A Multi-Agent System to Implement a Collaborative Learning Method

Hanaa Mazyad, Insaf Tnazefti-Kerkeni and Henri Basson
Laboratoire d'Informatique, Signal et Image de la Côte d'Opale
Université de Lille Nord de France
Calais, France
e-mail: {mazyad, kerkeni,basson}@lisic.univ-littoral.fr

Abstract— **This paper presents a multi-agent system with Peer-to-Peer architecture that implements a pedagogic method called MAETIC. Indeed, the system objective is to allow the collaborative learning of project management. However, collaborative e-learning imply new roles for tutors as well as for learners. It is therefore essential to identify the users' needs and integrate, in the system, the functionalities which allow satisfying such needs. In addition, it is essential to avoid the failure and/or desertion of learners by providing tutors and learners with the opportunity to obtain information about the progress of their learning processes as well as the level of collaboration and sociability of each learner in the group**

Keywords- Collaborative learning; Multi-agent systems; Peer-to-Peer systems

I. INTRODUCTION

The Information and Communication Technologies in Education (ICTE) allowed users dispersed geographically to work and thus maximize the creativity and efficiency of group learning. However, the effectiveness and efficiency of collaborative learning depend on the motivation of its members to collaborate, on the number of members, on the time they can consecrate to this work and on their skills. Thus, the responsibility becomes collective.

In this paper, we are interested in the field of collaborative learning and especially on a teaching method called MAETIC (from french « Méthode pédAgogique InstrumentEe par les TIC), equipped with ICT (Information and Communication Technologies), aiming at developing a project-based learning pedagogy [1]. The main purpose of the deployed pedagogy is to provide real support to the management of project-based learning by group of students. For this, a learning platform has been established to offer students a support for the development of their projects and monitoring by the teacher. However, the services offered by the system suffer from a lack of support tools for managing the archiving of interactions in the system. Indeed, in the field of collaborative learning, management of traces is important since it allows analyzing the collected information concerning the learner or group of learners and provides the tutor with accurate and adequate information for his needs on the individual and collective evolution of learning.

In this paper, we describe a multi-agent system (MAS) for the collaborative learning of project management. This system implements the MAETIC method and provides users with functionalities to keep a detailed history of all the actions of the groups when they access the system in order to assess the group's life and its evolution.

This paper is organized in 6 sections. In Section II, the MAETIC method is briefly introduced. Some related work is presented in Section III. Then, in Section IV, an agent-oriented collaborative learning system that implements MAETIC is proposed. In Section V, some obtained results are presented. Finally, conclusion and some perspectives are given in Section VI.

II. MAETIC METHOD

This section describes the MAETIC method and presents the needs to be taken into account in order to implement this method. In addition, it defends the use of multi-agents and Peer-to-Peer (P2P) systems by presenting their contribution.

A. Description of the method

MAETIC is a pedagogic method instrumented by the ICT (Information and Communication Technologies). It is a teaching method which, as part of pedagogy for project-based learning, describes a set of formalized and applied procedures according to defined principles. Thus, the objective of MAETIC is to allow students to develop requested knowledge and skills. For the teachers, MAETIC's objective is to promote the establishment of a process that facilitates their educational activities.

MAETIC is based on five stages commonly adopted in the process of project management [2]: the initialization, the preparation, the planning, the project monitoring and the revenue (Figure 1). Each stage establishes activities, requires the production of one or more deliverables, and takes place over one or several sessions. Since the work is collective, MAETIC advocates the establishment of an organization in the group project (description of roles) that promotes the acquisition or strengthening of transversal skills needed for teamwork. The fact of making the group produce deliverable develops qualities related to the written production.

Thus, each project team must establish its weblog. This weblog aims to describe the life of the project. Besides the general information on the project (subject, members), it is responsible of storing all the notes concerning the project's life and is also responsible for collecting developed deliverables.

