A multiagent system for formative assessment support in Learning Management Systems

Joice Lee Otsuka, Vítor Sexto Bernardes, Heloísa Vieira da Rocha
Institute of Computing – State University of Campinas
Nucleus of Informatics Applied to Education – State University of Campinas
P.O. Box 6176   13083-970 - Campinas, SP – Brazil
Phone +55 19 3788-5842
{joice,vitor.bernardes,heloisa}@ic.unicamp.br

Abstract

Formative Assessment has been adopted as an alternative to the traditional model based on cumulative exams and has revealed itself as a crucial importance to distance education, by helping the learners' behavior perception and the problems identification and making possible more effective advice in the time of learning process, even at distance. Nevertheless, the educators need to take extra steps to monitor learners' participation and to regulate the learning processes in distance courses, resulting in the educators' overload and, consequently, the formative assessment high costs.

The possibility of saving all interactions in computer-based courses for later analysis can be explored to provide formative support. This paper presents the main results of a doctoral research that proposes a multiagent system (MAS) to support formative assessment in Learning Management Systems. The proposed MAS architecture definition process and some considerations about the implementation of a prototype that has been integrated into TelEduc LMS are discussed.

Keywords: multiagent systems, distance education, formative assessment.

1 Introduction

The formative assessment has been used as an alternative to the traditional assessment based on cumulative exams, and allows the continuous monitoring and orientation of the learning process. This assessment approach presents informative and regulatory roles, providing information to the two main actors of the teaching and learning process: the educator, that will be informed about the real effects of his/her pedagogical actions, and will be able to regulate them; and the learner, that will have the opportunity to be conscious of his/her difficulties and, possibly, to recognize and to review his/her own mistakes [Hadji 2001].

In distance education, the formative assessment is even more important since the online assessment presents some intrinsic difficulties, such as the absence of the face-to-face interactions feedback, the lack of instructor control over the assessment and the authentication problem (who is performing the assessment?). The informative and regulatory characteristics of this assessment approach have revealed themselves as a crucial importance to distance education, by helping the learners' behavior perception and the problems identification, making possible more effective advice in the time of learning process, even at distance.

Nevertheless, nowadays most Learning Management Systems (LMSs) do not present features specially designed to help this assessment approach. Thus, the educators need to take extra steps to monitor, to analyze and to advise the learning process. This scenario has been the main problem of formative assessment, face-to-face as well as at distance.

This paper presents the main results of a doctoral research that proposes a multiagent system (MAS) to support formative assessment in LMSs, aiming to reduce the instruct overload and consequently, the formative assessment costs.

The paper is structured in five sections. Section 2 shows the main phases of the definition process that resulted in the proposed MAS architecture. Section 3 presents the agents knowledge base definition and section 4 presents some considerations about the implementation of a prototype of the proposed architecture that has been implemented and integrated into TelEduc[1] [Rocha 2002, Rocha *et al.* 2002], a LMS that has been developed at State University of Campinas since 1997, and which is available as free software under the open-source GNU Public License. At last, section 5 presents some final considerations about this work.

2 The Multiagent System Architecture definition

The definition process of a MAS architecture to support formative assessment in learning management systems was composed by the following main processes: the requirements specification and the process of analysis and design based on Gaia methodology [Wooldridge, Jennings & Kinny 2000]. The next subsections present the main results obtained in these phases and the resulted proposed architecture.

2.1 Requirements Specification

The first step in the definition of the proposed MAS architecture was the main requirements specification. These main requirements have been obtained through researches on education [Hadji 2001] [Perrenoud 1998], the research group's practical experience in computer-mediated distance courses [Otsuka & Rocha 2002] [Ferreira, Otsuka & Rocha 2003] and the analysis of other researches and experiences on learning assessment in distance education [Thorpe 1998] [Hopper 1998] [Nelson 1998] [Cerny 2001] [Gomez 2001].

