OLAP Presentation Modeling with UML and XML

Andreas S. Maniatis

Knowledge and Database Systems Laboratory (KDBSL)
Department of Electrical and Computer Engineering
National Technical University of Athens (NTUA)
15780 Athens, Greece
Email: andreas@dblab.ece.ntua.gr

Abstract. On-Line Analytical Processing (OLAP) is a trend in database technology, based on the multidimensional view of data. Multidimensional data (called multicubes) form the basic logical data model for OLAP applications and a significant number of proposals exist for cube models, without – in most cases – the distinct description of an equivalent "presentational" layer or model for the presentation of multidimensional data and their hierarchies. In this paper we show how a formal presentation model for multidimensional data, namely the *Cube Presentation Model (CPM)* can be modeled using an extension of UML, by means of stereotypes, to more elegantly represent the most common presentational aspects of OLAP. Further more, we exploit the eXtensible Markup Language (XML) to capture the necessary semantics of the presentation model that can be used in OLAP visualization applications. Finally, we demonstrate all these using Rational Rose.

1 Introduction

In the last years, On-Line Analytical Processing (OLAP) and data warehousing has become a major research area in the database community [1, 2, 3, 4, 5, etc]. A fundamental issue faced by vendors of OLAP tools as well as by researchers in the OLAP domain is the *modeling of data*. Vendors have adopted various models, while standardization bodies and researchers have developed and studied additional ones. All those models share some common concepts like measures, dimensions and dimension hierarchies, but there is still no formally defined and widely accepted (logical or conceptual) data model. An overview and comparison of the previously referenced models can be found in [6, 7, 8].

A second, equally important issue faced by vendors, researchers and, mainly, users of OLAP applications is the *visualization of data*. Presentational models are not really a part of the classical conceptual-logical-physical hierarchy of database models; nevertheless, since OLAP is a technology facilitating decision-making, the presentation of data is of obvious importance. Research-wise, data visualization is presently a quickly evolving field and dealing with the presentation of vast amounts of data to the users [9, 10, 11].

As of today, the Unified Modeling Language (UML) has been widely accepted as the standard object oriented modeling language for practically every aspect of a soft-

ware system and for multidimensional modeling as well, as is elegantly demonstrated in [5, 12, 13]. Especially important are the extensibility attributes of UML, in the sense that it incorporates the needed mechanisms (stereotypes, tagged values and constraints) that facilitate the definition and introduction of new, domain specific elements and notions [14, 15].

In this paper we demonstrate how a newly introduced presentation model for OLAP *(CPM – Cube Presentation Model)* can be modeled using specifically designed UML extensions so that presentational schemas and models can be more easily perceived by designers and programmers and mapping to common visualization techniques can be more straightforward and comprehensive. We will then show how these extensions can be incorporated into Rational Rose [16], to augment their applicability and – finally – we use the eXtensible Markup Language (XML) [22] to store the respective CPM models so that the semantics of each presentation can be used in various OLAP visualization applications.

The remainder of this paper is structured as follows: In Section 2, we informally and briefly present the logical and the presentation layers of CPM. In Section 3, we introduce the necessary UML extensions, specific to the CPM domain model and demonstrate how these extensions can be defined and used in Rational Rose. In Section 4 we briefly show how XML can be effortlessly used to capture and store the semantics of a specific multidimensional presentation model, for use in visualization applications and finally, in Section 5 we conclude our results and present topics for future work.

2 The Cube Presentation Model

The *Cube Presentation Model (CPM)* [17] is a novel proposal towards a discrete and autonomous presentation model for OLAP screens, intuitively based on the geometrical representation of a cube and its human perception in space. CPM brings together and formalizes terms and notions from existing proposals. The first is from the industrial field, where Microsoft has issued a commercial standard for multidimensional databases and where the presentational issues form a major part [18]. In this approach, a powerful query language is used to provide the user with complex reports, created from several cubes (or actually subsets of existing cubes). The second is an academic approach, the Tape Model [11], based on the notion of "Tapes", called thus due to their look and feel. Tapes are infinite and can overlap (if they contain shared data dimensions), or intersect with each other. A two dimensional intersection is called a matrix and represents a kind of cross-tab between the corresponding dimensions. Each tape comprises of a variable number of *tracks*. The most important operations on tapes include: (a) insertion and deletion of tracks, (b) changing the sequence of tracks (i.e., sorting) and (c) scrolling on tracks. The model offers the possibility of defining *hierarchical structures* within a tape.

