A Model for Interoperability in Computer Supported Collaborative Learning

Georgios Kahrimanis*, Andreas Papasalouros**, Nikolaos Avouris*, Simeon Retalis***
*Human-Computer Interaction Group, ECE Department, University of Patras, Greece
** Software Engineering Lab, National Technical University of Athens, Zografou, Greece
***Dep. of Technology Education and Digital Systems, University of Piraeus, Greece
kahrimanis@ee.upatras.gr, avouris@upatras.gr, andpapas@softlab.ntua.gr, retal@unipi.gr

Abstract

Computer Supported Collaborative Learning activities involve combination of complex software tools that often need to interoperate in a wider context of learning. This paper proposes a data model that accommodates requirements of typical collaborative learning situations and facilitates interoperability of tools and interchange of products of collaboration and evaluation data. The model has been tested against various typical tools used for both synchronous and asynchronous collaboration of groups of students.

1. Introduction

Computer Supported Collaborative Learning (CSCL) is a rapidly developing research field, inspired by social theories of learning and based on today's ubiquitous network technologies.

In this research context considerable number of computer tools that support Collaborative Learning activities and of tools for analysis of these activities, have been recently proposed and used.

However, since CSCL constitutes a relatively new research area, little attention has been paid so far on how these tools can interoperate. So, in many cases researchers face compatibility problems when trying to take advantage of the features of these computer tools in the frame of research activities and no common understanding and exchange of data and research results can be easily achieved.

The first step towards interoperability is taken when certain CSCL developing and research teams try to make the tools that they use compatible with each other. But this practice is obviously limited to the tools that each team uses. A model that covers a wide range of different tools is needed.

Some existing similar proposals towards this objective refer to specific cases of CSCL activities such as collaboration through a shared workspace [8], collaboration through a shared workspace and dialogue tools [1], or dialogue tools that force participants to annotate the messages they send [10].

However, a really interoperable model should cover a wide range of CSCL activities conducted with the use of various tools. First of all it should ensure that a common logging format, suitable for most CSCL tools is defined. Furthermore, a key requirement is that the format takes into account the typical tools, supporting analysis of CSCL activities as well as additional data produced during the analysis phase. Finally, it should be based on a data model of typical CSCL activities as they take place in the real world. Data stemming from different sources should be described in an integrated way. It is also important to stress that the model should not be restrained in describing actions and properties of CSCL-specific collaboration and analysis tools, but it should describe any tools commonly used in CSCL activities (including discussion for a file management systems etc.)

In order for the model to cover these needs it has to be *operational*, formally defined and structured, *generic*, so that it is applicable to data of various sources and *flexible*, so that it can be used in various specific cases.

The most relevant approach found in the literature was conducted in [7]. The model targets at similar purposes, however, it is restrained to the description of interaction analysis and shares a subset of our goals, namely to provide a common interface so that log data gathered by CSCL tools can be used for analysis purposes.

The rest of this paper is structured as follows: section 2 negotiates in detail the need for a CSCL interoperability model, section 3 presents the desired properties that the model should have to satisfy these needs, section 4 proposes how the model can be technically applied, section 5 reports the development process of the model, section 6 describes the most important parts of the model, and section 7 reports the conclusions of this work thread and proposes how it can be further expanded and exploited.

2. The need for a CSCL Data Log Interoperability model

The needs that motivate the development of a CSCL Interoperability model are stated in the following subsections.

2.1 Interoperability between CSCL tools

It is often the case that each CSCL tool logs data in order to fulfill just its internal functionality needs regardless of any concerns on interoperability with other tools. A noticeable exception to this is found in cases of tools which may have been developed by the same developers. However, a common logging format would offer new possibilities towards interoperability among distinct tools.

Some classic architectures of Computer Supported Collaborative Work (CSCW) tools [4],[5] allow sharing low level information that is interpreted by the environment of each user and reproduced according to certain environment's user interface functionalities. In the case of a common logging format, it would be possible that two different tools could be used in the same CSCL session regardless of their additional functionalities for representing log file data in the user interface. Other prerequisites for the realization of such a scenario is that distinct tools function equivalently at the level of user interface constituent components (widgets) and a common protocol for coordination messages is also shared.

2.2 Compatibility between CSCL tools and analysis tools

Analysis of a CSCL activity constitutes an important part of the integration of CSCL activities in an educational environment. Teachers need to analyze CSCL activities in order to assess their students in terms of grades, to evaluate the activities in order to examine their impact on the wider educational activity and improve their practices according to the evaluation findings and conclusions. Researchers and tool developers also need to analyze data in order to shed light into important aspects of NSCL practices and improve NSCL tools accordingly.

