Using Multi-Agent Systems to Facilitate the Acquisition of Workgroup Competencies

Diego Blanco, Alberto Fernández-Isabel,
Rubén Fuentes-Fernández, and María Guijarro

Abstract. The acquisition of competencies for teamwork is gaining relevance in higher education curricula. However, lecturers face increasing difficulties to monitor and guide the student evolution regarding these issues. As opposed to traditional face-to-face interactions in classrooms, the current educational context promotes students' autonomy to organize their learning activities, with an intensive use of Learning Management Systems (LMS) and other online facilities. This paper presents an architecture for Adaptive Multi-Agent Systems (AMAS) that supports workgroup learning in this setting. It embeds expert knowledge about participant profiles in teams and their evolution, and resources to work the related competencies. An AMAS uses this knowledge to carry out assessments of students and continually monitor their activities looking for patterns in behaviours and interactions. This information allows notifying to lecturers events that need their attention and suggesting them resources to improve students' competencies. The system is currently undergoing the validation of its knowledge basis, which this paper also reports.

Keywords: Workgroup, Acquisition of competencies, Higher education, Multi-agent system, Application support, Adaptation to user, User evaluation.

1 Introduction

Higher education is undergoing important changes triggered by emergent educational needs. Current students are supposed to acquire the knowledge and technical skills of their disciplines of choice, but also competencies required for a successful

Diego Blanco · Alberto Fernández-Isabel · Rubén Fuentes-Fernández · María Guijarro
Facultad de Informática,
Universidad Complutense de Madrid, Madrid, Spain
e-mail: {diego.blanco,afernandezisabel,
ruben,mguijarro}@fdi.ucm.es

J.B. Pérez et al. (Eds.): Highlights on PAAMS, AISC 156, pp. 223–230.
springerlink.com

performance in the modern information societies. Among others, students also need to get capabilities for an autonomous and long-life learning, and to use their skills in multidisciplinary groups that address problems in innovative ways [11].

In order to facilitate this learning, universities are adopting student-centred approaches focused on tailoring the learning experience to the specific needs of each student [12]. They are also increasing the relevance of group activities, as means to create communities of practice between students that encourage their constructive and critical peer collaboration and where they can practice teamwork [6].

The intensive use of software tools, such as Learning Management Systems (LMS) or web 2.0 applications, facilitates the adoption of these new learning strategies, though their support is still insufficient. These tools are intended to facilitate interactions but not to reflect on their results. They provide raw information (e.g. use statistics or authored contents) that lecturers need to analyse to establish the student and group states. From that analysis, lecturers have to find out the best way to guide the evolution of the class. These tasks demand from lecturers increased efforts and analysis skills on teamwork that are not so common in their background.

Our work addresses these issues with an architecture for Adaptive Multi-Agent System (AMAS) intended to boost class teamwork. It supports monitoring, analysis and guiding of the class applying expert knowledge extracted from Social Sciences. These AMAS are integrated with LMS to get information and set up activities using their infrastructure. Lecturers act as the final supervisors that accept and customize the AMAS suggestions on learning resources.

The applied expert knowledge includes participant profiles [1], interaction patterns [5], and resources to expose students to certain experiences [9]. It crystallizes in questionnaires and *social properties*. *Social properties* [4] formalize the description of group settings and workflows following the Activity Theory (AT) [8, 14]. The properties are used to document learning resources and situations, and to detect relevant observations from LMS using pattern-matching techniques. In both cases, the theoretical background of AT supports lecturers interpreting the information.

The architecture includes four main types of agent. *Student* and *group assistant* agents monitor and evaluate individual students and their groups respectively. They try to determine if their targets are accomplishing the proposed activities and fulfilling the related goals. *Lecturer assistants* use the reports provided by other assistants to produce digests for lecturers. They also communicate with *resource advisor* agents to get information on the resources available to articulate and guide teamwork. For instance, advisors can suggest a setting useful to initiate a student in the role of group leader. All the assistants learn from the feedback of their users the kind of information that they should push forward in order to reduce their workload.

