A Rule-based Data Warehouse Model

Cécile Favre, Fadila Bentayeb, and Omar Boussaid

ERIC Laboratory, University of Lyon 2,
5 av. Pierre Mendès-France
69676 Bron Cedex, France
{cfavre,bentayeb}@eric.univ-lyon2.fr
omar.boussaid@univ-lyon2.fr

Abstract. A data warehouse is built by collecting data from external sources. Changes that occur have to be reflected in the data warehouse thanks to schema updating or versioning. However a data warehouse has also to evolve according to users' analysis needs. This evolution is rather driven by knowledge than by data. To take into account these changes, we propose a new Rule-based Data Warehouse (R-DW) model in which rules integrate users' knowledge to dynamically create dimension hierarchies. The R-DW model is composed of a fixed part which is a fact table related to its first level dimensions, and an evolving part which contains the rules. Our model allows analysis context evolution and increases interactions between users and the decision support system.

1 Introduction

A data warehouse constitute an effective support in managing an increasing mass of data from heterogeneous sources and in providing an answer to analysis needs. Exhaustively establishing the users' needs is a complex task. Sometimes, users have knowledge which is not represented in the data warehouse and which may be needed to have a complete analysis. We therefore have to make the data warehouse evolve to be able to take into account users' knowledge. But this kind of evolution is not easy to achieve in the traditional rigid data warehouse models.

Indeed, a data warehouse schema evolution is performed following two different ways, namely schema updating and schema versioning. The first approach consists in updating a schema and transforming data from an old schema into a new one [1]. The second approach, on the contrary, keeps track of all versions of a schema [2]. These two approaches constitute a solution to the dimensions evolution, when the latter is induced by the evolution of data. However, once the data warehouse made up, the users may only carry out analysis provided by the model. These approaches do not take into account new analysis needs driven by the expression of knowledge ; thus they are not flexible enough.

Works aiming at an increased flexibility in data warehouses generally use rules to either define the data warehouse schema from source schemas [3, 4] or support various types of constraints [5] in order to keep the data and the analysis coherent, or manage the exceptions during the aggregation process [6]. The rule-based languages contributed in making the data warehouse more flexible. We

want to bring such a flexibility for the evolution of analysis needs driven by knowledge.

Thus, we propose a new Rule-based Data Warehouse (R-DW) model, in which rules create new granularity levels in dimension hierarchies by integrating the users' knowledge. Our R-DW model has several advantages as compared to existing data warehouse models. It allows (1) to dynamically create hierarchies ; (2) to make analysis on evolving contexts ; (3) to increase the interaction between users and the information system since they can integrate their own knowledge.

Section 2 explains principles of our R-DW model and an example is described in Section 3. Its implementation and a case study are presented in Section 4. Section 5 concludes this paper and provides future research directions.

2 The R-DW model

The R-DW model is composed of a fixed part and an evolving part (Figure 1a). The fixed part is a star schema, it is composed of a fact table and first level dimensions. The evolving part is composed of rules which generate new granularity levels in dimension hierarchies based on the user's knowledge.

The metamodel described in Figure 1b is a generalization of the R-DW model. It contains a definition of the fixed part of the model represented by the *Fact table* and *Dimension* classes. The evolving part is represented by *Rule defined extensionally* and *Rule defined intentionally* classes. Rules are defined extensionally when they are based on well-known values that can be enumerated ; on contrary, they can be defined intentionally using functions.
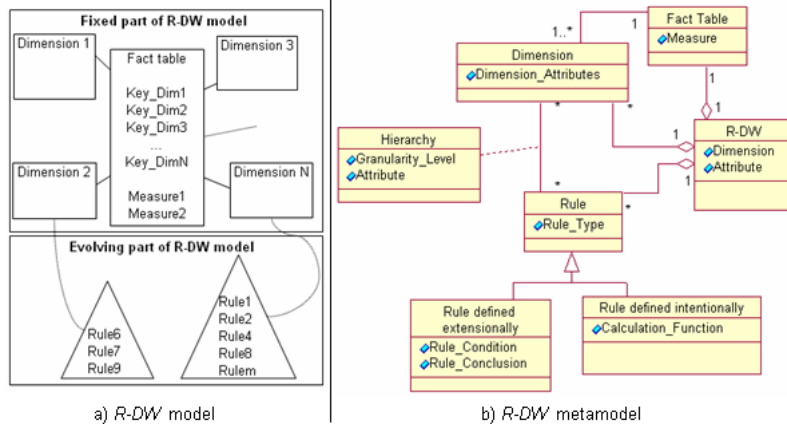

Fig. 1. R-DW model and metamodel

Rules defined extensionally are "if-then" rules. The then-clause contains the definition of a higher granularity level. The if-clause contains conditions on the lower granularity levels. Rules defined intentionally allow inferences on the granularity level according to lower levels. For instance, to obtain the department of a customer, we just have to extract the first two characters of the postal code.

