**Doulkifli Boukraa*, Omar Boussaïd**, Fadila Bentayeb**, Sabine Loudcher****

*National High School of Computer Science, Oued Smar, Algiers, Algeria
d_boukraa@esi.dz
**ERIC Laboratory, University of Lyon, LumièreLyon 2, Bron, France
{omar.boussaid, fadila.bentayeb, sabine.loudcher}@univ-lyon2.fr,

# OLAP OPERATORS FOR A COMPLEX OBJECT-ORIENTED MULTIDIMENSIONAL MODEL

**Abstract:** The data warehousing process should be adapted in response to complex data. Only few works address the issues of managing and analyzing complex data. Warehousing complex data involves different issues regarding their structure, storage, processing and analysis. Dimensional modeling must be adapted to complex data to take into account their specificities. In this paper, we propose an Object-Oriented (O-O) Data Warehouse Model capable to efficiently manage and analyze complex data. We also define a set of O-O OLAP operators in order to construct data cubes. In this context, semantic richness of O-O models will not only facilitate integration, but will also help analysts to understand the real meaning of data. Our O-O model has the main advantage of integrating various aspects of data complexity and thus of providing different points of view of the same data for a better decision making.

**Keywords:** data warehouse design, multidimensional model, complex objects, OLAP operators, Object-Oriented Model.

## 1. Introduction

Data warehouses support business decisions by collecting, consolidating and organizing data in a multidimensional way for reporting and analysis purposes with tools such as OLAP or data mining [Chaudhuri, Dayal 1997]. In recent years, special attention has been paid to relational data warehouse systems consolidating data from relational database systems. As companies collect huge amounts of heterogeneous and complex data, some efforts are needed to structure them and to make them as homogeneous as possible. In the case of numerical data, data warehousing systems often provide tools to assist in this process and several commercial data warehousing systems have been introduced into the market, e.g. Oracle Express Server, DB2, etc. Unfortunately, standard tools are inappropriate for producing relevant analysis axes when data are complex [Boussaïd et al. 2006]. Complex data are not only numerical or symbolic, but they may be represented in various formats (databases, texts, images, sounds, videos, etc.), diversely structured (relational databases, object databases, etc.), native from several different sources, described by several channels or points of view and changing in terms of definition or value such as temporal databases, periodical surveys, etc. [Darmont et al. 2005]. A literature review reveals the increasing interest of researchers and industrials in complex data warehousing and analysis. First, regarding the multiplicity of data sources, many works are considering Web data as a valuable source for a data warehouse [Bhowmick, Madria, Ng 2003]. Other works are motivated by the increasing popularity of XML as an industrial standard and valuable source for a data warehouse [Pedersen, Jensen 1999; Park, Han, Song 2005; Golfarelli, Rizzi, Vrdoljak 2001]. Secondly, from a data format point of view, the need to expand data warehousing to other fields than business has led to consider more data formats, such as images in geographical applications [Gómez et al. 2009] and medical applications [Wong et al. 2001]. Thirdly, regarding data structures, most of the reviewed works focus on semi-structured data, such as [Wong, Lam, Orgun 2001] whereas others propose to analyze unstructured text data [Inokuchi, Takeda 2007; Keith, Kaser, Lemire 2006].

Another important aspect in complex data warehousing is the conceptual multidimensional modeling. In this context, the object-orientation is pervasive in many works [Trujillo, Palomar 1998; Nassis et al. 2004; Ravat, Teste 2000; Lujàn--Mora 2002; Li, An 2005]. Other models are based on trees [Golfarelli, Rizzi, Vrdoljak 2001; Vrdoljak, Banek, Rizzi 2003] or an adapted star schema [Kimball, Ross 2002] by including spatio-temporal features, such as maps [Gómez et al. 2009], textual measures [Keith et al. 2006] and document references [Mothe et al. 2003]. Associated with the multidimensional models are OLAP operators. The proposed operators are either inspired from the traditional OLAP or are specific to the new models. These include spatio-temporal operators and textual operators [Park, Han, Song 2005] to mention a few.

In this paper, we propose a new multidimensional model for complex data and a set of related OLAP operators. Our model is based on the concept of complex object proposed in [Boussaïd et al. 2007] as a solution to the problem of complex data integration. Compared with existing models, the main novelties of our work can be summed up as follows:

1. We model a complex object as a container of complex data to be analyzed as a whole, such as a patient record in healthcare systems, a scientific publication, etc.

2. We consider two kinds of hierarchies: those composed by objects and those composed by attributes.

3. We provide a conceptual space of data visualization which is independent from any OLAP visual solution and propose related analysis operators. The visualization space can then be mapped on different solutions, such as cross-tabs and spreadsheets.

The remainder of this paper is organized as follows. In section 2, we present an illustrative example. In section 3, we introduce the main concepts of our object-oriented multidimensional model. The construction and projection operators are presented in section 4. In section 5, our proposed O-O data warehouse model and O-O OLAP operators are discussed. Finally, we conclude and give some research perspectives.

