LogForum

ISSN 1734-459X
2007
Vol. 3
Issue 3
No 1

> Elektroniczne czasopismo naukowe z dziedziny logistyki <

http://www.logforum.net

A CONCEPTUAL DISTANCE LEARNING ARCHITECTURE USING SEMANTIC WEB BASED MULTI-AGENT SYSTEMS

Ata Onal[1], Semih Otles[2], Inanc Seylan[1]

1) Department of Computer Engineering, Ege University, Izmir, Turkey
2) Department of Food Engineering, Ege University, Izmir, Turkey

ABSTRACT. In this paper, a conceptual distance learning architecture based on a semantic web enabled multi-agent platform is introduced. The need for such architecture arises to automate every day management tasks of a distance education system. Agents take care of such automation by using a common vocabulary of terms defined in ontologies. The users of the system are each given an agent to act on their behalf. These agents are then used to query, negotiate and cooperate with other agents to organize tasks and inform their users.

Key words: distance learning architecture, education system, ontology.

INTRODUCTION

Distance education is defined as a teaching form where the teachers and the students are in different locations [Silva]. In the beginning, it made use of postal service. Then new technologies such as radio, TV and computers were incorporated into it. Later, with the wide use of Internet, distance education environments are moved to the WWW. Many tools have been created such as WebCT [WebCT] to allow the content to be created, published and managed. However, the interaction between students and instructors doesn't go further than chat rooms, course materials or submitted assignments. Both users of the web based distance education environments require lots of human intervention. Tracking student progress requires manual methods to collect information and interpret that information.

Distance education is big business [Martz and Morgan]. Putting aside the economical facts, the amount of information that is needed to be managed in a distance learning environment is huge. It requires modern information management techniques which just don't store data but are able to fully categorize it, define relationships and rules; moreover, infer new knowledge. Therefore, distance learning provides opportunities that take advantage of ontologies. By ontology, we specifically mean ontologies written in W3C standard web ontology language, OWL [Web Ontology Language].

In the semantic web vision, the information published is both machine processable and human readable [Berners-Lee et al. 2001]. The computer programs that are agreed upon to process this information are software agents. Although software agents have a long history in the field of AI, research on them has increased in the last ten years. Despite its history, there isn't a universally-agreed definition of the term software agent. Different researchers describe it according to the key principals

Citation: Onal A. et al., 2007, A conceptual distance learning architecture using semantic web based multi-agent systems. LogForum 3, 3, 1
URL: http://www.logforum.net/vol3/issue3/no1
Accepted: 10.08.2007, on-line: 1.09.2007.

of their systems. [Woolridge and Jennings 1995] discusses that the defining characteristics of agency that are agreed upon by many researchers are: proactivity, reactivity, social ability and autonomy. Autonomous assistant agents representing their owners can help in management of a distance education environment, interact with other agents in the environment, thus making use of the services provided by a multi-agent system and automate everyday tasks of their owners.

There are some efforts to use software agents in distance learning. [Sheung at el 2005] uses autonomous interface agents to assist course coordinators in routine tasks. This architecture is different from ours because we model a multi-agent system in which each actor (student, academician) is represented by a software agent which is capable of communicating with other agents. We also make use of semantic web technologies. [Keleberda et al. 2004] focuses on the educational material selection based on student profiles, thus focusing on a single aspect of the proposed architecture. OntoEdu [Guang-zuo 2004] is a flexible educational architecture for e-learning with support for device adaptation. However it doesn't make use of the agent technology and task automation that comes with it.

This paper proposes a conceptual architecture for distance education that makes use of a multi-agent platform. The rest of this paper is organized as follows. Section 2 overviews the Seagent Platform which form the basis of the proposed distance learning environment. Section 3 discusses the architecture of this distance learning environment. Section 4 concludes the work and gives the direction of future research.

