MDA Approach for Designing and Developing Data Warehouses: A Systematic Review & Proposal

https://doi.org/10.3991/ijoe.v17i10.24667

Mohamed Hanine(✉), Mohamed Lachgar, Sara Elmahfoudi, Omar Boutkhoum
University of Chouaib Doukkali, El Jadida, Morocco
hanine.m@ucd.ac.ma

Abstract—A data warehouse (DW) is a vast repository of data that facilitates decision-making for businesses and companies. This concept dates back to the 1980s and it has been widely accepted. One of the key points for the success of the process of data warehousing lies in the definition of the warehouse model depending on data sources and analysis needs. Once the data warehouse is designed, the content and structure of the data sources, as well as the requirements analysis are required to evolve, therefore, an evolution of the model must take place (diagram and data). In this context, several approaches have been developed to design and implement data warehouses. Nevertheless, there is no standard process that deals with designing all of the data warehouse layers, also, there is no software that encompasses this type of problem. In general, the majority of these approaches focus on a particular aspect of data warehouse such as data storage, ETL process, OLAP, reporting, etc, and does not cover its entire lifecycle. A Model-Driven Architecture (MDA) is a standard approach, its aims to support all phases of software manufacturing by promoting the use of models and the transformations between them. Moreover, this approach aims to automate the process of software engineering, thereby decreasing the cost of software development and enhancing its productivity. In this study, we present a systematic review of various works on the data warehouse design methods. We compare and discuss these works according to the criteria that seem relevant for this issue. We present a new design approach for multidimensional schemas construction from relational models using MDA techniques, we also develop the resulting research perspectives.

Keywords—Data Warehouse, ETL Process, OLAP, Model Driven Architecture, Software Engineering, Business Intelligence

1 Introduction

Data warehouses (DW) are characterized by a complicated architecture, they are built from transactional sources via ETL (Extract - Transform - Load) processes. DW is often represented in a multidimensional format to aid decision-making. The old proposed DW design methods and most of the available storage technologies assume that the conceptual model of DW is set in stone. However, in practice, DWs are characterized by a dynamic that affects not only the stored data but also their structures. It is therefore difficult to definitively determine the model of a DW from the design phase, it is often obligatory to modify it after its installation. The continuous evolution of source diagrams is present because there is always an evolution over time of the business processes of the operating system within the company, and DW depends on this source and it cannot be definitive because it is inevitably affected by its diagram. Basically, in the DW design process, the steps currently used are tedious and require significant human expertise, in this context, several solutions have been proposed within the framework of a Model-Driven Architecture (MDA). MDA is an approach that aims to separate the implementation of the functionality of a system from the specification of the functionality on a platform. It is model-based, it allows a higher level of abstraction during development, and also allows the separation between platform independent models (PIM) and platform specific models (PSM). We wish to provide solutions to this problem. To do this, we will review the existing literature dealing with the problem of data warehousing, so we can respond to the various limits that block the standardization of the MDA approach.

The structure of this present study is as follows. In section 2, we will talk about some concepts related to data warehousing and MDA. In section 3, we will present our systematic reviews, in section 4 by summarizing the different authors' proposals about the MDA Approach. The analysis of the systematic review and the proposed approach are presented in sections 5 and 6 respectively. Finally, the last section presents the conclusions along with future work.

2 Background

2.1 Decision support system

A decision support system is all the IT tools (hardware and software) that allow the analysis of operational data from the system business information. This data is transformed into a decision-maker-oriented vision then analyzed via adapted manipulations and restitutions.

As shown in Fig 1, there are four categories of tools used in decision support systems:

- **The supply phase**: collect, clean, and consolidate data using ETL (Extract Transform Load) tools: This phase involves ETL processes which will be responsible for retrieving data from different storage sources, formatting, cleaning, and consolidating them. Although the standardization of exchanges between various IT tools is improving, business information systems are still heterogeneous and data formats are still disparate. This disparity is the main technological obstacle to the wide exchange of information.
- **The modeling and storage phase**: the data warehouse: This phase allows the data to be stored in a suitable form. Because decision-making requests consume a lot of machine resources, the data must be stored in a specialized database like a data warehouse. This later is responsible for storing and centralizing the data to build the decision-making information system.

