A Project Approach to Teaching and Learning with Technology: A Case Study with Microworlds Project Builder

John Eklund
Faculty of Education A35
University of Sydney 2006
eklundj@edfac.usyd.edu.au

Abstract

The Faculty of Education at the University of Sydney has introduced a new course entitled Computing Studies 1 for the second year Technology and Applied Studies (TAS) strand of the Bachelor of Education. This is a three hour per week full year course which is preparatory in content and teaching methodology for the Computing Studies courses in Junior Secondary School. This course uses both Microworlds Project Builder (MPB) and HyperCard as applications through which course outcomes are realised.

This paper focuses on the MPB section of the course, introduces the software and investigates its potential as a multimedia authoring environment, a teaching tool, an appropriate platform to achieve specific course objectives (both in the Junior Secondary Syllabus and in the university TAS strand), and finally as an educational philosophy based on Logo. Its potential as an across-curriculum tool is investigated, and the overall structure of the course in developing student-centred approaches to teaching and learning with technology are detailed.

The extent to which elements of content and methodology are derived from a ‘new basics’ approach, stressing the development of metacognitive and procedural skills, is placed in the perspective of the Secondary Computing Studies Syllabus. The paper presents elements of the course evaluation which underscore desirable changes in student attitudes to teaching and learning with technology.

Keywords

Logo, teaching and learning, student-centred, authoring, curriculum.

1. Acknowledgements

Many thanks to Dr John Harvey for his ongoing efforts to improve the teaching and learning in the course on which this paper is based.

2. Introduction

The corpus of academic writing on the classroom use of technology and the role of the teacher is quite large, and informs us about practical techniques to attain curriculum goals. This area remains a fruitful one for action research as to the nature of the relationship between the teacher, the learner and the content. In particular, giving students responsibility for their own learning, working collaboratively on projects and problem-solving are process skills that are being emphasised in recent curriculum documents, which refer to the importance of developing critical thinking and reasoning skills that can be used to solve problems that arise outside the classroom. Computers seem to have a special role to play in developing investigative skills and are transforming the more traditional forms of the teacher-learner relationship (McCune, 1992). An instance is the Secondary Computing Studies Syllabuses, which have as their central aim to instill in students the ability to design, create and implement computer-based solutions to real-world problems. In *Mindstorms*, Papert (1980) emphasised the importance of concrete learning experiences and how they allow students to develop a personalised understanding of many abstract concepts, as well as to aid in the critical thinking process. The relationship between student controlled learning and the development of such skills is that '...problem solving is more than just following predestined pathways or making yes / no choices, it involves children being in control of their learning and having freedom of choice within clearly defined parameters.' (Jarred, 1993, p. 31) Even in an environment where the Logo language is being used as purely recreational, students are exploring, experimenting, analyzing, predicting and planning. These thinking skills have been developed and strengthened by using Logo and, according to Papert, this natural mode of learning is far superior to that of conventional, instructional techniques. Coding, exploration, prediction, creativity, planning and debugging are all constantly being used in these microworlds. Reiber (1995) describes a microworld as '...one of the few constructivist examples of a learning environment [to have] some real "teeth" in it.... one doesn't just study content or ideas, one lives them.' In these environments the student is using the computer as an extension of their capabilities, a tool to solve problems, and all the time acquiring a sense of mastery over modern technology.

There is little doubt that the power of Logo, combined with the enhancements that Microworlds Project Builder provides, could be used to develop problem-solving skills in older students, such as those in the second-year course at The University of Sydney. The action-research question that was posed in this study was whether those attitudes to teaching with technology could be communicated to the students.

3. The course at The University of Sydney

Our challenge was to introduce a course for second year Education students which demonstrated this approach to teaching and learning with technology to the students by teaching with it. As platforms we chose the software *Microworlds Project Builder* (MPB, described in some detail later in this paper), and *HyperCard*, and the course was delivered to groups of about 15 students over 3 hours per week for 12 weeks for each. A feature of the course was the emphasis on a major project, and specifically the development of the project through the submission of a project proposal, a draft, a written report and a demonstration. This aspect of the assessment purposefully reflects current practice within the Technological and Applied Studies (TAS) Key Learning Area (KLA), and places a priority on the process of learning. This course focuses on the use of applications software for teaching computing studies in Secondary Schools, it is not a programming course, but has the additional benefit of using *HyperCard* and *Project Builder* as applications which are based on programming languages.

The course has as its main outcomes:

- to increase student's familiarity with particular applications software likely to be found in NSW schools
- to enable students to experience certain student centred approaches in teaching and learning with the software, and to apply this to their own practicum and later teaching experiences
- to integrate the use of the software with the TAS KLA through the development of a major project using HyperCard and Project Builder

- to develop skills in an authoring environment based on an approved programming language
- to give students an understanding of how using applications software can be indirectly integrated into teaching programs developed from both the senior and junior syllabuses in Computing Studies

4. Course delivery

The course was taught with minimal direct instruction, with the assessment project a focus from the beginning. Students were asked to design an environment and ‘make it happen’ through their own investigations. The MPB software epitomises the problem-solving ‘project’ approach to learning, and is best taught through that same approach. The motivation was intrinsic: their learning was driven by their desire to know how a particular action could be made to work, and to this end the teacher, the pre-built example projects and the reference manual were the resources available to them to solve the problem. Some initial instruction on the use of the environment was given, where students familiarised themselves with the basic tools it provides through a series of more structured exercises. Then a design brief for the project was introduced, and students were free to be creative, investigate possibilities, test ideas and enhance their project with the teacher’s help, that of each other, or through reference to the manual.

