

Transformación de modelos conceptuales mediante uso de XSLT

Juan Pablo Fernández Taurant, Claudia Castro, Marcelo Marciszack

Dpto. Ingeniería en Sist. de Información/ Facultad Regional Córdoba/ Universidad Tecnológica Nacional

{ jtaurant, ingclaudiacaastro, marciszack }@gmail.com

Resumen

El objetivo de este trabajo es presentar el análisis de una alternativa que brinde soporte a la gestión de requerimientos de software, facilitando su trazabilidad. Concretamente, nos centraremos en el uso del lenguaje de transformaciones XSLT, que nos permitirá definir diferentes reglas de transformación, las cuales, utilizaremos para realizar transformaciones automáticas bidireccionales entre los modelos generados durante el proceso de captura de requerimientos, permitiendo mantener la trazabilidad de los mismos, independientemente de los cambios o modificaciones que puedan sufrir a lo largo de su ciclo de vida.

Palabras clave: *requerimientos, Transformación de modelos, conceptual, trazabilidad, XSLT, UML, BPMN.*

Contexto

El presente trabajo se encuentra en el marco del proyecto “Validación de Requerimientos a través de Modelos Conceptuales”, que se encuentra consolidado dentro de la línea de investigación de Sistemas de Información en el Dpto. de Ingeniería en Sistemas de Información de la Universidad Tecnológica Nacional, Facultad Regional Córdoba.

El cual, busca dar solución a uno de los principales problemas de la ingeniería en

sistemas relacionado a la elicitación y especificación de requerimientos, que vincula las distintas etapas del proceso de desarrollo de software manteniendo la trazabilidad de los mismos hasta su validación e implementación.

Objetivos de trabajo

Los objetivos del presente trabajo, se encuentran alineados dentro de los objetivos propuestos dentro del Proyecto de investigación, los cuales son los siguientes son los siguientes:

- Establecer marco teórico metodológico para la gestión de requerimientos de Software.
- Definir propuesta metodológica para la especificación de requerimientos.
- Validación de modelos conceptuales en la especificación de Requerimientos.
- Posibilitar en todo momento la trazabilidad de requerimientos desde su captura, modelado y seguimiento.
- Construir e implementar una herramienta de software para la gestión de requerimientos y transformación automática de modelos.

Introducción

El proceso de captura de requerimientos es esencial para conocer las actividades de una organización y clave para el éxito del sistema de información a desarrollar, por lo que resulta de vital importancia mantener la trazabilidad de los mismos a lo largo del proceso de desarrollo. Por otra parte, están comprendidas en este proceso todas las actividades necesarias para crear y mantener los requerimientos de un sistema, estableciendo los lazos de comunicación entre el dominio del modelado del negocio y el dominio del sistema.

Sin embargo, esta comunicación no siempre es tan sencilla, ya que con frecuencia se ve dificultada por la existencia de una distancia semántica entre los dominios organizacional e informático[1], y por diferencias de formación y vocabulario entre las partes involucradas.

Para cerrar estas brechas de comunicación existen en la industria del modelado de negocio estándares ampliamente utilizados como Unified Modeling Language(UML)[2], y Business Process Modeling Notation (BPMN)[3]. Siendo BPMN más orientado al dominio organizacional, de fácil lectura, y más entendible para las partes.

Independientemente de la metodología utilizada, mantener la trazabilidad de requerimientos resulta una actividad compleja. Para esto, todos los modelos creados durante el proceso de desarrollo deben ser mantenidos y actualizados. La creación y mantención de modelos se realiza de forma manual, generando con gran frecuencia inconsistencias entre modelos, en detrimento de la trazabilidad de requerimientos.

En función de estas necesidades, sería de gran utilidad una herramienta que nos permita lograr la automatización total o parcial del

proceso de captura de requerimientos, en donde los requerimientos de software son obtenidos a partir del modelo de negocio realizando diferentes transformaciones entre modelos.

La aplicación de estas transformaciones permitirá que los modelos creados se mantengan actualizados, ya que los cambios realizados en uno de ellos serán propagados a los modelos restantes. De esta forma se contribuye a la trazabilidad de requerimientos, facilitando su gestión y validación.

- **Aplicación de transformaciones**

Las transformaciones entre modelos pueden llevarse a cabo de diferentes maneras. Una de ellas es mediante el estándar para transformación de modelos Query View Transformation Language (QVT) [4], creado y mantenido por la Object Management Group (OMG).

El principal problema que posee este estándar es la falta de disponibilidad en el mercado de herramientas de modelado con procesador QVT, más aún con soporte BPMN, ya que generalmente soportan sólo UML. Otro problema radica en la falta de interoperabilidad, ya que cada herramienta posee su propia representación interna de los modelos. Estos inconvenientes evidencian la necesidad de seleccionar una forma alternativa de transformación de modelos.

