A Meta-ontological Framework for Multi-Agent Systems Design

Marina V. Sokolova[1,2] and Antonio Fernández-Caballero[1]

[1] University of Castilla-La Mancha, Polytechnical Superior School of Albacete,
Campus Universitario s/n, 02071-Albacete, Spain
`{marina,caballer}@dsi.uclm.es`
[2] Kursk State Technical University, Kursk, ul.50 Let Oktyabrya, 305040, Russia

Abstract. The paper introduces an approach to using a meta-ontology framework for complex multi-agent systems design, and illustrates it in an application related to ecological-medical issues. The described shared ontology is pooled from private sub-ontologies, which represent a problem area ontology, an agent ontology, a task ontology, an ontology of interactions, and the multi-agent system architecture ontology.

1 Introduction

The aim of this work is to apply an ontological analysis for complex situation descriptions which should be modelled as multi-agent systems (MAS) in a natural manner. The example used is that of a complex problem that includes a set of factors, which are grouped into indicators of environmental issues in a city caused by traffic, industrial activity and man-made sphere pollution, and indicators of public health represented through morbidity. In spite of numerous and interesting works in this sphere, this problem continues to be actual, as new techniques and tools that can be utilized for the examination appear, and new facts and knowledge about the problem are discovered.

Therefore, the main practical objective of the paper is the creation of an agent-based system for state situation assessment, monitoring the environment pollution and following the corresponding changes in public health, and generating a set of alternatives for successful and sustainable situation management. The MAS paradigm helps reducing the complexity of such a system and pooling the optimal solutions produced by autonomous and semiautonomous entities (agents) into a general strategy or plan [1]-[2].

When analyzing the problem of environmental impact upon population health [3], we come to the conclusion that this task has to be seen in relation to the studied region as a root stable entity. The region, in turn, is characterized by some environmental situation (resulted from industrial development, transport activity, water, air and soil pollution, etc.) and some health indicators, which are represented by morbidity and the number of diseased people grouped into classes endogeneous and exogeneous diseases.

The main step in organizing the terminological and informational foundation for further analysis and usage (MAS creation, simulation, alarm awareness, etc.)

supposes describing a distributed meta-ontology framework. This step is a basic one and states the initial quality of further study processes and correct treatment of the concepts.

According to Guarino et al. [4], an ontology can be understood as an intentional semantic structure which encodes the implicit rules constraining the structure of a piece of reality. There are a number of approaches to ontology creation, mostly induced by the specificity of the domain of interest and the nature of the tasks to solve (e.g. [5]), from which we can induce and convert to our aims an algorithm of distributed ontology creation:

1. Situation description in natural language.
2. Vocabulary creation (extraction of concepts describing the situation).
3. Taxonomy creation.
4. Distributed meta-ontology structure creation.
5. Domain of interest ontology statement.
6. Description of tasks to solve and creation of the respective private ontology.
7. Description of MAS roles, agents and creation of the system architecture ontology.
8. Description of agent ontology.
9. Agent environment ontology statement by specifying interaction and communication protocols.
10. Ontologies mapping.
11. Data Bases filling for a MAS.
12. Data Sources delivering to agents.

To look briefly the steps of a given algorithms, it is worth noting that step 1 - problem description - serves for a better understanding the aims of the research and structure of the functionality of the situation. This initial analysis helps defining concretely the problem at hand and recovering the concepts, their characteristics and relations to examine. On this stage expert information, which is supplemented by statistical data and multimedia references related to the problem, is used. The consequentially following task (2) is the creation of a vocabulary, which includes the necessary and sufficient information about the concepts. The further step 3 consists in adding a set of relations (including hierarchical ones) between the concepts to a vocabulary, which results into a taxonomy. As in our work we use the inductive method of ontology creation, then, on step 4 we determine the general structure of the meta-ontology and extract the main functionally and semantically separated components. On steps 5 to 8 we create private ontologies for the extracted components of the meta-ontology, namely domain of interest, MAS architecture, tasks, agents and interactions. At steps 9 and 10 the private ontologies are mapped together. Finally, we fill data bases for a MAS (11) and deliver the real data to agents (12). In the following part of the article the distributed meta-ontology and the private ontologies, as well as the mapping procedure, are described in detail.

