

Monitoring pattern-based CSCL scripts: a case study

María Jesús Rodríguez-Triana, Alejandra Martínez-Monés, Juan Ignacio Asensio-Pérez, Iván Manuel Jorrín-Abellán, and Yannis Dimitriadis

GSIC-EMIC, University of Valladolid, Spain

{chus@gsic, amartine@infor, juaase@tel, ivanjo@pdg, yannis@tel}.uva.es

<http://www.gsic.uva.es>

Abstract. Two strategies have been proposed in CSCL to foster effective collaboration: structuring the learning scenario by means of collaboration scripts and monitoring interactions among participants in order to detect and regulate potential deviations from the initial plan. In order to help teachers in this endeavor, we propose to combine these two approaches by means of a process where design takes into account the especial requirements posed by monitoring, and monitoring is informed by the characteristics of the scripts that must be met to achieve the learning goals. These desired features are obtained from the constraints defined by the collaborative-learning flow patterns on which the scripts are based. The result is an automated and higher-level view about the evolution of the learning process, integrating the data gathered from the different tools. This paper also presents a case study based on an authentic experience in higher education where these ideas were tried out.

Keywords: CSCL, scripting, monitoring, Collaborative Learning Flow Patterns

1 Introduction

Computer-Supported Collaborative Learning (CSCL) scenarios are a clear example within TEL where Information and Communication Technologies (ICT) are employed in order to improve learning and teaching [17] [21]. Since in these scenarios learning occurs largely through interactions between participants, promoting effective collaboration is one of the main concerns [3]. *Scripting* and *monitoring* are two techniques long discussed in the research community aimed to foster effective collaborative learning interactions [13]. On the one hand, CSCL scripting structures the learning scenario and provides students with a set of instructions that guide potentially fruitful collaboration; on the other hand, monitoring the collaboration facilitates the intervention in order to redirect the group work in a more productive direction.

Within the wide range of CSCL scripting techniques, we can find the category of *macro-scripts* [7] (scripts hereafter), which mainly describes how groups (and individual roles) should perform a set of interrelated activities [4]. Such descriptions can even be made computer-interpretable by means of Educational Modeling Languages such as IMS Learning Design [12]. Since modeling potentially effective CSCL scripts is a difficult task, the use of patterns that reflect best practices in structuring collaborative learning has proved to be helpful [6] [19] [23]. For instance, *Collaborative Learning Flow*

Patterns (CLFPs) are a particular type of CSCL scripting patterns that describe well-accepted ways of arranging activities in collaborative learning scenarios [11]. Among other elements, CLFPs describe what types of interaction among learners and teachers should occur within activities.

Even though design is one important factor in building effective collaborative learning [8], having a "plan" is not enough to ensure that desired collaboration occurs: eventualities and contingencies require modifications of the course of the learning situation, leading teachers to adapt the original design [3]. At this point, monitoring the learning scenario and comparing its actual and expected states may provide useful information that the teacher can use to regulate collaboration [20].

In these scenarios collaboration takes place, at least in part, through computers. Analyzing these computer-mediated interactions may help to better understand collaboration [20]. This has been a strong trend in CSCL in the last years, but it has mostly focused on detailed interaction-analysis approaches aimed at letting researchers gain insight into the collaborative processes [14]. However, teachers need a more abstract idea of collaboration, easier to interpret so that they can react on time if needed. Additionally, the use of diverse ICT tools in the classroom would oblige teachers to integrate data from all the tools and environments where the learning process takes place. This is time-consuming and error-prone, if not unfeasible.

This paper proposes a method to get an automated and higher level view about the evolution of the learning process by combining the two mentioned approaches, i.e., structuring by means of pattern based scripting (to be more precise, supported by CLFPs), and monitoring collaboration. We propose to use the knowledge about the expected collaboration provided by the script, together with the data provided by the different sources in the technological infrastructure used for the enactment (VLE, Web 2.0 tools, etc.). We hypothesize that monitoring at a coarse level using these data is enough to provide teachers with relevant information about the current state of the learning process, helping them to react in time to redirect the course of the learning situation.

We present a pilot study based on data from an experience in higher education where these ideas were tried out. While the students collaborated following a script, we analyzed the data obtained from log-files and compared them with the expected collaboration as defined by the script. With these results, we provided teachers with visualizations of the process that helped them to understand and detect situations where actions needed to be taken (e.g. solving technical problems, extending the deadlines or modifying the distribution of the groups). The results were positive and set the lines for future research plans.

The remaining of this paper is structured as follows: section 2 introduces the proposal; section 3 is devoted to explain the case study (description, life-cycle and findings); finally conclusions and future work are summarized in section 4.

2 Combining scripting and monitoring to enhance collaboration

As mentioned before, the work presented in this paper combines both scripting and monitoring during the whole CSCL learning lifecycle to enhance collaboration. This

section describes the role that scripting and monitoring play in the proposal. The ideas are illustrated by means of an example.

