PUC

ISSN 0103-9741

Monografias em Ciência da Computação
n° 52/03

Supporting Groups and Group Formation through Software Agents in a Web based Learning Environment

Leonardo Magela Cunha
Hugo Fuks
Carlos José Pereira de Lucena

Departamento de Informática

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO
RUA MARQUÊS DE SÃO VICENTE, 225 - CEP 22453-900
RIO DE JANEIRO - BRASIL

PUC RIO - DEPARTAMENTO DE INFORMÁTICA ISSN 0103-9741

Monografias em Ciência da Computação, Nº 52/03

Editor: Carlos J. P. Lucena Novembro, 2003

Supporting Groups and Group Formation through Software Agents in a Web based Learning Environment

Leonardo Magela Cunha

Hugo Fuks

Carlos José Pereira de Lucena

Trabalho patrocinado pelo Ministério de Ciência e Tecnologia da Presidência da República Federativa do Brasil

Supporting Groups and Group Formation through Software Agents in a Web based Learning Environment

Leonardo Magela Cunha

Hugo Fuks

Carlos José Pereira de Lucena

{leocunha, hugo, lucena}@inf.puc-rio.br

Monografias em Ciência da Computação, Nº 52/03

Editor: Carlos J. P. Lucena Novembro, 2003

Abstract:

In this work, it is presented how an environment for distance learning, AulaNet, was prepared to support groups. Besides that, software agents were used to help teachers in setting groups of learners. The complexity of work and the dissemination of the information and communication technologies value and make group work a potential allied for the educational or commercial organizations. The AulaNet supports learning in groups and can also be used to support group work, so it is a groupware system that is based on the research from Computer Supported Cooperative Work. In Software Engineering, multi-agent systems provide a properly level of abstraction for the treatment of complex and distributed problems as those characterized by distributed environments such as AulaNet.

Keywords:

Group Formation, Groupware, Software agents, Distance Learning.

Resumo:

Em Engenharia de Software a utilização de sistemas multi-agentes possibilita um nível de abstração mais adequado para o tratamento de problemas complexos e distribuídos. Um exemplo é o caracterizado pelos ambientes de trabalho e aprendizagem em grupo. Atualmente a complexidade do trabalho e a disseminação das tecnologias da informação e comunicação valorizam e potencializam o trabalho em grupo. O apoio computacional fornecido para este trabalho, denominado groupware, baseia-se na pesquisa de Computer Supported Cooperative Work. Este trabalho apresenta o estudo e a forma como foi implementado um sistema multi-agentes para o auxílio à formação de grupos no ambiente AulaNet.

Palavras-chave:

Formação de grupos, Groupware, Agentes de software, Educação a Distância

1. Introduction

With the dissemination of computational environments to support work, investigations have been carried out about how to assist workers and learners in order to produce more and improve the quality of what they do. Another objective of these environments is to make it possible to record activities, generating a type of work memory to assist both organizations as well as individuals. Group work is one of the major demands of organizations today. Most of them want their members to be capable of exercising their social skills within the work environment in order to achieve greater or better job results.

In order to support individual work through the Web, it seems to be sufficient to offer the proper training and tools. But how would it be possible to work in a group? It is not enough just to offer the training and the tools and leave the individuals to their own devices. It is necessary to support them and provide the assistance that allows them to find other individuals with whom they can communicate. Based upon this communication, commitments emerge that must be complied with. The individuals will interact, cooperating or competing, and thus the need for the coordination of such interaction - that is, group work - arises.

Since it is teaching institutions that must prepare individuals for the job market, they require mechanisms that are capable of reflecting what happens in real life. Thus, it is possible to imagine that learners also need to find peers or colleagues in order to communicate between themselves, to subsequently interact and coordinate in order to learn.

Group learning is not a new idea but with the dissemination of computer technology networks a new vigor has emerged in the field. There is a great variety of new applications and possibilities, although one of the major challenges is the transition period required for transporting the applications, methods, methodologies and techniques of the real world to the virtual world. This is where the work in the fields of Computer Supported Cooperative Work (CSCW) and Software Engineering join up. Through the studies for group work and the methods for developing software it is possible to model and design software that is proper for supporting groups which is called groupware.

Reproducing the interaction that occurs in the workplace of the connected society is one of the ways of preparing learners for this workplace. The first objective of this paper is to show how support for groups was implemented in the AulaNet environment (Lucena, Fuks, Milidiú, Laufer, Blois, Choren *et al.*, 1999), which is a groupware developed for Web-based teaching and learning.

The access to large quantities of information distributed by the networks and the complexity of the real world has quickly changed how people learn. Agent technology has been presented as a promising strategy to be applied to the current challenges of modern educational environments that are increasingly more influenced by technologies such as the Internet and Artificial Intelligence. With the emergence of Distributed Artificial Intelligence (Weiss, 1999) it can be seen the strengthening of the software agents concepts and also a redirecting of these concepts towards Software Engineering (Jennings & Wooldridge, 2000).

According to Weiss (1999), one of the reasons for the use of Distributed Artificial Intelligence is the capacity of multi-agent systems to exercise a fundamental role in Computer Science, both in the future as well as in the present. Modern computing platforms and information environments are distributed, large, open and heterogeneous. These characteristics apply directly to the Web-based instruction environments. Software agents can influence different fields in educational systems. They supply new educational paradigms, support theories and can assist both students and teachers in the task of computer-assisted learning (Aroyo & Kommers, 1999).

Thus, the second objective of this paper is to present how a multi-agent system (MAS) was implemented for establishing groups of learners in classes during a course taught in the AulaNet environment.

This paper is structured as follows: the next section contains some considerations about groups. Next, we present how the AulaNet environment was redesigned in order to allow groups from different classes within a course to be used. Based upon this support, an investigation into software agent technologies was conducted and we show how a multi-agent system was used to support group formation within the environment. Some related work also is presented and based upon the comparisons with the MAS implemented, we came aware of future work. One future work that also is discussed is the creation of a federation of AulaNet servers, where the formation of groups with members (on different servers) of different classes within a course is possible. And last, final considerations about the work that was carried out are presented.

2. Working and Learning in a Group

In the last decade we have seen an explosion in the use of groups, mainly influenced by the success achieved by Japanese companies in the 1970s and 1980s through the effective use of small workgroups. Another factor that also contributed to this explosion was the large quantity of investigations carried out by U.S. and European researchers in the 1950s and 1960s. (Lipman-Blumen & Leavitt, 2001).

In the field of education, according to Barker and Barker (2001), the interest for groups stems from the 1960s. But the first group studies appeared during the decade of the 1940s when researchers such as Lewin (1948) examined groups departing from the behavior of their members. The work of Vygotsky (1987) impelled the use of groups since it presented theories which grounded the experience that through discussion there is consolidation of knowledge and the discovery of new solutions.

In their book, *Peopleware*, DeMarco and Lister (1999) present information about how software developers use their time in different work modes. On a typical day, 50% of their time is spent working with one other person, 30% is spent working by themselves and the remaining 20% is spent with two or more other people.

Some of the advantages that can be gained through working in a group are developing and improving individual skills for the use of knowledge, accepting responsibilities for individual and group learning, developing the capacity to reflect about one's own suppositions by submitting ideas to the group and developing social and group skills.

Besides these advantages, there also are disadvantages, fears and even resistances that need to be overcome. In the field of education one notes a fear of not being able to cover the full course content, resistance of some learners in assuming a more active role, underdeveloped social and group skills and indecision about how to issue grades.

2.1. Some considerations about groups

One of the ways of distinguishing one group from another is to analyze their objective and subjective limits, just as people have their physical and psychological limitations (Jaques, 2000). The objective limits of a group are, for example, the duration and the space that will be reserved for that group's carrying out an activity. As highlighted by Lipman-Blumen and Leavitt (2001), a group does not last forever and the lifetime of a group can be defined by a number of factors, such as the departure of a member or the end of an activity, among many others.

The subjective limits can be exemplified by the limit of the activity which is being carried out by the group; that is, what the group should or should not do. Another example is the limit of entry of participants; that is, the process a candidate must undergo to become a member of the group. This limit can clearly be perceived when a new member arrives in a previously established group.

The size of a group is one of its objective limits and there are researchers who believe that the minimum number of members for group-like behavior to occur is three, although pairs of individuals also is a form of group that should be considered. For example, according to Kay (2001), pairs play an important role in learning, creating natural opportunities for learners to articulate their knowledge, to reflect and justify their actions. According to Jaques (2000) the size can influence other group characteristics, such as structure, organization, frequency of interaction between members and the need for sub-groups, among others.

The size of the group also influences how members "feel" the group. In a small group it is complicated to distinguish the feelings of individuals from the feelings of the group. However, in larger groups this distinction is clearer. Another aspect that also is influenced by the size of a group is the quantity of work produced by its members. For example, in discussion groups with many members the quantity of messages per member is not as large as in smaller groups, where it is necessary that each participant sends more messages for there to be a considerable volume of them.

A group can be evaluated by taking into account the objective limits as well as the subjective ones, but great caution still needs to be maintained to avoid lack of motivation. According to (Lipman-Blumen & Leavitt, 2001), evaluating individuals can awaken anger and frustration independently of the fact that the work of these individuals is associated with groups. However, when evaluation occurs in a group, the best thing to do would be to support, to encourage and to evaluate the group as a whole and let the group evaluate its members.

For Perkins (1993), based upon the point of view of distributed cognition, evaluating a group work through individual performance is as lacking in meaning as that of evaluating a painter without his brush. It is not rare that in a work environment the manager does not have suffcient resources for knowing who in a group works a lot and who is only "out for a stroll," neither can the teacher count on resources to know how much a learner in a work group is really doing. Thus, it is important to provide managers or teachers with measurement tools that apply in those cases when they possess some resource for identifying individual progress and for when this does not occur.

2.2. Collaborative Learning

The theory of multiple intelligence (Gardner, 1993) offers a more encompassing vision than the purely academic view about learning. According to Gardner, intelligence is the capacity to solve problems or to fashion products that are appreciated in one or more cultural environments.

The classification presented in the work cited above is composed of eight learning styles, and one of these is interpersonal intelligence. Interpersonal intelligence refers to the needs for interaction and sharing between one learner and the others. The learners benefit from the cooperative aspects of the work within a group and the other tasks that promote interpersonal relationships, both within and outside the classroom.