This work is illustrated with the discussion of the current validation of the expert knowledge in the architecture. Experiments are undergoing in a Software Engineering course of a Computer Science School in a Spanish university using Moodle[1].

The rest of the paper is organized as follows. Section 2 introduces the basis of our work in AT and social properties. Sections 3 and 4 describe the proposed AMAS,

[1] http://moodle.org

focusing on the expert knowledge used and the architecture respectively. Then, Sect. 5 reports the preliminary results with the AMAS, that Sect. 6 compares with related work. Finally, Sect. 7 discusses some conclusions on the approach and future work.

2 Social Properties

The Activity Theory (AT) [8, 14] is a paradigm from Social Sciences. Its key concept is the *activity*, which is a contextualised human act.

An *activity* [8] is a social process carried out by a *subject*. The subject uses *tools* to transform *objects* into *outcomes*. These *outcomes* can satisfy some subject's *objectives*. There is also a social dimension whose central concept is the *community*. The *community* is a set of subjects that share social meanings and artefacts. Communities are related to the previous elements through *divisions of labour* and *rules*. Both of them represent the knowledge, artefacts and procedures of the community, but have different scopes. The *division of labour* focuses on the current activity. It covers aspects such as roles and skills required in tasks or team organization. *Rules* affect the current activity but are not specifically targeted to it. Examples of them are laws, economic systems or religions. The instances of all these concepts related to an activity constitute its *activity system* [3]. The activity system includes the physical and social context of the activity and its historical development. Social systems are represented as networks of activity systems interconnected by shared artefacts. At this analysis level, there is no difference between physical and mental activities.

The original AT studies are not well suited for engineering settings: they make an intensive use of textual discussions and require a high-level of expertise in social studies to understand and apply them. *Social properties* [4] address this issue. They replace free text with a template that combines diagrams and structured text.

The template describes individual settings and their *related properties*. These links explain the relation between the related properties and the correspondences of their concepts. This allows, for instance, connecting a conflict with a potential solution and explaining how to rework the original setting to get the fixed one.

Diagrams are specified with UML-AT. It is a UML profile with stereotypes for the concepts and relationships of AT. It also adds extra primitives such as the generic *artefact* concept or the *change of role* relationship to indicate that an artefact adopts a new role in a system (e.g. the outcome of an activity becomes the tool of another).

The analysis of information with social properties is made by identifying them in specifications described with UML-AT using pattern-matching algorithms. This process is semi-automated, as an expert makes the final validation of its results.

3 Expert Knowledge for Teamwork

The proposed architecture pursues supporting lecturers and students in the learning activities for teamwork skills, trying to alleviate their workload and improve the results. It performs assessments of the situation of students and groups, and proposes

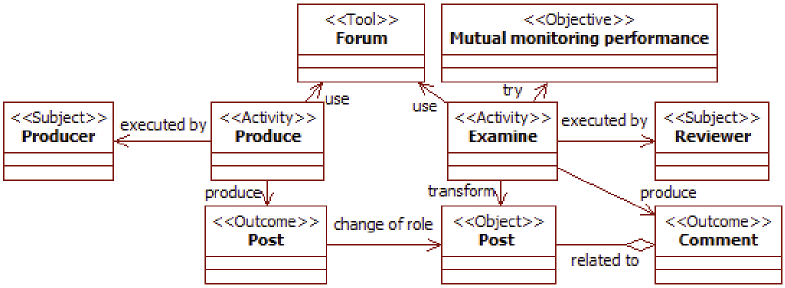

Fig. 1 UML-AT representation of mutual members monitoring in forums.

resources for the learning activities according to the pursued goals. These activities use different types of knowledge.

