An on-line educational system for Blended Learning. Magadi

Ainhoa Alvarez, Isabel Fernández-Castro, Maite Urretavizcaya
Dpt. de Lenguajes y Sistemas Informáticos, UPV/EHU
Apto. 649, 20080 Donostia,
{ainhoa.alvarez, isabel.fernandez, maite.urretavizcaya}@ehu.es

Abstract

MAGADI is an educational environment whose main goal is to provide training activities, on several subject-domains, adapted to the student according to his/her acquired knowledge, previous interactions and cross-curricula competencies. This system tries to answer the needs of blended-learning (B-Learning) by cohesively integrating in a Web environment, the student learning characteristics obtained from both the automatic inferences of an adaptive e-learning system and those detected in face-to-face interactions during traditional learning sessions.

This paper describes the general agent-based architecture of MAGADI, the basis of its knowledge representation and the elements and behaviour of the student workspace. All together make MAGADI an on-line component candidate for B-Learning.

1. Introduction

Since early seventies the computer-aided instruction area has been constantly evolving to respond to new educational requirements. The Intelligent Tutoring System (ITS) paradigm aroused in middle eighties as a solution for individualized learning. These systems try to emulate the human instructor by helping the student in an adaptive way, taking into account the student’s characteristics and behaviour. They share a common structure composed of four main modules: Domain Model, Student Model, Pedagogical Module and Communication Interface. Although this approach showed positive results during a time, it was exceeded by new educational tendencies, such as learning multi-domain competencies, constructivism vs. instructivism, or collaboration.

Currently, Learning Management Systems (LMS) form the major tendency adopted by Universities as they cover basic needs of teachers and administrative personal [6]. Diverse systems have been developed in this field, such as Moodle [2], WebCT or Blackboard [1], but most of them centre mainly on administrative and management tasks. Although they allow a subsequent automatic data recording that might be advantageously used for course improving, at the moment LMS scarcely provide adaptation to the learners.

The application of ITS adaptation capabilities in web context gives raise to the Adaptive Web-based Educational Systems (AWBES). In these systems, the adaptation mechanisms must be defined during the development process, in such a way that adding improvements on the pedagogical knowledge can be addressed only by new developments. So, we would say they have a static adaptation style [16]. An AWBES includes a restricted set of functionalities, so, generally, the teacher needs to use several of them to support his/her teaching style [6]. In this context, although several platforms, such as KnowledgeTree [6] or MEDEA [18], allow the integration of different AWBES, still a small change on the adaptation style of one of them requires new development and integration processes.

On the other hand, recent studies [10] point out that the reason of the apparent absence of success of current e-learning systems is the lack of a tighter integration among traditional classrooms and on-line learning environments. There exists a consensus that technological solutions should not neglect the important role of the teacher, but facilitate his/her work. Besides, from the student viewpoint, it should be interesting to interact with a system integrating both learning approaches taking advantage of both of them.

Another main educational trend, which is clearly visible in the reforms promoted by the Bologna Process in order to define the European Higher Education Area, is the changing focus
from contents to skills or abilities. The objectives of a programme are specified in terms of the learning outcomes and competences to be acquired. These competences can be either related to a specific subject or cross-curricular, what allows taking a unified viewpoint study that profit from inter-subject synergies.

Recently, the Blended-learning (B-Learning) approach has proposed learning scenarios that combine face-to-face instruction and computer-mediated instruction [9]. We believe that the opportunistic application of B-learning inspired tools can solve the novel depicted educational landscape, for which MAGADI [3] is our proposal. This is a domain independent, open and adaptive learning platform whose adaptation style can be anytime modified by the teacher, and where different components can be integrated providing a technological solution for B-Learning with real integration of online and offline activities.

This paper describes the general agent-based structure of MAGADI and its student oriented functionalities. It is organized as follows. The first section outlines MAGADI and a deeper description of its knowledge representation schema follows. Then, the student workspace is presented to better show the system behaviour. Finally the very first evaluation of the system and some conclusions are presented.

2. MAGADI and B-Learning

The adaptivity of a technological component for a B-Learning solution should allow the flow of information between the on-line and off-line environments. This is, the results of the on-line activities ought to be taken into account by the off-line activities and the other way round [12]. For example, the results of online activities, such as exercises, should be available for the teacher to prepare the following face-to-face interaction, whereas the results of the last one should be included into the online system to be taken into account in subsequent interactions. In this way the educational experience, taken as a whole, profit from the advantages of both, whereas the other one solves the drawbacks of each.

