



PETRÓLEOS DE VENEZUELA S.A.



MINISTERIO DEL PODER POPULAR
PARA CIENCIA Y TECNOLOGÍA



Fundación Instituto de
Ingeniería (FII-MCT)



Fondo Nacional de Ciencia,
Tecnología e Innovación
(FONACIT-MCT)



MINISTERIO DEL PODER POPULAR
PARA EDUCACIÓN SUPERIOR



Fundación Gran Mariscal
de Ayacucho (FGMA-MES)



GOBIERNO DE NUEVA ESPARTA



CÁMARA VENEZOLANA DE EMPRESAS DE
TECNOLOGÍAS DE LA INFORMACIÓN

Microsoft® MICROSOFT

X WORKSHOP IBEROAMERICANO DE
INGENIERÍA DE REQUISITOS Y AMBIENTES DE SOFTWARE

<http://kuainasi.ciens.ucv.ve/ideas07>

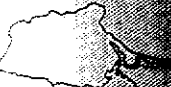
X Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes de Software



Primer Encuentro Venezolano sobre
Tecnologías de Información e
Ingeniería de Software

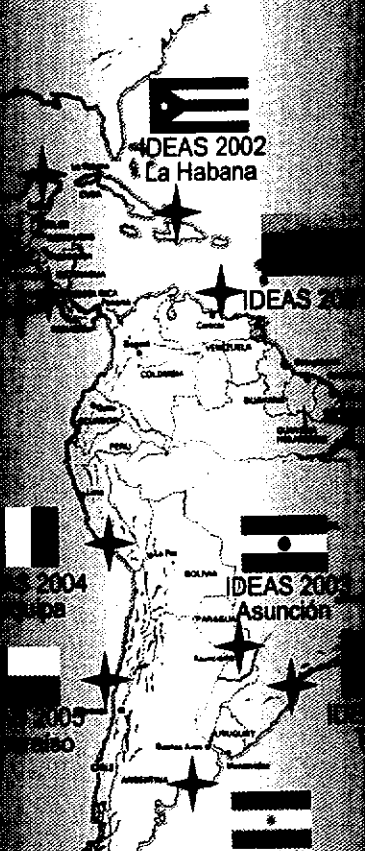
Del 7 al 11 de Mayo de 2007

Iberoamericano
de Requisitos
de Software



Estado

Del 7 al 11 de



MEMORIAS

X Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes de Software



Editores

Francisca Losavio
Guilherme Horta Travassos
Vicente Pelechano
Isabel Díaz
Alfredo Matteo

Isla de Margarita, Venezuela
Del 7 al 11 de Mayo de 2007

MEMORIAS

**X Workshop Iberoamericano de
Ingeniería de Requisitos
y Ambientes de Software**

Ficha Técnica

Memorias del X Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes de Software (IDEAS'07)
Editores: F. Losavio, G. H. Travassos, V. Pelechano, I. Díaz, A. Matteo
Mayo, 2007 – Caracas, Venezuela

Copyright © 2007 by IDEAS'07

All rights reserved

Prohibida la reproducción total o parcial de esta obra,
por cualquier medio, sin la autorización de sus editores

ISBN:978-980-325-323-3

If58120076201398

Agradecimiento

La publicación de estas memorias fue posible gracias al apoyo de las siguientes instituciones venezolanas:



Fondo Nacional de Ciencia Tecnología e Innovación (Fonacit)
del Ministerio del Poder Popular para Ciencia y Tecnología



Consejo de Desarrollo Científico y Humanístico (CDCH)
de la Universidad Central de Venezuela

Presidencia IDEAS'07

Francisca Losavio

Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación

Comité Directivo de IDEAS

Ernesto Pimentel

Universidad de Málaga
España

Jaelson Brelaz de Castro

Universidad Federal do Pernambuco
Brasil

Luca Cemuzzi

Universidad Católica Nuestra Señora de la Asunción
Paraguay

Luis Olsina

Universidad Nacional de La Pampa
Argentina

Miguel Katrib

Universidad de La Habana
Cuba

Oscar Pastor

Universidad Politécnica de Valencia
España

Silvia Gordillo

Universidad Nacional de La Plata
Argentina

Comité Organizador IDEAS'07

Alfredo Matteo

Co-Presidente
Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación

Isabel Díaz

Co-Presidente
Universidad Central de Venezuela
Facultad de Ciencias Económicas y Sociales
Escuela de Economía

Comité de Programa

Guilherme Horta Travassos
Co-Presidente
Universidade Federal do Rio de Janeiro
Brasil

Vicente Pelechano
Co-Presidente
Universidad Politécnica de Valencia
España

Miembros del Comité de Programa

- (1) **Alessandro Garcia**, Lancaster University – United Kingdom
- (2) **Alfredo Matteo**, Universidad Central de Venezuela – Venezuela
- (3) **Altigran Soares da Silva**, Universidade Federal do Amazonas – Brasil
- (4) **Álvaro Arenas**, CCLRC Rutherford Appleton Laboratory – Reino Unido
- (5) **Amador Durán**, Universidad de Sevilla – España
- (6) **Antonio Brogi**, Universidad de Pisa – Italia
- (7) **Antonio Ruiz**, Universidad de Sevilla – España
- (8) **Antonio Valleclillo**, Universidad de Málaga – España
- (9) **Carne Quer**, Universitat Politècnica de Catalunya – España
- (10) **Claudia Maria Lima Werner**, COPPE/Universidade Federal do Rio de Janeiro
- (11) **Emília Mendes**, University of Auckland – Nueva Zelanda
- (12) **Ernesto Pimentel**, Universidad de Málaga – España
- (13) **Ernest Teniente**, Universitat Politècnica de Catalunya – España
- (14) **Esperanza Marcos**, Universidad Rey Juan Carlos – España
- (15) **Fernanda Aletar**, Universidade de Pernambuco – Brasil
- (16) **Francisco Ruiz**, Universidad de Castilla-La Mancha – España
- (17) **Gustavo Rossi**, Universidad Nacional de La Plata – Argentina
- (18) **Itana Gimenes**, Universidade Estadual de Maringá – Brasil
- (19) **Jaelson Castro**, Universidad Federal de Pernambuco – Brasil
- (20) **João Araújo**, Universidade Nova de Lisboa – Portugal
- (21) **João Falcão e Cunha**, Universidade do Porto – Portugal
- (22) **Jonás Montilva**, Universidad de Los Andes – Venezuela
- (23) **José Carlos Maldonado**, Universidade de São Paulo – Brasil
- (24) **Juan Hernández**, Universidad de Extremadura – España
- (25) **Judith Barrios**, Universidad de Los Andes – Venezuela
- (26) **Júlio Leite**, Pontificia Universidade Católica do Rio de Janeiro – Brasil
- (27) **Kátia Marçal de Oliveira**, Universidade Católica de Brasília – Brasil
- (28) **Luca Cernuzzi**, Universidad Católica Nuestra Señora de la Asunción – Paraguay
- (29) **Luis Olsina**, Universidad Nacional de La Pampa – Argentina
- (30) **Maria Lancastre**, Universidade de Pernambuco – Brasil
- (31) **Manoel Mendonça**, Universidade Salvador UNIFACS – Brasil
- (32) **Marcello Visconti**, Universidad Técnica Federico Santa María – Chile
- (33) **Márcio Delamaro**, Centro Universitário Eurípides de Marília – Brasil
- (34) **Márcio de Oliveira Barro**, Universidade Federal do Estado do Rio de Janeiro – Brasil
- (35) **María Angélica Ovalles**, Universidad Simón Bolívar – Venezuela
- (36) **Mario Plattini**, Universidad de Castilla-La Mancha – España
- (37) **Miguel Katrib**, Universidad de La Habana – Cuba
- (38) **Nancy Zambrano**, Universidad Central de Venezuela – Venezuela
- (39) **Natalia Juristo**, Universidad Politécnica de Madrid – España
- (40) **Oscar Pastor**, Universidad Politécnica de Valencia – España
- (41) **Pere Botella**, Universitat Politècnica de Catalunya – España
- (42) **Regina Maria Maciel Braga**, Universidade Federal de Juiz de Fora – Brasil
- (43) **Ricardo de A. Falbo**, Universidade Federal do Espírito Santo – Brasil
- (44) **Sandra Fabbri**, Universidade Federal de São Carlos – Brasil
- (45) **Silvia Gordillo**, Universidad Nacional de La Plata – Argentina

