produced is of an experimental nature. We see many Web sites showing examples of what can or could be done, or how a specific technology might be used. Creating interactive course material takes a long time, and it is only now that the first fully-fledged, wholly interactive courses are beginning to appear. Over the last year, I have developed an

Online Writing Course for the English Language Centre at the University of Victoria. This paper is a description of the process by which the course was created and implemented, and an attempt to pass on some of the many things I learned, both technical and non-technical, during the project.

The English Language Centre decided to go online, and to make our first venture a writing course, for several reasons. A team had already been creating multimedia CD-ROM-based material for several months, and we wished to continue to build on our commitment to new technologies. The University is blessed with a wealth of computer resources, including reliable Web-servers and a large number of student-access workstations, as well as many staff committed to the development of computer-based teaching and training programs. Thirdly, our ESL student population, drawn from many countries from South America to East Asia, are generally with us for two or three semesters only, and during that time they are anxious to capitalize on the Canadian environment by focusing on their speaking and listening skills; we felt that it would be useful if they could work on writing skills from their home countries both before and after their stay with us. Finally, we felt that the state of Web technology would allow us to deliver a course focused on writing even over low-bandwidth connections and to countries where the Internet backbone remains in its infancy; this is not yet possible, or at least reliable, for audio and multimedia materials.

Equipment, Tools and Personnel

One member of staff (myself) was assigned to the task, and the Language Centre provided a computer to work on, and access to other PCs and Macs on the network for testing. I arranged a Web site account on the University server and assembled my tools:

1. An HTML editor (*HotDog Pro* version 3)
2. A graphics package (*CorelDraw* version 6)
3. FTP and telnet applications
4. A collection of browsers in different versions: *Netscape Navigator* 2.x and 3.x for Mac and PC, and *MS Internet Explorer* 3.x for Mac and PC.

Like many Web developers, I tend to write HTML code in a text editor and test it directly in a browser, so it would have been perfectly feasible to proceed without a specialized HTML editor. However, HotDog provided a number of extremely useful and time-saving tools, including:

the ability to insert common HTML tags with the click of a button

the ability to define custom tags for one-click insertion (to insert, for example, a tag to change text to my own favoured shade of red)

the ability to do multiple-file find-and-replace, changing the same word or phrase

in 50 documents instantly (vital for managing a large and developing Web site).

The graphics package was essential, and was most commonly used for these tasks:

creating new graphics for logos, headings and icons

modifying public-domain graphics by cropping, resizing, sharpening or changing colour depth

converting graphics from non-Web file types to *.gif and *.jpg formats for the Web

Combining or altering clipart from the package to create new graphics (for instance, combining a hockey player, a baseball player and a runner to create a generic "sports" logo).

The FTP and telnet applications were needed for uploading and managing files on the server. The collection of different browser programs (all free, incidentally, for educational use) enabled me to check the appearance and functionality of pages to ensure that students would see more or less the same thing, whatever platforms or applications they were using.

The final, vital ingredients were a good working location and a useful set of colleagues. I was able to do my work in the Research and Development laboratory of the CALL facility,

where a number of other staff were working on other projects involving Web design, graphics and programming. The atmosphere was supportive and the enthusiasm for the technology palpable. A team spirit quickly developed, and I often found myself spending several hours working with someone else to help solve a problem on their project, just as others often devoted hours to helping me solve problems with mine.

Early Decisions

An important early decision concerned the level of technology we would assume our students had access to. A survey of computers available to students in our own facilities revealed that many would not readily and stably run *Netscape Navigator 3* or handle Java applets. The computers available to students in their own homes tended to be even older and slower. Considering this, and the fact that we hoped to deliver the course to students all over the world, many of them logging in by modem, we resolved to use a low level of technology, and keep bandwidth requirements to a minimum, avoiding Java, JavaScript, ActiveX, large graphics and animations. We elected, instead, to stretch HTML to its limits to create interactive materials, while keeping pages to 40K or less to reduce download times.

It was also decided to ensure that a large number of the exercises would be e-mailed in to the instructor (directly from a form on the Web page), marked, and returned to the student, since this would be the main form of interaction

between student and instructor (see 2, 3 and 4 below).

I designed five main types of content page:

1. Interactive multiple-choice exercises for presenting new ideas, skills and language items. These are interactive in the sense that when the student chooses an answer, right or wrong, specific feedback is given on that particular answer. In the case of a wrong answer, this might show why it is wrong; in the case of a right answer, the response may reinforce the point or supply extra useful information.
2. Mail-in practice exercises using option buttons and check boxes for multiple-choice questions.
3. Mail-in practice exercises based on sentences and short paragraphs.
4. Longer mail-in assignments.
5. Non-interactive "handout-style" pages explaining key points of grammar or writing skills.

The course was planned in ten units, beginning with a review of paragraph writing and working towards a full research essay.

Finally, some design decisions were made at the beginning. I settled on a limited palette of

colours for text — a dark blue, a dark red, a purple and a dark green — and decided that black text on a white background would be predominant, with red and blue used for highlighting and contrast. Having no graphics or design experience, I made these decisions on the basis of intuition, but I think the choice of a simple, restrained and paper-like appearance for the pages was fortuitous. I also designed a set of matching "header" graphics for the different exercise types, and picked out some simple icons for use as navigation buttons.