Our model provides to the end-users a mean to define their own rules to determine new dimensions hierarchies. Then users can therefore analyze data according to the new granularity levels. The data warehouse model becomes thus more flexible for analysis.

3 Example

As an example, we use the case study of Le Crédit Lyonnais (LCL). The annual Net Banking Income (NBI) is the profit obtained from the management of customers account. This is a measure studied according to customers, agencies and years. The students portfolio manager of LCL knows that some agencies have students as only customers. Our *R-DW* model (Figure 2) can integrate this knowledge to carry out an analysis taking into account the student dedicated agencies. The fixed part of the model is made up of the fact table *TF_NBI* and the dimension tables *CUSTOMER*, *YEAR* and *AGENCY*. To the fixed part we add the evolving part containing rules that describe the manager's knowledge on student agencies (R1 and R2).The induced model makes it possible to carry out new analysis based on the user's knowledge. A new granularity level has been added in the hierarchy. Therefore our *R-DW* model allows us to build aggregates, by considering that the facts to aggregate concern a student agency (R1), or a classical agency (R2).
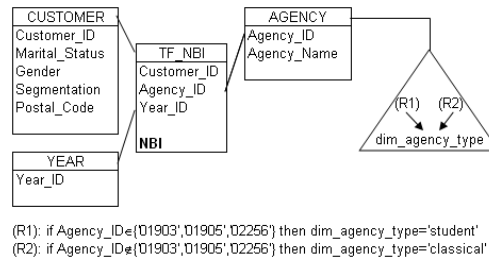

(R1): if Agency_ID∈{'01903','01905','02256'} then dim_agency_type='student'
(R2): if Agency_ID∉{'01903','01905','02256'} then dim_agency_type='classical'

Fig. 2. Rule-based Data Warehouse model for the NBI analysis

4 Implementation and case study

We developed a Web platform (HTML/PHP) behind Oracle DBMS used to store the fact table and dimension tables. Two additional tables contain the rules defined extensionally and intentionally. The Web platform allows the user to define the rules that generate the analysis axes. We initially restricted ourselves to decisionnal queries with an aggregation according to one granularity level. This aggregation is computed by a PL/SQL stored procedure.

The *R-DW* model for the NBI analysis previously presented is enriched by the granularity levels which are created by the evolving part of the Figure 3. From this model, the user will be able to run NBI analyses, not only by considering first level dimensions, but also by considering new dimension levels like agency type, department, age classes...

Fig. 3. Evolving part of the *R-DW* data warehouse for the NBI analysis

5 Conclusion

We proposed a rule-based data warehouse model. Rules introduce users' knowledge into the data warehouse for new analysis purposes. Our *R-DW* model is composed of two parts: a fixed part that contains a fact table and first level dimensions ; and an evolving part that is defined by rules defining granularity levels of dimension hierarchies. The *R-DW* model presents the advantage of being able to dynamically create dimension hierarchies, according to users' knowledge therefore satisfying their analysis needs. The implementation we developed was applied on the LCL case study and gave some promising results.

The perspectives opened by this study are numerous. First, we have to extend the analysis possibilities of our implementation. Then we intend to measure the performance of our approach in terms of storage space and response time. Furthermore we plan to define constraints on rules and a language that allows us to validate these rules. Moreover, we think it would be interesting to use non supervised learning methods for discovering new rules.

References

1. Blaschka, M., Sapia, C., Höfling, G.: On Schema Evolution in Multidimensional Databases. In: DaWaK'99: 1st International Conference on Data Warehousing and Knowledge Discovery. (1999) 153–164
2. Eder, J., Koncilia, C.: Changes of dimension data in temporal data warehouses. In: DaWaK'01: 3rd International Conference on Data Warehousing and Knowledge Discovery. (2001) 284–293
3. Kim, H.J., Lee, T.H., Lee, S.G., Chun, J.: Automated Data Warehousing for Rule-Based CRM Systems. In: 14th Australasian Database Conference on Database Technologies. (2003) 67–73
4. Peralta, V., Illarze, A., Ruggia, R.: On the Applicability of Rules to Automate Data Warehouse Logical Design. In: CAiSE Workshops. (2003)
5. Carpani, F., Ruggia, R.: An Integrity Constraints Language for a Conceptual Multidimensional Data Model. In: SEKE'01: XIII International Conference on Software Engineering Knowledge Engineering. (2001)
6. Espil, M.M., Vaisman, A.A.: Efficient Intensional Redefinition of Aggregation Hierarchies in Multidimensional Databases. In: DOLAP'01: 4th ACM International Workshop on Data Warehousing and OLAP. (2001)