A multi-agent system model to integrate Virtual Learning Environments and Intelligent Tutoring Systems

Giuffra P., Cecilia E., Silveira Ricardo A.
Departament of Informatic and Statistics
Federal University of Santa Catarina, Brasil

***Abstract* — Virtual learning environments (VLEs) are used in distance learning and classroom teaching as teachers and students support tools in the teaching–learning process, where teachers can provide material, activities and assessments for students. However, this process is done in the same way for all the students, regardless of their differences in performance and behavior in the environment. The purpose of this work is to develop an agent-based intelligent learning environment model inspired by intelligent tutoring to provide adaptability to distributed VLEs, using Moodle as a case study and taking into account students' performance on tasks and activities proposed by the teacher, as well as monitoring his/her study material access.**

***Keywords* — Agents, Intelligent Tutoring Systems, Multi-Agent System, Virtual Learning Environments.**

I. Introduction

THE number of students with computer access has increased substantially in recent years. A qualitative change in the teaching–learning process happens when we can integrate within an innovative view all technologies, including telematics, audiovisual, textual, oral and physical [11]. The fact of students seeking information on the computer converts them into more active students. “There are activities that can be performed with the computer, forcing the student to seek information, process it and use it to solve problems, allowing the understanding of what makes and the construction of their own knowledge” [18].

Virtual learning usually offer the same learning experience, during the course, for all students, without considering their specific needs. The problem is that the students are treated as if they always had the same profile, the same goals and the same knowledge [12].

In order to provide adaptability to learning environments, according to student characteristics, and to allow a greater interactivity degree between the learning environment and the users, the research points to the use of resources provided by artificial intelligence (AI) and in particular the use of multi-agent system-based architectures [15].

In agreement with this emerges the motivation of this research: to enhance the teaching–learning process in virtual learning environments using artificial intelligence techniques to make the environments more adaptive and more interactive. This paper proposes the use of agent-based intelligent tutoring systems architectures to get personalized teaching strategies, taking into account the student profile and his/her performance, exploring their skills as best as possible, in order to have better and more effective learning in an intelligent learning environment.

This paper is structured as follows: the second section presents the theoretical reference and related works, the third section presents the definition of the model, the fourth section presents an explanation about the model implementation and the last section presents the conclusions.

II. Background

Virtual learning environments are technological tools and resources using cyberspace to lead content and enable pedagogical mediation through the interaction between the educational process actors [14]. The use of these environments has increased significantly by the strong possibility of interaction between student and teacher that they offer, and by easy access anywhere and anytime. The virtual learning environments provide tools for interaction, such as forums and chats, and enable the provision of materials by teachers about the content of the course.

For Dillenbourg [6], virtual learning environments are not only restricted to distance learning. Web-based education is often associated with distance learning; however, in practice it is also widely used to support classroom learning. The author also comments that the difference between these two types of education is disappearing. Many students in distance courses do not live far from school, but have time constraints. Often they work. In addition, there are courses that combine distance and presence, which makes for more robust learning environments.

Virtual learning environments, at first, were used primarily in distance learning; now they also serve as support in classroom courses, as a teacher's tool to provide materials, to review tasks, to keep track of the students on course (activity

DOI: 10.9781/ijimai.2013.217

logs) and also to evaluate them. For students, the environment facilitates the delivery of tasks, the obtaining of materials for the course and the monitoring of their evaluation.

Virtual learning environments can be enhanced with artificial intelligence techniques, using intelligent agents, having intelligent learning environments as result. An agent is an abstraction of something that can perceive its environment through sensors and can act upon that environment through actuators [16]. Intelligent agents are those that have at least the following characteristics: autonomy, reactivity, proactivity and social ability[21].

In practice, systems with only one agent are not common. The most common are the cases of agents that inhabit an environment containing other agents. There are two major types of multi-agent systems: reactive and cognitive. The reactive acts under a stimulus-response scheme; the cognitive has, in general, few agents because each agent is a complex and computationally heavy system[4].

A rational agent is one who chooses his/her actions according to their own interests, given the belief that he/she has about the world. The Belief, Desire, Intention (BDI) model recognizes the importance of beliefs, desires and intentions in rational actions [20].

The BDI model represents a cognitive architecture based on mental states, and has its origin in the human practical reasoning model. An architecture based on the BDI model represents its internal processes through the mental states: belief, desire and intention, and defines a control mechanism that selects in a rational way the course of actions [7].

In the context of this work, an agent is considered as an autonomous entity, able to make decisions, respond in a timely manner, pursue goals, interact with other agents, and has reasoning and character. This agent is a of type BDI, with beliefs, desires and intentions, and operates in a virtual learning environment as an intelligent tutor.

Intelligent tutoring systems (ITS) are complex systems involving several different types of expertise: subject knowledge, knowledge of the student’s knowledge, and pedagogical knowledge, among others. According to Santos et al. [17], an ITS is characterized for incorporating AI techniques into a development project and acts as a helper in the teaching–learning process.

According to Conati [5], intelligent tutoring systems are an interdisciplinary field that investigates how elaborate educational systems provide adapted instructions to the needs of students, as many teachers do.

