IMS Learning Design Support for the Formalization of Collaborative Learning Patterns

Davinia Hernández Leo, Juan I. Asensio Pérez, Yannis A. Dimitriadis
Department of Signal Theory, Communications and Telematics Engineering
University of Valladolid, Camino del Cementerio s/n, 47011 Valladolid, Spain
{dherleo, juaase, yannis}@ulises.tel.uva.es

Abstract

Collaborative Learning Patterns (CLPs) are detailed descriptions of best practices in collaborative learning. These patterns provide a way for a representation of key aspects of CSCL (Computer-Supported Collaborative Learning) that is easy to understand by software developers. To formalize these CLPs we have focused our attention on IMS Learning Design (IMS-LD). IMS-LD provides a means of expressing many different pedagogical approaches (including collaborative learning), however we have found some limitations in reflecting learning experiences that are group-based. Although this specification supports multiple roles in a learning activity, it is not possible to specify how they are going to interact. This paper points out this deficiency and proposes an extension of IMS-LD. The process that can be followed in order to obtain a unit of learning based on a CLP is illustrated with an example.

1. Introduction

The application of Information and Communication Technologies in order to enhance education has always been present. The Computer-Supported Collaborative Learning (CSCL) domain is based on a new and strongly interdisciplinary paradigm of research and educational practice [9]. Its main features include highlighting the importance of social interactions as an essential element of learning [5], as well as the role of participatory analysis and design of the whole community when creating new technological environments. CSCL applications have to include support for collaborative activities and to offer the functionality desired by the set of potential actors that can participate in collaborative learning situations (teachers, students, and pedagogy experts, among others). The effort involved in the development of useful CSCL applications is only justified if they can be applied to a large number of learning situations and if they can survive the evolution of functional requirements and technological changes [10].

The creation of an environment that consists of modular integrated tools would provide great benefits for the development of reusable, flexible, and customizable CSCL applications. In so doing, the identification and dimensioning of tools are vital problems [2,6]. The fulfillment of this task largely depends on how the principles of the domain of interest are understood by software developers. In CSCL this problem is particularly important due to the big separation among abstractions used by experts in Collaborative Learning and those used by software developers. Traditional efforts for establishing a common ground among experts in the Collaborative Learning domain and software developers include top-down and bottom-up approaches.

The authors experience [6] shows how the intermediate approach of Collaborative Learning Patterns (CLPs) arises as a promising alternative for identifying reusable CSCL tools. A CLP can be understood as a way of describing a collaborative learning technique, easily understandable by software developers. Collaborative learning techniques dictate common ways of structuring interactions among participants in different collaborative learning activities, as well as the information they interchange [1]. CLPs are identified and described by collaborative learning practitioners, and validated by pedagogy experts using a formalism based on natural language. This fact makes the information provided by CLPs difficult to be used by computer-based applications such as authoring tools that could help teachers to select and integrate the CSCL tools they need in order to support a collaborative learning class. Therefore, a computer-oriented formalization of CLPs is required so as to broaden their applicability in CSCL scenarios.

To formalize these CLPs we are exploring the use of IMS Learning Design (IMS-LD) [8]. This Educational Modelling Language (EML) expresses the flow of any learning process in a formal way. Furthermore, it states that it provides a means of expressing many different pedagogical approaches. However, within the formalization of CLPs, we have found some limitations in reflecting learning experiences that are group-based. Thus, this paper proposes some extensions to IMS-LD in order to solve these deficiencies and illustrates the process for obtaining a CLP-based unit of learning.

Therefore, this paper is structured as follows: section 2 introduces the concept of CLP; section 3 analyses the requirements for the description of CLP-based collaborative learning scenarios and proposes an extension to the IMS-LD specification; an example of the process that can be followed in order to obtain units of learning based on CLPs is illustrated in section 5; and, finally, section 6 concludes this document and points out our future work.

