

Integrando Requisitos Organizacionais à Modelagem de Sistemas Multiagente Normativos

Emmanuel Sávio Silva Freire¹ e Mariela Inés Cortés²

¹Instituto Federal do Ceará (IFCE)
Rua Deoclécio Lima Verde, s/n – 63.507-110 – Iguatu – CE – Brasil
savio.essf@gmail.com

²Universidade Estadual do Ceará (UECE)
mariela@larces.uece.br

Abstract. *Normative multiagent systems are used to development complex systems. Because this, many languages and techniques are proposed to help the modelling and programming phases of these systems. However, the gathering and analysis of requirements phases is essential for a best understanding of users' needs. In this context, this paper addresses a integration between gathering and analysis of requirements phase and modelling phase. Thus, a mapping between an istar framework and NorMAS-ML language is proposed. Because it is a preliminary work, it doesn't have results to compare with related work.*

Resumo. *Os sistemas multiagente normativos são utilizados para o desenvolvimento de sistemas complexos. Para isso, muitas linguagens e técnicas foram propostas para auxiliar as fases de modelagem e implementação destes sistemas. Entretanto, a fase de levantamento e análise de requisitos é essencial para um melhor entendimento das necessidades dos usuários. Neste contexto, este artigo aborda a integração entre as fases de levantamento e especificação de requisitos e a fase de modelagem. Para tanto, um mapeamento entre o framework istar e a linguagem NorMAS-ML é proposta. Vale ressaltar que, por se tratar de um trabalho preliminar, ainda não se dispõem de resultados para comparação com trabalhos relacionados.*

1. Introdução

Os sistemas multiagente normativos (SMAN's) são formados por um conjunto de agentes inteligentes que interagem entre si para alcançarem os seus objetivos. Entretanto, necessitam observar as normas definidas nesses sistemas para verificar quais as ações que são permitidas, obrigadas ou proibidas de serem executadas [Silva, Braga e Figueiredo, 2010].

Neste contexto, linguagens de modelagem [Freire et. al, 2012] [Silva, Braga e Figueiredo, 2010] e de implementação [Lopes et.al, 2011] [Rocha Jr, Freire e Cortés, 2013] foram propostas para auxiliar o desenvolvimento de tais sistemas. Dentre elas, destaca-se a linguagem NorMAS-ML. Por meio dela, é possível modelar todas as entidades e as características desses sistemas. Entretanto, essa linguagem, mesmo tendo uma ferramenta de modelagem, não dá suporte a fase de levantamento e análise de requisitos.

Segundo Sommerville (2007), a fase de levantamento e análise de requisitos propicia um melhor entendimento das reais necessidades dos usuários do sistema. Assim, é possível detalhar e priorizar os requisitos permitindo que aqueles essenciais ao sistema possam ser implementados inicialmente. Com isso, o usuário tem uma maior probabilidade de receber um sistema que resolva as suas necessidades.

Neste sentido, o *framework istar* (i*) [Yu, 1995] foi proposto para auxiliar a fase de requisitos por meio de um modelo conceitual composto pelo: (i) modelo de dependência estratégica e (ii) modelo estratégico de razão. Por meio deles, os engenheiros de requisitos conseguem modelar os *stakeholders* como atores e suas intenções como objetivos. Assim, consegue-se obter uma compreensão sobre os relacionamentos da organização e sobre as razões envolvidas nos processos de decisão [Yu, 1995].

No contexto de SMAN's, a metodologia Tropos utiliza o *framework istar* para a fase de requisitos integrada com a fase de modelagem. Entretanto, Tropos não permite a modelagem das características das normas presentes nos SMAN's. Assim, o presente artigo propõe a integração do *framework istar* (fase de requisitos) com a linguagem NorMAS-ML, que permite a modelagem de SMAN's, possuindo uma ferramenta de modelagem integrada juntamente com uma linguagem de implementação (JAMDER 2.0 [Rocha Jr, Freire e Cortés, 2013]).