ITS research has been investigating how to make computer-based tutors more flexible, autonomous and adaptive to the needs of each student by giving them explicit knowledge of the relevant components of the teaching process and reasoning skills to convert this knowledge into intelligent behavior.

To Giraffa and Viccari [9], ITS developments consider a cooperative approach between student and system. The goal of ITS is to complement or replace a human tutor, with the advantage of monitoring the student in each learning step [13].

Research in intelligent tutoring systems is concerned about the construction of environments that enable more efficient learning. [8].

Intelligent tutoring systems offer flexibility in the presentation of material and have the major ability to respond to students’ needs. They seek, in addition to teaching, learning relevant information about the student, providing an individualized learning. Intelligent tutoring systems have been shown to be highly effective in improving performance and motivation of students [10].

Intelligent tutoring systems in virtual learning environments potentiate the teaching–learning process, making the virtual environment into an intelligent learning environment. Intelligent learning environments use AI techniques to respond to students’ needs, making that learning personalized [10].

According [15], the intelligent learning environment must build and update the student model in terms of what he/she already knows, which can vary significantly from one student to another.

Related Works

In order to know the current status of recent research about virtual learning environments and the use of intelligent agents as tutors in these environments, we performed a systematic literature review. Among them, there were three items which were most closely related to the purpose of this study.

“Approach to an Adaptive and Intelligent Learning Environment” [1], which proposes an agent-oriented approach for the design and implementation of an adaptive and smart component for a virtual learning environment. The adaptivity in the model is defined as a system’s ability to create and, during the learning process, uniformly upgrade the curriculum that satisfies the student’s needs. The proposed model has three parts that describe the main features of intelligent and adaptive component. First, the student chooses the courses based on his/her needs, the level of excellence that he/she wants to achieve, and his/her preference concerning the type of study material. Second, the system will decide how to act – for example, show the material to the student, based on the belief (student model) that the system has about it. In the last part it is decided when to propose an evaluation test for the student or any other activity that can evaluate any specific knowledge in relation to the curriculum. After completing the evaluation, the belief about the student is updated.

“Cluster Analysis in Personalized E-Learning” [22]: this is a proposed system architecture in which teaching techniques and appropriate layouts are set to groups of students with similar preferences, created by applying clustering techniques. Teaching materials and content can be adapted to the needs of each group and different learning paths can be created. New students fill out a questionnaire to determine their learning style and their choices of usability and, according to this, the appropriate group is chosen for them. The idea of the proposed solution is to divide the process into two steps: first, to look for groups of students with a great similarity and detect those isolates. In the second step the groups are mixed in larger groups if necessary and the isolates are indicated. The objective of the experiment was to examine the performance of the proposed clustering technique for different student's data sets, depending on the choice of parameters.

"Supporting Cognitive Competence Development in Virtual Classrooms" [19]. The approach described in this article implements a mechanism to adaptively create self-assessment questionnaires in a Moodle environment. The Learning Management System (LMS) is capable of saving all online activities of the students in log files. This information can be used also to automatically generate intelligent feedback to the student. The questions are derived from an ontology of skills that is also used for indexing learning materials. The student traces through the learning materials used to determine the current state of "expected knowledge" or skills. The system includes two main agents: the goals manager agent, which guides the student in planning activities; and the content manager agent, which guides the student during the resources review. In this paper, an extension of the Moodle LMS – in which ontologies are used to structure the learning process by providing resources and generating questionnaires automatically for self-assessment of the students – is presented.

In the conducted research to analyze the state of the art, papers were found dealing with adaptability in virtual learning environments, taking into account the students' needs, learning styles, usability preferences and their activities report (log).

In the first related paper, adaptability is based on the preferences of students regarding the study material, where the agent provides this material according to the information that he/she has about the student's preferences. In the second paper, the proposal is to provide different layouts to the students, taking into consideration similar preferences with regard to learning styles and usability choices. In the last related paper, self-assessment questionnaires are used, adaptively created in the virtual learning environment Moodle, and ontologies are used to index the learning materials. This paper proposes to join data that can be obtained from the database (information of student performance and logs) that most virtual learning environments widely used usually have, in order to set the discipline in a personalized way for each student, with regards to the available material for the student as the activities proposed to him/her, exploring his/her skills and bypassing disabilities, always having a baseline with material and compulsory activities, and activities outside of this line divided into different levels of difficulty.

III. Definition of the Model

The aim of this work is to create an agent architecture and a knowledge base of these agents that compose an intelligent tutoring system with information obtained from the database of a teaching–learning virtual environment. For this, a case study is done based on the Moodle platform architecture, chosen because it is a platform widely used today, in addition to being consolidated from the standpoint of operation, and also to be formally used in the institution where the research is performed.

The classical model of ITS contains the pedagogical model, the student model, the domain base and the control. In the proposed model, two types of agents, called "Bedel" and "tutor" are used. The Bedel Agent and all their knowledge and interaction structure corresponds in the classical model of intelligent tutoring to the Pedagogical Model. The Tutor Agent and all their structure corresponds in the model of intelligent tutors to the Student Model. The content of the discipline, in turn, may be associated with the abstraction of the Domain Base (Fig. 1).

