U.N.L.P.

UNIV. NACIONAL
DE LA PLATA

U.C.

UNIV. CATÓLICA
NTRA. SRA.
DE LA ASUNCIÓN

Una propuesta de transformación M2M para el análisis de la fase ASM de MoWebA

TESIS PRESENTADA PARA OBTENER LOS GRADOS DE

MAGÍSTER EN INGENIERÍA DE SOFTWARE
POR LA U.N.L.P.

&

INGENIERO EN INFORMÁTICA
POR LA U.C.

Autor:
Daniel BONHAURE

Tutor:
Ing. Magalí GONZÁLEZ

Co-Tutor:
M.Sc. Nathalie AQUINO

Co-Director:
Dra. Claudia PONS

22 de abril de 2016

1. Introducción

El Desarrollo de Software Dirigido por Modelos (MDD - del inglés “Model-Driven Development”) es una disciplina relativamente nueva dentro de la ingeniería del software. Aunque, de un tiempo a esta parte, ha adquirido gran importancia y ha alcanzado avances considerables, al punto que se la considera como “un nuevo paradigma de desarrollo de software” [29].

Emulando a los lenguajes de alto nivel e inspirado en su éxito, el enfoque MDD propone el uso de un mayor nivel de abstracción en la especificación tanto del problema como de la solución, en relación con los métodos tradicionales de desarrollo de software [29].

El enfoque MDD es la evolución natural de la ingeniería de software basada en modelos, más específicamente, del Desarrollo Basado en Modelos (MBD - del inglés “Model-Based Development”), enriquecida mediante el agregado de transformaciones automáticas entre modelos. De hecho, el adjetivo “dirigido” (driven) en MDD, a diferencia de “basado” (based) en MBD, enfatiza el hecho que en este paradigma los modelos son sumamente importantes, tanto o más que el código fuente, ya que la idea central es lograr el mayor grado posible de automatización a la hora de generar código, partiendo de los modelos generados en la etapa de diseño.

Existen varias propuestas concretas para MDD. Una de las más extendidas actualmente es la Arquitectura Dirigida por Modelos o MDA, propuesta para MDD desarrollada por el Object Management Group y promovida a partir del 2000. MDA es una arquitectura que proporciona un conjunto de guías para estructurar especificaciones expresadas como modelos, siguiendo el proceso MDD [29].

En el contexto de MDA es posible que existan múltiples modelos para un sistema determinado, los cuales van sufriendo sucesivas transformaciones hasta llegar al código fuente. En MDA, la funcionalidad del sistema es definida primero como un Modelo Independiente de la Plataforma o PIM. Estos PIM son traducidos luego, mediante transformaciones modelo a modelo (M2M), a uno o varios Modelos Específicos de la Plataforma o PSM. Y finalmente, los PSM son traducidos a código mediante transformaciones modelo a texto (M2T) [29] [9].

Enmarcado dentro de lo que sería el framework MDA, encontramos a MoWebA [14] (del inglés, Model Oriented Web Approach); un enfoque metodológico MDD para el desarrollo de aplicaciones WEB que adopta el estándar MDA en sus diferentes fases.

MoWebA se define en [13] como una aproximación basada en modelos para el desarrollo de aplicaciones Web que, además de definir elementos relacionados al proceso de desarrollo del software, contempla aspectos metodológicos (etapas, productos, dimensiones) y los contempla en un entorno que incluye herramientas de modelado y generación automática de código, uso de estándares, arquitectura robusta a través de la separación de conceptos, etc.

Una de las problemáticas abordadas por MoWebA se hace evidente al analizar la adopción, dentro de MDD, de las arquitecturas¹ RIA² (del inglés, Rich Internet Application). Resulta que un gran número de las metodologías MDD existentes en la actualidad adoptan las arquitecturas RIA extendiendo sus modelos PIM, más específicamente, la notación de sus modelos PIM, con primitivas adicionales o patrones [14]. Si bien estos cambios derivan en metodologías muy similares a las originales, estas son, en esencia, nuevas metodologías. Evidencia de esto es que, por ejemplo, WebML define WebML RIA [4]; UWE, UWE para RIA [18]; OO-H, OOH4RIA [21]; etc.

Esta necesidad de definir modelos PIM específicos para arquitecturas diferentes, va en contra del principio de portabilidad del PIM pretendido tanto por MDD como por MDA, y, como ya se mencionó, es una de las problemáticas abordadas por MoWebA. En concreto, el problema de portabilidad de los modelos PIM es abordado por MoWebA mediante la inclusión de un nuevo nivel de abstracción a los ya establecidos por el framework MDA.

Este nuevo nivel de abstracción lo constituye el modelo ASM (del inglés, Architecture Specific Model). El ASM está conformado por un modelo intermedio entre el PIM y el PSM cuya finalidad es la de enriquecer los modelos previos con información adicional referente a la arquitectura del sistema [12]. Arquitecturas como RIA, SOA, REST son ejemplos claros de las arquitecturas que referimos.

Al capturar toda la información referente a la arquitectura, el modelo ASM propuesto por MoWebA permite trasladar la definición de esta, desde el dominio del problema, donde se sitúa cuando es definida en el PIM, al dominio de la solución, donde se sitúa cuando es definida en el ASM y, lo más importante, libera a los modelos PIM de contenerla. Esta separación planteada, permite alcanzar la tan ansiada portabilidad del PIM, contribuyendo incluso a la evolución del software al permitir la creación de modelos PIM más legibles, y por ende, fáciles de mantener.

¹Consideradas como arquitecturas por MoWebA [14] y WebML RIA [4].

²Aplicaciones Web con características similares a las aplicaciones de escritorio.

El problema con el enfoque adoptado por otras metodologías que, a diferencia de MoWebA, ubican la definición de la arquitectura del sistema en el dominio del problema (el PIM) en lugar de en el dominio de la solución (el ASM), es que cuando se requiere un cambio de arquitectura es necesario re-escribir todos los modelos PIM del sistema, algo que, adoptando el enfoque planteado por MoWebA, no hubiese sido necesario.

Si bien la separación de conceptos es considerada como beneficiosa por la comunidad toda y, al menos en principio, la inclusión de la fase ASM definida por MoWebA conlleva importantes beneficios para el ciclo de desarrollo propuesto por MDD, no existen aún suficientes estudios como para validar estas afirmaciones o evaluar el grado de automatización de las transformaciones M2M que podría ser alcanzado mediante la inclusión de esta nueva fase.

Justamente esto es lo que se pretende en el presente proyecto. Sin embargo, antes de llevar adelante un análisis de este tipo, es necesario establecer reglas de transformación M2M que permitan traducir modelos PIM a modelos ASM para alguna arquitectura específica, ya que MoWebA aún no dispone de estas reglas para ninguna.

El presente trabajo ha sido desarrollado con el apoyo financiero del Consejo Nacional de Ciencia y Tecnología (CONACYT, Paraguay) en el marco del proyecto denominado “Mejorando el proceso de desarrollo de software: propuesta basada en MDD” (14-INV-056); el cual es llevado adelante en conjunto entre el Departamento de Electrónica e Informática (DEI) de la Universidad Católica “Nuestra Señora de la Asunción” y el Centro de Investigación en Métodos de Producción de Software (PROS) de la Universidad Politécnica de Valencia, España.

