Indicators computation from modeled traces in the context of computer Human Learning environment

TAREK DJOUAD [(1),(2)], MOHAMMED BENMOHAMMED [(1)], ALAIN MILLE[(2)]
[(1)] LIRE Laboratory, Mentouri Constantine University, Algeria
tarek_djouad@hotmail.com, ben_moh123@yahoo.com
[(2)] Liris Laboratory, Claude Bernard Lyon1 University, France
Alain.mille@liris.cnrs.fr

Abstract

We present in this paper TBS-IM a Trace Based System to calculate collaborative and individual human learning Indicators in Moodle. The system we propose is based on the concept of modeled trace, and allows users to build and reuse indicators using transformations trace models without using programming code.

Key words: *Modeled trace, indicators calculus, collaborative learning, Moodle.*

1. Introduction

We are interested in this paper by different methods used to facilitate the analysis of traces in collaborative learning situations. These methods are based on a Models Driven Engineering (MDE) approach to design and calculate effectively collaborative indicators, which are learning platforms independent. We will give in this section an overview of different works based on indicator calculus using traces. We present in section two our proposal method to compute indicators from modeled traces, and we prove this method in section three with a system we develop TBS-IM, and we conclude with our future works in section 4.

According to [1] an indicator is a mathematical variable which has a list of characteristics. It is a variable that takes values represented by digital, alphanumerical or graphical forms. The value has a status: the value is calibrated to other variables or interpreted directly by users (ex: teachers). A lot of works have been published about indicators, generally respecting this definition. For example in [2] we calculate the cohesion and the centrality in social networks from discussion forums. The platform ACOLAD [3] provides to tutors a tool which gives information about the activity triplet: Assiduity, Availability, and Involvement. [4] offers a tool that calculates from the interactions, the degree of involvement of each learner during the learning unit. It identifies: participative learner, useful learner, non-collaborative learner, learner which takes initiatives, and communicative learner. Other indicators are qualitatively interpreted as [5] where the density of the social network is interpreted using histograms. In [6], we measure agreement and disagreement between learners. The "ForumExplor" platform [7] works on forums' archives. It simplifies large forum analysis by offering different tools: thematic views and overviews of forums' archives. The thematic view is based on lexical recognition and pre-registered themes. Each theme is associated to a word list and a color.

All these works use data processing integrated to learning platform, and for a specific indicator calculus. This for example, will not allow to reuse indicator calculus in other learning platforms, or to calculate other indicators in the same learning platform. In the next section we explain a generic method to calculate and to reuse indicators in different learning platforms

2. Indicator computation from Traces

We present here a method to calculate indicators with a Traces-Based System (TBS). TBS, transformations and M-traces are defined in [8].

In the method proposed here, we calculate collaboration indicators from trace model transformations. This method includes three steps: Collecting data, trace transformation used to prepare the calculation of the indicator, and finally the calculation step. Figure 1 shows the order of these steps: collecting data, transformation of trace models and finally indicator calculus:

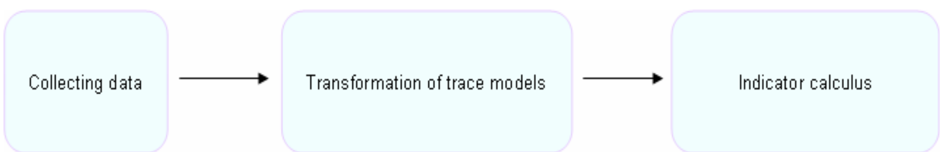

Figure 1: Our proposed approach in three steps

- Data collection consists in selecting pertinent/useful data within the tracing sources. On the one hand this collection depends on learning platforms, and on the other hand it depends on the primary trace model. The primary trace model defines what information is needed and the tracing sources, what is available. We use in this phase of collection, collaborative activity models, and, according to them we collect all obsels that can inform us about the behavior of collaborative learners.
- Starting from the primary trace, we propose transformation sequences using transformation model operators. These operators modify the model or the instances. Once the indicator is built, if the model changes, it becomes easy to modify the transformation sequence. We propose a library of models for transformed traces, where each trace model is associated to its transformations. We can then reuse the transformations to generate other models of M-Traces and therefore, other indicators.
- In indicator calculus we associate to each indicator “I” its trace model. This model helps to calculate the indicator directly. We define an indicator “I” by: I = {RI, TMRI} With: RI: the calculus Rule, TMRI the trace model used for calculation. Figure 2 illustrates this proposition, where a trace Trace(I), corresponding to its model TMRI, is associated to a collaborative indicator Indicator(I). The indicator value is derived from Trace(I) by applying a calculus rule RI. For example, we can calculate the indicator "Active Agent" by using a calculus rule we call RActiveAgent.