## 2. Illustrative example

Let us consider an academic institution that wants to warehouse data about scientific publishing in order to perform diverse analysis tasks. Typical analysis tasks would be for instance: (1) assessing the quality of a publication according to different metrics, such as its rating, (2) analyzing the publishing frequency of a researcher, (3) analyzing the impact of co-authorship on the quality of a publication, and (4) assessing the quality of journals and conferences. Scientific publishing data are complex: they are coming from multiple sources (e.g. DBLP,[1] ACM SIGMOD[2]), have multiple formats (such as images and videos included in conferences' websites), and hold multiple structures (e.g. semi-structured publications, unstructured acknowledgments). To perform the previous analysis tasks, these data can be physically integrated into a data warehouse according to a multidimensional model [Rafanelli 2003]. Different approaches can be adopted to design such a model: top down, bottom up or hybrid approach. In our case, we adopt a top down approach. Thus, we present a set of general modeling requirements as follows:

1. Modeling complex data as facts or dimensions. For instance, publications can be analyzed according to authors. Yet, both publications and authors can hold complex data.

2. Modeling explicit and semantic relationships between complex data, e.g. the publications are related to the authors on the one hand and to conferences or journals on the other.

3. Symmetric treatment of facts and dimensions. This means that, through different contexts of analysis, data of the same nature may play the role of fact, dimension or both simultaneously. Hence, facts and dimensions have to be instantiated dynamically for each context of analysis. For example, publications can be analyzed according to authors in one context and authors may be analyzed according to publications in another. Besides, publications can be analyzed according to other publications via the reflexive relationship of citation.

4. Modeling complex hierarchies. On the one hand, complex data may be organized hierarchically, e.g. publications, journal issues, journal volumes and journals form a hierarchy. On the other hand, attributes of a given object can show a hierarchical structure, e.g. in a publication, subsections, sections and the whole publication form a hierarchy.

[1] http://dblp.uni-trier.de/.

[2] http://www.sigmod.org/sigmod/anthology/index.htm.

## 3. The complex object multidimensional model

In order to meet the above-mentioned modeling requirements, we propose a multidimensional model based on the concept of complex object proposed in [Boussaïd et al. 2007]. Our multidimensional model is composed by complex objects being linked by different kinds of relationships. In addition, some of the complex objects and some of their attributes may be organized as hierarchies. Each concept of our model can be described at the instance level or at the class level (e.g. complex object class/instance, relationship class/instance, etc). However, due to space limitation, we define each concept at the class level unless otherwise specified.

### 3.1. Concepts and definitions

**Complex Object:** a complex object is an abstract or a physical entity that is meant to be analyzed as a subject (fact) or as an axis (dimension). It results from instantiating the generic model proposed by [Boussaïd et al. 2007]. However, at this stage, we model a complex object as a set of attributes; each one may have one or more values. A special mono-valued attribute plays the role of object's identifier. A complex object can hold two kinds of attributes: those that are independent from its nature such as *name* and *size*, and those that are nature-specific such as *length* for a text, the *color* for an image, etc. The attributes of an object can be independent from each other or linked via different kinds of relationships, among others hierarchical ones. At this stage, we consider only hierarchical relationships because of their special interest in the context of data warehousing. Other kinds of relationships will be considered in future work.

**Definition 1:** A complex object is defined by a couple $Obj = (id^{Obj}, A^{Obj})$ where $id^{Obj}$ is the identifier of $Obj$ and $A^{Obj} = \{a_i^{Obj} / i \in N\}$ is the set of its attributes.

**Complex relationship:** a complex relationship is an explicit link between two (or more) complex objects. We consider many kinds of relationships such as association, inheritance, composition, etc. However, it is worth noting that we deal only with binary relationships, i.e. between pairs of complex objects. As a matter of fact, in a data warehouse, all relationships are binary (i.e. between facts and dimensions or between two consecutive components of a hierarchy in a snowflake schema). Therefore, we believe that it is preferable to decompose all non-binary relationships into binary ones. A relationship is characterized by its name and by the two complex objects that it links.

**Definition 2:** A complex relationship is defined by a couple $R = (Obj_s^R, Obj_c^R)$ where $Obj_s^R$ is the source object of $R$ and $Obj_c^R$ is the target object of $R$.

**Complex attribute hierarchy:** a complex attribute hierarchy is a special relationship that is defined between attributes of one object. It is characterized by its name and by its members, including or not the object's identifier. To form such a hierarchy, an ordering relation $<^a$ orders the attributes from the finest-grained member up to the least fine-grained one. In addition, we add a dummy attribute called $All^a$ as the last member of the hierarchy. This special attribute is needed when we will define the OLAP operators.

**Definition 3:** An attribute hierarchy is defined by $AH^{Obj} = (\{a_i^{Obj} \text{ in } A^{Obj}\} \cup \{id^{Obj}\} \cup \{All^a\}, <^a)$.

