

Novática, revista fundada en 1975 y decana de la prensa informática española, es el órgano oficial de expresión y formación continua de **ATI** (Asociación de Técnicos de Informática), organización que edita también la revista **REICIS** (Revista Española de Innovación, Calidad e Ingeniería del Software). **Novática** co-edita, asimismo **UPGRADE**, revista digital de **CEPIS** (Council of European Professional Informatics Societies), en lengua inglesa, y es miembro fundador de **UPENET** (**UPGRADE** European **NET**work).

<<http://www.ati.es/novatica/>>
 <<http://www.ati.es/reicis/>>
 <<http://www.cepiss.org/upgrade/>>

ATI es miembro fundador de **CEPIS** (Council of European Professional Informatics Societies) y es representante de España en **IFIP** (International Federation for Information Processing); tiene un acuerdo de colaboración con **ACM** (Association for Computing Machinery), así como acuerdos de vinculación o colaboración con **AdaSpain**, **AIZ**, **ASTIC**, **RITSI** e **HispaLinux**, junto a la que participa en **ProInnova**.

Consejo Editorial

Ignacio Aguillo Sousa, Guillem Aínsa González, María José Escalona Cuaremas, Rafael Fernández Calvo (presidente del Consejo), Jaime Fernández Martínez, Luis Fernández Sanz, Didac Lopez Viñas, Celestino Martín Alonso, José Onofre Montesa Andrés, Francesc Noguera Puig, Ignacio Pérez Martínez, Andrés Pérez Payeras, Viktu Pons i Colomer, Juan Carlos Vilgo López

Coordinación Editorial

Llorenç Pagés Casas <pages@ati.es>

Composición y autedición

Jorge Llácer Gil de Ramales

Traducciones

Grupo de Lengua e Informática de ATI <<http://www.ati.es/ql/engua-informatica/>>

Administración

Tomas Brunete, María José Fernández, Enric Camarero, Felicidad López

Secciones Técnicas - Coordinadores

Acceso y recuperación de la Información

José María Gómez Hidalgo (Optenet), <mgomez@oai.uhu.es>

Manuel J. María López (Universidad de Huelva), <manuel.mana@diessia.uhu.es>

Administración Pública electrónica

Francisco López Crespo (MAE), <flc@ati.es>

Arquitecturas

Enrique F. Torres Moreno (Universidad de Zaragoza), <enrique.torres@unizar.es>

Jordi Tubella Morgades (DAC-UPC), <jordit@ac.upc.es>

Auditoría SITIC

Marina Touriño Troilito, <marinatourino@marinatourino.com>

Manuel Palao García-Suelto (ATI), <manuel@palao.com>

Derecho y tecnologías

Isabel Hernando Collazos (Fac. Derecho de Donostia, UPV), <isabel.hernando@ehu.es>

Elena Davara Fernández de Marcos (Davara & Davara), <edavara@davara.com>

Enseñanza Universitaria de la Informática

Cristóbal Pareja Flores (DSIP-UCM), <cpajeflores@dsip.ucm.es>

J. Ángel Velázquez Turbellida (DLSI, URJC), <angel.velazquez@urjc.es>

Entorno digital personal

Andrés Martín López (Univ. Carlos III), <amarin@uicm.es>

Diego Gachet Pérez (Universidad Europea de Madrid), <gachet@uem.es>

Estándares Web

Encarna Quesada Ruiz (Virat), <encarna.quesada@virat.com>

José Carlos del Arco Prieto (TCP Sistemas e Ingeniería), <icarco@gmail.com>

Gestión del Conocimiento

Juan Baiget Solé (Cap Gemini Ernst & Young), <jbaiget@ati.es>

Informática y Filosofía

José Ángel Olivares Varela (Escuela Superior de Informática, UCLM), <joseangel.olivares@uclm.es>

Karim Gherab Martin (Harvard University), <kgherab@gmail.com>

Informática Gráfica

Miguel Chover Sellés (Universitat Jaume I de Castellón), <chover@lsl.uji.es>

Rodrigo Vivó Hernández (Eurographics, sección española), <rvivo@dsic.upv.es>

Ingeniería del Software

Javier Dolado Cosín (DLSI-UPV), <dolado@si.ehu.es>

Daniel Rodríguez García (Universidad de Alcalá), <daniel.rodriguez@uah.es>

Inteligencia Artificial

Vicente Boti Navarro, Vicente Julián Inglada (DSIC-UPV), <vbotti.vinglada@dsic.upv.es>

Interacción Persona-Computador

Pedro M. Latorre Andrés (Universidad de Zaragoza, AIPO), <platorre@unizar.es>

Francisco L. Gutiérrez Vela (Universidad de Granada, AIPO), <lgutierrez@ugr.es>

Lengua e Informática

M. del Carmen Ugarte García (ATI), <cuarte@ati.es>

Lenguajes Informáticos

Oscar Belmonte Fernández (Univ. Jaime I de Castellón), <belleim@lsl.uji.es>

Inmaculada Coma Tassy (Univ. de Valencia), <inmaculada.coma@uv.es>

Lingüística computacional

Xavier Gómez Guinovart (Univ. de Vigo), <xgg@uvigo.es>

Manuel Palomar (Univ. de Alicante), <mpalomar@dsi.ua.es>

Mundo estudiantil y jóvenes profesionales

Federico G. Mon Trotti (RITSI), <gnu.trode@gmail.com>

Mikel Salazar Peña (Asociación Jóvenes Profesionales, Junta de ATI Madrid), <mikelboi_uni@yahoo.es>

Profesión Informática

Rafael Fernández Calvo (ATI), <rfcalvo@ati.es>

Miguel Santes Gurió (ATI), <msantes@ati.es>

Redes y servicios telemáticos

José Luis Marzo Lázaro (Univ. de Girona), <joseluis.marzo@udg.es>

Juan Carlos López López (UCLM), <juancarlo.lopez@uclm.es>

Robótica

José Cortés Arenas (Sopra Group), <jccortes@gmail.com>

Juan González Gómez (Universidad Carlos III), <jian@izarobotics.com>

Seguridad

Javier Arellito Bertolin (Univ. de Deusto), <jaarellito@eside.deusto.es>

Javier López Muñoz (ETS Informática-UMA), <jlm@cc.uma.es>

Sistemas de Tiempo Real

Alejandro Alonso Muñoz, Juan Antonio de la Puente Alfaro (DIT-UPM),

<aalonso.jpunte@dit.upm.es>

Software Libre

Jesús M. González Barahona (Universidad Politécnica de Madrid), <israel.herraz@upm.es>

