EXPLAINING DATA WAREHOUSE DATA TO BUSINESS USERS - A MODEL-BASED APPROACH TO BUSINESS METADATA

Stefanov, Veronika, Women's Postgraduate College for Internet Technologies, Institute of Software Technology and Interactive Systems, Vienna University of Technology, Favoritenstr. 9-11, 1040 Vienna, Austria, stefanov@wit.tuwien.ac.at

List, Beate, Women's Postgraduate College for Internet Technologies, Institute of Software Technology and Interactive Systems, Vienna University of Technology, Favoritenstr. 9-11, 1040 Vienna, Austria, list@wit.tuwien.ac.at

Abstract

Data Warehouse systems today represent a single source of information for analyzing the development and results of an enterprise organization in a changing environment. The data in the data warehouse describes events and statuses of business processes, products and services, goals and organizational units, and generally mirrors every aspect of the structure and behavior of the organization.

Business users are accustomed to their own vocabularies and concepts, and data interpretation is greatly improved by knowledge about context.

Surprisingly, information about the relationship between the data warehouse data and the organization is not made available to the data warehouse users or even recorded in a suitable way. In this paper, we present an approach that uses enterprise models and modeling techniques to record the at present mainly implicit knowledge about this relationship. We use models to derive Business Metadata, *which forms an additional level of abstraction on top of the data-oriented data warehouse structure. Business metadata makes context knowledge easily accessible and improves the data interpretation for the business users.*

Keywords: Data Warehouse, Enterprise Model, Business Metadata

1 INTRODUCTION

Data Warehouse (DWH) systems represent a single source of information for analyzing the development and results of an organization (List and Machaczek 2004). Measures, such as the number of transactions per customer or the increase of sales during a promotion, are used to recognize warning signs and to decide on future investments.

By describing events and statuses of business processes, products and services, goals or organizational units, the data in the data warehouse mirrors the structure and behavior of the organization. In the organization, information about this relationship between the data warehouse data and the business processes, products, etc. is usually available in the form of enterprise models and documents, and is used during the design phase of the data warehouse.

Surprisingly, the knowledge about this relationship is not made available to the data warehouse users or even recorded in a suitable way. Due to the data-oriented nature of Data Warehousing, the knowledge of how the data warehouse measures relate to business processes or products is not easily accessible to data warehouse users. As it is mainly implicit knowledge, it is also more likely to be lost or forgotten.

Business users are accustomed to their own vocabularies and concepts, and data interpretation is greatly improved by knowledge of context. Using and understanding traditional data-oriented Data Warehousing frontends therefore requires additional effort from the users. If knowledge about the business context is left to chance, data analysis is bound to miss important points and becomes more errorprone.

We identify a need for describing the relationship between the data in the data warehouse and the organization that surrounds it. We propose an approach that allows us to show:

- Which parts of the data warehouse data is created by which business process or part of it
- How business processes have impact on the values of the data warehouse data
- Which parts of the data warehouse data measures which (sub)process or products
- Which organizational units and roles are measured along with the processes
- Where the products and deliverables of the processes are mirrored in the data warehouse data structure

Indeed, adding context and background information to a data warehouse has been an open question in Data Warehousing for years. The term *business metadata* is used for data that describes the business context of the core data, its purpose, relevance, and potential use. There is general agreement on the usefulness and desirability of business metadata. But how to create or derive business metadata is still very much an open question.

Our approach makes use of the knowledge already available in the organization in enterprise (meta-) models to derive business metadata. We use a *weaving model* to store and manage the relationships between the data warehouse data model and a model describing the enterprise organization. The business metadata provided via the weaving model helps to improve the understanding and interpretation of the data warehouse data by the users. It basically creates an additional level of abstraction on top of the data warehouse data, which is aligned to the concepts that are well-known to the users, e.g. a business process-oriented view.

The approach provides the following contributions:

- It makes the implicit relationships between the data in the data warehouse and the structure and behavior of the enterprise organization visible and accessible.
- By relating the measures in the data warehouse to organizational concepts, users are better able to interpret the performance of the enterprise, and to understand the implications.
- Data warehouse requirements analysis and (re-)design are notoriously challenging tasks, because the business context of a data warehouse is difficult to extract from user interviews and practically impossible to store directly in the multidimensional data structures. Weaving enterprise models with data models makes context information accessible, and does so without disrupting the involved models.
- As a by-product, the weaving model can be used for model validation, as is identifies missing or superfluous tables and measures in the data warehouse, as well as omissions in the enterprise model.

We already investigated the relationship between Data Warehousing and a subset of enterprise models, enterprise goal models, in a previous paper (Stefanov & List 2006). As data warehouses can be used to measure the level of achievement of the goals of an enterprise organization, we created and introduced an approach to business metadata based on enterprise goals. This paper builds on and substantially extends our previous work to include business processes, products and the organizational structure.

