

# OntoDW: An approach for extraction of conceptualizations from Data Warehouses

Tiago Outerelo da Silva, Fernanda Baião, Kate Revoredo

Department of Applied Informatics -  
Federal University of the State of Rio de Janeiro (UNIRIO)  
Avenida Pasteur 458 – Urca – CEP 22290-240 – Rio de Janeiro - RJ

{tiago.dasilva,fernanda.baiao,katerevored}@uniriotec.br

**Abstract.** *Business Intelligence (BI) fosters proper decision-making in organizations, mainly by providing the means to analyze historical data stored in repositories called Data Warehouses (DW). However, formal representation of which concepts are implemented in a DW rarely exists, which would be important to clarify and semantically describe the domain concepts behind the data stored in a DW, as well as the analytical concepts that are available for the BI tools. Examples of important pieces of knowledge that are frequently hidden into the DW are: which domain concepts are available as analysis perspectives (dimensions), how the domain concepts relate to each other, which metrics (facts) are available and what do they mean, which domain perspectives are considered for each metric and how metrics may be aggregated. On the other hand, one of the relevant uses of an ontology for the Computer Science area is as a codified artifact that formally represents a shared conceptualization about a universe of discourse. Therefore, ontologies can be used to represent both domain and analytical concepts codified and stored in a DW. However, extracting these concepts from an already-in-production DW is not a trivial task, especially in medium and large organizations, often with tens of metrics and tens (even hundreds) of dimensions and potential aggregations. In this paper, we define a set of mapping rules from DW constructs to conceptual elements (concepts and relationships), towards automatically extracting an ontology codified in OWL. The proposal was successfully evaluated in a real scenario of a Brazilian financial institution.*

## 1. Introduction

Organizations are overloaded by the increasing amount of data, which is continuously generated and stored in corporate repositories, to be analyzed for proper decision-making [Sidorova and Towers 2014]. Definitions of business strategies, decisions on product prices and customer behavior trends are examples of scenarios benefiting from this data analysis [Andoh-Baidoo et al. 2014]. Business Intelligence (BI) solutions provide the means to gather information and to derive knowledge through analysis tools for decision-making [Sell et al. 2011]. They help in the analysis of large volumes of data, transforming them into meaningful, useful and enlightening information.

Despite the importance of analytical tools provided by BI solutions for current organizations, there are challenges to leverage their impact on the decision-making process [Sell et al. 2011]. Users do not have a clear definition of all information at their disposal, not even of the possible relations among the available data. This may occur due

to the lack of integration between business semantics artifacts and the analytical tools [Sell et al. 2011]. Towards this integration, a formal representation may be used to semantically describe the concepts implemented in the BI solution.

On the other hand, a relevant use of ontologies for the Computer Science area is as an artifact that formally represents a shared conceptualization of a domain of discourse, through its key elements: concepts and relationships. Therefore, it is a natural artifact to describe the the semantics behind the data and metadata stored in a DW, thus providing a rich, explicit and upgradeable conceptual representation of the organizational data. Therefore, it would be very useful an application ontology describing the concepts implemented in the DW, in order to support the data analysis task, to explicit the concepts available and the relationships between them and the possible operations to be performed.

In this paper, we define a set of mapping rules from DW structural constructs to conceptual elements (concepts and relationships), towards automatically extracting an ontology codified in OWL, within BI systems. The proposal was applied in a real scenario.

This article is organized as follows: Section 2 presents the theoretical basis of the work, Section 3 presents our proposal, Section 4 describes the application scenario and the application example and Section 5 describes related works on automatic generation of ontologies from relational databases. Section 6 contains final considerations.

## **2. Business Intelligence environments**

Business Intelligence (BI) is a set of theories, methodologies, architectures and technologies in order to recover and transform raw data into meaningful and useful information, allowing the operational, tactical and strategic levels of an organization make better and more agile decisions [Airinei and Homocianu 2009]. Briefly, BI systems integrate data from multiple sources to generate information to support decision making. Among the components which form a BI environment, we can highlight the OLAP tools and the Data Warehouse.

OLAP (Online Analytical Processing) is the ability to analyze and manipulate information from multiple perspectives. OLAP tools provide its users with this capability through interactive interfaces that enable the execution of analytical operations defined as OLAP operators.

[Inmon 2002] define a Data Warehouse as a subject-oriented data collection, nonvolatile, integrated and time-variant, to support decision-making. A DW is subject-oriented because data relates events or objects of real life; it is nonvolatile because data is not updated or deleted; it is integrated because it merges information from several different sources; and it is time-variant because data is presented with historical views. A Data Warehouse is built by integrating information from the organization's business processes, from different sources of information and the holding of periodic loads.

