An MDA Approach and QVT Transformations for the Integrated Development of Goal-Oriented Data Warehouses and Data Marts

Jesús Pardillo, University of Alicante, Spain
Jose-Norberto Mazón, University of Alicante, Spain
Juan Trujillo, University of Alicante, Spain

ABSTRACT

To customize a data warehouse, many organizations develop concrete data marts focused on a particular department or business process. However, the integrated development of these data marts is an open problem for many organizations due to the technical and organizational challenges involved during the design of these repositories as a complete solution. In this article, the authors present a design approach that employs user requirements to build both corporate data warehouses and data marts in an integrated manner. The approach links information requirements to specific data marts elicited by using goal-oriented requirement engineering, which are automatically translated into the implementation of corresponding data repositories by means of model-driven engineering techniques. The authors provide two UML profiles that integrate the design of both data warehouses and data marts and a set of QVT transformations with which to automate this process. The advantage of this approach is that user requirements are captured from the early development stages of a data-warehousing project to automatically translate them into the entire data-warehousing platform, considering the different data marts. Finally, the authors provide screenshots of the CASE tools that support the approach, and a case study to show its benefits.

Keywords: Conceptual Modeling, Customization, Data Mart, Data Warehouse, Goal-Oriented Requirement Engineering, MDA, Model-Driven Engineering, QVT

INTRODUCTION

A corporate data warehouse is a repository that provides decision makers with a large amount of historical data concerning the overall enterprise strategy. A data-warehousing architecture defines a set of data repositories and their relationships to support the decision-making process in a given organization. Several architectural options (Cabibbo & Torlone, 2001; Jarke et al., 2001).
1999; Jukic, 2006; Samos et al., 1998; Watson et al., 2001) and methodologies (Bonifati et al., 2001; Giorgini et al., 2008; Luján-Mora & Trujillo, 2006a; Mazón et al., 2007a; Sen & Sinha, 2005) have been proposed to develop these repositories. Specifically, two foundational data-warehousing alternatives have been broadly discussed (Breslin, 2004): the top-down approach originally stated by Inmon (2005) and the bottom-up approach stated by Kimball and Ross (2002). The basis of these approaches consists of which data repositories should be developed first: a corporate data warehouse in which an organization’s data are stored and integrated in a single repository (top-down) or departmental data marts in which data are aggregated and customized for particular information needs (bottom-up). Although the former is considered to be the most elegant solution from a theoretical point of view, it is usually hard to implement since the project scope involves the whole organization (Watson et al., 2001), and the second approach is thus more suitable for agile developments despite the problems that arise during data-mart integration (Watson et al., 2001; Chaudhuri & Dayal, 1997). Both approaches fail when they attempt to derive the second data repositories (i.e., data marts or corporate data warehouse, respectively) due to the inherent high cost associated to the integration of huge amounts of data (top-down) and to the duplicated integration tasks done by data marts (bottom-up). In order to overcome these limitations, Kimball and Ross (2002) have also proposed a bus architecture articulated by conformed dimensions. These dimensions account for 90 percent of the integration efforts made in order to tie data marts together (Kimball & Ross, 2002). They are obtained through the agreement of the entire organization, thus supporting truly cross-departmental decision-making processes. Despite all this, this solution is designed at the logical level (i.e., by using relational schemata), and does not therefore provide suitable mechanisms to drive complex developments such as methodologies (Bonifati et al., 2001; Giorgini et al., 2008; Luján-Mora & Trujillo, 2006; Mazón et al., 2006; Mazón & Trujillo, 2008) based on conceptual modeling (Abelló et al., 2006; Golfarelli et al., 1998; Hüsemann et al., 2000; Luján-Mora et al., 2006). Furthermore, existing matching methods do not cover the particular problems of integrating data warehouse and data mart schemas (Evermann, 2008).

However, we believe that the surrounding architectural debate (Breslin, 2004) has been overlooked by the current development approaches which are mainly based on conceptual modelling. These approaches have focused on capturing information requirements by means of multidimensional modelling (Kimball & Ross, 2002; Chaudhuri & Dayal, 1997) which organizes data in terms of facts and dimensions of analysis, but does not specify how data repositories (i.e., corporate data warehouse and their dependent data marts) are built from them. For instance, departmental data marts may be built by different development teams in isolation. They therefore lack incorporated conformity issues to solve the integrated development of data marts and corporate data warehouses, in order to assure cross-departmental information needs such as those answered by drill-across operations during “on-line analytical processing” (OLAP) (Chaudhuri & Dayal, 1997).

In this article, we present an approach based on goal-oriented requirement engineering (Yu & Mylopoulos, 1994) and model-driven engineering (Bézivin, 2006) technologies to solve the architectural debate (Breslin, 2004) by supporting Kimball’s insights (Kimball & Ross, 2002) at the conceptual level. Goal elicitation was identified by Ang et al (1995) as the third most critical success factor in enterprise projects, being mandatory to begin any project with a conceptualisation of its goals and the ways to achieve them (Slevin & Pinto, 1987). This solution is based on our previous works (Luján-Mora et al., 2006; Luján-Mora & Trujillo, 2006b; Mazón et al., 2007a; Mazón & Trujillo, 2008; Pardillo et al., 2008) which propose a modelling framework in terms of the goals that the data warehouse should achieve together with the information required to conform to analysis dimensions; and also, a transformation architecture based on the
“model-driven architecture” (MDA) (OMG, 2008; Siau & Cao, 2001) approach through which to automatically derive both the corporate data warehouse and its dependent data marts in an integrated manner. We thus enable decision makers to respond to their cross-departmental information needs. It is worth noting that this modelling framework is not associated to any particular data-warehousing development method, thus, the proposed diagramming techniques and model transformations can be applied whenever a data-warehousing architecture should be designed.

