CMCD: A Data Warehouse Modeling Framework Based on Goals and Business Process Models

Houda Chakiri
Sidi Mohammed Ben Abdellah University, Fez, Morocco,
h.chakiri@aui.ma

Mohammed El Mohajir
Sidi Mohammed Ben Abdellah University, Fez, Morocco,
m.elmohajir@ieee.ma

Nasser Assem
Al Akhawayn University in Ifrane, Morocco
n.assem@aui.ma

***Abstract*— The decision making process is mainly data driven, which makes Data Warehousing play a crucial role in providing a reliable and efficient tool to meet analytical needs and provide end users and decision makers with useful insights. However, several surveys show that a significant percentage of Data Warehouse (DW) projects fail to meet their business goals. Indeed, it is mainly due to the lack of any formal relationship between the Contextual Requirements analysis and the DW conceptual modeling phase. This work is a first step towards the development of a CMCD framework (from Contextual Model to Conceptual Design) that consists of a unified approach to automatically extract the DW multidimensional model from Goal modeling frameworks and Business Process Diagrams. In order to illustrate the feasibility of our approach, we implemented it within a case study related to the development and implementation of a DW to monitor and assess good local governance in Moroccan local municipality offices.**

Keywords— Data warehouse; Goals models; Business process; Requirements analysis; Multidimensional modeling.

I. Introduction

Data Warehouses integrate numerous heterogeneous data sources in multidimensional structures in support of the decision-making process. In order to be successful, recent DW development approaches, such as in [1], consist of using the i* modeling framework coupled with a requirements analysis method based on goal modeling. Accordingly, the requirements of individual decision makers, who are the users of the DW system, are gathered by means of questionnaires and interviews.

The main drawback of the techniques used during the requirements analysis phase is that they obtain only a partial view of the problem from each decision maker point of view. Moreover, these partial views may not always be aligned with the strategic business plan, which, together with the gap between IT specialists and business analysts, make it difficult to validate the goal models.

This problem arises whenever the business goals are not considered during the requirements analysis phase. Our approach adopts contextual modeling for requirements analysis, such as the Goals Diagram, the Rational Diagram and the Business Process Diagrams, from which we intend to automate the design of the future DW schema.

This paper is organized as follows: After an introduction, section 2 is dedicated to the description of related work, followed by section 3 describing our proposed framework called CMCD (from Contextual Model to Conceptual Design). Then, section 4 presents an application of the proposed framework within the e-Government field in Morocco. The final section presents conclusions and ongoing/future work.

II. Related Work

Most of the attention in Data Warehouse requirements analysis has been focused on requirements elicitation. A survey was conducted in [11] in order to compare different approaches to semi automatize the design of DWs.

The proposed technique in [2] focused on a demand driven approach in gathering user requirements by means of interviews and questionnaires in order to design the DW. In [1], a model driven goal oriented requirements engineering approach was used to design the DW based on decision makers' goals specifications, which makes it difficult to design the optimal DW that answers all aspects required by end users. The proposed methods in [3] were based on including functional and non-functional requirements for DW conceptual design, whereas in [4] they were based on a goal oriented approach for DW design; however they did not succeed to automatically extract facts and dimensions from the obtained diagrams.

Also in [5], a goal-oriented approach, based on the goal-decision-information model, is proposed. Even though this approach shares some similarities with ours, it mainly focuses on requirements analysis and does not describe how to move from requirements to the DW design. The proposed approach in [6] can be considered similar to our approach as it uses the Goal Question Matrix (GQM) for making explicit user requirements in the first stage of DW design activity; however the main difference is that organizational modeling is not supported and requirements analysis starts from the goals of decision makers.

978-1-5386-2775-4/17/$31.00 ©2017 IEEE

Our purpose is therefore to develop a new framework capable of offering a unified approach that allows for automatically extracting facts and dimensions to generate the design of the future DW from the early goals and business contextual models.

III. CMCD Framework: from Contextual Modeling to Conceptual Design

Our approach adopts the i* contextual modeling framework for requirements analysis [7] (Goals Diagram, Rational Diagram and Business Process Diagrams) in order to serve as a basis to automate the design of the future data warehouse.

