An Efficient and Flexible Technical Approach to Develop and Deliver Online Peer Assessment

An Efficient and Flexible Technical Approach to Develop and Deliver 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: Peer assessment is a special form of collaborative learning, in which peer students learn through assessing others' work. Recently, the design of collaboration scripts is a new focus area within the CSCL community. In this paper, we present a method based on open e-learning standards to script peer assessment processes. A standard-compatible tool can help users to script various forms of peer assessment in a machine-interpretable form. Such peer assessment scripts then can be executed on today's open technical e-learning infrastructure. In comparison with typical software development approaches to support online peer assessment, this technical approach is more efficient and flexible.

1. Introduction

Falchikov (2001) defines peer assessment as "the process whereby groups rate their peers". Somervell (1993) states that peer assessment engages students in making judgments on the other students' work. Researchers have generally agreed that peer assessment stimulates student motivation and encourages deeper learning and understanding (Freeman 1995; Topping 1998; Pope 2001). As Weaver and Cotrell (1986) pointed out, peer assessment can be seen as a means by which ability in the learner to make independent judgments of their own and others' work can be developed and practiced. A peer assessment can encourage a greater sense of involvement and responsibility, establish a clearer framework and promote excellence, direct attention to skills and learning and provide increased feedback. Peer assessment can be seen as a special type of collaborative learning (Freeman 1995; Brindley and Scoffield 1998; Keppell, Au et al. 2006). It not only promotes students' confidence in their ability to assess the work of others, but also provides the opportunity to develop skills for working in a team. In principle, no a single form of peer assessment can fit all situations. In practice, various forms of peer assessment are designed and used.

Although peer-assessment may be a comprehensive learning process in some ways, there are also some identified pitfalls (Falchikov 2002). Many of the associated problems may occur because it is a complex procedure and students are not very experienced to conduct peer assessment. The success of peer assessment depends greatly on how the process is set-up and subsequently managed. In recent years, many computer-based tools have been developed for supporting peer assessment. For examples, Many Using and Creative Hypermedia system (MUCH) (Rada, Acquah et al. 1993; Rushton, Ramsey et al. 1993), Peers (Ngu, Shepherd et al. 1995), Peer Grader (PG) (Gehringer 2001), and Self and Peer Assessment Resource Kit (SPARK) (Freeman and McKenzie 2002) are multi-user tools that support collaborative learning and have been successfully used to undertake peer assessment. These software tools are developed in a typical software development method. Normally, software developers make quite a lot efforts and invest much time to develop a peer assessment tool. In addition, after a tool is developed, it is difficult to change and add new functions to fit changing learning contexts and specific needs.

Recently in CSCL community, the design of collaborative learning scripts is a new focus area. The basic idea is to describe collaboration processes formally by using a scripting language and then to scaffold a group of students communicate and collaborate by executing collaboration scripts (O'Donnell and Dansereau 1992; Dillenbourg 2002; Kollar, Fischer et al. 2005; Miao, Hoeksema et al. 2005; Weinberger, Stegmann et al. 2005). However, so far there is no scripting language which is suitable to model various forms of peer assessment (see next section) and furthermore no corresponding system provides rum-time support. In this paper, we present an approach based on today's open e-learning standards to develop and deliver online peer assessment. In comparison with typical software development approaches to support peer assessment, we argue that our approach is more flexible and efficient. This paper is organized as following. First, we briefly introduce peer assessment and analyze the

characteristics of peer assessment from the perspective of collaboration scripts. Then we present an open e-learning standard based approach to support peer assessment. We present how users will be supported to script a peer assessment process by using an authoring tool and to execute a peer assessment script in today's open technical e-learning infrastructure. After discussing the advantages and disadvantages of this approach, we present conclusions and indicate the future work directions.

2. Various Forms of Peer Assessment

As mentioned above, there are various forms of peer assessment available. The variables could include levels of time on task, engagement, and practice, coupled with a greater sense of accountability and responsibility (Topping, Smith et al. 2000). To analyze the characteristics of peer assessment, we used Topping's aforementioned typology (Topping 1998), shown in Table 1. This typology consists of a survey of variables found in reported systems of peer assessment in higher education.

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

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?

In this section we investigate these variables from the perspective of scripting peer assessment. Some variables have no directly effect on scripting. They can be treated as certain kinds of metadata for describing and retrieving scripts. These variables are var. 1, var. 2, var. 6, var. 9, var. 10, var. 12, var. 13, and var. 14. Then we clustered the reminding variables into two categories: task-relevant variables and process-relevant variables.

2.1. Variety in Assessment Tasks

The variable concerning assessment tasks is variable 4. Various types of tasks may be performed in peer assessment for both providing evidences and for giving feedback. The usual task types, as described in variable 4, are tests/marks/grades or writing an essay. As reported in (Kane and Lawler III 1978), different types of tasks can be performed in peer assessment: peer ranking, which consists of having each group member rank all of the others from best to worst on one or more factors; peer nomination, which consists of having each member of the group nominate the member who is perceived to be the highest in the group on a particular characteristic or dimension of performance; and peer rating, which consists of having each group member rate each other group member on a given set of performance or personal characteristics, using any one of several kinds of rating scale. In knowledge

convergence script (Weinberger, Fischer et al. 2004), peer students use open-questions to write articles and to comment on peers' articles as well.