Otro tipo de herramientas disponibles para la representación y transformación de modelos son las basadas en Extensible Markup Language (XML) [5]. Para la representación de modelos se encuentran los estándares XML Metadata Interchange (XMI) [6] de la OMG, y XML Process Definition Language (XPDL) [7], creado por la Workflow Management Coalition (WfMC).

Por otro lado se encuentra el estándar Extensible Stylesheet Language

Transformations (XSLT) [8], también basado en XML, que permite realizar las transformaciones entre modelos.

- **Propuestas para la obtención de requerimientos**

Analizaremos dos propuestas para la obtención de requerimientos de software partiendo del modelo de negocio. Utilizaremos un dominio de ejemplo donde aplicaremos cada propuesta, indicando las ventajas y desventajas de cada una en función de los resultados obtenidos.

En la primera propuesta la obtención de requerimientos de software se realizará a partir del modelo de negocio en BPMN. En el que analista deberá seleccionar en forma manual las actividades de negocio que deben ser mapeadas al SI.

Para esto utilizaremos algún estereotipo de BPMN que servirá como marca para detectar si una actividad debe ser mapeada o no al SI. En principio, una de las opciones es utilizar el estereotipo “Script” de BPMN, que se utiliza para representar acciones que se realizar en forma automática.

Teniendo como objetivo liberar al analista de la tarea de seleccionar cada una de las

actividades, asumiremos que todas serán mapeadas al SI, exceptuando las indicadas por el analista, que representaremos con el estereotipo “Manual” de BPMN.

El siguiente paso consiste en generar mediante transformaciones XSLT, un diagrama de actividades UML por cada actividad de negocio que deba mapearse al SI. Cada diagrama de actividades generado deberá ser completado por el analista con las actividades de SI que correspondan.

El proceso inicial de transformación de BPMN a Diagrama de actividades UML (UML-AD) se realiza por dos razones. Una de ellas es debido a las similitudes existentes entre ambas notaciones. La otra se debe a la posibilidad de representar diferentes patrones de modelado con ambas notaciones, lo que nos permite realizar la transformación de un modelo a otro con relativamente poca pérdida de información.

Adicionalmente, obtendremos a partir de cada diagrama de actividades, diagramas de casos de uso (UML-UC) y Diagramas de clases de análisis mediante transformaciones XSLT.

Podemos observar esta propuesta en la figura 1, donde también se indica la etapa en la que se realizará la validación de requerimientos.

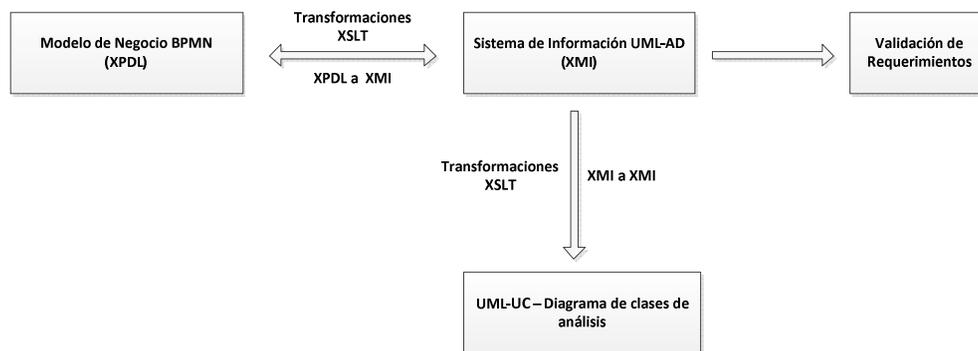


Figura 1 – Transformaciones BPMN a UML-AD

En la segunda propuesta se propone al igual que en la primera, la obtención de

requerimientos de software a partir del modelo de negocio en BPMN. En este caso

generaremos por medio de transformaciones un nuevo diagrama en BPMN para cada actividad del modelo de negocio que deba ser mapeada al SI. Cada diagrama se utilizará como plantilla para que el analista realice la descripción en forma manual de la actividad de negocio en términos de nuevas actividades BPMN que corresponderán cada una de ellas, a requerimientos del sistema de información a desarrollar. También obtendremos a partir de transformaciones XSLT diagramas de

actividades, diagramas de casos de uso y diagramas de clases de análisis que servirán como vistas compatibles con UML.

En la figura 2 se puede observar la estructura de la propuesta y sus transformaciones, indicando la etapa donde se validarán los requerimientos.

Posteriormente, evaluaremos los resultados obtenidos mediante la aplicación de ambas propuestas en un mismo dominio de ejemplo.

