Chapter II

Design and Representation of Multidimensional Models with UML and XML Technologies

Juan Trujillo, Universidad de Alicante, Spain
Sergio Luján-Mora, Universidad de Alicante, Spain
Il-Yeol Song, Drexel University, USA

Data warehouses (DW), multidimensional databases (MDB), and OnLine Analytical Processing (OLAP) applications are based on the Multidimensional (MD) modeling. Most of these applications provide their own MD models to represent main MD properties, thereby making the design totally dependent of the target commercial application. In this chapter, we present how the Unified Modeling Language (UML) can be successfully used to abstract the representation of MD properties at the conceptual level. Then, from this conceptual model, we generate its corresponding implementation into any market OLAP tool. In our approach, the structure of the system is specified by means of a UML class diagram that considers main properties of MD modeling. If the system to be modeled is too complex, we describe how to use the package grouping mechanism provided by the UML to simplify the final model. To facilitate the interchange of conceptual MD models, we provide an eXtensible Markup...
Language (XML) Schema which allows us to represent the same MD modeling properties that can be considered by using our approach. From this XML Schema, we can directly generate valid XML documents that represent MD models at the conceptual level. Finally, we provide different presentations of the MD models by means of eXtensible Stylesheet Language Transformations (XSLT).

**INTRODUCTION**

Multidimensional (MD) modeling is the foundation for Data warehouses (DW), multidimensional databases (MDB), and OnLine Analytical Processing (OLAP) applications. The benefit of using MD modeling is twofold. On one hand, the MD model is close to data analyzers’ ways of thinking; therefore, it helps users understand data. On the other hand, the MD model supports performance improvement as its simple structure allows us to predict final users’ intentions.

Some approaches have been proposed lately (presented in the Related Work section) to accomplish the conceptual design of these systems. Unfortunately, none of them have been accepted as a standard for DW conceptual modeling. These proposals try to represent main MD properties at the conceptual level with special emphasis on MD data structures. A conceptual modeling approach for DW, however, should also concern other relevant aspects, such as users’ initial requirements, the behavior of the system (e.g., main operations to be accomplished on MD data structures), available data sources, specific issues for automatic generation of the database schema, and so on. We claim that object orientation with the UML provides an adequate notation for modeling every aspect of a DW system (MD data structures, the behavior of the system, etc.) from user requirements to implementation.

We have previously proposed an object-oriented (OO) approach to accomplish the conceptual modeling of DW, MDB, and OLAP applications that introduces a set of minimal constraints and extensions of the UML (Booch, Rumbaugh & Jacobson, 1998; OMG, 2001) needed for an adequate representation of MD modeling properties (Trujillo, 2001; Trujillo et al., 2001). These extensions are based on the standard mechanisms provided by the UML to adapt it to a specific method or model (e.g., constraints, tagged values). We have also presented how to group classes into packages to simplify the final model in case that the model becomes too complex due to the high number of classes (Luján-Mora, Trujillo & Song, 2002). Furthermore, we have provided a UML-compliant class notation to represent OLAP users’ initial requirements (called cube class). We have also discussed issues, such as identifying attributes and descriptor attributes that set the basis for an adequate semi-automatic generation of a database schema and user requirements in a target commercial OLAP tool.