A Tool for Data Warehouse Multidimensional Schema Design using Ontology

M.Thenmozhi[1], K.Vivekanandan[2]

Assistant Professor[1], Professor[2]
Department of Computer Science and Engineering
Pondicherry Engineering College
Puducherry

Abstract

The data warehouse has become a necessary component for an effective analysis of large businesses. It is widely accepted that the data warehouse must be structured according to the multidimensional model to facilitate OLAP analysis. The two driving force for the design of a multidimensional model are data source and business requirements. In recent years in addition to operational databases, web data has become an important source for data warehouse. The web data may be semi-structured or unstructured and exhibit several heterogeneity issues. Hence the design of multidimensional model has become a tedious task. The use of semantic web technologies such as ontology helps to represent such web data into coherent information which can be incorporated in a data Warehouse. Ontology also helps to represent the business requirements in a formal way, which need to be used during the design. In this paper we present an ontology driven tool which helps to automatically derive the conceptual model and logical model for the data warehouse from data source and business requirements. We make use of ontology and its reasoning capabilities at various stages of our tool to facilitate the design task.

Keywords: *Data Warehouse Design, Multidimensional Model, Multidimensional Schema Design, Ontology based Data Warehouse Schema Design*

1. Introduction

Data integration is the process by which several autonomous, distributed and heterogeneous data sources are integrated into a single data source and associated with a global schema [1]. It recently received a great attention due to many data management applications such as data warehouse, peer-to-peer systems, e-commerce, and web services. The primary objective of data warehousing is to bring together information from disparate sources and put it into a format that is conductive to make business decisions. The Web has become one of the richest sources of external data hence the data warehouses must also integrate increasing amounts of web data to assist in decision support [4]. The web data is mostly unstructured or semi-structured and exhibit different types of heterogeneity [3]. The different types of heterogeneity that occur in these data are Structural, Syntactic and Semantic heterogeneity. All these characteristics make it difficult to integrate the data.

Nowadays, the semantic web is increasingly seen as a powerful infrastructure to build reusable and sharable knowledge on the web. It provides XML, RDF and OWL to describe web contents that enable automated information access on the machine processable semantics of information and service. Ontologies are the core of the semantic web for the reuse of formalized knowledge. Ontology is the term referring to the shared understanding of some domains of interest, which is often conceived as a set of classes (concepts), relations, functions, axioms and instances [7]. Ontology definition is adopted from Gruber [7] where an 'ontology is a formal, explicit specification of a shared conceptualization. The ontology is used to solve the problem of syntactic and semantic heterogeneities that exist between different data sources [10]. It is also used to analyze the knowledge related to a specific field of modeling the relevant concepts and facilitates the distinction of the different domain concept. Ontology can bring benefits to data warehousing developments at different phases, as it can enhance the semantics of data sources, integrate heterogeneous schemas, automate ETL process and facilitate OLAP in data analysis [11]. In recent years, researchers have proposed various approaches to bring semantic web and data warehousing to solve several data integration issues.

Our work mainly focuses on data warehouse schema design. In this paper we propose an ontology-driven tool to automate the design of the multidimensional schema for data warehouse. Following are the contribution of our work i) Representation of requirements formally using DW(Data warehouse) requirement ontology ii) Automatically deriving multidimensional elements present in the data source ontology iii) Formally match requirements with the data source to filter results iv) Generation of Logical Schema v) OBDWSD (ontology based data warehouse schema design) tool development.

2. Multidimensional Schema Design

Concerning data warehouse design, there is a general agreement that a conceptual or logical design activity should precede the actual implementation [8]. Typically, the design activity is based on a multidimensional model, whereas the implementation is carried out either within relational or multidimensional databases. The conceptual design allows having closer ideas about the ways that a user can perceive an application domain. In fact, it is considered as a key step that ensures the success of the data warehouse projects since it defines the expressivity of the multidimensional model [6]. The result of this step is a graphical notation which facilitates to the designer and the user for understanding and managing the conceptual model. The multidimensional model consists of a central *fact* which represent the subject of analysis. The *fact* contains numeric attributes which represent the *measure* of a business. The *fact* is related to a set of *dimensions* which represents the different perspective by which the *fact* is analyzed. The attributes of dimension are either in a hierarchy or just descriptive. The hierarchies allow for obtaining views of data with different granularity, i.e. summarized or detailed through roll-up and drill-down operations respectively.

