THE ROLE OF ONTOLOGIES IN AGENT-BASED SIMULATION OF INTELLIGENT TUTORING SYSTEMS

Vita Graudina
Janis Grundspenkis
Department of Systems Theory and Design
Riga Technical University
1 Kalku str., LV-1658, Latvia
E-mail: {vita, jgrun}@cs.rtu.lv

KEYWORDS

Ontologies, Agent-Based Simulation, Intelligent Tutoring Systems, Multi-Agent Systems.

ABSTRACT

The paper presents a novel architecture of agent-based simulation of teaching and learning process. We propose a conceptual architecture of such system. There are presented several aspects, how the usage of ontologies for the control of students progress can improve the efficiency of intelligent tutoring systems. Main attention is paid to the usage of ontologies for agent communication and formal description of learning content and process.

INTRODUCTION

The paper presents ongoing research on the role of ontologies in the development of the intelligent tutoring systems. The intelligent tutoring system should be considered as a testbed for agent-based teaching and learning process simulation. Technology based learning stands for all forms of computer supported learning, including distance, on-line, virtual learning, E-learning, etc., (Anohina 2003). A lot of work has been done to improve tutoring process based on E-learning systems (Paulsen 2003; Race 1994), Web-based learning environments (Groeneboer and Stockley 1997), and WebCT (available at http://www.webct.com/), etc.

Among wide varieties of applications of technology based learning, one of the most interesting and promising is the development of intelligent tutoring systems. Usually the research of these systems is focused on the development of such modules as Expert Model, Student Model, Course Management System, (Capuano et al. 2002) that are, as a rule, implemented in an intelligent tutoring system's architecture by using multi-agent based architecture (Capuano et al. 2000; Sklar et al. 2004; Dorca et al. 2003; Garro and Palopoli 2002).

Our search for the literature in the field reveals that the main activities in simulation of the teaching and learning process are done mainly in two directions. First, simulation of tutor and student activities is considered. So called pedagogical agents (see, for example, Johnson 2003; Dorca et al. 2003) simulate tutor, while the student model is used for simulation of students actions (Greer and McCalla 1994). Second, there are attempts to simulate policies of education process (Sklar et al 2004).

To make experiments with real systems, it is necessary to develop a simulator of the system. So, our final goal is the implementation of a simulation system that will enable tutors to pose "what if" questions about the effects of their decisions. Experiments are needed to test different teaching methods, techniques and pedagogical approaches, different course material representation techniques and different sequences of course contents. Multi-agent architecture allows to emulate a human tutor and a student group, i.e., the interaction between agents in the hierarchically organized community, where each agent can only take a role of a tutor or a role of a student, i.e. the roles are mutually exclusive. It is the case when simulation of interactions should be used to evaluate the efficiency of delivering course materials and different ways of their visualisation. Our search for information sources on the usage of ontologies in intelligent tutoring systems shows that their role is underestimated. There are only few papers focused on the usage of ontologies in this field, and mainly they are about management of course materials and learning objects (Brace and Nejdl 2004; Baumann et al. 2002; Tane et al. 2003; Garro and Palopoli 2002).

Ontologies play major role in agent interaction by providing shared representation of the domain and the concepts that agents need to use (Sycara and Paolucci 2004; Esteva et al. 2002). The approach discussed in this paper is based on the assumption that in result of interactions during teaching and learning process agents should reach a common shared ontology. This is not the case at the starting point of tutoring when each student has his/her own domain ontology, which may pretty much differ from the tutor's ontology.

Usually ontologies are used to describe some teaching components of intelligent tutoring systems, like learning objects of the course (Capuano et al. 2002; Garro and Palopoli 2002), theories about learning (Meisel et al. 2003), management of personalized information (Weissenberg et al. 2004; Garro and Palopoli 2002).

The paper is an attempt to highlight the role of ontologies in the intelligent tutoring systems. We propose architecture of the system where course content

Proceedings 19th European Conference on Modelling and Simulation
Yuri Merkuryev, Richard Zobel, Eugène Kerckhoffs © ECMS, 2005
ISBN 1-84233-112-4 (Set) / ISBN 1-84233-113-2 (CD)

as well as interactions between agents are described by ontologies.