According to Dillenbourg (1999), there is no common definition about collaborative learning that is accepted by everyone. This occurs due to the indiscriminate use of the word “collaboration”, currently greatly in vogue but he defines collaborative learning as a *situation* in which *two or more* people *learn* or try to learn something *together*. Each element of this definition can be interpreted in different ways.

The number of learners can vary from pairs, to small groups and even large organizations containing thousands of people. The term “to learn” can mean everything from monitoring a course to learning from lifelong work practice. The form of interaction described by the term "together" can mean face-to-face interaction or computer mediated, asynchronous or synchronous, that occurs frequently or not, that is carried out through joint effort or where work is divided up.

According to the concept of collaborative learning, Dillenbourg defines three dimensions within the space for this type of learning. These consist of the variety of the scales and the meaning of learning and collaboration.

The first of the dimensions directly influences the scalability of research since empirical results obtained from small groups that are learning just a few topics cannot be generalized for large groups working within a course, and vice-versa. The variety of the scales also can be seen in the view of the individual as a distributed cognitive system, while the group is seen by Distributed Cognition as a single system (Minsky, 1987). It is strange to speak about collaboration of one with itself, but the ideas of Piaget and Vygotsky (1987) that thinking is the result of internalized dialogues makes this view less chaotic and plausible.

For Dillenbourg the variety of the uses of the word “learning” leads to two distinct understandings of collaborative learning. One is the pedagogical method that says that two or more individuals should collaborate and it is expected that they will learn. The other is the psychological process where individuals are observed and collaboration is seen as a mechanism that causes learning. The confusion between these two types of learning can lead to overstatements about the effectiveness of collaborative learning.

Thus, the words “collaborative learning” describe a situation in which particular forms of interaction between two people are expected and unleash learning mechanisms, but there is no guarantee that this will occur. So it is necessary to increase the probability that some type of interaction takes place, which can be achieved in different ways.

The first is to supply the initial conditions, carefully designing the situation so that there is a greater probability of interaction. A second manner is to overvalue the “collaboration contract” through a scene setting based on role playing, such as creating discussions where groups of learners defend different points of view, even not being their own.

The third way is to support more productive interaction within the learning environment (assisted by computer). In a face-to-face learning session it is common to decide upon a discussion topic. In Computer Supported Cooperative Learning (CSCL) it is possible to reinforce these communication mechanisms by structuring and categorizing, as presented by Gerosa, Fuks and Lucena (2001). Some of the advantages of using these strategies are greater depth to the discussion and a reduction in information overload.

Finally, the fourth way would be to monitor and regulate the interaction. These strategies could be used by the teachers to offer tips or counseling regarding the direction being taken by work groups, thus facilitating learning.

However, the last dimension within collaborative learning is the significance of the term “collaboration.” According to Dillenbourg (1999), collaboration is related to four different aspects of learning: situation, interaction, mechanisms and the effects of collaborative learning. A situation can be more or less collaborative; for example, it is easier for collaboration to occur between colleagues than between a subordinate and his or her boss. Interaction also has different levels of collaboration; for example negotiation seems to be more collaborative than giving of orders. Some learning mechanisms are intrinsically more collaborative.

The last of the aspects (the effects of collaborative learning) is not used to define collaboration but, rather, is taken into account for corroborating to the terminological confusion in the field due to the different ways of evaluating collaborative learning. Thus, in order to understand collaborative learning it is necessary to understand the relationship between the four items presented. At first, the situation generates interaction standards; this interaction activates cognitive mechanisms that, for their part, generate cognitive effects. However, this linearity is a simplification, since the majority of relationships are reciprocal.

With the concepts of groups and collaborative learning in mind, the question that emerges is how to support and make interaction of groups possible through software. In order to answer this question, in the next section some concepts about software for group work are presented.

2.3. Groupware

Groupware is a type of software that supports interaction between individuals - that is, the interaction between the members of a work group to carry out a common objective. For Ellis and Wainer (1999), groupware is the technology of hardware and software that supports the interaction of groups. Khoshafian and Buckiewicz (1995) define groupware as a technology that refers to the vast fields of collaboration, human-computer interaction and human-human interaction through digital media, bringing about substantial improvements and changes to organizations. Baecker (1993) differentiates Groupware from Computer Supported Cooperative Work (CSCW), pointing to CSCW as the field of study for computer-assisted collaborative activities while groupware is software used to support CSCW. Ellis and Wainer (1999) share a similar view, where CSCW is the field of research that studies the use of communication and computation technologies to support group activities. Also according to them, CSCW has an entire area of discussion concerned about theories, frameworks and mathematical models.

Thus, CSCW includes the theoretical development of models of teams, organizations and social systems. We next present a collaboration model for group work and, subsequently, will provide a brief description of the IMS Enterprise specification and its relationship with the work groups of an organization.

Colaboration assisted by groupware mechanisms can be represented in the model presented in Figure 1 (Fuks, Gerosa & Lucena, 2002). This model is based on the Communication, Coordination and Cooperation triad (Ellis, Gibbs & Rein, 1991) (Borghoff & Schlichter, 2000); that is, in order to work in a collaborative manner an individual must share ideas (communicate), be in tune with the other members of the group (coordinate) and carry out actions in a satisfactory fashion in a shared workspace (cooperate) (Fuks, Laufer, Choren & Blois, 1999).

Figure 1 – The collaboration model

One sees in Figure 1 the occurrence of a cycle indicating that individuals should communicat in order to coordinate their work efforts and to cooperate towards achieving an objective. For cooperation, there is a need for communication, whether it be direct or through information obtained from the environment where the work occurred. In each relationship, there exists stimulus supplied by awareness information that makes it possible for shared understanding to take place regarding a task or the entire process (Fuks & Assis, 2001). Information overload (Fussel, Kraut, Lerch, Scherlis, Mcnally & Cadizz, 1998) is one of the undesired effects of collaboration and cannot be underestimated since its influence in the work can mean a quantitative or qualitative reduction in productivity.

2.4. The IMS Enterprise Specification

In organizations, one of the secrets for achieving good human-machine integration is to let workers decide what resources they need in order to make their work more productive, thus maximizing the overall performance of the company (Dertouzos, 2001).

A great number of organizations, educational or not, have systems for managing training, human resources, students, finances, libraries and others resources. Thus, it is to be expected that instruction systems can be integrated into these enterprise systems (IMS ENT, 2001).

Groups of academic, commercial, governmental and industrial consortia are collaborating in order to define the means of making inter-operability possible between products and educational content. The projects of the Instructional Management Systems Global Consortium (IMS) (IMS, 2001), Aviation Industry Computer-Based Training Committee (AICC) (AICC, 2001) and IEEE Learning Technology Standards Committee (IEEE, 2001) are the main ones (Silva, Lucena & Fuks, 2001).

One of the specifications proposed by the IMS is the IMS Enterprise, whose objective is to define a set of structures that can be used for the exchange of data between Internet-based systems for instructional management and other enterprise systems used to support the operations of an organization. These structures supply standardized data bindings that make it possible for software developers to create instructional management processes that inter-operate with other systems that have been independently developed (IMS ENT, 2001).

The conceptual view of the data model of the IMS Enterprise specification is supported through the use of three data objects. These are:

Person - contains elements describing an individual of interest to the instructional management environment;

Group - this object contains elements describing a group. There are a number of types of groups that can be shared between the enterprise systems and the instructional management systems, such as classes in a course or the (sub)groups within a class. A group also can have any number of relationships with the other groups;

Group Membership - contains elements that describe the participation of a person within a group. Members of a group can be instructors, learners, managers, administrators and others.

Based upon groupware concepts as well as a specification for the exchange of information about groups in an instructional management environment, it will be presented the AulaNet environment and how it was prepared to support groups.

3. The AulaNet Environment

AulaNet (Lucena *et al*., 1999) is a teaching and learning environment on the Web whose development has been carried out since June 1997 at the Software Engineering Laboratory (LES) of the Catholic University of Rio de Janeiro (PUC-Rio). AulaNet is based upon cooperative work relationships manifested through the interaction of learners with their instructors, with other learners and with didactic content.

3.1. The Structure of a Course

Figure 2 presents, using a diagram of UML classes, the structure of a courses within the AulaNet environment. Depending upon the configuration chosen by the coordinator at the moment of creation and/or updating of a course, didactic content is coupled to it, including “webliographies”, lessons, notices, bibliographies and documentation.

Most of the UML classes presented are mapped out in tables within the database. In the Content table, records are identified by name, a description and other sorting and presentation attributes. When there is a need for an extra attribute, such as is the case of Webliography where a URL (uniform resource locator) is necessary, the UML class Webliography inherit from Content and specialize it. In the database, this means that there is a table that references Content and merely incorporates the new attribute.

Figure 2 - Structure (UML class diagram) of a course in the AulaNet environment

Exams and Tasks also are coupled with the course. Thus, all of the classes within a course must take the same exam or resolve the same task. It was perceived through the use of the environment that this is common, although there is a need to apply different tasks and exams to different classes. An example of this is occurs in the course Software Systems Project of the PUC-Rio Computer Science Department. There are two classes in this course: one for undergraduate students and the other for post-graduate students. They make use of the same content but their deadlines and even their tasks are different.

A course is composed of one or more classes. And each class has messages associated with it from the Discussion Group (Discussion), Contact with the Teachers (TeacherMessageContact) and the Conferences (Interest).

Of the services that are available in the environment, some can not be seen in the diagram presented in Figure 2: Debate, Message for Participants, Teach Co-Authorship, Learner Co-Authorship, Follow-up Reports and Download.

The Debate service does not allow the storage of the chat sessions within the environment. However, the participants have available to them the possibility of receiving a copy of the debate transcript at an electronic address specified when they register on the environment. The participation of the members of the course in a debate is registered by the course mediators and stored according to the standard established to generate the Follow-up Reports (Fuks & Assis, 2001).

Regarding the Message for Participant service, no information is stored in the environment. This service has two operating modes, depending on whether the participant who a person wants to make contact is connected to the environment or not. In the former situation, communications occurs in real time between the participants through browser windows, much like when using ICQ software for instant messaging. In the latter case, the client of the electronic mail that is installed in the person's machine is invoked to send a message that will not be recorded in the environment.

