XIX Seminarium

ZASTOSOWANIE KOMPUTERÓW W NAUCE I TECHNICE' 2009
Oddział Gdański PTETiS
Referat nr 18

A PROTOTYPE OF EDUCATIONAL AGENT IN DISTANCE LEARNING ENVIRONMENT - VIRTUAL STUDENT ASSISTANT

Agnieszka LANDOWSKA[1]

1. Politechnika Gdańska, Wydział Elektroniki, Telekomunikacji i Informatyki
tel: (58)347-29-89 fax: (58)347-27-27 e-mail: nailie@eti.pg.gda.pl

Abstract: E-learning is supported with many different types of software, and among those are educational agents, which aim to assist a learner during an educational process. One of the educational agent types are virtual assistants. The article presents a program called WAS, an assistant agent implemented at Gdańsk University of Technology. Functions of the WAS prototype were described as well as its cooperation with e-learning environment. Article presents the agent construction and some implementation details. Main problems in agents implementation were pointed out, showing challenges of further research in the field of educational agents. The agents are believed to be some solution to selected problems in e-learning environments.

Key-words: intelligent tutoring systems, educational agents, e-learning

1. INTRODUCTION

An educational agent, sometimes called tutor or mentor, is a type of application for technology enhanced learning. It is designed to help the learner to follow training paths. Most important feature of the agents is that they introduce social context in e-learning environment – the agents can play role of a virtual teacher or a virtual learning companion. Agents are usually visualized with an avatar and they seem to be 'human face' of educational systems. Agents fall into one of three categories: mentors, learning companions and information bots, depending on their main objective in the learning environment. Mentor agents are virtual teachers or coaches, that aim to provide the learner with an illusion of learning with the real teacher [1, 2]. Their role is to provide knowledge and to motivate the learner to follow the designed educational path – they can ask questions or even answer ones, give tips, etc. Learning companion agents are designed to enhance training efficiency by introduction of competition [3, 4]. Last, but not least, information bots provide information about the systems, resource access, training features, etc. They can be general (guides) or personalized (assistants). This article will show an example of the educational agent called WAS, that falls into category of information bots, subcategory of learner assistants.

2. WAS AGENT

WAS is a prototype of an agent implemented at Gdańsk University of Technology as a result of author's research. Acronym WAS comes from polish name of Virtual Student Assistant (pol. Wirtualny Asystent Studenta). As a personalized assistant, WAS aims to accompany a student in e-learning environment. The agent functionality includes educational resources search and acquisition as well as student's progress tracking.

2.1. WAS agent functionality

The main task of the WAS agent is educational content searching using external servers and Web services. The search is based on a subset of metadata attributes, selected from the IEEE standard [5]. Nowadays educational resources are distributed among many sites in the Internet and can differ significantly in size, format, quality and difficulty, which makes manual search process quite hard. Content interoperability problem is addressed by many standard organizations, like Advanced Distributed Learning. However, some of educational content producers set up repositories and central search registries to publish the resources and to make them searchable. The registries like UDDI servers are able to manage the metadata and links to resources, allowing to search and download the content. The WAS agent is designed to communicate with UDDI registries and content repositories by means of web services and can help learner to search and obtain educational resources. The content is downloaded on demand, and the role of the agent is also to manage a local repository of educational content. The agent can check for new versions of data or wait in case a server is down. These functions are transparent to the user [6].

The second function of the WAS agent is to track student's activity with the educational content. The agent can cooperate with resources compliant with the SCORM model [7]. Any downloaded content can contain new or continued training. Trainings accessed by the user can be in several states: not-downloaded, not-started, started-but-not-finished, finished-but-not-passed, passed, etc. The WAS agent's role is to remember the state of each user training. It is user preference, whether the agent reminds of the learning path states or not, and which states and trainings are monitored.

Recenzent: Dr inż. Piotr Szprynger – Wydział Elektroniki, Telekomunikacji i Informatyki Politechnika Gdańska

The agent stores results of finished tasks and can share it with SCORM-compliant LMS [6].