 ISBN: 978-1-61208-280-6

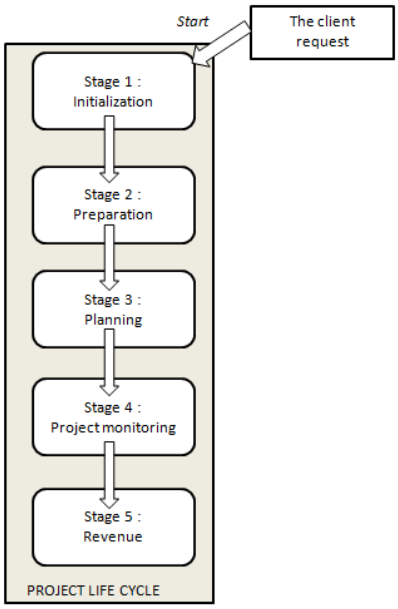

Figure 1. The five stages of MAETIC method

B. *Platform needs and contribution of multi-agents system*

MAETIC, as we have seen, is primarily concerned with implementing a project-based learning pedagogy to promote the learning of knowledge and skills that are essential to exercise a profession. Communication between the different actors: the director, teachers and learners in the initial platform is via the Weblog. Weblog technology was chosen for its maneuverability. The establishment phase of a weblog is very short and its use is accessible to all. In addition, weblogs can create a social bond with the students and seems to facilitate the writing of learners through the "posts" [3]. However, this technology provides a hardware consequent, very easy to collect but more difficult to analyze. The time spent by the teacher to monitor and analyze the activities of the learner is higher than that spent in the classical classes.

In addition, in the context of e-Learning, it is essential to avoid the failure and/or desertion of learners, to propose for tutors automatic tools, with assistance and decision support. These tools help to keep traces of all interactions related to students belonging to a given group and report the indicators of progression status in this learners group and its durability.

In a collaborative learning system, each member must manage and exchange his knowledge and cooperate with others in order to achieve his goals. Compared to these aspects, a Peer-to-peer system is particularly suitable to develop a collaborative learning system due to the following capabilities:

- It supports autonomy: every member of the system is seen as a peer that manages and has control over a set of local technologies, applications and services;
- It is decentralized: the community of peers is able to achieve its goal independently from any specific member or component;
- It is cooperative: in order to join and use the system, each member must provide resources or services to the others;
- It is dynamic: peers and resources can be added or removed at any time.

The multi-agent system is also an appropriate framework for realizing a P2P applications. The characteristics that they have, especially (a) their capability to allow the sharing or distribution of knowledge, and (b) that they assemble a set of agents and coordinate their actions in an environment to accomplish a common goal, are needed in this P2P application.

III. RELATED WORK

Several works on collaborative e-learning exist in the literature. Formid [4] for example, is an object-oriented platform with client-server architecture. This platform provides teachers with information about the realization of students' activities. However, we are interested by agent-oriented work. So, we review the approaches presented in the literature for providing multi-agents platforms for collaborative learning.

In the field of Artificial Intelligence and education, several approaches have been developed. For example, Guizzardi et al. [5] have developed a Peer-To-Peer system called "Help & Learn". This system was modeled using an agent-oriented language called AORML [6]. It is an open, centralized system that is designed to support the extra-class interactions between students and teachers. "Help & Learn" is limited to providing assistance to students who request it. Other systems have been developed. Fougeres and Ospina [7] have proposed a based-agent mediation system for the project management platform called iPédagogique. This system, modeled in AUML, serves as an interface between the human and the application to enhance their relationship and is used to promote collaboration among users. None of these two systems cares about monitoring learning and therefore, cannot trace user activities. Another system was proposed recently called I-MINDS [8]. This system includes a teacher agent for supporting the instructor. The teacher agent allows the instructor to interact with students, manage questions and answers (Q&A) sessions, administer quizzes, post evaluations, form groups and monitor individual and group performances. For computer-supported collaborative learning, I-MINDS provides a student agent for each student. Each student agent monitors and models its user. I-MINDS is interested in monitoring learning. However, it is not sufficiently independent and does not start up alerts to prevent teachers if there is a problem with a student or group. In addition, it does not give information about learners and groups collaborative level.

