Semantic Enrichment of Strategic Datacubes

Claudia Diamantini
Dipartimento di Ingegneria Informatica,
Gestionale e dell'Automazione
Università Politecnica delle Marche
via Brecce Bianche, 60131, Ancona, Italy
diamantini@diiga.univpm.it

Domenico Potena
Dipartimento di Ingegneria Informatica,
Gestionale e dell'Automazione
Università Politecnica delle Marche
via Brecce Bianche, 60131, Ancona, Italy
potena@diiga.univpm.it

ABSTRACT

In the information system view, the reference architecture for strategic and decision support is based on the Data Warehouse architecture, that enables flexible and multidimensional analysis of strategic indexes by means of OLAP tools and reports. In this paper we propose a novel model for semantic annotation of Data Warehouse schema that takes into account domain ontologies as well as a mathematical ontology. Such an ontology describes mathematical formulas underlying elements of the datacube schema, including the semantics of operands and operators. In particular, we discuss and apply the proposed model for the semantic annotation of the schema of a datacube, that is the basis for OLAP analysis and contains information derived from Data Warehouse schema. In the paper, an illustrative case study together with some examples of analysis based on this kind of annotation are provided.

Categories and Subject Descriptors

H.4.2 [**Information Systems Applications**]: Types of Systems—*Decision support*

General Terms

Design, Theory

Keywords

Datacube, Data Warehouse, Mathematical Ontology, Semantic Annotation, Strategic Index

1. INTRODUCTION

In any organization, strategic planning is one of the most critical activities that top management has to deal with. As stated in [12, p.37], strategic planning is a complex process of "dynamic, continuous activities of self-analysis" oriented to the definition of a strategy. As a matter of fact, at the strategic and decision levels, an organization develops complex planning and control cycles, where a model of the organization is compared against a "to-be" state, being it either the realization of a given vision and mission, a reference best practice, or a periodic budget. The models considered at this level by strategy experts and top management are defined by a set of high-level measurable performance indexes, that are analyzed along a number of different dimensions.

Data Warehousing (DW) has been introduced as a technology to make managers' work simpler are more effective, enabling flexible analysis of performance indexes by means of OLAP tools over multidimensional datacubes [3]. Indeed, a full exploitation of these tools requires a full understanding of the exact meaning of datacube elements, both in terms of business concepts familiar to managers and of the way they are built. In fact, conceptual definitions can only give reference to abstract concepts, while the actual implementation of these concepts in the datacube depends on the way indexes are computed. Consider for instance the "Return on Investment" (ROI) index: while the abstract, conceptual definition of ROI (the income that an investment provides in a year) is shared enough by managers, the economic literature reports different ways to calculate a ROI, known as ROI trees, depending on the economical framework adopted [9, 13]. Hence, it can happen that an apparently shared conceptual definition hides subtle operational misunderstandings. Besides limiting comprehension, differences in the calculus of an index are also a source of semantic heterogeneity that limits interoperability. As a matter of fact, although theoretically enterprises could/should define a standardized set of indexes, this is not actually the case for many true enterprises. The reason is related to the existence of some form of autonomy for organization units, e.g. public administrations, multiple division structures, franchising etc. This autonomy can lead organization units to define their own measures and hence to heterogeneous indexes definition. As another source of heterogeneity, indexes can change in time, due to different analysis needs, or modified external and internal conditions like changes in enterprise rules or national/international laws. In these scenarios, only advanced tools for making the calculus of an index explicit would allow managers to discover subtle discrepancies between apparently similar indexes, enhancing communication, comprehension, and reconciliation of analyses produced by different units or at different times. However, at present this information ultimately lies in IT worker's mind and in procedures implementing the ETL process. Hence, in order to fully support managers' activities, information contained in

DOLAP'08, October 30, 2008, Napa Valley, California, USA.

a data warehouse and in particular strategic indexes should be semantically enriched, associating each information to an explicit and formal description of its meaning and its derivation process. Nowadays, semantic enrichment is almost a synonym of annotating source data with formal descriptions of concepts in a domain ontology, and it is mainly considered in the Semantic Web [1] scenario. From the original definition, the Semantic Web vision has pervaded the research in database and information systems, and it is applied to data and information structures as different from Web pages as scientific documents, databases, business processes. In this paper we propose a model for semantic annotation of datacubes that takes into account domain ontologies as well as a mathematical ontology, that is the formal representation of formulas and a conceptualization of the mathematical domain. The choice to discuss the model for datacubes is guided by two main reasons: *(i)* datacubes are the basis for OLAP analysis and represent the information directly available to managers; *(ii)* being the datacube derived from the DW on the basis of specific analysis requirements, in a datacube schema we can find more information than in the DW schema, e.g. derived and forecasting indexes. Hence, our approach could be easily applied to the whole DW schema as well.

The rest of the paper is organized as follows. Section 2 discusses related works, section 3 gives a brief introduction to the notion of semantic annotation. Section 4 provides a discussion about the kind of information contained in a datacube and how it can be semantically enriched. Then, in Section 5 the semantic annotation model is proposed, and in Section 6 an illustrative case study is provided. Finally, Section 7 ends the paper.