Israel Herráz Tabernero (UAJ), <isra@herraz.org>

Tecnología de Objetos

Jesus Garcia Molina (DIS-UM), <jmolina@um.es>

Gustavo Rossi (LPIA-UNLP, Argentina), <gustavo@sol.inf.unlp.edu.ar>

Tecnologías para la Educación

Juan Manuel Dodero Beardo (UC3M), <dodero@inf.uc3m.es>

César Pablo Córcoles Briongo (UOC), <ccorcoles@uoc.edu>

Tecnologías y Empresa

Didac Lopez Viñas (Universitat de Girona), <didac.lopez@ati.es>

Francisco Javier Cantas Sánchez (Indra Sistemas), <fjcasas@gmail.com>

Tendencias tecnológicas

Alonso Alvarez Garcia (TID), <aad@tid.es>

Gabriel Martí Fuentes (Interbits), <gabi@atinet.es>

TID y Turismo

Andrés Aguesyo Maldonado, Antonio Guevara Plaza (Univ. de Málaga), <aguyao.guevara@cc.uma.es>

Las opiniones expresadas por los autores son responsabilidad exclusiva de los mismos. **Novática** permite la reproducción, sin ánimo de lucro, de todos los artículos, a menos que lo impida la modalidad de © o copyright elegida por el autor, debiéndose en todo caso citar su procedencia y enviar a **Novática** un ejemplar de la publicación.

Coordinación Editorial, Redacción Central y Redacción ATI Madrid

Padilla 66, 3º dcha., 28006 Madrid
 Tfn. 91 4029391; fax. 91 3093685 <novatica@ati.es>

Composición, Edición y Redacción ATI Valencia

Av. del Reino de Valencia 23, 46005 Valencia
 Tfn./fax 963330392 <secreval@ati.es>

Administración y Redacción ATI Cataluña

Via Llaletana 46, ppal. 1º, 08003 Barcelona
 Tfn. 93 41 25235; fax. 93 41 27713 <secregen@ati.es>

Redacción ATI Aragón

Lagasca 9, 3-B, 50006 Zaragoza.
 Tfn./fax 976235181 <secretara@ati.es>

Redacción ATI Andalucía

<secreand@ati.es>

Redacción ATI Galicia

<secregal@ati.es>

Suscripción y Ventas <<http://www.ati.es/novatica/interres.html>>, ATI Cataluña, ATI Madrid

Publicidad

Padilla 66, 3º dcha., 28006 Madrid
 Tfn. 91 4029391; fax. 91 3093685 <novatica@ati.es>

Imprenta: Derra S.A., Juan de Austria 66, 08005 Barcelona.
 Depósito legal: B 15.154-1975 - ISSN: 0211-2124; CODEN NOVACE

Portada: Luces técnicas - Concha Arias Pérez / © ATI
 Diseño: Fernando Agresta / © ATI 2003

editorial

La fuerza de trabajo en Informática

> 02

en resumen

Construcción sistemática de edificios del conocimiento

> 02

Llorenç Pagés Casas

Noticias de IFIP

Reunión del TC 13 (Human-Computer Interaction)

> 03

Julio Abascal González

Reunión del Consejo de IFIP

> 06

Ramon Puigjaner Trepal

Coordinación editorial

> 07

monografía

Ingeniería de Sistemas de Aprendizaje Electrónico

(En colaboración con UPGRADE)

Editores invitados: José Luis Sierra Rodríguez, Antonio Sarasa Cabezuolo, Demetrios G.

Sampson, Kinshuk, Ignacio Aedo Cuevas

Presentación. Diseño y construcción de Sistemas de Aprendizaje Electrónico

> 08

Ignacio Aedo Cuevas, Kinshuk, Demetrios G. Sampson, Antonio Sarasa Cabezuolo,

José Luis Sierra Rodríguez

Ingeniería de Sistemas de Aprendizaje Electrónico y esfuerzos de estandarización:

Un caso de estudio relativo a los Almacenes de Objetos Didácticos

> 10

Antonio Sarasa Cabezuolo, Daniel Rodríguez Cerezo, José Luis Sierra Rodríguez

Sistemas PDP de contenidos educativos audiovisuales:

Desarrollo de una solución para la Universidad de La Laguna

> 13

Carina S. González González, David Cabrera Primo, Daniel López Barrios, Antonio Barroso Díaz

Patrones de comunicación en sistemas tutores inteligentes basados

en componentes

> 17

Géraldine Ruddeck, Dennis Maciuszek, Martina Weicht, Alke Martens

Sistemas Tutores Inteligentes con SCORM

> 23

Gustavo Soares Santos, Álvaro Reis Figueira

Servicios eLearning en Moodle 2.0

> 29

Miguel A. Conde González, Alberto del Pozo de Dios, Francisco J. García Peñalvo

Un caso práctico de integración de agentes y servicios en entornos de

eLearning mediante espacios de tuplas

> 34

Francisco Jurado Monroy, Ana I. Molina Díaz, Miguel A. Redondo Duque, Manuel Ortega Cantero

Hacia la invocación de servicios de e-Learning adaptativo: Aplicación a un

simulador de navegación y pesca

> 40

Valérie Monfort, Maha Khemaja

Una metodología de diseño para el aprendizaje ubicuo

> 46

Jihen Malek, Mona Laroussi, Alain Derycke, Henda Ben Ghezala

secciones técnicas

Estándares web

Una aproximación basada en modelos para la ejecución automática

de modelos de procesos de negocio

> 51

Valeria de Castro, Javier Fabra Caro, Pedro Álvarez Pérez-Aradros, Esperanza Marcos Martínez

Redes y servicios telemáticos

Diseño e Implementación de Video Streaming en Redes de Sensores Inalámbricas

> 57

David Rodenas Herráiz, Antonio Javier García Sánchez, Felipe García Sánchez

Robótica

Arquitecturas de control de robots

> 62

José Cortés Arena

Referencias autorizadas

> 65

sociedad de la información

Programar es crear

Mi número de Erdos (solución)

> 72

Julio Javier Castillo, Diego Javier Serrano

La Forja

Creación de un Clúster de Alta Disponibilidad con software libre (solución)

> 74

Miguel Vidal López, José Castro Luis

asuntos interiores

Socios Institucionales

> 77

Valeria de Castro¹, Javier Fabra Caro², Pedro Álvarez Pérez-Aradros², Esperanza Marcos Martínez¹

¹Grupo Kybele, Universidad Rey Juan Carlos; ²Departamento de Informática e Ingeniería de Sistemas, Universidad de Zaragoza

<{valeria.decastro,esperanza.marcos}@urjc.es>, <{jfabra,alvaper}@unizar.es>

Una aproximación basada en modelos para la ejecución automática de modelos de procesos de negocio

Este artículo fue seleccionado para su publicación en **Novática** de entre las ponencias presentadas a las VI Jornadas Científico-Técnicas en Servicios Web y SOA (JSWEB2010) celebradas en Valencia y de las que ATI fue entidad colaboradora.

