Proceedings of the Second IASTED International Conference
EDUCATION AND TECHNOLOGY
July 17–19, 2006, Calgary, AB, Canada

USING SOFTWARE AGENT TECHNOLOGY TO SUPPORT *E*-LEARNING

Weidong Pan and Igor Hawryszkiewycz
Faculty of Information Technology
University of Technology, Sydney
PO Box 123 Broadway, NSW 2007, Australia

ABSTRACT

This paper presents an agent-based approach to supporting *e*-learning. In contrast with other work, we facilitate learning of learners by providing them with a set of services that are useful to construct knowledge. The autonomy of learners in learning is sufficiently supported and meanwhile the supportive services implemented by software agents assist them to get through a learning process. In this paper, the features of *e*-learning are first briefly reviewed and then the challenges for supporting *e*-learning - are analyzed. An agent-based approach to supporting *e*-learning is proposed. Then, the development of the UOL (units of learning) database is explored. After that, the preliminary implementation of the services is reported.

KEY WORDS

e-learning, software agents, supportive services, learner control, UOL database

1. Introduction

E-learning is being recognized as an important means for provision of learning opportunities for more learners at a low cost. Learners are confronted with some particular challenges in *e*-learning. Because they are physically separated with both their peers and teachers in space and/or time, they have to independently get through the learning process. For some learners, they may have problems to choose an appropriate learning material, determine a proper learning method to conduct relevant learning activities for their learning goals. Some learners may even have no ideas for how to evaluate their own learning outcomes and align learning towards their goals based on practical learning progress. No debate, supportive services are necessary to help them solve these problems; otherwise they would be in an isolation and disorientation situation.

Yet, the support for *e*-learning is still quite limited. This paper presents an agent-based approach to supporting *e*-learning. It differs from the ways that intelligent tutoring systems (ITS) and intelligent computer-aided instruction (ICAI) systems have adopted. ITS and ICAI enhance learning through using a variety of learner models, expert models and tutorial models [1]. They *pre-design* all possible learning routes for a course based on those models and then appoint one to a learner based on the category that the learner belongs to according to the learner model being used. The learner can only follow the learning path chosen by the system and has little or no choice about what and how he or she would like to learn. Clearly, those *pre-set* paths cannot possibly specify all of the ways in which learners may go about trying to solve a problem [5], because learners may use their own preferred ways to study due to their different backgrounds, interests, styles, motivations, capabilities, etc. Frequently learners are forced down a learning path that does not suit them, or even limits the development of their cognitive abilities. By contrast, we provide services for learners to *assist* them to conduct learning activities, and we offer the services in *non-intrusive* ways. Learners are *not* forced to take any learning activities and their autonomy in learning is sufficiently supported. Meanwhile a broad range of services is supplied for them to guide and scaffold the learning process [9]. These services go beyond simply presenting course materials, but supply a wider range of technological facilities, tools and services to support the learning process on the *e*-learning environments.

In the following, the agent-based approach will be presented. The UOL (units of learning) database that provides information for the agents to supply individualized supportive services for different learners based on practical learning scenes will be explored. The preliminary implementation of the services to support *e*-learning will be reported.

2. Agent-based approach to support *e*-learning

2.1 Challenges to support *e*-learning

As it can be seen, learners in *e*-learning are assumed with the responsibility to select the topics they want, control the pace and sequence in which they study the topics. Early research into learning has indicated that a great degree of autonomy should be given to learners in order to enable them to realize their full potential. They should be allowed to take a full control over their learning activities. On the other side, however, not all learners are equally capable of adequately making these decisions [8], e.g. some may have problems to independently choose a learning resource and determine a proper method to perform the learning process toward a particular goal. This clearly suggests the necessity to provide supportive services for learners to assist them to solve these problems. For this purpose, the system has to take control of the learning process. The more assistance the system offers, the greater degree of control the system needs to take. How to solve this problem? This is in fact in a dilemma. It is a challenge to provide learners with services to assist them to solve the problems in *e*-learning and at the meanwhile the autonomy of learners can still be supported.

