FIPA COMPLIANT PEDAGOGICAL AGENTS IN DISTRIBUTED INTELLIGENT LEARNING ENVIRONMENTS

Ricardo Azambuja Silveira
Instituto de Física e Matemática - DMEC
Universidade Federal de Pelotas
Pelotas, RS, Brasil
rsilv@ufpel.tche.br

Eduardo Rodrigues Gomes
Instituto de Física e Matemática - DMEC
Universidade Federal de Pelotas
Pelotas, RS, Brasil
ergomes@ufpel.tche.br

ABSTRACT

Over the last years, many organizations started to use Distance Teaching tools as instruments in employees qualification programs, creating what we may call E-learning or Virtual Training in Human Resources Development Programs. However, usually these organizations tend to use technological resources already available, and do not shape their technological platform into a pedagogical project. Recent advances in the field of Intelligent Teaching Systems have proposed the use of Artificial Intelligence through architectures based on agents' societies. Teaching systems based on Multi-Agent architectures make possible to support the development of more interactive and adaptable systems. The objective of the paper is to discuss the feasibility of implementing Distributed Intelligent Learning Environment – DILE based on the Multi-Agents Architecture approach, aiming at the achievement of human resources qualification through Virtual Training. Besides, we present a proposal of an architecture based on the approach of FIPA compliant Multi-Agents systems named Java Agent Framework for Distance Learning Environments..

KEYWORDS

Intelligent Learning Environments, Artificial Intelligence, Multi-agents systems.

1. INTRODUCTION

Computer Science, together with Psychology and Education, has been trying to refine teaching computational tools towards personalized self-learning. Everyday, new approaches to the use of Computer and Education are bringing new perspectives to this area. The evolution of Computer and Education became computational teaching environments an excellent choice for Distance Learning, by bringing new vigor to this field of science. Computer Networks and Multimedia fields have provided tools for the development of Tutoring Systems based on client-server architectures. The popularity of Internet along with the extensive development and use of standard protocols and services make Internet very attractive for Distance Learning. There has been a big boom of tools and mechanisms available for implementation and support of Distance Learning. The traditional Computer Assisted Instruction Systems approach (CAI) lacks to provide an adaptable learning process according to each individual student.

The state of the art in Intelligent Tutoring Systems and Intelligent Learning Environments fields points to the use of Agent Society-Based Architectures. The fundamentals of the Multi-Agent systems have

demonstrated to be very appropriate to design tutoring systems, since the teaching-learning problem could be handled in a cooperative approach (Cheikes 1995; Giraffa et al. 1998; Johnson & Shaw 1997; Mathoff & Hoe 1994; Norman & Jennings 2000). Using Multi-Agents Systems approach to design Intelligent Tutoring Systems can result in more versatile, faster and at lower costs systems. The introduction of AI techniques and, specifically, the use of Multi-Agents architecture in these environments aim to provide student-modeling mechanisms (Johnson & Shaw 1997). We believe that these concepts can be used in modeling and implementation of Intelligent Distance Learning platforms aimed at qualification programs in organizations.

The objective of the paper is to discuss the feasibility of implementing Distributed Intelligent Learning Environment – DILE based on the Multi-Agents Architecture approach, aiming at the achievement of human resources qualification through Virtual Training. Besides, we present a proposal of an new architecture for Java Agent Framework for Distance Learning Environments (Silveira & Viccari 1997; Silveira 1997; Silveira 2000). This architecture represents an evolution over the old one toward a more robust communication framework among the agents by using a well known Agent Communication Language (ACL): FIPA-ACL over a agents communication platform FIPA-OS.

2. THE PROJECT

This project, based on Java Agent framework for Distance learning Environments proposes an infrastructure of project, development and implementation of Distributed Intelligent Learning Environments – DILE, based on the approach of FIPA like Multi-Agents architecture. The Java Agent Framework for Distance Learning Environments was born in 1997 as a thesis project. In this project we use the Eletrotutor prototype (http://www.inf.ufrgs.br/~rsilv/eletro31/eletro.html) as a test bed. Eletrotutor is a teaching environment for Electrodynamics teaching, and in each version we have been refining the agents architecture.

2.1 Architecture

This architecture encompasses a very short Multi-Agent family composed of just three types of agents (Fig 1): an agent responsible for the student interface , one responsible for student model, and a set of agents responsible for tasks related to teaching tactics (Pedagogical Agents), where each agent may have its tasks specified according to its goal.

This agents are implemented according to FIPA recommendation . In order to improve the FIPA compliance the FIPA-OS platform (http://fipa-os.sourgeforge.net) was used The FIPA-OS framework provides several JAVA classes for FIPA compliant agents construction, a agent runtime environment and a set of FIPA message interchange services. All the FIPA-ACL agent's message interchanging is controlled by FIPA-OS.

The system contains a special agent responsible for each teaching strategy (Pedagogical Agents), that is, for the domain knowledge retrieval over each point to be presented to the student, for the task of proposing exercises and evaluating proposals, examples and extra activities.

The Student's Model agent takes all actions of student's data accessing. When a Pedagogical agent is required to update the student's history, this agent sends to the Student Model agent the data to be updated, as well as any other change in the student's cognitive state.

The Interface Agent performs the communication between the student and the system. He recognize the student actions over the graphical interface and generates the corresponding messages to other agents and produces content displaying according to the received messages from them.

