Customer-Driven Development for Rapid Production of Assessment Learning Objects

Andrew Adams and Shirley Williams
School of Systems Engineering, University of Reading, UK
a.a.adams@reading.ac.uk
shirley.williams@reading.ac.uk

Abstract: Customer-Driven Development is a technique from the software development method called extreme Programming (XP) where customers (most importantly including end users of all levels) are closely involved in the software design and redesign process. This method of producing software suitable for customers has been adapted to help in the production of e-learning material, in particular, Assessment Learning Objects (ALOs) consisting of multiple-choice questions. Asking undergraduate students to produce multiple-choice questions as part of their formal assessment processes facilitated this.

The outcome shows two distinct benefits to this process. Firstly, the students who took part in this project benefited from the encouragement to participate in reflective learning, both on the specific topic on which they chose to produce a multiple choice question, and in the methods and purposes of multiple choice questions (which form a significant part of their self-assessment regime and summative assessment exam). Secondly, of the questions produced by students a significant number of them were of suitable quality to be used for future cohorts and to be made available to the wider community. This gives two important benefits to staff: developing a wide range of questions is difficult and time consuming; student insight into misunderstandings of material can often be greater than that of staff. Resources for the development of ALOs are scarce and given that students benefit directly from being asked to develop their own questions, the year-on-year expansion of a question set produced by students can be a very useful resource.

Keywords: Learning Objects, Multiple Choice questions, Extreme Programming, Computer Aided Assessment

1. Introduction

Learning Objects (LOs Wiley 2000a) is a design concept for creating e-learning that can be reused in multiple combinations. Where an LO is designed for assessment it can be called an Assessment Learning Object (ALO). This approach to online learning material is based on the programming concept of object-orientation. Given that authoring high quality (A)LOs requires significant effort, it is reasonable to consider methods for their production also drawn from programming methods. Such methods obviously cannot be directly applied to the production of LOs, but the reasoning behind the approach must be considered and re-applied to the new task at hand. Earlier work (Adams et al, 2004) presented the application of the Pair Programming method of software development to the task of developing ALOs. Pair Programming is one of a suite of “agile” programming methods collectively known as “eXtreme Programming” (XP, Beck, 2000). Given the successful evaluation of Pair Programming, it is an obvious step to consider other elements of XP for ALO development. In this paper, the Customer-Driven Development software development method is adapted for ALO production.

In “traditional” software engineering (Sommerville 2004) input is sought from the “customer” for a software project only at a small number of explicit points in the development cycle. This corresponds to “traditional” teaching methods, which involve the students in curriculum and teaching material development only in highly constrained ways. In contrast, the Customer-Driven Development approach involves end users in the development of a program at many stages. This approach has parallels in the Student-Centred Learning educational approach. The term "Student Centred Learning" stems from the works of Myers and Briggs, and Cornelius (2000) provides a detailed review of the different types of learner styles.

Taking these two inspirations the authors have trialled the practice of requesting students to create the basic content of ALOs as part of their current course. The ALOs produced by the students have been assessed for quality on a number of measures, and in addition the effect on the students producing the ALOs has been assessed. The results of the initial study case were very positive and further investigations are under way.

Students on a course in Functional Programming (using the language Caml Light) were set the task of producing a new multiple-choice question, with guidance in the shape of existing ALOs available for self-assessment (a set of ALOs produced from the study undertaken in (Adams et al, 2004)). This task was a small part of their assessed coursework carrying a weight of 1% of the overall module. This very low weighting was due primarily

Reference this paper as:
Adams A and Williams S (2006) “Customer-Driven Development for Rapid Production of Assessment Learning Objects” *The Electronic Journal of e-Learning* Volume 4 Issue 1, pp 1-6, available online at www.ejel.org

to the unknown nature of any educational benefits to students of undertaking such a task. The ALOs already available, and the ALOs students were requested to produce, are purely formative in nature: they are available to the students as part of their self-assessment of progress and understanding and to give pointers to associated course material covering material which the students have demonstrated they do not understand.