**Complex object hierarchy:** an attribute hierarchy is a special relationship that is defined between a set of complex objects. It is characterized by its name and by its members. To form such a hierarchy, an ordering relation $<^{Obj}$ orders the objects from the finest-grained member up to the least fine-grained one. In addition, we add a dummy object called $All^{Obj}$ as the last member of the hierarchy. This special object is also needed when we will define the OLAP operators.

**Definition 4:** An object hierarchy is defined by $OH = (\{Obj_i\} \cup \cup \{All^{obj}\}, <^{Obj})$.

**Complex multidimensional schema:** a complex multidimensional schema is composed by (1) the set of complex objects, (2) the set of complex relationships, (3) the set of attribute hierarchies and (4) the set of object hierarchies.

**Definition 5:** A complex multidimensional schema is a four-tuple defined by $SCM = (SO, SR, SAH, SOH)$ where $SO = \{Obj_i / i \in N\}$, $SR = \{R_j / j \in N\}$, $SAH = \{AH_k / k \in N\}$ and $SOH = \{OH_m / m \in N\}$.

## 3.2. Illustration

In the scientific publishing field, we can identify many complex objects, e.g. *Publication*, *Author*, *Proceedings*, *Journal* and *Time*. A *Publication* is characterized for example by the attributes *Title*, *Pages*, *URL*, *Keyword*, *Section* and *Subsection*. Besides, different relationships link the objects, e.g. the *Written by* relationship links *Publications* to *Authors*. An example of attribute hierarchies in the object *Publication* is $Subsection <^a Section <^a ID\_Pub <^a All^a$. Finally, an example of an object hierarchy is $Conference\ proceedings <^{Obj} Conference <^{\mathrm{Obj}} All^{Obj}$. Figure 1 depicts a complete example of a multidimensional schema of complex data. In this figure, the objects, relationships, attribute hierarchies and object hierarchies are represented respectively with rounded rectangles, lines, single-lined arrows and double-lined arrows. Formally, we call this schema *SCM_Pub*. Then, *SCM_Pub* = {*SO*, *SR, SAH*, *SOH*} where *SO* = {*Publication*, *Publication_Type*, *Time*, *Author*, *Conference_Proceeding*, *Conference*, *Journal_Issue*, *Journal_Volume*, *Journal*}, *SR* = {*Written_by*, *Reviewed_by*, *Cited_in*, *Date_publication*, *Publi_journal*, *Publi_conference*}, *SAH* = {*H_Pub*, *H_Time*}, where *H_pub* is associated with the object *Publication* such that *H_pub* = ({*Subsection*, *Section*, *ID_Pub*, $All^a$ },$<^a$) and *H_Time* is associated with the object *Time* such that *H_Time* = ({*Day*, *Month*, *Year*, $All^a$},$<^a$) and *SOH* = {*H_Type_Pub*, *H_Conf*, *H_Journal*}, where *H_Type_Pub* = ({*Publication*, *Publication_Type*, $All^{obj}$},$<^{Obj}$), *H_Conf* = ({*Conference_Proceeding*, *Conference*, $All^{obj}$}, $<^{Obj}$) and *H_Journal* =({*Journal_issue*, *Journal_Volume*, *Journal*, $All^{obj}$},$<^{Obj}$).

## 4. The O-O OLAP operators

The multidimensional model of complex objects is, by definition, independent from any context of analysis. In other words, there is not an *a priori* analysis subject or analysis axes. As a matter of fact, the subject and the axes are designated when a specific analysis is defined. In addition, only relevant elements are to be considered since not all the relationships, attribute hierarchies and object hierarchies are relevant for an analysis. Therefore, in order to define the analysis subject and axes as well as relevant elements, the multidimensional schema will be subject of further operations. These operations project the multidimensional schema on subsets of its components to produce an analysis-ready structure called complex cube. In what follows, we present the set of operators to construct a complex cube and the set of operators to analyze its data.

### 4.1. Construction operators

**Complex cubic projection:** This operation projects the multidimensional model onto one object to play the role of the fact. This operation results in projecting the following elements. (1) All the relationships that connect the fact to other objects. (2) All the objects that are directly connected to the fact. (3) All members of object hierarchies that contain the projected objects in (2). (4) All attribute hierarchies relating to the projected objects in (2) and (3). In addition, each projected object hierarchy (respectively attribute hierarchy) will be reduced by removing some of its members so that the finest-grained member becomes the object that is directly connected to the fact (respectively becomes the object's identifier).

**Definition 6:** Let $SCM = (SO, SR, SAH, SOH)$ be a multidimensional schema. The complex cubic projection of $SCM$ onto a complex object $Obj$ is defined by $\Pi_{cc}$ $Obj$ $(SCM) = C = (F, SR^C, SD, SAH^C, SOH^C)$ where $F$ is the fact, $SR^C = \{R_i^C / i \in N\}$ is the set of projected relationships such that $SR^C \subseteq SR$, $SD = \{D_j^C / j \in N\}$ is the set of dimensional objects of the cube such that $SD \subseteq SO$, $SAH^C = \{AH_k^C / k \in N\}$ is the set of projected attribute hierarchies after reduction and finally $SOH^C = \{OH_m^C / m \in N\}$ is the set of projected object hierarchies after reduction.

