THE AUTONOMOUS SYSTEMS LABORATORY
UNIVERSIDAD POLITÉCNICA DE MADRID

A Survey on Ontologies for Agents

From Theory to Practice

J. Bermejo–Alonso, R. Sanz and I. López

ASLab ASL-A-2006-XXX v 1.0 Draft
June 12, 2006

Ontologies and agents are two research areas that have become intertwined in recent years. Ontologies have started to be developed aiming at agent–based applications. Agents have benefited by the use of ontologies in heavily information–based processes.

From the theory of ontology and agenthood to its application in practice, we have reviewed the available literature. Based on our research, we summarize the state–of–the–art in ontology–based agent applications.

1 Introduction

Knowledge Engineering (KE) and Artificial Intelligence (AI) research have addressed the use and development of ontologies as a mean to improve knowledge processes. Ontologies allow to define the concepts and relationships within a universe of interest. The goal is not to define what is and what it is not, but to define useful concepts for knowledge– and computer–based systems.

Another area under continuous development is agent technology. Different agents, architectures, methodologies and languages have been developed. Agents have proven to be useful to handle information and knowledge–based processes.

It was a matter of time that agents would need to incorporate ontologies. Agents can benefit from the use of ontologies in heavily knowledge–oriented tasks, and specially when the knowledge involved constitutes the very foundation of system–wide performance (as it is the case in the domain of agents in distributed embedded control systems).

Our research has initially addressed ontologies and agents from a theoretical viewpoint. Later, an analysis on how ontologies are applied within the agent community from a practice viewpoint. The analysis has provided useful conclusions for our further research on ontology– agent–based systems for the control domain.

2 A Review of Fundamentals of Ontologies and Agents

Regarding ontologies, we aimed at providing a brief review of the state of the art for theoretical foundations of ontologies as a grounding for systematic real–time agent engineering. We have let aside the review of the methodologies, languages and tools developed for ontologies, as they fall outside the scope of this paper. Excellent reviews could be found in [20], [40].

In relation to agents, our approach was not to provide an exhaustive description of fundamentals regarding agent's definition, architectures and methodologies. The reasons are twofold. Firstly, it has already been done in the literature [18], [5], [25], [29] [14]. Secondly, agent theory is not the main focus of this paper. Hence, only the most relevant concepts are presented, for later reference within the analysis of ontologies in agent–based systems.

2.1 On the Notion of Ontology

Ontology was originally linked to Philosophy where it means the philosophy of being (ontos=being and logos=treatise).

The term ontology became relevant to the Knowledge Engineering community where it is understood as a systematic account of Existence. Ontology, in knowledge engineering contexts, was defined by Gruber as *an explicit specification of a conceptualization* [21], [22]. This definition was quickly adopted by the AI community. The concept of *conceptualization* was originally defined as "a set of extensional relations describing a particular state of affairs" [19]. To specify the conceptualization, an ontology consists of classes, instances, functions, relationships and axioms. All these elements allow for the definition of the entities in the domain, as well as establishing the constraints and bonds among them.

According to [23] an ontology is " a logical theory accounting for the *intended meaning* of a formal vocabulary, i.e., its *ontological commitment* to a particular *conceptualization* of the world. The intended models of a logical language using such vocabulary are constrained by its ontological commitment. An ontology indirectly reflects these commitments (and the underlying conceptualization) by approximating these intended models".

A comprehensive definition was provided by [43]: "An ontology is a formal, explicit specification of a shared conceptualization" where *conceptualization* is defined as an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon; *explicit* refers to explicitly defining concepts and constraints; *formal* implies that the ontology should be machine-readable, and *shared* specifies that an ontology captures knowledge accepted by a group. This group may be just a collection of software agents, or more generally, a collection of knowledge–based agents (human and computational) that use this shared knowledge during system exploitation or as a basis of human–centric engineering processes for agent construction, deployment and decommission. This last case, i.e. the integration of engineering and exploitation knowledge, is of major importance in the case of high dependability, real–time systems.

2.2 Ontological Commitments

Ontologies are to be used and shared by different users, either humans or machines. Therefore, all the actors involved should agree on how the vocabulary representing the conceptualization is to be used. The term *ontological commitment* is used to describe "agreements about the objects and relations being talked about among agents, at software module interfaces or in knowledge bases" [22]. The more specific the domain to be modeled, the more the ontological commitments to be defined [6].

