



RESUMEN DE TESIS: Análisis y Diseño de Sistemas Multiagente Normativos Abiertos

Emilia Garcia, Adriana Giret

Universitat Politècnica de València
{mgarcia,agiret}@dsic.upv.es

Abstract Este artículo resume las contribuciones principales de la tesis con el título "Engineering Regulated Open Multiagent Systems". Esta tesis se centra en el análisis y diseño de sistemas normativos abiertos mediante tecnología multiagente. Específicamente esta tesis ofrece: (1) un metamodelo que permite especificar todas las características de estos sistemas; (2) una metodología que cubre el análisis y diseño mediante la especificación detallada del proceso de desarrollo y guías específicas que incluyen la identificación y formalización del entorno normativo del sistema; (3) una herramienta CASE que integra el diseño del sistema con la verificación formal de su entorno normativo.

Keywords: Multi-Agent Systems, Software engineering, Contracts, Model-Driven Software Development, Model Checking

1 Introduction

Actualmente existe una creciente demanda de sistemas flexibles, adaptables y con gran escalabilidad para apoyar las interacciones de personas e instituciones distribuidas en entornos heterogéneos. Esto se debe principalmente al incremento en la necesidad de trabajo colaborativo y la descentralización de los procesos en muchos dominios de aplicación. Por lo general, estas aplicaciones de software deben seguir legislaciones y normativas específicas, es decir, las entidades que participan en el sistema tienen derechos, deberes y restricciones específicas. Al igual que en otros trabajos del área [8], en esta tesis se utiliza el término *sistemas normativos abiertos* para referirse a los sistemas de este tipo.

El desarrollo de sistemas normativos abiertos puede producir importantes beneficios para las compañías que los usen, ya que permiten la comunicación de instituciones, entidades heterogéneas y diferentes dispositivos con el fin de lograr tanto los objetivos globales del sistema como los individuales de cada institución y entidad. Sin embargo, también hay algunas cuestiones importantes que potencialmente pueden complicar el análisis, diseño e implementación de estos sistemas. La mayoría de estos problemas están relacionados con la interoperabilidad de sus procesos, la privacidad, la combinación de los objetivos individuales y la combinación de las restricciones y la legislación de cada una de las entidades del sistema. Por lo tanto, es necesario el uso de métodos de ingeniería del software y herramientas de desarrollo para hacer frente a estos problemas y guiar a los desarrolladores durante el proceso de desarrollo.

La tecnología basada en sistemas multiagente (SMA) es considerada una buena candidata para el desarrollo de sistemas normativos abiertos [7]. Durante los últimos años, el uso de las tecnologías SMA se ha incrementado no sólo en el ámbito académico, sino también en el desarrollo e implementación de aplicaciones industriales. Los SMA se han establecido como un paradigma de la ingeniería del software para la creación de sistemas adaptativos complejos, en entornos distribuidos y heterogéneos.

Esta tesis se centra en el análisis y diseño de sistemas normativos abiertos utilizando la tecnología SMA. Algunas metodologías SMA se dedican al desarrollo de sistemas de este tipo [1, 3, 2]. Sin embargo, después de analizar en qué medida las metodologías SMA actuales soportan el análisis y el diseño de estos sistemas, podemos concluir que todavía hay importantes problemas a resolver en el área. Algunos de estos problemas son la integración del contexto normativo del sistema durante el proceso de desarrollo, la falta de directrices para identificar y formalizar este contexto normativo, la falta de técnicas de validación y verificación que garanticen la coherencia del diseño final respecto a los requisitos del sistema, la coherencia entre los objetivos individuales y la coherencia de las restricciones de cada entidad respecto al contexto normativo del sistema global.

2 Contribuciones

Esta tesis contribuye al estado del arte mediante un método de modelado llamado ROMAS (Sistemas Multiagente Regulados y Abiertos). Con el objetivo de soportar completamente las etapas de análisis y diseño de procesos para el desarrollo de sistemas multiagente normativos abiertos, la propuesta de ROMAS está compuesta por un lenguaje de modelado específico, una metodología que guía el proceso y una herramienta CASE que soporta el uso de dicha metodología. El metamodelo y la metodología se han definido siguiendo el estándar FIPA *Design Process Documentation Template*.