2.1 The approach throughout the CSCL learning lifecycle

The process proposed in this work spans different phases of the CSCL learning lifecycle [9] (see Figure 1): the preparation of the learning scenario (design); the deployment/contextualization of the designed activities to address the concrete tool instances, participants and groups that will participate in their execution (particularization and instantiation); the execution of the activities themselves (enactment) and, eventually, the revision and refinement of those activities (evaluation).

First of all, the **design** must take into account the especial requirements posed by monitoring, in a similar way to existing proposals in which other tasks, such as assessment or evaluation are considered in advance [18]. As a consequence, some decisions must be taken explicitly, together with the rest of the learning process before the learning situation is implemented: defining how collaboration is expected to happen, choosing suitable tools for these collaboration purposes, identifying the key collaborative aspects and planning how they will be monitored. The result of this phase is what we have called the *monitorable collaboration schema*. It contains pointers to those phases of the script that have to occur in order to obtain its expected benefits.

During **enactment**, *actual monitoring results* are obtained from those sources that offer relevant information for collaboration analysis purposes (e.g., looking for evidence of expected interaction during a discussion activity in the logging data of a chat tool).

Finally, during **evaluation**, the *actual monitoring results* are contrasted with the *monitorable collaboration schema*, obtaining graphical visualizations of the progress of the learning situation, with the aim of helping teachers to detect potentially critical situations where actions must be taken.

The key issue in this process lies in the identification of the script constraints since this information guides the whole process. Next section deals with this issue in depth.

2.2 Pattern-based guidance of the monitoring of the collaborative scripts

The design phase of the process described in the previous section is to be informed by an analysis of the collaboration scripts. During this phase, authoring tools will guide designers to identify those aspects of the script that must be accomplished in order to achieve the learning goals. Several authors [5] [2] [15] have analyzed scripts in order to identify which features are modifiable (extrinsic constraints) and which ones have to be accomplished in order to maintain their pedagogical intentions (intrinsic constraints). In this section, we focus on determining which collaborative interactions should take place in order not to compromise the fulfillment of the intrinsic constraints of the script.

As mentioned before, the scripts considered in this work are based on a specific type of patterns, the CLFPs. For this reason, the requirements that the script must satisfy in order to achieve the learning objectives derive from the *pattern constraints*. These *constraints* are the conditions that should be met in a learning situation to be considered an implementation of such a pattern. Attending to the classification of the structural

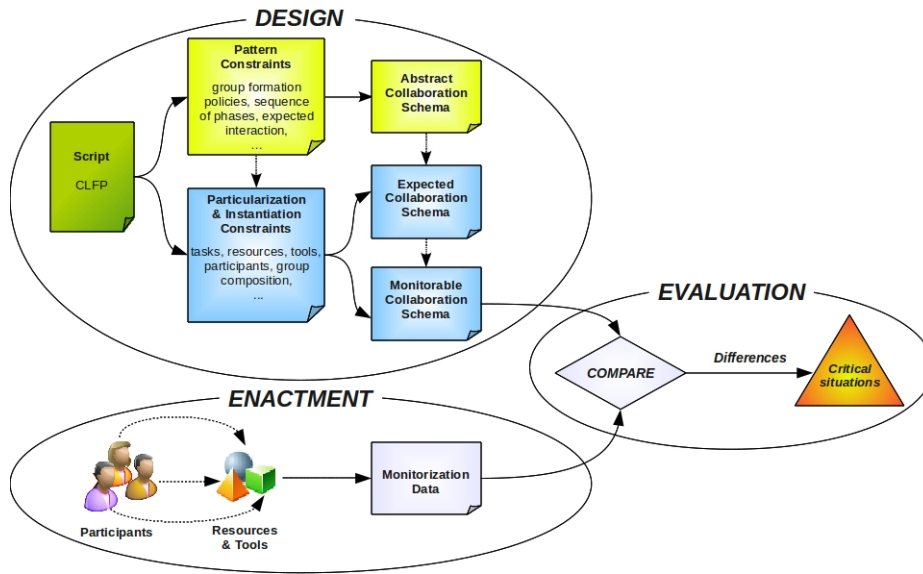


Fig. 1. Overview of the approach: associated tasks and expected results.

components of the scripts [3] [16], these constraints are represented by the group formation policies, the sequence of phases, the expected interactions, etc. The rest of the constraints of a script are the result of the *particularization and instantiation* of the pattern to each specific context [24], so they are modeled by means of elements like concrete tasks, resources, tools, participants, group composition, etc.

The determination of the script constraints has significant implications for the monitoring of the learning situation. First, by means of the pattern constraints it is possible to extract the heuristics that define the expected sequence of phases, as well as some collaboration guidelines and the rules for structuring groups. These heuristics constitute what we have called an *abstract collaboration schema*. Secondly, particularization and instantiation constraints provide the description of the context and the details about how such collaboration should happen (face to face, by means of a collaborative tool or a shared resource, synchronous or asynchronous...). This information is used to build the *expected collaboration schema*. Additionally, it is possible to establish which collaboration may be computationally monitored depending on the expected interaction and the computer support (e.g. which resources or tools provide history data), obtaining with it the *monitorable collaboration schema*. During the enactment of the learning situation, the monitorable collaboration schema guides the data gathering, focusing the process on obtaining evidence about the expected collaboration.