THE SEAGENT PLATFORM

SEAGENT is a new agent development framework and platform that is specialized for semantic web based multi agent system development. The communication and plan execution infrastructure of SEAGENT looks like other existing agent development frameworks such as DECAF [Graham et al 2003], JADE [Bellifemine et al. 2001], RETSINA [Sycara et al. 2003]. To support and ease semantic web based multi agent system development, SEAGENT includes the following built-in features that the existing agent frameworks and platforms do not have:

- Agents created using SEAGENT handle their internal knowledge base using semantic web standards and the platform provides specifically designed interfaces to manage and query the internal knowledge without being dependent on a particular application programming interface.
- The directory service of SEAGENT is implemented in a way that the directory knowledge is held in semantic web standards and the directory service supports semantic matching of the agent capabilities to find the semantically similar agents.
- FIPA-RDF content language [FIPA] has been used to transfer semantic content in the agent communication language messages and OWL-QL [Fikes at al. 2003] is integrated to the FIPA-RDF content language to query the agents and services.
- SEAGENT introduces a new service for managing and translating ontologies. It provides a means to define mappings between platform ontologies and external ontologies. The translation process is based on these defined mappings.
- SEAGENT supports discovery and dynamic invocation of semantic web services by introducing a new platform service for semantic service discovery and a reusable agent behavior for dynamic invocation of the discovered services.

The following section discusses the overall architecture of SEAGENT.

THE ARCHITECTURE

In this section, we discuss SEAGENT's layered software architecture. Each layer and packages of the layers have been specially designed to provide build-in support for semantic web based multi agent system (MAS) development. The overall architecture is shown in Figure 1. Although the introduced architecture is the implemented architecture of the SEAGENT platform, we believe that it is generic enough to be considered as a conceptual architecture of semantic web based multi agent systems. In the following, we briefly discuss each layer with an emphasis on the semantic support given by that layer.

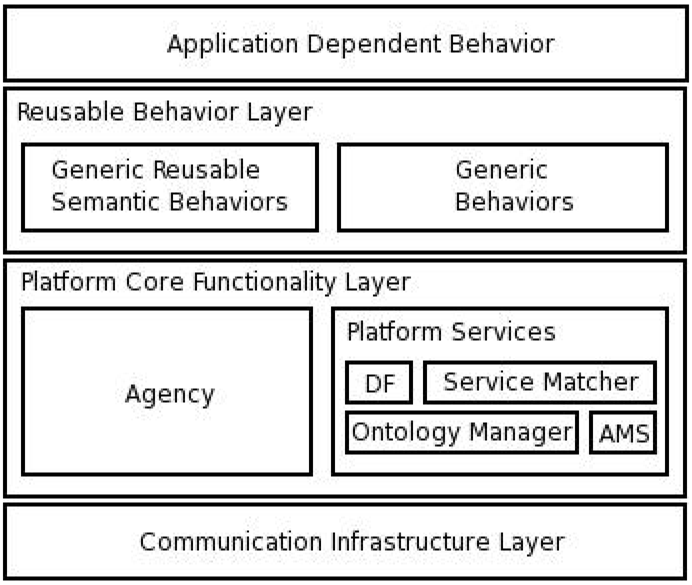

Fig. 1. Seagent Platform Overall Architecture
Rys. 1. Architerktura platformy Seagent

The bottom layer is responsible of abstracting platform's communication infrastructure implementation. SEAGENT implements FIPA's Agent Communication and Agent Message Transport specifications [FIPA] to handle agent messaging. Although Communication Infrastructure Layer can transfer any content using FIPA ACL and transport infrastructure, SEAGENT platform only supports FIPA RDF content language since it is very suitable to transfer semantic web enabled content.