Our approach mainly considers two works on student's profiles to assess their capabilities. The Belbin's model [1] identifies nine roles that should appear in a well-balanced team that works smoothly and effectively. Our teams are composed following this model and considering that some students are able to play several roles. The *Knowledge, Skill, and Ability requirements for Teamwork* (KSAT) [13] is a catalogue of desirable features of people working in teams. The work in our teams is aimed at giving students a suitable environment to improve their KSAT. For each team, lecturers and students collaborate to provide 360-degree evaluations that include superiors, peers and subordinates. When students gain new competencies, lecturers reconfigure teams to work out other KSAT.

Lecturers facilitate the previous evolution in order to get a successful learning environment. For this purpose, they gather feedback from the interactions happening in LMS, interpret it, and use different resources to promote new dynamics. The knowledge for this process is obtained from literature on teamwork, adapted to LMS functionality, and represented with social properties for a semi-automated use.

Fig. 1 shows an example of property extracted from [7] and adapted to forums about the group behaviour *identifying mistakes and lapses in other team members' actions*. The UML-AT diagram shows two activities, *produce* and *examine*. Both of them make use of the *forum* tool to share information in a team: *produce* to post new *content*, and *examine* to get *content* and include new information as *comments*. The diagram makes explicit the objective pursued by the *reviewer* subject when executing the *examine* activity. The subject tries to achieve the *mutual monitoring performance*, which according to [7] is one of the aspects that make up teamwork.

The architecture uses these properties to define what can be observed and interpret it. For instance, Fig. 1 shows the use of a forum: when a student posts a new element and opens a thread, s/he is executing the *produce* activity with certain artefacts (e.g. information, tools and collaborators). The student assistant agents must provide an implementation able to recognize this activity. When there is a match between actual observations and a property, that match is stored as a *fact* about students and their teams. Additional information is inferred from observations, e.g. the

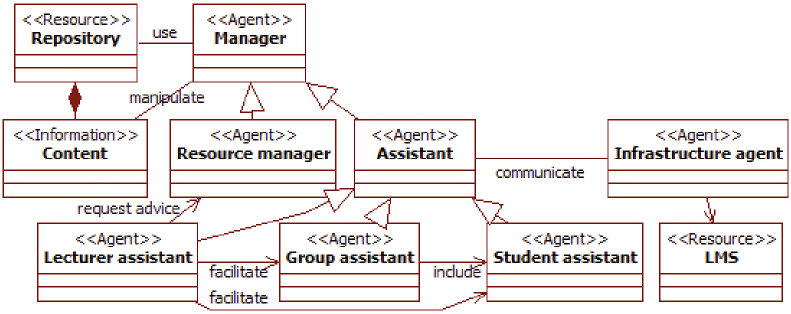

Fig. 2 AMAS architecture.

related objectives. In this example, when a student performs the *examine* activity, it is assumed that s/he is pursuing the *mutual monitoring performance* objective.

Social properties are also used to document the workflows used to achieve certain learning objectives. For this purpose, a property specifies the activities involved in the workflow, their requirements as artefacts, and the expected learning results.

4 The AMAS Architecture

The architecture for AMAS includes two layers of agents. The *infrastructure layer* is the interface with the LMS. The *user layer* includes the four types of agents mentioned in the introduction. Fig. 2 summarizes this architecture.

Infrastructure agents receive events, get information and act on LMS. They use these capabilities to process information and requests on behalf of *manager* agents. For the LMS events, they support a simple subscription and notification mechanism. They also translate information between LMS interfaces and the AMAS internal representation with UML-AT. For instance, in the case of the property in Fig. 1, the infrastructure agent accesses a forum and represents each comment there as an execution of either *produce* or *examine* activities with their related activity systems. Then, the agent notifies the new information to the interested *assistant* agents.