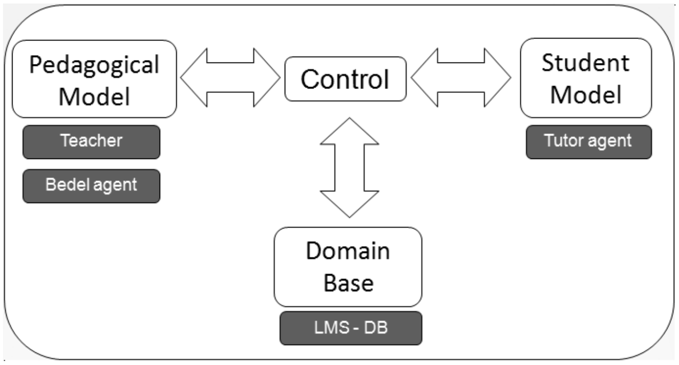


Fig. 1 Classical model with proposed model

The model of agents "Bedel" (agent of the discipline) and "Tutor" (agent of the student) is defined; they are connected with the learning virtual environment through the database. The interface in which the teacher sets the priority and levels of the resources and tasks in the environment is developed. The database is adapted with the creation of the table of grade profiles of the students and the table of dependencies of resources and activities, configured by the teacher by means of a Moodle block into his discipline, and is made the integration of the actions of agents with the virtual learning environment Moodle. The agents model (Fig. 2) shows the agents "Tutor" and "Bedel" their actions, messages and perceptions, as well as their connection to the database. The actor "teacher" is the figure of the discipline teacher who inserts the resources and activities into the learning environment, sets the type of profile and sequence, and this information is stored in the database, so the agent "Bedel" knows how to show them to students.

The database has information concerning the student, such as personal data, performance data and data from student interaction in the system. Every student interaction in the environment is saved in the base in the form of a log. Similarly, the student's performance in each of the activities and tasks is stored in the database and updated constantly every access and interaction of the student, providing rich material for the agent's performance. The agents share the information from the database. The "Tutor" updates the student profile and, if necessary, shows a message to him/her about his/her performance. The "Bedel" obtains from the database the configuration of resources and activities in the discipline made by the teacher; it sets their preview and verifies if tasks were evaluated to send a message to the "Tutor" who, upon receiving the message, updates the data of the student profile.

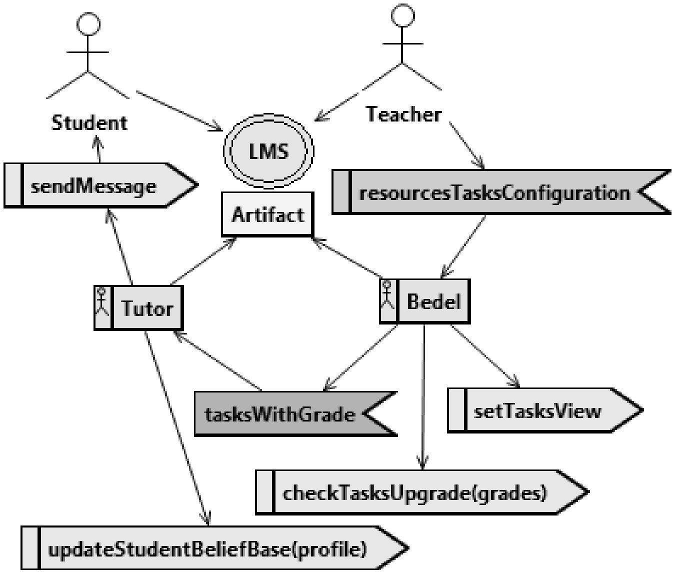
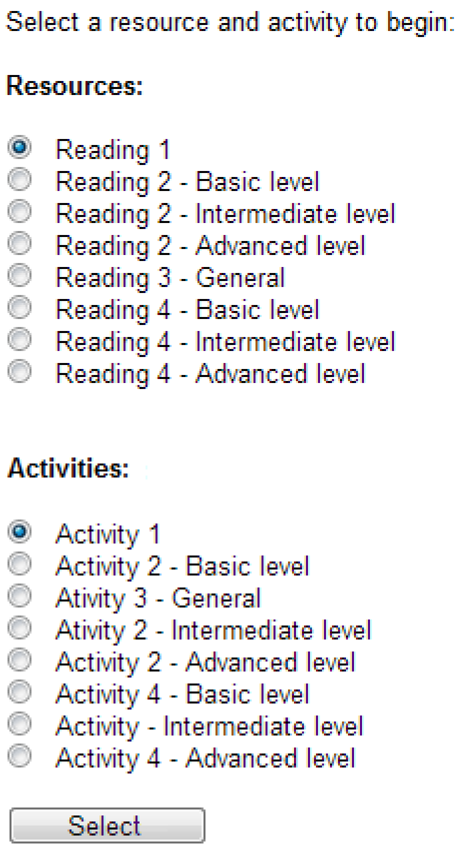