Clearly it is not acceptable if the system takes fully control over the learning process like that in ICAI or ITS, because in that case learners can only *passively* follow a predetermined learning route chosen by the system. We tackle the challenge by making all the supportive services tailored to learners on an individual basis. The system provides individualized supportive services in an adaptive manner according to the learning characteristics of individual learners so that the individual needs and preferences can be accommodated. To enable learners to take the responsibilities for their own learning, most of our supportive services are presented in a form of suggestion or advice. Furthermore, learners are offered opportunities to choose a greater or lesser degree of support from the system. In our approach, different learners are provided with different service contents, by using different manners, and with different intervention degrees. Learning characteristics of learners are characterized by the following three dimensions: 1) learning history; 2) cognition level; and 3) learning styles. A profile is built for each individual learner who logs in for study and dynamically updated as the learning proceeds.

2.2 Overall architecture of the agent supported learning system

In order to support the autonomy of learners in learning and meanwhile provide supportive services to assist learners to solve various potential problems in *e*-learning, we propose an agent-based approach to providing supportive services to promote *e*-learning.

The overall architecture of the agent supported learning system is depicted in Figure 1. It utilizes electronic learning space technology to create an online learning environment conducive to constructivist learning [6], and incorporates software agents into the environment to provide supportive services to facilitate and assist individual learners to construct knowledge. Learners interact with the electronic learning spaces to manipulate the problem under study, construct their own knowledge representations, and communicate with other learners to share and exchange ideas and views. The agents work, independently of

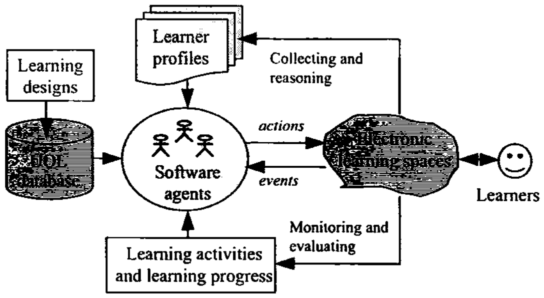

Figure 1. Overall system architecture

learners, observe and monitor the learning of individual learners and make suggestions and advice to assist them to construct meaningful understandings and develop skills relevant to high-order problem solving. They take *events* in the learning spaces as input and provide *actions* to the learning spaces as output. Any change taking place in a learning space made by an individual learner is represented as an event. Events can be detected by the agents. Learning behavior of individual learners is evaluated through evaluating the detected events. The learner profiles are built and timely updated through collecting the detected events and reasoning from them. The agents generate suggestions or advice for learners according to the practical learning scenarios and learner profiles. This is implemented based on a BDI based proactive reasoning [11] and with the help of the UOL database developed through careful learning designs.

2.3 Process agents

A software agent used in our work is as defined an active, persistent software component, situated in an online learning environment, that is capable of taking *autonomous* and *reasonable* actions to the change of the learning spaces and interacting with other agents to solve dependencies between them [11]. In the application presented in this paper, the agents are *process agents* that promote *e*-learning not through understanding the academic contents of the subject that learners are learning, but rather through providing a broad range of process-oriented services to assist learners to get through the process. They assist learners to get access to learning materials, provide learning plans for learners to achieve their learning goal, suggest discussion forums for learners to participate in the discussion about the theme under study, aid learners to timely evaluate learning outcomes and adjust learning whenever necessary. Process agents differ from the *pedagogical agents*, which may act as the *mentors* to students [2], the learning *partners* [3], or the *tutees*, and those agents which facilitate learning by supporting some particular functionality in a learning process, e.g. interface agents or navigational agents, etc. Pedagogical agents are only applicable for some specific circumstances because they are not designed for a generic learning process. In contrast with pedagogical agents, process agents focus on a generic learning process rather than on some particular roles in learning. They work independent of individual learners monitoring the change that happens in the learning spaces. They *autonomously* monitor and evaluate the learning behavior of individual learners and present them with a variety of assistance and scaffolding whenever they deem necessary. They guide learners to get through the learning process and are generic in this sense. They are domain independent and thus they do not need to understand the knowledge of a particular subject as pedagogical agents usually do.

3. UOL database

3.1 Role of the UOL database

The goal of the work presented in this paper is to utilize software agents to provide supportive services for individual learners to facilitate their knowledge construction. In order for the agents to provide services that meet individual learner's *just-in-time* needs or even *just-for-me* needs in the learning process, the agents need knowledge about the learner's learning styles, the learning activities being conducted and the corresponding services. A UOL database is used to supply the links between a learning scenario and the service to the learner with particular learning characteristics. It is composed of a collection of records each of which corresponds to a UOL. A UOL is an abstract term used to refer to any delimited piece of education and training [4]. A UOL describes *what*, *why* and *how* a unit can be studied for different learners and how the corresponding supportive services can be provided for different learners.