Revisores Colaboradores

- (1) **Alberto Abelló**, Universidad Politècnica de Catalunya – España
- (2) **Alicia Díaz**, Universidad Nacional de La Pampa – Argentina
- (3) **Antonia Reina Quintero**, Universidad de Sevilla – España
- (4) **Arló Claudio Dias Neto**, COPPE/Universidade Federal do Rio de Janeiro – Brasil
- (5) **Heman Melgratti**, Universidad de Pisa – Italia
- (6) **Isabel Díaz**, Universidad Central de Venezuela – Venezuela
- (7) **Isi Castillo**, Universidad Central de Venezuela – Venezuela
- (8) **Ismael Navas**, Universidad de Pisa – Italia
- (9) **Javier Bazzocco**, Universidad Nacional de La Pampa – Argentina
- (10) **Joaquín Peña Siles**, Universidad de Sevilla – España
- (11) **Jobson Luiz Massolar da Silva**, COPPE/Universidade Federal do Rio de Janeiro – Brasil
- (12) **José María Conejero**, Universidad de Extremadura – España
- (13) **Juan Manuel Murillo**, Universidad de Extremadura – España
- (14) **María Istela Cagnin**, Centro Universitário Eurípides de Marília – Brasil
- (15) **María José Escalona**, Universidad de Sevilla – España
- (16) **Pedro J. Clemente**, Universidad de Extremadura – España
- (17) **Pedro Valderas**, Universidad Politècnica de Valencia – España
- (18) **Razvan Popescu**, Universidad de Pisa – Italia
- (19) **Rodrigo de Oliveira Spinola**, COPPE/Universidade Federal do Rio de Janeiro – Brasil
- (20) **Thalzel Fuentes**, Universidad de Pisa – Italia
- (21) **Valter Vieira de Camargo**, Centro Universitário Eurípides de Marília – Brasil
- (22) **Victoria Torres**, Universidad Politècnica de Valencia – España

PRÓLOGO

Este volumen contiene los trabajos aceptados y presentados en el *X Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes de Software: IDEAS'07* celebrado en la Isla de Margarita, Venezuela, del 7 al 11 de mayo de 2007.

Muy brevemente mencionaremos los números y los pasos que dimos durante el proceso de evaluación. Inicialmente se enviaron 104 resúmenes que finalmente se concretaron en el envío de 82 artículos. Cada artículo se asignó a 3 revisores. En la difícil decisión de aceptar o rechazar los trabajos se discutieron las discrepancias con respecto a un mismo artículo para alcanzar, donde fuera posible, un consenso. Además de esto adoptamos el criterio de aceptar como artículos todos y sólo aquellos que obtuvieron un promedio igual o superior a 4 sobre 6. Así, el resultado final ha sido que de los 82 artículos enviados, 29 han sido aceptados para su presentación.

La cantidad de trabajos enviados se ha mantenido, lo que constituye un síntoma claro del interés que existe por IDEAS. Esto nos hace pensar que, pasados 10 años, IDEAS ha lograrse consolidarse como un foro de primer nivel en Ingeniería del Software para Iberoamérica. Al mismo tiempo, el porcentaje de aceptación indica un nivel de exigencia interesante para la realidad Iberoamericana que puede ayudar a situar a IDEAS a una altura que permita obtener un mayor reconocimiento de las calificaciones de los autores en sus respectivos países.

Evidentemente, todo esto no hubiera sido posible sin la valiosa colaboración de distintos actores a quienes se dirigen nuestros más sinceros agradecimientos. Entre ellos cabe mencionar a los *autores*, por su esfuerzo en investigación, los *revisores*, *miembros del Comité de Programa* y sus *colaboradores*, quienes han realizado un esfuerzo particularmente intenso por la cantidad de trabajos recibidos y el proceso adoptado, los *miembros del Comité Organizador*, los *ponentes* de los tutoriales y conferencias invitadas, el *Comité Directivo de IDEAS* y todas las demás *personas e instituciones patrocinantes* que de distintas formas han contribuido para que este evento se constituya en una ocasión importante y beneficiosa para la comunidad científica en nuestra región.

Esperamos disfruten de IDEAS'07 así como de la cálida acogida y exuberante belleza de la Isla de Margarita.

Francisca Losavio
Presidenta IDEAS'07

Guilherme Horta Travassos
Co-Presidente Comité de Programa
Vicente Pelechano
Co-Presidente Comité de Programa

ÍNDICE DE CONTENIDOS

SESIÓN 1: LENGUAJES, MÉTODOS, PROCESOS Y HERRAMIENTAS (Parte 1)	1
Extracting the Best Features of Two Tropos Approaches for the Efficient Design of MAS	3
María Jocelia Silva, Paulo Roberto Maciel, Rosa C. Pinto, Fernanda Alencar, Patrícia Tedesco, Jaelson Castro Universidade Federal de Pernambuco, Brasil	
Soporte Automatizado a la Ingeniería de Requisitos de Seguridad	17
Daniel Mellado ¹ , Moisés Rodríguez ² , Eduardo Fernández-Medina ² , Mario Piattini ² ¹ Ministerio de Trabajo y Asuntos Sociales, España ² Universidad de Castilla-La Mancha, España	
SESIÓN 2: MODELADO ORGANIZACIONAL	31
Construcción de Modelos de Requisitos a partir de Modelos de Procesos y de Metas	33
José De la Vara, David Anes, Juan Sánchez Universidad Politécnica de Valencia, España	
Modelagem de Requisitos Organizacionais, Não-Funcionais e Funcionais em Software Legado com Ênfase na Técnica I*	47
Victor F. Araya S. ^{1,2} , André A. Vicente ² , Fabio G. Köerich ² , Jaelson Brelaz de Castro ³ ¹ Universidad de Talca, Chile ² Universidade Estadual do Oeste do Paraná, Brasil ³ Universidade Federal de Pernambuco, Brasil	
SESIÓN 3: MEDICIÓN Y EXPERIMENTACIÓN	61
Validando la Usabilidad y Mantenibilidad de los Modelos de Procesos de Negocio: un Experimento y su Réplica	63
Elvira Rolón ¹ , Félix García ² , Francisco Ruiz ² , Mario Piattini ² ¹ Universidad Autónoma de Tamaulipas, México ² Universidad de Castilla-La Mancha, España	
Soporte de Información Contextual en un Marco de Medición y Evaluación	77
Hernán Molina, Luis Olsina Universidad Nacional de La Pampa, Argentina	
Propuesta de Marco para la Selección de Técnicas de Educación de Requisitos ...	91
Dante Carrizo ¹ , Oscar Dieste ² ¹ Universidad Complutense de Madrid, España ² Universidad Politécnica de Madrid, España	