El resto del documento se organiza de la siguiente manera: la sección 2 describe el estado del arte del área de interés del presente trabajo; la sección 3 presenta los objetivos perseguidos; la sección 4 describe la propuesta planteada y la sección 5 presenta el estado actual de las actividades establecidas para la realización de esta tesis.

2. Estado del Arte

La importancia adquirida por MDD dentro del campo de la Ingeniería del Software se debe, en parte, a las promesas realizadas por este paradigma y, en parte, a la necesidad de la comunidad de encontrar nuevas formas de enfrentar los problemas cada vez más complejos que conlleva la producción de software.

Si bien los problemas per se no cambiaron mucho desde que iniciara la *crisis del software*³, la complejidad de estos no hizo más que ir en aumento ya que cada vez se acortan más los tiempos disponibles para producción; aumentan las exigencias en cuanto a confiabilidad, usabilidad, robustez y demás características del software; las funcionalidades demandadas son cada vez más complejas, hasta el punto de requerir complicadas interacciones con el medio ambiente (un ejemplo es la realidad aumentada en teléfonos inteligentes), etc.

El enfoque MDD es una de las propuestas actuales que buscan enfrentar la crisis del software. Emulando a los lenguajes de alto nivel e inspirado en su éxito, el enfoque MDD propone el uso de un mayor nivel de abstracción en la especificación tanto del problema como de la solución, en relación con los métodos tradicionales de desarrollo de software [29].

En un primer momento los programas se escribían usando lenguajes de bajo nivel como el Assembler. Más tarde se crearon lenguajes de alto nivel, incluso, actualmente existen lenguajes de tan alto nivel que tienen un parecido considerable con el lenguaje natural y el hecho de que éstos son traducidos a programas escritos en lenguajes de bajo nivel es algo que pasa totalmente desapercibido. Es más, actualmente se considera que los compiladores son tan buenos que difícilmente un programador logre código tan eficiente como el producido por estos.

La gran promesa de MDD es lograr lo mismo, pero basándose completamente en modelos, es decir, realizando un trabajo similar al compilador, pero partiendo de modelos en lugar de código en algún lenguaje de alto nivel. ¡Una meta ambiciosa!, sin embargo, en su momento también se consideró ambiciosa la meta propuesta por los lenguajes de alto nivel.

Si bien MDD presenta actualmente los mismos inconvenientes que presentaron en su momento los compiladores, es decir, produce código poco eficiente, requiere eventualmente de la intervención del programador, entre otras cosas; esto no es motivo para no creer en ella, sino todo lo contrario, es un incentivo para seguir mejorando este paradigma.

En otras palabras, MDD plantea una forma completamente nueva de entender el desarrollo y mantenimiento de sistemas de software en donde los modelos son utilizados para dirigir las tareas de comprensión,

³Término acuñado en la primera conferencia organizada por la OTAN sobre desarrollo de software en 1968.

diseño, construcción, pruebas, despliegue, operación, administración, mantenimiento y modificación de los sistemas [29].

Para lograr esto MDD identifica cuatro tipos de modelos [29]:

- CIM:

Los CIM (Computational Independent Model), son los modelos con más alto nivel de abstracción y son independientes de cualquier metodología computacional. Los CIM representan el contexto, los requerimientos y el propósito de la solución pero todo esto sin ninguna implicación computacional. Usualmente se hace referencia a los CIM como modelos del dominio ya que en su construcción se utiliza vocabulario que resulta familiar para los expertos en el dominio en cuestión. En principio, hay partes del CIM que ni siquiera pueden ser mapeadas a la implementación final del software.
- PIM:

Los PIM (Platform Independent Model), son modelos independientes de la plataforma, es decir, independientes de cualquier tecnología de implementación. Este es el nivel que describe el comportamiento y la estructura de la aplicación pero de una manera en la que resulta independiente de la plataforma de implementación [5]. Dentro del PIM el sistema se modela desde el punto de vista de cómo se soporta mejor al negocio, sin tener en cuenta como va a ser implementado [29]. Es importante notar que el PIM es solo para las partes del CIM que serán resueltas mediante una solución basada en software.
- PSM:

Los PSM (Platform Specific Model), son modelos específicos para una plataforma dada. Cada PSM es la proyección del PIM a una plataforma específica [29]. Incluso si el PSM no será ejecutado directamente, lo que sí puede llegar a ocurrir por ejemplo al interpretar código [5], esta representación debe tener toda la información pertinente al comportamiento y la estructura de una aplicación con respecto a una plataforma específica de tal manera que los desarrolladores puedan traducirla linealmente a código.
- IM,

Los IM (Implementation Model), son los modelos que representan el código fuente. Constituyen el paso final del desarrollo basado en modelos (MDD), son básicamente la traducción o transformación de cada PSM a código.

Cada uno de estos modelos representan o modelan al mismo sistema mostrando diferentes niveles de abstracción, cuando mayor es el nivel de abstracción, más general es la información que provee el modelo sobre el sistema modelado.

En general las diferencias entre el modelo de máximo nivel de abstracción y el de mínimo nivel de abstracción son bastante grandes, sin embargo, agregando los niveles intermedios obtenemos modelos con la diferencia justa como para que, a pesar de las diferencias, sea posible ir transformando uno en otro con relativa facilidad.

Estas transformaciones son justamente las que permiten a MDD obtener código partiendo del diseño del sistema y por tanto, constituyen uno de los pilares de MDD junto con los modelos y las herramientas que dan soporte, ya sea a la creación de modelos, como a su posterior transformación.

Básicamente, una transformación entre modelos puede verse como un programa de computadora que toma un modelo como entrada y produce un modelo como salida [29]. Dentro de MDD se manejan dos tipos de transformaciones: aquellas que producen como resultado otros modelos y aquellas cuyo resultado es un conjunto de artefactos textuales. A las primeras se las conoce como transformaciones modelo a modelo o M2M y las segundas como transformaciones modelo a texto o M2T.

Ambos tipos de transformaciones, M2M y M2T, cuentan con lenguajes específicos que permiten su especificación y ejecución. Dos de los lenguajes de transformación M2M más populares son ATL (ATLAS Transformation Language) y QVT (Query/View/Transformation). Los lenguajes de transformación M2T, sin embargo, son bastante más numerosos y entre los más reconocidos podemos encontrar a Acceleo, AndroMDA, JET, MOFScript, Xpand.

Siguiendo con el enfoque MDD, cabe mencionar la existencia de varias propuestas concretas para este paradigma. Una de las más extendidas es MDA, arquitectura que proporciona un conjunto de guías para estructurar especificaciones expresadas como modelos, siguiendo el proceso MDD [29].

La idea clave detrás de MDA es la de separar la lógica del negocio (la funcionalidad de una aplicación) de su implementación en una determinada plataforma. De este modo, el desarrollador crea modelos de alto nivel e independientes de la plataforma (PIM), que posteriormente, mediante transformaciones modelo a modelo (M2M), generan de manera automática modelos dependientes de la plataforma (PSM). Y finalmente, al igual que en MDD, el código de la aplicación es derivado de los modelos independientes de la plataforma (PSM) mediante transformaciones modelo a texto (M2T) [9].

Siguiendo las directrices del OMG, las dos motivaciones principales de MDA son:

- La interoperabilidad, es decir, la independencia de los fabricantes a través de estandarizaciones, y
- La portabilidad, o sea la independencia de la plataforma.