The conceptual design approaches usually follow one of the following: i) Supply-driven, in which the multidimensional model for the data warehouse is derived by thoroughly analyzing the data source. Here, requirements are used at the end of design to filter the results. ii) Demand-driven, in which the design is carried out based on the requirements to generate the multidimensional model. The data source is considered only when populating the data warehouse [20]. Existing approaches claim to fully automate the design task, but in supply-driven they rely on discovering as much multidimensional knowledge as possible from the data sources. As a consequence, supply-driven generate too many results, which misleads the user. The demand-driven approach assumes that the requirements are exhaustive and hence they do not consider the data sources to contain some interesting elements for analysis. In order to overcome these drawbacks, the most promising solution consists of formally considering both the data sources and the requirements in a hybrid approach [13]. In this scenario automation is essential as it removes the dependency on an expert's ability to perform the design and also the need to analyze the data sources. Existing approaches claim to fully automate the design task but lack mechanisms through which to formally match the data sources with information requirements in the early stages of the development, thus making it highly complex to populate the data warehouse in a proper manner. Hence the proposed approach provides a mechanism by which data source and requirements are analyzed in parallel to obtain the multidimensional elements which facilitate the construction of data warehouse schema.

3. Related Work

[18] Proposed a method and tool for designing a data warehouse from a global ontology which integrates a set of OBDBs. They consider user requirements along with sources in order to carry out the design process. They follow goal oriented approach to develop the requirement model. The data warehouse design process is carried out by extracting data warehouse ontology from global ontology and annotating the concepts with multidimensional role by analysis of defined goals of the requirement model. [16] In this paper the authors followed a re-engineering process to design the multidimensional schema from an ontology representing the domain. They derive the multidimensional elements by fully analyzing the sources without considering the requirements by following a set of multidimensional constraints. The obtained knowledge is used for requirement elicitation to derive the conceptual schema. In [17] the authors proposed a semi-automatic approach to derive a conceptual representation of multidimensional schema and ETL process. They represent the business requirements in XML format. The requirements are validated using the ontology representing data sources. The requirements are enriched using information gathered from the sources. For each business requirement an annotated subset ontology is derived from source ontology and validated for multidimensionality. The conceptual multidimensional schema and ETL operations are identified using the merged annotated ontology. [14] In this paper the authors propose a framework which provides a means for building Multidimensional Integrated Ontology (MIO) for Semantic Data Warehouses. By integrating the concepts and properties of several ontologies coming from the same application domain, a MIO establishes the topics, measures, dimensions and hierarchies required by a specific data analysis application.

4. Proposed OBDWSD Tool

The OBDWSD (ontology based data warehouse schema design) tool proposed in this paper is an extension of our previous work [19]. In the our method we use a hybrid methodology to derive the multidimensional model. The requirements and the data source are represented using ontology. Using ontology reasoning capability the multidimensional elements such as fact, measure and dimension are derived from the data source. The concepts from the requirements are matched with the multidimensional elements to filter the results. After

including user suggestion the resultant conceptual model is represented graphically. The tool also facilitate to convert the conceptual to the logical model. Finally quality of the logical multidimensional model is assessed.

The application domain used for our work is EU-Car Rental domain. The case study is a specification developed by the Business Rules Group [5]. This application needs to handle the rapidly evolving business policies governing car rentals. For example, the application must deploy rules that can manage activities: i) Accepting new reservations for new and existing customers ii) Selecting the best promotional offer plan at each customer touch point. The customer may receive the best pricing for the rental agreement made. This is done regardless of promotions that were in effect when the reservation was made. Hence pricing must be evaluated at each customer point i.e., making a reservation, picking up the car, returning the car, etc., The car rental company may need some performance indicators to monitor the pricing. In order to generate a data warehouse schema for the given Car Rental domain, following sections provides the detailed steps of our approach.