O presente artigo está estruturado como segue: a Seção 2 apresenta o referencial teórico acerca do *framework istar* e da linguagem NorMAS-ML. A Seção 3 discute como a fase de requisitos é incluída no processo de desenvolvimento de SMAN's juntamente com a proposta de integração e mapeamento de *istar* e NorMAS-ML. Finalmente, a Seção 4 apresenta as conclusões e os trabalhos futuros.

2. Referencial Teórico

2.1. O Framework istar

O *framework istar* [Yu, 1995] possui uma estrutura conceitual para reconhecer motivações e intenções sobre as características de um processo. Com isso, auxilia no processo de levantamento e análise de requisitos. Ele é composto por dois modelos: (i) de Dependência Estratégica (SD), que fornece uma descrição intencional do processo em termos de uma rede de relacionamentos de dependência entre atores do ambiente, e (ii) Estratégico da Razão (SR) que apresenta uma descrição estratégica do processo, fornecendo uma análise dos meios para permitir que os objetivos possam ser cumpridos por meio das contribuições dos atores. Os dois modelos são utilizados nas fases iniciais do Tropos para capturar as intenções dos *stakeholders* e as responsabilidades dos sistemas.

2.2. A Linguagem NorMAS-ML

A linguagem NorMAS-ML (*Normative Multi-Agent Systems Language*) foi definida por Freire et. al (2012) por meio da extensão da linguagem MAS-ML [Silva, Choren e Lucena, 2008] por meio da inclusão dos conceitos estáticos das normas [Silva, Braga e Figueiredo, 2010]. Vale ressaltar, que todas essas linguagens são extensões de UML. Por meio dela, é possível modelar as características dos SMAN's juntamente com os

aspectos estáticos das normas. Adicionalmente, NorMAS-ML possui uma ferramenta de modelagem capaz de gerar código a partir dos seus modelos.

3. Discussão

Por meio de NorMAS-ML, pode-se gerar modelos para representar os SMAN's, tanto na fase de modelagem quanto na fase de implementação. Entretanto, essa linguagem não dá suporte a fase de levantamento e análise de requisitos. Segundo Sommerville (2007), a fase de requisitos propicia um maior entendimento do sistema a ser desenvolvido, pois o analista de requisitos pode interagir com os usuários para conhecer as suas necessidades.

Neste sentido, Yu (1995) propôs o *framework istar* que permite suporte à fase de requisitos por meio da identificação das necessidades de cada um dos *stakeholders* juntamente com o motivo para cada uma delas. Assim, Tropos inclui os modelos apresentados no *framework istar* na fase inicial de desenvolvimento. Adicionalmente, Melo et. al (2015) também apresentou um mapeamento do *framework istar* para modelos da UML, mais especificamente, para o diagrama de classes.

Como NorMAS-ML é uma extensão de UML, infere-se que se pode fazer um mapeamento entre *istar* e essa linguagem. Entretanto, na literatura, apenas trabalhos relacionando *istar* com SMA's são apresentados. Assim, precisa-se incluir o conceito normativo presente em NorMAS-ML. Segundo Silva, Braga e Figueiredo (2010), as normas são restrições sobre as ações dos agentes, indicando quais as ações que podem, devem ou não devem ser executadas. Por outro lado, Sommerville (2007) afirma que a especificação de sistemas necessita dos requisitos funcionais e não funcionais. O primeiro está relacionado com a funcionalidade em si, enquanto o segundo está relacionado com restrições funcionais do sistema. Assim, poderia pensar em mapear as normas como requisitos não funcionais.

Entretanto, ao analisar o conceito de regras de negócio, Sommerville (2007) indica que as regras são restrições impostas pelo processo de negócio da organização que irá utilizar o sistema. Logo, comparando este conceito com o de normas, pode-se verificar a semelhança entre eles. Portanto, as normas podem ser mapeadas como regras de negócio.

Adicionalmente, é necessário verificar como o mapeamento será realizado para as entidades presentes na linguagem. Vale ressaltar que Melo et. Al (2015) apresentou a formalização do mapeamento entre *istar* e o diagrama de classes da UML. Logo, como NorMAS-ML possui um diagrama de classes, pode-se utilizar a mesma estratégia utilizada por Melo et. al (2015). Por outro lado, as entidades ambiente, organização e agente presentes na linguagem precisam ser representadas em *istar*.