Enmarcado dentro de lo que sería el framework MDA encontramos a MoWebA, un enfoque metodológico MDD para el desarrollo de aplicaciones WEB que adopta MDA en sus diferentes fases.

Un aporte diferenciador de MoWebA es la inclusión de un nuevo nivel de abstracción a los ya definidos por el framework MDA. Este nuevo nivel lo constituye el modelo ASM [12] (del inglés, Architecture Specific Model) que permite, entre otras cosas, omitir la definición de modelos PIM específicos para cada tipo de arquitectura, fortaleciendo de esta manera la portabilidad de los mismos [14].

El modelo ASM constituye un nivel de abstracción intermedio en el PIM y el PSM cuyo principal objetivo es capturar todos los aspectos referentes a la arquitectura del sistema en desarrollo. En este sentido, la mayor parte de los trabajos realizados sobre arquitecturas para el modelo ASM de MoWebA contemplan las arquitecturas RIA.

Analizando el estado actual de la implementación de las arquitecturas RIA dentro de MoWebA, podemos notar que este enfoque MDD ya cuenta con meta-modelos/modelos ASM para arquitecturas RIA. Sin embargo, la mayoría de los trabajos, ya sean estos previos o paralelos a este (algunos enmarcados dentro del mismo proyecto de investigación que esta tesis), se orientan a automatizar solo las transformaciones M2T, permitiendo de esta manera la transformación de modelos ASM a código final pero manteniendo manual la transformación de modelos PIM a modelos ASM.

Por lo tanto, la automatización de la transformación de modelos PIM a modelos ASM para arquitecturas RIA constituye una importante oportunidad de investigación, sobre todo considerando que esto permitirá completar el proceso de generación automática de la aplicación final, partiendo del PIM, y pasando por el ASM, el PSM y el código.

Es por esto que, considerando lo mencionado en los párrafos previos, sobre todo el hecho que permitirá completar la automatización del ciclo de desarrollo de MoWebA, el presente trabajo adopta a RIA como la arquitectura a ser utilizada para el análisis de la fase ASM.

Como parte de la revisión bibliográfica llevada a cabo en el contexto de este trabajo, se ha realizado un Mapeo Sistemático de la Literatura o SMS (del inglés, Systematic Mapping Study) con el objetivo de identificar qué se ha hecho y qué se debe hacer en el futuro, en el contexto de las metodologías MDD con enfoque RIA.

La elección de un SMS por sobre una SLR obedece a que el mapeo ofrece un alcance más amplio que las revisiones al permitir la inclusión de todos los artículos que tratan sobre un área de interés y no solo aquellos que contengan o abarquen estudios empíricos [11].

El principal objetivo del mapeo es analizar las propuestas y los trabajos de investigación existentes con el fin de identificar áreas poco estudiadas, trabajos pendientes, y en general, cualquier oportunidad de investigación que pueda o requiera ser abordada en el futuro en el contexto del desarrollo MDD de aplicaciones RIA y sus transformaciones M2M, pero sobre todo, demostrar la validez de lo mencionado hasta el momento.

La elaboración del SMS demandó que fueran reunidas, clasificadas y analizadas todas las investigaciones sobre MDD para arquitecturas RIA y transformaciones M2M realizadas entre el 2002, año de aparición del término RIA [1], y el 2015 (ambos inclusive).

El mapeo se realizó siguiendo las directrices definidas por Genero et. al en [11]. Por lo tanto, según lo indicado por estos autores y en concordancia con los objetivos del mapeo mencionados en los párrafos anteriores, en el Cuadro 1 se presentan las preguntas de investigación definidas para su realización.

Preguntas	Motivación
PI1: ¿Cuáles son los objetivos perseguidos en la investigación sobre MDD para RIA y sus transformaciones M2M?	Determinar donde se encuentra la mayor parte del interés de investigación y qué áreas han sido poco investigadas, explorando conceptos básicos, recopilando conocimiento de prácticas actuales o con el objetivo de avanzar la práctica a través de la ciencia del diseño (Design Science).
PI2: ¿Qué tipo de métodos se utilizaron en la investigación relacionada con MDD para RIA y sus transformaciones M2M?	Determinar si las propuestas en este campo de investigación son más prácticas o más de investigación básica y también identificar oportunidades para realizar futuras investigaciones.

PI3: ¿Cuál es el alcance de las propuestas en los trabajos presentados?	Encontrar el tipo de resultados obtenidos en cada uno de los trabajos analizados, por ejemplo, nuevas propuestas, herramientas implementadas, etc.
PI4: ¿Cuáles de las fases de modelado son contempladas?	Identificar qué fases de MDA adoptan cada una de las propuestas analizadas.
PI5: ¿La propuesta cuenta con herramientas que la soporten? ¿Cuál es el estado de desarrollo de estas herramientas?	Identificar si la propuesta llegó a ser más que solo eso, es decir, si existe alguna herramienta que permita probarla.
PI6: ¿Cuál es el IDE señalado para la utilización de la herramienta propuesta?	Se pretende identificar el IDE para el cual fue desarrollada la herramienta propuesta.
PI7: ¿Cuál es el nivel de automatización de la transformación M2M contemplada?	Identificar el nivel de automatización de las transformaciones M2M planteadas.
PI8: ¿Pertencen, los modelos origen y destino, al mismo nivel de abstracción?	Identificar el nivel de abstracción de los modelos origen y destino y analizarlos.
PI9: ¿Cuál es el lenguaje en el que se expresan los modelos origen y destino?	Diferenciar si los modelos origen y destino están expresados en lenguajes diferentes o no.
PI10: ¿Cuál es la relación entre los modelos origen y destino?	Identificar si el modelo destino constituye un nuevo modelo o se obtiene mediante una actualización del modelo origen.
PI11: ¿Qué relación hay entre el número de modelos origen y destino?	Se pretende identificar si se permiten o no múltiples modelos origen y/o múltiples modelos destino.
PI12: ¿Qué lenguaje de la transformación modelo a modelo utiliza la propuesta analizada?	Se busca identificar el lenguaje de transformación M2M utilizado.

Cuadro 1: Preguntas de Investigación

Además, se estableció un esquema de clasificación teniendo en cuenta cada una de las preguntas de investigación (ver Cuadro 1) y la literatura relevante (ver Cuadro 2), para de esta forma restringir las respuestas posibles a cada una de las preguntas de investigación.

Sin embargo, antes de presentar el mencionado esquema de clasificación, es importante profundizar en la taxonomía de las transformaciones M2M ya que algunas de ellas son consideradas en este esquema.

Si bien existen numerosas formas de categorizar las transformaciones M2M, en el presente estudio nos limitaremos a cuatro solamente. Estas cuatro son consideradas debido a su trascendencia a la hora de arrojar luz sobre la situación actual del estado del arte de las transformaciones M2M o el grado de adopción del framework MDA en las metodologías MDD analizadas.

En [24] podemos encontrar tres de las clasificaciones utilizadas. Estas son las correspondientes a las preguntas de investigación 8, 9 y 11. Mientras que en [7] encontramos la clasificación restante, la correspondiente a la pregunta de investigación 10.

En la pregunta de investigación 8, “¿Pertencen, los modelos origen y destino, al mismo nivel de abstracción?”, se ha considerado la clasificación propuesta por [24] y que refiere al nivel de abstracción de los modelos origen y destino. Concretamente, según esta categorización contamos con dos opciones diferentes a la hora de clasificar las transformaciones M2M: transformaciones horizontales y transformaciones verticales.