SESIÓN 4: INGENIERÍA DE REQUISITOS	105
Discovering Group Communication Requirements	107
Igor Miranda ¹ , Renata Araujo ² , Marcos Borges ¹ ¹ Universidade Federal do Rio de Janeiro, Brasil ² Universidade Federal do Estado do Rio de Janeiro, Brasil	
Comparaç�o do Impacto do Uso de um Processo de Engenharia de Requisitos entre Grupos de Desenvolvimento de Software - Um Estudo de Caso	121
Elias Canhadas Genvigir ¹⁻² , Nilson Sant'Anna ¹ ¹ Universidade Tecnol�gica Federal do Paran�, Brasil ² Instituto Nacional de Pesquisas Espaciais, Brasil	
Extensi�n al Modelo de Separaci�n Multi-Dimensional de Concerns en Ingenier�a de Requisitos	135
Carlos A. Ospina, Carlos A. Parra, Luis F. Londo�o, Raquel Anaya Universidad EAFIT, Colombia	
SESI�N 5: PROCESOS DEL NEGOCIO	149
Una Propuesta Basada en Modelos para la Construcci�n de Sistemas Ubicuos que den Soporte a Procesos de Negocio	151
Pau Giner, Victoria Torres Universidad Polit�cnica de Valencia, Espa�a	
Experiencia en Transformaci�n de Modelos de Procesos de Negocios desde BPMN a XPDL	165
Beatriz Mora, Francisco Ruiz, F�lix Garc�a, Mario Piattini Universidad de Castilla-La Mancha, Espa�a	
Verification of Models in a MDA Approach for Collaborative Business Processes ...	179
Pablo Villarreal ¹ , Jorge Roa ¹ , Enrique Salomone ¹⁻² , Omar Chlotti ¹⁻² ¹ Universidad Tecnol�gica Nacional, Argentina ² INGAR-CONICET, Argentina	
SESI�N 6: LENGUAJES, M�TODOS, PROCESOS Y HERRAMIENTAS (Parte 2)	193
Towards a Standardized Description and a Systematic Use of Social Patterns	195
Carla Silva ¹ , Jo�o Ara�jo ² , Ana Moreira ² , Jaelson Castro ¹ ¹ Universidade Federal de Pernambuco, Brasil ² Universidade Nova de Lisboa, Portugal	
Aplicaci�n de QVT al Desarrollo de Almacenes de Datos Seguros: Un caso de Estudio	209
Emilio Soler ¹ , Juan Trujillo ² , Eduardo Fern�ndez-Medina ³ , Mario Piattini ³ ¹ Universidad de Matanzas, Cuba ² Universidad de Alicante, Espa�a ³ Universidad de Castilla-La Mancha, Espa�a	

SESI�N 7: MDA Y TRANSFORMACI�N DE MODELOS	223
Marco de Referencia para la Evaluaci�n de Herramientas basadas en MDA	225
Juan Bernardo Quintero, Raquel Anaya de Paez Universidad EAFIT, Colombia	
Composici�n de Transformaciones de Modelos en MDD basada en el �lgebra Relacional	239
Roxana Giandini ¹ , Gabriela P�rez ¹ , Claudia Pons ¹⁻² ¹ Universidad Nacional de La Plata, Argentina ² Universidad Abierta Interamericana, Argentina	
OOWS Suite: Un Entorno de Desarrollo para Aplicaciones Web basado en MDA ...	253
Francisco Valverde, Pedro Valderas, Joan Fons Universidad Polit�cnica de Valencia, Espa�a	
SESI�N 8: LENGUAJES, M�TODOS, PROCESOS Y HERRAMIENTAS (Parte 3)	267
Utilizando a Tecnica I* para Modelar a Concep�o de Vigotski Visando Auxiliar o Processo de Desenvolvimento de Software Educacional para Pessoas com Defici�ncia Visual	269
Victor F. Araya S. ¹⁻² , Dorivaldo Rodrigues da Silva ² , Andr� Abe Vicente ² , Jaelson de Castro ³ ¹ Universidad de Talca, Chile ² Universidade Estadual do Oeste do Paran�, Brasil ³ Universidad Federal de Pernambuco, Brasil	
Intercambio de Modelos UML y OO-Method	283
Beatriz Mar�n, Giovanni Giachetti, Oscar Pastor Universidad Polit�cnica de Valencia, Espa�a	
Apoio Automatizado � Ger�ncia de Riscos Cooperativa	297
Victorio Carvalho, Alexandre Coelho, Ricardo Falbo Universidade Federal do Esp�rito Santo, Brasil	
SESI�N 9: LENGUAJES, M�TODOS, PROCESOS Y HERRAMIENTAS (Parte 4)	311
Um Modelo Integrado de Requisitos com Casos de Uso	313
Michel Fortuna ¹⁻² , Claudia Werner ¹ , Marcos Borges ¹ ¹ Universidade Federal do Rio de Janeiro, Brasil ² Universidade Federal do Julz de Fora, Brasil	
Planejamento Integrado das Atividades de Codifica�o e Testes em Orienta�o a Objetos no N�vel de Granularidade dos M�todos.....	327
Tatiane Lopes, Clovis Fernandes Instituto Tecnol�gico de Aeron�utica, Brasil	
Desenvolvimento de Interface com Usu�rio Dirigida por Modelos com Gera�o Autom�tica de C�digo	341
Lucas Issa ¹ , Clarindo P�dua ¹ , Rodolfo Resende ¹ , Stenio Viveiros ¹ , Pedro Neto ² ¹ Universidade Federal de Minas Gerais, Brasil ² Universidade Federal do Piaul, Brasil	

SESIÓN 10: REQUISITOS Y COLABORACIÓN	355
OO-Sketch: Una Herramienta para la Captura de Requisitos de Interacción	357
José Ignacio Panach, Sergio España, Inés Pederiva, Oscar Pastor Universidad Politécnica de Valencia, España	
Colaboração e Negociação na Elicitação de Requisitos	371
Danilo Freitas, Marcos Borges, Renata Araujo Universidade Federal do Rio de Janeiro, Brasil	
SESIÓN 11: SERVICIOS WEB Y COMPONENTES	385
Automated Generation of BPEL Adapters	387
Antonio Brogi, Razvan Popescu Universidad de Pisa, Italia	
Un Enfoque Dirigido por Modelos para el Desarrollo de Servicios Web Semánticos	401
César J. Acuña, Esperanza Marcos, Mariano Minoli Universidad Rey Juan Carlos, España	
Un Perfil UML para la definición de Componentes Inteligentes	415
José Luis Pastrana ¹ , Ernesto Pimentel ¹ , Miguel Katrib ² ¹ Universidad de Málaga, España ² Universidad de La Habana, Cuba	
EVETIS'07: PRIMER ENCUENTRO VENEZOLANO SOBRE TECNOLOGÍAS DE LA INFORMACIÓN E INGENIERÍA DE SOFTWARE	427

Sesión 1

Lenguajes, Métodos, Procesos y Herramientas (Parte 1)