2 Description of the Ontological Basis of the Multi-agent Architecture

It is well-known that ontology creation is based on expert knowledge about the problem area and the developer's experience and understanding. Though, the more generalized approach implies extracting the main group of concepts, semantically and functionally connected. As it is shown in [6], a typical ontology for a MAS includes the following models: domain of interest, aims and tasks, agents, interaction, and environment. Having accepted this model of distributed meta-ontology creation, the structure shown in Fig. 1 is proposed as a framework for meta-ontological MAS design:
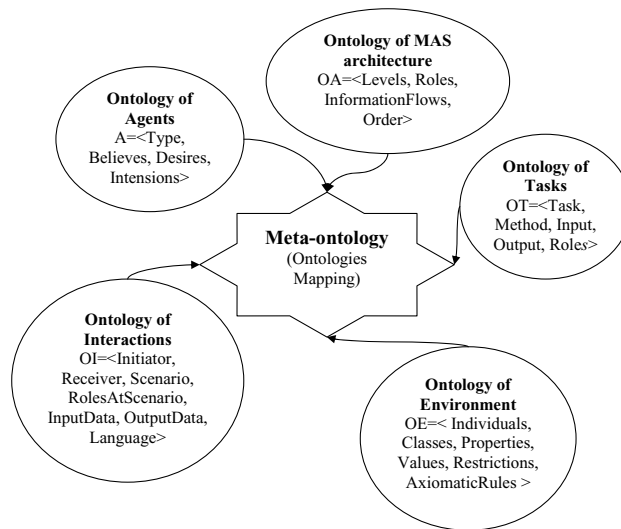

Fig. 1. The components of the distributed meta-ontology

This meta-ontology model specifies the private ontologies and gives opportunities to generalize knowledge about the MAS and the problem area. In the following subsections the focus is set on the components of the proposed meta-ontology.

2.1 The Domain of Interest Ontology

If defining the ontology O in terms of algebraic system, we have the following three attributes:

$$O = (C, R, \Omega) \tag{1}$$

where C is a set of concepts, R a set of relations between the concepts, and Ω a set of rules. Formula (1) proposes that the ontology for the domain of interest

(or the problem ontology) may be describe by offering proper meanings to C, R and Ω.

As we have used the ontology editing software Protégé [7], after widening the formula for our system, we get the following specialization for equation (1) :

$$OE = < Individuals, Classes, Properties, Values, Restrictions, AxiomaticRules >$$
$$(2)$$

Individuals are entities representing regions under study. These have certain mortality and pollution levels; concretely, in our current research, the possibilities for region are Spain, Castilla-La Mancha, Albacete, etc.

Classes are interpreted as "sets containing individuals", and are organized in a taxonomy in accordance with the hierarchical superclass-subclass relations.

Properties are binary relations on individuals, which enable asserting facts about classes and individuals and can be functional, inverse functional, symmetric, or transitive. The properties are used in restrictions and in axioms. *Values* contains the values that can be assigned to individuals. *Restrictions* state the permitted and extreme ranges. Generally speaking, *Restrictions* impose constraints on the properties of the classes. *AxiomaticRules* use restrictions, boolean algebra and some other concepts such as general classes to create properties and class axioms.

Let us take a look at the *Classes* with respect to our domain of interest. The general illustration to our understanding of the domain of interest includes regions, which are characterized with some environmental pollution and human health level. The accent is made on regions, which are represented by instances as Toledo, Albacete, etc. The ontology for the dimension "Regions, pollution, health population indicators", as created in Protégé 3.2, is represented in Fig. 1 and described as:

$$Morbidity = M^t = < m_j, m_{j,k}^{t,g,ag} >$$

where m is the set of nosologies, $m \in M$, t represents the time of registration (year), $j = 1 \ldots |M|$, k stands for a general class of disease (endogeneous or exogeneous) $k = 1, 2$, g is the gender, and ag stands for the age.