In addition, variable 4 mentions oral presentations or other skilled behaviors. That is, in an online peer assessment, task-specific application tools may be used to demonstrate their progress and capabilities and to evaluate peers' work. Pellegrino, Chudowsky et al (2001) described the use of concept mapping to assess knowledge structures, or the use of latent semantic analysis to interpret student essays. Therefore, scripting peer assessment requires explicitly modeling various types of tasks.

2.2. Variety in Assessment Processes

Peer assessment that are embedded in an institutional context, require more stipulation of the processes of assessment and rely on higher levels of student involvement (Sluijsmans, Brand-Gruwel et al. 2004). Var. 5 concerns whether staff is involved in the process and what a kind of role s/he will actually has. Variables concerning the composition of the feedback groups are var. 11, var. 12, and var. 13. Variables concerning the interaction of the students are var. 7 and var. 8. In peer assessment processes, various tasks are carried out by many students with multiple roles in sequence or in parallel. A large quantity of information is produced in performing various tasks in different phases. Students interact with each other through exchange of information. They may exchange in one-way, reciprocal, or mutual manner. In knowledge convergence script (Weinberger, Fischer et al. 2004), peer students transfer their articles and comments in a rotate manner. Variable 3 concerns whether a peer assessment is integrated with other learning activities. Peer assessment has a vital role to play in formative assessment, but it can also be used as a component in a summative assessment package. Therefore, in order to support online peer assessment, a complex workflow with the involvement of multiple users/roles should be modeled.

In summary, there are various forms of peer assessment. They vary in using different task types and in different interaction processes. Basic requirements to script peer assessment are to model various types of assessment tasks and various forms of group interaction.

3. An Approach Based on Open E-learning Standards

This section presents two open e-learning standards which are suitable to support various types of assessment tasks and various assessment processes, respectively. Our approach is based on these two international e-learning standards.

3.1. IMS Question and Test Interoperability

The IMS Question and Test Interoperability (IMS QTI 2006) is an open e-learning standard which describes a data model for the representation of question (assessment_item) and test (assessment_test) and their corresponding results reports. The diagram below the dash line in Figure 1 illustrates the main concepts and their relations. For the purpose of this paper, we omit a lot of detail of IMS QTI conceptual model. General speaking, an assessment_test consists of a set assessment_items. An assessment_item contains not only information about question itself, but also relevant information such as time_dependent, adaptive, stylesheet, modal_feedback, and some kinds of declarations. In Figure 1, only item_body (representing questions) and outcome_declaration (representing results like a score), response_declaration (capturing user's response), and response_processing (handling results according to user's responses) are drawn and emphasized. An item_body can have one or more interactions. IMS QTI defines a set of interaction types such as choice_interaction, text_entry_interaction, extended_text_interaction, match_interaction, order_interaction, slider_interaction, an so on. Each interaction is associated with a response variable which captures user's response. User's responses will be used to determine the outcome according response_rules (not drawn in Figure 1) specified in response_processing. So IMS QTI provides sufficient flexibility to grow into the advanced constructed-response items and interactive tasks we envisage as the future of assessment elaborates the assessment items in detail (Almond, Steinberg et al. 2001). Furthermore, it provides mechanisms to design structured assessment and control branches and calculate weighted scores. That is, various types assessment tasks and even structured assessment tasks needed in peer assessment can be supported by using IMS QTI tools.

However, IMS QTI is concerned with individual learners only, although it does not prohibit usage in contexts involving other actors (e.g., instructors, supervisors, and peers). It does not support explicitly the definition of a variety of roles or sequencing behaviors that result from participation of other actors. Therefore, it can not be

used to support the multiple roles/users interaction that are needed to model peer assessment. Additionally, IMS QTI does not support specific assessment tasks which need specific assessment tools.

3.2. IMS Learning Design

IMS Learning Design (IMSLD 2003) is an open e-learning standard based on the Educational Modeling language (EML) developed by Open University of the Netherlands (Koper 2001). The diagram of upper part in Figure 1 (excluding grey rectangles) illustrates the main concepts and their relations in IMS LD. It is a conceptual model represented by using UML notations. Some concepts (e.g., learning objective, activity-structure, and concrete expressions) and some relations (e.g., hierarchical structure of role or environment, association relation between act and notification) are not shown in Figure 1 for the sake of simplicity and readability. As illustrated in Figure 1, a learning design (unit of learning is its operational object with necessary resources) consists of a set of components such as roles (including learners and staff), activities (including learning activities and support activities), environments (containing learning objects and services), and properties (including personal, role-based local-/global-properties, not shown in Figure 1). They are organized by using theatrical metaphors like plays, acts, and role-parts as a hierarchically structured and process-oriented method. Conditions, as a part of the method, consist of expressions (e.g., logical expressions, arithmetic expressions, and IMD LD specific expressions not shown in Figure 1) and actions (e.g., show/hide, notification, and change-property). IMS LD is a pedagogical neutral language which can be used to model a wide range of pedagogical strategies (Koper and Olivier 2004). In general, IMS LD can be used to script different forms of group interaction involved with multiple roles/users.