Assistant agents support participants in the class. They store and manage the information required for this purpose in *repositories*. Current types of information include: student *profiles* and their *evolution* [1, 13]; *questionnaires* to determine these profiles and evaluate student and group performance and work [9]; and *social properties* [4] and *facts* observed in LMS described with UML-AT. The *evolution* information includes data about the history of users' interactions with the system, e.g. preferences or activities already performed. The management of this information includes its preliminary interpretation through matching with social properties and deciding the relevant events to notify to users. The *student assistants* forward metrics on their users' activities, e.g. frequency and size of their contributions. The *group assistants* also consider member interactions to detect potential isolation issues. The *lecturer assistant* receives the previous notifications and general digests about course activity.

The lecturer studies this information to decide the resources to apply in the course. S/he uses a *lecturer assistant* to ask the *resource manager* for these resources. The retrieval relies on a simple tagging mechanism based on keywords. The application of the resource is currently a manual task.

5 Preliminary Results

The proposed architecture is undergoing preliminary evaluation in a Computer Science School of a Spanish university. The target course belongs to a 4-year degree and the subject is Software Engineering, which is delivered in the second year. The course has 46 students enrolled, but only 26 assisting to class, 23 men and 3 women, with an average age of 20 years. The LMS used is a tailored version of Moodle.

The initial questionnaires had heterogeneous results, with students that did not consider themselves good for any role and others that were good for several roles. The distribution of Belbin's profiles [1] among the later was: 9 *completers*, 10 *implementers*, 9 *team workers*, 2 *specialists*, 6 *evaluators*, 8 *coordinators*, 3 *resource investigators*, 2 *plants* and 12 *shapers*. This uneven distribution made impossible to set up groups with students that played all the 9 profiles. Lecturers decided to make up 4 teams of 6-7 members with the maximum possible of different roles. These groups have 9, 7, 8 and 7 out of the 9 profiles. The number of members is close to that of required profiles in an effective team according to [1], but still manageable by students with little experience in teamwork.

First observations on the dynamics of these groups showed some malfunctions. After the initial outburst of activity, communication in forums sharply declined as it was expected. More interesting is that the application of the social property in Fig. 1 about monitoring inside teams warned lecturers about 43% of students not posting at all and 21% with no-comment on any post of their team fellows. Among students commenting, it showed that they interacted with other 2 members of their teams in average. This pointed out low values of the *KSA to communicate openly and supportively* [13].

Lecturers considered these rates did not help to start building the required social bonds in teams. The resource suggested for this case was a general post in the forum advising to improve communication in teams. No specific figures on interactions were provided in order to avoid fake messages that were useless for teams.

6 Related Work

This work is related to two main lines of research: analysis of teamwork and semi-automated tools to support it.

Our research tries to provide a general architecture able to incorporate different analysis techniques for teamwork. The current prototype of the AMAS incorporates: works on profiles [1] and skills [13] required for teamwork with the purpose of assessing student evolution; and resources based on good practices for teamwork in learning settings [9]. Nevertheless, other works regarding, for instance, team evolution [3] or cooperative learning [6], have not been included. A key problem here is building a cohesive theoretical framework for the support provided by the AMAS. The research area is highly fragmented, with works focused on very specific aspects of teamwork and relying on different and partial theories. Moreover, most of works make only public their results, but not the complete description of the tools they used to elicit information, e.g. questionnaires or interviews. This makes difficult applying their analysis in other works.

Regarding tools, literature mainly focuses on the study of standard applications such as LMS. The common conclusion is that lecturers need to make an important effort with these tools in order to get the proper teamwork settings [10]. There are some aids, but their utility and scope are limited. The review in [2] highlights that acquiring teamwork competencies is a multi-faceted problem, and the related support tools need to incorporate and integrate its requirements. Our work addresses several of the open problems pointed out there. It tries to provide an open and flexible architecture that supports new collaboration and analysis methods. Other interesting work from the perspective of monitoring and analysis is [7]. It reports different visualizations for the use information of tools such as wikis or subversion systems. These visualizations provide insights on who is interacting with whom, and the frequency and size of people contributions.

7 Conclusions

This paper has presented an architecture for AMAS intended to support learning of teamwork competencies. It pursues several goals discussed below.