2. RELATED WORKS

The work presented in this paper can be set in the research area on metadata management. Metadata and metadata management have been recognized as an essential element of a warehousing architecture since the beginning [3, 24]. In [3], Chauduri classifies metadata into three categories: administrative, business and operational. Administrative metadata are necessary to set and use the warehouse. They include the description of source databases as well as the data warehouse schema. Business metadata includes business terms, while operational metadata includes information collected during the operation of the data warehouse. It is now generally accepted a distinction between low-level *technical* metadata and higher-level *semantic* (or business) metadata. Another distinction regards descriptive versus transformational metadata [24]. The former category includes information related to *structures* (of data sources, data warehouses, and data marts), while the latter consider information associated with *data processing*.

Technical metadata management have been studied mainly for the purpose of integration of heterogeneous sources [2, 21]. More recently, in the Model Driven Architectures framework the definition of meta-models is considered. For instance, the Object Management Group, facing the problem of interoperability and integration of data warehouses, introduced the Common Warehouse Model (CWM) [6] as a way to exchange metadata via a shared metadata model. In [8] the Behavioural Meta-model, part of the CWM standard, is used to describe business indicators in order to abstract from database implementation and enhance interoperability between different software products. The work goes in the direction of representing transformational metadata, however it remains at the technical level, without considering the semantic level. Semantic metadata are intended for business end users, who are not familiar with warehouse description formats and need a business-oriented view on technical metadata. Semantic metadata enables more effective analysis and increases end user's confidence. In the direction of linking technical and semantic metadata, [24] proposes a unifying UML model. Similarly, [15] introduces a weaving model between enterprise goals and DW data, in order to enhance the way users access and interpret data in the DW. Both models are defined at a conceptual level by UML class diagrams. Techniques borrowed from the semantic web domain, and in particular the use of domain ontologies, have been recently applied to DW, for linking the technical and semantic levels. In particular, in [7] these techniques are used to represent hierarchical relationships among datacubes. Support to datacube design is considered in [20, 26], while improvement OLAP functionalities are presented in [11]. However, as discussed above and in [4] strategic information has peculiar characteristics that require novel techniques and models, since traditional mapping of terms to ontology concepts cannot by itself express the whole body of semantic information appearing at the strategic level. Rather, languages and techniques should be defined which are expressive enough to semantically describe transformational metadata.

3. SEMANTIC ANNOTATION

Webopedia (*http://www.webopedia.com*) defines an *annotation* as "A comment attached to a particular section of a document. Many computer applications enable you to enter annotations on text documents, spreadsheets, presentations, and other objects. This is a particularly effective way to use computers in a workgroup environment to edit and review work. . . ".

Annotations are used to describe the content of "something" and therefore can be considered as metadata. Annotations may be provided under different forms, ranging from a completely unstructured text to formal structures. Also, they can be embedded in the annotated object or be reachable through links. We are especially interested in formal *semantic annotation*, that is annotation expressed in a given structured language which have also a formal semantics. In fact, the more formal the semantics of the language is, the more the machine-readability of the annotation increases.

We distinguish an annotation language to describe the schema of the annotation and an annotation content expressing the semantic information carried by the annotation itself. As a matter of fact, an annotation could simply be a link to a Web resource, and in this case the meaning of the link is evident. However one like to describe more characteristics of the annotation (e.g. the author of the annotation or the date when the annotation has been written). In this case a language to organize such characteristics is needed, which is also shared by the annotation provider and the annotation consumer. This is typically done by exploiting XML-based languages. In Section 6 an example of such a language is given.

In addition to the annotation definition language used, a common understanding of the annotation content is needed. Typically such common understanding is given by referring to one or several *domain ontologies*, that provide "a representation of a shared conceptualization of a particular domain" [22]. It means that the conceptualization has to be agreed by the annotation providers and by the annotation consumers, and that the annotation contents are linked to concepts in the ontology. Ontologies are formally expressed by logical languages, in particular the Web Ontology Language (OWL) is widely adopted in the literature.

4. STRATEGIC DATACUBE SEMANTICS

A *datacube* consists of a set of measures (i.e. indexes) and a set of dimensions with respect to which a measure is analyzed by exploiting OLAP functionalities. In this section we first analyze the different kinds of information associated with each element of the datacube schema, and then we discuss techniques suitable for making this information explicit.

First of all, we like to note that the two basic elements of a datacube carry different kinds of information. A dimension is typically a multilevel hierarchy of simple concepts. Typical dimensions are temporal and geographical dimensions. Other dimensions involve domain specific concepts like a taxonomy of products, types of project, classes of cost, and so on. The name of any dimensions and of their levels is the key to associate them with the meaning they assume in a specific domain, e.g. a "Boing 737" is a kind of airplane, a "fiscal year" is a period used for calculating annual financial statements, etc. Furthermore, starting from the name, and exploiting their own knowledge, managers can deduce more properties like the fact that an airplane is formed by fuselage, engines, wings, cockpit, landing gears, etc; or that a fiscal year is different from a solar year, is formed by days and months, and so on.