The *Cube Presentation Model (CPM)* is composed of two parts: (a) a *logical* layer which involves the formulation of cubes and (b) a *presentational layer* that involves the presentation of these cubes (normally, on a 2D screen). The main idea behind CPM lies in the separation of *logical data retrieval* (which we encapsulate in the

logical layer of CPM) and *data presentation* (captured from the presentational layer of CPM). This duality provides the flexibility of possibly replacing one of the two layers with an alternative proposal smoothly. The logical layer that we propose is based on an extension of a previous proposal [20] to incorporate more complex cubes. Replacing the logical layer with any other model compatible to classical OLAP notions (like dimensions, dimension hierarchies and cubes) can be easily performed. The presentational layer, at the same time, provides a formal model for OLAP screens. To our knowledge, there is no such result in the related literature.

The logical layer of the Cube Presentation Model

In this section, we present the logical layer of CPM; to this end, we extend a logical model for OLAP, in order to compute more complex cubes. We briefly repeat the basic constructs of the logical model and refer the interested reader to [20] and to Appendix A of this paper for a detailed presentation of this part of the model. The most basic constructs are:

- A dimension is a lattice of *dimension levels* $(\mathbf{L}, \prec)$, where $\prec$ is a partial order defined among the levels of $\mathbf{L}$.
- A family of monotone, pairwise consistent *ancestor functions* $anc_{L_1}^{L_2}$ is defined, such that for each pair of levels L_1 and L_2 with $L_1 \prec L_2$, the function $anc_{L_1}^{L_2}$ maps each element of $dom(L_1)$ to an element of $dom(L_2)$.
- A *data set* DS over a schema $S=[L_1,\ldots,L_n,A_1,\ldots,A_m]$ is a finite set of tuples over S such that $[L_1,\ldots,L_n]$ are levels, the rest of the attributes are *measures* and $[L_1,\ldots,L_n]$ is a primary key. A *detailed data set* DS^0 is a data set where all levels are at the bottom of their hierarchies.
- A *selection condition* φ is a formula involving atoms and the logical connectives $\wedge$, $\vee$ and $\neg$. The atoms involve levels, values and ancestor functions, in clause of the form $x \partial y$. A *detailed selection condition* involves levels at the bottom of their hierarchies.
- A *primary cube* c (over the schema $[L_1,\ldots,L_n,M_1,\ldots,M_m]$), is an expression of the form $c=(DS^0,\varphi,[L_1,\ldots,L_n,M_1,\ldots,M_m],[agg_1(M_1^0),\ldots,agg_m(M_m^0)])$, where:
  - DS^0 is a detailed data set over the schema $S=[L_1^0,\ldots,L_n^0,M_1^0,\ldots,M_k^0]$, $m \le k$.
  - φ is a detailed selection condition.
  - $M_1, \ldots, M_m$ are measures.
  - L_i^0 and L_i are levels such that $L_i^0 \prec L_i$, $1 \le i \le n$.
  - $agg_i \in \{sum,min,max,count\}$, $1 \le i \le m$.

  The limitations of primary cubes is that, although they accurately model SELECT-FROM-WHERE-GROUPBY queries, they fail to model (a) ordering, (b) computation of values through functions and (c) selection over computed or aggregate values (i.e., the HAVING clause of a SQL query). To compensate this shortcoming, we extend the aforementioned model with a certain number of auxiliary entities (see Appendix A) that assist in the definition of the following construct:

– A *secondary cube* over the schema $S=[L_1,\ldots,L_n,M_1,\ldots,M_m,A_{m+1},\ldots,A_{m+p},\ RANK]$ is an expression of the form: $s=[c,[A_{m+1}:f_{m+1}(\mathbf{A}_{m+1}),\ldots,A_{m+p}:f_{m+p}(\mathbf{A}_{m+p})],O_{\mathbf{A}}^{\theta},\psi]$ where $c=(DS^0,\varphi,\ [L_1,\ldots,L_n,M_1,\ldots,M_m],[agg_1(M_1^0),\ldots,agg_m(M_m^0)])$ is a primary cube, $[\mathbf{A}_{m+1},\ldots,\mathbf{A}_{m+p}]\subseteq[L_1,\ldots,L_n,M_1,\ldots,M_m]$, $\mathbf{A}\subseteq S-\{RANK\}$, $f_{m+1},\ldots,f_{m+p}$ are functions belonging to $\mathbf{F}$ and ψ is a secondary selection condition. With these additions, primary cubes are extended to secondary cubes that incorporate: (a) computation of new attributes ($\mathbf{A}_{m+i}$) through the respective functions ($\mathbf{f_{m+i}}$), (b) ordering ($O_{\mathbf{A}}^{\theta}$) and (c) the HAVING clause, through the secondary selection condition ψ.

The presentational layer of the Cube Presentation Model

In this section, we give an intuitive and informal description of the *presentation layer* of CPM. To make the discussion easier, we present the full metamodel of the presentation layer in Fig. 1, using "flat" UML modeling.

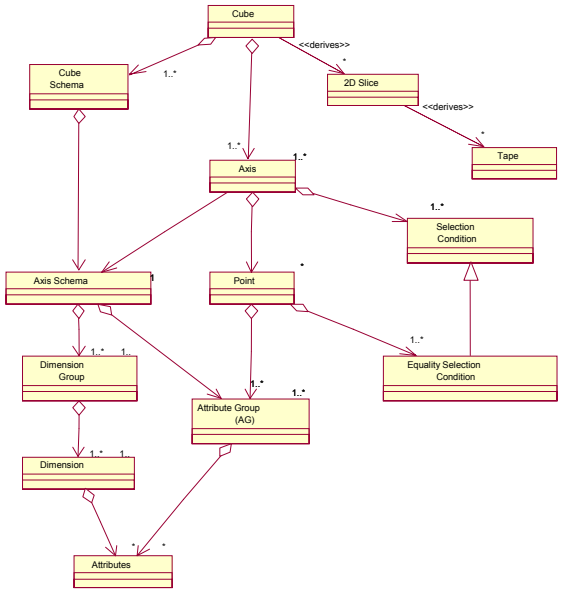

Fig. 1: A "flat" UML Metamodel for the presentational layer of CPM

The most important entities of the logical layer of CPM include:

– **Points:** A *point over an axis* resembles the classical notion of points over axes in mathematics and geometry. Still, since we are grouping more than one attribute per axis (in order to make things presentable in a 2D screen), formally, a point is a pair comprising of a set of attribute groups (with one of them acting as primary key) and a set of equality selection conditions for each of the keys.

– **Axis:** An axis can be viewed as a set of points. We introduce two special purpose axes, Invisible and Content. The Invisible axis is a placeholder for the levels of the data set which are not found in the "normal" axis defining the multicube.

The Content axis has a more elaborate role: in the case where no measure is found in any axis then the measure which will fill the content of the multicube is placed there.

- **Multicubes.** A multicube is a set of axes, such that (a) all the levels of the same dimensions are found in the same axis, (b) Invisible and Content axes are taken into account, (c) all the measures involved are tagged with an aggregate function and (d) all the dimensions of the underlying data set are present in the multicube definition. In our motivating example, the multicube MC is defined as MC={Rows,Columns,Sections,Invisible,Content}.
- **2D-slice:** Consider a multicube MC, composed of K axes. A *2D-slice over MC* is a set of (K-2) points, each from a separate axis. Intuitively, a 2D-slice pins the axes of the multicube to specific points, except for 2 axes, which will be presented on the screen (or a printout). In Fig. 2, we depict such a 2D slice over a multicube.
- **Tape:** Consider a 2D-slice SL over a multicube MC, composed of K axes. A *tape over SL* is a set of (K-1) points, where the (K-2) points are the points of SL. A tape is always parallel to a specific axis: out of the two "free" axes of the 2D-slice, we pin one of them to a specific point which distinguishes the tape from the 2D-slice.
- **Cross-join:** Consider a 2D-slice SL over a multicube MC, composed of K axes and two tapes t_1 and t_2 which are not parallel to the same axis. A *cross-join over* t_1 *and* t_2 is a set of K points, where the (K-2) points are the points of SL and each of the two remaining points is a point on a different axis of the remaining axes of the slice.