IV. TOWARDS AN AGENT-ORIENTED COLLABORATIVE LEARNING SYSTEM

The agentification of MAETIC method needs the use of a methodology. Several methodologies were proposed for the

 ISBN: 978-1-61208-280-6

development of MAS. We will follow Aalaadin methodology [9]. Aalaadin supports:

- Distributed systems where their members are geographically distributed.
- Opened systems that allow the integration of new members or the departure of a member.
- Complexity of elaborated systems.
- Autonomy of agents.
- Cooperative behavior of agents to achieve their goals.

A. *The system modeling*

Aalaadin is an organizational method developed by Gutknecht and Ferber. It is, first, a background for developing multi-agent systems, providing methodological guidance and secondly, a prototyping and running environment for agents based on notions of group and role through the AGR (Agent/Group/Role) model.

The AGR model of Aalaadin methodology is based on three primitive concepts, Agent, Group and Role that are structurally connected and cannot be defined by other primitives.

Agent: an agent is an active and communicating entity that plays roles within groups. An agent may hold multiple roles, and may be a member of several groups. An important characteristic of the AGR model is that no constraints are placed upon the architecture of an agent or about its mental capabilities. Thus, an agent may be reactive as an ant, or clever as a human.

Group: a group is a set of agents sharing some common characteristics. A group is used as a context for a pattern of activities, and is used for partitioning organizations. Two agents may communicate if and only if they belong to the same group, but an agent may belong to several groups. This feature will allow the definition of organizational structures.

Role: the role is the abstract representation of a functional position of an agent in a group. An agent must play a role in a group, but an agent may play several roles. Roles are local to groups, and a role must be requested by an agent. A role may be played by several agents.

1) The analysis phase : This phase identifies the system functions and dependencies within the identified communities. The functionalities of the system have been organized into the following objectives: a) Provide learners and tutors with an environment for project-based learning with tools allowing collaborative learning; b) Support the learners and help them in the achievement of their project by providing them with functionalities for monitoring their activities and those of the group ; c) Analyze the information collected about the learner or group of learners; d) Provide tutors with accurate and adequate information for their own needs on the individual and collective evolution of learning; e) Determine the level of productivity of each learner in terms of achieving pedagogic activities, as well as his level of communication with other members of the group; f) Determine the progress of each group regarding to the achievement of activities but also to the present, absent, inactive members; g) Determine the levels of activities achievement by all the groups and adjust the calendar if necessary; h) Provide users with information about the use of collaborative tools; i) Determine the collaborative work level in each group; j) Enable learners to also have access to this information which supposed to empower them, show their performance and enable them to compare this performance with other learners in the same group.

2) The design phase : This section presents the system design. Indeed, to design the system, the functionalities listed in the analysis phase will be taken into account to design the system groups' structure. In addition, the requirements of P2P systems will be followed to design databases.

a) Databases Design : P2P systems require from each peer to be client and server at the same time. Thus, there is no centralized server which contains all data and files of each user. Thus, in the proposed system, each user will have its local database (L_DB) containing his data. The user must