Regarding measures, besides the meaning of terms used to represent them, there are other kinds of properties that are to be taken into account, like the unit of measurement and the data type. Also, in a strategic analysis an index is presented as a high level synthetic measure, that is aggregated at different granularity levels and recursively derived from other indexes. So, the meaning of an index is completed with the aggregation function (e.g. sum, average, min, max) and the derivation rule used to compute the index itself. Moreover, a particular kind of strategic index is a forecasting index, that estimates quantities which are not already known, e.g. the net income for the next month. Fundamental information to understand the meaning of a forecasting index are the forecasting model (i.e. the derivation rule) and the actual method (for instance, linear or non linear regression) used to estimate the unknown quantity, on the basis of known data.

From this brief analysis of the kinds of information contained in a datacube, we derive a classification of the semantic content with respect to the semantic annotation techniques that can be used to describe it, as follows: (a) Metric unit information (b) Meaning of a term (c) Temporal and Geographical information (d) Aggregation and derivation rules.

Metric information is perhaps the most standardized and well-known kind of information. Probably it is not necessary to define specific ontologies. Simple annotations reporting the symbol of the unit measure (\$, %,...) might suffice. In some situation, information about unit conversion (e.g. from Dollar to Euro, or from inch to centimeter) might be useful. In this case, one can express a fixed conversion or a variable conversion. If a variable conversion must be reported, like in the exchange rate, it is preferable to establish a link to an external, official exchange organization. Let us note that, despite its simplicity, information about unit measure gives important insights to understand the meaning of numbers in a report.

The meaning of terms can be made explicit by linking the terms to concepts of a common domain ontology. Different ontologies can be used to fully express terms semantics. For instance, the Edinburgh Enterprise Ontology (EO) [23] can be taken into consideration for describing products and relationships among products and their components, and the FEF Financial Ontology [16] can be used to explain the meaning of financial and economic concepts.

The semantic content of temporal and geographical dimensions, can similarly be provided by resorting to ontologies. We distinguish this information from the terms meaning since special relations among concepts exist, like sequence of events or interval overlapping in the temporal domain, or adjacency of regions in the geographical domain. As an example, the OWL-Time ontology [19] describes temporal concepts and properties like interval, instant, sequence of events, and so on.

In order to describe formally and without ambiguity aggregation and derivation rules, we have to make the underlying operational semantics explicit. It can not be described by traditional domain ontologies. This information can be given by a link to a suited mathematical ontology. Regarding forecasting indexes, information about method and model has to be given by referring to formal descriptions of both. Formal description of the method means the description of the process which leads to the estimate, given in some process description language. It can refer either to an abstract description of the method or to the actual program which performs calculation. Finally, formal description of the model means for instance to give the parameters of the linear/non linear equation interpolating known data. Model description languages like the Predictive Model Markup Language (PMML) [5] can be used to this end.

Concluding, it results that in order to give a semantic annotation of any elements of the datacube schema, it is needed to refer to various domain ontologies and to a mathematical ontology. With mathematical ontology we mean the formal representation of implicit formulas of the datacube schema and a conceptualization of the mathematical domain.

5. SEMANTIC ANNOTATION OF DATACUBES

This section presents the model we propose for the semantic annotation of a datacube. An overview of the proposal is shown in Figure 1. The main idea represented in the figure is that the semantics of a datacube is made explicit by linking any element of the schema to both domain and mathematical ontologies. We group together the different domain ontologies in the *Business Ontology*, while pull apart the *Mathematical Ontology* on the basis of the formal model used to represent it. As a matter of fact, domain ontologies, expressed in description logics, are not able to semantically describe a mathematical formula, with its operators and operands. These ontologies contain the description of any explicit or implicit elements of the datacube.

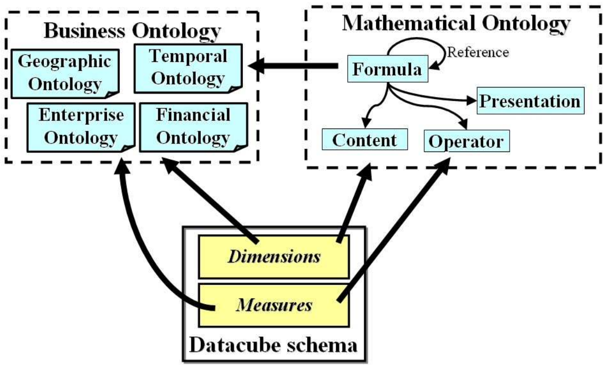


Figure 1: The semantic annotation model.

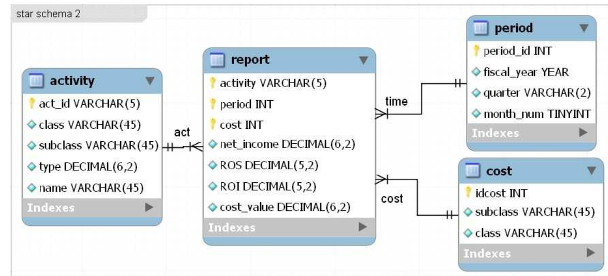


Figure 2: Case study: the datacube star schema.