4.1 DWRequirement Ontology

A requirement analysis stage for data warehouse aims at obtaining informational requirements of decision makers. These requirements are related to interesting measures of business processes and the context for analyzing these measures. Since the decision makers are concerned about the goals which the data warehouse should satisfy, they ignore how to suitably describe information requirements. Therefore, a requirement analysis phase for data warehouse should start by discovering goals of decision makers. From these goals, the information requirements can be more easily discovered. Finally, the information requirements can be related to the required multidimensional concepts, i.e., the measures of the business process or the context for analyzing these measures.

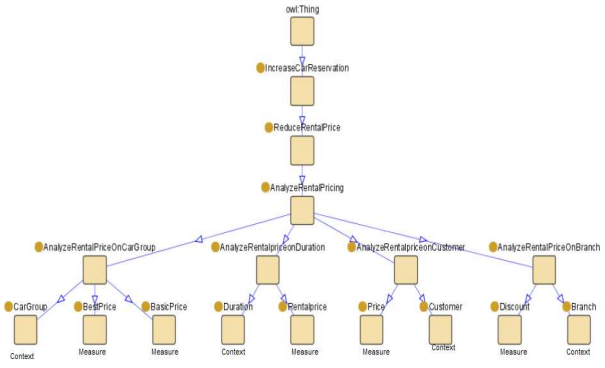

Fig. 1 DWRequirement Ontology

Semantic web technologies such as ontology have found many valuable applications in the field of requirements engineering. In this section, we focus on modeling the data warehouse requirements using ontology. The use of ontologies for the representation of requirements knowledge has several advantages such as communication, traceability, completeness, and consistency. It also supports the detection of redundant or conflicting requirements. Here we assume that a formal requirement analysis has been carried out based on goal oriented requirement analysis for data warehouse. For modeling the information requirements we developed DWRequirement ontology based on i* modeling framework [12]. We developed a GUI where the designer can enter the different goals, context and measures identified in the requirement analysis task. Based on this the DWRequirement ontology is automatically constructed. The developed DWRequirement ontology is shown in Fig 1. for the EU car rental domain. The main objective of the car rental domain is to "Increase car reservation" which is the strategic goal of the organization. To fulfill the above objective the decision goals are "Reduce rental price" and "Open new branches". In order to achieve the goal "Reduce rental price" the information goal "Analyze the rental pricing" is derived. "Analyze rental price on date ","Analyze rental price on car group", "Analyze rental price on Duration " and "Analyze rental price on Branch" are the information requirements for the above information goal. The context and measures for each requirement are annotated as shown in Fig 1. The main purpose of developing the DWRequirement ontology is to formally map the requirements with the data source to construct the conceptual model.

4.2 Data Source Analysis

In data warehouse design along with the requirements the data source needs to be analyzed which provides interesting concepts for constructing the conceptual model. We make use of ontology and its reasoning capability to perform the analysis task. Since ontology can capture the concepts of a domain in a formal and meaningful way, each data source of the data warehouse can be represented using ontology. Each source ontology can then be merged using ontology merging to represent a global ontology. This global ontology captures the knowledge about the domain to be analyzed.

In this paper we assume that such global ontology exists for our EU car rental domain. The Fig. 2 shows the EUCar-Rental ontology [9]. Here we propose an algorithm to automatically identify the interesting concepts available in the global ontology representing the source. The output of the Algorithm 1 is a list of facts, measures and dimensions present in the source. The global ontology is

annotated with the multidimensional elements identified.

