Journal of Educational Media, Vol. 28, Nos. 2–3, October 2003 Carfax Publishing Taylor & Francis Group

Using Blended Learning to Improve Student Success Rates in Learning to Program

TOM BOYLE[1], CLAIRE BRADLEY[1], PETER CHALK[2], RAY JONES[2] & POPPY PICKARD[3]

[1]*Learning Technology Research Institute and* [2]*Department of Computing, Communications & Maths, London Metropolitan University, and* [3]*Department of Computing and Electronics Technology, Bolton Institute, UK*

ABSTRACT *The aim of the project was to improve student success rates in learning to program. The project team introduced a number of changes in module organization, tutorial support and online resources. The blend represents a mixture of traditional and novel elements, with the novel elements more marked in the online developments. More than 600 students in two higher education institutions used the new blended learning environment. The results showed marked improvements in pass rates in both institutions. Detailed evaluation was conducted on students' use of the new environment. The results indicate a generally positive evaluation of the main elements of the blend, and widespread use of the new online features.*

Introduction

This paper reports on the development and use of a blended learning environment for learning introductory programming at university level. Modern programming courses at university level use industry strength languages. However, the resulting complexity and abstraction cause significant problems for students. Even universities with selective student intakes report significant problems (Jenkins, 2002; Jenkins & Davey, 2001). The problem is multi-faceted, especially with weaker students, requiring intense support at a number of levels to make a significant impact on success rates in programming.

At London Metropolitan University introductory programming is taught on three modular courses: B.Sc., HND and a conversion M.Sc. The student numbers are large with more than 600 students studying introductory programming across the three courses. Reflecting the national picture, there were problems in student

ISSN 1358-1651 (print)/ISSN 1469–9443 (online)/03/02&30165-14

DOI: 10.1080/1358165032000153160

retention and success in these modules. It was decided to initiate a project using a blended learning approach to tackle this problem.

The scale of the problem required that a full project be conducted, involving problem analysis, development, implementation and evaluation. The initial analysis phase investigated the nature of the multiple problems that the students encountered. The second major stage then involved the planning and development of the offline and online changes. This work took place over a period of 6 months. It involved changes in the organization of the face-to-face delivery and the development of a significant new e-learning facility. These changes were then integrated in the delivery of the courses over a period of one semester. A particular feature was the integration of conventional teaching with novel e-learning resources, including the use of learning objects. A thorough evaluation of the impact of the changes was conducted, which examined both student responses and the impact on module success rates.

The structure of this paper reflects these major phases in the project. The first section reports on the collection of information on the problems faced by the students. This is followed by a description of the major design changes made. An important element in these changes was the development of significant e-learning support. The third section of the paper discusses the integration of these changes into module delivery. This leads into a discussion of the evaluation of the project impact. This section reports on both students' subjective evaluation and the comparison of pass results against baseline data.

Identifying the need for change

The first stage in the project was to undertake a detailed analysis of the operation of the existing programming modules. This analysis involved collating existing statistics, canvassing academic opinion through focus groups, e-mail discussions and individual conversations. Student feedback was extracted from all the existing quality monitoring documents and from two questionnaires completed at the beginning and middle of the semester. The lecturer conducting the analysis also gathered informal feedback through her role as first year tutor. Current practice in the wider academic community was explored through publications.

Student numbers had increased by over 50% on the B.Sc. during the previous 3 years (from 347 to 529 starting the module). The module was delivered by dividing this cohort into three or four groups each with its own lecturer and tutorial staff. In line with school practice the module had a convener with overall responsibility, ensuring a common scheme of work and common assessment. From this common scheme lecturers usually developed their own lecture materials and practical exercises. Some lecturers put their materials on the Web; others, including the module convener, made materials available through publicly accessible directories on the university network. Tutorial staff worked in the same computer laboratory each week and students who regularly returned to the same laboratory would develop a working relationship with a particular tutor. However, the onus was on the student to do this as there was no formal tracking of student attendance or achievement

other than the module assessments, the first of which took place about halfway through the module. It was observed that some students moved between computer laboratories and occasionally between different lecturers. The first student questionnaire (completed by 50% of students starting the module) gave an overview of the student body. The mid-semester questionnaire (completed by 25% of those starting the module) sought student comment on their experience of the module. When asked how the module could be improved 25% requested more tutorial support. Some said greater attention should be given to students in laboratories; a few said the quality of the tutorial staff should be improved. About 20% cited the lectures saying they could be made more interesting, fun, have better content, and take a problem from its inception to final code. Lecturing on code alone was insufficient; 10% of students requested more exercises and feedback, including weekly graded exercises. About 10% felt no improvements were necessary.

