

Collaborative Learning Models on Distance Scenarios with Learning Design: A Case Study

Luis de la Fuente Valentín, Abelardo Pardo,
Carlos Delgado Kloos
Department of Telematics Engineering, Carlos
III University of Madrid
{lfonte, abel, cdk}@it.uc3m.es

Juan I. Asensio-Pérez, Yannis A. Dimitriadis
School of Telecommunication Engineering,
University of Valladolid
{juaase, yannis}@tel.uva.es

Abstract

Collaborative learning models are widely used in educational institutions. These models require a high interaction level among students and are mainly oriented towards in-class scenarios. But when collaborative models are deployed in a distant scenario, user expressiveness is significantly reduced thus creating a gap that hinders the effectiveness of this collaboration. A computer-supported model provides a set of tools to compensate for the distant scenario and reduce this gap. This paper presents the issues and solutions derived from the design and deployment of a complex collaborative model in a distant scenario. The course structure was captured using the Learning Design specification, and an architecture based on Virtual Network Computing was used to provide the required collaborative tools. The course was included as part of a regular undergraduate program in three higher educational institutions.

1. Introduction

Collaborative learning, where students interact with peers to achieve certain objectives in a set of activities is a technique that has been shown to significantly improve the effectiveness of a learning experience [12]. Rather than working in isolation, students are organized in groups where certain interaction patterns are suggested. Tutors supervise this interaction and support students to guarantee progress towards the objectives.

Due to the numerous technological resources available to students and tutors, collaborative learning is usually supported by computers in what is known as computer supported collaborative learning (CSCL) [13]. This level of support may range from purely administrative tasks, to full support of the interaction through platforms such as videoconferencing.

One of the main challenges when designing a learning experience with collaboration is that of structuring the overall learning process as to trigger productive argumentation among students [8]. This structure needs to specify numerous details, from the upper level resources to be used, to the low level administrative tasks such as role manipulation. In CSCL, the difficulty is frequently answered by capturing a description of the interaction with the so called “scripts” (see [6, 9]). Furthermore, the IMS Global Learning Consortium released in February of 2003 the first version of the Learning Design specification [5]. This specification offers a generic framework to capture a wide range of pedagogical strategies, included those based on collaboration. In fact

, the expressive power of Learning Design to model collaborative scripts has been previously studied in the literature [6] showing how typical CSCL interaction patterns may be well captured with Learning Design although a detailed knowledge of the specification is required.

In addition to the design of the experiences, new difficulties arise when deploying and enacting them in real-life technological scenarios. Interestingly, many experiences take place within the scope of a Learning Management System (LMS), and therefore they need to be tightly coupled with it. However most of the tasks required for a collaborative experience need the support of one or several particular computing tools, not necessarily provided by the LMS. It is when those tools need to be integrated within the LMS, structured as a single learning experience, and deployed for a set of real students when a new set of problems appear.

In this document, the issues that arise when designing, deploying and enacting a fully collaborative learning experience with remote students have been empirically observed. A set of students from three geographically distant higher level institutions was selected to participate in a course based on collaborative learning. The structure of the course was

captured as a Unit of Learning using Learning Design and deployed in a conventional Learning Management System. Student interaction was conceived to use several patterns and multiple technological resources were offered to support it. The interaction was supported by a set of tools all seamlessly integrated through Virtual Network Computing (VNC) architecture.

The rest of this document is organized as follows. Section 2 describes the collaborative model used in the experience. The deployment and adoption of such model to the learning scenario is described in Section 3. The paper concludes with the analysis of the obtained results as well as the conclusions of the study.

2. Collaborative Learning Model

The case study presented in this paper is a shared experience among three higher-education institutions, and it was included as part of their regular undergraduate programs. The experience took place in nine two-hour sessions during November and December, 2007.

2.1. Student Profiles

A total of 31 participants took part in the experience: 27 students and 4 tutors. Participants at two of the institutions were undergraduate students of the Telecommunications Engineering degree and for them the course was optional. On the contrary, for the rest of the participants, undergraduate Computer Science students, it was part of a larger non-optional programming course.

The learning structure was then defined for a group of nine students, and three identical course replicas were enacted independently from each other. In order to maximize availability, tutors were not assigned to any particular instance. Henceforth, the described learning flow refers to a group of nine students.

Since all participants are computer engineering students, they are assumed to have basic computer skills. It was also assumed that students from the same institution had similar programming skills, and therefore three different profiles were created. The motivation for this division is that collaboration with different skilled peers from a different institution may increase the discussion effectiveness [3]. It is worth mentioning that one of the institutions participating in the course follows a distant education paradigm for their regular courses. As a consequence, students from this institution have different scheduling requirements, and in most cases their involvement in the activities is purely asynchronous.