- [11] Oat Systems and MIT Auto-ID Center. The Savant. Technical Report MIT-AUTOID-TM-003, MIT Auto-ID Center, May 2002.
- [12] T. Kindberg et al. People, Places, Things: Web Presence for the Real World. In WMCSA 2000, Monterey, USA, Dec. 2000.
- [13] G. Abowd, E.Mynatt. Charting Past, Present and Future Research in Ubiquitous Computing. ACM Transactions on Computer-Human Interaction, vol. 1, no. 7, March 2000, pp. 29-58.
- [14] M. Satyanarayanan. Pervasive Computing: Vision and Challenges. IEEE Personal Communications, August 2001, pp. 10-17.
- [15] K. Lyytinen, Y. Yoo. Issues and Challenges in Ubiquitous Computing. Communications of the ACM, vol.45, no12, December 2002, pp. 63-65.
- [16] M. Langheinrich, V. Coroama, J. Bohn, M. Rohs. As we may live - Real world implications of ubiquitous computing. Technical Report, Institute of Information Systems, Swiss Federal Institute of Technology, Zurich, Switzerland, 2002.
- [17] A. Stone. The Dark Side of Pervasive Computing. IEEE Pervasive Computing, January-March 2003, pp. 4-8.
- [18] Henricksen, K., Indulska, J., and Rakotonirainy, A. Modeling context information in pervasive computing systems. In LNCS 2414: Proceedings of 1st International Conference on Pervasive Computing (Zurich, Switzerland, 2002), F. Mattern and M. Naghshineh, Eds., Lecture Notes in Computer Science (LNCS), Springer, pp. 167-180.
- [19] D. Thevenin and J. Coutaz, "Plasticity of User Interfaces: Framework and Research Agenda", Proc. of INTERACT'99 (Edinburgh, August 1999), IOS Press, 1999.
- [20] Javier Muñoz and Vicente Pelechano. Building a Software Factory for Pervasive Systems Development. In CAISE 2005, Porto, Portugal, June 13-17, volume 3520 of LNCS, pages 329-343, May 2005.
- [21] Paterno, Mancinii and Meniconi, 1997. "ConcurTaskTress: a Diagrammatic Notation for Specifying Task Models", In Proceedings of INTERACT 97, Chapman & Hall, 362-366
- [22] Torres, V, Pelechano, V., Giner, P.: Generación de Aplicaciones Web basadas en Procesos de Negocio mediante Transformación de Modelos. In: Riquelme, J.C., Botella, P. (eds.): Ingeniería del Software y Bases de Datos. JISBD 2006. 443-452
- [23] S. Domnitcheva: Location Modeling: State of the Art and Challenges, Location Modeling for Ubiquitous Computing Workshop at Ubicomp 2001, Atlanta.
- [24] Ouyang, C., van der Aalst, W.M.P., Dumas, M., ter Hofstede, A.H.M.: Translating BPMN to BPEL. BPM Center Report BPM-06-02, BPMcenter.org (2006)
- [25] S. A. White. Business Process Modeling Notation (BPMN) Version 1.0. Business Process Management Initiative, BPMI.org, May 2004.
- [26] OMG. MOF QVT Final Adopted Specification. ptc/05-11-01, Nov 2005.
- [27] Pastor, O., Gomez, J., Infran, E., Pelechano, V. The OO-Method Approach for Information Systems Modelling: From Object-Oriented Conceptual Modeling to Automated Programming. Information Systems 26, pp 507-534 (2001).
- [28] Christian Emig, Christof Momm, Jochen Weisser, Sebastian Abeck: Programming in the Large based on the Business Process Modeling Notation, Jahrestagung der Gesellschaft für Informatik (GI), Bonn, September 2005.
- [29] R. Heckel and M. Lohmann. Towards Model-Driven Testing. Electronic Notes in Theoretical Computer Science 82(6), 2003.
- [30] Spinola, Rodrigo, Massolar, Jonson, Travassos, G. H. Towards a Conceptual Framework to Classify Ubiquitous Software Projects. In: SEKE 2006 - International Conference on Software Engineering and Knowledge Engineering, 2006, San Francisco. Proceedings of the SEKE 2006, 2006.

Experiencia en transformación de modelos de procesos de negocios desde BPMN a XPDL.

Beatriz Mora, Francisco Ruiz, Félix García, Mario Piattini

Universidad de Castilla-La Mancha, Escuela Superior de Informática, España
{Beatriz.Mora | Francisco.RuizG | Felix.Garcia | Mario.Piattini}@uclm.es

Resumen. En este trabajo se presenta una primera experiencia de aplicación de MDA a la transformación de modelos de procesos de negocio, desde BPMN a XPDL. Se trata de transformaciones entre niveles M1 y M1 (en terminología MDA), es decir, entre modelos y modelos; pero con la particularidad de que los modelos BPMN son de naturaleza conceptual mientras que los modelos XPDL son de naturaleza lógica. Esto ha originado problemas a la hora de establecer las equivalencias entre elementos de BPMN y elementos de XPDL y, como consecuencia, se han sufrido complicaciones importantes para definir las transformaciones. Para llevar a cabo la experiencia se ha trabajado con el entorno de desarrollo ADT, funcionando sobre la plataforma ECLIPSE. Con esta herramienta se han utilizado diversos estándares soportados por MDA (MOF, QVT y OCL 2.0) y el lenguaje KM3 (Kernel MetaMetaModel) para describir los metamodelos. En este trabajo se explica la manera en que se ha llevado a cabo la experiencia y se comentan los principales problemas encontrados; entre los cuales caben resaltar dos: ausencia de una correcta impedancia entre ambos metamodelos, y algunas limitaciones de la herramienta. También se presentan las principales lecciones aprendidas.

Palabras Claves: MDA, Transformación de modelos, Procesos de negocio, BPMN, XPDL, ADT.

1 Introducción

La Arquitectura MDA [13], las transformaciones de modelos y los procesos de negocios son conceptos que están cobrando cada vez más importancia en la ingeniería del software. Estos conceptos son el principal motor de esta experiencia, consistente en la aplicación de la arquitectura MDA en una transformación de modelos de procesos de negocio, representados en BPMN [10], a XPDL [19].

Las transformaciones se han realizado entre niveles M1 a M1, según la terminología MOF [12], pero con la peculiaridad que los dos metamodelos del nivel M2 subyacentes son de distinta naturaleza: BPMN es de naturaleza conceptual mientras que XPDL es de naturaleza lógica. Es una situación con un cierto parecido a los modelos conceptuales entidad/relación frente a los modelos lógicos relacionales, en el campo de las bases de datos.

Para llevar a cabo las transformaciones se ha utilizado el lenguaje ATL (Atlas Transformation Language) [7] sobre el entorno de desarrollo ADT (ATL Development Tools) [2], ambos desarrollados por la Universidad de Nantes (Francia).

Son tres los objetivos que han impulsado la realización de la experiencia. En primer lugar, probar la tecnología. En este sentido, interesaba utilizar un entorno de desarrollo intuitivo, amigable, que ayudase a realizar transformaciones aplicando los principios de MDA, y que estuviera basado en la tecnología Eclipse. ADT cumple todos estos requisitos. El segundo objetivo era trabajar con modelos y metamodelos de procesos de negocios. El tercer y último objetivo buscado era enfrentar el reto de una transformación compleja, como es el caso de BPMN vs XPDL.

El resto de este trabajo está organizado en los siguientes apartados: una continuación de la introducción, donde se comenta un aspecto que puede ser clarificador para el lector: estándares empleados para el modelado de procesos de negocio. En segundo lugar se explican las características del entorno tecnológico empleado durante la experiencia. En el tercer apartado se presentan los únicos trabajos relacionados conocidos. A continuación, en el tercer apartado se desarrolla la experiencia, detallando paso a paso lo que se ha realizado, también comentando los problemas encontrados. Y por último, un apartado de conclusiones.

1.1 Modelado de Procesos de Negocios

Mientras que el objetivo de un proceso software es obtener un producto software [1], el objetivo de un proceso de negocio es obtener resultados beneficiosos (generalmente un producto o servicio) para los clientes u otros interesados (stakeholders) [15]. Un modelo de proceso de negocio describe cómo funciona el negocio [6], es decir, describe las actividades involucradas en el negocio y la manera en que se relacionan unas con otras e interactúan con los recursos necesarios para lograr la meta del proceso.

El modelado de procesos de negocios se utiliza para capturar, documentar o rediseñar procesos de negocio. Para llevarlo a cabo se pueden emplear diversos lenguajes, con diferente naturaleza, características y objetivos.

BPMN (Business Process Modeling Notation) es una notación gráfica para el modelado conceptual de procesos de negocio. Proporciona la capacidad de entender y definir procesos de negocio, tanto internos como externos, a través de un diagrama de procesos de negocio. Es un estándar de facto que provee una representación gráfica (mediante diagramas) para expresar procesos de una empresa. Se diseñó con el objetivo de facilitar la comprensión por parte de todos los implicados (expertos TIC, analistas de negocio, directivos, etc.) que participan en el proceso.