The Teacher Co-Authoring service allows other teachers within the environment to help the teacher who is the coordinator of a course to create and maintain the course. In order to make it possible to identify authorship, there is an attribute in the majority of the elements of the course that a person has authored. For example, it is possible to determine who is responsible for a given didactic content or task.

The Learner Co-Authoring service allows learners to create content that subsequently will be validated by the coordinator of the course, and possibly incorporated into it. The coordinator also is permitted to grade the content created by the learners and this evaluation, as well as in the debate service, is conducted through the Follow-up Reports data standard.

3.2. Supporting Groups on AulaNet

On AulaNet it is possible to create a course, and a course can have different classes and their respective mediators. One need that has been noticed as a result of the use of the environment was the formation of groups within these classes - for example, to resolve a task, write an essay or design software among others.

In order to make it possible to support groups within the classes the use of the Follow-up Reports data structure was contemplated. This would make it possible to record the production of groups in activities such as tasks, exams and messages sent to the environment.

However, the need to maintain an individual record of the participation of the members of the group and to record which groups would execute what activities was seen to exist. The data structures cited above call for the recording of the learners' production but not for assigning an activity to a learner. Thus it was seen that these structures were not ideal for providing group support.

On the AulaNet, the only service that made it possible to decide who would carry out a given activity was Learner Co-Authorship. In this service, the coordinator defines who will be the co-authoring learners. After the creation of content by the learner, the coordinator can grade the learner's contribution and use it on the course. Thus the contribution record that is stored for the Follow-up Reports can be a learner co-authorship that was used or a learner co-authorship that was not used, but the assingnment is not recorded through this data structure. A group on the AulaNet environment is related directly to a class and the members of this group are all learners of this class.

In this paper, we do not discuss some social and institutional factors that appear in large groups, such as definition of roles for the members of the group and the institution of rules or rules of conduct, because we are dealing with a class subdivision. This does not imply that the research into group interaction is less important, but rather that at this moment what is intended is to supply an initial support for groups.

Upon recording the attribution of a group to an activity, it is possible to have the same group associated with different activities. For example, a group might tackle a first activity and subsequently use groups of different learners in the other activities. At the end, the initial groups can be used again for discussion, resolving a final activity or generating an activity report.

In order for group support to be implemented, some changes were necessary to the structure of the courses, as presented in Figure 3. The first modification was the "generalization" of the class activities represented by the ClassContent UML class. This generalization also included the change of some associations, such as Tasks and Exams that previously had been associated with a course and then became associated with a class.

The need for other adjustments also was seen, such as the standardization of some terms and functions of the environment. In the AulaNet, there is the certification concept. Upon creating content (Webliography, bibliography, notice, documentation or lesson, the teacher (Coordinator or Teacher Co-Author) can decide if the content is certified or not. Upon certifying a content, it will be exhibited to the participants of a course. However, this was not possible when a task was created: all of the tasks were automatically certified. Like the tasks, the majority of the UML classes associated with the student UML class did not possess the certification attribute. With the creation of ClassContent, it was possible to create a single identifier for all types of activities related to the class that could be associated with groups. And it also was possible to resolve the certification problem, creating an attribute in the ClassContent UML class.

Figure 3 - New structure (UML class diagram) of a course in the AulaNet environment

The structure of the groups presented in Figure 3 and that was implemented in the AulaNet is inspired in the IMS Enterprise specification presented in Section 2.3.1. One sees that some of the mandatory elements of the specification were not directly related to the groups implemented on the AulaNet. It was clear that some of these elements were not group attributes but rather consisted of the relationship of a group and an activity (ClassContent), such as a Task of the AulaNet.

The essence of the AulaNet activities is diversified, thus for the implementation of the groups to be compatible with the IMS Enterprise specification it is necessary to analyze each one of the relationships between the groups and the activities. Based upon the experience acquired using the environment, the Tasks service was the first to be altered in order to support groups. The mapping of the relationship of the AulaNet groups and Tasks service to the mandatory elements of the IMS specification can be found in (Cunha, 2002).

3.3. Groups, The AulaNet Environment and Software Agents

The use of information technologies and the alliances between organizations will ease the process of the (self)generation of groups among these federations and make them possible (Lipman-Blumen & Leavitt, 2001). We believe that in teaching environments new characteristics must be considered such as learning styles, personal preferences, points of view and, one way or another, new ways of evaluation by the teachers or by the learners themselves.

Another aspect that should be considered and studied is the importance of the relationships between group members. Freud (1921) said that people become members of and remain in groups because of emotional ties to other members. In the subconscious of a group member there is the introjection of a preferred person or the qualities of this person, and the projection of the negativities of a(nother) member. The analysis of these ties can lead to a greater exchange between groups of people with educational, and possibly work, environments. For this to happen interchange between enterprise systems and instructional management is important.

Based upon group support implemented on the AulaNet environment it was possible to apply and use the concepts of software agents to provide a service that supports the formation of groups of learners, as will be presented in the next sections. With the study of and the use this technology, one also sees the great potential for application of the group and agent technologies to equip the environment with services that are even more customized, distributed or even more collaborative. With the creation of a federation of AulaNet servers, greater interchange between groups of learners from different servers will be possible. Thus, more collaborative experience will be carried out within the environment and, also, with the use of the IMS Enterprise specification in other environments.

Next we highlight concepts regarding software agents and how this technology can be used in environments for Web-based instructional management. Of particular interest is the explanation of why use a multi-agent system to support group formation.

4. Software Agents

For Jennings, Sycara and Wooldridge (1998), autonomous agents and multi-agent systems represent a new way of analyzing, designing and implementing complex software. The agent abstraction has a wide gamut of applications, ranging from the creation of personal assistants to air traffic control systems, electronic commerce and the group work support.

According to Huhns and Singh (1998), with the need to make data available anywhere and at any time, information environments are becoming increasingly larger, more complex and distributed. In order to join together the different systems found, many designers have used the agent paradigm to provide middleware that is uniform in syntax and semantically consistent. However, the dynamic characteristic and complexity of these environments lead to the need for more customized interfaces, and here again personal agents can help users.

4.1. Conception

In 1995, the concept of software agents was emerging(Wooldridge & Jennings, 1995). In different research groups in order to define an agent different characteristics could be necessary, leading to a large variety of definitions and characteristics.

Franklin and Graesser (1996) provided a general view about the definition of autonomous agents. A number of concepts and points of view of what are agents are presented in order to arrive at a definition that contains the essence of the agency concept, and that can more define the class of agents more widely. For them, an autonomous agent is a system that senses and acts within the environment, of which it is a part, following its own agenda and acting to satisfy it.

Wooldridge (1999) presents a definition of agents as being a system that is found in an environment and that is capable of autonomous actions within this environment to achieve its objectives. He also reaffirms that even researchers do not agree regarding a definition of agents nevertheless there are important characteristics that define an agent, of which the most accepted is autonomy. Other attributes vary according to importance, pending upon the application domain of the agents.

The Green Paper (OMG, 2000) of the Object Management Group (OMG) presents an extensive list describing various characteristics that a software agent can present:

- Autonomy: an autonomous agent is capable of acting without external user interference or interference from other systems. A good way to talk about autonomy is not regarding its presence or absence but rather through a scale - that is, how much an agent has control over its internal state and actions based upon its own experiences;
- Interactivity: this characteristic is present in agents that are capable of communicating with other agents and with the environment where they are found;
- Mobility: this consists of the capacity to move from one environment to another;
- Pro-activity: a pro-active agent is one that is capable of making decisions without being asked to do so, but that also satisfies some objective. In this case, the agent does not react only to the environment, but it also has its own proposals and acts oriented to goals.
- Rationality: rational agents are capable of choosing an action to be carried out based upon their goals and their knowledge that a particular action will move them closer to concluding their goal plan.

When one adopts an agent-oriented view, the majority of the systems cannot be implemented as a single agent but, rather, a set of them - that is, a multi-agent system. It is expected that the agents are able to interact and coordinate themselves in order to achieve a common objective. However, in many cases a single agent can be sufficient, such as in the case of personal assistants that do not need to interact with other agents.

Multi-agent systems are ideal for representing problems with multiple methods for solving them, multiple perspectives and/or multiple entities that resolve them (Jennings *et al*., 1998). According to Poslad, Buckle and Hadingham (2000), in multi-agent systems distributed heterogeneous services are represented as autonomous software agents that interconnect using an Agent Communication Language (ACL) that is based upon the theory of speech acts (Searle, 1969).

For the OMG (2000), one should not create an agent that does everything since there is a great possibility of facing problems of reliability and efficiency. Upon dividing up functions between a number of agents, it is possible to achieve modularity, flexibility, maintainability and greater extension. Also, one should not centralize all knowledge in a single agent since, normally, specific knowledge is distributed and when necessary all that is needed can be merged, thus generating a wider and updated view.

4.2. Agents for the AulaNet Environment

According to Aroyo and Kommers (1999), agents can influence different aspects in educational systems. They supply new educational paradigms, support theories and can be very helpful both for learners and for teachers in the task of computer-aided learning. The application of agents in the educational sector comes about mainly in the form of personal assistants, user guides, alternative help systems, dynamic distributed system architectures, human-system mediators and others.

As a result of all of the changes that have taken place in the educational system, one now sees the increasing emergence of complex and dynamic educational infrastructure that needs to be efficiently managed and, corroborating this, new (types of) educational mechanisms and services now need to be developed and supplied.

In particular these services need to satisfy a series of requirements such as personalization, adaptation, support for user mobility, support for users while they are dealing with new technologies, among others. Agents emerge to provide solutions for these requirements in a way that is more efficient when compared to other existing technologies (Aroyo & Kommers, 1999).

Lees and Ye (2001) believe that the application of the agent paradigm to CSCW potentially can: make the exchange of information more fluid among the participants of groupware systems (as decision-making systems), help in control of the process flows and also supplie groupware interfaces. These ideas also are applicable to other domains, such as is the case of interactive learning.