Extended Material

This article is an extended version of the short paper (Pardillo & Trujillo, 2008) presented at the ER’08 conference. In this version the short paper has been extended and improved by: (i) introducing a methodology based on MDA according to the proposed novel UML extensions, (ii) describing each of the design artefacts involved in greater detail and presenting more UML modelling diagrams in order to clarify their application, (iii) introducing two new UML profiles which provide a modelling language for data-mart requirements and data-warehousing repository modelling, (iv) introducing a set of QVT relations/transformations in order to be able to deal with automatic transformations for the novel UML profiles presented in this article, and finally (v), enlarging the case study throughout the entire article in order to clarify how to apply our method in real world projects. This extended version therefore constitutes a natural and significant evolution of the approach presented in (Pardillo & Trujillo, 2008).

The remainder of this article is organized as follows: the following section introduces a motivating example in order to illustrate common conformity problems. The section “Integrated Development of Model-Driven Goal-Oriented Corporate Data Warehouses and Data Marts” presents our goal-oriented model-driven approach for the integrated development of data warehouses. The section “Conceptual Model-
ers cannot drill across them to fulfill their cross-departmental information needs since they are not integrated.

With regard to the dimensions involved (see Figure 1), several problems arise from an isolated data-mart development process:

- **Store dimension.** Stores are described at different granularities (Kimball & Ross, 2002) in each data mart (i.e., a ZIP aggregation level in the inventory and the city level in the sales fact). Cities are aggregated into different levels in each model (i.e., countries for retail stores and states for sales stores).
- **Date dimension.** The concept of date is described by different formats (i.e., day vs. dates). Years in the inventory data mart are fiscal years whereas years in the sales data mart are calendar years (although they are given the same name). Dates are aggregated by months or weeks depending on the data mart.
- **Product dimension.** Products are managed at different granularities (i.e., pieces vs. the products themselves). Products have different semantics (i.e., assembled products). Aggregation levels depend on the actual semantics of products (i.e., brand grouping pieces vs. products). Product types may be related to their sizes (and should therefore be correctly related).
- **Promotion dimension.** This dimension is required by only one of the data marts.

These facts signify that, in the current approaches, drilling across these dimensions is not possible (Kimball & Ross, 2002; Abelló et al., 2002) and data marts remain isolated.

Kimball and Ross (2002) propose the conformity of dimensions through an agreement between every data-mart development team, later providing a foundational definition of conformed dimensions: “two dimensions are conformed if the fields that are used as common row headers have the same domain” (Kimball, 2003). However, this definition is oriented towards the logical level (Hüsemann et al., 2000), in which the name matching of logical structures (i.e., tables, columns, and rows) is necessary to enable drill-across operations to take place by sorting and merging relational database structures. Some authors (Abelló et al., 2002; Cabibbo & Torlone, 2004) therefore generalize the conformity constraints at the conceptual level. For example, (Abelló et al., 2002) supports conformity by finding functional dependencies between dimension instances. On the other hand, other authors (Hurtado et al., 1999) in literature establish more general schema equivalences in terms of their information capacity.

We assume an adaptation of the definition stated in (Kimball, 2003) for the conceptual level: sharing dimensions between conceptual multidimensional models implies that they can be reused through data marts in order to permit cross-departmental decision-making processes. Therefore, our approach is based on using agreement to discover information needs through data marts and combining them in a master conformed template that fulfills all these needs.

Moreover, our approach can also be used to provide the integrated development of the corporate data-warehouse that populates the data marts, thus reducing the expensive efforts involved in data integration (Vassiliadis, 2000).

**INTEGRATED DEVELOPMENT OF MODEL-DRIVEN GOAL-ORIENTED CORPORATE DATA WAREHOUSES AND DATA MARTS**

In this section, we present a development approach for data warehouses based on: (i) discovering information needs for each data mart by applying goal-oriented requirement engineering techniques (Yu & Mylopoulos, 1994), (ii) permitting the dimension-related requirements obtained to conform by using a conformity authority that assures agreement and commitment between every data-mart stakeholder, (iii) providing a conceptual framework.
to model the underlying data repositories, and (iv) automatically translating the information requirements obtained from the very early stages of development to the final implementation by using a model-driven engineering approach (Bézivin, 2006), specifically the well-known MDA proposal (OMG, 2008; Siau & Cao, 2001) which has been successfully employed in our previous works (Luján-Mora et al., 2006; Luján-Mora & Trujillo, 2006b; Mazón et al., 2006; Mazón et al., 2007a; Mazón & Trujillo, 2008; Pardillo et al., 2008).

Goal-Oriented Reasoning to Permit Data Marts to Conform

Kimball and Ross (2002) advocate a "dimension authority" such as the stakeholder responsible for managing conformed dimensions by defining, maintaining and publishing them for each data mart; hence, conformity implies organizational commitment rather than simply a technical decision. Nevertheless, this author does not provide any mechanisms to support this.

Therefore, we propose to enrich the organizational modelling in goal-oriented approaches (Bonifati et al., 2001; Giorgini et al., 2008; Mazón et al., 2007a) by also taking into account conformity issues by explicitly establishing a dimension authority. It is worth mentioning that we can not only respond to data-mart needs, but also integrate them into the strategic policies of the whole organization.