1. Introducción

La rápida evolución de las tecnologías de servicios Web y del paradigma de Computación Orientado a Servicios (*Service-Oriented Computing*) en los últimos años ha motivado el interés en ambos en el ámbito de la gestión de los procesos de negocios o *Business Process Management* (BPM) [24]. BPM se define como un conjunto de tecnologías y estándares para el diseño, ejecución, administración y supervisión de los procesos de negocio, y como una forma de ayudar a la gestión de los frecuentes cambios en los negocios y en las cadenas de valor de las organizaciones [10][24].

Este nuevo ámbito de trabajo, unido a la proliferación de la Arquitectura Orientada a Servicios (*Service Oriented Architecture*, SOA) y al desarrollo basado en servicios, ha impulsado la aparición de nuevos lenguajes para el diseño y la implementación de procesos de negocios basados en servicios Web, tales como el *Business Process Execution Language* (BPEL) [1], que se ha convertido en el estándar *de facto* para la descripción de procesos de negocios ejecutables. Sin embargo, BPEL y, en general, la mayoría de los lenguajes para la implementación de procesos de negocios, presentan problemas importantes tales como sus limitaciones a la hora de ser utilizados en etapas tempranas del proceso de desarrollo de software [23], su dependencia de una tecnología concreta (por ej., servicios Web), o la carencia de una semántica formal por parte de los procesos resultantes que impide el análisis y la verificación de los mismos. Estos problemas incrementan la brecha existente entre los analistas de negocio y los desarrolladores de software, lo que representa una de las principales limitaciones en el área de BPM.

La Ingeniería de Software Dirigida por Modelos (ISDM) o *Model Driven Engineering* (MDE) constituye hoy en día una de las principales herramientas con las que se cuenta a la hora de tratar el problema de la alineación entre el punto de vista de los procesos de negocios de alto nivel y las tecnologías de la información que implementan tales procesos. Una de las mayores propuestas que pone

Resumen: La gestión de procesos de negocio intenta definir un marco para que las organizaciones puedan afrontar de manera eficiente cambios frecuentes en sus procesos. Uno de los mayores desafíos en este entorno sigue siendo la brecha que existe entre los procesos de negocios de alto nivel y quienes los gestionan, y las tecnologías de información que dan soporte a los primeros. La Ingeniería de Software Dirigida por Modelos (ISDM) proporciona herramientas que pueden ayudar en la alineación de modelos de procesos de alto nivel con sus implementaciones correspondientes. La posibilidad de ejecutar modelos de procesos de negocio es uno de los principales beneficios de la ISDM. Este trabajo se centra en este aspecto y presenta una propuesta para el análisis, desarrollo y ejecución de modelos de procesos de negocio en la que se integran dos propuestas anteriores, SOD-M y DENEb. La primera define una aproximación basada en modelos para el desarrollo orientado a servicios de Sistemas de Información que permite obtener modelos de procesos a partir de modelos de negocios de alto nivel; la segunda proporciona una plataforma basada en redes de Petri para la ejecución de dichos modelos de proceso. Para que la integración sea posible es necesario llevar a cabo la transformación de modelos que se describe en este trabajo, y que se ilustra por medio de un caso de estudio desarrollado.

Palabras clave: Desarrollo Orientado a Servicios, ejecución de modelos, Ingeniería de Software dirigida por Modelos, procesos de negocio, transformación de modelos.

es práctica los principios de la ISDM es sin duda, la especificación MDA (*Model Driven Approach*) propuesta por OMG [15].

MDA constituye una importante herramienta para conseguir la alineación descrita anteriormente, dado que proporciona una estructura conceptual que se extiende desde los modelos utilizados por los analistas de negocio hasta los diversos modelos utilizados por los desarrolladores de software, organizándolos de manera tal que unos puedan ser transformados (de forma semiautomática o automática) en otros más detallados o derivados a partir de ellos, permitiendo además llegar a la generación de código.

Sin embargo es evidente que tras varios años desde la aparición de MDA y varias propuestas que utilizan los principios de la ISDM, la utilidad de esta aproximación para el desarrollo de software sigue siendo cuestionable [20]. La principal controversia en este tema se relaciona con la falta de un soporte tecnológico que permita explotar de forma completa los beneficios de la generación automática de código a partir de modelos [21]. La habilidad para generar código abriría nuevas posibilidades a este tipo de soluciones. Por ejemplo, podría ser usada como base para la verificación de procesos de negocio (¿permiten los

procesos de negocio cumplir los requisitos de negocio establecidos?) y su análisis de propiedades. Así, la generación a partir de modelos de código ejecutable se convierte en uno de los retos clave para potenciar la utilización de los principios de la ISDM en el desarrollo de software.

Este trabajo pretende centrarse en este último aspecto, es decir, en aprovechar los beneficios que la ISDM puede aportar a la hora de desarrollar software, y presenta un marco de trabajo basado en modelos para el análisis, desarrollo y ejecución de modelos de procesos de negocio en el que se integran dos propuestas anteriores, SOD-M y DENEb [4][7]. SOD-M define una aproximación basada en modelos que sigue un enfoque orientado a servicios para el desarrollo de software y propone un proceso en el que, partiendo de un modelo de negocio de alto nivel, es posible obtener mediante transformaciones sucesivas un modelo de composición de servicios. Este modelo es un diagrama de actividad que puede después transformarse de manera automática en un proceso ejecutable. DENEb aparece así como una plataforma basada en redes de Petri para la ejecución de procesos flexibles, capaz de ejecutar directamente un modelo de *workflow* generado como resultado de una transfor-

mación que se aplica al modelo de composición de servicios.

La principal aportación de esta propuesta radica en la definición de un marco completo de análisis, desarrollo y ejecución de procesos de negocio que potencia uno de los principales beneficios de la ISDM y que, además, permite acortar la brecha que existe entre los analistas de negocio y desarrolladores de software, que es una de las principales limitaciones a las que se enfrentan en la actualidad las propuestas de BPM.