It is needed to point out that at the present phase of the research we take a rather simple approach where student's progress is related only with step by step construction of his/her ontology. The ontology represents the student's knowledge base, which includes the particular knowledge domain being learned. At the same time we agree with other researchers that cognitive (internal) factors such as motivation, emotion, and, ability to learn as well as external factors (such as tutor's personality, relationships with class mates, training appliances, environment, etc.) jointly affect teaching and learning process (Sklar et al. 2004). We suppose to include more factors in the simulation system in future.

ARCHITECTURE

Agents play a dual role in simulation of teaching and learning process: on the one hand, they are goal-directed autonomous problem solvers; on the other hand, they have a social dimension because they interoperate as a part of a multi-agent system (MAS) (Weiss 1999; Wooldridge 2002). Ontologies describe the type of entities that agents encounter, their properties, and the relations between them. Agents in a MAS necessarily interact with other agents. Ontologies provide the basic representation to the agents that allows them to reason about interactions needed for the solution of the problem. Moreover, ontologies provide agents with shared knowledge that they can use to communicate and work together (Sycara and Paolucci 2004). This paper focuses on the modelling of ontologies that help agents in their interaction in the intelligent tutoring systems.

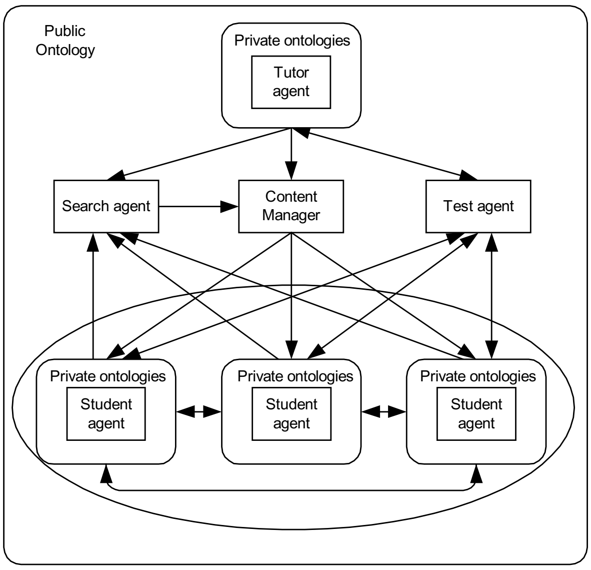

Figure 1: Architecture of the proposed system

The proposed model is a MAS, where agents interact to achieve the goal – perform students tutoring (Student agents' Domain ontology starts to align to one that is intended by Tutor agent). Agents of different types constitute the system (see Figure 1), while ontologies of different types support interactions between agents.

Agents

Different types of agents are used in the proposed MAS. First, Student agents, second, tutoring support agents (Search agent, Content Manager and Test agent), and third, the Tutor agent (see Figure 1).

Figure 2 shows all information and data flows between agents, which could be considered and called as agent perceptions. All flows between students' community and other agents are also possible for each student agent individually not only for the whole community.