In fact, Goals Diagram analysis aims to identify non-functional requirements while rational diagrams that relate contextual goals to stockholders are used to capture facts and dimensions. Similarly, the Business Process Diagrams are used to identify the time granularity for the historical dimension and the measures for the fact tables.

In this work, we used Prolog language to extract facts and dimensions from the Rational Diagram and Business Process Diagrams. We also used a UML 2.0/OCL profile to enrich the conceptual schema using Goal Diagram as Fig. 1 shows.

Our approach is organized around four phases:

1. Contextual modeling.
2. Extraction of facts and dimensions.
3. Extraction of measures and time granularity.
4. Extraction of non-functional requirements.

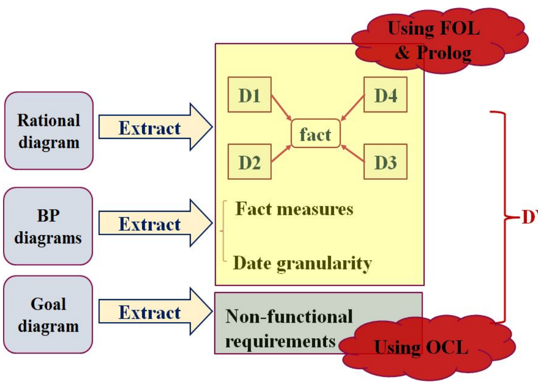

Fig. 1. CMCD Framework design

A. Contextual Modeling

Contextual modeling is a form of modeling that draws the structure of an organization and the relationships between the different stakeholders, departments or occupations within that organization. In our case, we use three types of contextual diagrams:

1. Goals Diagram: it represents goals, their dependencies and their properties.
2. Rational Diagram: it is used to represent the logical foundations that rule the relationships between actors. It appears as a balloon within which goals of a specific actor are analyzed and dependencies with other actors are established.
3. Business Process Diagram: A business process is a collection of activities, which are executed to create products.

B. Extraction of Facts and Dimensions

In order to extract dimensions and facts, we will use the rational diagram and some new concepts that will be defined using the first order logic [9].

In order to facilitate the understanding of the domain of study, we give Fig. 2 that shows an example of a Rational Diagram:

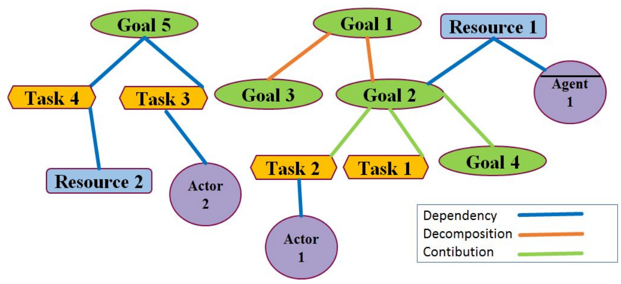

Fig. 2. Rational Diagram Example

D = {Goal1, Goal2, Goal3, Goal4, Goal5, Task1, Task2, Task3, Task4, Actor1, Actor2, Agent1, Resource1, Resource2}

In our case, we assume that a fact is a resource that can be measured. The user of our framework must identify all existing facts in his/her Rational Diagram. Then, a link, which is a relationship between two objects, can be considered a decomposition, a dependency, a contribution or an association as in Rule 1.

$$\forall X,Y \in D: \text{Link}(X,Y) \rightarrow \text{Decomposition}(X,Y) \lor \text{Dependency}(X,Y) \lor \text{Contribution}(X,Y) \lor \text{Association}(Y,X) \quad (1)$$

We do not differentiate between the types of links because we are mostly interested in identifying the objects related with each other. So, considering resource 1 as a fact, the Rational Diagram will be as shown in Fig. 3.