El resto del artículo se organiza de la siguiente forma. En la **sección 2** se analizan algunas de las principales propuestas relacionadas con este trabajo. En la **sección 3** se introducen brevemente las dos propuestas que sirven de base en este trabajo de integración, SOD-M y DENEb. El proceso de transformación definido entre ambas propuestas se presenta en la **sección 4** y se ilustra en la **sección 5** a través de un caso de estudio. Finalmente, la **sección 6** concluye el trabajo destacando sus principales aportaciones y las líneas de trabajo para el futuro.

2. Estado del arte

La Ingeniería de Servicios (*Service Engineering*, SE) trata de combinar los beneficios de la arquitectura SOA con BPM. De esta forma, se consigue una alineación entre los procesos de negocio y las tecnologías de la información (TI) que facilita el desarrollo rápido de procesos ágiles, capaces de responder a los cambios en los requisitos. Recientemente, la convergencia entre SE y la ISDM ha puesto de manifiesto cómo la utilización de MDA facilita aún más el desarrollo de procesos y servicios basados en SOA a través de modelos [17][15][24]. MDA propone la utilización de diferentes niveles para representar los diferentes puntos de vista de un sistema, desde el mayor nivel de abstracción hasta un nivel tecnológico concreto: nivel independiente de computación (CIM), nivel independiente de plataforma (PIM) y nivel específico de plataforma (PSM), respectivamente [15]. Existen diferentes propuestas que asocian estándares de BPM a estos niveles [14][19].

En el nivel CIM destaca la notación *Business Process Modelling Notation*, BPMN [16]. BPMN permite modelar diagramas de procesos mediante flujos, eventos, actividades y resultados, además de permitir modelar subprocesos. En el nivel PIM, la mayoría de los trabajos apuestan por el *Business Process Definition Metamodel*, BPDM, que normalmente se utiliza junto con UML [14]. Finalmente, en el nivel PSM el estándar predominante es *Business Process Modelling Language*, BPEL [1]. BPEL permite describir la lógica de ejecución de procesos de negocio definiendo sus flujos de control y mediante mecanismos de interacción, gene-

rando como resultado soluciones interoperables y robustas basadas en SOA. Resulta importante destacar que, pese a que en muchos trabajos se comparan BPEL y BPMN como dos iniciativas similares, normalmente el primero es utilizado por programadores y técnicos, mientras que los analistas de negocio usan el segundo.

Partiendo de la descripción de procesos de negocio basados en estándares, resulta clave la generación automática o semiautomática del código fuente correspondiente a su implementación. Pese a que la utilización de los estándares debería facilitar esta tarea, la realidad es todo lo contrario: resultan complicados y tediosos de utilizar por los analistas de negocio en las primeras etapas de desarrollo software [23]. Los principales problemas son que estos estándares apuntan a soluciones basadas casi exclusivamente en servicios Web, no incluyen una plataforma para realizar un diseño *top-down* y no aportan herramientas para realizar el análisis posterior de los procesos. Por tanto, la transformación de un modelo de alto nivel a un lenguaje de composición que implemente el proceso no resulta trivial.

Son varios los trabajos que han propuesto la transformación de BPMN a BPEL. En [25], los autores proponen una transformación entre las estructuras de grafo de BPMN (*flow elements*) y los bloques BPEL (*sequence element*). Sin embargo, no muestran una metodología firme ni una herramienta que implemente la transformación. El *Oracle Business Process Analysis Suite* sí aporta una implementación de la transformación entre estos estándares [6], aunque existen diferencias semánticas entre el modelo inicial y el final, por lo que esta solución sólo es viable para modelos técnicos basados en servicios y no se recomienda para escenarios complejos. En [18], los autores integran el *XML Process Definition Language*, XPDL, con BPMN y BPEL para generar el código BPEL final automáticamente. Sin embargo, esta transformación se limita al intercambio de datos, y las formas más complejas de interacción quedan fuera del objetivo del trabajo. Otra serie de trabajos proponen la utilización de WebML [2], UML [4][2], o xUML [9] para generar esqueletos de código como resultado de la transformación. Resulta evidente que la generación completa de código depende, pues, de una intervención humana posterior.

La finalidad de MDA como puente de unión entre BPM y SOA es la automatización de estos procesos de transformación. Sin embargo, las diferentes alternativas presentadas evidencian serias carencias y limitaciones. En este trabajo abordaremos la posibilidad de integrar dos soluciones a diferentes niveles para proporcionar un enfoque completo, válido y flexible que resuelva los problemas expuestos.

3. Propuestas previas

Este trabajo es el resultado de la integración de dos propuestas previas que, unidas mediante un proceso de transformación de modelos, permite cubrir las etapas que van desde el desarrollo de modelos de procesos de negocios de alto nivel hasta la obtención de modelos de procesos de bajo nivel y la ejecución de los mismos.

Por un lado, se parte de un método para el desarrollo orientado a servicios de Sistemas de Información (SI) llamado SOD-M (*Service Oriented Development Method*) [4]. El proceso de modelado de SOD-M incluye modelos que están en correspondencia con los tres niveles diferentes de abstracción considerados por MDA: CIM, PIM y PSM. A nivel CIM, el proceso comienza con la construcción de los modelos de valor y de proceso y, a través de un proceso de transformaciones de modelos, permite obtener como resultado un modelo de composición de servicios [5]. Por motivos de espacio, en este trabajo sólo nos centraremos en describir el modelo de composición de servicios (*Service Composition Model*). Una descripción detallada del resto de los modelos de SOD-M puede consultarse en [4].

El modelo de composición de servicios es la entrada del proceso de transformación de modelos definido en este trabajo y que permite generar como salida un modelo que podrá ser ejecutado en el marco de la segunda propuesta que integramos en este trabajo.

