Investigations
in university teaching and learning vol. 6 (1) spring 2010 ISSN 1740-5106

Reflective Practice for Programming Students: adjusting the blend to improve skills

Dafna Hardbattle, Ken Fisher & Peter Chalk
Faculty of Computing
London Metropolitan University

Keywords: *blended learning, learning object, reflective practice, reflective writing, programming*

Context

This paper reports on the findings of a study to engage first-year programming students in reflective practice by means of an online interactive learning object (LO). The aim of the newly developed LO is to allow students to learn online, in their own time, about reflective practice (RP) in relation to writing about Java programming problem-solving tasks.

This first-year, first-semester, 'Introduction to Programming' module in the Faculty of Computing (FoC) at LondonMet has been taught in a blend of face-to-face and selected web-based technologies. With the aim of improving students' learning, text-based LOs authored as HTML pages and multimedia-based LOs authored in *Macromedia Flash* were incorporated into the module back in 2002 as a successful project by Boyle *et al.* (2005). Since then, however, the module has been substantially transformed. The core text has twice been replaced and new online materials developed. Throughout, however, relevant online materials have been linked into the syllabus weekly; including LOs, which provide visual examples of abstract concepts in Java programming.

Until recently, the assessment instruments for the module mainly focused on students' technical skills. However, it has been acknowledged that RP can encourage critical thought about what has been experienced and could be used to improve students' thinking processes, but it was largely missing from the computing discipline (Beale, 2007) and more specifically in the FoC (Chalk & Hardbattle, 2007).

A recent review of all Faculty modules enabled us to change the blend. Personal reflective reports were incorporated into the assessment process to give students the opportunity to engage in RP and provide insight into their level of understanding. The new RP learning object (RPLO) was developed to introduce students to the concept of RP and to support them with their reflective tasks.

The aim was to improve students' reflective writing skills and also to investigate the relationship between these skills and programming skills with the aim of enabling our students to develop a RP approach to learning to help them become proficient programmers. Our main concern was the willingness of students to engage with the reflective tasks and their perception of their relevance in improving their programming skills. While there is a long tradition of reflective writing in the humanities, it was considered that its use in scientific disciplines, including computing, might not be widely accepted by students (or staff) or even recognised as a 'valid' learning/teaching activity.

Rationale

In order to become proficient in a new programming language, students are expected to grasp a number of complex and abstract concepts and need to learn a new way of thinking (Jenkins, 2002). Computer programming demands a great deal in terms of cognitive skills. This includes critical and creative thinking and problem-solving skills. Learning to program also requires students to continuously plan and monitor their own progress, and to evaluate their efforts (Breed, 2004).

Reflection is the process of an individual examining his/her actions during or after the execution of those actions. While this process is not new, the practice of reflection has increased since the publication of Schön (1983), who asserts that effective RP guides professionals to examine and re-think their professional creations or course of action, **during** *(reflection-in-action)* and **after** *(reflection-on-action*) the accomplishment of the creation/action process. Schön believes that reflection improves ability and performance within a profession.

The importance of reflection in the context of software development derives from the complexity involved in developing software systems. Analysis of the field of Software Engineering supports the adoption of the RP perspective (Hazzan, 2004), where it is suggested that the developer must improve the understanding of his/her own mental processes by adopting a reflective mode of thinking.

Russell (2005) challenges educators who promote RP but do not provide explicit instructions or strategies for helping novice professionals. He believes that nurturing RP requires more than just telling learners to reflect. Agreeing with Russell, we ensured that the RPLO demonstrates the different levels involved in reflective writing. Students are thus able to test their understanding of each level.

Methodology

The results presented below are based on data collected over five months from c.80 undergraduate students on this module in 2008-09. Early in the semester, before the RPLO was introduced in the fifth week, students' reflective writing ability was measured by a textual analysis approach, using data from a tutorial reflective task related to a programming exercise. This provided the baseline.

Students' writing from assignments submitted in weeks seven and twelve were also analysed and rated after the exposure to the RPLO. Further data were collected via an online questionnaire (33 students) and short structured interviews (17 students). In addition, the use of the RPLO by students was tracked through the *WebLearn* virtual learning environment and reports were generated to obtain session data and identify learning patterns.