Introductory programming had been taught using two languages. The students studied Visual Basic in semester 1, and then switched to Java in semester 2. The reason for this approach was to introduce students to an easier language before tackling Java. However, it was clear that this did not work effectively. Tutors complained that the students focused on the surface features of manipulating the screen interface in Visual Basic. They therefore did not get the experience in constructing underlying code that was essential in tackling Java. This was reflected in the module results where the failure rate in Visual Basic was unacceptably high. After discussion with the tutors it was decided that Java would be taught over both semesters. The development of new resources would be concentrated on this language.

A report on the finding and recommendations was produced. The report identified a number of problems ranging from course organization, through to the poor e-learning support. The School Management Committee considered the report and agreed that a major project should be implemented to improve the students pass rates in introductory programming.

Design of the new blend

A team was set up to plan the new modules and develop the resources required to deliver them. The team included the course tutors who would be teaching the modules, a research evaluator and the team leader. The services of a multimedia developer who worked for a central academic support unit in the university were also secured. The group started work in Spring 2002, preparing for new courses starting in September, and worked continuously through the summer period. The team's experience overall thus echoes the advice of Aycock *et al.* (2002) that effective blended learning requires a 'full course redesign'. They stress the need for 'release time' and 'summer contracts' to allow staff to properly integrate face-to-face and on-line material, both of which were required in our case.

The team's progress mirrored closely the model proposed by Voos (2003) who describes the four 'key factors' needed for successful blended learning. One, staff expertise is required in the use of a virtual learning environment and the production

of online teaching material. Two, we were able to choose from "the largest set of instructional methods and learning situations … to meet the specific needs of the discipline and the level of the course" (Voos, 2003, p. 4). Three, training courses were set up for tutorial assistants (some 20 in all) before the term started and, for students, guides were produced in the form of learning objects and lecture notes. Four, we ensured that university technical support and the computer laboratories were sufficient to meet the needs of the course.

In order to address the problems identified and to provide more support for the learner, changes were made in three main areas:

- a common curriculum was developed that would provide a core for all the modules;
- a new e-learning environment was created; and
- improvements were made to the organization of the modules.

Common curriculum

Although the modules formed parts of a number of different courses, all were aimed at students who had little or no exposure to computing. Certainly, no experience in programming was assumed. Each module would, therefore, begin at the same point and would be required to cover a common set of topics; the overall scope and pace might vary from one module to another but the core would be the same.

The decision to design a common curriculum had a significant effect. Instead of having a number of individual academics producing materials for their own modules, a team-based approach was taken. Thus, once the overall structure of the curriculum had been agreed, sections of it were allocated to team members for development. This approach allowed staff to concentrate on the development of a subset of the materials and this supported a richer development process. The team-based approach naturally facilitated peer-review of an individual academic's work, as the team approved the adoption of all of the material produced before it was deployed.

It was felt that in traditional courses students were overwhelmed by large amounts of material, while at the same time they had to come to grips with concepts that might be quite foreign to them. In order to reduce the overload in material to be assimilated, a spiral approach to the curriculum was adopted where a topic is first introduced in a limited way and then re-visited and elaborated on later (Boyle, 1997). The advantage of this approach, to novice programmers, is that they do not have to cope with such a large volume of material in order to start writing meaningful programs. They learn just enough programming to deal with the exercises that are set for them and then return to enhance their understanding of particular topics at a later stage. The consequence of this approach is that the repetition of similar topics throughout the syllabus allows concepts to be reinforced and thus more effectively embedded in the students' knowledge.