**Dimensional projection:** The number of relationships resulting from the complex cubic projection will define the initial dimensionality of the cube. The dimensional projection is then the operation of selecting relevant relationships for the analysis.

**Definition 7:** Let $C = (F, SR^C, SD, SAH^C, SOH^C)$ be a complex cube scheme such that $SR^C = \{R_1^C, R_2^C, \ldots, R_n^C\}$, where $n$ denotes the number of relationships of $C$. Let $R_1^{C'}, R_2^{C'}, \ldots, R_k^{C'}$ be $k$ relationships of $SR^C$ where $k \leq n$. The dimensional projection operation of $C$ onto the $k$ relationships $R_1^{C'}, R_2^{C'}, \ldots, R_k^{C'}$ is defined by $\Pi_D\, R_1^{C'}, R_2^{C'}, \ldots, R_k^{C'}\, (C) = C' = (F', SR^{C'}, SD', SAH^{C'}, SOH^{C'})$ where $F' = F$, $SR^{C'} = \{R_1^{C'}, R_2^{C'},$

$\ldots, R_k^{C'}\} \subseteq SR^C$, $SD' \subseteq SD$, $SAH^{C'} \subseteq SAH^C$ and $SOH^{C'} \subseteq SOH^C$. The sets $SD$, $SAH^C$ and $SOH^C$ will be reduced following the removal of the non-projected relationships.

**Object hierarchy projection:** This operation projects a cube schema containing a set of object hierarchies onto a subset of hierarchies that are relevant to the analysis.

**Definition 8:** Let $C = (F, SR^C, SD, SAH^C, SOH^C)$ be a complex cube where $SOH^C = \{OH_1{}^C, OH_2{}^C, \ldots, OH^C{}_n\}$ such that $n$ denotes the number of object hierarchies. Let $OH_1{}^{C'}$, $OH_2{}^{C'}$, …, $OH_k{}^{C'}$ be $k$ object hierarchies of $SOH^C$ such that $k \leq n$. The object hierarchy projection of $C$ onto the $k$ hierarchies $OH_1{}^{C'}$, $OH_2{}^{C'}$, …, $OH_k{}^{C'}$ is defined by $\Pi_{OH} OH_1{}^{C'}, OH_2{}^{C'}, \ldots, OH_k{}^{C'}\,(C) = (F', SR^{C'}, SD', SAH^{C'}, SOH^{C'})$, where $F' = F$, $SR^{C'} = SR^C$, $SD' \subseteq SD$, $SAH^{C'} \subseteq SAH^C$ and $SOH^{C'} = \{OH_1{}^{C'}, OH_2{}^{C'}, \ldots, OH_k{}^{C'}\} \subseteq SOH^{C'}$.

**Attribute hierarchy projection:** This operation projects a cube containing a set of attribute hierarchies onto a subset of hierarchies that are relevant for the analysis. An attribute hierarchy projection is denoted by $\Pi_{AH}$. Its formal definition is similar to that of the object hierarchy projection. However, due to space limitation, we do not present it.

**Measure projection:** so far, the cube schema does not contain any explicit measure. The measure projection is then the operation of naming some of the fact's attributes as measures to be analyzed. The identification of relevant measures may be intuitive or by using some techniques such as data mining. Moreover, each measure is associated with (1) a set of relationships along which its values can be aggregated depending on its additivity and (2) a set of suitable aggregation functions to ensure aggregation semantics.

**Definition 9:** Let $C = (F, SR^C, SD, SAH^C, SOH^C)$ be a cube where $F = (ID^F, A^F)$ such that $ID^F$ denotes the fact's identifier and $A^F$ the set of the fact's attributes. Let $SAF = \{af_i \,/\, i \in N\}$ be a set of aggregation functions. The measure projection of $C$ onto a set of measures $SM$ is defined by $\Pi_M SM(C) = (F, SM, SR^{C'}, SD', SAH^{C'}, SOH^{C'})$ such that $SR^{C'} = SR^C$, $SD' = SD$, $SAH^{C'} = SAH^C$, $SOH^{C'} = SOH^C$ and $SM = \{M_i \,/\, i \in N\}$ where $M_i \in \{ID^F\} \cup A^F$. We also define two functions *AggRel* and *AggFun* as follows: *AggRel* associates each measure $M_i$ of $SM$ with a set $SR^{Mi} \subset SR^C$ and *AggFun* associates each measure $M_i$ with a set $SAF^{Mi} \subset SAF$.