Student
Teacher
LMS
Artifact
sendMessage
resourcesTasksConfiguration
Tutor
Bedel
tasksWithGrade
setTasksView
checkTasksUpgrade(grades)
updateStudentBeliefBase(profile)

Fig. 2. Agent System Model

The model works as follows:

1) The teacher inserts the resources and creates tasks in the Moodle environment as usual. After that, he/she adds the tutor block in his/her discipline and configures the Bedel setting the first resource and activity (Fig. 3), the dependencies (Fig. 4) of activities and resources, and their level (basic, intermediate and advanced), also by means of the Moodle environment. In addition, the teacher has the option to choose resources that must be shown to all students in a general way. The first reading (resource) and the first activity are shown for all students; therefore the teacher needs to indicate which they are. This information is stored in the database, where some tables which are necessary for the model are added. With this information, the "Bedel" knows about resources and activities of the discipline and knows how the course should be developed for each type of student.
2) In the environment's database the grade_profile table (Fig. 5) is also added, which contains a numeric value, calculated with the grades of the activities performed by students and the access made by them in different files provided by the teacher. This table is updated each time a teacher updates the worksheet with the grades of some of the tasks that he/she provides to students.
3) The grade_profile average of all students is computed and students are separated by profile into groups – basic, intermediate or advanced – according to their grade_profile. Whoever has average grades is in the average profile, whoever has grades below average is in the basic profile, and those who have grades above average are in the advanced profile.
4) Tasks are provided independently for each student, according to their performance in the previous tasks, and their access to previous reading material. The availability of tasks and resources is made by the "Bedel" using the conditional access resource of Moodle.
5) Each time the Bedel calculates the grade_profile it updates the belief base of the current profile of the student, which can go from basic to intermediate or advanced and vice-versa during the time that the course is offered.


Select a resource and activity to begin:

Resources:

● Reading 1
○ Reading 2 - Basic level
○ Reading 2 - Intermediate level
○ Reading 2 - Advanced level
○ Reading 3 - General
○ Reading 4 - Basic level
○ Reading 4 - Intermediate level
○ Reading 4 - Advanced level

Activities:

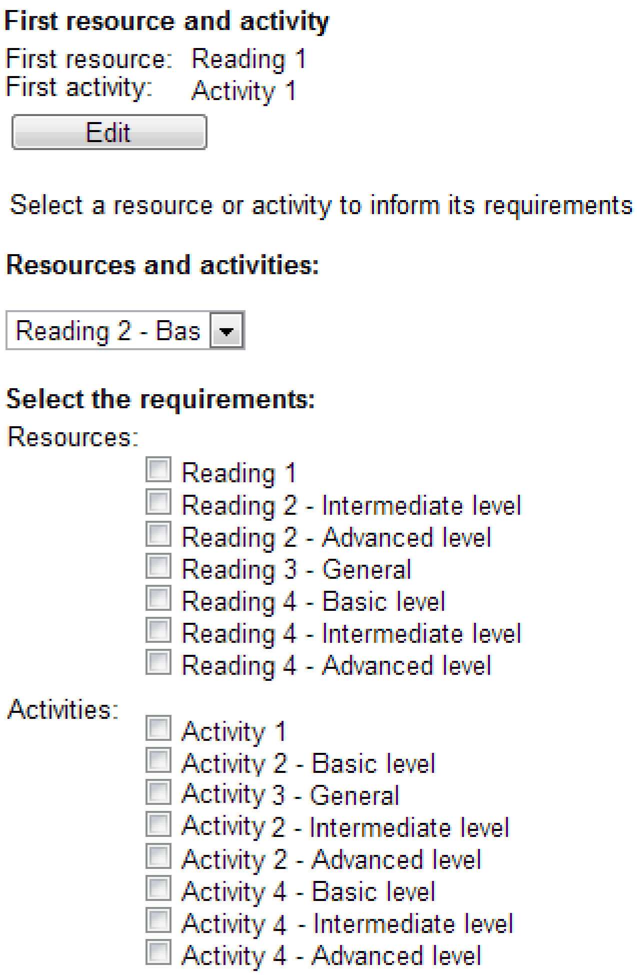
● Activity 1
○ Activity 2 - Basic level
○ Ativity 3 - General
○ Ativity 2 - Intermediate level
○ Activity 2 - Advanced level
○ Activity 4 - Basic level
○ Activity - Intermediate level
○ Activity 4 - Advanced level

Fig. 3 First resource and activity

The calculation of grade_profile is done as follows: The student grade of the last activity assessed by the teacher and the grade of the student access on reading that is a pre-requisite for activity are summed. If the student accesses the reading, two points in the activity grade are added. If he/she does not access, four points are added. This difference is given for increasing the possibility to the student who does not access the reading, to have a higher grade_profile and then go to a higher level task than the profile that he/she would belong to if he/she had a lower grade, stimulating him/her to read before accomplishing future activities.

After this, the average value of the grade_profile field is computed for all students. The lowest value considered for the profile average is 6; if the profile average is lower, it automatically becomes 6. For the student to be at an average profile, his/her grade_profile must be between 5.5 and 6.5; if the student has a grade_profile less than 5.5 he/she is on a basic profile and if he/she has grade_profile greater than 6.5

he/she is on an advanced profile.