The superclass-subclass relations are stated through the indexes, as it is shown for theclass *Morbidity* in the Fig. 2.

$$Pollution = P_t = < p_i, pp_{i,j}, pp_{i,j}^t >,$$

where p represents the set of main pollutants, $p \in P$, pp is a sub-pollutant from class p, $pp \in PP$, $PP \subseteq P$, t is again the time of registration (year), $i = 1 \ldots |P|$, $j = 1 \ldots |PP|$. The *Environment* class is represented in similar manner as the class *Morbidity*, as it shown in Fig.3.

As stated previously, the *Morbidity* class includes two subclasses of diseases: endogeneous and exogeneous, which are detailed into nosologies in accordance with the International Classification of Diseases [9]. The *Environment* class includes the following performance indicators: water pollution, dangerous wastes, transport activity, and industrial activity parameters revealing dangerous emissions during energy life-cycles (use of energy, and so on).
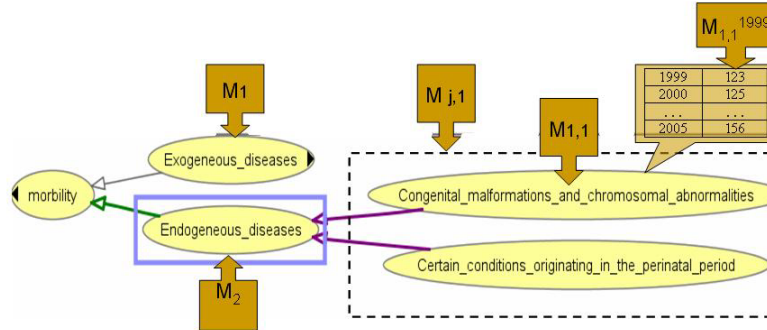
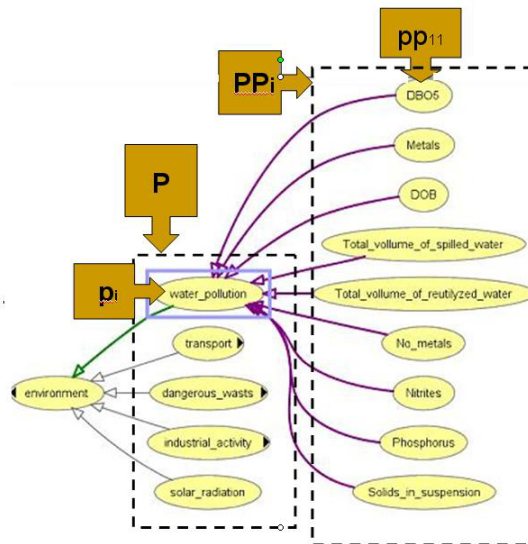**Fig. 2.** Illustration of *Morbidity* class representation

Fig. 3. Illustration of *Environment* class representation

2.2   The MAS Architecture Ontology

The initial analysis of the system was carried out with the Gaia methodology [10] and resulted in revealing and describing the system roles and protocols. The ontology for MAS architecture is stated as:

$$OA = < Levels, Roles, InformationFlows, Orders > \qquad (3)$$

where *Levels* correspond to logical levels of the MAS (see Fig. 1), *Roles* is a set of determined roles, *InformationFlows* is a set of the corresponding input and
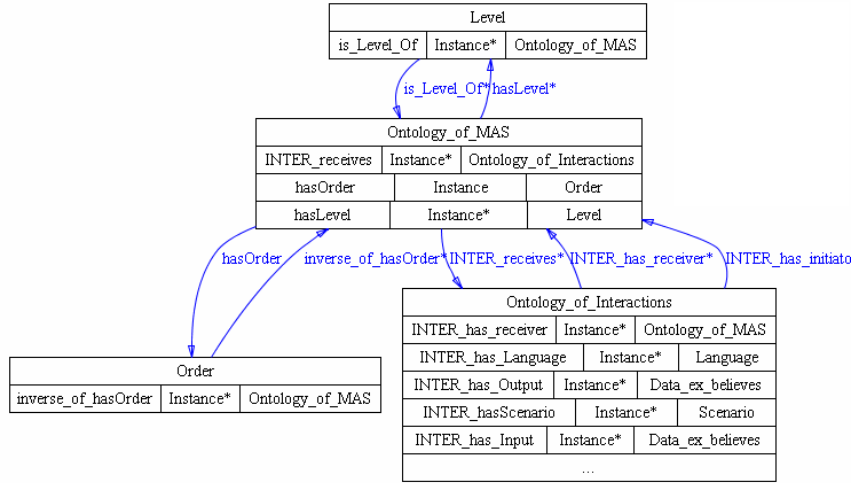
Fig. 4. The MAS Ontology

output information, represented by protocols. Lastly, the set *Order* determines the sequence of execution for every role.

The system consists of three levels. The first level is aimed for meta-data creation, the second one is responsible for hidden knowledge discovering, and the third level provides real-time decision support making, data distribution and visualization. This architecture satisfies all the required criteria to decision support systems as it includes the necessary procedures and functions.

In Fig. 4 there is an ontology created in accordance with the given formal description (3), which presents also its connection with the Interaction ontology (see 2.5), which determines protocols.

The first level is named "Information fusion" and it acquires data from diverse sources and preprocess the initial information to be ready for further analysis. The second layer is named "Data Mining" and there are three roles at this level, dedicated to knowledge recover through modelling, and calculation impact of various pollutants upon human health. The third level, "Decision Making", carries out a set of procedures including model evaluation, computer simulation, decision making and forecasting, based on the models created on the previous level. The main function of this level is to provide a user - actually, a person who makes decision - with the possibility to run online real-time "what - if" scenarios.

The end-user, that is to say the person making decisions, interacts with the MAS through a System-User Interaction protocol, which is responsible for human-computer interaction. The user chooses the indicator he wants to examine and initiates a computer simulation.

2.3   The Tasks Ontology

In order to fulfil the assigned aims, the MAS have to realize the set of tasks and subtasks. The task ontology is represented by the following components:
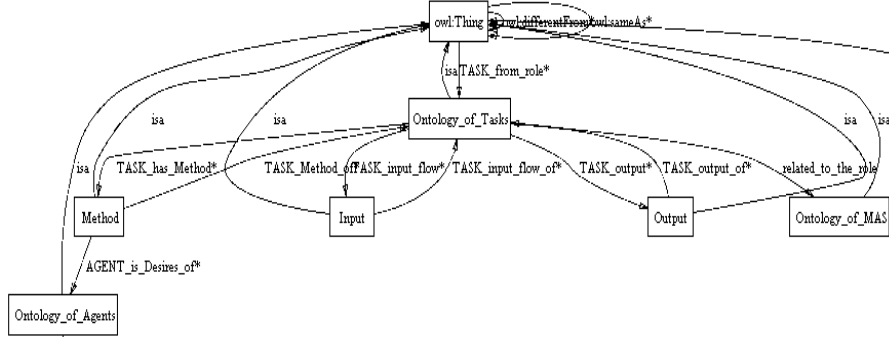
Fig. 5. The Task Ontology

$$OT = <Task, Method, Input, Output, Role> \qquad (4)$$

where *Task* is a set of tasks to be solved in the MAS, and *Method* is a set of activities related to the concrete task, *Input* and *Output* are input and output data flows, *Role* is a set of roles, which utilize task.

In Fig. 5 there is a private ontology created in Protégé, which demonstrates the formal model of the task ontology. The component "Role" is inherited from the Ontology of MAS. The tasks are shared and can be accomplished independently, in accordance with an order, which is inherited from the MAS architecture ontology through the *Role* component. The tasks delegation is being delivered for every type of agent. Actually, all the types of agents solve particular tasks and have determined responsibilities. This fact let us relate our system to organizational MAS [12], which is strictly organized and does not require any kind of control agents.

