Towards a Script-Aware Monitoring Process of Computer-Supported Collaborative Learning Scenarios

María Jesús Rodríguez-Triana, Alejandra Martínez-Monés, Juan I. Asensio-Pérez, and Yannis Dimitriadis

GSIC-EMIC, University of Valladolid, Spain
(chus@gsic, amartine@infor, juaase@tel, yannis@tel).uva.es
http://www.gsic.uva.es

Abstract. The increasing complexity of CSCL scenarios makes the classroom management highly demanding. Teachers invest considerable effort to design the learning scenario and to be aware of what happens during the enactment. We hypothesize that providing monitoring information closely related to the teachers' pedagogical intentions will help them to understand the unfolding of the learning situation, empowering them to intervene. This article presents a script-aware monitoring process which uses the knowledge about the learning design to guide the analysis of the educational data generated throughout the learning processes. The proposal is illustrated by an example based on a real CSCL scenario in an university course. This example shows how the script-aware monitoring provides useful feedback for the teacher and reduces the effort devoted to management tasks.

Keywords. Computer-Supported Collaborative Learning (CSCL), Learning Analytics, Learning Design, monitoring, scripting.

1. Introduction

Putting in practice Computer-Supported Collaborative Learning (CSCL) scenarios is usually a non-trivial task (Dillenbourg et al., 2009). Teachers have to plan a priori the activities to be carried out, organize students into groups, and prepare the technological environment that will support the learning process. Afterwards, during the enactment, teachers need to supervise the learning process, going through different learning resources and tools, looking for evidence of the student's work. Both the design and the management of the learning process are even more challenging when using Distributed Learning Environments (DLEs) (MacNeil and Kraan, 2010). DLEs integrate learning platforms with external tools, and are becoming increasingly popular in current pedagogical practice.

A concept related to Learning Design refers to CSCL scripting, which aims to provide students with a set of instructions that guide them towards potentially fruitful collaboration (at least partially) mediated by computers (Kobbe et al., 2007). Our focus in this paper is on macro-scripts, as opposed to micro-scripts, which are characterized by their coarse granularity and their emphasis on the orchestration of activities (Dillenbourg and Jermann, 2007). Hereafter we will refer to macro-scripts simply as scripts.

Once the learning process has been defined, the next step is to understand how such process unfolds. The emerging research field of Learning Analytics may provide an insight into that question. The Society for Learning Analytics Research (SoLAR[1]) refers to Learning Analytics as *“the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments in which it occurs”*. One way of applying Learning Analytics could be to monitor the learning process, and use the information collected to analyse and report data about the learners to the teachers, so that they can identify unexpected situations, and react to them on time.

However, teachers may not be able to make sense of these analyses. They need meaningful information, connected to their pedagogical intentions (Sutherland et al., 2012). To face this problem, several authors have highlighted the potential synergies that may emerge with the alignment of Learning Design and Analytics. Lockyer and Dawson (2011) stated that this tandem offers the opportunity of better understand the student behaviour and provide pedagogical recommendations where deviations from pedagogical intention emerge. Looney and Siemens (2011) considered the learning plan as an efficient learner hypothesis that may be compared with the learner on-going activity for adaptation and personalization. In addition, Martínez-Monés et al. (2011) emphasized that the alignment between pedagogical and informational needs is crucial in order to integrate learning analytics into mainstream CSCL practices. A similar approach, aligning learning design and assessment techniques (e.g. e-portfolios), has been followed in the NEXT-TELL[2] project.

In this work, we focus on the integration of *scripting* and *monitoring* for the design and analysis of CSCL scenarios. This article presents the so called *script-aware monitoring process* where the script information guides the gathering and analysis of the educational data available in a learning scenario. We hypothesize that, focusing the analysis on the teacher’s pedagogical intentions may provide them with meaningful information for managing the learning situation.

The proposal is illustrated with an example extracted from a real CSCL scenario in higher education, in which the students collaborated following a script in a DLE conformed by a virtual learning environment (VLE) and external tools. We analysed the data obtained from log-files of the different elements of the DLE and teacher’s observations, and compared the obtained results with the script definition. We provided the teacher with visualizations of this comparison, which helped her to understand and detect situations which required corrective actions. This example illustrates some of the benefits that the proposal may bring for the teacher (e.g. reducing the effort required for monitoring the learning process, and facilitating the detection of potential problems).

The rest of this article is structured as follows: Section 2 introduces the overall approach of the proposal; Section 3 describes the script-aware monitoring process and Section 4 presents an illustrative scenario and discusses how the proposal maybe suitable for different scenarios; finally, Section 5 summarizes the conclusions and future work.

2. General Approach: Scripting and Monitoring Alignment in CSCL scenarios

According to the literature, the lifecycle of CSCL scripts goes through several phases. Though there is no widely-agreed consensus, they could be roughly summarized in the following ones (see Figure 1): the **design** of the learning scenario (Vignollet et al., 2008; Sobreira and Tchounikine, 2012); the **instantiation** of the designed activities to address the concrete tool instances, participants and groups that will be involved in the learning scenario (Dillenbourg and Tchounikine, 2007; Sobreira and Tchounikine, 2012); the execution of the activities themselves and their run-time **management** (Dillenbourg and Tchounikine, 2007; Sobreira and Tchounikine, 2012); and, eventually, the **evaluation** of those activities (Vignollet et al., 2008). Though this article deals with the management phase, in this section, we introduce our overall proposal, describing the connections between the design and management phases.