We hypothesize, without loss of generality, that the schema of the datacube is described in XML. In this way, we can annotate each dimension and each measure of the datacube by simply adding attributes to the related XML element. In particular we introduce two optional attributes: the first attribute allows us to link the datacube element to a concept in (one of the ontologies of) the Business Ontology, while the other attribute points to the formula underlying the element.

The main concept in the Mathematical Ontology is the *Formula*, that is described by the name, its presentation and the formal description of its operators and operands. The formula presentation describes how the formula should be visualized. In order to give a simple example, let us consider the formula for the variance of the price of a given product, its presentation is: $\frac{1}{N}\sum_{i=1}^{N}(p_i-\mu)^2$, where p_i are prices and μ their average value. The operators of this formula are product, summation, difference and raising to the second power; while operands are N, p_i and μ. Each operand can be described both by referring to a concept of the Business Ontology that helps to understand its meaning (e.g. p_i is a price, that is a property of a product,...), and recursively by a link to another formula that explains how the operand is computed (e.g. μ is linked to the formula for the average price, that has a name, a presentation, and so on). In order to complete the formula description, the Mathematical Ontology also represents for any operators their scope (e.g. the difference is applied to p_i and μ) and mathematical meaning (e.g. raising x to the second power is equivalent to multiply x by itself, it is the inverse of the square root, if x is zero then then the operation returns 1, etc). Finally, the formula itself can be contextualized in the domain by referring to a concept of the Business Ontology. In the proposed model, the semantic annotation of the formula is based on the exploitation of recent standards for the semantic description of mathematical formulas, namely the MathML [25] and the OpenMath [14] standards.

MathML (i.e. *Math*ematical *M*arkup *L*anguage) is an XML-based language conceived for the description of both the format and the content of a mathematical formula. To this end, MathML offers two different kinds of markup: *presentation* markup and *content* markup. Presentation and content markups can be used either separately or jointly, depending on the goal of the markup.

Differing from MathML, the OpenMath standard is solely concerned with the semantic representation of mathematical objects. The main difference with the MathML content markup is that OpenMath allows also to describe the semantics of mathematical operators by an XML-encoded collection of mathematical symbols and their definition called *Content Dictionary (CD)*. Both standards suggest to describe the semantics of a mathematical formula by exploiting the MathML content and the OpenMath CD jointly.

Referring to Figure 1, MathML is exploited for the *presentation* level of the formula and for describing its *content*, while OpenMath CD is used for giving an ontological description of *operator*s.

In an organization usually the task of defining a new datacube is entrusted to the IT department. In the same way, in our vision the IT worker is the annotation provider, that is responsible for manually populating the Mathematical Ontology. Although the formal explicitation of a formula is a time-consuming task, we hasten to note that activity has also two main advantages: it contributes to make explicit in a formal way the knowledge patrimony of the organization, and allows us to develop mechanisms to support the design of a datacube and new OLAP functionalities. Furthermore, note that various user-friendly graphical formula editors for MathML and OpenMath exist [17, 18].

In the next section, we present and discuss a case study in order to give a practical example of the annotation model and of how this annotation can be exploited to design tools supporting strategic analysis.

6. CASE STUDY

Let us consider the datacube star schema in Figure 2, that is designed for the financial evaluation of activities of an organization. In particular, the analysis dimensions are the taxonomy of an activity, the time and kind of costs.

The fact table “report” is characterized by the cost chargeable to an activity, the income from sales (net_income), the Return on Sales (ROS) and the Return on Investment (ROI). The last two measures are two of the most adopted financial indexes: ROS measures the operating profit margin of an activity, defined as the ratio between the revenue and the net_income, while ROI can be defined as the ratio between gain (or loss) on an investment and the value of the investment itself. In particular in this example, we assume that ROI is computed according to the following formula:

$$ROI = ROS \cdot \frac{net\_income}{total\_costs}.$$

Note that this is only one of the possible formulas for ROI proposed by the economic literature [9, 13].

```
<Schema>
  <Cube name="case_study">
    <Table name="report"/>
    <Dimension name="Time" foreignKey="period">
      <Hierarchy hasAll="true" allMemberName="Every period" primaryKey="period_id">
        <Table name="period"/>
        <Level name="Fiscal Year" column="fiscal_year" type="Numeric"
               uniqueMembers="true" BO="http://www.w3.org/2006/time#fiscal_year"/>
        <Level name="Quarter" column="quarter" uniqueMembers="false" BO="http://www.w3.org/2006/time#quarter"/>
        <Level name="Month" column="month_num" type="Numeric"
               uniqueMembers="false" BO="http://www.w3.org/2006/time#month"/>
      </Hierarchy>
    </Dimension>
    <Dimension name="Activity" foreignKey="activity" BO="financialOntology.owl#project">
      <Hierarchy hasAll="true" allMemberName="All activities" primaryKey="act_id">
        <Table name="activity"/>
        <Level name="Name" column="name" uniqueMembers="true"/>
      </Hierarchy>
    </Dimension>
    <Dimension name="Cost" foreignKey="cost">
      <Hierarchy hasAll="true" allMemberName="All costs" primaryKey="idcost">
        <Table name="cost"/>
        <Level name="Class" column="class" uniqueMembers="true"/>
        <Level name="Subclass" column="subclass" uniqueMembers="false"/>
      </Hierarchy>
      ...
    </Dimension>
    <Measure name="Net Income" column="net_income" BO="financialOntology.owl#sales" MO="net_income"/>
    <Measure name="ROS" column="ROS" BO="financialOntology.owl#PR_ROS" MO="ReturnOnSales"/>
    <Measure name="ROI" column="ROI" BO="financialOntology.owl#PR_ROI" MO="ROI"/>
    <Measure name="Cost" column="cost_value" BO="financialOntology.owl#cost"/>
  </Cube>
</Schema>
```