The multidimensional modeling is a subject-oriented data modeling technique, widely used in BI environments in Data Warehouse projects. The basic elements of multidimensional models are *facts* and *dimensions*. Facts are indicators (measures / metrics) to be analyzed and dimensions are analytical views on stored facts. Facts and dimensions are stored in different tables. A particular type of fact table is the Factless

Fact table, because it has no measure columns. The information represented is the relationship between elements of dimension tables referenced by the fact table.

In a DW implemented in a relational database, fact tables are related to dimension tables. In the star schema type model, the dimensions are denormalized tables and each can store multiple levels of the same analysis. An example would be the **Time** dimension, which can store in a single table the analysis (dimension levels) **Day**, **Month** and **Year**. In snowflake type model, the dimensions are normalized tables and each table is a analysis level. For the same example where the **Time** dimension was used, there would be a table (dimension) for the analysis **Day**, related to another table to **Month**, and this related to another table for **Year**. Both the types, or a hybrid of them, are used in DW implementations and shows that exists a diversity in their data structure that can make the identification of concepts stored a complex task to the BI analysts (IT analyst responsible for BI systems).

In typical BI systems scenarios, a knowledge representation would allow a business analyst or a BI analyst to know the measures available for analysis, for which analysis views are available (granularity / summarizability) and the relationships between these views, for example.

### 3. DW to Ontology

Consider a real scenario of a BI system about employees of a financial institution participating in a pension fund. Business analysts are provided with an OLAP tool for building analysis, reports and interactive dashboards. However, business analysts are highly dependent on IT specialists to build reports and perform analysis due to difficulty in knowing the information available and crossing possibilities between them. However, the IT department does not have sufficient availability to attend the demand and the available documentation does not follow the changes that occur in the technological environment. In this scenario, a description of the concepts and their relationships available for analysis in the BI environment provide the users with knowledge allowing them to make better decisions. Moreover, it can also help BI analysts to explain discrepancies between the data schema and the application layer and support data integration demands.

For instance, suppose that the business area needs to perform an analysis about the financial impact of employees retirement. The retirement value of an employee is based, among other variables, on the amount of his salary. With the available OLAP tool, a business analyst easily visualizes a metric with the employees salary. However, other information such as the grouping or filtering possibilities of such information, the possible dimensions of use or the granularity of information are not provided. Additionally, other information related to the chosen one would provide the analysts with more insights towards a better decision, such as the quantity of dependents, gender or city of residence. An ontology could represent **Salary** and **Quantity of dependents** as metrics associated to a temporal dimension and to other dimensions such as **Gender** and **Residence city**. Thus, this ontology can be used as an artifact for providing extra knowledge towards a better decision.

In this work, we propose OntoDW, an approach for automatically extracting an ontology from the Data Warehousing structural constructs (schema metadata) and contents (data) within a BI system. The elements of the generated ontology are obtained by specific mapping rules of our proposed method. The hypothesis for this proposal is

that it is possible to generate ontologies from data warehouses through the use of specific mapping rules, and this ontology will reflect the knowledge about OLAP analysis task present in the data and metadata of the Data Warehouse.

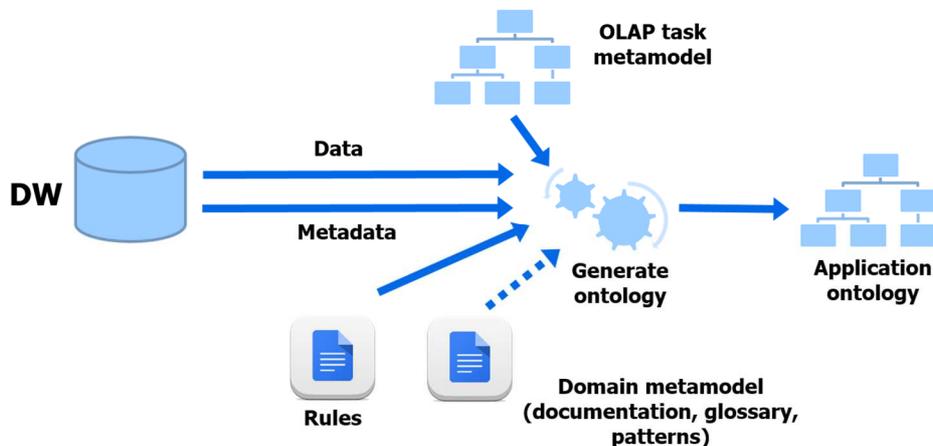


Figure 1. Proposed process for ontology generation

The obtained ontology should include not only explicit knowledge about the data structures (such as translating from tables to classes, for example), but also about the semantics (such as class categorization). A domain ontology comprises the concepts present in the Data Warehouse, without specifying the possible operation possibilities to perform. For this reason, the proposed solution generates an ontology and includes classes relating to an OLAP task metamodel.