XPDL (XML Process Definition Language) es un lenguaje para la definición de flujos de trabajo de procesos. Fue creado por la WfMC en el año 2001 y el 3 de octubre del 2005 se liberó la versión 2.0. XPDL forma parte de la interfaz uno (definición de procesos de negocio). Utiliza un formato de archivo basado en XML [18], que se puede usar para intercambiar modelos de procesos de negocio entre herramientas diferentes. Es un formato de archivo que representa el "dibujo" del grafo que representa la definición del proceso. Por esta razón, incluye elementos tales como el tamaño y las coordenadas X e Y de un nodo o las líneas que indican el camino a seguir. Los nodos y las líneas tienen atributos que pueden especificar información

sobre la ejecución del proceso, tales como roles, descripción de actividades, temporizadores, llamadas a servicios (web o de otro tipo), etc. La versión 2.0 incluye extensiones para poder representar todos los elementos de BPMN. Éste último hecho demuestra la importancia real que han cobrado ambos lenguajes. Por ello, puede resultar interesante y útil establecer mecanismos de transformación automatizados entre ambos, razón por la cuál cobra sentido la experiencia aquí presentada.

2 Entorno Tecnológico Empleado

Para este experimento interesaba utilizar un entorno de desarrollo que permita transformaciones siguiendo la filosofía de MDA y que esté basado en la tecnología Eclipse (www.eclipse.org). La arquitectura de plugins de Eclipse permite, además de integrar diversos lenguajes sobre un mismo IDE, introducir otras aplicaciones accesorias que pueden resultar útiles durante el proceso de desarrollo; tales como diseñadores en UML, editores visuales de interfaces, ayuda en línea para librerías, etc. Por todas estas razones, en esta experiencia se optó por emplear un entorno tecnológico que esté soportado por Eclipse. El entorno utilizado en este experimento, y que satisface estos requisitos, es ADT (ATL Development Tools), el cual proporciona herramientas para trabajar con el lenguaje de transformaciones ATL. Los elementos que se pueden encontrar en ADT son los típicos que pueden aparecer en un entorno de desarrollo: asistentes (wizards) para crear proyectos ATL que permiten ejecutar transformaciones, editor de textos, vistas de propiedades, iconos para especificar todos los elementos específicos de ATL, "builder" asociado al tipo de proyecto, distintas perspectivas, informes de errores de compilación, depurador, etc.

ATL es un lenguaje de transformación de modelos basado en los estándares del OMG, MOF, QVT [11] y OCL 2.0 [14]. Es un lenguaje híbrido, ya que se trabaja con construcciones declarativas e imperativas. Las construcciones declarativas son la opción preferida para escribir las transformaciones, puesto que son claras y precisas. Permiten expresar correspondencias, entre los elementos del modelo fuente y del modelo destino, a partir de una serie de composiciones de reglas. Adicionalmente, las construcciones imperativas proporcionan constructores para facilitar la especificación de correspondencias que de forma declarativa serían mucho más complejas de implementar.

El principio que hay que tener en cuenta a la hora de abordar una transformación en ATL, es que una transformación también es un modelo y, por tanto, cumple todas las propiedades de un modelo. Aunque con ATL es posible realizar diversos tipos de transformaciones (M1 a M2, M2 a M1, M3 a M1, M3 a M2, etc.), los objetivos de nuestro experimento han supuesto que sólo se hallan experimentado transformaciones M1 a M1 (de modelo BPMN a modelo XPDL). De todas formas, la filosofía MDA obliga a tener en cuenta y a utilizar los niveles superiores de modelado, ya que cada modelo se define conforme a un metamodelo, y éste a su vez se define respecto de un metametamodelo universal. MOF, ya comentado, y Ecore [4], que se introdujo con "Eclipse Modelling Framework" (EMF) [5], son los metametamodelos más significativos.

La transformación de modelos proporciona las facilidades para generar un modelo Mb (definido a partir de su metamodelo MMB) a partir de otro modelo Ma (definido

acorde a su metamodelo MMA). El modelo de la transformación es un tercer modelo que define las reglas necesarias para realizar la transformación. Por supuesto, también se define a partir de su metamodelo que en el caso que nos ocupa es el metamodelo de ATL. Todos estos metamodelos que intervienen en la transformación (el de entrada, el de salida y el de transformación) están a su vez definidos en base a un único metamodelo, tales como MOF o Ecore.

En nuestro experimento, el modelo de transformación es un archivo ATL (MMA2MMb.atl), que está basado en el metamodelo ATL (ver Fig. 1). Este archivo contiene las reglas necesarias para la transformación de los modelos, incluyendo las correspondencias entre los metamodelos de entrada y de salida.

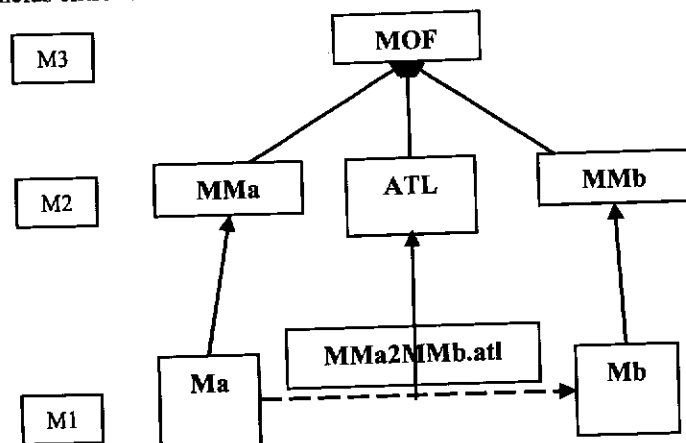


Fig. 1. Esquema de una transformación de modelos con ADT.

Para facilitar la codificación de los metamodelos se emplea KM3 [8]; un lenguaje creado por el mismo grupo de investigación de ATL. Este lenguaje proporciona una sintaxis textual similar a la notación de Java. La sintaxis abstracta de KM3 está basada en Ecore y eMOF 2.0. El formato de archivo .km3 se puede serializar en los formatos Ecore y MOF 1.4. Una vez escritos los metamodelos en KM3, ADT permite pasarlos a Ecore, traducción que es obligatoria porque las transformaciones ATL requieren que los metamodelos y modelos estén escritos en Ecore.

3 Propuestas relacionadas

Debido a su novedad, casi no existe información ni estudios relacionados con la transformación entre los lenguajes BPMN y XPDL. Se sabe que la última versión de XPDL, la 2.0, define las extensiones necesarias para poder representar BPMN pero, aún así, no existe ningún documento oficial que represente las transformaciones y correspondencias entre ambos lenguajes. El único trabajo conocido al respecto es el llevado a cabo por alguna empresa privada [17], en el cual se presentan algunas correspondencias entre los dos lenguajes que han servido de apoyo para la experiencia. La mayor limitación de esta propuesta es que no entra en detalle en el

análisis de los elementos del primer lenguaje que corresponden con otro u otros del segundo lenguaje. Por esta razón, dicho trabajo ha tenido que ser realizado durante la realización de nuestro experimento y, de hecho, ha supuesto la mayor parte del esfuerzo realizado.

4 Transformación de BPMN a XPDL

En ATL, y por tanto en nuestro experimento, a la hora de llevar a cabo una transformación, es necesario realizar las siguientes etapas:

- 1) **Creación del proyecto ATL:** Se emplea "ATL Project", un módulo del entorno que ayuda a la creación de los proyectos ATL mediante un asistente.
- 2) **Creación de los metamodelos KM3:** en esta experiencia, los metamodelos son los correspondientes a BPMN y XPDL.
- 3) **Creación del archivo ATL de transformación:** que almacena las reglas de transformación entre los modelos.
- 4) **Creación del modelo de entrada:** para probar la experiencia, es necesaria la elaboración de un modelo de entrada, en este caso, el modelo de entrada corresponde a un modelo de BPMN utilizado de ejemplo.
- 5) **Ejecución de la transformación para obtener el modelo de salida:** se emplea el módulo "ATL builder" para obtener, en nuestro caso, el modelo XPDL correspondiente.

En los sub-apartados siguientes se detallan algunos de los aspectos más significativos durante la realización de estas etapas.

