An Ontology for Collaborative Tasks in Multi-agent Systems

Daniela Schmidt [1], Rafael H. Bordini [1], Felipe Meneguzzi [1], Renata Vieira [1]

[1]Postgraduate Program in Computer Science
School of Informatics (FACIN)
Pontifical Catholic University of Rio Grande do Sul - PUCRS
daniela.schmidt@acad.pucrs.br
{rafael.bordini,felipe.meneguzzi,renata.vieira}@pucrs.br

Abstract. *This paper proposes an ontology for task representation and inference. The ontology was developed to support reasoning about tasks, such as task recognition and relocation. Our proposal is formalized in OWL (Web Ontology Language) and SWRL (Semantic Web Rule Language). We show one scenario to exemplify reasoning situations based on the axioms and rules of our ontology. This knowledge-level representation of tasks can be explored to support reasoning about activities for groups of people. The knowledge asserted and inferred in the ontology is useful in multi-agent systems to enhance agent coordination and collaboration through reasoning over tasks. An evaluation of the proposed ontology is presented.*

1. Introduction

Ontology and agent-based technologies have received significant attention, but little focus has been given in their integrated use [Hadzic et al. 2009a]. Ontologies allow the sharing of common knowledge among people and software agents. For multi-agent systems (MAS) development, ontologies are key for the common understanding and reuse of domain knowledge.

Our work aims to integrate an ontology that represents collaborative tasks with a multi-agent framework to enable queries and inferences about shared tasks. It provides knowledge about tasks for the execution of plan recognition, and for the negotiation and relocation of tasks. In this paper, we describe our task ontology in detail, as well as its integration in a multi-agent system. Based on the proposed ontology, we present an application in *health care* which consists of a family group that takes care of an elderly person. The elderly needs constant monitoring to perform his daily tasks, so it is necessary that the group collaborates in the distribution of tasks related to his care.

The paper is organized as follow: Section 2 introduces main aspects of ontologies, and technological alternatives for their representation such as OWL and SWRL. Section 3 describes related work. Next, in Section 4 we present our proposed ontology. Section 5 presents the ontology evaluation. Section 6 shows some final remarks and future directions to extend our proposed ontology.

2. Ontology

Ontology was originally the philosophical study of reality to define which things exists and what we can say about them. In computer science, ontology is defined as an "explicit specification of a conceptualization" [Gruber 1993]. A conceptualization stands for an abstract model of some world aspect that specifies properties of important concepts and relationships. Therefore, ontologies are knowledge representation structures composed of concepts, properties, individuals, relationships and axioms. A *concept* (or class) is a collection of objects that share specific restrictions, similarities or common properties. A *property* expresses relationships between concepts. An *individual* (instance, object, or fact) represents an element of a concept; a *relationship* instantiates a property to relate two individuals; and an *axiom* (or rule) imposes constraints on the values of concepts or individuals normally using logic languages (which can be used to check ontological consistency or to infer new knowledge).

Nowadays there are prominent ontology languages, such as OWL (Web Ontology Language) [Bechhofer et al. 2004], which is a semantic web standard formalism to explicitly represent the meaning of terms and the relationships between those terms. OWL is a language for processing web information that became a W3C recommendation in 2004 [Bechhofer et al. 2004]. Ontologies empowers the execution of semantic reasoners, such as Pellet [Sirin et al. 2007]. Semantic reasoners provide the functionalities of *consistency checking*, *concept satisfiability* and *classification* [Sirin et al. 2007]. In other words, reasoners infer logical consequences from a set of axioms, which in the current technology can be done, for example, through the application of the rules coded in SWRL (Semantic Web Rule Language) [Horrocks et al. 2004].

