# An adaptive Multi-Agent based Architecture for Engineering Education

Jara-Roa D., Valdiviezo-Díaz P., Agila-Palacios M., Sarango-Lapo C.
Virtualization Department
Universidad Técnica Particular de Loja - UTPL
Loja, Ecuador
{dijara, pmvaldiviezo, mvagila, cpsarango}@utpl.edu.ec

Rodriguez-Artacho M.
LSI Department
Universidad Nacional de Educación a Distancia (UNED)
Madrid, España
miguel@lsi.uned.es

*Abstract*— **This paper proposes an agent-based adaptive Architecture to extend *Moodle* in order to support instructional decisions and adaptive behaviour in engineering education. The paper describes the characteristics, functions, and interactions of the agents which take part in each module of the adaptive architecture. In addition, we describe the origin and function of ToDei, the proposed intelligent agent for Instructional Decisions Making. This agent is in charge of collecting information generated by the rest of agents and deciding what is best for the final users, tutors and students, taking into account their attitudes towards the learning environment.**

***Keywords*:** *Intelligent agents; adaptive educational systems; ntelligent systems; user modelling; tutor modelling*

## I. INTRODUCTION

One of the challenges of engineering education is to tackle with complex learning experiences which in many cases include the use of a diversity of tools and environments, drill and practice processes, virtual laboratories, etc. In this context, the use of adaptive content has been proven to be an important contribution to facilitate learning [1][9][10], provided as an extension of a given learning architecture. From all the VLE open sources available in the web, Moodle is the most widespread and used around the world. Currently, there are 56357 sites from 210 countries registered as Moddle[1] sites. They are supported by a development community that provides current updates and support. Due to the previously mentioned services and to the portability and modularity that characterizes this community, it has been easy to present a proposal with an adaptive focus that contributes to the growth and exponential development of the VLE open source with tendency to an Adaptative Educational System (AES).

The Moodle approach is based on three main components: the professor, the classroom in which the educational process is given, and the students. They work together in the following way:

The professor looks for and produces all the information necessary for the course development. He/she organizes it in weekly blocks or in thematic blocks and enters the topic orientation, the supporting resources and tools (files, documents, videos, audio, etc.), and the activities that must be developed by the student.

The student is in charge of developing the proposed activities, downloading or checking the resources and interacting with his/her partners according to the orientations received by the professor. The teaching-learning process is developed in the classroom, and the information generated here is stored in a relational data base that is fed and accessed through the user interface. The Moodle's structure allows providing facilities in order to give each one of the student's different resources and activities, to give advice to students, and to facilitate the interaction between students to students, students to professors and vice - versa. If there is not information related to the user preferences, the navigation sequence, the document evaluation, and the knowledge level, the teaching-learning process will be developed in a general way for all the members of a course. And a personalized service will not be offered to each student.

One alternative to such problems is to focus Moodle on students' needs, trying to adapt this system to each one of the students and training them according to their learning styles as well as adjusting the system to the students' needs. For this reason, a structure that allows Moodle to be adaptive is necessary.

The proposed architecture is based on the main areas of adaptation defined in [2], providing presentation and navigation adaptation using intelligent agents associated to different modules in Moodle. This multi-agent methodology has recently appeared as a good companion of adaptive distributed educational systems. User adaptation using intelligent agents has been developed using several approaches. For instance as users' self test, like in [3] that provides an adaptive testing tool to fit the student's level of

[1] Moodle.org. http://moodle.org/sites

knowledge, where questions are generated automatically; or integrating agents in VLE as in [10] where educational content is made available by means of a process of re-categorization carried out using intelligent agents within the VLE.

Furthermore, it is considered a multi-agent structure that needs to be easily applied to the modular architecture of Moodle and offers a dynamic adaptation through learning. This learning is developed by each one of the agents that intervene in each component .

## II. ADAPTATIVE LOGICAL ARCHITECTURE PROPOSED FOR MOODLE

The proposed architecture of AES for *Moodle*, has four principal modules: tutor module, student module, user interface module, and knowledge base.

Each of the first three has an intelligent agent that performs tasks for each module. For communication and interaction among agents of each component, the agent called ToDei has been defined. Its goal is to take the information of each module and decide the best method of instruction for the final user. In the knowledge base module, the development of one or more agents can be considered, depending on the way the information is structured (taxonomy or ontology, for example). For the present case, no agent has been defined in this module since its basic functions will be performed by the ToDei agent.