The generated ontology will be composed of concepts that reflect the multidimensional data schema (fact tables and dimension tables) and concepts associated with the analysis operations in BI systems (such as summarizability, the possible analyzes to perform). For this, the following input elements for the ontology generation process are used: the DW, an OLAP task metamodel, a domain metamodel and the set of mapping rules defined in this proposal, as in Figure 1.

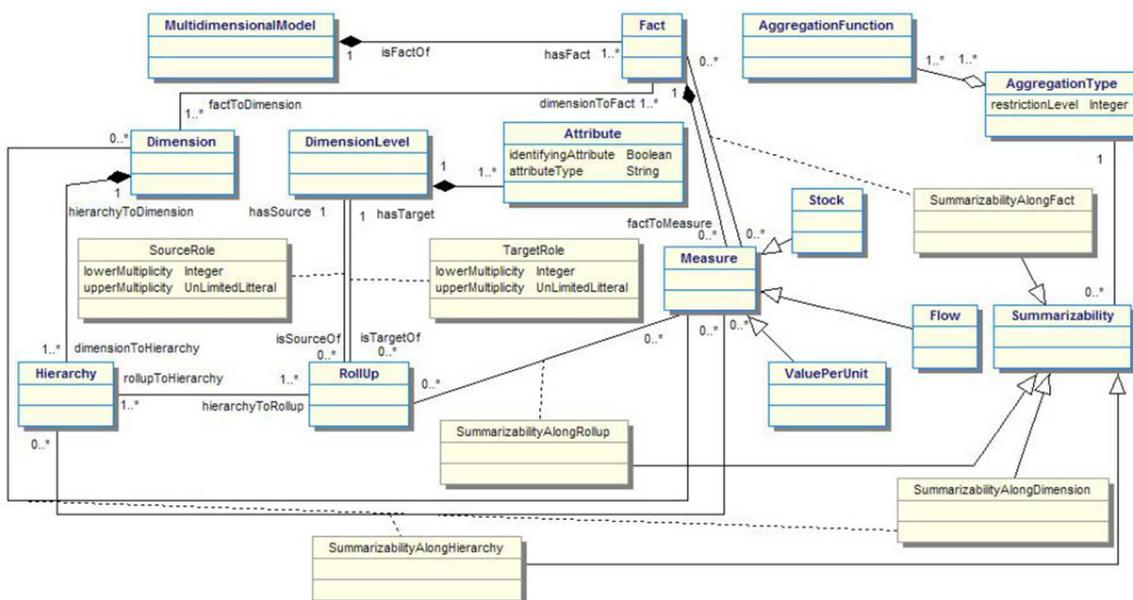


Figure 2. OLAP task metamodel [Prat, Megdiche and Akoka 2012]

The Data Warehouse metadata (logical schema of the database) is taken into account by the proposed mapping rules. Additionally, access to DW data is required in cases where the metadata does not provide enough information to identify concept instances.

The OLAP task metamodel presents predefined concepts and relationships associated to information analysis in BI systems, such as Measure and Dimension concepts. In this work, we adopted the OLAP task metamodel proposed by Prat et al. [Prat Megdiche and Akoka 2012], illustrated in Figure 2.

The domain metamodel presents domain specific concepts on which the system is included and may be represented by a data dictionary, a terminology standard or a glossary, which are simple components that are traditionally found in organizational environments. The domain information will be used to name concepts according to business terms already established.

The current implementation of our proposal defined mapping rules (described below) to the following concepts of the metamodel: **Fact**, **Dimension**, **DimensionLevel**, **Measure** and **SummarizabilityAlongDimension**. These concepts do not represent all the classes present in the task metamodel, but are the main concepts for a rich and aligned ontology for the BI system.

### 3.1. Mapping rules

This section describes the mapping rules defined from Data Warehouses elements (data and metadata) to some ontologies concepts of the used OLAP task metamodel (Figure 2). The OntoDW ruleset differs from the rules defined by Prat et al. [Prat, Akoka and Comyn-Wattiau 2012] [Prat Megdiche and Akoka 2012] because they do not use data and data structures as input elements, only logical models definitions.

