

A Knowledge-Based Approach to the Specification and Verification of MAS Design

Anarosa A. F. Brandão

Computer Science Dept at PUC-Rio

R. Mq de S. Vicente, 225 - RJ

22453-900 - Brazil

+55 21 2540 6915 #131

anarosa@inf.puc-rio.br

Viviane Torres da Silva

Computer Science Dept at PUC-Rio

R. Mq de S. Vicente, 225 - RJ

22453-900 - Brazil

+55 21 2540 6915 #131

viviane@inf.puc-rio.br

Carlos J. P. de Lucena

Computer Science Dept at PUC-Rio

R. Mq de S. Vicente, 225 - RJ

22453-900 - Brazil

+55 21 2540 6915 #103

lucena@inf.puc-rio.br

Categories and Subject Descriptors

D.2.2 [Software Engineering]: Design Tools and Techniques:
Design verification

General Terms

Design, Verification.

Keywords

Multi-agent systems, design verification, ontology.

1. INTRODUCTION

The development of multi-agent systems (MAS) is a complex task due to the specific characteristics associated with the software agent abstraction. Although several modeling languages ([1],[2] [3]) were created to facilitate the development of MAS, even the modeling languages are complex by themselves. Therefore, it is expected that some inconsistencies occur while modeling the structural and dynamic aspects of MAS. The early verification of such inconsistencies can avoid several implementation problems. In general, modeling languages don't have a precise semantics thus there is a need for formally specifying the models to proceed their verification. By verification we mean the process of checking if application models conform to the specification of the modeling diagrams defined in the language metamodel. To provide a formal specification of MAS design models we propose the definition of an ontology based on the language metamodel.

2. PROBLEM DESCRIPTION

The inconsistencies that can occur when modeling an application are classified as intra-model inconsistencies and inter-model inconsistencies. Intra-model inconsistencies are the ones related to the non conformity between a diagram and its specification defined by the language metamodel. Inter-model inconsistencies result from the interdependencies between the diagrams. Due to inter-model inconsistencies, the overall MAS

modeling may become inconsistent, however each model is consistent itself. Given a modeling language and its associated metamodel, our problem consists of providing means to tame both types of inconsistencies by answering questions such as: 1. how can we describe what are these inconsistencies while modeling a MAS?; 2. how can we (automatically) detect such inconsistencies?; 3. how can we (automatically) avoid, when it is possible, the emergence of such inconsistencies?; 4. how can we solve these inconsistencies, when it is possible, or how can we provide guidelines for the designers to solve them? To answer these questions we propose a knowledge-based approach beginning with an ontology to formally describe the modeling language metamodel. The ontology contains the specification of all diagrams defined in the language metamodel. The diagrams (and their components) are directly mapped to ontology instances, generating a knowledge base (KB) which can be checked and manipulated through the use of KB reasoners. In addition, the ontology also introduces axioms that explain or restrict its components and describe the properties that may lead to intra-model inconsistencies. Therefore, the KB becomes inconsistent whenever a property is violated and this can be checked automatically by using a KB reasoner. To tame inter-model inconsistencies, the rules that verify their existence are described by defining queries that are used to query the KB composed of the overall application model. The queries results, provided by a KB reasoner, present such inconsistencies to the designer whenever they occur. As a proof of concept of the proposed solution, we consider MAS-ML [2], a modeling language that extends UML in a conservative way to describe the MAS specificities that cannot be described by using UML.

3. ACKNOWLEDGMENTS

This work is partially supported by CNPq/Brazil under the project "ESSMA", number 5520681/2002-0.

4. REFERENCES

- [1] Odell,J., Parunak, H. and Bauer,B. Extending UML for Agents.In *Proceedings of the Agent-Oriented Information Systems Workshop*, Austin, 2000, pp. 3-17.
- [2] Silva, V. and Lucena,C. From a Conceptual Framework for Agents and Objects to a Multi-Agent System Modeling Language, *Journal of Autonomous Agents and Multi-Agent Systems*, vol 9, 2004, pp. 145-189.
- [3] Wagner,G. The Agent-Object-Relationship Metamodel: Towards a Unified View of State and Behavior, *Information Systems*, vol 28, n 5, 2003, pp. 475-504..

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

AAMAS'05, July 25–29, 2005, Utrecht, NL.

Copyright 2005 ACM 1-59593-094-9/05/0007...\$5.00.