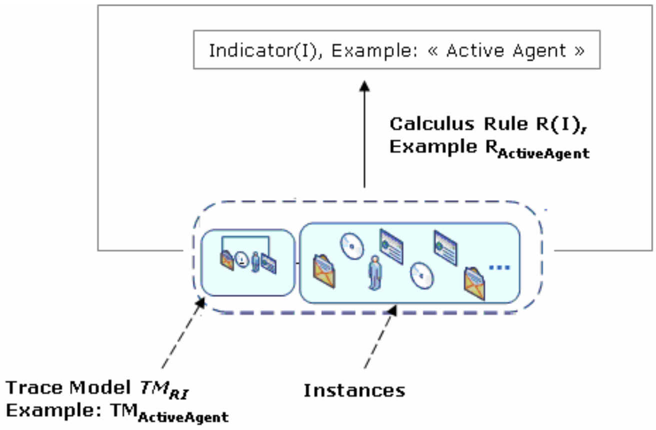

Figure 2: associate for each indicator a trace model to calculate it Results

3. Implementation

We will present here the implementation of various steps presented in section two. The implemented Trace-based system uses the platform Moodle [9] as tracing source, but its architecture is opened to other learning platforms.

3.1 Collecting data in Moodle

We propose here a specialized collecting phase for various collaborative learning activities. We are interested in synchronous and asynchronous activities, where actors work together on the same resources. The resources used in Moodle are: Wiki, chat, text resources, and private messages. In a previous work [10], we proposed a model for the primary trace Moodle. We extended the model to support additional actions that manage contacts. This additional information helps researchers to explain the collaborative behavior of learners using the generated indicators values. Our tool provides a selector to choose the list of data the user wants to collect. This choice depends on trace models the user wants to build in order to calculate the target indicator. Using Moodle, the tool connects to Moodle database, imports necessary data, and instantiates the trace model in OWL format. This format respects the syntax of Jena parser [11]. We use a graphical user interface (Fig. 3) to select data. As shown in Figure 3, the dialog box gives a list of observable actions we can pick up. Here, the user chose only 5 of 18 possible actions. Plugins we use collects data from tracing sources and instantiates the primary trace model. The result of this collecting phase is primary trace in OWL format.