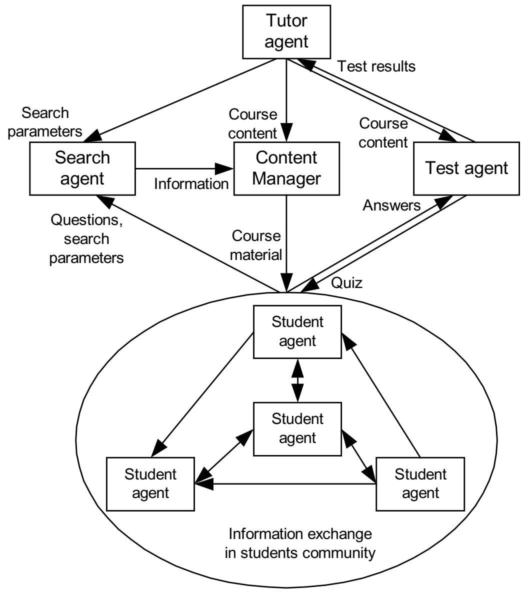

Figure 2: Perceptions of Agents

Student agents and the **Tutor agent** simulate the corresponding human beings that are considered to be knowledge workers, whose activities are supported by different agents surrounding them (Grundspenkis 2003). These agents usually are filtering agents, search agents, recommender agents and other personal agents. In our case we consider the Tutor agent and Student agents as knowledge workers, but tutoring support agents as personal agents. All agents have their own ontologies and can use shared ones. Student agents constitute a student community. During communication process Student agents change and update each other's Domain ontologies. Student agents can send some search parameters to Search agent, if they need some additional information about delivered course material.

The Tutor agent is used to simulate the tutor's work. This agent supports a tutoring process in proposed system. The Tutor agent coordinates interactions between other tutoring support agents to reach the tutoring goals by providing information about course contents, pedagogical approaches, etc.

The **Search agent** is needed to support Student agents with appropriate course material. According to given search parameters of Tutor agent and Student agents,

the Search agent derives necessary information to Student agents. In principle, to provide better search results the Search agent could interact with other search agents (like, Web crawlers - also known as Web spiders, robots, or wanderers (Chau and Chen 2003)) or database agents, but it needs more research on it.
The **Content Manager** supports representation of information found by the Search agent in the form, which is understandable and easy to use. The deliverable course material depends on course contents, tutor's teaching style and student's learning style. Only after a course material has been corrected, it is delivered to the Student agent.
The **Test agent** compares Student agent's Domain ontology with Tutor agent's Domain ontology, and returns results of comparison to the Tutor agent. Student agents receive a course material from the Content Manager, tests and quizzes from the Test agent, and give test answers to the Test agent.

Ontologies

It is worth to underline that there are inherent difficulties encountered in implementing coordinated behaviour in any MAS communication, interaction, coherence and coordination (Capuano et al. 2000). These difficulties may be overcome using ontologies. Ontologies describe communication protocols, provide ways how agents can interact with each other and help agents to find solutions and to achieve their goals. According to dual agents' behaviour (Sycara and Paolucci 2004) ontologies of two types are needed. Private ontologies support the individual problem solving purposes of agents while Public ontologies support social interactions of agents.
Following (Guarino 1998), there are other subtypes of ontologies. Domain ontologies, Task ontologies and Application ontologies are Private ontologies, but Top-level ontologies are Public ontologies. Figure 3 shows the hierarchy of proposed ontologies. A Task ontology and a Domain ontology specify concepts from the Top-level ontology, and an Application ontology specifies the Domain ontology and the Task ontology.

