José Alfonso Aguilar Calderón

A goal-oriented approach for managing requirements in the development of Web applications

– PhD. Thesis –

Advisor: Irene Garrigos, Jose-Norberto Mazón López

Depto. Lenguajes y Sistemas Informáticos
Universidad de Alicante

Cada uno da lo que recibe,
y luego recibe lo que da,
nada es más simple,
no hay otra norma:
nada se pierde,
todo se transforma.

Jorge Drexler, de su canción "Todo se transforma".

Tu beso se hizo calor, luego el calor, movimiento, luego gota de sudor que se hizo vapor, luego viento que en un rincón de La Rioja movió el aspa de un molino mientras se pisaba el vino que bebió tu boca roja. Tu boca roja en la mía, la copa que gira en mi mano, y mientras el vino caía supe que de algún lejano rincón de otra galaxia, el amor que me darías, transformado, volvería un día a darte las gracias. Cada uno da lo que recibe y luego recibe lo que da, nada es más simple, no hay otra norma: nada se pierde, todo se transforma.

El vino que pagué yo, con aquel euro italiano que había estado en un vagón antes de estar en mi mano, y antes de eso en Torino, y antes de Torino, en Prato, donde hicieron mi zapato sobre el que caería el vino. Zapato que en unas horas buscaré bajo tu cama con las luces de la aurora, junto a tus sandalias planas que compraste aquella vez en Salvador de Bahía, donde a otro diste el amor que hoy yo te devolvería... Cada uno da lo que recibe y luego recibe lo que da, nada es más simple, no hay otra norma: nada se pierde, todo se transforma.

Agradecimientos

Un trabajo de investigación nunca es un esfuerzo individual, sino que se realiza siempre con el apoyo y la ayuda de muchas personas, tanto en lo profesional como en lo personal. Imposible enumerarlas a todas, así que a todas ellas: gracias. Por el apoyo en el trabajo, por esa charla delante de un café, por esa ayuda con el inglés, por esa palmada en la espalda en los días malos, por esos desayunos temprano por la mañana, por esos momentos de desconexión haciendo deporte, por esas comidas diarias con *tupper* incluido, por ser el faro en la noche... Por todo.

A Inma, por convertir los malos momentos en una sonrisa. Este trabajo es tanto tuyo como mío.

A toda mi familia por hacerme sentir especial, en particular a mis padres porque todo lo que soy es gracias a ellos.

A Juan Carlos Trujillo, por todo lo compartido en este tiempo y por todo el apoyo, decisivo y fundamental, para seguir adelante.

A los que me "sufrieron" más directamente en el trabajo diario: Jesús Pardillo y Octavio Glorio, con los que he colaborado estrechamente en la realización de varios trabajos de investigación, así como en la implementación de la herramienta de modelado que se muestra en esta tesis. Por los momentos compartidos en estos años y por los momentos que quedan por venir...

A Emilio Soler, por todos estos años de "conexión cubana" y por esta recta final de trabajo intenso, codo con codo.

A Luisa Micó y Jose Manuel Iñesta porque con ellos empezó mi periodo de becario en el Departamento de Lenguajes y Sistemas Informáticos de la Universidad de Alicante.

A mis compañeros del grupo de investigación Lucentia del Departamento de Lenguajes y Sistemas Informáticos de la Universidad de Alicante, Cristina Cachero, Jose Jacobo Zubcoff, Sergio Luján, Rafael Romero y Lily Muñoz por las discusiones de trabajo y toda la colaboración que me brindan.

A los miembros del grupo de investigación IWAD del Departamento de Lenguajes y Sistemas Informáticos de la Universidad de Alicante, en especial a Jaime Gómez, Irene Garrigós y Santiago Meliá por el apoyo recibido durante este período de investigación.

A todos los miembros del grupo de investigación GPLSI del Departamento de Lenguajes y Sistemas Informáticos de la Universidad de Alicante por todos los momentos compartidos.

También quisiera expresar mi más profundo agradecimiento al Prof. Dr. Gottfried Vossen y al Dr. Jens Lechtenbörger por facilitarme la realización de una estancia de investigación en la Universidad de Münster durante el verano de 2007. Su interés en mi trabajo, sus valiosas discusiones y su apoyo han tenido gran influencia en mi investigación.

Finalmente, también quisiera agradecer al Prof. Dr. Eric Yu y al Dr. Jordi Cabot su apoyo durante mi estancia en la Universidad de Toronto durante el verano de 2008.

Acknowledgments

Research is not an individual effort, but requires the professional and personal support of many people. It is impossible to list everybody here, so thanks to all of you. Thanks for supporting my work, for that talk during the coffee break, for that assistance with the English, for that pat on the back, for those breakfasts early in the morning, for those "sport moments", for those daily *tupper* meals, for being the lighthouse in the night... Thanks for everything.

To Inma, for turning difficult times into a smile. This work is as much yours as it is mine.

To my family for making me feel special, particularly to my parents since all I am is due to them.

To Juan Carlos Trujillo, for all the shared moments and for his decisive and crucial encouragement to go on.

To Jesús Pardillo and Octavio Glorio, for their friendship, for all the discussions, for their support and for their involvement in the implementation of this research.

To Emilio Soler, during these years of "Cuban connection" and during this last sprint, working side by side.

To Luisa Micó and Jose Manuel Iñesta, for allowing me to start working in the Department of Software and Computing Systems of the University of Alicante some years ago.

To my colleagues in the Lucentia research group in the Department of Software and Computing Systems of the University of Alicante, Cristina Cachero, Jose Jacobo Zubcoff, Sergio Luján, Rafael Romero and Lily Muñoz for their assistance, fruitful discussions and support.

To all the members of the IWAD research group in the Department of Software and Computing Systems of the University of Alicante, and particularly to Jaime Gómez, Irene Garrigós and Santiago Meliá for their continued support.

To all the members of the GPLSI research group in the Department of Software and Computing Systems of the University of Alicante, for all the shared moments.

I would also like to express my deepest gratitude to Prof. Dr. Gottfried Vossen and Dr. Jens Lechtenbörger for hosting my stay at the University of Münster during the summer of 2007. Their interest in my work, valuable discussions and encouragement had a great influence on the progress of my research.

Finally, I wish to thank Prof. Dr. Eric Yu and Dr. Jordi Cabot for their support during my research period in the University of Toronto in the summer of 2008.

Preface

Designing a multidimensional model of a data warehouse is a highly complex, prone to fail, and time consuming task, due to the fact that (i) the information needs of decision makers and the available operational data sources that will populate the data warehouse must both be considered together in a conceptual multidimensional model, and (ii) summarizability-aware non-trivial mappings must be performed to obtain the final implementation of this conceptual multidimensional model. However, no significant effort has been made to take these issues into account in a systematic, well structured and comprehensive development process. To overcome the lack of such a process, this PhD Thesis proposes a model-driven approach for the development of a hybrid multidimensional model at the conceptual level and for the automatic derivation of its logical representation as a basis of implementation. A normalization process is also proposed to avoid summarizability problems at the conceptual level and in further implementation stages. Finally, an *Eclipse*-based tool has been implemented as a proof of concept of this research. This tool has been used in a case study, which shows each step of the presented approach.

This PhD Thesis is composed of a set of published and submitted papers. In order to write this PhD Thesis as a collection of papers, several requirements must be taken into account as stated by the University of Alicante. With regard to the content of the PhD Thesis, it must specifically include a summary which is devoted to the description of initial hypotheses, research objectives, and the collection of publications itself, thus justifying its coherence. It should be underlined that this summary of the PhD Thesis must also include research results and final conclusions. This summary corresponds to part I of this PhD Thesis (chapter 1 has been written in Spanish while chapter **??** is in English).