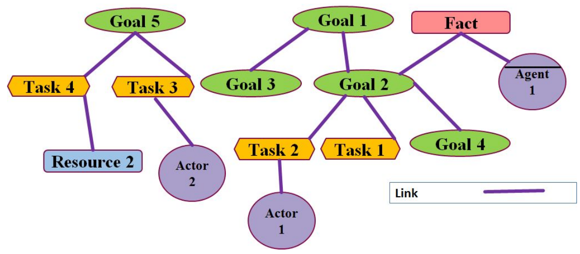

Fig. 3. Fact Identification

We define a sub object, a root object and a neighbor as shown in Fig. 4, and Rules 2 – 4.

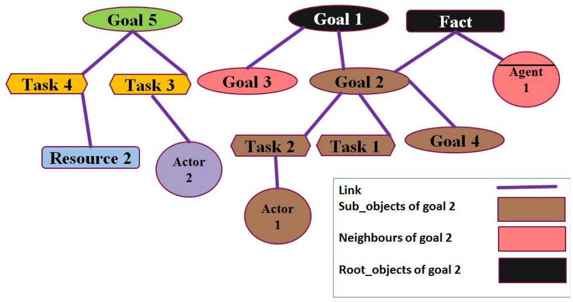

Fig. 4. Sub object, Root object and Neighbor for a Goal

$\forall X,Y \in D$: Sub_object$(X,Y) \rightarrow$ Link(X,Y) $\vee$

Sub_object$(X,Y) \rightarrow \exists Z \in D$: Link$(Z,Y)$ $\wedge$ Sub_goal_task(X,Z) (2)

$\forall X,Y \in$ D: Root_object$(X,Y) \rightarrow$ Link(Y,X) $\vee$

Root_object$(X,Y) \rightarrow \exists Z \in D$: Link$(Y,Z)$ $\wedge$ Root_object (X,Z) (3)

$\forall X,Y \in D$: Neighbor$(X,Y) \rightarrow \exists Z \in D$: Root_object$(X,Z)$ $\wedge$ Root_object(Y,Z) (4)

The dimensions are the agents, the actors and the roles that relate to a fact directly or indirectly as show in Rule 5.

$\forall X,Y \in$ D: Dimension(X,Y) $\rightarrow$ Fact(X) $\wedge$ (Root_object(X,Y) $\vee$ Neighbor(X,Y)

$\vee$ Sub_object (X,Y)) $\wedge$ (Agent(Y) $\vee$ Actor(Y)) (5)

So, to identify the dimensions, we go through two stages:

1. Identify all the objects related to the fact as shown in Fig. 5.
2. Among the identified objects, keep just the agents, the actors and the roles as shown in Fig. 6.

Indeed, in this example, the dimensions are Agent1, Actor1 to which we add the date/time dimension.

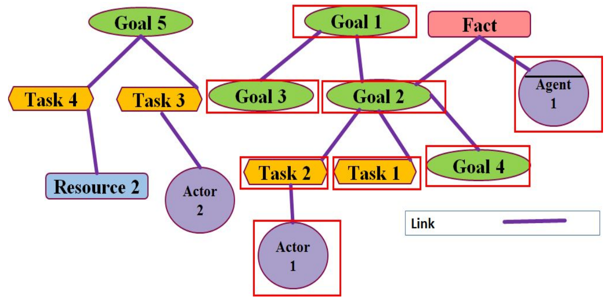

Fig. 5. Identification of Objects Related to a Fact

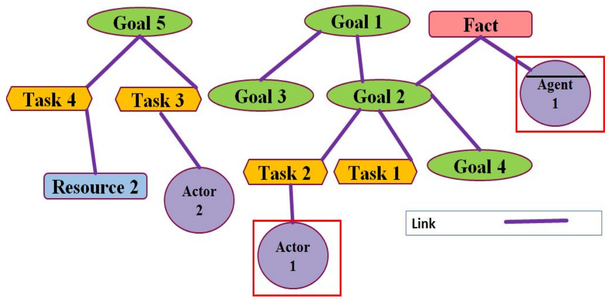

Fig. 6. Dimensions' Identification

C. *Extraction of Measures and Time Granularity*

In this phase, we use the business process diagram related to a fact and we give for each activity a time attribute which characterizes its execution time. We also give to the activities a measure if possible. For example, for the activity "receive payment", we associate the payment amount as a measure.