DENEb (*Development and Execution of iNteroperable dynamic wEB processes*) proporciona un entorno para el desarrollo y ejecución de procesos de negocio flexibles [7]. En DENEb, un proceso de negocio está compuesto por un conjunto de tareas que deben ser ejecutadas en base a determinadas restricciones de orden. La ejecución de una tarea concreta puede representar la invocación a un proveedor externo (un servicio Web, otro proceso de negocio, etc.) o un sistema interno (una base de datos, un recurso hardware, etc.). El núcleo de DENEb consta de tres componentes clave: el *intérprete de workflows*, el *intérprete de conversaciones* y el *broker de mensajes*. Los dos primeros componentes son responsables de la ejecución de la lógica de negocio (*workflows*) y de la lógica de interacción (*roles*) de los procesos, respectivamente. El *broker de mensajes* tiene un doble papel. Por un lado, desacopla la implementación de los dos intérpretes de DENEb, facilitando la comunicación y coordinación entre ambos. Esta comunicación se produce a través del repositorio de mensajes del *broker*, por medio del intercambio de mensajes de control y sincronización. Por otro lado, el *broker* es el responsable de gestionar la comunicación con los proveedores externos (servicios Web, procesos de ne-

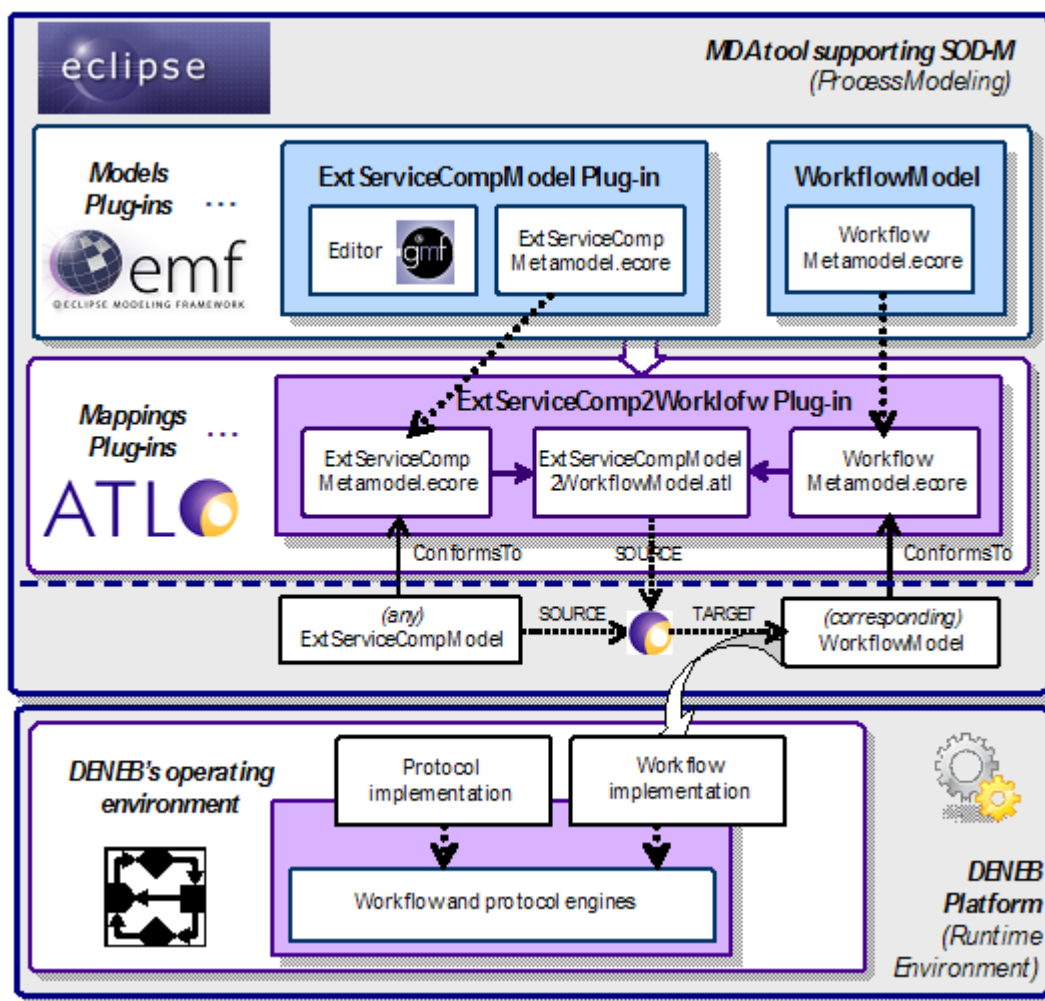


Figura 1. Proceso de transformación propuesto.

gocio, etc.) a través de una serie de componentes especiales, denominados mediadores [7].

SOD-M y DENEb ofrecen a nivel individual contribuciones relevantes desde el punto de vista del ciclo de vida del desarrollo de software. En este trabajo se presenta la integración de estas propuestas con el fin de ofrecer una solución completa para el análisis, diseño, desarrollo, despliegue y ejecución de procesos de negocio. Esta integración está basada en la idea de generar automáticamente procesos de negocio directamente ejecutables por DENEb a partir de los modelos de SOD-M (más concretamente, a partir del modelo de composición de servicios). Desde el punto de vista de otras propuestas similares, nuestra contribución reside en dos hechos: primero, el proceso de generación de código es automático y, segundo, los procesos que se obtienen como resultado son directamente ejecutables (es decir, no son esqueletos de procesos o fragmentos de código que requieren un trabajo extra por parte de los programadores).

Las propuestas ya existentes están basadas fundamentalmente en lenguajes estándar de especificación de procesos (por ejemplo,

BPEL). Estos lenguajes están fuertemente acoplados con las tecnologías de servicios Web (BPEL es un buen ejemplo de este hecho). En este sentido, nuestra propuesta es más flexible gracias a las características de DENEb. Un proceso no sólo puede integrar servicios Web, sino cualquier otro tipo de proveedor accesible a través de la Red (por ejemplo, un sistema GRID). Simplemente se requiere un mediador que facilite esta interacción y que SOD-M represente en sus modelos la ejecución de una tarea por parte de este tipo de proveedor. Por tanto, la integración de ambas propuestas ofrece una solución completa con contribuciones relevantes desde un punto de vista común.

4. Proceso de transformación de modelos

En esta sección describimos la transformación de modelos propuesta. En concreto, nos centraremos aquí en la transformación que permite obtener el modelo de *workflow*, que se ejecutará en la plataforma DENEb, a partir de un *modelo de composición de servicios* de SOD-M. La obtención de este último modelo a partir de modelos de más alto nivel definidos por SOD-M se ha presentado en [5].