It should be mentioned that this PhD Thesis has been developed within the PhD program *"Aplicaciones de la Informática"* of the Department of Software and Computing Systems (*Departamento de Lenguajes y Sistemas Informáticos*, DLSI) of the University of Alicante. This PhD work was funded by the Spanish Ministry of Education and Science under the FPU grant AP2005-1360.

Finally, this research was developed under the following projects: METASIGN (TIN2004-00779) and ESPIA (TIN2007-67078) projects from the Spanish Ministry of Education and Science; DADASMECA (GV05/220) and DEMETER (GVPRE/2008/063) projects from the Valencia Ministry of Enterprise, University and Science (Spain); and MESSENGER (PCC-03-003-1), DADS (PBC-05-012-2) and QUASIMODO (PAC08-0157-0668) projects from the Castilla-La Mancha Ministry of Education and Science (Spain).

Alicante, October 2008 *Jose Norberto Mazón López*

Contents

Part I

Summary

1

Síntesis en Castellano

La presente tesis doctoral se ha realizado mediante la modalidad de compendio de artículos. Por tanto, este capítulo está dedicado a describir los objetivos, hipótesis y el conjunto de trabajos que forman parte de la tesis, quedando justificada su unidad temática. Cabe destacar que en este capítulo inicial también se sintetiza el contenido científico de la tesis, presentando un resumen global de los resultados obtenidos así como de las conclusiones finales. Por último, resaltar que el contenido de este capítulo ha sido escrito en castellano, mientras que el capítulo siguiente corresponde a su traducción en inglés.

1.1 Tesis Doctoral como Compendio de Artículos

Los requisitos que debe cumplir una tesis doctoral para ser realizada en la Universidad de Alicante mediante un compendio de publicaciones fueron definidos por el Pleno de la Comisión de Doctorado de fecha 2 de marzo de 2005. A continuación, se exponen aquellos directamente relacionados con el contenido de la tesis:

1. *"La tesis debe incluir una síntesis, en una de las dos lenguas oficiales de esta Comunidad Autónoma, en la que se presenten los objetivos, hipótesis, los trabajos presentados y se justifique la unidad temática."*
2. *"Esta síntesis debe incorporar un resumen global de los resultados obtenidos, de la discusión de estos resultados y de las conclusiones finales. Esta síntesis deberá dar una idea precisa del contenido de la tesis."*
3. *"Los trabajos deben ser publicados, o aceptados para la publicación, con posterioridad al inicio de los estudios de doctorado. Los artículos en periodo de revisión pueden formar parte de la tesis como apéndices del documento, que debe presentarse adjunta a los artículos publicados."*

Con el propósito de satisfacer los requisitos mencionados anteriormente, la estructura de la tesis queda constituida en tres partes. La primera parte (Parte I) consiste en una síntesis de la tesis y se encuentra dividida en dos capítulos. El primer capítulo corresponde a la síntesis en castellano (capítulo 1) y el segundo capítulo a su versión en inglés (capítulo **??**). La Parte **??** presenta el conjunto de artículos publicados que forman el núcleo principal de la tesis. Finalmente, la Parte **??** consiste en un apéndice donde se presentan dos trabajos que se encuentran actualmente en proceso de revisión.

Asímismo, es muy importante subrayar que la tesis doctoral ha sido materializada gracias al apoyo económico otorgado por el Consejo Nacional de Ciencia y Tecnología (CONACYT) México, por medio del Programa de Becas de Estudios de Posgrado en el Extranjero. Finalmente, es necesario destacar el interés y apoyo otorgado por parte de la Universidad Autónoma de Sinaloa, a través del Programa de Formación de Recursos Humanos en Áreas Estrátregicas.

1.1.1 Publicaciones Pertenecientes a la Tesis Doctoral

En este apartado se describen brevemente las cuatro publicaciones seleccionadas para que formen parte de la tesis doctoral. El criterio utilizado para la selección consistió en la relevancia y contribución científica de cada una de las publicaciones. Es decir, fueron seleccionadas los artículos publicados en revistas indexadas en JCR *Journal Citation Report* y en congresos unbicados en la clasificación CORE *Computer Research and Education.*

Capítulo ??

J.A. Aguilar, I. Garrigós, J.-N. Mazón, J. Trujillo. Web Engineering Approaches for Requirements Analysis - A Systematic Literature Review. 6th Web Information Systems and Technologies (WEBIST 2010), Vol. 2, pp. 187-190, 2010.

Este capítulo presenta los temas de estudio que dieron origen a la investigación asociada a la tesis doctoral, así como el material básico de referencia para comprender los detalles de la especificación, análisis y modelado de requisitos en ingeniería Web. Principalmente, el capítulo se enfoca en la revisión bibliográfica y el estado de la cuestión (ingeniería de requisitos en el dominio Web). Por último, presenta un análisis de las aproximaciones metodológicas más importantes en el ámbito de la ingeniería Web enfocado únicamente en aspectos como el análisis y especificación de requisitos, trazabilidad y las herramientas de soporte ofrecidas por cada una de ellas.

Capítulo ??

J.A. Aguilar, I. Garrigós, J.-N. Mazón, J. Trujillo. An MDA Approach for Goal-oriented Requirement Analysis in Web Engineering. Journal of Universal Computer Science (J.UCS), 16(17): 2475-2494 (2010).

Este capítulo describe la propuesta base de la tesis. En el capítuo anterior, se realizó una revisión sistemática de la literatura para estudiar las técnicas ingenieríles en el desarrollo de aplicaciones Web. Los resultados demuestran que la mayoría de las aproximaciones se enfocan en las etapas de análisis y diseño, por tanto, no ofrecen un soporte integral a la fase de requisitos. La aproximación descrita en este capítulo, ha tomando como sustento las carencias detectadas en el capítulo anterior para desarrollar una aproximación basada en el marco de modelado orientado a objetivos *i** y en MDA (*Model-Driven Architecture*). Con la propuesta, es posible derivar los modelos conceptuales que conforman una aplicación Web 1.0 (*Platform Independent Models*) a partir de la especificación de requisitos (*Computational Independent Model*). Lo anterior, por medio de un conjunto de transformaciones descritas de manera formal utilizando el lenguaje QVT (*Query/View/Transformation*).

Capítulo ??

J.A. Aguilar, I. Garrigós, J.-N. Mazón. Impact Analysis of Goal-Oriented Requirements in Web Engineering. The 11th International Conference on Computational Science and Its Applications (ICCSA 2011), June 20-23, 2011, Santander, Spain. Part V, Lecture Notes in Computer Science, Vol. 6786, pp. 421-436, 2011.

En capítulos anteriores se ha resaltado la importancia de la etapa de análisis y especificación de requisitos en la ingeniería Web, obligada, principalmente, por la audiencia heterogénea y por la evolución constante en las tecnologías de implementación. Estas carácteristicas particulares de las aplicaciones Web ocasionan, en la mayoría de los casos, inconsistencias entre los requisitos. Por consiguiente, es importante conocer las dependencias entre los requisitos para garantizar, en lo posible, que la aplicación Web satisfaga las necesidades y expectativas de los usuarios. Comprender y análizar las dependencias entre los requisitos le permite al diseñador brindar una mejor gestión y mantenimiento de la aplicación Web. En este capítulo se presenta un algoritmo para manejar las dependencias, entre los requisitos funcionales y los requisitos

no-funcionales de la aplicación Web, en un contexto orientado a objetivos (*goal-oriented*). Con el algoritmo, es posible comprender cuál es el impacto en los requisitos procedente de un cambio en los modelos conceptuales que conforman la aplicación Web, así como saber qué requisitos necesitan ser implementados para cumplir, en medida de lo posible, los proósitos establecidos en el análisis orientado a objetivos.