Summarizing, by means of the analysis of the CLFPs, we expect to support CSCL practitioners during the design process in the decision-making and the identification of the key aspects of collaboration. The analysis of two CLFPs, *Brainstorming* and *Pyramid* [10], is presented in sections 2.3 and 3.2. The result of this process, the monitorable

collaboration schema, will guide the data gathering during the enactment of the learning situation, focusing on monitoring those key aspects of the script.

2.3 An illustrative example

This section illustrates the proposal with an example. Let us suppose that a teacher is designing a collaborative learning situation that implements a *Brainstorming* CLFP [10]. In this activity, students, working *in pairs*, have to collect the relevant keywords related to a lesson, putting them altogether in a list. Then, they have to review the list and draw a concept map using the terms included in the list. The activity lasts one week, combining *face-to-face* sessions with *distance* work mediated by several ICT tools. Participants will have at their disposal a shared document (*Google Document*¹), to elaborate the list, and a shared board (*Dabbleboard*²), to draw the concept map.

This description of the learning situation contains some issues that are key to achieve the learning objectives: there are two phases, the brainstorming and the review, and intra-group collaboration is expected in both of them. Such issues, are represented by the *abstract collaboration schema* depicted in Figure 2 (step 1).

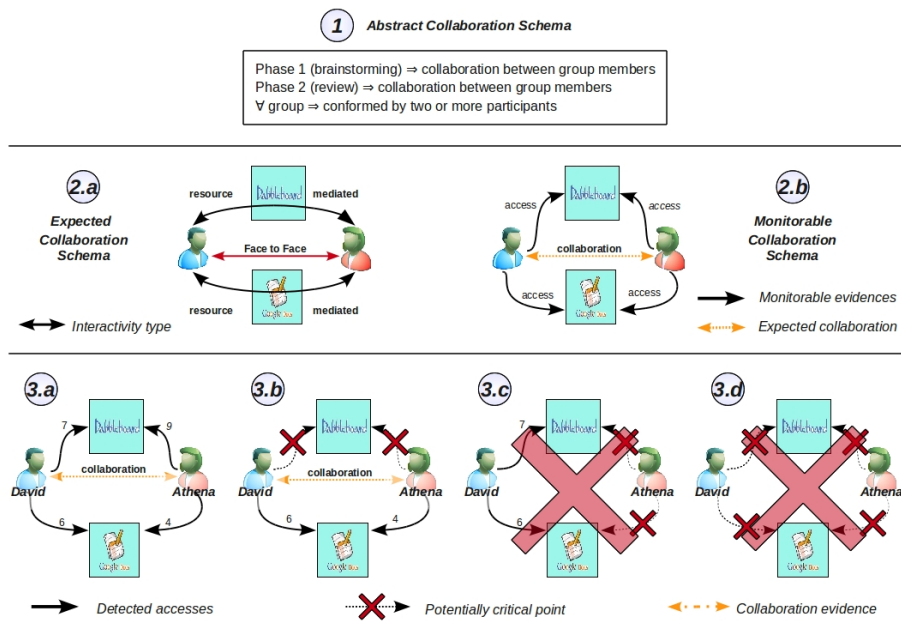


Fig. 2. Collaboration schemas and potential monitored scenarios.

Taking into account that the expected collaboration from students during the learning situation is to take place both face to face and through the shared resources (Google-

¹ <http://docs.google.com/>

² <http://www.dabbleboard.com/>

Docs and Dabbleboard instances), the heuristics obtained in the first step appear particularized in the *expected collaboration schema* (step 2.a of Figure 2). Due to the lack of observers during the enactment, face to face interactions are not monitored; nevertheless, the software used in the learning scenario registers when users access to the shared resources. Based on this, the evidences that are used to detect collaboration are shown in the *monitorable collaboration schema* (step 2.b of Figure 2).

The third level of Figure 2 presents four hypothetical situations that might happen during the enactment of the script. A number appears next to each arrow, representing how many times the user has accessed to each resource and, in that cases where there is no evidence of collaboration, it has been pointed out by a red cross over a dotted line. The first case (3.a), shows a scenario where both members of the dyad have accessed the shared resources, so it seems that they have co-worked as it was expected. The second one (3.b), exemplifies a situation where students have not worked with all the resources they have to. Despite there is evidence of collaboration, the second phase of the learning situation does not seem to have been carried out yet, so the teacher may consider to intervene. The last two examples (3.c and 3.d) illustrate two potentially critical situations where there is no collaboration evidence: the former shows a learning scenario that seems to have been developed by just one participant; in the latter, the activities probably are not being carried out by any of them.

Using this information, and enriching it with their knowledge about face-to-face sessions, teachers should be aware of the participants' progress during the course of the activity, and be able to react on time to undesired situations.