For each agent, the principal functions are defined. Therefore, the one who works on the tutor module has didactic-pedagogical functions and tutor modelling, and the student module carries out the creation of student models and information updating. The user interface module (student-tutor) determines the most appropriate interface for each user based on the hardware and software used for the connection. The ToDei agent communicates and determines the best instruction for the user.

The components of the different modules proposed for AES are based on research based on an analysis of available information on the Moodle virtual platform, which permits definition of the necessary task adaptation to acquire an adaptive approach. As stated clearly before, the components and intelligent agents that will intervene in an adaptive architecture, using the Moodle virtual platform as a base, are shown.

### A. Tutor Model

The instructor, who is an actor in (traditional) Moodle, becomes a component in this type of architecture, in which the best way to make information available to students is identified. This function is performed by an intelligent agent, which will make decisions according to various variables that could be considered to provide knowledge to students.

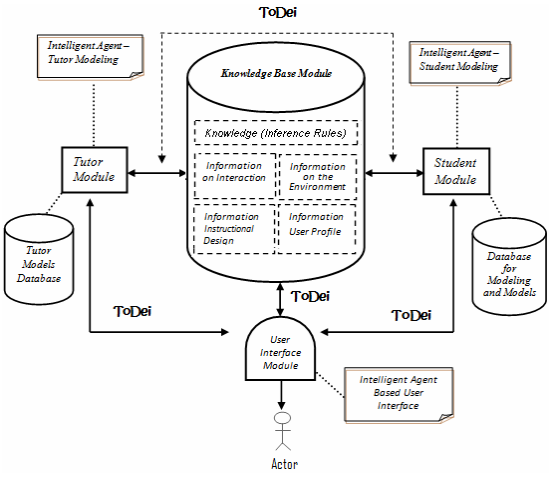



Figure 1. Moodle's Adaptative Logical Arquitecture

This activity is supported by the instructional design and the different versions of the material made by instructors [8]. The different tutor models generated by the agent are stored in this module. At the same time, this agent is in contact with the knowledge base module in order to retrieve and store information. The intelligent agent, which is associated with this module, is the intelligent agent for tutor modeling that must perform the following functions:

- Pedagogical-didactical (teaching style)
- Tutor modeling (implementation of contents)

*Intelligent Agent for Tutor Modeling*

The tutor modeling agent has been designed to perform the following functions:

- *Didactical-Pedagogical*. Each instructor has his/her own style to encourage learning among students, which is perceived differently by each student. This situation is what the intelligent agent attempts to simulate and learn in order to guide students in what and how to learn.
  According to students' acceptance and response to a particular style, the agent will be able to classify what teaching form is the most accepted by a particular student profile. Thus, once the agent has learned, it can choose the most appropriate style for each user based on his or her profile.
- *Tutor Modeling.* The main feature of this function is the analysis of the organization and implementation of contents according to the experience that the agent has acquired. This process will begin with a modeling that gathers information on activities and resources proposed to students and on their success rate. Based on this experience, the agent will learn the best way to implement contents.

After analyzing this information, some tutor models, which can be applied to determined student models, can be defined. Tutor models may vary depending on the level of learning and experience acquired by the agent. This will permit the definition of more concrete models that can be best adjusted to a particular type of student.

Considering the classification made by [4] and [5] regarding the structure of an agent, the tutor modelling agent can be categorized within the *informed agents that maintain a design of the world* due to a) the constant updating of the environment in which this agent operates, b) the updating that must be done in the tutor modelling, and c) the decisions about what content should be offered to the student.

The use case according to the task that this agent must complete is shown in Figure 2:

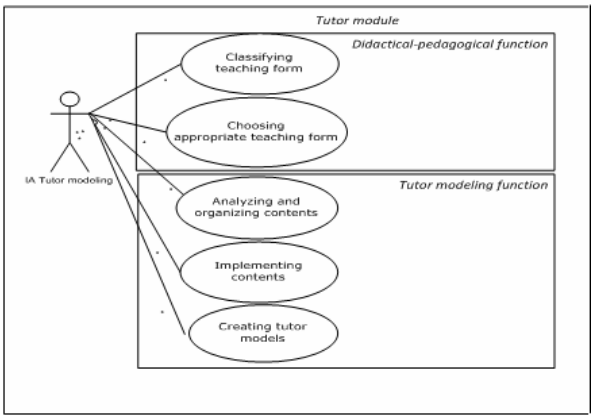



Figure 2. Use Case of Tutor Model

### B. Knowledge Base Module

This module has the initial knowledge of the system, expressed in statements of general validity as like as inference rules or probability distributions [7], these are used by the agent to infer a conclusion or new knowledge, used various information sources such as a) student's personal data, b) interaction data, c) environment data, and d) information on instructional design. This Knowledge can be organized into different structures such as taxonomies, conceptual maps, thesaurus, and even ontologies.