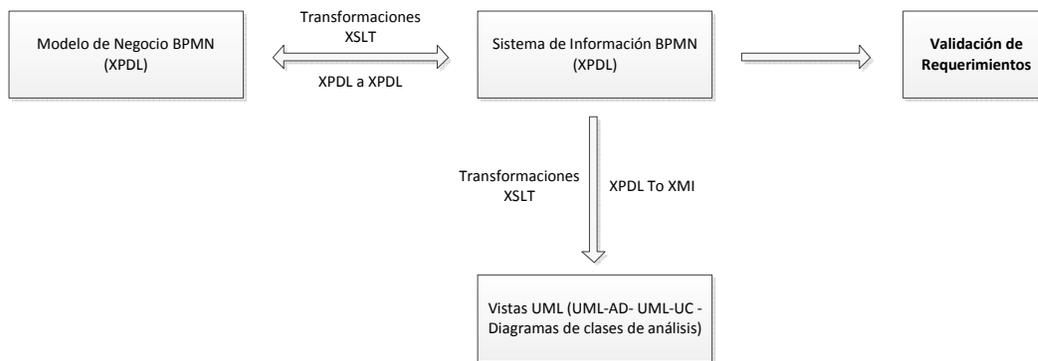


Figura 2 – Transformaciones BPMN a BPMN

Resultados obtenidos

Analizando los resultados de la aplicación de cada propuesta en un caso de estudio de ejemplo, nos encontramos con que la obtención de requerimientos puede realizarse en forma parcial, ya que la pérdida de información producida en cada transformación se ve reflejada en los modelos resultantes. Dependiendo del tipo de modelo que se transforme podemos observar en el proceso la pérdida de relaciones, asociaciones, multiplicidades y hasta de elementos completos.

En la primera propuesta se plantea el uso de la notación BPMN solamente para el modelado de negocio. Mediante la transformación del modelo de negocio en

BPMN a UML-AD, el proceso de obtención de requerimientos se obtiene realizando transformaciones entre artefactos UML. La segunda propuesta en cambio, utiliza la notación BPMN para el modelado de negocio y para describir los requerimientos del SI, esto representa una desventaja si tenemos en cuenta que notación BPMN fue creada para modelar procesos de negocio.

Desde el punto de vista de la obtención de requerimientos a partir del modelo negocio, se hace evidente la necesidad de intervención del analista para la descripción de cada procesos de negocio.

Otro factor de gran importancia además de la obtención de requerimientos es su trazabilidad a lo largo de todo el proceso de transformación. En la primera propuesta, el

uso de diferentes notaciones hace que la pérdida de información asociada a la transformación entre artefactos sea mayor que en la segunda propuesta, que propone el uso de la misma notación tanto para modelar el negocio como los requerimientos del SI. Esta pérdida de información dificulta el proceso de generación de transformaciones inversas que nos permita reconstruir el modelo de nivel superior frente a los cambios realizados en los modelos de menor nivel, impidiendo la trazabilidad de requerimientos.

Formación de Recursos Humanos

Este proyecto al estar circunscripto dentro de otro proyecto de investigación de I+D consolidado y al desarrollarse dentro del ámbito académico de la Universidad, prevé la transferencia de los conocimientos que se vayan adquiriendo, como así también la formación de los recursos humanos que participan del mismo. El mismo cuenta con tres becarios de investigación, dos tesis de la maestría en Ingeniería en Sistemas de Información y un doctorando de Ingeniería de Software que desarrollan sus trabajos en el ámbito del proyecto. Se prevé vinculaciones con otras redes de investigación, tratando de contribuir el conocimiento científico-técnico para su posterior transferencia a nivel social.

Bibliografía

1. Taylor-Cummings, A., 1998. Bridging the user-IS gap: a study of major information systems projects. *Journal of Information Technology*, 13, 29-54.
2. OMG. Unified Modelling Language: Superstructure Version 2.0 (online), Julio 2005, <http://www.omg.org>
3. Object Management Group. Business Process Modeling Notation (BPMN). http://www.omg.org/technology/documents/b_r_pm_spec_catalog.htm, version 1.2, 3 January 2009.
4. Object Management Group: MOF Query / Views / Transformations. Version 1.0, April 2008. http://www.omg.org/technology/documents/modeling_spec_catalog.htm.
5. World Wide Web Consortium: Extensible Markup Language (XML). Version 1.0(fifth edition), 26 November 2008. <http://www.w3.org/XML/>
6. Object Management Group: XML Metadata Interchange (XMI). version 2.1.1, 1 December 2007. http://www.omg.org/technology/documents/modeling_spec_catalog.htm#XMI.
7. <http://www.wfmc.org/xpdl.html>.
8. World Wide Web Consortium: XSL Transformations (XSLT). version 1.0, 16 November 1999. <http://www.w3.org/TR/xslt>