Teaching through electronic mail

Stuart Watt

Knowledge Media Institute and Department of Psychology

KMI-TR-39

September 1995

An unpublished report on the DMZX863 experimental Internet course — Common Lisp for Artificial Intelligence

Teaching through electronic mail

Stuart Watt
Knowledge Media Institute and Department of Psychology
Open University
Walton Hall
Milton Keynes. MK7 6AA.

Email: S.N.K.Watt@open.ac.uk

Abstract

In November 1994 we (members of the former Human Cognition Research Laboratory) ran an experimental version of the Masters level course DM863, 'Common Lisp for Artificial Intelligence,' taught as far as possible entirely through the Internet. This report describes this course and outlines some of the experiences and ideas that evolved during this course.

Introduction

The intention behind this course was to explore the extent to which it is possible to teach a course entirely through the Internet. The work was derived from two sources, first Tony Hasemer's work on transferring the basic DM863 'Common Lisp for Artificial Intelligence' course onto the FirstClass conferencing system, and second, the virtual summer school work exploring teaching through the Internet. The main difference in this course was that we tried a 'lowest common denominator' approach—we wanted to see how effectively we could teach using communications media so widespread that we could have global reach. In a sense, its aim was the exact opposite of the virtual summer school—instead of exploring how rich a teaching experience could be created using new technologies, we wanted to see how much teaching could be achieved using the most basic electronic communications technologies.

In this case, by 'Internet' we assumed an 'IP' (Internet protocol) of some kind between computers. The minimum specification that we assumed was an electronic mail connection of some kind between students and tutors. In practice, this mostly manifested itself as an electronic mail system using the standard Internet mail protocol, 'SMTP.'[1]

[1] In fact, the reach of the course was wider than this implies. There are several other computer networks in the world, and almost every one of them has good connections to and from the Internet for the purposes of electronic mail. Other kinds of communication between the networks are less well supported. One student was not on the Internet at all, but on a different network, Bitnet. Because we used electronic mail for communication, and this is a sound lowest common denominator, this was transparent to the other participants on the course.

Students were recruited by a message posted on the Internet's global conferencing system, the Usenet. Normally, the ethics of the Internet prevent its use for advertising, but the experimental nature of this course, and the academic nature of its content, meant that we were able to recruit students this way. (But similar postings since, attempting to recruit paying students, have been remarkably unsuccessful.)

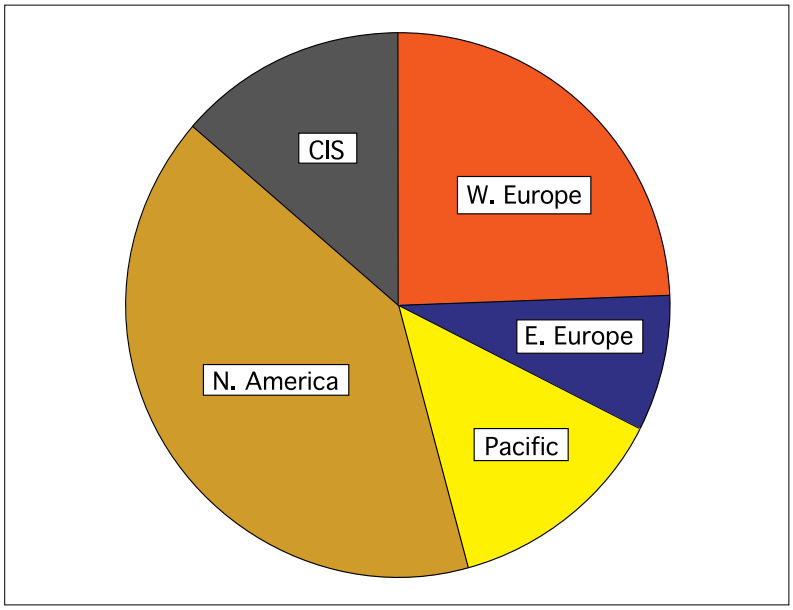

Figure 1. Student breakdown by country.