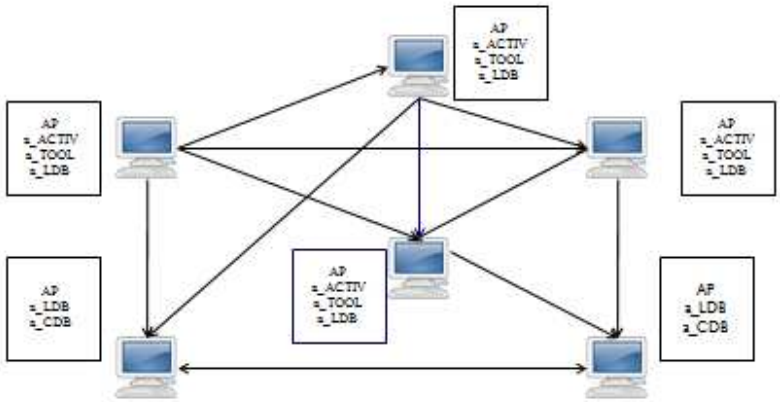

Figure 2. Agents deployment

allow other users of his group to have access to this database in order to recuperate files or data. However, to enable the user to know the list of his group users, a centralized entity is used. Indeed, a hybrid architecture is adopted which means that user machines are connected to a machine called super-peer that plays the role of a server containing a directory. This directory contains the user information (username, IP address, etc.). However, this central database will not contain user files. On the contrary, file and data exchange will be based on the P2P model where user machines are both client and server.

b) Groups Design : There are two group structures in our system: the tutors' group structure and the learners' group structure. The tutors' group structure has the roles played by tutors: a) Guide the learners and adjust the pedagogic scenario; b) Encourage collaborative work; c) Motivate struggling learners. The learners' group contains the roles below:

- Helper: giving help to learners;
- Help consumer: asking questions to other learners;
- Author: upload documents to give it to tutors
- Learner: download documents given by the tutors

Each learner and tutor has a personal assistant agent (AP). The "AP" agent plays roles enumerated above and executes all the request of the user.

To collect and manage traces, we need to define the following roles:

- Supervisor of activity
- Supervisor of tools
- Local database manager (L_DB)
- Central database manager (C_DB).

We attribute each one of these roles to one agent:

- a_ACTIV: supervises users' activities during a session. It provides statistics concerning the progression of each activity. It reminds learners about deadlines and notifies the late groups by sending alerts.
- a_TOOL: supervises the use of tools. It provides statistics about the use of collaborative tools (Email, Chat, Forum, etc.). In fact, the "a_TOOL" agent saves in the "L_DB" of each user, tools that he used;
- a_LDB: manage the interactions between system agents and the "L_DB" and between "L_DB" and "C_DB".
- a_CDB: manages the central database.

Figure 2 shows the agents deployment. The agent "a_CDB" is a centralized entity which exists only in the tutor's space. It should be noted that the agents "a_ACTIV" and "a_TOOL" do not exist in the tutor's space due to the fact that the system does not collect their information.

3) The realization phase : We have chosen to deploy the system on the multi-agents platform Madkit [10] [11]. This choice is taken due to the fact that MadKit is intended for the development and the execution of multi-agents systems and more particularly for multi-agents systems based on organizational criteria (groups and roles). However, MadKit does not impose any particular architecture to the agents. MadKit communication is based on a peer to peer mechanism, and allows developers to quickly develop distributed applications using multi-agents principles. Concerning the database management, Java has the Java