First resource and activity

First resource: Reading 1
First activity: Activity 1

Select a resource or activity to inform its requirements

Resources and activities:

Reading 2 - Bas

Select the requirements:

Resources:

☐ Reading 1
☐ Reading 2 - Intermediate level
☐ Reading 2 - Advanced level
☐ Reading 3 - General
☐ Reading 4 - Basic level
☐ Reading 4 - Intermediate level
☐ Reading 4 - Advanced level

Activities:

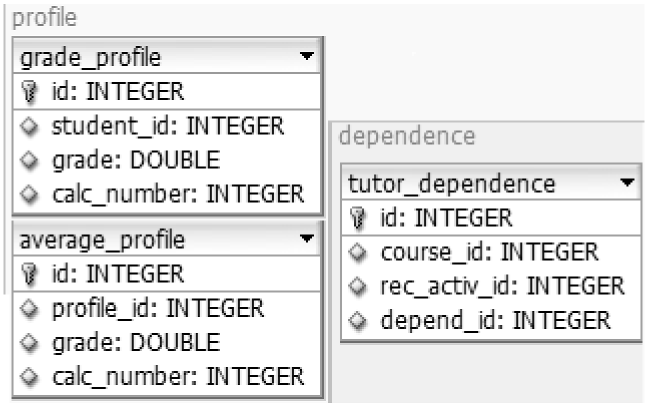
☐ Activity 1
☐ Activity 2 - Basic level
☐ Activity 3 - General
☐ Activity 2 - Intermediate level
☐ Activity 2 - Advanced level
☐ Activity 4 - Basic level
☐ Activity 4 - Intermediate level
☐ Activity 4 - Advanced level

Fig. 4 Dependencies configuration

The maximum value considered for the profile average is 8. If the average profile is greater than that, it automatically becomes 8. For the student to be at average profile, his/her grade_profile must be between 7.5 and 8.5, if the student has a grade_profile less than 7.5 he/she is on the basic profile and if he/she has a grade_profile greater than 8.5 he/she is on the advanced profile.

The student belongs to the average profile if his/her grade_profile is 0.5 less or more than the average grade_profile in his/her class; for example, if the grade_profile average is 7.5, he/she will be in the intermediate profile if he/she has a grade_profile between 7 and 8. The student who has a higher grade with more than 0.5 of diference with the average will be in the advanced profile and the student who has a lower grade with more than 0.5 of diference will be in the basic profile.

The student will have access to the material of their profile (basic, intermediate, advanced), according the configuration of resources and tasks made by the teacher.



profile

grade_profile
id: INTEGER
student_id: INTEGER
grade: DOUBLE
calc_number: INTEGER

average_profile
id: INTEGER
profile_id: INTEGER
grade: DOUBLE
calc_number: INTEGER

dependence

tutor_dependence
id: INTEGER
course_id: INTEGER
rec_activ_id: INTEGER
depend_id: INTEGER

Fig. 5. Profile and dependence tables

IV. Implementation of the Model

The model integrates concepts of intelligent tutoring systems architectures with VLE's that have their use consolidated as Moodle, which are not adaptive for itself only, and can be potentiated with artificial intelligence techniques, resulting in intelligent learning environments which are shown to be adaptive and more suitable to the implementation of teaching defiant methodologies for the student.

The use of agents in the implementation of this model is important because of the agent's ability to adapt to environment changes, showing resources and activities to students in a personalized way, according to their performance in the discipline, and taking into account the teacher's initial settings.

For the agent implementation the Jason tool was used, which is an interpreter for an extended version of AgentSpeak, oriented agent programming language, implemented in Java. The basic idea of AgentSpeak is to define the know-how (knowledge about how to do things) of a program in the form of plans [3].

One of the most interesting aspects of AgentSpeak is that it is inspired by and based on a model of human behavior that was developed by philosophers. This model is called the Belief Desire Intention (BDI) model. The language interpreted by Jason is an extension of AgentSpeak, based on BDI architecture. A component of the agent architecture is a beliefs base and an example of what the interpreter does constantly, without being specifically programmed, is to perceive the environment and update the beliefs base accordingly with this [3].

The teaching–learning virtual environments are designed to enable the knowledge-building process. Different to conventional software, which seeks to facilitate the tasks achievement by user, learning environments incorporate the complexity to more flexible different forms of users (students), relations, to learn and to practice content, and to collaborate. These environments are used by students of various cognitive profiles [2].

The version of virtual learning environment Moodle used for this work is 2.2, where the task condition resource is available, which allows the provision of content and activities with a restriction. This feature must be activated by the administrator of Moodle in the environment advanced settings, enabling the option “Enable tracking of completion” and “Enable conditional access.” Moreover, in course settings, in “student progress” topic, the teacher must enable the completion tracking option.

With this feature enabled, tasks can be made available only to students who perform the pre-requisites set, which can be: a grade on a specific activity; the viewing of a resource; or his/her grade.