Figure 3: Case Study: A fragment of the XML (Mondrian) representation of the datacube.

In order to have an XML representation of the datacube schema, in Figure 3 we refer to the schema definition language of Mondrian [10], an open-source project for the development of Data Warehouse servers. The schema in Figure is divided into two main parts, the first one defines the dimensions hierarchies, while in the second part the measures of the cube are defined. In our proposal, we simply extend the Mondrian schema description by adding tags **BO** and **MO**, referring to concepts of Business Ontology and to formulas of Mathematical Ontology respectively.

Note the use of multiple domain ontologies. In particular, we refer both to the public *OWL-Time ontology* for describing levels of the time dimension, and to the *financialOntology*, that is a local and private domain ontology, for the description of other elements of the datacube schema.

Figure 4 shows, as an instance of Formula in the Mathematical Ontology, the formula of the ROI. Each Formula is represented by an XML-based language, where the basic elements of the language are reported in the Figure. First the name of the formula is given, together with an optional reference to the Business Ontology. In the example, the *BO* attribute is used to link the formula named *ROI* to the instance *PR_ROI* of the financialOntology. Then, the <semantics> tag provides a description of the mathematical formula. Observing Figure 4, it is evident the threefold structure used to describe the semantics of a formula. As a matter of fact, Presentation, Content and Operator properties of a formula are described through the three <annotation-xml> tags referring to MathML Presentation, MathML Content and OpenMath languages respectively. Note in particular the combined use of MathML Content and OpenMath: for each operator the former makes its scope explicit through the <apply> tag, and the latter provides the formal description by linking to a specific concept in a specific Content Dictionary. In the example, the *times* operator is applied to the operand *ROS* and the result of the *divide* operator, then the same *times* operator is linked to the homonymous concept in the *arith1* CD, as shown in <OMS>. Finally, for each operand in the formula, the <reference> tag presents optional references to the corresponding formula in the Mathematical Ontology and concept in the Business Ontology, e.g. the operand named *ROS* is associated both with the concept *PR_ROS* in the financialOntology and the formula *ReturnOnSales* in the Mathematical Ontology.

The repeated use of MO and BO attributes allows us to define semantic paths among elements of the datacube schema and the two ontologies. The graph shown in Figure 5 represents the semantic path regarding the concept of ROI. Squares represent formulas in the Mathematical Ontology, ellipses are concepts of the Business Ontology, and hexagons are objects of the datacube schema or components of formulas. In this graph is evident our idea: adding semantics to a Data Warehouse implies the description of both explicit and implicit objects; each object, has both an ontological and a mathematical description.

Annotations allow us to define new functionalities for business manager support, managing mathematical formulas and terms semantics in addition to classical OLAP operators. In particular, at this stage, we have developed a functionality named the *semantic drill down* of indexes. Semantic drill down allows to expand formulas definitions in terms of the business ontology concepts and of the functional relations among them defined by formulas. This enables managers to look at the index definition in terms of business concepts, hiding the technical definition of DW elements, to explore

```
<formula>
  <name BO="financialOntology.owl#PR_ROI">ROI</name>
  <semantics>
     <annotation-xml encoding="MathML-Presentation">
        <mi> ROI </mi>
        <mo> &equals; </mo>
        <mi> ROS </mi>
        <mo> &middot; </mo>
        <mfrac>
           <mi> net_income </mi>
           <mi> total_costs </mi>
        </mfrac>
     </annotation-xml>
     <annotation-xml encoding="MathML-Content">
        <apply>
           <eq/>
              <ci> ROI </ci>
              <apply>
                 <times/>
                    <ci> ROS </ci>
                    <apply>
                       <divide/>
                          <ci> net_income </ci>
                        <ci> total_costs </ci>
                    </apply>
              </apply>
        </apply>
     </annotation-xml>
     <annotation-xml encoding="OpenMath">
       <OMA xmlns="http://www.openmath.org/OpenMath">
           <OMS cd="arith1" name="times"/>
           <OMS cd="arith1" name="divide"/>
           <OMS cd="relation1" name="eq"/>
     </annotation-xml>
  </semantics>
  <reference>
     <obj BO="financialOntology.owl#PR_ROS"
          MO="ReturnOnSales" name="ROS">
     <obj BO="financialOntology.owl#sales"
          MO="net_income" name="net_income">
     <obj BO="financialOntology.owl#total_cost"
          MO="total_costs" name="total_costs">
  </reference>
</formula>
```

Figure 4: Case Study: The formula of ROI in the Mathematical Ontology.

them and to understand their ultimate meaning and correctness. Figure 6 shows a report, built with JPivot, a tool for OLAP analysis of reports derived from Mondrian DWs.