2.2. Activity Flow

On a collaborative learning model, students are supposed to be active participants. Success highly depends on student motivation [1] so the course topic must be carefully selected in order to ensure positive participation. The described case was based on the study of Drupal, an open-source Content Management System written in the PHP scripting language¹. This topic was selected based on the following observations:

- PHP is one of the most popular scripting languages used for fast web development, and yet, it was not fully covered by any of the regular undergraduate courses. Therefore, a course on the subject was expected to be attractive for those students interested in improving their curricula.
- The Drupal platform is a real-life application that allows students to modify current features and see the effects quickly in a mature environment, thus opening the possibility of new developments.
- The learning model for this type of platform requires certain degree of self-study. Drupal users' community has created a set of complete, well-structured set of information where students easily can find additional resources.

The proposed learning flow for the course is based on a combination of collaborative learning flow patterns, or CLFPs [10]. The overall structure leads user activities to a global goal, being problem-based learning [4] the underlying strategy.

In the described case study, the activity sequence is given as follows:

On the first individual stage students examine provided documentation. The requested output -a brief summary of the readings- creates a link with the next activity. In a second phase, students work in three-member groups to peer-review their summaries and agree on a common conceptual map of the discussed ideas according to the "Peer-reviewing" CLFP.

Once students receive the statement of the problem itself, a group decision sets the development strategy, which will be put in practice individually. All groups work on the same problem. However, each group assignment is focused on a different aspect of the solution. Students become then *experts* of a given aspect.

Next, students are re-organized into *jigsaw* groups, where members are experts on different aspects, promoting positive interdependency.

¹ <http://www.drupal.org>

In the last activity, according to the “Jigsaw” CLFP, developers have to work in their jigsaw groups and join their partial work to obtain a complete solution. An overview of the learning flow is depicted in Figure 1 for one particular participant. The “shape” of the student (e.g. triangle) represents his/her expertise while the “color” (e.g. white) indicates the *jigsaw* group he/she belongs to.

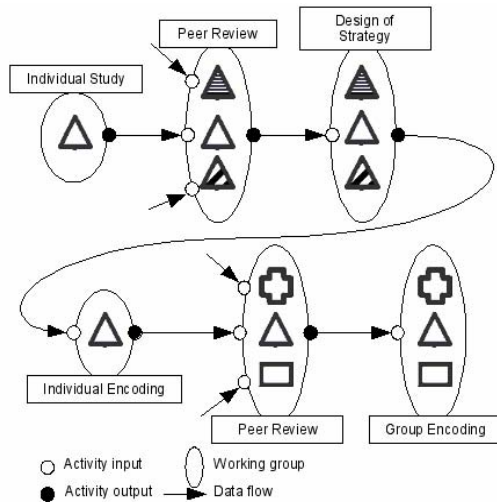


Figure 1: Example of learning flow for the student “red-triangle”.

To avoid excessive distance between user skills and provide a framework where peer-review is effective, students are encouraged not to use any extra time to finish activities.

3. Adaptation to the scenario

The learning model presented in the previous section has no particular reference to the deployment scenario, being typically deployed in a face-to-face environment.

When applying this model to a distant scenario - as done in the described experience - both student expressiveness and model flexibility are reduced. The course deployment phase has to be adapted to the scenario in order to minimize the above problems. The use of supporting tools allows the inclusion of certain degree of flexibility in the model without conflicting with intrinsic constraints [9].

3.1. Distance gap reduction

Tools for collaborative work allow multiple users to synchronously manipulate different resources, providing a framework where collaboration is more fluid. However, student profiles do not guarantee

availability for synchronous sessions. As a consequence, an average of one student of each group does not use synchronous tools.

To support different profiles, the use of both synchronous and asynchronous tools is merged in the activities: discussion recordings were used as input for asynchronous students, who are assigned to review discussed topics and arguments. A forum is used for the review, so that a parallel discussion is encouraged.

The joint use of synchronous collaborative tools and asynchronous communication facilities reduces the gap created by the distant scenario and allows students with different profiles to work together. The influence of this approach on the learning model is discussed in the following section.

3.2. Computational representation and model flexibility

Course deployment on distant scenarios requires a design involving technological requirements. Learning model representation can be done by a computer interpretable script, enhancing reusability of the model.

The Learning Design specification [5] provides support for a wide range of pedagogies to be expressed. Its capabilities for modelling collaborative activities [6] match with scenario requirements. The specification is divided in three levels, each of them increases the modeling capabilities of the script. Level A introduces the core system and allows activities to be sequenced or delivered in parallel. Level B adds the *course state* concept, modeling it with the so-called properties. A tracking system allows conditional behavior of contents depending on the course state. Level C allows communication facilities to be added to the learning model.