Figure 2 shows the general overview of the stakeholders involved in a data-mart development after including a dimension authority with the i* notation (Yu & Mylopoulos, 1994). The process starts with the elicitation of the information requirements for a particular decision maker in the department (e.g., a sales or inventory manager in Figure 2). By using goal-modelling terminology (Yu & Mylopoulos, 1994), a data-mart developer intentionally depends (represented as \( \rightarrow \)) on other organizational actors (\( \odot \)), i.e., decision makers, in order to obtain the resource (\( [\text{resource}] \)) of his/her particular information requirements. These dependencies are modelled by means of strategic-dependency diagrams like that shown in Figure 2. A data-mart developer therefore depends on the dimension authority to make dimensions conform as a result of the corporate agreement. On the other hand, for this aim, the dimension authority needs the dimensions to conform for the different data-mart teams. Hence, data marts can deploy already conformed data structures which enable decision makers to fulfil their information needs. The conformity achieved signifies that data-mart coalescing queries (Cabibbo & Torlone, 2004) can be employed during cross-departmental decision-making processes. It is worth noting that each data mart is designed from a set of predefined queries that are derived from the decision-maker's goals. This fact does not mean that decision makers cannot evolve their analyses by smoothly getting into completely new and unknown queries as usual in the OLAP technology.

In order to design an integrated modelling framework for this kind of requirements, we have defined a UML profile with which to model the conformity of multidimensional requirements (called Conformity@MultidimensionalRequirements). The UML profile Conformity@MultidimensionalRequirements defines the modelling architecture that is shown in Figure 3. This is articulated by means of two additional profiling layers, namely, the UML profile for modelling multidimensional requirements in the framework of i* (MultidimensionalRequirements@iStar) and the UML profile for modelling i* by using UML based on the previous work (iStar@UML). The UML (OMG, 2008; Siau & Cao, 2001) thus enables us to articulate a coherent modelling architecture with which to incrementally extend UML, thus providing goal-oriented modelling, followed by data-warehousing requirements, and finally conformity. UML profiles are thus an easy mechanism with which to extend a modelling language. These profiles contain stereotypes of the modelling element to be extended (metaclasses of the language) which manage the tag definitions used to describe the extension attributes and the constraints used to specify
Figure 1. Data models of two independent data marts: retail inventory and sales analysis
well-formedness rules over the models. The designed extensions are explained, from the bottom to the upper layer, as follows.

**iStar@UML Profile**

The i* modelling framework (Yu & Mylopoulos, 1994) provides mechanisms with which to represent actors, their dependencies, and the structuring of the business goals that an organization wishes to attain. This framework establishes two models: the strategic dependency (SD) model with which to describe the dependency relationships among various actors in an organizational context, and the strategic rationale (SR) model, used to describe actors' interests and concerns, and how they might be addressed. From here on, we shall focus on describing the SR models used to model decision makers' goals and information requirements.

The SR model (modelled with the SR stereotype and represented as \( \text{\textbullet} \text{\textcircled{}} \)) provides a detailed means of modelling the internal intentional elements and relationships of each actor \((\text{iActor}, \text{\textcircled{}})\). We use intentional elements such as goals \((\text{Goal}, \text{\textcircled{}})\), tasks \((\text{Task}, \text{\textcircled{}})\), resources \((\text{Resource}, \text{\textbullet})\); and intentional relationships such as means-end \((\text{MeansEnds}, \rightarrow)\) to represent alternative means of fulfilling goals, or task-decomposition \((\text{Decomposition}, \rightarrow)\) representing the necessary elements that a task should perform.

**MultidimensionalRequirements@iStar Profile**

In order to define SR models, goals, tasks, and resources are represented as intentional elements for each decision maker. Decision makers’ goals are defined by using the Strategic, Decision, and Information stereotypes are defined by using the inheritance white-head arrow to specialize the previously defined Goal stereotype, and the intentional means-end relationships \((\text{MeansEnds}, \rightarrow)\) between them. Information requirements \((\text{Requirement})\) are derived from information goals and are represented as tasks.

Furthermore, the requirements analysis of data warehouses necessitates the addition of certain multidimensional concepts (in the sense of (Giorgini et al., 2008)). This extension is achieved by using the UML profile for modelling multidimensional requirements by using \(\text{i*(MultidimensionalRequirements@iStar, see Figure 3)}\). The following concepts are therefore added as resources: business processes related to decision makers’ goals \((\text{BusinessProcess stereotype})\), relevant measures related to decision makers’ information requirements \((\text{Measure})\), and contexts needed for analyzing these...
measures (Context). Any foreseen relations between the context of analysis are also modelled. For instance, the city context and the country context are related because cities can be aggregated in countries. These relationships were modelled by using the UML (shared) aggregation relationship (Association UML metaclass, represented as $\rightarrow \circ$).

Conformity@MultidimensionalRequirements Profile

As previously stated, conformity issues require mechanisms to manage the involved rationale with which decisions are made in order to obtain conformed dimensions. Given the previous UML profiles, every stakeholder has a rationale which is necessary to accomplish his/her strategic dependencies in the organization. Thus, we can apply the same principles in order to additionally model the rationale of the dimension authority by means of SR diagrams. The extension achieved by the Conformity@MultidimensionalRequirements profile is defined through the use of the following stereotypes:

- **DataMartUser** is the decision maker that defines a set of information requirements that should be made to conform in order to design a data mart in the context of an integrated data-warehousing architecture.
- **DimensionAuthority** is the project’s stakeholder that plays the role of obtaining the agreement for conformed dimensions of the data marts involved.
- **ConformableContext** is the information items that must be made to conform. They may be conformed (modelled as a tag definition of this stereotype) if and only if they have been agreed on by means of the dimension authority.
- **ConformityAgreement** is the dependency that links the dimension authority to the context to be conformed.

