Enabling Teachers to Develop Pedagogically Sound and Technically Executable Learning Designs

Open Universiteit
www.ou.nl

Enabling Teachers to Develop Pedagogically Sound and Technically Executable Learning Designs

Yongwu Miao, Marcel van der Klink, Jo Boon, Peter Sloep, and Rob Koper
Centre for Learning Sciences and Technologies, Open University of the Netherlands
Emails: (Yongwu.Miao, Marcel.vanderKlink, Jo.Boon, Peter.Sloep, Rob.Koper)@ou.nl

ABSTRACT

A learning design describes a sequence of learning activities that learners undertake in order to help them achieving particular learning objectives, including the resources and services needed to complete the activities. Unfortunately, there is no learning design language which can be used by teachers to explicitly describe pedagogical strategies and then can be interpreted by machine. This paper proposes an approach to support teachers' design processes by providing pedagogy-specific modeling languages for learning design. Based on activity theory, we have developed a conceptual framework, which can be used to specify pedagogical semantics and operational semantics of a pedagogy-specific modeling language. We present our approach by using peer assessment as an example. Through analyses we conclude that enriching pedagogical semantics and operational semantics of vocabularies of the modeling language may be a promising approach to developing a new generation of learning design languages, which enable teachers to develop pedagogically sound and technically executable learning designs.

1. Introduction

Open and distance learning provides learners with more accessibility and flexibility. However, distant learners and teachers may have difficulties to coordinate their interactions because they lack rich communication channels available in face-to-face contexts. There is a need to provide computational coordination mechanisms to orchestrate the activities. In addition, more and more computer application tools are used in distance education. It would be efficient for a computer-based mechanism to

automatically configure computer-based teaching and learning environments for the right people at the right time.

The emergence of the "learning design" concept holds promise as a possible solution to meet requirements identified above. A learning design is a description of a pedagogical strategy using a learning design language, an educational process modeling language. If a learning design is represented as a formal model, a machine-interpretable form, it can be used to coordinate interactions and to configure workspaces in a language-compatible execution environment (Koper & Olivier, 2004). A technically executable learning design can support teachers and students to focus on their substantive teaching and learning activities, minimizing concerns about coordination and logistical activities (Koper & Bennett, 2008). As a consequence, a formal learning design, if well-documented, can enhance the efficiency of the teaching and learning process in distance learning.

In the past decade many learning design languages have been developed. Falconer and Littlejohn (2008) distinguish two categories of learning designs: those meant for learning (the executable design) and those meant for teaching (the inspirational design). The audience of the former learning designs is a machine, and the audience of the latter learning designs is the teacher. In order to represent a learning design that can be processed automatically by a machine, the learning design languages in this category such as IMS Learning Design (LD 2003), E2ML (Botturi, 2006), LAMS (Dalziel, 2003) have to be specified in explicit syntax and semantics. These learning design languages enable to describe the detail, the logistics, and the technical services/tools required to execute the learning design (Agostinho, 2008). However, as noted by Oliver and Littlejohn (2006), these learning design languages can clearly describe the "mechanics" of a learning design, but can not illustrate clearly the pedagogical ideas of the design. That is, most pedagogical ideas are embedded in code unfamiliar to teachers, except for those partially represented in the attributes such as title and description.

In contrast, the learning design languages for teachers such as LDVS (Agostinho, et al. 2008, Bennett, et al. 2005), LDLite (Oliver and Littlejohn, 2006), and 8LEM (Verpoorten

et al. 2006) usually have no explicit syntax and semantics specified. The pedagogy and the rationale of the actual design are described informally through the use of textual description or/and visual diagrams. Such learning design languages are easy to use, but the learning design represented in these languages may be ambiguous and hence it is difficult and even impossible to be interpreted by computer based on current artificial intelligence technologies.

Oliver (2006), Waters and Gibbons (2004), and Agostinho (2008) have called for a common learning design language that can be applied and consistently understood by practitioners (like teachers), researchers, and technical support stuff. They have suggested that a notation system, similar to that found in other disciplines such as music and dance, is needed for learning design. Unfortunately, there is currently no such a notation system that can facilitate communication and sharing of learning designs among practitioners, researchers, developers, and the computer.

The objective of our work described in this article is to support practitioners to develop pedagogically sound and technically executable learning design. We explore a possible solution to design notations which can be used by practitioners to represent pedagogical strategies. Meanwhile, the syntax and semantics of the notations are defined explicitly so that they can be interpreted by machine. We argue that a pedagogy-specific modeling language may be a promising approach to help teachers to develop, communicate, understand, adopt, and adapt learning designs and enable a computer to interpret, instantiate, and automate learning designs in practice as well.

In this article, we first present a conceptual framework based on activity theory. Then we explain how to develop a pedagogy-specific modeling language by using peer assessment as an example pedagogy. Section 4 describes how to represent a peer assessment design using the peer assessment modeling language. After discussing several issues, we present our conclusions.

2. An Activity Theory Based Conceptual Framework

A learning design is a description of a sequence of teaching and learning activities. Activity is the central component of a learning design language. In this section, we briefly introduce activity theory, which provides us a powerful socio-cultural lens through which we can analyze most forms of human activity. Based on activity theory, we developed a conceptual framework which can be used for specifying the pedagogical semantics and operational semantics of notations.