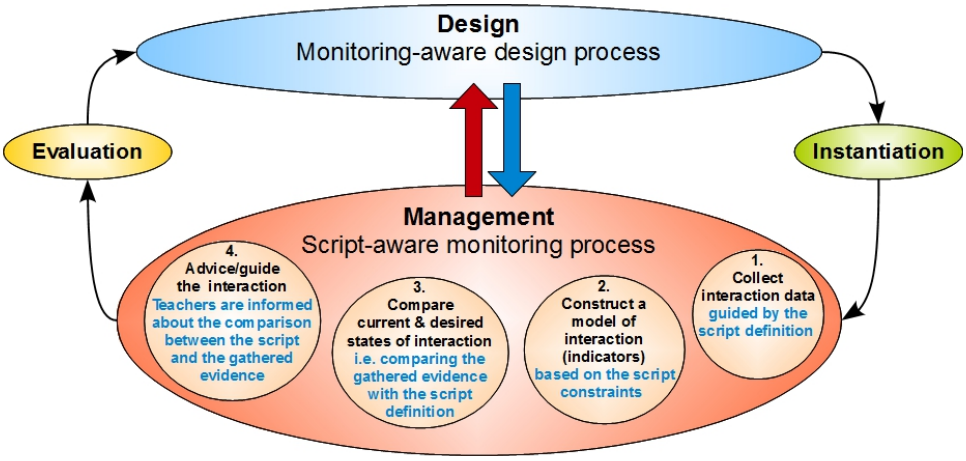

Figure 1 Alignment of scripting and monitoring techniques throughout the lifecycle of CSCL scripts

In our previous work, we have defined a **monitoring-aware design process** (Rodríguez-Triana et al., 2012a). This process aims to take monitoring into account within the learning design. Teachers are therefore prompted to consider those *constraints imposed by the design* (Dillenbourg and Tchounikine, 2007) that require special attention, and thus can be the focus of monitoring (e.g. constraints derived from the pedagogical patterns implemented by the script, or from the configuration of the activities); to *reflect on the decisions that affect monitoring* (such as the selection of ICT tools, or the selection of additional data sources that may inform about the students' work), and to *specify some details of the learning scenario necessary to set up the analysis* (see Table 1).

Table 1 Information gathered during the design process and used to guide the analysis of students' interactions

Pattern	Activity	Teacher's monitoring decisions
Activity flow Collaboration Group formation policies	Deadlines Resources (tools, contents) Participants Groups Social level Interactivity type Location Participation	Monitoring periods Interactions to be monitored Activities to be monitored Resources to be monitored Expected use of resources

In order to define the learning design information that could be relevant for monitoring purposes, we built on existing models of CSCL scripts (e.g. Kollar et al., 2006; Dillenbourg and Tchounikine, 2007; Kobbe et al., 2007), and on models for the analysis of computer-mediated interaction (Harrer et al., 2009; Miao et al., 2005). We selected elements that model CSCL scripts (participants, groups, roles, activities and resources) and elements that model computer-mediated interactions (participants, groups, roles, resources, and actions). With this information and the lessons learnt from previous experience in monitoring CSCL authentic scenarios, we defined the **monitoring-aware scripting model** (Rodríguez-Triana et al., 2012b). Table 1 shows the parameters considered in this model, classified in those related to the pedagogical pattern(s), to the definition of the activity, and to the teacher's monitoring decisions. These elements synthetize the main aspects that must be incorporated to a script that aims at providing monitoring capabilities when enacted.

The complementary part of the monitoring-aware design process in the proposed cycle is **the script-aware monitoring process** that defines the steps to carry out at the management phase (see Figure 1). Section 3 describes this monitoring process and Section 4 provides an example of its application in a real scenario.

3. Script-Aware Monitoring of CSCL Scenarios

This section presents a script-aware monitoring process that covers the management phase of the CSCL scripts lifecycle depicted in Figure 1. The key aspect of the proposal is the use of scripts designed by teachers to guide the data analysis of students' interactions. We hypothesize that the comparison between the script and the evidence gathered from the learning scenario may provide the teachers with information aligned with their pedagogical interests.

There are several proposals in the literature devoted to conceptualize the data analysis processes. Many of them define a data-driven approach obtaining indicators based on the data available and trying to extract meaning from them, e.g. Mitra et al. (2002), Dron and Anderson (2009) and Chatti et al. (2012). Others follow a model-driven approach in which the collected data are compared with a pre-specified model that guides the analysis. Since we try to relate the data gathered from the learning scenario with the teacher's pedagogical intentions predefined in the scripts, we focus on the second approach, concretely in the *collaboration analysis process* proposed by Soller et al. (2005).

Soller et al. identified five steps in the management of collaborative learning interactions: (1) the collection of interaction data, (2) the construction of an interaction model, (3) the comparison between the current and the desired state of interaction, (4) the advisement and/or guidance in case of discrepancies between the current state of interaction, and (5) the evaluation of interaction assessment and diagnosis. Connecting this framework with the script lifecycle presented in Section 2, the four first steps of the framework belong to the management phase, whereas the fifth corresponds to the evaluation phase.

We adopt the aforementioned collaboration analysis process as a conceptual framework for our proposal. The following subsections describe how the script-aware monitoring process implements the four management steps of the framework, using the script parameters listed in Table 1 to guide the data collection and build the interaction models.

3.1. Collect Interaction Data

Instead of gathering all the data available, in our proposal we select a priori the data to be included in the analysis. Table 2 defines a set of heuristics to relate low-level data (from now on "interactions") with the definition of the activities and the teacher's monitoring decisions (detailed in Table 1). To consider a participant's interaction relevant for the analysis of one *activity*, it should meet the following requirements: the interaction must be time-stamped within the activity *deadlines*, carried out in the *resources* that the teacher decided to monitor, classified as a type of *interaction to be monitored*, and performed by the activity *participants.* In addition, if participants or groups have to work with a specific subset of resources, only their interactions with these resources will be taken into account.