La **figura 1** muestra el proceso de transformación de modelos propuesto en este trabajo y detalles sobre la implementación del mismo. Como puede verse en la figura, por un lado se han desarrollado los *plug-ins* necesarios para la representación de los modelos utilizados, y por otro el *plug-in* necesario para la transformación (*mapping*). La implementación de los modelos se ha desarrollado sobre el *Eclipse Modeling Framework (EMF)* [22], y en concreto, se han definido dos *plug-ins*, uno para la representación del modelo de composición de servicios (modelo origen) y otro para el modelo de *workflow* (modelo destino). La implementación de la transformación se ha realizado utilizando el *ATLAS Transformation Language, ATL* [13]. Para ello, las reglas de transformación se han definido previamente en lenguaje natural y posteriormente se han codificado en ATL (*ServiceCompModel2WorkflowModel.atl*). Como puede verse en la **figura 1**, para llevar a cabo la transformación, el programa ATL codificado utiliza la información de los metamodelos de los modelos origen y destino implicados (representados en los ficheros con extensión *.ecore*). En el momento de ejecutar una transformación particular, el programa

Origen	Elementos del modelo de origen	Descripción de la regla	Elementos del modelo de destino	Destino
Modelo de Composición de Servicio		Para cada <i>ServiceActivity</i> del modelo de origen se genera un bloque en el modelo de destino que contiene un lugar (<i>place</i>) de entrada y uno de salida		Modelo de workflow
		Para cada <i>ServiceActivity</i> marcada como inicial, se genera un lugar marcado unido al lugar de entrada al bloque.		
		Para cada <i>ServiceActivity</i> marcada como final, se identifica una transición y un lugar finales unidos al lugar de salida del bloque.		
		Por cada nodo de decisión en el modelo de origen se genera una bifurcación con una condición de guarda en el modelo destino.		
		Por cada nodo de unión de decisión en el origen se genera una unión en el destino.		
		Por cada nodo de unión en el origen se genera una estructura de unión en el destino.		
		Por cada nodo de bifurcación en el origen se genera una estructura de bifurcación en el destino que habilita ramas de ejecución paralelas.		
		Por cada nodo ejecutable en el origen se genera un rol en DENEb y la estructura necesaria en el <i>workflow</i> para instanciarlo y ponerlo en ejecución. Esta estructura se construye usando los valores etiquetados como <i>endpoint</i> , <i>operations</i> , etc., normalmente de acuerdo a un esquema genérico.		

Tabla 1. Reglas de Transformación de SOD-M a DENEb.

toma una instancia del modelo de composición de servicio concreto y genera una instancia del modelo de *workflow* correspondiente.

Por motivos de espacio, en este trabajo solo presentaremos las reglas de transformación entre los modelos, las cuales se ilustrarán con un ejemplo en la siguiente sección. Los metamodelos de cada uno de los modelos utilizados pueden consultarse en [3] y [26].

En la **tabla 1** presentamos las reglas de transformación definidas en lenguaje natural para transformar un modelo de composición de servicios en un modelo de *workflow*. Estas reglas se ilustran en la siguiente sección a través de un caso de estudio.

5. Ejemplo de aplicación

En esta sección se ilustra el proceso de transformación descrito en el apartado anterior, utilizando como caso de estudio el sistema GesIMED, un sistema Web para la gestión y el procesamiento de imágenes médicas desarrollado en la Universidad Rey Juan Carlos [8][11]. El objetivo del sistema es ofrecer a los investigadores en neurociencias una base de datos de imágenes médicas disponible a través de la Web, sobre la que sea posible realizar todo tipo de consultas, y una serie de procedimientos normalizados. Esto permite a dichos investigadores realizar el procesamiento y análisis de las imágenes almacenadas, cuyos resultados son también almacenados en la misma base de datos, de modo que

puedan ser utilizados en estudios futuros. Este sistema ofrece a los investigadores en neurociencias tres servicios concretos: *servicio de almacenamiento y consulta de imágenes médicas*, *servicio de procesamiento de imágenes médicas* y *servicio de visualizaciones de imágenes médicas* (creación de imágenes 3D a partir de varias imágenes 2D).

Este sistema se ha desarrollado de forma completa utilizando los modelos propuestos por SOD-M en trabajos previos [4]. En este artículo nos vamos a centrar sólo en uno de los modelos de composición de servicios construidos, y vamos a mostrar el modelo de *workflow* generado como resultado del proceso de transformación.

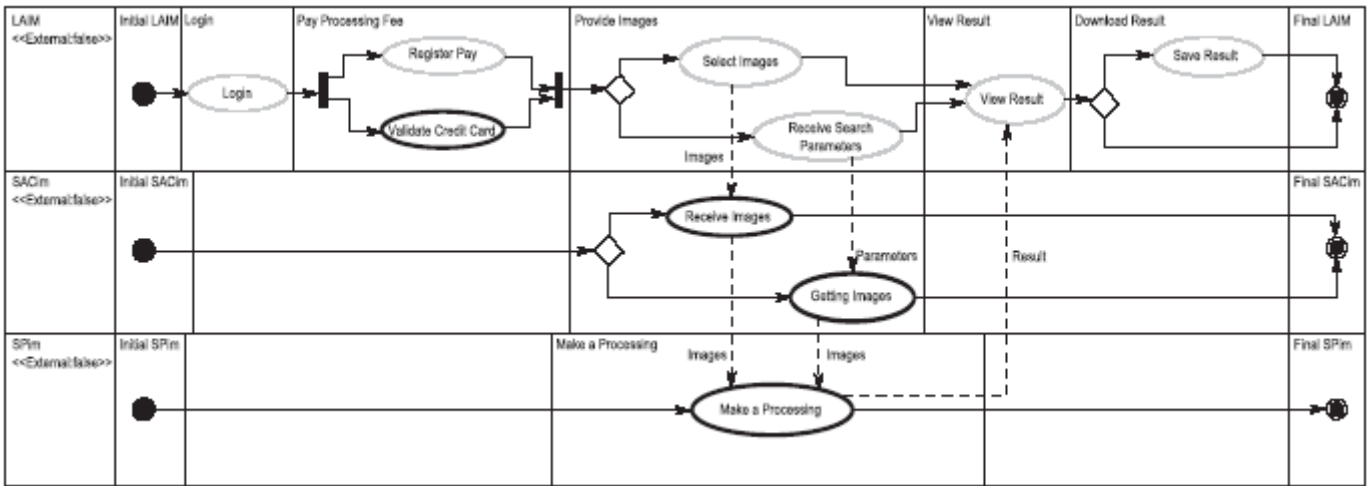


Figura 2. Modelo de composición de servicios para el servicio *Obtain Processed Images*

La figura 2 muestra el modelo de composición de servicio construido para el servicio de procesamiento de imágenes (*Obtain processed images*). Para llegar a la construcción de este modelo, SOD-M define un conjunto de transformaciones que toman como elementos de entrada elementos del modelo de valor y del modelo de procesos de servicio.