Normally, the course DM863 is taught by communicating electronically with the students, apart from the assignments book and the set book, which are printed materials sent normally by post. Tuition is carried out using the synchronous conferencing system FirstClass, which ensures that a student's or tutor's messages are immediately visible to all members of the course. The course uses a 'team-teaching' format, where all the students and all the tutors share a single conference. Assignments are posted electronically; using a PT3 form designed within FirstClass. These forms carry the submitted assignments between a student, their tutor, and the assignment handling and administration departments, all within the FirstClass conferencing system.

The goals of the Internet version of the course were several:

1. to explore the extent to which very basic (and widespread) media could be used for successful tutorial support,

2. to explore the extent to which the Internet could be used as a teaching medium for Open University courses,

3. to explore the effects of the media restrictions imposed on the teaching and learning processes by the use of the Internet, and

4. to explore the extent to which the University's administrative systems could in time encompass courses taught remotely through the Internet and other similar computer networks.

At the beginning of the course, all the students were sent (a) an electronic version of the study guide, (b) an electronic version of the assignments book, and (c) an electronic version of the first chapters of the set book, 'Common Lisp Programming for Artificial Intelligence,' by Hasemer and Domingue. All were converted from word processor files to plain text so that we could guarantee they would be readable by every student, even though this meant that some of the style and structure of these texts was lost. In parallel, PostScript versions of all these texts were made available by the File Transfer Protocol ('FTP'), on a computer which was accessible to Internet users. This meant that every student with access to a FTP client program could get hold of these texts, and if they had a PostScript printer, they could have a good paper copy of the texts, with all the style and structure of the original set book.

In practice, not every student had access to an FTP client program, notably our Russian students who were effectively limited to the use of electronic mail. There is a gateway ('FTPMAIL') which allowed people with only electronic mail access to get access to FTP archives, but the size of these archives often makes this impractical.

Students were initially recruited by a single message posted to the Internet's conferencing system, the Usenet. On this system there is a single special interest conference for Common Lisp; a single message posted in this conference raised about eighty messages expressing an interest, and slightly less than sixty students registered to take the course.

Originally the course was purely experimental, evaluating the use of the minimal communication media provided by the Internet. Although students were registered for the course, to begin with this registration was purely nominal and in no way involved any of the University's official administration departments.

As interest in the course grew, it was decided that much of the benefit in this experiment would be in evaluating the extent to which the University's administration systems could be extended to enclose an Internet based course. This meant that we asked our students to register retrospectively, completing an official registration form. This, too, was mailed to our students in both a plain text form and in PostScript which could be printed. The form could not be returned electronically because a real signature was required, so the students sent back the forms by surface or air mail directly. This was the first point in the course where it was found that it was not possible to mediate all interaction with our students electronically.

There was one other significant problem—and not one that we could control in any way. One of the set books for the course was the second edition of Guy Steele's "Common Lisp: The Language." This book turned out to be out of print in many countries, and there were only a few copies available even in the US and UK. Fortunately, the book was available in electronic form, and so could be obtained and printed by many of our students. Unfortunately, the book is over a thousand pages long and the printable file is quite a few megabytes. This meant that the cost was often prohibitive, and some students were without a copy of this essential book for some or all of the course duration.

As far as possible, the experiment retained the normal structure of the course. Assignments were written, then submitted to, and marked by a student's tutor in the normal way. The course involved the usual three assignments, the first of which was purely formative and did not contribute to the student's grade, the other two counted towards the final mark for the course.

Building a tutorial group

When we were looking to map the University's teaching systems into purely electronic media, we were faced with a number of options. There are a lot of different communication routes involved in traditional Open University teaching, and our goal was to preserve as much as possible of the experience of this style of teaching.