2.4   The Agent Ontology

In our approach we model by BDI agents; their architecture consists of three data structures: *Beliefs*, *Desires* and *Intentions* (which include a plan library). The *Beliefs* are usually represented as facts or in form of information files, data bases, and correspond to the information the agent has about its environment. *Desires* are actions or goals that the agent wants to achieve, and *Intentions* are the desires that the agent chooses under the given circumstances. Intentions are realized in form of actions, which are formed in a plan library, which consists of sequences of steps the agent can execute to achieve its goals. Actually, *Intentions* is a subset of *Desires*. Hence, we describe every agent as a composition of the following components:

$$Agent = <Beliefs, Desires, Intentions, Type>, \qquad (5)$$

Every agent has a detailed description in accordance with the given ontology, which is offered in a form of BDI cards, in which the pre-conditions and
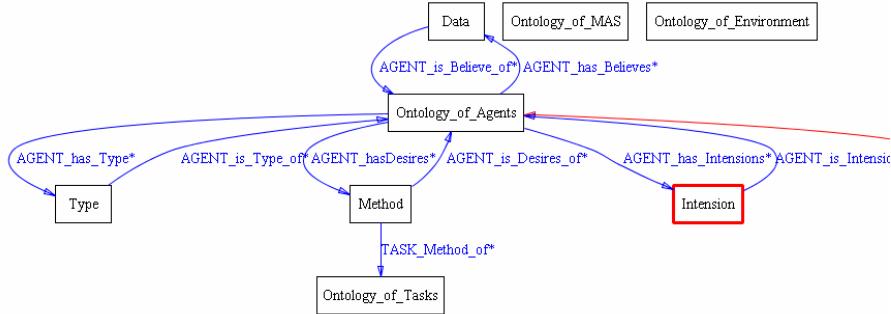
Fig. 6. The Agent Ontology

post-conditions of agent execution, explaining necessary conditions and resources for the agent successful execution, are stated. There is also a collaborator, in case if there are two or more agents needed to solve the task.

The Agent Ontology created is represented in Fig. 6 with general details that include its components. The Beliefs are referenced to Data ontology class, which determine information data resources, necessary for every agent. Desired include methods, stored in the Task Ontology, Intensions call Desires, necessary for every activity or task specification.

The component *Type* determines the common function of an agent and indicates its relation to a certain level of a MAS, and has a value in the range *DB Handling agents, Analysing agents, Evaluation agents, Simulation agents, Distribution agents*. *DBHandling agents* deal with raw initial data sources, which may be distributed and heterogeneous. Agents of this type fuse the information, clear fused meta-data from outlets, noise, deal with missing values, etc. *Analyzing agents* realize statistical analysis of meta-data and organize and execute data mining procedures (function approximation, classification, etc.). *Evaluation agents* provide model evaluation; they are called for model acceptance and decision making. *Simulation agents* organize and execute model simulation, alarm check and forecast. *Distribution agents* deliver actual information for end-users in the form of electronic documents, files, e-mails.

2.5   The Interactions Ontology

The interactions between the agents include an initiator and a receiver, a scenario and the roles taken by the interacting agents, the input and output information and a common communication language.

The private ontology is setup as:

$$OI = < Initiator, Receiver, Scenario, RolesAtScenario, InputData, OutputData, Language > \qquad (6)$$

Actually, as *Initiator* and *Receiver* we use roles, which are delegated to split the information and deliver to proper agents. A *Scenarios* corresponds to a protocol. *RolesAtScenario* is a set of roles the agents play during the interaction,
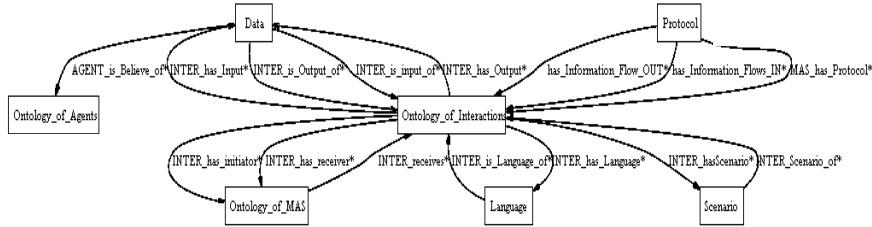
Fig. 7. The Interactions ontology

InputData and *OutputData* are represented by informational resources, read and created, respectively. *Language* determines the communication language.