## 4.2. Illustration

Let us consider the previous example. Suppose we want to analyze publications according to different dimensions. Then, the object *Publication* represents the fact. The complex cubic projection projects the multidimensional schema *SCM* onto *Publication*. Then we write $C\_Pub = \Pi_{cc}$ *Publication* (*SCM_Pub*). Now suppose that publications are to be analyzed only according to their authors, conferences, journals, time and other publications. In this case, the dimensional projection will result in removing the relationship *Reviewed by*. Then we write $C\_Pub1 = \Pi_D$ *Written_by*, *Cited_in*, *Date_publication*, *Publi_journal*, *Publi_conference* (*C_Pub*) where $SR^{C\_Pub1}$ = {*Written_by*, *Cited_in*, *Date_publication*, *Publi_journal*, *Publi_conference*}. Let us now suppose that the only interesting object hierarchies for the analysis are *H_Conf* and *H_Journal*. The third hierarchy, namely *H_Type_Pub*, will be removed. So, the cube *C_Pub1* is projected onto the hierarchies *H_Conf* and *H_Journal*. Formally, $C\_Pub2 = \Pi_{OH} H\_Conf, H\_Journal$ (*C_Pub1*) such that $SOH^{C\_Pub2}$ = {*H_Conf*, *H_Journal*} and *SD* = {*Publication*, *Time*, *Author*, *Conference_Proceedings*, *Conference*, *Journal_Issue*, *Journal_Volume*, *Journal*}. Similarly, if we want to keep the attribute hierarchy *H_Time*, we project the cube *C_Pub2* onto *H_Time*. Then we write $C\_Pub3 = \Pi_{OH} H\_Time(C\_Pub2)$ where $SAH^{C\_Pub3}$ = {*H_time*}. Finally, we suppose that we want to get the rating of each publication and the set of its keywords. The ratings are to be analyzed by author and by period of time whereas the keywords are to be analyzed by author and by conference. For the rating measure, we want to get the maximum, minimum and average rating (respectively denoted by max_rating, min_rating avg_rating). For the keyword measure, we want to get the top keyword of a set of keywords (denoted by *Top_keyword*). We call *C_Pub4* the cube resulting from the measure projection of *C_Pub3* onto the set $SM^{C\_Pub4}$ = {*Rating*, *Keywords*}. Then, $C\_Pub4 = \Pi_M SM^{C\_Pub4}(C)$ where *AggFunc*(*rating*) ={*Avg_Rating*, *Max_Rating*, *Min_Rating*}, *AggRel*(*rating*) = {*Written_by*, *Date_publication*}, *AggFunc*(*keywords*) = {*Top_keyword*}, *AggRel*(*keywords*) = {*Written_by*, *Publi_conference*}. Figure 2 depicts the cube *C_Pub4*.[3] The grey rounded rectangle represents the fact whereas the other rectangles represent the dimensional objects.

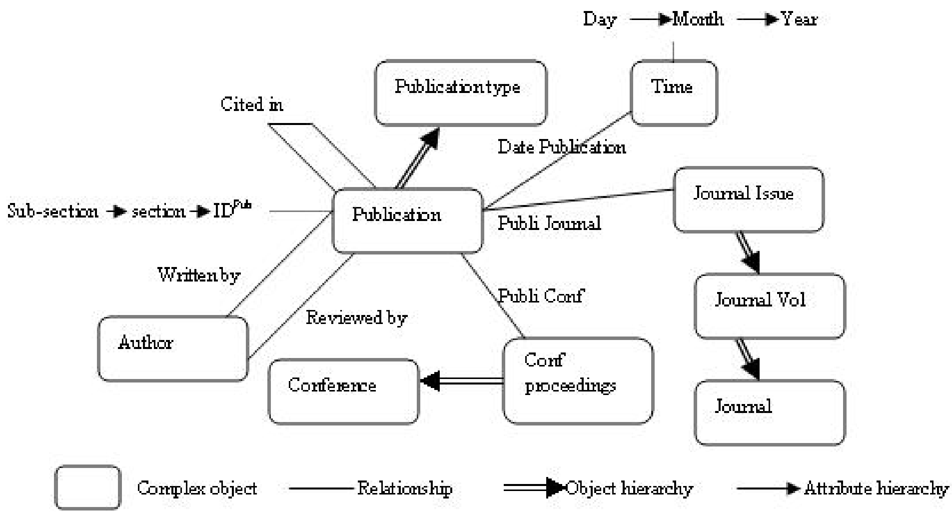


**Figure 1.** Publications multidimensional schema

[3] For the sake of clarity, we do not represent the measures in the schema.