Ontologies and rules are established paradigms in knowledge modeling that can be used together. SWRL is a rule extension of OWL that adheres to the open-world paradigm [Horrocks et al. 2004]. SWRL adds to the OWL's expressiveness by allowing the modeling of certain axioms which lie outside the capability of OWL; including an abstract syntax for Horn-like rules in ontologies. The rules are defined as an implication between an antecedent (body) and a consequent (head). The intended meaning can be read as: whenever the conditions specified in the antecedent hold, then the conditions specified in the consequent must also hold (be true). A SWRL rule [Horrocks et al. 2004] has the form of $antecedent \Rightarrow consequent$ where both antecedent and consequent are conjunctions of atoms written as $a1 \wedge ... \wedge an$. Variables are prefixed with a question mark (*e.g.*, `?x`), and an atom may be unary (*e.g.*, a class expression) or binary (*e.g.*, an object property), and arguments in atoms can be individuals or data values. In this syntax, a rule asserting that the composition of parent and brother properties implies the uncle property can be written as [Horrocks et al. 2004]: $parent(?x, ?y) \wedge brother(?y, ?z) \Rightarrow uncle(?x, ?z)$.

This section provided a background on ontology technologies employed in this work. Our proposal is formalized in OWL and SWRL, and applies the semantic reasoner Pellet [Sirin et al. 2007] for executing inferences over the modeled concepts, properties, individuals, axioms and rules. Next, we present previous approaches to task ontologies.

3. Related Work

This section describes previous work that use ontologies for activity representation and recognition. Activity modeling in ontologies consists in defining formal semantics for human tasks by means of the operators in ontological languages. Then, ontological reasoning can be used to recognize that the user is performing a certain activity starting from some facts (*e.g.*, sensor data, location of persons and objects, properties of actors involved) [Riboni and Bettini 2011b].

Smith *et al.* [Smith et al. 2011] propose an ontology containing rules to reason about task characteristics to enable an effective coordination of activities. To implement such mechanism, agents need to reason and communicate about activities, resources and their properties. The semantics for the coordination mechanisms are given by rules coded in an ontology. The idea is to enable agents to reason about the relationships of their activities with the activities of other agents [Smith et al. 2011].

The OWL-T ontology [Tran and Tsuji 2007] was designed to allow the formal and semantic specification of tasks using a high-level knowledge abstraction. The *Task* class is the central concept in OWL-T, and can be hierarchically decomposed into simple or complex tasks. According to Tran and Tsuji [Tran and Tsuji 2007], an example of complex task is planning a trip, that requires the sub-tasks of book flight, book hotel and rent car.

Riboni and Bettini [Riboni and Bettini 2011a] propose an ontology of activities for task representation that combines ontological reasoning with statistical inference to enable activity recognition. Their solution uses statistical inference on raw data retrieved from body-worn sensors (*e.g.*, accelerometers) to predict the most probable activities. Then, symbolic reasoning refines the results of statistical inference by selecting the set of possible activities performed by a user based on the context. The ActivO (ontology for activity recognition) contains a set of activities, as well as context data that can be useful to recognize them. The set of activities and context data defined in ActivO is non-exhaustive, but it is claimed that it can be used to model many pervasive computing scenarios.

The work on [Chen et al. 2012] introduces a knowledge-driven approach to activity recognition and inferencing based on multi-sensor data streams in smart homes. The ontology represents the correlated domains of smart homes contexts and Activities of Daily Living (ADL). For example, the task of brushing teeth normally takes place in the bathroom twice a day and usually involves using toothpaste, toothbrush and water, which is the context for this activity. Contextual information is obtained by sensors that are linked to physical and conceptual entities such as objects, locations, and states. In addition to the ontology of contexts, an ontology for activity recognition was created to model an activity hierarchy in which each class denotes an ADL type. ADL ontologies can be viewed as activity models that establish links between activities and contextual information.

Garcia *et al.* [Garcia et al. 2013] propose one approach based on ontologies to solve problems related with resource sharing in pervasive environments. The ontological model is composed by a set of ontologies that represent the elements involved in a collaborative environment. Ontologies refers to types of managed resources (human, physical and virtual) and other characteristics such as environment and organizational aspects. This set of ontologies is part of RAMS architecture (Resource Availability Management Service). According to the authors, a set of ontologies is better than one unique ontology and these proposed ontologies can be extend by adding new concepts.