The second layer includes packages, which provide the core functionality of the platform. The first package, called as Agency, handles the internal functionality of an agent. Agency package supports the creation of general purpose and goal directed agents. In this sense, Agency package provides a built-in 'agent operating system' that matches the goal(s) to defined plan(s), which are defined using HTN planning formalism [Paolucci et al. 2000]. It then schedules, executes and monitors the plan(s). From semantic web based development perspective, an agent's internal architecture must support semantic web ontology standards for messaging and internal knowledge handling to simplify semantic based development. For this purpose, Agency package provides a build-in support to parse and interpret FIPA RDF content language to handle semantic web based messaging. On the other hand, Agency provides two interfaces for semantic knowledge handling, one for local ontology management and the other one for querying. Although the current version includes the JENA [JENA] based implementation of these interfaces, other semantic knowledge management environments and query engines can be integrated to the platform by implementing these interfaces.

The second package of the Core Functionality Layer includes service sub-packages, one for each service of the platform. SEAGENT provides all standard MAS services such as Directory Facilitator

(DF) Service and Agent Management Service (AMS) following the previous platform implementations and FIPA standards. But these standard services are implemented differently using the capabilities of a semantic web infrastructure. For example, standard functionality of the DF is to store agent capabilities and return the matched agent(s) upon a capability-matching request. In SEAGENT implementation, DF uses an OWL [Web Ontology Language] ontology to hold agent capabilities and includes a semantic matching engine to be able to return agent(s) with semantically similar capabilities to the requested ones. Similarly, AMS stores agents' descriptions in OWL using FIPA Agent Management Ontology [FIPA] and can be queried semantically to learn descriptions of any agent that is currently resident on the platform.

Besides implementing standard services in a semantic way, SEAGENT platform provides two new services to simplify semantic web based MAS development. The first one is called as Semantic Service Matcher (SSM). SSM is responsible for connecting the platform to the semantic web services hosted in the outside of the platform. SSM uses 'service profile' construct of the Web Ontology Language for Semantic Web Services (OWL-S) standard for service advertisement and this knowledge is also used by the internal semantic matching engine for discovery of the service(s) upon a request. SSM and DF services are implemented by extending a generic semantic matching engine architecture. The second unique service is the Ontology Manager Service (OMS). It behaves as a central repository for the domain ontologies used within the platform and provides basic ontology management functionality such as ontology deployment, ontology updating, querying etc. The most critical support of the OMS is its translation support between the ontologies. OMS handles the translation request(s) using the pre-defined mapping knowledge which is introduced through a specific user interface. Through the usage of the ontology translation support, any agent of the platform may communicate with MAS and/or services outside the platform even if they use different ontologies.

Third layer of the overall architecture includes pre-prepared generic agent plans. We have divided these generic plans into two packages. Generic Behavior package collects domain independent reusable behaviors that may be used by any MAS such as well know auction protocols (English, Dutch etc.). On the other hand, Generic Semantic Behaviors package includes only the semantic web related behaviors. In the current version, the most important generic semantic behavior is the one that executes dynamic discovery and invocation of the external services. This plan is defined as a pre-prepared HTN structure and during its execution, it uses SSM service to discover the desired service and then using OWL-S 'service grounding' construct it dynamically invokes the found atomic web service(s). Hence, developers may include dynamic external service discovery and invocation capability to their plan(s) by simply inserting this reusable behavior as an ordinary complex task to their HTN based plan definition(s).

THE DISTANCE LEARNING ENVIRONMENT

ACTORS

The human actors of the system are considered to be students, academic staff, and system administrators. Every human actor is represented by a software agent in the system that is also of the same role as the human actor. Assigning roles to agents implicitly ensures that actors don't attempt of doing something out of their privileges.

It is a choice of implementation where these agents live. They may reside on local machines of the users or on system servers which users interact through a web interface. However, for performance considerations, it is better that they follow the more distributed approach of working on local machines. If this is the case, there must be mechanisms to reflect the updates of agent plans to the already working agents. After logging into the system, all agents may ask for updated versions of their plans to a dedicated agent. The goal of this agent is to manage the plans by keeping the versions and the roles of agents which can use them in an OWL ontology. The updates can be further categorized into being mandatory or optional so that users don't have to wait for an optional update to be finished to use their agent. It would be better that these decisions given by the actor were recorded and followed in the same scenarios.

