A Domain-specific Modeling Approach to the Development of Online Peer Assessment

A Domain-specific Modeling Approach to the Development of Online Peer Assessment

Yongwu Miao and Rob Koper

Educational Technology Expertise Center,
Open University of The Netherlands,
Valkenburg 177, Heerlen, The Netherlands
Email: yongwu.miao@ou.nl, rob.koper@ou.nl

Abstract: Modelling a peer assessment using IMS LD and IMS QTI is difficult for average practitioners. In this paper, we apply domain-specific modelling technologies to develop a peer assessment modelling language, in which notations are directly chosen from the concepts and rules used to describe peer assessment. Thus, practitioners can easily understand such a high-level language and use it to specify online peer assessment. The paper also discuss some related issues to develop an authoring tool for modelling with the peer assessment modelling language and to map a peer assessment model represented in the peer assessment modelling language to a corresponding executable model represented in IMS LD and IMS QTI.

Keywords: Domain-specific modeling, peer assessment; IMS LD; IMS QTI.

1 Introduction

Peer assessment is an arrangement for peers to consider the level, value, worth, quality or successfulness of the products or outcomes of learning of others of similar status (Topping, et al. 2000). Researchers have generally agreed that peer assessment stimulates student motivation and encourages deeper learning and understanding (Topping, 1998, Gipps, 1999; Boud, Cohen, & Sampson, 2001). As e-learning is more and more popular, a number of software tools supporting online peer assessment have been developed such as Peers (Ngu, et al. 1995), Peer Grader (Gehringer, 2001), NetPeas (Liu, et al. 2001), SPARK (Freeman & McKenzie 2002), Espace (Volder, et al. 2007), Turnitin Peer Review (Turnitin), and so on.

Contrast to traditional software development approaches to peer assessment tools listed above, we adopted a process modelling approach to support online peer assessment (Miao & Koper, 2007). Concretely speaking, we use IMS Learning Design (LD, 2003), IMS Question and Test Interoperability (QTI, 2006), and assessment-specific services to model peer assessment processes. The resulting peer assessment process models with necessary resources can be played in any standard-compatible run-time environment. In comparison with typical software development approaches, our technical approach is more efficient and flexible (Miao & Koper 2007). In particular, our approach can support seamless integration of peer assessment with learning activities.

However, this technical approach has limitations. The required level of knowledge of LD and QTI and technical knowledge of process modelling for those authoring assessments is significant. To acquiring such technical knowledge is very difficult for average practitioners. In addition, if a peer assessment process is extremely complex, the modelling work will be very difficult even for technical experts (Miao & Koper 2007).

In this paper, we address the limitations of the standard-based approach by applying domain-specific modelling (DSM) technologies. As the first attempt in this direction, we develop a peer assessment modelling language for practitioners to model peer assessment processes. The remaining part of the paper is structured as follows: domain-specific modelling and peer assessment are briefly introduced in Section 2 and Section 3, respectively. Section 4 presents a peer assessment modelling language. Some issues are discussed in section 5. Finally, we present conclusions and point out the future work.

2 Domain-specific Modelling

Domain-Specific Modeling (DSM) or Domain-specific language (DSL) are more expressive and therefore tackle complexity better, making modeling easier and more convenient. More importantly, they allow automatic, full code generation, similar to the way today's compilers generate Assembler from a programming language like JAVA.

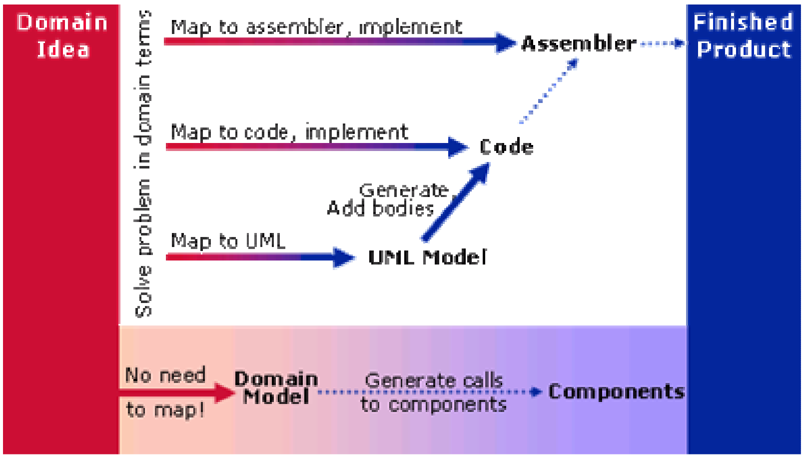

Figure 1: DSM and traditional software development approaches (from DSM forum)

DSM raises the level of abstraction beyond programming by specifying the solution in terms of concepts and associated rules culled from the very domain of the problem being solved. The final software products are generated from these high-level specifications (DSM forum). Notations in a domain-specific model are a whole level of abstraction higher than those in Unified Modelling Language (UML). As shown in figure 1, normally software developers will implement the final product by mapping the domain concepts to assembler, code, or UML model. By adopting DSM, a meta-model of the problem domain will be constructed as a modeling language by domain experts. Domain-specific code generators and executable components will be developed by experienced technical experts. Then, less experienced developers can use the meta-model to build actual solutions, which will be automatically transformed into existing component code.