2.1 Activity Theory

Activity theory is "a philosophical framework for studying different forms of human praxis as developmental processes, both individual and social levels interlinked at the same time" (Kuutti, 1996). The founder of this theory, Vygotsky developed the idea of mediation and claimed that human activities are mediated by culturally established instruments such as tools and language (Vygotsky, 1978). Leontiev further suggested that activities are also mediated by other human beings and social relations (Leontiev, 1981).

[Insert Figure 1 about here]

Activity theorists (Engeström, 1987, 1998; Kuutti, 1991) considered the activity to be the minimally meaningful unit of study. Engeström (1987) proposed the structure of human activity as illustrated in Figure 1. In this model, the subject refers to an individual or a group of individuals who are involved in an activity, for example, evaluation. The object refers to raw material (e.g., an article) or an abstract thing (e.g., an idea) to be transformed into outcome (e.g., comments). The instrument can be any tool (e.g., a pen) and any sign (e.g., jargon) that can be used to help the transformation process. The community is a group of people (e.g., teachers and students in a project) who share the same general object. The rules are laws, standards, norms, customs, and strategies that govern behaviors of community members within the activity system (e.g., a student has to

review at minimum 3 and at maximum 5 articles of his peer students in a week). A division of labor is the distribution of tasks and powers between the members of the community (e.g., student A is responsible for review the articles concerning the topic of constructivism).

[Insert Figure 2 about here]

Leontiev (1978) proposed a three-level structure of activity. As depicted in Figure 2, human activity is driven by an objective-related motive and carried out by a community. The activity consists of goal-directed actions or chains of actions performed by individual (or group). Actions are realized through operations, which are driven by the conditions of the concrete situation and are related to routinized behaviors performed automatically (Kutti, 1996).

Although we do not explore the full richness of the theory in this article, we will further demonstrate that it is fit for the purpose of our goal namely to decompose a pedagogical scenario into different components.

2.2 A Conceptual Framework

In this section we describe the various components of our conceptual framework.

Stage: The *stage* is used to describe the goal and intention of activities performed in a period of time. Multiple activities can/have to be performed in a stage to achieve the goal. A stage is completed when the goal is achieved or aborted. A sequence of stages makes up a learning/teaching process to be described in a learning design.

Role: The *role* is used to distinguish different types of people in the community. A role can be decomposed into sub-roles if necessary. Thus, a group or an organization can be modeled in a hierarchic structure. From the perspective of process modeling, a role can

have attributes and memberships. It is necessary to distinguish nominal roles (e.g. teacher and students) and behavior roles (e.g., tutor and chair).

Activity and Activity Structure: The *activity* is used to specify a logical unit of a task performed individually or collaboratively. It is very important but difficult to identify an activity in process modeling. For example, the statement 'each student in a pair writes a report' means that there are two individual activities. The statement 'two students in a pair write a synthesis' means that there is one collaborative activity. Sometimes, a complex process consists of a set of activities performed in sequence or/and in parallel. For example, first each student writes a draft independently and then they merge their drafts into a synthesis. For modeling such a complex process clearly, the concept of *activity structure* is introduced. Four types of activity-structures can be specified: *sequence-structure* (all activities will be performed in a prescribed sequence), *selection-structure* (a given number of activities selected from a set of candidate activities will be performed in any order), *concurrent-structure* (a set of activities is performed concurrently by the same individual or by different individuals), and *alternative-structure* (one or more activities will be selected to perform from a set of candidate activities according to a prescribed condition). The activity has generic attributes like title, description, state, and completion condition.

Object, Outcome, Artifact, and Information Resource: The *artifact* is used to represent an outcome that is created and introduced within an activity and then may be used and transformed in the same or/and other activities as an intermediate product and/or a final product. Because an artifact produced in an activity may be used as an input of another activity, artifact is a kind of object of the activity as well. The *information resource* is used to represent another kind of object, which is available (e.g., learning material) before and during the learning process and will not be modified and transformed in the whole process. For example, if an assessment form is available in design time and will not be changed during the learning process, it should be defined as an information resource. If an assessment form will be created by a participant in execution time, it should be defined as an artifact. Usually an artifact refers to a tangible or a digitalized

object such as an article, a physical model, a questionnaire, or a comment. An intangible outcome such as a friendship and a consensus is not an artifact. However, if an intangible outcome is explicitly represented in a form of digitalized entity and can be accessed later on, it can be regarded and modeled as an artifact as well. For example, a consensus can be represented as an artifact with two attributes: achieved (true/false) and a description of the consensus.

Service: According to activity theory, the instrument refers to all the means which the subjects have at their disposal for influencing the object and for achieving the goals. In a computer-based distance learning environment, we use the term *service* to specify a computer-based application for handling certain types of artifact/information resources (e.g., a test tool or a simulator) or for facilitating communication (e.g., a chat or a forum).

Action: The *action* is used to specify an elementary part of work such as allocating a task, finishing an activity, transferring an artifact and calculating the mean of a collection of scores.

Condition: The *condition* is used to represent a situation which can be interpreted in mathematic expression. Examples of conditions are whether an activity is completed, whether an artifact is available, and whether the scheduled time is over.

Rule: The *rule* is used to specify dynamic features of a learning process. It consists of conditions, actions and/or embedded rules in a form of 'if (condition) then (actions/rules) else (actions/rules)'. Sometimes, a rule can have no condition.