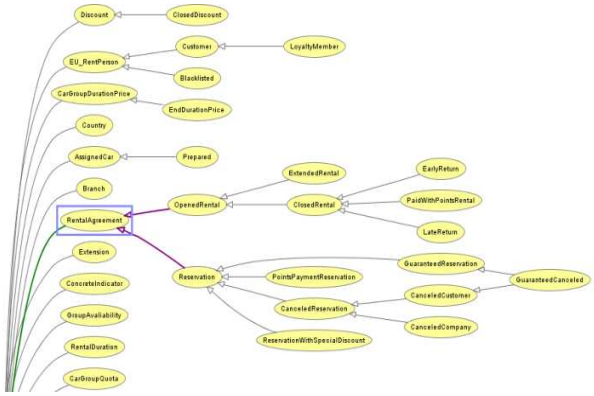

Fig. 2 EUCar-Rental Ontology

Algorithm 1 : Compute Facts and Dimensions
Input : Global ontology O
Output : Multidimensional elements

```
1 Compute_fact(O)
2   for each class in O do
3     for each dataproperty in class do
4       if isnumeric(dataproperty.range)then
5         num_list= num_list+ dataproperty;
6         na++;
7       else
8         nonnum_list= nonnum_list+dataproperty;
9       end if
10    ta++;
11    end for
12    rna = na / ta;
13     if (rna >  trna) then
14      fact=class;
15      fact.measure_list= num_list;
16      fact.level_list= nonnum_list;
17      Print(fact,fact.measure_list,fact.level_list);
18       Compute_dimension(fact);
19    end if
20    Compute_dimension(fact)
21        directconcepts= reasoner. getSubclasses(fact);
22        for each concept in directconcepts do
23          if (fact. objectproperty allValueFrom
            concept&& maxCardinality = = 1) then
24            fact.dimension_list = fact.dimension_list
              +concept;
25          end if
26        end for
27        Print(fact.dimension_list);
28  end for
```

Steps 1–19 of the algorithm 1 computes facts and measures of the given global ontology. For each data property of the concept class we compute, ratio of numerical properties (rna) = na / ta, where na is the number of numerical properties for a concept and ta is the total number of data properties for a concept. Concepts with rna > trna are marked as facts. Where, trna is the threshold value for numerical properties which can be set by the designer. The numerical properties of the fact are identified as measures (num_list) and non numerical properties (nonnum_list) are identified as a level. From step 20-28, the algorithm computes the dimensions for each fact identified in the previous step. Using reasoner we find concepts involved in a subsumption relationship with fact (i.e., fact c subsume of c'). Here the concepts c' with a many-to-one relationship with fact are identified as dimensions (dimension_list). The results obtained after filtering facts and dimensions from global ontology is displayed to the user in our tool as shown in Fig 3. For fact *RentalAgreement* the dimensions identified are *Branch, CarGroup, Country* etc.,

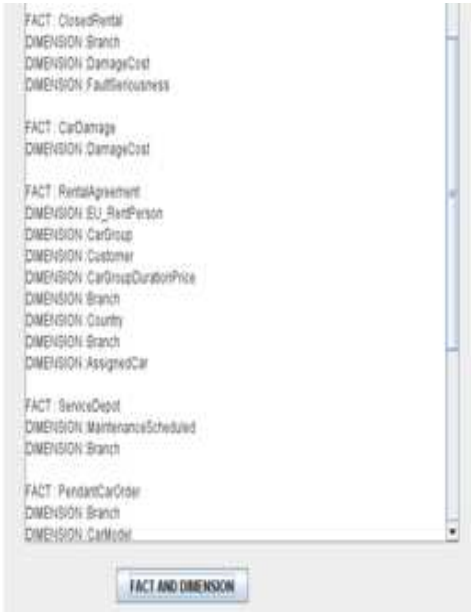

Fig. 3 Facts and Dimensions in Data Source

4.3 Matching Requirements with Data source

To construct the conceptual model we proceed this phase by filtering the results obtained using the Algorithm 1 by matching it with the information requirements. We use ontology matching such as Lexical matching and Semantic matching to perform matching between DWRequirement ontology and global ontology. Here the measures and context represented in the DWRequirement ontology is automatically matched with the measures and dimensions respectively from source. For our car rental domain, the contexts of our requirement are: *Branch, Customer, CarGroup and Duration* which finds a matching with

dimensions of fact *RentalAgreement*. The matched entities are displayed to the user. Any ambiguity in the requirements can be resolved with the knowledge obtained from the source ontology. Thus the conceptual model is generated from the requirements. Fig. 4 represents the screen shot for the matching results performed between global ontology and DWRequirement ontology.