The AulaNet environment was developed based upon CSCW umbrella and contains a range of different pedagogical opportunities as represented by its services. The following are highlights of possible elements of the environment that would benefit from the application of the multi-agent system paradigm:

- Course content: educational content could be dynamically linked through the use of pedagogical agents that determine the best sequence of presentation or method of exhibition to learners, for example based upon their profiles;
- Asynchronous communication: a greater exchange between the participants of a course could be obtained by using personal agents to filter messages (Maes, 1994) as well as the creation of link structures for messages related to the interests of the participants;
- Synchronous communication: the use of agents can assist the teachers to mediate online debates (Jaques,Andrade, Moraes & Móra, 2000) in a manner that improves the learning that takes place during these synchronous exchanges;
- Support of group work: software agents can be used for the formation (Olguín, Delgado & Ricarte, 2000) and the support for group work, classes or even entire courses;
- Other possibilities are the use of agents to exchange content and to the use of virtual reality for distance learning courses.

As presented in the above list, a learning environment can become complex enough to instigate the use of the agent paradigm. With the development of technology based on mobility, through the use of personal assistants (PDAs) and cellular telephones, a new challenge also has been presented regarding the way of accessing and presenting educational content.

In terms of the geographical distribution of the participants, which is one of the most publicized advantages of the Web-based education environments, there are much to gain through the use of the agent paradigm. Through computation distribution there can be a significant reduction in the demand for computational resources on the servers and greater customization for the clients. How to configure the different personal agents to carry out all of these tasks, or how to configure a course to make use of all of these types of agents are questions that still are open and lacking in research.

The use of agents for the formation of groups within the AulaNet is the first attempt to incorporate a multi-agent system into the environment. Next, we present the justification for the use of the agent-oriented paradigm.

- Autonomy: the use of the concept of autonomy permits the encapsulation of the interests of the participants of course. For example, a learner's agent can give preference to participation in groups where the learner has a greater level of interest in the topics;
- Interactivity: for there to be communication between the agents in order to discover partners it is necessary that they use a communications protocol; that is, it is necessary that they are interactive;
- Collaboration: the learner agents need to collaborate, supplying and receiving information about which ones will be the best partners in order to put together the group as intended by the teachers of the course;
- Pro-activity: the capacity to act with or without the need for user interference can be used to permit agents to suggest the formation of groups, based upon the identification of the interests of the participants and given topics and the activities related to a course. Another form of applying pro-activity would be the awareness of the specific needs of a group of learners and the suggestion of the formation of a workgroup;
- Mobility: the mobility of the agents would make it possible to use systems for forming groups from different servers, which is especially interesting in organizations that are far-flung geographically.

5. Group Formation on the AulaNet

To design a multi-agent system (MAS) to support group formation it was necessary to model the learners of the AulaNet environment. We next present some considerations regarding the modeling of these learners, their respective implementation in the environment as well as concepts about how to find agents that supply the services or the information desired by other agents. Then we present the MAS for the group formation and some work related to the application of agents in education and group formation. Subsequently the concept of a federation of AulaNet servers is presented where it would be possible to create groups with members (on different servers) of different classes of the same course. And finally, some related work that is compared to the MAS developed for the AulaNet environment suggests future work that could be carried out.

5.1. Modeling the Learners

According to Kay (2001), in the first computer-assisted teaching environments the idea was to build "teachers" who could transmit knowledge to the learners. Currently, these types of environments are more geared up for exploration on the part of the learners, designing, building and using adaptive systems as tools. These environments also are being built to give greater responsibility to the learners regarding aspects of the learning process, and especially regarding control of its model, which is the central aspect in the adaptability of the tools.

For McCalla, Vassileva, Greer and Bull (2000), learner models may have a variety of purposes depending upon the type of knowledge that needs to be stored and processed. For them, the computation of all of the learner (sub)models of an environment can be computationally expensive and not always necessary. In the work cited four purposes are presented for a model: reflection, validation, matchmakers and negotiation.

The first proposal is reflection. In this proposal the learner models can be seen by all of the other learners. Questions such as, "How do my colleagues see me?" can help the learners re-evaluate their attitude within a group or allow the evaluation of the other members. For its part, validation can be seen as a special type of reflection. Through validation, a learner for example can confirm opinions through different points of view of the model. Instead of requesting some information, the learner begins with an opinion that will be validated.

The matchmaking proposal occurs, for example, when one desires to find an appropriate peer to help in an activity. In this case matchmaker agents can be used. With access to different learner models, these agents are able to adapt to the needs of those who are seeking and those who are offering help that is based on the models. The last of the proposals is negotiation. Negotiation can be part of the matchmaking process or it may occur for other reasons, such as the exchange of knowledge between two agents.

For Kay (2001), there are potential problems from the learners' point of view. One is the increase in the power of choice and control over the model. This could increase the learners' workloads or even turn into a distraction. In this case, the learners should take advantage of the moments such as the end of a course or a topic to evaluate and reflect upon their participation and the learning process.

By making it possible for learners to control their models, or part of them, the designers of the environment subject them to a number of risks. If learners have control of their models, they may provide incorrect information; they may use the learning environment in an improper way, intentionally or even accidentally, reducing the effectiveness of the educational process; if asked to conduct self-evaluation, they may underestimate or overestimate their knowledge. In the future, it is intended that learning environments take all of these aspects into consideration. One of the ways of getting around the problem of incorrect data being supplied by the learners is to store what type of information they are providing and what type the environment generates.

To implement the learner models within the AulaNet environment, it was decided to use the specification defined by the IMS (IMS, 2001) of Reusable Competency Definitions (RCDs) (IMS RCD, 2001). The work of Soltysiak and Crabtree (1998) is recommended for a detailed review of the use of user modeling for agents.

In the specification of the IMS, the word competence is used in a general manner, including meanings such as skill, knowledge, task and learning outcome. It was thus seen that the best meaning for using in the environment was as knowledge, making it possible to record reusable knowledge definitions and, subsequently, the creation of learner models based upon these definitions.

According to the IMS, the reusable competence definitions provide a means for creating common understanding that appear as part of a career plan, prerequisites for a course or for educational objectives. They can be used for exchange between learning environments or human resources systems among others.

The reusable definitions of competence were created for an exchange between machines although the information they currently contain is for human understanding. Basically, a definition contains a unique identifier and a non-structured textual description.

In order to make it possible for AulaNet participants to organize their models and create groups, the reusable competence definitions were related with the courses, the class content (ClassContent) and the participants, as shown in Figure 4.

Figure 4 - The employment of Reusable Competence Definitions in the AulaNet environment

It is left up to the teachers of a course (Coordinators and Teacher Co-Authors) to relate a set of RCDs with their specific courses and/or activities (ClassContent). For example a course about databases could be related with these RCDs: Database Architecture; Database Models; Relational Model and SQL (Structured Query Language), among others.

The participants are allowed to provide information about the set of RCDs of the server or of a specific course. The information that will be supplied by the participants is a grade or concept about how much they know about or are interested in the RCD; it also is possible to insert a comment about the attribution of the concept they are making.

Considering the need to distinguish between the information of the learner model generated by the environment and that supplied by the learner, it was decided to use the "Type" attribute in the records for the RCDParticipant table. There are three dimensions or types of information of an RCD in the AulaNet: Interest, Qualification and Competence.

Interest is the information that is supplied by the learners about their level of interest regarding a given RCD. Qualification is also information that is supplied by the learners regarding their experience with a given RCD. And, last, Competence is the information generated by the environment based on the evaluation of a learner by the teachers for a given activity associated with an RCD.

The use of learner modeling in the AulaNet environment matches one of the concerns of modern organizations, which is the knowledge management. For Yiman and Kobsa (2000), systems for finding individuals with given expertise are gaining importance as organizations begin to seek new ways to exploit their in-house knowledge capital and improve collaboration between employees.

5.2. Matchmaking and Brokering Concepts

One of the problems in a multi-agent system project is the way to discover which agents have a specific piece of information or skill. Many environments and specifications define agents that offer white page services, which are directories of agents; and yellow page services, which directories with the features offered by the agents. These are the cases of FIPA-OS (Poslad, Buckle & Hadingham, 2000) and SACI (Hübner & Sichman, 2000). Some agent communication languages such as KQML also offer special performatives for this behavior, such as Recruit, Broker and Forward (Decker, Williamsom & Sycara, 1996).

For Ivezic, Barbacci, Libes, Potok and Robert (2000), matchmakers and brokers work as intermediate agents between agents that supply services and agents that need these services. Next, we describe the interpretations of the matchmaking and brokering processes found in (Decker *et al*., 1996).

The matchmaking process allows Agent A, with an objective, to get to know Agent B through a matchmaker M. Either the objective cannot be achieved by A, or A understands that the objective can be better achieved by the other agent. The objective can be a goal, a piece of information or a service. From the moment that Agent A, through the mediation of M, gets to know Agent B, it can negotiate with B, for example, hiring a service.

The brokering process involves how an agent with an objective can get that objective to be carried out by another agent. The process involves Agent A, that requests the carrying out of an objective and Broker B, who knows the other agents (C1, C2, …, Cn) and their capabilities. B announces its capabilities as a function of the capabilities of these other C agents. From the point of view of Agent A, there is no difference between a broker and the other C agents, except for the response time and, possibly, the "price" of the service offered. For their part, the C agents are committed to B to carry out a set of predefined objectives.

For Foner (1996), the use of centralized architecture for matchmaking can be valid, for example, in cases where the agents are unable to discover each other and request that a central "entity" provide a solution for the problem. However, there also are disadvantages to this type of architecture - for example, its tolerance for mistakes is low since it has a central point where attacks or even incidents may occur. Another disadvantage is the potential computational bottlenecks that could arise as a result of the increase in the number of agents.

He also states that the use of some techniques that already have been applied in networks, such as the hierarchical organization of entities (such as on the Internet's domain name systems and newsgroups) does not reduce problems like the computational bottleneck. This occurs because of the non-existence of a standard hierarchy. For example, why would the interests of one agent come ahead of another? In order to propose a solution to these problems, Foner used some ideas based on computational ecology (Huberman, 1988). The main ideas are:

- To compare agent information in a decentralized manner (peer-to-peer);
- To use references from one agent for the others and an algorithm that remember hill-climbing to find other partners to;
- Build clusters or clumps of agents with common interests, and;
- To use these clusters of agents with common interests to present users to each other;
- To use a persistent agent that is active for long periods of time and not an agent that the user initiates, obtains a result and then deactivates. In this way, more agents can be consulted and more appropriate clusters can be formed.

Next, we present a multi-agent system that helps teachers in forming groups of learners.