We do not focus the contribution of the paper on the extension of the underlying logical model or CPM itself; still, we refer the interested reader to the full version of CPM for more intuition and rigorous definitions [19].

3 CPM Modeling through UML Extensions

UML provides many mechanisms – internal to its structure – that allow for the definition and introduction of domain specific new elements and entities, suitable for more comprehensive modeling of these domains. These mechanisms include *stereotypes*, *tagged values* and *constraints* [15]. As previous proposals have successfully demonstrated [5, 12, 13], UML is suitable for general multidimensional modeling, as a natural extension to database and persistent modeling. In this section we focus in demonstrating how UML can be properly extended for OLAP presentational modeling as well.

UML extensions for CPM

We have defined a number of stereotypes to represent the most notable and CPM specific elements as they are depicted in the CPM metamodel of Fig. 2, presented using a 'flat' UML notation which makes it difficult to read and understand by readers non-familiar to the domain under examination. All of them are a specialization of

the general Class model element and are visualized using specifically drawn icons to resemble their notion of the element they represent. Fig. 2 lists all the icons used to represent the respective stereotypes.

(a) Cube (b) Dimension (c) Axis (d) Tape (e) 2D-Slice (f) Point

Fig. 2: Stereotype icons for CPM modeling.

We adhere to the rules and examples for the definition of stereotypes as described in [15] and for the sake of completeness and comprehensibility, we include some hints of the extension definitions:

- Name: **Cube**
- Base class: Class
- Description: Represents the notion of the CPM entity "*Cube*"
- Icon: Figure 2 (a)
- Constraints: None
- Tagged Values: None

- Name: **Dimension**
- Base class: Class
- Description: Represents the notion of the CPM entity "*Dimension*"
- Icon: Figure 2 (b)
- Constraints: None
- Tagged Values: None

- Name: **Axis**
- Base class: Class
- Description: Represents the notion of the CPM entity "*Axis*"
- Icon: Figure 2 (c)
- Constraints: None
- Tagged Values: None

- Name: **Tape**
- Base class: Class
- Description: Represents the notion of the CPM entity "*Tape*"
- Icon: Figure 2 (d)
- Constraints: None
- Tagged Values: None

- Name: **2D-Slice**
- Base class: Class
- Description: Represents the notion of the CPM entity "*2D-Slice*"
- Icon: Figure 2 (e)
- Constraints: None
- Tagged Values: None

- Name: **Point**
- Base class: Class
- Description: Represents the notion of the CPM entity "*Point*"

- Icon: Figure 2 (f)
- Constraints: None
- Tagged Values: None

CPM Modelling using Rational Rose

To demonstrate the applicability of the previously described notions on real case implementations, we have used Rational® Rose, one of the most well-known and widely used modeling tools. It embeds all the necessary functionality to implement the UML stereotype extensions for CPM previously defined, through specific features of the Rose Extensibility Interface [16], allowing thus for the domain specific "adaptation" of the tool and the usage of more comprehensive elements for OLAP presentational modeling. We have developed the add-in that customizes stereotypes – by means of a stereotype configuration file – to use Rational® Rose for the design of OLAP presentation models.