### C. Student Module

Regarding AESs, students generate the underlying information for adaptation. For this reason, in an adaptive architecture, this actor turns into an independent module in which, based on the students' characteristics, needs and preferences [6], different models are generated.

For this module, the following use cases can be identified:

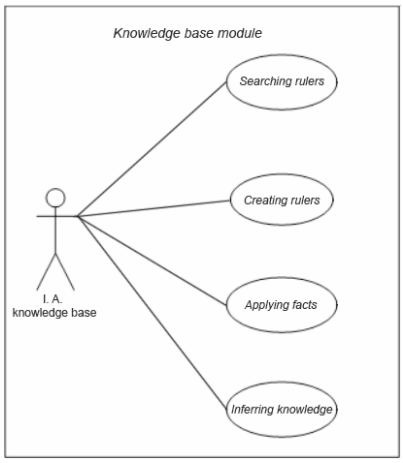



Figure 3. Use Case Knowledge Base Module

*Intelligent Agent for Student Modeling*

The agent for student modeling performs some functions in this module. These functions include:

- Creation of Student Models. Creation of student models based on a previous model, in which explicit and implicit variables are defined and include navigation environments, learning styles, learning levels, cooperation levels and interaction levels. Once models have been created, the agent will be able to categorize the different student profiles according to the model with which they share similarities. This will favor a more significant learning experience.
- Information Update. This function collects information about the student from the moment the student accesses the system. This information is updated by the agent, which is in charge of monitoring various activities, routes, and actions of the student in the system. After comparing some characteristics that students accumulate in their profiles, and if necessary, the agent will be able to change the model to which the user was assigned and determine the progress of the user in the system.

This agent interacts with the knowledge base through the instructional decision making agent (ToDei), from which data of each profile are retrieved and saved once the student has been identified in the system or when the student exits the system.

Returning to the classification made by [4], the agent for tutor modelling can be classified within the *informed agents that maintain a design of the world* due to a) the constant updating of the environment in which this agent operates, b) the updating that must be made in the student modelling, and c)

the categorization of student profiles regarding the models that were created.

The use case diagram for this agent is shown below:

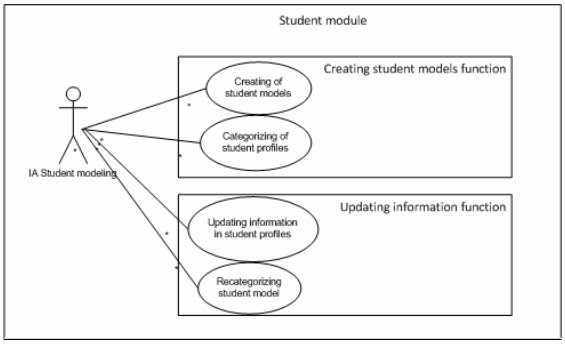



Figure 4. Use Case Studen Module

### D. *Interface Module for Users*

This is the component that shows all the information to the students, trying to capture their attention and keeping them motivated, through redaction of messages type "Inverted Pyramids" [11]. In this module, we identify characteristics such as the type of browser used, type of device used for access, and the available connections.

*Intelligent agent for user interface*

The objective of this agent is to determine the best interface to be offered to each user based on the hardware and software used for the connection.

The means of connection, the software, and the equipment used are different for each user and for this reason each has different presentation needs. For example, if a student accesses the system from a cell phone, he would only need the most necessary files, but if a student accesses from a carry-on computer with a good bandwidth, he would be able to watch videos and animated films and listen to audio files.

The intelligent agent for user interface should learn these conditions and determine the most appropriate type of system interface for each one. This learning can be acquired when students can interact with the offered interface and when the identification is accepted or denied. This will help classify it as appropriate or inappropriate.

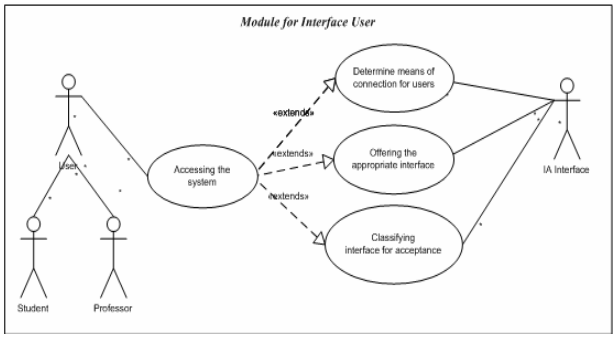



Figure 5. Use Case User Interface Module

Continuing with what [4] stated, this agent has been categorized inside the *informed agents that maintain a design of the world*, since the constant updates from the environment that surrounds it demand different answers depending on each case. For example, it must be able to recognize if a user accesses from a PALM and then accesses from a PC the same day.