An inter-rater comparison tool based on Moon's categories of reflection (Moon, 2001) was used for textual analysis. To simplify the process, and make it more comprehensible to the students, Moon's four levels of reflection - from the lowest level of "descriptive writing" to the highest level of "critical reflection" - were renamed as 'Description', 'Analysis', 'Evaluation' and 'Conclusion'. Students' writing was independently rated by the authors.

Findings

Tracking

One concern, based on the literature and the authors' prior experience, raised at the beginning of this project was that computing students may be dismissive of the concept of reflection. However, as can be seen from the tracking results presented in Table 1, students engaged with the RPLO and accessed it approximately 250 times for a total accumulated time of 17 hours, 1 minute and 16 seconds. In fact, the RPLO came 4th ('Home Page' aside) of the learning-related items, in terms of total time and number of visits.

Table 1. Tracking data for top six (of 175) items held in WebLearn

Item	Visits	Average Time per Visit	Total Time	% Total Visits
Total	12036	03:50:33	489:48:29	100.00%
Home Page	5359	00:00:27	41:15:05	8.42%
Learning Materials URL	2611	00:05:33	242:11:44	49.45%
Coursework 1	1640	00:03:21	91:52:33	18.76%
Coursework 2	1004	00:03:34	59:51:12	12.22%
Reflective Practice LO	251	00:04:04	17:01:16	3.48%
Tutor contact list	138	00:03:38	08:22:46	1.71%

Questionnaire and Structured Interviews

Analysis of questionnaire results also indicates that students responded positively to RP, the RPLO and how these relate to their work and skills. In response to the question "The reflective practice learning aid helped me to think/write reflectively", 85% of the students either agreed or strongly agreed. In response to the question "I think that reflective practice helps my programming skills", 67% of students either agreed or strongly agreed.

Examples of positive responses were:

"It makes you look at your own work in a critical way. It also makes you think about the reasons you are programming in the way that you are";

"being able to plan each stage of compiling my work";

"it helped me to understand the way i had gone wrong";

"aided my thinking and problem solving style and technique".

Examples of negative responses were:
"It doesn't do much for me - sorry to say that";

"it didn't really helped me with my programming skills";

"I don't think it does".

Data from the short structured interviews suggest that, in general, students felt that reflective practice/writing helps them with their work:

"you understand the work more if you reflect through it";

"reflect on errors, try to fix";

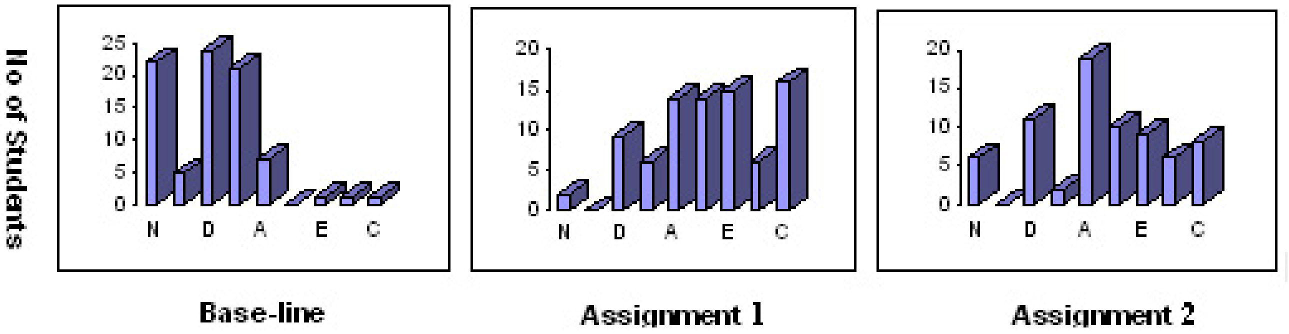
"It gave me more ideas, what you need to think about to be a programmer, how to manage myself to be a programmer, not to ask for help".

Most students seemed to greatly appreciate the clarity of the RPLO, in particular the demonstration of the different levels of reflection.