Our main focus was on interaction between students and tutors, both individually and as a group. Direct interaction between a student and a tutor was easy—this is simply provided by electronic mail. Keeping the feel of a tutorial group was far more difficult. I am not sure that this was successfully achieved, probably because the drop in student numbers quickly took the numbers participating down to a level where interaction was less frequent.

There is one important difference here, between 'synchronous' and 'asynchronous' interaction. In 'synchronous' conferences, all the participants are involved at the same time and responses are immediate. In asynchronous interaction, participants are not required to be connected at the same time; instead, people will connect and pick up messages that might have been left earlier by other participants.

The Internet provides both: synchronous interaction in the 'Internet Relay Chat' (or 'IRC') protocol, and asynchronous interaction in the Usenet. But for a variety of different reasons, varying from technical, psychological, and cultural, the Usenet is far more widespread as a means of communicating. Internet Relay Chat is still used occasionally, but mainly for special occasions such as live conferences.

This is probably an oversimplification. There are a lot of other newer systems emerging with quite startling rapidity. One that was pressed on us from the very early days of this course was 'MUD's or 'MOO's. These (the name derives from 'Multi User Dungeon') resemble 1970s style 'adventure' games; they usually use a teletype command language interface, where people connect up to the system and then act to command an agent which acts for them in a simple textual version of virtual reality. Some of these environments have been customised for teaching, and provide 'rooms' where people can meet, and where a whiteboard, desks, and all the usual school teaching paraphernalia are provided. We evaluated this form briefly, as a possible complement to the other teaching tools used on the course, but found that it requires a lot of computer awareness and tolerance to cope with the rather hostile interface and command language (although this may well improve with time and further development.) Perhaps more worrying was the large amount of connection time that might be involved—this is particularly of importance in the UK where even local calls to the University's network can be expensive. In the US, where these systems are more widespread, local calls are often free.

The other main alternative was the Usenet, but this proved to be so open that it would probably be hard to use. The Usenet works by registering one computer with others so that new messages are exchanged in both directions periodically. This meant that either we would have to ask students to use our machine directly as a server, or we would have to accept the loss of privacy associated with public distribution of our discussions. Neither of these were acceptable, so we ruled out the Usenet as a teaching medium.

In the end, we chose the safest alternative, which was to use an electronic mailing list system. We set up a number of mailing lists, where all mail sent to a number of special addresses

would be forwarded to all registered subscribers. We used a free mailing list administration system, which was functionally good enough for our purposes, but was hard to administer. This was probably because we didn't choose the best mailing list administration program available.

We used three mailing lists for the course. The first, `dm863-work@open.ac.uk`, had all students, all tutors, and members of the course team as subscribers. This list was not moderated; any message received from any student would automatically be sent to all subscribers registered to this list. This list was intended for work-related issues; that is, questions relating to the academic aspects of the course: questions, the set book, and the content of the course. The second list, `dm863-chat@open.ac.uk`, was set up with the same participants, but the content was very different: this list was mainly for people to interact socially. Cats were a major theme on this list. The third list, `dm863-materials@open.ac.uk`, was again set up with the same participants, but it was moderated. This list was used to post the course materials direct to all the students; the study guide, the assignments book, and chapters from the set book were mailed out on this list in plain text form.

In practice, the mailing list solution was not as primitive as it sounds. As tutors, we used a mail reader called 'Eudora,' which provides automatic filtering. That is: we could set up the mail system so that it would recognise which list had distributed a message, and it would gather all the messages from each list into a different folder automatically. This achieved much of the effect of a true conferencing system, such as FirstClass. It was my feeling, however, that even though the systems were virtually identical with regards to the interface, an unquantifiable feeling of presence that is part of synchronous conferencing systems was still absent. This phenomenon remains a topic for further research.