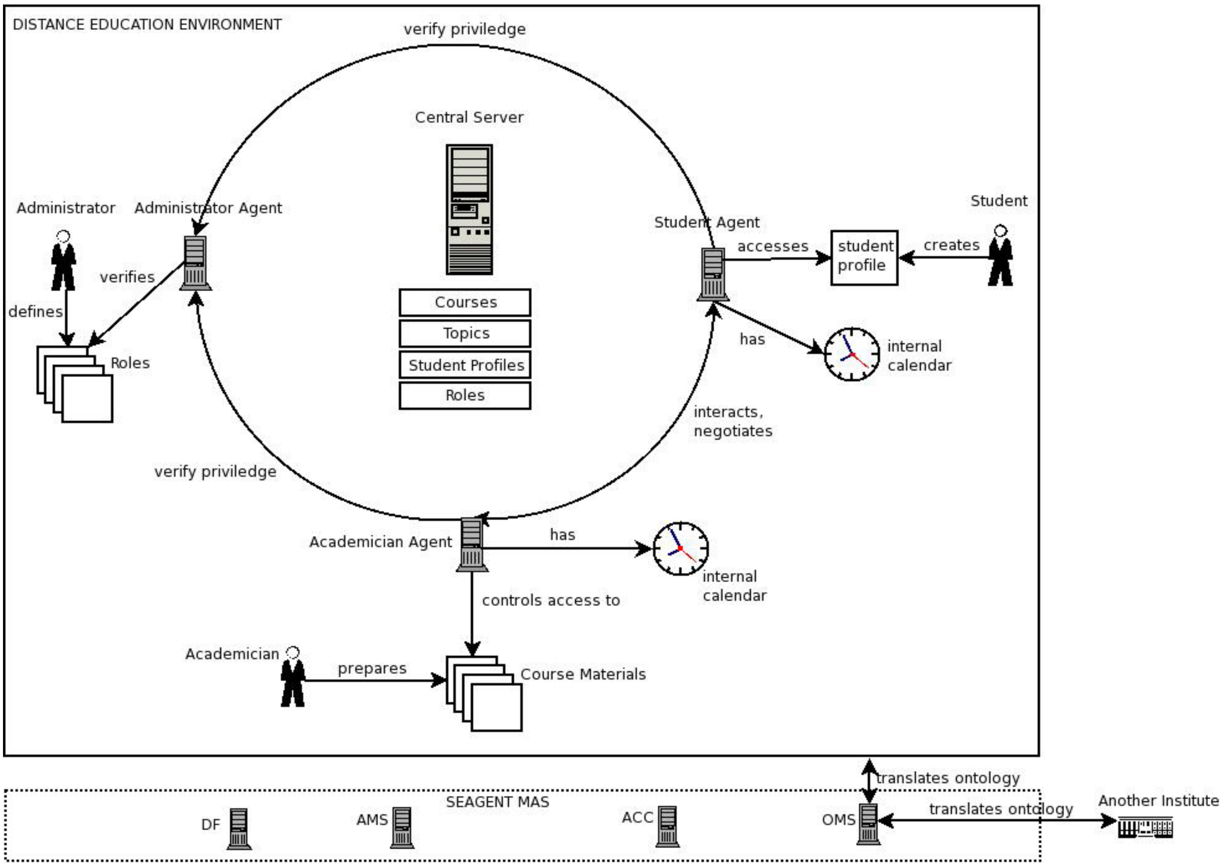

Fig. 2. The actors of the system
Rys. 2. Aktorzy systemu

It is important that the agents either have access to a calendar of the actor or they provide the calendar themselves. The latter is preferred because the events can then be customized for the education domain such as assignment due, examination. It is the responsibility of the users to provide their agents with a detailed schedule.