This paper is structured as follows: The next section gives a short overview over the concepts our approach is based on: Enterprise models in general, the Architecture of Integrated Information System (ARIS) and the Event-Driven Process Chain (EPC), which we use to describe the enterprise organization, the multi-dimensional data model for describing the data warehouse data structure, and finally the concept of model weaving, for bringing it all together. Section 3 gives the details of our weaving model and explains how the business metadata is derived. An example of how business metadata looks like in a tool is shown. Section 4 treats related work, followed by our conclusion.

2 BACKGROUND

2.1 Enterprise Models

An enterprise model formally represents the basic building blocks of an organization, its structure, behavior and goals. It is usually organized into several aspects that can be modeled individually but also related to each other (Whitman & Ramachandran & Ketkar 2001). The Architecture of Integrated Information Systems (ARIS) (Scheer 1999) is a typical example for such an enterprise model. Other similar approaches include CIMOSA (Kosanke & Vernadat 2002) and MEMO (Frank 2001).

Figure 1 shows the outline of a generic enterprise model, organized into five aspects: The enterprise strives to achieve *goals*, acts through *processes*, has an *organizational structure*, produces *products* and uses *software applications*. In the enterprise model, an organization chart can be used to describe the organizational structure, i.e. the dependencies between the departments, groups and roles that exist within the organization. Similarly, business process models describe the structure of business processes with control flows, inputs and outputs. The products, applications, and strategic goals can also be modeled separately, as well as connected to the other aspects in a single model. Such an overview model can connect all models to show for example how processes fulfill goals, are performed by organizational roles, fall into the responsibility of departments, and use applications to produce products for other departments.

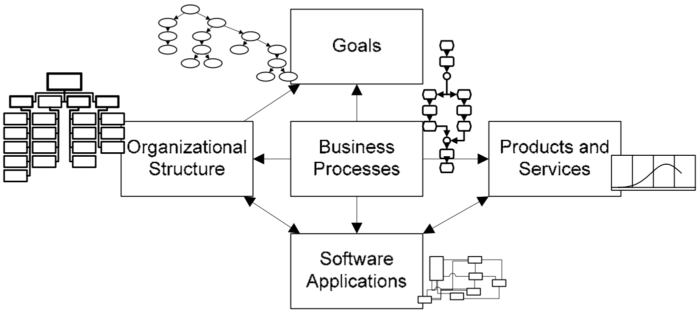

Figure 1. Example of a generic enterprise model

All aspects of the enterprise model are related to the data in the data warehouse, because data in the data warehouse is based on and mirrors the structure and behavior of the enterprise. We therefore use enterprise models for our approach.

2.2 The Architecture of Integrated Information System (ARIS) and the Event-Driven Process Chain (EPC)

The Architecture of Integrated Information System (ARIS) concept (Scheer 1999) involves dividing complex business processes models into separate views, in order to reduce the complexity. There are three main views focusing on functions, data, and the organization, and an additional view focusing on the integration of the other three.

The EPC has been developed within the framework of ARIS and is used by many companies for modeling, analyzing, and redesigning business processes. It is the key component of SAP R/3's modeling concepts for business engineering and customizing. The EPC is based on the concepts of stochastic networks and Petri nets. EPCs describe processes on the level of their business logic, and are targeted to be easily understood and used by business people (Keller & Nüttgens & Scheer 1992). A basic EPC consists of *Functions* and *Events*. The extended EPC additionally contains *Organization Units* or *Roles* and *Information Objects.*

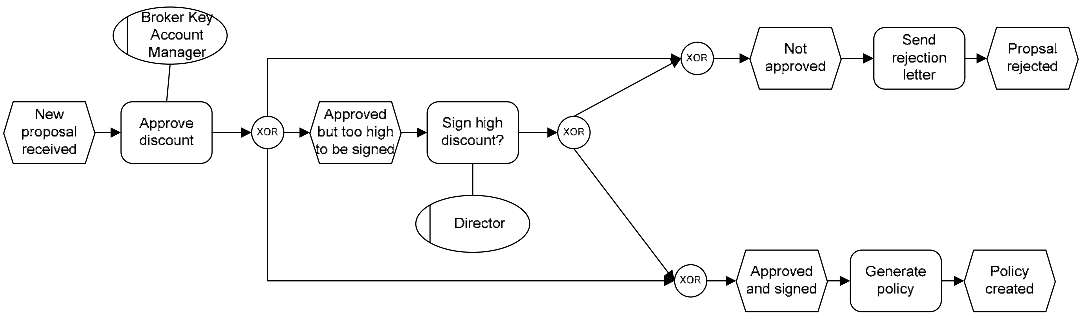

Figure 2. Example for an Event-Driven Process Chain (EPC): Insurance policy creation

We have chosen the EPC for our approach because of its wide-spread use in many companies for modeling business processes, and because of its flexible view concept, that allows to separate the different aspects of a business process.

Figure 2 shows a simplified example process from an insurance company. The process starts with the arrival of a proposal for an insurance policy (as created and sent by an insurance broker for example). The broker key account manager checks whether the discount is appropriate. There are three possible outcomes of this check: The proposal can be either approved and signed, or not approved by the key account manager, or, if it is approved but the discount is above a certain amount, it has to be signed by a director. For each signed proposal, an insurance policy is generated, whereas if it was not approved, a rejection letter is sent.