Las transformaciones M2M horizontales son aquellas en las que los modelos origen y destino se encuentran en el mismo nivel de abstracción. Mientras que las transformaciones M2M verticales son aquellas en las que los modelos origen y destino se encuentran en niveles de abstracción diferentes.

En la pregunta de investigación 9, “¿Cuál es el lenguaje en el que se expresan los modelos origen y destino?”, se ha considerado la clasificación propuesta por [24], la cual apunta al lenguaje de modelado en el que están escritos los modelos origen y destino. De acuerdo a esta taxonomía las transformaciones son clasificadas en dos categorías: transformaciones M2M endógenas y transformaciones M2M exógenas.

Las transformaciones M2M endógenas son aquellas en las que los modelos origen y destino están escritos en el mismo lenguaje de modelado. Las transformaciones M2M exógenas, por el contrario, son aquellas en las que los modelos origen y destino están escritos en lenguajes de modelado diferentes.

En la pregunta de investigación 10, “¿Cuál es la relación entre los modelos origen y destino?”, se ha considerado la clasificación propuesta por [7]. Aquí se propone como criterio de clasificación la relación existente entre los modelos origen y destino, lo que deriva en dos tipos de transformaciones M2M. Una opción en que el modelo destino sea uno totalmente nuevo y la otra es que el modelo destino sea creado a partir de la modificación del modelo origen.

En la pregunta de investigación 11, “¿Qué relación hay entre el número de modelos origen y destino?”, se ha considerado la clasificación propuesta por [24]. Donde el criterio de clasificación es el número de modelos origen y destino involucrados en el proceso de transformación y se definen cuatro categorías diferentes. Estas son: 1 a 1, 1 a *, * a 1, * a *.

Las transformaciones 1 a 1 son aquellas en las que un solo modelo origen genera un solo modelo destino. Las transformaciones 1 a * son aquellas en las que un solo modelo origen deriva en múltiples modelos destino. Las transformaciones * a 1, por el contrario, son aquellas en las que múltiples modelos origen producen un solo modelo destino. Y finalmente, las transformaciones * a * son aquellas en las que múltiples modelos origen son transformados en múltiples modelos destino.

Dimensiones	Categorías
PI1: Objetivo de la investigación	Evaluar, Mejorar, Proponer. Evaluar y Mejorar, Evaluar y Proponer.
PI2: Tipo de evidencia/método de la investigación	Empírico: experimento, estudio de caso, encuesta. No empírico: ejemplos y sus variantes, proyectos de materia, revisiones ⁴ y/o mapeos ⁵ de la literatura.
PI3: Tipo de resultado de la investigación	Método (técnica, metodología, proceso). Herramienta, Método y Herr., Trazabilidad.
PI4: Fases de MDA	CIM, PIM, PSM. Combinaciones entre CIM, PIM y PSM. Ninguna.
PI5: Estado del desarrollo de una Herramienta que aborde la propuesta	Nivel 0: no se propone una herramienta. Nivel 1: definición teórica o trabajo futuro. Nivel 2: implementación parcial. Nivel 3: implementación total. Nivel 4: disponibilidad de la herramienta. Nivel 5: disp. de herr. y documentación.
PI6: IDE propuesto	Eclipse. MagicDraw. Otro.
PI7: Nivel de automatización de la transformación M2M	Automática. Semi-Automática. Manual. Automática y Semi-Automática. Semi-Automática y Manual.
PI8: Clasificación según el nivel de abstracción de los modelos origen y destino	Transformación M2M Horizontal. Transformación M2M Vertical.
PI9: Clasificación según el lenguaje de los modelos origen y destino	Transformación M2M Endógena. Transformación M2M Exógena.
PI10: Relación entre los modelos origen y destino	Se puede dar que el destino sea un nuevo modelo, se actualice el modelo origen o ambas situaciones.
PI11: Número de modelos origen y destino	Uno a Uno. Uno a Muchos. Muchos a Uno. Muchos a Muchos.
PI12: Lenguaje de la transformación M2M	ATL, QVT o una combinación de ambos. Otro lenguaje de transformación.

Cuadro 2: Dimensiones / Categorías

⁴Las SLR son clasificadas como estudios no empíricos en [11].

⁵Los SMS son clasificados como estudios no empíricos en [11].

Los términos de búsqueda, ya sean estos principales, sinónimos, palabras alternativas o términos relacionados con los términos principales se presenta en el Cuadro 3.

Términos principales	Términos alternativos
MDD, MDE, MDA	DSDM, MDSD, MDSE, Model-Driven, Model Driven
M2M, ATL, QVT	transformation, Model-to-Model
RIA	Rich Internet Applications

Cuadro 3: Términos de Búsqueda

La cadena de búsqueda se define a partir del Cuadro 3 mediante la ejecución de las siguientes instrucciones:

1. El término principal se une a los términos alternativos mediante el operador lógico OR.
2. Los términos principales se unen unos con otros mediante el operador lógico AND.

Resultando lo siguiente:

(“MDD” OR “DSDM” OR “MDSD” OR “MDE” OR “MDSE” OR “MDA” OR “Model-Driven” OR “Model Driven”)
AND
 (“transformation” OR “M2M” OR “Model-to-Model” OR “ATL” OR “QVT”)
AND
 (“RIA” OR “Rich Internet Applications”)

La cadena de búsqueda se aplicará a cada una de las fuentes de búsqueda señaladas en el Cuadro 4, buscando, siempre que el buscador de las bibliotecas digitales lo permita, en el título y el resumen, y cuando las bibliotecas digitales no lo permitan, sobre el texto completo.

Fuentes de búsqueda	Art. Encontrados	Art. Seleccionados
IEEE Xplore	31	7
Springer Link	57	19
Literatura Gris	3	3

Cuadro 4: Resultados por Fuente

Para la selección de estudios fueron definidos los siguientes criterios de inclusión y exclusión.

Criterios de Inclusión
Publicaciones relacionadas con MDD para RIA y transformaciones M2M.
Revistas académicas, artículos de conferencias y talleres.
Publicaciones en inglés, español y portugués.
Trabajos publicados entre el 2002 y el 2015.
Criterios de Exclusión (revisando los títulos y el resumen)
Publicaciones que no se centren en MDD para RIA o transformaciones M2M.
Artículos de los cuales se disponga solo el resumen.
Publicaciones que no hayan sido revisadas por pares.
Publicaciones sobre el mismo estudio, duplicadas en diferentes fuentes.
Criterios de Exclusión (revisando el texto completo)
Publicaciones en las que se mencionen los criterios de búsqueda, pero no constituyen el objeto de estudio.
Publicaciones sobre el mismo estudio, duplicadas en diferentes fuentes.

Cuadro 5: Criterios de Inclusión y Exclusión

Además de las fuentes de búsqueda consultadas para la realización del mapeo, en el Cuadro 4 también es posible observar, organizados por fuente de búsqueda, la cantidad de artículos encontrados y la cantidad

de artículos seleccionados luego de la aplicación de los criterios de inclusión y exclusión señalados en el Cuadro 5.

Finalmente, los artículos utilizados para la realización del mapeo son presentados en el Cuadro 6.

