**Investigations**
**in university teaching and learning** vol. 7 spring 2011 ISSN 1740-5106

# Developing a Generic Design Pattern for a Learning Object to Support Student Reflection

Dafna Hardbattle
and
Ken Fisher
Faculty of Computing
London Metropolitan University

**Keywords**: *learning object, reflective writing, pedagogical pattern, reflective practice,*

## Introduction

This paper reports on findings of a funded project to discover and implement a generic pattern for learning objects to support reflective writing. This research is built on a recent project (Hardbattle, Fisher, & Chalk, 2010), which investigated the relationship between reflective writing skills and programming skills and aimed to encourage deep learning and to support and improve reflective report writing in first-year computing students. The reflective practice learning object (RPLO), which was developed via that earlier project based on a constructivist perspective, has attracted positive responses from academics and developers. It worked successfully towards improving students' engagement and skills in both programming and reflective writing. However, it was not easily adaptable for reuse as it was written in *Flash* and in a specific context. In order to expand the interactive learning experience to benefit students on other courses, the aim of this subsequent project has been to investigate a generic pattern for reflective practice learning objects. It is expected that the new template will allow lecturers easily to re-purpose and create reflective practice learning objects to suit different topics or subject areas.

## Context

Traditionally, the approach to designing reusable learning objects was to separate the content from the context (Boyle, 2003). This approach, however, appeared to be insufficient for our purpose and did not achieve good adaptability and reuse. Research into existing tools that are designed for learning object reuse suggests that there are very few tools available. Evaluation of these tools, such as the open source tool RELOAD, also suggests that they are not particularly user-friendly, especially for lecturers with limited technical abilities. Other tools were found limited in their interactive end-result functionality and are mainly text based, such as the wiki-type editor MURLLO.

Boyle (2006) advocates that, in order to achieve greater re-usability, there is a need to understand the underlying structure of the learning object. When we develop a generative learning object, the surface form has to be separated from the content of the learning object so that the forms can be reusable rather than the content. However, capturing the surface forms is only the first stage. There is also a need to identify the conceptual structure and the underlying pedagogical pattern. These should be the real basis for reuse.

Patterns for pedagogy were originally suggested by Susan Lilly in 1996, who developed the idea of reusable pedagogical design patterns. This was followed by the Pedagogical Patterns Project which was initiated by Sharp *et al.* (2003), who set out to collect and disseminate successful practices of educators. The notion of pedagogical patterns stemmed from the concept of design patterns in the field of object-oriented software design, which had a great impact on software development. Lilly said that pedagogical patterns should be repeatable and easy to adapt for different teaching sessions and by different tutors. They are aimed at capturing the best practice of teaching and learning in a specific domain and are designed to solve problems that normally recur in different contexts.

