Modeling Ontologies for Robotic Environments

Antonio Chella
Dipartimento di Ingegneria Informatica
Università di Palermo
Viale delle Scienze, 90128 Palermo, Italy
+39.091.42667929
chella@unipa.it

Massimo Cossentino
Centro di Studi sulle Reti di Elaboratori-Consiglio Nazionale delle Ricerche
Viale delle Scienze, 90128 Palermo, Italy
+39.091.6566274
cossentino@cere.pa.cnr.it

Roberto Pirrone
Dipartimento di Ingegneria Informatica
Università di Palermo
Viale delle Scienze, 90128 Palermo, Italy
+39.091.426679
pirrone@unipa.it

Andrea Ruisi
Dipartimento di Ingegneria Informatica
Università di Palermo
Viale delle Scienze, 90128 Palermo, Italy
+39.091.6566273
andrea_ruisi@hotmail.com

ABSTRACT

On the basis of a multiple abstraction levels specification process, we developed a representational model for environmental robotic knowledge through the definition of a set of ontologies using a multi perspective approach. A general ontological model for typical indoor environments has been first developed, followed by its specialization using an implementation perspective. Actual software implementation of the ontology has been obtained via a XML-based markup language, used to build a repository for robotic environmental knowledge.

Keywords

Multi Agent Systems, Ontologies, Robotics.

1. INTRODUCTION

An ontology can be defined as a formally specified model of bodies of knowledge defining the concepts used to describe a domain and the relations that hold between them [4]. In the context of Artificial Intelligence, an ontology deals with what categories of real entities can be identified and how they are related. Knowledge-based system refer to entities and relations in the real world; to build such systems, a well-formalized global ontology is needed to specify what kinds of things exist, what their general properties are, and the interactions among them.

Furthermore, an ontology plays a central role in the creation of agent-based infrastructures to support queries over open and dynamic collections of heterogeneous and distributed information sources [4]. Here, the term agent may be intended in its wider meaning as a software agent or as a human. Two agents can not cooperate if they do not share the same language and the same semantics; an ontology becomes the common background for every kind of interaction, and the creation of an ontological model supporting terms related to human concepts allows an user friendly dialogue among humans and robots.

Our objective in this work is to produce a system for describing, upgrading and sharing knowledge about operating environments of a robot fleet. We can provide an *a priori* environmental description through the use of a specifically developed graphical tool, and robots can interact with it thanks to services provided by an agent-based map server.

In order to support this approach we structured the ontology definition process in two steps: Ontology Identification Phase (where we identify entities and their relationships in the domain) and Ontology Description Phase (where we describe entities' properties in a detailed way, taking into account implementation issues). From these two phases we produce a XML-based description of the knowledge, representing the actual software realization of our logical concepts.

2. REFERENCE FRAMEWORK

The base architecture we adopt here is based on the assumption that an Intelligent Autonomous Agent (IAA) can obtain environmental experience from three different and conceptually divided channels: (i) the metric channel, giving quantitative information about the environment (laser, sonar, odometry); (ii) the visual channel, giving snapshots of the environment (cameras); (iii) the semantic channel, giving support for the association of a *semantic* valence (a category) to spatial entities.

We use the term functional semantics to refer to different valences that a human or a robot give to a spatial entity. Sources of this channel can be human operators giving an a priori symbolic map or IAAs.

Let us suppose to have a chair in a room. From the point of view of a robot programmed just to move and avoid obstacles, the only functional semantics that is associated to the chair is that it is an obstacle to its motion. At a human eye, however, the value of functional semantics is highly increased: a chair can be used to sit on it, a chair can be moved, etc. We adopt a semantic channel to associate concepts to spatial entities without considering acquisition problems that are irrelevant for the representational task. In our approach, an environment is said to be structured when a certain number of categories of objects and places that can be encountered in it is defined.