Bae [Bae 2014] presents and approach to activities of daily living (ADL) recognition called RADL (Recognizing Activities of Daily Living). RADL is a system that

Table 1. Comparing the domains of related ontologies

Ontology	Scope	Main Concepts
Coordination ontology [Smith et al. 2011]	Tasks and their positive/negative relationships to enable agent coordination.	**Activity**, interdependence types, agent, resource, operational relationship, and coordination rule.
OWL-T [Tran and Tsuji 2007]	Business processes using high-level abstraction.	**Task** (divided into simple and composite), input, output, pre/post conditions, preferences, effects.
ActivO [Riboni and Bettini 2011a]	Concepts to help in activity recognition, specially in the health care field.	**Activity**, artifact, communicationRoute, **person**, symbolicLocation, timeExtend.
SH and ADL [Chen et al. 2012]	Daily-living activities in the smart home domain.	**Activity**, actor, **location**, resource, environment, entities, duration, goal, effects, conditions, time.
RAMS [Garcia et al. 2013]	Represent actions that users can execute about resources.	Process, **activity**, group, role, calendar, time interval, human resource.
RADL [Bae 2014]	Represent activities of daily living related to elderly person in a smart home environment.	**Person**, **activity**, **location**, device, device status, sensor, sensor status, service, daily life service, message service, safety service.

detects and monitors ADL's standards for smart homes equipped with sensors. RADL is exemplified in a smart home scenario where one elderly person lives alone. The ontology proposed by the author is able to reason about ADL's standards and provide semantic discovery of locations, devices, activities and other relevant information. The ontology is divided in three parts. The first represents concepts about daily life services like: air conditioner on or off and open or closed window for instance. The second represents safety services like: fire alarm activated. And the third part represent messages services like: sleeping message, wake up message and so on are described.

As we see, semantic representations of tasks through ontologies are starting to appear as promising research directions. These representations enable agents to reason about tasks, for example, to implement activity recognition approaches. In Table 1 we compare the related work presented in Section 3, highlighting the main concepts of each ontology. In our proposed ontology, we reused the most common concepts in the domain (as highlighted in bold in Table 1). We then included new aspects to those found in previous works, since our focus was on the representation of tasks performed collaboratively (to be presented in the next section). No previous work was found with such orientation.

4. Proposed Task Ontology

This section presents our proposed ontology, the integration of our ontology in the multi-agent system and its application. The goal is to represent the knowledge of where and when the tasks might occur, who is responsible for them and what are the tasks particularities. Based on this ontology representation, we may use logical rules and apply semantic reasoners to infer new knowledge about tasks that may be useful for agent programmers willing to implement task reasoning mechanisms, such as techniques of task recognition, task negotiation and task relocation.

The *Task* is the main concept in the ontology; it represents an activity that is executed by one or more people. We can also say that the execution of a *Task* may happen in a particular location and time, and normally involves an object. Therefore, the main and most generic concepts of the proposed task ontology are: *Task*, *Person*, *Location*, *Object*, *TimeInterval* and *TaskPurpose* (see Figure 1 a)).