3.2 Architecture of the UOL database

The UOL database is composed of a collection of UOL records. A UOL record uses an architecture shown in Figure 2 to store the description of a UOL. The architecture is developed building upon earlier work of research in the descriptions of learning activities and processes, especially the Educational Modelling Language [7]. A UOL record includes seven composite fields, i.e. *metadata*, *roles*, *content*, *methods*, *assessments*, *case studies* and *learning plans*. Each of them contains more elementary fields, constructing a complex structure. The roles of these fields for implementing services to assist learners to build knowledge have been reported in [10].

3.3 Development of the UOL database

The development of the UOL database is to carefully design every UOL record. A UOL is a learning unit that satisfies one or more learning objectives. It can correspond to a course, a module, or even a single learning activity such as a discussion to elaborate on some ideas. As a learning goal can be divided into several sub-goals, a UOL may invoke several UOLs to respectively achieve the sub-goals. This way enables a hierarchical architecture of a module or a subject to be defined if required.

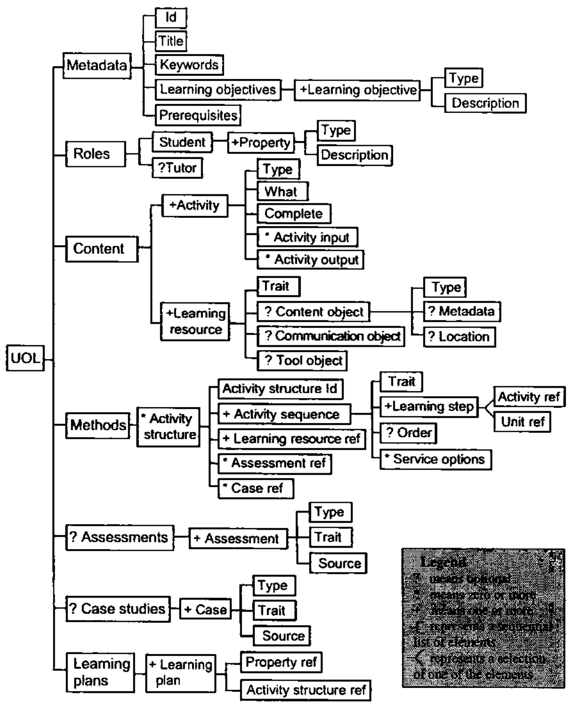

Figure 2. The architecture of a UOL

Designing a UOL should be based on constructivist learning theory [6]. Conceptually it includes the following processes:

- *designing the overall learning objectives* that are achieved by learners who complete the UOL.
- *designing how to achieve the goal.* Multiple learning routes for learners to reach the goal designed for the UOL are designed. Firstly, different learning routes are designed for different learners based on their cognitive levels, learning styles and so on. Secondly, *optional* learning routes are designed for a unique category of learners so that they can select an optimized one. Each learning route includes detailed learning steps respectively for a sub-goal and their sequences.
- *specifying required learning materials.* Learning materials that can be adopted by learners to achieve the goal designed for the UOL are specified. The learning materials can be in any forms but are mostly Web resources. Multiple *optional* learning materials are specified for individual learning requirements.
- *specifying related evaluation methods and case study materials.* The evaluation methods that can be used by learners to assess their learning for the UOL are specified, which may be a quiz or an interactive program. The case study materials that can be used to scaffold learners to accomplish the learning for the UOL are specified. Again for the purpose of supporting

individual knowledge construction, multiple *optional* evaluation methods and case study materials are specified.

3.4 Implementation of the UOL database

The requirement analyses to the UOL database indicate that there is no massive amount of data that needs to be inserted, updated or deleted on a daily basis. The major operations are related to searching for a particular UOL from the database and parsing the matched UOL. The structure of the database is not only very complex but also has no standard format as well. Clearly these features make it inadequate to use a relational database management system, such as Sybase or Oracle, to implement the UOL database. Instead, XML technology [12] is particularly suitable for the implementation of the UOL database.