The OntoDW rules for concepts extraction do not contain rules defined by Prat et al. However, for the generation of OWL ontology after concepts identification, some these rules were used, with adjustments. The used rules define the concepts as subclasses of the appropriate classes that represents in the OLAP task metamodel used. Ex.: *“Transformation T2.1: Each dimension of the multidimensional model is defined as a subclass of the class Dimension in the OWL-DL ontology”* [Prat, Megdiche and Akoka 2012]. The T2.1 transformation was used in this work, with the setting of each identified dimension table was defined as an instance of the Dimension class, not a subclass. The definition of concepts such as instances was made for better handling of ontology, with the clear separation of the model relationships of metamodel relationships, and the non-use of instances to represent system data such as records of the dimension table, for example.

#### 3.1.1. Class Fact

It is assumed that there is a Fact (or fact table in the DW schema) for each table that has at least one column as a foreign key, but that is not referenced by any foreign key of another table DW schema. This rule is justified by the own definition of star schema. It is assumed that F is a Factless Fact table if, additionally, there are no numeric type columns outside the primary key.

**Rule R1:** For each table T1, T1 is mapped to a fact F1 if there is not a table T2 (T2 ≠T1) that references T1 via foreign key and T1 references a table T3 via foreign key. Let

$PK=\{C_1, \dots, C_i\}$  be the F1 columns subset that composes its primary key and  $NK=\{C_{i+1}, \dots, C_n\}$  the F1 columns subset that not composes its primary key. For each fact F1, F1 is classified as Factless Fact if there is no X column ( $X \in NK$ ) of numeric type as foreign key.

### 3.1.2. Class Dimension

It is assumed that there is a dimension (or dimension table in the DW schema) for each table that is referenced by a foreign key from another table of the DW schema. This rule is justified by the own definition of star schema.

**Rule R2:** For each table T1, T1 is mapped to a dimension D1 if there is a table T2 ( $T2 \neq T1$ ) that references T1 via foreign key.

### 3.1.3. Class DimensionLevel

A dimension level is a subdivision of a dimension and represents an analysis view. A dimension can have more than one level, if the table from which it was mapped is denormalized. It is assumed that there is a dimension level for each set of columns of a dimension where no column is a foreign key and, for each value of a column of this set, the same corresponding value occurs in another column of this same set. The columns which are foreign keys are not considered because they represent the relationship to another level. The restriction on the columns values is justified by the fact that a record in a dimension level must be unique, like a dimension table.

**Rule R3:** For each dimension D1, let  $ND=\{C_1, \dots, C_j\}$  be a D1 columns subset. ND is mapped to a dimension level N1 if,  $\forall a, C_a$  ( $C_a \in ND$ ) is not a foreign key and,  $\forall b, a$  value in  $C_a$  always has the same corresponding value in  $C_b$  ( $C_b \in ND$ ).

### 3.1.4. Class Measure

There are two scenarios for mapping measures. In scenario 1, it is assumed that in a F1 fact, a numeric type column that is not part of the primary key and that is not a foreign key is a measure. The column must be numeric to enable aggregation operations on its values, such as the sum or average, for example. In scenario 2, if F1 is a Factless Fact, there is a measure M1 with no corresponding column in the DW table.

**Rule R4:** For each fact F1, let  $NK=\{C_i, \dots, C_n\}$  be the F1 columns subset that does not compose its primary key. For each  $C_a$  ( $C_a \in NK$ ),  $C_a$  is mapped to a measure M1 if it is a numeric type column and it is not a foreign key.

**Rule R5:** For each fact F1 classified as Factless Fact, F1 is mapped to a measure M1.

### 3.1.5. Class SummarizabilityAlongDimension

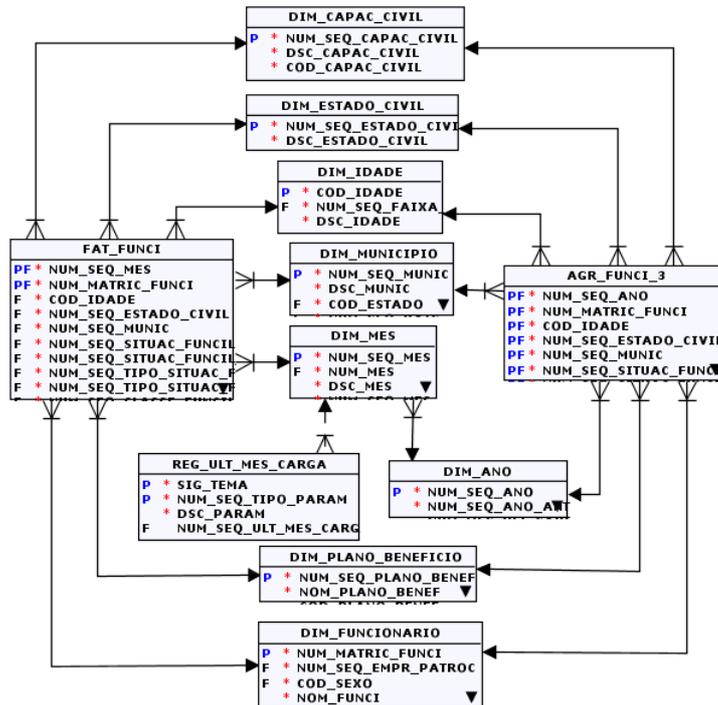
The summarizability along dimensions of a M1 measure represents all M1 relationships with the mapped dimensions through the fact tables that contain M1.