Designers can use these stereotypes to model the problematic elements of the conformed dimensions, as is shown next. Interestingly, this modeling extension enables us to define a set of model-to-model transformations (see “Conceptual Modeling Mapping: from Goals to Data Structures”) which automate the design of the database structures that implement both data marts and the corporate data warehouses.

Application of the Modeling Framework

Figure 4 illustrates the rationale for our case study (see Figure 1). This figure shows a data mart user (sales manager) whose rationale concerning data analysis is modelled. This model is created by first considering the business process sales which s/he has to manage. Given this, this user has a strategic goal to increase sales that may be refined into two decisional goals decrease price and determine promotion. In decisions, other informational goals are involved which have associated information requirements, respectively: analyze price, which implies aggregate total amount by product and brand and analyze sales, which implies aggregate quantity sold by product type and promotion code. After the goal refinement, information requirements are discovered in the form of the measures total amount and quantity sold and the contexts that must be made to conform (brand, product, and so on). Moreover, the contexts are aggregated to each other (e.g., products into brands or product types).

A dimension authority should manage each conformable context discovered. Figure 5 shows this management for the contexts related to products. First, the most granular context is identified (product for the sales manager and pieces for the inventory manager). They are then made to conform to the project’s dimension authority. To do this, the conformity on product dimension conformity agreements are modelled in the strategic-dependency diagram. The dimension authority’s rationale (sales & inventory authority) is then employed to make the context of analysis of both data marts conform. They are related to the same strategic goal conform product dimension. So, several decisions are made by
the dimension authority in order to make these contexts conform: subclassify products, merge brands, and preserve pieces, sizes, service types & product types. These decisions finally involve the conformed contexts that enable both data marts to be integrated: product, assembly, and so on. The problems of the initial product dimension (Figure 1) are solved by the conformed contexts because they have been defined by taking into account the requirements of both data marts.
Conceptual Modeling of the Data-Warehousing Architecture

In this section, we enrich our conceptual modelling framework for data warehouses (Luján-Mora et al., 2006) (denominated in Figure 7 as MultidimensionalData@UML) in order to tailor the represented schemata for specific data repositories, i.e., data marts or the corporate data warehouse itself. This extension is achieved by using the modelling architecture shown in Figure 7. This is a two layer architecture of UML profiles in which the bottom layer is occupied by the UML profile for multidimensional data (hereafter denominated as MultidimensionalData@UML profile) and the upper layer is occupied by the UML profile for modelling data-warehouse repositories (Repository:Architecture@MultidimensionalData). Since the first profile was discussed during the motivating example (“Motivating Example”), the following discussion is focused solely on the Repository:Architecture@MultidimensionalData profile.

First, we extend the previous three-layer packaging architecture (Luján-Mora et al., 2006) by also including the deploying data repositories (i.e., data marts and corporate data warehouse). Figure 6 shows the relationships between the different packages. The entire model of a corporate data warehouse is composed (represented as –→ (OMG, 2008)) of all the data-mart models. Furthermore, each model of a departmental data mart is composed of several star packages (a conceptualization of a logical star schema (Kimball & Ross, 2002; Moody & Kortink, 2000)), and each one is additionally composed of several fact packages. Moreover, we define a dimension library as a catalogue with which to publish master conformed dimensions that can be obtained from the dimension authority’s rationale (explained in the following section) and reused in each data-mart model. Hence, the model is composed of several dimension packages containing the project’s conformed dimensions which are (<<import>>) dependency relationships (OMG, 2008)) imported by the star packages in order to describe the contained facts.

In Figure 7, we show the RepositoryArchitecture@MultidimensionalData profile containing several stereotype and tag definitions.

\[\text{Figure 4. Discovering information requirements in products to be conformed}\]

Copyright © 2011, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
by extending some of the UML metaclasses, i.e., its modeling elements. In this case, we only need to extend UML packages and models. In addition, this profile imports the definition of the MultidimensionalData@UML profile, as has previously been explained. Hence, we also extend raw dimensions from (Luján-Mora et al., 2006) by enriching their semantics in order to additionally include conformity issues. For each stereotype, we provide a representative icon that allows developers to easily understand and recognize the proposed modeling elements (Table 1). The semantics of the provided stereotypes and tag definitions is the following:

- **ConformedDimension** defines the dimensions that may be conformed depending on the conformed tag value (it is true by default).
- **ConformedDimensionPackage** defines the package that contains a conformed dimension. It is made to conform depending on the conformed tag value (it is true by default).
- **DataArchitecture** defines an entire architectural model, which, technically speaking, acts as a container of the remaining modelling elements.
- **DimensionLibrary** defines the catalogue of conformed dimensions which data marts use in order to define an integrated data-warehousing solution.
- **DataRepository** (abstract) represents general data repositories in the architecture. It is additionally specialized by the other stereotypes.
- **DataWarehouse** represents a repository of historical data with the properties stated by Inmon (2005). It can be specialized by using their scope or type of analysis.
- **CorporateDW** is a data warehouse for an overall organization. It can be integrated if it has dependent data marts.
- **DataMart** is a departmental data repository that contains aggregated and customized data for responding to specific information needs. It can be dependent of a corporate data warehouse.
- **OLAP** is a data warehouse, which is oriented towards OLAP analysis. It is therefore designed by using the multidimensional-modeling paradigm. In addition, the technology used can also be specified.
- **FlatFile** is a data store for analysis applications without any special information needs as regards data structures.
- **OLTP** is a data repository, which is oriented towards the “on-line transaction processing” which is typically used for populating data warehouses.
- **ODS** is an operational data store that serves as a staging area for the population of a data warehouse.