2.3 The Definition of Agency

When approaching the topic of agents, the first step should be to find out what an agent is or what researchers consider an agent to be. However, there is not an agreement on a definition of agent.

An intelligent agent has been defined as a computer system which exhibits certain properties (autonomy, social ability, reactivity, pro–activeness, adaptivity) and which is implemented using concepts usually related to humans [52].

The notion of agent has evolved with time and further research in the agent community. Attributes and roles have been further detailed to address a particular focus of interest.

Agents, therefore, could be *rational* (acting on its environment, and which chooses to act to fulfil its own best interests [47]); *autonomous* (agent that should achieve autonomously goals by making decisions and carrying out actions in an environment [18], [16], [14], [1]); *mobile* (agents can move freely in an electronic network, communicating with objects of the environment such as information resources of other agents [5]); *cognitive*(an autonomous agent with human-like cognitive features but, in general, any agent that exploits explicit knowledge [3]).

When several problem–solving agents form part of a whole, it is referred to as a *multi–agent systems* (MAS). Usually the agents are heterogeneous both on their capabilities and their goals. Therefore, coordination, negotiation and communication among agents are essential features of a MAS to avoid duplication of efforts, inference in achieving goals, provide robustness by redundancy or exploit of agent's capabilities [29].

3 Ontologies for Agents

3.1 Reasons to Use Ontologies for Agents

Ontologies are widely used, not only in agent–based applications. Nevertheless, they provide specific benefits for agent applications. These benefits could be summarized as follows [7], [36], [54]:

1. Ontologies clarify the structure of knowledge: performing an ontological analysis of a domain allows to define an effective vocabulary, assumptions and the underlying conceptualization. The analysis also allows to separate domain knowledge from operational or problem–solving one.

2. Ontologies help in knowledge scalability: knowledge analysis can result in large knowledge bases. Ontologies help to encode and manage in a scalable way.

3. Ontologies allow knowledge sharing and reuse: by associating terms with concepts and relationships in the ontology as well as a syntax for encoding knowledge in them, ontologies allow further users and agents to share and reuse such knowledge.

4. Ontologies increase the robustness of an agent–based system: agents can draw on ontological relationships and commitments to reason about novel or unforeseen events in their domain.

5. Ontologies provide a foundation for interoperability among agents.

6. Ontologies, that have as focus the domain of software engineering of agent–based systems, do help sustain development teams and software processes, and may even render useful during exploitation phases as a foundation of cognitive understanding and integration of agents including cognitive self–reflection capabilities.

3.2 A Classification of Ontology–based Agent Developments

To understand how ontologies are used within the agent community, we have reviewed the available literature. In our analysis we started by considering the traditional roles played by ontologies as described in [46], [26], [7], [49], [48]. Nevertheless, our review of the literature highlighted the necessity of considering new roles, as a result of combining agent technology with ontologies.

3.2.1 Ontology as Specification

In this case, the ontology is created to aid in the development of software systems in a given domain. This is the approach described in [4], where an ontology is used as a *specification of an information system*. A domain ontology is combined with different method ontologies, which allow the definition of an application ontology. Such ontology allowed to specify both the application's functionalities and the knowledge required for the application to carry out its tasks.

3.2.2 Common Access to Information

The underlying idea is to use the ontology to share terms among users, either humans or agents.

Ontologies have shown its utility in systems that operate close to a human conceptual level, as it happens in a Virtual Enterprise environment. Partners should negotiate upon common standards of which an ontology is part of. The approach was implemented as an agent–based platform, ForEV [30], using a multiple ontology approach [49]. Ontologies and agents technologies have also been combined to *solve the semantic heterogeneity problem in e–commerce negotiations* and transactions [8], [31].

Sharing terms are also important in robotics applications. An ontology has been used within the RoboCup robot soccer domain [32] both to share knowledge, but also to ground *sensory information* to symbols used to represent real world objects in the software applications.