The use case diagram that corresponds to this agent is presented below.

### E. *Intelligent Agent for making instructional decisions (ToDei)*

The adaptive architecture that has been proposed is a dynamic structure and multi-agent that allows learning, communication, and independence of each component of the system. In all the components there are agents that facilitate the execution of the process and a particular defined process, but in order to fulfil the requirement of offering the use of the adaptive effect, the communication and interaction of all the components of the architecture are necessary. Consequently, the intelligent agent ToDei is used.

The objective of this intelligent agent is to fulfil these functions as well as to transmit the content to the user. Furthermore, considering the characteristics and greatest needs, it decides the best way to offer information generated in this process.

ToDei receives information through the module for user interface, specifically the one generated by the agent for interface. It is the first to have contact with the user because it communicates with the module for students where the information regarding the profile (stored in the knowledge basis) and the model student (stored in the module for students) are restored. They are then sent to the module for the tutor where, according to the tutor model assigned for the model student, the most appropriate instructional design is offered. Finally, ToDei displays the results of this process to the user. At this stage, the intelligent agent chooses how to show through the interface the content generated based on the style best suited to the learning process of the student.

As observed, ToDei is the only agent present in this architecture which has the function of restoring, transmitting, and sending information through the different modules of the system. Consequently, according to [4] and [5], it has been categorized as an *informed agent that maintains a design of the world* since its function of interacting with all the

components of the system forces it to update its internal state based on constant changes of the environment.

Structurally, the intelligent agents are formed by sensors, effectors, states, and the environment [5]. The elements regarding the ToDei agent are described in the following paragraphs.

- ***Sensors:*** These are the parts of the agent in charge of detecting changes, requests, and requirements needed by the environment. Furthermore, they become the source of information and the means of updating the states stored by the agent. For example, if the environment of the agent shows that the user has accessed the system, the sensor will detect this change and send it to a state where the action-reaction rules will be applied. This will determine the student model to which the user belongs and will show through the effectors the appropriate content for this model.
- ***Effectors:*** Their function is to take the final actions to be applied to the agent environment, which were perceived by the sensors, allocated in a state, and responded to by the action-reaction rules.
- ***States:*** They are the changes that have been detected by sensors and the answers shown through the effectors of the agent according to its corresponding environment. These are stored in such a way that they allow learning from the agent, keeping for example all the actions taken according to certain perceptions and their effect on the environment. The states are part of the memory of the agent.
- ***Environment:*** It is the environment designed for the agent in which the agent acts and operates. Regarding ToDei, the environment corresponds to the interconnections of the system components.

In this proposal, ToDei has different inputs which are derived from the modules: user interface, students, knowledge basis, and tutor. They generate communication and reception of information through the agents that interact inside each module. The changes generated by such agents are received through the sensors of the ToDei agent; they are processed in the different action-reaction rules (rules of inference) of the agent. Finally, the resulting actions are sent through the effectors to the components that have generated the environment inputs of ToDei and in the same way are stored in the agent memory (states) for its self-learning.

The structure and the process carried out in ToDei are illustrated in the following chart

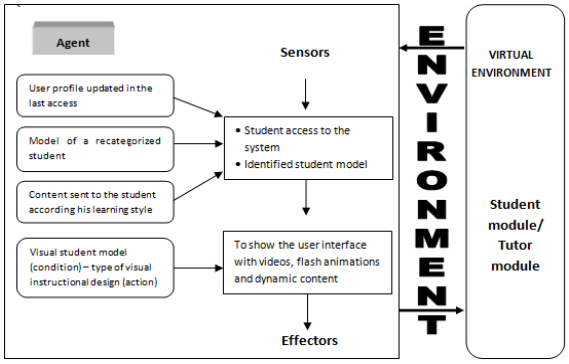
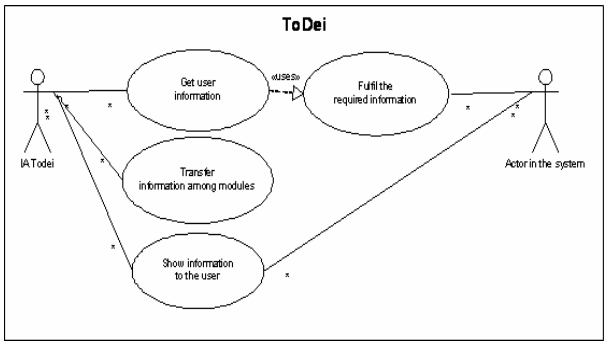