JPivot interface has been enriched, by adding the capability to visualize the formula annotating a given measure. In the example, the ROI formula is visualized. By clicking on each term, if the term is annotated, then the formula is expanded by showing the definition of each element. Drill down of the ROI formula is shown in Figures 7 and 8.

In particular, Figure 7 shows a high level view of the concept of ROI. Besides the formula, the tool reports all the information contained in Mathematical and Business Ontologies. From the former the formula operators and variables are extracted, while from the latter it is shown the meaning, a list of aliases, its relationships and dependencies, and the class hierarchy. Now, we are able to browse the ROI concept, both by expanding the underlying formula and by visualizing the meaning of linked information. In figure 8, the concept of *total_costs* has been expanded and the meaning of *divide* is shown.

Index semantic drill down is the basic drill down functionality. In particular, it is an expansion of the intensional definition of the index. An enhanced functionality is the

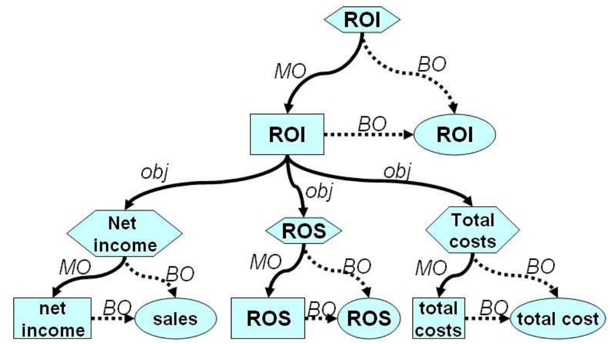
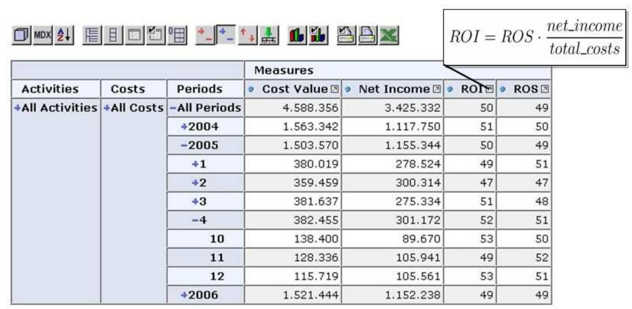


Figure 5: Case Study: Graph of semantic links.



			Measures			
Activities	Costs	Periods	Cost Value	Net Income	ROI	ROS
+All Activities	+All Costs	-All Periods	4.588.356	3.425.332	50	49
		+2004	1.563.342	1.117.750	51	50
		-2005	1.503.570	1.155.344	50	49
		+1	380.019	278.524	49	51
		+2	359.459	300.314	47	47
		+3	381.637	275.334	51	48
		-4	382.455	301.172	52	51
		10	138.400	89.670	53	50
		11	128.336	105.941	49	52
		12	115.719	105.561	53	51
		+2006	1.521.444	1.152.238	49	49

Figure 6: Case Study: A report with a ROI column and its definition.

extensional drill down of the index, that is the visualization of values taken by the operands that compose the index definition. In practice, besides traditional dimensions of multidimensional models, more dimensions are introduced, that represent the different ways a measure can be derived (instead of aggregated). In the example, extensional drill down could be useful to analyze the *ROS* index, whose component elements, namely revenue and the income from sales, are not shown in the report. Allowing extensional drill down the ROS column is decomposed into two columns reporting the values of revenues and sales, allowing to check whether a hypothetical growth of the index is due either to a growth of sales volumes or to a reduced amount of expenses. The implementation of the extensional drill down concept requires to conceive an extended multidimensional model, and to implement a sort of query rewriting system and will be the subject of future research.

Semantic annotation of indexes can be exploited also for formula rewriting, in order to move from a definition to a different one, e.g. the ROS index from ROI definition, provided that a symbolic mathematical reasoner is given. We like to note that the MathML content markup language and the OpenMath CD used to represent semantics of mathematical formulas provide all the information needed to design it. A symbolic reasoner would be capable to simply perform the following transformations:

$$ROI \rightarrow ROS \cdot \frac{net\_income}{total\_costs} \rightarrow \frac{revenue}{total\_costs}$$