2.3 The Multidimensional Data Model

Data warehouse applications involve complex queries on large amounts of data, which are difficult to manage for human analysts. Kimball, Reeves, Ross and Thornthwaite (1998) state that relational data models “are a disaster for querying because they cannot be understood by users and they cannot be navigated usefully by DBMS software”. The main logical data model in Data Warehousing is the multidimensional model, also called star schema (Chaudhuri & Dayal 1997). It is said to provide intuitive and high performance data analysis (Kimball et al. 1998).

The multidimensional paradigm allows data access in a way that comes more natural to human analysts. The data is imagined as located in n-dimensional space, with the dimensions representing the different ways the data can be sorted (e.g., sorted by time, by customer type, etc.), analogously to a two-dimensional spreadsheet.

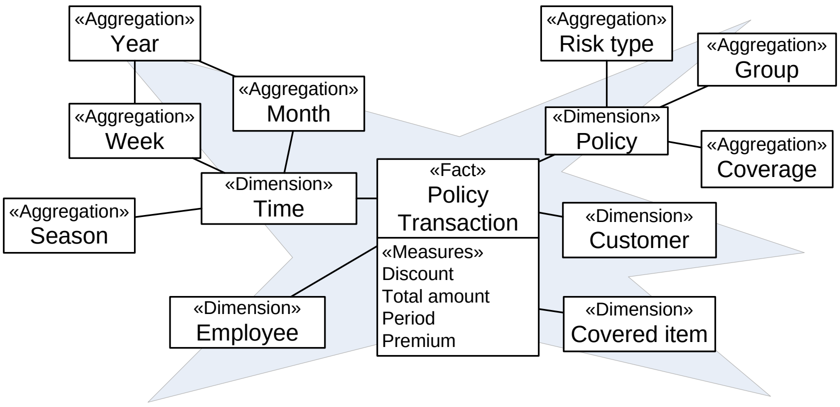

Figure 3 . Multi-dimensional data model of 5-dimensional cube, i.e. a fact table with 5 dimensions and 4 measures, in UML notation (cf. Luján-Mora et al. 2002). Aggregation levels are shown only for the Time and Policy dimensions.

A multidimensional model, also called star schema or fact schema, is basically a relational data model (as originally introduced by Chen in 1976) in the shape of a star or snowflake (see Figure 3 for an example). At the center of the star there is the *fact* table. It contains data on the subject of analysis (e.g., policy transactions, or sales, repairs, admissions, expenses, etc.). The attributes of the fact table (e.g., amount, duration, cost, revenue, etc.) are called *measures*. The spokes/points of the star represent the *dimensions* according to which the data will be analyzed (e.g., by employee and month or by customer group and covered items). The dimensions can be further broken down into hierarchies that are useful for aggregating data (e.g., day, month, year). Several stars can share their dimensions, thus creating a web of interconnected schemas that makes drill-across operations possible.

There are many approaches to modeling the multidimensional data structures of data warehouses, as described and compared by Vassiliadis and Sellis (1999) and Blaschka, Sapia, Höfling and Dinter (1998). Some of them are object-oriented models or based on the Unified Modeling Language (UML) (Abelló et al. 2002, Nguyen & Tjoa & Wagner 2000, Trujillo & Palomar & Gómez & Song 2001). For our purpose, we need a data model that supports model weaving (see below). We choose the object-oriented approach first presented by Trujillo et al. (2001) and further developed to a UML profile by Luján-Mora, Trujillo, and Song (2002).

A UML profile is a domain-specific extension to the UML modeling language (Object Management Group 2005). The profile used here adapts the UML class diagram for multi-dimensional modeling, i.e. the base class of the stereotypes is *Class*. Figure 4 shows the main elements of the Profile and their relationships as a metamodel: A *Fact* table can have any number of optional *Measures* and must have at least two *Dimensions* connected to it. Dimensions may be shared between facts and have one or more *Aggregations*, which form the aggregation hierarchy.

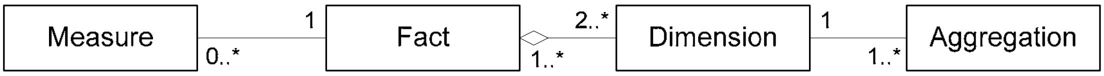

Figure 4 . *Core elements of the metamodel for multi-dimensional modeling (cf. Luján-Mora et al. 2002)*

There is no universally accepted, generic meta-model for multi-dimensional modeling. In this paper, we use the meta-model shown in Figure 4 in lieu of generic metamodel. The corresponding model in this case is a UML model. Yet, for our approach, the data model does not necessarily have to be a UML model. The only prerequisite for the data model is that its metamodel is available, and that it allows to model facts, dimensions and measures.