The UOL database is implemented by an XML document. All the UOL records are represented in the document. The DOM and SAX approaches [12] are comprehensively applied to parsing the UOL database for the purpose of high efficiency and low cost. When searching for a particular UOL from the UOL database, the SAX approach is adopted to avoid the requirement for loading the whole UOL database into memory, which requires a huge memory resource. When parsing the content of a UOL record, the DOM approach is used to avoid the requirement for repeatedly loading the UOL record from the UOL database. In this way, not only the high speed of processing has been attained but also the excessive memory usage is avoided.

4. Implementation of the services

4.1 Services to support *e*-learning

The services we have identified to support *e*-learning are major concentrated on the ones that assist individual learners to get through the learning process [9]. They include: 1) providing access to appropriate learning materials and learning strategies that meet their learning requirements and match to their unique learning characteristics; 2) fostering meaningful interactions with content, teachers, and other learners; 3) supporting personalized learning for individual learners; 4) promoting collaborative learning among learners in groups; and 5) aiding to timely and accurately evaluate learning outcomes. All these services are implemented by software agents. Moreover, all these services are tailored to learners on individual basis. Examples include: 1) when a learner builds a learning plan for a desired goal, an agent will, based on his/her profile, advise him/her of several plans that suit him/her to achieve his/her goal; 2) when a learner executes a plan to reach his/her desired goal, an agent will offer him/her a wide range of supportive services for the relevant learning activities, including the advice on the choice of learning resources for the current theme, the related discussion forums, the assessment approaches, and the people to get further assistance, etc; and 3) when a learner is engaged in his/her learning activities, agents will monitor and evaluate his/her learning behaviors and advise him/her to align learning whenever necessary.

Different learners are provided with different supportive services, depending on their learning characteristics stored in profile. Moreover they are presented with the services using different modes and in different intervention degrees.

4.2 Implementation approaches

A prototype of the system has been developed and some of the supportive services have been successfully implemented. This section briefly describes the approaches to implementing the typical services.

4.2.1 Directing learners to get through a learning process

For some learners, the agents *actively* offer a *step-by-step* guide to help them get through a learning process. Figure 3 is an example where a learner, *Peter*, is forwarded to subject "*Database design*" and presented a suggestion in a popup window regarding what he should do next, when the agents detect he has set up a goal relevant to "*Database design*". Here the suggestion is generated based on his learning history and learning characteristics stored in his profile. The purpose for offering him such a suggestion is to guide him to perform the learning. Note that the suggestion content for other learners may be different, for example it may recommend learning "*Conceptual design*" or other topic, because it is generated based on the profile of the learner but the aim is the same: to guide the learner to effectively learn the current theme.

For the suggestion offered by the agents, *Peter* has the option to accept it or reject it. If he does not like it, he can ignore it by clicking on any other topic in the interface he would like to study. Alternatively, he can click on the "*More suggestions*" in the popup window to read those suggestions offered by the agent. If he does so, he will capture suggestions regarding suitable learning materials he can use for learning the current theme. He will also get advice on suitable tutor he can contact to attain assistance, suitable fellow learner he can study together, and suitable discussion forum he can enter to participate in the discussion about the current theme.

Supposing *Peter* would like to accept the suggestion, he clicks on "*Design techniques and models*" on the interface. He is then forwarded to an interface, shown in Figure 4, where the agents, again, present him a suggestion relevant to how to learn the current theme. Obviously such *step-by-step* guide can effectively assist *Peter* to get through a learning process. As it can be seen from the popup window, *Peter* can select "*Do not display the step-by-step suggestion again*" if he does not need such kind of assistance.

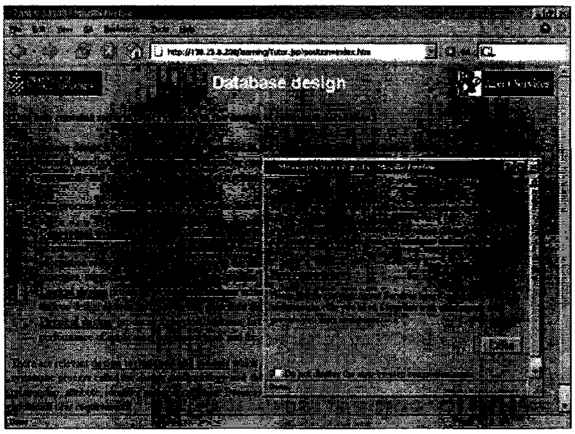

Figure 3. Step-by-step guide (1)