Table 2 Heuristics used to select the interactions considered in the analysis

An interaction is included in the analysis if:

- **The interaction happens within the activity deadlines:**
  {activity.begining >= interaction.timestamp <= activity.end}
- **The author(s) of the interaction belong(s) to the activity participants**
  {interaction.actor ⊆ activity.participants}
- **The author(s) of the interaction is(are) supposed to use the resource**
  {interaction.actor ⊆ resources.users}
- **The interaction involves a resource to be monitored during the activity**
  {interaction.resource ∈ activity.resources_to_be_monitored}
- **The type of interaction must be monitored in a given resource:**
  {interaction.type ∈ resource.interactions.to_be_monitored}

Though technology enhanced learning contexts such as CSCL scenarios offer the possibility to store and analyse large amounts of educational data (Siemens et al., 2011), there are some problems that hinder the data gathering and integration (Martínez-Monés et al., 2011). For example, some tools do not register any kind of data about the user's interactions; there is no standard format to store and model these data, so each tool/environment follows its own approach; and frequently, applications do not provide ready- to-use data (such as streamed data or low level interactions). These obstacles increase when the technological context is heterogeneous and decentralized, as it happens in DLEs, or when the data is generated not only automatically by the technology but also ad-hoc by the participants. Thus, the use of architectures that integrate the different data sources plays a crucial role. Three architectures that may be used to this end are: *Tin Can API*[3], *GLUE!-CAS* (Rodríguez-Triana et al., 2011) or *CAM* (Scheffel et al., 2010). The *Tin Can API* is a specification for capturing data about a person or group's activities from multiple technologies. This specification requires that each tool implements a REST service to send statements -in the form of "Noun, verb, object"- to a learning record store. *GLUE!-CAS* is an architecture for data gathering and integration in DLEs based on adapters (Gamma et al., 1995). Each VLE or tool requires a REST adapter that returns the users' interactions on demand. Users' interactions are modelled according to the "Common format" defined by the European Kaleidoscope NoE to enable interoperability among CSCL and collaboration analysis tools (Harrer et al., 2009). A similar approach is followed in the *CAM* solution to gather data from PLEs, where each tool must offer a REST interface to provide the user's data on demand, following the *CAM* format.

3.2. Construct a Model of Interaction

According to our proposal, the selection of the indicators used in the model of interaction is based on the parameters of the script. We have identified the following aspects needed to model the interaction: *participation* (involvement of an individual or group in the activity); c*ollaboration* (interactions among groups and/or group members); *use of resources (*interactions between participants and resources); *group formation policies* (requirements that groups should accomplish in terms of criteria such as size or type of participants); *activity flow dependencies* (activity parameters that affect other activities, e.g. reused resources, groups, or deadlines).

Table 3 and Table 4 present indicators related to some of the aforementioned aspects (participation, collaboration, and use of resources) and how they are used to define the current and desired state of the interaction. For instance, *current participation* is the sum of interactions analysed at individual or group level; *current collaboration* is measured by the face-to-face interactions and/or interactions mediated by shared resources; and *current use of resources* is analysed by means of the number of individual and group interactions.

On the other hand, the desired state of the interaction is derived from the values of the script parameters defined at design time. For example, if the social level is *individual* and the participation is *mandatory*, there should be at least one piece of evidence of participation; if the activity is *collaborative* then there should be evidence of at least two participants interacting face-to-face or through shared resources; and, if the use of a resource is *mandatory* for groups, at least one participant of the groups should interact with the resource.

Also the pedagogical pattern(s) may set some constraints in the design. For instance, patterns such as CLFPs (Hernández-Leo et al., 2006) define guidelines related to the activity flow, the collaboration, and the group formation policies. These guidelines may be used to identify dependences between activities and, based on these dependences, predict the impact of an expected situation in future activities.

The main role of the aforementioned indicators is to detect when there is no evidence of a specific type of activity taking place (e.g., one student has not submitted its assignment). This is complemented by showing the teachers very simple data (e.g. number of accesses to a tool), that they are expected to interpret in their own contexts.

The reason why our indicators are minimalistic (at least at this stage) is based on the fact we are dealing with DLE and blended learning scenarios, where the data obtained by the system can be very simple (in the case of DLE, we cannot assure we will get deep data) (Krüger et al., 2010), and incomplete (in the case of blended scenarios, a lot of activity is carried out of the classroom and of the DLE. In our approach, we assume that the complexity of the results obtained from the analysis must be in line with the depth and reliability of the data you get from the scenario. Indeed, we hypothesize that in spite of their simplicity, these indicators are useful for the teacher to manage his classes.

Table 3 Definition of the current state of the interaction according to the indicators related to *participation, collaboration, and use of resources.*

Participation: applied to individuals or groups depending on the social level.

Individual participation: the involvement of each participant in the activity is measured by:

- *The number of interactions that (s)he develops*:

  {∑interaction | interaction.actor.id = participant.id}

Group participation: the involvement of each group in the activity is measured by:

- *The number of interactions that the group members develop*:

  {∑interaction | interaction.creator ⊆ group.participants}

Collaboration among group members: for each group configured according to the social level, the collaboration is measured by the interactions in shared resources:

- *The interactions that the group members develop*:

  {∑interaction | interaction.creator ⊆ group.participants}
- *The interactions that involve shared resources in the group:*

  {∑interaction | interaction.resource ∈ group.resources_to_be_monitored}

Use of monitored resources: for each monitored resource that supports the activity, the use that participants make of it is measured by:

- *The interactions that participants make*:

  {∑interaction | interaction.creator ⊆ activity.participants}

Table 4 Constraints associated to the considered indicators based on their expected values, i.e. the script content.