DSM enables significant improvements in the productivity of the software development process and the quality of the resulting products. Industrial experiences of DSM consistently show it to be 5-10 times faster than current practices, including current UML based implementations of Model Drive Architecture (MDA). Since experienced technical experts specify the code generators and components, the resulting code is better than that most developers write by hand (DSM forum). Significant improvements can be achieved, mostly because the complexity is limited by focusing on a single, well defined problem domain (Tolvanen, 2004).

3 Peer Assessment

Peer assessment is a process consisting of various cognitive activities such as reviewing, summarizing, clarifying, providing feedback, diagnosing errors, and identifying missing knowledge or deviations (Van Lehn et al., 1995). In literatures many peer assessment process models are described (Liu et al., 2001; Sluijsmans, 2002; Sitthiworachart and Joy, 2003; Volder, et al, 2007). Typically a peer assessment process can be divided into three separate stages. In stage 1, candidates complete their assignments and then submit assignment outcomes. In stage 2, each reviewer assesses peer assignment outcomes and then gives feedback. In stage 3, each candidate reads and evaluates the received feedback and they may improve their original assignment outcomes based on peer feedback. Note that various forms of peer assessment are available in practice. For example, stage 2 and stage 3 may be repeated for several rounds until the final version of the assignment outcome is produced. The assignments and assessment form should be either pre-defined or designed before the stage 1 (called stage 0).

No.	Variable	Range of Variation
1	Curriculum area/subject	All
2	Objectives	Of staff and/or students? Time saving or cognitive/affective gains?
3	Focus	Quantitative/summative or qualitative/formative or both?
4	Product/output	Tests/marks/grades or writing or oral presentations or other skilled behaviors?
5	Relation to staff assessment	Substitutional or supplementary?
6	Official weight	Contributing to assessee final official grade or not?
7	Directionality	One-way, reciprocal, mutual?
8	Privacy	Anonymous/confidential/public?
9	Contact	Distance or face to face?
10	Year	Same or cross year of study?
11	Ability	Same or cross ability?
12	Constellation Assessors	Individuals or pairs or groups?
13	Constellation Assessed	Individuals or pairs or groups?
14	Place	In/out of class?
15	Time	Class time/free time/informally?
16	Requirement	Compulsory or voluntary for assessors/ees?
17	Reward	Course credit or other incentives or reinforcement for participation?

Table 1: A typology of peer assessment in higher education (Topping 1998)

In addition, the variables of the peer assessment could include levels of time on task, engagement, and practice, coupled with a greater sense of accountability and responsibility (Topping, et al. 2000). Topping (1998) developed a typology, as shown in Table 1, which consists of a survey of variables found in reported systems of peer assessment in higher education.

4 Peer Assessment Modelling Language

Definition of a peer assessment modelling language can start from choosing the terminologies used in the domain of peer assessment. Such terminologies provide natural concepts that describe peer assessment in ways that practitioners already understand. They do not think of solutions in coding terms. Starting from the existing vocabulary also means that there is no need to introduce a new, unfamiliar set of terms, or create a mapping between two sets of terms.