The components of the conceptual framework have been described above. Their relationships can be summarized as: following certain rules, people with various roles perform activities in sequence/parallel within various stages to produce outcome by using objects and services. We recognize that activity theory entails more components than we described. Our use is rather instrumental but it encouraged our conceptual thinking on how to define activities as a hierarchical structure consisting of several more detailed

components. In the next section, we will present how to develop a pedagogy-specific modeling language according to this conceptual framework through the use of peer assessment as an example pedagogy.

3 A Peer Assessment Modeling Language

In this section, we first describe the development of a peer assessment modeling language and then discuss how to represent a peer assessment design with the peer assessment modeling language.

3.1 The Development of a Peer Assessment Modeling Language

Peer assessment can be characterised as the process in which students collaborate and evaluate their own performance as well as those of fellow students (Sluijsmans, 2002). As is the case with any complex skill, becoming competent in peer assessment requires training and support in, for example, defining assessment criteria, judging the performance of peers and providing feedback for further learning. Research indicates that, especially in the initial development stages of the peer assessment skills, clear guidelines and supporting tools are essential for students and their teachers (Fastré, Van der Klink, Sluijsmans & Van Merriënboer, 2008). If peer assessment is not designed well then it can easily result into an ineffective learning experience (Boud, Cohen & Sampson 1999). The success of peer assessment depends greatly on how the process is set-up and subsequently managed. It would be very useful if a learning design language could be developed for teachers to represent and communicate the best practice of peer assessment, to make a peer assessment plan, and then to scaffold students conducting an online peer assessment in distance education.

A peer assessment can be represented in IMS-LD, which is an open e-learning technical standard. It is a pedagogy neutral learning design language that can be used to describe a

wide range of pedagogical strategies as computer executable models. However, it is impossible for practitioners to handle technical complexities required for authoring complicated learning designs like peer assessment using IMS-LD (Miao & Koper, 2007). Instead of making adaptations to IMS-LD, we adopt a domain-specific modeling (DSM) approach to develop an advanced learning design language. DSM is a kind of model-driven approach to developing software applications. It was originally applied to enable end-users to model business processes. A domain-specific modeling language raises the level of abstraction beyond programming by specifying the solution in terms of concepts and associated rules selected from the very domain of the problem being solved (DSM forum). In education, the concept of "domain" can be understood as the pedagogy. In a particular pedagogy, the vocabularies used to describe the pedagogy can have more specific and more elaborate meaning. For example, in peer assessment, the terms of 'responding', 'questionnaire', and 'score' have more specific and concrete pedagogical semantics than the generic terms like 'learning activity' and 'property' used in IMS-LD. In addition, the operational semantics of the vocabularies can be explicitly defined as well. For example, the data type of a 'score' is a number which can be assigned by the reviewer and then can be calculated and presented. If we choose these vocabularies, on the one hand, practitioners can intuitively apply these vocabularies to describe their pedagogical strategies. They don't need to describe a learning process by mapping the pedagogy-specific concepts to the coding terms or to very generic notations like activity, role, and property. On the other hand, a learning design represented using these vocabularies can be interpreted by a computer and transformed into executable models represented in a low-level modeling language like IMS-LD.

s

By adopting DSM to develop a peer assessment modeling language, we began by selecting various categories of vocabularies based on the conceptual framework and clarifying their pedagogical semantics and operational semantics. We reviewed relevant literature (e.g., Joosten-ten Brinke et al. 2007; Liu, 2001; Sitthiworachart, 2003; Sluijsmans, 2002; Topping, 1998; and Volder, 2007) to choose and compare vocabularies in an activity-centric manner. For each type of activity, we needed to specify in which stage and for which goal it would be performed. This raised a series of questions, such as:

which role would carry out this type of activity?; what kind of artifacts and possible information resources would be used as input and what kind of artifact would be expected outcome in this type of activity?; what kinds of services would be needed to complete the activity?; and what possible rules might be applied to specify how to accomplish an activity and how artifacts should be distributed? For example, we can define an activity called *responding* by specifying its semantics and articulating the relations to other vocabularies. The semantics of *responding* can be clearly defined within the context of peer assessment. *Responding* is an activity performed within the *evidence creation* stage by a *candidate* using a *test tool* to provide a *response* to a *questionnaire.* All vocabularies such as *evidence creation*, *candidate, test tool*, *questionnaire,* and *response* are pedagogically meaningful in a peer assessment. For each type of activity, certain rules can be selected. If necessary, parameter values should be assigned when the rule is applied. For example, a rule *if a time limit ( ) is over then completing the activity* can be defined and a time period (e.g., one hour) can be assigned to the parameter *time limit*.

In this way, we have identified all vocabularies and specified the semantics and syntax for the peer assessment modeling language. When we put all vocabularies together and specify their relations, a meta-model of peer assessment processes has been defined. According to the meta-model, four types of stages make up a peer assessment process: *assessment design, evidence creation, assessment,* and *reaction*. In the *assessment design* stage, one or more various activities such as *constructing an assessment form, designing an assignment, preparing material,* and *designing criteria* may take place. A *designer* can perform one or more activities and one activity can be performed by one or more *designers*. Performing design activities may produce *assignment description, assessment forms,* and/or *criteria*. Note that the *assessment design* stage may or may not be included in a peer assessment, because sometimes the *assignment description* and the *assessment form* have been pre-defined before a peer assessment process starts. Regardless of whether the *assessment design* stage is included, a peer assessment actually starts from the *evidence creation* stage, in which one or more *candidates* work on assignments such as *responding* to a *questionnaire* or *performing* tasks according to an *assignment description*. Sometimes, the performance will be observed and recorded by the *observer*.