Participation: depending on the social level (individual/group), the expected participation values are:

Individual participation: for each participant:

- *If activity.expected_participation =optional* → participant.participation ≥ 0
- *If activity. expected_participation =mandatory* → participant.participation ≥ 1

Group participation: for each group:

- *If activity.expected_participation =optional* → group.participation ≥ 0
- *If activity. expected_participation =mandatory* → group.participation ≥ 1

Collaboration among group members: if the activity is collaborative, for each group the expected collaboration values are:

- *At least two group members participate* → ∃ participant1.participation ≥ 1 & participant2.participation ≥1

Use of monitored resources: for each monitored resource that supports the activity, according to the **expected use**:

- *If resource.expected_use= mandatory* → activity.resource.use ≥ 1
- *If resource.expected_use= recommended* → activity.resource.use ≥ 0

3.3. Compare Current and Desired States of Interaction

For each monitoring period, current and desired states of the interaction are compared, checking the constraints. In those cases where the evidence does not satisfy the expected values, the analysis triggers warnings highlighting the problem (e.g. lack of participation, lack of collaboration, unexpected use of resources). Once the state of each activity is analysed, its impact on future activities is also ckecked (for instance, unavailable resources or unstructured groups).

3.4. Advise/Guide the Interaction

This step aims at informing the teacher about the commonalities/discrepancies between the current and the desired states of the interaction. Especially we warn the teacher about the lack of evidence of expected participation, collaboration, use of resources, etc. In addition, the pattern allows us to predict the impact of unexpected situations in future activities. With this information the teacher could manage the learning situation according to his/her needs.

4. Illustrative Learning Scenario

This section illustrates the script-aware monitoring process presented in this article with an example based on one of out currently running validating cases.

The section is structured as follows: first, we introduce briefly the learning scenario; then, we describe the script identifying the parameters that will guide the analysis; next, we explain how the four steps of the script-aware monitoring process were applied to this learning scenario; and finally, we end with a discussion.

4.1. Context of the Illustrative Learning Scenario

The study took place in a CSCL scenario at the University of Valladolid, in a course of a Master's Degree for Pre-service Secondary Education Teachers. 13 students and a teacher with previous experience in CSCL scenarios participated in the study.

During the study, we followed the cycle described in Figure 1. First, the teacher defined the script according to the monitoring-aware design process (Rodríguez-Triana et al., 2012a) and, later on, the students' activity was monitored during the enactment of the script (from February 17^{th} to March 9^{th}, 2012). The gathered evidence was compared with the desired state derived from the script definition, and the teacher was informed about the coincidences and differences at the end of each learning activity. We gathered teacher's feedback during the study and carried out an interview with her at the end of the intervention. Although in this article we only present part of the learning scenario, we will bring out some ideas extracted from the teacher's feedback.

Regarding the technological support of the CSCL scenario, we used a DLE based on *GLUE!*[4] (Group Learning Unified Environment). *GLUE!* is an architecture that integrates third-party tools in VLEs. *GLUE!* offers a set of functionalities that make it suitable for the application of this proposal. Dealing with the design, the *GLUE!-PS*[5] (*GLUE!*- Pedagogical Scripting) module supports teachers in the instantiation of their designs across DLEs, and provides a computational representation of the script. By means of GLUE!-PS teachers can provide the students with specific technological placeholders (e.g., a GoogleDocs spreadsheet, a Dropbox folder,…) where they can elaborate or upload their learning outcomes. Regarding monitoring, the *GLUE!-CAS* (Collaboration Analysis Support for *GLUE!*) module facilitates the data gathering and integration of users interactions data coming from the heterogeneous components of the DLE (Rodríguez-Triana et al., 2011). In order to automatize the script-aware monitoring process, we built a tool called *GLIMPSE* (Group Learning Interaction Monitor for Pedagogical Scripting Environments). This tool takes the script generated by *GLUE!-PS* and, based on its content, guides the data gathering (asking *GLUE!-CAS* for users' interactions registered in specific periods and tools); applies the heuristics described in Table 2; defines the desired state according to the indicators presented in Table 3 and Table 4; compares the desired and the real state; and generates a monitoring report with the results.

4.2. Monitoring-Aware Design Process of the scenario

The study is framed within a course on "Learning methods for Technology and Computer Science". During this course, students had to analyse different learning methods applicable to secondary education. In order to help them in understanding and internalizing these topics, they were asked to study a specific educational context and to decide which methods could be the most appropriate, providing an example of their application to that context, thus illustrating their decision. To perform this task, the students worked in groups in a blended CSCL setting, interleaving face-to-face with distance activities mediated by ICT tools.

Though the real script contained several additional activities, in this article we will focus on a subset of them that implemented a *Jigsaw* Collaborative Learning Flow Pattern (CLFP) (Hernández-Leo et al., 2006). The Jigsaw pattern guided the **student grouping**, as well as the **activity flow**. In the first activity students, individually, analysed two learning methods and elaborated a summary about such methods. During the second activity, those students who had worked about the same methods joined to form expert groups. Each expert group developed a collaborative mind map with the group's main ideas. In the third activity, students, working in jigsaw groups (i.e. in groups formed by experts in each of the topics), chose the most suitable methods to learn with ICT tools and elaborated a poster with the proposal. These activities are summarized in Table 5.