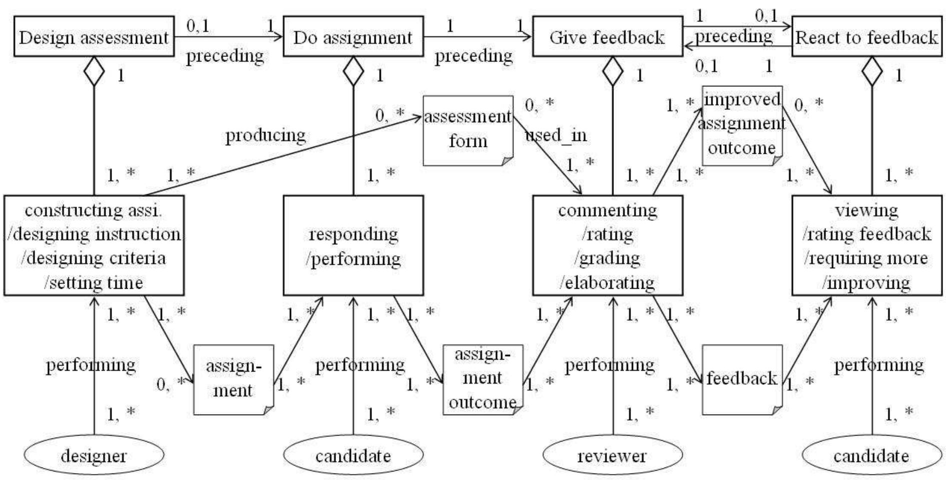

Figure 2: The meta-model of peer assessment

Based on the peer assessment process models and the typology described in the last section, we develop a meta-model by deriving many of the modeling concepts and the constraints. As shown in figure 2, a peer assessment process consists of four stages: *design assessment, do assignment, give feedback,* and *react to feedback.* In a *design assessment* stage, there is one or more various activities such as *constructing assignment, designing assignment/feedback instruction, designing assessment criteria,* and *setting time.* A *designer* can perform one or more activities and one activity can be done by one or more *designers*. One or more design activities may or may not produce *assignments/assessment forms*. Note that the *design assessment* stage may or may not be included in a peer assessment, because sometimes the *assignment* and the *assessment form* are pre-defined before a peer assessment starts. If the *design assessment* stage is included, it must precede a *do assignment* stage. In the *do assignment* stage, one or more *candidates* may be engaged one or more activities such as *responding question(naire)* or *performing tasks* according to the *assignment.* The *assignment outcomes* will be distributed to the activities in a succeeding *give feedback* stage, in which one or more *reviewers* will assess the allocated *assignment outcomes* according to the *feedback instruction* and *assessment criteria* by providing *feedback* in forms of *comments, rates, grades*, and so on. In certain *summative assessment*, the process may terminate here. Normally, a *react to feedback* stage will follow, in which the *candidate* will *view* the received *feedback*. Sometimes, a peer assessment process can be design in ways that *candidates* can *improve* their own *assignment outcomes* and even ask reviewers to *elaborate feedback* and/or to *review* the *improved assignment outcome*, In certain extreme situations, additional *react to feedback* stages and *give feedback stages* can be repeated for many rounds.

Such a meta-model can be used as a modeling language to specify various peer assessment scenarios. It is important to note that this diagram just illustrates the first-class concepts of the meta-model and primary relationships between them. Many details of the modeling language are represented as alternatives, constraints, and rules, which can not be illustrated in the diagram. When modeling a peer assessment scenario, one has to describe the scenario by representing the design decisions in the modeling language. For example, how many participants will be engaged and what roles they will play; which kinds of assignments (e.g., an essay or a list of multiple-choice questions) will be used and whether each candidate has a different assignment or the same one; whether each reviewer can review only one or more assignment outcomes of their peers; whether assignment outcomes will be distributed in a rotated, reciprocal, or mutual manner. In order to help practitioners to make design decisions, the modeling language defines default values for certain design variables. For example, all candidates are reviewers as well. In addition, certain design decisions are related in a way if one design decision has been made then the relevant decisions will be made accordingly. For example, if a summative assessment is selected as the purpose of a peer assessment, then the activity *improving assignment outcome* in *react to feedback* stage and the activity *elaborating feedback* in *give feedback* stage will be excluded accordingly. Thus, it is necessary to guide practitioners specifying a peer assessment by employing a sequence of decision-makings. All of these decisions could be easily captured and recorded by using the meta-model, and thus made available for subsequent use and refinement in the process of modeling. Because of the limitation of the space, these issues will be not discussed in detail in this paper.

5 Discussion

In order to support practitioners to develop online peer assessment using the peer assessment modeling language, the things below should be provided: an authoring tool for modeling with the peer assessment modeling language, a domain-specific component library, and a domain-specific code generator. This section discusses these issues.

An authoring tool: A tool should enable practitioners to specify a peer assessment, as mentioned above, by guiding practitioners to make a series of decisions. On the one hand, the tool makes it possible that practitioners don't need to specify every detail by employing default values and relevant decisions. On the other hand, the tool makes it possible that practitioners can specify any detail if they like.

A domain-specific component library: Because LD and QTI are executable code, LD components (e.g., activity and role) and QTI component (e.g., choice interaction and response-handling) can be regarded as basic components. More complicated components like certain templates represented as a fragment of LD/QTI code can be defined and stored in a library (Miao, Burgos, et al. 2007).

A domain-specific code generator: In our case, the code generator just generates LD and QTI code. Because the functions to generate LD code have implemented in LD authoring tools like RELOAD (RELOAD) and COSMOS (Miao, 2005), we just need to develop mapping functions to translate peer assessment modeling language into LD concepts. For example, a *stage* maps to the *act*, a *commenting* maps to an *activity*, and a *reviewer* maps to a *role*.