- **The restitution or distribution phase**: This step involves data retrieval tools in order to distribute and facilitate the accessibility of information according to functions and types of use. The EIP (Enterprise Information Portal) decision-making portal performs the function of distributing information to all of the company's internal partners.
- **The analysis and exploitation phase**: End-users intervene in this last step to exploit and analyze the data provided to them. Depending on the needs, different types of extraction and exploitation tools are available such as OLAP tools for multidimensional analyzes, data mining tools (extraction) to search for barely visible correlations, dashboards that present the key indicators to drive performance, and reporting tools to communicate on performance.

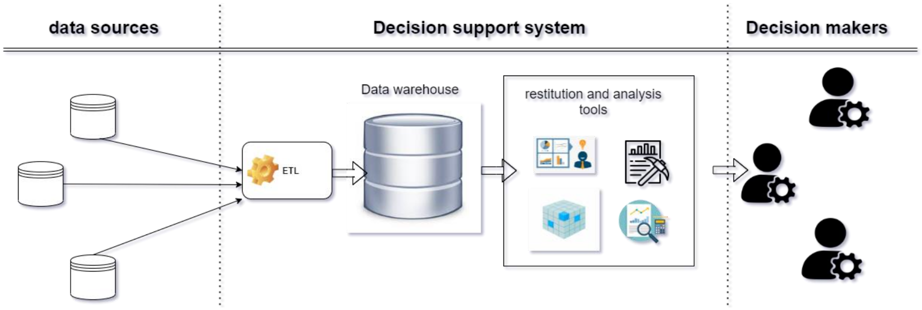

Fig. 1. Decision support system (Adapted from [1])

2.2 Data Warehouse

A data warehouse is a multidimensional database designed for data queries and analyzes, decision making, and Business Intelligence (BI) activities rather than for transaction processing or other traditional database uses. The data stored in the data warehouse is historical and provides an overview of the different transactions that have taken place over time. Redundant data is often included in data warehouses to provide users with multiple views on the information. This is the reason why the data stored in the warehouse is often aggregated to make it easier for users to access it. In addition to a relational database, a data warehouse environment integrates a tool for extracting, transporting, transforming, and loading data (ETL). There is also an online analytical processing engine (OLAP), customer analysis tools, and other applications to manage the processing of the collected data. One of the main features of a data warehouse is that the information is classified by subject (customers, products, etc.). In fact, what really defines a data warehouse is the type of data it contains and the people who use it.

According to [2], the inventor of the term, data warehouses have four specific characteristics. They must be subject-oriented, integrated, non-volatile, and “time-variant”.

- **Data Warehouses must be subject-oriented**: which means that it must be possible to define them by their subject. For example, a warehouse can be deployed specifically to analyze data related to company sales. This Data Warehouse will be used to answer questions such as "who were the best customers for which product in the past year".

- **Data Warehouse must be integrated**: which means that it must be able to assemble data from different sources in a consistent format, it should resolve issues such as name conflicts and inconsistencies in terms of units of measure.
- **Data Warehouses must be non-volatile**: This means that once a piece of data has entered the Warehouse, it should not change. This allows the user to analyze the data as it was stored in the Warehouse.
- **Data Warehouse must be time-variant**: This means that it allows analyzes to focus on changes over time from large datasets, in order to uncover trends. This is what sets Data Warehouses against OLTP systems whose operational data is atomic and only reflects the current value of the last transaction.

To design a data warehouse, there are three types of modeling:

- **Star modeling**: star modeling is the simplest model and the one most commonly used in the design of Data Warehouses, in this model, the fact table is at the center of the diagram and is surrounded by dimension tables. It visually looks like a star. This modeling has a business orientation, each fact table corresponds to an object of study: sales, purchases, logistics, production, etc. The fact table contains all the facts and measurements associated with the object of study - most of the data it contains are figures: amounts, quantities, rates, etc.
- **Snowflake modeling**: Snowflake schema is a type of star schema that includes the hierarchical shape of dimensional tables. In this diagram, there is a fact table made up of different dimension and sub-dimension tables linked by primary and foreign keys to the fact table. It is called the snowflake because its structure resembles a snowflake. It uses normalization which divides data into additional tables. Splitting helps reduce redundancy and prevents memory loss. A snowflake diagram is easier to manage but complex to design and understand. It can also reduce the efficiency of navigation because more joins will be required to execute a query.
- **Constellation modeling**: The constellation model is therefore made up of several fact tables with their respective dimension tables. The dimension tables' common to the different fact tables are not subject to redundancies: this is one of the main advantages of this modeling. This helps reduce the storage space required. Ideally, the shared dimension tables should be identical and contain the same values, the same attributes. Otherwise, adjustments are necessary to make the shared dimension tables suitable for both business needs.