Teaching programming requires the introduction of a number of abstract concepts from the very beginning and this can be difficult for students to cope with

(Kölling & Rosenburg, 2001). In order to support students in understanding these concepts, a 'visual approach' was adopted where programs produce graphical shapes. Changes in program code are reflected in visible changes in the attributes and behaviour of these graphical objects (a similar approach had been taken, with some success, at Williams College by Bruce *et al.*, 2001). This change was introduced to facilitate novice programmers' understanding of program behaviour. About halfway through the modules a switch was made to the more traditional non-visual command line environment.

New e-learning environment

To support the new syllabus, new learning materials were developed by the academic team along with multimedia developers. The materials were developed as fairly small 'learning objects' (Bradley *et al.*, 2003). These learning objects included sophisticated animations that demonstrated particular programming constructs or operations. The team identified topics that were traditionally difficult to learn for the initial round of learning objects, particularly in the area of object-oriented programming.

It was agreed that the materials would be presented through WebCT, the university virtual learning environment (VLE). WebCT provides a structured way to present information about a module online. The WebCT site would give the overall organization of the module articulated into separate weekly blocks. It would provide access to online materials and act as an electronic collection point for the exercises that students complete on a weekly basis. WebCT would thus provide a unified access to the online resources.

Organization of course delivery: a blend of old and new elements

Before the start of this development, there was a debate among the team about the efficacy of lectures in a practical subject such as programming. There was a suggestion that the online materials could replace lectures altogether. However, the consensus of opinion was that there were two main reasons for keeping them: first, there were some theoretical topics that were best dealt with in a lecture; and second, it was useful for students to be in contact with the lecturer, who is seen to be the subject expert, on a regular basis and not just interact with the teaching assistants.

It was decided to reduce the staff student ratio in the teaching laboratories from approximately 25: 1 to 15: 1. Thus was agreed by the School Management Committee. It was argued that improvements in student retention and pass rates would justify the extra expense involved. It was envisaged that the extensive new e-learning environment would also provide a common set of information for all the teaching assistants working on the course.

Thus, the re-designed course was a blend of traditional methods of delivery (the lectures and the practical laboratory sessions) but with significant changes including the development of considerable e-learning support. The organization of the course followed a traditional week-by-week delivery of topics and assessment, but the

provision of online copies of all material and the additional learning objects meant that students could return to earlier material whenever they chose.

Delivery of the new blended learning environment

The delivery of the new modules commenced in late September 2002. At London Metropolitan University three modules were affected: B.Sc., HND and conversion M.Sc. The combined number of students involved was more than 600. Slightly different blends of the core materials were delivered to the undergraduate students, the masters' students and the HND (non-degree) students. A member of the development team changed posts during the summer. She implemented the blended course at her new institution, Bolton Institute, where 120 B.Sc. students used the new teaching and learning environment. Again there were variations to suit local circumstances. In Bolton all the laboratory sessions were covered by the lecturer and one other tutor, whereas in London there were a large number of teaching assistants (many part-timers) that needed to be managed.

The teaching assistants in London were offered two training sessions before the semester started. One session was devoted to the Java programming environment and the pedagogic approach of the course, including the use of the learning objects. The second was a session about the VLE chosen for the online delivery of the course, including online assessment. This was WebCT, which was also used in Bolton. The training sessions also addressed the face-to-face aspects of the blend: managing the tutorial group, and the methods of assessment.

Additional just-in-time training sessions were offered to teaching assistants during the semester to ensure conformity of marking assignments and using the VLE for feedback. Regular communication was maintained between module leaders and teaching assistants through meetings during tutorial times and weekly e-mail reports. A staff resource website gave support to teaching assistants (many of whom had never taught Java before) and model solutions provided for all exercises set.