Following the terminology used in [2], we call landmarks "those entities belonging to a sub-set of objects' categories considered significant in the environment". A landmark can be formally defined as a couple: {entity, category}. From a representational point of view, to specify the category of objects we can simply name it. When using this information, an IAA can associate proper semantic valence to objects according with its reasoning and acting abilities.

Cranefield and Purvis have investigated UML potentialities as an ontology modeling language in [4], adopting as a representative formalism a static model that is composed by class and object diagrams.

All UML class relationships and multiplicity indicators are admitted (generalization, aggregation, association), and all attributes are considered to have public visibility (because the ontology is intended to be a shared and public view of the domain). This formalism is widely adopted in our work.

UML is recognized to be a good alternative to specialized formalisms such as KIF and KL-ONE because of its large and rapidly expanding user community and because of the availability for it of a standard graphical representation. These issues are particularly meaningful when adopted in agent-based systems, where sharing ontologies is fundamental for agent communication. Also, a formally well-described ontology leads to its accessibility over project and development teams, and allows a common language for Knowledge Engineers, Software Engineers and developers.

3. ONTOLOGY IDENTIFICATION PHASE

In this phase, a meta-ontology framework is defined, to contain the application specific ontology describing the operating environments of the robot. We will use several different ontologies to represent the multiple aspects characterizing the operating environment of the robot. In other words, the ontology identification phase proposes an inner multi-perspective approach to the identification of different kinds of the environmental knowledge needed for the resolution of robot motion problems. This leads to a model (Ontology Identification Model, OIM) that encompasses a set of different ontologies, each one representative of a different *view* of the world.

OIM represents a set of ontologies (see Figure 1): each one (for example: *geometrical*, *topological*, *semantic* and so on) representing the world from a specific point of view and related to a particular inner formalism.

Ontologies can be referred to a more general category using a generalization relationship according to the dichotomy between quality and quantity.

As an example, the semantic ontology is obviously a qualitative description of the environment, while the geometrical ontology is a quantitative one. In defining each single OIM ontology, we identify *what* to capture from the real world and define the basic *entities* representing what is captured. This is completed by the identification of the entities' *properties* within the perspective proper of the adopted view.

For example, we can look more in detail to the semantic ontology as composed by *things*, that are given by the association of a category (a property) to a given entity in the environment (i.e: a table is an entity that belongs to the *table* category).

OIM is an abstract knowledge model in which quantitative and qualitative information are linked through consistency relations. Generally speaking, quantitative information is characterized by measurability, while qualitative information is not.

OIM is influenced by the SSH theory [8], particularly by its intuition of multiple interacting representations in cognitive processes, and by Maio-Rizzi's work about layered knowledge architectures [2][3]. However, we do not include behavioral levels in our model, because these (like control models) are more related to agent implementation issues than to knowledge representation.

Properly qualitative and quantitative knowledge can be present under different forms, on the basis of the different perspectives used to look at the environment. At present, qualitative OIM views are the topological one, the semantic one, and the clustered one (corresponding to successive clustering of entities in the environment). Quantitative knowledge is present as a typically robotic grid-based view, and as a purely geometric one, where entities are merely geometries of real entities.

Consistency between those kinds of information is determined by the presence, in the real world, of a causal model that acts as a bridge between them. Consistency can not be obtained without causal relations: the fact that to reach room B from room A we have to cross corridor C (a causal schema), represents the link between environment qualitative topology and the real underlying quantitative geometry. The causal model is dependent upon both kinds of information (because causality can be built only after consistency), and on their hand these are dependent on the causal model (because of their inner consistency).