Fuentes	Artículos
IEEE Xplore	[19], [2], [26], [31], [21], [22], [37]
Springer Link	[38], [32], [25], [15], [35], [23], [16], [33], [6], [36], [34], [28], [10], [8], [27], [17], [20], [18], [40]
Literatura Gris	[39], [4], [30]

Cuadro 6: Artículos por Fuente

Gracias al análisis cuantitativo realizado sobre los datos obtenidos con la realización del Mapeo Sistemático o SMS (ver Cuadro 7) fue posible elaborar las siguientes conclusiones, una por cada pregunta de investigación:

1. La mayor parte del esfuerzo de investigación de la comunidad científica se centra en crear o proponer nuevas propuestas en lugar de estudiar mejor aquellas pre-existentes.
2. Tan solo el 48 % de los artículos respaldan sus afirmaciones con estudios empíricos.
3. Si bien la mayoría de las propuestas presentan tanto métodos como herramientas, estas herramientas no son de fácil acceso o solo se mencionan como trabajo futuro.
4. La mayoría de las propuestas adoptan las fases PIM y/o PSM, mientras que ninguna propone nuevas fases a las ya definidas por el estándar MDA.
5. Si bien muchas de las propuestas estudiadas proponen herramientas, pocas las desarrollan y publican.
6. El IDE más utilizado para el desarrollo de herramientas que soporten a los métodos propuestos, es Eclipse, ya sea creando nuevos IDEs a partir de este, como creando plugins para el Eclipse Modeling Framework (EMF).
7. Son pocas las herramientas que automatizan completamente el proceso de desarrollo, son más comunes los procesos semi-automáticos.
8. Según la taxonomía propuesta en [24], el tipo de transformaciones M2M más frecuente es la Transformación M2M Vertical.
9. Según la taxonomía propuesta en [24], el tipo de transformaciones M2M más frecuente es la Transformación M2M Endógena.
10. Según la taxonomía propuesta en [7], el tipo de transformaciones M2M más frecuente es aquella cuyos modelos destino son creados desde cero y no a partir de la modificación de los modelos origen.
11. Según la taxonomía propuesta en [24], el tipo de transformaciones M2M más frecuente es la transformación Muchos a Muchos.
12. En cuanto al lenguaje de transformación M2M, la misma cantidad de propuestas adoptan ATL (ATLAS Transformation Language) y QVT (Query/View/Transformation).

PI	Categoría	Artículos
PI1	Evaluar	0
	Mejorar	2
	Proponer	13
	Evaluar y Mejorar	1
	Evaluar y Proponer	13

PI2	Empírico	Experimento	1
		Caso de Estudio	13
	No Empírico	Ejemplo	6
		Running example	3
		Example scenario	1
		Proyecto de materia	1
		SMS y/o SLR	0
Sin validación	4		
PI3	Método	7	
	Herramienta	4	
	Método y Herr.	17	
	Trazabilidad	0	
	Sin clasificar	1	
PI4	CIM	0	
	PIM	2	
	PSM	3	
	CIM y PIM	1	
	CIM y PSM	1	
	PIM y PSM	15	
	CIM, PIM y PSM	5	
	Otro	0	
	Ninguna	2	
PI5	Nivel 0	No usa herramientas	3
		Usa herr. existentes	5
	Nivel 1	4	
	Nivel 2	4	
	Nivel 3	10	
	Nivel 4	1	
Nivel 5	2		
PI6	Eclipse	15	
	MagicDraw	2	
	Otro	3	
	Sin clasificar	9	
PI7	Automática	7	
	Semi-Automática	15	
	Manual	0	
	Automática y Semi-Automática	2	
	Semi-Automática y Manual	1	
	Sin clasificar	4	
PI8	Horizontal	2	
	Vertical	18	
	Ambas	3	
	Sin clasificar	6	
PI9	Endógena	20	
	Exógena	3	
	Sin clasificar	6	
PI10	Nuevo	20	
	Actualización	0	
	Ambos	3	
	Sin clasificar	6	
PI11	1 a 1	4	
	1 a *	4	
	* a 1	3	
	* a *	10	
	Sin clasificar	8	
PI12	ATL	11	
	QVT	11	

ATL y QVT	1
Otro	2
Sin clasificar	4

Cuadro 7: Resultados por Pregunta y Categoría

Además del análisis cuantitativo, fue llevado a cabo un análisis cualitativo. La razón de este segundo análisis es la posibilidad de expresar algunas conclusiones derivadas de la interpretación de los datos obtenidos, las cuales son presentadas a continuación.

En primer lugar es importante resaltar que ninguna de las propuestas analizadas proponen nuevas fases a las ya definidas por el estándar MDA. Además, ninguna de estas propuestas contempla la fase ASM introducida por MoWebA [13] [14] [12].

En general, las propuestas que extienden enfoques MDD pre-existentes para que soporten arquitecturas RIA, lo hacen extendiendo los modelos PIM. Por lo tanto, en detrimento de la portabilidad de estos modelos independientes de la plataforma, cuya portabilidad es uno de los pilares de MDD.

Debido fundamentalmente a la inmadurez del campo, la comunidad científica todavía centra sus esfuerzos en definir nuevas propuestas MDD para RIA. Y por ende, el número de herramientas desarrolladas y distribuidas es aún muy bajo. En este punto es importante recordar que otro de los pilares de MDD es la creación de potentes herramientas que permitan la automatización del proceso de desarrollo.

Fueron detectadas varias oportunidades de investigación. Entre ellas merecen una mención especial dos que consideramos particularmente importantes para MDD. Una es la falta de trabajos empíricos, sobre todo experimentos, que permitan validar aquellas propuestas que aún no fueron validadas con este nivel de rigurosidad. La otra es contribuir con el desarrollo de herramientas que soporten cada una de las propuestas existentes, lo que aporta a la comunidad de dos maneras diferentes: facilitando la realización de trabajos empíricos y fomentando la adopción de MDD por parte de la industria.

Finalmente, resulta importante mencionar que además de aportar al presente estado del arte, el mapeo realizado fue documentado en un Reporte Técnico, referenciado en este documento como [3] y que se buscará publicar en alguna conferencia.

3. Objetivos

En el marco del enfoque MoWebA, el objetivo general del presente proyecto final de carrera se centra en analizar el impacto del nuevo nivel de abstracción planteado por MoWebA, el modelo ASM, considerando para el efecto una arquitectura basada en RIA.

En pos de la consecución de este objetivo general será necesario avanzar sobre los siguientes objetivos específicos.

- Definir reglas de transformación que permitan transformar modelos PIM a modelos ASM para una arquitectura RIA.
- Validar el proceso de transformación PIM-ASM, garantizando que las reglas definidas se ajusten adecuadamente tanto al enfoque MoWebA como a las arquitecturas RIA.
- Definir y llevar adelante experiencias que aporten datos relevantes al objetivo general del presente trabajo de investigación.

Resulta importante señalar que los objetivos, tanto el general como los específicos, fueron propuestos considerando las oportunidades de investigación identificadas en el mapeo presentado en la sección anterior. Específicamente fueron abordadas las dos oportunidades señaladas al final del estado del arte: el bajo porcentaje de trabajos con validación empírica y la escasa disponibilidad de herramientas de automatización.

Una vez alcanzados los objetivos, MoWebA contará con reglas de transformación M2M para la transformación automática de modelos PIM a modelos ASM, lo que será un gran aporte tanto para MoWebA como para la comunidad toda, sobre todo considerando que, en el ámbito del proyecto de investigación del cual forma parte este trabajo, se llevan adelante otras investigaciones que abarcan el resto de las etapas planteadas por MoWebA, llegando incluso a la obtención de código final.