2.2. WAS agent in e-learning environment

At the beginning, e-learning environments were based on the few-to-many model, where few educational content producers offered it to, potentially, many students. But, due to public access to publishing any content in the Internet, much content was created independently by different educational organizations as well as individuals. Additionally, the evolution of Web 2.0 technologies, like forums, chats, blogs, group work tools, influenced the e-learning environments, creating Education 2.0.

Education 2.0 is based on many-to-many model, where knowledge (and educational content) can be produced and published by academic or industry professionals as well as learners. In education 2.0 competences are created by knowledge sharing, collaboration and teamwork of a community of students. As a result the content can be created almost by anybody, raising at least two major problems: content search and quality assessment.

WAS agent is not capable of solving quality problems, but is designed to address the issue of content search and acquisition. WAS agent in the e-learning environment is shown in figure 1.

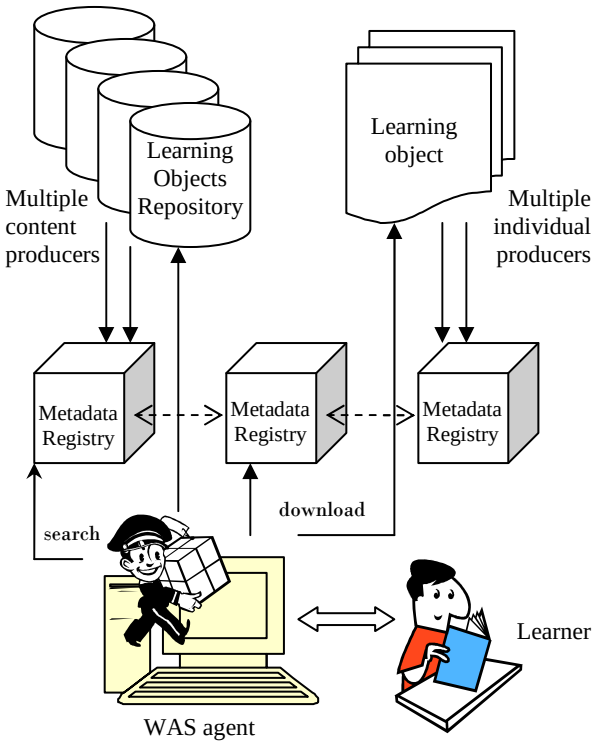

Fig. 1. WAS agent in e-learning environment

With multiple organizational and individual content producers as well as multiple search engines, educational resource acquisition can be very complex and time-consuming. WAS agent can act on behalf of the learner and download content. The process starts with content producers, who publish educational content divided into smaller units called learning objects and then register objects' metadata in one of the search repositories, f.e. UDDI registries. Metadata is a set of attributes that describes the learning object – its content, format, copyrights, etc. UDDI registries containing metadata attributes can synchronize their content with each other, so contacting one of them gives access to complete available content list. While one of the servers is down, the others can be accessed making search mechanisms available. WAS agent interacts with user, who defines topics of interest, potential formats and financial constraints (f.e. only fee-free content). Then the agent monitors metadata registries looking for learning objects that fit the learner requirements and then downloads content from individual sites or learning object repositories. The user can change requirements and define a profile including languages, formats or preferred agent actions.

2.3. WAS agent interface

The main window of the WAS agent comprises three sections: the display area, agent visualization and conversation area. Results of main agent tasks, i.e. resources found or content tree are shown in the display area. The visualization of the agent is placed at the right bottom corner, while its messages are placed just above it. WAS agent interface is shown in figure 2.

Main menu of WAS agent consists of the following sections: learning objects (pol. obiekty edukacyjne), categories (pol. kategorie), languages (pol. języki) and help (pol. pomoc). Learning objects submenu allows the user to browse and manage local repository of learning objects, i.e. move, delete objects, as well as to set up new tasks concerning content search. Categories submenu allows the user to add, edit or delete categories, which are used for grouping downloaded learning objects within user-defined taxonomy.