3 Case Study: monitoring an authentic learning scenario

In this paper, the authors address one main research question: "does the combination of pattern based scripting and monitoring collaboration contribute to provide useful information about the evolution of the learning process?" In our attempt to understand in depth this research question, an instrumental case study [22] was carried out on an authentic CSCL scenario. The authors followed the presented approach in a learning experience in which a CLFP-based script was applied. First, the teacher was guided during the design process and, later on, participants' activity was monitored during the enactment of the experience.

This section is structured as follows: first, we present briefly the main characteristics of the case; then, following the steps presented in the approach, we explain how the design guided the monitoring process, as well as some examples of how some undesired collaboration situations were detected; and finally, the section ends with the discussion of the results obtained from the case study.

3.1 Context and methodologies of the study

The case study presented here was developed in November 2010, and took place within a third year course (out of five) on "Network traffic and Management" of Telecommunications Engineering degree, at the University of Valladolid (Spain), with 46 students attending the course. During this course, students had to develop a chat tool using data

network protocols. In order to help them to plan and anticipate problems for a subsequent programming assignment, they were asked to elaborate a sequence diagram of their software design. To elaborate this diagram, students worked in a blended CSCL situation, interleaving *face-to-face* with *distance activities mediated by ICT tools*.

The collaboration script implemented a two-level *Pyramid CLFP* [10]. At level-1, *groups* of 2 participants attended to a face-to-face lab session to carry out the first activity. In this activity, students had to draw a preliminary version of the sequence diagram and write a report with a summary of the main decisions and open issues. At level-2, groups joined to conform *super-groups* (composed of 4 groups) that had to accomplish both a distance and a face-to-face activity. During the former, each *group* had to review and provide feedback on the reports produced by their *super-group* mates; in the latter, they had to discuss and produce a joint version of the diagram, as well as perform an oral presentation with a common version of the conclusions and open issues.

Regarding the technological support, teachers used the VLE *Moodle*³ to centralize the access to all the resources and activities. To accomplish the drawing tasks, students were provided with a shared board (*Dabbleboard*), and in order to explain, review and discuss, they had at their disposal shared documents and presentations (*Google Documents* and *Google Presentations*). Since these tools can not be automatically integrated in Moodle, the *GLUE!* architecture [1] was used to integrate them into the VLE.

With this case study we aimed at illuminating three issues that would help us to gain insight into our research question. These issues were: "does the CLFP description help to identify the constraints of the collaboration script?", "does the comparison between desired an actual collaboration provide teachers with relevant information about the current state of the learning process?" and, in a learning scenario characterized by the integration of ICT tools, "which are the required conditions for collecting relevant information about the key collaborative aspects?".

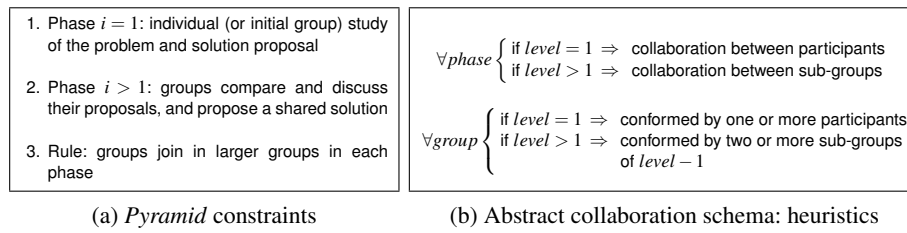
Our results were triangulated with data coming from observations of the face-to-face activities; a questionnaire that was handed to the students at the end of the experience; a group interview with students that volunteered to participate at the end of the activity, and an interview to the teachers.

3.2 Collaboration schemas of the proposed script

According to the process proposed in section 2.1, the teacher is guided during the design to define two schemas, namely: the expected collaboration schema, which defines how collaboration should occur during the enactment, and the monitorable collaboration schema, which describes the evidence of that collaboration that can be gathered from the technological environment. It is a long-term goal to support the designer in these tasks so, for this pilot study, the authors themselves defined these schemas.

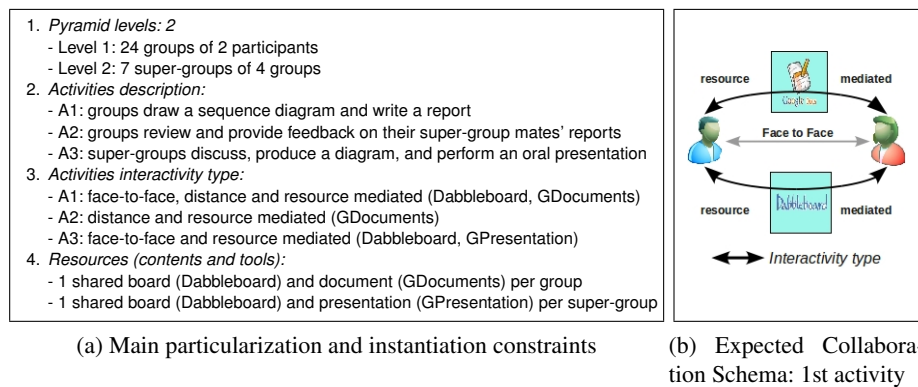
First of all we will go through the pattern constraints. The *Pyramid CLFP* defines a collaborative learning flow for a context in which several participants face the same problem, whose resolution implies the achievement of gradual consensus [10]. Therefore, the pattern defines a sequence of at least two phases with some particular collaboration objectives, as well as a rule for structuring groups (see Figure 3a).