5. The project

Students were asked to create an environment using MPB within two main parameters: It had to relate to an aspect of the TAS syllabus, and it had to incorporate a sense in which the user of the environment was learning by investigating. Students were required to prepare a word-processed submission outlining the environment that was intended; discuss the ways that it may be used in teaching and learning TAS and what possible classroom settings it would support; make a list of the primitives used; include a statement as to how it works; list possible scenarios students would use in the project and how the project deals with them; incorporate ‘screen shots’ of the project pages into the submission; list improvements that were made or provide a time-line showing how the idea embodied in the project had developed; discuss the elements that have been able to be incorporated, and those that were not, and why; make suggestions for further work or improvements; include a list of references where necessary; and finally to provide a demonstration of the project to the class.

Students were also asked to apply the additional criteria:

- does the chosen area provide enough scope to incorporate a variety of functions of the Microworlds / HyperCard environment? That is, can a number of different techniques, keywords, procedures etc. be used in the context of the chosen project?
- how will the final project be used in a classroom situation? Does it allow for flexible teaching styles or learner-centred activities?
- why is MPB particularly suitable for the chosen project? Could it be better achieved using another authoring / programming environment?
- is the chosen project one which allows students flexibility in its use? Can it easily be adapted to a variety of teaching and learning situations?
- does the chosen project allow for continued development or is the concept which underlies the project exhausted by the implementation?

These project criteria formed the basis for the assessment for the course. Ultimately the project became a problem solving exercise for those students who selected an idea with enough scope for development.

6. The software

Microworlds Project Builder™ (MPB) (Logo Computer Systems Inc, 1993, A$130 single or site license $995) is a Logo based authoring environment '... designed for grade levels 4 to 8' (LCSI, 1993). At The University of Sydney, we are using it effectively with students in Year 2, some 7 to 11 years older! It not the intention of the author to review the software in this paper, (*see* Eklund, 1995 for a more comprehensive review), but it is necessary to briefly describe the software for those readers unfamiliar with the enhancements MPB provides over traditional forms of Logo.

MPB combines the command based graphical environment of the more traditional form of Logo with a set of tools which significantly enhances the environment, empowering students to create simulations, courseware and multimedia projects with relative ease. Its features include a tool palette with a select tool, a shapes center, a paint center, a toggle to the procedures page, a hatch turtle tool, a text box tool, a button tool, and a tool for creating a slider. The shapes center contains a number of pre loaded shapes which may be edited. Shapes may also be imported from clipart using the clipboard. These shapes may be used to stamp onto a turtle to change the turtle's basic shape, or to decorate the page. The paint center contains basic paint tools and many colors, which may be programmed to interact with the turtle or the mouse. The procedures page is a page where all of the Logo procedures are written. The hatching turtle allows the student to generate multiple turtles, each of which may wear a different shape and be programmed to carry out a task. The Text box tool allows the student to create text-boxes on a page, the text may be fixed or variable by writing procedures to talk to a particular text box. Buttons may be created and named. Once a button is named it may carry out an instruction when pressed. eg: We could name a button **fd 5** and the turtle would move forward 5 spaces (either once or many times) or we could name a button **go** and define the procedure **go** on the procedures page. Sliders may also be created which allow the value of a variable to be input by the student. The name, minimum and maximum value of the slider is defined in the slider set-up dialogue box.

7. Results and conclusion

Results were obtained from three sources, the student course evaluations, our own classroom observations and marking of assigned work, and through informal discussion with students. The course evaluation forms offered a number of questions about the course and the teaching in the course with a rating of All of the time, Most of the time, Half of the time, Little of the time, and None of the time. The specific aspects of the teaching evaluations which relate to the attitudes of students to teaching with technology are mentioned in this section.

The results for the first semester were encouraging, with students producing high-quality work and gaining excellent skills in using the software, skills which will be easily transferable to other environments. Students clearly enjoyed the course and were enthusiastic about their work, some in particular spending a good deal of time beyond expectation in order to enhance their projects. Further, the course evaluations showed the students were complimentary of the autonomy that they were given. They appreciated the time to think in class, work at their own pace and develop their own ideas within the confines of the requirements for the project. In informal discussions with students it was ascertained they generally found the teaching styles presented entirely appropriate and that they would use similar techniques when teaching themselves. Importantly, they understood the link between the teaching methodology used and the content. They also seemed to interpret the autonomy as "the lecturer treats students with respect", and "responds to student needs" (83% All of the time, 17%

most of the time). By the end of the twelve-week semester, they did not interpreted the minimal amount of direct teacher-instruction as a lack of structure, with the students reporting that “the lecturer was well-prepared for each session” (17% all of the time, 84% most of the time). Students also reported that they had “stretched their minds” (50% All of the time), and acknowledged that the skills they had acquired and on which they had been assessed had been process-oriented.

The study of teaching and learning with technology is an excellent area in which to engage in action research, where the results of the enquiry may be fed back into the course to improve outcomes for students. The use of an authoring environment such as MPB, together with a student-centred teaching approach and a major project has shown in this instance to be an effective method of attaining specific course outcomes and linking these to syllabus aims. It is ultimately also an excellent way of teaching problem-solving.

8. References

Eklund, J. (1995). Logo the next generation: Microworlds Project Builder reviewed, *Information Transfer*,Vol. 15, pp.38-39.

Jarred, A. (1993). Can computers stand the test of PEWIT? *Information Transfer*, Vol. 13, No. 2, pp.31-34.

Logo Computer Systems Inc. (1993). *Microworlds Project Builder™* .

McCune, S. (1992).*Creating the future: Perspectives on educational change*, London:Aston Clinton.

Papert, S. (1980). *Mindstorms: Children, computers and powerful idea*s, Harvester Press Ltd.

Reiber, L. (1995). Re: Web threading... interjection and frustration, *ITFORUM communication*, 9.02 AM 24/8/95.