Introducing Safety Requirements Traceability Support in Model-Driven Development of Robotic Applications*

Pedro Sánchez[1], Diego Alonso[1], Francisca Rosique[1], Bárbara Álvarez[1], and Juan A. Pastor[1]

División de Sistemas e Ingeniería Electrónica (DSIE)
Universidad Politécnica de Cartagena, 30.202, Cartagena, España
{pedro.sanchez, diego.alonso}@upct.es

Summary

Teleoperated service robots are used to perform hazardous operations in hostile environments such as nuclear reactors, space missions, warehouses, etc. Since they have to interact with both the environment and human operators, it is essential that they be so designed as to involve no risk to the operators, the environment, or the robot itself. Where it is impossible to eliminate the risk, this at least must be limited.

The work described in this article was developed in the context of the European Union V Framework Programme EFTCoR project (*Environmental Friendly and Cost-Effective Technology for Coating Removal*), which addressed the development of a solution to the problem of retrieval and confinement of sub-products from ship maintenance operations. Given the experience of the DSIE research group in both the design of component-based software applications for tele-operated service robots [1], and the combined use of safety standards (like ANSI/RIA 15.06-1999 and European Standard EN 61508:2001) with specific methodologies for safety systems development (like Rapid Object-Oriented Process for Embedded Systems, ROPES) [2], we decided to develop an integrated development framework.

The Model-Driven Software Development (MDSD) approach can provide a suitable theoretical and technological support for integrating different facets of safety critically system in a global development framework. To implement a system that assures safety, a component-based approach, and traceability, there are a number of conceptual and development (technical, tools, etc.) requirements that have to be met. The absolutely essential ones are basically:

1. An integrated framework of safety management requirements.
2. A catalogue of architecture solutions for these requirements.
3. A framework for automatic generation of code for different implementation platforms.
4. Support for traceability of the various artifacts involved in the development.
5. Tools which enable fully automated handling of all the information involved in the process.

* This work has been partially supported by the Spanish CICYT Project EXPLORE (ref. TIN2009-08572), the Séneca Project MISSION-SICUVA (ref. 15374/PI/10), and the Spanish MEC FPU Program (grant AP2009- 5083).

A. Ruíz, L. Iribarne (Eds.): JISBD'2012, pp. 519-520, ISBN:978-84-15487-28-9.
Jornadas SISTEDES'2012, Almería 17-19 sept. 2012, Universidad de Almería.

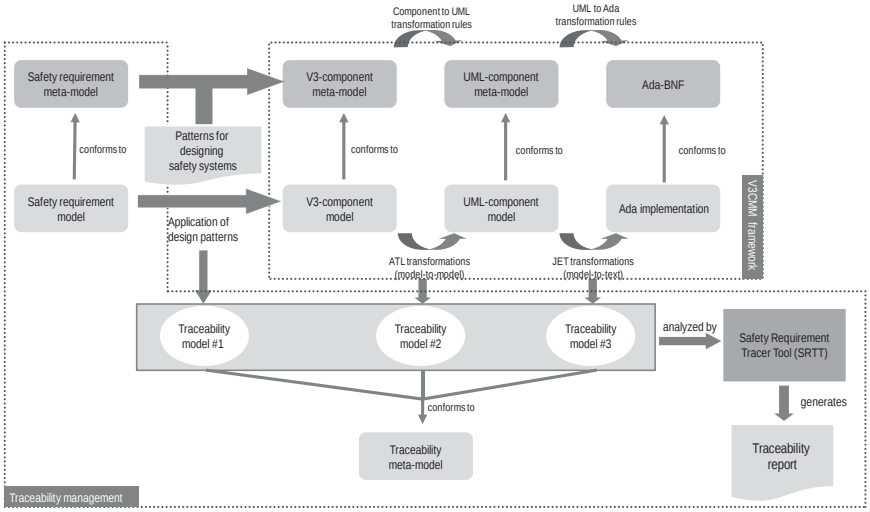

Fig. 1. General MDSD development framework for component-based applications, completed with safety requirements traceability management.