Capítulo ??

J.A. Aguilar, I. Garrigós, J.-N. Mazón. A Goal-Oriented Approach for Optimizing Non-Functional Requirements in Web Applications. The 8th th International Workshop on Web Information Systems Modeling (WISM 2011), held in conjunction with the International Conference on Conceptual Modeling (ER 2011), 31 October - 03 November 2011, Brussels, Belgium. Part X, Lecture Notes in Computer Science, Vol. X, pp. XXX-XXX, 2011.

Recientemente, la idea de considerar los requisitos no-funcionales desde las etapas iniciales (requisitos) del proceso de desarrollo con el fin de mejorar la aplicación a desarrollar ha sido objeto de investigación en el contexto del desarrollo dirigido por modelos (*Model-Driven Development, por sus siglas en inglés*). La idea se fundamenta en la implementación de los requisitos funcionales a partir de los requisitos no-funcionales. En este sentido, los requisitos no-funcionales deben de ser priorizados de acorde al contexto de los usuarios de la aplicación Web. En este capítulo se presenta una adaptación del algoritmo de Pareto (Frontera de Pareto) para evaluar y seleccionar la configuración de requisitos óptima que maximice los requisitos no-funcionales de la aplicación Web. La solución del algoritmo proporciona al diseñador de la aplicación Web un conjunto de configuraciones de entre las cuales podrá elegir qué requisitos funcionales implementar (configuración óptima) considerando la prioridad de los requisitos no-funcionales.

1.1.2 Artículos en Proceso de Revisión Pertenecientes a la Tesis Doctoral

En este apartado, se presentan dos trabajos que forman parte de la tesis doctoral pero que están actualmente bajo proceso de revisión.

Apéndice ??

Requirements in Web engineering: a systematic literature review. Este artículo se ha enviado a la revista XXXXXXXX.

En este trabajo se realiza una profunda revisión del estado de la cuestión, en lo referente al análisis, especificación y trazabilidad de requisitos en ingeniería Web. Concretamente, se analizan: (i) las técnicas utilizadas por las metodologías ingenieríles en la etapa de análisis y especificación de requisitos, (ii) el tipo de requisitos y la terminología utilizada por cada metodología Web, (iii) el soporte para trazabilidad y (iv) las herramientas de soporte que ofrecen.

Apéndice ??

A Goal-Oriented Requirements Engineering Approach to Distribute Functionality in RIAs. Este artículo se ha enviado a 12th International Conference on Web Information System Engineering (WISE 2011).

Como es sabido, la Web evoluciona constantemente y las metodologías Web deben ser adaptadas para lidiar, por ejemplo, con las tecnologías de implementación. Parte de esta evolución son las aplicaciones RIAs (*Rich Internet Applications, por sus siglas en inglés*), las cuales ofrecen, entre otras cosas, una mejor interactividad con el usuario, similar a la ofrecida por las aplicaciones *software* de escritorio. En este trabajo, se presenta la adaptación de la propuesta descrita en el capítulo **??** para auxiliar al diseñador Web en la distribución entre el cliente y el servidor de la funcionalidad de la aplicación RIA. Para lograrlo, se adaptó el algoritmo de Pareto para lograr, en lo posible, un equilibrio entre los requisitos funcionales y los no-funcionales.

1.1.3 Otras Publicaciones

En el transcurso de la investigación asociada a la tesis doctoral se han publicado (o enviado) cinco artículos a distintos eventos nacionales y/o internacionales. Los trabajos no han sido incluídos en el núcleo de la tesis, sin embargo complementan el progreso de la investigación.

J.A. Aguilar, I. Garrigós, J.-N. Mazón. Aproximaciones en Ingeniería Web para el Análisis de Requisitos: una Revisión Sistemática de la Literatura. *Actas del IV Congreso Nacional de Informática y Ciencias de la Computación (CNICC 2009), Mazatlán, Sinaloa, México, 2009. ISSN: XXXXXXX.*

J.A. Aguilar, I. Garrigós, J.-N. Mazón. Modelos de *weaving* para Trazabilidad de Requisitos Web en A-OOH. *Actas del VII Taller de Desarrollo de Software Dirigido por Modelos (DSDM 2010) en XV Jornadas de Ingeniería de Software y Bases de Datos (JISBD 2010), en conjunto con el Congreso Español de Informática (CEDI), pp. 146-155. SISTEDES, Valencia, España, 2010.*

J.A. Aguilar, I. Garrigós, J.-N. Mazón. Modelo Requisitos y Modelo de Dominio, trazabilidad mediante modelos de *Weaving. Actas de VIII Jornadas para el Desarrollo de Grandes Aplicaciones de Red (JDARE 2010). GrupoM, Alicante, España, 2010. ISBN: XXXXXXX.*

J.A. Aguilar, I. Garrigós, J.-N. Mazón. Automatic Generation of Conceptual Models from Requirements Specification in Web Engineering using ATL. *Actas de IX Jornadas para el Desarrollo de Grandes Aplicaciones de Red (JDARE 2011). GrupoM, Alicante, España, 2011. Enviado.*

J.A. Aguilar, I. Garrigós, J.-N. Mazón. Una Propuesta Orientada a Objetivos para el Análisis de Requisitos en RIAs. *Actas de XVI Jornadas de Ingeniería de Software y Bases de Datos (JISBD 2010). La Coruña, España, 2011. Enviado.*

1.2 Objetivos de Investigación e Hipótesis Inicial

De forma similar a los sistemas software desarrollados exclusivamente para un entorno de escritorio, los sistemas Web necesitan la aplicación de conceptos de ingeniería para obtener éxito en la aplicación Web final. Para lograrlo, es necesario definir técnicas y enfoques que consideren la gran variedad de usuarios, plataformas y entornos para su implementación. En este sentido, uno de los factores de éxito más importantes en el desarrollo de *software* es la elicitación, gestión y análisis de requisitos. Sin embargo, en el desarrollo de *software* en ingeniería Web llevar a cabo una correcta gestión de los requisitos es una tarea complicada. Principalmente, esto se debe a que la ingeniería Web enfrenta continuos cambios que dificultan la etapa de requisitos a razón de las carácteristicas particulares de las aplicaciones Web, como el caso de: (i) la gran cantidad de información que ofrecen (contenido), (ii) el acceso a los diferentes escenarios donde ofrecen esa información (navegación), (iii) como proveer dicha información al usuario o grupos de usuarios (funcionalidad) del sitio Web y (iv), la audiencia heterogénea que tiene acceso a la Web. Como consecuencia de estos factores, los analistas, desarrolladores y diseñadores se enfrentan a retos cada vez más complejos para gestionar el diseño y mantenimiento de las aplicaciones Web. Por lo tanto, definir los requisitos (funcionales y no-funcionales) que el sistema debe cumplir para satisfacer las necesidades de los usuarios es una tarea que necesita una atención especial.

Actualmente, existen una notable cantidad de aproximaciones metodológicas para el desarrollo de aplicaciones Web (A-OOH, UWE, NTD, OOWS, etc.) [4] que toman en cuenta la aplicación de distintas técnicas para llevar a cabo la etapa de desarrollo. Sin embargo, la mayoría de las aproximaciones contemplan técnicas de ingeniería de *software* para gestionar correctamente los requisitos de los usuarios, como el caso de UWE (casos de uso). La mayoría de las técnicas utilizadas por las metodologías resultan insuficientes para representar carácteristicas muy particulares de las aplicaciones Web, tales como: la navegación y la especificación de las necesidades de los diferentes actores implicados en la aplicación Web (audiencia heterogénea). Por tal motivo, es necesaria la inclusión de nuevas técnicas para lidiar con las carácteristicas particulares de las aplicaciones Web.