To obtain these results, adaptivity in the online component of a B-Learning system must not only be based on the student-system interactions (as it happens on classical ITS or AWBES). It is also imperative to allow the adaptivity style to be both flexible, based on criteria specified just for the current session -i.e. just in time adaptation-, and anytime modifiable directly by the teacher – in order to obtain adaptivity based on face-to-face interaction. Moreover, an integral approach for actual education tendencies, such as the defined by the Bologna Process, demands learning environments able to integrate the inter-related representation of several domains in order to share cross-curricular competencies.

To answer the mentioned needs, MAGADI allows an interleaving of uses -in the sense that authoring and learning can happen at the same time and so influence one another. This is opposed to the strictly and usual sequential life cycle of ITS, where an authoring phase establishes the outcome system, with a concrete set of adaptivity criteria, which is then used by the students in the subsequent learning phases.

In order to obtain the mentioned goals, the system is organized around three related workspaces. It dedicates an agent-based [13, 17] workspace for each target user –teacher, student, author– and provides a common shared background composed of several databases. These support knowledge and information flow among the different workspaces and contain both domain dependent and independent knowledge about the represented areas. Figure1 shows the type of interaction of the target users with the shared databases described in section 3.

These properties allow MAGADI to work with cross-curricular synergies and make it a B-Learning environment with effective blend of delivery methods where students and teachers interact by means of a flexible and dynamic dispatching of learning sessions as it is described in section 4.

3. Knowledge Representation in Magadi

Adaptive on-line educational systems are supported by three main categories of knowledge -student, subject domain and pedagogical. MAGADI shares this categorization and organizes it according to its direct predecessors INTZIRI and DETECTIVE [9]. These systems use a set of knowledge bases whose classes and instances form the basis for deriving a multi-layer meta-ontology. In contrast, MAGADI organizes the related information by means of several stand-
alone database servers, at the same time that maintains the multi-layer ontology.

<table>
<thead>
<tr>
<th>Teacher workspace</th>
<th>Student workspace</th>
</tr>
</thead>
<tbody>
<tr>
<td>*Defines the pedagogical criteria to be used for students, domains and groups</td>
<td>Pedagogical</td>
</tr>
<tr>
<td>*Introduces the domain content</td>
<td>Domain</td>
</tr>
<tr>
<td>*Can visualize and modify the student knowledge</td>
<td>Student</td>
</tr>
<tr>
<td><strong>Uses the stored information to configure the learning session</strong></td>
<td><strong>Uses the stored information to show domain contents to the student</strong></td>
</tr>
<tr>
<td><strong>Uses the stored information, actualises the information taking into account the learning</strong></td>
<td></td>
</tr>
</tbody>
</table>

Figure1. Knowledge categories’ use

On the one hand, ontologies [8] provide a common vocabulary for a fluent communication among the agents, and also an underlying conceptual schema for the defined databases. So, ontologies and databases taken together provide a data structure appropriate for information description and exchange. Moreover, the created ontologies integrate the CLAI instruction theory model, making MAGADI a theory-aware system [11].

On the other hand, the database servers allow anytime access and modification by both the teacher and the interaction and tutoring inferences arising from learning activities coming from any other B-Learning component. In either case, the information changes are instantly considered by MAGADI, what is fundamental for a B-Learning solution to offer a real integration of face-to-face and on-line learning.

3.1. Domain Knowledge

Several works propose a multi-domain approach for adaptive systems, for instance the pluggable domain modules described in [19] tries to solve this question. However, as the domain models are “plugged” in an independent way, the resulting system cannot profit from their interrelationships and impedes the representation of cross-curricula competences.

We have defined a general and distributed domain repository for MAGADI, supported by both a database and the domain meta-ontology. They articulate the set of basic learning units (elements to be learned), the instructional objectives (skills or competencies) to be developed by the student and the inter-element pedagogical and structural relationships.

The database stores information about several subject-domains and the interrelationships among their components so that dedicated knowledge agents on the different workspaces allow exploiting inter-domain competences.

Figure2 shows a fragment of the domain meta-ontology as defined in PROTÉGÉ, and figure3 shows the associated part of the domain database.