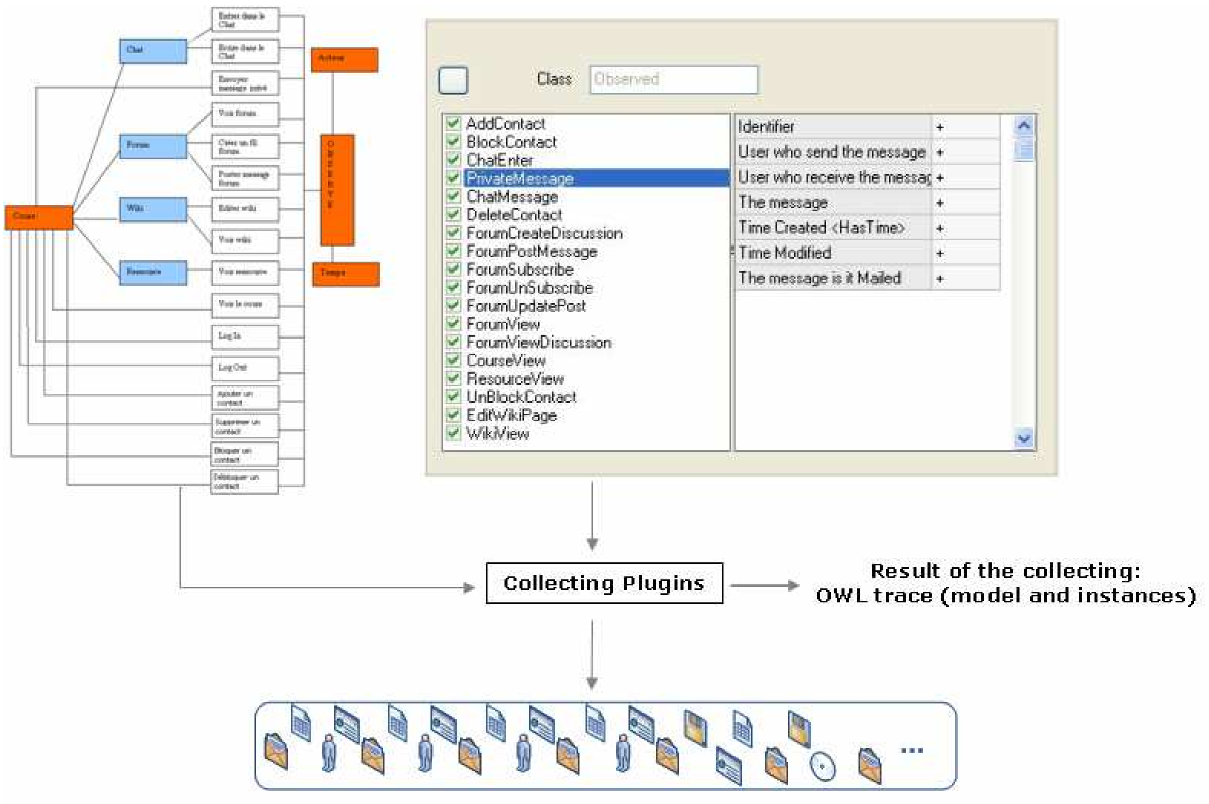

Figure 3: User interface used to collect observed actions

3.2 The transformation process

We implement in TBS-IM a lot of operators to transform (or not) traces models and instances. These operators are: slection, pruning, fusion, matching… more details about these operators are in [12]. For example the Matching operator identifies a sequence of observed actions using patterns. To find a pattern in such a sequence, we use an algorithm proposed by [13]. This operator can be used in the following way:

TraceX = Matching (pattern) [TraceY]

Where TraceX is the resulting trace, pattern is the pattern used to define our research criteria, while TraceY is the source trace where the matching function looks for episodes that fit the given pattern. For example, we can say that an effective enter in a Chat is sequence of actions such as " Chat Enter" followed by writing a message in the Chat “ChatWhriteMessage".

3.3 The indicators calculus. Example: the division of labor indicator

We show in this section how to apply operators on the M-trace to build transformation sequences, and then generate M-trace for collaboration indicators. We show that our implemented TBS can easily build indicators. And to illustrate the M-traces transformation, we will give a concrete example of the division of labor indicators calculus.

The division of labor calculus is based on the sum of the differences (SD) of two instances associated to each user on each tool. There is no specific rule in the CSCL (Computer Supported Collaborative Learning) literature to calculate this indicator. We propose the following transformation sequence to calculate the indicator. We use the selection and pruning operators to build the sequence (Fig. 4).