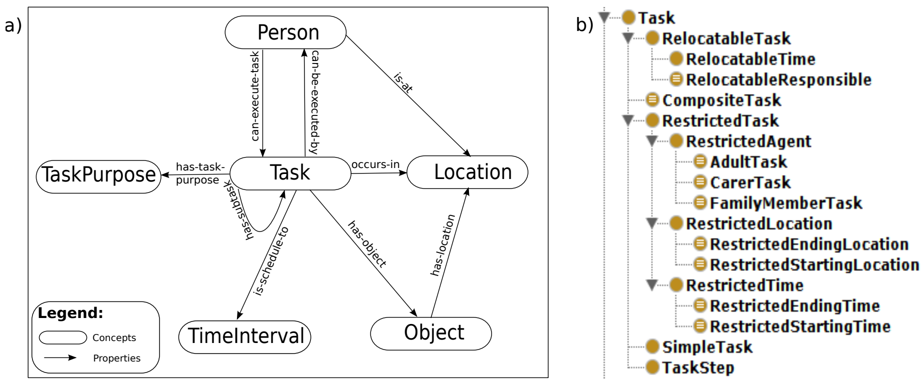

Figure 1. a) Task ontology main concepts. b) Taxonomy of task concepts

Collaborative tasks may have restrictions as to who can execute them, when and where they occur. Then, to address these issues our ontology specialized the task concept (according to Figure 1 b)). In our ontology, the concepts were defined based on restrictions and other logical characteristics related to collaborative tasks. For example, the *CompositeTask* concept is equivalent to a task that has sub-task. This concept definition used the existential quantifier, as follows:

$$CompositeTask \equiv \exists has\text{-}subtask.Task$$

The *RestrictedTask* concept is subdivided in three kinds of restrictions that are presented bellow:

RestrictedAgent: in this case, the concepts can be used to define features that agents or people may to perform certain tasks. In our ontology there are three concepts to specify restrictions regarding agents. They are: *(i)* the concept *AdultTask* restricts the tasks that may be performed only by adults, like driving a car, for instance. *(ii)* the concept *CarerTask* restricts the tasks that may be performed only by carers, *(iii)* the concept *FamilyMemberTask* that represents tasks that can only be performed by family members. The setting of restrictions on concepts specified above is as follows:

$$AdultTask \equiv \forall \mathit{can\text{-}be\text{-}executed\text{-}by}.Adult$$

$$CarerTask \equiv \forall \mathit{can\text{-}be\text{-}executed\text{-}by}.Carer$$

$$FamilyMemberTask \equiv \forall \mathit{can\text{-}be\text{-}executed\text{-}by}.FamilyMember$$

Note that the restrictions regarding agent may vary according to the application, in the case of our application carer and family members are types of agents, but in other applications these types may differ. More details on the application is described in the sequence.

RestrictedLocation: a task can be classified according to the location where it occurs. To represent these restrictions the concepts *RestrictedStartingLocation* and *RestrictedEndingLocation* were included. One task instance belongs to any of these concepts if it has the property *possible-starting-location* or *possible-ending-location* respectively. This definition is specified as follows:

$$RestrictedStartingLocation \equiv \exists \mathit{possible\text{-}starting\text{-}location}.Location$$

$$RestrictedEndingLocation \equiv \exists \mathit{possible\text{-}ending\text{-}location}.Location$$

RestrictedTime: similar to the location constraints, a task can have time restrictions. The concepts *RestrictedStartingTime* and *RestrictedEndingTime* are used when a task has restrictions regarding the start or end time of it execution (a physiotherapy session, for instance). A task instance is classified as *RestrictedTime* if it has any restrictions through the properties *has-beginning* or *has-end* respectively.

In addition to the specializations and restrictions, our task ontology allows agents to negotiate about relocation of tasks. To provide this information, the concept *Task* has the sub-concept *RelocatableTask* whose function is to describe the possibilities of relocation in terms of time and responsible for the execution of a task.

RelocatableTask: this concept describes the relocation task possibilities. It is divided in two sub-concepts called *RelocatableResponsible* and *RelocatableTime*. The first refers to a task instance which has the property *can-be-relocated-to*. Rules in SWRL were created to define who is able to perform each task. For instance, some tasks can be executed only by adults. Already, the temporal relocation can occurs when one task instance is not *RestrictedTime*. The following we define the concept *RelocatableResponsible* and the rules in SWRL that allow inferences instances of people for that a task can be relocated.

$$RelocatableResponsible \equiv \exists \mathit{can\text{-}be\text{-}relocated\text{-}to}.Person$$

$$Task(?x) \wedge Person(?y) \Rightarrow \mathit{can\text{-}be\text{-}relocated\text{-}to}(?x, ?y)$$

$$AdultTask(?x) \wedge Adult(?y) \Rightarrow \mathit{can\text{-}be\text{-}relocated\text{-}to}(?x, ?y)$$

Furthermore, a *Task* may contain restrictions based on locations where it can happen, what objects are related, and so on (see Figure 1). These aspects are addressed below.