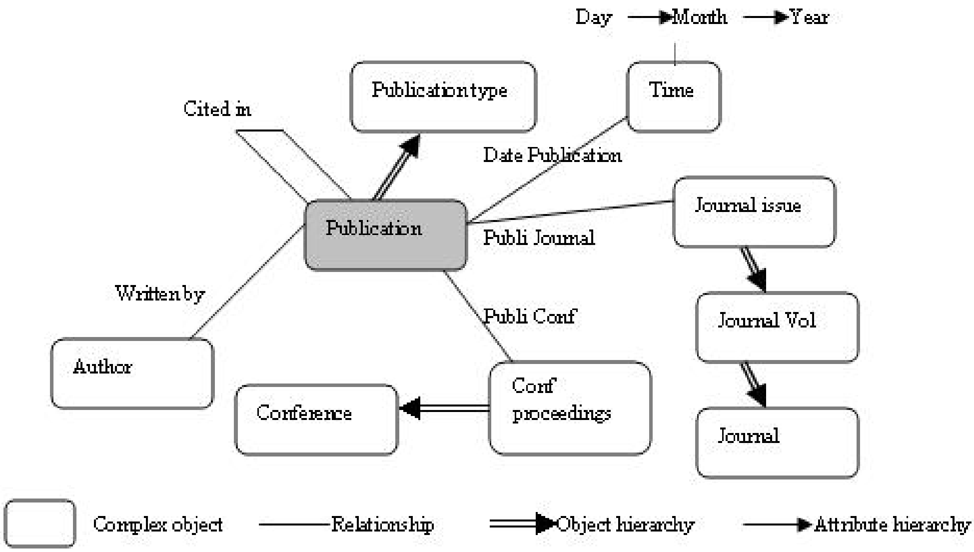


**Figure 2.** The structure of the cube C_Pub4

### 4.3. Analysis Operators

In this section, we present the OLAP operations that aim at analyzing a complex cube. We consider only the granularity-related operations (rollup, drill-down, cube generation) and the set-based operations (dimension-based selection, fact-based selection). To this end, we define a conceptual space to visualize the data cube, called a view over a complex cube.

**View over a complex cube:** A view over a cube is a multidimensional space that allows observing a measure along its assigned relationships. These relationships determine the dimensions of the view. The measure is then observed according to a member of an object hierarchy or an attribute hierarchy if there is any. If the dimension has no related hierarchy, then the measure is observed either according to the object that is directly connected to the fact or according to the dummy object $All^{Obj}$. The values of the measure are then aggregated using one of its assigned functions along all the dimensions. Figure 3 illustrates the notion of view over a complex cube. The measure $M$ is being analyzed according to four relationships, namely $R1$, $R2$, $R3$ and $R4$. The dashed arrows point to the observation elements (grey-colored elements). The grey arrows represent implicit links to the dummy object $All^{Obj}$ in case the dimension has no related hierarchy. The granularity-related operators produce a new view over the same cube. These operators are represented by blank-headed arrows. The granularity-related operations are described below.

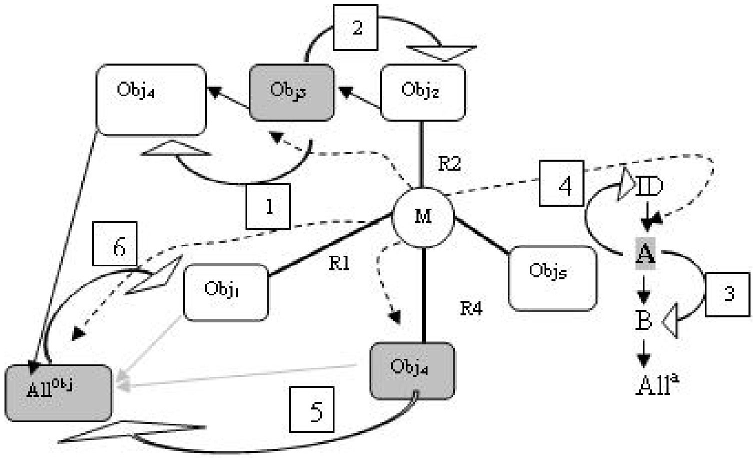


**Figure 3.** Granularity-related operations

**Rollup:** We define three kinds of rollup operations. The object hierarchy-based rollup (arrow no. 1) consists in moving the observation element from the current member of an object hierarchy to the upper member while aggregating all the values of the measure. If the initial observation element is the least fine-grained member of the hierarchy (different from $All^{Obj}$), the next observation element becomes $All^{Obj}$. Analogously, the attribute hierarchy-based rollup (arrow no. 3) consists in moving the observation element from the current member of the attribute hierarchy to the upper member while aggregating all the values of the measure. If the initial observation element is the least fine-grained member of the hierarchy, the next observation element becomes $All^a$. Finally, the hierarchy-less rollup, (arrow no. 5) consists in removing the current relationship from the view. The next observation element becomes the dummy object $All^{Obj}$.

**Drill down:** We define three kinds of drill-down operations: the object hierarchy-based drill-down (arrow no. 2), the attribute hierarchy-based drill-down (arrow no. 4) and the hierarchy-less drill-down (arrow no. 6). These operations consist in moving the observation element from the initial member of a hierarchy to the lower member. Particularly, the operation of moving down from the dummy object $All^{Obj}$ or from the dummy attribute $All^a$ level to the lower level introduces back a relationship into the view.

**Cube generation:** This operation consists, for each measure and for a given aggregation function, in applying successively the rollup operation described above for each relationship and along one of its related hierarchies if there is any. In case there is no hierarchy related to a relationship, the hierarchy-less rollup is applied.