2.3 MDA: Model Driven Architecture

MDA It is an approach that aims to separate the implementation of the functionality of a system from the specification of the functionality on a platform. It is model-based, it allows a higher level of abstraction during development, and also allows the separation between platform independent models (PIM) and platform specific models (PSM). It is an approach that has two main aspects in the development process, the first aspect is the business aspect which represents the functions of the application, and the second is the technical aspect which encompasses the technologies to implement the application. Each aspect has sets of models, which contain the information needed to build the application. MDA defines levels of abstraction models (Fig 2): [3]

- **CIM (Computational Independent Model)**: it is the basic analysis model of the business or the field of application, it is independent of any computer system and describes the concepts of the business activity, the know-how of the processes, the terminology, and the management rules (high level), it also describes the situation in which the system is used and it is only modified if knowledge or business needs change (very long lifespan). The requirements modeled in the CIM will be taken into account in the construction of the PIM (Platform Independent Model) and PSM (Platform Specific Model).
- **PIM (Platform Independent Model)**: it is a design model which describes the system independently of any technical platform and any technology used to deploy the application, this model represents the business logic specific to the system (functioning of entities and services) and it is sustainable over time, it consists of UML class diagrams (with OCL constraints). The different levels of PIM specify the choices of persistence, transaction management, security, etc.
- **PDM (Platform Description Model)**: it contains information for the transformation of models to a platform, it is specific to a platform and it is a transformation model to enable the transition from PIM to PSM.
- **PSM (Platform Specific Model)**: it is used to generate executable code for specific technical platforms and describes how the system will use the platform. It is platform dependent. This model has many levels: The first level, resulting from the transformation of a PIM by adapting UML models to the specificities of the platform, the other PSM levels are obtained by successive transformations taking into account the language (Java, C #, PHP ...), the design choices ... The last level, or implementation PSM, describes, among other things, the program code, the schemas of the tables, the libraries used, the deployment descriptors, etc.[3]

3 Systematic Review

A systematic review is the result of a rigorous scientific approach made up of several well-defined steps, including a systematic literature search, an assessment of the quality of each study considered and a summary, quantified or narrative, of the results obtained. The result of this work makes it possible to conclude, for example, on the effectiveness of a treatment, the risk of side effects, or the performance of a diagnostic test. Sometimes authors can only see the lack of rigorous scientific data.

A systematic review attempts to assemble all the empirical evidence that meets predefined eligibility criteria in order to answer a specific research question. It uses explicit and systematic methods that are chosen to minimize and to get reliable results

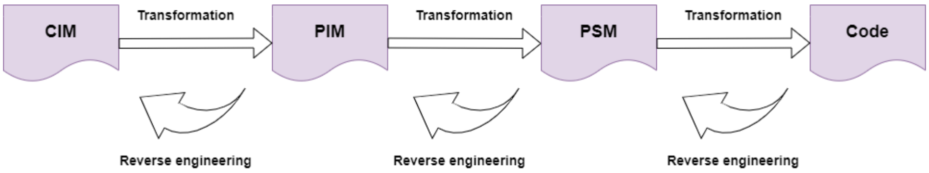

Fig. 2. Model Driven Architecture levels [4]

in order to draw conclusions and make decisions. The main characteristics of a systematic review are:

- a set of clearly defined goals with predefined eligibility criteria for studies;
- an explicit methodology that can be reproduced;
- a systematic search that attempts to verify all studies that meet the eligibility criteria;
- an assessment of the validity of the conclusions of the included studies;
- a systematic presentation and synthesis of the characteristics and results of the included studies.

The result of a systematic review usually is communicated in the form of a technical report, a section of a doctoral thesis or in a journal or conference article. In our case, we will review the literature that deals with the problems related to the data warehouse, the selected articles are presented in Table 1.