Además, se generarán datos y experiencias que permitirán validar de manera empírica los aportes de la inclusión del modelo ASM al ciclo de desarrollo propuesto por el enfoque MDD, permitiendo hacer luz sobre lo beneficiosa o no de esta inclusión y abordando otra de las posibilidades de investigación detectadas por el mapeo, la falta de validación empírica en muchos de los trabajos analizados.

4. Propuesta de Trabajo

En esta sección se presenta la problemática abordada por el presente trabajo de investigación, así como también, la solución propuesta y un escenario motivador que permite comprender tanto el problema planteado como la solución.

4.1. Problemática

La mayoría de las metodologías MDD existentes en la actualidad, excluyendo la adoptada para la realización de esta tesis, MoWebA, adoptan las arquitecturas RIA extendiendo sus modelos PIM, más específicamente, la notación de sus modelos PIM, con primitivas adicionales o patrones.

Esta extensión de los modelos PIM deriva en la existencia de modelos PIM específicos para diferentes arquitecturas, lo que, más que una simple adaptación/extensión de una metodología base, a fines prácticos, es más parecido a utilizar una nueva propuesta. Esto debido a que, una vez que los modelos PIM son especializados mediante la extensión de su notación, constituyen, a pesar de las semejanzas que pudieran existir, modelos diferentes, y por lo tanto, un cambio de arquitectura requerirá la creación de nuevos modelos PIM o, en el mejor de los casos, la modificación de los modelos PIM pre-existentes.

Otro aspecto a considerar es que la definición de modelos PIM específicos para una arquitectura dada viola el principio de portabilidad del PIM pretendido tanto por MDD como por MDA, así como también el principio de separación de conceptos promovido por esta última. Además, la complejidad añadida al modelado PIM por los detalles relacionados con la arquitectura dificulta el mantenimiento del sistema al aumentar innecesariamente la complejidad de estos modelos.

A fin de comprender mejor la situación, a continuación se plantea un ejemplo práctico; si utilizando alguna de las metodologías tradicionales (WebML, UWE, etc), pretendemos desarrollar aplicaciones bajo diferentes tipos de arquitectura (por ejemplo RIA, SOA y REST), nos veremos obligados a utilizar tres definiciones diferentes de modelos PIM, una para RIA (WebML RIA, UWE para RIA), otra para SOA (WebML SOA, UWE para SOA -si existiesen-), etc. Lo que, entre otras cosas, implica la necesidad de crear tres modelos PIM diferentes, triplicando el trabajo generado por un cambio que requiera una modificación a nivel de PIM.

Claramente este problema con la portabilidad del PIM es un problema que merece ser abordado, así como también aquellas dos oportunidades de investigación detectadas por el SMS: el bajo porcentaje de trabajos con validación empírica y la escasa disponibilidad de herramientas de automatización.

4.2. Propuesta de Solución

Una vez identificada la problemática, y a sabiendas de la existencia de MoWebA, un enfoque metodológico que la aborda; considerando además las conclusiones del SMS llevado a cabo como parte de la revisión bibliográfica del presente trabajo de tesis, fundamentalmente aquella que resalta la existencia de muchas propuestas incompletas y/o pendientes de validación y la necesidad de seguirlas estudiando; resulta claro que la propuesta de solución más apropiada sería adoptar MoWebA como base para extenderla/completarla.

Confrontando la propuesta de MoWebA (el modelo ASM), con la problemática identificada en la subsección anterior (el problema de portabilidad del PIM), es posible notar que la utilización de esta etapa intermedia entre el PIM y el PSM devolvería a los modelos PIM su portabilidad, ya que, por ejemplo, bastaría la definición de un solo modelo PIM para desarrollar las tres variantes del sistema planteado en el ejemplo anterior. Además, al librar al PIM de la definición de aspectos relacionados con la arquitectura se mejora su legibilidad, y por ende, su mantenibilidad.

Sin embargo, MoWebA, uno de los enfoques más nuevos dentro de lo que sería MDD orientado a arquitecturas RIA, todavía no es capaz de automatizar las transformaciones M2M requeridas para transformar un PIM a un ASM y este a un PSM. Y por lo tanto, las ventajas y/o desventajas que pudiera arrojar la inclusión del ASM al ciclo de desarrollo, sobre todo las ventajas prometidas al plantearse su inclusión, aún no fueron debidamente estudiadas y validadas con métodos lo suficientemente rigurosos.

En resumen, la solución propuesta consiste, en primer lugar, y como algo imperativo para la consecución del objetivo final de esta tesis, en la definición de reglas de transformación que permitan transformar modelos PIM a modelos ASM (para una arquitectura RIA). En este punto es importante remarcar que la correcta definición de estas reglas de transformación es crucial para evitar sesgos o desviaciones en los análisis posteriores. De esta manera, una vez estén completamente definidas y validadas las reglas de transformación, se llevarán a cabo distintas experiencias, recabando de ellas la mayor cantidad de datos

posibles, los cuales posteriormente serán analizados rigurosamente para determinar los aportes, ventajas y desventajas de la adopción del modelo ASM.

4.3. Escenario Motivador

El escenario motivar es básicamente una extensión del ejemplo presentado en las sub-secciones anteriores.

Supongamos el desarrollo de un sistema siguiendo alguna metodología MDD tradicional (WebML, UWE, etc), es decir, alguna que no plantee una etapa intermedia entre el PIM y el PSM que capture los detalles referentes a la arquitectura utilizada. Es más, supongamos un sistema que no adoptó una arquitectura específica, una sistema web tradicional y estático.

Asumiendo los supuestos del párrafo anterior, el proceso de desarrollo del sistema sería el siguiente: primero se crearían manualmente los modelos PIM, estos serían transformados a modelos PSM mediante transformaciones M2M automáticas y, finalmente, los PSM serían transformados a código mediante transformaciones M2T automáticas. El único proceso manual sería la creación de los PIM.

En concreto, la problemática abordada se manifiesta cuando el sistema anterior requiere ser modificado. Específicamente, cuando se requiere un cambio de arquitectura, o la adopción de una en nuestro caso.

En este punto se pueden dar dos situaciones bastante diferentes, dependiendo de la metodología MDD que haya sido adoptada para el desarrollo del sistema.

Si la metodología adoptada captura los detalles referentes a la arquitectura en los modelos PIM, entonces para realizar el cambio necesario se tendría que seguir el siguiente ciclo: primero se tendrían que volver a crear los modelos PIM, incluyendo esta vez los detalles referentes a la arquitectura, es decir, adoptando el PIM específico para la arquitectura deseada. Este es un proceso manual exactamente igual al realizado al desarrollar el sistema por primera vez. Luego, se procedería a transformar estos modelos PIM a modelos PSM con transformaciones M2M automáticas. Para, finalmente, transformar estos PSM a código mediante transformaciones M2T que también son automáticas.

Consideremos ahora el caso en el que la metodología adoptada captura los detalles referentes a la arquitectura en un modelo intermedio entre el PIM y el PSM. Consideremos, más específicamente, el modelo ASM de MoWebA. En este caso, el proceso necesario para realizar el cambio requerido es completamente automático ya que, se reutiliza el modelo PIM pre-existente y se lo transforma, mediante transformaciones M2M automáticas, a modelos ASM. Luego, estos modelos ASM son transformados a modelos PSM, también mediante transformaciones M2M automáticas y, finalmente, los modelos PSM son transformados a código mediante transformaciones M2T que también son automáticas.