The requirements of two main actors in formative assessment process, educators and learners, were identified. Thus, the requirements to support the **educators'** needs are:

- To receive, continuously, relevant and reliable information about learner's participation at each proposed activity, throughout the course;
- To receive the consolidation of learner's participation at all activities proposed, making possible a global monitoring of learners' participation and the identification of the learners' participation and learning styles;
- To have mechanisms to help the analysis and interpretation of the above information, just in time, with coherence and impartiality;
- To receive help on the assessment of activities, by means of the preanalysis of learners' participation based on criteria defined in the activities plan, and the elaboration of a detailed and consistent feedback about these assessments;
- To have flexibility to define the assessment criteria of each activity.

The requirements to support **learners'** needs are:

- To receive, continuously, monitoring reports and their learning profiles, that provide a feedback of their participation in each proposed activities and in the whole course, helping them to be conscious of their difficulties and progresses;

[1] http://teleduc.nied.unicamp.br

- To consult other learners' learning profiles, making possible the knowledge of the group learning profile, what might lead to improvements in the collaborative learning.

The next step was the specification of agents "roles", that is, the behaviors that need to be implemented by the MAS to support the requirements specified earlier. The roles specification was based on Gaia methodology, and it is presented in the next subsection.

2.2 Analysis and design based on Gaia methodology

The Gaia methodology [Wooldridge, Jennings & Kinny 2000] was adopted because it is simple and efficient, and provides effective support to the specification of agents behavior (Roles Model), the inter-role protocols definitions (Interaction Model), the agent types definition (Agent Model), as well as the identification of communication links between agent types (Acquaintance Model).

At analysis phase the Roles Model and the Interaction Model were defined. The Roles Model specifies a set of agent behaviors needed to support the requirements presented at subsection 2.1. The Interaction Model identifies the communication protocols that will be specified to make possible the inter-role interactions.

At design phase the Agent Model and the Acquaintance Model were defined. The Agent Model defines the agent types, that is best thought of as a set of agent roles. To improve the system performance, the roles were grouped trying to obtain a minimal number of agent types. This process resulted in an Agent Model composed by 3 agent types (Figure 1):

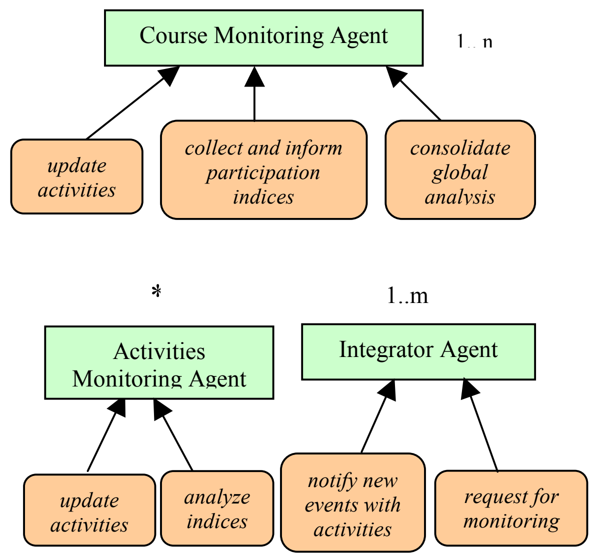

Figure 1 – The Agent Model of the proposed MAS: the agent types are represented by rectangle and the agent roles are represented by rounded rectangles. The agents cardinality is represented by: * (0 or more), 1..n (1 to n)

- **Course Monitoring Agent:** this agent type is responsible for ongoing monitoring of learners' participation throughout a course. It acts accessing the course database and collecting indices from learners' participation in each activity proposed (for example, synchronous and asynchronous discussion, projects, reports and so on). The collected indices are sent to the appropriate Activity Monitoring Agent responsible for the analysis of activities. This agent type is also responsible for a global analysis of learners' participation in one course, creating the **learning profiles.** The MAS can have *1* to *n* agents of this type, where *n* is the number of current courses in the LMS;
- **Activities Monitoring Agent:** each agent of this type is responsible for the analysis of learners' participation indices in every activity developed in one specific tool of the LMS (such as *Discussion Forums, Chat, Portfolio, Mail, Bulletin Board and so on*). The MAS can have *1* to *m* agents of this

type, where *m* is the number of monitored tools in the LMS;

- **Integrator Agent:** this agent type provides the MAS interface with the educator: it receives events from LMS and informs them to the appropriated agents in the MAS. A new Integrator Agent should be created when one educator logs in the LMS and should stay "alive" until the end of the session.