In order to extract time granularity, we compute the execution time of the process t, and define the time granularity as follows:

If 1 sec $< t \leq$ 1 min, then the granularity is the minute;
else if 1 min $< t \leq$ 1 hour, then the granularity is the hour;
else if 1 hour $< t \leq$ 1 day, then the granularity is the day; etc…

D. *Extraction of Non-functional Requirements*

In this phase, we use the goals diagram for extracting non-functional requirements. In our future work, we will focus on representing the non-functional requirements using an extension of the approach proposed in [8] based on UML 2.0/OCL profile.

IV. Framework Application

In order to validate the viability of our approach, we selected the Moroccan Local Registry Office (BEC) for our business analytics case study [10]. It aims at setting up a Data

Warehouse to help decision makers in making informed strategic decisions, and to measure empirically the efficiency and effectiveness of BECs in different communes' daily work and services for citizens and consequently monitor and assess the goodness of local governance.

The next sections describe the multidimensional modeling phases of the target DW using our proposed framework.

A. *Contextual Modeling*

We use only the Rational Diagram as shown in Fig. 8 and Fig. 9, and the Business Process Diagrams as shown in Fig. 12. The Goal Diagram is used in our ongoing work as shown in Fig. 7, with an extension of the approach proposed in [8].

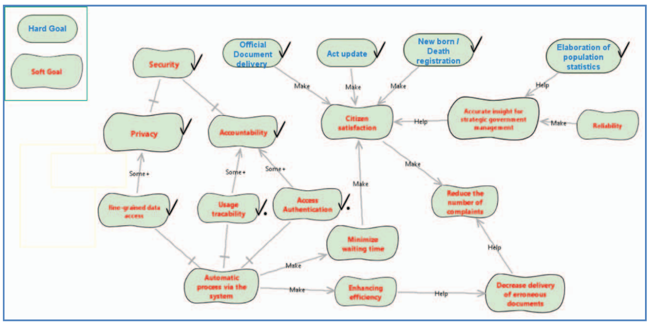

Fig. 7. Case study – Goal Diagram

The principal goals in a general municipality office (BEC) are: (i) generating population statistics, (ii) delivering official documents, (iii) updating citizens' record, and (iv) recording (registering) new life events (birth, death, marriage, divorce, name change, etc…). All these goals contribute to satisfying the citizens' requests and meeting their expectations.

As secondary goals, we can enumerate the following:

- Security, composed of privacy (as citizens' records contain sensitive data which should be kept private) and accountability (creating standards against which individuals providing service and service delivery can be held accountable.)
- Automatic process via the system, composed of fine-grained data access, contributing to increased privacy, usage traceability and access authentication, thus contributing to accountability.
- Automatic process via the system, with a positive impact on minimizing waiting time and enhancing efficiency, by decreasing the number of erroneous documents delivered.
- Minimizing waiting time, meeting citizens' expectations, reducing the number of complaints.
- Creating visibility in the service workflow for citizens through an automated service delivery.
- Issuing population statistics, with accurate insight for strategic government management.

Fig. 8 represents the processing tasks in the Moroccan local municipality office and their relationships with different external stakeholders.

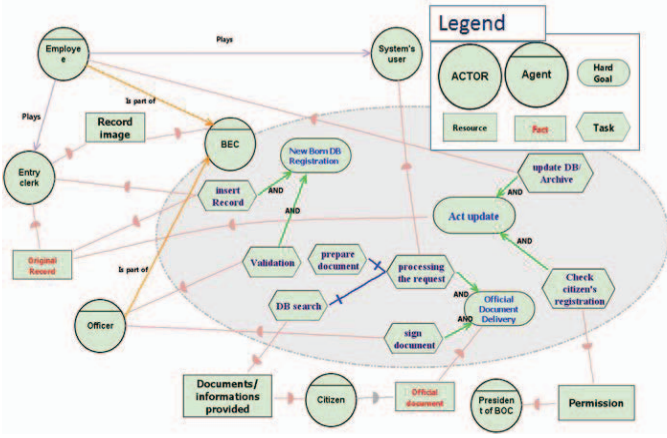

Fig. 8. Case study – Rational Diagram

B. *Extraction of Facts and Dimensions*

First, we use the rational diagram described in Fig.8. In our case, we have two facts: Official Document and Original Citizen's Record. So, we try to extract dimensions from the Official Document fact. Then, we keep only the candidate dimensions delimited by a rectangle as shown in Fig. 9.