2.1 Metamodelo

Los sistemas normativos abiertos son especificados en ROMAS como sistemas multiagente organizacionales, donde los agentes interactúan por medio de servicios estándares, y donde las relaciones sociales y contractuales se formalizan mediante normas y contratos. Tal y como muestra la Figura 1, en la metodología ROMAS, *agentes*, *roles* y *organizaciones* se definen a través de una estructura social formal basada en un arquitectura orientada a servicios. Aquí, las organizaciones representan un conjunto de personas e instituciones que tienen que coordinar recursos y servicios a través de fronteras institucionales. En este contexto, los agentes representan a las partes individuales que asumen roles en el sistema, dentro de una organización (por ejemplo una empresa), que pueden ofrecer y consumir servicios como parte de las funciones que desempeñan. Más allá de esto, las organizaciones también pueden ser construidas para coordinar recursos y servicios a través de los límites institucionales. Las *normas* definen permisos, obligaciones y prohibiciones que restringen el comportamiento de las entidades del sistema. Los *contratos* se utilizan para formalizar las relaciones entre las entidades. A pesar de que los contratos se especifican completamente en tiempo de ejecución, durante el diseño del sistema se deben especificar plantillas de contrato que predefinen características que todos los contratos de ese tipo deben cumplir. En nuestro enfoque, podemos diferenciar entre dos tipos de contratos: *acuerdos contractuales* y *contratos sociales*.

El metamodelo de ROMAS se describe mediante 4 vistas: (1) vista organizacional en la que se especifican los objetivos globales, la funcionalidad requerida y ofrecida al entorno y la estructura social de la organización; (2) vista interna en la que se especifica la funcionalidad interna, capacidades, creencias y objetivos de cada entidad del sistema; (3) vista de la plantilla del contrato en la que se especifican restricciones predefinidas que todo contrato de un determinado tipo debe cumplir; (4) vista de actividad en la que se especifican los protocolos de interacción y la secuencia de actividades en las que las tareas y servicios se descomponen. Dicho metamodelo y sus vistas puede consultarse en [4].

2.2 Metodología

La metodología ROMAS define un proceso de desarrollo compuesto por actividades y tareas que guían a los desarrolladores en cada etapa del proceso (ver descripción detallada en [6]). ROMAS proporciona un conjunto de guías que facilitan el proceso de toma de decisión en puntos críticos del diseño tales como la identificación y formalización del marco normativo del sistema, así como las comunicaciones y los intercambios de servicios y recursos. El proceso de desarrollo está compuesto por 5 fases: (1) *Especificación del sistema* en la que se analizan los requisitos del sistema desde un punto de vista global; (2) *Especificación de la organización* en la que se analiza y diseña la estructura social y funcional del sistema; (3) *Especificación del contexto normativo* en la que se identifican y formalizan las normas y

contratos del sistema; (4) *Especificación de la actividad* en la que se detallan las tareas, protocolos de interacción y servicios del sistema; (5) *Especificación de los agentes* en la que se analizan y diseñan las entidades individuales del sistema.

2.3 Herramienta CASE

ROMAS ofrece una herramienta de modelado textual y gráfica que soporta el metamodelo de ROMAS [5]. Esta herramienta ha sido implementada utilizando tecnología de desarrollo dirigida por modelos. Específicamente se ha utilizado tecnología Eclipse. Por lo tanto, dicha herramienta es totalmente modulable e interoperable con otras herramientas basadas en el estándar Ecore.

Esta herramienta de modelado integra técnicas de *model checking* que permiten verificar formalmente partes del modelo. En la actualidad ROMAS permite la verificación de la coherencia del marco normativo de un sistema, es decir, la coherencia entre las restricciones y compromisos de cada entidad y la especificación global del sistema. Para implementar esta funcionalidad se ha integrado la herramienta de modelado con un plug-in para Eclipse del model checker Spin [9] y se ha creado un plug-in que transforma los modelos gráficos en código verificable.

La herramienta CASE ROMAS también incluye un módulo de generación de código para la plataforma de agentes Thomas [10]. Cabe destacar que el trabajo de implementación de dicho módulo fue desarrollado fuera del ámbito de esta tesis.

3 Conclusiones y limitaciones

Esta tesis presenta ROMAS, una nueva propuesta para el desarrollo de sistemas normativos abiertos. ROMAS analiza y diseña este tipo de sistemas desde un punto de vista global del sistema e individual de cada entidad autónoma que lo compone. Para especificar la visión global del sistema, ROMAS utiliza abstracciones de alto nivel tales como organizaciones, contratos, normas, roles, agentes y servicios. Estas abstracciones resultan cercanas a los expertos en el dominio y facilitan las tareas de análisis y diseño. Además, el uso de contratos para desarrollar este tipo de sistemas les otorga flexibilidad. Esto se debe a que la especificación de contratos permite conocer de antemano el comportamiento esperado de cada entidad sin restringir cómo va a llevarlo a cabo ni qué tecnología va a usar.