The *Location* concept represents physical places where *Task* instances happens.

Table 2. Domain and range of task ontology properties

Domain	Object Property	Range
Task	has-subtask	Task
Task	is-part-of	Task
Task	can-be-execute-by	Person
Task	can-be-relocated-to	Person
Task	has-feature	Feature
Task	has-object	Object
Task	has-task-purpose	TaskPurpose
Task	occurs-in	Location
Task	is-schedule-to	TimeInterval
Person	can-execute-task	Task
Person	is-at	Location
Object	has-location	Location
Object	is-used-for	Task
Patient	has-carer	CarerGroup
CarerGroup	carer-of	Patient

Location has two sub-concepts to differentiate between internal and external locations. The relationship of the task concept and location is given by the property *occurs-in*.

The *Person* concept represents the group of people.The relationship between Task and Person occurs through the property *can-be-executed-by*. In order to specialize the ontology according to the application, the *person* concept has two sub-concepts called *CarerGroup* and *Patient*. The first is divided into *Carer* and *FamilyMember*. Instances of carer group are responsible to take care of patient instances. To provide this relationship, we create the object property *has-carer* (see following).

$$Patient \equiv \exists has\text{-}carer.CarerGroup$$

The *Object* concept represents the objects involved in the task execution. The *TaskPurpose* concept represents the specialization of tasks (*e.g. entertainment, hygiene, etc.*). The relationship between this concept and the Task concept occurs through the property *has-task-purpose*.

The *TimeInterval* concept includes information about temporal restrictions. This concept is related to task concept through of the property *is-schedule-to*. *TimeInterval* has three sub-concepts called *ClosedInterval*, *LeftClosedInterval* and *RightClosedInterval*. A task instance is classified with restricted time according to follow rules.

$$ClosedInterval \equiv \exists has\text{-}beginning.string \wedge \exists has\text{-}end.string$$

$$LeftClosedInterval \equiv \exists has\text{-}beginning.string$$

$$RightClosedInterval \equiv \exists has\text{-}end.string$$

The relationship between the concepts occurs through of properties. In Table 2 we present the main properties of our proposed ontology.

4.1. Ontology Integration in the Multi-agent System

The task ontology is part of a multi-agent framework and allows queries and inferences about tasks in a multi-agent environment. It provides knowledge to plan recognitions and task negotiation and relocation. Figure 2 shows a view of the application of our ontology in the multi-agent framework (more details about the framework and the artifact which allows agents to interact with ontologies can be found in our other papers [Freitas et al. 2015], [Panisson et al. 2015]).

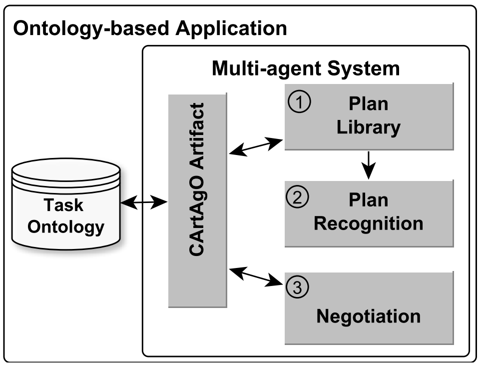

Figure 2. Application of task ontology in the multi-agent framework

The *Plan Library (1)* can be created by instances and restrictions of tasks modeling in the ontology. Consider a plan called *prepare-meal*. In the ontology, there is a instance of *CompositeTask* called prepare-meal. This instance is classified as a top-level plan in the plan library and is decomposed with the sub-tasks instances like: prepare-breakfast, prepare-lunch and so on. The hierarchy between the top-level plan and the sub tasks occurs through the has-subtask property that allows differentiate between sequence or decomposition. In Figure 3 we show a plan makes by ontology instances.