Fig. 4 Matching Data Source with Requirements

4.3 Construction of Logical Schema

This phase transforms the conceptual model into a logical schema expressed with the multidimensional elements identified in the previous phase. The logical multidimensional schema is generated by means of the transformations detailed below. The logical design is carried out by the following steps:

Definition of facts: The class annotated as fact in the global ontology is represented as a fact table in the logical model. In our application domain *RentalAgreement* becomes the fact table.

Definition of measures and dimensions : For the identified fact the annotated properties of the global ontology become the measure attributes to the fact table. *BestPrice* and *BasicPrice* attributes are the measures for the fact *RentalAgreement*. The concepts either class or properties annotated as dimension becomes the dimension table. Here *Branch, Customer, CarGroup* and *RentalDuration* are the dimesnion tables. For each dimension class the property that relates to the fact class becomes the attributes of the fact table.

Definition of dimension hierarchies : In the global ontology, concepts annotated as a dimension that are matched to requirements are considered in this step to shape the dimension hierarchy. The Algorithm 2 automatically computes the levels from the global ontology for each dimension to form the hierarchy.

Algorithm 2: Compute Hierarchy
Input : Dimension List
Output : Dimension Hierarchy

```
1 Compute_hierarchy(fact,dimension_list)
2   for each dimension in dimension_list do
3     directconcepts= reasoner.getSubclasses(dimension);
4       if(directconcepts != null)then
5         for each concept in directconcepts do
6                 if(dimension.objectproperty allValueFrom
                  concept && maxCardinality==1)then
7                   if(concept!=fact &&
                    concept!=dimension)then
8                     level=concept;
9                     dimension.level_list=
                      dimension.level_list+level;
10                  end if
11                end if
12          end for
13          Print(dimension,dimension.level_list);
14          Compute_hierarchy(dimension.level_list);
15      end if
16  end for
```

In step 3, we make use of the reasoner to compute the directly related concepts. The direct concepts with 1:n relationship with dimension are identified as levels (level_list). Each dimension is recursively traversed to identify the dimension levels. The global ontology is annotated with the levels identified. Table 1 represents the concepts identified in the first level for *Branch* and *CarGroup* dimensions. The result of the algorithm is displayed to the user for any suggestion. For our logical schema we selected *Country* to be kept as level for the *Branch* dimension table.

Table 1: Dimension Levels

Dimension Name	Levels
Branch	Branch Type, Country, Service depot
CarGroup	Car, Discount

Definition of attributes in dimension and dimension levels : The properties of the class annotated as a dimension and levels are identified as attributes describing the dimension and dimension hierarchies. For example *Customer-id, Customer-name, Customer-address* etc., becomes the attributes of *Customer* dimension tables.

Definition of different aggregation functions : The aggregation functions are normally {SUM, AVG, MIN, MAX, MED, VAR, STDDEV, COUNT} applied to the measure. For our set of measures the aggregation function

that can be applied are {SUM, AVG, MIN, MAX, COUNT}.

The final multidimensional schema, represented by the graphical notation appears in Fig 5. The generated logical schema is validated for *disjointness* (any two dimension concepts belonging to a fact must be disjoint), *orthogonality* (each instance of fact is related to at-least and at-most one instance of dimension) and *summarizability* (roll-up and drill down facilities). Our output satisfies the above constraints.

