

Ingeniería de Software Dirigida por Modelos aplicada a sistemas robóticos usando los estándares de la OMG

Claudia Pons, Gabriela Pérez, Carlos Neil, Roxana Giandini, Marcelo de Vincenzi

CIC, Universidad Abierta Interamericana (UAI) y Facultad de Informática de la UNLP
Buenos Aires, Argentina

Resumen

El objetivo general de esta investigación es contribuir al mejoramiento de los procesos de desarrollo de software de los sistemas robóticos, a través del análisis del paradigma de desarrollo MDD aplicando los estándares definidos por la OMG para el modelado de estos sistemas y complementado con componentes (CBD) y la arquitectura orientada a servicios (SOA).

Palabras clave: sistemas robóticos, ingeniería de software, desarrollo dirigido por modelos, estándares de la OMG.

1. CONTEXTO

Los sistemas robóticos (RSS Robotic Software Systems) desempeñan un papel cada vez más importante en nuestra vida cotidiana. La necesidad de sistemas robóticos en todos los entornos (industriales, educativos) aumenta y sus requisitos se vuelven más exigentes. Están hechos de diferentes componentes y sensores, lo que resulta en una arquitectura muy compleja y altamente variable. Actualmente, la mayoría de los sistemas de software de robótica se basan todavía en software propietario y están estrechamente ligados al hardware específico, las plataformas de procesamiento o la

infraestructura de comunicación. En consecuencia, estos robots solo pueden ser armados, configurados y programados por expertos. Los enfoques tradicionales utilizados en el proceso de desarrollo de este tipo de sistemas están basados principalmente en codificar las aplicaciones sin ningún tipo de técnica de modelado. Y, aunque estas aplicaciones se utilizan en diferentes sistemas robóticos, se pueden identificar algunos problemas. Entre ellos, vale la pena mencionar que no hay documentación clara sobre las decisiones de diseño que se toman durante la fase de codificación, por lo que se dificultan tanto la evolución como el mantenimiento de estos sistemas. Además, al utilizar lenguajes de programación específicos perdemos la posibilidad de generalizar conceptos que pueden ser extraídos, reutilizados y aplicados en otros sistemas, lo que nos permitiría evitarnos recodificar todo desde cero cuando se lo necesite.

Por otro lado, a medida que los sistemas robóticos crecen para ser cada vez más complejos, la necesidad de aplicar los principios de ingeniería de software para su proceso de desarrollo se convierte en un reto obligatorio en estos días.

Desde esta perspectiva, se acepta el hecho de establecer nuevos enfoques para satisfacer las necesidades del proceso de

desarrollo de sistemas robóticos tan complejos como los de hoy. El desarrollo basado en componentes (component-based development o CBD), la arquitectura orientada a servicios (Service-oriented architecture o SOA) [5], así como la Ingeniería de software dirigida por modelos (MDE), y el modelado específico de dominio (DSM) son algunas de las tecnologías más prometedoras en el dominio de los sistemas robóticos.

Actualmente se promueve la integración de los componentes de los sistemas robóticos a través de la adopción de estándares de la OMG [1, 11]. Se han lanzado cuatro especificaciones: para la interacción en los sistemas robóticos (Robotic Interaction Service – ROIs) [3], para la localización de los sistemas robóticos (Robotic Localization Service - RLS), para el modelado de los componentes (Robotic Technology Component - RTC) [2] y para el despliegue dinámico y su configuración (Dynamic Deployment and Configuration for Robotic Technology Component - DDC4RTC).

Es en este contexto en el que el Desarrollo Dirigido por Modelos (MDD, Model Driven Development)[7, 8, 9] aparece como una alternativa viable para aplicar técnicas de ingeniería de software en el desarrollo de este tipo de sistemas. Su uso logra un nivel de abstracción superior, permitiendo utilizar los estándares propuestos para robótica, y así obtener ventajas como generalidad, reutilización, claridad, expresividad. Estas son todas cualidades inherentes a un proceso de creación de software eficiente y eficaz.