A course written in IMS LD is packaged in a self-contained file, which can be deployed several times to create course replicas. This feature is especially relevant when a given model must be run several times in parallel.

Learning Design can also provide run time flexibility of the model as defined in [7]. To allow run time modifications on the course behavior they must be considered during the design. LD properties can be used to add flexibility to the model. They have been used in the described design to achieve the following purposes.

Group set-up: the high level of interaction required in the model is reflected in a complex grouping model. The supporting system must be flexible enough for the management. In our design, working groups are related to LD property values. Thus, group behavior is defined

at course instantiation and the tutor can modify it during the course if required.

Adaptation issues: synchronous collaboration-based activities are not always suitable for all students, especially when they are from different institutions. In our case, one of the participant institutions follows a distant learning model for regular courses so in most cases students schedule does not match with peers. The asynchronous students are given an adapted assignment, where tasks are modified to match the case.

The students are also capable to modify their own profile. This fact, used in conjunction with the capability of conditional delivery of contents of IMS LD adds more flexibility to the model.

However, the use of a scripting approach such as Learning Design also introduces difficulties in the experience. First, the design becomes a complicated task: existing authoring tools do not always hide the complexity of the underlying language and thus the author requires a deep knowledge of the specification to achieve good results.

Another disadvantage would appear if severe deficiencies are found in the model at run time. Despite the described flexibility in the model, supporting software does not allow changes in the learning flow (run time script edition) once the course has been deployed. This is especially relevant in a scenario where the design task is considered error prone [2].

The system architecture that supports the model is composed of two different servers. The LD server contains the learning flow and delivers the assignments to students. These assignments are linked to VNC based collaborative tools, hosted on a second server. Relevant data is automatically synchronized between servers to avoid database inconsistencies.

4. Results and discussion

Since all requirements in the course design and deployment were covered, the experience has been considered successful. However, the feedback obtained from the participants offered multiple observations that should be taken into account in future versions.

The data gathered from the study has been collected through questionnaires that participants filled at the beginning of the course (to know their expectations and background), as well as after the experience was over (to obtain their viewpoints). Some additional evaluation data such as messages posted in the course forums and systems logs were also available. Questionnaires had a mixture of quantitative and qualitative questions and their analysis followed the principles of a mixed evaluation method [11].

Evidences extracted with quantitative data are supported by the analysis of different data sources, including qualitative data. Main findings are summarized in Table 1 and discussed in this section.

Table 1: Main findings including selected supporting data and data sources.

Finding	Sample of Supporting Data	Source
Supporting Technology is not mature enough	- "Technical failures derived on delays when doing activities"	quest
Students did not get a perception of having an integrated solution	- "You did not have freedom to move between tools. Their slowness make them appear as something external"	quest
Collaboration was perceived as a positive factor on a course	- Opinion about collaboration level with peers [0-5]: 3,49	quest
Asynchronous support widely used. Users missed synchronous support	- 195 participations on forums - "Sometimes we had doubts that need to be solved just at the moment"	forum quest
Complexity of learning flow not appropriate for students experience	- "Did peers with the same shape receive a different statement? Or... is there any mistake?" - "I think groups must not change during the course"	forum quest

One of the main conclusions is that supporting technology was not mature enough. Tools supporting communication in collaborative environments are mainly used in pilot programs. Very few experiences where a complex model is deployed in a real-life course are found in the literature, and therefore, there is no best practice guide to face the difficulties that appeared at runtime. Participants ranked an average of 2.20 (in a scale of 0 to 5) the technological support in the course. This result is mainly due to the lack of robustness of the tool prototypes. Nevertheless, several participants recognized the added value and potential of the presented technical solution for supporting distance collaboration. For instance, one of the participants said: *"I think it is a very useful way of working with participants who are not located in the same place, in spite of the failures in the technical support"*.

Although the tools covered all the required functionality to support the course, they failed to address some basic usability issues. The result was an environment that is fully functional but not user-friendly enough for a real case. In practice, it took some time for students to get used to the interface. This had a special impact on the experience because of the time restrictions explained in Section 2. A training session addresses this problem. However, the most effective solution would be to use tools with intuitive

interfaces providing a low adoption threshold. In this sense, even though it was conceived as part of the architecture, participants did not perceive the various distributed applications as an integrated solution (an average 1.58 in a 0-5 scale). Additionally, some opinions pointed out this low level of integration as a source of difficulties. For example, one of the participants stated that “...you did not have the freedom to go from one application to another and, furthermore, you perceived them as something external...”.