**Application of the Modeling Framework**

Specifically, Figure 8 shows the library of master conformed dimensions obtained from the dimension authority’s rationale of our case study. It is worth noting that the dimension library is the foundation through which to later automatically derive the dependent data-mart schemata in an integrated manner. For instance, the required product dimensions are combined in the conformed product dimension which allows inventory and sales facts to be described through the commitment of every particular information requirement.

The proposed packaging architecture is applied to our example scenario and is shown in Figure 9. Inventory and sales data marts are modelled as packages (correctly stereotyped) which depend (<<import>> dependencies) on the corresponding {conformed} dimensions (correctly tagged) that are contained in the retail dimension library. Moreover, the retail corporate data warehouse is modelled as a package which contains every data mart (  

---

Copyright © 2011, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
with \( \rightarrow \) relationships) in the data-warehousing architecture. The conceptual modelling framework provided can be used to translate the previously exposed goal reasoning into multidimensional models. These conceptualize required data structures in order to deploy not only data marts but also the entire corporate repository.

**CONCEPTUAL MODELLING MAPPING: FROM GOALS TO DATA STRUCTURES**

Any of the presented models (i.e., goal-based or multidimensional) can be mapped in order to automatically derive data structures for both a corporate data warehouse and its dependent data marts.

Our solution is based on the best-known initiative for model-driven engineering (Bézivin, 2006), namely the “model-driven architecture” (MDA) (OMG, 2008). This proposal enables us to specify model-to-model transformations by means of “query/view/transformation” (QVT) (OMG, 2008) language. It contains a declarative element which enables us to easily design the required model mappings in a visual form. Our transformation chain, as it is called in QVT terminology, is divided into three stages concerning each modelling framework: the “computer-independent model” (CIM), the
Figure 7. A UML profile for data-warehousing repository modeling

Table 1. Iconography for the defined stereotypes at the conceptual level

<table>
<thead>
<tr>
<th>Stereotype</th>
<th>Iconography</th>
<th>Stereotype</th>
<th>Iconography</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConformedDimension</td>
<td>![ConformedDimension Icon]</td>
<td>DimensionLibrary</td>
<td>![DimensionLibrary Icon]</td>
</tr>
<tr>
<td>ConformedDimensionPackage</td>
<td>![ConformedDimensionPackage Icon]</td>
<td>FlatFile</td>
<td>![FlatFile Icon]</td>
</tr>
<tr>
<td>CorporateDW</td>
<td>![CorporateDW Icon]</td>
<td>ODS</td>
<td>![ODS Icon]</td>
</tr>
<tr>
<td>DataMart</td>
<td>![DataMart Icon]</td>
<td>OLAP</td>
<td>![OLAP Icon]</td>
</tr>
<tr>
<td>DataWarehouse</td>
<td>![DataWarehouse Icon]</td>
<td>OLTP</td>
<td>![OLTP Icon]</td>
</tr>
</tbody>
</table>
“platform-independent model” (PIM), and the “platform-specific model” (PSM). These therefore allow us to smoothly isolate the deployment platform by means of different abstraction levels, thus tackling complex projects such as data warehousing.

Figure 10 shows the QVT relations designed to map the requirements models based on i* into the discussed multidimensional models. Each relation has a source modelling element (marked as checkonly by QVT) which is translated into a modelling element on the target metamodel (marked as enforce). For instance, data-mart users in our i* extension are translated into data-mart packages at the conceptual level (by means of a DataMartUser2DataMart relation). Moreover, in order to translate a given modelling element, a relation may require others. These dependencies are also represented in Figure 10. The dependency graph roots the entire model mapping, i.e., strategic dependency diagrams (SD) into data architectures (DataArchitecture). Packaging-related mapping can then be performed. For example, the dimension authority in i* can be mapped into the dimension library in multidimensional modelling. There are thus several mappings that translate the relationships and modelling elements contained in the previously mapped packages. In this layer, for example, dimensions are translated from a conformable context in i*, and relationships between data marts and a conformable dimension library are translated from conformity agreements. These relations are rendered in QVT by means of a visual notation such as that presented in Figures 11 to 13.

A QVT relation (see Figure 11 as example) has two modelling languages (named istar and md) over which two modelling elements are labeled as the relation domain. These are used to specify certain modelling elements in order to define a pattern through which the source modelling will be checked in order to match or generate (since the target model is marked as enforced) the target modelling elements. For instance, Figure 11 shows the mapping from data-mart users into data marts. Data-mart users are represented by means of two primitives: a domain class (dm_c) and the stereotype which extends it (dm_s). In addition, this class is related to a given strategic dependency diagram which is also represented in a dual form due to the usage of UML profiles (package sd_p and stereotype sd_s). By means of pattern matching, the source structure is related to the target source in which domains are first matched by their names (value n). Moreover, due to its dependencies, this QVT relation has a when clause that acts as a mapping precondition. In this case, the precondition establishes that sd_p and da_m are related by means of the QVT relation SD2DataArchitecture. A QVT engine thus takes these declarative definitions and executes them in a particular order given the when and where clauses.