2. OBJETIVOS, HIPÓTESIS Y RESULTADOS ESPERADOS

El objetivo general de esta investigación es contribuir al mejoramiento de los procesos de desarrollo de software de los sistemas robóticos, a través del análisis del paradigma

de desarrollo MDD aplicando los estándares definidos por la OMG para el modelado de estos sistemas y complementado con componentes (CBD) y la arquitectura orientada a servicios (SOA).

3. LINEAS DE INVESTIGACIÓN

Para cumplir con el objetivo detallado anteriormente se desarrollan los siguientes temas:

- Modelos y Lenguajes de modelado
- Transformación de modelos
- Herramientas automáticas para transformación de modelos
- Técnicas de desarrollo de componentes (CBD)
- Arquitectura orientada a servicios (SOA).
- Lenguajes estándar definidos por la OMG para los sistemas robóticos, como RTC, ROIs
- Fundamentos teóricos para la composición de transformaciones.

4. PROPUESTA

Este trabajo tiene como objetivo general estudiar los sistemas robóticos y cómo aplicar técnicas de ingeniería de software para desarrollarlos. Este objetivo está siendo abordado a través de los siguientes sub-objetivos:

- Estudiar las propuestas de la OMG para el modelado de sistemas robóticos, y analizar como pueden ser aplicadas en el desarrollo de los sistemas robóticos.
- Analizar las interrelaciones entre los distintos estándares.
- Analizar cómo pueden ser integrados los paradigmas SOA y CBD con lo propuesto por la OMG
- Estudiar como los modelos realizados en dichos estándares pueden ser

transformados para crear otros modelos, y eventualmente, crear código.

5. METODOLOGÍA DE INVESTIGACIÓN Y DESARROLLO PROPUESTA

A partir del estudio de los sistemas robóticos actuales, se observa que son artesanales, y que su desarrollo esta basado principalmente en el código. A pesar de que existen propuestas para el modelado de los sistemas robóticos, aun no se aplican y tampoco cubren la posibilidad de la integración de dichas propuestas.

Nuestra hipótesis de trabajo consiste en la posibilidad de integrar los estándares con otras técnicas de ingeniería de software para poder aplicar MDD en el desarrollo de los sistemas robóticos.

Respecto a los lenguajes de modelado y transformación nos basaremos en los estándares de la OMG, en particular MOF [6], UML, OCL y QVT[10].

Finalmente, todos nuestros resultados teóricos serán plasmados en herramientas de desarrollo de software de código abierto, preferentemente sobre la plataforma Eclipse [15].

6. ESQUEMA DE PLAN DE TRABAJO C/ACTIVIDADES

El proyecto viene desarrollándose a través de las siguientes actividades:

- _ Estudio de los lenguajes estándares para modelado de sistemas robóticos
- _ Estudio del estándar como base para el modelado de software, en particular MOF (Meta Object Facilities), RTC (Robotic Technology Component), ROIs (Robotic Interaction Service), DDC4RTC (Dynamic Deployment and Configuration for Robotic

Technology Component), RLS (Robotic Localization Service).

- _ Estudio del estándar para modelado de servicios SOA.
- _ Analizar las relaciones entre dichos estándares y como se complementan para ser utilizados en el proceso de desarrollo.
- _ Analizar mecanismos de transformación entre modelos
- _ Plasmar lo analizado en herramientas de desarrollo de software de código abierto
- _ Evaluación de la propuesta mediante la aplicación de la misma en proyectos reales.

7. FORMACIÓN DE RECURSOS HUMANOS Y RESULTADOS

El equipo está integrado por cinco investigadores senior, provenientes de 2 universidades (UNLP y UAI), quienes combinan sus conocimientos en robótica e ingeniería de software. El resto del equipo está integrado por estudiantes de pregrado y postgrado. En el contexto de este proyecto se están desarrollando 5 tesis de licenciatura en Informática, 5 tesis de Maestría en TI y 5 tesis doctorales (1 financiada por CONICET y 1 por el programa DoctorAR). Se trabaja además en colaboración con la Universidad de Viena a través de un convenio bi-lateral financiado por el Mincyt [16]. Los resultados preliminares del proyecto se han publicado en [17], [18], [19], [20] y [21].