Comparando ambos procesos de modificación, resulta evidente que la existencia del modelo ASM mejora los tiempos de desarrollo debido a que el cambio de arquitectura, al utilizarlo, resultó ser un proceso más automatizado. Por el contrario, la modificación utilizando PIM específicos requirió que los modelos PIM sean re-escritos de manera completamente manual, lo que implica un trabajo similar al llevado a cabo cuando el sistema fue desarrollado por primera vez.

5. Estado Actual del Proyecto

A continuación se describen brevemente las actividades que fueron llevadas a cabo y las que aún se encuentran pendientes.

Actividad	Estado
Investigación y revisión bibliográfica.	Finalizada
Instrucción en los diagramas definidos por MoWebA.	En Progreso
Definición y validación de las reglas de transformación PIM-ASM.	Pendiente
Validación empírica y análisis de resultados.	Pendiente
Elaboración del libro de tesis y preparación de la defensa final.	En Progreso

Cuadro 8: Estado de actividades del proyecto

5.1. Actividades finalizadas

Como se puede observar en el Cuadro 8, solo la primera actividad, “Investigación y revisión bibliográfica.”, ha sido finalizada hasta el momento, la razón es que esta contempla las actividades iniciales del proyecto, sobre todo la revisión de la literatura. Además de este documento, en esta etapa se llevó a cabo un Mapeo Sistemático de la Literatura y se redactó un reporte técnico que detalla, entre otras cosas, la metodología utilizada en la realización del mencionado estudio y los resultados obtenidos.

5.2. Actividades en desarrollo

La segunda actividad, “Instrucción en los diagramas definidos por MoWebA.”, es una actividad que se encuentra actualmente en proceso y que abarcará, más o menos, hasta que finalice la etapa siguiente, ya que en todo momento durante la definición de las reglas de transformación, será posible, y hasta necesario, profundizar el conocimiento de estos diagramas.

La quinta actividad, “Elaboración del libro de tesis y preparación de la defensa final.”, es una actividad que se encuentra actualmente en progreso ya que todos los documentos generados, este inclusive, serán utilizados para la redacción final del libro de tesis. No obstante, es probable que la mayor parte del trabajo de redacción se realice en las etapas finales del proyecto.

5.3. Actividades pendientes

La tercera actividad, “Definición y validación de las reglas de transformación PIM-ASM.”, es una etapa crucial de la que depende todo el trabajo a ser realizado, ya que una definición incorrecta de las reglas de transformación puede invalidar todos los análisis realizados en las etapas posteriores. Esta etapa todavía continúa pendiente y será abordada apenas sea posible.

La cuarta actividad, “Validación empírica y análisis de resultados.”, es una actividad que depende completamente de la actividad previa, todavía se encuentra pendiente y no será abordada hasta que finalice la tercera etapa. Además esta será la etapa con mayor contribución para con la comunidad científica relacionada a MDD.

Referencias

- [1] Allaire, J.: Macromedia Flash MX-A next-generation rich client. Macromedia White Paper pp. 1–2 (2002)
- [2] Bernardi, M.L., Lucca, G.A.D., Distante, D.: Model-driven fast prototyping of rias: From conceptual models to running applications. In: *Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on)*. pp. 250–258. IEEE (Sept 2014)
- [3] Bonhaure, D., González, M., Aquino, N., Cernuzzi, L.: Transformaciones modelo a modelo para arquitecturas ria: un Mapeo Sistemático de la Literatura. Tech. rep., Universidad Católica “Nuestra Señora de la Asunción” (UCA) (2016)
- [4] Bozzon, A., Comai, S., Fraternali, P., Carughi, G.T.: Conceptual modeling and code generation for rich internet applications. In: *ICWE 2006. Proceedings of the 6th international conference on Web engineering*. pp. 353–360. ACM (2006)
- [5] Brambilla, M., Cabot, J., Wimmer, M.: *Model-Driven Software Engineering in Practice*. Morgan&Claypool Publishers (2012)
- [6] Casalánguida, H., Durán, J.E.: Current Trends in Web Engineering: ICWE 2013 International Workshops ComposableWeb, QWE, MDWE, DMSSW, EMotions, CSE, SSN, and PhD Symposium, Aalborg, Denmark, July 8-12, 2013. Revised Selected Papers, chap. A Method for Integrating Process Description and User Interface Use during Design of RIA Applications, pp. 172–186. Springer International Publishing (2013)
- [7] Czarnecki, K., Helsen, S.: Classification of Model Transformation Approaches. *OOPSLA* (2003)
- [8] Dannecker, L., Feldmann, M., Nestler, T., Hübsch, G., Jugel, U., Muthmann, K.: Current Trends in Web Engineering: 10th International Conference on Web Engineering ICWE 2010 Workshops, Vienna, Austria, July 2010, Revised Selected Papers, chap. Rapid Development of Composite Applications Using Annotated Web Services, pp. 1–12. Springer (2010)