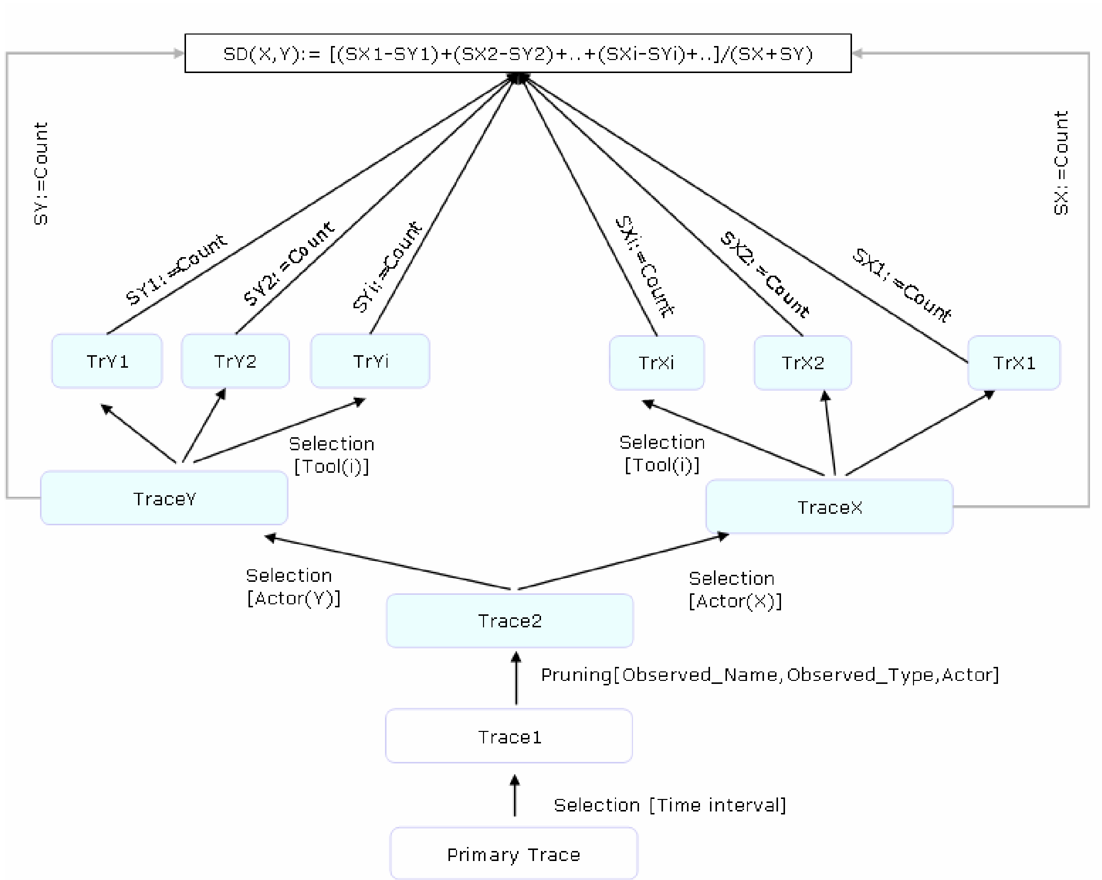

Figure 4: Generic transformations Sequence to calculate the indicator "Division of labor" between two actors ActorX and ActorY

We can notice that the transformations sequence for ActorX is the same for ActorY, which will allow us to reuse a part of the same sequence to build all the sequence, which means the ability to define and reuse small transformation sequences in other transformations. Figure 5 shows the division of labor calculus between actors "15" and "16" on tools "ToolChat1" and "Private Message".