The basic structure of the modules followed the traditional format. The students had a weekly lecture. During this lecture they were given a task sheet (one side of A4) that set out the laboratory tasks and assessed exercise for the week. The PowerPoint slides for the lectures and the task sheets were also mounted in the WebCT site for the module. Each module maintained its own WebCT site. These sites followed a similar structure and re-used the common material developed by the group. They contained variations to suit student needs. For example, the HND students moved at a slower weekly pace than the B.Sc. students. Answers to exercises were submitted online through WebCT. The new learning objects (both text-based and multimedia) developed for the courses were also accessed through the WebCT environment. This was the most innovative aspect of the new blend. The design and use of the learning objects are described in detail in other papers (Boyle, 2003; Bradley *et al.*, 2003). The students could access all the online learning material from home as well as from the university laboratories. Examples of these learning objects are available online (http://www.unl.ac.uk/ltri/learningobjects).

The major programming assignment, also submitted online, was graded according to a viva held in the final tutorial, which authenticated the online submission and provided a major barrier to plagiarism. Plagiarists were also caught out by the online quizzes and the final examination (also automatically marked multiple-choice questions). Thus, the assessment strategy contributed to the blending of the face-to-face and online environments and the balance between the two was subject to much discussion in the project team and varied between the four modules.

In summary, the new blended learning course delivery consisted of:

- integrating the traditional face-to-face approach with on-line delivery;
- providing support for students in tutorials and through e-learning materials;
- supporting the learning of difficult topics by providing learning objects; and
- monitoring performance through continuous assessment that was managed online and subject to viva.

Evaluation

A detailed evaluation of the project was conducted. A researcher, who was not part of the teaching team, was appointed to carry out this evaluation. The evaluation used a series of techniques to study the effectiveness of the new blended learning environment and its major components. A series of three online questionnaires were given to the students, at the beginning, middle and end of the semester. These questionnaires were supplemented with short structured interviews with 36 students in Week 5. This provided richer qualitative information than that provided by questionnaire data.

Towards the end of the course the teaching assistants and module leaders were also asked for their views via short questionnaires delivered by e-mail. This produced rich, qualitative feedback. The teaching assistants delivered the laboratory part of the course, but were not part of the course development team. We also gathered extensive tracking data provided by the WebCT system on the student use of the online resources. Finally, we compared the pass results on completion of the modules with the baseline results for the previous year.

Overall module results

Table I reports the percentage increase in pass rates for the new blended learning modules compared with the results obtained in the previous year. The improvements obtained across both institutions are dramatic.

The impact of the introduction of the new blended learning environment seems to be substantial. The detailed evaluation carried out during the operation of the modules provides a richer picture of the factors that may have contributed to this positive outcome. More detailed evaluation and analysis was conducted at London Metropolitan University, and it is this data that are reported here. The discussion of the information obtained is grouped under the three areas of curriculum changes, organizational changes and the new e-learning environment.

TABLE I. Increase in pass rates

Course	Increase in pass rate (%)
HND London Metropolitan	+ 19
B.Sc. London Metropolitan	+ 15
B.Sc. Bolton Institute	+ 23
M.Sc. London Metropolitan	+ 12

Curriculum and organizational changes

A significant innovation in the curriculum was the use of a graphics-based approach in the first half of the semester. A special graphics library constructed for the course supported this approach. This resource enabled novice students to construct programs that produced and manipulated simple graphical shapes. In Questionnaire 2 (mid-semester) the students were asked what they thought about this graphics-based approach to learning programming. The results are summarized in Figure 1, and show that 35% of the learners thought this was a 'very good' approach, with 95% judging the approach as 'good' or 'very good'. Observation in the programming laboratories supported this finding of a positive response by the students to this innovation.

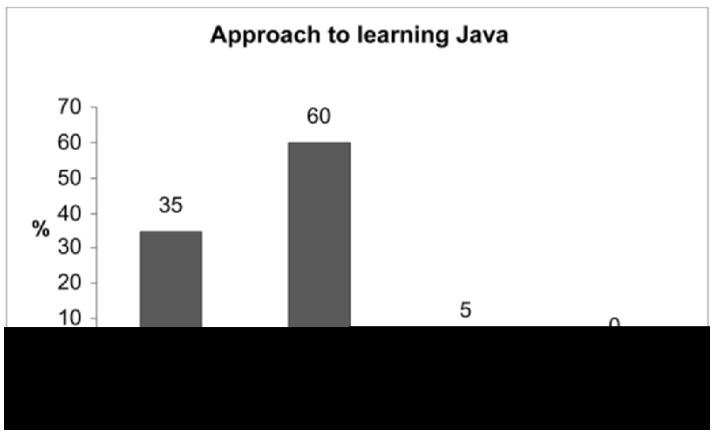

FIGURE 1. Approach to learning Java.
Source: Questionnaire 2 (223 responses).