³ <http://www.moodle.org>

**Fig. 3.** Pattern analysis

These pattern constraints are translated into the heuristics shown in Figure 3b, which constitute what we have called *abstract collaboration schema*. This information will allow us to establish connections between those students or groups that are expected to collaborate in each phase of the *Pyramid*.

Secondly, to particularize and instantiate the pattern to the specific scenario, several details were fixed: the levels of the *Pyramid*, the definition of learning objectives, the set of activities and the interactivity type (e.g. face-to-face, distance or resource mediated), the provision of necessary resources (contents and tools), and decisions about completion of activities (e.g. control of time), as well as the creation of actual groups and the binding of participants to the groups of each level of the *Pyramid*, etc. The values of the most relevant constraints are summarized in Figure 4a.

**Fig. 4.** Particularization and instantiation analysis

While the pattern constraints define how to establish collaborative connections (abstract collaboration schema), the constraints derived from the particularization and instantiation let us draw the *expected collaboration schema*. This schema, as Figure 4b shows, specifies for each activity how the collaboration should occur (face-to-face, distance or both) and by means of which resources (shared documents and boards). However, not all kinds of collaboration are computationally monitorable. That is why, the next step deals with the analysis of the capabilities of the ICT tools involved experience to produce data that can be used for monitoring collaboration.

As mentioned in section 3.1, the software used to support the aforementioned learning scenario included a virtual learning environment (Moodle), a number of external Web 2.0 tools (Dabbleboard, Google Documents and Google Presentations), and an

implementation of the GLUE! architecture for the integration of external tools into the VLE. Table 1 describes both the kind of monitoring data and the way in which these tools provide access to this information.

Software	User activity information	Retrievable from
Dabbleboard	None	Not possible
Google Tools	Document revision history: <i>user, date, time and document version</i>	User interface
Moodle	Event history: <i>date, time, IP address, user name, action, resource used</i>	User interface or database
GLUE!	History of accesses to the integrated resources <i>user, date, time, resource</i>	Event logs

Table 1. Description of the software used in the case study in terms of (1) the information provided about the user activity and (2) the way this information is accessible.

Table 1 shows the high heterogeneity of both the data provided and the harvesting methods: for example, Dabbleboard does not provide any kind of user activity data and Google's tools do not offer a programmatic interface to obtain this information. Regarding Moodle, it would be necessary to query the database manually in order to complete the history information offered by the VLE. Taking into account that GLUE! offers largely the same information that would be obtained from Moodle, we decided to use GLUE! as the sole source of activity data. Since GLUE! registers every access to the instances of the integrated tools (Dabbleboard and Google suite in this case), the computational monitoring of the collaboration, in this case study, is based on the evidence of instance accesses provided by GLUE!.

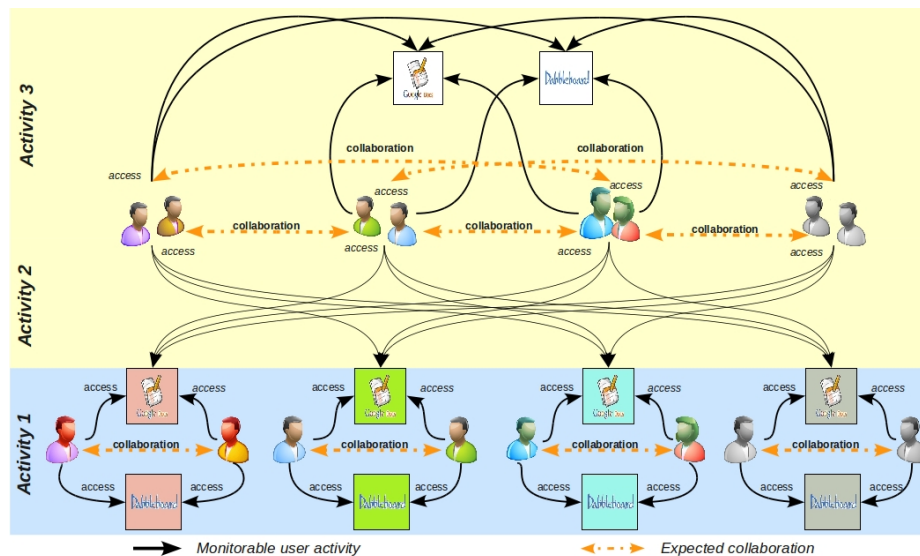


Fig. 5. Monitorable collaboration schema of the experience studied in this case.