Neste contexto, pode-se definir pontos de extensão em *istar* considerando os conceitos de NorMAS-ML. Para tanto, precisa-se analisar a estrutura do *framework* e da linguagem e compará-las com o intuito de identificar as semelhanças existentes. Em seguida, deve-se verificar quais alterações devem ser realizadas no *framework* e/ou na linguagem e, finalmente, verificar a compatibilidade do mapeamento e da extensão. Para a validação da extensão, deve-se utilizar o estudo de caso modelado por [Freire et. al (2012) utilizando NorMAS-ML juntamente com os requisitos representados em *istar* analisados por Santos (2008). Por se tratar de um mesmo estudo de caso, pode-se modelá-lo com a extensão proposta para verificar a compatibilidade.

4. Conclusão e Trabalhos Futuros

Este artigo apresentou uma proposta de integração entre as fases de requisitos e de modelagem para permitir um melhor entendimento e modelagem de SMAN's. Para tanto, é proposta a utilização do *framework istar* juntamente com a linguagem NorMAS-ML para essa integração. Decidiu-se utilizar *istar* por conta da sua relevância para a análise e representação de requisitos. Por outro lado, resolveu-se escolher NorMAS-ML por conta da sua estrutura baseada em UML que permite a modelagem as entidades típicas dos SMAN's e possui ferramenta de modelagem integrada com um *framework* para desenvolvimento. Adicionalmente, espera-se, como trabalhos futuros: (i) analisar e integrar as entidades de NorMAS-ML e do *istar*, (ii) formalizar a integração por meio de ontologias, e (iii) verificar a consistência da extensão por meio de um estudo de caso.

Referências

- Freire, E. S. S., Cortés, M. I., Gonçalves, E. J. T., Lopes, Y. S. (2012). NorMAS-ML: A Modeling Language to Model Normative Multi-Agent Systems. In: 14th International Conference on Enterprise Information Systems (ICEIS), 2012, Wroclaw (Poland). Proceedings of the 14th International Conference on Enterprise Information Systems.
- Lopes, Y. S., Gonçalves, E. J. T., Cortés, M. I., Freire, E. S. S. (2011). Extending JADE Framework to Support Different Internal Architectures of Agents. In: The Ninth European Workshop on Multi-agent Systems (EUMAS), 2011, Maastricht (The Netherlands). Proceedings of the Ninth European Workshop on Multi-agent Systems.
- Melo, J., Sousa, A., Agra, C., Júnior, J., Castro, J., Alencar, F. (2015). Formalization of Mapping Rule from Istar to Class Diagram in UML. In: 29th Brazilian Symposium on Software Engineering, 2015.
- Rocha Jr., R. M., Freire, E. S. S., Cortés, M. I. (2013). Estendendo o Framework Jamder para Suporte à Implementação de Sistemas Multi-Agente Normativos. In: IX Simpósio Brasileiro de Sistemas de Informação (SBSI), 2013, João Pessoa. Anais do IX Simpósio Brasileiro de Sistemas de Informação (SBSI).
- Santos, B. S. (2008). IStar Tool - Uma proposta de ferramenta para modelagem de i*. Dissertação de Mestrado. Recife: Universidade Federal de Pernambuco, Centro de Informática.
- Silva, V., Braga, C., Figueiredo, K. (2010). A Modeling Language to Model Norms. In: Workshop on Coordination, Organization, Institutions and Norms in agent systems at International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS10), Toronto, p. 25-32.
- Silva, V. T., Choren R., Lucena, C. (2008). MAS-ML: a multi-agent system modelling language, In IJAOSE, Modeling Lang. for Agent Systems,(2)4pp.382-421.
- Sommerville, I. (2007). Engenharia de Software. 8 ed. São Paulo: Pearson Addison-Wesley.
- Yu, E. (2002). Agent-Oriented Modelling: Software Versus World, In: Proceedings of the Agent-Oriented Software Engineering (AOSE'01), Edited by Wooldridge, M., Weiss, G. and Ciancarini, P., LNAI, Vol. 2222, Springer-Verlag, p. 206 – 225.