3.2. Student Knowledge

In order for the teacher to influence the system and vice versa, the MAGADI adaptive behaviour requires to store knowledge and skills about students on several subject domains, and to allow the teacher to include his/her findings derived from face-to-face interactions. The student knowledge is represented by a generalized overlay model; it comprises the whole set of subjects learned by the student, in such a way that each subject domain includes its particular information (i.e. presented elements, taken activities, acquired basic learning units, and so forth) and inter subject issues. The student database can include also some domain-independent information such as teacher or own comments. This structure resides in a centralised user modelling server [6, 7, 14, 15]. Thus, it separates the user modelling functionality from the user-adaptive application systems obtaining some interesting advantages [15]: reduces redundancy and allows different applications to anytime actualise and dispose of the information on the server.
These servers are not part of but rather independent from application systems, i.e. they are not functionally integrated into the application but communicate with it [15]. So each MAGADI's workspace provides a knowledge agent for server interaction capable of generating a view as required for the current educational use.

3.3. Pedagogical Knowledge

The pedagogical knowledge required to provide adaptivity in MAGADI falls into two categories: first, the criteria used to adapt the instructional plan that guides the learning session [4] and to generate the initial set of exercises for practice sessions; and, second, the criteria required to determine the availability of learning session types'. These criteria are stored in the Pedagogical Database.

![Diagram of Conceptual Model](image1)

![Diagram of Domain Ontology](image2)

![Diagram of Domain Database Model](image3)
Several planning approaches give rise to a range of tutoring styles. In this aspect, MAGADI is based on the general schema of the IRIS tutors [4] which share a dynamic incremental planning schema of minimum commitment; some others take similar approaches [5]. IRIS tutors integrate a pragmatic cognitive theory of instruction, the CLAI Model, and are based on a minimum agreement instructional planning process where, a priori, the tutor only plans the initial content or basic learning unit. The planning process is completely opportunistic, which in this context means the student’s request can be accepted at any time. A rule-based planner performs the process of building the Instructional Plan, which in turn determines the activities of the tutor. MAGADI provides a configuration stage in the student workspace that allows varying, among sessions, the underlying criteria for generating the instructional planner that will guide the student activities. This behaviour, together with the possibility of anytime changing the pedagogical database by the teacher, provides the system with a great flexibility and an adaptivity style based also on the results of face-to-face interaction.

The idea of learning session types provides the system with an extra degree of flexibility. We have identified some session types, such as exam, guided session or free exercise session, which can anytime be extended. The student can select any of them but it is also possible to prune the availability of session types. For example, if today Paul selects to work on the Geometry domain but the teacher determined an exam in advance, today Paul will not be able to select any other session type but the exam over that subject domain.

4. Student Workspace

The system behaviour can be analysed from any of its different workspaces. In this section we describe MAGADI from the student workspace viewpoint as it helps to better show the system main characteristics. In MAGADI, each workspace is composed of several information, task, interface and knowledge agents. Concretely, the basic agent architecture of the student workspace follows the classical module structure of ITS, such as INTZIRI [4].

For the student to develop a learning session, his/her workspace must get through two stages: configuration and dispatching. The Configuration stage establishes the adaptivity criteria that must be used in the immediate session (taking into account the updated information in the database, i.e. just in time adaptivity), and makes up the system agent structure. So, next, on the dispatching stage, the just created agents will conduct the session development. Some agents, those devoted to knowledge management, participate in both stages (see Figure 4).

An Interface Agent allows communication between the user and the system and a Manager Agent carries out coordination activities. Three knowledge agents are devoted to control communication events with the shared databases Domain, Student and Pedagogical. Figure 4 shows the student workspace’s architecture, and next the concrete agents of each stage are described.

4.1. Dispatching Stage

The Dispatching Stage contains the agents to conduct a student session: a Planner –generated in the configuration stage as described next– and an Evaluator.

The main actions developed by Planner Agent are to select the subject material to show next and the next pedagogical action to be carried out. This selection process is based on the IRIS [4] rule-based planner due mainly to its domain independent approach. This was a main property required by our planner as it must be capable to work with different subject domains.

The system evaluation capabilities are provided via the Evaluator agent that links the diagnostic capabilities of DETECTIVE [8]. Again, this system was selected mainly due to its domain independent evaluation capacity.

The agents of the dispatching stage are generated by the configuration stage at session beginning, taking into account the updated information of the databases, so they can vary among sessions.