Agents knowledge is implemented as a relational data bank. The Student Model agent performs all actions related to knowledge base retrieval and updating. When a pedagogical agent needs to update the student's historic, for example, it will send data to the Student Model Agent. The major roles of the Student Model Agent are: To load the current student's state; to generate the overall student's historic; to generate the report of every student's steps; to generate the assessments results; to select the teaching strategies; to check the last access date and to verify tactics available for a certain lesson.

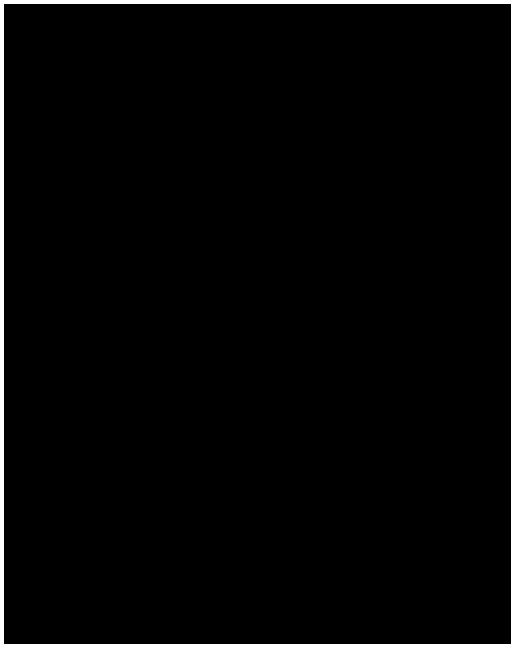

Figure 1 Architecture System.

The Pedagogical Agents are generated from a tactics previously defined by the course specialist. Their tasks are defined according to the agent's needs. However, as the tutor is based on the content presentation (HTML pages presentation), some tasks are previously defined for all pedagogical agents:

- **Show current content:** when the pedagogical agent receives this request, it communicates with the Student Model manager agent to retrieve from the knowledge base the content that is being presented to the student and sends it to the student's browser.
- **Advance:** with that request, the pedagogical agent communicates with the Student Model Manager to retrieve from the knowledge base which content will be presented to the student.
- **Return:** the pedagogical agent retrieves, from the knowledge base and through the Student Model Manager, which is the content previous to the one the student sees at that moment.
- **Options:** if requested, the pedagogical agent can propose some tools or resources to the student, according to the teaching tactics she/he is performing
- **Update historic:** at every task implemented, the pedagogical agent must register at the Student Model the actions that were performed, as for example, date and hour the student left the current content, date and time of a new content input, etc.
- **Evaluation:** the agent has evaluation mechanisms for the tactics the agent implements.

3. CONCLUSION

In this work we intend to bring some important contributions, refining the efficacy of learning environments, aggregating concepts of different areas to establish a methodology for the implementation of Distance Education projects, and stressing the use of cooperative problem solving paradigm using Multi-agent architecture.

The use of some FIPA-compliant communication framework (FIPA-OS) improved the message interchanging among the agents and provide more adaptability and flexibility to the system.

Further work will integrate the implementation of this pedagogical agents with commercial or well-known academic learning environments or frameworks. This integration takes advantage of the pedagogical and administrative resources of these environments and improves their adaptability using cognitive modeling and solving problem strategies of the framework.

ACKNOWLEDGEMENT

This paper is granted by Brazilian research agencies: CNPq and FAPERGS

REFERENCES

Cheikes, B. A. , 1995 Gia: agent based architecture for intelligent tutoring systems. *Proceeding of the cikm workshop on intelligent information agents*.

ELETROTUTOR III. 2003. Retrieved: January 15, 2003. from http://www.inf.ufrgs.br/~rsilv/eletro31/eletro.html

FIPA: The foundation for Intelligent Physical Agents. 2003. Retrieved: January 15, 2003. from http://www.fipa.org

FIPA-OS: Fipa Open Source. 2003. Retrieved: January 15, 2003. from http://fipa-os.sourgeforge.net

Giraffa L. M., Viccari R. M.; Self, J. 1998. Multi-Agent based pedagogical games. *Proceedings of ITS*, 4.,1998.

Johnson, W. Lewis; Shaw, Erin, 1997. Using agents to overcome deficiencies in web-based courseware. *Proceedings of world conference on artificial intelligence in education*, ai-ed, 8.

Mathoff, J.; Van Hoe, R. 1994. Apeall: A multi-agent approach to interactive learning environments. *Proceedings of European Workshop On Modeling Autonomous Agents Maamaw*, 6., Berlin.

Norman, Timothy J.; Jennings, Nicholas R. 2000. Constructing a virtual training laboratory using intelligent agents. *In International Journal of Continuos Engineering and Life-long Learning.*

Silveira, R. A.; Viccari, R. M. 1997. Projeto Eletrotutor: Desenvolvimento e Avaliação de Ambientes Inteligentes de Ensino-Aprendizagem. *Proceedings Conferencia Latino-Americana De Informatica*, 23. Valparaíso.

Silveira, Ricardo. 1997. Desenvolvimento e avaliação de duas abordagens de ambientes de ensino inteligentes. *Proceedings of Simpósio Brasileiro De Informática Na Educação*, 8. São José dos Campos. Brazil.

Silveira, Ricardo Azambuja. 2000. Modelagem Orientada a Agentes Aplicada a Ambientes Inteligentes Distribuídos de Ensino: JADE - Java Agent framework for Distance learning Environments Porto Alegre: PPGC da UFRGS, Doctoral Thesis (Portuguese).