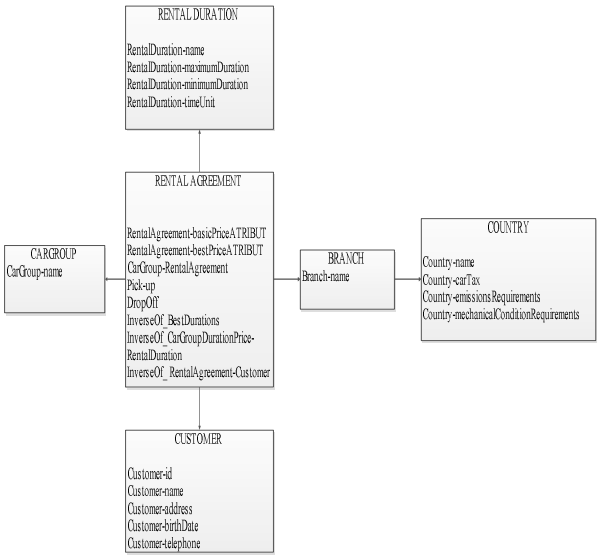

Fig. 5 Generated Logical Schema for Car Rental Domain

5. Tool Implementation

The OBDWSD (ontology based data warehouse schema design) tool is developed using Java (j2sdk 1.4.2) and Jena 2.1, the Java API for ontology development and processing. The global ontology can be constructed by means of an ontology merging. The prompt plug-in for protégé (an ontology editing tool) can be used for the merging task. Pellet reasoner API is used for reasoning tasks in Algorithm 1 and Algorithm 2. We make use of ontology matching algorithm to perform matching between requirements and data source ontology. Our tool has GUI features which help the designer to assist in generating the multidimensional logical schema. The different components that are integrated in our tool is shown in Fig. 6.

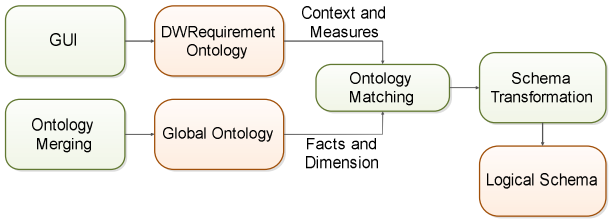

Fig. 6 Components of OBDWSD Tool

6. Comparison of Ontology based DW Design Approaches

Data warehouse schema design in the past was carried out mainly considering relational sources. At present, due to increasing amount of web source and the popularity of semantic web lead to different ontological approaches for data warehouse schema design. These approaches followed supply driven or demand driven and only a few follow the hybrid approach. The Table 2 represents the comparison our proposed approach with the existing ontological approaches for data warehouse schema design. In [18], they follow a hybrid method to derive the conceptual and logical model of the data warehouse. Though they represent the requirements using an ontology, which is the motivation of our proposed approach they do not provide a formal way of analyzing the source. The main advantage of our proposed approach is that it provides a thorough analysis of data source and a formal way of mapping requirements with it to derive the multidimensional model.

7. Conclusions

In this paper we proposed a conceptual design phase aimed at analyzing the data source and matching it with requirements to express the multidimensional concepts they contain. The resulting conceptual model is then transformed into a logical multidimensional schema. Elaborating from our previous work, we have developed an ontology driven tool to carry out the modelling task to help the data warehouse designer. The output schema developed using our tool ensures multidimensionality, disjointness and summarizability. In a data warehouse environment the data source and requirements are not static hence they may evolve. Our future work includes developing an ontology based approach to handle the impact of data sources and requirements evolution on data warehouse schema and ETL process.

Table 2: Comparison of Ontology based Approaches

Features	**[18]**	**[16]**	**[17]**	**[14]**	**Proposed**
Automation	Semi-automatic	Semi-automatic	Semi-automatic	Semi-automatic	Fully
Design Approach	Hybrid	Supply-Driven	Demand-Driven	Demand-Driven	Hybrid
Requirement Representation	Ontology	-	XML Format	Natrual Language	Ontology
Data source Representation	Ontology	Ontology	Ontology	Ontology	Ontology
Data source Analysis	No	Yes	No	No	Yes
Formal Algorithm	Yes	Yes	Yes	No	Yes
Conceptual Design	Yes	Yes	Yes	No	Yes
Logical Design	Yes	No	No	Yes	Yes
Physical Design	No	No	No	Yes	No
User Suggestion	No	Yes	Yes	No	Yes
Tool	Yes	No	Yes	No	Yes

References

[1] Buccella, A., Cechich, A., Brisaboa, N.R., 2003. An ontology approach to data integration. Journal of Computer Science and Technology 3 (2), 62–68.

[2] Cheng Hian Goh. 1997. Representing and Reasoning about Semantic Conflicts in Heterogeneous Information Systems. Ph.D. Dissertation. Massachusetts Institute of Technology. AAI0597943.