A typical message from the mailing list system is shown in figure 2. Note that some of the message's header lines are set to show the mailing list information, while others show who actually wrote the message.

```
Date: Thu, 23 Feb 95 10:24:58 GMT
Reply-To: dm863-work@open.ac.uk
Originator: dm863-work@open.ac.uk
Sender: dm863-work@open.ac.uk
Precedence: bulk
From: J.B.Domingue@open.ac.uk (John Domingue)
To: s.n.k.watt@open.ac.uk
Subject: Re: Accessing and hiding slots
X-Comment: DM863 Work

>I could use advice on a lot of stuff, but I'll start with information
>hiding. How is this done in CLOS?
>
>As I understand it, accessors and methods (and the generic functions
>they generate) are the primary means for a class to present a public
>interface. How then do you imbed code in a class that you want to be
>hidden and inaccessible from outside the object?

you can not hide things in CLOS. This seems to horrify OOP purists but i
think its ok. When you program things in clos if you want to hide a slot
you do not define an accessor for it (but presumably some other method will
use it using slot-value or with-slots). External objects use the methods
you've defined and are /encouraged/ to not use slot-value and with-slots.

The philosophy in clos is that it is 'good style' to access objects via the
defined (public) methods but you always have a back door if you really need
it

john
```

Figure 2. A typical message from the mailing list system.

Figure 3 below shows the appearance of a single folder from the Eudora mail system: this time showing some of the messages in the ‘DM863 Work’ mailing list.

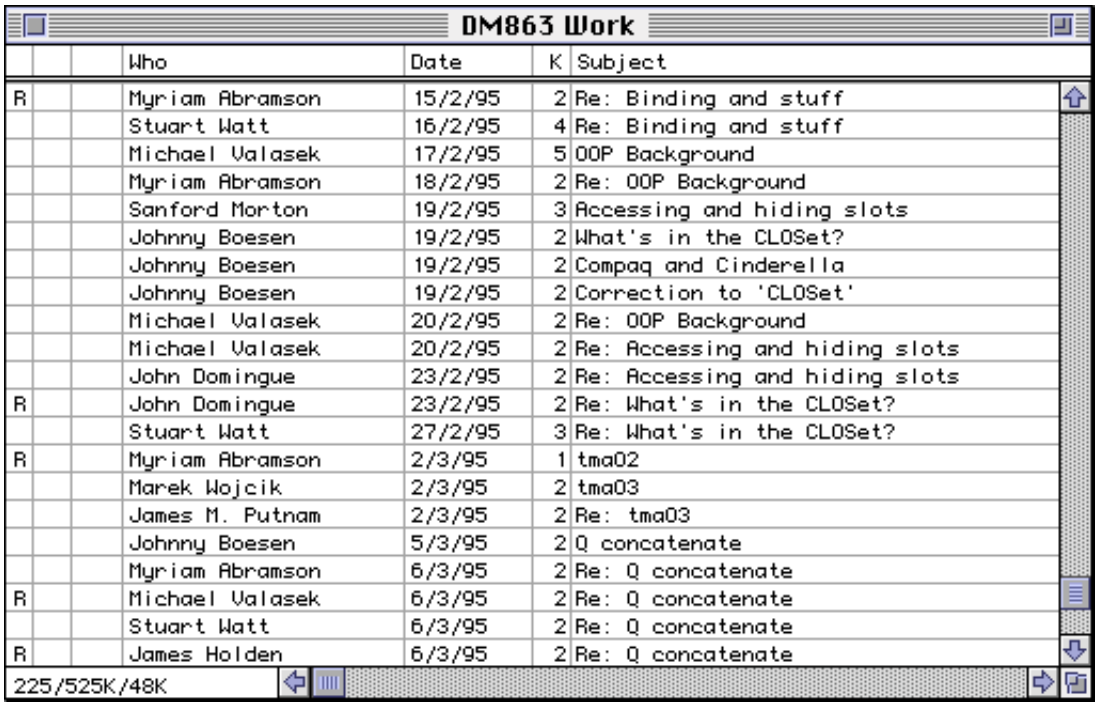

Figure 3. The main work mailing list, as presented by Eudora.