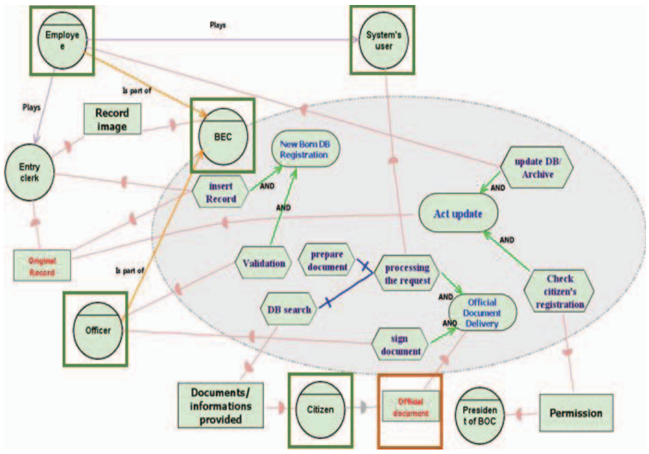

Fig. 9. Case study – Dimensions' Identification

Indeed, according to the CMCD framework, the dimensions related to the Official Document fact are: Officer, Citizen, Employee, Document, System's User, BEC and Date.

We automated this process using Prolog language and SWI-Prolog tool. We translated the diagram to a Prolog file as shown in Fig.10, and SWI-Prolog tool extracts automatically the dimensions related to the facts as shown in Fig.11.

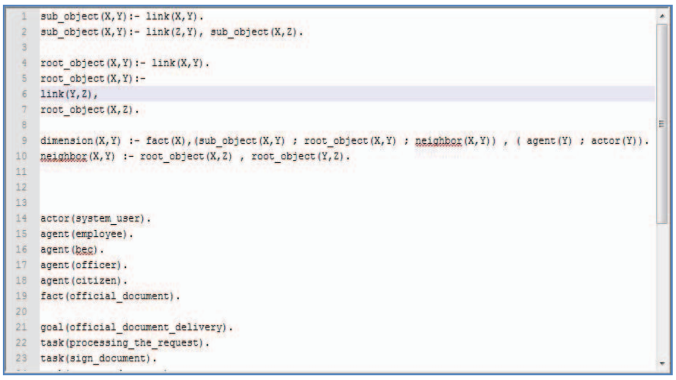

Fig. 10. Rational Diagram Translated to Prolog

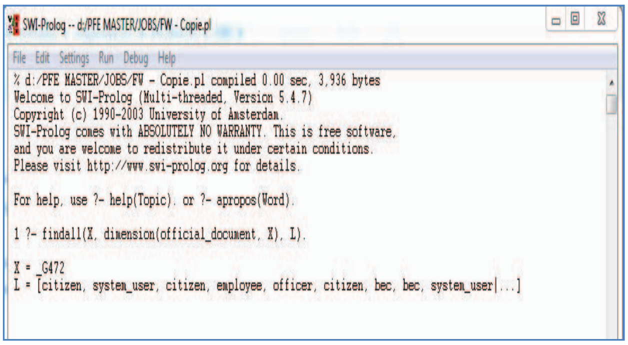

Fig. 11. Dimensions' Extraction with Prolog

C. *Extraction of Measures and Time Granularity*

Presently, we use the Business Process Diagram, as shown in Fig. 12, to extract the measures and the time granularity. In our case, the measures are the number of document copies issued and the payload. The execution time is equal to 5 min 45 sec (1 min < $t \leq$ 1 hour). So, the granularity is (hour, day, month, year).