The graphical representation of the protocol (depicted in Fig. 7) shows the main components, their properties and connections with other sub-ontologies.

2.6   The Distributed Meta-ontology

The Distributed Meta-Ontology is obtained as a result of private ontologies mapping, and is pooled by their common use and execution. This is achieved at step 10 of the algorithm proposed in the Introduction section. The shared ontological dimension, filled with the data, provides agents with correct addressing to proper concepts and synchronizes the MAS functionality.

The problems that appear at this stage are mostly concerned with data heterogeneity. Indeed, data might be stored in different sources, represented by various identifiers and be measured unequally. These procedures can be solved by different methods and are marked as a future step of our work.

3   Conclusions

Ontology creation may be viewed as a crucial step in MAS design as it determines the system knowledge area and potential capabilities. In this article a model of distributed meta-ontology has been proposed that serves as a framework for MAS design. Its components - private ontologies - have been described in extensive with respect to application area and in terms of used semantics. Our future work will be dedicated to the problem of sharing data in the MAS through information fusion.

Acknowledgements

References

1. Bradshaw, J.M. (1997). Software Agents. The MIT Press.
2. López-Jaquero, V., Montero, F., González, P., Fernández-Caballero, A. (2005). A multi-agent system architecture for the adaptation of user interfaces. International Central and Eastern European Conference on Multi-Agent Systems, CEEMAS 2005. Lecture Notes in Artificial Intelligence, 3690, pp. 583-586.
3. Sokolova, M.V., Fernández-Caballero, A. (2007). A multi-agent architecture for environmental impact assessment: Information fusion, data mining and decision making. 9th International Conference on Enterprise Information Systems, ICEIS 2007.
4. Guarino, N., Giaretta, P. (1995), Ontologies and knowledge bases: Towards a terminological clarification. In: Mars, N.J.I. (ed.), Towards Very Large Knowledge Bases, IOS Press 1995, pp. 25-32.
5. Samoilov, V., Gorodetsky, V. (2005). Ontology issue in multi-agent distributed learning. Autonomous Intelligent SAystems: Agents and Data Mining, AIS-ADM 2005. Lecture Notes in Computer Science, 3505, pp. 215-230.
6. Gómez-Sanz, J., Pavon, J. (2003). Agent oriented software engineering with INGENIAS. International Central and Eastern European Conference on Multi-Agent Systems, CEEMAS 2003. Lecture Notes in Computer Science, 2691, pp. 394-403.
7. Protégé. http://protege.stanford.edu/
8. OWL Web Ontology Language. (2004). http://www.w3.org/TR/owl-features/
9. International Classification of Diseases (ICD). http://www.who.int/classifications/icd/en/
10. Wooldridge, M., Jennings, N.R., Kinny, D. (2000). The Gaia Methodology for Agent-Oriented Analysis and Design. Journal of Autonomous Agents and Multi-Agent Systems, 3, 285-312.
11. ISO 14031:1999. Environmental management - Environmental performance evaluation - Guidelines
12. Weiss, G. (2000). Multi-agent Systems: A Modern Approach to Distributed Artificial Intelligence. The MIT Press.
13. Georgeff, M., Pell, B., Pollack, M., Tambe, M., Wooldridge, M. (1998). The Belief-Desire-Intention model of agency. Intelligent Agents V: Agent Theories, Architectures, and Languages, ATAL'98. Lecture Notes in Computer Science, 1555, pp. 1-10.