Figure 5 summarizes the outputs of all the previous mentioned steps. The collaboration heuristics that conform the abstract collaboration schema are represented by means of the dashed yellow arrows, highlighting those key collaborative aspects to be monitored. The specific sequence of activities, the resources required for each one of them, and the way the students are grouped are derived from the particularization and instantiation constraints (see Table 4a). Finally, among all the expected ways of collaboration (expected collaboration schema), continuous black arrows point to the information that may be collected in order to detect collaboration evidence. This view of the learning scenario depicts what we have called *monitorable collaboration schema*.

3.3 Comparing monitorization data against the collaboration schema

During the enactment phase of the learning experience, we retrieved and interpreted the content of GLUE!’s logs. The analysis of the logs focused on detecting evidence from the key collaborative aspects highlighted in the abstract collaboration schema (dashed black arrows in Figure 5) by means of the accesses to the shared resources (continuous black arrows in Figure 5). Figure 6 displays the actual collaboration of one super-group. The labels on the links specify how many times each participant (or group) accessed the resource. Red crosses over a dotted line represent those cases where no evidence of collaboration could be deduced from the collected data.

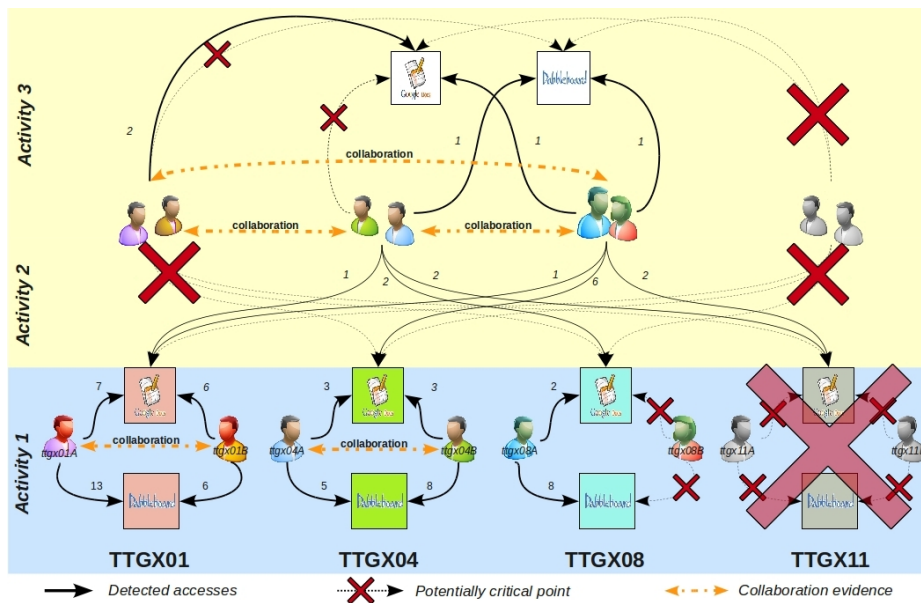


Fig. 6. Monitoring report of one super-groups that participated in the case study.

By means of this information, we detected some potentially critical situations. For instance, during the first activity, two potentially undesired situations were identified.

As shown in Figure 6, in group TTGX08 just one of the group members interacted with the resources; and in TTGX11, none of the participants had accessed the resources, which was a pointer to a potential undesired situation.

Teachers were informed periodically about the potentially undesired situations. Being aware of them, they reviewed carefully the resources of those groups that showed irregularities to confirm (or not) our assumptions. In case of confirmation, teachers took preventive measures, for instance, by sending to the students a reminder of the deadlines or asking them whether they had found any problem during process to finish the activity.

From the researchers' perspective, in order to validate whether our assumptions based on the monitorable data were correct, we triangulated these results with the rest of the data sources. We found out that in those cases where monitorable data showed expected behaviour, participants had actually collaborated following the script (as extracted from the interviews with both teachers and students). Furthermore, all the cases of collaboration breakdown had been detected based on the monitorable data. However some cases that had been interpreted as potentially critical situations were indeed "false positives". These were cases where learners collaborated in unexpected ways. For example, from the observations and questionnaires we could see that students interacted face to face when they were expected to collaborate by means of the ICT resources. Despite this limitation, teachers stated in the interview that this was a minor problem, since it had been easy for them to confirm or disprove the initial assumptions, based on their own knowledge of the classroom dynamics from their informal observations.

3.4 Findings and discussion

As mentioned in the Introduction, the main goal of the case study was to get evidence on whether the combination of structuring (by means of pattern based scripting) and monitoring collaboration contributes to provide an automated and higher level view about the evolution of the learning process. For achieving that goal we proposed three research issues (see section 3.1). We discuss the results according to them.

The first issue considers the usefulness of CLFPs for identifying the key collaborative aspects of the script. Results indicated that the analysis of the *Pyramid* constraints (abstract collaboration schema) has eased the identification of those constraints of the script that require accomplishment (expected collaboration schema). Indeed, with the information provided about the ICT tools involved in the learning scenario, it has been possible to identify which collaboration evidence could be obtained in order to inform about the key aspects of the script (monitorable collaboration schema). And finally, this schema guided the monitoring process to focus on those aspects of the expected interaction that could jeopardize the script collaborative purposes. It must be highlighted, that this analysis of the *Pyramid* pattern could be applied in different contexts. For this reason, we expect to carry out this analysis for each CLFP in order to guide designers automatically by means of authoring tools.