3.2.3 Ontology–based Search

The role of the ontology is to aid in the search of information within a repository. In this context, many of the developments of ontologies have been focused in the Semantic Web as promoted by the World Wide Web Consortium. The Semantic Web is a project that intends to create a universal medium for information exchange by placing documents with computer-processable meaning on the World Wide Web by using standards, mark–up languages and related processing tools. Ontologies are used as common *metadata vocabularies*, to allow document creators to know how to mark up their documents so that agents can use the information in the supplied metadata. Research in Spain related to the Semantic Web has been summarized in [9].

3.2.4 Reuse of Knowledge

The ontology enables the reuse of knowledge to address the domain knowledge needs of potential new applications.

This role has been described in [4], where an ontology, PHYSSYS, was constructed to describe the knowledge for physical systems such as heating systems, automotive systems and machine tools. The reusability aspects were addressed by using *ontology projections* to describe technical components, physical processes and mathematical relations.

3.2.5 Integration of Heterogenous Information Sources

The ontology is used to support the integration information task.

An example is the definition of an *ontology–based agent system related to different domains* as described in [35]. It was defined as a global domain ontology to be used by agents to communicate about hazardous waste measurements. Different types of agents were modeled to address translation issues between the global ontology to the user's and resource's local ontologies.

A *multiagent framework for collaborative understanding of distributed ontologies* is described in [34]. The framework consists of two set of components: The ontological components allow the agents to communicate and understand each other. The operational components describe the query processing, action planning and query composition. A *dynamic ontology integration for a multi-agent environment* is described in [15]. Each agent holds the ontologies of other agents of its interest (acquaintances). The integration has to be carried out whenever a new acquaintance is added or when the local ontology of an acquaintance changes. The most interesting concept handled by the two aforementioned approaches is the notion of neighborhood or acquaintances, which implies that an agent can communicate or is interested only with a few agents, while the other agents are ignored. Such a concept allows to handle to some extent the scalability problems which usually appear on large multi–agent systems.

3.2.6 Ontologies for Modeling

A new role of ontologies within agent–based systems is modeling. Ontologies are used to model the concepts the agents need and the internal operations or tasks that agents carry out.

As an example, an *ontology associated with the FIPA Request Interaction Protocol* was defined in [12], [13]. In this case, concepts referred to message types, the reason for the request and the precondition to be fulfilled. Internal agent operations were modeled as a combination of classes and objects defining the operation to perform, the implementation of the operation and the invocation of the operation.

Ontologies have also been used to *model the world for autonomous vehicles* [45]. Having an accurate description of the environment (obstacles, paths, etc.) is a key issue for such types of vehicles. In this case, the ontology was used to model the obstacles to support the navigation task of the vehicle.

To develop ontologies for modeling agent systems, the Unified Modeling Language (UML) [38] has been used. Although UML was not initially developed for ontology development, there are ongoing efforts to make possible its application within the ontology domain [37]. As examples, UML has been used to prove its suitability to *model software agents systems* in general [10], [2] and agents applied to a travel booking scenario [11].

3.2.7 Ontologies for Semantic Interoperability

Semantic interoperability is defined as "the problem of achieving communication between two agents that work in the same or overlapping domains, even if they use different notations and vocabularies to describe them" [17]

Agents are highly heterogeneous in real applications. They are likely to be incapable of fully understanding each other, so syntactic and semantic mismatches can arise. Moreover, agents are characterized by different views of the world which are explicitly defined by ontologies.

Common, global or shared ontologies are used to overcome the semantic heterogeneity among agents. A commitment to the shared ontology permit the agents to interoperate and cooperate while maintaining their autonomy. A common ontology built up either by sharing, merging or translating ontologies has been proposed as a possible solution to address the semantic interoperability [44], [35].

Despite the use of a common ontology, some issues still remain. Firstly, common ontologies are useful as long as they stay within the context they were defined. They are not that easily portable to other domains. Neither the concepts nor the agents' roles are capable of evolving as the context changes and requires. Secondly, commitment to a common ontology may guarantee consistency but not completeness [33]. Agents sharing an ontology might not be totally committed to it, as each agent would work with both a local ontology and only a part of the common ontology as described in [42].