The Acquaintance Model was obtained directly from the Agent Model (Figure 2), and identifies the communication links among the three different agent types with the purpose of identify any potential communication bottlenecks, that could cause problems at run-time [Wooldridge, Jennings & Kinny 2000].

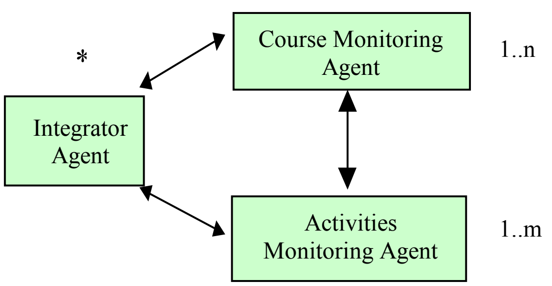

Figure 2 – The Acquaintance Model of the proposed MAS

2.3 The multiagent system architecture

The MAS architecture has been obtained from the two process presented in the previous subsections. Figure 3 shows the proposed architecture integrated into a LMS that contains ***n*** courses and ***m*** tools to be monitored. Thus, to support this LMS the proposed MAS architecture could include ***n*** Course Monitoring Agents and ***m*** Activity Monitoring Agents. At each time an educator logs in the LMS, one Integrator Agent is created to provide the interface between the MAS and the educator in the current session.

The proposed architecture seems to be easily extended to support the monitoring of new tools incorporated into the LMS. This is achieved by the implementation and addition of a new Activity Monitoring Agent, responsible for the analysis of learners' participation in the context of the new tool. The Course Monitoring Agent's roles may be slightly changed to be able to deal with the new tool monitoring. Nevertheless, they are precise changes, that will not interfere in the previously defined roles (see discussion on section 4.3).

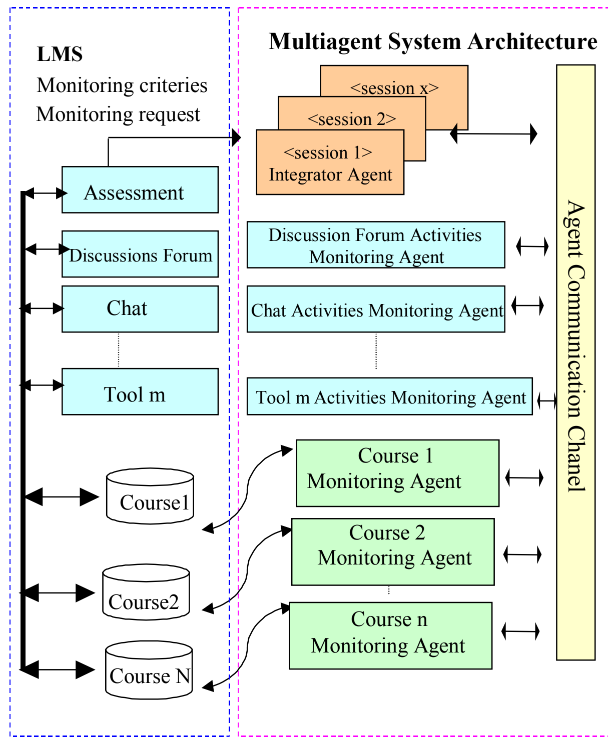

Figure 3 – The proposed MAS integrated with a LMS

3 The knowledge base definition

The roles of Course Monitoring Agents and Activities Monitoring Agents involve information analysis and new knowledge creation. Thus, the definition of these agents' knowledge bases was needed. A knowledge base has been defined for each Activity Monitoring Agent to make possible the analysis of learners' participation in activities developed in the monitored tool.