At this stage in the semester, we also asked students how they felt about their progress to date, and their motivation towards learning to program in Java (see Table 2). As this is the stage in the module when students are likely to drop out if they are struggling with the subject or are unhappy, these results are encouraging. While 22% are disappointed with their progress, the fact that they express high levels of motivation indicates that low satisfaction with progress can be overcome.

One organizational change was not as effective as the team had hoped. The aim

TABLE II. Student level of progress and motivation

	Student level of progress and motivation at the mid-semester stage			
Progress	Very happy, 9%	Happy, 66%	Disappointed, 22%	Very disappointed, 3%
Motivation	Very high, 28%	High, 56%	Low, 15%	Very low, 1%

Source: Questionnaire 2 (223 responses).

to reduce the ratio of students to teaching assistants to 15: 1 and to maintain continuity of tutor throughout the module was successful in some respects but not in others. We were able on the whole to maintain this lower ratio by employing more teaching assistants, but trying to retain consistency of students to teaching assistants was more problematic. Some of the larger computer laboratories that were used made this difficult as multiple student groups were housed in the same room, and some students kept switching between teaching assistants.

The online learning environment: WebCT and the learning aids

The module leaders used WebCT extensively for student management and delivering assessments (including the final examination). They found WebCT a valuable tool for helping to manage the large numbers of students taking the module. The teaching assistants, who covered the laboratory classes, generally viewed the use of WebCT positively though some had reservations about the limitations of VLE systems such as WebCT.

Figure 2 shows the distribution of WebCT log-ins by the 304 B.Sc. students who completed the module, taken from the system tracking data. Each bar indicates the number of students making that number of accesses to WebCT. A total of 127,190 discreet log-ins have been recorded, giving a mean average number of accesses per student of 418 over the semester. However the range and variation of individual log-ins is broad: two students accessed the system five times, and one student made 1469 visits. The online environment was thus used extensively.

The development and introduction of learning objects was probably the most innovative aspect of the project, and considerable attention was paid to evaluating their success. These learning resources each focused on one clear learning goal (often a basic technique or construct in the language). These learning objects were developed in html and Flash, and were designed to be reusable outside this particular course. The learning objects were stored on a separate server from WebCT and loaded into WebCT at runtime. To the students, however, they appeared a part of the WebCT environment, and were viewed as 'learning aids' in that environment. Questionnaire 3 (end of semester) showed a positive rating of the learning aids by the students (Figure 3). This finding was supported by observation in the laboratories.