Table 1. Studies selection

Authors	Paper name	Source
Taktak, S. et al., 2017	The Power of a Model-Driven Approach to Handle Evolving Data Warehouse Requirements [5]	Open Archive Toulouse
Azzaoui, M. et al., 2019	A Model Driven Architecture Approach to Generate Multidimensional Schemas of Data Warehouses [6]	The learning and Technology Library
Moukhi, N. E.et al., 2018	Towards a new method for Designing Multidimensional Models [7]	InderScience
Letrache, K.et al., 2017	The automatic creation of OLAP Cube using an MDA Approach [8]	Online Library
Khouri, S. & Bellatreche, L., 2017	Design Life-Cycle Driven Approach for Data Warehouse Systems Configurability [9]	SpringerLink
Srai, A., et al., 2017	An MDA Approach for The Development of Data Warehouses from Relational Databases using ATL Transformation Language [10]	RiPublication
El Beggar, O, et al., 2021	DAREF: MDA Framework for Modelling Data Warehouse Requirements and Deducing the Multidimensional Schema [11]	DigitalLibrary SpringerLink
El Beggar, O et al.,2016	Towards an MDA Oriented UML Profiles for Data Warehouses Design and Development [12]	IEEE Xplore
Moukhi, N E et al., 2021	Towards A New Automatic Data Warehouse Design Method [13]	DOAJ
Kalna, F et al., 2019	A Meta-Model for Diverse Data Sources in Business Intelligence [14]	Science Publishing Group
Mazón, and Trujillo, 2008	An MDA approach for the development of data warehouses [15]	ScienceDirect

4 Reviews Results

The articles that we have included have been distributed based on several classification criteria in order to answer our problem as illustrated in Table 2. These criteria are the source, the target, the modelization, the type of mapping, the type of construction, and the degree of automation of the proposed process.

Table 2. Classification of the selected studies

Paper	Source	Target	Modelization	Mappage	Construction Type	Degree of automation
Taktak, S. et al., 2017 [5]	UML Model	Data Warehouse	UML	QVT	Textual	Semi-automatic
Azzaoui, M. et al., 2019 [6]	UML Model	Data Warehouse	UML	QVT	Textual	Semi-automatic
Moukhi, N. E.et al., 2018 [7]	UML Model	Data Warehouse	UML	XSLT	Graphic	Semi-automatic
Letrache, K.et al., 2017 [8]	UML Model	OLAP	UML Profiles	ATL	Textual	Semi-automatic
Khouri, S. & Bellatreche, L., 2017 [9]	UML Model	ETL, Data Warehouse	UML	CVL	Graphic	Semi-automatic
Srai, A., et al., 2017 [10]	UML Model	Data Warehouse	UML	ATL	Textual	Semi-automatic
El Beggar, O, et al., 2021 [11]	UML Model	Data Warehouse	UML Profiles	ATL	Textual	Semi-automatic
El Beggar, O et al., 2016 [12]	UML Model	Data Warehouse OLAP	UML Profiles	ATL	Textual	Semi-automatic
Moukhi, N E et al., 2021 [13]	UML Model, XML Schema	Data Warehouse	UML	QVT	Textual	Semi-automatic
Kalna, F et al., 2019 [14]	UML Model, Fichiers XML	Data Warehouse	UML	ATL	Textual	Semi-automatic
Mazón, and Trujillo, 2008 [15]	UML Model	Data Warehouse	UML Profiles	QVT	Textual	Semi-automatic

After studying the contributions, we can conclude that most of these contributions have data warehouses as a target, for the source, the use of the UML Model was dominant, as for the modeling, UML profiles was the most used, for the mapping there were several languages used: ATL, QVT, and XSLT. After in-depth analysis, the proposed processes are semi-automatic as presented in Fig 3.