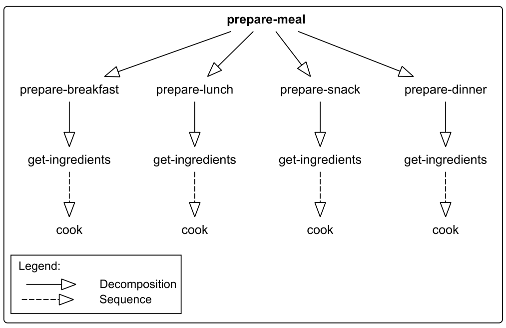

Figure 3. Plan *prepare-meal*

The *Plan Recognition (2)* module is responsible to recognize the agent plans. For this, we used a plan recognizer developed by Avrahami-Zilberbrand and Kaminka [Avrahami-Zilberbrand and Kaminka 2005]. The plans are based on the structure specified in the plan library that was generated from the ontology. In this context, ontology's role is to provide subsidies for the construction of plans and a set of features that help the plan recognizer to identify what task is running, if it will fail or if the plan needs to start a process of negotiation.

The *Negotiation (3)* module performs queries in the ontology to verify two types of information. *(i)* when agents need to know if a task is relocatable temporally and *(ii)* when agents need to know if a task can be relocated to another member of the group. In the first case, it is checked if the task instance belongs to the concept *RelocatableTime*, if the answer is positive, the task can be relocated temporarily. In the second case, the agent asks the ontology if one task instance has the property *can-be-relocated-to*. This property relates a task instance that can be relocated to other group members who are able to execute it. In this case, the application proceeds to relocation between the members related to the task. The existence of this property takes into account constraints such as tasks that can be performed only by adults, for example.

4.2. Task Ontology Application

According to Bae [Bae 2014], recognition and monitoring of daily activities can provide important opportunities for applications whose focus is the care of the elderly. Moreover, according to the author, the correct way to represent knowledge in household, including behavioral rules systems, is through concepts and information modeled in ontologies. In the other words, ontologies provide readable and understandable knowledge for machine and perform an important role in knowledge representation, sharing and data management, information retrieval, among others. Computational agent-based systems are used to support distributed computing, dynamic information retrieval, automated discovery of services, etc. Therefore, ontologies and agent-based systems are two different but complementary technologies where the ontology is responsible for providing knowledge to the system while the agents provide dynamism that the system needs [Hadzic et al. 2009b].

Our application corresponds to a family group with an elderly man living alone called *Joao*. He has health problems and needs constant monitoring to perform their daily tasks. *Joao* has two children called *Paulo* and *Stefano*. *Paulo* lives next door with his wife *Jane* and their two children (*Pedro* 12 years old and *Maria* 14 years old). *Stefano* lives in the same city, but about 10 kms away from *Joao*'s house. To help with daily tasks, *Joao* has two professional carers that help him (one for the day and another at night). *Joao* has a routine of activities that includes walk in the park, physiotherapy, stimulation activities (memory games, for instance), as well as feeding and medicines at specific time. The group's tasks are related with the care of the elderly. Whereas the elderly needs to follow up full-time, then, the group established a routine tasks that starts when the elderly wakes up and extend across the rest of the day. Thus the ontology is designed to represent all aspects of the tasks of daily living (ADL) of elderly and its relationship with the other members of the group.

Each application will require a specific instantiation, we instantiate a health care group. In Figure 4 it is possible to see one instance of task concept and their inferences. This instance is the same demonstrated in the plan library (Figure 3). In this example, you can see the specialization of the *Task* concept, as the relationship that it has with the concepts *Person* (through the property *can-be-executed-by*) and *Location* (*occurs-in*). One of inferences refers to the type *CompositeTask* that occurs because the concept