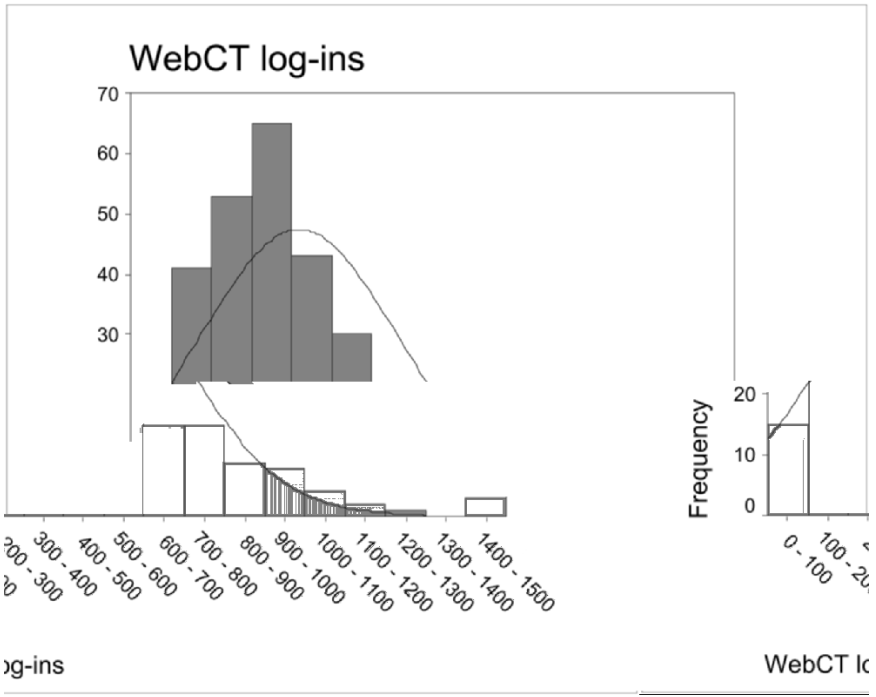

FIGURE 2. Distribution of WebCT log-ins among B.Sc. students.

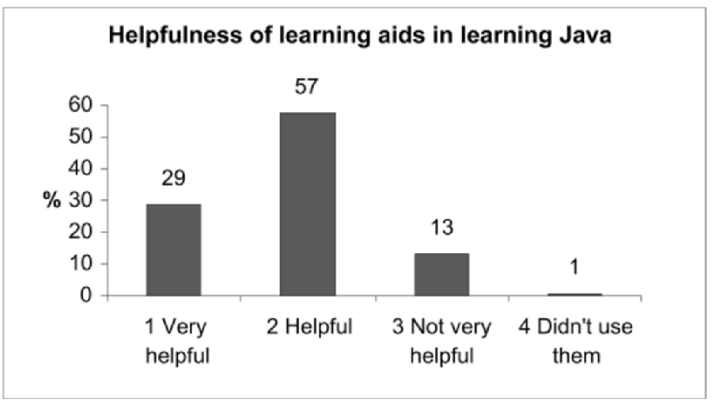

FIGURE 3. Helpfulness of learning aids in learning Java.
Source: Questionnaire 3 (117 responses).

Which aspects of the course blend are most successful?

In Questionnaire 2 (mid-semester) we asked students how useful they found each of the major teaching components: the lectures, laboratory exercises, textbook, and text and animated learning aids. The results are presented in the bar chart in Figure 4.

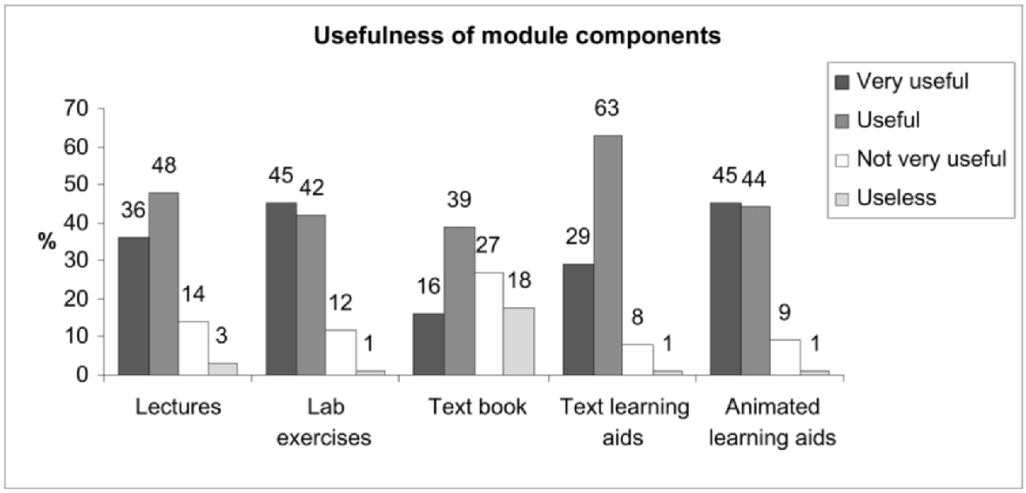

FIGURE 4. Usefulness of module components to students.
Source: Questionnaire 2 (223 responses).