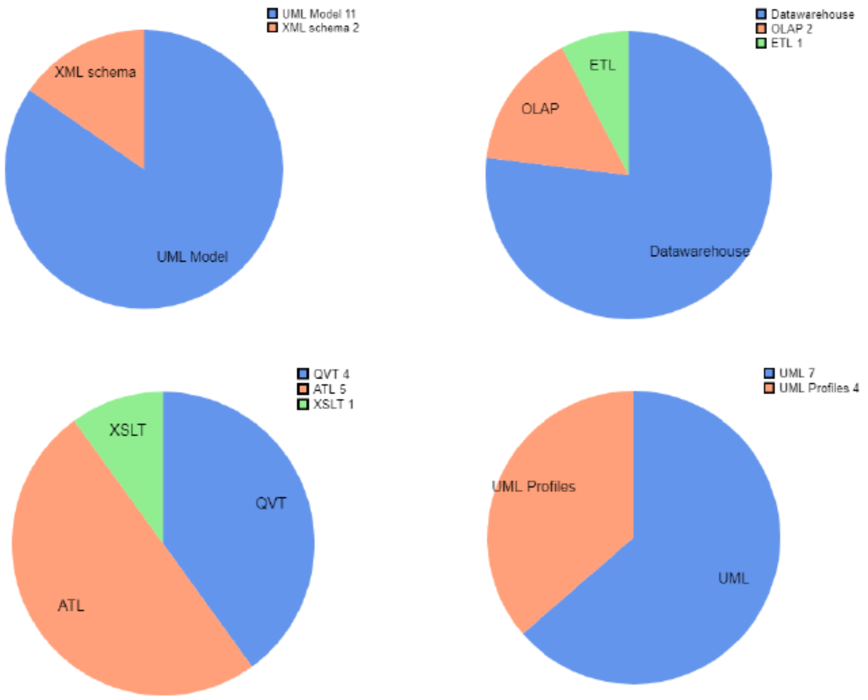

Fig. 3. Comparison between analysis criteria

5 Findings and Analysis

The article [5] implements an MDA approach, its aims to automate the propagation of changes raised by decision-makers, the solution is based on three evolution models:

- **DS Evolution Model**: it describes what can affect the multidimensional structures
- **Requirements Evolution Model**: it concerns the new needs of de decision-makers in term of axes analysis
- **M2M transformation**: it generates DWEM from REM implemented in QVT
- **M2T transformation**: it deploys the code generated

In the article [6], the authors presented an MDA approach for the development of DWs, they applied the transformations from a UML PIM to generate a DW PIM with the use of QVT language, so they were able to obtain a multidimensional model from a simple UML model.

Concerning the article [7], the authors define a list of rules that identify the components of a data warehouse, they develop an engine written in XSLT language that transforms the relational model into the multidimensional model.

The article [8] focuses on the OLAP part, the authors proposed a Meta-Model for each level of MDA abstraction and the rules required to automate the passage between these levels. Thus, at the level of CIM, the authors have proposed a UML profile that extends the BPM and use case diagrams to analyze data warehouse requirements. Then the multidimensional scheme (PIM) is extracted from the precedent CIM. The authors have used a profile UML which extends the UML class diagram and covers all concepts and types of OLAP associations for designing the multidimensional model. Then, they got 2 PS Models, relational and OLAP, which the authors modeled by using the Open Information Model (OIM) Meta-Models. In addition, for automating the transition from PIM to OLAP, the authors proposed a PDM model to describe the target platform.

The authors in the article [10] propose an MDA approach for developing and designing data warehouses based on relational models, the solution is to develop a data warehouse with a list of transformations from PIM to PSM using ATL Language.

In [11], the authors begin by proposing an independent model of multi-level computation (CIM) based on UML profiles. In fact, the UML profile offers a CIM architecture composed of two models, where each represents a level of requirement specifications. In the level of the initial requirements, the author provided a goal-based model, which extends the Meta-Model of UML use cases and represents the inceptive view of requirements DW. The approach aims to facilitate and improve communication, by making models easier to understand, and at the same time preserving the well-defined definition of UML notation.

In conclusion, the solutions that have been developed in the context of data warehouse design are not 100% automatic and they do not cover the whole decision chain, so they have mainly focused on UML: not being able to render the database design less sensitive to changes.

6 Proposal of an MDA approach for the development of data warehouses

By reviewing the literature, we notice that several researchers have applied the MDA approach for the development of data warehouses, each contribution focused on a part of the approach and proposed Meta-Models for levels of abstraction of the MDA approach. Several works have dealt with existing standard Meta-Models for the design of data warehouses, there are also some which have proposed UML profiles customizing UML.

For our solution, we propose an approach as shown in Fig 4 based on UML Profiles oriented MDA for the development of data warehouses. This solution consists first of all in defining UML profiles for the modeling of a DW. In fact, at the CIM model level, we will offer UML profiles that extend the use case and Business Process Model (BPM) Meta-Models to define user needs. The next phase is to adopt the UML profiles proposed to represent the platform independent model (PIM). Then, the relational and OLAP models considered as platform specific models (PSM) are obtained from the latter. Finally, the SQL and XMLA codes are generated to respectively build the DW and its equivalent OLAP cubes. All model transformations are implemented using the ATL transformation language. Table 3 summarizes the proposed approach.