## Developing a reflective practice pedagogical pattern

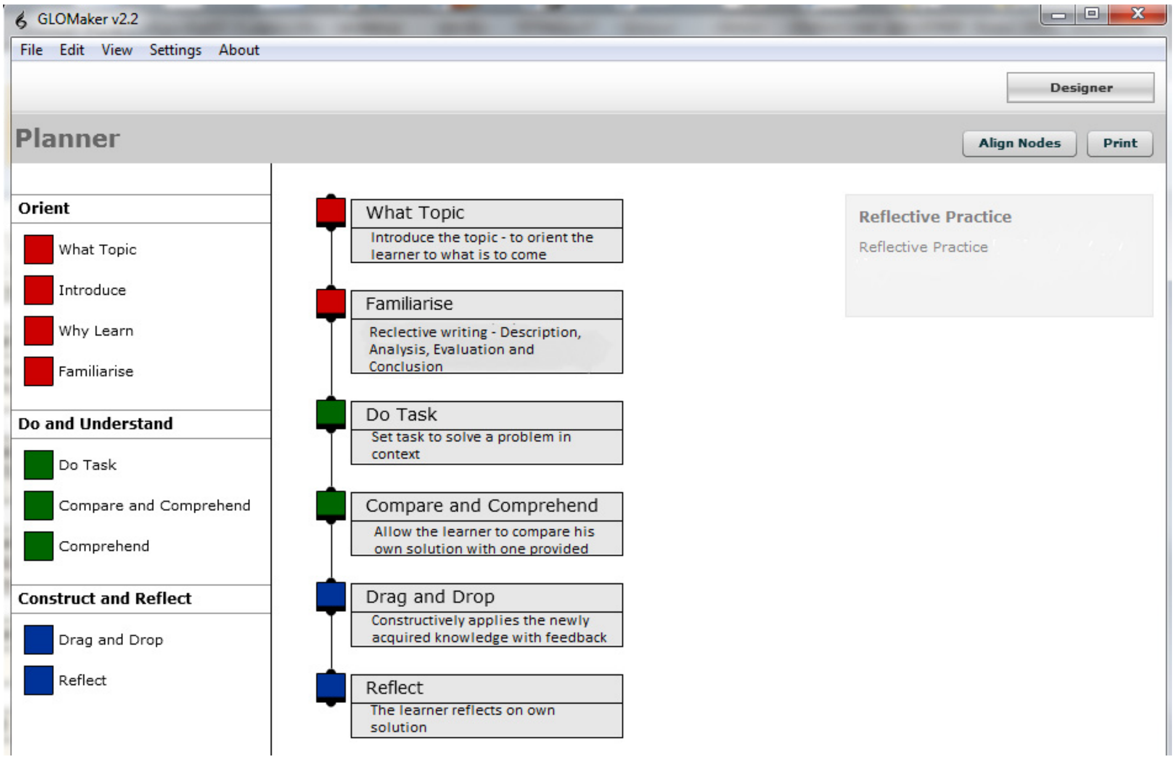
For our study, we adopted the generative learning object (GLO) approach, which is a template-based approach, and we focused on extracting a reflective practice pedagogical pattern (RPPP) in order to make this pattern the basis for reuse. We used the Generative Learning Objects Maker (GLO-Maker) tool that was developed by the Centre for Excellence in Teaching and Learning in Reusable Learning Objects (RLO-CETL). The GLO-Maker currently has two pedagogical patterns built into it. The first pattern is Explain and Show Amplified (EASA). It was originally developed to help learners to engage in skills of an abstract nature such as programming or mathematics. The second pattern is Evaluating Multiple Interpretations (eMI). This pattern was designed to encourage critical thinking and deeper understanding when learners have access to different perspectives. The tool also includes a Freestyle template.

### *Method*

The method of eliciting the RPPP involved two processes. The first process was the evaluation of the RPLO content and structure by academics, and the second was the use of the GLO approach to identify the hierarchical structure of the learning object.

Lecturers who were interested in using the original reflective practice learning object in their own teaching contexts, evaluated the learning object in terms of its surface forms and its content. They made decisions about what content/text they wanted to keep as generic text and what kinds of problems they wanted to present to learners. An in-depth discussion took place with each academic to understand their needs and views. The information was collated, analysed and integrated into a generic content/structure.

The second process, involving the application of an analytic approach to the RPLO itself, was undertaken by the researchers. The underlying hierarchical structure of the object was clarified and a reflective practice pedagogical pattern was elicited. The pattern is shown in Figure 1. At the very simplest level, the RPPP comprises three elements: “Orient[ate]”, “Do and Understand”, and “Construct and Reflect”. This basic structure was expanded further to include specific pedagogical functions such as “Familiarise”, “Compare and Comprehend” and “Reflect”.



What Topic
Introduce the topic - to orient the learner to what is to come

Familiarise
Reclective writing - Description, Analysis, Evaluation and Conclusion

Do Task
Set task to solve a problem in context

Compare and Comprehend
Allow the learner to compare his own solution with one provided