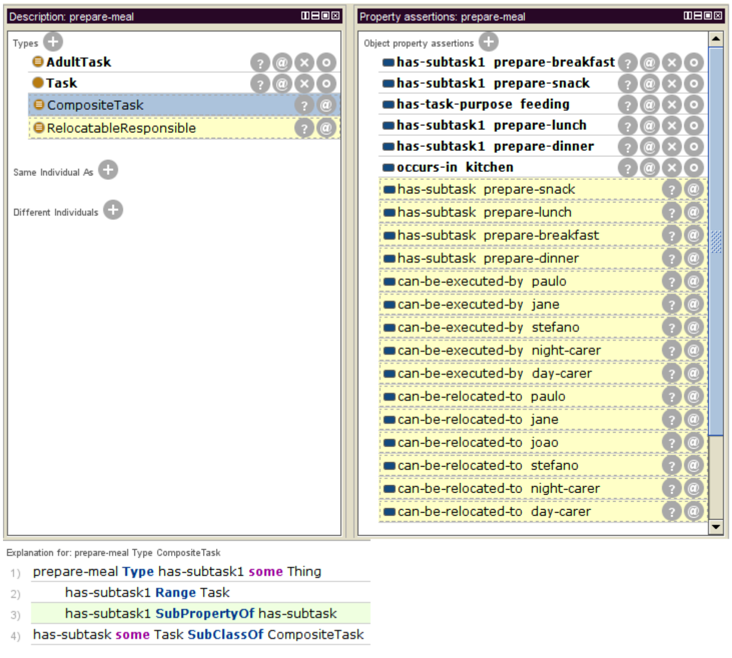

Figure 4. Instance of task concept (*prepare-meal*)

CompositeTask is equivalent to a task that has sub-tasks (the explanation about why the reasoner infers the *CompositeTask* concept can be visualized in Figure 4).

5. Task Ontology Evaluation

We evaluated the task ontology with the group of people that used the ontology as a resource for the Multi-Agent application development. Our goal was to identify if the ontology was considered suitable for their needs in the development. The subjects were: two phd candidates, two masters candidates and one developer. The evaluation consisted of a set of open and closed questions. The closed questions are based on the Likert scale of five points, regarding the following points: *Q1: Do you consider the concepts represented in the ontology relevant for the collaborative multi-agent application?*; *Q2: Do you consider the terminology adequate?*; *Q3: Do you consider that the ontology provides subsidies for the negotiation process?; Q4: Do you consider that the ontology representation is adequate for the plan library generation?*. Table 3 presents the answers. All participants considered it relevant and the terminology was evaluated as adequated. Similarly, the ontology was considered efficient to provide information for negotiation and to the plan library modelling.

In the open questions, the participants could suggest changes in the task concepts (*Q5*), none was suggested. The next questions (*Q6* and *Q7*), ask about advantages and disadvantages of using ontologies in multi-agent systems. Regarding the advantages they mentioned: *(i)* possibility of knowledge reuse; *(ii)* applications development become independent of the domain which may vary according to the ontology and; *(iii)* inferences of new knowledge. Regarding the disadvantages they mentioned: *(i)* performance of the application and *(ii)* the developer needs to know about ontologies.

Table 3. Evaluation of proposed ontology

	Question 1	Question 2	Question 3	Question 4
Strongly Agree	4	4	3	3
Agree	1	0	1	2
Undecided	0	1	1	0
Disagree	0	0	0	0
Strongly Disagree	0	0	0	0

According to the answers, we consider our ontology adequate for the representation of collaborative tasks in multi-agent systems. Issues regarding performance still must be investigated.

6. Final Remarks

This paper presented a new task ontology, based on an extensive literature review of how ontologies are being used as semantic models for task representation and reasoning. We explained how the concepts, properties, and rules were defined in our ontology, and then we exemplified the kind of inference processes that it allows. We have shown how the model allows for knowledge inference about tasks that may be used in the coordination of activities in groups of agents.

The full version of our ontology consists of 34 concepts, 31 object properties, 4 date properties and 73 instances. It allows for different inferences as *(i)* classification tasks into simple and composite, *(ii)* classification tasks that can be performed only by adults, *(iii)* relocation responsible, *(iv)* time constraints, *(v)* classify people as adults or not, among others.