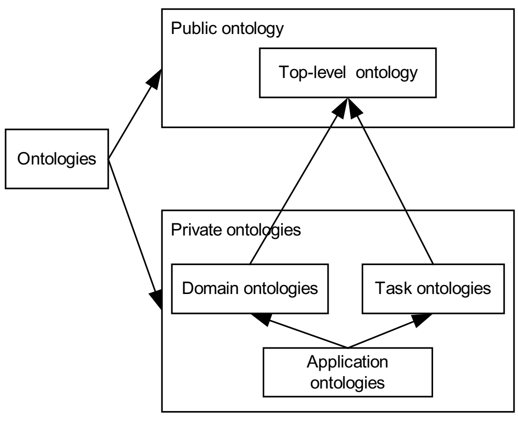

Figure 3: Hierarchy of Ontologies

A **Public ontology** is necessary to support social interactions between agents in the MAS. It supports communication and information exchange. It also provides description of the infrastructure of the whole MAS and involved agents. Involved agents are described in terms of their capabilities, interaction protocol, etc. Public ontology is shared between agents in the MAS. A **Top-level ontology** describes general terms used overall the problem domain, such as time, space, etc. It is possible to use some already created top-level ontologies, like WordNet (Miller 1995), Cyc (Lenat 1995), etc. Top-level ontology as a pubic ontology is also shared between all agents of the system.
Private ontologies are created for each agent individually, depending on the type of an agent, its purpose, goals, acting domain, actions etc. **Domain ontologies** and **Task ontologies** describe vocabulary that is connected with agents' domain (like, education, pedagogical approaches) or task, or actions (like, searching, planning or learning). These ontologies specify concepts from the Top-level ontology. Some agents in our proposal are domain independent, so they need only Task ontologies. For example, for a Content Manager we specify only a Task ontology, because this agent is domain independent and there are no needs to create a special Domain ontology. **Application ontologies** describe concepts depending on both a particular domain and a task (like, order of delivering course material or information about which lessons should be learned before particular one). The Application ontology also defines commands, parameter names and interpretation of them needed to control the system from outside. The Application ontology provides possibility to human intervention in agents' behaviour to make experiments. In some cases it isn't needed to use Application ontologies, it is enough with Domain and Task ontologies, like in case with a Test agent. This agent has only a Task ontology, because it provides concepts about different kinds of quizzes as well as concepts about process of testing.
A **Student agent's Domain ontology** describes the current state of Student agent's knowledge about the domain. This ontology is updated during communication between Student agents or after they have learned a course material provided by the Content Manager. For the Tutor agent's Domain ontology it is possible to use some already created ontologies (ACM Computer Classification System available on http://www.acm.org/class/1998 can be used for courses in Computer Science or bodies of knowledge, for example, the Software engineering body of knowledge available on http://www.swebok.org).
Initially Student agents' Domain ontologies have some concepts associated with the course, but at the beginning of learning they may be different for each Student agent. An initial Student agent's Domain ontology could be empty or may present an initial ontology that is obtained by experience or intuitionally. An empty initial ontology simulates those students who have no prerequisite knowledge about particular lecture

course. An empty ontology, as a rule, is for beginners' level courses, as we suppose that students don't have a prior knowledge.

An initial Domain ontology that includes only some concepts simulates students who have already some knowledge about the course, for example, if the offered course is "Advanced Artificial Intelligence", it is more possible that Student agents' Domain ontologies already have concepts from "Fundamentals of Artificial Intelligence" ontology. An initial ontology with some concepts is obtained from real experiments (tests) with real students.

Student agent's Domain ontology is updated after delivering of each portion of course material. The Test agent compares Student agents' Domain ontologies with Domain ontology provided by the Tutor agent. Depending on results of testing the next portion of the course material is delivered.

Candidate concepts are dealt out from the delivered course material by using ontology learning methods and approaches (Maedche and Staab 2004). Candidate concepts are concepts, which could be included in the Student agent's Domain ontology in case of successful learning of delivered material. A Test agent is used to find candidate concepts. By using quizzes it is tested has student learned candidate concepts or not. If he/she has, then candidate concepts are included in the Student agent's Domain ontology in accordance with following rule: $Ontology^{old} \cup Learned\_concepts \rightarrow Ontology^{new}$.

The question is still open how to simulate the outcome of student communication that may result in learning new concepts. If it is the case, we somehow need to update the Student agents' Domain ontologies. At the moment it is unclear what kind of agents and methods can be used.

A **Student agent's Task ontology** is used to describe concepts corresponding to a learning process as well as generic concepts about learning styles. A Student agent's Task ontology is a basis for further simulation of students learning styles and preferences.

A **Student agent's Application ontology** can be considered as a complete Student model, because it contains student knowledge level as well as preferred learning style. It is used for the same purpose as a student model in the intelligent tutoring systems (Capuano et al. 2000). A Student agent's Application ontology stores and describes characteristics, preferences, learning style and problem-solving behaviour of the particular student. Concepts of the Student agent's Application ontology are updated after Student agent's action. The Student agent's Application ontology influences the Content Manager agent with concepts about particular student's preferences and learning style.

A **Tutor agent's Domain ontology** contains ontology of the offered course, i.e. it can be considered as the ontology of the course. The aim of whole teaching process is to update Student agents' Domain ontologies until they match the Tutor agent's Domain ontology.

A **Tutor agent's Task ontology** contains concepts about the order of course material delivering, teaching goals, teaching techniques, methods and styles.