Research on the topic addresses the aforementioned issues. A proposed solution [28] is to follow the ROADMAP methodology [27], as extension of the GAIA methodology [53], combined with the EXPLODE methodology in the development of a multi–agent system [24]. The two first ones describe the models to be considered within the multi–agent systems, both considering a role hierarchy and an agent hierarchy. EXPLODE allows the development of the knowledge model considered in the system without depending on the ontology structure, by a multi–

stage approach combined with continuous integration.

Semantic Interoperability is especially relevant for those agent– and multi–agent–based applications related to the Semantic Web. Agents need to be capable of communicating and sharing knowledge without semantic mismatches. Just as two recent examples within the extensive literature on the topic, this issue has been addressed in [39], [51].

3.2.8 Dynamic Ontology

As agents work in real changing environments, ontologies should also evolve to cope with those changes. Researchers refer to this type as *dynamic ontology*, which could be described as a shared ontology that adapts to an application domain and evolves with time as the concepts in that domain change. Some attempts to come up with such an ontology are described in [50], [8].

4 Conclusions

4.1 On Ontologies

Ontologies are still under debate and development within the KE and AI communities. Several definitions have been provided, each one stressing a different viewpoint.

An ontology can be seen as a vocabulary describing the terms of a domain or task (or more general if being an upper ontology). However, the key point is not the ontology as a vocabulary but the underlying meaning and commitments of that vocabulary. By such, we mean that an ontology could not just be a more or less detailed list of terms belonging to a domain expressed in particular language (either formal or informal) but, in some sense, may include operational issues concerning the use of it. The meaning, relationships, constraints and axioms of these terms are what builds up the vocabulary to transform it into an ontology.

4.2 On the Concept of Agent

Despite the widely use of agent, there has not been a consensus in work dealing with agents and agency. The term agent is ubiquitous in the literature on intelligent systems, software engineering, and complex systems. However, it is rarely defined, and its possible interpretations are somehow vague.

We have even participated in the discussion of a new type of agent, called *sapient agent*. Such an agent exhibits wisdom and sapience, understood as the capabilities of providing meaning to the other agents of the society its belongs to [41]. It is still unknown if the term will take in within the agent community.

We have pinpointed the difficulty in agreeing on the underlying concept involved in the term agent. As a possible solution, we suggest to turn the problem over: instead of defining what an agent is, let's find common situations where agents are involved in. Therefore, attributes and features of agents could be used to characterized them.

4.3 On the Application of Ontologies for Agents

Our first conclusion regards to the kind of agents used. Within the agent literature, the class of agent (autonomous, mobile, cognitive, etc.) used in the application is described in full detail. When describing the ontology–based agent systems, such type is however skipped. Therefore, is a particular agent type more suitable to be used with an ontology?. It remains an open question.

Communication or ontological commitments among agents using different ontologies either at a local vs. global level or when merging/combining from different ontologies are another focus of interest. It seems that the agent communication is not so straightforward as desired. Most of the research carried out so far addresses in detail the communication/merging/integration problems and their solution. However, the analysis is not so much made from a meaning or semantical level but rather from a symbol level viewpoint. The concerns focus on how different vocabularies are used and understood, not how the agent "understands" the meaning associated with the terms of the vocabulary. The sharing of ontologies depends heavily on a precise semantic representation of concepts and their properties

To make matters worse, ontology–based agent systems usually encompass several agent roles (mappers, translators, coordinators, etc) to address the previous interaction problems. Once again, the roles definitions and underlying assumptions change from deployment to deployment. If several roles are needed, we point out the necessity to establish a common definition or features for agents' roles. Otherwise, ontology–based agent research might end up with the same vagueness as shown in agent–based research.

Furthermore, the ontology–based agent systems are in general, with some exceptions [28], developed ad–hoc. None of the existing agent methodologies has been used. Neither ontology development methodologies. Therefore, the development of such systems seem to be too time and effort consuming, not to mention the several problems encounter from requirements to implementation. We address the research on reusable components or patterns to reduce such efforts.

5 Further Work

The research conducted allowed us to identify fundamentals concepts, trends and research on the topic of ontologies for agents. Being the ontology–based agent systems a domain under ongoing research, benefits, drawbacks and further experiences should also be considered.