**Rule R6:** Be  $F1=\{F_i, \dots, F_j\}$  the set of all DW fact tables that contain the M1 measure and  $D=\{D_m, \dots, D_n\}$  the set of all DW dimensions. For each measure M1, M1 is mapped to a summarizability along dimension instance AD1 of M1, related to M1 and to dimension D1 ( $D1 \in D$ ), if,  $\forall a, D1$  is related to  $F_a$  ( $F_a \in F1$ ) via foreign key.

## 4. Application Example

### 4.1. Scenario Description

This Section illustrates the application of our proposed solution in the scenario of a pension fund of one of the largest financial institutions in Brazil. The chosen domain comprised employees information, on which we focus in this paper since it is easier to understand for non-experts; moreover, the chosen DW scheme applied a variety of multidimensional modeling techniques and the data stored in the DW is known to be consistent; finally, it is an strategic subject for both the Business Intelligence area of the institution and the business area responsible of this data.



**Figure 3. Part of DW schema model of the application example**

The DW stores data loaded since 1997, totaling tens of millions of records in the tables. This data relates to 275,000 employees and former employees of the financial institution, in various analysis views, composing a rich environment of information that can be used for management actions and analysis on the actuarial calculation and monitoring of the staff. Every month, the registration information of the employees is loaded in the DW and integrated with information about the pension funds in which each employee participates. This DW scheme contains 62 tables (50 dimension tables, 11 fact tables and 1 control table). The DW is implemented on DBMS Oracle 11g, the same platform used for the development of OntoDW.

Figure 3 shows a cutout of the physical model of the data schema. There are 2 fact / aggregation tables (**FAT\_FUNC1** and **AGR\_FUNC1\_3**) and some of the existing dimension tables that relate to them, storing analysis views of the measures / metrics available. A control table that keeps track of which data is already loaded into the DW is also present (**REG\_ULT\_MES\_CARGA**).

The fact / aggregation tables of Figure 3 present only a subset of their columns, since the total number of columns in the tables is very high (50 columns for the

FAT\_FUNCI table). This is due to the high number of dimensions and metrics existing for this subject in the DW and also to the fact that it is an old data structure, which has undergone several evolutionary and corrective maintenance changes. This makes it more difficult for analysts to perform analysis of the model for identifying information in the DW. For example, the **Salary** measure is stored by the **VAL\_SALAR\_PARTIC** column. However, it is present in these two tables with a confusing identification and different possible analysis views from the data model.

## 4.2. Results

To illustrate the results of our application example, we chose an excerpt of the ontology generated by OntoDW on top of the DW schema of Figure 3, related to aggregability along dimensions. It contains conceptual elements representing possible analysis from business analysts on top of the data in the Data Warehouse and to present different extracted concepts. The validation criteria for the proposed approach is that the generated ontology reflects the knowledge present in the data and metadata of the DW.