For comparison, figure 4 below shows the equivalent display from FirstClass. The displays may look very similar but, between them, there is the fundamental difference that the mail system in figure 3 is asynchronous, and in figure 4 the system is synchronous.

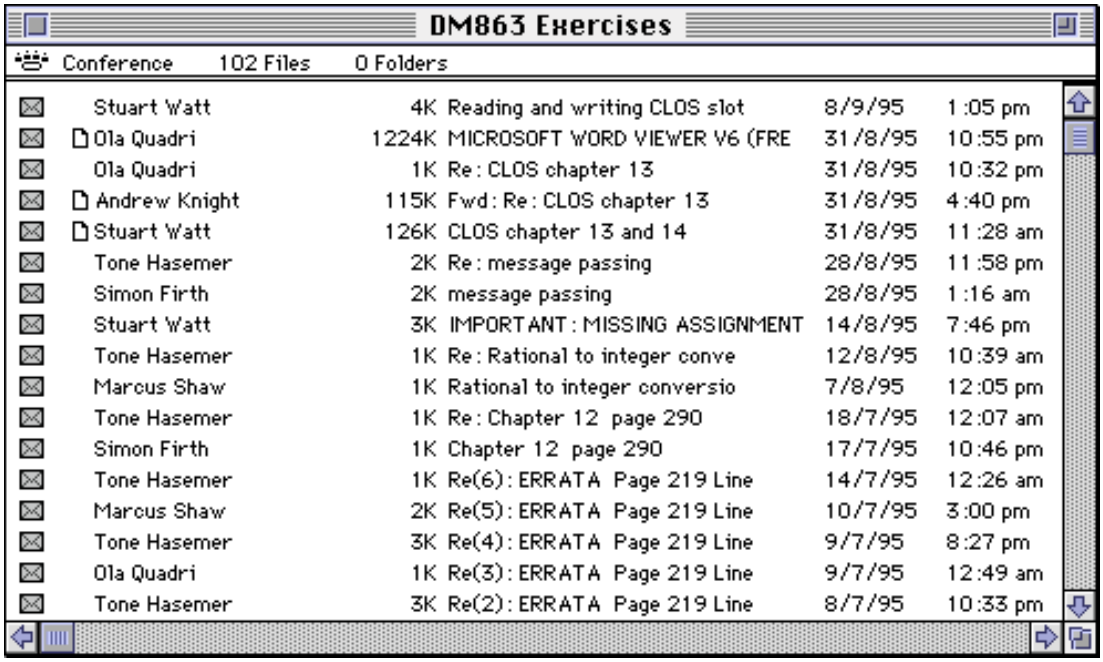

Figure 4. The corresponding conference in FirstClass.

This course suffered an extremely high dropout rate. We suspect that many students registered an interest while unaware of the substantial amount of work involved—and this course

is difficult even compared to many of the other courses in the same Masters' profile. However, the biggest drop was after the final assignment; only one student completed the exam. This was probably because (a) the course was offered in isolation, and therefore there was little benefit in the credit points that might be gained, and (b) the cost of the exam, which could not be waived with the very limited funding available for this experiment, was quite substantial for our many overseas students.