Animated learning aids and laboratory exercises receive the highest ratings for 'very useful' at 45%, followed by lectures (36%) and text learning aids (29%), but the text-based aids had by far the highest rating for 'useful' at 63%. The textbook attracted the least useful ratings overall.

Table III. What students liked most and least about the module

Module aspect	Number of students who liked this most about the module	Number of students who liked this least about the module
Learning Java/programming	13	0
Lectures	6	7
Tutors/tutorials	4	8
WebCT	8	0
Learning aids	2	0
Text book	2	1
Teaching/teaching method	7	2
Graphics approach (Weeks 1–5)	12	5
Non-graphics approach (Weeks 6–10)	14	9
Practicals	5	0
Creating working programs	8	1
Exercises/tasks/assignments/assessments/	12	6
Pace of module	0	8
Nothing	1	8

Source: Questionnaire 3 (117 responses).
Note: Table reports the categories of response using the students' own terms. Hence there is scope for overlap in the categories used. It seemed better to capture the students' own responses rather than collapse them into categories of the evaluator's choosing.

In Questionnaire 3 (end of semester), students were asked separate questions about what they liked most and least about the module. These were free text answers, in which the students could enter their own response about anything to do with their module. These comments have been grouped into categories, and recurring aspects of the module are summarized in Table III.

These results indicate that a blended learning approach provides for a range of likes and dislikes for students. What one student likes the most, another dislikes. As our student population is large and quite diverse in terms of prior experience of programming, this approach can provide for the needs of a larger body of students overall, and enables students to learn from the approach that suits them best. Similar experiences have been noted elsewhere, for example in students' preferences for text-based or multimedia learning aids.

Conclusions

This paper has reported on the use of a blended learning approach to tackle a significant problem – the teaching and learning of introductory programming. The nature of the changes made was influenced by two factors: the analysis of the problem, and pedagogical ideas for producing improved learning experiences for the students. The blended learning environment resulted in marked improvements in pass rates and positive student evaluations. Evidence from increased student attendance and consistently high levels of usage of the online resources supports the argument of Aycock *et al.* (2002) that the hybrid course, another name for a blended environment, "increases student engagement".

There are several advantages in deploying a blended learning approach. When a problem is multi-faceted, tackling only some problems and leaving others untouched is unlikely to be successful. The blended approach taken operated by identifying each of the significant problem areas and developing methods and resources to tackle these problems. Sometimes the appropriate changes seemed to reside in the (offline) organization of the course. A key issue, for example, was improving tutorial support given to students. For other areas the development of specialized e-learning resources seemed to be required. The overall blend aimed to provide appropriate support and resources to help students deal with the range of problems they faced.

There is a major issue in handling course transition. Both staff and students are used to a particular format for teaching and learning. Too abrupt a change may produce resistance from tutors and/or students. This resistance could undermine the impact of new methods introduced to tackle the educational problem. The blended approach thus consists of a mixture of traditional and new approaches. In general, the traditional approaches are more associated with the offline running of the course and the novel elements are in the e-learning components. The blend aims to provide a stable transition of familiar and new features. This is an important characteristic of the overall blend produced.

This produces a potential limitation in that the blend must retain a significant conservative element. This could, over time, restrict progress. However, this is only a problem if the blend remains static. It is the team's view that the blend should be

dynamic and evolve over a number of years. The role of evaluation is crucial in this respect. For logistic and tactical reasons, only a certain number of changes could be made in the project. However, the evaluation conducted provides evidence for the success or failure of the new elements introduced in the blend. Demonstrable successful changes can be built on and extended in subsequent cycles of development and delivery. The strategic approach, therefore, is one of continuous improvement, where subsequent implementations can build on and extend successful features of the blend. In this way the blend can evolve dynamically, in a managed way, towards more innovative and successful configurations.