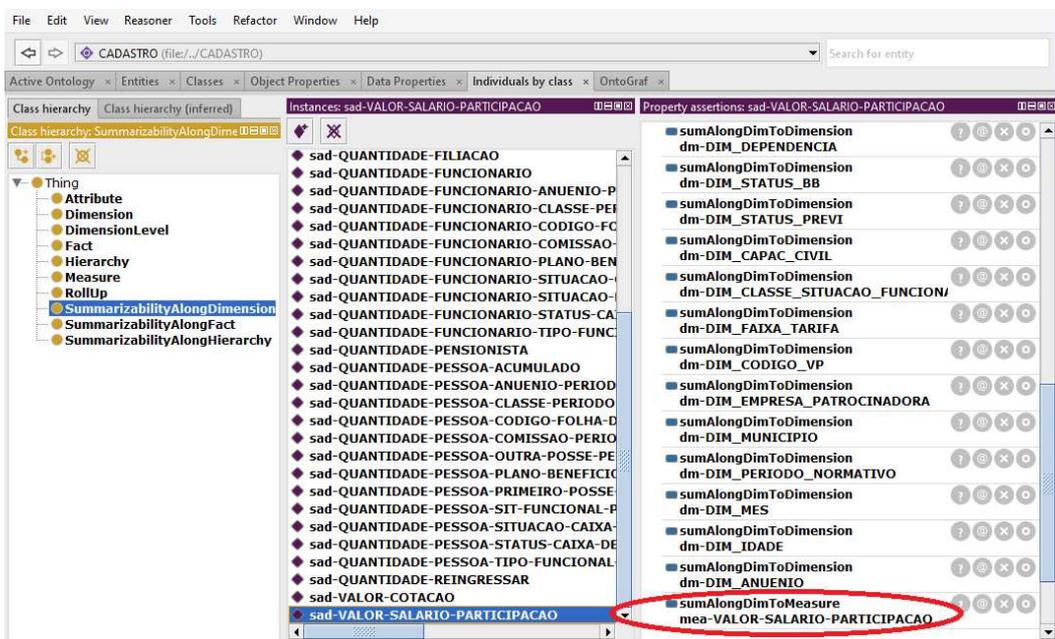
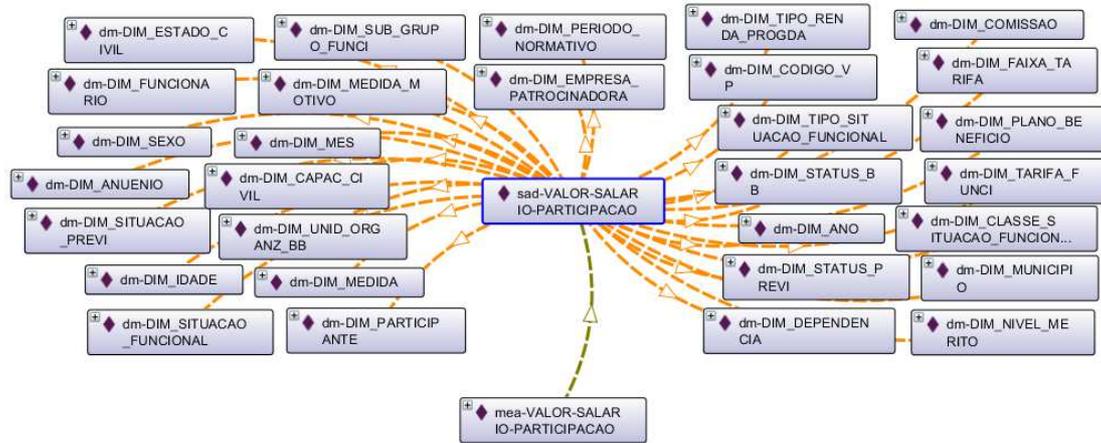


Figure 4. Screenshot from Protégé with ontology instances

Figure 4 is a screenshot from Protégé ([www.protege.stanford.edu/](http://www.protege.stanford.edu/)) showing instances of the **SummarizabilityAlongDimension** class found by OntoDW. The screen display is divided into three parts: the leftmost subdivision shows the classes defined in the ontology, the central subdivision shows instances of the selected and rightmost subdivision shows the metric that has its summarizability represented (highlighted in red) and the dimensions by which it is possible to analyze the metric.

The **mea-VALOR-SALARIO-PARTICIPACAO** measure highlighted in Figure 4 enables the analysis of the employees salary considered by the pension fund to actuarial calculation, benefits payment and revenue collection. The name of the metric was defined from the name of the column in the fact table that stores its data (**VAL\_SALAR\_PARTIC**). Using the separator (“\_”) as a parameter, the terms have been extracted and were consulted in the glossary of organizational business terms; if found, the term is replaced by the original term in the glossary. The terms are then concatenated

with the other separator (“-“), also defined as a parameter. An identifier prefix of the class that defines the instance was also defined (“mea”), to help BI analysts.



**Figure 5. Screenshot from Protégé with part of the resulting ontology**

The named instance **sad-VALOR-SALAR IO-PARTICIPACAO** (where “sad” stands for “summarizability along dimension”) relates to **mea-VALOR-SALAR IO-PARTICIPACAO** measure, connecting it to all its possible dimensions for analysis. Figure 5 is an excerpt of the resulting ontology and presents the same instances of Figure 4, but represented graphically by OntoGraf plugin of Protégé.

