ABSTRACT

Multidimensional (MD) modeling is the basis for data warehouses (DW), multidimensional databases (MDB) and on-line analytical processing (OLAP) applications. In this paper, we present how the unified modeling language (UML) can be successfully used to represent both structural and dynamic properties of these systems at the conceptual level. The structure of the system is specified by means of a UML class diagram that considers the main properties of MD modeling with minimal use of constraints and extensions of the UML. If the system to be modeled is too complex, thereby leading us to a considerable number of classes and relationships, we describe how to use the package grouping mechanism provided by the UML to simplify the final model. Furthermore, we provide a UML-compliant class notation (called cube class) to represent OLAP users’ initial requirements. We also describe how we can use the UML state and interaction diagrams to model the behavior of a data warehouse system. To facilitate the interchange of conceptual MD models, we provide a Document Type Definition (DTD) which allows us to represent the same MD modeling properties that can be considered by using our approach. From this DTD, we can directly generate valid eXtensible Markup Language (XML) documents that represent MD models at the conceptual level. We believe that our innovative approach provides a theoretical foundation for simplifying the conceptual design of MD systems and the examples included in this paper clearly illustrate the use of our approach.

Keywords: data warehouses; multidimensional databases; OLAP; conceptual modeling; UML; object orientation; ODBMS; XML

INTRODUCTION

Multidimensional (MD) modeling is the foundation for data warehouses (DW), multidimensional databases (MDB) and on-line analytical processing (OLAP) applications. The benefit of using MD modeling is two-fold. On one hand, the MD model is close to data analyzers’ way 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 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 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 (Unified Modeling Language) (Booch, Rumbaugh, and Jacobson, 1998; OMG, 2001), needed for an adequate representation of MD modeling properties (Trujillo, 2001; Trujillo, Palomar, Gómez, and Song, 2001b). 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 the model becomes too complex due to the high number of classes (Luján-Mora, Trujillo, and Song, 2002). Furthermore, we have provided a UML-compliant class notation to represent OLAP users’ initial requirements (called cube class). From these cube classes, we then describe the use of state and interaction diagrams to model the behavior of the system based on the applied OLAP operations (Trujillo, Palomar, and Gómez, 2000). 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.

The UML can also be used with powerful mechanisms such as the Object Constraint Language (OCL) (Warmer and Kleppe, 1998; OMG, 2001) and the Object Query Language (OQL) (Cattell et al., 2000) to embed DW constraints (e.g. additivity and derived attributes) and users’ initial requirements in the conceptual model. In this way, when we model a DW system, we can obtain simple yet powerful extended UML class diagrams that represent main MD properties at a conceptual level.

On the other hand, a salient issue these days in the scientific community and in the business world is the interchange of information. The eXtensible Markup Language (XML) (W3C, 2000) is rapidly being adopted as the standard syntax for the interchange of un-structured, semi-structured and structured data. XML is an open neutral platform and vendor independent meta-language, which allows us to reduce the cost, complexity, and effort required in integrating data within and between enterprises.

From these considerations, in this paper we present the following contributions. We believe that our innovative approach provides a theoretical foundation for the possible use of Object-Oriented Databases (OODB) and Object-Relational Databases (ORDB) for DW and OLAP applications. For this reason, we provide the representation of our approach into the standard for OODB proposed by the Object Database Management Group (ODMG) (Catell et al. 2000). We also believe that a relevant
Related Content

Modeling and Implementing Scientific Hypothesis
[www.igi-global.com/article/modeling-and-implementing-scientific-hypothesis/142069?camid=4v1a](www.igi-global.com/article/modeling-and-implementing-scientific-hypothesis/142069?camid=4v1a)

MDD Approach for Maintaining Integrity Constraints in Databases
[www.igi-global.com/chapter/mdd-approach-maintaining-integrity-constraints/20698?camid=4v1a](www.igi-global.com/chapter/mdd-approach-maintaining-integrity-constraints/20698?camid=4v1a)
An Overview on Signature File Techniques
www.igi-global.com/chapter/overview-signature-file-techniques/20750?camid=4v1a

Database Pointers in Navigational and Object–Oriented Database Management Systems: A Comparison
www.igi-global.com/article/database-pointers-navigational-object-oriented/51155?camid=4v1a