Although EML can support assessment, however, assessment tools and strategies are excluded in IMS LD (IMSLD 2003) when it was adopted by IMS (considering the existence of IMS QTI). As a consequence, IMS LD can not explicitly model various types of assessment tasks within a peer assessment process. However, IMS LD supports to include assessment content. In addition, as illustrated in Figure 1, IMS LD offers an approach to integrating application tools as services. Although only four internal services are explicitly specified in IMS LD, in theory, any software tool can be integrated in a learning design as an external service. Therefore, with an appropriate interface, any specific assessment tool (e.g., a concept-mapping tool or a simulator) can be integrated into a unit of learning.

3.3. Supporting Peer Assessment through a Combined Use of IMS QTI and IMS LD

IMS QTI version 2 provides the possibility to integrate IMS QTI with IMS LD. The primary motivation for integrating IMS LD and IMS QTI stems from use cases involving formative assessment and summative assessment using items with traditional question types (IMSQTI 2006). We extend the application areas of an integration of IMS LD and IMS QTI and improve the benefit of their combined use. As a consequence, a peer assessment can be modeled as a unit of assessment, a special unit of learning with assessment-specific entities.

Figure 1 shows an extended IMS LD conceptual model with an integration of IMS QTI. The grey rectangles represent extended assessment-specific concepts. A unit of assessment contains, at minimum, one assessment activity performed by assessee or assessor in a manner exploiting IMS QTI documents or/and assessment-specific services. It is important to note that such an extension is at conceptual level, without changing IMS LD at operational level except to explicitly add a new resource type "imsqti". For example, an assessment activity should be defined still as a learning activity or a support activity. Assessee or assessor will be defined as sub-roles of staff or/and learner in a normal way. If an external service will be used as an assessment tool, it will be defined in a normal way to specify other external services. Only if a QTI item such as a multiple-choice, an ordering, or an open-question will be used in the assessment activity, the definition of the resource has to be handled in a IMS LD-aware manner. As illustrated in Figure 1, a resource referring to an assessment_test or an assessment_item has to be explicitly defined as an "imsqti" type. With such an indication, the run-time environment will call a QTI player as a generic assessment service to render questions according to the referred QTI document. In addition, any assessment-relevant property in IMS LD should be defined in a way that the identifier of the property is defined as a combination of the identifier of the assessment_item and the identifier of the outcome. In this way, a property and a outcome will be coupled. Three solid lines represent the connections between IMS LD and IMS QTI.

When scripting a peer assessment through such a combined use of IMS LD and IMS QTI, a peer assessment can be modeled and wrapped as a special unit of learning, which include a set of coordinated learning activities, support activities, and assessment activities performed by a group of peer students (and sometimes including tutor). An assessment activity may be performed by using a specific assessment service or by referring a

QTI document directly in its activity-description or indirectly through a learning object within an associated environment (see Figure 1). The scripted peer assessment then can be delivered in an integrated execution environment. The following two sections will present this approach in detail using a peer assessment example.