**Set-based operators:** The set-based operators consist is selecting sub-sets of the data cube. In this paper, we present two operations. The *dimension-based selection* selects a subset of instances of a dimensional object for which a predicate holds true.

This operation automatically filters out the instances of other members of the same dimension as well as the instances of the fact that are not connected to the selected instances. Once the fact instances are selected, this operation filters out the instances of the other dimensional objects that are not connected to the selected fact instances. The *fact-based selection* selects a subset of fact instances for which a predicate holds true. This operation automatically filters out the instances of the dimensional objects that are not connected to the selected fact instances.

## 5. Discussion

In this section, we compare our complex O-O multidimensional model with existing models. The main novelty that we bring is to consider the facts and dimension members as complex objects. In fact, existing multidimensional models structure the facts and dimension members as (UML) classes. In our model, a fact or a dimension member represents a package of strongly coupled classes to be treated as a whole, which we believe is closer to reality. The second novelty of our work is that we model two kinds of hierarchies (object and attribute hierarchies). In fact, in existing models, hierarchies are either composed by classes [Li, An 2005; Lujàn-Mora 2002] or by attributes [Golfarelli, Rizzi, Vrdoljak 2001; Vrdoljak, Banek, Rizzi 2003]. Furthermore, in mixed (star/snowflake) schemas, the coexistence of entity and attribute hierarchies is rather a tradeoff between normalization and performance optimization. In our model, the object and attribute hierarchies are modeled from a conceptual point of view. In other words, an attribute hierarchy cannot be transformed into an object hierarchy and vice versa. In fact, the first transformation would produce a complex model holding too many objects to manage. Conversely, the second transformation would induce the loss of independency between objects while decreasing the number of possible cubes to construct. The third novelty of our work is that we explicitly address the question of measure additivity. In this context, among the reviewed literature, only Lujàn-Mora [2002] tackles this problem by using UML constraints in order to express additive rules. The solution consists in associating each measure with the set of relationships along which it can be analyzed simultaneously.

## 6. Conclusion and perspectives

In this paper, we presented a complex object-based multidimensional model. The concept of complex object allows capturing and describing multiple facets of data complexity (format, structure, source, etc.). The main advantage of our model consists in providing different points of view of the same data, which leads to better decision making. In addition, our model is well adapted to capture different kinds of relationships, such as associations and inheritance, within or between complex objects. We also provided a set of OLAP operators to manipulate the O-O model. Moreover, we defined OLAP operators onto our O-O multidimensional model. The construction OLAP operators allows transforming the multidimensional model into an analysis-ready model called complex cube. Thus, it is possible to construct different cubes in order to meet multiple analysis needs. The analysis operators allow analyzing the complex data cube in terms of aggregation and selection. For future work, we plan to complete the formal framework of complex object analysis by providing formal definitions of aggregation and selection operators. Next, we will provide a mapping of the multidimensional model and the cube model into logical and physical models using XML. In addition, we plan to implement our proposals using a complete case study. Performance issues will then be investigated, especially by providing a suitable physical model for our complex O-O multidimensional model.

## References

Boussaïd O., Ben Messaoud R., Choquet R., Anthoard S. (2006), X-warehousing: An ML-based approach for warehousing complex data, [in:] *10th East-European Conference on Advances in Databases and Information Systems (ADBIS 2006)*, Eds. Y. Manolopoulos, J. Pokorný, T.K. Sellis, *Lecture Notes in Computer Science*, Vol. 4152, Springer, 2006, pp. 39-54.

Boussaïd O., Tanasescu A, Bentayeb F., Darmont J. (2007), Integration and dimensional modelling approaches for complex data warehousing, *Journal of Global Optimization*, Vol. 37, No. 4, pp. 571-591.

Bhowmick S.S., Madria S.K., Ng W.K. (2003), *Web Data Management: A Warehouse Approach*, Springer Verlag,. New York.

Chaudhuri S., Dayal U. (1997), An overview of data warehousing and OLAP technology, *SIGMOD Record*, Vol. 26, No. 1, ACM, pp. 65-74.

Darmont J., Boussaïd O., Ralaivao J.C., Aouiche K. (2005), An architecture framework for complex data warehouses, [in:] *7th International Conference on Enterprise Information Systems (ICEIS 2005)*, *Computing Research Repository*, Vol. abs/0707.1534.

Golfarelli M., Rizzi S., Vrdoljak B. (2001), Data warehouse design from XML sources, [in:] *Proceedings of the 4th International Workshop on Data Warehousing and OLAP (DOLAP 2001)*, ACM, pp. 40-47.

Gómez L.I., Kuijpers B., Moelans B., Vaisman A.A. (2009), A survey of spatio-temporal OLAP, *International Journal of Data Warehousing and Mining*, Vol. 5, No. 3, pp. 28-55.