Textual Analysis

The results from the textual analysis (*Figure 1*) show a dramatic increase in the average level of students' reflection in the assignments, after the introduction of RPLO. Although the gap in the mean level of reflection is significant, we need to be careful with our conclusions, as the baseline task was not assessed: so students could be *less* motivated.

Figure 1 – Students' level of reflection resulting from the textual analysis



N = none (writing illegible or does not exist), D=description, A = analysis, E = evaluation and C = conclusion

Student Learning and Achievements

Results from the module in the academic year 2008-09 show a significant improvement in students' overall module marks in comparison with previous year. However, as the form of assessment on this module has changed in a number of ways, it is difficult to assess which element of the 'blend' has had the most impact.

Conclusion

It is difficult to claim for certain which element of the 'blend' has been most effective, but adjusting the blended learning environment appeared to work successfully. There is evidence that students improved their skills in both programming and reflective writing and in their engagement with the assessment. Furthermore, students' feedback on the RPLO suggests that it was used and appreciated.

We believe that the design and use of the RPLO has contributed to the success of the project. However, the learning object embeds examples and exercises of reflective practice within the context of Java programming and as such is not easily reusable in other contexts. In future work we aim to investigate techniques to create reflective practice learning objects that can be easily re-purposed to suit different subject areas.

Acknowledgement
This project is supported and funded by the "Write Now" Centre for Excellence in Teaching & Learning (CETL), www.writenow.ac.uk.

References

Beale R. (2007), 'Blogs, reflective practice and student-centred learning', In: *Proceedings of the 21st BCS/HCI Group Conference*, 3-7 September 2007, Lancaster University, UK. http://www.bcs.org/upload/pdf/ewic_hc07_sppaper1.pdf [accessed 24/04/10]

Boyle, T., Bradley, C., Chalk, P., Fisher, K. and Pickard, P. (2005), 'Introducing a virtual learning environment and learning objects into higher education courses', *International Journal of Learning Technology*, **Volume 1 Number 4,** pp 383-398

Breed, B. (2004), 'The reflective abilities of expert and novice learners in computer programming', *SA-eDUC JOURNAL* **Volume 1, Number 1**, pp. 6–14 http://www.puk.ac.za/opencms/export/PUK/html/fakulteite/opvoed/educ/artikels/Gekeurde/Breed.pdf [accessed 24/04/10]

Chalk, P. and Hardbattle, D. (2007),' Does Reflective Writing in the PDP Improve Science and Engineering Students' Learning?', *Investigations in university teaching & learning*, **Volume 4 Number 2** , http://www.londonmet.ac.uk/londonmet/library/o64237_3.pdf [accessed 24/04/10]

Hazzan, O. (2004), 'The reflective practitioner perspective in software engineering', position paper presented at *CHI 2004 One Day Workshop - Designing for Reflective Practitioners,* http://www.ics.uci.edu/~redmiles/chiworkshop/papers/Hazzan.pdf [accessed 24/04/10]

Jenkins, T. (2002),'On the Difficulty of Learning to Program', http://www.psy.gla.ac.uk/~steve/localed/jenkins.html, [accessed 25/04/10]

Moon, J. (2001), 'Reflection in Higher Education Learning', PDP Working Paper 4, LTSN Generic Centre, available at: http://www.heacademy.ac.uk/resources/detail/resource_database/id72_Reflection_in_Higher_Education_Learning [accessed on 24.6.2010]

Schön, D. A. (1983), *The Reflective Practitioner*, New York, Basic Books

Russell, T (2005). 'Can reflective practice be taught?', *Reflective Practice*, **Volume 6 Number 2**, pp199-204

Biographical note:

Dafna Hardbattle is a Senior Lecturer in the Faculty of Computing at LondonMet, currently leading a module on educational applications of digital technologies.
Email: d.hardbattle@londonmet.ac.uk

Dr Ken Fisher is a Senior Lecturer in the Faculty of Computing at LondonMet, currently leading a module on introductory programming in Java. Email: k.fisher@londonmet.ac.uk

Peter Chalk is Associate Dean in the Faculty of Computing at LondonMet, teaching computing and conducting research into web-based leaning. Peter's publications on e-learning date back to 1985. Email: p.chalk@londonmet.ac.uk