Following this approach, certain changes have been made for next year's delivery. The graphical approach was very successful with the students. This has been extended from the first half of the semester to the whole semester. The original textbook has been replaced by one that takes an explicitly graphical approach to introducing Java (Bell & Parr, 2002). Crucially the majority of the learning objects have not been affected by these changes, as they were explicitly designed to be reusable (Boyle, 2003). The 'reusability' of learning objects provides a powerful feature for blended learning, as the same objects can be reused in a variety of different blends. As anticipated, new learning objects have been developed and added to augment the list. The blend is thus evolving, and the changes made will be monitored and evaluated to inform the future development of the course blend.

Acknowledgements

The authors would like to acknowledge the support of the LTSN Centre for Information and Computer Sciences who provided funding support for the project.

Note

Examples of the learning objects mentioned in the paper are available online (http://www.unl.ac.uk/ltri/learningobjects).

Notes on Contributors

TOM BOYLE is the Director of the Learning Technology Research Institute (LTRI) at London Metropolitan University. He has written a book and over 100 refereed papers in the areas of multimedia, and learning technology.

CLAIRE BRADLEY is a Research Fellow in the LTRI at London Metropolitan University. She has worked during the past 7 years on a number of UK and European research projects involved in e-learning, online communities and in the general use and application of ICT in teaching and learning.

PETER CHALK is an Academic Leader at London Metropolitan University with research interests in Web-based learning.

RAY JONES is a senior member of the teaching staff in the Department of Computing at London Metropolitan University. His research interests are in the re-use and re-purposing of learning objects, and the application of XML and related technologies in learning object design.

POPPY PICKARD has 20 years' experience of teaching in further education and secondary schools, and has worked for more than 10 years in higher education. Her research interests are in pedagogy and e-learning.

Correspondence: Professor Tom Boyle, Director of the Learning Technology Research Institute, London Metropolitan University, 166–220 Holloway Road, London N7 8DB, UK. Tel: +44 20 7749 3757; fax: +44 20 7749 3781; E-mail: t.boyle@londonmet.ac.uk; website: http://www.londonmet.ac.uk/ltri

REFERENCES

AYCOCK, A. GARNHAM C. & KALETA R. (2002) Lessons learned from the Hybrid Course Project, *Teaching with Technology Today*, 8. Available on-line: www.uwsa.edu/ttt/articles/garnham2.htm (accessed 6 March 2003).

BELL, D. & PARR, M. (2002) *Java for students*, 3rd edition (Harlow, Prentice Hall).

BOYLE, T. (1997) *Design for multimedia learning* (London, Prentice Hall).

BOYLE, T. (2003) Design principles for authoring dynamic, reusable learning objects, *Australian Journal of Educational Technology*, 19(1), pp. 46–58. Available on-line: http://www.ascilite.org.au/ajet/ajet19/boyle.html (accessed 30 July 2003).

BRADLEY, C., BOYLE, T. & HAYNES, R. (2003) Design and evaluation of multimedia learning objects, in: D. LASSNER & C. MCNAUGHT (Eds) *Proceedings of Ed-Media 2003, 23–28 June 2003, Honolulu, Hawaii, USA* (USA, AACE).

BRUCE, K., DANYLUK, A. & MURTAGH, T. (2001) A library to support a graphics based object-first approach to CS1, *Proceedings of SIGCSE 2001, 21–25 February, Charlotte, North Carolina, USA* (New York, ACM).

JENKINS, T. (2002) On the difficulty of learning to program, *Proceedings of 3rd annual conference of the LTSN-ICS, 27–29 August 2002, Loughborough* (University of Ulster, LTSN Centre for Information and Computer Sciences).

JENKINS, T. & DAVY, J. (2001) Diversity and motivation in introductory programming. *Italics*, 1(1). Available on-line: http://www.ics.ltsn.ac.uk/pub/italics/issue1/tjenkins/003.html (accessed 30 July 2003).

KÖLLING, M. & ROSENBURG, J. (2001) Guidelines for teaching object orientation with Java, *Proceedings of ITiCSE 2001, 25–27 June, University of Kent* (New York, ACM).

VOOS, R. (2003) Blended learning – what is it and where might it take us? *Sloan-C View*, 2.1. Available on-line: www.aln.org/publications/view/v2n1/blended1.htm (accessed 30 July 2003).