2.4 Model Weaving

For many modeling purposes, one large general “one size fits all” model is not advisable (Breton & Bézivin 2002). Creating several smaller, specific domain models allows for a better separation of concerns. Separate issues can be expressed in separate domain models, which are then related to each other by the use of model weaving, which was introduced by Del Fabro, Bézivin, Jouault, Breton and Gueltas (2005), as a linking mechanism.

Model weaving is an operation that links two or more (meta-)models (Bézivin & Jouault & Touzet 2005). The result is a weaving model containing links between elements from the involved models. The links may contain additional mapping information such as calculation formulas. Weaving models are models of the relationships between (other) models. They allow for easier handling of complex relationships between models. Weaving does not imply model or data transformation, but may be a prerequisite to these tasks. A weaving model is a “normal model” that can be stored and edited, accessed and analyzed with modeling tools. Among the application areas of model weaving are database schema matching, model transformation specification, visual programming, and ontology mapping (Bézivin & Jouault & Valduriez 2004).

Weaving works best if the participating (meta-)models are based on the same (meta-)meta-model. The metamodels in this paper are compliant to the Meta Object Facility (MOF) (Object Management Group 2003).

3 DERIVING BUSINESS METADATA

In this section, we present a weaving model for connecting enterprise models to data warehouse data, in order to derive business metadata. Business metadata allows the data warehouse users to access the context of the data warehouse data. The weaving model stores information about the relationship between the data warehouse and the structure and behavior of an enterprise organization, e.g., which business process or part of it impacts which part of the data warehouse.

Figure 5 shows the weaving model (strong lines) between the two existing metamodels (grey): On the left side the core model elements of a multidimensional data model, as described in Section 2.3, and on the right side a subset of the model elements of the ARIS Framework's metamodel, as described in Section 2.2. Both metamodels are actually larger, as hinted by the blended edges.

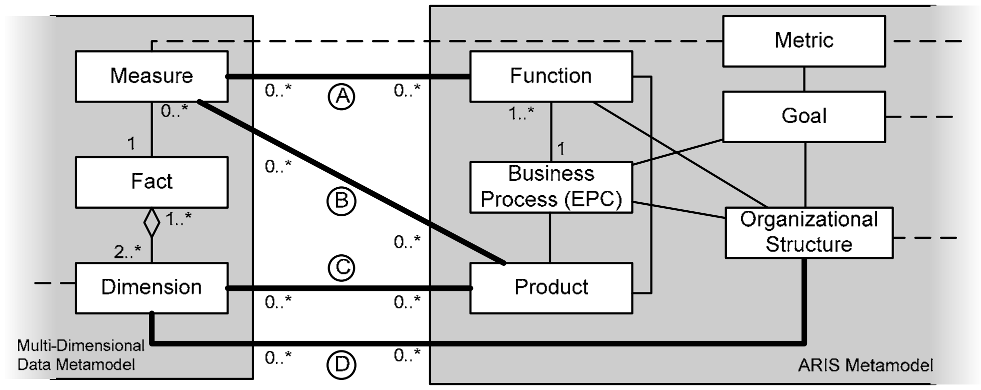

Figure 5 . *Connecting the data model to the enterprise model: A Weaving model with four links (strong lines). Only subsets of the models that are being connected by the weaving links are shown. Other links (dashed line) were introduced and discussed by the authors in Stefanov and List (2006).*

Between these two (meta-)models we have introduced four weaving links (strong lines, (A) – (D)). They store the relationships between the two domains and allow us to derive business metadata.

We use the knowledge already available in the organization in the form of enterprise models for business metadata. The business metadata helps to improve the understanding and interpretation of the data warehouse data by the users. For the viewpoint of the users, it is an additional level of abstraction on top of the data warehouse data. Business metadata should be aligned to the concepts that are well-known to the users, e.g. provide a business process-oriented view.

The weaving model makes formerly implicit relationships between the data in the data warehouse and the structure and behavior of the enterprise organization visible and accessible. The four weaving links shown in Figure 5 are described in the following.

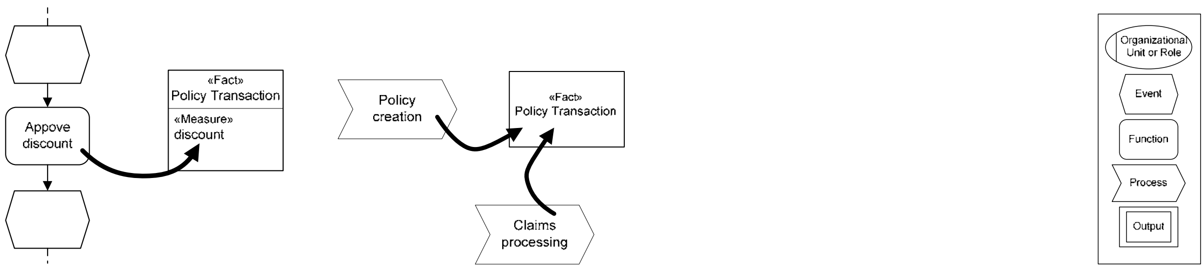

Figure 6. *A function creates the value of a measure in the data warehouse. Several processes impact on the values of a fact table.*