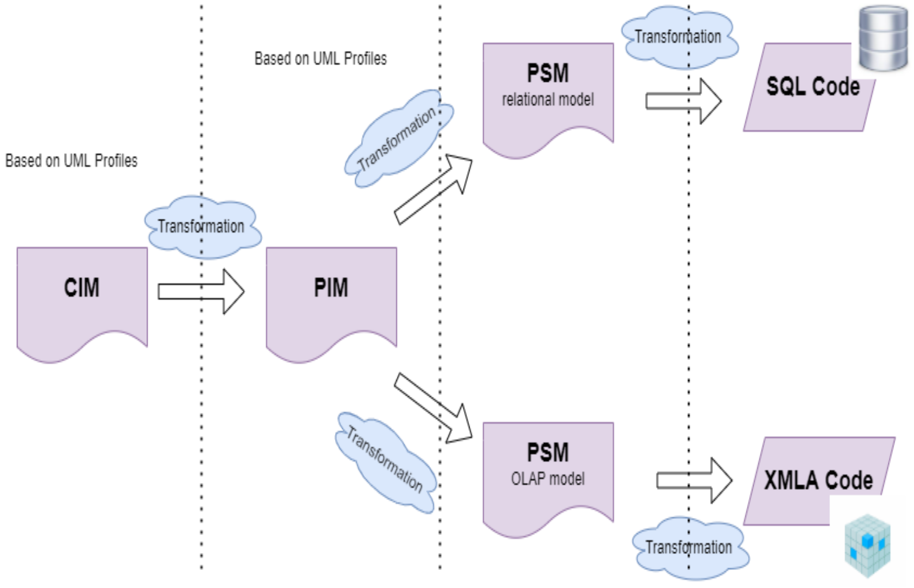

Fig. 4. The MDA Approach proposed

Table 3. Proposed criteria of MDA Approach

Source	Target	Modelization	Mapping
UML Model	Data warehouse, OLAP	UML Profiles	ATL

7 Conclusion and future work

In this research work, we have presented a systematic review of various works on the data warehouse design methods. Then, an MDA approach for the development and design of data warehouses has been proposed. The solution was based on the link between the layers of the MDA approach with the different modeling points of the data warehouses. The CIM platform was represented by the requirements of decision-makers, the PIM platform specified the multidimensional conceptual level and the PSM platform specifies for each technology the reference DW and its OLAP cube, thus, we contribute to the continuous improvement of the productivity and performance of decision support tools.

Our future work consists of improving the proposed approach and its optimization, so we can put end to the universal issue of data warehouse design for better decision-making.

8 References

[1] I. Ahmad, S. Azhar, et P. Lukauskis, « Development of a decision support system using data warehousing to assist builders/developers in site selection », Autom. Constr., vol. 13, no 4, p. 525‑542, juill. 2004, https://doi.org/10.1016/j.autcon.2004.03.001

[2] W. H. Inmon, « Creating the data warehouse data model from the corporate data model », PRISM Tech Top., vol. 1, no 2, 2000.

[3] Y. Rhazali, A. El Hachimi, I. Chana, M. Lahmer, and A. Rhattoy, "Automate Model Transformation From CIM to PIM up to PSM in Model-Driven Architecture," Advances in Information Security, Privacy, and Ethics, pp. 262–283, 2020, https://doi.org/10.4018/978-1-7998-1082-7.ch013

[4] A. G. Kleppe, J. Warmer, J. B. Warmer, et W. Bast, MDA Explained: The Model Driven Architecture : Practice and Promise. Addison-Wesley Professional, 2003.

[5] S. Taktak, S. Alshomrani, J. Feki, and G. Zurfluh, "The Power of a Model-Driven Approach to Handle Evolving Data Warehouse Requirements," Proceedings of the 5th International Conference on Model-Driven Engineering and Software Development, 2017, https://doi.org/10.5220/0006209001690181

[6] A. Azzaoui, O. Rabhi, and A. Mani, "A Model Driven Architecture Approach to Generate Multidimensional Schemas of Data Warehouses," International Journal of Online and Biomedical Engineering (iJOE), vol. 15, no. 12, p. 18, Aug. 2019. https://doi.org/10.3991/ijoe.v15i12.10720

[7] N. E. Moukhi, I. E. Azami, and A. Mouloudi, "Towards a new method for designing multidimensional models," International Journal of Business Information Systems, vol. 28, no. 1, p. 18, 2018. https://doi.org/10.1504/ijbis.2018.091161

