Development of Agent-Driven Systems: from i* Architectural Models to Intentional Agents' Code

Maurício Serrano and Julio Cesar Sampaio do Prado Leite,

Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro,
Rua Marquês de São Vicente 225, Ed. Padre Leonel Franca 13o. andar,
Rio de Janeiro, Brasil
mauserr@les.inf.puc-rio.br, http://www-di.inf.puc-rio.br/~julio/

Abstract. The intentionality concept can improve the cognitive capacity of software agents, especially if the proposed intentional reasoning engine deals with softgoals at runtime. In this scenario, the use of an intentionality-based technological set to develop agent-driven systems from i* models to code is adequate. In this paper, we propose heuristics to improve the development of agent-driven systems from i* models to Belief-Desire-Intention-based code. Moreover, we apply a fuzzy-logic-based mechanism to deal with softgoals "on the fly", improving the reasoning engine of intentional agents. We compare our efforts with related work and illustrate our contributions with a case study.

Keywords: i* models, model-driven development, intentional agents, transformational heuristics, fuzzy-logic-based reasoning engine.

1 Introduction

According to [1], a multi-agent system (MAS) is a system composed of many intelligent agents that work in a collaborative manner in order to achieve their goals. In other words, these intelligent agents tend to automatically find the best solution for the delegated goals – i.e. "without intervention". Therefore, the agents can assume different responsibilities – normally represented as different roles or capabilities. Moreover, they can perform several tasks by also adapting themselves to better achieve their goals.

A multi-agent system can be driven by behavioral agents [2] or intentional agents [3]. On one hand, the former orientation is more appropriate to deal with systems based on agents that can be modeled and implemented by using specific behaviors, such as: eating, sleeping, running, reading, and so on. Therefore, the behavior-based orientation seems proper to represent, for example, self-organization in social colonies (e.g. ant colonies). On the other hand, the latter orientation can improve the reasoning and learning capacity of intelligent agents by using the intentionality abstraction. According to [3], an intentional multi-agent system represents an adequate way to deal with human practical reasoning and to improve the goals formation and human-mental states interpretation. The BDI (Belief-Desire-Intention) model is an intentional model, intensely investigated in the Artificial Intelligence field. More recently, this model has been applied to intelligent agents by allowing them to deal with (i) **beliefs** – i.e. the agent's knowledge about, for example, the real world or a specific context; (ii) **desires** – i.e. the goals to be achieved by the agents; and (iii) **intentions** – i.e. a sequence of tasks that must be performed by the agents to achieve the delegated goals (in the BDI model, intentions have the semantics of actions with the purpose to achieve the desires).

In order to guide the development of multi-agent system, TROPOS [4] offers an agent-oriented methodology. This method is centered on the i* Framework [5], which abstractions (e.g. goals, softgoals, tasks, resources, beliefs and others) are used to model the requirements and design details of the MAS under development. There are different approaches [6] [7] centered on the TROPOS methodology that propose solutions to respectively: (i) develop MAS by applying heuristics to go from i* models to BDI-based code; and (ii) perform the development of MAS from organizational architectures in i* to architectures based on agents by also applying some Agent UML diagrams to capture the agents' intentionality. Section 3 highlights how our work differs from previous work.

In this paper, we propose specific design and implementation heuristics to drive the development of MAS from the requirements to code centered on intentional agents. Therefore, we propose: (i) using i* for both the requirements model as well as the design model; and (ii) the implementation of the modeled system based on the BDI model of the JADEX [8], an add-on to the JADE MAS platform. JADEX is also centered on the intentionality abstraction. Moreover, we offer an approach centered on fuzzy-logic to analyze the impacts of tasks on softgoals implemented as a reasoning engine for the intentional agents.

This paper is organized in Sections: Section 2 discusses the main objectives of our research; Section 3 presents some scientific contributions; Section 4 summarizes the proposal by presenting the final considerations; and finally, in Section 5 we consider the ongoing and future work.

2 Objectives of the research

In order to facilitate the presentation of our research, we use a case study – the Lattes-Scholar [9]. It uses two specific services: (i) the Lattes service [10], which consists of a base of scientific and technological curriculums maintained by the Brazilian Government; and (ii) the well-known Google Scholar service. The objective is to list the publications from an author's curriculum sorted by the number of citations.