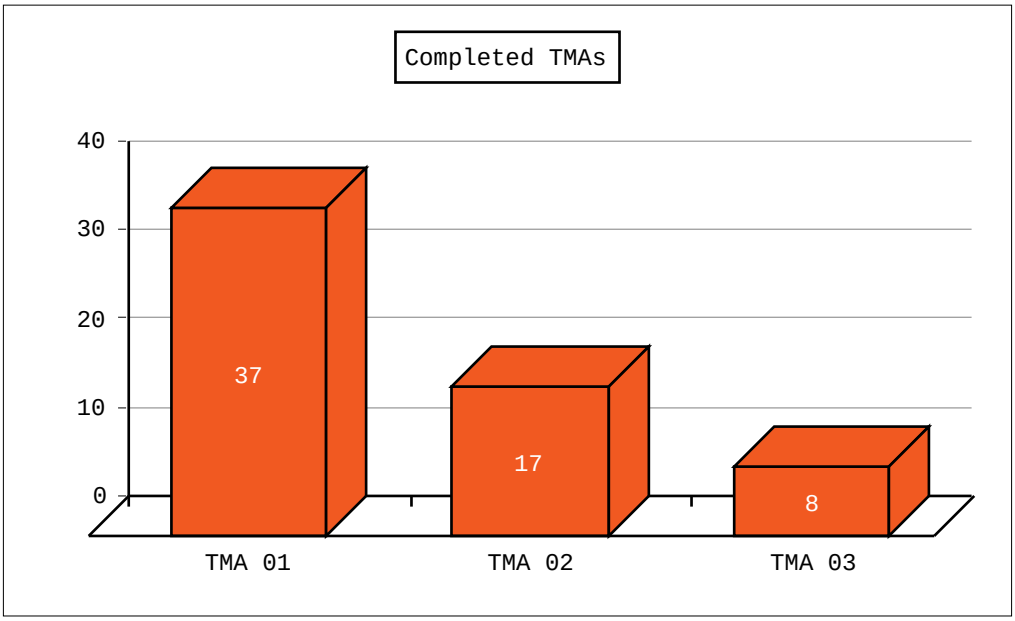

Figure 5. Student numbers for completed assignments.

Scaling the course

This course is never very big, and it is always possible for all the tutors and all the students to share conferences without losing the dynamics of interaction. We did feel, though, that there is often a optimal conference size, which depends on the interaction medium, but also on the electronic volubility of the individual students and tutors involved. We did, therefore, think quite seriously about the problems that would be associated with scaling a course of this type.

The result of these deliberations is the ring model shown in figure 6 below[2] . In this diagram the heavy border delimits one particular teaching conference. The darker shading shows the participants who normally interact through this conference; the lighter shading shows the participants who observe the conference, but who do not normally interact through it. In figure 6, for example, the course team can observe the teaching in every conference for monitoring purposes (although they clearly cannot intervene in private interaction between any two participants) but they will not normally actively participate.

On the FirstClass version of this course, there is a private conference which allows the students to communicate with each other privately, without the tutors able to see their messages. This was not mirrored on the mailing list system, partly because students could interact with one another privately anyway, but also because the University would have had to run the mailing list, and this could have prejudiced the privacy.

[2] These models were developed jointly during discussions on scaleability of electronically taught courses between Marc Eisenstadt and Stuart Watt.

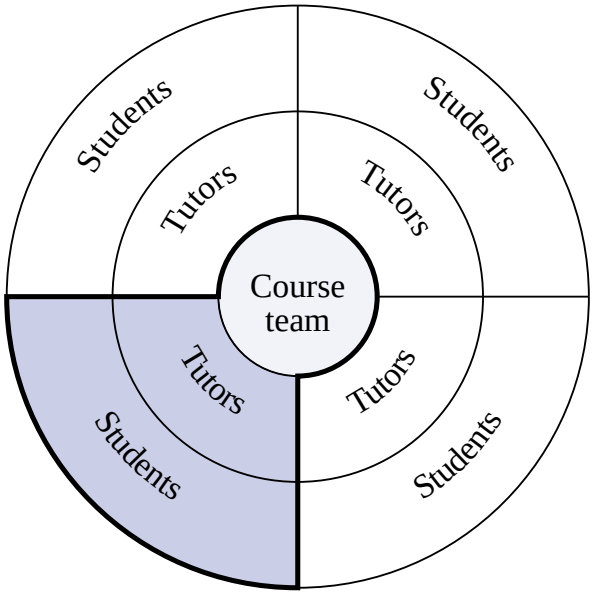

Figure 6. Basic ring model for teaching with electronic conferencing

This model can be complemented with other conferences, which allow the course team and the tutors to interact through the electronic media. This was not an issue for this course, because the tutors shared adjacent office space, and could talk directly when an issue arose.