Many analysis tools have been developed to aid research in the broader scope of behavioral sciences [9]. These tools face the limitation that most data they use as input are not structured and need hand-coding in order to be processed automatically. On the contrary, analysis tools designed specifically for CSCL evaluation studies [3] take advantage of data captured automatically by CSCL tools. However, it is usually the case that each tool can handle data produced by a limited number of related CSCL tools

A solution to this problem may be supported by the definition and adoption of a common logging format. In that case, diverse analysis tools providing support for various analysis techniques would be able to manipulate data produced by any CSCL tool. This widens the horizon of research from various methodological standpoints. It offers possibilities that would not easily be feasible if one had to be restricted to certain tools with specific analysis orientations.

There is also the case that compatibility between CSCL tools and analysis tools is needed both ways. Although this seems paradoxical, it constitutes a real need because in many cases CSCL environments contain integrated analysis tools and analysis results may need to be fed back to the collaborating partners.

2.3 Integrated description of data produced by a wide range of computer tools used in a CSCL activity

Researchers and activity designers in the field of CSCL often seem to have a fallacious perception of the nature of CSCL activities. They sometimes develop strict educational scripts, provide certain CSCL tools to the students and restrict the students to conduct an activity according to their directives [14]. However, in practice, students prove to be surprisingly flexible in terms of computer tool usage: they adopt alternative media in order to interact with their peers. Usage of mailing applications, instant massagers and asynchronous discussion forums are the most common examples.

This reality introduces a new problem when analyzing CSCL activities: Researchers examine CSCL interaction using the logs of the tools that they provide, missing important information conveyed via other channels of communication.

This introduces the need for a holistic description of data used as input for analysis, i.e. including all computer tools used during the CSCL activity regardless whether they are CSCL-specific or not.

Furthermore, in the case of face-to-face CSCL activities, other data, like video or even hand-coded data that describe other channels of communication such as oral dialogue or gestures, should be integrated in the common description.

3. Desirable properties of the model

In order for the model to cover the external demands stated in the previous section, it should satisfy the following characteristics.

First of all, it has to be *interoperable.* It should therefore be formally defined and structured in a standard language. We developed the model as an XML Schema [12] since it is a widely accepted standard that is sufficient for our purposes.

It should also be *generic* enough so that data of different sources and nature can be described in the model. There is a trade-off between the need for structure that interoperability imposes and the disparity of different data sources. The use of required and optional elements and the abstract form that some elements' meanings selectively take, is one mechanism used for setting an optimal balance for this trade-off..

There is also the need for *flexibility* of the model. Apart from being generic, the model should also be adaptable in various modes of usage according to the needs of different researcher standpoints. Thus, we define a model that is both human and machine interpretable. Moreover, flexibility refers to the need for the model to be applicable in specific circumstances with peculiar needs. For this reason we provide space in the model for some XML elements without a specified meaning, which can include information that is not explicitly described elsewhere in the model.

4. Application of the model

In order for the model to be actually used, the functionality of potentially compliant tools should be changed, so that they log data in an XML file following the model's format. However, this approach demands a great deal of effort, in terms of resources and time, which is not always affordable to the institutions or companies that are proprietors of the tools.

An alternative approach is that the tools' logging mechanisms remains the same and their log file is transformed into an equivalent log file compliant to the model. In case a tool logs data in the form of an XML file, XSL Transformations (XSLT) technology can be used for the conversion [13]. In cases where a tool stores data in a database or uses another markup language such as RDF, XML parser applications should be used.

5. Formation and development of the model

The development of the model followed a spiral process. It started from examples of typical CSCL tools and was followed by the examination of other tools one after the other.

ModellingSpace [2] (www.modellingspace.net) and *Synergo* [15] (www.synergo.gr) were the first CSCL tools taken into account. Afterwards, *ColAT* [3], which is an analysis tool compliant in a certain extent to the logging format of the aforementioned tools, was also tested. New elements that provided support for extra analysis facilities of ColAT, such as multiple-level grouping of actions were added to the model. Two non-CSCL open source tools, namely *PHPbb* and *Moodle*, were then tested. The process went on with the study of Noldus *Observer*™, which is a widely used analysis tool, suitable for a wide spectrum of research areas. Finally a similar model developed in the scope of Kaleidoscope [6] was examined. This model aims at providing unidirectional compatibility from CSCL tools to analysis tools. Nine learning support tools (four of which are CSCL tools) and seven analysis support tools were taken into account while developing that model. The authors of the model committed to provide the functionality for transforming the logging mechanism of all these tools so that they are compliant to their model. Our model was tested and was in large extend in concordance with model discussed in [6]. The mapping among the models has been defined so that our model can be compliant to all the tools taken into account in [6].