In this work, the availability of resources and activities is done taking into account the student’s performance and his/her access in the system and is made available to the student depending on his/her grade_profile, computed according to his/her performance and participation in the discipline. The information between the agent system and the virtual learning environment are exchanged through the database of the learning environment which contains information about the pre-requisites and profiles of tasks and resources, defined by the teacher at a time to configure the Bedel. The development was made in 4 steps. On Fig. 6 we can see the Bedel sequence diagram, with the Artifact working like an interface between the database and the Bedel Agent.

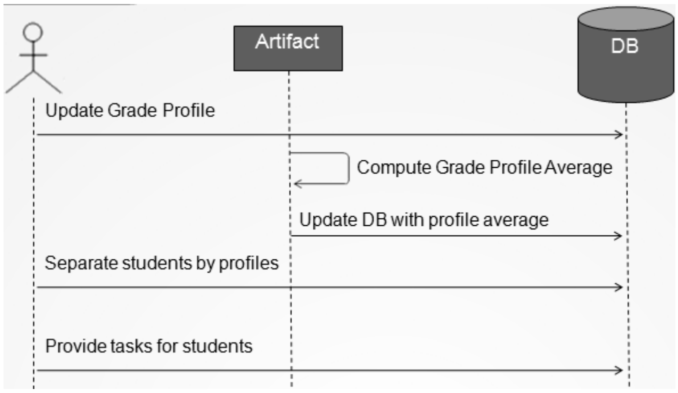

Artifact
DB
Update Grade Profile
Compute Grade Profile Average
Update DB with profile average
Separate students by profiles
Provide tasks for students

Fig. 6 Bedel Sequence Diagram

A. Development of the tutorblock in Moodle LMS.

It was developed a moodle block for the teacher to configure the agent. This block was created following the Moodle standard programming for creating blocks. The name was defined as “Tutor Block”. This block have to be added by the teacher and is used by him/her to set the Bedel Agent, after the insertion of resources and activities in the environment.

After the teacher configures the agent through the block, he/she can see the dependency graph (Fig. 7) generated after setting all prerequisites. This graph shows the relations between the resources and activities setting by the teacher.

B. Development of the Bedel Agent. Programming student profile

In this step was implemented the code for the Bedel Agent to calculate de student profile, using the information available on database. The Bedel Agent access the database trough the Artifact and execute the plans it has for calculating this profile. Then, the Bedel Agent calculate the profile average of all the students and separate them into profiles (basic, intermediate and advanced).

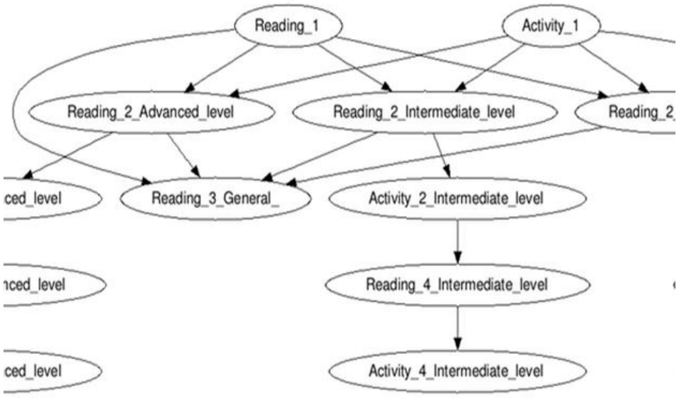
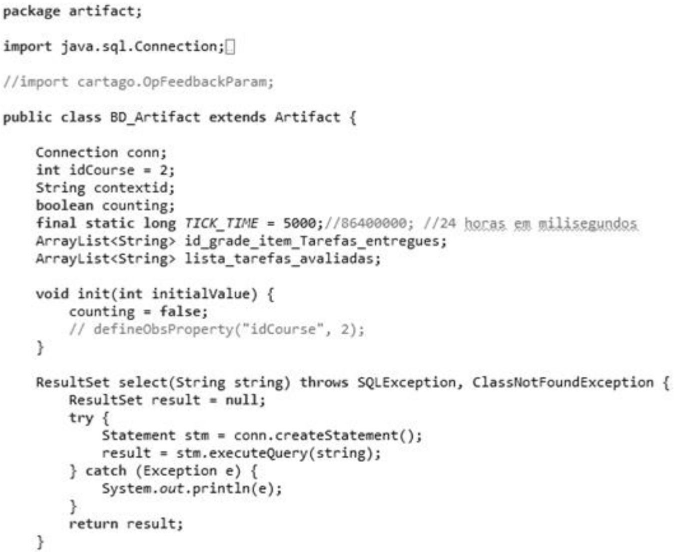

Reading_1
Activity_1
Reading_2_Advanced_level
Reading_2_Intermediate_level
Reading_2_
ced_level
Reading_3_General_
Activity_2_Intermediate_level
nced_level
Reading_4_Intermediate_level
ced_level
Activity_4_Intermediate_level

Fig. 7 Dependency graph

The Bedel Agent implementation has a definition file, where is specified the infrastructure, the Cartago environment used to program the artifacts (Fig. 8) and the Tutor and Bedel Agents.