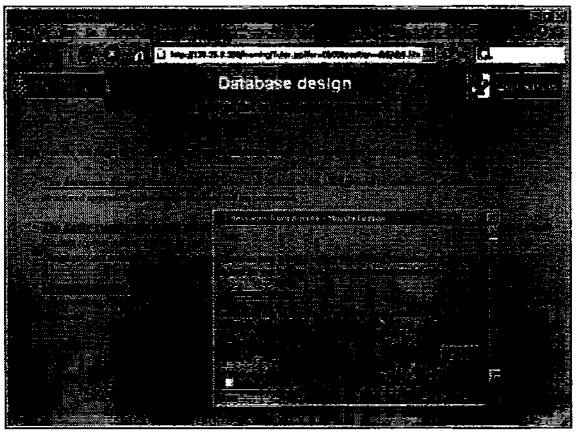

Figure 4. Step-by-step guide (2)

4.2.2 Assisting learners to develop personalized learning plans

The agents assist an individual learner to develop his/her personal preferred learning plan to reach his/her desired learning goal through advising him/her of several plans for the goal. These plans are the routes to achieve the learning goal, including the relevant learning activities to be taken and their conduct sequences. They are extracted from the UOL database by the agents based on the practical learning scenario and the specific learning characteristics of the learner stored in his or her profile. Figure 5 shows a typical scene where the agent for a learner has, based on his/her profile, generated three suitable plans for the learner to learn unit *database design*, and is presenting these three plans to him/her. The learner has the option to accept any one of them or reject them building his/her own plan. If the learner wants to accept a plan recommended by the agent, he/she only needs to click the "*Follow the plan*" in the plan line and the agent will then forward him/her to the due learning step defined in the plan. The learner can also look at the detailed learning steps of a plan by clicking the "*See the steps*" in the plan line. The popup window at the bottom of Figure 5 displays the detailed learning steps of "*The best suitable plan*" after the learner clicks on the "*See the steps*" in its plan line.

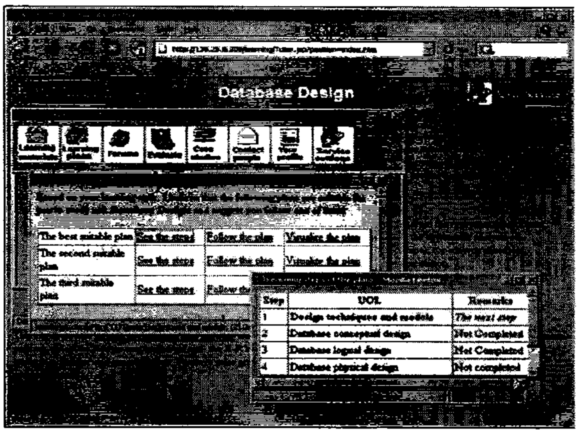

Figure 5. Assisting learners to develop learning plans

To help learner analyze a learning plan, the agent provides a visual presentation of the learning plan. After clicking "*Visualize the plan*" in a plan line, the learner will be presented with an interactively visual interface where a tree-structured graphical representation of the learning plan is displayed on the left panel of the interface and the detailed information of the focused step in the plan is concurrently displayed on the right panel. Through interacting with the interface, the learner can not only inspect the learning steps in the learning plan but also view their hierarchical architecture.

4.2.3 Guiding learners to dynamically align learning

Guiding individual learners to align learning towards their goals is realized through keeping tracks on individual learner's learning. The agents perform the work with the help of the two lists: *activity list* and *checklist*. Every learner is associated with those two lists. When a learner initiates the learning for a UOL, the unit, its learning goal (i.e. the goal unit), and the adopted learning plan for achieving the goal are together put into the *activity list*. Meanwhile all the learning tasks of the UOL are put into the *checklist*. When a learner has achieved the objective of a unit, namely has completed all its sub-tasks, the unit in the *checklist* will be set as *completed*. The *activity list* and *checklist* are dynamical updated and maintained by the agents based on the result attained through observing and monitoring individual learner's learning.

With these two lists, the agents are able to capture the information about what an individual learner is learning, what goal it is for, and which plan the learner is currently adopting for the goal. The agents are able to keep individual learner's learning on track because they can know what an individual learner has completed for a particular goal. Furthermore, they, by the comparison of these two lists, are able to find what an individual learner has done for his/her goal and what he or she needs to do next for achieving the goal. These make it possible for the agents to manage individual learner's learning and suggest learning adjustment when it is necessary.