6. Description of the model

This section presents the main aspects of the model. The first part describes the *context* of the model and the second the *actions* taken place during an activity.

6.1 Description of context of a CSCL activity

The context part of the model was included due to several reasons: Firstly, there is a certain need for providing a structured description of the context of a CSCL activity. Secondly, data that refer to *user* descriptions, their *roles* and their assignments in *groups* should be stored once and not repeated recurrently accompanying actions. In addition, the description of the tools that are used and the information about the user-interface object types that they provide are included in the context part in favor of *modularity* of the model. Each action logged makes an explicit reference to the identifier of an object type and the property that it causes to change, instead of repeating object type specific information needlessly.

6.2 Description of actions in a CSCL activity

The first part of the action branch of the model concerns modeling of actions that commonly occur in CSCL activities. Concerning this part, what discriminates the model from previous approaches [7] and constitutes its innovative proposal is the way objects are instantiated and referenced. As stated in the previous section, an object is instantiated with a certain action. Its type, the tool that it belongs to and its property types (that an action may cause to change), are referenced in the context part of the model. In addition, a further distinction to previous models is that the model explicitly supports non-CSCL tools, even hand-scripted interactions occurring during face-to-face interactions.

6.3 Logging of actions in an analysis activity

The second part of the action part of the model includes information that relates to data used for analysis of CSCL activities. The most common and trivial action that an analysis tool facilitates and the model supports is the addition of a comment or a description to an action. The model takes also into account cases in which an action is classified according to a coding scheme. Usually, messages are coded and assigned to a category of a classification scheme so that they indicate an “agreement”, a “proposal” etc. The coding may be defined by the users prior to posting the messages (explicitly: by selecting a category, or implicitly: by choosing sentence openers mapped to categories). Alternatively, the assignment into categories may take place a posteriori during the analysis of the activity by researchers or teachers engaged in content analysis studies [11].

However, in that way, the model prescribes log files that are just treated as an input from the analysis tools. Thus, we proceeded the development of the model and integrated descriptions of actions taken by researchers and logged by analysis tools.

A more sophisticated facility of analysis tools is the building of an interpretative model of an activity in the form of a multilevel structure. Collaboration Analysis Tool (ColAT) [3], which was studied in the scope of this work, follows an Activity Theory approach. It allows grouping of actions in three distinct levels, namely *Operations*, *Actions* and *Activities*. ColAT logs information about multiple-level grouping in its peculiar database. In our model we provide a complex element called *level annotation* that contains the name of the level as an attribute (e.g “Operation level”) and the name of the certain instance of that level as a sub-element. Consequently, the model satisfies the need of a multilevel interpretation and it not restricted to certain interpretations of that kind as well.

7. Validation of the model

Log files produced during real-world CSCL activities by all the tools mentioned in section 6 were transformed into their equivalents in the format of the Interoperability Model in order to validate it. In this paper we present just an example of such validation. A typical example of a logged action (dispatch of a chat message) in ModellingSpace format is shown in Fig. 1.

```
<event>
  <time>09 : 47 : 55</time>
  <time2>00 : 07 : 44</time2>
  <user>thodoris</user>
  <action>Chat message</action>
- <attribute>
    [isws kai enas server na einai ontothta giati exei p.x mia dieuthunsh ]
  </attribute>
  <typology/>
  <comments/>
  <added_by_user>false</added_by_user>
  <id_event>49</id_event>
</event>
```

Figure 1 - Action log according to ModellingSpace’s format

The equivalent log in the model we presented would take the form that is shown in Figure 2.

```
- <action id="49" abs_time="09:47:55" rel_time="00:07:44" successful="1">
    <user_ref>2</user_ref>
    <type>Chat_change</type>
    <object_type_ref>3</object_type_ref>
    <object_ref>41</object_ref>
  - <property_change property_ref="1">
      isws kai enas server na einai ontothta giati exei p.x. mia diauthunsh
    </property_change>
  </action>
```

Figure 2 – Action log according to Interoperability model format

8. Conclusions

During the process of developing the Interoperability Model, proposed in this paper, it was found that most log files of CSCL tools use equivalent semantics to store and describe log file data despite their technical variations. This implies that no major modifications would be needed in order for most tools to comply with the interoperability model.

Concerning analysis tools, we noticed that they provide flexible formats that can be appropriate for various analysis approaches, however not much attention is paid to the output that they produce. This imposes a problem for such tools to be fully compatible to the interoperability model. Additional functionality should be supported so that analysis tools can create log files according to the model.