The outcome shows two distinct benefits to this process. Firstly, the students who took part in this project benefited from the encouragement to participate in reflective learning, both on the specific topic on which they chose to produce a multiple choice question, and in the methods and purposes of multiple choice questions (which form a significant part of their formative self-assessment regime and summative assessment exam). Secondly, of the questions produced by students a significant number of them were of suitable quality to be used for future cohorts and to be made available to the wider community. This gives two important benefits to staff: developing a wide range of questions is difficult and time consuming; student insight into misunderstandings of material can often be greater than that of staff who are generally much more expert in the general and specific topics being taught than their students. Resources for the development of ALOs are scarce and given that students benefit directly from being asked to develop their own questions, the year-on-year expansion of a question set produced by students can be a very useful resource.

2. Assessment learning objects

Learning Objects are a relatively new way of presenting learning content. They are digital entities designed to facilitate reuse. Instructional designers can build small (relative to the course) instructional components that can be reused in different contexts. The term Assessment Learning Object (ALO) is used to describe an object designed to provide a summative or formative assessment of the learning.

The use of the term "object" itself has its roots in Computer Science, where the term object is used to describe an entity consisting of data and related operations. The origins of the term are Kay's work in the late sixties (Kay, 1968 & 1969). Wiley (2000b) states: "Learning objects are elements of a new type of computer-based instruction grounded in the object-oriented paradigm of computer science."

ALOs are in a number of ways the closest analogue to programming objects in terms of development goals (and therefore appropriate development methods). ALOs are primarily designed to have a functional interface: they must be "executed" in some way in order to produce a "result", i.e. the assessment of the student's knowledge. The requirement for their correctness is very high (100% for many of their aspects and near 100% for the remainder). Whether used for summative or formative assessment, the utility of an ALO is completely destroyed by many of the possible errors.

ALOs have many types, but one of the most familiar is the multiple-choice question. These have been popular for many years as a way of assessing the basic knowledge of students at many levels (Bull and Dalziel, 2003). They are particularly useful for large classes since they present a mechanically marked process that, when well design, can nevertheless provide useful feedback on gaps in student knowledge. While the marking of multiple choice questions is a quick and mechanical process compared to many other forms of assessment (and can be done completely automatically for e-learning systems) the production of useful multiple choice questions for a specific topic is not such a quick process. While creating a question and the correct answer may be relatively easy for an author experienced in the topic to hand, creation of sensible distracter incorrect answers is much more difficult, particularly in focussing the distracters to identify single points of confusion, allowing for the automated production of feedback to students who make mistakes.

3. Customer-driven development and student-centred learning

XP is what is called an agile design method. Incorporating this approach in to instructional design facilitates the introduction of agility into the production of learning material.

Within XP the customer plays a vital role. To quote from Wells (1999) the customer should aim:

> *"Not only to help the development team, but to be a part of it as well."*

With traditional software development techniques the customer is usually kept external to the project team. The customer takes part in initial requirements capture (Sommerville, 2004) and then is not involved in any detail until acceptance testing. This sometimes leads to problems with the software that is produced matching the customer requirements but not actually being what the customer wanted. Customer-driven development incorporates the customer into the design process. In particular, individual end-users from the customer organisation are engaged in the development cycle during decision points ranging from functionality to interface choices.

Within an e-learning environment, the Customer community contains the students who are studying this e-learning material. Involvement of students in the production of ALOs for the material is therefore the equivalent of customer involvement in interface or functionality development.

Such an approach ties in with the ideas of student-centred learning, where students are encouraged to take overall responsibility for directing their own learning and understanding the purpose of the tasks and information they are presented with by educators. By giving students the task to develop an ALO for possible use by future students they also gain an understanding of the mechanism of assessment implemented by such ALOs. This should enhance their own learning and in particular they're use of formative ALOs and their performance with summative ALOs. The relationship to student-centred learning is not exact, however, as student-produced content is principally used for other students than those creating them.

4. Student suggestion of ALO content

So, the precise instantiation of the XP method customer-driven development is Student Suggestion of ALO Content. The primary intent of this is two-fold: to reduce the effort needed by indivuals or small teaching teams to produce a large set of ALOs; to improve the quality of ALOs by involving students in their development. The burden on students producing the ALOs should not detract from their education, and ideally should contribute to it.