2. Collaborative Learning Patterns

The term Collaborative Learning Pattern is derived from the notion of "Collaboration Design Pattern" introduced in [4] and defined as a way of describing "*[…] best practices in collaborative learning*". As it was described in the previous section, they are intended to reduce the conceptual gap between the collaborative learning field and the software development world and, therefore, they are useful in advancing towards the desired goal of obtaining reusable, customizable, and integrated CSCL software tools.

CLPs are represented according to a formalism, shown in Table 1, that enlarges the one previously described for "Collaboration Design Patterns" [4]. That table also shows an example of a CLP, drawn from a larger set that resulted from our analysis, defining a well-known practice in collaborative learning: *pyramid*. CLPs are supposed to collect knowledge from collaborative learning practitioners and, as it can be appreciated in Table 1, they do not contain any technical information.

Table 1 shows how a CLP provides software developers with information about the *flow* of learning activities types that are expected to happen during a collaborative learning scenario based on that CLP. Using this information, software developers can identify what type of CSCL tools could be needed in order to support collaborative learning scenarios compliant with the same CLP. Moreover, software developers can be confident on the fact that an important subset of those tools could potentially be reused in the support of several of those scenarios. That is why we propose CLPs as a good option for software developers to obtain information from the collaborative learning domain and, at the same time, fulfil the goal of reusability and adaptability of CSCL applications.

Furthermore, the information provided by CLPs could be used by software-based authoring applications that would guide teachers to select and integrate the CSCL tools that they need in order to support a collaborative learning classroom. However, this is not a trivial problem: the description of CLPs is based on natural language due to the fact that they are proposed by non-technical people. That means that software tools cannot process CLP definition. Therefore the following section explores the use of the IMS-LD specification towards a more formal description.

Table 1. CLP structure and its application to Pyramid-like collaborative learning activities

Facet	Explanation	Example #1	
Name	Name of the CLP	**Pyramid**	
Problem	Learning problem to be solved by the CLP	Complex problem, usually without a specific so-lution, whose resolution implies the achievement of gradual consensus among all the participants	
Example	A real-world learning activity capable of being structured according to the CLP	Collaborative proposal of the design of a computing system where each participant contributes with a design that is subsequently compared with other contributions and refined	
Context	Environment type in which the CLP could be applied	Several participants facing the collaborative resolution of the same problem	
Solution	Description of the proposal by the CLP for solving the problem	Each individual participant studies the problem and proposes a solution. Groups of participants compare and discuss their proposals and, finally, propose a new shared solution. Those groups join in larger groups in order to generate new agreed proposal. At the end, all the participants must propose a final and agreed solution	
Actors	Actors involved in the collaborative activity described by the CLP	- Teacher - Learner - Evaluator	
Types of Tasks	Types of tasks, together with their sequence, performed by the actors involved in the activity. (NOTE: due to space restrictions only types of tasks performed by learner and teacher are shown)	**Learner:** 1.Access to the information 2.Individual study of the problem 3.Individual solution proposal [REPEAT 4.Group formations 5.Group discussion 6.Common solution proposal] (Until only one group remains) 7.Process self-evaluation	**Teacher:** 1.Global problem definition 2.Provision of useful information 3.Group dimensioning 4.Decisions about control of time 5.Activity progress monitoring 6.Result evaluation
Types and structure of Informa-tion	Description of the types of information identified in the colla-borative activity and how they are related	- Input information needed for global problem resolution - Intermediate resolution proposals - Global problem resolution proposal - Correct global problem resolution (optional)	
Types and structure of Groups	Description of the types of groups of learners identified and how they are related	- Growing pyramid groups	

3. IMS-LD extension for CSCL

E-learning standardization efforts are now moving from content delivery resources to Educational Modelling Languages (EML), which are focused on the performance of individual and group learning activities [3]. We have chosen IMS-LD for the formalization of the CLPs because of its pedagogical flexibility. Its objective is to provide a framework of elements that can describe any learning design in a formal way [8]. A *learning design* is a description of a *method* enabling learners to attain particular *objectives* by performing learning *activities* in a certain order in the context of a learning *environment*. The environment consists of the appropriate *learning objects* and *services* to be used during the performance of the activities. A method contains the *play*, which is modelled according to a theatrical play with *acts* and *role-parts*.