Then the *assignment outcomes* or/and *record* will be produced and distributed to the activities in a subsequent *assessment* stage, in which one or more *reviewers* will evaluate the allocated *assignment outcomes* using the *assessment form* and *criteria*, and finally provide *feedback* in forms of *comments, rates, grades*, *ranking,* and so on. In summative peer assessments, the process may end here. In the formative peer assessment, typically a *reaction* stage will follow, in which the *candidate* may *view or review feedback.* Sometimes, candidates further *improve* assignment outcomes and even *require elaborate feedback*. In the latter case, the reviewer may *elaborate* on or provide additional *feedback*. In some situations, *reaction* stages and *assessment* can be repeated several times. Note that the generic activities like *reading, writing, presenting, discussing,* and *searching* can be performed in all stages and we will not discuss these types of activities here.

The meta-model can be used as a high-level process modeling language to specify various peer assessment models. In fact, many details of the meta-model are represented as alternatives (e.g., score, grade, rating, and comment), constraints (e.g., time), and rules (e.g., a given number of review tasks per reviewer). When specifying a peer assessment model, one has to represent the design decisions to be made in terms of the modeling language. Thus one should specify, for example, how many participants will be involved and what roles they will play; which kinds of assignments (e.g., an essay, a list of multiple-choice questions, or/and a demonstration) will be performed and whether individual candidates have a different assignment or the same one; whether each reviewer can review only one or more assignment outcomes of their peers; whether assessment is anonymous/confidential/public; whether assignment outcomes will be distributed reciprocally or mutually; etc. When all necessary design decisions are represented, a peer assessment model is created.

3.2 Representation of a Peer Assessment Design Using the Peer Assessment Modeling Language

Oliver and Littlejohn (2006) suggested representing pedagogical practice in an appropriate form that teachers can easily apply, adopt, adapt, and reuse. They developed a learning design language *LDLite*, with which the practitioners could represent a lesson plan with a sequence of online and face-to-face activities by using a matrix. In order to enable practitioners to use the peer assessment modeling language easily, we offer a similar form for practitioners to document a peer assessment design. However, our form is based on the meta-model and will be implemented as an authoring tool to support practitioners to represent a peer assessment design easily. As illustrated in Table 1, a peer assessment design represented in our peer assessment modeling language consists of two parts. Table 1.a describes the generic features of a peer assessment design. Table 1.b articulates the assessment procedure.

[Insert Table 1.a and Table 1.b about here]

In Table 1.a the cells with bold text are not editable area. A designer can represent design ideas by inputting information or selecting options. A designer, for instance, a teacher has to provide information for mandatory items which are indicated with the symbol '(m)' and others indicated by '(o)' are optional. For example, all participants, nominal roles, and their organizational structures must be listed. If there are expected outcomes that will be produced in the peer assessment, they should be listed as artifact items. In Table 1.b, a teacher can design a peer assessment process by choosing a stage from a set of pre-defined stages, for instance, an *assessment design* stage. Then the teacher can choose a type of activity which can be performed in the *assessment design* stage, for example, a *designing assignment* activity. If necessary, the teacher can specify attributes of the activity such as the *title* and *description* of the activity. Designing with an authoring tool, the teacher will be prompted to select a type of assignment such as a *text-based description* or a *questionnaire*. If the teacher selects a *questionnaire*, an instance will automatically appear in the outcome column at the same row as the *design assignment* activity. Then the teacher will be prompted to select an existing questionnaire or create a new questionnaire. For the former choice, the teacher should give a link to a Web page or to a local file where the questionnaire is accessible. For the latter, a questionnaire

authoring tool can be launched when the teacher clicks the questionnaire. The teacher also needs to specify which participants should play the role of the *designer* for this activity. If necessary, the teacher can create an *information resource* in the object column or drag an *information resource* item in the list from the first part of the form and drop it into this cell. For some types of activities, a special *service*, such as a *forum*, can be defined. By right-clicking the mouse-button in the rule column, situated *rules* will be presented from which the teacher can choose. For instance, by choosing *if a time limit ( ) is over then completing the activity* the teacher can assign a time period (e.g., one hour) to specify this rule. The authoring tool can check for conflicts and incomplete definitions on demand. Guided by the authoring tool in this manner, the teacher will represent a peer assessment design step by step until it is complete.

4 An example peer assessment design

Before implementing the target authoring tool, we conducted an internal test to investigate the possibility for teachers to represent peer assessment designs using the modeling language. The test was undertaken by two researchers who had substantial teaching experience in higher education but who did not participate in any of the previous development stages of the modeling language. Prior to the test they received instructions about the philosophy, background and a description of the modeling language. Then they selected two authentic peer assessment scenarios and decided how their peer assessment elements matched the modeling language. Because of the limitations on the space, we just briefly describe the first scenario, which was part of a course in learning theories for students attending a Masters program in educational sciences. The second example, which entails a peer assessment in traditional face-to-face teacher education, is not included in this paper.