Figure 12 also shows the QVT relation for mapping conformable contexts into conformed dimensions. In this mapping, the when clause defines the condition on the context to be translated, which should be the most granular context in the aggregation hierarchy of context. Thus, its counterpart in the multidimensional model is a conformable dimension (which shares the same value as its conformed property). Conversely, if it is not the most granular context, this rule will not be executed. The Conformable2Base will be executed instead. The when clause also defines a condition for the related strategic rationale (SR), which should be the rationale of the dimension authority. The utility function isStereotypedBy is used for this purpose.

As a final example of the QVT relations designed, Figure 13 shows the relation between conformity agreements (which are represented by means of stereotype ca_s and association class ca_ac) and import relationships (import element metaclass) between data marts and conformed dimensions. In this case, a where clause is also specified to identify the most
granular context which is, in fact, the context used for the conformity agreement.

**Dealing With Existing Data Marts**

It is worth noting that our approach is focused on a scenario in which the data warehouse project is in early stages of development, so data marts are not deployed yet. However, there is a different scenario in which our approach can be useful: in later stages of the development, when some of the data marts are already deployed. This scenario can be easily considered if we applying reverse engineering techniques to the existing data marts in order to obtain their corresponding conceptual multidimensional models. In our previous work (Mazón & Trujillo, 2007), an approach for the development of a data warehouse as a modernization process addressed the analysis of the available data sources to discover multidimensional structures with which to derive a conceptual multidimensional model. Therefore, this modernization process can also be applied to an existing data mart before starting the design process defined in this paper in order to obtain the corresponding conceptual multidimensional model.

**Application of the QVT Mappings**

Figure 14 shows the model-transformation chain for our case study. The transformation chain begins from *i* diagrams (Yu & Mylopoulos, 1994) in our conceptual modelling framework.

We decompose the transformation process into **measure-fact** and **context-dimension** mappings. Whereas the mappings presented in (Mazón et al., 2007a) are oriented towards the generation of a single data repository in isolation, the information requirements to be translated herein are spread out: (i) rationales of data-mart decision makers for measure-fact mappings, and (ii) dimension authority’s rationale for context-dimension mappings. Thus, we translate the dimension authority’s rationale, which holds contexts of analysis, to the conceptual library which contains the translated conformed dimensions. However, together with each obtained dimension, we also map it into the required package structure (see Figure 6).

The mapping of decision maker’s rationales, such as those of the inventory manager, implies a model merging with the dimension library (shared by data marts across every department). As Figure 8 shows, since conformed dimensions are already mapped into the dimension library, each context discovered in these rationales (conformed by the dimension authority) is translated into an `<import>` dependency from the related fact to the conformed dimension. We thus ensure that the facts in each data mart can be drilled across the conformed dimensions. Moreover, the model of the entire (corporate) data warehouse that contains every data mart is also automatically derived from the whole strategic-dependency model (Figure 2). Once again, the packaging scaffolding (Figure 6) is also taken into account for deriving the models of data marts and the corporate data warehouse.

The conceptual modelling of data repositories can also be used to automatically derive the deployment metadata that implement them. These transformations are carried out by model mappings adapted from (Mazón et al., 2006; Mazón & Trujillo, 2008; Pardillo et al., 2008). Essentially, the mappings involved match each multidimensional concept with both the data structures of the data repository and the client metadata to query them by following a multidimensional view. Specifically, facts and dimensions, together with their measures and aggregation hierarchies, are mapped at the logical level (Hösemann et al., 2000) into the corresponding tables and columns of a star schema (Kimball & Ross, 2002; Moody & Kortink, 2000) with regard to the relational model. Given the model of the corporate data warehouse that collects all the dependent data marts, by applying the mappings in (Mazón et al., 2006; Mazón & Trujillo, 2008), we obtain the data structures which the corporate repository implements. Given a data-mart model, we obtain the corresponding aggregated and customized version of the entire repository. It is worth noting that these data structures are conformed by
Figure 8. Dimension library for modelling conformed dimensions of the retail data marts
their dimensions. We can therefore automatically obtain their deployment counterparts in an integrated manner. Moreover, using our previous work (Pardillo et al., 2008) as a basis, we also automatically generate the required metadata in order to query them by using OLAP applications (Chaudhuri & Dayal, 1997), thus ameliorating the tedious work of their manual definition. Concerning platform-specific design decisions such as indexing or partitioning, continue being able to be managed by similar QVT transformations (in latter design phases) to which the article presents.

DEVELOPMENT PLATFORM AND IMPLEMENTATION

The related standards that we employ are also shown in Figure 14. On the one hand, for
Figure 11. QVT relation for mapping data-mart users over i* diagrams into data marts at the conceptual level

Figure 12. QVT relation for mapping conformable contexts over i* diagrams into conformed dimensions at the conceptual level

goal-oriented requirement engineering, we employ i* diagrams (Yu & Mylopoulos, 1994) supported by our UML profile presented in (Mazón et al., 2007a). With regard to the multidimensional modelling of data warehouses, we use the UML profile presented in (Luján-Mora et al., 2006), which is enriched for their architectural modelling as has previously been described. In addition, the “common warehouse metamodel” (CWM) (OMG, 2008), which is
widely used to deal with the interoperability of applications (Zhao & Siau, 2007), is employed to represent the deployed data structures for both the underlying databases and OLAP applications (Chaudhuri & Dayal, 1997) in a vendor-independent manner.