A **Tutor agent's Application ontology** similarly as a Student agent's Application ontology is used to provide capability of making experiments with different pedagogical approaches. A Tutor agent's Task ontology together with a Tutor agent's Domain ontology constitutes complete Tutor model. Therefore, we don't need a specific Tutor agent's Application ontology, as it is in case of a Student's agent.

A **Search agent's Domain ontology** describes the concepts about possible area where an agent can search additional course materials. This ontology contains concepts, like Internet, intranet, available databases, etc.

A **Search agent's Task ontology** describes process and methods of searching for additional course materials.

A **Search agent's Application ontology** describes the interpretation of the received search parameters from Student agents and a Tutor agent. The Application ontology also specifies concepts introduced in the Search agent's Domain ontology and the Search agent's Task ontology.

A **Content Manager's Task ontology** describes concepts about process and techniques (text, slides, diagrams, pictures, etc.) of information visualisation. Forms of visualisation are defined by methodological considerations depending on specific goals and requirements of each course. It is also desirable that at least part of the course material is available in several forms of visualisation because students have different priorities of using them due to their different background knowledge and abilities to perceive materials (somebody prefers full text, somebody prefers "condensed" text presented in slides, while others prefer more diagrams, picture, etc.). Students' preferences are collected during students' actions and they are stored in the Student agents' Application ontologies.

A **Content Manager's Application ontology** describes the interpretation of a Student agent's Application ontology and a Tutor agent's Application ontology. These ontologies have concepts about student's learning style and preferences, tutor's teaching style, and visualisation concepts from the Contents Manager's Task ontology.

A **Test agent's Task ontology** has concepts about different kinds of quizzes (Race 1994), which can be used to test whether a student has learned the delivered course material. It also describes process of making different kinds of quizzes to provide testing of candidate concepts.

RELATED WORK

Our search for related works confirms that the proposed architecture represents a novel approach because we have not find exactly similar approaches. The previous works can be divided in some groups, which only to the certain extent can be considered to be similar with our conception (some parts of them are similar with our approach). These similar works can be divided into three groups: works related to ontologies, works related to usage of agent technologies, and works related to simulation in education.

♦ Related works on ontologies can be divided as follows:

- e-Learning, where ontologies are used to support management and retrieval of the course materials (Brace and Nejdl 2004; Baumann et al. 2002; Tane et al. 2003), as well as visualisation of delivered course material (Abecker and van Elst 2004));
- Agent technologies, where ontologies are used to support communication, coordination, interactions and information exchange between agents in MAS (Sycara and Paolucci 2004).

♦ Related works in the field of agents:

- Agents in education (information search, retrieval and representation, and different pedagogical agents for tutoring different learning courses (Zhong et al. 2003, Capuano et al. 2000; Dorca et al. 2003; Garro and Palopoli 2002);
- Agent-based simulation for different purposes (for example, modelling of different decision making (Sklar et al. 2004) or industrial processes).

♦ Related works in the field of simulation in education:

- Agent-based simulation in education (we have found only SimEd for simulation and modelling of search for the optimal educational policy (Sklar et al. 2004);
- Simulators, which are used for training some specific skills (like, simulators in aviation for pilots trainings or manufacture workers (Ho and Vance 1995)).

Figure 4 shows overlapping of four fields (agents, simulation, education and ontologies) to illustrate connections between them. It shows related works and position of our proposal in these areas.

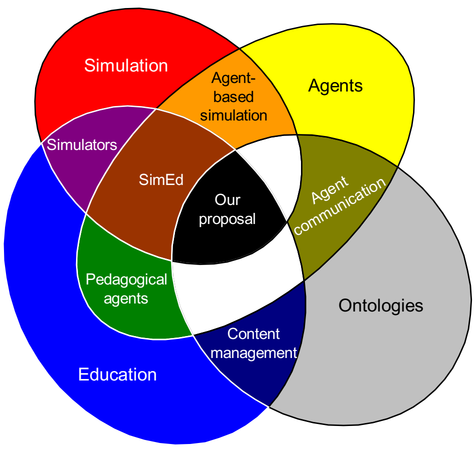

Figure 4: Areas of related works

The presented work differs from related works because specific interactions in specific knowledge domains are simulated using different kinds of ontologies. Our approach integrates these ontologies into multi-agent system. The long-term goal of this research is to simulate on the individual bases the interactions that take place between the tutor and the group of students. We hope that this approach will allow tutors to experiment with their decisions and their impact on the teaching and learning process considering teaching methods, sequence of topics, course material representation techniques, assessment methods of student progress, etc.