Considering all the concepts that were automatically explicated by OntoDW in the generated ontology, the application example is considered successfully performed. From the data and metadata of the DW, the instances of classes **Fact**, **Dimension**, **DimensionLevel**, **Measure** and **SummarizabilityAlongDimension**, and the relationships between them, were mapped. With the graphically represented ontology, the identification of the instances and the relationship among them can be made more quickly and intuitively. Returning to the example analysis described above, the BI analyst can easily deduce from Figure 5 the metric analysis possibilities associated with salary amount in relation to the dimensions found in DW.

## 5. Related works

We carried out a bibliographic search looking for works related to the problem in question, and found some studies that have explored the generation of ontologies from data structures. Although there are the different approaches in the literature for automatic generation of ontologies [Prat, Akoka and Comyn-Wattiau 2012] [Prat Megdiche and Akoka 2012] [Dou, Qin and Lependu 2010], such approaches require the existence of other external data sources, out of the BI system, such as data models or other ontologies.

Prat et al. [Prat, Akoka and Comyn-Wattiau 2012] [Prat Megdiche and Akoka 2012] address the generation of OWL-DL ontology from a multidimensional data model. They, however, premised on the existence of a conceptual data model, which poses a huge limitation for its applicability in practice. Moreover, our proposed ruleset differs from the rules defined by Prat et al. [Prat, Akoka and Comyn-Wattiau 2012] [Prat Megdiche and Akoka 2012] because they do not consider the DW data and metadata, only logical models definitions, and their rules do not extract BI systems concepts, only maps concepts already identified in the logical model to concepts in the output ontology.

Gil et al. [Gil and Martín-Bautista 2014] [Gil, Martín-Bautista and Contreras 2010] present a methodology for ontology learning (SMOL) composed of phases over a structured process. However, techniques or methods to the generation of ontology and process steps are not described.

[El Idrissi, Baina and Baïna 2013] present a practical survey of methods using databases structures as inputs to the ontology learning process. The authors conclude from this survey that there is no tool that automatically extracts an ontology from the database structure.

[Dou, Qin and Lependu 2010] proposed a framework for automatic discovery of mapping between database schemas and ontologies, and a query translation algorithm, but does not provide the generation of ontology with the application concepts. This framework, given different ontologies or schemas and their associated data, will be able to mine a set of first-order mapping rules that describe how the input ontologies or schemas relate to each other. Therefore, it is expected that there is an initial system ontology to generate an output ontology.

Moreira et. al [Moreira et. al 2014] [Moreira et. al 2015] presents an ontological approach for the derivation of multidimensional schemas, using categories from a foundational ontology (FO) to analyse the data source domains as a well-founded ontology. Initially, a domain ontology is created and this ontology is derived to a database schema. This approach has two limitations that does not allow the use in the solution presented in Section 3. The first limitation is that the approach includes only multidimensional modeling concepts, leaving out the concepts of OLAP applications. The second limitation is that the generation of the multidimensional tables in the database schema is always performed with the same technique. To use these rules to reverse process of generating the ontology from the database schema (objective of this work), is necessary that the approach covers techniques present in the models of star schema type and snowflake schema type.

The analysis of the aforementioned studies showed the absence of solutions for generating ontologies automatically from DWs. In particular, the use of another source of data (other than the multidimensional data structure) for generating an ontology would require the existence of up-to-date documentation in synchrony with the concepts implemented to throughout the system life cycle, which is unrealistic in practice. It is very difficult to keep another source of information available for use in the generation of an ontology, that is always current. On the other hand, in a BI system based on a DW, the multidimensional data structure is part of the implemented system.

## **6. Final considerations**

This article proposed a set of mapping rules for automatically generating an ontology for BI systems from Data Warehouses, contributing to solve the lack of a formal knowledge representation that explicitly and semantically describe the data and metadata of BI systems stored in the DW.

Advantageously, the use of DW elements to generate an ontology provides a source of information shared with the BI system, ensuring alignment between the concepts implemented in the system and domain concepts that the extracted ontology proposes to represent. In addition, this source of information allows the inference of BI domain concepts, such as summarizabilities, task more difficult to perform using an operational database. The characteristics of stored data, such as volume and sparsity,

may also be used to infer the ontology elements. For example, an aggregate table of employees by age group tends to be less voluminous than a fact table in the level of employee or age.

The ontology generation from DWs presents as challenges some issues that are inherent characteristics commonly found in BI systems, such as the large volume of data stored in data structures, that make it difficult to manipulate the data stored in the repository and the structure that contain them, and the denormalisation of data models that make it difficult to identify the relationship between the classes and their properties.