5.3. A Multi-Agent System for Group Formation

It is possible on AulaNet to create a course, and within the course it is possible to have different classes, each with their respective mediators. The environment was prepared for the use by groups within classes using the Tasks service. Furthermore, it is possible to form groups manually. As presented in section 4.1, an agent is found in an environment and interacts with this environment. In comparison with the environment concept, in the SACI tool there is a concept of society where the agents are united and can communicate through the common agent communication language using their identities. An identity is a name that uniquely identifies an agent within its society (Hübner & Sichman, 2001).

In the SACI tool, an agent's life cycle is as follows. The agent enters a society and receives an identification. In this society it can send or receive messages from other agents of the same society, announce its skills to the society and, finally, it may leave the society, thereupon losing its identity.

The creation of groups in a class is related to the dynamics of the course - that is, the person responsible for creating groups is the mediator of the class. The mediator requests that a mediating agent (AgMediator) create groups in his class, supplying the information that is necessary for this purpose - such as the number of groups, what are the RCDs that must be analyzed in the learner models, what is the level of difference between learners and if the learners can be repeated in the groups, among other data.

In order to determine if the maintenance of agents representing learners in execution during a long period of time is feasible or not, it would be necessary to analyze the average number of participants on the AulaNet servers. Thus, it was decided to permit that the AgMediator enter into the society of the class that requested the creation of the groups and instantiate an agent for forming groups (AgGroups) that, for its part, instantiates the agents for all of the learners (AgLearner).

On AgGroups' initialization, it receives from the AgMediator which RCDs and respective aspects it should represent. The AgLearner agents are initialized and enter into the society, publishing which learner they represent and if they can satisfy the needs of the RCDs and aspects requested by the AgGroups. The AgMediator then requests that the AgGroups form the groups, presenting the goal containing the levels of difference of each RCD and its respective aspects.

The AgGroups search the society for the AgLearners that can respond to the formation of groups and passes along the request of the AgMediator to the AgLearner. Those that are able to form groups collaborate among themselves to suggest groups that satisfy the request.

It can be seen that the formation of K groups with n participants satisfying a given degree of difference between the participant models is a NP-Complete problem. The 3-Dimensional matching problem (Garey and Johnson, 1978) is a problem that can have polinomial time in the case the elements are repeated in the matchings that are carried out, or if the dimension is less than 3. However, in our case, it is possible that we will have numbers of groups greater than or equal to 3.

One heuristic adopted to solve this problem allows agents to pass along references from other agents they know, thus reducing the quantity of messages exchanged. Another heuristic adopted was the use of a greedy strategy (Cormen, Leiserson, Rivest & Stein, 1991), in which as of the moment an AgLearner succeeds in forming the groups that were requested in accordance with the parameters received from AgGroups, it stops collaborating, so informing the AgGroups and leaving the society.

The AgGroups filter these groups, if there are repeated groups. AgGroups supplies this information to the AgMediator. For its part, the AgMediator gets back the information to the mediator who requested the formation of the groups through the AulaNet's group formation interface. In case a timeout occurs during the attempt to form the groups determined by the AgMediator, it requests that the AgGroups and the AgLearner give up to carry out the formation, halting collaboration and leaving the society.

Each AgLearner knows the model of its learner. This model is composed of the learner's competence, interest and qualification aspects from the RCDs. Upon requesting the formation of a group, the AgGroups define which aspects of the model must be taken into consideration for negotiation by the AgLearners.

For example, a mediator might want learners, independent of their preferences, to form groups where all have the same prior knowledge about a given RCD. He should define that the interest and competence aspects will not be taken into account in the negotiation and that the degree of difference be minimal in the qualification aspect. Other combinations of aspects may be created, assisting the mediator to apply different tactics for forming groups.

The collaboration model for learner agents was inspired in the matchmaking algorithm found in (Foner, 1996). Thus, each AgLearner has the following data structures:

- Cluster cache: this is a list containing the names of the agents known by this agent and who satisfy the group formation criterion presented by the Agroups. Since the degree of difference concept was adopted, this structure can be divided into a number of others:

  Upon requesting the formation of a group, a mediator decides which aspects (Asp*x*) of the RCDs (RCD*y*) and what degree of difference (GD*z*) must be considered, represented by the (RCD*y*, Asp*x*, GD*z*) tuple. Suppose that a mediator then establishes the following goal for the formation of groups (SQL, Interest, 1) and (Relational Model, Competence, 2). And suppose that the class has three learners Ap1, Ap2 and Ap3, as presented in Table 1.

  Table 1 – A plan for group formation

  It is possible to perceive that Ap1 can be in a cluster cache of Ap2, and also that Ap3 can be in a cluster cache of Ap2, since both Ap1 and Ap3 satisfy the same degree of difference established by the mediator. However, Ap1 and Ap2 cannot be in the same cluster cache of Ap2.

  As shown in the example, the cluster cache of an AgLearner can be constructed using various tactics. In this work, we chose to make the cluster cache of the AgLearners a flexibility point in the multi-agent system. Considering a goal consisting of n tuples we would have 2^n tactics to discover if an agent belongs to a cluster cache. Using the flexibility point, two types of cluster caches were implemented:

  - A positive cluster cache: these are the names of the learner agents that possess all of the RCD values and aspects at a degree of difference higher than the AgLearner in question;
  - A negative cluster cache: these are the names of the learner agents that possess all of the RCD values and aspects at a degree of difference lower than the AgLearner in question.

- Rumor cache: these are the names of the agents and the respective information of their models that were used to verify the criterion presented by the AgGroups of the last *r* agents with whom this agent communicated. It was arbitrarily defined in this paper that the value of *r* should be 5 (five), but tests are required to check to see if this value is appropriate;
- Pending-contact list: these are the names of the agents that still have not been contacted and the respective cluster cache to which they could be part of. The names of the agents are discovered through the references of other agents that have been contacted and their appropriateness to the cluster cache is checked through the processing of the rumor cache received as reference.

To initiate the collaboration, the learner agents need to discover at least one other agent who also satisfies the criteria established by the AgGroups. This can be obtained by asking the agents to publish this skill within the society. The AgLearner chooses one of these agents randomly and they begin a conversation.

Suppose that Agent A discovers that Agent B possesses the same RCD that is being requested by the mediator for the formation of the groups and that all of the aspects (Interest, Qualification and Competence) must be taken into consideration during the negotiation.

Agent A requests from Agent B its values for the goal established by the AgGroups. If it is the case that the degree of difference is satisfied and Agent B is available, it is added to the corresponding cluster cache (positive or negative). Automatically B is added to the rumor cache of Agent A. An agent always is available if the learners can repeat in the groups and is not available when the learners cannot repeat and the agent already has assumed a commitment to remain in the cluster cache of another agent.

After this comparison phase, Agent A asks Agent B for the references of other agents that could assist it in the formation of the groups. Agent B then supplies its rumor cache. In possession of the Agent B rumor cache, Agent A decides which of the agents contacted by B that can be inserted into its pending-contact list for a future contact. The rumor cache of A is updated with the rumor cache of B. The entire process is reciprocal: Agent A also supplies its aspect values to B, as well as its rumor cache, so that B can continue its search. After the conclusion of the collaboration with B, Agent A continues contacting other agents on its pending-contact list, and if it still is necessary the agent contacts the other AgLearners that can satisfy the AgGroups' goals.

Regarding the requirements passed along by the mediator, the size of the group is what defines the end of the proposed algorithm. As in our case, the AgLearners are created only for the formation of groups and are not in execution during a long period of time, this quantity will be the limiting factor for the size of the cluster cache. In the greedy strategy adopted, when the cluster cache of an AgLearner satisfies the size requested by the AgMediator, it sends its suggestion of a group.

A strategy that could be implemented in the future in the event that AgLearner remains in use for a long period of time would be to always request the cluster cache independent of its size and then proceed to divide up these cluster caches to obtain, for example, more uniform group sizes.

One of the questions that arises from the use of software agents is: what do these agents learn? Do they learn from their interaction with the users or with the other agents? According to the way the system was designed and implemented, it is not possible to learn more about the interaction between agents unless they continue to execute for a longer amount of time. Regarding the learners, the agents could learn more about them through the changes that occur in their competence, as of their use of the environment.

Why do the AgLearners collaborate? Would it not be better to create a central agent for matchmaking? The decision to use a distributed structure was based on (Foner, 1996) who states that for a reduced number of agents the computational gains are not considerable in relation to the central structure. In classes with few learners the centralized structure perhaps would be more efficient; however, this paper deals with future versions of the AulaNet environment and, thus, supports the formation of inter-class and inter-server groups. For example, in Figure 5, it is possible to foresee a future step for the formation of inter-class groups within the same course.

5.4. Federation of AulaNet Servers

For example, if a coordinator wants to create groups of learners for co-authorship, he can request that the mediators of the course offer him groups or learners with the particular characteristics found in some RCDs. These groups could be used directly for attribution to the desired activity. Or, in the case of the learners, their agents could be asked to move to the coordinator's society and collaborate in the formation of groups with members from different classes.

Figure 5 - Group formation using software agents - future step

Besides the possibility of the creation of inter-class groups, ones of the purposes of the AulaNet project is the creation of server federations, where it would be possible for the participants and the courses to possess single identifiers even though they were acessing different servers. These federations would offer support for user mobility besides being an even more fertile field for the use of agents due to the distributed nature of the proposal.

One of the needs already perceived in the AulaNet project is for an exchange of contents between the different Web-based instructional management servers, whether or not they are AulaNet servers. Silva *et alli* (2001) propose a framework for the interoperability of educational content using the IMS specifications called ContentNet. The ContentNet proposes an architecture composed of two modules: Content Search and Iserver.

The Content Search module acts as a Search Server that centralizes the information about the Content Servers while the IServer serves for communication between the Content and the Search Servers. The IServer, Intermediator Server, supplies the Search Server with the information from the Content Server, and supplies the Content Server with the information available from the Search Server. In this way communication between different types of Content Servers that make use of the IMS specification is possible. The proposed architecture is centralized. There is also a BackUp Server that acts as a Content Search server when the former is not available.

A federation of servers in this paper is a set of servers with a single identification, for example the server's IP address, that is capable of exchanging information about the learners and their interaction in different courses - and not only educational content, as is the case of ContentNet.

A course in a federation of servers is created on one server and has classes on it and/or on other servers. The interaction of the learners is stored on the class servers and the content relating to the entire course is stored on the course server. Thus, this server must be capable of handling a larger number of accesses.