Recientemente, el desarrollo dirigido por modelos (*Model Driven Development*, MDD) se ha convertido en una alternativa valiosa para resolver los problemas asociados con el desarrollo de software de manera sistemática, estructurada, integrada y completa mediante la utilización de modelos como artefactos principales en el proceso de desarrollo de las aplicaciones Web. El desarrollo dirigido por modelos es una aproximación al desarrollo de software basado en el modelado del sistema software y su generación a partir de los modelos. Al ser únicamente una aproximación, sólo proporciona una estrategia general a seguir en el desarrollo de software, pero no define técnicas a utilizar, ni fases del proceso, ni ningún tipo de guía metodológica. El impacto de MDD en la ingeniería Web ha permitido la llegada de la ingeniería Web dirigida por modelos (*Model Driven Web Engineering*, MDWE) como una nueva aproximación para el desarrollo de aplicaciones Web. Su supuesto básico es la consideración de los modelos como entidades de primera clase que impulsan el proceso de desarrollo desde el análisis de requisitos hasta la implementación final. Básicamente, cada paso del proceso consiste en la generación de uno o más modelos de salida a partir de uno o más modelos de entrada. Por lo tanto, las transformaciones entre modelos son la clave para completar cada paso del proceso del proceso de desarrollo dirigido por modelos.

En este contexto, la arquitectura dirigida por modelos (*Model Driven Architecture*, MDA) es un estándar del OMG (*Object Management Group*) que promueve el MDD y que se ha aplicado con resultados favorables al MDWE. MDA está formada por un conjunto específico de capas y transformaciones que proporcionan un marco conceptual en donde encontramos tres tipos de modelos, el primero de ellos es modelo independiente de la computación (*Computational Independent Model*, CIM), utilizado para la especificación de los requisitos de la aplicación a desarrollar, el segundo es el modelo independiente de la plataforma (*Platform Independent Model*, PIM), como su nombre lo indica, se caracteriza por ser independiente de la plataforma de implementación de la aplicación, finalmente, el modelo específico de la plataforma (*Platform Specific Model*, PSM), el cual es obtenido del PIM y contiene la información sobre una plataforma de desarrollo o alguna tecnología en especifico donde será implementada la aplicación final, esto es, el código fuente de la aplicación [9].

Actualmente, MDA ha sido aplicado para el desarrollo de aplicaciones Web, tal es el caso de NDT y UWE (CITA). Sin embargo, el trabajo presentado en la tesis doctoral es el primero que aborda el modelado conceptual de aplicaciones Web a partir del nivel CIM de MDA utilizando técnicas orientadas a objetivos para la obtención automática de modelos conceptuales a nivel PIM asegurando que sean correctos semánticamente.

El **objetivo de investigación** de esta tesis doctoral es la propuesta de una metodología para el análisis de requisitos para aplicaciones Web 1.0 que considere:

- Una etapa de análisis de requisitos orientada a objetivos para representar las expectativas reales del usuario de la aplicación Web.
- Mecanismos para la comprensión de los objetivos de negocio que debe lograr la aplicación Web gracias al uso del análisis de requisitos orientada a objetivos.
- Soporte para la gestión de los requisitos en aspectos como la trazabilidad y el análisis de impacto.
- Considerar los requisitos no-funcionales desde la etapa de análisis y especificación de requisitos.
- Asistir al diseñador al momento de la selección de los requisitos funcionales a implementar a través de alternativas de diseño que consideren el balance y maximización de los requisitos no-funcionales.
- Semi-automatizar el desarrollo de aplicaciones Web por medio de un conjunto de transformaciones formales para obtener los modelos conceptuales a partir de la especificación de los requisitos.

Cabe destacar que la hipótesis de partida de la investigación asociada a la tesis doctoral consiste en que si es factible o no el desarrollo de una metodología MDD-MDA que contemple una etapa de análisis de requisitos que permita comprender los objetivos y expectativas reales de la audiencia heterogénea de una aplicación Web.

1.3 Resumen del Contenido de la Tesis Doctoral

El objetivo de investigación de esta tesis doctoral se aborda en dos etapas, la primera es la definición de una propuesta orientada a objetivos para el análisis y especificación de requisitos en ingeniería Web. La finalidad de la primera etapa es la obtención de los modelos conceptuales de la aplicación Web (dominio y navegación). La segunda consiste en la especificación y aplicación de técnicas para la gestión de requisitos, concretamente, aquellas relacionadas a la trazabilidad de requisitos, análisis de impacto y maximización de requisitos no-funcionales.

1.3.1 Una Aproximación Orientada a Objetivos para el Análisis de Requisitos en Ingeniería Web

El modelado MD de ADs debería tener en cuenta tanto los requisitos de usuario como las fuentes de datos desde las etapas tempranas de desarrollo. Además, el proceso de diseño debería establecer un conjunto de transformaciones formales para obtener automáticamente la implementación final del modelo MD. Para tratar estas cuestiones, en esta tesis doctoral, se describe una propuesta dirigida por modelos[1] la cual (i) combina las estrategias dirigida por datos y dirigida por requisitos en una propuesta híbrida de tal manera que el AD cubra las necesidades de los usuarios a la vez que concuerde con las fuentes de datos, y (ii) contiene un repositorio de transformaciones formales donde se incluye el conocimiento sobre cómo obtener automáticamente una representación lógica apropiada a partir del modelo MD conceptual de tal manera que los diseñadores puedan ahorrar tiempo y esfuerzo en la implementación del AD.

Se ha definido cada una de las partes de la propuesta global de modelado MD (ver Fig. **??**) de manera separada durante el desarrollo de esta tesis doctoral [**?**, **?**, **?**, **?**, **?**, **?**], siendo el trabajo descrito en [**?**] un punto de partida de esta investigación. La novedad de esta propuesta reside en considerar un punto de vista híbrido para el modelado MD de manera completa, sistemática y bien estructurada, a la vez que se definen un conjunto de transformaciones formales que apoyan al diseñador en la obtención automática de la implementación del modelo MD. Esta propuesta se basa en el *Model Driven Architecture* (MDA) [**?**] especificada por el *Object Management Group* (OMG) como un estándar para llevar a cabo el MDD. Tal y como se muestra en la Fig. **??**, se define un modelo MD conceptual del AD (*Platform Independent Model*, PIM) a partir de un modelo de requisitos (*Computation Independent Model*, CIM) que se obtiene de los usuarios del AD [**?**]. Este PIM inicial se debe reconciliar con las fuentes de datos [**?**, **?**], obteniendo un PIM híbrido. Además, a partir de este PIM híbrido se pueden derivar varios modelos lógicos (*Platform Specific Models, PSMs)* considerando diferentes plataformas de implementación (relacional, multidimensional, etc.). Finalmente, el código para la implementación del modelo MD se obtiene a partir de cada PSM. Cabe destacar que se ha desarrollado una herramienta basada en *Eclipse* como prueba de concepto de esta investigación.

A continuación, se resume cada una de las partes de la propuesta basada en MDA para el diseño MD. Se remite al lector a los capítulos específicos de esta tesis doctoral para una explicación más detallada.

CIM Multidimensional

El primer paso de la propuesta presentada en esta tesis doctoral es la obtención y el modelado de los requisitos de información de los usuarios. Esto se describe con más detalle en el *capítulo* **??**.