- To demonstrate the utility of this method, it was necessary to experiment with one or more groups of students and to meet the following specific targets in the experiment:
- A simple explanation of the task should be produced, which the majority of the students submitting coursework understood and were able to make reasonable attempts to complete.
- Student feedback from the cohort performing the coursework should indicate they did not find the task a burden.
- The questions submitted by students should provide suitable material which can, or could be, translated into ALOs with less work by the educator than would otherwise be required to produce an equivalent set of ALOs.

5. Functional programming

Functional Programming, for those that are not aware of it, is a different approach to programming from the most familiar concept, imperative programming. As the name suggests, functional programming involves defining functions, which have one or more inputs and produce exactly one output. At the University of Reading, Functional Programming has been taught in the first year of a Computer Science course, in parallel with the better-known imperative programming. The module is taught using lectures, practical classes and non-assessed weekly practical sheets. Online support includes self-assessment quizzes and discussion boards. Summative coursework is set twice per year, the first weighted at 10% of the overall module and the second weighted at 20% of the overall module. The "student suggestion of ALO content" task was set as part of the second assignment, so students had completed classes covering all the major elements of the language by this point and were being taught integrated example projects, which combined elements in various ways. Final assessment of the module was by an exam, half of which was multiple-choice questions of similar style to the online quizzes, although drawn from a separate pool.

6. Experience of using customer-driven development for the development of ALOs

6.1 The coursework

As part of the coursework for the functional programming module, the class of 2004/5 were set the following assignment:

Select one of the topics:

- Values and Expressions
- Records, Constructed and Enumerated Types
- Lists
- Simple Functions
- Pattern Matching Functions
- Recursive Functions

Create one multiple-choice question suitable for helping students to learn Programming in Caml Light. The multiple-choice question should have three incorrect answers and a correct answer. Incorrect answers should be "distracters", that is plausible wrong answers which demonstrate a specific misunderstanding of an element of the language. Each incorrect answer should have a one or two sentence explanation of what a student who chose incorrectly needs to learn. You may wish to view the online quizzes available on Blackboard as inspiration.

The question represented only 5% of the coursework mark. This represented a maximum of 1% of the marks for the module. Nevertheless, 44 students, of the 52 who submitted the coursework, attempted this assignment. The allocation of marks is as shown below:

Table 1: Shows the allocations of marks

Mark	#Students
0	0
1	4
2	3
3	15
4	19
5	3

Only a few students produced ideal questions (those three marked at 5 out of 5), i.e. those that could be copied into a learning environment and made available with only correctness, spelling and grammar checking applied. Many, however (those thirty four marked at 3 or 4 out of 5), produced questions, which were very usable as source material for question production. The commonest failing for these was the lack of explanations for the distracter answers, or a poor presentation of such explanation. The upgrading of these questions for use required less than half the time needed to produce such questions from scratch. In addition, the production of questions on the same topic by multiple people removes the boredom factor for an educator in coming up with "yet another variant on pattern matching for lists" or other commonly misunderstood topics. Only seven of the forty-two questions submitted (those receiving 1 or 2 out of 5) were not usable as ALO source material.

6.2 Sample work

The best questions produced could be used immediately, for example one student submitted the following:

Which of the following will assign a list containing 1, 2, 3 and 4 to the variable x?

1) Let x = [1::2::3::4];;

2) Let x = (1,2,3,4);;

3) Let x = 1::2::[3;4];;

4) Let x = 1,2,3,4;;

1) Incorrect. In order to use the "::" constructor the list must be bound to an empty list at the end. Also, the square brackets around the list will make this a list of a list if it is corrected.

*2) Incorrect. This is a type int*int*int*int, not an int list.*

3) Correct, the "::" constructor can be used to add numbers to the front of an existing list.

*4) Incorrect, this is a type int*int*int*int, not an int list.*

This piece of work could be directly transformed into an ALO that could be immediately deployed. The answers are all realistic within the context of programming with CAML light and the explanations of why an answer is a distracter are correct and appropriate.

Other good questions could not be used directly because of the way the student provided feedback. For example in this question:

Which of these evaluates to [1;2;3;4;2]?

a) conc([1;2;3], [4;2]);;

b) [1;2;3] @ [4;2];;

c) [1;2;3]+[4;2];;

d) [4;2]::[1;2;3];;

Answer a indicates that the student is probably guessing, assuming the non-existent conc function is used to concatenate lists.

Answer c probably means the student is confusing list concatenation in CamlLight with string concatenation in Delphi.