Results of the WAS agent actions are visible in the display area. Figure 2 shows the content of local repository. Downloaded content is assigned initially by the agent to overall category New (pol. Nowe). The user can create multiple categories and sub-categories creating a taxonomy of his own. Exemplary taxonomy is shown on Display area in figure 2.

The WAS agent is able to have a simple natural language communication with the user. Conversation area of WAS interface displays messages from the agent to the user. At the beginning the agent introduces itself, saying "Hello, I'm your assistant". Then the messages are chosen from a predefined set of conversation scenarios. The choice is based on several criteria: the actions undertaken by the user, latest finished tasks and events, interval from last message. Conversation history can be browsed by the learner.

Natural language conversation is a quite complex task. WAS is a prototype – conversation is only one-way. More advanced two-way conversation will be available in the next version of WAS agent, called KAY. WAS agent visualization is a static picture, however KAY will be represented with an animated avatar. More information on the KAY project is available at www.snowqueen.edu.pl website.

3. WAS AGENT IMPLEMENTATION

WAS agent architecture is a result of several major requirements, especially the cooperation with SCORM compliant content and non-functional feature of portability. WAS agent was supposed to perform tasks of content search and download independently from the user switching his computer on or off. Therefore a client-server architecture was chosen. Only small part of the WAS agent functionality was implemented at the server side i.e. resource search and acquisition.