Our aim is to gain a thorough insight to be applied for the development of an Ontology for Autonomous Systems (OASys) with the purpose of defining the concepts, relationships and architecture to be used in a multi–agent system within the real–time and embedded control systems. This ontology is part of the ASys Long Term Project conducted by the Autonomous Systems Laboratory (ASLab) to create a science and technology for the construction of highly autonomous systems.

The underlying idea of the ontology for ASys is one where the ontology should express the concepts, consider the constraints or relationships in an explicit way under some ontological commitments but most importantly build the ontology to be readable by computers. This way the ontology will become an engineering artifact within a software process developed to define and implement autonomous systems. The ontology so understood is a mapping of the philosophical meaning of ontology into agent or knowledge–based systems epistemology.

References

[1] R. Ashri, I. Rahwan, and M. Luck. Architectures for negotiating agents. In Mueller J. Marik V. and Pechoucek M., editors, *Multi-Agent Systems and Applications III*, volume 2691 of *Lecture Notes In Artificial Intelligence*, pages 136–146. Springer, 2003.

[2] F. Bergenti and A. Poggi. Exploiting uml in the design of multi-agent systems. *Lecture Notes in Computer Science*, 1972:106–113, 2001.

[3] M. Bogner, U. Ramamurthy, and S. Franklin. Consciousness and conceptual learning in a socially situated agent. In K. Dautenhahn, editor, *Human Cognition and Social Agent Technology*. Amsterdam: John Benjamins, 2000.

[4] W.N. Borst. *Construction of Engineering Ontologies*. Centre for Telematica and Information Technology, University of Tweenty. Enschede. The Netherlands, 1997.

[5] W. Brenner, R. Zarnekow, and H. Wittig. *Intelligent Software Agents: Foundations and Applications*. Springer, 1998.

[6] B. Chandrasekaran, J.R. Josephson, and V.R. Benjamins. Ontology of tasks and methods. In *11Th Workshop on Knowledge Acquisition, Modeling and Management (KAW'98)*, Banff, Canada, 1998.

[7] B. Chandrasekaran, J.R. Josephson, and V.R. Benjamins. What are ontologies and why do we need them? *IEEE Intelligent Systems*, 14(1):20–26, 1999.

[8] Q. Chen and U. Dayal. Multi–agent cooperative transactions for e–commerce. In *Conference on Cooperative Information Systems*, pages 311–322, 2000.

[9] O. Corcho, A. Gómez Pérez, and M. Fernández López, editors. *CAEPIA2005: Taller Ontologías y Web Semántica*, Santiago de Compostela, Spain, Noviembre 2005.

[10] S. Cranefield, S. Haustein, and M. Purvis. Uml-based ontology modelling for software agents. In *Workshop on Ontologies in Agent Systems, OAS'2001*, 2001.

[11] S. Cranefield, J. Pan, and M. Purvis. A uml ontology and derived content language for a travel booking scenario. In *3rd Workshop on Ontologies in Agent Systems, 2nd International Conference on Autonomous Agents and Mas, AAMAS'03*, pages 55–62, 2003.

[12] S. Cranefield, M. Purvis, and M. Nowostawski. Is in an ontology or an abstract syntax?: Modelling objects, knowledge and agent messages. In *Workshop on Applications of Ontologies and Problem–Solving Methods*, pages 16.1–16.4, 2000.

[13] S. Cranefield, M. Purvis, M. Nowostawski, and P. Hwang. Ontologies for interaction protocols. In V. Tamma, S. Cranefield, T.W. Finnin, and S. Willmott, editors, *Ontologies for Agents: Theory and Experiences*, Whitestein Series in Software Agent Technologies, pages 1–17. Birkhäuser, 2005.

[14] P. Davidsson. *Autonomous Agents and the Concept of Concepts*. PhD thesis, Department of Computer Science, Lund University, 1996.

[15] S.M. Deen and K. Ponnamperuma. Dynamic ontology integration in a multiagent environment. In *20th International Conference on Advanced Information Networking and Applications, AINA'06*, pages 373–378, 2006.

[16] E.D. DeJong. *Autonomous Formation of Concepts and Communication*. PhD thesis, Vrije Universiteit Brussel, 2000.