4.2. Configuration Stage

Three main agents shape the Configuration Stage, which makes the structure for the dispatching session: Session Selector, Pedagogical Selector and Exercise Selector.
The pedagogical knowledge that will be active for the current planner is determined at the beginning of each session, taking into account the information in the pedagogical database. The Session Selector determines the set of available session types for the target student on the selected subject. It uses the adaptation criteria stored on the Pedagogical database, which is obtained via the Pedagogical Agent. The Pedagogical Selector determines the set of rules to be used in the current session by the planner; it communicates with the pedagogical database agent in order to obtain the most up to date pedagogical constraints. As the teacher can anytime modify this information, several criteria can produce different planning behaviours to be used with the same subject matter and student, but on different moments. As a direct consequence, MAGADI facilitates a progressive authoring, even when the system is being used, in the same way as good teachers change their teaching strategy according to new emerged parameters –bad student results, for instance.

Finally, the Exercise Selector selects the initial set of exercises for practice sessions using the information stored in the shared background.

All the configuration agents obtain the information from the shared databases. As a direct consequence, this configuration stage allows the on-line activities to take into account not only the results of the previous online sessions (with Magadi or other online systems that modify the shared databases) but the results derived from offline activities (introduced in the database by the teacher).

4.3. An Example

Let us suppose that the student selects to work on a given domain, then the Session Selector must provide the available session types for that student on the selected domain. For that, it must analyse the pedagogical, domain and student databases to determine whether, for example, the student has a near exam, or there is a near lecture for which the teacher has defined the prerequisites; thus, it can decide which session type to recommend (review of some concepts, guided exercises and so on). Figure 5 shows the interaction among the agents for this phase.

Once the student decides to follow a type of session, the Pedagogical Selector agent taking into account the current domain, the selected session type and the student knowledge on the concepts to be developed generates a Planner agent. If the session is an exercise based one, the planner needs to request the Exercise Selector agent the initial set of exercises to begin with. Once all these tasks have been developed the system begins with the dispatching of the selected session.
5. Prototype Implementation and Evaluation

A prototype of MAGADI has been developed using JADE, JSP and Jess. The basic behaviour of all the agents has been implemented; especially, the basic ontologies and the agent interactions. The agents reasoning capability is implemented as Jess behaviours whereas the interfaces of the different users have been implemented as Java Server Pages, and PROTEGE has been used to define the ontologies. Finally, MYSQL has been used as the underlying database management system.

Once the mentioned elements have been implemented and the basic behaviour tested with the JADE testing tools, we have developed a very first evaluation of the prototype that will be followed by an evaluation with actual students.

The Teacher Workspace has been tested while building the subject domain *Introduction to Programming-loops*; besides, new students and some basic pedagogical knowledge definitions have been created by means of a restricted authoring interface based on JSP pages. After that, we have populated the subject domain database. The KADI [8] environment has been used to define the exercises as this first prototype is specially centred on practice sessions.

The test of the Student Workspace has been focussed just on the technological aspect, so some simulated students have been used to test different interactions among system agents, during both the configuration and dispatching stages on different exercise-based session types. The results of these preliminary tests have been positive.

6. Conclusions and future work

Blended Learning is an educational paradigm that can answer the needs of actual educational tendencies. However, nowadays, the main on-line components of such environments are LMS, which do not adapt to the target students. Adaptive systems bring two main problems: their mono-domain property and their behaviour based on fixed adaptation criteria. To tackle both problems, we have presented MAGADI, which is based on results from the authoring tools field and the agent technology.

MAGADI is founded on the concept of blended-learning and tries to cohesively integrate in a Web environment the student learning characteristics obtained from both the automatic inferences of an adaptive e-learning system and those detected in face-to-face interaction during traditional learning sessions.

The main MAGADI objective is obtained by integrating three user-oriented workspaces by means of a set of shared databases (see section 3) and by splitting two functional stages on the student workspace, configuration and dispatching (see section 4). The configuration stage allows the generation of tutoring knowledge dynamically at execution time according to the changing characteristic emerging for each session. This is, it provides just-in-time adaptation to the student’s current learning necessities, determined by his/her
learning history and teacher’s objectives. It is done by allowing the teacher to anytime modify the system pedagogical characteristics. Moreover, MAGADI extends the IRIS student and domain models in order to allow them to represent multi-subject elements. In addition, the planner that determines dynamically the instructional plan on the student workspace follows a set of domain-independent rules so that it can work with different subject-matters.

In the near future we plan to test the prototype in the course of databases in our University to contrast the claimed benefits of the system in a B-learning environment both from the student and the teacher perspective.

Acknowledgments

This work is supported by the Spanish Ministry of Education of Science MCyT, grant TIN2006-14968-CO2-01, and the University of the Basque Country EHU 06/111.

References