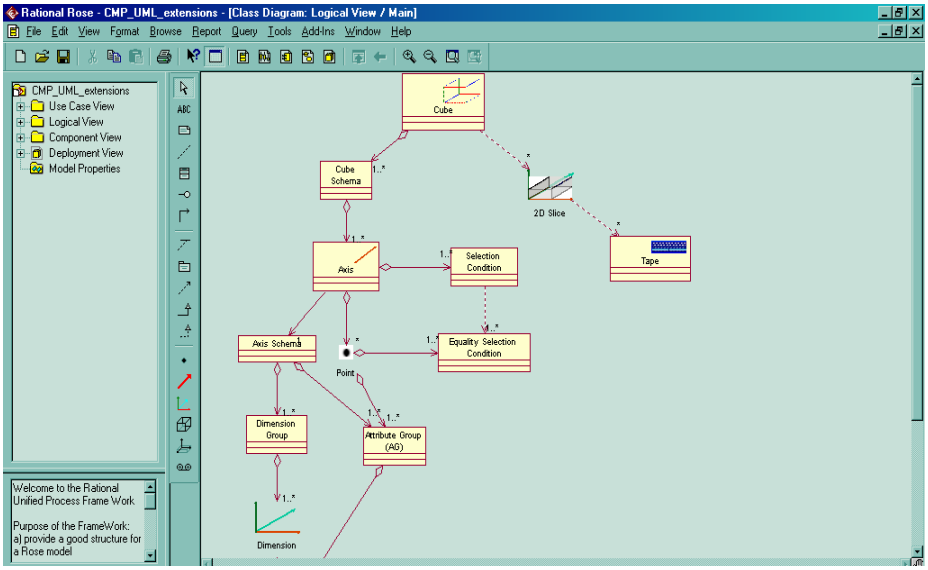

Fig. 3: The CPM metamodel of Fig. 1 using stereotypes in Rational Rose

Fig. 3 shows the enhanced CPM metamodel of Fig. 1. It is quite clear in this example that the model has become more "readable" and understandable, even for domain non-experts. Further more, we exploit certain presentational features of Rational Rose concerning CPM elements that makes the design and display of them more straightforward. For example, stereotypes can be displayed in various formats: The `Cube`, `Axis` and `Tape` classes are displayed with the stereotype icon inside the class frame while the `Dimension`, `Point` and `2D Slice` classes are displayed using their respective stereotype icons only.