The second issue referred to whether the comparison between the desired and actual collaboration helps teachers to be aware of the progress of the learning scenario. The evidence points out that it is feasible to detect and inform teachers about potentially critical situations, facilitating the adaptation tasks. The proposed way of visualizing the learning process has been a fruitful first attempt to provide teachers with feedback. In

this paper we have presented a prototypical solution that has served to test the proposal. The positive results obtained from the study lead us to plan future work, dealing with the automatic generation of the collaboration models for each CLFP.

The third and last issue is related to technological restrictions that appear in learning scenarios characterized by the use of diverse ICT tools, i.e., which are the required conditions for collecting relevant information about the key aspects of collaboration in these settings. We have shown that, just by monitoring the access to the tool instances, it has been possible to offer a general view about the collaborative activities in this specific scenario. With this simple information, we have given response to a very common problem: the lack of monitoring data in some ICT tools [18] (as it happened with Dabbleboard). As shown in the case, this information can still be enriched, including for example the data collected from each individual tool or even with interactions registered by means of face-to-face observations. Thus, one of the main issues in our upcoming work will delve into the problem of data integration.

Finally, we collected some impressions and feedback from both students and teachers. From the teachers point of view, they argued that, despite the limitations of the reduced variety of monitored events, results were pretty close to the real facts. By means of the monitored schemas developed for each super-group, teachers could realize at a glance which participants seemed not to be working, which ones were isolated and which was the best way to re-distribute them. Moreover, this information prevented teachers from going through all the 64 instances to check the activities progress, saving them a long time. On the other hand, students complained about the lack of information about their mates work, which forced them to connect and review their resources frequently to check whether there was any change. Therefore, this points to the potential usefulness of the monitoring results to facilitate students collaboration.

These findings provide evidences on the capabilities of the presented approach to generate relevant information about the evolution of the learning process. This information has had educational value, especially regarding the improvement of the regulation tasks of CSCL activities in real practice.

4 Conclusions and future work

This paper proposed an approach where CLFP-based scripting and monitoring methods and tools are combined to enhance collaboration. We have shown how the analysis of pattern constraints helps to identify those aspects of collaboration that should be looked after when monitoring the activity. This way, monitoring can focus on these aspects, which are easier to detect and to follow than fine-grained analysis of the activities. Finally, by comparing the actual and desired collaboration schemas, teachers can identify undesired situations and regulate the collaborative process.

This proposal has been put in practice through a case study, carried out in a higher education learning scenario, that has shown initial evidence on the suitability of the approach for the detection of potentially critical situations. The approach has been followed manually so next steps will deal with the process automation. The analysis of other CLFPs constrains to guide designers during the process, and the visualizations of the collaboration schemas constitute two main lines of future work.

These results could be helpful for facilitating the regulation tasks. For instance, the work done in [5] [2] and [15] analyzing the scripts flexibility and the ways to adapt them when eventualities appear, could be simplified by our proposal. Once teachers know which constraints are not being satisfied in real time, it will be easier for them to address the issues hampering collaboration (e.g. solving technical problems, extending the deadlines or modifying the student distribution). This benefit is even more remarkable using heterogeneous scenarios where several ITC tools are involved.

This work is connected to the effort towards the integration of VLE and external ICT tools in [1]. The use of an architecture that integrates all the software used to support the learning scenario, simplifies data gathering and provides relevant high level information about the progress of the learning activities. This trend underlines the need of developing architectures for the retrieval of monitoring data from decentralized learning environments. However, the integration of different data sources is a complex task due to the heterogeneity of their content [18]. These two pending issues, gathering data from different data sources and their integration, constitute other lines of future work.

Acknowledgements

This research has been partially funded by the Spanish Ministry of Science and Innovation Projects TIN2008-03-23 and IPT-430000-2010-054, and the Autonomous Government of Castilla and León, Spain, Project VA293A11-2. The authors would also like to thank the rest of GSIC/EMIC Group at the University of Valladolid for their support and ideas, and all the students who participated in the evaluation process.