The mapping of non-CSCL tools was also proved to be semantically straightforward, but some of these tools also have the problem of logging data just for their internal functionality purposes (usually in a database), regardless of interoperability concerns. However, we believe that it is worth the effort to make tools used in a CSCL activity compliant to the interoperability model, due to the new range of possibilities that this standard would provide.

Further work, extending this research thread can follow two directions. Firstly, concerning the development of the model, more tools of wider diversity could be used in order to test and reform the model. We noticed that the model converged to a certain form as its formative evaluation in relation to certain tools proceeded. However, there is always space for further reform or enrichment of the model with additional features. Secondly, further long-term application of the model in various real-world circumstances should reveal possible shortcomings of the model or inspire further research. The direction of proposing the model as a standard and making it compatible with other existing standards in the field may enhance its use and should be pursued.

Acknowledgement. The reported work has been conducted in the frame of the e-learning TELL2003-4721/001-001 EDU-ELEARN project, funded by the European Union. The ideas have also been discussed in the frame of IKYDA project of German-Greek Scientific Collaboration supported by IKY and DAAD.

10. References

[1] Avouris, N., Dimitracopoulou, A., and Komis, V., "On analysis of collaborative problem solving: an object-oriented approach". Journal of Human Behavior", 19(2), 2003, 147-167.

[2] Avouris N., Margaritis M., Komis V., Saez A. and Melendez R., "ModellingSpace: Interaction design and architecture of a collaborative modelling environment". In Proceedings of 6th Conference on Computer Based Learning in Science (CBLIS) (ed. C. Constantinou), University of Cyprus, Nicosia, Cyprus, 2003, pp. 993–1004, available from http://hci.ece.upatras.gr.

[3] Avouris N., Komis V., Fiotakis G., Dimitracopoulou A., and Margaritis M., "Method and Tools for analysis of collaborative problem-solving activities" , Proceedings of ATIT2004, First International Workshop on Activity Theory Based Practical Methods for IT Design , Copenhagen, Denmark, September 2004, pp. 5-16, available from http://hci.ece.upatras.gr.

[4] Calvary, G., Coutaz, J., and Nigay, L: From Single-User Architectural Design to PAC*: a Generic Software Architecture Model for CSCW, Proceedings of CHI 97, ACM publ., 1997, pp. 242-249.

[5] Dewan, P, "Multiuser Architectures", in Proceedings EHCI'95, Working Conference on Engineering Computer Human Interaction, 1995.

[6]Harrer A. et. al., The Common Format, Chapter 4, in A. Martinez, Library of Interaction Analysis Methods, Deliverable D31.2.1, Kaleidoscope NoE (available from www.noe-kaleidoscope.org), 2005.

[7] Martinez A., de la Fuente P., and Dimitriadis Y., , "An XML-based representation of collaborative interactions", In B.Wasson, S. Ludvigsen & U. Hoppe (Eds) Computer Support for Collaborative Learning: Designing for Change in Networked Learning Environments, (CSCL 2003), Bergen, Norway, 2003, pp. 379-384

[8] Muhlenbrock, M., "Action-based collaboration analysis for group learning", IOS Press, Amserdam, 2003.

[9] Noldus LPJJ, "The Observer: a software system for collection and analysis of observational data". Behav. Res. Methods Instrum 2, 1991, pp. 415–429

[10] Soller, A., "Supporting social interaction in an intelligent collaborative learning system", International Journal of Artificial Intelligence in Education, 12, 2001, pp. 40-63.

[11] Strijbos, J. W., Martens, R., and Prins, F., "Content analysis: What are they talking about?". In B. Wasson, R. Baggetun, U. Hoppe & S. Ludvigsen (Eds.), CSCL 2003: Community events, communication and interaction, Bergen: Intermedia, University of Bergen, 2003, pp. 74-76.

[12] XML 1.0 (Third ed.) W3C Recommendation, 2004, available on line at: http://www.w3.org/TR/REC-xml/.

[13] XSL Transformations (XSLT) Version 1.0 W3C Recommendation, 1999, available on line at: http://www.w3.org/TR/xslt

[14] Dillenbourg, P. (2002). Over-scripting CSCL: The risks of blending collaborative learning with instructional design. In P. A. Kirschner (Ed.) Three worlds of CSCL. Can we support CSCL (pp. 61-91), Heerlen, Open Universiteit Nederland.

[15] Avouris N., Margaritis M., and Komis V., Modelling interaction during small-group synchronous problem-solving activities: The Synergo approach, 2nd International Workshop on Designing Computational Models of Collaborative Learning Interaction, ITS2004, 7th Conference on Intelligent Tutoring Systems, Maceio, Brasil, September 2004.