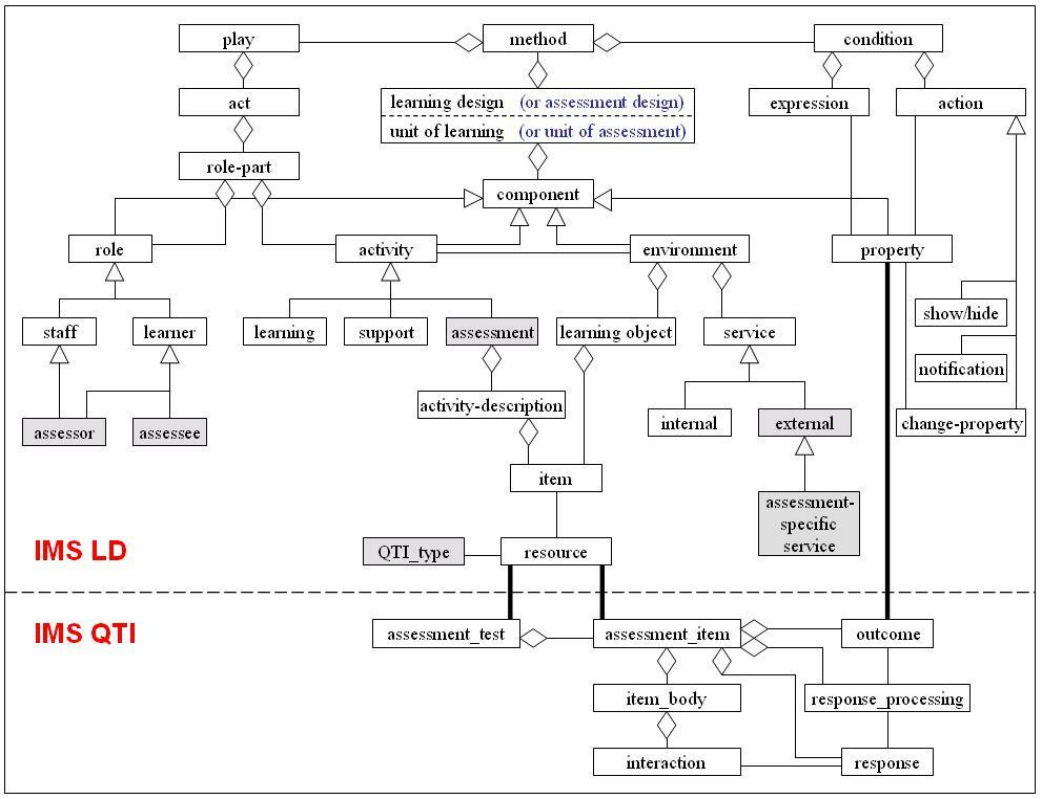

Figure 1. Extended IMS LD Conceptual Model with Integration of IMS QTI

4. Scripting a Peer Assessment

4.1. A Peer Assessment Example

For the purposes of presenting modeling method, a case study is introduced that is originally described in (Orsmond 2004). This case study describes a peer assessment exercise – writing and reviewing an article for a scientific magazine. The following steps describe the principal stages:

1. A tutor explains the peer assessment procedure and instructs students to select an interesting, recent paper from the primary scientific literature.
2. Each student selects a different paper and reads it.
3. Each student then prepares a brief article (400-500 words) about their chosen paper in the style of the "This Week" section of New Scientist magazine.
4. Pairs of students then exchange articles and review each other's work, using an evaluation sheet very similar in overall style to that used by scientific journals. The reviewer must assess the article and (i) decide whether the article is acceptable without change or whether minor/major revision is required (ii) provide specific feedback on any points raised by commenting on the article.
5. Student reviewers then return the article and evaluation sheet to the original author, who has then to consider their response to the review, using a response form. Students must decide whether to (i) modify their article, whether they feel that the reviewer's comments are appreciate and (ii) prepare a written response to each of the points raised by the reviewer. Then students hand in all documents for final assessment.
6. The tutor then marks on students' exercises in a way that the quality of the original version of the article, the student's response to peer review, and the student's effectiveness as a peer reviewer will be considered as 30%, 30%, 40% of the overall mark, respectively.

4.2. Scripting the Peer Assessment Example by Using an Authoring Tool

The peer assessment example is modeled and shown in Figure 2. In this peer assessment example there are two kinds of roles: tutor and learner. In order to explicitly model the tasks of each peer student and the exchange of information between them, learner1 and learner2 are defined as two sub-roles of the learner. The tutor and peer students are assigned to do different tasks. The tasks are modeled as learning activities (e.g., selecting/reading paper1 and responding review1) and support activities (e.g., final assessment1) in the model. Each activity has an element called activity-description, some of which (e.g., writing article1 or reviewing article2, final assessment2) refer to QTI documents. The overall assessment process is defined as a play with six acts illustrated in the Figure 2. Each act consists of more than one role-part. In the first act, the tutor teaches learners how to conduct this peer assessment and what is expected. In the second act, two peer students select a different paper respectively and read the selected papers. In the third act each student writes an article. In the fourth act students review the articles of their peers and comment on them. In the following act they response to the reviews of their peers and revise the original article if necessary. In the last act, the tutor assesses the students' work and give them scores. All acts are executed in sequence. The arrows with solid lines in Figure 3 indicate the control-flows of the process