The knowledge representation has been done by means of frames and production rules. This process has been supported by

Protégé-2000[2] and Algernon[3] tools that are presented at section 4.1.2. An initial set of rules has been defined with the participation of researchers of Nucleus of Informatics Applied to Education (Nied) at State University of Campinas, based on group's experiences with formative assessment in distance education.

At the time of an activity creation, the educator selects some criteria that are going to reflect the rules to be considered on this activity monitoring. The educator also can define some reference indices that will be used as parameters of rules at the execution time, providing some flexibility to these rules.

Thus, when new monitoring indices are collected through a Discussion Forums' Activity and instantiated into the *Discussion Forums' Activities Monitoring Agent'* knowledge base, these rules are executed and new knowledge can be produced.

The knowledge base definition is at initial phase and may be refined with users' suggestions. In order to make this refinement possible, these knowledge bases modeling have been focused on an easily extended and not source-code dependent solution. In order to make easier the new rules creation, the design of an interface that makes possible the creation of new rules directly by final users is proposed as future work.

4 Implementation Considerations

A prototype of the proposed MAS has been implemented and it is going to be integrated into the TelEduc LMS, composing the solution for formative assessment support proposed for this system in [Otsuka *et al.* 2003]. The following subsections present some considerations on the technological support that has been explored in the implementation, and on the integration of the proposed MAS into TelEduc.

4.1 Technology investigation

Some tools have been analyzed in order to make easier the implementation of the proposed MAS, namely, tools to support the development of multiagent systems, tools to support the construction of the knowledge base and inference systems. In the implementation of the prototype, the following tools have been adopted: the Jade MAS development framework[4], the Protégé-2000 knowledge base editor[5] and the Algernon rule-based inference system[6], all of them are briefly described in the following subsections.

4.1.1 Jade

JADE is free software and is distributed by TILAB, as open source software under the terms of the LGPL (Lesser General Public License Version 2). It is implemented in Java[7] and aims to make easier the development of multiagent systems by providing a comprehensive set of services and an API to support the development of agents in compliance with the FIPA specifications [Gluz & Viccari 2003].

The JADE framework provides an API with several classes that represent common entities in MAS, supporting the definition of agents, agents' behaviors, FIPA-ACL-compliant messages, interaction protocols, and so on [Bellifemine *et al.* 2003]. In the time of vocabulary and semantics definition of the agents' communication content, a solution based on Java objects usage as the content of the messages exchanged among the agents has been adopted, as proposed in [Vaucher & Ncho 2004]. The JADE utilization has been essential to the agents implementation in the proposed MAS,

[2] http://protege.stanford.edu/
[3] http://smi.stanford.edu/people/hewett/research/ai/algernon/

[4] http://sharon.cselt.it/projects/jade/
[5] http://protege.stanford.edu/
[6] http://smi.stanford.edu/people/hewett/research/ai/algernon/
[7] http://java.sun.com/

simplifying and significantly speeding up the process.

4.1.2 Protégé–2000 and Algernon

Protégé-2000 is an extensible, platform-independent ontology and knowledge-base editor, developed by Stanford Medical Informatics at the Stanford University School of Medicine. Protégé is available as free software under the open-source Mozilla Public License [Gennari *et al.* 2002].

Protégé provides a simple and powerful graphical interface, integrating the ontology definition and creation tasks and the insertion of instances into knowledge base task. Another very interesting feature is the Protégé-2000's extensibility, since it is a platform that can be extended with graphical widgets such as tables, diagrams and animation components. Also, due to its plugin architecture, it is possible to customize and extend Protégé's features according to the project specific needs [Musen *et al.* 2000].

In the prototype implementation, the Algernon Tab plugin[8] has been used, which is responsible for the integration of Protégé and Algernon, an inference system that provides backward and forward chaining rules to infer data in knowledge bases. This plugin makes possible to execute Algernon commands directly in the Protégé-2000 GUI, and also operates directly on the latter's knowledge bases, requiring no mapping between different memory areas. This direct manipulation of Protégé-2000 data is very interesting, because it provides some performance gain [Hewett 2003]. Algernon is available as free software under the Mozilla Public License as well.