[Insert Table 2.a and 2.b about here]

Table 2.a and 2.b illustrate a representation of the first peer assessment scenario, which was created by the two researchers using the peer assessment modeling language. The test results reveal that it is possible for teachers to represent their peer assessment practices using the modeling language, although it remains difficult. The difficulties can be summarized as: distinguishing the role of planner at design-time and the role of designer at run-time, using advanced modeling features (e.g., activity-structure and distribution rules), and representing highly complex peer assessments practices. These test results are valuable for the improvement of the modeling language and the development of the authoring tool.

5 Discussion

This section discusses two specific issues concerning the peer assessment modeling language and two relevant generic issues.

The first specific issue is whether a peer assessment modeling language can help teachers to develop pedagogically sound learning design. Because the conceptual framework of the modeling language is based on activity theory, it is hoped that teachers are able to easily understand the conceptual framework and think and reflect their design according to the conceptual framework. Furthermore, the vocabularies of the peer assessment modeling language are close to those teachers use in daily practice. They do not need to describe a learning process by mapping the peer assessment specific concepts to the generic notations like 'activity' and 'property' used in IMS-LD. The results of our test reveal that teachers, with little training, may be able to represent their own peer assessment scenarios and understand the peer assessment designs developed by others, although it remains difficult sometimes. Moreover, semantics and syntax of the language are also clearly defined as the meta-model. For example, *responding* activity has to be performed in *evidence creation* stage by a *candidate* to answer a *questionnaire* and to produce a *response* using a *test tool*. Another example is that the task load for each candidate would be better balanced. That is, pedagogical knowledge and principles are

embedded in the design of the peer assessment modeling language. They can affect the teachers' design in two ways. Firstly they can be used as guidance for teachers to make appropriate design decisions. For example, when a teacher wants to define an activity in an *evidence creation* stage, only appropriate activity types including *responding* will be suggested by the authoring tool. When the teacher makes a design decision, for instance, selects a *responding*, the associated design decisions (e.g., specifying a test tool, a questionnaire, and a response in the responding activity) will be further suggested and even automatically made by the authoring tool as default. Secondly, the pedagogical knowledge and principles embedded in peer assessment modeling language can be used to check whether a teacher's design violates certain pedagogical principles. For example, when the authoring tool checks up the design and finds that some reviewers have to review too many *assignment outcomes*, while others have been assigned too little work, warning messages will be presented to the teacher. Thus, the quality of the peer assessment design will be guaranteed to some extent in terms of pedagogical soundness.

The second specific issue is whether a peer assessment design can be executed in computer-based learning environment. Instead of developing an execution environment compliant with the peer assessment modeling we are going to transform the peer assessment design into a technically executable model represented in IMS-LD and IMS-QTI. In a comparison of the conceptual framework used for defining the peer assessment modeling language with the constructors of IMS-LD, we can find many one-to-one mappings from the components of the conceptual framework to the elements of IMS-LD. For examples, a *stage* can map to an *act*; a specific activity like a *responding* or *commenting* can map to a *learning activity* or a *supporting activity*; a specific role like a *candidate* or a *reviewer* can map to a *learner* or a *staff*; an information *resource* can map to a *learning object*; a *service* can map to a LD *service*. Although there is no corresponding concept of *artifact* in IMS-LD, an artifact like a *questionnaire*, an *assignment description*, a *comment*, a *status*, or a *grade* can be interpreted as an IMS-QTI assessment test or an IMS-LD *property* with a certain data type (e.g., a file, a string, a Boolean, or an integer). An environment will be generated for each activity within which all associated artifacts, outcomes and services will be included. Because of the technical

complexity and limited space, we will not discuss the transformation algorithm further in this article. In summery, a peer assessment design can be transformed into a executable model codified with IMS-LD and IMS-QTI.

Finally we discuss to generic issues. The first one is related to the language flexibility. We compare pedagogy-neutral approach and pedagogy-specific approach to developing an educational modeling language. As Koper (2001) pointed out, there are a lot of different stances when answering questions about learning. However, there are a lot of commonalities in education. Through abstracting these commonalities, IMS-LD was developed as a pedagogy-neutral modeling language. Because the elements defined in IMS-LD are quite generic, it is flexible enough to represent a wide range of pedagogies. In order to provide specific support for describing various peer assessment scenarios, a peer assessment modeling language was developed as a pedagogy-specific modeling language. In fact, a more generic assessment process modeling language (for modeling various forms of assessment processes) and a more specific peer assessment modeling language (e.g., for modeling the pure peer assessment involved with only two students) could be developed for working at different levels of abstraction. In general, there are hundreds of theoretical and practical theories and models of learning and instruction such as competence-based learning, problem-based learning, mastery learning, experiential learning, and case-based learning (Koper 2001). Teaching and learning based on a particular model has more commonalities than those in overall education. For example, although there are various strategies based on problem-based learning (PBL), there are many commonalities among the diverse PBL strategies. These PBL-specific commonalities could be abstracted to develop a PBL-specific modeling language. In particular, the commonalities could be abstracted at different levels, so that more specific PBL modeling languages could be developed used in different contexts. In theory, a more pedagogically generic modeling language can be used to represent a wider range of teaching/learning strategies. However, the user has to construct pedagogically specific 'mechanics' using generic notations. A more pedagogically specific modeling language is likely to be easier for teachers because the vocabularies are more pedagogically meaningful to them. However, such a language will lose flexibility to some extent

because the semantics and syntax of the notation have been more elaborately and specially defined.