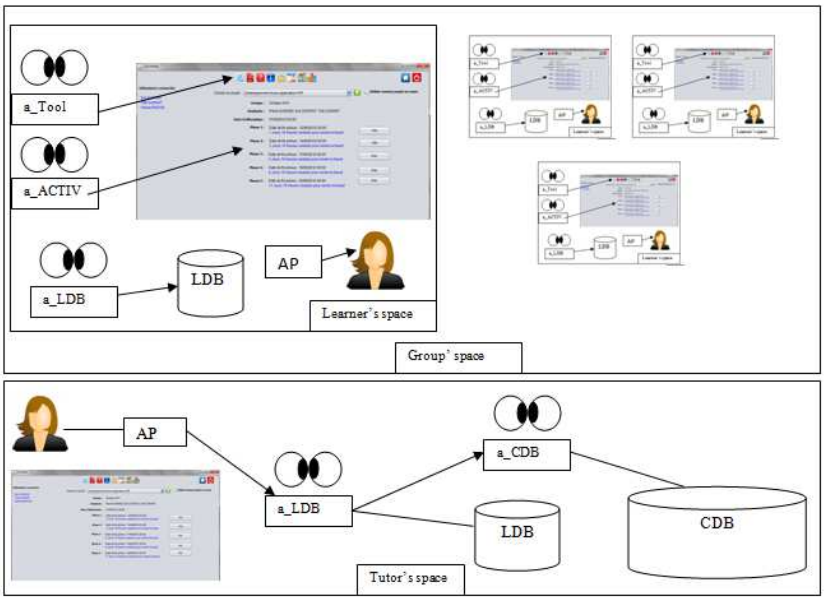

Figure 3. The system agents in their respective space

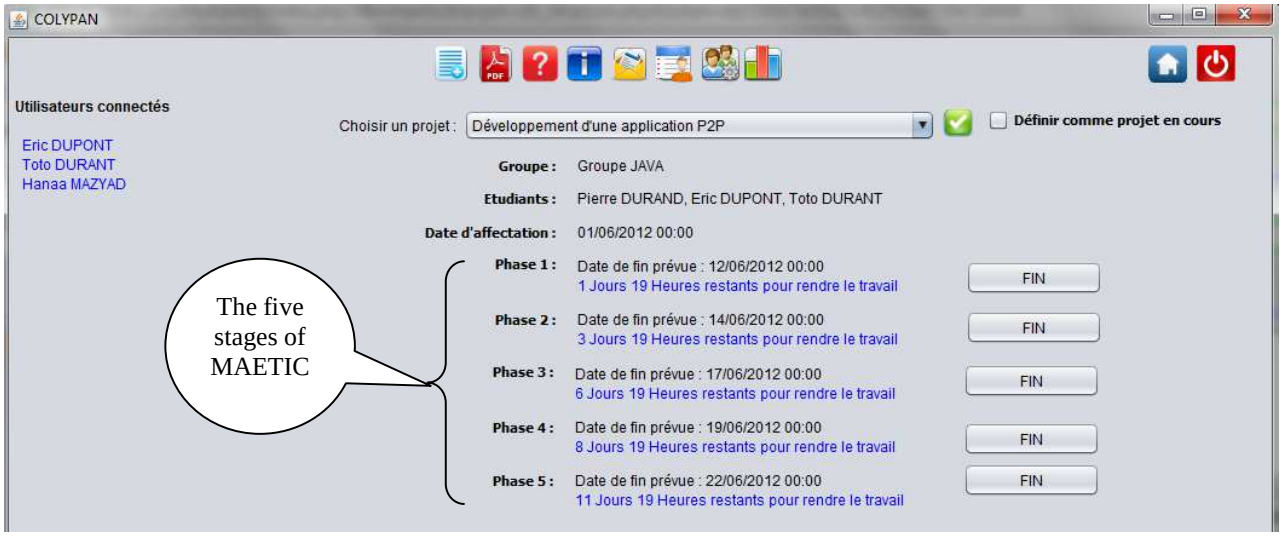

Figure 4. The graphic interface

Database Connectivity (JDBC) API that allows the connection to databases and is independent of database management system (DBMS). MySQL is chosen as DBMS because it is a popular server used for free. It runs on most operating systems and is often used in conjunction with Java.

As the hybrid model of P2P requires, our system consists of peers connected to a super-peer. Thus, machines with low bandwidth are called peers. Machines with good bandwidth are super-peers. Super-peers play the role of localization server. This model has several advantages

- It decreases of the number of connections on each server which helps in avoiding bandwidth problems.
- It uses a mechanism of P2P decentralized model to update a client directory and files indexes from information coming from other servers. Thus, a server can provide any client with all the information on the network.
- It allows identifying the system users which is essential in the learning context, while keeping the advantages of P2P systems.