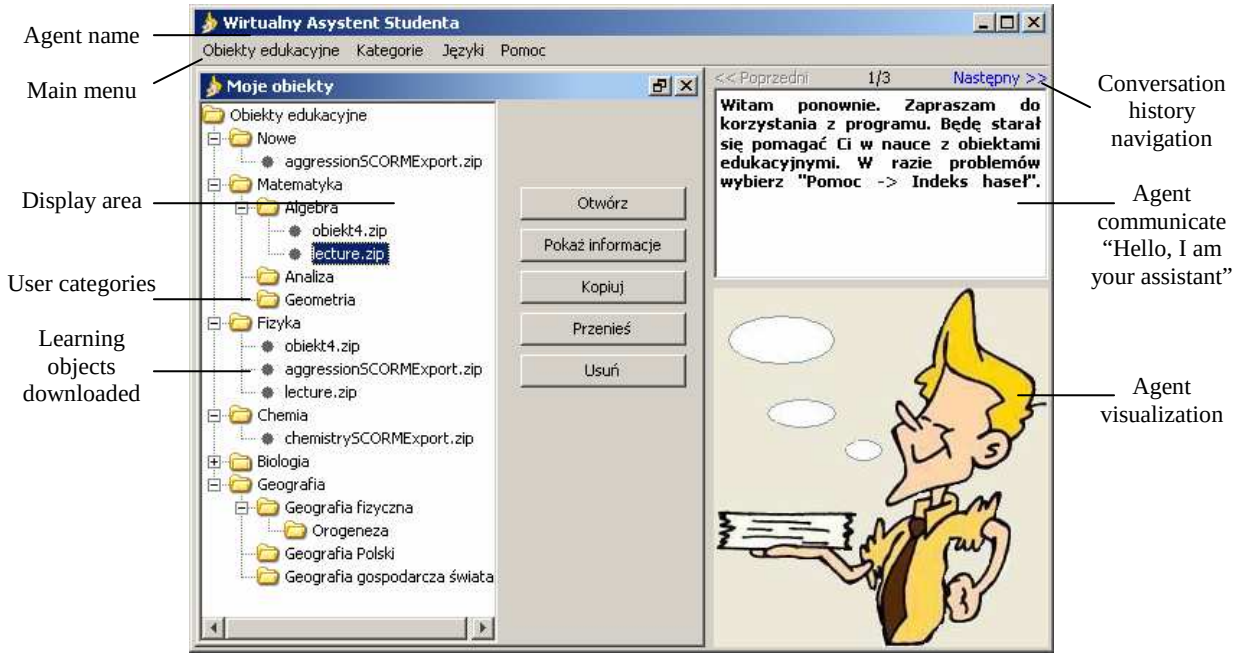

Fig. 2. WAS agent interface

The WAS agent has client-server architecture. The main part of the application is client-side. Results of WAS server-side modules actions are downloaded if client-side application is active. Both server-side and client-side parts of the WAS agent are implemented with Java programming language. Educational agents, including WAS assistant, should not be seen as examples of software agents in the sense of distributed programming. Most of educational agents are not mobile and the agent cooperation is not required (however, sometimes possible). Most of the WAS application data is locally managed and stored on the user's personal computer. Server-side modules store current tasks and its results only as long as the client application is switched off. This project decision was made as a result of analysis of typical learning object content. Most of the learning objects have medium size, consist of multiple files and are designed to be displayed in Internet browsers, which is compatible with SCORM reference model. Many e-learning courses are filled with multimedia and animations, many contain self-assessment tests and other scripts. To avoid delays in learning object browsing or display, a decision was made to store them locally on learner's personal computer. Educational content repository is stored using Apache DERBY database in embedded mode – integrated with the client-side application, so that no additional step during installation process is necessary. System configuration and WAS agent conversation scenarios are stored independently in file system.

Client-side application is assumed one-user only in order to make individual categorization of learning content possible. Server-side application is shared among multiple users and web services technology was used for client-server communication. Search function of WAS server-side application is supported with UDDI server and a set of metadata attributes was selected, including keywords, difficulty, aggregation level and publication language.

4. WAS AGENT ASSESMENT

There are several criteria of the educational agents usability assessment, especially functionality, reliability, efficiency and ergonomics. Incorrect behavior or delays in communication may result in the learner switching the agent off or not paying attention to the information provided by it.

WAS agent passed all of the designed functional tests, concerning organizational tasks: content search and download, student's progress tracking, etc. Only educational tasks were left to the student himself, which was the initial assumption.

WAS agent reliability is mostly dependent on the server-side part of the application. Client-side desktop application is stable, however the communication with server-side modules is dependent on the Internet connection. WAS desktop application works even if the connection to the server-side modules is broken, tasks would start immediately after the connection is restored. Efficiency of WAS agent is satisfying, content search and download takes less than few seconds. However, this feature is dependent on server power and load, especially with many client applications active. WAS agent ergonomics assessment was also positive. Most of the users stated, that the interface is intuitive. Agent functions are two- or three- clicks ahead from the main WAS window. WAS agent can be minimized to the icon on the system task bar. The agent allows the learner to concentrate on the training, which was the goal of the WAS agent.

High quality of the agent is the basis for guaranteeing a more important feature – credibility [9]. Credibility means, that the agent is reliable enough in the role it plays, that the information it provides can be trusted. A learner needs to be ensured that knowledge presented by the assistant is valid. A good assistant agent should also accompany the learner during the learning process, but should not be insistent. WAS agent can be assessed as discrete, as it follows the learner rather than initiates an action.

However WAS prototype lacks some of the features influencing overall credibility as an assistant – the possibility of two-way natural language conversation and agent visualization are missing. Those two attributes would influence the impact of WAS agent by the creation of illusion of interacting with human-being – social context introduction is one of the expected features of educational agents.

5. PROBLEMS IN AGENT CONSTRUCTION

Construction of a credible assistant agent is rather complex. The issues arising can be divided into technical problems and AI limitations. The main technical problem with assistant agents and other LMS platforms is interoperability of educational content. The problem is addressed by some of the standards, but remains unsolved, due to diversity of formats and a large number of content producers.

Efficient and credible educational agent should be able to make an illusion of working with a human-being. There is a problem of limitations of artificial intelligence methods, like human-computer natural language conversation, decision making or expressing emotions, resulting in Turing test not passed since 1950 until now. However, despite the limitations, some of the educational agents proved to be useful already.