This work is part of a greater research project involving various AI techniques such as MAS development, plan recognition, negotiation, among others. Our focus is the representation of collaborative tasks to provide the required knowledge to other modules developed in the project (more details about the project can be found in our other papers [Freitas et al. 2015], [Panisson et al. 2015]). The ontology was modeled with a level of abstraction that allows it to be reused by other applications whose focus is the representation of collaborative tasks.

The ontology was instantiated allowing reasoning and querying. After we evaluated the ontology by means of a questionnaire that was answered by developers that used the ontology as a source of knowledge for their modules. As future work, we intend to expand the instantiation of tasks to new scenarios of family care and reuse the ontology in the development of other group applications.

Acknowledgment

Part of the results presented in this paper were obtained through research on a project titled "Semantic and Multi-Agent Technologies for Group Interaction", sponsored by Samsung Eletrônica da Amazônia Ltda. under the terms of Brazilian federal law No. 8.248/91.

References

Avrahami-Zilberbrand, D. and Kaminka, G. A. (2005). Fast and complete symbolic plan recognition. In *IJCAI-05*, pages 653–658.

Bae, I.-H. (2014). An ontology-based approach to adl recognition in smart homes. *Future Gener. Comput. Syst.*, 33:32–41.

Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D. L., Patel-Schneider, P. F., and Stein, L. A. (2004). OWL Web Ontology Language Reference. Technical report, W3C, http://www.w3.org/TR/owl-ref/.

Chen, L., Nugent, C. D., and 0001, H. W. (2012). A knowledge-driven approach to activity recognition in smart homes. *IEEE Trans. Knowl. Data Eng.*, 24(6):961–974.

Freitas, A., Panisson, A. R., Hilgert, L., Meneguzzi, F., Vieira, R., and Bordini, R. H. (2015). Integrating ontologies with multi-agent systems through CArtAgO artifacts. In *2015 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT*.

Garcia, K., Kirsch-Pinheiro, M., Mendoza, S., and Decouchant, D. (2013). An Ontological Model for Resource Sharing in Pervasive Environments. In *Procedings of International Conference on Web Intelligence (WI) and Intelligent Agent Technology (IAT)*, IEEE/WIC/ACM, pages 179–184.

Gruber, T. R. (1993). Toward principles for the design of ontologies used for knowledge sharing. In *International Journal of Human-Computer Studies*, pages 907–928. Kluwer Academic Publishers.

Hadzic, M., Wongthongtham, P., Dillon, T., and Chang, E. (2009a). *Ontology-based Multi-Agent Systems*. Springer, Berlin Heidelberg, Germany, 1st edition.

Hadzic, M., Wongthongtham, P., Dillon, T., and Chang, E. (2009b). Significance of ontologies, agents and their integration. In *Ontology-Based Multi-Agent Systems*, volume 219 of *Studies in Computational Intelligence*, pages 93–110. Springer Berlin Heidelberg.

Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosof, B., and Dean, M. (2004). SWRL: A Semantic Web Rule Language combining OWL and RuleML. W3c member submission, World Wide Web Consortium.

Panisson, A. R., Freitas, A., Schmidt, D., Hilgert, L., Meneguzzi, F., Vieira, R., and Bordini, R. H. (2015). Arguing about task reallocation using ontological information in multi-agent systems. In *12th International Workshop on Argumentation in Multiagent Systems (ArgMAS)*.

Riboni, D. and Bettini, C. (2011a). Cosar: Hybrid reasoning for context-aware activity recognition. *Personal Ubiquitous Comput.*, 15(3):271–289.

Riboni, D. and Bettini, C. (2011b). OWL 2 modeling and reasoning with complex human activities. *Pervasive Mob. Comput.*, 7(3):379–395.

Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A., and Katz, Y. (2007). Pellet: a practical OWL-DL reasoner. *Web Semant.*, 5(2):51–53.

Smith, B. L., Tamma, V. A. M., and Wooldridge, M. (2011). An ontology for coordination. *Applied Artificial Intelligence*, 25(3):235–265.

Tran, V. X. and Tsuji, H. (2007). OWL-T: A Task Ontology Language for Automatic Service Composition. In *ICWS*, pages 1164–1167. IEEE Computer Society.