This generation of ontology should be automatic because of problems relating to their manual construction. This facilitates keeping the consistency of the ontology with the DW elements along the application life cycle. In cases of changes due to evolutionary system maintenance, when new facts and dimensions are frequently included, the proposed approach may be reexecuted so as to update the existing conceptualization.

The proposed mapping rules extends the state of the art in the generation of ontologies from BI environments. These rules deal with more specific aspects of multidimensional modeling and takes both the data and metadata present in the DW data structures into account.

The contributions of this proposal are the creation and improvement of mapping rules from data warehouses elements to ontology concepts, which address specific aspects of multidimensional modeling and OLAP applications to use the data and metadata in the DW, and implementation of a tool for automatic generation of ontologies, using mapping rules, the system domain information and an OLAP task metamodel, besides the Data Warehouse.

As a future task, a survey with BI professionals will be conducted to evaluate the extraction of concepts rules using their theoretical knowledge and experience. Users of the BI application scenario will also validate the premise that a representation of knowledge can support the analysis of data in the DW. The survey form will present issues containing parts of the generated ontology and the participant shall provide its opinion on the usefulness of representations submitted. .

## **Acknowledgements**

The authors would like to thank FAPERJ (E-26/203.446/2015 - BBP), FAPES, CAPES and CNPq for partially funding their research projects.

## **References**

- Sidorova, A. and Torres, R. (2014). Business Intelligence and Analytics: A Capabilities Dynamization View. In: *Twentieth Americas Conference on Information Systems, Savannah, Georgia, 2014*.
- Andoh-Baidoo, F., Villa, A., Aguirre, Y. and Kasper, G. (2014). Business Intelligence & Analytics Education: An Exploratory Study of Business & Non-Business School IS Program Offerings. In: *Twentieth Americas Conference on Information Systems, Savannah, Georgia, 2014*.
- Sell, D. et al (2011). Adding Semantics to Business Intelligence: Towards a Smarter Generation of Analytical Tools. In: *BUSINESS INTELLIGENCE–SOLUTION FOR BUSINESS DEVELOPMENT*, p. 33, 2011.
- Airinei, D., and Homocianu, D. (2009). DSS vs. business intelligence. In: *Revista Economica*.

- Tong, G. et al (2009). Application of Ontology-Based Information Integration on BI System. In: *Software Engineering, 2009. WCSE'09. WRI World Congress on. IEEE, 2009. p. 171-175.*
- El Idrissi, B., Baina, S. and Baïna, K. (2013). Automatic generation of ontology from data models: a practical evaluation of existing approaches. In: *Research Challenges in Information Science (RCIS), IEEE Seventh International Conference on (pp. 1-12).*
- Inmon, W. (2002). *Building the Data Warehouse*, 3rd ed. Wiley Computer Publishing, 428p.
- Prat, N., Akoka, J. and Comyn-Wattiau, I. (2012). Transforming multidimensional models into OWL-DL ontologies. In: *Research Challenges in Information Science (RCIS), 2012 Sixth International Conference on (pp. 1-12). IEEE.*
- Prat, N., Megdiche, I. and Akoka, J. (2012). Multidimensional models meet the semantic web: defining and reasoning on OWL-DL ontologies for OLAP. In: *Proceedings of the fifteenth international workshop on Data warehousing and OLAP (pp. 17-24). ACM.*
- Gil, R. and Martín-Bautista, M. J. (2014). SMOL: a systemic methodology for ontology learning from heterogeneous sources. In: *Journal of Intelligent Information Systems, 42(3), 415-455.*
- Gil, R., Martín-Bautista, M. J. and Contreras, L. (2010). Applying an ontology learning methodology to a relational database: University case study. In: *Semantic Computing (ICSC), 2010 IEEE Fourth International Conference on (pp. 313-316). IEEE.*
- Dou, D., Qin, H. and Lependu, P. (2010). OntoGrate: Towards automatic integration for relational databases and the semantic web through an ontology-based framework. In: *International Journal of Semantic Computing, 4(01), 123-151.*
- Moreira, J., Cordeiro, K., Campos, M. L. and Borges, M. (2014). OntoWarehousing—multidimensional design supported by a foundational ontology: a temporal perspective. In: *International Conference on Data Warehousing and Knowledge Discovery (pp. 35-44). Springer International Publishing.*
- Moreira, J., Cordeiro, K., Campos, M. L. and Borges, M. (2015). Hybrid Multidimensional Design for Heterogeneous Data Supported by Ontological Analysis: an Application Case in the Brazilian Electric System Operation. In: *EDBT/ICDT Workshops (pp. 72-77).*