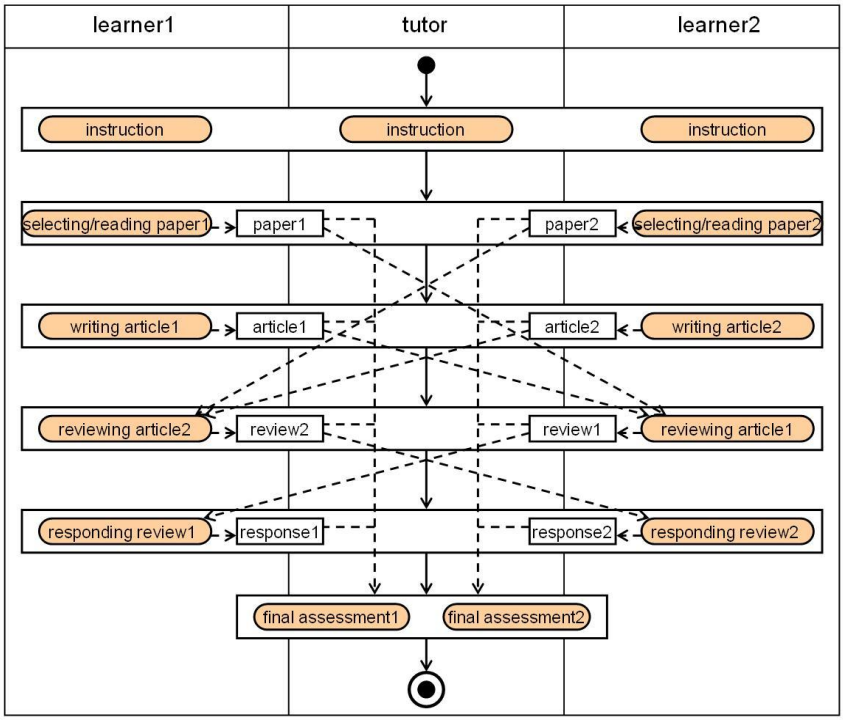

Figure 2. Process Model of a Peer Assessment Example

Properties should be defined to represent products and assessment results (e.g., article1 and review1) in the peer assessment script. Meanwhile, corresponding outcome variables of assessment_items have to be defined as well. The identifier of a property titled articile1 has to be defined in a way like article1_qtiitem.content by combining identifier of assessment_item (defined as article1_qtiitem) and identifier of outcome (defined as content). Such definition enables data transference from QTI document to IMS LD property. In addition, as we see in Figure 2, data (e.g., article1, article2, review1, review2, and so on) are produced by a learner in an activity and will be used by another learner in another activity. The arrows with dash lines indicate the data-flows in the process. Viewing the value of a property is realized by using "view-property" element in a XHTML document, which is modeled as a learning resource and will be referred by an item. The item is defined in a learning object within an environment. We define two environments for storing data regarding to the work of two learners, respectively. For example, the environment named "information about article1" will be associated with all activities handling article 1 such as selecting/reading paper1, writing article1, reviewing article1, responding review1, and final assessment1. Since all data concerning article 1 is collected in this environment, this shared environment can be used by learner1 writing article1, by learner2 reviewing article1, and by tutor assessing learner1's work.

An compatible authoring tool can be used to script this peer assessment and then to generate IMS LD code and IMS QTI documents automatically. This tool is developed based on CoSMoS (Miao 2005), a tree-form-based IMS LD authoring tool and now is extended to integrate functions for editing IMS QTI item. Although not all QTI edit functions have been developed, as shown in Figure 3, a user can script a learning design and edit necessary QTI documents in an integrated authoring environment with a unified user interface. The Figure 3 shows the user interface of editing the review form with a multiple-choice interaction and an open-question interaction. It is important to note that the coupling of a property (e.g., comment1) in peer assessment script with an outcome variable (e.g., comment) in the assessment_item titled "review1" can be defined by dragging the icon of the property and dropping into the input-filed of outcome. Then the identifier of the property titled "comment1" will be assigned as "review1.comment" automatically.