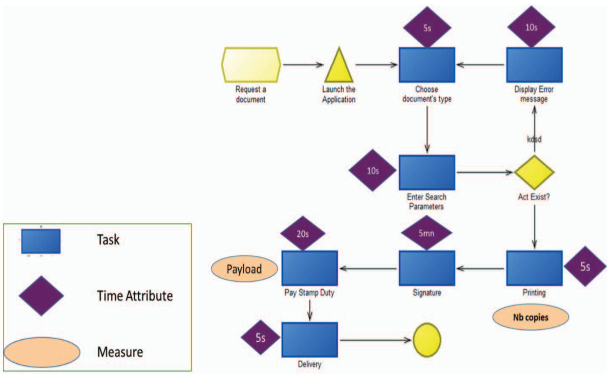

Fig. 12. Business Process Diagram

Fig. 13 shows the extracted Data Mart related to the first fact (Official Document).

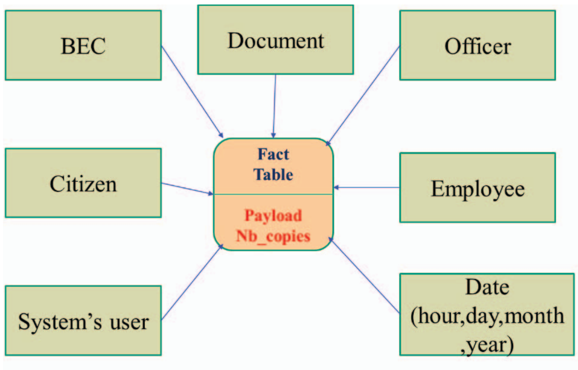

Fig. 13. Extracted Data Mart 1

We followed the same steps, and obtained the schema corresponding to the other fact as shown in Fig. 14.

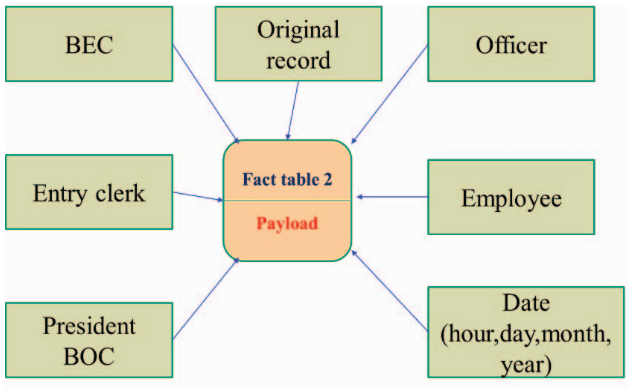

Fig. 14. Extracted Data Mart 2

We used the fact constellation model to represent the obtained schema as shown in Fig. 15.

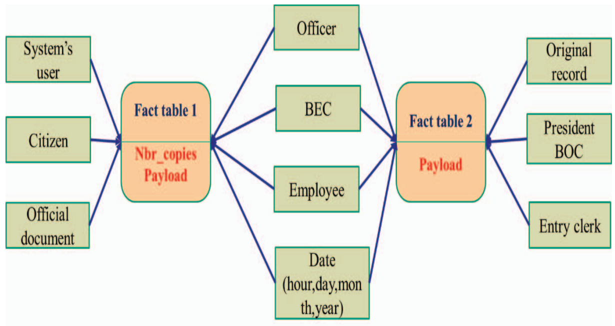

Fig. 15. Produced Data Warehouse Schema

In order to align the obtained schema with the data sources and increase the performance, we refined it as shown in

Fig.16. The related resulting reports and results are described in [10].