Table 5 Description of the activities detailed in the illustrative scenario (ITC - Inside the classroom / OTC - Outside the classroom). The activity in bold has been used to illustrate the analysis process

Activity	Social level	Participation	Interactivity type	Physical location	Monitored resources & interactions
Individual work	Individual	Mandatory	Through computers	OTC	* *Editions/uploads* in *MediaWiki*
Expert consensus	**Expert groups**	**Mandatory**	**Blended**	**ITC & OTC**	*** *Attendance* to the lab session** *** *Accesses Dabbleboard*** *** *Editions/uploads* in *MediaWiki***
Poster development	Jigsaw groups	Mandatory	Blended	OTC	* *Uploads* in *MediaWiki*

In addition to the group configuration and the tasks to be carried out in each one of the activities, the teacher complemented the script definition with decisions that affected monitoring. As mentioned in Table 1, these parameters included the activity **deadlines** (with explicit starting and ending points), the specification of the **social level** (individually/by groups/whole class), the **interactivity type** (face-to-face, through computers or blended), the required **participation** (optional, mandatory for individuals, mandatory for groups), and the **physical location** (inside and/or outside the classroom).
The activities required ICT **tools** that satisfied the pedagogical needs, and, at the same time, provided data about the users' interactions. The teacher chose *MediaWiki*[6] to centralize the access to all the resources and activities (as a VLE), and to support the writing tasks; and *Dabbleboard* to accomplish the drawing tasks. Besides, we used the aforementioned *GLUE!* architecture to integrate *Dabbleboard*[7] into *MediaWiki*. Additionally, the teacher specified the **resources and interactions to be monitored** as well as the **expected use** of such resources (optional/mandatory, individual/by groups).

Since one of the activities was partially developed in the classroom, the *Expert consensus*, the teacher planned to control the attendance to the lab session in order to be aware of who interacted face-to-face. Thus, the evidence gathered from the technological support was enriched with the attendance evidence collected by teacher.

Finally the teacher defined the **periods of analysis** and asked the research team for a monitoring report at the end of each activity.

4.3. Script-Aware Monitoring Process

This subsection explains how we applied the four steps of the script-aware monitoring presented in Section 3 during the enactment of the aforementioned CSCL script.

Collect Interaction Data. Based on the definition of the activities and the teacher's monitoring decisions presented in the previous subsection, we obtained a set of heuristics to collect and filter the interaction data. For instance, during the *Expert* *consensus* (see Table 5), the interactions selected for the analysis were those generated by the students between February 24th at 10:00 and 27th at 22:00. The considered resources and interactions were those specified by the teacher at design-time: the *editions and uploads* in *MediaWiki*, the *accesses* to *Dabbleboard*, and the *attendance* to the lab sessions registered by the teacher (see column "Monitored resources & interactions" in Table 5). Since each expert group had been assigned its own wiki pages and *Dabbleboard* instance, only the interactions developed by the group members were considered.

We used the *GLUE!-CAS* module for the data gathering and integration. While *MediaWiki* registers and provides information about the users' activity (e.g. by means of the database), *Dabbleboard* did not offer any kind of information. Then, we used the *GLUE!* logs to control, at least, when the students accessed the tool. To gather the information from the *Attendance register*, we provided the teacher with a spreadsheet where she included the information.

Construct a Model of Interaction. The purpose of the analysis is defined in this step of the process, based on the pedagogical pattern(s) and the parameters of the script. The script implements a *Jigsaw* CLFP (Hernández-Leo et al., 2006). This pattern imposes some constraints that must be accomplished for achieving the pedagogical goals. Table 6 shows the group formation policies for the three phases of the activity flow, as well as the expected collaboration in each one of them.

Regarding the script parameters, the *Expert consensus* (see Table 5), was designed to be a *collaborative* activity carried out in *expert groups*. The activity was developed during and after the lab session, combining face-to-face and computer-mediated interactions. The teacher provided each group with a collaborative drawing tool as well as the wiki pages where they had to submit their proposals. Table 7 shows the target users and the expected use (mandatory vs. optional) of the resources involved in the activity, according to the script definition.

As it is shown in Table 7, during the *Expert Consensus*, the *individual participation* and the *collaboration among group members* were measured, taking into account computer-mediated and face-to-face interactions (based on the edition and uploads in *MediaWiki*, the accesses to *Dabbleboard*, and the attendance to the lab sessions). The group participation was informed by the use of *MediaWiki* (the mandatory resource they have to employ to complete the activity tasks). And, finally, the *use of each resource* was studied based on the interactions gathered.

Table 6 Constraints that the Jigsaw CLFP imposes to the illustrative scenario**. X** represents that the restriction must be satisfied in that specific phase of the pattern (individual, expert and jigsaw).

Structuring constraints	**Individual** (individual)	**Expert** (collaborative)	**Jigsaw** (collaborative)	**Description**
group sizes		**X**	**X**	The group size must be ≥ 2 to allow collaboration.
expert group sizes		**X**	**X**	The group sizes must be ≥4 to provide participants to each jigsaw group.
jigsaw group sizes			**X**	The group sizes must be ≥3 to gather experts from all areas.
no. of subproblems	**X**	**X**	**X**	The number of subproblems must be ≥2 and ≤ 6 (half the number of participants) to allow for collaboration in the expert groups
no. of expert groups	**X**	**X**	**X**	The number of expert groups must be ≥3 (the number of subproblems) and ≤ 6 (half the number of participants) to allow for collaboration in the expert groups.
no. of jigsaw groups			**X**	The number of jigsaw groups must be ≤ 4 (the number of experts of each area).
group dependencies			**X**	There must be experts of all areas in each jigsaw group.

Table 7 Resources monitored, and related monitoring aspects, in the analysis of the *Expert consensus* activity.

Resources	**Target users**	**Expected use**	**Participation**		**Collaboration**	**Use of resources**
			Individual	**Group**		
MediaWiki	Expert groups	Mandatory	X	X	X	X
Dabbleboard	Participants	Optional	X		X	X
Attendance to the lab session	Participants	Optional	X		X	X

Compare Current and Desired States of Interaction. In order to compare the current and the desired states of the *Expert Consensus*, we used the *GLIMPSE* tool. In terms of *participation*, found evidences indicate that 12 of 13 students were involved in the activity, while no computer-mediated or face-to-face interactions were detected for the remaining student (Helen, see Figure 2). Regarding *collaboration,* all three groups presented collaboration face-to-face and/or through the *Dabbleboard* and *MediaWiki*. However, two groups did not use *MediaWiki* as expected (no upload or edition was registered in expert groups A and C, see Figure 2).