IMS-LD affirms that it supports group and collaborative learning of different kinds. It enables the design of processes that include several *roles*, each of which can be played by several people (a group). A collaborative learning experience can be described by associating multiple people and/or multiple roles to the same learning activity. Furthermore, IMS-LD enables their activities to be specified in coordinated “learning flows” that are analogous to groupware workflows [11]. Therefore, IMS-LD is a reasonable candidate as a language with which to formalize the CLPs.

However, while a main feature of CSCL applications is the set of mechanisms that support the collaborative interaction, IMS-LD provides no means to specify how the members of a group interact within each learning activity. It only states that if multiple individuals are to collaborate or work together at the same time, this has to be done through a *service* in their assigned environment which supports this collaborative capability [8]. Therefore, the concept of *service* is central in IMS-LD for CSCL. An IMS-LD service specification describes the characteristics of a *resource* that supports a learning activity. When applying an IMS-LD learning design to a actual learning scenario the learning designer must specify the resources that, at last, provide the implementation of the defined services, thus obtaining a so-called *unit of learning*. These resources range from a simple blackboard or a paper sheet to a complex e-learning or CSCL application.

IMS-LD only proposes and defines four basic services, two of which are (to some extent) collaborative: discussion forum and e-mail. IMS-LD states that other needed services (collaborative or not) should be specified by the designers of learning scenarios. The problem is that IMS-LD does not allow the aforementioned designers to describe collaboration-related capabilities when defining a new service: type of awareness information needed and provided by the service, floor control policy that guides learners actions, communication skills required to these learners, etc. In this context we propose an extension of the IMS-LD service definition consisting of the definition of a special type of service, called g*roupservice*, whose main characteristics are summarized in Figure 1.

This generic characterisation of collaborative services, together with the definition of "learning flows" provided by IMS-LD, would enable scenarios in which existing CSCL tools could be selected and integrated in order to support a complete (and potentially complex) set of learning activities. Furthermore, and thanks to the formal nature of IMS-LD, this selection and integration of CSCL tools could be automatically performed by authoring tools thus hiding software engineering problems to learning designers (e.g. teachers).

The proposed extension to IMS-LD is also useful for the formalization of CLPs [7]. In this sense, a CLP can be understood as an "incomplete" learning design that has to be customized in order to generate a complete one. An IMS-LD definition of a CLP includes the description of *groupservices* in which some of their collaborative characteristics are not specified. Nevertheless, the level of *groupservice* descriptions provided by a CLP is enough for the identification of types of services needed by all the learning designs that could be derived from the same CLP. This fact, in a CSCL environment, helps software developers to decide what characteristics a CSCL tool should possess in order to be potentially reused in the different learning designs that are compliant with the same CLP.