Figure 6 shows a typical scene where the agent for *Peter* advises him to align learning because he wants to start learning a new unit but he has not completed all the units planned to learn *prior* to the one he is going to study. The process for the agent to recognize the requirement for the adjustment is described as follows. When the agent for *Peter* perceives *Peter* starts the learning for a new unit, it will autonomously check if it is appropriate to do so according to his learning history and the learning plan he is using for the goal. It thus first captures the learning plan that he is currently executing for the goal unit through a lookup table in the *activity list*. It then compares the *checklist* against the learning tasks scheduled in the plan. If all the units planned to learn *prior* to the one *Peter* is going to study have not been completed, the agent will suggest *Peter* adjust learning.

5. Conclusion

The paper presented an agent-based approach to supporting *e*-learning through providing learners with supportive services. All the supportive services tailored to learners on an individual basis. Most of them are presented to learners in a forma of suggestion or advice to encourage them to *active*ly learn. The paper described the overall system architecture, explored the development of the UOL database, and reported the preliminary implementation of the supportive services.

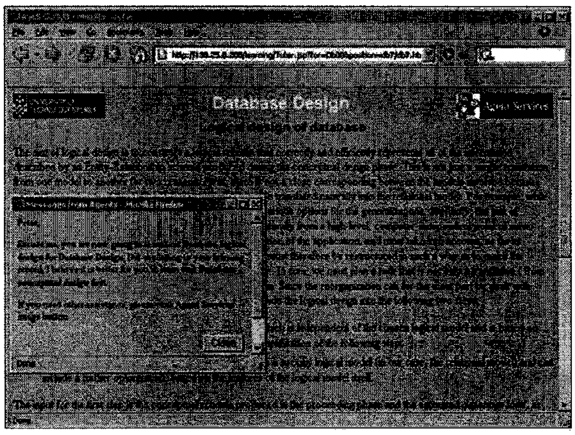

Figure 6. Assisting learners to align learning

References

[1] Anderson, J. R., Boyle, C. F., Corbett, A. T. & Lewis, M. W. (1990). Cognitive modelling and intelligent tutoring. *Artificial Intelligent*, 12, pp. 7-49.

[2] Baylor, A. L. (2004). Designing pedagogical agents to address diversity in learning. International Conference of the Learning Sciences, Los Angeles, California.

[3] Graesser, A.C., Lu, S., Jackson, G.T., Mitchell, H., Ventura, M., Olney, A., & Louwerse, M.M. (2004). AutoTutor: A tutor with dialogue in natural language. *Behavioral Research Methods, Instruments, and Computers*, 36, pp. 180-193.

[4] IMS. (2003). IMS Learning Design Specification V1.0.
Available at: http://www.imsglobal.org/learningdesign/ index.cfm (10/07/2004)

[5] Jonassen, D. H. (2000). *Computer as Mindtools for Schools: Engaging critical thinking*. Columbus, OH: Prentice-Hall.

[6] Jonassen, D. H. (1999). Constructivist Learning Environments on the Web: Engaging Students in Meaningful Learning. *Educational Technology Conference (EdTech 99)*. Singapore, Feb 9-11.

[7] Koper, R. (2001). Modeling units of study from a pedagogical perspective: the pedagogical model behind EML. http://eml.ou.nl (10/07/2004)

[8] Large, A. (1996). Hypertext instructional programs and learner control: A research review. *Education for Information*, 14, pp. 95-105.

[9] Pan, W. & Hawryszkiewycz, I. (2005). Facilitating Constructivist Learning through Managing Learning Processes. *IASTED International Conference on Computer and Advanced Technology for Education*, Oranjestad, Aruba, August 29-31. pp. 197-202.

[10] Pan, W. & Hawryszkiewycz, I. (2004). A method of defining learning processes. In R. Atkinson, C. McBeath, D. Jonas-Dwyer & R. Phillips (Ed.), *Beyond the comfort zone: Proceedings of the 21st ASCILITE Conference*. Perth, Australia, Dec 5-8. pp. 734-742.

[11] Wooldridge, M. (2002). *An Introduction to Multiagent Systems*. John Wiley & Sons, Ltd, England.

[12] World Wide Web Consortium, Extensible Markup Language (XML). Available at:http://www.w3.org/XML/ (10/10/2005)