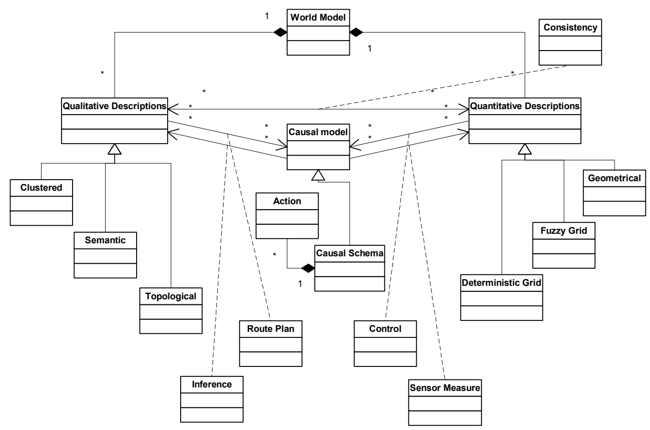

Figure 1. Ontology Identification Meta-Model

Various models [1][5][6][7] have been proposed in order to combine the points of strength and weakness of the two kinds of information (quantitative and qualitative) alone. Those models, however, are referred to canonical topological and grid-based representations only. OIM extends those approaches by combining a wider set of representations and cognitive models and by introducing successive layers abstractions and hierarchical structures. OIM's single ontologies can all be intended in a dynamical way. In general, assuming that an environment is dynamic leads to a world model that is time dependent, and this can be intended as bringing an additional *temporal channel* to our domain, that can be formally introduced by considering a timestamp as an additional property for each entity in each OIM ontology.

4. ONTOLOGY DESCRIPTION PHASE

When moving from identification to description phase, we are adopting a change in our general perspective, from the abstract and conceptual one to the implementation one. The resulting model (Ontology description model, ODM) is created at a lower abstraction level than OIM. The latter, in fact, does not show the way in which entities are effectively related to attributes specifying them.

As an example, let us suppose to look at a room from an highly abstracted level. What we see is tables, chairs, walls, and the fact that a table has four legs or five is not relevant. Looking at the same room in a geometrical way, instead, this same fact has *relevance*, while the fact that the object is a table is not.

Using an highly abstract perspective is exactly what we have done when identifying each OIM ontology. As an example, for the semantic ontology, we were simply interested in specifying the presence of objects (entities) and their category (property of the entity). The use of a geometrical ontology, instead, would have lead us to the identification of geometrical properties (the four or three legs of the table), without identifying the table.

The multi-perspective approach used in the ontology identification phase allowed us to identify concepts and properties in OIM models without the "overhead" of their full description, that becomes now necessary in an implementation perspective. When moving from OIM to ODM, we have to specify which parameters and general attributes have to be specified to obtain an *implementable* knowledge representation model.

OIM focal point is to pose emphasis on consistency (related to the presence of causal models). However, because consistency is essentially determined from *interpretation* of the world (that is: from inferential processes), when moving to an implementation perspective, it is something that has to be considered external from the representational model, i.e. obtained by techniques such as those proposed in [5][6]. Therefore, ODM (see Fig. 2) does not include layers corresponding to OIM causal schemas, and turns its attention to purely representation issues.

Each OIM ontology relates to an ODM *layer*, obtained applying a finer-grained specification to it. The presence of a semantic acquisition channel, however, allows some ODM layers to be the combination of OIM's semantic ontology with other ones.

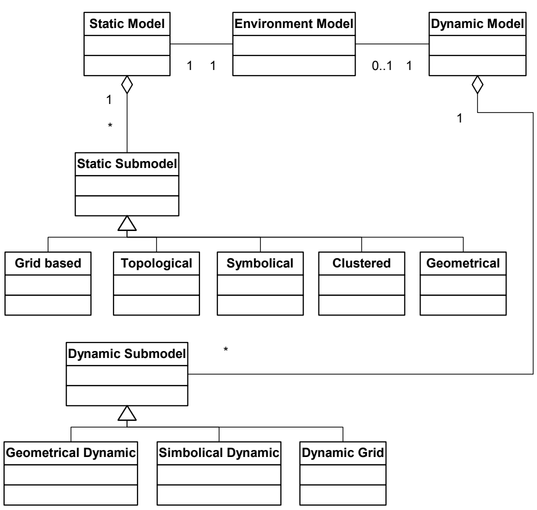

Figure 2: Ontology Description Model

As an example, ODM's symbolical layer is obtained by augmenting the geometrical OIM ontology with its semantic one. ODM topological layer, also, is a topological graph in which nodes are not merely free zones, but representative of abstractions such as rooms and corridors, obtained by augmenting significant places with semantics obtained from the semantic acquisition channel. As we formerly mentioned, in our representational task this is simply obtained through naming entities in their categories. An IAA can use this information to properly behave according with its acting abilities.