[17] D. Dou, D. Mcdermott, and P. Qi. Ontology translation by ontology merging and automated reasoning. In V. Tamma, S. Cranefield, T.W. Finnin, and S. Willmott, editors, *Ontologies for Agents: Theory and Experiences*, Whitestein Series in Software Agent Technologies, pages 73–94. Birkhauser, 2005.

[18] S. Franklin and A. Graesser. Is it an agent, or just a program? a taxonomy for autonomous agents. In *Third International Workshop on Agent Theories, Architectures and Languages*, Intelligent Agents III, pages 21–35. Springer Verlag, 1997.

[19] M.R. Genesereth and L. Nilsson. *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann, Los Altos, California, 1987.

[20] A. Gómez Pérez, M. Fernández López, and 0. Corcho. *Ontological Engineering: With Examples from the Areas of Knowledge Management, E-Commerce and the Semantic Web*. Advanced Information And Knowledge Processing. Springer, 2004.

[21] T.R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli, editors, *International Workshop on Formal Ontology in Conceptual Analysis and Knowledge Representation*, Padova, Italy, 1993. Kluwer Academic Publishers.

[22] T.R. Gruber. A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2):199–220, 1993.

[23] N. Guarino. Formal ontology and information systems. In N. Guarino, editor, *Formal Ontology in Information Systems, FOIS 98*, pages 3–15, Trento, Italy, June 1998. Ios Press.

[24] M. Hristozova and L. Sterling. An extreme method for developing lightweight ontologies. In S. Cranefield, T. Finin, and S. Willmott, editors, *Workshop on Ontologies in Agent Systems*, Bologna, Italy, July 2002. CEUR Workshop Series, Vol. 66.

[25] C. Iglesias, M. Garrijo, and J. González. A survey of agent-oriented methodologies. In Jorg Muller, Munindar P. Singh, and Anand S. Rao, editors, *Proceedings Of The 5Th International Workshop On Intelligent Agents V : Agent Theories, Architectures, And Languages (Atal-98)*, volume 1555, pages 317–330. Springer-Verlag: Heidelberg, Germany, 1999.

[26] R. Jasper and M. Uschold. A framework for understanding and classifying ontology applications. In *12th Workshop on Knowledge Acquisition, Modeling and Management (KAW'99)*, Banff, Alberta, 1999.

[27] T. Juan, P. Pearce, and L. Sterling. Roadmap: Extending the gaia methodology for complex open systems. In *First International Joint Conference on Autonomous Agents and Multi–Agent Systems*, pages 3–10, Bologna, Italy, 2002.

[28] K. Lister, M. Hristozova, and L. Sterling. Reconciling implicit and evolving ontologies for semantic interoperability. In V. Tamma, S. Cranefield, T.W. Finnin, and S. Willmott, editors, *Ontologies for Agents: Theory and Experiences*, Whitestein Series in Software Agent Technologies, pages 121–144. Birkhauser, 2005.

[29] M. Luck and M. D'Inverno. *Understanding Agent Systems*. M. D'Inverno And M. Luck, Springer, 2004.

[30] A. Malucelli, D. Palzer, and E. Oliveira. B2b transactions enhanced with ontology–based services. In *ICETE'04 – 1St International Conference on E–Business And Telecommunication Networks*, pages 10–17, Portugal, August 2004.

[31] A. Malucelli, D. Palzer, and E. Oliveira. Combining ontologies and agents to help in solving the heterogeneous problem in e-commerce negotiations. In *International Workshop on Data Engineering Issues in E-Commerce (DEEC 2005), IEEE Computer Society*, pages 26–35, Tokyo, Japan, April 2005.

[32] R. Mendoza and M-A. Williams. Ontology based categorisation for robots. In *Australasian Ontology Workshop, Aow2005*, Sydney, Australia, December 2005.

[33] M.H. Nodine and J. Fowler. Agent to agent talk: “is anybody out there?”. In V. Tamma, S. Cranefield, T.W. Finnin, and S. Willmott, editors, *Ontologies for Agents: Theory and Experiences*, Whitestein Series in Software Agent Technologies, pages 43–72. Birkhauser, 2005.