The adoption of Protégé-2000 and Algernon in the prototype implementation has made possible using Protégé-2000's graphical interface directly to create ontologies and knowledge-base rules. This is completely independent of the agents Java code, thus making the maintenance of knowledge bases easier.

4.2 Technology integration

The three different technologies discussed are Java-based and offer APIs that make easier the integration among Jade, Protégé-2000 and Algernon. As seen before, building an agent with the Jade framework is realized by extending classes already defined in the framework. The manipulation of one agent's knowledge base is possible by importing classes from Algernon's APIs.

The server runs by starting a Jade platform, which is the environment where the agents will live in, and then creating Activity and Course Monitoring agents. These agents' knowledge bases edited with the Protégé-2000 GUI are stored in files that can be accessed using Algernon's API in the agents code. This API allows creating an instance of the Algernon inference machine, adding a knowledge base to the list of KBs this Algernon instance accesses, and querying and inserting data into them.

Thus, after the definition of frames, slots and relations between frames, all using Protégé-2000's graphical interface, the insertion of instances into an agent's knowledge base is done using Algernon's API methods. When these data are inserted into the knowledge base, the previously defined rules (using the Algernon Tab plugin) are automatically triggered in the Algernon inference machine to process the new entries and to try to recognize any pattern. Whenever the inference machine discovers something, a new instance is created reporting this inferred situation, which can be queried later by the community agents.

4.3 Integration of the proposed MAS into TelEduc

The prototype integration into the TelEduc LMS has started by focusing on the learners' participation monitoring in two of the

[8] http://algernon-j.sourceforge.net/doc/algernon-protege.html

communication tools of this LMS, the Discussion Forums and the Portfolio. This decision is explained by the relevance of these two tools in the context of formative assessment. These tools, since they are asynchronous, allow deeper reflection by the learner at the moment of his/her participation, resulting in more relevant and consistent contribution. The asynchronous characteristic also helps the educator's action in the monitoring and regulating learners' participation throughout the activities development, although this process is generally too expensive due to the huge volume of information to be analyzed.

The Discussion Forums tool interface has been adapted to make possible select the degree of relevance (not relevant, low relevance, medium relevance, relevant, high relevance) of a message. Thus, it will be possible to record the participation as well as to analyze the records of the learners' participation assessments.

The integration of the proposed MAS into TelEduc has been done through the Integrator agent that runs at an user's browser's applet and is responsible for the communication with server-side agents notifying relevant events at LMS. The Integrator agent is activated when necessary by invoking one of its methods, which is done by a Javascript function call in the LMS code, that passes on to the applet, and thus to the Integrator agent in it, whatever information is needed. The Integrator agent, once receives these data, must pass it to the server-side agents (Activity and Course Monitoring agents), by sending an ACL-message through HTTP using Jade's HTTP MTP (Message Transport Protocol). Once the server-side agents receive the data, they process it according to predefined protocols, and the communication between the Integrator agent and the server-side agents is done for that given piece of information that server-side agents needed to know.

After the implementation of the MAS for the monitoring these two tools, it will be possible to easily extend the system to monitor new tools, since the core of the system will already be implemented and most of the existing classes will possibly be used and/or extended to implement new agents. Thus, the extension of the system to monitor a new tool is expected to be done in 3 main steps:

- Creation of new methods for the Course Monitoring Agent (to collect participation indices related to the new tool);
- Creation of new methods for the Integrator Agent (to receive requests for reports about the new tool monitoring and to present the analysis results);
- Implementation and instantiation of a new Activity Monitoring Agent and creation of the knowledge base of this new agent to analyze the learners' participation indices in activities developed with the new tool.

5 Final Considerations

This paper presented the definition process of a MAS architecture to support formative assessment in learning management systems. The main goals of this process was to propose one solution that: (1) tries to minimize the number of agents, avoiding one fast explosion of this number when many courses and tools must be monitored; (2) is extensible, making possible to easily extend the system to monitor new tools; (3) is generic, making possible to easily adapt it to other LMSs.