The use of centralized content servers alerts us to problems such as the need for more efficient backup strategies as well as the need for secure access verification mechanisms. The current lack of a commerce business model to the re-use of content also is a challenge to be overcome. In fact, this is a very interesting field of research, one that will benefit from the maturing of the electronic commerce area as well as the use of interchange standards between Web-based instructional systems.

Among the advantages that could be obtained by using server federations, we can point to the fact that institutions interested in setting up distance education or knowledge management solutions could invest in powerful servers for the content of their courses and less-powerful servers to record the interaction of the learners in their respective classes. Another possibility that should be explored is the customization of content; based upon a central point of access, the content can be customized by software agents for use by different servers. On these servers this content could be customized once again according to the user models or with the devices being used to access it.

Based upon a federation of servers, the use of software agents for forming groups, or even for the customization of content, stands out as an appropriate solution. Especially for the formation of groups, we fear that the storage of centralized information about the learners could overload the course server, whereas maintaining class servers means the knowledge remains distributed and it is easily accessible by the agents. The agents can travel around and supply information or negotiate with the agents of other learners in order to meet goals that have been established, for example, by the coordinator of the course.

5.5. Related Work

Collaborative agents are capable of cooperating with other agents in order to achieve a common goal. According to Nwana (1996), the main features of collaborative agents are autonomy, social skills, reactivity and pro-activity.

In order to use collaborative activities in education, it is necessary to use synchronous and asynchronous communication tools. These tools present the advantage of making it possible to access the experience and the collaboration of others, both in the creation of a shared product as well as in the exchange of information.

In general, a collaborative agent can assist in the activities of both a learner as well as a teacher. Upon communicating and cooperating with their peers (artificial agents or humans) by monitoring and acting autonomously, the agents can help attain objectives or carry out tasks designated for the users.

It has been seen that with the popularization of the Internet and the agent paradigm as one of the solutions for distributed problems, new techniques are being applied in distance education through the World Wide Web. Many of these projects are experimental or prototypes. O'Riordan and Griffith (1999) presented ways of taking advantage of the possibilities offered by the Web through the use of agents in an educational environment. For their part, work conducted by Vassileva and Deters (2001) presents a multi-agent system to help peers that was designed to help learners in activities that require them to solve problems. In the following paragraphs, we highlight work that is representative of the field of application of this paper.

In the work of Olguín *et alli* (2000), it is presented the development of an agent architecture to support group formation. The groups formed have a well defined profile. The groups that are formed by the AulaNet multi-agent system also have a profile that is designated by the mediator; although this profile is derived from the learners' model, it may be modified over the life of a course and the group can be used again in activities other than the one for which it was created.

Also in the work of Olguín *et alli*, a learner profile stores information about its competence and skills in a set of topics, as well as information about their performance in group activities. However, the Interest dimension, used by the AulaNet, is not stored, despite the fact that, in the cited system, group participation depends upon user approval. On the AulaNet, this user acceptance dependency is not implemented yet, although in the future the environment will support study or learning groups independently of an activity. So, it will be necessary also to endow learners with a way of using the MAS and setting up the groups.

The Guardian Agent (Whatley, Beer & Staniford, 2001) foresees support for group work by monitoring group activities. Despite possessing a learner model, it is not concerned about how to form groups. What most stands out is the attribute of specific roles of group members according to their models. The learners' skills are obtained through questioning them regarding the "task areas" which a learner likes, is good, does not like or is not good. That is, what we have here is binary information (like or does not like, knows or does not know) supplied by the learners for the model to agents.

The support of group work is an important concern in Internet learning environments, as is pointed out by Kojiri, Ogawa and Watanabe (2001). For them, full support of learners and activity management is necessary for example, not to engage only in interactive discussions about the learning processes but also so as not to remain passive. With the initial support for the use of groups on the AulaNet, the next step is to supply this more specific support to groups within the activities.

The MATHNET ((Labidi, Silva, Coutinho, Costa & Costa, 2000) and (Coutinho, Labidi, Serra & Teixeira, 2000)) is not concerned about the formation of groups, despite having a specific agent for modeling the learners. The idea of this modeling is to offer intelligent tutoring. But it also could be used to form groups, thus helping the teachers who use the MAS.

The work that has been mentioned here show something of the diversified panorama regarding the use of agents for forming and using groups. Furthermore, it also is possible to see the differences in the AulaNet implementation and the other work and to learn from it lessons that serve to stimulate future work in the environment.

6. Final Considerations

As emphasized by Dertouzos (2001), both group work and group education is important in the new era in which computing is becoming geared to human beings (human-centered computing). This paper is inserted in both of these fields, supplying a technological perspective for the formation of workgroups. This perspective is represented by the CSCW technology, incremented through the use of software agents.

The use of workgroups within the AulaNet environment provides a view of the needs of the learners and teachers and collaborates with a contextualized survey. Moreover, the concern with using IMS standards and the groupware approach adopted by the system makes it possible to have easy similarity between the educational world and the workplace of the market job. The use of the AulaNet as a tool for supporting work is being researched and looks promising. The resemblance of a course with a project, of a class with a team, of a learning group with a workgroup, is a very stimulating view in this research process.

Based upon the literature that was surveyed, it is believed that the use of agents could favor a number of psycho-pedagogical aspects of Web-based education. By providing support for forming and working in a group, the software agents also are supporting project learning and collaborative learning. When the groups that have been formed demonstrate a high degree of heterogeneity, interdisciplinary attitude and practice also may benefit. In the same way that these aspects are influenced, it is necessary to point out that professional skills can be developed and influenced through group work, such as the capacity for self-monitoring, listening, presenting new ideas and persuasion, among others.

7. Acknowledgements

The AulaNet project is partially financed by the Fundação Padre Leonel Franca, by the Ministry of Science and

Technology through its Program of Excellence Nuclei (PRONEX) grant nº 76.97.1029.00 (3366). And also through individual grants awarded by CNPq – Brazilian National Research Council to: Carlos José Pereira de Lucena nº 300031/1992-0, Hugo Fuks nº 352820/1996-9 and Leonardo Magela Cunha nº 140895/2002-7.

8. References

AICC. (2001). Aviation Industry Computer-Based Training Committees (online). <http://www.aicc.org> (Consultation: 2001, July 3rd).

AROYO, L. & KOMMERS, P. (1999). Preface - Intelligent Agents for Educational Comp uter-Aided Systems. *Journal of Interactive Learning Research*, v. 10, n. 3/4, p. 235-242.

BAECKER, R. (1993). *Readings in Groupware and Computer-Supported Cooperative Work: Assisting Human-Human Collaboration*. San Mateo, CA, EUA: Morgan-Kaufmann.

BARKER T. & BARKER J. (2001) How Group Working Was Used to Provide a Constructive Computer-Based Learning Environment. In: *The Fourth International Conference on Cognitive Technology, University of Warwick, United Kingdom*. p. 203.

BORGHOFF, U.M. & SCHLICHTER, J.H. (2000). *Computer-Supported Cooperative Work: Introduction to Distributed Applications*. Springer, USA.

CORMEN, T. H.(ed), LEISERSON, C. E., RIVEST, R. L. & STEIN, C. (2001). *Introduction to Algorithms*, MIT Press. 2nd edition.

COUTINHO L. R., LABIDI S., SERRA G. & TEIXEIRA G. A.(2000) Learner Modelling Agent for Cooperative Learning. *Simpósio Brasileiro de Informática na Educação. Maceió, Alagoas, Brazil*. p. 17-23.

CUNHA, L. M. (2002) *Formação de Grupos de Trabalho Utilizando Agentes de Software*. M. Sc. Dissertation. Computer Science Department, Pontifical Catholic University of Rio de Janeiro: PUC-Rio. [In Portuguese]

DECKER K., WILLIAMSON M. & SYCARA K. (1996). Matchmaking and Brokering. In: *The Second International Conference on Multi-Agent Systems (ICMAS-96)*, Kyoto, Japan.

DEMARCO, T. & LISTER, T. (1999). *Peopleware: productive project and teams*. 2. ed., New York: Dorset House Publishing Co., Inc.

DERTOUZOS, M. L. (2001). *Unfinished revolution: human-centered computers and what they can do for us*. New York: HarperCollins Publishers, Inc.

DILLENBOURG, P. (1999) What do you mean by collaborative learning? In: DILLENBOURG, P. (Ed) *Collaborative-learning: Cognitive and Computational Approaches*. Oxford: Elsevier. p. 1-19.

ELLIS, C. A. & WAINER, J. (1999). Groupware and Computer Supported Cooperative Work. In: WEISS, G. (ed) *Multiagent Systems – A Modern Approach to Distributed Artificial Intelligence*. Cambridge, MA: MIT Press. p. 426-457.

ELLIS, C. A., GIBBS, S. J. & REIN, G. L. (1991) Groupware: Some Issues and Experiences. *Communications of the ACM*, v. 34, n. 1, p. 38-58.

FONER, L. (1996). A Multi-Agent Referral System for Matchmaking. In: *The First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM 96), London*. p. 245-262.

FRANKLIN, S. & GRAESSER, A. (1996). Is it an agent, or just a program?: A taxonomy for autonomous agents. In: *ECAI'96 Workshop on Agents Theories, Architectures, and Languages: Intelligent Agents III*, Berlin: Springer. p. 21-36.

FREUD, S. (1921) *Group Psychology and the Analysis of the Ego*, edited and translated by STRACHEY, J. (1975) New York: Norton.

FUKS, H. & ASSIS, R. L. (2001). Facilitating Perception on Virtual Learningware-based Environments. *The Journal of Systems and Information Technology*, Edith Cowan University, v. 5, n. 1, p. 93-113.

FUKS, H., GEROSA, M. A. & LUCENA, C. J. P. (2002) Using a Groupware Technology to Implement Cooperative Learning via the Internet - A case study. In: *HICSS'35 - Thirty-fifth Annual HAWAII International Conference on System Sciences. Big Island of Hawaii*. p. 4.

FUKS, H., LAUFER, C., CHOREN, R. & BLOIS, M. (1999) Communication, Coordination and Cooperation in Distance Education. In: *V AMCIS'99 - 1999 Americas Conference on Information Systems. Association for Information Systems (AIS), Milwaukee, USA*. p. 130-132.