Drag and Drop
Constructively applies the newly acquired knowledge with feedback

Reflect
The learner reflects on own solution

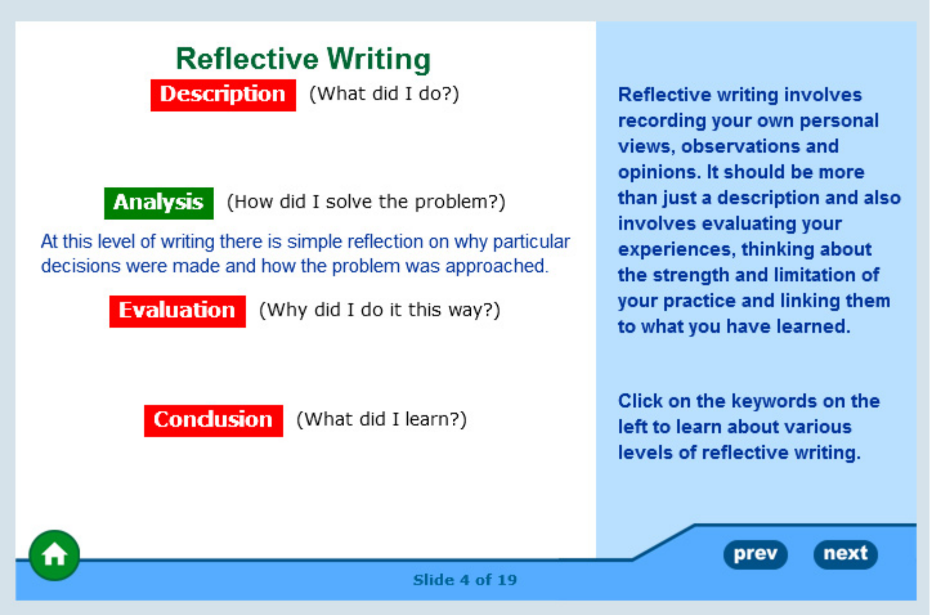
*Figure 1 – An Example of a Pedagogical Sequence, Based on the RPPP*

While creating generic content and structure in the context of programming or other computing topics was reasonably simple, identifying a generic structure to accommodate other contexts was more complex. Comparing the new pedagogical pattern, RPPP, to the existing patterns in the GLO-Maker tool shows similarities to both the EASA and the eMI pedagogical patterns. This is not surprising. The EASA pattern was originally extracted from learning objects designed in a programming context and comprises the Orient[ate] – Understand – Use structure. The eMI focuses on evaluation of different perspectives in an Arts and Humanities context, and comprises the Orient[ate] – Do Task – Reflect structure. The RPPP embraces problem solving in an abstract topic, construction of new knowledge and the reflection on what was learned, and therefore combines elements from both the EASA and the eMI patterns.

## Extending the functionality of the GLO-Maker tool

The GLO-Maker tool's architecture combines three components: an authoring tool, or interface, for tutors to create learning objects; an XML file, which is used to capture the output from the authoring tool; and a player, which presents the actual learning object. The rationale behind the tool is to allow educators to use it to develop highly adaptable multimedia learning objects (Boyle, 2006). When the content of a learning object is complete, the user can generate a multimedia file in the form of a *Shockwave Flash* object, which can then be displayed online. The tool is also extendible and allows new components (plug-ins) to be added as necessary.

Our evaluation of the GLO-Maker's authoring tool components suggested that, in order to provide the interactivity achieved by our original RPLO, the tool functionality had to be extended. Three new plug-ins were identified as being necessary. The first was a *Drag-and-Drop* functionality, which was needed when the learner is tested on the four levels of reflection. This activity provides the learner with scaffolding/feedback. The second was an *Input-Output* functionality, which was needed to allow learners to actively use the learning object by inputting their own text, for example when the learners are requested to input their own solutions. The text is then kept in the memory of the learning object until it needs to be extracted and presented again. This allows the learners to compare their own solutions with the solution provided, or to reflect on their own writing. The third functionality, *On-Click-Text,* was needed to allow the learner to interact with the learning object by clicking on text to reveal further information. Figure 2 shows the use of the *On-Click-Text* plug-in where the learner is exploring the four levels of reflective writing.



Reflective Writing

Description (What did I do?)

Analysis (How did I solve the problem?)

At this level of writing there is simple reflection on why particular decisions were made and how the problem was approached.

Evaluation (Why did I do it this way?)

Conclusion (What did I learn?)