The knowledge base definition was concerned with: (1) the flexibility, making possible the rules adaptation according to the educator's needs; (2) the easy rule set extension, by disconnecting the rules' edition and system implementation.

The Gaia methodology employment was fundamental in the analysis and design phases of the MAS architecture, helping mainly in the roles and agent types definition. In the implementation phase, the utilization of Jade, Protégé-2000 and Algernon technologies was fundamental because they significantly made easier and sped up this process. The next steps of this project involve: (1) to finish the agents' behaviors implementation; (2) to refine the knowledge bases of Discussion Forums and Portfolio Activities Monitoring Agents; (3) to design the MAS user interface, that challenges to present a huge volume of information and to maintain the consistence with the current TelEduc user interface.

The proposed MAS is part of the formative assessment support model proposed to TelEduc in [Otsuka 2002b, Otsuka *et al.* 2003]. The other researches that compose this model are briefly described below:

- One proposes and implements the Assessments Tool, aiming to make easier the assessment management (by storing and loading observations and grades assigned by educator during the proposed activities development) [Ferreira 2003]. These research results have been used in the proposed MAS architecture, providing important learners' participation qualitative indices to be analyzed by the agents of the proposed architecture;
- Another that explores interface agents to do an initial selection of the relevant messages of one chat session, according to educator's interests [Lachi, Otsuka e Rocha 2002; Lachi 2003], in order to reduce the educator overload during the analysis of a chat session log. In order to integrate this new chat tool into the proposed architecture, its interface should be adapted to make possible indicate the relevance degree of each selected message (not relevant, low relevance, medium relevance, relevant, high relevance). Besides, the proposed architecture should be extended to make possible monitor this new chat tool;
- The last one aims the development of a set of exercises management tools (multiple choice questions, true or false and subjective questions) to supply TelEduc users' needs [Freitas 2002]. The proposed MAS architecture should be appropriately extended to support these tools monitoring.

This model explores the ease of recording anything that happens in a computer-mediated distance course, in order to provide formative assessment support, by means of agent technology.

6 References

Bellifemine, F. ; Caire, G.; Trucco, T.; Rimassa, G (2003) . JADE Programmer's GUIDE. Available at: http://sharon.cselt.it/projects/jade/

Cerny, R.Z. (2001). Uma reflexão sobre a avaliação formativa na educação a distância. UFSC, 2001. Available at: http://www.anped.org.br/24/tp1.htm

Ferreira, T. B. (2003). Gerenciador de Avaliações: Uma Ferramenta de Auxílio à Avaliação Formativa para o Ambiente de Educação a Distância TelEduc. Master Thesis, Institute of Computing/State University of Campinas, December, 2003.

Ferreira, T. B.; Otsuka, J.; Rocha, H. (2003). Interface para Auxílio à Avaliação Formativa no Ambiente TelEduc. In: Proceedings of XIV Simpósio Brasileiro de Informática na Educação (SBIE 2003). Rio de Janeiro, November, 2003.

Freitas, C. E. F. (2002). Desenvolvimento de uma Ferramenta de Avaliação Objetiva para Ambientes Computacionais de EaD. Master Qualify, Institute of Computing/ State University of Campinas, December, 2002.

Gennari, J. ; Musen, M. A.; Fergerson, R. W. ; Grosso, W. E.; Crubézy, M.; Eriksson, H. ; Noy, N. F. ; Tu, S. W. (2002) The Evolution of Protégé: An Environment for Knowledge-Based Systems Development, 2002. Available at: http://protege.stanford.edu/

Gluz, J. C.; Viccari, R. M. (2003) Linguagens de Comunicação entre Agentes: Fundamentos Padrões e Perspectivas. In Vieira, R., Osório, F., and Rezende, S., eds., Proceedings of III Jornada de Mini-Cursos de Inteligência

Artificial, vol. 8. Campinas: SBC. 53-102, July, 2003.