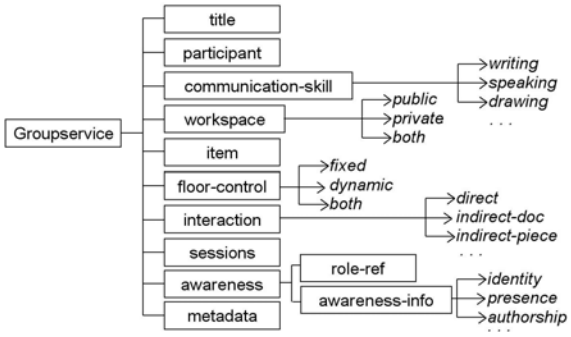


Figure 1. Scheme of the proposed extension to the IMS-LD service element

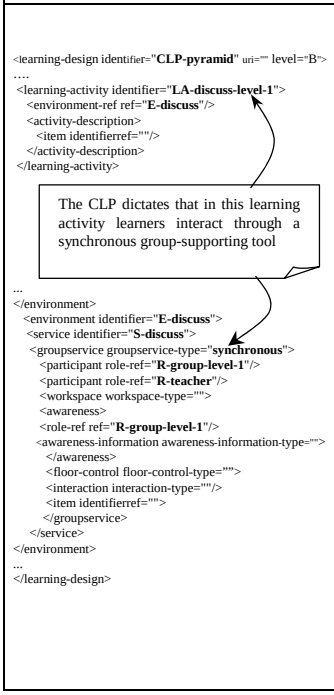
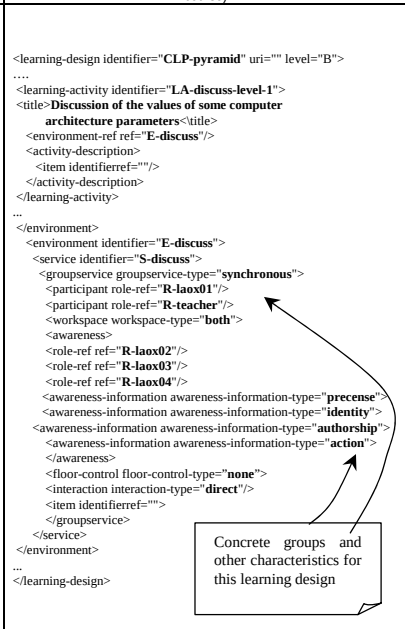
4. Units of learning based on CLPs

In this section, we present an example which illustrates how a *unit of learning* can be achieved using a CLP. A CLP-based unit of learning consists of a customized CLP-based learning design and a set of particular resources that depend on the concrete learning scenario.

Table 1 showed the example of the pyramid CLP, where several individuals join successively in larger groups in order to reach an agreed solution of the same problem. The learning flow of the CLP can be expressed in an IMS-LD play. The play consists of a sequence of acts. Each act represents a pyramid level, i.e., whenever people join in a larger group to compare and discuss their proposals, and propose a shared solution. In each act, different activities are set for the different roles (learner, teacher and evaluator) and are performed in parallel. Column 1 of table 2 illustrates the IMS-LD description of a pyramid CLP learning activity. When an act is completed, the next act starts until the completion requirements for the learning design are met (last level of the pyramid). The interactions of the individuals of each group in the pyramid are mediated by a type of service described as a *groupservice*.

The pyramid CLP has been applied by the authors to the specification of a learning design that supports a course on computer architecture for Telecommunications Engineers in our University. Column 2 of table 2 represents the teacher customisation of the pyramid CLP IMS-LD description (column 1) for the course. This is an example of a pyramid CLP-based learning design. When the teacher determines the binding of this learning design with concrete CSCL tools, an example of a pyramid CLP-based unit of learning is achieved. Column 3 of table 2 shows a unit of learning in which a particular implementation of a *groupservice* is referenced within the CLP-based learning design. This *resourse* is a collaborative labelling of parameters tool that enables the discussion and agreement of some computer cache design parameters. In order to support that unit of learning (column 3), the authors have developed a component-based CSCL application [6]. Thus, table 2 illustrates the three-stage process for obtaining a unit of learning based on a CLP.

Table 2. A partial example of the process for obtaining a unit of leaning based on a CLP

1. IMS-LD description of the pyramid CLP (The description of a well-known best practice)	2. A pyramid CLP-based learning design (The teacher customizes the previous description of the CLP for a course)	3. A pyramid CLP-based unit of learning (The binding of the previous learning design for the course with concrete CSCL tools)
		

5. Conclusions

This paper has introduced the concept of Collaborative Learning Pattern (CLP) as a promising approach for establishing a conceptual common ground among collaborative learning practitioners and software developers of CSCL applications. The paper has also motivated and described the process of applying IMS-LD in order to formalize CLPs. This process has detected several deficiencies in IMS-LD for CSCL. An adequate IMS-LD support of collaborative learning activities is expected to enable scenarios of easy, teacher-oriented integration and reuse of existing CSCL applications. In order to solve these deficiencies, the paper has proposed a set of extensions to IMS-LD focused on the definition of a new type of services for supporting group interactions. These extensions still have some limitations: they support very limited awareness and floor control models, and they do not allow the specification of privileged roles in group interactions, among others.