La mayor contribución de la metodología ROMAS y lo que la distingue de otras propuestas es que cubre el análisis y el diseño de estos sistemas teniendo en cuenta su contexto normativo. ROMAS ofrece un conjunto de guías de desarrollo que facilitan las tareas críticas del proceso tales como la identificación y formalización del contexto normativo.

ROMAS está soportada por una herramienta CASE que no sólo permite el modelado de sistemas normativos abiertos sino que lo integra con la verificación formal del contexto normativo mediante técnicas

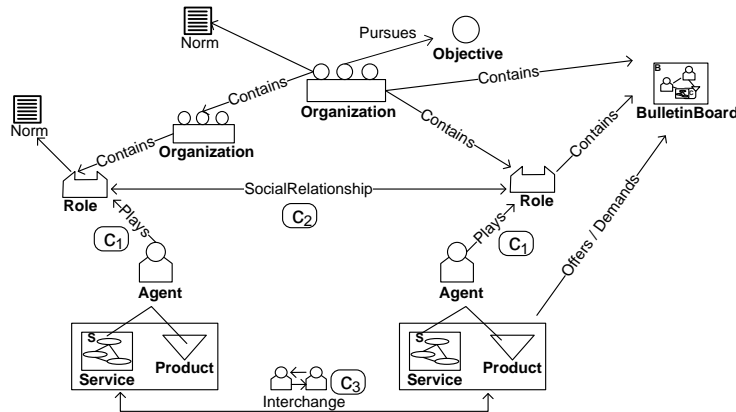


Figure 1: Esquema general de los elementos de ROMAS

de model checking.

Algunas de las limitaciones de esta tesis son la falta de guías metodológicas para especificar el protocolo de interacción más adecuado en cada caso, la integración de las guías metodológicas en la herramienta de modelado y los problemas de escalabilidad del módulo de verificación formal mediante model checking. Estas limitaciones son el eje de nuestro trabajo futuro.

Agradecimientos

This work is partially supported by the TIN2008-04446, TIN2009-13839-C03-01, PROMETEO 2008/051 projects, CONSOLIDER INGENIO 2010 under grant CSD2007-00022.

References

- [1] Scott A. DeLoach. Developing a multiagent conference management system using the o-mase process framework. In *Proc. Int. Conf. on Agent-oriented software engineering VIII*, pages 168–181, 2008.
- [2] Scott A. DeLoach, Lin Padgham, Anna Perini, Angelo Susi, and John Thangarajah. Using three aose toolkits to develop a sample design. In *International Journal Agent-Oriented Software Engineering*, volume 3, pages 416–476, 2009.
- [3] V. Dignum. *A model for organizational interaction: based on agents, founded in logic*. PhD thesis, Utrecht University, 2003.
- [4] Emilia Garcia, A. Giret, and V. Botti. Regulated Open multi-agent Systems based on contracts. In *Int. Conf. on Information Systems Development*, 2010.
- [5] Emilia Garcia, A. Giret, and V. Botti. A Model-Driven CASE tool for Developing and Verifying Regulated Open MAS. *Science of Computer Programming*, page In Press doi: 10.1016/j.scico.2011.10.009, 2011.
- [6] Emilia Garcia, A. Giret, and V. Botti. Developing Regulated Open Multi-agent Systems. In *International Conference on Agreement Technologies*, pages 12–26, 2012.
- [7] Emilia Garcia, G. Tyson, S. Miles, M. Luck, A. Taweel, T. Van Staa, and B. Delaney. Analysing the Suitability of Multiagent Methodologies for e-Health Systems. In *Agent-Oriented Software Engineering XIII*, volume 7852, pages 134–150. Springer-Verlag, 2013.
- [8] Christopher D. Hollander and Annie S. Wu. The current state of normative agent-based systems. *J. Artificial Societies and Social Simulation*, 14(2), 2011.
- [9] Tim Kovše, Boštjan Vlaovič, Aleksander Vreže, and Zmagor Brezočnik. Eclipse plug-in for spin and st2msc tools-tool presentation. In *Proceedings of the 16th International SPIN Workshop on Model Checking Software*, pages 143–147, 2009.
- [10] Mario Rodrigo, Soledad Valero, Carlos Carrascosa, and Vicente Julian. Tool and integrated application development environment. Document identifier: AT/2011/D6.2.3/v1.0 Project: CSD2007-0022, INGENIO 2010 <http://www.agreement-technologies.org/>. 2011.