by application of the formula

$$ROS = \frac{revenue}{net\_income},$$

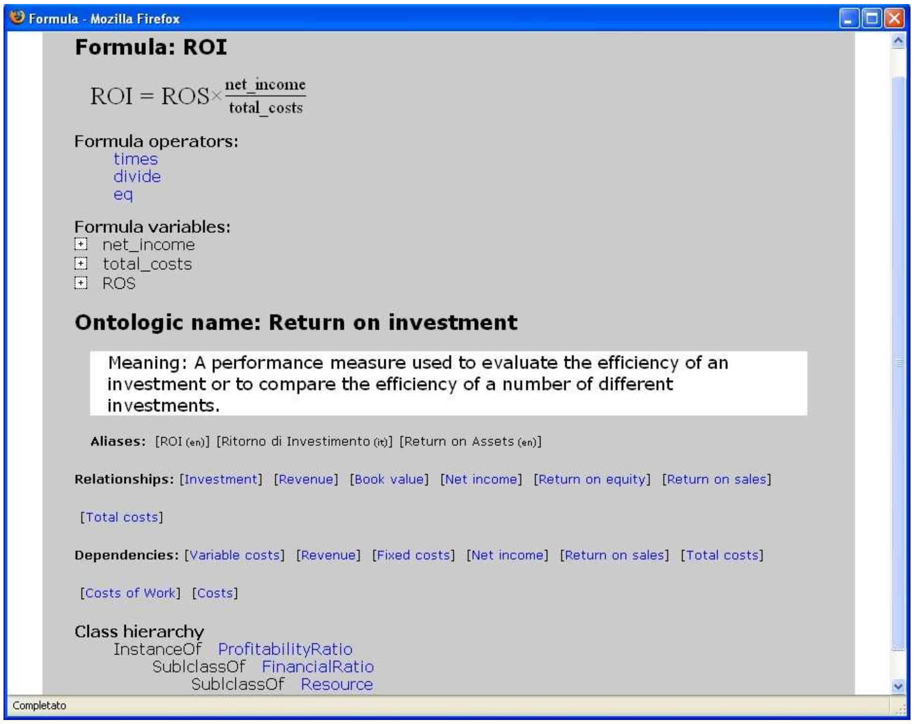


Figure 7: Case Study: semantics of ROI index.

and simplification, and then derive

$$costs \rightarrow \frac{revenue}{ROI} - cost\_of\_work,$$

by application of the formula

$$total\_costs = costs + cost\_of\_work.$$

This is a new formula that is not explicitly contained in the Mathematical Ontology.

7. CONCLUSIONS

In the paper we considered the problem of representing the semantics of strategic datacubes, that is datacubes designed for supporting business managers to assess their strategies and decisions. It is observed that the features of these datacubes and in particular of strategic indexes are such that traditional semantic, ontology-based, annotation techniques are of limited use in explaining the meaning and reconciling heterogeneities. In fact, indexes are compound objects, derived from others, more basic elements. Hence, in order to fully provide index semantics, a representation of mathematical formulas is needed. We then proposed a semantic annotation model that exploits MathML and OpenMath languages in order to formally describe such formulas in a Mathematical Ontology. Operands in formulas and indexes themselves are also linked to concepts of a Business Ontology, in order to solve traditional heterogeneity problems and to provide final users with a business-oriented description of the datacube content. Finally, an example of annotation of a datacube schema is given, considering an extension of the Mondrian XML metadata definition language, and it is shown how annotations can be exploited to perform new kinds of OLAP analyses, like the drill down of index definitions, that is the expansion of formulas, and index comparison. We note that, since a datacube schema contains more information than the Data Warehouse one, e.g. derived and forecasting indexes, our approach can be easily extended to the whole Data Warehouse schema.

In the paper the semantic annotation model and the implementation of a prototype is presented. Future work will be devoted to fully implement the framework and to define novel types of OLAP analyses. In particular, the previously called extensional drill down of indexes will be considered, where the expansion of a formula is used to pass from values of a compound index to the set of values of its component elements.

8. REFERENCES

[1] T. Berners-Lee, J. Hendler and O. Lassila. The Semantic Web. *Sci. American*, 284(5):34–43, 2001.

[2] D. Calvanese, M. Lenzerini, D. Nardi, and R. Rosati. Source Integration in Data Warehousing. *Int. Jour. of Cooperative Information Systems*, 10(3):237–271, 1998.

[3] S. Chaudhuri and U. Dayal. An Overview of Data Warehousing and OLAP technology. *SIGMOD Rec.*, 26(1):65–74, 1997.

[4] C. Diamantini and N. Boudjlida. About Semantic Enrichment of Strategic Data Models as Part of Enterprise Models. In *Proc. of the BPM'06 Int. Workshop on Enterprise and Networked Enterprises Interoperability*, Vienna, Sept. 2006.

[5] R. Grossman, M. Hornik, and G. Meyer. Evolving Data Mining into Solutions for Insights: Data Mining Standards Initiatives. *Comm. of the ACM*, 45(8):59–61, August 2002.