La actividad *ValidateCreditCard* se representa como un servicio Web en este modelo, puesto que es un servicio estándar ofrecido por varias organizaciones. También las funcionalidades ofrecidas por los servicios de procesamiento de imágenes médicas (*SPm*) y almacenamiento y consulta de imágenes médicas (*SACm*) se han identificado como

servicios Web, dado que este comportamiento se ofrece como resultado de una invocación del servidor central de la aplicación a los servicios correspondientes.

Una parte del modelo obtenido como resultado de la transformación aplicada al modelo de la figura 2 se muestra en la figura 3.

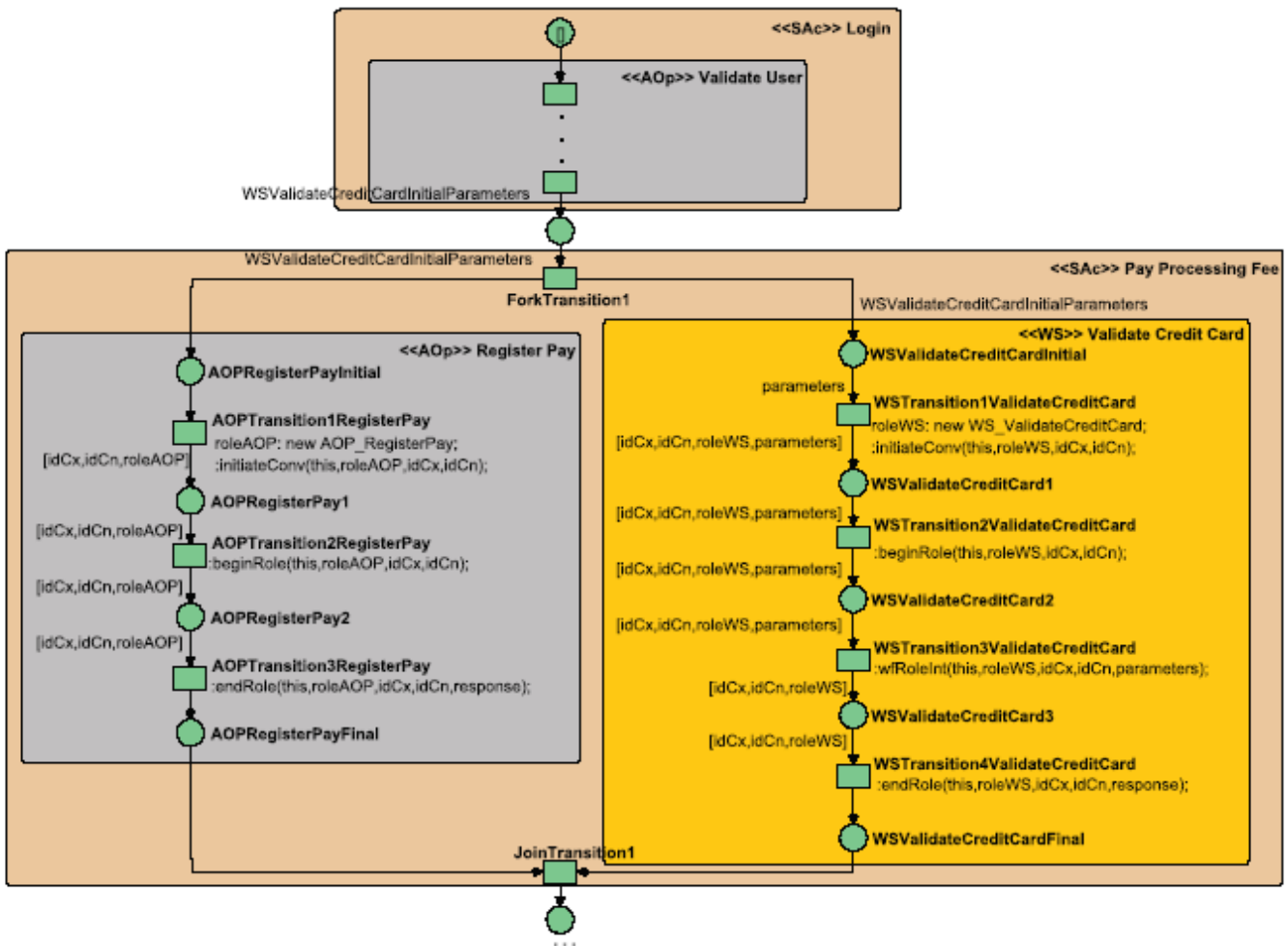


Figura 3. Modelo de *workflow* correspondiente al servicio *Obtain Processed Images* y generado mediante la transformación.

Esta figura representa la parte del *workflow* del proceso DENEb resultante que comprende de la actividad de registro del pago (*RegisterPay*) y el servicio Web de validación de tarjeta de crédito (*ValidateCredit Card*). Debemos aclarar que el modelo de *workflow* obtenido como resultado de la transformación ATL tiene una estructura de árbol. Sin embargo, para mejorar la comprensión del caso de estudio analizado, en la **figura 3** dicho modelo se muestra con el editor de modelo de la herramienta DENEb, lo que permite visualizarlo gráficamente como una red de Petri de alto nivel.

El modelo de la **figura 3** se ha obtenido de forma automática a partir de la implementación de las reglas recogidas en la **tabla 1**. Centrándonos en la **figura 3**, podemos ver que el pago por el procesamiento de la imagen se ha transformado en un bloque etiquetado como `<<AOp>> RegisterPay` (el estereotipo *AOp* se utiliza para indicar que es una operación y no un servicio Web).

Por otro lado, el servicio Web de validación de tarjetas de crédito se ha transformado en un bloque etiquetado como `<<WS>> ValidateCreditCard`. Ambos bloques se encuentran incluidos en un bloque de nivel superior que se ha etiquetado como `<<SAc>> PayProcessingFee`. Las inscripciones y elementos que forman parte de los bloques se obtienen como se explica en la **tabla 1**, usando los valores etiquetados como *endpoint*, *operations*, etc. de las actividades y de acuerdo a un esquema genérico predefinido. Por razones de espacio no podemos mostrar los protocolos generados ni describir como se lleva a cabo la ejecución de modelo de *workflow* obtenido sobre la plataforma DENEb, aunque los detalles de dicha ejecución pueden consultarse en [7].