We modeled the requirements and the detailed design of Lattes-Scholar by using i*, specifically the Strategic Dependency (SD) and Strategic Rationale (SR) models. Fig. 1 shows the architecture of Lattes-Scholar modeled in i*.

There are some differences between the i* modeling and the specification in BDI notation, such as: (i) the i* modeling represents a network of social actors, while the BDI-based specification represents the internal and mental architecture of a unique software agent; and (ii) the i* modeling represents sofgoals as well as how tasks, goals and softgoals impact on these softgoals by using positive and negative contribution links.

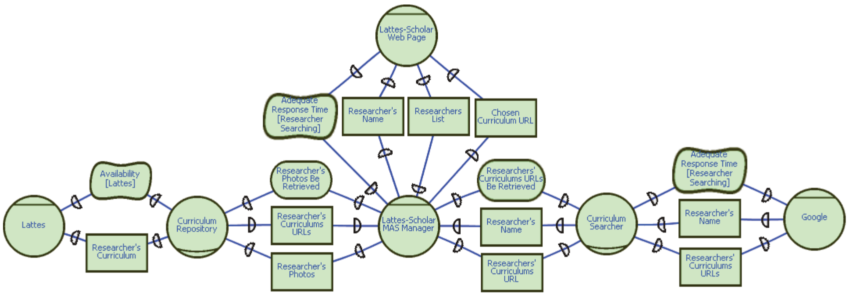
Lattes-Scholar Web Page
Adequate Response Time [Researcher Searching]
Researcher's Name
Researchers List
Chosen Curriculum URL
Availability [Lattes]
Researcher's Photos Be Retrieved
Researchers' Curriculums URLs Be Retrieved
Adequate Response Time [Researcher Searching]
Lattes
Researcher's Curriculum
Curriculum Repository
Researcher's Curriculums URLs
Researcher's Photos
Lattes-Scholar MAS Manager
Researcher's Name
Researchers' Curriculums URL
Curriculum Searcher
Researcher's Name
Researchers' Curriculums URLs
Google

Fig. 1. Lattes-Scholar's architecture modeling in i*

Although there are semantic differences between the models i* and BDI, these models share common concepts, such as: actor/agent, goal, task/intention, belief, resource/belief, and others. Therefore, it is possible to associate the abstractions of these models, as presented in Fig. 2. Table 1 describes some of these (design) transformational heuristics to produce a BDI specification from SD and SR i* models.

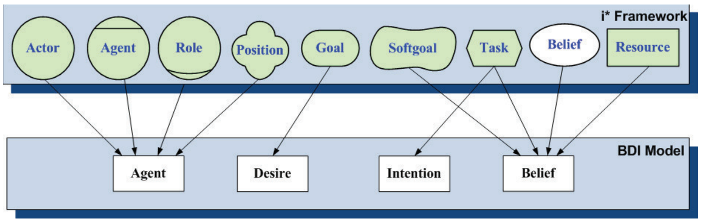

Fig. 2. Association between i* abstractions and BDI abstractions

It is important to consider that – after applying the design heuristics – there are some gaps in the BDI model of the agents, such as *types*. The software engineer must manually fill in these gaps.

Now, our interest is in a lower abstraction level, on which intelligent agents centered on BDI specifications are implemented as intentional agents in JADEX. The JADEX Framework implements a BDI architecture for agents of the JADE platform. The intentions defined in BDI specifications are implemented as Java object classes – i.e. Plan – in JADEX. Furthermore, JADEX is an add-on for the JADE platform, which implements the FIPA agent communication protocol and messages exchange. Fig. 3 illustrates the associations between the abstractions of the BDI specification and the MAS code in JADEX. We omitted the implementation heuristics (from BDI models to JADEX agents' code) due to lack of space.

Basically, the executable agents are implemented as JADEX agents. Non-executable agents (originated from roles and positions) are implemented as capabilities, which are acquired at runtime by software agents of the MAS platform. Desires are translated as goals and maintained or performed according to the tag "type" from the BDI specification. The intentions are translated as plans of agents – i.e. Java classes that extend the "Plan" class of JADEX. Beliefs with 0..1 or 1..1

cardinalities are translated as beliefs of agents, while beliefs with 0..n or 1..n cardinalities are translated as a set of beliefs.