Se necesita una fase explícita de análisis de requisitos para modelar las necesidades de información de los usuarios y derivar un modelo MD conceptual que las satisfaga plenamente, incrementando el éxito de un proyecto de desarrollo de un AD. Los usuarios de un AD ignoran

[1] Aunque este trabajo de investigación se centra en describir una propuesta para el modelado MD del AD, dicha propuesta se ha contextualizado dentro de un marco de trabajo global donde se considera cada una de las partes del AD: procesos ETL, herramientas de análisis de datos, etc. (ver capítulo **??**).

frecuentemente cómo describir apropiadamente los requisitos de información, ya que dichos usuarios son más conscientes de los objetivos de alto nivel que el AD ayuda a cumplir. Por tanto, una fase de análisis de requisitos para ADs debe comenzar descubriendo los objetivos de los usuarios. Los requisitos de información se descubrirán más fácilmente a partir de estos objetivos.

Los objetivos relacionados con el AD se especifican a tres niveles: *Objetivos estratégicos*, los cuales representan los principales objetivos de un proceso de negocio: "incrementar ventas", "incrementar el número de clientes", "decrementar el coste", etc. *Objetivos decisionales*, que permiten definir las acciones que se deben tomar para cumplir con un objetivo estratégico, por ejemplo, "definir algún tipo de promoción" o"abrir nuevas tiendas". Finalmente, *Objetivos informacionales* que se relacionan con la información que requiere un objetivo decisional para poder cumplirse, como por ejemplo "analizar compras de clientes" o "examinar niveles de inventario". Una vez definidos estos objetivos, se pueden obtener los requisitos de información directamente de los objetivos informacionales. Los diferentes elementos MD que aparecerán en el modelo MD conceptual del AD, tales como *hechos* o *dimensiones*, podrán descubrirse a partir de los requisitos de información.

Con el fin de modelar esta jerarquía de objetivos y los correspondientes requisitos de información, se ha extendido el marco de modelado i^* [**?**] para el dominio del AD. Este marco de modelado define mecanismos con los que representar los diferentes actores, sus dependencias y los diferentes objetivos que se desean alcanzar. Específicamente, se ha usado UML (*Unified Modeling Language*) [**?**] con el fin de poder modelar objetivos y requisitos de información en un CIM mediante la definición de (i) un *profile* UML para i^* y (ii) un *profile* UML que adapta i^* al dominio del AD. Ambos *profiles* se describen en [**?**] y en la sección 1.3.1.

PIM Multidimensional Inicial

Una vez que los requisitos de información se han especificado en un CIM, debe derivarse un modelo MD conceptual como un PIM. Para ello se han definido varias reglas de transformación QVT (*Query/View/Transformation*) [**?**], de tal manera que se obtiene un PIM inicial que contiene los elementos MD necesarios para suministrar la información requerida por los usuarios del AD. Estas reglas QVT aseguran la trazabilidad entre objetivos y requisitos de información en el CIM y elementos MD en el PIM. Esta parte de la propuesta se describe en el *capítulo* **??** y en [**?**].

La definición de este PIM se basa en un *profile* UML para el modelado MD conceptual presentado en [**?**] (ver Sect. 1.3.1).

PIM Multidimensional Híbrido

Tal y como se ha descrito anteriormente, el PIM inicial se deriva directamente del CIM, por lo que asegura que el AD será útil para cumplir con los objetivos de los usuarios. Sin embargo, ese PIM inicial se define independientemente de las fuentes de datos operacionales y puede que no concuerde con estas fuentes debido a que los usuarios tienen una visión limitada de ellas. Debido a esto, el PIM inicial podría no ser *fidedigno* (quizás no pueda poblarse con las fuentes de datos existentes) ni *completo* (quizás no capture la potencia de análisis suministrada por las fuentes de datos). Por lo tanto, si se pretende evitar estos errores, entonces este PIM inicial debe reconciliarse con las fuentes de datos disponibles. Esta parte de la propuesta se describe en los *capítulos* **??** *y* **??**.

Existen varias formas normales multidimensionales [**?**] para razonar de manera rigurosa acerca de varias propiedades deseable de un modelo MD conceptual derivado de las fuentes de datos (entre otras el que sea fidedigno y completo). Por consiguiente, en el marco de esta tesis doctoral se ha desarrollado un conjunto de relaciones QVT basadas en estas formas normales [**?**] para asegurar que el PIM inicial sea fidedigno y completo con respecto a las fuentes de datos, obteniendo así un PIM híbrido.

El enfoque adoptado para la obtención de un PIM híbrido se compone de dos fases principales. La primera se basa en considerar el diseño de un modelo MD como una tarea de

modernización del software [**?**]. El objetivo de esta tarea es enlazar los elementos de las fuentes de datos con conceptos MDs [**?**]. Debido a que las fuentes de datos operacionales son verdaderos sistemas heredados, la documentación no está disponible de manera general, no puede obtenerse o es demasiado compleja para comprenderse [**?**]. Por tanto, esta primera fase comienza usando mecanismos de ingeniería inversa de datos para obtener una representación lógica de las fuentes de datos. Una vez hecho esto, se aplican una serie de reglas QVT para identificar conceptos MDs (hecho, dimensión, medida, etc.) en la representación lógica, obteniendo así un modelo lógico marcado.

La segunda fase consiste en la reconciliación del PIM inicial con el modelo lógico de las fuentes de datos marcado con conceptos MD, para la obtención de un PIM híbrido, mediante el uso de un conjunto de relaciones QVT basadas en las formas normales multidimensionales (propuestas en [**?**, **?**]). Estas relaciones se basan en la detección de dependencias funcionales (FDs) en el PIM inicial y en las fuentes de datos. Concretamente, para que un modelo sea *fidedigno* se debe asegurar que las FDs que se encuentran en el PIM inicial sean un subconjunto de aquellas que se observan en las fuentes de datos (de otra manera, el modelo MD representaría estructuras que no podrían ser pobladas con los datos existentes), mientras que para que un modelo sea *completo* se debe asegurar que las FDs entre niveles de dimensión que aparecen en las fuentes de datos estén representadas en el PIM y que las FDs entre conjuntos de medidas de las fuentes de datos estén representadas en el PIM mediante formulas de derivación (de otra manera se perdería potencial de análisis en el modelo MD). Además, las formas normales multidimensionales aseguran que cada medida se asigne a un hecho al nivel de detalle "correcto" (sin redundancias). El conjunto de relaciones QVT definidas se basa en formas normales multidimensionales para forzar estas propiedades mediante la eliminación, borrado o modificación de elementos en el PIM inicial, obteniendo de esta manera el PIM híbrido.

El gran beneficio de este PIM híbrido radica en que representa fielmente las fuentes de datos, manteniendo totalmente su potencial de análisis, mientras simultáneamente se capturan los requisitos de información de usuario.

PSM Multidimensional

Un PSM representa el modelo del mismo sistema especificado por el PIM, pero capturando también cómo el sistema hace uso de una plataforma o tecnología específica. En el modelado MD, "específico de plataforma" significa que el PSM se diseña para un tipo determinado de tecnología de bases de datos: *tecnología relacional* (representación de estructuras de datos MDs mediante el uso de bases de datos relacionales) tal y como se describe en el *capítulo* **??**, *tecnología multidimensional* (representación directa de los datos en estructuras MDs) tal y como se muestra en el *capítulo* **??** o cualquier otra tecnología.