Besides, regarding the pattern constraints, the analysis triggered two warnings. On the one hand, the group formation policies set by the pattern (see Table 6) might be affected. According to the pattern definition, each jigsaw group has to be formed by at least one expert on each area. Thus, a student that does not participate during the expert phase might not contribute with her expertise to her jigsaw group. On the other hand, the unexpected use of the resources (two expert groups did not submit their proposals in *MediaWiki*) could affect the activity flow, as the products of this phase were to be used in the *Poster development in Jigsaw groups.*

Advise/Guide the Interaction. Figure 2 shows a snippet of the monitoring report sent to the teacher with the analysis of the *Expert consensus* (participants' names have been modified to remain anonymous). Aware of the potential problems detected, the teacher reviewed the students' work in *MediaWiki* and realised that the Expert Groups A and C had uploaded their proposals in a wrong wiki page. Therefore, she asked the groups to modify the location of their submissions, avoiding any negative effect in the next activity. Additionally, to verify the individual participation, the teacher contacted Helen and her group, conforming that the student had interacted with her team using other means of communication (such as e-mails or face-to-face meetings out of the classroom).

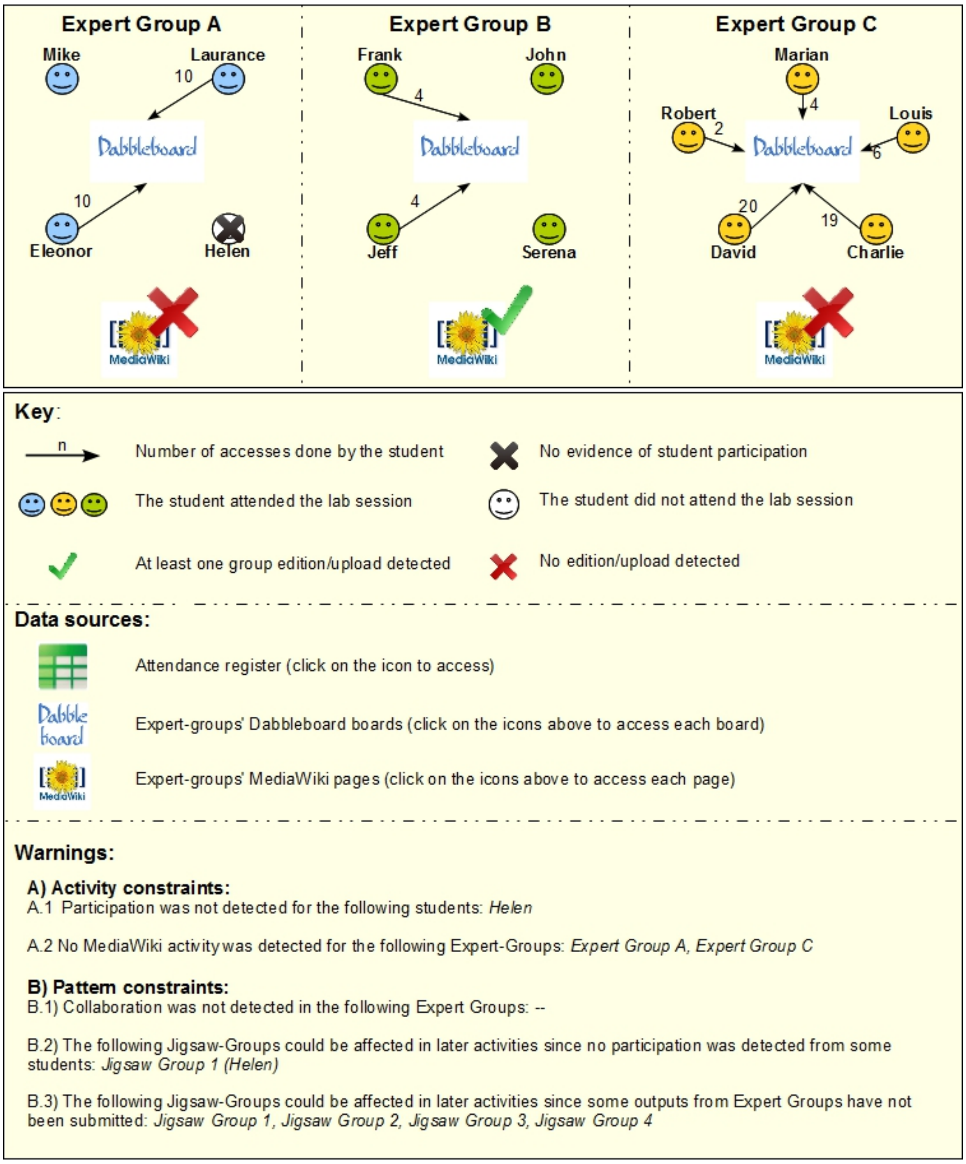

Figure 2 Example of the information sent in the monitoring report of the activity *Expert consensus* (participants' names are anonymized).

4.4. Discussion

As mentioned beforehand, our intention is not only to use the example to illustrate the proposal, but also to open the discussion about its applicability and the potential benefits for teachers in other contexts.

The aformentioned example shows that it is **feasible** to guide the monitoring process by means of the parameters collected from the script (see Table 1). Moreover, we could automatize part of the process by means of the *GLUE!-CAS* and the *GLIMPSE* prototypes, namely data gathering, definition of the current and desired states, and comparison between them.