One of the promises offered by distance education is less rigid time constraints. Indeed, it is the reason many working individuals prefer distance education. So the system must take this to its advantage. Instead of the instructor just announcing a date for an exam, he can also announce a time duration that the exam can be taken. When the student agents connect with the system, and see the exam event, they can first check their schedule. In case of schedule overlaps, they can try to negotiate with other students agents, and even the instructor agent to decide on a free date for all them.

Inactive students (those who have not submitted their assignment, or did not view the course material for a specified time limit) can be detected by the agent of the instructor of that course and the student agent is then sent a warning if it's currently online. If not, the student himself is sent an e-mail. Students can also be reminded about approaching due dates. Letting the academician agent take care of such monitoring tasks gives more time to the instructor in preparing better course material. [Sheung at al. 2005] has implemented a similar monitoring system and it got positive feedback from the students because they thought that their teacher was enthusiastic about their studies and aware of their learning progress and performance in the course. However, the students were not aware that these mails were sent by a software agent.

ONTOLOGY KNOWLEDGE BASE

It is essential to define the terms in our domain of interest, distance learning, in ontologies. They must be categorized. But an ontology is more than just a pure classification of things. Such a classification is called a taxonomy. Ontologies also offer the power to express the relationships that can exist among things.

In the system, ontologies are used to store student profiles and course related information. Courses have relationships with other courses such that a course could be prerequisite of another one. It is likely that these courses share an amount of same teaching material. Even courses at different levels or grades might need to use materials that are already published.

Initially, the whole list of topics that the online courses cover must be defined in ontologies. For example, the term "multi-agent systems" is a field of "AI", but there are also applications of it in "Information Management". Thus, it can be declared as the sub class of "AI" and can link to "Information Management" by its "related Field" property. It is a difficult and detailed job to define the topics this way but it has many advantages. First of all, the instructor just needs to select the topics his course covers from the topics ontology and the relationships among the chosen topics are implicitly there. In addition, it helps to form a question-base where test questions prepared by the academic staff are kept. A test can be prepared anytime by providing a query on wanted topics, the ratio of hard/easy questions, etc. to the test preparation agent.

Furthermore, ontologies can help in exchange of knowledge between different institutions based on a common vocabulary. In such a case, the OMS (Ontology Management Service) of the SEAGENT Platform could be used to translate the concepts from one ontology to the other, hence providing university collaboration and joint programs.

Students are the consumers of the course material. At any time, they are taking courses and they have a list of token ones with passing or failing grades. According to the student's history, which is kept in his profile, he can be suggested directions that are tested to lead to success.

CONCLUSIONS AND FUTURE WORK

The proposed architecture is mainly based on our experience with agent and semantic web technologies. We are looking for domains that makes use of the expressiveness of ontologies and distributed, social, autonomic nature of software agents to test our agent platform. Therefore, we see distance learning as an exciting application area because of the opportunities explained in this paper. Our future work will include incrementally implementing this environment, testing it on samples of students from different departments and get their feedback. We also want to integrate the strategies defined in [Olt 2002] to online assessment coordinated by a software agent.

REFERENCES

Bellifemine, F., Poggi, A., and Rimassa, G., 2001, Developing Multi-agent Systems with a FIPA-compliant Agent Framework, Software Practice and Experience, 31 103-128.

Berners-Lee, T., Hendler, J., Lassila, O. 2001, The Semantic Web, Scientific American.

FIPA, 'FIPA Specifications', http://www.fipa.org

Fikes, R., Hayes, P. and Horrocks, I., 2003, OWL-QL - A Language for Deductive Query Answering on the Semantic Web, Knowledge System Laboratory, Stanford University, available at http://ksl-web.stanford.edu/KSL-Abstracts/KSL-03-14.html.

Guang-zuo CUI et al., 2004, OntoEdu: Ontology based Education Grid System for e-learning, GCCCE2004 International conference, Hong Kong.