The final issue is the degree of formality of the modeling language. Usually teachers are not accustomed to designing lesson plans formally. They may prefer to represent their design ideas informally. Even if their learning designs are not complete and accurate, teachers can interpret the designs using their knowledge and experience. However, a computer-based learning environment is currently not sufficiently intelligent to interpret incomplete and ambiguous representations. In order to make a learning design interpretable and executable, the learning design has to be represented in a formal modeling language and articulate all technical details such as appropriate date types and value domains. It is difficult and even impossible for ordinary teachers to handle the technical complexities. In order to enable both teachers and machines to interpret learning designs, our approach is to formally define high-level notations and rules as a meta-model. For example, when defining *feedback*, teachers only need to choose a *comment,* a *grade,* or another type of feedback. They do not need to choose a personal-/role-relevant property with a data type of a string or an integer. The operational semantics of the *comment* and *grade* have been defined in the language. Another example is the rule *exchange artifacts mutually*. This rule is formally defined in the language. When it is applied, the articulation of artifact-flows among the peers will be automatically specified according to the semantics of this rule. That is, if a modeling language is formally defined at a high-level appropriately, the teachers can avoid handling technical details and complexities. Meanwhile, the high-level notations and rules can be interpreted by a machine according to the operational semantics defined in the language.

Conclusions

In this paper, we have presented a conceptual framework for an educational process modeling language based on activity theory. We have outlined our technical approach to

apply the DSM paradigm to the development of a peer assessment modeling language based on this conceptual framework. Through analysis, we have drawn two conclusions. First, the commonalities of learning and teaching can be abstracted at different levels. The more generic the notations, the more flexible the modeling language, which can be used to describe a wider range of pedagogical strategies. The more specific the modeling language, the more support can be provided, particularly for inexperienced designers to develop a learning design. If pedagogical knowledge and principles are implemented in the pedagogy-specific modeling language, the corresponding authoring tool can use them as guidance and checking mechanisms to improve the quality of learning designs on pedagogical aspects. Second, through formally specifying the operational semantics of pedagogically meaningful vocabularies of an educational modeling language, a learning design represented in such a language is interpretable by a machine. Teachers will benefit from the use of such a language, because it is not necessary for them to handle technical details and complexities needed for the execution in computer-based learning environment.

As the next step, we will implement the peer assessment authoring tool based on the peer assessment modeling language. We will implement the mapping algorithms to translate a peer assessment design into IMS-LD code. In this paper, we emphasize that the language and authoring tool are developed for distance and e-learning settings, but a spin-off may be that it could also be used by teachers in face-to-face situations.

The internal test has detected some of the issues that require further attention in the development of both the language and the authoring tool. However, further evaluation definitely needs to be undertaken, in particular, during and after the development of the authoring tool. Substantial groups of teachers employed in various educational sectors will need to be included in future evaluation activities in order to establish more comprehensive insights into the usability of this language in daily teaching practice. In addition, the work can be extended in future to developing other pedagogy-specific modeling languages and corresponding authoring tools.

ACKNOWLEDGEMENT

REFERENCES

Agostinho, S. (2008). Learning Design Representations to Document, Model, and Share Teaching Practice, In Lockyer, L., Bennet, S., Agostinho, S., Harper, B. (Eds.). *Handbook of Research on Learning Design and Learning Objects: Issues, Applications and Technologies*, 1-19, IDEA group.

Agostinho, S., Harper, B., Oliver, R., Hedberg, J., & Wills, S. (2008). A visual learning design representation to facilitate dissemination and reuse of innovative pedagogical strategies in university teaching. In L. Botturi & T. Stubbs (Eds.), *Handbook of visual languages for instructional design: Theories and practices.* Information Science Reference.

Bennett, S., Agostinho, S., Lockyer, L., Harper, B., & Lukasiak, J. (2006). Supporting university teachers create pedagogically sound learning environments using learning designs and learning objects. *International Journal on WWW/Internet*, 4(1), 16-26.

Boud, D., Cohen, R. & Sampson, J. (1999). Peer learning and assessment. *Assessment & Evaluation in Higher Education*, 24(4), 413-426.

Boud, D. (2000). Sustainable assessment. Rethinking assessment for the learning society. *Studies in Continuing Education,* 22(2), 151-167.

Botturi, L. (2006). E2ML: A visual language for the design of instruction. *Educational Technology, Research and Development*, 54(3), 265-293.

Dalziel, J. (2003). Implementing learning design: The Learning Activity Management system (LAMS). In G.Crisp, D. Thiele, I. Scholten, S. Barker & J. Baron (Eds), *Interact, Integrate, Impact: Proceedings of the Conference of the Australasian Society for Computers in Learning in Tertiary Education*. Retrieved September 15th, 2008, from http://www.ascilite.org.au/conferences/adelaide03/program/conf_prog_index.htm

DSM forum, Retrieved September 15th, 2008, from http://www.dsmforum.org/,

Engeström, Y. (1987). *Learning by expanding: An activity-theoretical approach to developmental research*. Helsinki: Orienta-Konsultit.

Falconer and Littlejohn (2008). Representing Models of Practice, In Lockyer, L., Bennet, S., Agostinho, S., Harper, B. (Eds.). *Handbook of Research on Learning Design and Learning Objects: Issues, Applications and Technologies*, 20-40, IDEA group.