This article presents an integrated development environment that furnishes these resources (see Fig. 1), and demonstrates the viability of the approach on the basis of experience acquired in the development of an industrial robotic system with safety requirements, which represents a relevant case study in synergy between the Robotics and Software Engineering domains. While MDSD is a well-known Software Engineering paradigm and Robotics is a discipline with a long tradition, as far as we know, there is no experience on Robotics that uses in an extensive way the MDSD approach. Nevertheless, there are tremendous opportunities to start with reusable and semantically well-defined designs of complex robotic software systems. The work described in this article offers developers an MDSD environment which: (1) promotes the reuse of the software artifacts generated in the safety systems construction process, and (2) considers the traceability of these artifacts in order to enhance the quality of the systems, since it facilitates both modification and analysis of the impact of the changes on the safety requirements. It is worth noting that these results can be easily extrapolated to other domains different to robotics, since the software artifacts developed and used are indeed domain independent.

References

1. A. Iborra, D. Alonso, F. Ortiz, J. Franco, P. Sánchez, and B. Álvarez: Design of service robots. IEEE Robotics and Automation Magazine, Special Issue on Software Engineering for Robotics, vol. 16, no. 1, pp. 24–33, 2009.
2. D. Alonso, P. Sánchez, F. Ortiz, J. Pastor, B. Álvarez, and A. Iborra: Experiences developing safe and fault-tolerant tele-operated service robots, a case study in shipyards. In Service robot applications. Ed. In-Tech, 2008, pp. 159–182.

Un repositorio NoSQL para acceso escalable a modelos

Javier Espinazo Pagán[1], Jesús Sánchez Cuadrado[2] y Jesús García Molina[1]

[1] Universidad de Murcia, España jespinazo@um.es, jmolina@um.es
[2] Universidad Autónoma de Madrid, España jesus.sanchez.cuadrado@uam.es

Abstract. La aplicación de la Ingeniería Dirigida por Modelos (MDE) en sistemas de escala industrial requiere de complejos modelos que pueden llegar a ser muy grandes y que deben ser almacenados de tal forma que puedan ser manipulados por aplicaciones cliente sin necesidad de ser cargados por completo. En este artículo presentamos Morsa, un repositorio para la manipulación escalable de modelos grandes usando carga bajo demanda, guardado incremental y un API de consultas; la persistencia de modelos es llevada a cabo por una base de datos NoSQL.

Keywords: persistencia de modelos, repositorio de modelos, escalabilidad, modelos grandes

1 Introducción

La creciente madurez de la Ingeniería Dirigida por Modelos (MDE) está promoviendo su adopción por grandes compañías [1][2] que se benefician en términos de productividad, calidad y reuso. Sin embargo, la aplicación de MDE en este contexto requiere de herramientas de escala industrial que puedan operar con modelos muy grandes y complejos. Una operación básica de dichas herramientas es la persistencia de modelos y su acceso, debiendo satisfacer dos requisitos esenciales: escalabilidad e integración.

Uno de los principales obstáculos para la adopción de MDE en la industria es la *escalabilidad* de las herramientas a la hora de acceder a modelos grandes. Como se dice en [3], *"la escalabilidad es lo que echa para atrás a un número importante de potenciales usuarios"*. Un enfoque para abordar la escalabilidad es la partición de modelos mediante mecanismos de modularización propios del lenguaje de modelado [3]. Sin embargo, la complejidad de los modelos grandes dificulta su particionado automático en fragmentos que sean fácilmente accesibles [4], por lo que es imprescindible tener una solución de persistencia de modelos escalable. Para la serialización (es decir, persistencia) de modelos se suele usar el formato XMI (XML Metadata Interchange) [5], el cual requiere la carga completa de un modelo para poder acceder a cualquiera de sus elementos, lo cual puede desbordar la memoria del cliente si dicho modelo es demasiado grande. Los

Este trabajo ha sido financiado por el Ministerio de Ciencia de España (proyecto TIN2009-11555) y la Fundación Séneca (beca 14954/BPS/10).

A. Ruíz, L. Iribarne (Eds.): JISBD'2012, pp. 521-534, ISBN:978-84-15487-28-9.
Jornadas SISTEDES'2012, Almería 17-19 sept. 2012, Universidad de Almería.