Gomez, M.V. Avaliação Formativa e Continuada da Educação Baseada na Internet. Proceedings of VI Congresso Internacional de Educação a Distância. ABED, 1999.

Hadji, C. (2001). Avaliação Desmistificada. Porto Alegre: Artmed, 2001.

Hewett M.S. (2003). Algernon in Java Documentation – Available at: http://algernon-j.sourceforge.net/doc

Hopper, M. (1998) Assessment in WWW-Based Learning Systems: Opportunities and Challenges. Journal of Universal Computer Science 4, no. 4 (1998): 329-347. Available at: http://www.jucs.org/jucs_4_4/assessment_in_www_based/paper.html

Lachi, R. L. Chapa: Um Agente de Interface para Ferramentas de Bate-papo em Ambientes de Ensino à Distância na Web. Master Thesis, Institute of Computing/ State University of Campinas, 2003.

Lachi, R; Otsuka, J.; Rocha, H. (2002). Uso de Agentes de Interface no Suporte à Análise de Sessões de Bate-Papo. In: Proceedings of 5th Workshop on Human Factors in Computer Systems (IHC 2002). Fortaleza, October, 2002.

Musen, M. A.; Fergerson, R. W. ; Grosso, W. E.; Noy, N. F.; Crubezy M., & Gennari, J. H..(2000). Component-Based Support for Building Knowledge-Acquisition Systems. In: Proceedings of Conference on Intelligent Information Processing (IIP 2000) of the International Federation for Information Processing World Computer Congress (WCC 2000), Beijing, 2000.

Nelson, G. E. (1998). On-Line Evaluation: Multiple Choice, Discussion Questions, Essay, and Authentic Projects. In: Third Teaching in the Community Colleges Online Conference, Kapiolani Community College, Hawaii, April 7-9, 1998. (ED 430 695).

Otsuka, J. L. (2002). Um Modelo Baseado em Agentes de Interface para o Suporte à Avaliação Formativa em Ambientes de Educação a Distância. Campinas: Instituto de Computação da UNICAMP. Doctoral Qualify. Institute of Computing / State University of

Otsuka, J. L.; Rocha, H. V. (2002) Avaliação Formativa em Ambientes de EaD. In: Proceedings of XIII Simpósio Brasileiro de Informática na Educação (SBIE 2002). São Leopoldo, 12-14 de novembro, 2002

Otsuka, J. L., Ferreira, T. B., Lachi, R. L., Rocha, H. V.(2003) Um Modelo de Suporte à Avaliação Formativa no Ambiente TelEduc. Revista Brasileira de Informática na Educação, SBC, 2003.

Perrenoud, P.(1999). Avaliação: da excelência à regulação das aprendizagens entre duas lógicas. Porto Alegre: Artes Médicas, 1999.

Rocha et al.(2002). Projeto TelEduc: Pesquisa e Desenvolvimento de Tecnologia para Educação a Distância. Prêmio de Excelência ABED/Embratel 2002. IX Congresso Internacional de Educação a Distância da ABED, September, 2002.

Rocha, H. (2002). O ambiente TelEduc para Educação à Distância baseada na Web: Princípios, Funcionalidades e Perspectivas de desenvolvimento. In: Moraes, M.C. (Org). Educação à Distância: Fundamentos e Práticas. Campinas, SP:Unicamp/Nied, 2002, pp. 197-212.

Thorpe, M. (1998) Assessment and 'Third Generation' Distance Education. Distance Education 19, n. 2, 1998: 265-286. (EJ 582 131).

Vaucher, J. Ncho, A. (2004). Jade Primer. Available at: http://www.iro.umontreal.ca/%7Evaucher/Agents/Jade/JadePrimer.html.

Wooldridge, M.; Jennings, N.R.; Kinny, D. (2000) The Gaia Methodology for Agent-Oriented Analysis and Design. Journal of Autonomous Agents and Multi-Agent Systems Vol. 3. No. 3 285-312.