A second model would be similar, but would allow other tutors to observe the teaching in the conference. This is shown in figure 7 below.

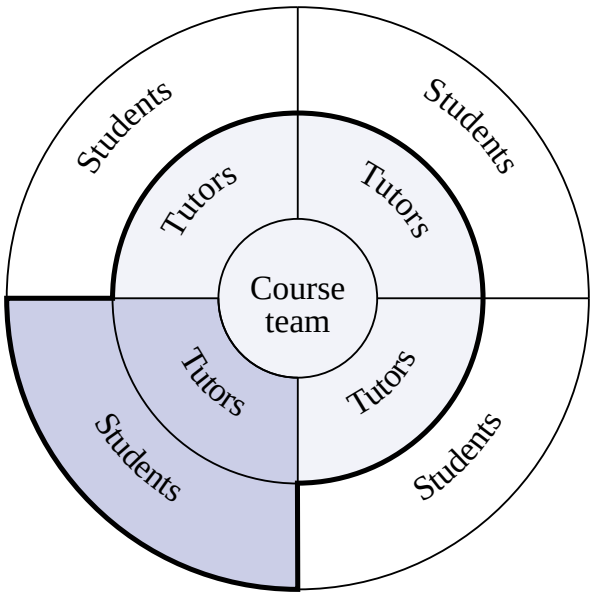

Figure 2. Enlarged ring model for teaching with electronic conferencing

With both of these models, the most important factor is choosing the right slice size, so the conference sizes are big enough that active discussion takes place. On DM863, both on the Internet and using FirstClass, the experience seems to be that at least 60 students are required to keep discussion reasonably active, and possibly more. This number might be more or less for different courses. With DM863, the student numbers are usually such that only a single conference is appropriate, so the model simply becomes layered.

Summary and discussion

This course was clearly not a success academically, in that the very high drop-out rate made the results exceedingly poor. However, as an experiment, it was very valuable, and many important lessons were learned.

There were some important successes. First, as the chart in figure 1 showed, the course did have a truly international set of students, with representatives everywhere from Aberdeen to Fiji. Second, it proved that, to an extent at least, it was possible to convert the course materials and to teach it through the very limited parts of the Internet that were used by the course. Third, it showed that it was at least possible to interact on a daily basis with students using this medium. The opportunities for this technology in remote areas seem promising, especially when used in combination with other approaches for interacting over a distance.

The down side of the course is that it clearly wasn't as successful as the normal approaches to teaching DM863 electronically through FirstClass. A lot of this may be due to the perceived benefit of the course being small to many of the external students. If it was possible to offer a full degree profile, the uptake might be much higher, and the dropout rate might be much lower. This remains a topic which needs a better market analysis.

In conclusion I would recommend that:

1. Careful attention should be paid to the design of the size and participation in the conferences used for electronic tuition. This seems to be an important factor in the group dynamics involved in tutorial groups, and it doesn't seem to follow the same patterns as tutorial group sizes for face to face tuition.

2. Where possible, distributing course materials electronically is useful, especially when this is complementary to sending them by post. Distribution of the materials was generally easy (except for very large amounts of data), it was also possible to do large mailings both quickly and directly by the course team.

3. Trying to teach purely using electronic mail is not really a good idea. For some students it works fine, but it is better complemented by mail, phone, and other communication media. This is because the human feeling of presence is not fully constructed by electronic mail systems on their own.

Acknowledgements

This version of 'Common Lisp for Artificial Intelligence' was made possible by many people, including John Domingue, Marc Eisenstadt, Jane Freebairn, Ben Palmer, Ortenz Rose, Stuart Watt, and Zdenek Zdrahal. This report, however, contains opinions that are solely those of the author, and should not be freely ascribed to others.