En la propuesta de esta tesis doctoral, se han definido una serie de transformaciones QVT para poder obtener cada tipo de PSM. En concreto, cada PSM se ajusta con un metamodelo concreto de CWM (*Common Warehouse Metamodel*) [**?**]. De manera resumida, CWM es una definición de metamodelos para el intercambio de especificaciones de ADs entre diferentes plataformas o herramientas. Estos metamodelos son lo suficientemente completos para modelar cada parte del AD, incluyendo fuentes de datos, procesos ETL, modelado MD, implementación relacional del AD, etc. Además, son representaciones genéricas y estándar de metadatos con el fin de asegurar su intercambio entre diferentes plataformas y herramientas. Cabe destacar que el uso de CWM es un requisito deseable para la gestión de metadatos en escenarios de inteligencia de negocio [**?**].

Finalmente, se han desarrollado un conjunto de transformaciones *Models to Text* (Mof2-Text) [**?**] para obtener el código de cada PSM. Por ejemplo, un PSM basado en tecnología relacional derivaría en código SQL. Además, como cada metamodelo CWM está relacionado íntimamente con una tecnología, derivar el código correspondiente es una tarea sencilla, la cual se trata sólo en la Sect. 1.3.1 de esta tesis doctoral.

Implementación

Después de probar varias plataformas, tales como *Rational* [**?**] o *Borland Together Architecture* [**?**], la propuesta definida en esta tesis doctoral se ha implementado en la plataforma de desarrollo *Eclipse* [**?**]. *Eclipse* puede extenderse por medio de *plugins* con el fin de añadir más características y nuevas funcionalidades. Se ha desarrollado un *plugin* que da soporte a cada parte de la propuesta [**?**]. Este nuevo plugin contiene los siguientes módulos:

Módulo CIM. Este módulo implementa el *profile* UML para usar i^* en el dominio de los ADs. Usando este módulo se implementa un CIM MD. El marco de modelado i^* suministra mecanismos con los que poder representar varios actores, sus dependencias y estructurar los objetivos que se pretenden alcanzar. Los principales elementos de i^* se describen en la Tab. 1.1. El *profile* de i^* para ADs (ver Fig. **??**) se ha desarrollado mediante la extensión de estos elementos por medio de nuevos estereotipos (los iconos correspondientes se muestran en la Fig. **??**).

Table 1.1. Principales elementos para el modelado en i^*

Elemento	Descripción
Actor	Es una entidad que lleva a cabo acciones para cumplir con sus objetivos. Se relaciona con varios elementos intencionales (objetivo, tarea o recurso).
Goal	Representa una condición o estado que el actor le gustaría alcanzar. En el contexto del AD, los objetivos pueden ser estratégicos, decisionales e informacionales.
Task	Representa una manera particular de hacer algo. En el contexto del AD, una tarea se relaciona con la manera en la cual se obtienen los datos (requisito de información).
Resource	Es una entidad que debe estar disponible para su uso. En el contexto del AD, un recurso se relaciona con un ítem de información, p.e. una medida.
Means-ends	Son asociaciones que describen cómo se alcanzan los objetivos al representar cuales son los elementos necesarios para su cumplimiento.
Decomposition	Son asociaciones que definen elementos adicionales necesarios para llevar a cabo una tarea.

Los objetivos de los usuarios del AD se definen mediante el uso de los estereotipos *Strategic*, *Decision* e *Information*. Los requisitos de información se derivan de los objetivos informacionales y se representan como tareas estereotipadas como *Requirement*. Además, el análisis de requisitos para ADs necesita del concurso de ciertos conceptos MDs (tal y como se describe en [**?**]). Por tanto, los siguientes conceptos se pueden añadir al CIM como recursos estereotipados: el proceso de negocio relacionado con los objetivos de los usuarios (estereotipo *BusinessProcess*), medidas relacionadas con los requisitos de información (*Measure*) y los contextos para el análisis de esas medidas (*Context*). De manera adicional, se pueden prever y modelar relaciones entre los contextos de análisis. Por ejemplo, los contextos ciudad y país deberían estar relacionados porque las ciudades pueden agregarse en países. Para modelar estas relaciones se usa la relación de agregación (compartida) de UML.

Módulo PIM. Este módulo implementa el *profile* de UML para modelado MD que permite definir un modelo MD conceptual en un PIM.
La definición del PIM se basa en el *profile* de UML para modelado MD conceptual presentado en [**?**]. Este *profile* contiene los estereotipos necesarios para representar propiedades MDs (ver Fig. **??**) a nivel conceptual mediante un diagrama de clases UML. Los principales elementos de este *profile* se describen en la Tab. 1.2 (los iconos utilizados se muestran en la Fig. **??**). Este *profile* se ha definido formalmente y posee una serie de restricciones

OCL (*Object Constraint Language*) [**?**] para expresar reglas bien formadas de los nuevos elementos, evitando su uso arbitrario.

Table 1.2. Principales estereotipos del *profile* UML para modelado MD de ADs

Estereotipo	Extiende	Descripción
Fact	Class	Las clases con este estereotipo representan hechos de un modelo MD, los cuales consisten en medidas (los valores a analizar).
Dimension	Class	Representan dimensiones en un modelo MD, las cuales contienen niveles de jerarquía.
Base	Class	Son niveles de una jerarquía de dimensión y se componen de atributos de dimensión.
FactAttribute	Property	Son atributos del hecho, es decir, medidas. Pueden representar medidas derivadas mediante una regla de derivación.
DimensionAttribute	Property	Representan información descriptiva de un nivel de jerarquía.
Descriptor	Property	Representan atributos descriptores de un nivel de jerarquía.
Rolls-UpTo	Association	Son relaciones entre dos clases *Base*. El rol r representa la dirección en la que la jerarquía se agrega (*rolls-up*) y el rol d representa la dirección en la cual se desagrega (*drills-down*).

Módulo PSM. Este módulo implementa la capa *Resource* de CWM con el fin de definir modelos lógicos para el AD. Esta capa consiste en una serie de metamodelos estándar que permiten representar la estructura de los datos según diferentes tecnologías. Por ejemplo, el metamodelo *relacional* (ver Fig. **??**) contiene clases y asociaciones que representan cada aspecto de una base de datos relacional: tabla, columna, clave primaria, clave ajena, etc.

Módulo QVT. Tras probar varios motores de transformaciones para implementar las reglas QVT definidas (como *mediniQVT* o *smartQVT*), finalmente se eligió el motor del *ATLAS Transformation Language* (ATL) [**?**]. Por tanto, este módulo aprovecha este motor para implementar y ejecutar todas las transformaciones definidas como parte de la propuesta guiada por modelos.

Módulo Code. Este módulo utiliza un motor de transformaciones llamado *MOFScript* [**?**] para la implementación del conjunto de transformaciones Mof2Text y su posterior ejecución para derivar el código de cada PSM.

Dentro de cada módulo se ha definido diferentes editores textuales y gráficos para crear una herramienta con la que diseñar los diferentes modelos y aplicar las transformaciones QVT y Mof2Text de manera integrada. La Fig. **??** muestra una visión general de la herramienta[2]. Los diferentes modelos (CIM, PIM y PSM) y el código se guardan en diferentes carpetas cuando se crea un proyecto (Fig. **??**). Se han creado también un par de paletas para poder dibujar los diferentes elementos de cada uno de los *profiles* definidos (Fig. **??**): una paleta para el *profile* UML de *i** en el dominio de los ADs (Fig. **??**) y otra para el *profile* UML para modelado MD (Fig. **??**). Además, cada una de las transformaciones puede ejecutarse mediante el uso de cada una de las opciones del menú *"Transform"* (Fig. **??**).