For example, in a museum guide robot, knowing to be into a corridor could activate a "follow corridor" behavior, while knowing to be into an exposition room could activate a "wait for attendant" behavior.

ODM, in other words, avoids OIM's ontological dichotomy between qualitative-semantic and quantitative-geometric information, augmenting quantity with quality and vice versa when possible. On the other hand, in order to solve the implementation problems posed by a fully adaptive map, and following the paradigm of decoupling static and dynamic knowledge [11], ODM introduces a dichotomy between static and dynamic knowledge by encompassing two parallel, not interleaved ontologies related to static and dynamics entities. Static knowledge is the result of a prior exploration phase or of the availability of a hand-crafted map. This choice is meaningless with regard to the development of a *representational* model. Corresponding static layers are the topological one, the symbolic one, the geometrical one, the grid-based one and the clustered one. Dynamic layers are obtained by replicating geometrical, symbolical and grid layers augmenting their content with timestamps.

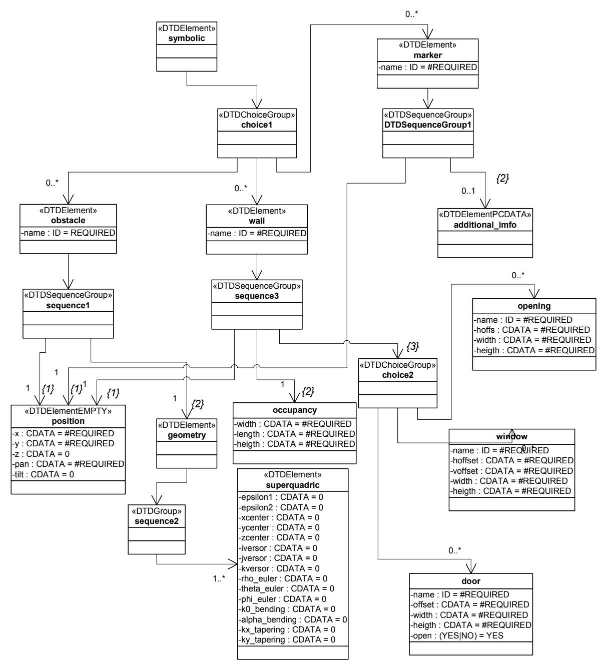

Figure 3. Static symbolic layer DTD

All ODM layers have been defined at a level of detail that is able to capture inner formalisms of each, but at the same time they remain independent from the particular software realization of those formalisms. The process of creating successively finer-grained ontological models on different abstraction levels is inspired from the well established successive abstraction process used in software engineering.

5. XML REALIZATION

After ODM layers description, we have formally specified models for each layer, but we still do not have any software *realization* of them. ODM models can be realized by building, i.e. Java or C++ classes. We have developed an XML-based ODM realization that has been used as data storage format for a software map server.

XML has recently been proposed [12] as a good candidate for knowledge representation in AI. Some points that justify the use of XML in this sense are: (i) portability; (ii) support for data structuring and encapsulation; (iii) support for hierarchical structures.

XML is receiving increasing support by CASE tools producers, both in the direction of using XML as a cross-platform data exchange format (XMI) for UML diagrams, and in the direction of supporting XML modeling via UML. We have adopted Rose™ conventions exposed in [9][10] to model DTDs using UML and allow their eventual migration to Schemas.

XML DTDs have been developed to represent all ODM layers. As an example, the UML DTD model for the symbolic layer is shown in Fig. 3. This DTD contains all the information we identified in the corresponding ODM submodel, though rearranged using XML rules, and it defines syntax and structure of a chunk of the language we used to write files to physically store the environmental knowledge base (KB).