[8] K. Letrache, O. E. Beggar, et M. Ramdani, "The automatic creation of OLAP cube using an MDA approach", Softw. Pract. Exp., vol. 47, no 12, p. 1887‑1903, 2017, https://doi.org/10.1002/spe.2512

[9] S. Khouri et L. Bellatreche, "Design Life-Cycle-Driven Approach for Data Warehouse Systems Configurability", J. Data Semant., vol. 6, no 2, p. 83‑111, juin 2017, https://doi.org/10.1007/s13740-017-0077-8

[10] A. Srai, F. Guerouate, N. Berbiche, et H. Drissi, « An MDA approach for the development of data warehouses from Relational Databases Using ATL Transformation Language », Int. J. Appl. Eng. Res., vol. 12, no 12, p. 3532‑3538, 2017.

[11] O. El Beggar, K. Letrache, and M. Ramdani, "DAREF: MDA framework for modelling data warehouse requirements and deducing the multidimensional schema," Requirements Engineering, vol. 26, no. 2, pp. 143–165, Sep. 2020, https://doi.org/10.1007/s00766-020-00339-9

[12] O. E. Beggar, K. Letrache, and M. Ramdani, "Towards an MDA-oriented UML profiles for data warehouses design and development," 2016 11th International Conference on Intelligent Systems: Theories and Applications (SITA), Oct. 2016. https://doi.org/10.1109/sita.2016.7772270

[13] N. E. Moukhi, I. E. Azami, A. Mouloudi, et A. Elmounadi, "Towards a new automatic data warehouse design method", Electron. J. Inf. Technol., vol. 0, no 11, Art. no 11, nov. 2018, Consulté le: mai 05, 2021.

[14] F. Kalna and A. Belangour, "A Meta-model for Diverse Data Sources in Business Intelligence," American Journal of Embedded Systems and Applications, vol. 7, no. 1, p. 1, 2019, https://doi.org/10.11648/j.ajesa.20190701.11

[15] J.-N. Mazón and J. Trujillo, "An MDA approach for the development of data warehouses," Decision Support Systems, vol. 45, no. 1, pp. 41–58, Apr. 2008, https://doi.org/10.1016/j.dss.2006.12.003

[16] M. Hanine, O. Boutkhoum, A. Tikniouine, and T. Agouti, "An application of OLAP/GIS-Fuzzy AHP-TOPSIS methodology for decision making: Location selection for landfill of industrial wastes as a case study," KSCE Journal of Civil Engineering, vol. 21, no. 6, pp. 2074–2084, Dec. 2016, https://doi.org/10.1007/s12205-016-0114-4

9 Authors

Hanine Mohamed is an Assistant Professor in Computer Science at the National School of Applied Sciences, University of Chouaib Doukkali, EL Jadida (Morocco). He received a Diploma degree (M.Sc) in computer science from the University of Cadi Ayyad (Morocco), in 2010. In 2017, he obtained the Ph.D. degree in computer science from the University of Cadi Ayyad (Morocco). Then in 2018 he joined the Department of Telecommunications, Networks and Computer Science at the National School of Applied Sciences, teaching engineering students in the area of Big Data and Business Intelligence. His research interests include Business Intelligence, Big data, and Decision Support Systems.

Lachgar Mohamed received his Ph.D. in Computer sciences at the University Cadi Ayyad in Marrakesh, Morocco. He is an Assistant Professor at the Department of Telecommunications, Networks and Computer Science, National School of Applied Sciences, University of Chouaib Doukkali. His research interests include issues related to Software engineering, Mobile technologies, and machine learning. He is author of research studies published at international journals and conference proceedings.

Elmahfoudi Sara is a computer engineer. Graduated on 2021 from the National School of Applied Sciences, El Jadida, (Morocco). She is now preparing a PhD in Business Intelligence and especially in Model Driven Architecture (MDA).

Boutkhoum Omar received the Ph. D. degree from the Faculty of sciences and technologies, Cadi Ayyad University, Marrakesh, Morocco, in July 2017. He is currently working as an Assistant Professor with the Department of Computing, Faculty of Sciences, University of Chouaib Doukkali, El Jadida, Morocco. His research interests include Decision support systems, Business intelligence, blockchain and Machine learning.

Article submitted 2021-06-09. Resubmitted 2021-07-09. Final acceptance 2021-07-09. Final version published as submitted by the authors.