All the modelling frameworks and model transformations with regard to our solution have been implemented in the Eclipse® development platform, since a key factor for the success of any MDE or MDA approach is an appropriate tool support (Saraiva & Silva, 2008).

Specifically, we employ several of its plugins that implement the MDA standards: for instance, “model development tools” (MDT) to support UML and UML profiles, the “eclipse modelling framework” (EMF) to specify CWM metadata representations in a vendor-independent manner, medini QVT and SmartQVT in order to specify and launch model-to-model QVT mappings with its declarative or imperative part, respectively, or MOFScript to design model-to-code Mof2Text (OMG, 2008) mappings in order to automatically implement the final data-warehousing solution. These have been combined to provide an “integrated development environment” (IDE) to manage data-warehousing projects based on model-driven engineering. This tool has been used to implement our case study as a proof of concept of our approach. It is shown in Figures 15 and 16. The first figure shows the modelling of conformed requirements by using i*. Both an outline of the strategy dependency diagram (left-hand side) and strategic rationales are represented (for the sales manager on the right-hand side and for the dimension authority at the bottom). With regard to the second figure, in this case the inventory data mart is modelled on the left-hand side in order to automatically transform it into the deployment OLAP metadata (shown on the right-hand side), by applying the QVT mapping at the centre of Figure 16.

**RELATED WORK**

The development approaches for data warehouses that appear in literature can be divided into those that present methods or guidelines
to capture information requirements such as (Bonifati et al., 2001; Giorgini et al., 2008; Luján-Mora & Trujillo, 2006a; Mazón et al., 2007a) and those that present modelling frameworks for data structures that respond to these requirements (Abelló et al., 2006; Golfarelli et al., 1998; Hüsemann et al., 2000; Luján-Mora et al., 2006).

Specifically, very few authors (Bonifati et al., 2001; Giorgini et al., 2008; Mazón et al., 2007a) have investigated goal-oriented approaches such as that presented here, and none of them have attempted to capture conformity
issues between decision maker’s requirements or to derive multidimensional data models for the every data repository involved in an entire data-warehousing architecture. Thus, those researchers that have investigated the conceptual modelling of data warehouses have mainly focused on intraschema properties such as additivity (Horner et al., 2004) and aggregation hierarchies (Malinowski & Zimanyi, 2006).

However, the research community has made great efforts in the related issues of data integration and view materialization, as can be observed by the number of research papers published in the field (Vassiliadis, 2000). For instance, Moody and Kortink (2000) recognize three kinds of multi-star data models (i.e., constellation, galaxy, and star cluster). These are, however, focused on the logical structures. Thus, real conformity mechanisms or data-repository deployments are not established. At the conceptual level, the best efforts for modeling interschema properties have been made by (Abelló et al., 2002; Abelló et al., 2006; Cabibbo & Torlone, 2004). In Abelló et al. (2002), several kinds of relationships between facts and dimensions to drill across different schemata are discussed. Nevertheless, this model is oriented towards complex data relationships that decision makers do not usually require (Pedersen, 2004). Once again, there are no mechanisms with which to achieve agreement and commitment between data-mart stakeholders. In addition, Cabibbo & Torlone (2004) propose the “dimension compatibility” notion to drill across data marts; nevertheless, this is not oriented towards the integrated development of the data repositories involved. It is also worth noting that data-warehouse design can never avoid the analysis of data sources. For this purpose, hybrid methods such as the presented by Mazón and Trujillo (2009) can be enriched with the presented approach. However, this article is focused on the conformity of data warehouses and data marts from user’s requirements, and therefore, is more focused on the goal-modeling part. If the reader wishes more information on the integration of data sources, we suggest the reading of (Mazón et al., 2007b).

Concerning the goal-oriented modelling, Prakash and Gosain (2008) studied the conceptual relationship between data analysis and the decision-making process. These authors stated that such relationship is made by decisions: database queries answer information needs, that support some decision, that in addition is taken.
in order to achieve some goal. Paim and Castro (2002) also proposed a goal-oriented framework for modeling data warehousing requirements. They studied many of the non-functional requirements that are involved in such technology and their interrelationship. In particular, these authors classify the issues dealt with by our proposal as part of the ‘multidimensionality’ non-functional requirement (“Ability to represent decision-support requirements as and provide access to dimensional and factual data.”), being actually identified as ‘conformance’ (“Ability to represent common data warehouse aspects in the same way across the entire data warehouse specification”).

Finally, the proposed library of conformed dimensions is a similar concept to the design pattern approach that has been investigated by Jones and Song (2008) in order to propose dimensional patterns for data warehouses. However, these patterns act as design guidelines, and they do not signify suitable conformance mechanisms which enable integrated architectural deployments. It is worth noting that other authors such as Niemi et al. (2001) also propose automatic mechanisms to generate OLAP schemata. However these mechanisms are not conceived to solve the conformance among dimensions or to drive an entire data-warehousing architecture.

CONCLUSION

Two alternative approaches for the implementation of data-warehouses have been widely discussed (Bresil, 2004):

- The top-down (Inmon, 2005) which aims to first design a corporate data warehouse in which an organization’s data are stored and integrated in a single repository;
- The bottom-up (Kimball & Ross, 2002) which conversely advocates first developing departmental data marts in which data are aggregated and customized for particular information needs.