The measures of the data warehouse cubes are linked to functions (link A) and products (link B) in the ARIS model. Functions supply measure data to the data warehouse as they create or change values of business objects. Data in the "Policy Transaction" cube (see Figure 3 in Section 2.3) is created each time a process creates or changes the values of a policy. For example, the measure "discount" of the policy transaction cube is set by the function "Approve discount" (see Figure 2 in Section 2.2). Knowing which function supplies a measure implies knowing the overall business process to which the function belongs. The same applies to the fact cubes of measures. This transitive relationship allows to derive business metadata for analysis needs on different levels of detail (Figure 6). We can show which business process the data warehouse or which part of it impacts a certain measure.

Regarding products and other deliverables, their values can be found in two different places in the data warehouse: They are represented as measures (link B) as well as dimensional data (link C). For example, the insurance policy as a product will be present as a dimension of several fact cubes of the data warehouse (see Figure 7 for an illustration): The data on policy transactions can be aggregated e.g. by policy risk type. But the individual attributes of a policy, such as its premium or period (which may be different for each instance and change over time with each transaction) will be found in the measures.

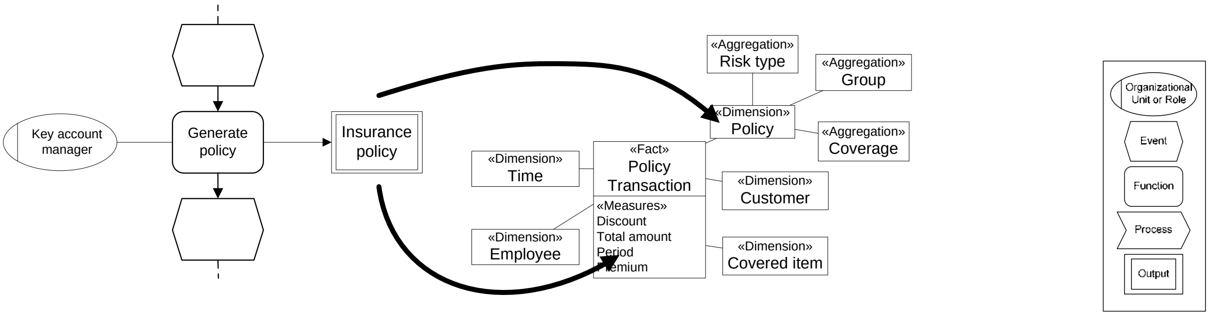

Figure 7. Products are represented in the data warehouse both as dimensional data and as measures.

Figure 8. Organizational roles are found in the data warehouse either directly in the dimensional data, or indirectly via other elements such as functions.

As illustrated in Figure 8, elements of the organizational structure, such as units or roles, may be mirrored in the data warehouse directly as dimensional data (link D), or indirectly through other elements such as the business process and functions (via link A) they are in charge of, their products (via links B and C), or goals. Directly as a dimension, the organization may appear as employees and/or departments. Indirectly, functions of business processes have organizational roles or units assigned to them, which can be evaluated through the measures recorded for these functions. The same applies to products or goals.

Figure 9, as a summary of the above, schematically illustrates the use of business metadata in Data Warehousing. The user accessing data on policy transactions has several questions: *Where do the values in this table come from? Which functions created them and where? What am I measuring here?*

The weaving links create the connections and show that the rows of the table are created by the business process *Policy creation* (1). The column *Policy ID* (2) refers to the *Insurance policy* which is the product of this process (3). If the users is interested in further details here, it can be shown that the insurance policy is actually the output of the function *Generate policy* within the process (4), and that the values in the column *Discount* are set by the function *Approve discount* of the same process (5). If the user is interested in the discount, more questions appear, such as *Who is responsible for that?* The user can see that the function that sets the value of *Discount* is performed by the organizational role *Key account manager* (6) which can also be found in the column *Employee ID* (7).

The table as it is shown here refers to the fact table “Policy Transaction” (see (8) and Figure 3 in Section 2.3). The columns are based on either measures (9) or dimensions (10).