5. Conclusiones y trabajo futuro

En este artículo hemos presentado la integración de SOD-M y DENEb. El resultado es un marco basado en modelos para el análisis, desarrollo y ejecución de procesos de negocio. Este marco integra la visión de los analistas de negocio (objetivos y requisitos) con la visión de los desarrolladores de software (código ejecutable). Esto es posible gracias a la generación automática de código ejecutable por DENEb a partir de los modelos de SOD-M. En relación a otras propuestas similares, las principales contribuciones de nuestra solución son: el código resultante implementa el proceso de negocio completo (en vez de fragmentos o esqueletos del proceso que deben ser completados por los desarrolladores), la generación de este código es automática y, por último, los procesos implementados pueden integrar servicios o aplicaciones programados en tecnologías heterogéneas (no sólo programados en base a servicios Web, como es el caso de la mayoría

de los lenguajes de procesos como, por ejemplo, BPEL).

Por otro lado, el código generado durante el proceso de transformación tiene una semántica formal. Esta característica abre una nueva vía de trabajo futuro: la verificación y análisis previo a su ejecución de los procesos de negocio resultantes. En [12] se propuso una metodología para verificar y analizar propiedades de comportamiento de procesos de negocio representados con el mismo formalismo. Los resultados de este trabajo podrían ser fácilmente adaptados para ser aplicados sobre los procesos obtenidos a partir de los modelos de SOD-M. Posteriormente, esta metodología de análisis y sus correspondientes herramientas podrían ser integradas en el marco objeto en este trabajo.

Agradecimientos

Este trabajo se ha llevado a cabo en el marco de los proyectos Model CAOS (TIN2008-03582) y TIN2010-17905, financiados por el Ministerio de Ciencia y Tecnología de España.

Referencias

- [1] OASIS. BPEL 2.0 specification, <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>.
- [2] M. Brambilla. Generation of WebML web application models from business process specifications. *6th International Conference on Web Engineering*, July 11-14, 2006, Palo Alto, California, USA.
- [3] J. Fabra, V. De Castro, P. Álvarez, E. Marcos. *Automatic execution of business process models: exploiting full benefits of Model-Driven*. Enviado a Journal of Systems and Software. Elsevier publishing.
- [4] V. De Castro, E. Marcos, R. Wieringa. Towards a Service-oriented MDA-Based Approach to the Alignment of Business Processes with it Systems: From the Business Model to a Web Service Composition Model. *International Journal on Cooperative Systems* 18(2), pp. 225-260, 2009.
- [5] V. De Castro, E. Marcos, J.M. Vara. Applying cim-to-pim model transformations for the service-oriented development of information systems. *Information and Software Technology* Vol. 53(19), pp. 87- 105, 2011.
- [6] L. Dikmans. *Transforming BPMN into BPEL: Why and How*. Oracle Technology Network, September 2008.
- [7] J. Fabra, P. Álvarez, J.A. Bañares, J. Ezpeleta. Runtime Protocol Binding: Flexible Service Integration by Means of Flexible Service Interactions. *2008 IEEE International Conference on Services Computing SCC'08*, pp. 291-298, 2008.
- [8] GesIMED System. <http://ariadna.escet.urjc.es/gesimed/>, 2008.
- [9] N. Guelfi, A. Mammar. "A formal framework to generate XPD specifications from UML activity diagrams". *2006 ACM symposium on Applied computing*.
- [10] M. Havey. *Essential Business Process Modeling*, O'Reilly Media, Inc., 2005. ISBN-10: 9780596008437.
- [11] J.A. Hernández, C. Acuña, V. de Castro, E. Marcos, M. López, N. Malpica. A WEB-PACS for Multi-center Clinical Trials. *IEEE Transactions on Information Technology in Biomedicine* 11,1 pp. 87-93, 2007.
- [12] M.J. Ibáñez, P. Álvarez, J. Ezpeleta. RDF Model Checking: A Technique to Verify Behavioral Properties in Semantically Annotated Business Processes. *Third IEEE International Conference on Semantic Computing (ICSC09)*, pp. 245-252.
- [13] F. Jouault, I. Kurtev. Transforming Models with ATL. *Satellite Events at the MoDELS 2005 Conference*, pp. 128-138.
- [14] R.K. Ko, S.S. Lee, E.W. Lee. Business process management (BPM) standards: A survey. *Business Process Management journal* 15 (5), 2009.
- [15] J. Miller, J. Mukerji. 'MDA Guide Version 1.0.1' Document number omg/2003-06-01, <http://www.omg.com/mda>, 2003.
- [16] OMG. *BPMN 1.1 - OMG Final Adopted Specification*, <http://www.omg.org/docs/formal/08-01-7.pdf>.
- [17] OMG. Estándares MDA, <http://www.omg.org>.
- [18] N. Palmer. Understanding the BPMN-XPDL-BPEL Value Chain. *Business Integration Journal*, No. November/December 2006, pp. 54-55.
- [19] S. Roser, B. Bauer. A categorization of collaborative business process modeling techniques. *Seventh IEEE International Conference on E-Commerce Technology Workshops*, IEEE Computer Society, Washington, DC, USA, 2005, pp. 43-54.
- [20] B. Selic. The pragmatics of Model-Driven development. *IEEE Software* Vol. 20, N° 5, pp. 19-25, 2003.
- [21] B. Selic. MDA Manifestations. *Upgrade, The European Journal for the Informatics Professional*, Vol. IX no.2, pp. 12-16, 2008. <http://www.cepis.org/upgrade/files/2008-II-selic.pdf>.
- [22] D. Steinberg, F. Budinsky, M. Paternostro, E. Merks. *Eclipse Modeling Framework* (2nd. Ed.). Addison-Wesley Professional, 2008. ISBN-10: 9780321331885.
- [23] L. Verner. BPM The Promise and the Challenge. *Queue of ACM*, Vol. 2 (4), pp. 82-91, 2004.
- [24] A. Watson. A Brief History of MDA. *Upgrade, The European Journal for the Informatics Professional*, Vol. IX no.º2, pp. 7-11, 2008. <http://www.cepis.org/upgrade/files/2008-II-watson.pdf>.
- [25] S. White. *Using BPMN to Model a BPEL Process*. IBM Corp., United States.
- [26] Grupo Kybele. *Metamodelos para la transformación de SOD-M a DENEb*. <http://www.kybele.etsii.urjc.es/TechnicalReport/SODM2DENEb/>.