CONCLUSIONS AND FUTURE WORK

In this paper a novel approach of multi-agent systems architecture based on ontologies is introduced. The proposed system should be considered as a testbed for agent-based simulation of teaching and learning process. The role of ontologies is described and ways how ontologies may be used in intelligent tutoring systems are discussed. This is only a initial phase of the development of the agent-based simulation of intelligent tutoring system in which different ontologies are used. This paper presents a conceptual level of the proposed approach.

Future work is connected with the development of a prototype of the intelligent tutoring system based on the proposed approach. After that the experimental campaign is planned. Further studies will concern the architecture and models of the proposed system. They will be defined in details after the end of the experimentation and analysis of results obtained. We are also planning to make experiments how to obtain students' initial ontologies more effectively and to work on how to prepare tests for establishing concepts and relations between concepts for students' initial ontology.

REFERENCES

Abeker, A. and L. van Elst. 2004. "Ontologies for Knowledge management". In *Handbook on Ontologies 2004,* S. Staab and R. Studer (Eds.). Springer-Verlag, 435-454.

Anohina, A. 2003. "Classification of the Terminology Used in the Field of Virtual Learning". In *Scientific Proceedings of Riga Technical University, 5th Series, Computer Science, Applied Computer Systems, V. 17*. Riga, RTU, 94-102.

Baumann, S.; A. Dengel; M. Junker; and T. Kieninger. 2002. Combining Ontologies and Document Retrieval Techniques: A Case Study for an E-Learning Scenario. In *Proceedings of the 13th International Workshop on Database and Expert Systems Applications (DEXA'02)*.

Brace, J. and W. Nejdl. 2004. "Ontologies and Metadata for eLearning". In *Handbook on Ontologies 2004,* S. Staab and R. Studer (Eds.). Springer-Verlag, 555-573.

Capuano, N.; M. De Santo; M. Marsella; M. Molinara; and S. Salerno. 2000. "A Multi-Agent Architecture for Intelligent Tutoring". In *Proceedings of the International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet SSGRR 2000.*

Capuano, N.; M. Gaeta; A. Micarelli; and E. Sangineto. 2002. "An Integrated Architecture for Automatic Course Generation". In *Proceedings of IEEE International Conference on Advanced Learning Technologies (ICALT 2002)*, 322-326.

Chau, M. and H. Chen. 2003. “Personalized and Focused Web Spiders”. In *Web Intelligence 2003,* N. Zhong; J. Liu; and Y.Y. Yao. (Eds.). Springer-Verlag, 197-217.

Dorca, F.A.; C.R. Lopes; and M.A. Fernandes. 2003. “A Multiagent Architecture for Distance Education Systems”. In *Proceedings of the 3rd IEEE International Conference on Advanced Learning Technologies (ICALT’03),* 368-369.

Esteva, M.; D. de la Cruz; and C. Sierra. 2002. “ISLANDER: an electronic institutions editor“. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AA-MAS 2002).* (Bologna, Italy, July). 1045-1052.

Garro, A. and L. Palopoli. 2002. “An XML Multi-Agent System for e-Learning and Skill Management”. In *Proceedings of the 3rd International Symposium on Multi-Agent Systems, Large Complex Systems, and E-Businesses (MALCEB'2002)* (Erfurt, Germany, October), 283-294.

Greer, J.E. and G.I. McCalla (Eds.). 1994. *Student Models: The Key to Individualized Educational Systems.* Springer Verlag.

Groeneboer, T.C.C. and D. Stockley. 1997. “Virtual-u: A collaborative model for online learning environments”. In *Proceedings of the Second International Conference on Computer Support for Collaborative Learning* (Toronto, Canada, December).