4.1 Metamodelos

Tal como se ha explicado, para la transformación entre BPMN y XPDL se necesitan sendos metamodelos escritos en KM3. En un principio, los metamodelos que sirvieron de base se encontraron en la página oficial de Eclipse (xpd.km3: última actualización 21/02/2006 y bpmn.km3: última actualización 20/09/2006). Al principio, pareció que encontrar estos metamodelos ya creados pareció solucionar la tarea más costosa de codificación, pero, analizando dichos metamodelos, se comprobó que el de BPMN estaba incompleto porque se refería a la versión 1.0 y sólo incluía algunos elementos (Activity, BpmnDiagram, Message, Pool, Properties, PropertiesType, Subproceso), dejando a muchos otros sin precisar. En cambio, el metamodelo de XPDL, sí coincidía casi por completo con la especificación de su lenguaje.

Para solventar el problema del metamodelo BPMN se pensó en escribir el metamodelo a partir de sus especificación correspondiente [10], pero resultaba un trabajo demasiado laborioso para los objetivos del experimento. Por ello, finalmente se optó por crear dos metamodelos parciales de BPMN y XPDL, que describieran sólo los elementos más representativos. Para alcanzar los objetivos de la experiencia era suficiente con dichos metamodelos parciales ya que, lo que interesaba, era que existiese una transformación entre lenguajes de modelado de procesos de negocio

conceptuales y lógicos. Además, en este punto cabe resaltar que se descubrió que no existen correspondencias plenas entre ambos metamodelos, es decir, muchos elementos de uno de los metamodelos no tienen un elemento equivalente (o similar) en el otro metamodelo. Por todo ello, los elementos que se incluyeron en los metamodelos parciales fueron sólo aquellos que tienen correspondencia plena, es decir, con equivalentes en el otro metamodelo y, en consecuencia, son transformables de BPMN a XPDL.

De esta forma, el metamodelo parcial de XPDL, utilizado como entrada en la transformación, incluye sólo los elementos más significativos que tienen equivalente en el lenguaje BPMN. Por otro lado, al estar incompleto el metamodelo de BPMN disponible en la web de Eclipse, se optó por utilizar como punto de partida, para el metamodelo de salida, el metamodelo subyacente de la herramienta de modelado de procesos de negocio "eBPMN Designer" [16]. La principal ventaja de ésta herramienta es que ya trabaja con la nueva versión oficial 2.0. La desventaja que hubo que superar es que hubo que realizar ingeniería inversa para disponer del metamodelo BPMN a partir del código de la herramienta, ya que no figura de forma explícita como un recurso, documento o archivo específico.

En la siguiente figura (Fig. 2) se muestra parte del código de ambos metamodelos, escrito en KM3. La imagen de la izquierda corresponde al metamodelo de XPDL y la imagen de la derecha al metamodelo de BPMN.

```

package XPDL{
  ..***** EVENT *****..
  abstract class Element{
    attribute id : String;
    attribute name [0-1] : String;
  }
  class Activity extends Element {
    attribute isStartActivity[0-1] : Boolean;
    attribute status[0-1] : StatusKind;
    attribute startMode : ModeKind;
    attribute finishMode : ModeKind;
    attribute startQuantity[0-1] : Integer;
    attribute isATransaction[0-1] : Boolean;
  }
}

package BPMN{
  ..***** EVENT *****..
  class ProcessObject{
    attribute categories : String;
    attribute documentation : String;
    --reference
  }
  class FlowObject extends ProcessObject{
    attribute name : String;
  }
  class Event extends FlowObject {
    attribute eventType : String;
    attribute implementation : String;
    reference pool : Pool oppositeOf events;
  }
}
    
```

Fig. 2. Parte del código del metamodelo de XPDL (izq.) y BPMN (der.) en KM3.

4.2 Correspondencias

Una vez que se tienen los metamodelos, el siguiente paso es la creación del fichero de transformación ATL. Dentro de éste fichero se definieron todas las reglas necesarias para la transformación entre modelos BPMN y XPDL, basados en los previamente referidos metamodelos parciales. Lo primero que hubo que abordar era la lista de correspondencias a implementar. Para ello nos basamos en la propuesta ya comentada en el apartado 2. El resultado fue la lista de correspondencias mostrada en la Tabla 1. Por las razones ya comentadas, los elementos considerados en nuestra experiencia fueron los mostrados en negrita en la citada tabla: Business Process

Diagram, Event, Task, SequenceFlow y Pool (este elemento no se encontraba en [17] y fue añadido).

Tabla 1. Correspondencias identificadas entre BPMN y XPDL

BPMN	XPDL
Business Process Diagram	Package
Process	WorkflowProcess
Subprocess	SubFlow
Activity	Activity
Gateway	Route (Child of Activity)
Gate	Transition (see below)
Event	Event
InputPropertyMaps	DataMappings
OutputPropertyMaps	
Task	Task (Child of Activity → Implementation)
Artifact	Artifact (Child of Activity)
Graphical Connecting Objects, SequenceFlow , MessageFlow, Association	Transition
Property	Field
Pool	Pool

En la figura 3 se muestra la correspondencia entre elementos Event de BPMN y XPDL.

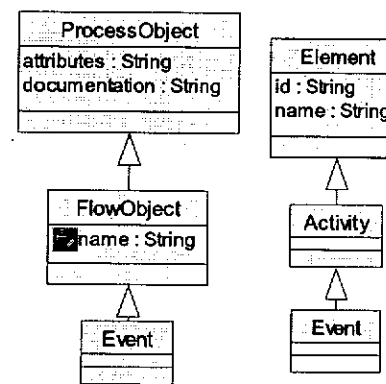


Fig. 3. Correspondencia Event (BPMN) - Event (XPDL).

En la siguiente imagen (Fig. 4), se hace lo mismo pero para los elementos de tipo Task (tareas). También, para mejorar la visibilidad, se han omitido las subclases de Task: ManualTask, ServiceTask, UserTask, ScriptTask, ReferenceTask, ReceiveTask y SendTask.

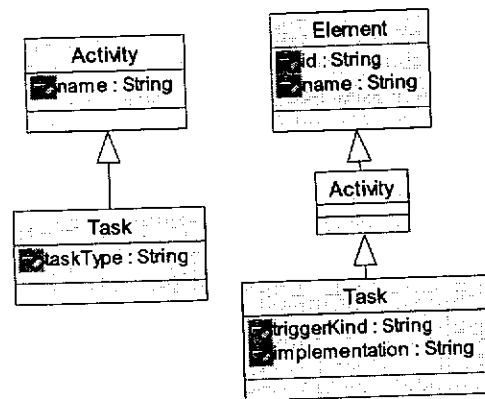


Fig. 4. Correspondencia Task (BPMN) - Task (XPDL).

La figura 5 representa la correspondencia entre elementos SequenceFlow (en BPMN) y elementos Transition (en XPD L).

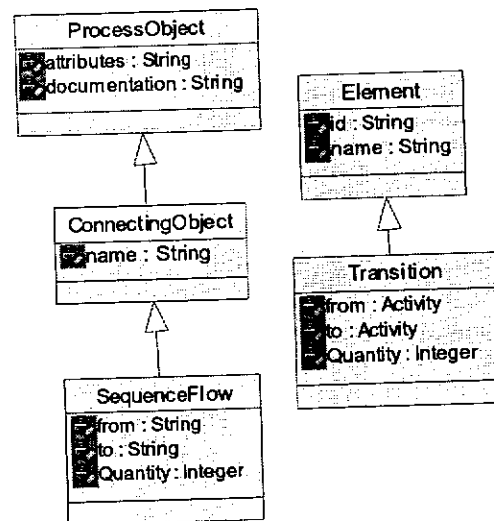


Fig. 5. Correspondencia SequenceFlow (BPMN) - Transition (XPDL).