Graham J. R., Decker, K. S. and Mersic, M., 2003, DECAF - A Flexible Multi Agent System Infrastructure, Journal of Autonomous Agents and Multi-Agent Systems, 7, 7-27.

Paolucci, M. et al., 2000, A Planning Component for RETSINA Agents, Intelligent Agents VI, LNAI 1757, N. R. Jennings and Y. Lesperance, eds., Springer Verlag.

JENA, A Semantic Web Framework for Java, http://jena.sourceforge.net/.

Keleberda, I. N., Lesna, N. S., Makovetskiy S. D., Terziyan V., 2004, Personalized Distance Learning Based on Multiagent Ontological System, ICALT.

Olt, M. R., 2002, Ethics and Distance Education: Strategies for Minimizing Academic Dishonesty in Online Assessment, Online Journal of Distance Learning Administration Vol. 5 Issue 3.

Martz, B., Morgan, M. S. Assessing the Impact of Internet Testing: Lower Perceived Performance. Distance Learning and University Effectiveness. pp. 177 - 189.

Sheung-On Choy, Sin-Chun Ng, Yiu-Chung Tsang, 2005, Software Agents to Assist in Distance Learning Environments, Educause Quarterly, Vol. 28 No. 2.

Silva, E., Moreira, D., Use of Software Agents for the Management of Distance Education Courses, http://citeseer.ist.psu.edu/silva00use.html

Sycara, K., Paolucci, M., Van Velsen, M. and Giampapa, J., 2003, The RETSINA MAS Infrastructure, Journal of Autonomous Agents and Multi-Agent Systems, 7, 29-48.

WebCT, http://www.webct.com

Web Ontology Language (OWL), http://www.w3.org/2001/sw/WebOnt/

Woolridge M., Jennings, N., 1995, Intelligent Agents: Theory and Practice. Knowledge Engineering Review. Vol. 10:2., pp. 115-152.

ARCHITEKTURA SYSTEMU EDUKACJI ZDALNEJ PRZY WYKORZYSTANIU SYSTEMÓW OPARTYCH NA WIELOAGENTOWYCH SIECIACH

STRESZCZENIE. W pracy przedstawiono zagadnienie architektury systemu edukacji zdalnej opartej na wieloagentowej sieci. Zapotrzebowanie na taki system wynika z potrzeby automatyzacji codziennej pracy w systemie edukacji zdalnej. Agent zajmuje się automatyzacją tej pracy, stosując powszechne słownictwo zdefiniowane w bazach. Każdy użytkownik ma swojego własnego agenta. Agenci tacy współpracują z innymi agentami w celu organizacji zadań i przekazywania informacji swoim użytkownikom.

Słowa kluczowe: edukacja zdalna, system edukacyjny, ontologia.

EIN KONZEPT ZUR FERNSTUDIENARCHITEKTUR UNTER NUTZUNG SEMANTISCHER WEBBASIERTER MULTI-AGENT-SYSTEME

ZUSAMMENFASSUNG. In diesem Artikel wird ein Konzept zur Fernstudienarchitektur unter Nutzung semantischer webbasierter Multi-Agent-Systeme vorgestellt. Der Bedarf für solch ein System entsteht durch die tägliche Notwendigkeit, Managementaufgaben des Fernstudiums zu automatisieren. Die Agenten erledigen solch eine Automatisierung unter Nutzung eines allgemein verbreiteten Vokabulars von definierten Begriffen in Ontologien. Jedem Nutzer des Systems wird ein Agent gegeben, der in ihrem eigenen Namen handelt. Diese Agenten werden dann gebraucht, um andere Agenten abzufragen, sich mit ihnen auszutauschen und zu kooperieren, um Aufgaben zu organisieren und ihre Nutzer zu informieren.

Codewörter: Fernstudienarchitektur, Ausbildungssystem, Ontologie.

Semih Ötleş,
Department of Food Engineering
Ege University
Bornova 35100 Izmir, Turkey
e-mail: semih.otles@ege.edu.tr