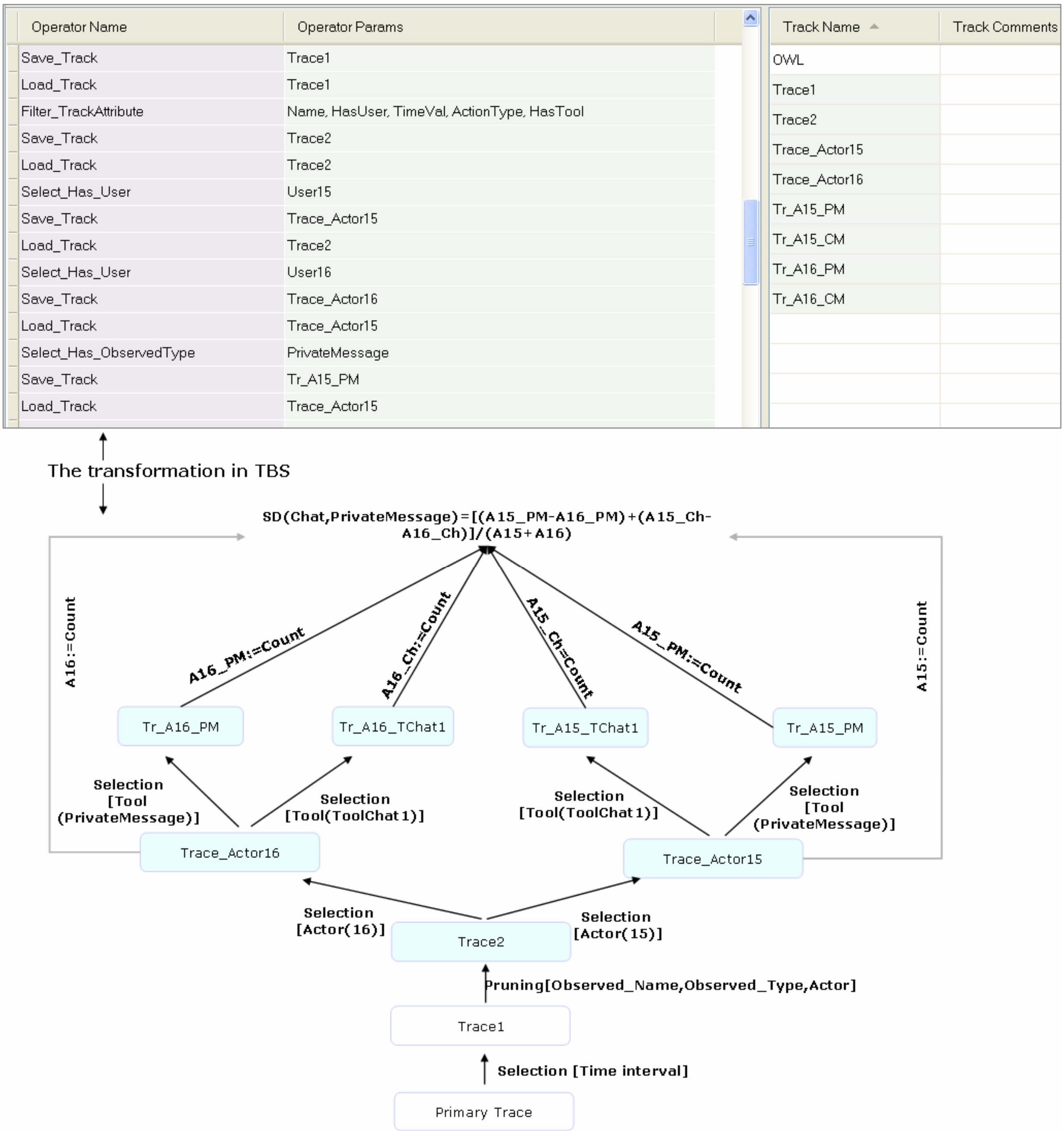

Figure 5: Example of a transformation sequence to calculate the division of labor (actors 15 and 16) on two tools: "ToolChat1" and "Private Message"

We propose to build the calculus rule in a toolbox taking the transformed traces as input. This toolbox offers arithmetic operators; visual forms "Pies and histograms"; and calls the Count operator. Figure 6 shows the calculus rule implementation for the indicator Proportion.