Some Statistics

These are some statistics on the completed course:

Total Web pages:	186
Total graphics:	194
Interactive exercises:	89
Non-interactive exercises:	19
Mail-in assignments:	44

Timetable for the Project

Half of my working day was allocated to this project over a period of six months. This was a tight schedule, and I found myself working many more hours in the evenings and at weekends in order to keep up. Writing the complex but repetitive HTML code for the interactive pages proved tedious and time-consuming, so I wrote a small Visual Basic program enabling me to enter questions, answers and responses, and then generate the

code automatically. (This was so useful that I have since written two more Windows 95 applications that generate HTML/JavaScript exercises in the same way.)

Despite the stress, having a clear deadline was essential. It is possible to spend infinite amounts of time finding or perfecting a single graphic, or minutely adjusting the layout of a page; however, it was soon apparent that the only way to complete a project like this is to accept a certain level of imperfection and keep pushing on with the endeavour. There is little point in spending two days prettifying a graphic that the student will see for less than five minutes; it is far more important to ensure that the content is effective and varied, and that there is enough of it to keep the student engaged and active.

In the midst of the pressure to keep moving, I tried to arrange my schedule in order to leave one afternoon a week — usually Friday — for playing, learning and thinking. I would spend this time on anything that engaged my enthusiasm — learning new code tricks, surfing the Web for ideas, collecting materials, finding sources of information, graphics and help, and even indulging the temptation to toy with a graphic for two or three hours. I think this helped to keep me engaged with the medium and enthusiastic about the project even several months into the work.

Putting the Course to Work

Our initial implementation of the course was in the form of an elective offered to our ESL students on campus. This allowed us to test the material, get student reactions, and discover how much work was involved on the teacher's part, without taking fees from students or having to do long-distance technical support. We learned a number of important lessons:

1. Learner training took two to three weeks, and even after that time, some students were still frustrated and struggling with the technology.
2. The marking load for the instructor was quite onerous. We calculated that, on average, the instructor would devote 16.5 hours to each student in the course. This means that a group of 30 students would constitute the same workload as a full-time teaching job.
3. The drop-out rate among students was a little higher than for normal elective classes. Two factors may be at work here. Some students were clearly uncomfortable with the technology and found this a stressful way to learn. Also, distance education typically has higher drop-out rates than ordinary classroom study, and it is possible that online learning is similar to distance education even when it takes place on the same campus.

Further Lessons: Password Paranoia

Following our deployment of the course and its initial trial, a new and unanticipated dilemma arose. The course was inevitably discovered by the "robots" used by Web search tools, and Web surfers began to find their way to it. Potential students began to do the exercises and submit their work, and others began to ask for permission to register. At the same time, those responsible for funding the project began to feel the need to protect their investment, and it was proposed that the course material be password-protected in some way, so that it could not be accessed by people not registered as students.

As a developer, I was a little uneasy at this. I had two objections. First, the course would basically be useless to those who were not registered as students, because no marked work would be returned to them; therefore, I felt, we had nothing to fear from casual surfers except that they might eventually register and pay. Secondly, I was proud of my work, and believed that the quality of the material was such that if potential students *were* able to surf through the course, they would appreciate the quality of the work and would be likely to register (if they were looking for an online writing course). Management, however, looked on the course as a product, which had cost my half-time salary over two semesters to produce, and were concerned to prevent anyone from enjoying this product without paying their money up-front. Eventually, a compromise was reached. I took advantage of the availability of a piece of course-hosting server software called *WebCT*

(from the University of British Columbia), which was being beta-tested, to put the course behind a firewall. However, I kept a "front-end" on the public area of the server, and made one unit of the course available for the curious to peruse.

However, potential students continued to contact us in ever-increasing numbers. Soon several students per day were asking to register. This was gratifying, of course, but it raised a new and fascinating problem: we had no systems in place for taking on external students in this way. Students were brandishing money at us, but we were unable to take it because no-one knew how to employ or pay teachers for this job; apparently, no precedent existed.

I was now teaching the course as an elective, and when management asked me to estimate the number of hours a teacher would devote to a student doing the course, I was able to confirm our previous prediction: around 16.5 hours. This seemed a shock to management; apparently, their conception of an online course did not involve a large teacher-student interaction component, while for me, the guiding principle when I wrote the course had been that in the absence of most other forms of human interaction, the flow of work between instructor and student should be the heart of the course experience. We were at an impasse. For months, no decision was made about how students could be taken on. Meanwhile, messages from would-be students continued to flow in, and I began to wish I had password-protected the whole site.

Conclusion

As I write, we are finally beginning to address the possibility of taking on external students. The University is reluctant to pay a full salary to teachers working in this way, uneasy about my workload predictions, and apt to think of the course material as "the course", and the teacher simply as an adjunct, an online "presence". My view of what constitutes an effective online course has, through my own experience, diverged so far from this that I would be reluctant to take on the job of teaching the course here; I view it as a highly skilled and specialized kind of teaching, and I would like to see it paid as well as (if not better than) classroom teaching.

If teachers are willing to teach it, though, there are some interesting new possibilities in

the way we might hire them, and enroll students. With an online course, there is no longer any need for students to start and finish the course at fixed times, or in groups. A student could begin the course at any time, be assigned to a teacher, and work at his or her own pace (within reason, of course). Teachers could be paid per student (as they often are in distance education); some might elect to take only one or two, while others might accept larger numbers and become "full time". Any number of students and teachers could be using the course at any time, working at different speeds. To institutions short of classroom space, and instructors needing to supplement meagre salaries, this has clear benefits. However, it will probably take time for institutions to adapt to this new paradigm and develop the administrative systems necessary to support it.

References

- Holmes, Martin. *University of Victoria Online English Writing Course*.
<http://web.uvic.ca/hrd/OLCourse> (3 October 1997).