Fastre, G., Van der Klink, M., Sluijsmans, D. & Van Merriënboer, J. (2008). Support in self-assessment in secondary vocational education. In G. Kanselaar, V. Jonker, P. A. Kirschner, & F. J. Prins (Eds.), *International perspectives in the learning sciences: Creating a learning world. Proceedings of the eight International Conference of the Learning Sciences, Part 3* (pp. 29-30). International Society of the Learning Sciences

IMS-LD. (2003). *IMS Learning Design Specification.* Retrieved September 15, 2008, from http://www.imsglobal.org/learningdesign/index.cfm.

Joosten - ten Brinke, D., Van Bruggen, J., Hermans, H., Latour, I., Burgers, J., Giesbers, B., & Koper, R. (2007). Modeling Assessment for Re-use of Traditional and New Types of Assessment. *Computers in Human Behavior*, 23, 2721-2741.

Koper, E.J.R. (2001). *Modeling Units of Study from a Pedagogical Perspective: the Pedagogical Meta-model behind EML*. Open University of the Netherlands.

Koper, E.J.R., & Olivier, B. (2004). Representing the learning design of units of learning, *Journal of Educational Technology & Society*, 7(3), 97-111.

Koper, E.J.R., & Bennett, S. (2008). Learning Design: Concepts, in Adelsberger, H., Kinshuk, Pawlowski, J.M., & Sampson, D. (Eds.) Handbook on Information Technologies for Education and Training, 135-154. Springer.

Kuutti, K. (1991). Activity theory and its applications to information systems research and development, In H.E. Nissen, H.K. Klein, & R. Hirschheim (Eds*.), Information systems research: Contemporary approaches and emergent traditions*, 529-549, North Holland: Elvsevier Science Publishers B.V.

Kuutti, K. (1996). Activity theory as a potential framework for human-computer interaction research, In B.A. Nardi (Eds.)*, Context and consciousness: Activity theory and human-computer interaction,* Cambridge, MA: MIT Press.

Leont'ev, A. L. (1978). *Activity, Consciousness, and Personality*. Englewood Cliffs, NJ, Prentice Hall, 1978.

Leont'ev, A.N. (1981). *Problems of the Development of Mind*. Moscow, 1981, Progress Press. English translation, (Russian original published 1947).

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. & Koper, R. (2007). An Efficient and Flexible Technical Approach to Develop and Deliver Online Peer Assessment. In Chinn, C.A., Erkens, G., Puntambekar, S. (Ed.) *Mice, Minds, and Society*, *Proceedings of the 7th CSCL 2007*, 502-511, New Jersey, USA.

Oliver, R., (2006). Learning designs for ICT-based learning settings. *ASCILITE Newsletter*. Retrieved September 15th, 2008, from
http://www.ascilite.org.au/index.php/Newsletter:_July_2006

Oliver, R., & Littlejohn, A. (2006). Discovering and describing accessible and reusable practitioner-focused learning. In Minshull, G. & Mole, J. (Eds.), *Proceedings of Theme 1 of the JISC Online Conference: Innovating e-Learning 2006 (pp. 30-33).* Retrieved September 15th, 2008, from http://www.jisc.ac.uk/elp_conference06.html

Sitthiworachart, J., and Joy, M., (2003) Web-based Peer Assessment in Learning Computer Programming, *Proc. of IEEE ICALT03*, 180-184, Athens, Greece.

Sluijsmans, D. (2002). Student involvement in assessment. The training of peer assessment skills. (Doctoral dissertation, Open Universiteit Nederland: Heerlen).

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.

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

Verpoorten, D., Poumay, M., & Leclercq, D. (2006). The 8 Learning Events Model: a Pedagogic Conceptual Tool Supporting Diversification of Learning Methods. *Proceedings of International Workshop in Learning Networks for Lifelong Competence Development*, 48-53, Sofia, Bulgaria. Retrieved Sep. 30th, 2008, from
http://dspace.learningnetworks.org

Volder, M.D., Rutjens, M., Slootmaker, A., Kurvers, H., Bitter, M., Kappe, R., Roossink, H., Goeijen, J., and Reitzema, H., (2007). Espace: A New Web-Tool for Peer Assessment

with In-Built Feedback Quality System, *in Proceedings of ABR & TLC Conference*, Hawaii, USA.

Waters, S.H., & Gibbons, A.S (2004). Design Languages, notation systems, and instructional technology: A case study. *Educational Technology, Research and Development,* 52(2), 57-68.
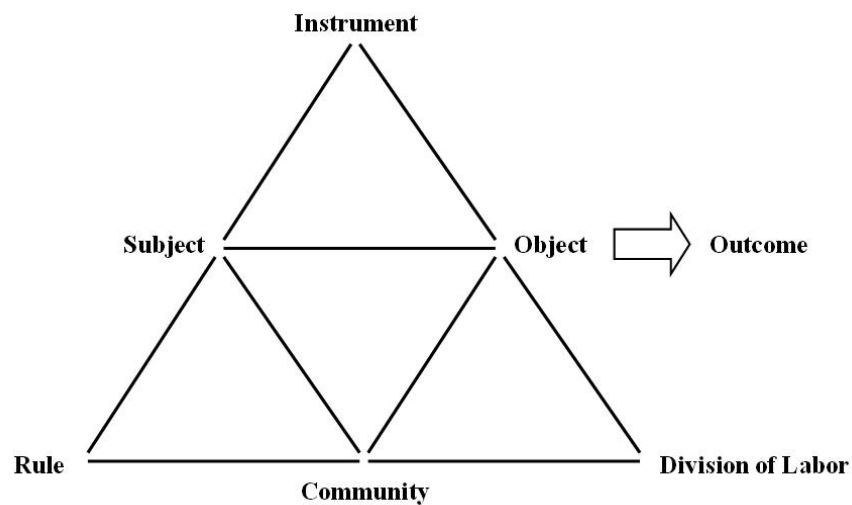
Figure and table caption