In the proposed system, the system's central database is installed on the super-peer. However, at no time user files will be stored on the server. Indeed, the server can localize these files and users can get them from the machine where they are stored in a shared folder and accessible remotely. In addition to this central database, each user has on his machine a local database that contains information about him and the data he owns. Users can access this local database as long as the owner allows them.

We created our agents using the Madkit agent « Designer ». It is a tool designed to facilitate the agent creation and launch. However, this agent does not require any running concept. In fact, it only provides methods to start the life cycle of an agent (Activate, live, end) and the user must program its behavior.

Figure 3 shows the agents interacting in users (Tutor, Learner) space. In each learner's space, "a_ACTIV" agent collects information about the learner activities and "a_TOOL" agent collects information about the use of tools in this space. Then, these two agents send the collected information to the "a_LDB" agent. Finally, "a_LDB" agent sends statistics based on this collected information to the "a_CDB" agent. In the system, these statistics will be sent to the "a_CDB" agent each 15 minutes as long as the learner is connected.

V. OBTAINED SYSTEM

Our system is a collaborative learning system that implements the MAETIC method. It aims to enable collaborative learning of project management. In this system, the tutor's role is to guide learners and facilitate their learning. The group's role is to motivate learners and create a social link that prevents the feeling of isolation. Learners learn by interacting with others. In addition, agents interact and communicate to achieve a common goal: satisfy the users (learner or tutor) requirements. Indeed, this system provides tutors and learners with information on group progress and on collaboration and sociability levels of each learner, and allows each learner to perceive his own learning situation. In addition, it is independent and proactive. It scrutinizes periodically the data of interaction and triggers alerts to prevent the tutor and the learner when the group is at risk of bursting or is experiencing educational failure.

Figure 4 shows the system's graphic interface. Firstly, learner should be a member of at least one group to be able to work on a project. Secondly, learner chooses the project that he wants to work on and then, he can start his learning process and system's agents start collecting information. In

 ISBN: 978-1-61208-280-6

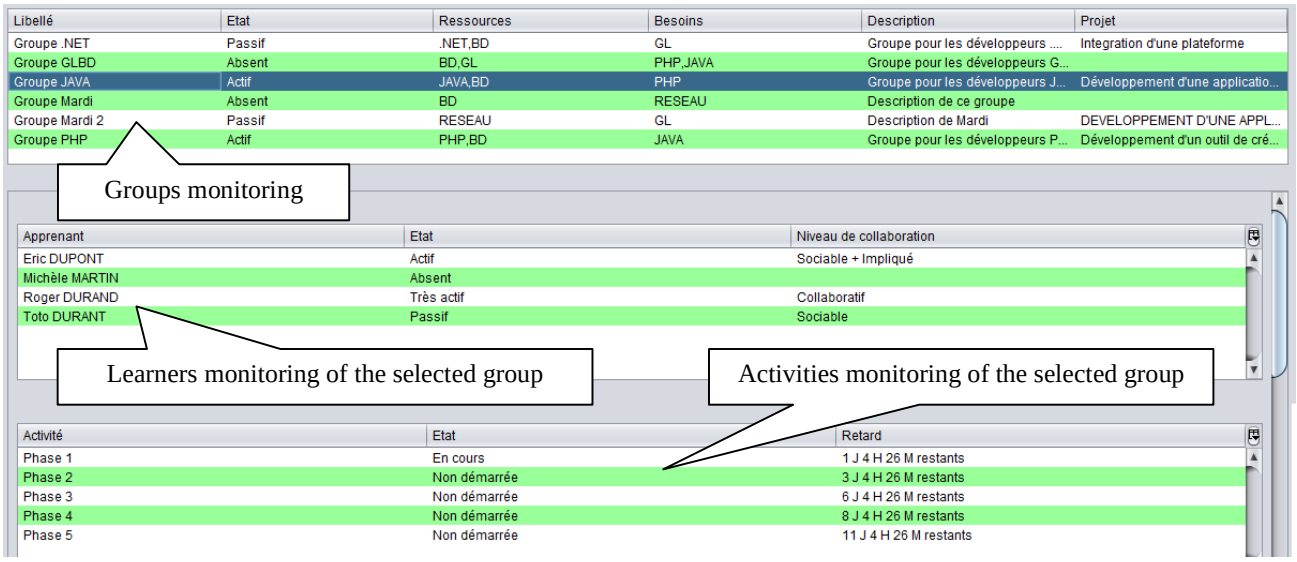

Figure 5. The group's monitoring

the system, learner can be member of several groups but in each group, he works on a single and different project.