Table 1. Transformational Heuristics: from i* Framework to BDI specification

#	When it is applied	Action	Exit (XML Fragment)
01	For each actor (or agent, or role, or position) of the system.	Create an agent in the BDI specification. Actors and agents have a tag runnable="yes". Roles and positions have a tag runnable="no".	<agent name="[actor_name]" runnable="[yes|no]"> <beliefs/> <desires/> <intentions/> </agent>

03	For each softgoal of an actor.	Create a belief with the type "Softgoal".	<belief
			name="[softgoal_name]"
			type="Softgoal" />

06	For each task that is performed to achieve a goal of an actor.	Create an intention and associate it to a desire (goal).	<intention name="[task_name]"> <desire>[goal_name]</desire> <script lang="" /> </intention>

07	For each task that is performed to achieve a goal of an actor.	Create a belief with the type "Task".	<belief
			Name="[task_name]"
			type="Task" />

14	For each dependency *per* goal.	Create a desire in the specifications of both involved agents.	<desire name="[goal_name]" type=""/>

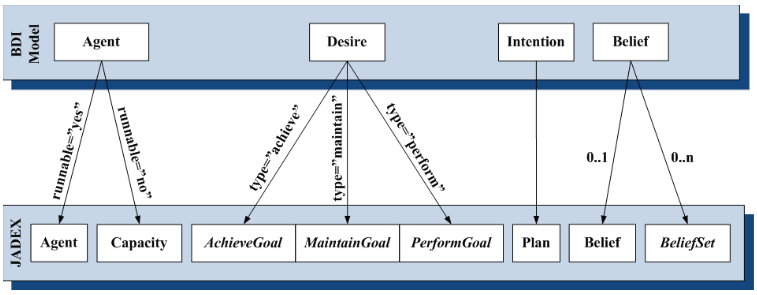

Fig. 3. Association between BDI abstractions and JADEX abstractions

We developed a qualitative reasoning engine [11] based on fuzzy logic to allow agents to perform their tasks by considering different quality criteria – i.e. modeled as softgoals in the i* models – and negative and positive contributions to them. The softgoals and contributions are available to the reasoning engine as agents' beliefs. The fuzzy logic usage allows agents to reason – "on the fly" – about softgoals by simulating the algorithms proposed in the i* Framework for models analysis. Other proposals [12] [13] are based on quantitative reasoning engines. According to our investigation by developing different intentional MAS, the use of a quantitative reasoning to deal with softgoals can be viewed as inadequate, especially if we consider their subjective and imprecise nature.

3 Scientific contributions

In [6], the authors suggest some heuristics that do not cover all abstractions of the i* Framework. For example, it does not deal with the role and position abstractions. They also do not address how to translate dependencies to interaction protocols. Moreover, they propose a reasoning engine based on softgoals priorities. In our proposal [11], we offer heuristics for all i* abstractions as well as our reasoning engine is enriched by a fuzzy-logic-based set to improve the agents' cognitive capacity on dealing with the intrinsic uncertainness of the modeled softgoals.

In [7], the authors use an extension of the UML to model the agents and their intentions. We argue that i* models can capture these intentions without the introduction of other notations or diagrams. However, some UML diagrams can be appropriate to deal with situations/aspects i* does not cover, such as temporality.

Summarizing the scientific contributions of our research, we can mention: (i) the usage of the i* as both requirements model and design model instead of using it to only model the requirements; (ii) the heuristics from i* abstractions to BDI abstractions – i.e. from i* models to BDI specification; (iii) the heuristics from BDI abstractions to JADEX BDI abstractions – i.e. from BDI specification to MAS code in JADEX; (iv) an approach centered on fuzzy-logic to analyze the impacts of tasks on softgoals; and (v) a reasoning engine for agents centered on this approach.