- [9] García Rubio, F.O., Vara Mesa, J.M., Chicote, C.V.: Desarrollo de Software Dirigido por Modelos: Conceptos, Métodos y Herramientas. Ra-Ma (2013)
- [10] Garrigós, I., Meliá, S., Casteleyn, S.: Web Information Systems Engineering - WISE 2009: 10th International Conference, Poznań, Poland, October 5-7, 2009. Proceedings, chap. Personalizing the Interface in Rich Internet Applications, pp. 365–378. Springer (2009)
- [11] Genero Bocco, M., Cruz-Lemos, J., Piattini Velthuis, M.: Métodos de investigación en ingeniería del software. Ra-Ma (2014)
- [12] González, M., Aquino, N., Cernuzzi, L., Pastor, O.: Developing Web Applications for different architectures: The MoWebA Approach. IEEE 10th International Conference on Research Challenges in Information Science (RCIS) (Jun 2016), (in press)
- [13] González, M., Cernuzzi, L., Pastor, O.: Una aproximación para aplicaciones web: MOWEBA. Congreso Iberoamericano en "Software Engineering"(CIbSE) (2011)
- [14] González, M., Cernuzzi, L., Pastor, O.: A Navigational Role-Centric Model Oriented Web Approach - MoWebA. International Journal of Web Engineering and Technology (IJWET) 11(1) (2016), (in press)
- [15] Hermida, J.M., Meliá, S., Martínez, J.J., Montoyo, A., Gómez, J.: Current Trends in Web Engineering: ICWE 2012 International Workshops: MDWE, ComposableWeb, WeRE, QWE, and Doctoral Consortium, Berlin, Germany, July 23-27, 2012, Revised Selected Papers, chap. Developing Semantic Rich Internet Applications with the Sm4RIA Extension for OIDE, pp. 20–25. Springer (2012)
- [16] Hermida, J.M., Meliá, S., Montoyo, A., Gómez, J.: Web Information Systems Engineering – WISE 2010 Workshops: WISE 2010 International Symposium WISS, and International Workshops CI-SE, MBC, Hong Kong, China, December 12-14, 2010, Revised Selected Papers, chap. Developing Semantic Rich Internet Applications Using a Model-Driven Approach, pp. 198–211. Springer (2011)
- [17] Hermida, J.M., Meliá, S., Montoyo, A., Gómez, J.: Applying model-driven engineering to the development of rich internet applications for business intelligence. Information Systems Frontiers 15(3), 411–431 (2013)
- [18] Koch, N., Pigerl, M., Zhang, G., Morozova, T.: Web Engineering: 9th International Conference, ICWE 2009 San Sebastián, Spain, June 24-26, 2009 Proceedings, chap. Patterns for the Model-Based Development of RIAs, pp. 283–291. Springer (2009)
- [19] Link, S., Schuster, T., Hoyer, P., Abeck, S.: Focusing graphical user interfaces in model-driven software development. In: Advances in Computer-Human Interaction, 2008 First International Conference on. pp. 3–8. IEEE (Feb 2008)
- [20] Mainetti, L., Paiano, R., Pandurino, A.: Web Engineering: 12th International Conference, ICWE 2012, Berlin, Germany, July 23-27, 2012. Proceedings, chap. MIGROS: A Model-Driven Transformation Approach of the User Experience of Legacy Applications, pp. 490–493. Springer (2012)
- [21] Meliá, S., Gómez, J., Pérez, S., Díaz, O.: A model-driven development for gwt-based rich internet applications with ooh4ria. In: Web Engineering, 2008. ICWE '08. Eighth International Conference on. pp. 13–23. IEEE (July 2008)
- [22] Meliá, S., Gómez, J., Pérez, S., Díaz, O.: Facing architectural and technological variability of rich internet applications. IEEE Internet Computing PP(99), 1–1 (2010)
- [23] Meliá, S., Martínez, J.J., Mira, S., Osuna, J.A., Gómez, J.: Web Engineering: 10th International Conference, ICWE 2010, Vienna Austria, July 5-9, 2010. Proceedings, chap. An Eclipse Plug-in for Model-Driven Development of Rich Internet Applications, pp. 514–517. Springer (2010)
- [24] Mens, T., Van Gorp, P.: A Taxonomy of Model Transformation. El Seiver (2005)
- [25] Pastor, Ó., Ruiz, M., España, S.: Software and Data Technologies: 6th International Conference, ICSOFT 2011, Seville, Spain, July 18-21, 2011. Revised Selected Papers, chap. From Requirements to Code: A Full Model-Driven Development Perspective, pp. 56–70. Springer (2013)

- [26] Pérez, S., Díaz, O., Meliá, S., Gómez, J.: Facing interaction-rich rias: The orchestration model. In: Web Engineering, 2008. ICWE '08. Eighth International Conference on. pp. 24–37. IEEE (July 2008)
- [27] Pérez, S., Durao, F., Meliá, S., Dolog, P., Díaz, O.: Web Information Systems Engineering – WISE 2010 Workshops: WISE 2010 International Symposium WISS, and International Workshops CISE, MBC, Hong Kong, China, December 12-14, 2010, Revised Selected Papers, chap. RESTful, Resource-Oriented Architectures: A Model-Driven Approach, pp. 282–294. Springer (2011)
- [28] Pleuß, A., Hußmann, H.: Human-Computer Interaction. Interaction Design and Usability: 12th International Conference, HCI International 2007, Beijing, China, July 22-27, 2007, Proceedings, Part I, chap. Integrating Authoring Tools into Model-Driven Development of Interactive Multimedia Applications, pp. 1168–1177. Springer (2007)
- [29] Pons, C., Giandini, R., Pérez, G.: Desarrollo de Software dirigido por Modelos (Conceptos teóricos y su aplicación práctica). Edulp - Editorial de la Universidad de la Plata (2010)
- [30] Preciado, J.C., Linaje, M., Comai, S., Sanchez-Figueroa, F.: Designing rich internet applications with web engineering methodologies. In: Web Site Evolution, 2007. WSE 2007. 9th IEEE International Workshop on. pp. 23–30. IEEE (Oct 2007)
- [31] Preciado, J.C., Linaje, M., Morales-Chaparro, R., Sanchez-Figueroa, F., Zhang, G., Kroiß, C., Koch, N.: Designing rich internet applications combining uwe and rux-method. In: Web Engineering, 2008. ICWE '08. Eighth International Conference on. pp. 148–154. IEEE (July 2008)
- [32] Robles Luna, E., Escalona, M.J., Rossi, G.: Software and Data Technologies: 5th International Conference, ICSoft 2010, Athens, Greece, July 22-24, 2010. Revised Selected Papers, chap. Modelling the Requirements of Rich Internet Applications in WebRe, pp. 27–41. Springer (2013)
- [33] Rodríguez-Echeverría, R., Conejero, J.M., Clemente, P.J., Pavón, V.M., Sánchez-Figueroa, F.: Current Trends in Web Engineering: ICWE 2012 International Workshops: MDWE, ComposableWeb, WeRE, QWE, and Doctoral Consortium, Berlin, Germany, July 23-27, 2012, Revised Selected Papers, chap. Model Driven Extraction of the Navigational Concern of Legacy Web Applications, pp. 56–70. Springer (2012)
- [34] Rodríguez-Echeverría, R., Conejero, J.M., Clemente, P.J., Villalobos, M.D., Sánchez-Figueroa, F.: Web Engineering: 12th International Conference, ICWE 2012, Berlin, Germany, July 23-27, 2012. Proceedings, chap. Extracting Navigational Models from Struts-Based Web Applications, pp. 419–426. Springer (2012)
- [35] Rodríguez-Echeverría, R., Conejero, J.M., Linaje, M., Preciado, J.C., Sánchez-Figueroa, F.: Web Engineering: 10th International Conference, ICWE 2010, Vienna Austria, July 5-9, 2010. Proceedings, chap. Re-engineering Legacy Web Applications into Rich Internet Applications, pp. 189–203. Springer (2010)
- [36] Rodríguez-Echeverría, R., Pavón, V.M., Macías, F., Conejero, J.M., Clemente, P.J., Sánchez-Figueroa, F.: Web Information Systems Engineering – WISE 2013: 14th International Conference, Nanjing, China, October 13-15, 2013, Proceedings, Part II, chap. Generating a Conceptual Representation of a Legacy Web Application, pp. 231–240. Springer (2013)
- [37] Sosa, E., Clemente, P.J., Conejero, J.M., Rodríguez-Echeverría, R.: A model-driven process to modernize legacy web applications based on service oriented architectures. In: Web Systems Evolution (WSE), 2013 15th IEEE International Symposium on. pp. 61–70. IEEE (Sept 2013)
- [38] Torres, V., Giner, P., Pelechano, V.: Developing bp-driven web applications through the use of mde techniques. *Software & Systems Modeling* 11(4), 609–631 (2010)
- [39] Urbieto, M., Urbieto, M., Rossi, G., Rossi, G., Ginzburg, J., Ginzburg, J., Schwabe, D., Schwabe, D.: Designing the interface of rich internet applications. In: Web Conference, 2007. LA-WEB 2007. Latin American. pp. 144–153. IEEE (Oct 2007)
- [40] Valverde, F., Pastor, O.: Web Information Systems Engineering - WISE 2009: 10th International Conference, Poznań, Poland, October 5-7, 2009. Proceedings, chap. Facing the Technological Challenges of Web 2.0: A RIA Model-Driven Engineering Approach, pp. 131–144. Springer (2009)