8. REFERENCIAS

1. OMG Robotics-DTF - Object Management Group - <http://robotics.omg.org/> (consultado en 2015)
2. Documento de especificación RTC 1.0 - Object Management Group - <http://www.omg.org/spec/RTC/1.0/> (consultado en 2015)

3. Documento de especificación Robotic Interaction Service (RoIS) – OMG - <http://www.omg.org/spec/RoIS/> (consultado en 2015)
4. Robotic Localization Service (RLS) – OMG - <http://www.omg.org/spec/RLS/>
5. Documento de especificación SOA – OMG - <http://www.omg.org/technology/readingroom/SOA.htm> (consultado en 2015).
6. Documento de especificación OMG's MetaObject Facility (MOF) Home Page - <http://www.omg.org/mof/>
7. Stahl, M Voelter. Model Driven Software Development. John Wiley, ISBN 0470025700.
8. Kleppe, Anneke G. and Warmer Jos, and Bast, Wim. MDA Explained: The Model Driven Architecture: Practice and Promise. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2003.
9. Object Management Group, MDA Guide, v1.0.1, omg/03-06-01 (2003).
10. MOF 2.0 Query/View/Transformations - OMG Adopted Specification. March 2005. <http://www.omg.org>.
11. Object Management Group (OMG) <http://www.omg.org>
12. Ledeczi, A., Bakay, A., Maroti, M., Volgyesi, P., Nordstrom, G., Sprinkle, J., Karsai, G. Composing Domain-Specific Design Environments. IEEE Computer 34 (2001)
13. Kleppe, Anneke. MCC: A Model Transformation Environment. A. Rensink and J. Warmer (Eds.): ECMDA-FA 2006, LNCS 4066, pp. 173 – 187, Spain, June 2006.
14. Atlas Model Weaver Project Web Page. <http://www.eclipse.org/gmt/amw/>, 2005.
15. Proyecto Eclipse - <https://eclipse.org/> (consultado en 2015).
16. Project Title: "Adoxx Meta-Model Compiler. Modelling Methods for Robotic Systems". Programa de Cooperación Científico-Tecnológica entre el Ministerio de Ciencia, Tecnología e Innovación Productiva de la República Argentina (MINCyT) y el Ministerio de Ciencia e Investigación de la República de Austria (BMWF).
17. C. Pons, G. Pérez, R. Giandini, G. Baum . A Model-Driven Approach to Constructing Robotic Systems. Journal of Computer Science & Technology. Vol. 14 - No. 1 – April 2014 - ISSN 1666-6038.
18. Jerónimo Irazábal and Claudia Pons. Metamodel independence in Domain Specific Modeling Languages. "Communications in Computer and Information Science CCIS Series" vol.411, "Software and Data Technology" pp.140-154, ISBN 978-3-642-45403-5. Springer-Verlag Berlin Heidelberg (2013).
19. Omar Martinez Grassi, Claudia Pons, Gabriel Baum. Variable-Based Analysis for Traceability in QVT-R Model Transformations. CibSE - SET 2015 (CibSE - Software Engineering Track). XVIII Conferencia Iberoamericana en "Software Engineering", CibSE 2015. Perú. April 2015.
20. Gabriela Pérez, Jerónimo Irazábal, Claudia Pons y Roxana Giandini. Applying

MDE tools to defining domain specific languages for model management. SADIO Electronic Journal of Informatics and Operations Research. ISSN 1514-6774 vol. 12, no. 1 (Sept 2013).

21. A systematic review of applying modern software engineering techniques to developing robotic systems. Claudia Pons, Roxana Giandini, Gabriela Arévalo. Vol. 32 No. 1 de 2012. Revista Ingeniería e Investigación. Tri-annual ISSN: 0120-5609. Indexada en el ISI.