Figure 5 shows the group monitoring. It presents information about each member of the group but also about the realization of group's activities.

VI. CONCLUSION

In this paper, we are interested in the design and the realization of a system that implements the MAETIC method. We have chosen to use multi-agents systems with P2P architecture to realize this collaborative e-learning system. In this system, each learner contributes in the learning process of the group, and in return, the group contributes in the learning process of its members. The consistency of the whole group allows achieving the goal. However, collaborative e-learning implies new roles for tutors as well as for learners. Thus, the user's needs are identified and functionalities which allow satisfying such needs are integrated into the system. In addition, this system provides tutors and learners with the opportunity to obtain information about the progress of their learning processes as well as the level of collaboration and sociability of each learner in the group.

At present, several points need to be explored. Without being exhaustive, the e-learning is a multidisciplinary field ranging from the social sciences to the hard sciences, in which users with heterogeneous profile interact. We think it will be interesting in the field of collaborative learning to propose analysis methods of chat and forums content.

REFERENCES

[1] D. Leclet and B. Talon, "Assessment of a Method for Designing E-Learning Devices". Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications, ED-MEDIA, Vienna, Austria, AACE/ Springer-Verlag (Ed.), 2008, pp 1-8.

[2] H. Marchat, Kit de conduite de projet. Paris: Organization editions, 2001.

[3] S. Fiedler, "Personal Web publishing practices and conversational learning". Symposium on Introducing disruptive technologies for learning:. ED-MEDIA Lugano, 2004, pp. 2584-2591.

[4] V. Guéraud and J-M. Cagnat, "Automatic Semantic Activity Monitoring of Distance learners Guided by Pedagogical Scenarios", Innovative Approaches for Learning and Knowledge Sharing, EC-TEL 06, Lecture Notes in Computer Science LNCS 4227, Springer 2006, pp.

[5] R. Guizzardi-Silva, LM. Aroyo and G. Wagner, "Help&Learn: A peer-to-peer architecture to support knowledge management in collaborative learning communities". Revista Brasileira de Informatica na Educação, 12 (1),2004, pp. 29-36.

[6] G. Wagner, "The Agent-Object-Relationship Meta-Model: Towards a Unified View of State and Behavior". Information Systems 28:5, 2003, pp. 475–504.

[7] A.J. Fougères, "Vers un Système de médiation pour les systèmes coopératifs ". Mémoire d'habilitation à diriger des recherches, Université de Technologie de Belfort-Montbéliard, 2010, pp. 76-78.

[8] N. Khandaker, L. -K. Kiat, L. D. Miller and H. Jiang, "Lessons Learned from comprehensive deployments of multi-agent CSCL applications I-MINDS and classroom wiki". IEEE Transactions on Learning Technologies, Volume 4, Number 1 (TLT4(1)), January 2011, pp. 47-58.

[9] J. Ferber and O. Gutknecht, "Aalaadin: a meta-model for the analysis and design of organizations in multi-agent systems ". In : Demazeau, Y., éditeur : 3rd International Conference on Multi-Agent Systems, Paris, IEEE, 1998, pp. 128–135.

[10] O. Gutknecht and J. Ferber, " Madkit : a generic multi-agent platform". Proceedings of the fourth international conference on autonomous agents, 2000, pp. 78-79.

[11] www.madkit.org

 ISBN: 978-1-61208-280-6