FUSSEL, S. R., KRAUT, R. E., LERCH, F. J., SCHERLIS, W. L., MCNALLY, M. M. & CADIZZ, J. J. (1998). Coordination, Overload and Team Performance: Effects of Team Communication Strategies. In: *CSCW '98,* The Association for Computer Machinery, Chapel Hill, USA. p. 275-84.

GARDNER, H. (1993). *Frames of Mind*. London: Fontana Press.

GAREY, M. & JOHNSON, D. (1978) *Computers and Intractability – A Gide to the Theory of NP-Completeness*. W.H. Freeman and Co., San Francisco.

GEROSA, M. A., FUKS, H. & LUCENA, C. J. P. (2001) Use of Categorization and Structuring of Messages in order to Organize the Discussion and Reduce Information Overload in Asynchronous Textual Communication Tools. In: *7th International Workshop on Groupware - CRIWG 2001, IEEE*. Darmstadt, Germany, p. 136-141.

HUBERMAN, B. A. (Ed). (1988). *The Ecology of Computation*. Elsevier Science Publishers B. V..

HÜBNER, J. F. & SICHMAN J. S. (2000). SACI: Uma ferramenta para Implementação e Monitoração da Comunicação entre Agentes. In: *Brazilian Symposium on Artificial Intelligence (SBIA)/7th Ibero-American Conference on AI (IBERAMIA). Atibaia, São Paulo*. p. 47-56. [In Portuguese]

HÜBNER, J. F. & SICHMAN, J. S. (2001). *SACI Programming Guide*. Version 0.8. July, 2001. Universidade de São Paulo.

HUHNS, M. N. & SINGH, M. P. (Ed). (1998). *Readings in Agents*. Morgan Kaufmann Publishers.

IEEE (2001) Learning technology Standards Committee (online) <http://ltsc.ieee.org> (Consultation: 2001, July 3rd).

IMS (2001) Instructional Management Systems Global Consortium, Inc. (online) <http://www.imsproject.org> (Consultation: 2001, July 3rd).

IMS ENT, (2001) Instructional Management Systems Global Consortium, Inc. (online). *IMS Enterprise Specification*. Version 1.01, released January 12, 2000 <http://www.imsproject.org/enterprise/index.html> (Consultation: 2001, July 3rd).

IMS RCD (2001) Instructional Management Systems Global Consortium, Inc. (online) *IMS Reusable Competency Definition Specification*. Version .01, approved by the IMS Technical Board in February 2000 <http://www.imsproject.org/rcd/index.html> (Consultation: 2001, July 3rd).

IVEZIC, N., BARBACCI, M., LIBES, D., POTOK, T. & ROBERT, J. (2000) An Architecture Analysis of Agent-based Systems. In: *Workshop Proceedings of ESAW'00. Engineering Societies in the Agents' World*.

JAQUES, D. (2000). *Learning in Groups*. 3. Ed. London: KoganPage Limited.

JAQUES, P. A., ANDRADE, A. F., MORAES, M. C. & MÓRA, M. C. (2000) Uma Arquitetura de Agentes para a Análise Qualitativa da Interação em Ambientes de Educação à Distância. In: *Simpósio Brasileiro de Informática na Educação, Maceió, Alagoas, Brazil*. p. 282-289. [In Portuguese]

JENNINGS, N. & WOOLDRIDGE, M. (2000). Agent-Oriented Software Engineering. In: BRADSHAW. J. (ed.), *Handbook of Agent Technology*, AAAI/MIT Press.

JENNINGS, N., SYCARA, K. & WOOLDRIDGE, M. (1998). A Roadmap of Agent Research and Development. *Autonomous Agents and Multi-Agent Systems*, Boston, n. 1, p. 7-38, Kluwer Academic Publishers, 1998.

KAY, J. (2001). Learner Control. *User modelling and User Adapted Interaction*, Netherlands, n. 11, p. 111-127, Kluwer Academic Publishers.

KHOSHAFIAN, S. & BUCKIEWICZ, M. (1995). *Introduction to Groupware, Workflow, and Workgroup Computing*. New York: John Wiley and Sons, Inc.

KOJIRI, T., OGAWA, Y. & WATANABE, T. (2001). Agent-oriented Support Environment in Web-based Collaborative Learning. *Journal of Universal Computer Science*, v. 7, n. 3, p. 226-239.

LABIDI, S., SILVA, J. C., COUTINHO, L. R., COSTA, N. S. & COSTA, E. B. (2000). Agent Based Architecture for Cooperative Learning Environment. . In: *Simpósio Brasileiro de Informática na Educação, Maceió, Alagoas, Brazil*. p.32-39.

LEES, B. & YE, Y. (2001). Preface of the *Proceedings of ASCW01 – Workshop of Agent-Supported Cooperative Work. In: The Fifth International Conference on Autonomous Agents. Montreal, CA.*

LEWIN, K. (1948). Frontiers of group dynamics. *Human Relations*, n. 1, p. 5-42.

LIPMAN-BLUMEN, J. & LEAVITT, H. J. (2001). *Hot Groups: Seeding Them, Feeding Them, and Using Them to Ignite Your Organization*. Oxford University Press (Trade).

LUCENA, C. J. P., FUKS, H., MILIDIÚ, R., LAUFER, C., BLOIS, M., CHOREN, R., TORRES, V. & DAFLON, L. (1999). AulaNet: Helping Teachers to Do Their Homework. In: *Multimedia Computer Techniques in Engineering Education Workshop. Graz, Austria: Technische Universitat Graz.*

MAES, P. (1994). Agents that Reduce Work and Information Overload. *Communications of the ACM*, v. 37, n. 7, p. 31-40.

MCCALLA, G., VASSILEVA, J., GREER, J. & BULL, S. (2000). Active Learner Modelling. In: GAUTIER, FRASSON & VANLEHN (Ed.). *Proceedings of ITS'2000*, Springer LNCS 1839, p. 53-62.

MINSKY, M. (1987). *The society of mind*. London: Wilian Heinemann Ltd.

NWANA, H. (1996). Software Agents: An Overview. *The Knowledge Engineering Review*, v. 11, n. 3, p. 1-40.

O'RIORDAN, C. & GRIFFITH, J. A (1999). Multi-Agent System for Intelligent Online Education. *Journal of Interactive Learning Research*, v. 10, n. 3/4, p. 263-274.

OLGUÍN, C. J. M., DELGADO, A. L. N. & RICARTE, I. L. M. (2000). An Agent Infrastructure to Set Collaborative Environments. *IFETS/IEEE Educational Technology & Society*. v.3, n.3, p.65-73.

OMG (2000). OBJECT MANAGEMENT GROUP. *Agent Technology – Green Paper*. Version 1.0, 2000, Object Management Group.

PERKINS, D. N. (1993). Person Plus: a distributed view of thinking and learning. In: SALOMON, G. (Ed). *Distributed Cognitions. Psychological and educational considerations*. Cambridge, USA: Cambridge University Press. pp. 88-110.

POSLAD, S., BUCKLE, P. & HADINGHAM, R. (2000). The FIPA-OS Agent Platform: Open Source for Open Standards. In: *International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology - PAAM 2000*. Manchester, UK. p. 355-368.

SEARLE, J. R. (1969). *Speech Acts*. Cambridge, UK: Cambridge University Press.

SILVA, V., LUCENA, C. J. P. & FUKS, H. (2001). ContentNet: A Framework for the Interoperability of Educational Content Using Standard IMS. *Computers & Education Journal*, v. 37, n. 3/4, Elsevier Science Press, p. 273-295.

SOLTYSIAK, S., CRABTREE, B. (1998). Knowing Me, Knowing You: Practical Issues in the Personalisation of Agent Technology. In: *International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology - PAAM'98*. London. p. 467-484.

VASSILEVA, J. & DETERS, R. (2001). Lessons from Deploying I-Help. In: *First International Workshop on Agents and Internet Learning, Autonomous Agents 2001*. Montreal, CA. p. 21-23.

VYGOTSKY, L. S. (1987) *The Collected Works of L.S.Vygotsky*. v. 1. New York: Plenum Press.

WEISS, G. (Ed). (1999). *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. Cambridge, MA: MIT Press.

WHATLEY, J., BEER, M. & STANIFORD, G. (2001). Facilitation of Online Student Group Project with a Support Agent. In: *Workshop of Agent-Supported Cooperative Work, Autonomous Agents 2001*. Montreal, CA. p. 19-23.

WOOLDRIDGE, M. & JENNINGS, N. (1995). Intelligent agents: Theory and Practice. *The Knowledge Engineering Review*, v. 10, n. 2, p. 115-152.

WOOLDRIDGE, M. (1999). Intelligent agents. In: WEISS, G. (Ed) *Multiagent Systems – A Modern Approach to Distributed Artificial Intelligence*. Cambridge, MA: MIT Press. p. 27-77.

YIMAM, D. S., KOBSA, A. (2000). DEMOIR: A Hybrid Architecture for Expertise Modeling and Recommender Systems. In: *The IEEE 9th International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*. Gaithersburg, MD.