A continuación, con el fin de ejemplificar el uso de esta herramienta, se presenta un caso de estudio basado en el Plan Estratégico de Formación de la Universidad de Alicante (`http://www.ua.es/es/presentacion/pe/psec/formacion/index.html`). Este plan determina los ejes, los objetivos y las acciones necesarias para articular una oferta formativa de

[2] A partir de aquí y en aras de facilitar una mejor explicación del entorno *Eclipse*, sólo se muestran partes detalladas de cada captura de pantalla.

calidad. En concreto, tras el análisis de este plan, el caso de estudio se centra en el desarrollo de un AD que apoye la toma de decisiones en el proceso de evaluación (*"Assessment"*) en la Universidad de Alicante. En este proceso está implicado un actor principal, el *"education manager"*, mediante el objetivo estratégico *"provide a good education program"*. A partir de este objetivo se obtienen tres objetivos decisionales diferentes: *"adapt education program to demand"*, *"achieve international recognition"* y *"having renowned program"*. Los objetivos informacionales derivados de estos objetivos decisionales son: *"evaluate environment demand"*, *"analyze international impact"* y *"study student performance"*. A partir de estos objetivos informacionales se pueden obtener los requisitos de información como tareas: *"percentage of students per city and province"*, *"percentage of foreign students"*, *"average of passed examination sessions by subject, degree and department"*. Una vez hecho esto, se debe determinar las medidas y los contextos de análisis y asociarlos a los requisitos de información como recursos: la única medida es *"examination session"* y los elementos que representan contextos de análisis son *"student"*, *"city"*, *"province"*, *"country"*, *"subject"*, *"degree"* y *"department"*. Parte de estos contextos de análisis (*"student"* y *"subject"*) pueden agregarse, por lo que se relacionan con otros contextos.

Cada uno de estos elementos se define en un CIM mediante el *profile* UML de *i** (Fig. **??**).

De manera resumida, para definir apropiadamente un CIM con *i** se deben realizar los siguientes pasos: (i) descubrir los actores (usuarios del AD), (ii) descubrir sus objetivos (estratégicos, decisionales e informacionales), (iii) derivar requisitos de información de los objetivos informacionales y (iv) obtener las medidas y el contexto de análisis relacionados con los requisitos de información.

Con el fin de automatizar el paso del CIM al PIM, se ha desarrollado una serie de reglas de transformación QVT [**?**]. Esta transformación tiene como entrada el CIM y crea como salida un PIM con los elementos MDs correspondientes (tal y como se muestra en la Fig. **??**): se crea una clase *Fact Assessment* con un *FactAttribute* llamado *ExaminationSession* (Fig. **??**) y, además, se crean dos clases *Dimension* y sus jerarquías de clases *Base* según los contextos de análisis definidos en el CIM (ver Fig. **??** y Fig. **??**).

El próximo paso es la obtención de un modelo de fuentes de datos y su marcado con conceptos MDs (tales como hecho, dimensión, etc.). En este caso de estudio existe una implementación de las fuentes de datos en *Oracle*. El proceso de derivación de un modelo relacional a partir del diccionario de datos de *Oracle* se ha implementado mediante *Java* dentro del entorno *Eclipse*. En concreto se ha usado la interfaz `java.sql.Connection` para realizar la conexión a la base de datos *Oracle* y ejecutar las sentencias SQL requeridas para obtener los metadatos del diccionario. Después de obtener los metadatos necesarios, se deriva el modelo correspondiente mediante el uso de *Eclipse* y el metamodelo relacional de CWM. Una vez que se tiene este modelo, se marca cada uno de sus elementos con conceptos MDs. La figura **??** muestra el modelo de las fuentes de datos del caso de estudio.

Una vez que se tiene el modelo de las fuentes y el PIM inicial, se debe proceder a realizar su reconciliación. Esta reconciliación se ha implementado en tres pasos siguiendo las formas normales multidimensionales. Primero, con el fin de asegurar que el PIM es fidedigno, para cada una de sus dependencias funcionales (FD) se comprueba que existe una FD equivalente en el modelo de las fuentes de datos, es decir, las FDs del modelo MD deben ser un subconjunto de aquellas observadas en las fuentes de datos. La opción *"Required Annotation"* (ver Fig. **??**) ejecuta una serie de reglas para hallar las FDs del PIM y comprobar que esas mismas FDs ocurren en el modelo de las fuentes de datos. Si esta comprobación falla, entonces el estado (*status*) de los elementos involucrados se etiqueta como *required* y se colorean en *rojo* para indicar que estos elementos aparecen en el PIM inicial pero no tienen equivalente en el modelo de fuentes de datos (es decir, son requeridos por el usuario pero las fuentes de datos no los suministran). Por ejemplo, en la Fig. **??**, *ExaminationSession*, *Country* y la asociación *RollsupTo* entre *Degree* y *Department* están en color rojo porque no existen elementos equivalentes en el modelo de fuentes de datos.

Una vez se comprueba que el modelo es fidedigno mediante la anotación de elementos requeridos, el segundo paso es asegurar la completitud del modelo. Se debe realizar una comprobación de las siguientes condiciones:

- Las FDs entre niveles de dimensión que aparezcan en las fuentes de datos deben representarse como asociaciones *Rolls-upTo* en el modelo MD. De lo contrario, el potencial de análisis se pierde (*completitud de agregación*).
- Las FDs entre conjuntos de medidas que se encuentran en las fuentes de datos deben representarse mediante fórmulas de derivación. De otra manera, se perderían estas relaciones en el modelo MD (*completitud de derivación*).
- Cada medida (*FactAttribute*) se debe asignar a un hecho (clase *Fact*) de tal manera que los niveles terminales de dimensión (clase *Dimension*) determinen funcionalmente y sin dependencias transitivas a la medida. De otro modo, la medida se guarda de manera redundante en un nivel de detalle "erróneo" (*eliminación de redundancias*).

Estas condiciones se comprueban mediante la ejecución de la transformación *"Supplied Annotation"* (ver Fig. **??**), la cual comprueba que las FDs del modelo de fuentes de datos tienen su equivalente en el PIM. Cuando esta comprobación falla, el estado (*status*) de los elementos involucrados se etiqueta como *supplied* y se colorean de *azul* para indicar que estos elementos aparecen en las fuentes de datos pero no en el PIM inicial (es decir, son suministrados por las fuentes de datos pero el usuario no los ha tenido en cuenta, bien porque no los necesitaba o bien porque no sabía de su existencia). Por ejemplo, en la Fig. **??** y en la Fig. **??**, se muestran nuevos elementos en azul que forman parte de las fuentes de datos pero no aparecen en el PIM inicial.

Hasta ahora cada elemento del PIM inicial se ha etiquetado con cierto estado (*status*) como *required* o *supplied*, o bien se ha dejado sin etiquetar (estado *none*). La tarea del diseñador es comprobar el modelo resultante y cambiar el estado de los elementos a *none* (ver Fig. **??**) si (i) un elemento requerido (*required*) puede ser suministrado por alguna fuente de datos externa o (ii) un elemento suministrado (*supplied*) puede ser útil para el usuario. Obviamente, en el siguiente paso, sólo se tienen en cuenta los elementos cuyo estado es *none* para poder derivar el PIM híbrido (ver Fig. **??** y Fig. **??**).

Una vez que se obtiene el PIM híbrido, el siguiente paso es obtener un PSM acorde a una tecnología específica. En este ejemplo, se deriva un PSM según la representación lógica más común de un modelo MD: el *esquema estrella* [**?**]. Éste es un esquema relacional que consiste en una tabla de hechos central con una clave primaria compuesta cuyos elementos forman una clave ajena a cada una de las tablas de dimensión (las cuales poseen una clave primaria única). Este esquema se muestra en la Fig. **??**.