Grundspenkis, J. 2003. “Development of Hybrid Intelligent Systems: Integration of Structural Modelling, Intelligent Agents and Knowledge Management Techniques”. In *Scientific Proceedings of Riga Technical University, 5th Series, Computer Science, Applied Computer Systems, V. 17*. Riga, RTU, 7-30.

Guarino, N. 1998. “Formal Ontology and Information Systems”. In *Proceedings of Formal Ontologies in Information Systems 1998* (Trento, Italy, June). N. Guarino (Eds.). IOS Press, 3-15.

Ho, L.K. and J.M. Vance. 1995. "Virtual Training in Manufacturing Systems Using TELEGRIP". In *Deneb User Group IX Meeting Proceedings.* (Ypsilanti, MI, October).

Johnson, W.L. 2003. “Using Agent Technology to Improve the Quality of Web-Based Education”. In *Web Intelligence 2003,* N. Zhong; J. Liu; and Y.Y. Yao. (Eds.). Springer-Verlag, 77-101.

Lenat, D.B. 1995. “CYC: a large-scale investment in knowledge infrastructure”. *Communications of the ACM* 38 No.11 (Nov), 32-38.

Maedche, A. and S. Staab. 2004. “Ontology Learning”. In *Handbook on Ontologies 2004,* S. Staab and R. Studer (Eds.). Springer-Verlag, 173-190.

Meisel, H.; E. Compatangelo; and A. Hoerfurter. 2003. “An ontology-based approach to intelligent Instructional Design support”. In *Proceedings of the 7th International Conference on Knowledge-Based Intelligent Information & Engineering Systems (KES’2003)*, volume 2773 of LNAI, Springer-Verlag, 898-905.

Miller, G.A. 1995. “WordNet: A Lexical Database for English”. *Communications of ACM* 38 No.11 (Nov), 39-41.

Paulsen, M.F. 2003. “Online education. Learning Management Systems”. NKI Forlaget, Norway.

Race, P. 1994. “The Open Learning Handbook: Promoting quality in designing and delivering flexible learning”, Kogan Page Ltd, London, Nichols Publishing Company, New Jersey.

Sklar, E.; M. Davies; and M.S.T. Co. 2004. “SimEd: Simulating Education as a Multi Agent System”. In *Proceedings of Third International Conference of Autonomous Agents and Multi Agent Systems (AAMAS-2004)*. 998-1005.

Sycara, K. and M. Paolucci. 2004. “Ontologies in Agent Architectures”. In *Handbook on Ontologies 2004,* S. Staab and R. Studer (Eds.). Springer-Verlag, 343-363.

Tane, J.; C. Schmitz; G. Stumme; S. Staab; and R. Studer. 2003. “*The Courseware Watchdog: an Ontology-based tool for finding and organizing learning material*”. Fachtagung Mobiles Lernen und Forschen, 6.11.2003, Universität Kassel.

Weiss, G. (Eds.). 1999. *Multiagent Systems: A Modern Approach to Distributed Modern Approach to Artificial Intelligence*. The MIT Press.

Weissenberg, N.; A. Voisard; and R. Gartmann. 2004. “Using Ontologies in Personalized Mobile Applications”. In *Proceedings of the 12th annual ACM international workshop on Geographic information systems* (Washington, DC, USA, Nov. 12-13), 2-11.

Wooldridge, M.J. 2002. *An Introduction to Multi – Agent Systems*. John Wiley & Sons.

AUTHOR BIOGRAPHIES

Vita Graudina is the Master student at Riga Technical University. She is working on the final version of her Master thesis. The topic of the thesis is the usage of ontologies in different information and knowledge-based systems. Contact her Riga Technical University, 1 Kalku str., LV-1658, Latvia; vita@cs.rtu.lv.

Janis Grundspenkis is the professor at Riga Technical University. He also is the dean of the Faculty of Computer Science and Information Technology, the director of the Institute of Applied Computed Systems, and the head of the Department of Systems Theory and Design. His research interests are agent technologies, knowledge engineering and management, and structural modelling. He has published more than 150 scientific papers. Contact him Riga Technical University, 1 Kalku str., LV-1658, Latvia; jgrun@cs.rtu.lv.