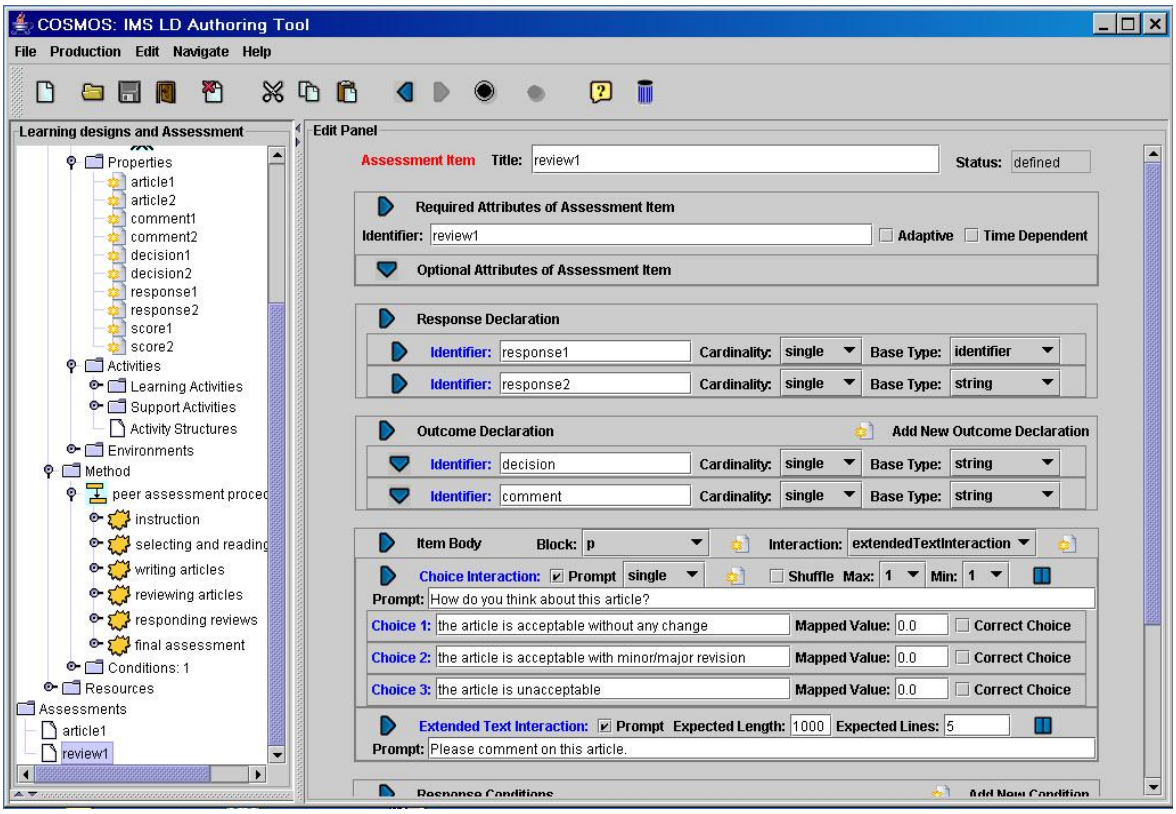

Figure 3. A Screenshot of an Integrated IMS LD and IMS QTI Authoring Tool

5. Delivering a Peer Assessment

This peer assessment example has been executed successfully in a web-based, integrated execution environment including Service-based Learning Design Player (SLeD 2004), an IMS LD client, CopperCore (Vogten and Martens 2004), an IMS LD engine, and APIS (APIS 2004), an IMS QTI player. They have been integrated through CopperCore Service Integration Architecture (CCSI) (Vogten, Martens et al. 2006). CCSI was developed with the integration of different kind of services in mind, especially those defined in the service section of LD although other types of services are conceivable. In the execution of the peer assessment, a user interacts with SLeD in a normal way to play a learning process following the script. When a QTI document is used, the CopperCore engine will send the QTI document to SLeD. Then SLeD will ask for service from APIS player and render corresponding question for the user. When user finishes the answering the question, SLeD will send to APIS again for handling user's response. The results will be transferred to CopperCore according the coupling between the property and outcome defined in the script. The detail handling procedure can be seen in (Vogten, Martens et al. 2006). Figure 4 shows a screenshot of the user interface when learner2 is reviewing article 1.