References

1. Alario-Hoyos, C., Wilson, S.: Comparison of the main Alternatives to the Integration of External Tools in different Platforms. In: Proceedings of the International Conference of Education, Research and Innovation. pp. 3466–3476. ICERI'10, Madrid, Spain (2010)
2. Demetriadis, S., Karakostas, A.: Adaptive Collaboration Scripting: A Conceptual Framework and a Design Case Study. In: Proceedings of the 2nd International Conference on Complex, Intelligent and Software Intensive Systems. pp. 487–492. CISIS'08, IEEE Computer Society, Washington, DC, USA (2008)
3. Dillenbourg, P.: Three Worlds of CSCL. Can We Support CSCL?, chap. Over-Scripting CSCL: the risks of blending collaborative learning with instructional design, pp. 61–91. Open Universiteit Nederland, Heerlen (2002)
4. Dillenbourg, P., Jermann, P.: Scripting computer-supported communication of knowledge – cognitive, computational, and educational perspectives, chap. SWISH: A model for designing CSCL scripts, pp. 1–10. Springer, New York (2007)
5. Dillenbourg, P., Tchounikine, P.: Flexibility in macro-scripts for CSCL. *Journal of Computer Assisted Learning* 23(1), 1–13 (2007)
6. E-LEN: Design expertise for e-learning centers (2004), project website, <http://www2.tisip.no/E-LEN/>. Last visit: April 2011
7. Fischer, F., Kollar, I., Haake, J., Mandl, H.: Scripting computer-supported communication of knowledge – cognitive, computational, and educational perspectives, chap. Perspectives on collaboration scripts, pp. 1–10. Springer, New York (2007)

8. Goodyear, P.: Educational design and networked learning: patterns, pattern languages and design practice. *Australasian Journal of Educational Technology* 21(1), 82–101 (2005)
9. Gómez Sánchez, E., Bote Lorenzo, M., Jorrín Abellán, I., Vega Gorgojo, G., Asensio Pérez, J., Dimitriadis, Y.: Conceptual framework for design, technological support and evaluation of collaborative learning. *International Journal of Engineering Education*. 25(3), 557–568 (2009)
10. Hernández-Leo, D., Villasclaras-Fernández, E., Asensio-Pérez, J., Dimitriadis, Y.: Handbook of Visual Languages for Instructional Design: Theories and Practices, chap. Diagrams of learning flow patterns' solutions as visual representations of refinable IMS Learning Design templates. IGI Global (2007)
11. Hernández-Leo, D., Villasclaras-Fernández, E., Asensio-Pérez, J., Dimitriadis, Y., Retalis, S.: CSCL Scripting Patterns: Hierarchical Relationships and Applicability. In: Proceedings of the 6th IEEE International Conference on Advanced Learning Technologies. pp. 388–392. ICALT, Kerkrade, The Netherlands (2006)
12. IMS Global Learning Consortium: IMS Learning Design Specifications (2003), retrieved from <http://www.msglobal.org/learningdesign/>. Last visit: April 2011
13. Jermann, P., Soller, A., Lesgold, A.: Computer Software Support for CSCL. In: Strijbos, J.W., Kirschner, P., Martens, R. (eds.) What we know about CSCL ... and implementing it in higher education, *Computer-Supported Collaborative Learning*, vol. 3, pp. 141–166. Kluwer Academic Publishers, Norwell, MA, USA (2004)
14. Kahrimanis, G., Avouris, N., Komis, V.: Technology-Enhanced Systems and Tools for Collaborative Learning Scaffolding, chap. Interaction analysis as a tool for supporting collaboration. An overview. Springer (2011)
15. Karakostas, A., Demetriadis, S.: Adaptation patterns in systems for scripted collaboration. In: Proceedings of the 8th International Conference on Computer Supported Collaborative Learning. CSCL'09, vol. 1, pp. 477–481. International Society of the Learning Sciences, Rhodes, Greece (2009)
16. Kobbe, L.: Framework on multiple goal dimensions for computer-supported scripts (2005), Kaleidoscope Noe. Deliverable D29.2.1
17. Koschmann, T.: CSCL: Theory and Practice of an Emerging Paradigm, chap. Paradigms Shift and Instructional Technology, pp. 1–23. Lawrence Erlbaum Associates, Mahwah, NJ USA (1996)
18. Martínez-Monés, A., Dimitriadis, Y., Harrer, A.: Analyzing Collaborative Interactions in CSCL: Methods, Approaches and Issues, chap. An interaction aware design process for the integration of interaction analysis in mainstream CSCL practices, pp. 269–291. Springer Verlag (2011)
19. PPP: The Pedagogical Patterns Project (2005), project website, <http://www.pedagogicalpatterns.org/>. Last visit: April 2011
20. Soller, A., Martínez, A., Jermann, P., Muehlenbrock, M.: From Mirroring to Guiding: a review of the state of the art in interaction analysis. *International Journal on Artificial Intelligence in Education* 15, 261–290 (2005)
21. Stahl, G., Koschmann, T., Suthers, D.: Cambridge handbook of the learning sciences, chap. Computer-supported collaborative learning: an historical perspective., pp. 409–426. Cambridge University Press, Cambridge, UK (2006)
22. Stake, R.: The art of Case Study Research. Sage Publications, Londong (1995)
23. TELL: Towards Effective networked supported coLLaborative activities (2005), project website, <http://cosy.ted.unipi.gr/tell/>. Last visit: April 2011
24. Villasclaras-Fernández, E., Hernández-Gonzalo, J., Hernández-Leo, D., Asensio-Pérez, J., Dimitriadis, Y., Martínez-Monés, A.: InstanceCollage: A Tool for the Particularization of Collaborative IMS-LD Scripts. *New Directions in Advanced Learning Technologies. Educational Technology & Society* 12(4), 56–70 (2009)