Reflective writing involves recording your own personal views, observations and opinions. It should be more than just a description and also involves evaluating your experiences, thinking about the strength and limitation of your practice and linking them to what you have learned.

Click on the keywords on the left to learn about various levels of reflective writing.

*Figure 2 – An Example of the Use of On-Click-Text to Explore the Four Levels of Reflection*

## Assessment of impact

Some of the GLO-Maker tool development, for example embedding the new pedagogical pattern and developing the *On-Click-Text* plug-in, was straightforward. However, for the rest of the functionalities, the tool needed to be upgraded more than once by the RLO-CETL team to repair previously undetected faults. Because of the delay, the intention for lecturers to use and evaluate the pedagogical sequence in the GLO-Maker themselves and to create their own learning objects did not take place. Further complications arose from a lack of the necessary administrator rights to install the new version of the tool on the University computers. Therefore, only the project team had the opportunity to use and evaluate the tool.

Nevertheless, the development of the tool and the embedding of the pedagogical pattern were eventually completed successfully. Using the upgraded tool, a new reflective practice learning object was generated for students in the context of a different module. We tracked the use of the learning object through WebLearn, and questionnaires were given to students for feedback. Analysis of the questionnaires revealed results very similar to our previous research. The majority of students either agreed or strongly agreed to each of the closed questions presented online through WebLearn. Seventy-three per cent of students said that they found the reflective practice learning object was either helpful or very helpful for writing the coursework report that required reflection. Seventy three percent of students (89% of those who accessed it) either agreed or strongly agreed that the RPLO helped them to think/write reflectively.

## Conclusion and Future work

A Reflective Practice Pedagogical Pattern was elicited and embedded into a tool for creating learning objects, to enable a successful reflective practice learning object to be easily re-purposed for different topics or subject areas. Three new plug-ins were also successfully embedded into the GLO-Maker tool and we are hoping that a new version of the tool will be released shortly for public use.

Future work will involve working with lecturers on other modules, in the Faculty of Computing and University wide, who have expressed interest in re-purposing the reflective practice learning object for their own teaching.

## Acknowledgements

This project was supported and funded by the *Write Now* Centre for Excellence in Teaching and Learning, *Learn Higher* Centre for Excellence in Teaching and Learning and the Centre for Academic Professional Development (CAPD). We would also like to thank Matteo Corazzin, who helped in the development of the plug-ins for the GLO-Maker, and the module tutors who participated in this research.

## References

Boyle, T. (2003). Design principles for authoring dynamic, reusable learning objects. *Australian Journal of Educational Technology*, 19(1), 46-58. Available at: http://www.ascilite.org.au/ajet/ajet19/boyle.html

Boyle, T. (2006). The Design and Development of Second Generation Learning Objects. In E. Pearson & P. Bohman (Eds.), *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications* 2006 (pp. 2-12). Chesapeake, VA: AACE.

Hardbattle, D., Fisher, K. and Chalk, P. (2010). Reflective Practice for Programming Students: adjusting the blend to improve skills, *Investigations in university teaching & learning*, Volume 5 Number 1, pp. 75-80.

MURLLO - Management, Use and Re-purposing of Language Learning Objects. Available at: http://www.elanguages.ac.uk/researchcommunity/projects/murllo/tools.html

Pedagogical Patterns Project (2006). Available at: http://www.pedagogicalpatterns.org/

RELOAD Content Packaging & Metadata Editor Available at: http://www.bolton.ac.uk/IEC/Educational%20Software/RELOAD%20Content%20Packagin.aspx

RLO-CETL (2009). Generative Learning Objects (GLOs) Available at: http://www.glomaker.org/about.html#v1

Sharp, H., Manns, M.L., & Eckstein, J. (2003). Evolving pedagogical patterns: The work of the Pedagogical Patterns Project. *Journal of Computer Science Education*, 13(4), 315 – 330.

## Biographical note:

**Dafna Hardbattle** is a Senior Lecturer in the Faculty of Computing at London Metropolitan University, who teaches computing and led 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 London Metropolitan University who leads a module on introductory programming in Java. **Email:** k.fisher@londonmet.ac.uk