Figure Caption

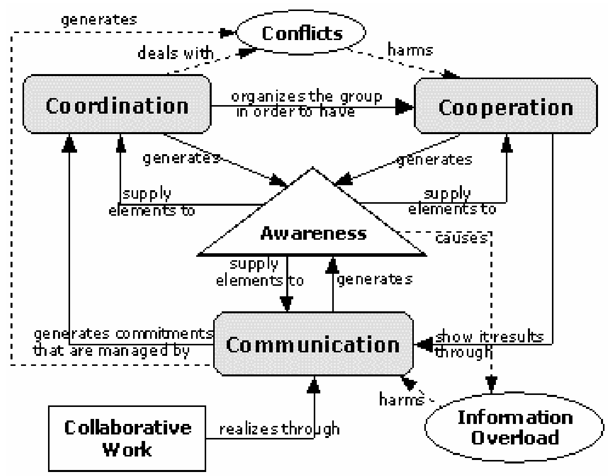

Figure 1 – The collaboration model

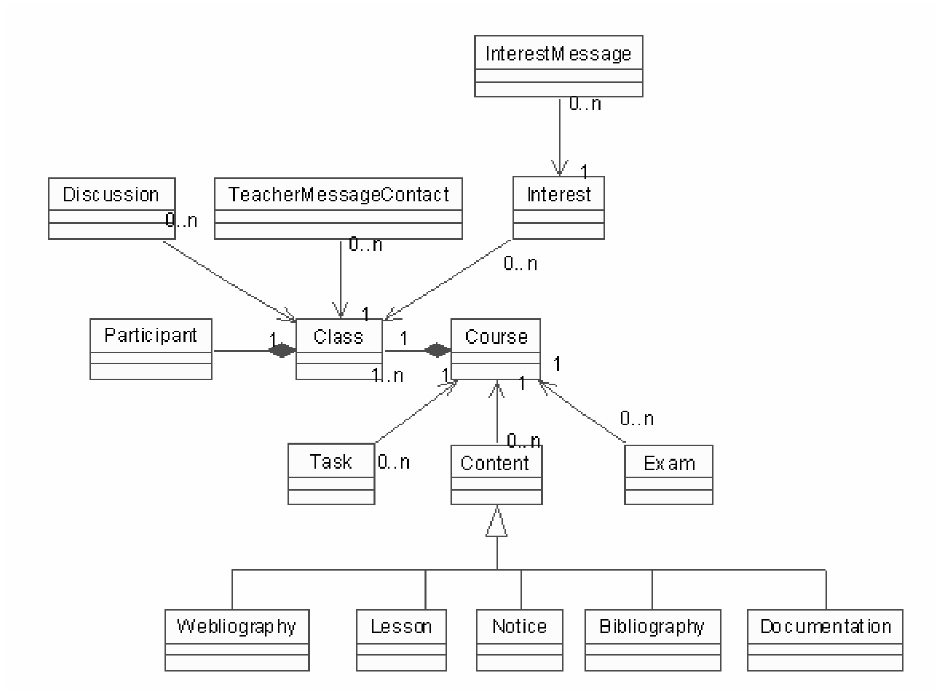

Figure 2 - Structure (UML class diagram) of a course in the AulaNet environment

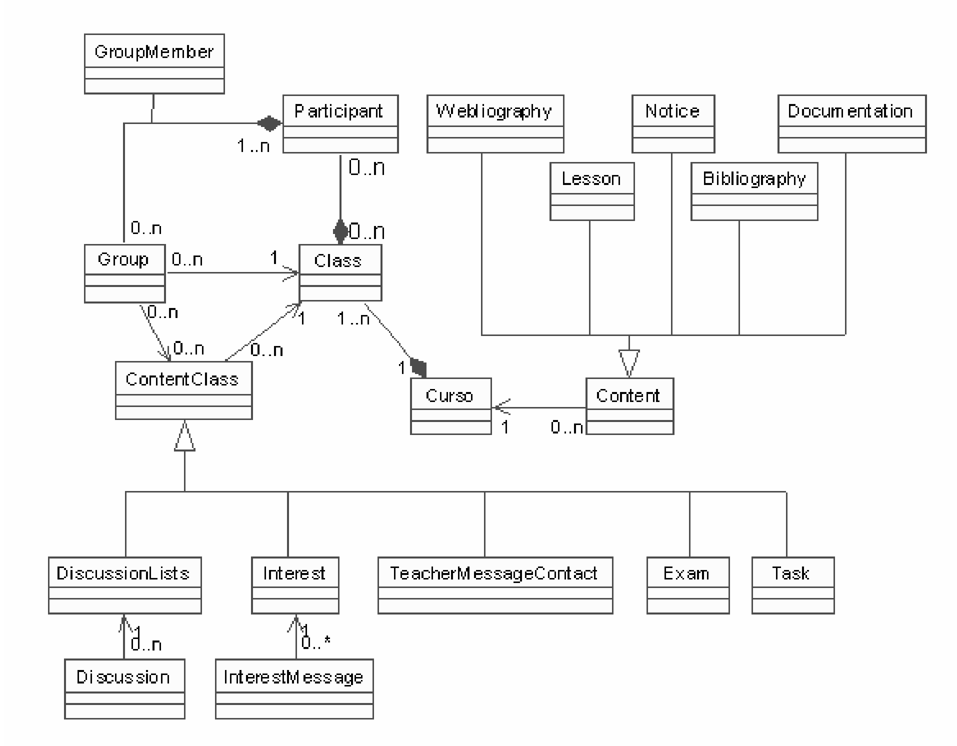

Figure 3 - New structure (UML class diagram) of a course in the AulaNet environment

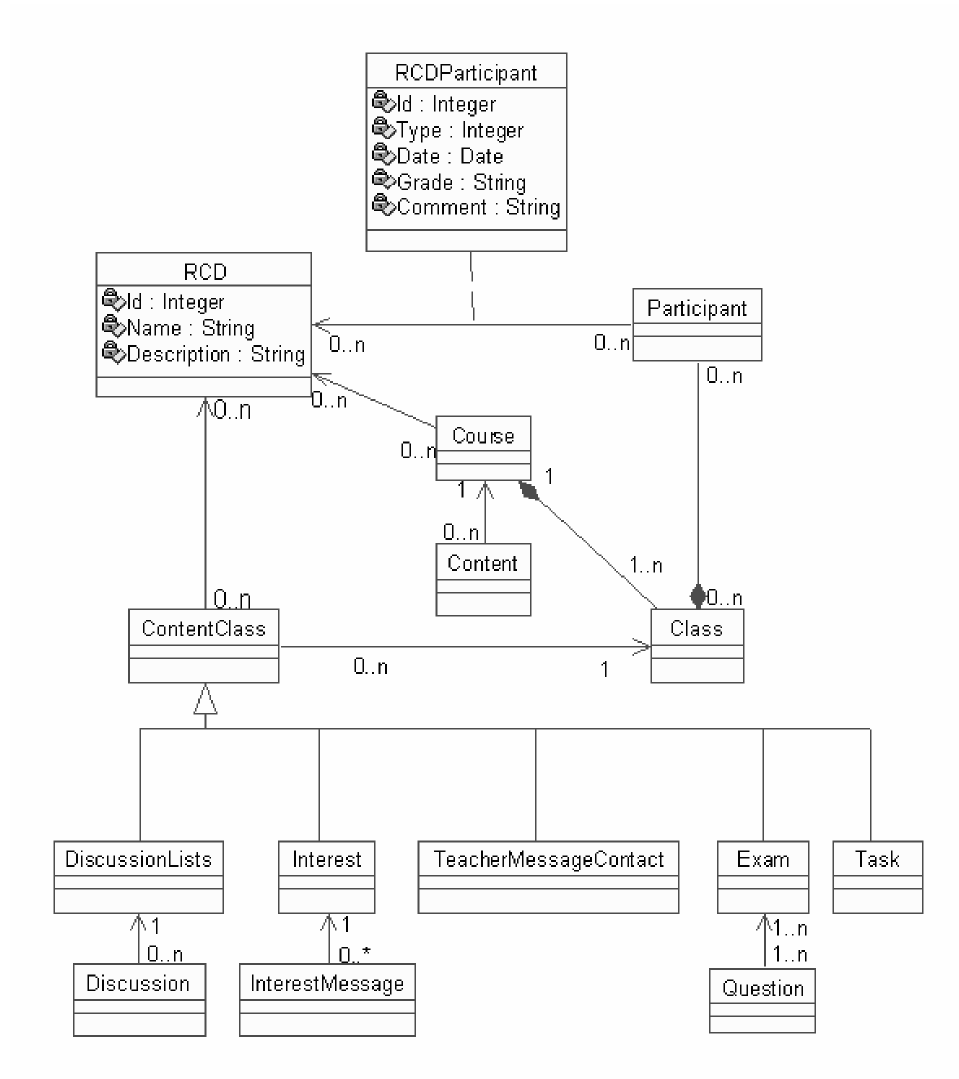

Figure 4 - The employment of Reusable Competence Definitions in the AulaNet environment

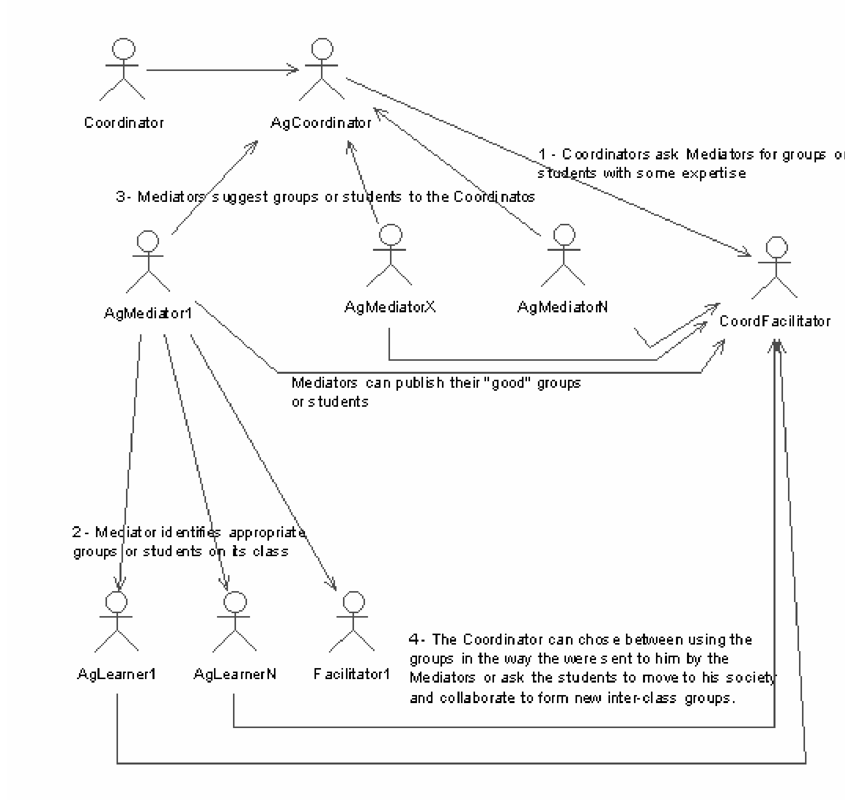

Figure 5 - Group formation using software agents - future step

Table Caption

RCD	**Aspect**	**Degree of Difference**	**Ap1**	**Ap2**	**Ap3**
SQL	Interest	1	2	3	4
Relational Model	Competence	2	4	2	4

Table 1 – A plan for group formation