Several short-term activities are under way within our research group in order to enhance the above contributions. First of all, we are enlarging the set of available CLPs provided by collaborative learning practitioners in order to deeply validate the CLP approach itself and also in order to have a broader knowledge of concepts and principles of the collaborative learning domain. Secondly, we are currently working on the improvement of the proposed extensions to IMS-LD by adding more collaborative-related expression capabilities. And a third activity is aimed at the use of IMS-LD definition of CLPs as the basis for the integration of concrete CSCL tools in order to support different learning scenarios within real courses in our University.

All this work is expected to implicitly validate IMS-LD as a potential standard for the definition of collaborative learning designs based on CLPs, and also for the generation of authoring and supporting software tools enabling teachers the easy integration and customisation of CSCL tools.

Acknowledgements

This work has been partially funded by European Commission Project EAC/61/03/GR009, the Autonomous Government of Castilla and León, Spain (project VA117/01), and the Ministry of Science and Technology, Spain (projects TIC2000-1054 and TIC-2002-04258-C3-02). The authors would also like to thank the rest of “Intelligent & Cooperative Systems Research Group” at the University of Valladolid for their support and ideas.

References

[1] Asensio, J. I., Dimitriadis, Y. A., Heredia, M., Martínez, A., Álvarez, F. J., Blasco, M. T., and Osuna, C. A., "From Collaborative Learning Patterns to Component-based CSCL Applications", *European Conference on Computer Supported Collaborative Work (ECSCW03), Workshop: From Good Practices to Patterns,* Helsinki, Finland, Sept. 2003.

[2] Bote, M. L., Vaquero, L. M., Vega, G., Asensio, J. I., Gómez, E., and Dimitriadis, Y., "GRIDCOLE: a Grid Collaborative Learning Environment", *Proceedings of the 4th International Symposium on Cluster Computing and the Grid (CCGrid 2004), Workshop on Collaborative Learning Applications of Grid Technology (CLAG 2004),* Chicago, IL, USA, Apr. 2004.

[3] Caeiro, M., Anido, L., and Llamas, M., "A Critical Analysis of IMS Learning Design", *Proceedings of the Computer Supported Collaborative Learning Conference, CSCL 2003*, June 2003, pp. 363-367.

[4] DiGiano, C., Yarnall, L., Patton, C., Roschelle, J., Tatar, D., and Manley, M., "Collaboration Design Patterns: Conceptual Tools for Planning for the Wireless Classroom", *Proceedings of the IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE'02),* 2002.

[5] Dillenbourg, P. *Collaborative Learning: Cognitive and Computational Approaches*, Oxford, UK: Elsevier Science, 1999.

[6] Dimitriadis, Y., Asensio, J. I., Martínez, A., and Osuna, C., "Component Based Software Engineering and CSCL: Component Identification and Dimensioning", *Upgrade (digital journal of European Profesional Informatics Societies), special issue on e-learning: Boarderless education*, vol. 4, no. 5, Oct. 2003, pp. 21-28.

[7] Hernández Leo, D. *From IMS-LD to MDA: Approaches to Component-based CSCL Applications Modeling (in Spanish)*, MSc. Thesis, University of Valladolid, Valladolid, Spain, 2003.

[8] IMS Global Learning Consortium. *IMS Learning Design v1.0 Final Specification*, Accessible in http://www.imsglobal.org/learningdesign/index.cfm, 2003.

[9] Koschmann, T. Paradigm shift and instructional technology. In: *CSCL: Theory and Practice of an emerging paradigm*, ed. Koshmann, T. Lawrence Erlbaum, 1996, pp. 1-23.

[10] Roschelle, J., DiGiano, C., Koutlis, M., Repenning, A., Phillips, J., Jackiw, N., and Suthers, D., "Developing Educational Software Components", *Computer*, vol. 32, no. 9, Sept. 1999, pp. 50-58.

[11] Vantroys, T. and Peter, Y., "COW, a Flexible Platform for the Enactment of Learning Scenarios", *Proceedings on the 9th International Workshop on Groupware (CRIWG 2003),* Grenoble, France, 2003, pp. 168-182.