[34] M.H. Nodine and J. Fowler. Distributed ontologies in a multiagent framework. In V. Tamma, S. Cranefield, T.W. Finnin, and S. Willmott, editors, *Ontologies for Agents: Theory and Experiences*, Whitestein Series in Software Agent Technologies, pages 95–120. Birkhauser, 2005.

[35] M.H. Nodine and J. Fowler. On the impact of ontological commitments. In V. Tamma, S. Cranefield, T.W. Finnin, and S. Willmott, editors, *Ontologies for Agents: Theory and Experiences*, Whitestein Series in Software Agent Technologies, pages 19–42. Birkhauser, 2005.

[36] N.F. Noy and D.L. Mcguinness. Ontology development 101: A guide to creating your first ontology. Technical Report KSL-01-05, Stanford Knowledge Systems Laboratory, March 2001.

[37] Object Management Group. *Ontology Definition Metamodel*, August 2004.

[38] Object Management Group. *Unified Modeling Language (UML), V. 2.0*, July 2005.

[39] B. Orgun, M. Dras, S. Cassidy, and A. Nayak. Dasmas–dialogue based automation of semantic interoperability in multi–agent systems. In T. Meyer and M. Orgun, editors, *Australasian Ontology Workshop, AOW'05*, volume 58, 2005.

[40] H.S. Pinto and J.P. Martins. Ontologies: How can they be built? *Knowledge and Information Systems*, 6(4):441–464, July 2004.

[41] R. Sanz and J. Bermejo. Meaning, wisdom and sapience in autonomous systems. In *IEEE International Conference on Integration of Knowledge Intensive Multi-Agent Systems (KIMAS'05), Workshop On Sapient Systems*, pages 401–405, Boston, Ma, 18-21 April 2005.

[42] H. Stuckenschmidt and I.J. Timm. Adapting communication vocabularies using shared ontologies. In *Second International Workshop on Ontologies in Agent Systems, OAS'2002*, 2002.

[43] R. Studer, V.R. Benjamins, and D. Fensel. Knowledge engineering: Principles and methods. *IEEE Transactions on Data and Knowledge Engineering*, 25(1-2):161–197, 1998.

[44] V. Tamma. *An Ontology Model Supporting Multiple Ontologies for Knowledge Sharing*. Phd, University Of Liverpool, 2001.

[45] M. Uschold, R. Provine, S. Smith, C. Schlenoff, and S. Balikirsky. Ontologies for world modeling in autonomous vehicles. In *18Th International Joint Conference on Artificial Intelligence, IJCAI'03*, 2003.

[46] A. Valente and J. Breuker. Towards principled core ontologies. In B.R. Gaines and M. Mussen, editors, *Proceedings of the KAW-96*, Banff, Ca., 1996.

[47] W. Van Der Hoek and M. Wooldridge. Towards a logic of rational agency. *Logic Journal of the IGPL*, 11(2):133–157, 2003.

[48] U. Visser and C. Schlieder. Modelling real estate transactions: The potential role of ontologies. In *The Ontology and Modelling of Real Estate Transactions in European Juristictions*, 2002.

[49] H. Wache, T. VÖgele, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann, and S. HÜbner. Ontology-based integration of information – a survey of existing approaches. In H. Stuckenschmidt, editor, *(IJCAI–01) Workshop: Ontologies And Information Sharing*, pages 108–117, 2001.

[50] P. Weinstein and P. Birmingham. Runtime classification of agent services. In *AAI'97 Spring Symposium on Ontological Engineering*, Stanford, Palo Alto, Ca., 1997 1997.

[51] A.B. Williams. Learning to share meaning n a multi–agent system. *Autonomous Agents aAnd Multi–Agent Systems*, 8(2):165–193, 2004.

[52] M. Wooldridge and N.R. Jennings. Intelligent agents: Theory and practice. *Knowledge Engineering Review*, 10(2):115–152, 1995.

[53] M. Wooldridge, N.R. Jennings, and D. Kinny. The gaia methodology for agent–oriented analysis and desing. *Journal of Autonomous Agents and Multi–Agent Systems*, 3(3):285–312, 2000.

[54] R.E. Wray, S.A. Lisse, and J.T. Beard. Ontology infrastructure for execution–oriented autonomous agents. *Robotics and Autonomous Systems*, 2004:113–122, 2004.