Having students from three different institutions increased the existing gap between profiles. Students' initial skills in the course topic were significantly different. Some participants argued that the course level was too high, while others performed the activities easily. Overall course difficulty was ranked with an average of 3.40 (in a 0-5 scale). Far from being transparent to students, this fact had a negative impact on their motivation. By appropriately measuring the gap between students, peer-reviewing groups can be reconsidered in order to optimize learning results, as shown in [3].

5. Conclusions

The design and deployment of a complex collaborative learning model in a distant scenario has been presented. Regarding required computer support, two main aspects have been discussed: computational representation of the model and the effective use of collaborative tools.

Learning Design has been used for computational representation. It has been shown how adaptability and flexibility provided by the specification can be used to consider different student profiles and also for grouping purposes. However, the difficulty of creating a course package with available tools hinders the task to non-expert users. Deployment of collaborative tools with the use of the VNC architecture has been discussed as an effective system to provide shared environments.

Experimental results show the potential of the deployment infrastructure: learning scripts used in conjunction with collaborative tools provide a framework where distant collaboration becomes possible even if different student profiles are involved in the course. However, the model highly depends on supporting tools performance. Low architecture robustness can derive in effectiveness reduction.

6. Acknowledgments

Work partially funded by *Programa Nacional de Tecnologías de la Información y de las Comunicaciones*, Project TSI2005-08225-C07-02/04

7. References

- [1] A. Chow and N. Law. "Measuring motivation in collaborative inquiry-based learning contexts" In *Proceedings of the 2005 Conference on Computer Support For Collaborative Learning: Learning 2005: the Next 10 Years!* (Taipei, Taiwan, May 30 - June 04, 2005)
- [2] L. Palomino-Ramírez, M.L. Bote-Lorenzo, J.I. Asensio-Pérez, L. de la Fuente-Valentín, and Y.A. Dimitriadis. "The Data Flow Problem in Learning Design: A Case Study", In *Proceedings of the International Conference on Networked Learning 2008* (May 5-6, 2008)
- [3] R. M. Crespo, A. Pardo and C. Delgado Kloos. "An Adaptive Strategy for Peer Review". *34th. Frontiers in Education Conference*, FIE 2004. Savannah, GA, EEUU. 20-23 October 2004.
- [4] J.A. Colliver (2000) "Effectiveness of problem-based learning curricula: research and theory." *Academic Medicine*. 75(3), pp. 259-66.
- [5] IMS, G. L. C. (2003). IMS Learning Design Information Model. v1.0, Technical Specification. www.imsglobal.org/learningdesign
- [6] D. Hernández Leo, D. Burgos, C. Tattersall, R. Koper, "Representing Computer-Supported Collaborative Learning macro-scripts using IMS Learning Design" *Second European Conference on Technology Enhanced Learning*, CEUR Workshop Proceedings, EC-TEL'07, Crete, Greece, September 2007.
- [7] P. Dillenbourg, P. Tchounikine (2007) Flexibility in macro-scripts for computer-supported collaborative learning. *Journal of Computer Assisted Learning* 23 (1), 1–13.
- [8] P. Jermann, P. Dillenbourg, "Elaborating new arguments through a CSCL scenario". *Arguing to Learn: Confronting Cognitions in Computer-Supported Collaborative Learning Environments*, 2003, p. 205-226 Amsterdam: Kluwer, 2003.
- [9] A. Weinberger; B. Ertl; F. Fischer; H. Mandl, "Epistemic and social scripts in computer-supported collaborative learning". *Instructional Science* 33(1), 1-30, Springer Verlag.
- [10] D. Hernández-Leo, J.I. Asensio-Pérez, Y. Dimitriadis, "Computational Representation of Collaborative Learning Flow Patterns Using IMS Learning Design" *Educational Technology & Society*. 8(4):75-89, October 2005.
- [11] A. Martínez, Y. Dimitriadis, B. Rubia, E. Gómez, P. de la Fuente, . "Combining qualitative evaluation and social network analysis for the study of classroom social interactions". *Computers & Education*, 41 (4), 353–368, 2003.
- [12] P. Dillenbourg. "Collaborative learning: cognitive and computational approaches". Oxford, UK: *Elsevier Science*.
- [13] G. Stahl, T. Koshmann, D. Suthers. "Computer-supported collaborative learning: a historical perspective". In R. K. Sawyer (Ed.), *Cambridge handbook of the learning sciences*, 2006, Cambridge, UK: Cambridge University Press.
- [14] X. Lu. "Construct Collaborative Distance Learning Environment with VNC Technology". In *Proceedings of the First international Conference on Semantics, Knowledge and Grid* (November 27 - 29, 2005).