Answer d shows that the student does not fully understand the construct:: as this can only be used to add an integer value to a list of integers.

The correct answer is b.

The student provided an explanation that was aimed at a marker, not at the person attempting the question. To convert this question into an ALO it would be necessary to rework the explanations into a format that could be used immediately as feedback. However, the question itself is useful and the distracters are suitably chosen.

Poor questions came from students who had a lack of understanding of the technical material and so were unable to devise questions or answers. These students needed additional coaching so that they could master the subject.

7. Future work and conclusions

The module on which this project was initiated was a specialised module that is only taught in a small number of institutions. The results were promising, some of the material produced was immediately usable and much more provided inspiration to aid the design of new material. Three goals were set above, each of which was achieved:

- A simple explanation of the task should be produced, which the majority of the students submitting coursework understood and were able to make reasonable attempts to complete.
- Thirty-seven of the forty four students submitting the coursework produced usable material and achieved good marks for their contribution.
- Student feedback from the cohort performing the coursework should indicate they did not find the task a burden.
- Student feedback indicated that the students found the task interesting and different. A number of them indicated that they had gained a significant amount of understanding of the purpose and structure of multiple-choice questions while performing the task. Far from being a burden they found that it was a useful educational exercise.
- The questions submitted by students should provide suitable material which can, or could be, translated into ALOs with less work by the educator than would otherwise be required to produce an equivalent set of ALOs.
- Thirty-seven questions provided such source material. Previous work (Adams et al, 2004) indicates that this would represent hours of work by educators working in a Pair Development method.

A second study is under way, using the same technique in an online imperative programming module for the C language (a language taught in a significant number of courses). In addition to the online course run over the summer term, this language will be used for teaching first year programming to over two hundred students starting this autumn. The production of suitable online self-assessment ALOs will be a necessary task in preparing for this, and it is expected that student suggested ALO content will provide excellent support for this task.

When the ALOs are deployed a more detailed evaluation of the work will be able to be undertaken. An approach will be used that applies data mining techniques to the log data collected on responses selected by the students (Lubega and Williams, 2004).

Acknowledgements

The 2004/5 classes of students taking CS1H2 – Functional Programming provided the example of customers described here. They originally used ALOs that were devised as part of earlier work. They then contributed to generating a larger pool of ALOs by attempting and assessment that required the production of an ALO.

Thanks to these students, their predecessors and our colleagues for their support of the work described here.

References

Adams, A. Lubega, J., Walmsley, S. and Williams, S. (2004) "The Effectiveness of Assessment Learning Objects Produced Using Pair Programming", Electronic Journal of e-Learning, Volume 2, Issue 2, ISSN 1479-4403, pp 247-256

Beck, K. (2000) *Extreme programming Explained: Embrace Change*; Addison-Wesley.

Bull, J. and Dalziel, J. (2003) "Assessing Question Banks", in "Reusing Online Resources" edited Allison Littlejohn, Korgan Page, pp 171-181

Cornelius, S. (2000) "Learning Online: Models and Styles" in Online Tutoring e-book, Editor Carol A. Higgison, http://otis.scotcit.ac.uk/onlinebook/otisT102.htm, (accessed July 2005)

Kay, A. (1968) *FLEX, a flexible extensible language,* M.S. thesis, Univ of Utah, May, 1968 (Univ. Microfilms).

Kay, A. (1969) *The Reactive Engine,* PhD. thesis, Univ of Utah, Sept., 1969 (Univ. Microfilms).

Lubega, J. and Williams, S. (2004) "Patterns of Use when interacting with a Managed Learning Environment", in Cantoni, L. and McLoughlin, C. (eds) Proceedings: EdMedia 2004 World Conference on Multimedia, Hypermedia and Telecommunication, Lugano, Switzerland, pp 5001-5005.

Sommerville, I. (2004) *Software engineering*, 7th Edition Addison Wesley.

Wells, D. (1999) "The Customer is always available", http://www.extremeprogramming.org/rules/customer.html, (accessed July 2005)

Wiley, D. A. (ed) (2000a) "The Instructional Use of Learning Objects": Online Version: http://www.reusability.org/read/

Wiley D.A. (2000b) *Learning Object Design and Sequencing Theory*. Doctor of Philosophy Dissertation, Brigham Young University,