Por último, en la figura 6, se muestra parte del código que implementa todas estas correspondencias mediante reglas de transformación. Como se observa en dicha figura, el archivo de transformación está compuesto de reglas (rules), de forma que existe al menos una regla por cada elemento. En cada una de estas reglas se indica el elemento origen de BPMN y su correspondencia con el elemento destino de XPD L.

```

***** TRANSITION - SEQUENCE FLOW *****
rule SequenceFlow2Transition{
    from
        b : BPMN!SequenceFlow
    to
        out : XPDL!Transition(
            quantity <- b.quantity,
            from_element<- b.from_element,
            to_element <- b.to_element
        )
}

rule Pool {
    from
        b : BPMN!Pool
    to
        out : XPDL!Pool(
            name <- b.name
        )
}

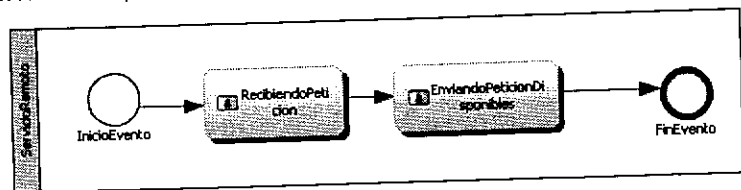
```

Fig. 6. Ejemplo del código del archivo de transformación bpmn2xpdl.atl.

4.3. Ejemplo

Tal como se explicó anteriormente, para ejecutar la transformación es necesario que los metamodelos se traduzcan al formato Ecore, operación que se realiza automáticamente por el entorno ADT. Por tanto, parece que el lenguaje KM3 sólo ha servido para facilitar la codificación en un estilo amigable a "lo Java" en lugar de emplear alguno de los estándares de metamodelado basados en XML (MOF/XML, Ecore).

Completada la fase de la escritura de los metamodelos, el siguiente paso es la creación del modelo BPMN de entrada (bpmnModel.ecore) que será transformado al correspondiente modelo XPD L de salida (xpdlModel.ecore), ambos escritos en Ecore. Con fines ilustrativos, se ha elegido como modelo BPMN el mostrado en la figura 7. Se puede observar que sólo se incluyen los elementos Pool, StartEven (subtipo de Event), EndEvent (subtipo de Event), UserTask (subtipo de Task) y SequenceFlow. El Pool es un servicio web formado por dos eventos (InicioEvento y FinEvento), dos tareas (RecibiendoPetición y EnviandoPeticiónDisponible) y tres flujos de secuencia que conectan InicioEvento con RecibiendoPetición, RecibiendoPetición con EnviandoPeticiónDisponible y EnviandoPeticiónDisponible con FinEvent.



```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xmi:XMI xmi:version="2.0" xmlns:xmi="http://www.omg.org
  <Pool name="ServicioWeb"/>
  <StartEvent name="InicioEvento" eventType="WebSe
  <EndEvent name="FinEvento" eventType="WebService
  <UserTask name="RecibiendoPeticiones" implementa
  <UserTask name="EnviandoPeticionesDisponibles" i
  <SequenceFlow name="flecha1" from_element="Inici
  <SequenceFlow name="flecha2" from_element="Recib
</xmi:XMI>
```

Fig. 7. Modelo BPMN de ejemplo creado con la herramienta eBPMN y su correspondiente codificación (parte) en Ecore.