4 Conclusions

In this paper we present an overview of our work in order to develop MAS centered on the intentionality concept. The proposed support offers heuristics to conduct this development from i* models to the BDI-based code in JADEX. In addition, the proposed reasoning engine covers all i* abstractions, including, for example, belief, role and position abstractions.

One contribution of our efforts consists of avoiding the introduction of different notations and diagrams to capture the actors and agents intentions. In this case, we propose the use of the i* models (SR and SD models) to graphically represent the requirements and design details.

An intentional MAS reasoning engine combined with a fuzzy-logic mechanism [11] improves the cognitive capacity of the agents by allowing, for example, to deal with softgoals at runtime. The main idea is to analyze – "on the fly" – the impacts of the tasks, specified in the i* models, on the sofgoals, also specified in these models.

Finally, we can argue that, based on the systematic and incremental application of our approach to the Lattes-Scholar case study, we seem to be in the right direction in terms of: (i) reducing the necessary models to specify the requirements and the design details; (ii) improving the cognitive capacity of software agents centered on the intentionality through the use of the BDI model as well as the fuzzy-logic-based support set; and (iii) conducting the development of an intentional MAS from the requirements to code.

5 Ongoing and future work

Since the beginning of 2010, we have been applying our proposal to the Lattes-Scholar case study. Among other contributions, this incremental and systematic development has allowed us to evolve both the Lattes-Scholar system and our proposal – i.e. the proposed heuristics, our fuzzy-logic-based approach and our agents' reasoning engine.

As future work, we intend to develop a tool support to help the transformation process from the i* models to the BDI-based code in JADEX. The main idea is to semi-automate this process using the proposed heuristics and intentionality.

References

1. Wooldridge, M.: An Introduction to MultiAgent Systems. John Wiley & Sons Ltd, ISBN 0-471-49691-X, 366 pages (2002)
2. Bellifemine, F., Caire, G., Grenwood, D.: Developing Multi-Agent Systems with JADE. John Wiley & Sons, ISBN-13: 978-0-470-05747-6, 286 pages (2007)
3. Bratman, M. E.: Intention, Plans, and Practical Reason. University of Chicago, ISBN: 1575861925, 208 pages (1999)
4. Castro, J., Kolp, M., Mylopoulos, J.: Towards Requirements Driven Information Systems Engineering: The Tropos Project. Information Systems, vol.27, n. 6, pp. 365-389 (2002)
5. Yu, E.: Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering. 3rd IEEE Int. Symp. on Requirements Engineering (RE'97), pp. 226-235, Washington D.C., USA (1997)
6. Perini, A. and Susi, A.: Automating Model Transformations in Agent-Oriented Modeling. In Agent-Oriented Software Engineering VI, Springer LNCS, vol. 3950, pp. 167-178, Berlin/Heidelberg (2006)
7. Silva, C., Dias, P., Araújo, J., Moreira, A.: De Arquitecturas Organizacionais em i* a Arquitecturas Baseadas em Agentes: Uma abordagem orientada a modelos. XIV Workshop on Requirements Engineering, page 48, April (2011)
8. Braubach, L., Lamersdorf, W., Pokahr, A.: Jadex: Implementing a BDI Infrastructure for JADE Agents. Distributed Systems and Information Systems, vol. 3, n. 3, pp.76-85, September (2003)
9. Lattes-Scholar: Requirements Engineering Group at PUC-Rio. Available at: http://www.er.les.inf.puc-rio.br/~wiki/index.php/Lattesscholar (May 2011)
10. Lattes Platform. Available at: http://lattes.cnpq.br/ (May 2011)
11. Serrano, M., Serrano, M. and Leite, J.C.S.P.: "Dealing with Softgoals at Runtime: A Fuzzy Logic Approach", to appear at the 2nd International Workshop on Requirements@Run.Time (2011)
12. Penserini, L., Perini, A., Susi, A., Morandini, M., Mylopoulos, J.: A Design Framework for Generating BDIAgents from Goal Models. 6th Int. joint Conference on Autonomous Agents and Multiagent Systems, DOI:10.1145/1329125.1329307, number 149 (2007)
13. Weber, C., Current, J., Benton, W.: Vendor Selection Criteria and Methods. European Journal of Operational Research, vol. 50, no. 1, pp. 2–18 (1991)