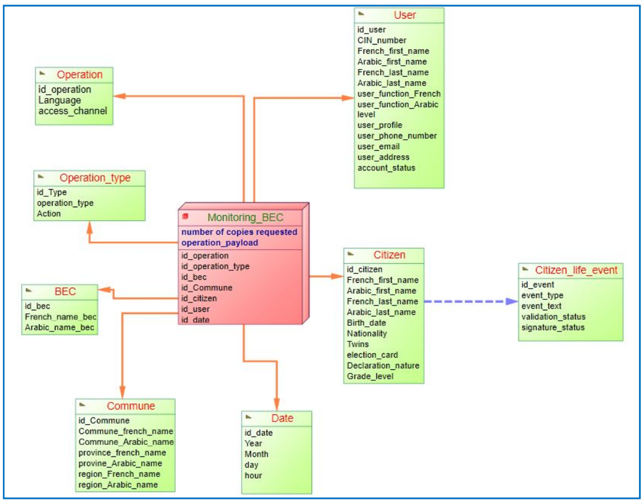

Fig. 16. Refined Schema for the Case study Data Warehouse.

V. Conclusion and Ongoing/Future Work

In this paper, we proposed a preliminary foundation for a CMCD framework (from Contextual Model to Conceptual Design) that can be used to automate the conceptual modeling of a Data Warehouse based on both Goal Models and Business Process Diagrams. The adoption of such a framework can be very helpful to the designer to reduce the risk of inaccuracies between contextual business requirements analysis and DW modeling design, leading to a more successful decision support project.

The CMCD framework was applied successfully to the Moroccan local municipality office case study. The project outcomes were instrumental to validating our work, and get insights for future developments.

As ongoing work, we are focusing on representing the non-functional requirements using an extension of the approach proposed in [8] based on UML 2.0/OCL profile. We will also transform the CMCD framework to an ontology that includes the contextual and the automated extraction of the conceptual design of the DW.

References

[1] J.N. Mazón, J. Pardillo and J. Trujillo, "A model-driven goal-oriented requirement engineering approach for data warehouses," in *Advances in Conceptual Modeling – Foundations and Applications*, 2007, pp. 255-264.

[2] R. Winter and B. Strauch, "A method for demand-driven information requirements analysis in data warehousing projects," in *Proc. 36th Hawaii Int. Conf. System Sciences*, 2003, pp. 1359-1365.

[3] M.El Mohajir and I.Jellouli, "Towards a framework incorporating functional and non-functional requirements for data warehouse conceptual design," in *Int. J. Computer Science and Information Systems*, vol. 9, no. 1, pp. 43-54, 2014.

[4] P. Giorgini, S. Rizzi and M. Garzetti, "GRAnD: A goal-oriented approach to requirement analysis in data warehouses," in *Decision Support Systems*, vol. 45, no. 1, pp. 4-21, Apr. 2008.

[5] N. Prakash and A. Gosain, "Requirements driven data warehouse development," in *CAiSE Short Paper Proc.*, 2003.

[6] A. Bonifati, F. Cattaneo, S. Ceri, A. Fuggetta and S. Paraboschi, "Designing data marts for data warehouses," in *ACM Tran. on Software Engineering and Methodology*, vol. 10, no. 4, pp. 452-483, Oct. 2001.

[7] P. Bresciani, P. Giorgini, F. Giunchiglia, J. Mylopoulos and A. Perini, "Tropos: an agent-oriented software development methodology," in *J. Autonomous Agents and Multi-Agent Systems*, vol. 8, no. 3, pp. 203-236, May 2004.

[8] R. Villarroel, E. Fernández-Medina, M. Piattini and J. Trujillo, "A UML 2.0/OCL extension for designing secure data warehouses," in *J. Research and Practice in Information Technology*, vol. 38, no. 1, pp. 31-43, Feb. 2006.

[9] F. Baader, "Logic-based knowledge representation," in *Artificial Intelligence Today,* Berlin, Heidelberg: Springer-Verlag, 1999, pp. 13-41.

[10] H. Chakiri and M. El Mohajir, "A data warehouse for local good governance monitoring and assessment - Case study of local registry office in Morocco," in *Proc. 4th IEEE Int. Colloq. Information Science and Technology*, 2016, pp. 52-58.

[11] W. Tebourski, W. Ben Abdessalem Karâa and H Ben Ghezala, "Semi-automatic data warehouse design methodologies: a survey," in *IJCSI Int.J. Computer Science Issues*, vol. 10, issue. 5, no. 2, pp. 48-54, Sep. 2013.