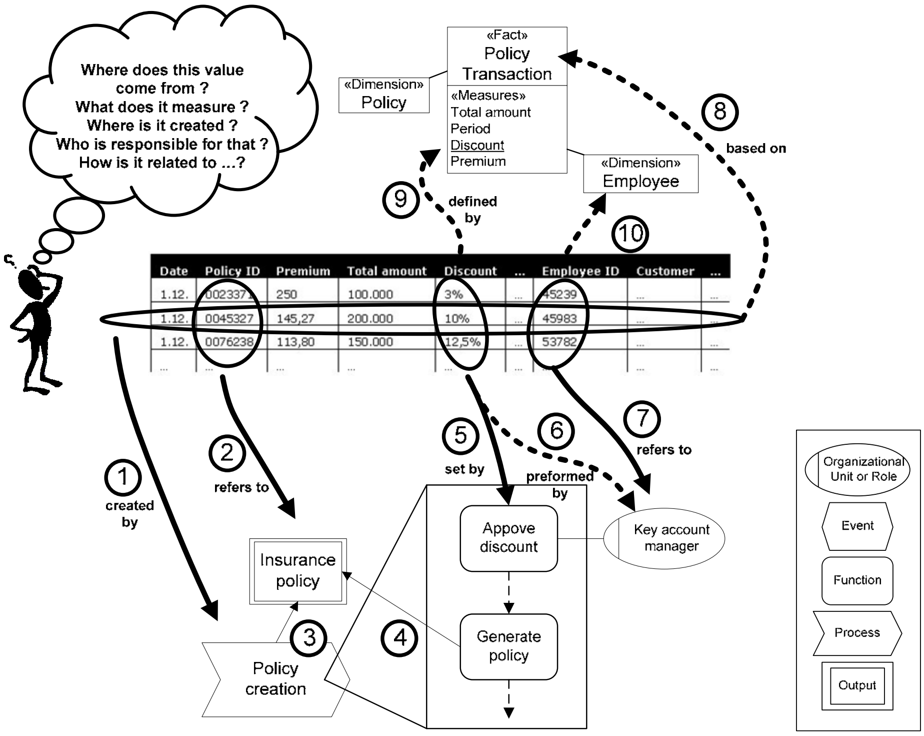

Figure 9. *Business metadata provides the data warehouse user with context and background information. Data items of the fact table are linked to the processes that produce them, the products they represent, or the organizational units involved.*

The knowledge captured by the weaving model can be exploited by analysis tools (also to offer better navigation or hints). Figure 10 shows how the business metadata can be displayed for the “policy transactions” cube introduced in Section 2.3. The organizational knowledge captured in the enterprise model becomes available to the user. Providing this information to the user directly within the analysis tool helps to improve data interpretation. The business metadata thus increases the usefulness of the data.

The business metadata is derived from the links shown in Figure 5. In the example in Figure 10, the user is browsing data on policy transactions, sorted by customer locations. One of the values of the measure “discount” is highlighted. The user can access a short textual summary of the metadata, as well as display a process diagram or an organizational chart with the current context highlighted.

The weaving links can also provide insights for data warehouse (re)design and maintenance, and requirements analysis. If the changes in the organizational context require changes in the data warehouse, weaving links can indicate where changes are necessary.

We are currently working on improving this prototype of a toolkit for the whole business metadata lifecycle. User are supported in choosing and integrating the enterprise models and data warehouse data models available to them. Then they create the instances of the weaving model that links the enterprise models to the data warehouse data. The weaving links are then used by a plug-in of the analysis tool to display the matching business metadata, as shown for example in Figure 10.

The prototype is based on the well-known Eclipse Rich Client platform and implemented in Java. It forms the basis for future case studies. Figure 11 gives an overview.