Figure 6. Example of the Structure of ToDei Intelligent Agent

The following use case diagram shows the ToDei process in the system that has been determined for this agent:

Figure 7. Use Case Diagram for ToDei Intelligent Agent

## III. CONCLUSIONS




Moodle contain basic information from tutors, students and contents; but no store the interactions path from the different actors. This aspect can be improved by including intelligent components, thus providing a personalized teaching-learning process.

The definition and implementation of intelligent agents in the components of a platform for a Moodle virtual platform do not imply the disappearance of the main actors: professor and student. On the contrary, this platform is a support that helps to reduce the teachers` workload and permits them to focus their efforts on creating good relationships with students to achieve success in learning.

Although they have the same structures, each one of the proposed agents is functionally different but depends on the module on which it is operating. The ToDei agent constitutes the main component inside this architecture since it allows visualization of the adaptive effect generated by the interaction of the components.

## IV. FUTURE PROJECTS

Currently, the user model and part of the *ToDei* agent are being developed:

1. The agent which belongs to the student module performs the following functions:
   - Creates student models based on a previous model.
   - Categorizes student profiles.
   - Monitors students while they interact with the platform.
   - Updates student information.
2. The *Todei* agent will present the information according to the categorization of the student profile

Once these components are developed, the construction of any modules that involves the definition of intelligent agents can be achieved. To carry out this, it is necessary to determine the learning type best suited to the available information and to each of the works to be developed. Also is important to consider in the development of each of the modules, the application of usability standards.

In the tutor model when instructional design is defined some mechanisms have to be found to maintain the student investigative spirit, in order to not create any dependency with resources shown by the SAE

## ACKNOWLEDGMENT

We thank Comunidad de Madrid for the support of E-Madrid Network of Excellence S2009 TIC-1650 and Ministerio de Ciencia e Innovación for the support of project TIN-2009-14317-C03-03.

## REFERENCES

[1] Brusilovsky, P. (2003). Developing adaptive educational hypermedia systems: from design models to authoring tools. Recuperado el 16 de enero de 2009 de: http://www.sis.pitt.edu/~peterb/papers/KluwerAuthBook.pdf

[2] Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. Recuperado el 25 de enero de 2009 de: http://www2.sis.pitt.edu/~peterb/papers/UMUAI96.pdf

[3] Brusilovsky, P., Millán E. (2007). User models for adaptive hypermedia and adaptive educational systems. Recuperado el 12 de enero de 2009, de: http://www2.sis.pitt.edu/~peterb/papers/1_BrusilovskyMillan.pdf

[4] Russell, S., Norvin, P. (2004). Inteligencia Artificial. Un enfoque Moderno. Segunda Edición. Pearson Education S.A. Madrid.

[5] Fritz, W. (2007). Sistemas inteligentes y sus sociedades. Recuperado el 20 de junio de 2008 de: http://intelligent-systems.com.ar/intsyst/indexSp.htm#IS

[6] Gaudioso, E. (2002). Contribución al modelado de usuario en entornos adaptativos de aprendizaje y colaboración a través de internet mediante técnicas de aprendizaje automático. Tesis Doctoral. Departamento de Inteligencia Artificial. Universidad Nacional de Educación a Distancia.

[7] Castillo, E., Gutiérrez, J., Hadi, A. Sistemas Expertos y Modelos de Redes Probabilísticas. Santander España

[8] Moodle. (2007). Documentación para profesores. Recuperado el 07 de Enero de 2009 de: http://docs.moodle.org/es/Documentaci%C3%B3n_para_Profesores

[9] Rosmalen, P., Brouns, F., Tattersall C., Vogten, H., Bruggen, J., Sloep, Peter., Koper, R. (2005). Towards an open framework for adaptive, agent-supported e-learning. Recuperado el 07 de enero de 2008, de: http://dspace.ou.nl/bitstream/1820/76/2/article-alfanet-121203.pdf

[10] Brusilovsky Peter, Peylo Christoph, Adaptive and Intelligent Web-based Educational Systems, International Journal of Artificial Intelligence in Education, v.13 n.2-4, p.159-172, April 2003

[11] Nielsen, J. (1996). Inverted Pyramids in ciberspace en http://www.useit.com/alertbox/9606.html [Fecha de Consulta: 01-12-09]