Inokuchi A., Takeda K. (2007), A method for online analytical processing of text data, [in:] *Proceedings of the 16th Conference on Information and Knowledge Management (CIKM 2007)*, Eds. M.J. Silva, A.H.F. Laender, R.A. Baeza-Yates, D.L. McGuinness, B. Olstad, Ø.H. Olsen, A.O. Falcão, ACM, 2007, pp. 455-464.

Keith S., Kaser O., Lemire D. (2006), Analyzing large collections of electronic text using OLAP, *Computing Research Repository*, Vol. abs/cs/0605127.

Kimball R., Ross M. (2002), *The Data Warehouse Toolkit*, John Wiley and Sons, New York.

Li Y., An A. (2005), Representing UML snowflake diagram from integrating XML data using XML schema, [in:] *Proceedings of the ICDE International Workshop on Data Engineering Issues in E-Commerce (DEEC 2005)*, IEEE Computer Society, pp. 103-111.

Luján-Mora S. (2002), Multidimensional modeling using UML and XML, [in:] *12th Workshop for PhD Students in Object-Oriented Systems (PhDOOS 2002), 16th European Conference on Object-Oriented Programming (ECOOP 2002)*, Ed. B. Magnusson, *Lecture Notes in Computer Science*, Vol. 2548, Springer, pp. 48-49.

Mothe J., Chrisment C., Dousset B., Alaux J. (2003), DocCube: Multi-dimensional visualisation and exploration of large document sets, *Journal of the American Society for Information Science and Technology*, Special topic section: *Web Retrieval and Mining*, Guest Editor H. Chen, Vol. 54, No. 7, pp. 650-659.

Nassis V., Rajugan R., Dillon T.S., Rahayu W. (2004), Conceptual design of XML document warehouses, [in:] *Proceedings of 6th International Conference on Data Warehousing and Knowledge Discovery (Dawak 2004)*, Eds. Y. Kambayashi, M.K. Mohania, W. Wöß, *Lecture Notes in Computer Science*, Vol. 3181, Springer, pp. 1-14.

Park B.K., Han H., Song I.-Y. (2005), XML-OLAP: A multidimensional analysis framework for XML warehouses, [in:] *Proceedings of 7th International Conference on Data Warehousing and Knowledge Discovery (Dawak 2005)*, Eds. A.M. Tjoa, J. Trujillo, *Lecture Notes in Computer Science*, Vol. 3589, Springer, pp. 32-42.

Pedersen T.B., Jensen C.S. (1999), Multidimensional data modeling for complex data, [in:] *Proceedings of the 15th International Conference on Data Engineering (ICDE 1999)*, IEEE Computer Society, pp. 336-345.

Rafanelli M. (2003), *Multidimensional Databases: Problems and Solutions*, Idea Group Publishing, Hershey, PA,.

Ravat F., Teste O. (2000), Object-oriented decision support system, [in:] *Proceedings of the 2nd International Conference on Enterprise Information Systems (ICEIS 2000)*, Eds. B. Sharp, J. Cordeiro, J. Filipe, Stafford, pp. 79-84.

Trujillo J., Palomar M. (1998), An object-oriented approach to multidimensional database conceptual modeling, [in:] *Proceedings of the 1st International Workshop on Data Warehousing and OLAP (DOLAP 1998)*, ACM, pp. 16-21.

Vrdoljak B., Banek M., Rizzi S. (2003), Designing Web warehouses from XML schemas, [in:] *Proceedings of the 5th International Conference on Data Warehousing and Knowledge Discovery (Dawak 2003)*, Eds. Y. Kambayashi, M.K. Mohania, W. Wöß, *Lecture Notes in Computer* Science, Vol. 2737, Springer, pp. 89-98.

Wong S.T.C., Hoo K.S. Jr, Knowlton R.C., Laxer K.D., Cao X., Hawkins R.A., Dillon W.P., Arenson R.L. (2001), Design and applications of a multimodality image data warehouse framework, *The Journal of the American Medical Informatics Association*, Vol. 9, No. 3, pp. 239-254.

Wong R.K., Lam F., Orgun M.A. (2001), Modelling and manipulating multidimensional data in semistructured databases, *World Wide Web*, Vol. 4, No. 1-2, pp. 79-99.

## OPERATORY OLAP DLA ZŁOŻONYCH OBIEKTOWO ZORIENTOWANYCH WIELOWYMIAROWYCH MODELI DANYCH

**Streszczenie:** w artykule przedstawiono proces zarządzania i analizy złożonych obiektów w hurtowniach danych. Specyfika złożonych obiektów w hurtowniach dotyczy ich struktury, pamięci, procesów przetwarzania i analizy. Autorzy artykułu zaproponowali obiektowo-zorientowany model hurtowni danych umożliwiający efektywne zarządzanie złożonymi strukturami danych i ich analizę. Zostały też zdefiniowane obiektowo zorientowane operatory OLAP działające na kostkach (ang. *data cube*) danych. Semantyczne walory modelu obiektowego nie tylko ułatwiają intergrację danych, lecz także pozwalają lepiej zrozumieć rzeczywiste znaczenie danych.