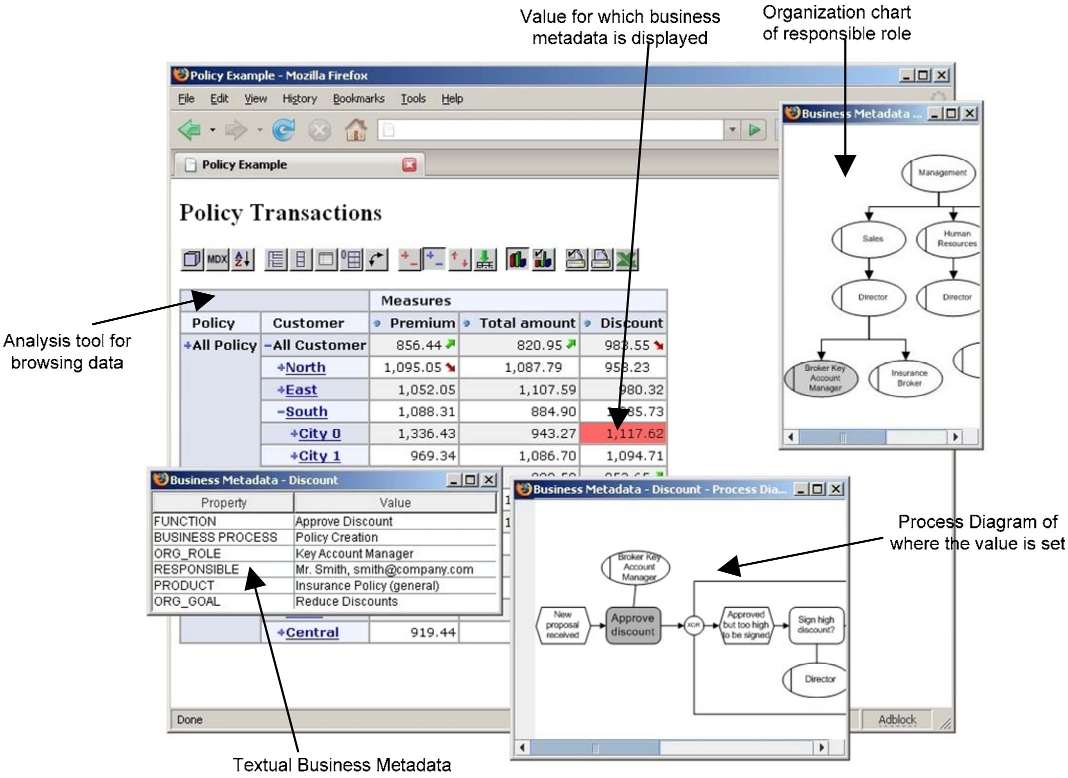

Figure 10. *Business metadata is displayed in form of text or diagrams.*

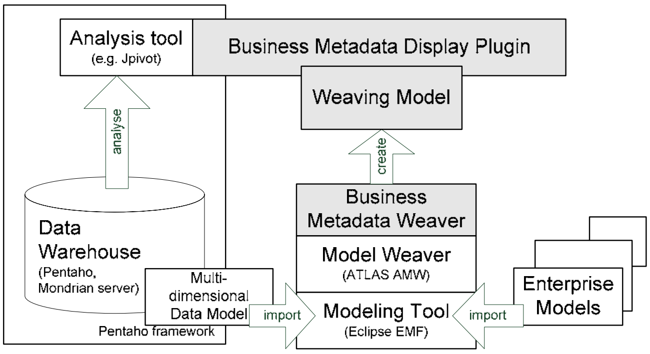

Figure 11. *Schematic description of the architecuture of the business metadata prototype.*

For the Data Warehousing part, it uses the open-source data warehouse platform Pentaho, which is partly built also on Eclipse and combines well-known DWH components such as the Mondrian server, JPivot for analysis, BIRT and JFreeReport as reporting tools, as well as data mining and dashboard components. In the screenshot shown in Figure 10, the business metadata is implemented as an extension of the JPivot library.

For modeling, the prototype uses the Eclipse Modeling Framework (EMF). The weaving links are created and managed with the ATLAS Model Weaver. All models are imported into EMF and stored in a model repository.

4 RELATED WORK

There are a lot of conceptual models available for business processes, data bases or data warehouses. But there are no models available that focus on the relationship between the data warehouse and the business processes. EPCs (Scheer 1999) incorporate a data view targeting operational data bases. EPC functions perform read or write operations on the databases and their entities. But they do not take the specific characteristics of data warehouses into account.

The Business Process Modeling Notation (BPMN 2004) provides data objects, which are used and updated during the process. The data object can be used to represent many different types of object, both electronic or physical.

An integrated view on Data Warehousing and business processes was introduced by Stefanov, List and Korherr (2005) in terms of a model that allows to show where and how a DWH is used by business processes, and which parts of the business processes depend on which parts of the DWH.

Mazon, Trujillo, Serrano and Piattini (2005) applied the MDA framework to data warehouse repository development, and aligned multidimensional conceptual data models with code. Speaking in MDA terms, they aligned a Platform Independent Model (PIM) with a Platform Specific Model (PSM) and defined a transformation between them. Our approach can be seen on top of this work targeting the Computational Independent Level (CIM) level, as we align enterprise context with the data warehouse conceptual data model.

Sarda (2001) linked data warehouse business metadata with technical metadata, in order to provide a better context for decision support. The business metadata is described with UML classes and associations and then linked directly to the technical metadata within the same model. The approach only covers metadata and reqires new separate metadata models.

The term “weaving” is also used in a different sense in aspect-oriented programming, where it denotes the integration of aspects into the base program (Kiczales 1997). See the AOSD Ontology by van den Berg, Conejero and Chitchyan (2005) for more general definitions that apply not only to the programming level, but also to modeling.

Breton and Bézivin (2002) apply model weaving to the area of workflow and process modeling. The build-time and the run-time workflow definitions are weaved together to create a binding between definition and execution of the process.

5 CONCLUSION

In this paper we have presented an approach to business metadata that is based on the relationship between the data warehouse data and the structure and behavior an enterprise organization. The information about the organization such as business processes and functions, organizational units and roles as well as the products produced by them, is taken from an enterprise model. Business metadata is derived by linking this knowledge about the organization to the data warehouse by means of a weaving model.

The business metadata can then be read directly from the weaving model. It improves data interpretation by explaining the relevance and context of the data, whereas the weaving model itself supports data warehouse requirements analysis, (re)design and evolution by making context visible and accessible. The approach is applied to an example.

References

Abelló, A., Samos, J., and Saltor, F. (2002). YAM2 (Yet Another Multidimensional Model): An Extension of UML. In IDEAS’02, pages 172–181. IEEE Computer Society.

van den Berg, K., Conejero, J. M., and Chitchyan R.. AOSD Ontology 1.0 - Public Ontology of Aspect-Orientation. AOSD-Europe-UT-01 D9, AOSD-Europe, May 2005.