In addition, the teacher's feedback supports our hypothesis: focusing the analysis on the teacher's pedagogical intentions captured in the script provided her with **meaningful** information for managing the learning situation. The monitoring reports offered an integrated view that allowed the teacher, at a glance, to identify issues in the activity that could require her reaction.

The example shows how our proposal faces usual problems that appear in CSCL settings, irrespective of whether they involve small or big groups of students. From the technological viewpoint, the scenario shows that supporting teachers in these complex settings requires gathering data from different sources and integrating heterogeneous data. From the teacher's perspective the to-do list is overwhelming even with a few students: many artefacts are modified and may need supervision; students must collaborate during the learning process but it is not straightforward to verify their participation; activities depend on each other and it is not trivial to foresee the potential impact of a deviation in the plan, etc. Besides, teachers usually have neither much time nor technological support to follow the learning process.

The described scenario interleaves face-to-face and computer-supported activities that happened inside and outside the classroom. This configuration of the learning context offers to the students many forms of interaction not restricted to the technological support. The example shows how the proposed indicators, despite being minimalistic, may provide relevant information for the management of the learning scenario.

In a more centralized scenario, supported maybe by a single LMS and its internal tools, we could assume that more reliable data can be obtained, and therefore, define higher-level indicators. Besides this, even the simple values we are considering at this moment might be adapted to norms of "little / medium / high" based on the characteristics of each context (e.g. based on comparison with peer groups if every group follows the same steps and with the same tools).

5. Conclusions and Future Work

Several works in the literature have pointed out the potential synergies of integrating Learning Design and Learning Analytics. Based on this assumption, we propose to link two well-known design and analysis techniques in CSCL: scripting and monitoring. This article presents a script-aware monitoring process where the data gathering and analysis are guided by the teacher's pedagogical intentions specified in the script, with the aim of providing teachers with relevant information for the management of the learning scenario.

To illustrate and better understand the proposal, we have presented a piece of an authentic scenario where we followed the CSCL script lifecycle presented in Figure 1. From the technological perspective, the example shows that guiding the monitoring process with the script information is feasible. Besides, from the teacher's point of view, the feedback obtained from the analysis provides relevant information for the management of the learning situation, helps to anticipate potential problems, and reduces the management effort.

Though the illustrative scenario showed the benefits of the proposal working with small groups, we foresee that this proposal may be especially useful in scenarios with large number of students and/or following on-line learning such as MOOCs, where developing CSCL activities is very complex due to the dropout rate. Being able to predict potential problems obtained from this kind of analysis could be used for adaptation, intervention and personalization (Looney and Siemens, 2011). However, to verify this assumption we need to explore suitable visualization techniques to present bigger amounts of information, a key issue in those contexts.

This proposal aligns learning design and analysis techniques in a similar way than the NEXT-TELL project integrates learning design and assessment. While we focus on how students follow the learning process, the NEXT-TELL project pays attention on detecting learning evidence. We believe that joining both lines of work could help to better support teachers and students in the learning process.

Though, from a pedagogical point of view, we have followed a teacher-centred approach in this stage, we plan to explore whether our proposal may be applied in learning scenarios in which the orchestration tasks are shared by teachers and students. One possible research line would aim at sharing with the students the information generated by the script-aware monitoring process. Extracting from the learning design the students' expected interaction goals and making assessment criteria explicit (Looney and Siemens, 2011), this approach could be used for self-monitoring, informing students about how to reach such goals. Another work line is to extend our proposal with personal learning environments (PLEs), a technological context that presents common problems with DLEs due to the distributed nature of tools and information. For that purpose, we consider that the Tin Can API and CAM architectures could be helpful to implement the proposal in PLEs, concretely to support the data gathering and integration.

6. References

Chatti, M.A., Dyckhoff, A.L., Schroeder, U. and Thüs, H. (2012), "A reference model for learning analytics", *International Journal of Technology Enhanced Learning*, Vol. 4 No. 5/6, pp. 318–331.

Dillenbourg, P., Järvelä, S. and Fischer, F. (2009), "The Evolution of Research on Computer-Supported Collaborative Learning", in Balacheff, N., Ludvigsen, S., Jong, T., Lazonder, A. and Barnes, S. (Eds.),*Technology-Enhanced Learning*, Springer Netherlands, pp. 3–19.

Dillenbourg, P. and Jermann, P. (2007), "Designing Integrative Scripts", in Fischer, F., Kollar, I., Mandl, H. and Haake, J.M. (Eds.),*Scripting computer-supported communication of knowledge – cognitive, computational, and educational perspectives*, Springer US, New York, pp. 275–301.

Dillenbourg, P. and Tchounikine, P. (2007), "Flexibility in macro-scripts for CSCL", *Journal of Computer Assisted Learning*, Blackwell Publishing Ltd, Vol. 23 No. 1, pp. 1–13.

Dron, J. and Anderson, T. (2009), "On the design of collective applications", *International Conference of Computational Science and Engineering*, pp. 368 – 374.

Gamma, E., Helm, R., Johnson, R. and Vlissides, J. (Eds.). (1995), *Design Patterns: Elements of Reusable Object-oriented Software*, Addison-Wesley Professional.

Harrer, A., Martínez-Monés, A. and Dimitracopoulou, A. (2009), "Users' data: collaborative and social analysis", in Balacheff, N., Ludvigsen, S., de Jong, T., Lazonder, A., Barnes, S. and Montandon, L. (Eds.),*Technology-Enhanced Learning. Principles and Products*, Springer Netherlands, pp. 175–193.