6. APPLICATIONS

A FIPA agent based map server has been coded to allow publication of environmental knowledge in a multi-robot operating environment. The server supports queries on the shared knowledge base and allows operating robots to update it when encountering new obstacles. This is obtained by including DOM parsers into the map server to parse the XML documents constituting the KB and to extract information used to answer to particular robots requests.

The presence of dynamical layers in the shared Knowledge Base allows a robot to obtain not only the a priori corpus of knowledge, but to augment it with the additional knowledge obtained by previous explorations by other robots.

7. CONCLUSIONS

We have described the process of creation of a spatial ontology model for indoor office environments. The obtained model, ODM, allows a layered structure for environmental knowledge representation, that supports multiple abstraction levels, and that is based upon integration of qualitative and quantitative knowledge.

We have developed an XML realization of ODM creating a new markup language for environmental knowledge representation.

A graphical editor allowing an *a priori* specification of an indoor office environment map has also been created (see Fig. 4). The editor explicitly supports the creation of ODM symbolical layer, and implements the exportation of files in the ad hoc built XML based markup language.

These files have been effectively used as data storage for a FIPA agent-based map server in order to publish knowledge and share it in a multi-robot environment.

The map server supports queries upon the Knowledge Base, allows knowledge update, and is able to derive some knowledge layers from other ones (i.e. grids from symbolical maps).

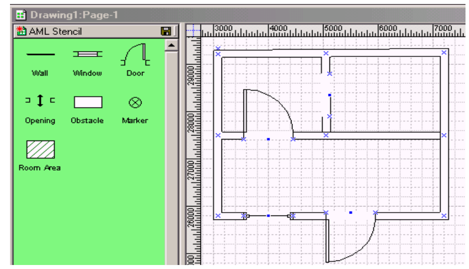

Figure 4. A snapshot from the editor

8. REFERENCES

[1] Thrun, S. and Bucken, A. Integrating grid-based and topological maps for mobile robot navigation. In Proc. of the 13th Conference on Artificial Intelligence (Portland, Oregon, August 1996)

[2] D. Maio and S. Rizzi, "Knowledge architecture for environment representation in autonomous agents", Proc. ISCIS VIII, Istanbul, 1993.

[3] D. Maio and S. Rizzi. A Multi-Agent Approach to Environment Exploration. Int. Journ. Cooperative Information Systems, 5(2-3):213--250, 1996.

[4] S. Cranefield and M. Purvis. UML as an ontology modelling language. In Proc. of the Workshop on Intelligent Information Integration, 16th International Joint Conference on Artificial Intelligence (IJCAI-99), 1999

[5] T. Duckett, A. Saffiotti, Building globally consistent gridmaps from topologies, in: Proceedings of the Sixth International IFAC Symposium on Robot Control (SYROCO), Wien, Austria, 2000

[6] Fabrizi, E. and A. Saffiotti (2000). Extracting topology-based maps from gridmaps. In: IEEE Intl. Conf. on Robotics and Automation (ICRA). San Francisco, CA.

[7] S. Thrun, J.-S., B. Kuipers et al. Integrating topological and metric maps for mobile robot navigation: A statistical approach. In Proceedings of AAAI-98. AAAI Press/The MIT Press, 1998

[8] B. J. Kuipers. The spatial semantic hierarchy. Artificial Intelligence Journal, 119:191--233, 2000

[9] G. Booch et al. UML for XML Schema Mapping Specification.. Rational Software white paper, 1999

[10] Migrating from xml dtd to xml schema using uml. Rational Software White Paper, 2000

[11] Fox D., Burgard W., Thrun S. Probabilistic methods for mobile robot mapping. In Proc. of the IJCAI-99 Workshop on Adaptive Spatial Representations of Dynamic Environments, 1999

[12] Popov, D. Using XML as the core language for Knowledge representation in AI. Proceedings of the 2nd International workshop on Computer Science and Information Technologies (CSIT'2000). Ufa, Russia, 2000