ATLAS Model Weaver AMW (2006). http://www.eclipse.org/gmt/amw/ (15.11.2006)

Bézivin, J., Jouault, F., Touzet, D. (2005). An Introduction to the ATLAS Model Management Architecture. Technical report, LINA

Bézivin, J., Jouault, F., Valduriez, P. (2004). First Experiments with a Model Weaver. In: OOPSLA 2004.
Blaschka, M., Sapia, C., Höfling, G., and Dinter, B. (1998). Finding Your Way through Multidimensional Data Models. In DEXA ’98, pages 198–203.
Breton E. and Bézivin J. (2002). Weaving Definition and Execution Aspects of Process Meta-models. In 35th Hawaii Int. Conf. on System Sciences, page 290.
Business Process Modeling Notation (BPMN). (2004). BPMN Specification 1.0, http://www.bpmn.org
Chaudhuri, S. and Dayal, U. (1997). An Overview of Data Warehousing and OLAP Technology. SIGMOD Rec., 26(1):65–74.
Chen, P. (1976). The Entity-Relationship Model - Toward a Unified View of Data”, ACM Transactions on Database Systems, vol. 1, issue 1, pp 9-36.
Del Fabro, M.D., Bézivin, J., Jouault, F., Breton, E., Gueltas, G. (2005). AMW: A Generic Model Weaver. In: IDM’05 Ingénierie Dirigée par les Modéles.
Eclipse Modeling Framework (2006). http://www.eclipse.org/emf/ (15.11.2006)
Eclipse Platform (2006). http://www.eclipse.org/ (15.11.2006)
Frank, U. (2001). Multi-perspective Enterprise Modeling (MEMO) - Conceptual Framework and Modeling Languages. HICSS 2002: 72
Keller, G., Nüttgens, M., Scheer, A.-W. (1992). Semantische Prozeßmodellierung auf der Grundlage "Ereignisgesteuerter Prozeßketten (EPK)" (in German), In: Scheer, A.-W. (Hrsg.): Veröffentlichungen des Instituts für Wirtschaftsinformatik, Heft 89, Saarbrücken.
Kiczales G., Lamping J., Mendhekar A., Maeda C., Lopes C. V., Loingtier J.-M., and Irwin J. (1997). Aspect-Oriented Programming. In Proceedings of ECOOP 1997, LNCS 1241, p.220–242.
Kimball R., Reeves L., Ross M., and Thornthwaite W. (1998). The Data Warehouse Lifecycle Toolkit. John Wiley & Sons, Inc.
Kosanke, K., Vernadat, F. (2002). CIM-OSA: A Reference Architecture for CIM. PROLAMAT 1992: 41-48.
List, B. and Machaczek, K. (2004). Towards a Corporate Performance Measurement System. In: ACM SAC’04, ACM Press.
Luján-Mora, S., Trujillo, J., and Song, I.-Y. (2002). Extending the UML for Multidimensional Modeling. In UML ’02, pages 290–304. Springer Verlag.
Mazon, J.-N., Trujillo, J., Serrano, M. and Piattini, M. (2005). Applying MDA to the development of data warehouses. In Proceedings DOLAP ’05, pages 57–66, New York, NY, USA. ACM Press.
Nguyen, T. B., Tjoa, A. M., and Wagner, R. (2000). An Object Oriented Multidimensional Data Model for OLAP. In Web-Age Information Management (WAIM) 2000), pages 69–82. Springer.
Object Management Group (OMG) (2003). Meta Object Facility (MOF) 2.0 Core Specification. http://www.omg.org/cgi-bin/apps/doc?ptc/03-10-04.pdf (15.11.2006)
Object Management Group (OMG) (2005). UML 2.0 Superstructure http://www.omg.org/docs/formal/05-07-04.pdf (15.11.2006)
Pentaho Project (2006). Pentaho Open-Source Business Intelligence. http://www.pentaho.org
Sarda, N. L. (2001). Structuring Business Metadata in Data Warehouse Systems for Effective Business Support. CoRR, http://arxiv.org/abs/cs.DB/0110020
Scheer, A.-W. (1999). ARIS - Business Process Modeling. Springer Verlag.
Stefanov, V and List, B (2006). Business Metadata for the Data Warehouse - Weaving Enterprise Goals and Multidimensional Models, Proceedings IWMEC Workshop at EDOC 2006, Hong Kong
Stefanov, V., List, B. and Korherr, B. (2005). Extending UML 2 Activity Diagrams with Business Intelligence Objects, Proceedings DaWaK 2005, LNCS 3589, Springer Verlag
Trujillo, J., Palomar, M., Gómez, J., and Song, I.-Y. (2001). Designing Data Warehouses with OO Conceptual Models. IEEE Computer, 34(12):66–75.
Vassiliadis, P. and Sellis, T. K. (1999). A Survey of Logical Models for OLAP Databases. SIGMOD Record, 28(4):64–69.
Whitman, L., Ramachandran, K., Ketkar, V.(2001). A taxonomy of a living model of the enterprise. In: WSC’01, IEEE Computer Society, 848–855