4 Representing CMP Entities Using XML

A last issue we phased in the contents of this paper is information interchange of CPM models with presentation and visualization applications or browsers. We have adopted the eXtensible Markup Language (XML) [XML02], due to its wide acceptance and adoption from the academic and commercial community as the neutral-platform, vendor independent, meta-language standard for information interchange.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDe-
fault="qualified" attributeFormDefault="unqualified">
    <xs:complexType name="Cube">
        <xs:annotation>
            <xs:documentation>The Cube Multidimensional Data Base Presentation
</xs:documentation>
        </xs:annotation>
        <xs:sequence>
            <xs:element ref="CS"/>
            <xs:element ref="Two-D-Slice" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="Dimension">
        <xs:annotation>
            <xs:documentation>Dimensions</xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Attributes" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="DS">
        <xs:annotation>
            <xs:documentation>The underlying dataset</xs:documentation>
        </xs:annotation>
    </xs:element>
```

Fig. 4: The CPM metamodel of Fig. 1 & 3, using stereotypes in Rational Rose

In Fig. 4 we present a part of the XML DTD schema of the CMP metamodel of Fig. 1 or 3. This schema can be accordingly processed and interpreted by any front end application or visualization platform used, by means of either internal processing structures or – more preferably – by applying XSLT stylesheets [21] to transform the original XML schema into as many different presentations of the original CPM model as needed, according to the kind and nature of the front-end application or visualization platform used.

5 Conclusions and Future Work

In this paper we have demonstrated how a newly introduced presentation model for multidimensional data, namely the Cube Presentation Model (CPM) can be modeled using the widely accepted and popular Unified Modeling Language and more specifi-

cally, by defining UML stereotype extensions to more elegantly and naturally represent the modeled domain. Further more, we demonstrated how we have adapted Rational Rose, one of the most well known modeling tools globally, to define and use these stereotype extensions. Finally, we used the eXtensible Markup Language for storing any CPM presentation model instantiation and interchanging data with presentation and visualization front end tools, by means of XSLT stylesheets.

Next steps in our research include the extension of the herein described work, by completing Rational Rose with a number of mechanisms and extensions for CPM schema validation and automatic generation of the respective XML DTD Schema and the design of XSLT stylesheets for mapping CPM instances to common OLAP visualization techniques (Pivot Tables, Table Lens, Hyperbox etc.). Further more, we are working on the introduction of suitable, novel visualization techniques for CPM, complying with current standards and recommendations as far as usability and user interface design is concerned and its extension to address the specific visualization requirements of mobile devices. All these will be embedded in a suitable visualization platform to be designed and developed.

References

[1] S. Chaudhuri, U. Dayal: An Overview of Data Warehousing and OLAP technology. *ACM SIGMOD Record*, 26(1), (1997)

[2] E.F. Codd: *Providing OLAP to User-analysts: an IT Mandate*. E.F. Codd and Associates, (1993)

[3] W.H. Inmon: *Building the Data Warehouse*. John Wiley, (1996)

[4] C. Sapia, M. Blaschka, G. Höfling, B. Dinter: Extending the E/R model for the Multidimensional Paradigm. *Proeedings. International Workshop DMDW*, (1998)

[5] J. Trujillo, Manuel Palomar, Jaime Gomez, Il-Yeol Song: Designing Data Warehouses with OO Conceptual Models. *IEEE Computer*, December (2001)

[6] A. Abello, J. Samos, F. Saltor: A Framework for the Classification and Description of Multidimensional Data Models. *Proceedings DEXA Conference*, Munich, Germany (2001) 668-677

[7] P. Vassiliadis, T. Sellis: A Survey of Logical Models for OLAP Databases. *ACM SIGMOD Record* 28(4), (1999)

[8] A. Tsois, N. Karayannidis, T. Sellis: MAC: Conceptual Data Modeling for OLAP. *Proceedings International Workshop DMDW* (2001)

[9] D.A. Keim. Visual Data Mining. Tutorials 23rd International Conference on Very Large Data Bases (VLDB), Athens, Greece, (1997)

[10] A. Inselberg. Visualization and Knowledge Discovery for High Dimensional Data. *Proceedings 2nd IEEE UIDIS Workshop,* (2001)

[11] M. Gebhardt, M. Jarke, S. Jacobs: A Toolkit for Negotiation Support Interfaces to Multi-Dimensional Data. *Proceedings ACM SIGMOD Conference*, (1977) 348–356.

[12] S. Lujan-Mora. Multidimensional Modeling using UML and XML. PhD Thesis, 2002.

[13] S. Lujan-Mora, J. Trujillo, I.-Y. Song: Multidimensional Modeling with UML Package Diagrams. *Proceedings ER Conference*, Tampere, Finland (2002)

[14] G.Booch, J. Rumbaugh, I. Jacobson: *The Unified Modeling Language: User Guide. Object Technology Series*, Addison–Wesley, (1999)

[15] Object Management Group (OMG): Unified Modeling Language Specification 1.5. Internet: http://www.omg.org (2003)
[16] Rational Software Corporation: Using the Rose Extensibility Interface, (2001)
[17] A. Maniatis, P. Vassiliadis, S. Skiadopoulos, Y. Vassiliou: CPM: a Cube Presentation Model for OLAP. *Proceedings DaWaK Conference*, Prague, Czech Republic, (2003)
[18] Microsoft Corp. OLEDB for OLAP (1998) Available at: http://www.microsoft.com/data/oledb/olap/
[19] A. Maniatis, P. Vassiliadis, S. Skiadopoulos, Y. Vassiliou. CPM: a Cube Presentation Model. http://www.dblab.ece.ntua.gr/~andreas/publications/CPM_dawak03.pdf (long version).
[20] P. Vassiliadis, S. Skiadopoulos: Modeling and Optimization Issues for Multidimensional Databases. *Proceedings CAiSE Conference*, Stockholm, Sweden (2000)
[21] World Wide Web Consortium (W3C): XSL Trasformations (XSLT) version 1.0. Internet: http://www.w3.org (1999)
[22] World Wide Web Consortium (W3C): eXtensible Markup Language (XML) 1.0. Internet: http://www.w3.org (2000)