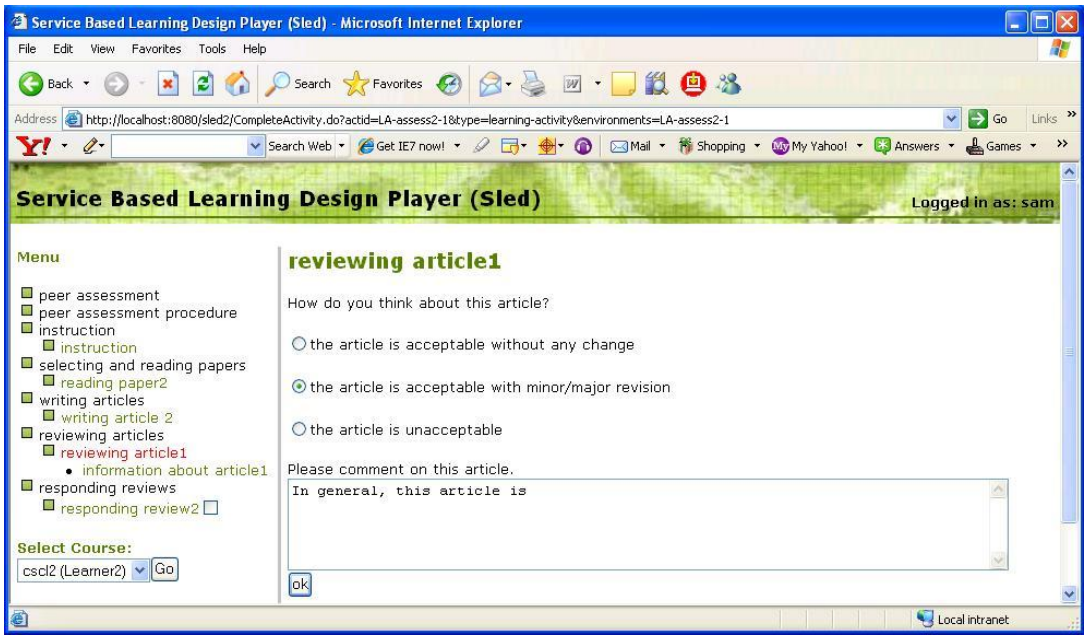

Figure 4. A Screenshot of Execution of the Peer Assessment Example

6. Discussion

In this section we discuss two issues: efficiency and flexibility. *Efficiency*: Rather than educational efficiency of a peer assessment, we discuss efficiency of technical approaches to develop and deliver an online peer assessment. As mentioned before, in typical software development methods, developers with programming competence has to spend about one man-year to design, code, compile, debug, and install a peer assessment tool. Our approach is fully based on open e-learning standards. As we have seen, standard-compatible authoring tools and run-time environments are available. The users with knowledge about programming and process modeling can be trained easily to script online peer assessment by using tools. To script a peer assessment process, one or several days may be enough for users who have process modeling competence that is possessed by most software programmers. In addition, because of interoperability, users can design a peer assessment based existing scripts of others through searching and modifying. It will extremely save a lot of time and efforts in development of online peer assessment. *Flexibility*: we discuss the flexibility of technical approach to develop and deliver a online peer assessment. As discussed in the second section, there are a variety of forms of peer assessment. The variation space of peer assessment is a combination of all variables changing in their value domains. Any software tool can only provide a limited flexibility. Additionally, once a software application tool has been developed, it is not easy to customize and add new functions to fit the changing contexts and specific needs. These software applications have their own data representation that is not usable by other applications. Their functions cannot be shared directly by other software tools as well. In contrast, our approach is based on open e-learning standards. A peer assessment script can be tailored and customized easily for their special requirements. They can be executed in any IMS LD player with any integrated IMS QTI player.

This technical approach has limitations. The required level of technical knowledge of IMS LD and IMS QTI for those authoring assessments is significant at the moment, because of the lack of easy to use graphical tools that support users in complex learning models. To acquiring such knowledge is not very difficult work for software developers and people with knowledge about programming and process modeling. However, when we try to extend a user group to include end-users like teachers and assessment designers, there is still a gap between the requirements of users and the functions that existing authoring tools can provide. In addition, if group interaction is extremely complex (e.g., in group composition, group dynamics, data structure of evidence, and data exchange patterns) and the number of roles and peer students increases, the complexity of the scripts will be too difficult to be handled even for experts. Therefore, new generations of authoring tools are expected to support practitioners to develop online peer assessment. One of the aims of the TENCompetence project (TENCompetence, 2006) is to develop such authoring tools.

7. Conclusions and Future Work

Peer assessment is a special pedagogical method that can be applied to develop critical thinking skills and improve communication skills. There is no such a form of peer assessment that "one size fits to all". Many different forms of peer assessment have been designed and reported. Existing tools supporting online peer assessment are developed in a typical software development method. A lot time and efforts will be spent for developing the tools. In addition, they can not be easily customized to fit the changing contexts and specific needs. We claim that a technical approach based on open e-learning standards can make the development and delivery of a peer assessment more efficiently and flexibly. In this paper, we analyze the strength and weakness of IMS QTI and IMS LD on supporting online peer assessment. We present a technical approach to script multiple users/roles involved group interaction needed in peer assessment by using IMS QTI and IMS LD complementarily. In order to help users to get benefits from this approach, design-time systems and run-time systems are developed and under development. Through using a peer assessment example, we present how users can be supported in scripting a peer assessment and in executing a peer assessment script. Through a discussion, we conclude that our approach based on IMS QTI and IMS LD, in comparison with typical software development methods, is a more efficient and flexible method to support online peer assessment.

However, existing IMS LD and IMS QTI authoring tools can not support average practitioners to script their own peer assessment. Our future work in this direction is to develop domain-specific language to represent the various facets of peer assessment. Such a language tends to support higher-level abstractions than general-purpose modeling language like IMS LD and IMS QTI, meaning that they require less effort and fewer low-level details to script a peer assessment. The scripts in such an assessment-specific language will be transformed into IMS LD code and QTI documents automatically, wrapped as a unit of assessment, and delivered in any standard-compatible execution environment.

References

Almond, R. G., L. Steinberg, et al. (2001). A sample assessment using the four process framework. CSE Report 543. Center for study of evaluation. Los Angeles, University of California.

APIS. (2004). Retrieved Oct. 20, 2006, from http://www.elearning.ac.uk/resources/1apisoverview.

Brindley, C. and S. Scoffield (1998). "Peer assessment in undergraduate programmes." Teaching in higher education 3 (1): 79-89.

Dillenbourg, P. (2002). Over-scripting CSCL: The risks of blending collaborative learning with instructional design. Three worlds of CSCL. Can we support CSCL. In P. A. Kirschner (Eds.), Heerlen, Open Universiteit Nederland: 61-91.

Falchikov, N. (2001). Learning Together: Peer tutoring in higher education. London, Routledge Falmer.

Falchikov, N. (2002). 'Unpacking' Peer Assessment'. Assessment: Case Studies, Experience & Practice from Higher Education. In P. Schwartz and G. Webb. (Eds.), London, Kogan Page.

Freeman, M. (1995). "Peer assessment by groups of group work." Assessment and Evaluation in Higher Education 20: 289-300.

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. In Proceedings of the 32rd SIGCSE Technical Symposium on Computer Science Education, 139-143, Charlotte, North Carolina.

IMSLD. (2003). "IMS Learning Design Specification." Retrieved September 15th, 2006, from http://www.imsglobal.org/learningdesign/index.cfm.