C. Development of the code to display resources and activities

For the availability of resources and activities according to the student profile was created the tutor_profile_availability table in the database, to store the information of the minimum and maximum grades of the intermediate profile. Verifying this information the resources and activities can be provided, according to the students profile.



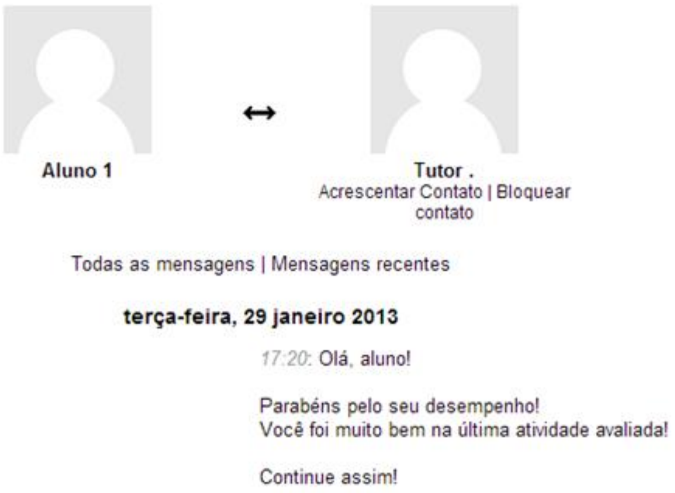
```
package artifact;

import java.sql.Connection;□

//import cartago.OpFeedbackParam;

public class BD_Artifact extends Artifact {

    Connection conn;
    int idCourse = 2;
    String contextid;
    boolean counting;
    final static long TICK_TIME = 5000;//86400000; //24 horas em milisegundos
    ArrayList<String> id_grade_item_Tarefas_entregues;
    ArrayList<String> lista_tarefas_avaliadas;

    void init(int initialValue) {
        counting = false;
        // defineObsProperty("idCourse", 2);
    }

    ResultSet select(String string) throws SQLException, ClassNotFoundException {
        ResultSet result = null;
        try {
            Statement stm = conn.createStatement();
            result = stm.executeQuery(string);
        } catch (Exception e) {
            System.out.println(e);
        }
        return result;
    }
```

Fig. 8 Artifact Source Code

D. Development of the Tutor Agent to send messages to the students

It was implemented the Tutor Agent, that is responsible for sending messages of encouragement to the student, taking into account his/her performance. This agent receives the student profile information from the Bedel Agent, updates the belief it has about this student and sends some message to him/her according to the student situation. If the student had a better performance than in the previous activity the Tutor Agent send a congratulation message (Fig. 9), if not, it sends a message to encourage him to be better the next time.

At this stage, also, the Bedel Agent is updated, inserting methods that enable communication between it and the Tutor Agent. Sending the student information to the Tutor Agent.


Aluno 1
↔
Tutor .
Acrescentar Contato | Bloquear contato
Todas as mensagens | Mensagens recentes

terça-feira, 29 janeiro 2013

17:20: Olá, aluno!

Parabéns pelo seu desempenho!
Você foi muito bem na última atividade avaliada!

Continue assim!

Fig. 9 Congratulation message (in Portuguese)

V. Conclusions

In this study is proposed a solution for virtual learning environments to assist teachers to provide activities and resources in a personalized way depending on the student's performance and his/her behavior in the discipline.

Students are assessed by their interaction in the discipline and the grades obtained in tasks, creating different profiles for groups of students with the same behavior. More advanced tasks are available for students who have improved performance, enabling more efficient learning, exploring students' skills, and maintaining a basic level for learning the discipline content.

Works related to virtual learning environments and adaptivity in general differentiate students by learning style – for example, a student who learns better with pictures than with reading lots of text. In this work students are distinguished by their performance, taking into account the grades obtained, and their participation (access) in the various resources available in the discipline, creating an adaptive environment that constantly updates the profile of students, and therefore, a student with a basic profile, at the end of the course may have an average profile. These profile changes can be studied and displayed to the teacher, in an extension of this model.

References

[1] Baziukaité, D.: Approach to an Adaptive and Intelligent Learning Environment. In: Elleithy, K., Sobh, T., Mahmood, A., Iskander, M., Karim, M. Advances in Computer, Information, And Systems Sciences, And Engineering, pp. 399–406. Springer Netherlands (2006)

[2] Boff, E.: Collaboration in Learning Intelligent Environments mediated by a Social Agent Probabilistic. In: Thesis (Ph.D.) Computer Science Course. Federal University of Rio Grande do Sul, Porto Alegre. (2008) (in Portuguese)

[3] Bordini, R.H., Hubner, J.F., Wooldridge, M.: Programming Multi-Agent Systems in AgentSpeak using Jason. Editora Wiley. England (2007)

[4] Bordini, R., Vieira, R., Moreira, A.F.: Fundamentos de Sistemas Multiagentes. In: Ferreira, C. E. (ed.). Jornada de Atualização em Informática (JAI'01), vol. 2, pp. 3–44. SBC, Fortaleza, Brasil (2001)