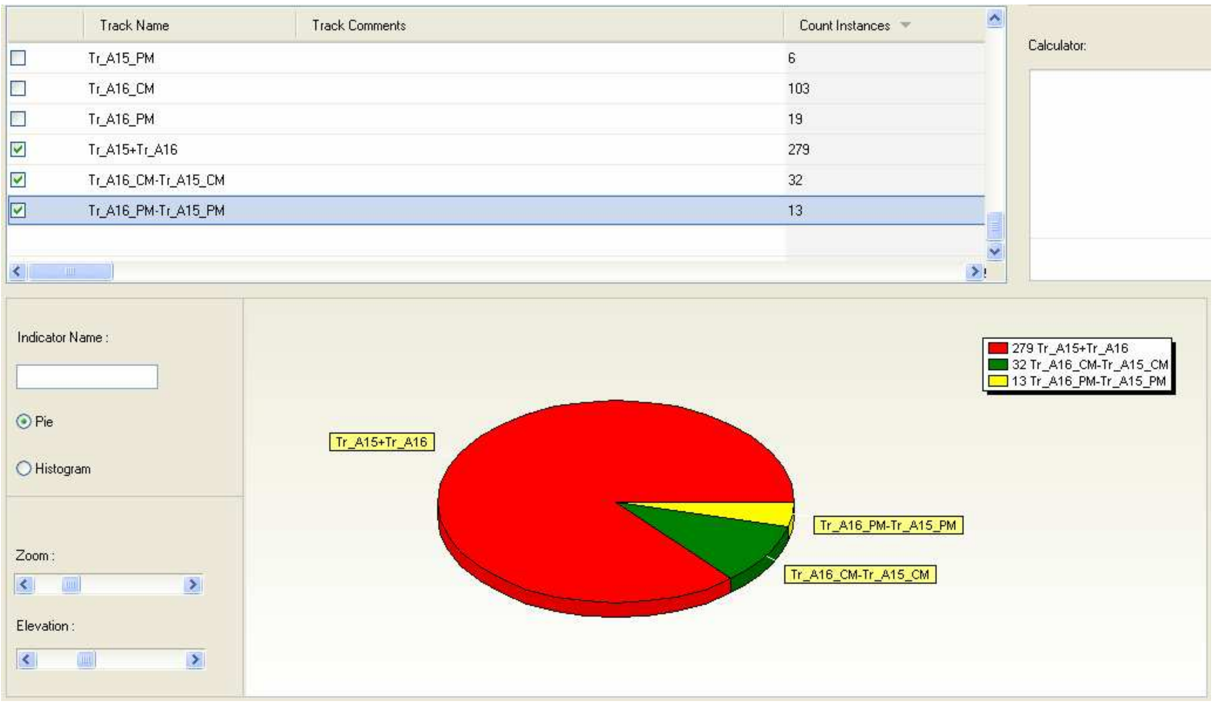

Figure 6: The "Division of Labor" calculus

4. Conclusion

We presented in this paper, a method and a tool to describe and calculate collaborative indicators when using a Trace-Based System. The method we propose is based on an MDE approach (trace models), and helps to build transformation sequences of models in order to calculate collaboration indicators. This method uses Trace-Based System to manage the traces we need to calculate indicators, and especially to manage models reuse at the design and calculation steps. We propose architecture and a tool to build and manage interaction's traces used to calculate indicators. It's essential and crucial for the proposed engineering to support the modelling process by facilitating the reuse of models for observables' collecting as well as for the indicators' calculus themselves.

The user can then reuse transformations' sequences to calculate the same indicators for other learning platforms, to ensure independence between the learning platform and how to calculate the collaborative activity indicator. The use and reuse of transformations' sequences applied on the M-Trace is the originality of our proposed method. This method applies also to social systems that involve a big number of participants, as well as for collaborative activities in small groups (2-3 persons).

In future works, our implemented TBS and the models transformations library, will be tested by independent researchers with other learning platforms. We also expect to add more transformation operators to give more richness to the transformation mechanism proposed by our method, particularly the rewriting operator.

5. References

[1] Dimitracopoulou A.. "State of the art of interaction Analysis: Interaction Analysis Indicators. Interaction and Collaboration Analaysis supporting Teachers' and Students' Self-regulation. (ICALTS)", JEIRP Deliverable D.26.1.1. Kaleidoscope NoE, pp. 153, December 2004.

[2] Reffay C., and Chanier T.. "How social network analysis can help to measure cohesion in collaborative distance-learning", Procs. Computer Supported Collaborative Learning Conference (CSCL'2003), Bergen, Norway, pp. 343-352, June 2003.

[3] Jaillet A.."Peut-on repérer les effets de l'apprentissage collaboratif à distance", Distances et savoirs, vol. 3, no. 1, pp. 49-66; 2005.

[4] Santos O. C., Rodríguez A., Gaudioso E., and Boticario J. G.. "Helping the tutor to manage a collaborative task in a web-based learning environment", Communication in the Workshop Towards Intelligent Learning Management Systems, Sydney, Australia, p.72-81, 2003.

[5] Martínez A., Dimitriadis Y., Gómez E., Rubia B., and de la Fuente P.. "Combining qualitative and social network analysis for the study of classroom social interactions", Computers and Education, vol. 41, no. 4, pp.353-368, 2003.

[6] Tedesco P. A.. "MArCo: Building an Artificial Conflict mediator to Support Group Planning Interactions", International Journal of Artificial Intelligence in Education, vol 13, pp.117-155, 2003.
[7] Lavallard A.. "Exploration interactive d'archives de forums : Le cas des jeux de rôle en ligne", Doctorate thesis, Université de Caen, Juillet 2008.
[8] Settouti L. S., Prié Y., Marty J. C., and Mille A.. "A Trace-Based System for Technology-Enhanced Learning Systems Personalisation", Proc. The 9th IEEE International Conference on Advanced Learning Technologies, Riga, Latvia. 2009.
[9] Moodle. Modular Object-Oriented Dynamic Learning Environment. http://docs.moodle.org/fr/Accueil
[10] Djouad T.. "Analyser l'activité d'apprentissage collaboratif : Une approche par transformations spécialisées de traces d'interactions", Proc. 2ième rencontre des jeunes chercheurs RJC-EIAH08, Lille, France, pp. 93-98,2008.
[11] Jena. A Semantic Web Framework for Java, http://jena.sourceforge.net/
[12] Djouad T., Mille A., Reffay C., and Benmohamed M.. "Calcul des indicateurs collaboratifs à partir des transformations spécialisées dans un SBT", Researsh report 2008. Disponible sur Internet : http://liris.cnrs.fr/~tdjouad/WebSite/rapport.pdf
[13] Mille A., Fuchs B., and Chiron B.. "Raisonnement fondé sur l'expérience : un nouveau paradigme en supervision industrielle", Revue d'intelligence artificielle, vol. 13, pp.97-128, 1999.