IMSQTI. (2006). "IMS Question and Test Interoperability Specification, Version 2.1." Retrieved September 15th, 2006, from http://www.imsglobal.org/question/index.html.

Kane, L.-. and E. E. Lawler III (1978). "Methods of peer assessment." Psychological Bulletin 85: 555-586.

Keppell, M., E. Au, et al. (2006). "Peer learning and learning-oriented assessment in technology-enhanced environments." Assessment & Evaluation in Higher Education 31(4): 453-464.

Kollar, I., F. Fischer, et al. (2005). Internal and external collaboration scripts in webbased science learning at schools. In: Proceedings of Computer Supported Collaborative Learning (CSCL2005): The Next 10 Years, 331-340, Mahwah, NJ: Lawrence Erlbaum Associates.

Koper, E. J. R. (2001). "Modelling units of study from a pedagogical perspective: The pedagogical meta-model behind EML". Document prepared for the IMS Learning Design Working Group. Heerlen: Open Universiteit Nederland.

Koper, R. and B. Olivier (2004). "Representing the Learning Design of Units of Learning." Journal of Educational Technology & Society 7(3): 97-111.

Miao, Y. (2005). CoSMoS: Facilitating Learning Designers to Author Units of Learning Using IMS LD. In: Towards Sustainable and Scalable Educational Innovations Informed by the Learning Sciences, Proceedings of the 13th International Conference on Computers in Education, 275-282, Singapore, IOS Press.

Miao, Y., K. Hoeksema, et al. (2005). CSCL Scripts: Modelling features and potential use. in: Proceedings of Computer Supported Collaborative Learning (CSCL2005): The Next 10 Years, 423-432, Mahwah, NJ: Lawrence Erlbaum Associates.

Ngu, A. H. H., J. Shepherd, 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.

O'Donnell, A. M. and D. F. Dansereau (1992). Scripted cooperation in student dyads: A method for analyzing and enhancing academic learning and performance. Interaction in cooperative groups: The theoretical anatomy of group learning. R. Hertz-Lazarowitz and N. Miller. (Eds), 120-141, London: Cambridge University Press.

Orsmond, P. (2004). Self- and Peer-Assessment: Guidance on Practice in the Biosciences. In e. a. Maw. (Eds), Teaching Bioscience Enhancing Learning.

Pellegrino, J. W., N. Chudowsky, et al. (2001). Knowing what students know: the science and design of educational assessment. Washington, DC: National Academy Press.

Pope, N. (2001). "An examination of the use of peer rating for formative assessment in the context of the theory of consumption values." Assessment & Evaluation in Higher Education 26(3): 235-246.

Rada, R., S. Acquah, et al. (1993). "Collaborative learning and the MUCH System." Computers and Education 20: 225-233.

Rushton, C., P. Ramsey, et al. (1993). "Peer assessment in a collaborative hypermedia environment: A case-study." Journal of Computer-Based Instruction 20: 75-80.

SLeD. (2004). "Service Based Learning Design System." from Website of the Service Based Learning Design System project: http://sled.open.ac.uk.

Sluijsmans, D. M. A., S. Brand-Gruwel, et al. (2004). "Training teachers in peer-assessment skills: effects on performance and perceptions." Innovations in Education & Teaching International 41(1): 60 – 79.

Somervell, H. (1993). "Issues in assessment, enterprise and higher education: the case for self-, peer and collaborative assessment." Assessment and Evaluation in Higher Education 18: 221-233.

TENCompetence. (2006). "TENCompetence Project Website." Retrieved November, 2006, from http://www.partners.tencompetence.org/.

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

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

Vogten, H. and H. Martens. (2004). "CopperCore." Retrieved 15 july, 2004, from http://www.coppercore.org.

Vogten, H., H. Martens, et al. (2006). CopperCore Service Integration - Integrating IMS Learning Design and IMS Question and Test Interoperability. In Proceedings of The 6th IEEE International Conference on Advanced Learning Technologies, Kerkrade, The Netherlands, IEEE Computer Society.

Weaver, W. and H. W. Cotrell (1986). "Peer evaluation: a case study." Innovative Higher Education 11: 25-39.

Weinberger, A., F. Fischer, et al. (2004). Knowledge convergence in computer-mediated learning environments: Effects of collaboration scripts. Proceeddings of the 85th Annual Meeting of the American Educational Research Association (AERA), San Diego, CA, USA.

Weinberger, A., K. Stegmann, et al. (2005). Computer-supported collaborative learning in higher education: Scripts for argumentative knowledge construction in distributed groups. In: Proceedings of Computer Supported Collaborative Learning (CSCL2005): The Next 10 Years!, 717-726, Mahwah, NJ: Lawrence Erlbaum Associates.

Acknowledgments

The work on this paper has been sponsored by the TENCompetence Integrated Project that is funded by the European Commission's 6th Framework Programme, priority IST/Technology Enhanced Learning. Contract 027087