After these facilities are developed, practitioners should be able to develop and customize a peer assessment as a high-level model, which will be transformed into a corresponding LD+QTI model. The later model can be executed in any LD+QTI compliant run-time environment. It is important to note that DSM can be applied to support the development of other pedagogical models such as problem-based learning and 360 degree feedback.

6 Conclusions

In this paper, we outline an approach to apply the domain-specific modelling paradigm to the development of peer assessment. We developed a peer assessment modeling language and proposed to support the modeling process as a sequence of design decision-makings. Based on the peer assessment modeling language and the decision sequence, we will implement an authoring tool, associated component library and mapping functions in the near future.

References

Boud, D., Cohen, R. & Sampson, J. (2001). Peer learning and assessment, In: Boud, D., Cohen, R. and Sampson, J. (Eds.) Peer Learning in Higher Education (pp. 67-84). London: Kogan Page.

DSM forum: http://www.dsmforum.org/

Freeman, M. and J. McKenzie (2002). "Implementing and evaluating SPARK, a confidential web-based template for self and peer assessment of student teamwork: benefits of evaluating across different subjects." *British Journal of Educational Technology* 33(5): 553-572.

Gehringer, E. F. (2001). Electronic peer review and peer grading in computer-science courses. Proceedings of the 32nd SIGCSE Technical Symposium on Computer Science Education, Charlotte, North Carolina.

Gipps, C. (1999). "Socio-cultural perspective on assessment." Review of Research in Education 24: 355-392.

LD. (2003). "IMS Learning Design Specification." from http://www.imsglobal.org/learningdesign/index.html

Van Lehn, K. A., Chi, M. T. H., Baggett, W., & Murray, R. C. (1995). Progress report: Towards a theory of learning during tutoring. Pittsburgh, PA: Learning Research and Development Center, University of Pittsburgh.

Liu, Z., Lin, S., and Yuan, S., (2001) Experiencing NetPeas : Another way of learning, Lecture notes in computer science, vol. 2198, pp. 584-588.

Miao, Y. (2005) CoSMoS: Facilitating Learning Designers to Author Units of Learning Using IMS LD. Proc. 13th International Conference on Computers in Education, Singapore, 2005, 275-282, IOS Press.

Miao, Y. & Koper, R. (2007) An Efficient and Flexible Technical Approach to Develop and Deliver Online Peer Assessment. In proceedings of CSCL 2007 conference, p. 502-511, July 2007, available at http://dspace.ou.nl/handle/1820/853.

Miao, Y., Burgos, D., Griffiths, D. and Koper, R. (accepted) Representation of Coordination Mechanisms in IMS Learning Design to Support Group-based Learning, in Lockyer, L., Bennett, S., Agostinho, S. and Harper B. (Ed.) *Handbook of Research on Learning Design and Learning Objects: Issues, Applications and Technologies.* available at http://hdl.handle.net/1820/930.

Ngu, A.H.H., Shepherd, J., et al. (1995). "Engineering the 'Peers' system: the development of a computer-assisted approach to peer assessment." Research and Development in Higher Education 18: 582-587.

QTI (2006). "IMS Question and Test Interoperability Specification." from http://www.imsglobal.org/question/index.html.

RELOAD: http://www.reload.ac.uk

Sitthiworachart, J., and Joy, M., (2003) "Web-based Peer Assessment in Learning Computer Programming", The 3rd IEEE International Conference on Advanced Learning Technologies: ICALT03, 180-184, Athens, Greece, 9-11 July 2003.

Sluijsmans, D.M.A. (2002). Student involvement in assessment: The training of peer assessment skills. Unpublished doctoral dissertation, Open University of the Netherlands, Heerlen. http://www.ou.nl/eCache/DEF/18/194.html

Tolvanen, J.P. (2004). MetaEdit+: domain-specific modeling for full code generation demonstrated [GPCE]. In OOPSLA'04: Companion to the 19th annual ACM SIGPLAN conference on Object-oriented programming systems, languages, and applications, 39-40, 2004.

Topping, K. J. (1998). "Peer assessment between students in colleges and universities." *Review of Educational Research* 68: 249-276.

Topping, K. J., Smith, E. F., Swanson, & Elliot. (2000). "Formative peer assessment of academic writing between postgraduate students." *Assessment and Evaluation in Higher Education* 25(2): 150-169.

Turnitin: http://turnitin.com/static/peerreview.html

Volder, M.D., Marjo R. et al (2007). Espace: A New Web-Tool For Peer Assessment With In-Built Feedback Quality System, in Proceedings of ABR & TLC Conference, Hawaii, USA, 2007.