Applying Ontologies and Agent Technologies to Generate Ambient Intelligence Applications

Artur Freitas, Daniela Schmidt, Alison Panisson,
Rafael H. Bordini, Felipe Meneguzzi, and Renata Vieira

Pontifical Catholic University of Rio Grande do Sul - PUCRS
Postgraduate Programme in Computer Science, School of Informatics (FACIN)
Porto Alegre - RS - Brazil
{artur.freitas,daniela.schmidt,alison.panisson}@acad.pucrs.br,
{rafael.bordini,felipe.meneguzzi,renata.vieira}@pucrs.br

Abstract. The specification of agent systems comprises different dimensions normally defined using distinct formalisms. Since this lack of a uniform representation makes harder to express how each level affects the others, we propose an ontology to integrate the formalisms that originally cover a single multi-agent system dimension. In doing this, we align semantic technologies and knowledge representation for agents, environments, and organisations providing agent-oriented designers with a unified approach for developing complex systems. In our approach, we represent the abstractions typical of each multi-agent system dimension as an ontology, and we exemplify both the use of such ontologies to model an eldercare application in the context of ambient intelligence and smart cities, as well as how the ontology concepts support coding in agent platforms. We discuss the implications of such integrated view for designing agents, and highlight its advantages for agent-based software development.

Keywords: ontology, multi-agent system, ambient intelligence.

1 Introduction

The use of ontologies in the context of Multi-Agent Systems (MAS) is still an open issue, especially in relation to integrated frameworks that consider the co-specification of their different dimensions. The development of MAS in JaCaMo [1] comprises three distinct dimensions, namely: agent, organisation, and environment. However, these dimensions are not uniformly integrated into a single formalism: agents are programmed in Jason [2] using the AgentSpeak language; organisations are specified in Moise [3] in an XML-based document; and environments are coded in Java using the CArtAgO API [4]. This approach makes difficult to keep track of problems because errors in one level can affect the other levels, and it also becomes cumbersome to explore interconnections between the different layers and requires the programmer the knowledge about different paradigms. To address these issues, we propose a unified representation which covers these three agent programming dimensions and integrates the various formalisms. Thus, we developed ontologies to represent the agency, environment, and organisation levels of MAS, which are aligned with a platform integrating these three

F. Koch et al. (Eds.): CARE/AVSA 2014, CCIS 498, pp. 22–33, 2015.

levels in agent programming (*i.e.*, JaCaMo [1]). Until now, JaCaMo platform does not address the ontological level of MAS. Hence, we discuss an integrated semantic model to represent these three dimensions based on ontologies that represent each MAS level.

In order to demonstrate the need for and advantages of an integrated view for knowledge representation when developing complex multi-agent systems, we show how our approach can be used to model a multi-agent assisted living application aiming at supporting home care for an elderly patient. In particular, we support the collaborative work of a team of people including family members and professional carers who work together to allow an elderly patient with various debilitating health conditions to live in his own home. We use multi-agent systems techniques and ambient intelligence to give such support, but beyond this, our vision is towards systems integration in smart cities through multi-agent systems techniques to allow full integration of data from city transport, health systems, social services, smart grids, and so forth. That is, we envision all such sources of information coming together to give as much support as possible to the patient's family and carers. This is a very important social scenario in Brazil where the predominant culture is for families to care for their elders themselves.

An integrated ontology model to represent these MAS dimensions enables semantic reasoning and can be used as a common vocabulary in agent-oriented programming. In our proposal these dimensions are sub-ontologies that may interconnect with each other, and reuse relevant concepts from each other MAS dimension. Each dimension details different aspects, and these interconnections when combined have to result in an integrated knowledge model with a clear correspondence to an integrated programming platform, such as JaCaMo [1]. Some proposals for the knowledge level, such as Moise [3], are already related to a programming framework, allowing to convert the ontology specification to a programming level [5]. This is desirable for all dimensions, but these levels have to be aligned for that to work as a common specification. Also, an MAS can be modelled, reused and extended in one dimension while maintaining the others, which allows the designer to work without going into specifics of the programming languages that define each dimension. In this context, an MAS design is more easily expressed and communicated, and the model can be more easily converted to a formal verification system. Thus, our work is a step towards a knowledge level integration of MAS dimensions and platforms.

This paper is structured as follows. Section 2 refers to previous ontologies related to MAS aspects: agents, organisations, and environments. Section 3 shows the need for the integration of such aspects at the knowledge level on the light of an example in the area of health care. Section 4 concludes this paper and points to our next steps.

2 Ontologies for Multi-Agent Systems

Agents are reactive systems that can independently determine how to best achieve their goals and perform their tasks [2] while demonstrating properties such as autonomy, reactivity, proactiveness and social ability. Agents are situated in an environment, where they can perceive and modify it, and they should be able to exchange information, cooperate and coordinate activities. Jason [2] is an AgentSpeak language platform implementation that focuses on agent actions and mental concepts. It is an open source