[3] Cui, Z., O'Brien, P., 2000. Domain ontology management environment. In: Proceedings of the 33rd Hawaii International Conference on System Sciences. Island of Maui, Hawaii, 4–7 January 2000.

[4] Etzioni, O., Banko, M., Soderland, S., & Weld, D. S. Open information extraction from the web. Communications of the ACM, 51(12), 2008, 68–74.

[5] Frias, L., Queralt, A., Olivé, A., EU-Rent Car Rentals Specification. Technical report, "Dept. de Llenguatges i Sistemes Inform`atics", 2003. www.lsi.upc.edu/dept/techreps/llistat_detallat.php?id=690.

[6] Golfarelli, M., 2009. From User Requirements to Conceptual Design in Data Warehouse Design–a Survey. Data Warehousing Design and Advanced Engineering Applications: Methods for Complex Construction, p.1–16.

[7] Gruber, T. A translation approach to portable ontology specification. Knowledge Acquisition, 5(2):199–220, 1995.

[8] H¨usemann,B.,Lechtenb¨orger,J., and Vossen,G., Conceptual data warehouse design. In Proc. DMDW,pages 3–9, 2000.

[9] http://www.lsi.upc.edu/~%20oromero/ EUCarRental.owl

[10] Isabel F. Cruz and Huiyong Xiao. The Role of Ontologies in Data Integration. Journal of EngineeringIntelligent Systems, 13(4):245–252, December 2005.
[11] Jesús Pardillo, Jose-Norberto Mazón. Using Ontologies for the Design of Data Warehouses. Journal of Database Management, 3(2), 2011.
[12] Maz´on, J.N., Trujillo, J., and Trujillo,J., A Model Driven RE Approach for Data Warehouses. In RIGiM '07: First International Workshop on Requirements, Intentions and Goals in Conceptual Modeling, 2007.
[13] Maz´on, J.N., Trujillo, J.: A hybrid model driven development framework for the multidimensional modeling of data warehouses. SIGMOD Record 38(2) (2009)
[14] Nebot,V.,Berlanga,R., Perez,J.M., Aramburu,J.M. and Pedersen,T.B., Multidimensional Integrated Ontologies: A Framework for Designing Semantic Data Warehouses. JoDS XIII, 5530:1{35, 2009
[15] Oscar Romero and Alberto Abelló. 2011. A comprehensive framework on multidimensional modeling. In Proceedings of the 30th international conference on Advances in conceptual modeling: recent developments and new directions (ER'11). Springer-Verlag, Berlin, Heidelberg, 108-117.
[16] Oscar Romero, Alberto Abelló: A framework for multidimensional design of data warehouses from ontologies. Data Knowl. Eng. 69(11): 1138-1157 (2010).
[17] Oscar Romero, Alkis Simitsis, and Alberto Abelló, GEM: requirement-driven generation of ETL and multidimensional conceptual designs. In Proceedings of the 13th international conference on Data warehousing and knowledge discovery, Springer-Verlag, Berlin, Heidelberg, (2011) 80-95.
[18] Selma Khouri, Ilyès Boukhari, Ladjel Bellatreche, Eric Sardet, Stéphane Jean, Michael Baron: Ontology-based structured web data warehouses for sustainable interoperability: requirement modeling, design methodology and tool. Computers in Industry 63(8): 799-812 (2012)
[19] Thenmozhi.M and Vivekanandan.K. An Ontology based Hybrid Approach to Derive Multidimensional Schema for Data Warehouse. International Journal of Computer Applications 54(8):36-42, September 2012.
[20] Winter, R., Strauch, B.: A Method for Demand-Driven Information Requirements Analysis in DW Projects. In: Proc. of 36th Annual Hawaii Int. Conf. on System Sciences, pp. 231–239. IEEE, Los Alamitos (2003)
[21] Zhan Cui and Paul O'Brien. 2000. Domain Ontology Management Environment. In Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 8 - Volume 8 (HICSS '00), Vol. 8. IEEE Computer Society, Washington, DC, USA, 8015-.