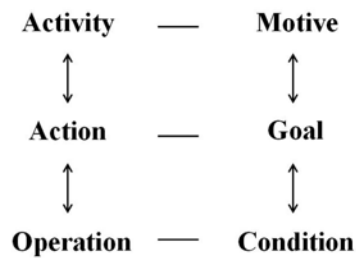Figure 1: Activity system [taken from (Engeström, 1987)]

Figure 2: The hierarchical structure of an activity [taken from (Kutti, 1996)]

Title (m)	A short name given to this design
Objectives (o)	Of staff and/or students? Time saving or cognitive/affective gains?
Prerequisites (o)	What work/knowledge is required before undertaking this assessment?
Description (o)	A narrative overview of the learning scenario
Design rationale (o)	Why make such a design?
Functions (o)	Diagnosis/placement/certification?
Estimated duration(o)	How long will this assessment take on average?
Privacy (o)	Anonymous/confidential/public?
Learning content and levels (o)	What qualifications will be assessed and at which level?
Population (o)	What are the characteristics of the target group?
Resources (o)	Available information units needed in this assessment
Artifacts (o)	Created information units as intermediate or final outcomes
Groups and Participants (m)	Who will be involved? - by assigning a label for each participant or nominal role

Table 1.a

stage	activity	role	object	outcome	service	rule
Choose one stage in each row from: *Design, Evidence creation, Assessment,* and *Reacting to feedback.* There is an option to specify how to complete the stage	Choose one or more activities within a certain stage. It is possible to choose an activity-structure with a type (sequence, concurrent, selection, and alternative).	In each type of stage, the behavior role has been defined such as *designer, candidate,* and *reviewer.* It is necessary to assign the participants who will play a behavior role for each activity. It is possible to assign multiple participants to perform an activity-structure.	Enumerating all objects needed to perform an activity or an activity-structure. There are two ways to specify objects. One is to specify an object by giving a web URL or a local file address. Another way is to indicate an artifact defined as an outcome.	Enumerating all outcomes produced within an activity or an activity-structure. A predefined type of outcome can be selected from a list of expected outcomes of each type of activity - tangible and intangible.	Enumerating all services needed to perform an activity or an activity-structure. A predefined type of service can be selected from a list of services useful for performing a given type of activity.	A list of pre-defined rules can be selected from to apply to each type of activity. In particular, it is necessary to specify the allocation of the activities and artifact exchange patterns.

Table 1.b

Table 1: Peer Assessment Design Form

Title	Peer assessment in a distance course on learning theories
Objectives	Increasing students' awareness of their own peer assessment skills
Prerequisites	The entire course entails nine tasks. Students need to finish their previous course tasks before they are allowed to start Task 6 with the peer assessment exercise
Description	•Each student needs to write an essay •The students upload the essay and the essay is distributed to a peer student •The peer student assesses the essay, writes a feedback report and uploads it •Students reflect on the received feedback, write a short note on how they will utilize the feedback for improving their own essay and then finalize their essay
Functions	The peer assessment exercise is obligatory for passing the exam of this course
Estimated duration	From start to finish approximately two weeks
Privacy	Peers provide anonymous feedback on the essay
Learning content and levels	Basic knowledge of Gagné's learning theory
Population	Students are adults and used to learning individually, at their own pace
Resources	•**Criteria**: List of criteria for the essay •**Assessment form:** Assessment form for providing feedback •**Assessment form:** Assessment form for reflecting on feedback
Artifacts	•**Assignment description**: General instructions •**Assignment outcome**: Concept essay1 •**Assignment outcome**: Concept essay2 •**Comment**: Feedback report1 •**Comment**: Feedback report2 •**Reflection**: Reflection report1 •**Reflection**: Reflection report2 •**Assignment outcome**: Final version essay1 •**Assignment outcome**: Final version essay2
Groups and Participants	•Teacher •Student1 •Student2

Table 2.a

stage	activity	role	object	outcome	servi-ce	rule
Design	Designing assignment: Prepare materials	Designer: Teacher		General instruction		
Evidence creation	Writing: Write essay1	Candidate: Student1	General instructions, list of criteria	Concept essay1		If (concept essay1) is submitted, then complete the activity.
	Writing: Write essay2	Candidate: Student2	general instructions, list of criteria	Concept essay2		If (concept essay2) is submitted, then complete the activity.
Assessment	Commenting: Review essay2	Reviewer: Student1	List of criteria, concept essay2	Feedback report1		
	Commenting: Review essay1	Reviewer: Student2	List of criteria, concept essay1	Feedback report2		
Reaction	Reflecting: Interpreting received feedback2	Candidate: Student1	Feedback report2	Reflection report1		
	Reflecting: Interpreting received feedback1	Candidate: Student2	Feedback report1	Reflection report2		
	Improving: improve report1	Candidate: Student1	Feedback report2 and concept essay1	Final version essay1		
	Improving: improve report2	Candidate: Student2	Feedback report and concept essay2	Final version essay2		

Table 2.b

Table 2: An example peer assessment design