[5] Conati, C.: Intelligent Tutoring Systems: New Challenges and Directions. Paper Presented at the Proceedings of the 21st International Joint Conference on Artificial intelligence (2009)

[6] Dillenbourg, P.: Virtual Learning Environment . EUN Conference 2000. Workshop on Virtual Learning Environment (2000). Available in: http://tecfa.unige.ch/tecfa/publicat/dil-papers-2/Dil.7.5.18.pdf

[7] Fagundes, M.: An environment for development of BDI agents. Course Completion Work. Federal University of Pelotas (2004) Available in: http://www.inf.ufsc.br/~silveira/INE602200/Artigos/TCC_Moser.pdf. (in Portuguese)

[8] Frigo, L. B., Pozzebon, E., Bittencourt, G.. The Role of Intelligent Agents in Intelligent Tutoring Systems. In: Proceedings of the WCETE - World Congress on Engineering and Technology Education, São Paulo, Brasil, pp. 667-671. (2004). (in Portuguese)

[9] Giraffa, L.M.M., Viccari, R.M.: The Use of Agents Techniques on Intelligent Tutoring Systems. In: Computer Science SCCC '98. XVIII International Conference of the Chilean Society, pp. 76-83. IEEE Computer Society Washington, DC, USA (1998).

[10] Lima R.D., Rosatelli, M.C.: An intelligent tutoring system to a virtual environment for teaching and learning. In: IX Workshop de Informática na Escola, Campinas. Anais do XXIII Congresso da Sociedade Brasileira de Computação. (2003). (in Portuguese)

[11] Moran, J.M.: Computers in Education: Theory & Practice. Porto Alegre, vol. 3, n.1. UFRGS. Postgraduate Program in Computer oin Education, pp. 137–144 (2000). (in Portuguese)

[12] Mozzaquatro, P.M., Franciscato, F., Ribeiro, P.S., Medina, R.D.: Modeling a Framework for adaptation of Virtual Learning Environments Mobiles to different cognitive styles. vol 7, n 3, pp. 253-263. Renote - New Technologies in Education Journal. ISSN 1679-1916 (2009). (in Portuguese)

[13] Oliveira, C.L.V.: AutoExplC: intelligent tutoring system to aid the teaching of "C" languagebased on learning by self-explanation of examples. PUC, Campinas (2005). (in Portuguese)

[14] Pereira, A.T.C., Schmitt, V., Álvares, M.R.C.: Virtual Learing Environments. Culture Bookstore. (2007). Available in: http://www.livrariacultura.com.br/imagem/capitulo/2259532.pdf. (in Portuguese)

[15] Silveira, R. A. Ambientes inteligentes distribuídos de aprendizagem. Porto Alegre: CPGCC da UFRGS, (1998)

[16] Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach. Prentice-Hall, Inc. New Jersey (2002)

[17] Santos, C.T., Frozza, R., Dahmer, A., Gaspary, L.P.: Doris - Pedagogical Agent in Intelligent Tutoring Systems. In: Cerri, S., Gouardères, G., Paraguaçu, F. Intelligent Tutoring Systems, pp. 91-104. Springer Berlin, Heidelberg (2002)

[18] Valente, J.A.: Computers in Education: conform or transform the school, pp. 42. (2009).

[19] Weinbrenner, S., Ulrich, H., Leal L., Montenegro, M., Vargas, W., Maldonado, L.: Supporting Cognitive Competence Development in Virtual Classroom In: 10th International Conference on Advanced Learning Technologies. IEEE Computer Society (2010)

[20] Wooldridge, M.: Reasoning about Rational Agents. The MIT Press. Cambridge, Massachussetts. London, England. (2000)

[21] Wooldridge, M.: An Introduction to Multiagent Systems, 2nd edition. John Wiley & Sons Ltd in Hoboken, NJ (2009)

[22] Zakrzewska, D.: Cluster Analysis in Personalized E-Learning Systems. Springer-Verlag, Berlin Heidelberg (2009)

Ms. Cecilia E. Giuffra P. Doctoral student of the postgraduate course in Engineering and Knowledge Management from the Federal University of Santa Catarina (UFSC). Master's (2013) and Bachelor's (2009) at Computer Science from the same University. Has experience in Computer Science, focusing on the following subjects: Artificial Intelligence, Learning Management Systems, Multiagent Systems, Computers in Education, Repositories, and Learning Objects. Has knowledge of the Moodle platform and had participation in teacher training and EaD staff.

Dr. Ricardo A. Silveira, PhD in Computer Science from Universidade Federal do Rio Grande do Sul (2001). Master's at Education from the Pontificial Catholic University of Rio Grande do Sul – PUC-RS (1992). Graduate at Electronic Engineering from PUC-RS from Departamento de Engenharia Elétrica (1980). Associate Professor of the Department of Informatic and Statistics (INE) on Federal University of Santa Catarina (UFSC). Has experience in Computer Science, focusing on Formal Languages and Automata, acting on the following subjects: Artificia Intelligence, Multiagent Systems, Computer in Education, Pedagogical Agents and Learning Objects.