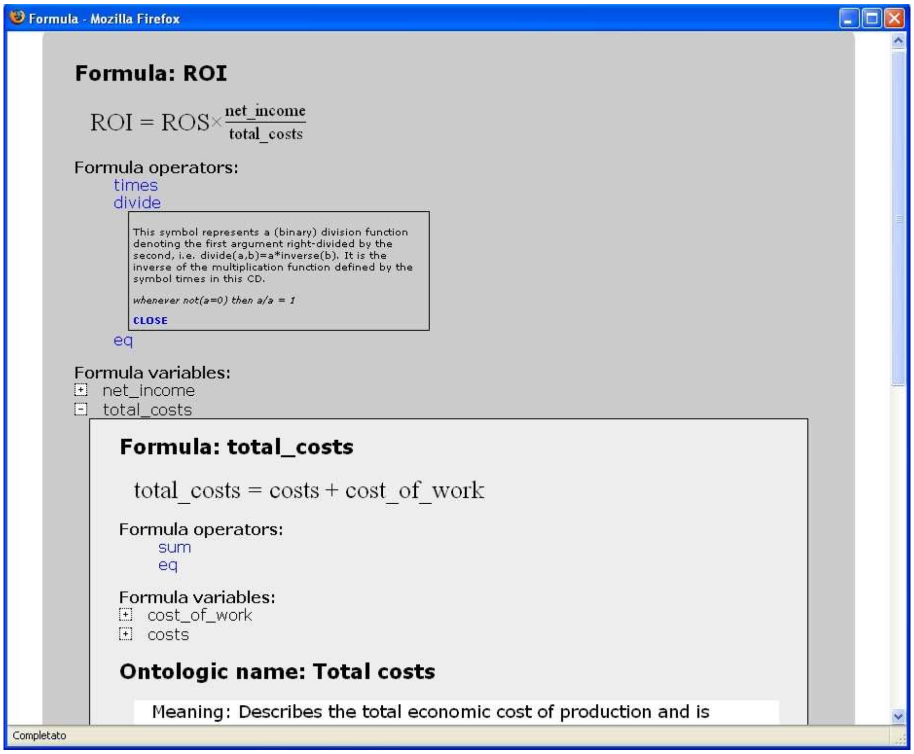


Figure 8: Case Study: ROI formula drill down.

[6] OMG Group. *Common Warehouse Metamodel Specification*. http://www.omg.org/technology/documents/formal/cwm.htm.

[7] S.-M. Huang, T.-H. Chou, and J.-L. Seng. Data Warehouse Enhancement: A Semantic Cube Model Approach. *Information Sciences*, 177(11):2238–2254, June 2007.

[8] A. Januszewski and T. Pankowski. Modeling Analytical Indicators Using Datawarehouse Metamodel. In *Proc. of the 17th DEXA Conference*. IEEE, 2006.

[9] P. Pratali. *Progettare il Profitto*. F. Angeli, 2000.

[10] Pentaho Analysis Services: Mondrian Project site. http://mondrian.pentaho.org/.

[11] T. Priebe, and G. Pernul. Ontology-Based Integration of OLAP and Information Retrieval. In *Proc. of DEXA Workshops*, pages 610–614, 2003.

[12] D. J. Rowley, H. D. Lujan, and M.G. Dolence. *Strategic Change in Colleges and Universities*. Jossey-Bass Publishers, San Francisco, CA, 1997.

[13] K. H. Silber. Calculating Return-On Investment, 2002.

[14] The OpenMath Society. *The OpenMath Standard Version 2.0*. http://www.openmath.org/, 2004.

[15] V. Stefanov and B. List. Business Metadata for the Datawarehouse. Weaving Enterprise Goals and Multidimensional Models. In *Proc. 10th IEEE Int. Enterprise Distributed Object Computing Conference Workshops*. IEEE, 2006.

[16] The Financial Exchange Framework Ontology. http://www.financial-format.com/fef.htm.

[17] The Java OpenMath Editor. http://www.activemath.org/projects/Jome/.

[18] The W3C MathML software list. http://www.w3.org/Math/Software/.

[19] The W3C Time Ontology in OWL. http://www.w3.org/TR/owl-time/.

[20] S. Toivonen and T. Niemi. Describing Data Sources Semantically for Facilitating Efficient Creation of OLAP Cubes. In *Poster Proc. of the 3rd Int. Semantic Web Conference*, 2004.

[21] F. S. Tseng and C.-W. Chen. Integrating Heterogeneous Data Warehouses Using XML Technologies. *Jour. of Inf. Science*, 31(3), 2005.

[22] M. Uschold and M. Grüninger. Ontologies: Principles, Methods and Applications. *The Knowledge Engineering Review*, 11(2):93–155, 1996.

[23] M. Uschold, M. King, S. Moralee, and Y. Zorgios. The Enterprise Ontology. *The Knowledge Engineering Review*, 13, 1998.

[24] A. Vaduva and K. R. Dittrich. Metadata Management for Data Warehousing: Between Vision and Reality. In *Proc. Int. Symp. on Database Engineering & Applications, 2001*, pages 129–135, 16-18 July 2001.

[25] W3C Math Working Group. *Mathematical Markup Language (MathML) Version 2.0 (Second Edition)*. http://www.w3.org/Math, Oct. 2003.

[26] G. Xie, Y. Yang, S. Liu, Z. Qiu, and X. Zhou EIAW: Towards a Business-friendly Data Warehouse Using Semantic Web Technologies. In *Proc. of the 6th Int. Semantic Web Conference*, Busan, Korea, Nov. 2007.