The architecture is designed to facilitate the integration of different theoretical perspectives and support infrastructures. Agents encapsulate well-defined tasks and the open AMAS easily support replacing them by others with similar skills. Knowledge is declaratively specified with UML-AT and social properties. This allows an iterative and incremental definition of the events and workflows to consider, grounding them in a widespread social theory. It also supports the description of resources applicable in a course and their goals and learning outcomes.

Providing agents the ability to manipulate the previous knowledge makes possible the adaption of the overall system to the evolution of the course. This is the case, for instance, of analysing the events to notify or the resources to apply to certain students according to the current state of their teamwork competencies. This adaptation and the flow of information in the system reduce the workload required to set up and carry out this kind of activities. Lecturers need to devote less time to monitor students, analyse their work and propose activities, and students get feedback from the system on potential fails or tips for their teamwork.

The proposed architecture is undergoing validation. Preliminary results have shown the usefulness of students' profiles and monitoring of their interactions to prevent some common malfunctions in their teams.

The approach still presents multiple open issues that need further work. First, the integration with LMS is quite limited. It is focused on the public exchange of information in forums and uploaded/downloaded resources. Additional interactions happen through mail or in the contents of resources, and they are relevant for teamwork. Second, the applied knowledge is basic and fragmentary. Research is needed to include a richer model on how teams interact and students learn from it. This would allow designing specific activities for teams and individuals to acquire certain competencies. Finally, additional courses are going to be enrolled in the experiments to validate the architecture and knowledge in a wider setting.

Acknowledgements. This work has been done in the context of the project ”SOCIAL AMBIENT ASSISTING LIVING - METHODS (SociAAL)”, supported by Spanish Ministry for Economy and Competitiveness, with grant TIN2011-28335-C02-01. Also, we acknowledge support from the Programa de Creación y Consolidación de Grupos de Investigación UCM-BSCH GR58/08.

References

1. Belbin, M.: Team roles at work. Butterworth-Heinemann (1993)
2. Dimitracopoulou, A.: Designing collaborative learning systems: Current trends & future research agenda. In: Proceedings of the 2005 Conference on Computer Support for Collaborative Learning (CSCL 2005), pp. 115–124 (2005)
3. Engeström, Y.: Learning by expanding: An activity-theoretical approach to developmental research. Orienta-Konsultit (1987)
4. Fuentes-Fernández, R., Gómez-Sanz, J., Pavón, J.: Understanding the human context in requirements elicitation. Requirements Engineering 15(3), 267–283 (2010)
5. Gersick, C.: Time and transition in work teams - toward a new model of group development. The Academy of Management Journal 31(1), 9–41 (1988)
6. Johnson, D., Johnson, R., Smith, K.: Cooperative learning: Increasing college faculty instructional productivity. ASHE-ERIC Higher Education Report 4 (1991)
7. Kay, J., Maisonneuve, N., Yacef, K., Reimann, P.: The Big Five and Visualisations of Team Work Activity. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 197–206. Springer, Heidelberg (2006)
8. Leontiev, A.: Activity, Consciousness, and Personality. Prentice Hall (1978)
9. Michaelsen, L., Knight, A., Fink, L.: Team-based learning - A transformative use of small groups. Praeger Publishers (2002)
10. Nijhuis, G., Collis, B.: Using a web-based course-management system: an evaluation of management tasks and time implications for the instructor. Evaluation and Program Planning 26(2), 193–201 (2003)
11. Reichert, S., Tauch, C.: Bologna four years after: Steps towards sustainable reform of higher education in Europe. European University Association (2003)
12. Scott, P.: The Meanings of Mass Higher Education. Open University Press (1995)
13. Stevens, M., Campion, M.: The knowledge, skill, and ability requirements for teamwork: Implications for human resource management. Journal of Management 20(2), 503–530 (1994)
14. Vygotsky, L.: Mind in society (1978)