Finalmente, se obtiene el código que implementa el modelo MD en una herramienta comercial. El código SQL para el esquema estrella se obtiene del PSM (Fig. **??**), mientras que del PIM híbrido se obtiene directamente el código que se usará en una herramienta de análisis de datos (Fig. **??**).

1.3.2 Resolución de Problemas de Sumarizabilidad

Un modelo MD conceptual (es decir, un PIM según la propuesta MDA presentada anteriormente) proporciona un alto nivel de abstracción para describir de manera exacta y expresiva situaciones reales. Hasta ahora se ha mostrado como, una vez que se diseña este modelo, se deriva la representación lógica correspondiente como base de la implementación del AD según una tecnología específica.

Sin embargo, aunque se diseñe un buen modelo MD conceptual, existe un intervalo semántico entre este modelo y su representación lógica que complica un adecuado tratamiento de la *sumarizabilidad*, lo que puede conducir a la obtención de resultados erróneos en las herramientas de análisis de datos. Las investigaciones llevadas a cabo hasta la fecha sólo plantean soluciones parciales, usando diferentes tipos de terminologías que dificultan el progreso en este área.

Por tanto, la propuesta dirigida por modelos para el diseño MD presentada en esta tesis doctoral se ha extendido mediante la inclusión de varias transformaciones que tienen en cuenta los problemas de sumarizabilidad en el paso del PIM al PSM. Esta parte de la tesis se describe en profundidad en los apéndices **??**, **??** y **??**.

Con el fin de motivar esta parte de la investigación, se ha realizado una revisión del estado de la cuestión (ver apéndice **??**), aportando una terminología unificada, para determinar (i) los puntos fuertes y débiles de las propuestas actuales en cuanto al modelado MD de estructuras complejas a nivel conceptual y (ii) los mecanismos existentes para evitar problemas de sumarizabilidad cuando se implementan modelos MDs conceptuales. Esta revisión sugiere que la sumarizabilidad se debe asegurar mediante un proceso global apoyado por una herramienta de diseño que permita:

1. La representación adecuada de las interacciones entre dimensiones y hechos [**?**].
2. La representación adecuada de las relaciones entre niveles de agregación en una jerarquía de dimensión. [**?**].

La noción de *sumarizabilidad* fue descrita por primera vez por Rafanelli y Shoshani [**?**] para bases de datos estadísticas, donde se refiere al cálculo correcto de valores agregados a bajo nivel de detalle a partir de valores de más alto nivel. Aunque este trabajo sólo trata las relaciones entre dos niveles de jerarquía de dimensión, las relaciones entre hecho y dimensión también pueden causar problemas de sumarizabilidad en un modelo MD [**?**]. Por tanto, para poder ser implementado, un modelo MD debe asegurar la sumarizabilidad en estos dos tipos de estructuras MD. De lo contrario, su violación puede derivar en resultados incorrectos y decisiones erróneas [**?**].

En vista de estas complicaciones, la manera tradicional de proceder es implementar un modelo MD sin incluir aquellas estructuras que pueden causar problemas de sumarizabilidad. Sin embargo, esta propuesta es muy simple y dificulta la tarea de los diseñadores que no pueden usar toda la expresividad de un modelo conceptual. Por tanto, los diseñadores desperdician mucho esfuerzo en implementar un modelo MD con constructores MD poco expresivos, debiendo ser muy cuidadosos para obtener una representación fidedigna de escenarios reales.

Por el contrario, según algunos autores [**?**, **?**, **?**, **?**, **?**] el modelado MD pretende representar cada elemento MD en un modelo conceptual independiente de la implementación con el fin de reflejar situaciones reales lo más precisamente posible. Una vez que se diseña un modelo MD conceptual, se deriva su correspondiente representación lógica como base de su implementación en una tecnología concreta. Sin embargo, la diferencia de expresividad entre los elementos representados en los modelos MD conceptuales y su representación lógica debe reducirse para preservar toda la información capturada por las estructuras MD complejas, mientras se resuelven los problemas de sumarizabilidad [**?**, **?**]. Además, derivar una representación lógica de un modelo MD conceptual es una tarea tediosa y sensible a errores por lo que debe ser automatizada tanto como sea posible.

Hasta ahora existen pocas propuestas que hayan considerado las cuestiones mencionadas anteriormente, de manera conjunta, para modelar cada tipo de jerarquía y relación entre hecho

y dimensión a nivel conceptual y, automáticamente, derivar una representación lógica que preserve la información definida a nivel conceptual evitando problemas de sumarizabilidad [**?**, **?**, **?**]. Además, las propuestas presentadas hasta la fecha tienen los siguientes inconvenientes: (i) sólo definen mecanismos informales para evitar problemas de sumarizabilidad, lo que dificulta soluciones automáticas o (ii) requieren de transformaciones a nivel de instancia (y no a nivel de esquema) para evitar problemas de sumarizabilidad por lo que se requiere de un preprocesado complejo que deriva en problemas de rendimiento y se producen instancias de datos "artificiales" difíciles de interpretar durante el análisis.

Para paliar estos problemas, se ha extendido la propuesta MDA previamente presentada con un *proceso de normalización*. Esencialmente, tal y como se muestra en la Fig. **??**, una vez que se define un PIM (el cual permite especificar estructuras MD complejas), se obtiene un PIM normalizado por medio de la ejecución automática de varias transformaciones QVT [**?**]. Este PIM normalizado captura la información representada en el modelo MD conceptual usando solamente aquel subconjunto de elementos que no viola la sumarizabilidad. A partir del PIM normalizado se puede derivar un PSM.

El proceso de normalización ha sido implementado junto con la herramienta basada en *Eclipse* presentada anteriormente. Se ha definido un *plugin* con todas las transformaciones QVT necesarias (implementadas con el motor de ATL [**?**]) para ejecutar el proceso de normalización de manera automática. En la Fig. **??** se muestra una captura de pantalla.

1.4 Conclusiones

Un AD es una colección integrada de datos históricos en apoyo a la toma de decisiones. Según esta definición, en el desarrollo de un modelo MD para un AD no es sólo importante considerar las necesidades de información de los usuarios (propuestas guiadas por requisitos), sino también las fuentes de datos existentes que poblarán el AD (propuestas guiadas por datos). Por tanto, se requiere de mecanismos formales para integrar estos dos puntos de vista en una propuesta híbrida. Además, el modelado MD del AD se asemeja a los métodos de diseño de bases de datos tradicionales [**?**] en cuanto a que debe estructurarse en varios pasos durante los cuales se desarrolla una fase de diseño conceptual, cuyos resultados se transforman en un modelo de datos lógico como base de la implementación del esquema. Esta manera de proceder posibilita la automatización de las transformaciones entre estas fases.

Para tratar con estas cuestiones, en esta tesis doctoral, se ha presentado una propuesta dirigida por modelos que permite (i) la especificación de un modelo MD híbrido a nivel conceptual de manera integral, sistemática y bien estructurada y (ii) la derivación automática de su representación lógica. Por lo tanto, esta propuesta permite a los diseñadores decrementar la complejidad del desarrollo de un AD, ahorrando tiempo y esfuerzo. Posteriormente, se ha añadido a esta propuesta dirigida por modelos un proceso de normalización con el fin de asegurar la sumarizabilidad de las estructuras MD complejas, como las jerarquías de dimensión y las relaciones hecho-dimensión. También, se ha desarrollado una herramienta basada en *Eclipse* que apoya cada parte de esta propuesta.

Finalmente, cabe destacar que esta tesis doctoral representa la primera propuesta de un proceso híbrido para el desarrollo del AD teniendo en cuenta requisitos de información y fuentes de datos, a la vez que se evitan los problemas de sumarizabilidad.