Hernández-Leo, D., Villasclaras-Fernández, E.D., Asensio-Pérez, J.I., Dimitriadis, Y. and Retalis, S. (2006), "CSCL Scripting Patterns: Hierarchical Relationships and Applicability", *Proceedings of the 6th IEEE International Conference on Advanced Learning Technologies*, ICALT'06, Kerkrade, The Netherlands, pp. 388–392.

Kobbe, L., Weinberger, A., Dillenbourg, P., Harrer, A., Hämäläinen, R., Häkkinen, P. and Fischer, F. (2007), "Specifying computer-supported collaboration scripts", *International Journal of Computer-Supported Collaborative Learning*, Vol. 2 No. 2-3, pp. 211–224.

Kollar, I., Fischer, F. and Hesse, F.W. (2006), "Collaboration Scripts – A Conceptual Analysis", *Educational Psychology Review*, Vol. 18 No. 2, pp. 159–185.

Krüger, A., Merceron, A. and Wolf, B. (2010), "A Data Model to Ease Analysis and Mining of Educational Data 1", *Proceedings of the 3rd International Conference on Educational Data Mining*, International EDM Society., Eindhoven, The Netherlands, pp. 131–140.

Lockyer, L. and Dawson, S. (2011), "Learning designs and learning analytics", *International Conference on Learning Analytics and Knowledge*, ACM, New York, USA, pp. 153–156.

Looney, J. and Siemens, G. (2011), *Assessment Competency: Knowing What You Know and Learning Analytics. It is Time for a Breakthrough*, *Promethean Thinking Deeper Research Papers Series*, Promethean Thinking Deeper Research Paper No.3.

MacNeil, S. and Kraan, W. (2010), *Distributed Learning Environments: A briefing paper*, JISC Center for Educational Technology and Interoperability Standards (CETIS).

Martínez-Monés, A., Harrer, A. and Dimitriadis, Y. (2011), "An interaction-aware design process for the integration of interaction analysis into mainstream CSCL practices", in Puntambekar, S., Erkens, G. and Hmelo-Silver, C.E. (Eds.),*Analyzing Interactions in CSCL*, Springer US, pp. 269–291.

Miao, Y., Hoeksema, K., Hoppe, H.U. and Harrer, A. (2005), "CSCL Scripts : Modelling Features and Potential Use", *Computer Supported Collaborative Learning Conference: the next 10 years!*, International Society of the Learning Sciences, Taipei, Taiwan, pp. 423–432.

Mitra, S., Pal, S.K. and Mitra, P. (2002), "Data mining in soft computing framework: a survey", *IEEE transactions on neural networks*, Vol. 13 No. 1, pp. 3–14.

Rodríguez-Triana, M.J., Martínez-Monés, A. and Asensio-Pérez, J.I. (2011), "Monitoring Collaboration in Flexible and Personal Learning Environments", *Interaction, Design and Architecture(s) Journal, special issue on: Evaluating Educative Experiences of Flexible and Personal Learning Environments.*, Vol. 11 No. 12, pp. 51–63.

Rodríguez-Triana, M.J., Martínez-Monés, A., Asensio-Pérez, J.I. and Dimitriadis, Y. (2012a), "Towards a monitoring-aware learning design process", *International Conference on Collaboration and Technology*, Duisburg, Germany.

Rodríguez-Triana, M.J., Martínez-Monés, A., Asensio-Pérez, J.I. and Dimitriadis, Y. (2012b), "Script-Aware Monitoring Model: Using Teachers' Pedagogical Intentions to Guide Learning Analytics", *Proceedings of the Workshop Towards Theory and Practice of Teaching Analytics, at the 7th European Conference on Technology Enhanced Learning*, Saarbrücken, Germany.

Scheffel, M., Wolpers, M. and Beer, F. (2010), "Analyzing Contextualized Attention Metadata with Rough Set Methodologies to Support Self-regulated Learning", *IEEE International Conference on Advanced Learning Technologies*, Ieee, pp. 125–129.

Siemens, G., Gasevic, D., Haythornthwaite, C., Dawson, S., Shum, S.B. and Ferguson, R. (2011), *Open Learning Analytics : an integrated & modularized platform Proposal to design , implement and evaluate an open platform to integrate heterogeneous learning analytics techniques*, Society for Learning Analytics Research (SOLAR).

Sobreira, P. and Tchounikine, P. (2012), "A Model for Flexibly Editing CSCL Scripts", *International Journal of Computer-Supporter Collaborative Learning*, Vol. 7 No. 4, pp. 567–592.

Soller, A., Martínez-Monés, A., Jermann, P. and Muehlenbrock, M. (2005), "From Mirroring to Guiding: a review of the state of the art in interaction analysis", *International Journal on Artificial Intelligence in Education*, Vol. 15, pp. 261–290.

Sutherland, R., Eagle, S. and Joubert, M. (2012), *A Vision and Strategy for Technology Enhanced Learning*, Report from the STELLAR Network of Excellence.

Vignollet, L., Ferraris, C., Martel, C. and Burgos, D. (2008), "A Transversal Analysis of Different Learning Design Approaches", *Journal of Interactive Media in Education*, North America, No. Comparing Educational Modelling Languages on the "Planet Game" Case Study.

[1] SoLAR: http://www.solaresearch.org/mission/about/ (last access, June 2013)
[2] NEXT-TELL Project: http://www.next-tell.eu/ (last access, June 2013)
[3] Tin Can API: http://tincanapi.com/ (last access, June 2013)
[4] GLUE!: http://www.gsic.uva.es/glue/ (last access, June 2013)
[5] GLUE! - PS: http://www.gsic.uva.es/glueps/ (last access, June 2013)
[6] MediaWiki: http://www.mediawiki.org (last access, June 2013)
[7] Dabbleboard: http://dabbleboard.wordpress.com/ (service no longer available)