Escrito el modelo BPMN, la transformación es automática. Tan sólo hace falta asociar los metamodelos en Ecore y el archivo de transformación al proyecto, y ejecutar la transformación. Con el ejemplo anterior, se obtuvo el código Ecore de la figura 8, que representa el modelo XPDL correspondiente.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xmi:XMI xmi:version="2.0" xmlns:xmi="http://www.omg.org/X
  <Pool name="ServicioWeb"/>
  <StartEvent name="InicioEvento" triggerKind="WebService"
  <EndEvent name="FinEvento" triggerKind="WebServiceasdf"
  <TaskUser name="EnviandoPeticionesDisponibles" implementa
  <TaskUser name="RecibiendoPeticiones" implementation="Oth
  <Transition from_element="InicioEvento" to_element="Recib
  <Transition from_element="RecibiendoPeticiones" to_elemen
</xmi:XMI>
```

Fig. 8. Codificación en Ecore (parte) del modelo XPDL obtenido.

4.4 Principales Dificultades Encontradas

A pesar de que la experiencia ha resultado satisfactoria al alcanzarse los objetivos buscados, han surgido algunos problemas y dificultades que merecen ser destacados.

La tabla de correspondencias encontrada en [17] sólo es orientativa y, en realidad, no resuelve el problema de la transformación entre BPMN y XPDL, es decir, no precisa la correspondencia atributo por atributo de los elementos. Esta labor ha sido realizada dentro de éste experimento de la forma más completa posible, pero con la limitación de que se han encontrado elementos cuyos atributos no coinciden totalmente entre ambos lenguajes. Por ejemplo, en el caso de "Activity" (actividad), elemento excluido en los metamodelos parciales probados (ver tablas 2 y 3), los atributos de actividad en BPMN no coinciden con los de XPDL, y tampoco se cumple

que los de BPMN son un subconjunto de los de XPDL o viceversa. Por tanto, al transformar desde BPMN a XPDL siempre se pierden atributos y también al transformar en dirección contraria. En cambio, para el elemento "Event" (evento), incluido en los metamodelos parciales, los atributos que tiene en BPMN son los mismos que los que tiene en XPDL, aunque con distintos nombres.

En la siguiente tabla (Tabla 2) se muestran los elementos excluidos del metamodelo BPMN para la transformación.

Tabla 2. Elementos excluidos de la transformación del metamodelo BPMN

BPMN	
Activity	InputPropertyMaps
Artifact	Lane
Assignment	LaneElement
Association	Loop
BPMNObject	Message
BusinessProcessUnit	MessageFlow
Cancel	MultiInstanceLoop
Compensation	OutputPropertyMaps
ComplexGateway	ParallelGateway
ConnectingObject	Participant
DataBasedExclusiveGateway	Process
DataObject	ProcessObject
EmbeddedSubProcess	Property
Error	ReferenceSubProcess
EventBasedExclusiveGateway	Rule
ExclusiveGateway	StandardLoop
Expresión	Subprocess
FlowObject	Swimlane
Gate	TextAnnotation
Gateway	Timer
Group	Transaction
InclusiveGateway	WebService
IndependentSubProcess	

En la Tabla 3 se listan los elementos excluidos del metamodelo XPDL para la transformación.

Otros problemas encontrados están en relación con el entorno de desarrollo ADT. Se han encontrado errores y carencias que dificultaron la tarea en el momento de la codificación. En primer lugar, el entorno no compila el archivo de transformación, por lo que aunque se incluyan elementos incorrectos en las reglas de transformación que no existan o estén mal escritos, el compilador no lo detecta. Otro problema es la actualización de los metamodelos, cada vez que se modifican los metamodelos hay que pasarlos de nuevo a Ecore, ya que no se realiza automáticamente por parte del entorno. Otro aspecto a resaltar es la utilización de KM3. Como ya hemos indicado, no parece justificarse el empleo de una nueva notación, si al final la transformación se realiza en Ecore obligatoriamente. En cuanto a la documentación sobre ADT y el

lenguaje ATL, se han echado en falta ejemplos de transformaciones que ilustren casos más complejos, a la altura de la transformación aquí abordada.

Tabla 3. Elementos excluidos de la transformación del metamodelo XPDL

	XPDL
Activity	Object
ActivitySet	ParametrizedApplication
Annotation	Participant
Application	ReferencedApplication
Artifact	ResultCompesation
Association	ResultError
BlockActivity	ResultMultiple
Category	Route
DataField	SubFlow
DataObject	Transition
Element	Trigger
Event	TriggerIntermediateMultiple
ExternalPackage	TriggerMultiple
ExternalReference	TriggerResultLink
Field	TriggerResultMessage
Gateway	TriggerRule
Group	TriggerTimer
Lane	TypeDeclaration
MessageFlow	WorkFlowProcess

5 Conclusiones y Trabajo Futuro

La experiencia aquí presentada ha sido satisfactoria en el sentido de que se alcanzaron los objetivos perseguidos: i) se han podido conocer las características, ventajas y limitaciones del lenguaje ATL y la plataforma ADT; y ii) también se han podido conocer y comprobar las dificultades reales que supone la "buena idea" de la transformación automática de modelos para un caso de complejidades realistas. De forma adicional, se ha podido descubrir que todavía no existen metamodelos adecuados para las versiones actuales oficiales de BPMN y de XPDL. Sin la disponibilidad de dichos metamodelos de forma pública y estandarizada, se hace muy difícil e improbable el éxito del nuevo paradigma de desarrollo dirigido por modelos.

De forma más concreta, se ha descubierto que al establecer las correspondencias entre elementos de BPMN y XPDL surge una dificultad bastante grande porque no están nada claras las equivalencias semánticas entre ambos lenguajes. En nuestra opinión, dicha dificultad es debida a la naturaleza conceptual de BPMN frente a la naturaleza lógica de XPDL. Los conceptos empleados en BPMN se refieren al dominio del problema (cómo funciona la empresa) mientras que los de XPDL se refieren al dominio de la solución (cómo es el flujo de trabajo adecuado para que funcione así).

Creemos que un trabajo importante de cara al futuro será desarrollar una adecuada metodología que identifique y establezca las adecuadas correspondencias conceptual-lógico, entre BPMN y XPDL, así como la definición de reglas que permitan la generación de las partes de XPDL que no tienen una equivalencia en BPMN. El resultado de este esfuerzo puede llegar a tener tanta importancia como tienen, desde hace muchos años, las reglas de transformación entidad/relación (el BPMN de las bases de datos) a esquemas relacionales en SQL (el XPDL de las bases de datos).

Por último, como continuación de la experiencia aquí presentada y buscando superar las limitaciones de ADT ya comentadas, se ha decidido probar con otros entornos tecnológicos diferentes. Por ello, actualmente se acaba de empezar otra experiencia más completa con el entorno MOMENT [3], [9].

Agradecimientos

Este trabajo ha sido parcialmente financiado por los proyectos ENIGMAS (Junta de Comunidades de Castilla-La Mancha, Consejería de Educación y Ciencia, PBI-05-058), ESFINGE (Ministerio de Educación y Ciencia, Dirección General de Investigación, Fondos FEDER, TIN2006-15175-C05-05) y COMPETISOFT (Programa Iberoamericano de Ciencia y Tecnología para el Desarrollo, 506AC0287).

Referencias

1. Acuña, S.T. y Ferré, X., *Software Process Modelling*. In Proceedings of the 5th. ISAS-SCI 2001. Orlando Florida, USA, 2001: p. pp. 237-242
2. Allilaire, F. y Tarik, I., *ADT: Eclipse development tools for ATL*. Université de Nantes. Faculté de Sciences et Techniques. LINA, 2005
3. Boronat, A., Oriente, J., Gómez, A., Ramos, I. y Carsí, J.A., *An Algebraic Specification of Generic OCL Queries within the Eclipse Modeling Framework*. Second European Conf. on Model Driven Architecture (ECMDA). Bilbao (Spain). Springer LNCS. July 10th-13th 2006,
4. Budinsky, F., Steinberg, D., Ellersick, R. y Grose, T.J., *Book, Chapter 5 "Ecore Modeling Concepts"*. 2004, Addison Wesley Professional
5. Budinsky, F., Steinberg, D., Ellersick, R. y Grose, T.J., *Eclipse Modeling Framework*. 2004, Addison Wesley Professional
6. Dufresne, T. y Martin, J., *Process Modeling for E-Business*. George Mason University, Spring 2003, INFS 770 - Methods for Informations Systems Engineering: Knowledge Management and E-Business., 2003
7. ATLAS group LINA & INRIA, *ATL: Atlas Transformation Language ATL. User Manual; Versión 9.7*. 2006.
8. ATLAS group LINA & INRIA, *KM3:Kernel MetaMetaModel Manual; Version 0.3*. 2004.
9. Grupo ISSI: MOMENT (A framework for Model Management), Consultado el 10-enero-2006. <http://moment.dsic.upv.es/>
10. Object Management Group, *Business Process Modeling Notation Specification*. 2006. Final Adopted Specification: dtc/06-02-01.
11. Object Management Group, *Meta Object Facility (MOF) 2.0 Query/View/Transformation Specification*. 2005. Final Adopted Specification ptc/05-11-01.

12. Object Management Group, *Meta Object Facility (MOF) Specification; version 1.4*. 2002. Document. formal/02-04-03.
13. Object Management Group, *Model-Driven Architecture (MDA) Guide Version 1.0.1*. 2003. OMG Document, omg/2003-06-01. <http://www.omg.org/mda/aspects.htm>
14. Object Management Group, *OCL 2.0 - OMG Final Adopted Specification*. 2003. ptc/03-10-14.
15. Sharp, A. y McDermott, P., *Workflow Modeling: Tools for Process Improvement and Application Development*. London: Artech House, 2001
16. Soyatec, *eBPMN Designer*. <http://www.soyatec.com/ebpmn/>
17. CAPE VISIONS. Software to Simplify Complexity, *XPDL with BPMN Extensions*. 2004. [http://www.omg.org/bpmn/Documents/WfMC/XPDL with BPMN Extensions.doc](http://www.omg.org/bpmn/Documents/WfMC/XPDL%20with%20BPMN%20Extensions.doc)
18. W3C, *Extensible Markup Language (XML) 1.0 (Fourth edition)*. 2006. A technical recommendation standard of the W3C. <http://www.w3.org/TR/REC-xml>
19. The Workflow Management Coalition Specification, *Process Definition Interface - XML Process Definition Language; Version 2.0*. 2005. Document Number WFMC-TC-1025. Document Status - Final.

Verification of Models in a MDA Approach for Collaborative Business Processes

Pablo D. Villarreal¹, Jorge Roa¹, Enrique Salomone^{1,2}, Omar Chiotti^{1,2}

¹CIDISI - Universidad Tecnológica Nacional - Facultad Regional Santa Fe, Lavaisse 610, 3000, Santa Fe, Argentina

(pvillarr@frsf.utn.edu.ar)

²INGAR-CONICET, Avellaneda 3657, 3000, Santa Fe, Argentina

({salomone,chiotti}@ceride.gov.ar)

Abstract. The application of a MDA approach for the automatic generation of B2B specifications from collaborative business process models enables enterprises to implement B2B collaborations with their partners, reducing the costs, time and complexity in the generation of technological solutions. Because of collaborative processes describe the behavior of the interactions between the partners, an important issue is the verification of the correctness of these processes, such as the absence of deadlocks and livelocks. In this work we describe how collaborative process models can be verified exploiting the benefits of a MDA approach. We describe the formalization of collaborative process models defined with the UP-ColBPIP language by using Petri Nets and the transformations of UP-ColBPIP models into Petri Nets specifications. Thus, a formal semantics for collaborative process models is provided, which enables the verification of these models through tools for verification of Petri Nets.

Keywords: MDA, Business-to-Business, Collaborative Business Processes, Petri Nets, Model Transformations

1 Introduction

The modeling and specification of collaborative business processes is a central issue in order to enterprises can implement Business-to-Business (B2B) collaborations. Through these processes, partners undertake to jointly carry out decisions to achieve common goals, coordinate their actions and exchange information.

Models of collaborative processes are defined in the business level of a B2B collaboration. They are designed from a business perspective by business analysts and system designers, who are not acquainted and do not want to deal with the technical details of the B2B collaboration. Hence, collaborative process models should be defined independent of any implementation technology [1].

Specifications of collaborative processes are defined in the technological level of a B2B collaboration, in which enterprises focus on the implementation of the B2B information systems that support the collaborative processes. In this level enterprises have to generate the specifications of both collaborative processes and partners'