The top-down approach is usually difficult to implement since the project scope involves the whole organization (Watson et al., 2001), and thus, the bottom-up approach is preferred in most cases (Chaudhuri & Dayal, 1997; Watson et al., 2001). However, this second approach fails when data marts are conformed to support cross-departmental decision making processes. To overcome this drawback, in this work we present an approach for the integrated design of an entire architecture for data warehousing from the very-early stages of development.

The cornerstones of our approach are as follows:

1. A goal-oriented requirements engineering approach based on i* (Yu & Mylopoulos, 1994) is used to capture information requirements at the early development stages, thus allowing us to anticipate risks at the very beginning of every project. A goal-oriented approach also allows developers to align particular information needs with the strategic policies of the whole organization. Finally, the agreement and commitment between distributed data-mart stakeholders to make their multidimensional models conform is coordinated by using this goal-oriented approach, and also provides artefacts with which to document these conformity agreements.

2. The usage of a model-driven approach is useful for complex developments (Bézivin, 2006) such as data-warehousing architectures from their inception to their final deployment. The use of a model-driven approach has the following advantages:
   - The data warehouse architecture is developed in an implementation independent manner, without concentrating on the underlying software platforms, this knowledge being delegated to the scaffolding model transformation architecture based on QVT.
   - It provides reusable assets for multidimensional models by means of a
catalogue of conformed dimensions, thus permitting the design of modular systems.

- The automatic and integrated deployment of both corporate data warehouse and dependent data marts by using QVT signifies a practical solution to the architectural debate (Breslin, 2004).

- It enables cross-departmental queries by automatically generating the OLAP metadata supported by conformed dimensions for each data mart involved.

Further investigations can be carried out in order to enrich the proposed approach and could, for instance, provide semantics-aware frameworks for conformity reasoning and information requirement discovery, suitability metrics to compare the ideal dimensions obtained from the data-mart requirements with the master conformed dimensions that we have designed, or recommendation capabilities over existing architectures. Importantly, our ongoing research is focused on assessing the suitability of the presented diagramming techniques, e.g., providing more concise visual primitives or richer mechanisms to decompose diagrams in manageable logical units. Importantly, our short-term future work consists of formalizing the specification of the underlying methodology in a process-oriented modelling language, accompanied by the appropriate empirical validation to assess the benefits of the presented framework and study its expressivity and completeness in real-world scenarios. It will be done by means of a collection of experiments in which the approach will be evaluated by first considering computer science students and then practitioners from a company in the sense of our previous experimental works (Serrano et al. 2004, 2007).

ACKNOWLEDGMENTS

This work has been partially supported by the MESOLAP project (TIN2010-14860) from the Spanish Ministry of Education and Science, and by the QUASIMODO project (PAC08-0157-0668) from the Castilla-La Mancha Ministry of Education and Science (Spain). Jesús Pardillo is funded by the Spanish Ministry of Education and Science under FPU grant AP2006-00332.

REFERENCES


Copyright © 2011, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
ENDNOTES

1  28th International Conference on Conceptual Modeling (ER'08)
2  UML profiles are herein denominated as guest@host where guest is the modelling language to be hosted, and the host is the modelling language that allocates the previous one.
3  http://www.eclipse.org (March 2008)

Jesus Pardillo graduated with honours in Information Systems Engineering in 2006 and obtained his Ph.D. in Computer Science in 2010 from the University of Alicante. He currently enjoys a research grant from the Spanish Ministry of Education and Science and works at the University of Montreal. He has publications on software engineering in international conferences such as MODELS, DOLAP, QUATIC, ER, DAWAK, or JISBD, and journals such as the Journal of Systems and Software, the International Journal of Intelligent Systems, Information Sciences, or Information and Software Technology. His research interests include: conceptual data modelling, programming languages, formal methods in software engineering, and information & software visualisation.

Jose-Norberto Mazón is Assistant Professor at the Department of Software and Computing Systems in the University of Alicante (Spain). He obtained his Ph.D. in Computer Science from the University of Alicante (Spain) within the Lucentia Research Group. He has published several papers about data warehouses and requirement engineering in national and international workshops and conferences, (such as DAWAK, ER, DOLAP, BNCOD, JISBD and so on) and in several journals such as Decision Support Systems (DSS), SIGMOD Record or Data and Knowledge Engineering (DKE). He has also been co-organizer of the International Workshop on Business Intelligence and the WEB (BEWEB 2010) and the International Workshop on The Web and Requirements Engineering (WeRE 2010). His research interests are: business intelligence, design of data warehouses, multidimensional databases, requirement engineering and model driven development.

Juan Trujillo is a Full-time Professor at the Department of Software and Computing Systems in the University of Alicante (Spain). His main research topics include business intelligence applications, data warehouses’ development, OLAP, data mining, UML, MDA, data warehouses security and quality, etc. He has published more than a 120 papers in different national and international highly impact conferences such as the ER, UML, ADBIS or CaSE, and more than 30 papers in highly ranked international journals indexed by JCR such as the DKE, DSS, IJISOF, IS, or JDBM. He has also been co-editor of five special issues in different JCR journals (e.g. DKE). He has also been PC member of different events and JCR journals such as ER, DAWAK, CIKM, ICDE, DOLAP, DSS, JDM, IJISOF or DKE, and PC Chair of DOLAP'05, DAWAK'05-'06 and FP-UML '05-'09. Further information on his main research publications can be found on: http://www.informatik.uni-trier.de/~ley/db/indices/a-tree/t/Trujillo:Juan.html.