6. CONCLUSIONS

There are several advantages of software agent introduction in distance education environment. Assistants aim to help a student to deal with organizational tasks, leaving more time for training itself. A good tutor or assistant agent can introduce social context into the learning environment, which aims to help a student to follow the learning process and therefore reduce the threat of resigning before training completion. By the introduction of interaction, especially natural language conversation, the learner is more motivated and concentrated on the tasks performed. The learning results are expected to be better and remain longer.

In the future WAS or other virtual assistants can be introduced in learning environments as well as other areas of application.

7. REFERENCES

1. Payr S.: The Virtual University's Faculty: An Overview Of Educational Agents, Applied Artificial Intelligence, vol. 17, Taylor&Francis, United Kingdom, 2003, pp. 1-19, ISSN: 0883-9514.
2. Baylor A. L.: Beyond butlers: Intelligent agents as mentors, Journal of Educational Computing Research, vol. 22(4/2000), Baywood Publishing Company, USA, 2000, pp. 373-382, ISSN: 0735-6331.
3. Rickel J., Johnson L.: STEVE: An Animated Pedagogical Agent for Procedural Training in Virtual Environments, Sigart Bulletin, vol. 8, number 1-4, ACM Press, USA, 1997, pp. 16-12, ISSN:0163-5719.
4. C. Y. Chou, T. W. Chan, and C. J. Lin, Redefining the learning companion: the past, present, and future of educational agents, Computers & Education, vol. 40, Elsevier, Netherlands, 2003, pp. 255–269, ISSN: 0360-1315
5. Learning Technology Standard Commitee IEEE, IEEE P1484.12.1-2002 Standard for Learning Object Metadata, http://ltsc.ieee.org/, access date 23.03.2005
6. Landowska A.: Let the student talk! Use of Conversational Agents in Distance Education Environments to Achieve Active Learning, Innovation in Learning Communities, Proceedings of 2009 EDEN Annual Conference, 10-13 June 2009, Gdańsk, Poland, pp.7, ISBN 978-963-87914-2-9.
7. Advanced Distributed Learning (ADL) Sharable Content Object Reference Model (SCORM), 2nd edition, 2004, http://www.adlnet.org/, access date 24.01.2005
8. Landowska A.: Rola agentów edukacyjnych w środowiskach zdalnego nauczania, XVII cykl seminariów Zastosowanie komputerów w nauce i technice, Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej, nr 25/2008, s. 83-86, ISSN 1425-5766.
9. Landowska A.: The role and construction of educational agents in distance learning environment, Proceeedings of the 1st International Conference on Information Technology Gdańsk, 19-21 May, 2008, pp.321-324, ISBN 978-83-60779-03-3.

PROTOTYP AGENTA EDUKACYJNEGO W ŚRODOWISKU ZDALNEGO NAUCZANIA – WIRTUALNY ASYSTENT UCZNIA

Słowa kluczowe: inteligentne systemy uczące, agenty edukacyjne, e-edukacja

W zdalnym nauczaniu pojawia się wiele systemów wspierających, z których niezwykle ciekawym przykładem są agenty edukacyjne. Wśród wielu rodzajów agentów edukacyjnych wyróżnia się osobistych asystentów, których rolą jest organizacyjna pomoc osobie zdobywającej wiedzę. Artykuł jest poświęcony zaimplementowanemu na Wydziale ETI Politechniki Gdańskiej prototypowi agenta edukacyjnego o nazwie WAS (Wirtualny Asystent Studenta). Pokazana została funkcjonalność zaimplementowanego prototypu – wyświetlanie komunikatów, poszukiwanie materiałów, sprawdzanie nowych wersji, katalogowanie pobranych zasobów. Zaprezentowana została współpraca agenta z rozproszonymi repozytoriami materiałów edukacyjnych. Omówiono architekturę asystenta WAS. Na przykładzie agenta WAS pokazano problemy i wyzwania budowy agentów edukacyjnych. Agenty edukacyjne mogą stanowić rozwiązanie dla pewnych problemów o charakterze społecznym występujących w procesach zdalnego nauczania.