International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-12, October 2019

A Scalable Business Intelligence Decision-Making System in the Era of Big Data

Fatima Kalna, Allae Erraissi, Mouad Banane, Abdessamad Belangour

***Abstract:** Transformation presents the second step in the ETL process that is responsible for extracting, transforming and loading data into a data warehouse. The role of transformation is to set up several operations to clean, to format and to unify types and data coming from multiple and different data sources. The goal is to get data to conform to the schema of the data warehouse to avoid any ambiguity problems during the data storage and analytical operations. Transforming data coming from structured, semi-structured and unstructured data sources need two levels of treatments: the first one is transformation schema to schema to get a unified schema for all selected data sources and the second treatment is transformation data to data to unify all types and data gathered. To ensure the setting up of these steps we propose in this paper a process switch from one database schema to another as a part of transformation schema to schema, and a meta-model based on MDA approach to describe the main operations of transformation data to data. The results of our transformations propose a data loading in one of the four schemas of NoSQL to best meet the constraints and requirements of Big Data.*

***Keywords:** Model Driven Engineering; meta-model; business Intelligence; Big Data.*

I. INTRODUCTION

Data integration is a major step in the decision-making process and is responsible for storing data from disparate and diverse sources for decision making. Several approaches have emerged to address data integration issues including those related to data sources. Among these approaches are ETL Extract Transform Load [1], EAI Enterprise Integration Application and EII Enterprise Information Integration. The ETL approach is often used in the case of a large amount of data from different sources that must undergo complex transformations. As for the EII approach is recommended in the case of an already existing data warehouse with the possibility of linking it to data sources of a medium volume.

Revised Manuscript Received on October 30, 2019.
* Correspondence Author

Fatima Kalna, Laboratory of Information Technology and Modeling, Hassan II University, Faculty of sciences Ben M'Sik. Casablanca, Morocco. Email: fz.kalna@gmail.com

Allae Erraissi*, Laboratory of Information Technology and Modeling, Hassan II University, Faculty of sciences Ben M'Sik. Casablanca, Morocco. Email: Erraissi.allae@gmail.com

Mouad Banane, Laboratory of Information Technology and Modeling, Hassan II University, Faculty of sciences Ben M'Sik. Casablanca, Morocco. Email: mouadbanane@gmail.com

Abdessamad Belangour, LTIM, Hassan II University. FSBM. Faculty of sciences Ben M'Sik. Casablanca, Morocco.

While the EAI approach ensures the integration of data from non-accessible sources directly and storing a small amount of data [2].

The objective of our work is to answer the problem of integration of data from relational databases, multidimensional, XML and NoSQL with its four figures (Key-value, Oriented columns, Document oriented and Oriented graphs). This variety of data types supported by these databases as well as the amount of data managed by NoSQL databases leads us to opt for the ETL approach in our integration process.

We focus in this paper on the transformation phase of the ETL approach and taking into account the sources and the nature of the data extracted through it we propose an approach based on two levels of transformations: the first transformation is a transformation diagram to SC2SC scheme which aims at the unification of the multiple schemas resulting from the extraction phase. The purpose of this first transformation is to avoid the problems that can arise when calling and processing different database schemas and to obtain a single, unified schema on which to base the second transform given to D2D data which aims to clean and standardize the data and their types.

Our goal is to successfully integrate data from different databases into a data warehouse. We present in this section the problematic dealt with in this paper as well as the state of the art.

This paper is organized as follows: We present in the first section the context of our study as well as the problematic and the state of the art related to the transformation of data from various sources. In the second section, we present our approach which will be based on two levels of transformations. In the following section, we detail the two transformations by presenting the rules for mapping from one schema to another and the transformation operations that will be applied to the data. In this part, we also present a meta-model based on the MDA approach and which will describe these transformation operations. We ended our paper with a conclusion in the last section.

II. PROBLEMATIC:

From a set of data sources representing structured, semi-structured and unstructured data we need to present a set of transformations needed to prepare the data that will feed into our data warehouse which will be based under a NoSQL database. In our study, we use the ETL approach where we will perform extractions from relational databases, multidimensional, XML and NoSQL [3], each of these data sources has its data model, its entities attached and their management rules, attributes, and name, mnemonic code, type, size, and format. As a result, the problem is to define the mechanisms to manage this diversity at the model and data level to obtain a unified model with its standardized data for all data sources and the design of the data schema.

The data warehouse and it's supply are running smoothly.

III. RELATED WORK

A Big data database contains a variety of data, that is, non-standard type data that are generally referred to as complex objects: text, graphics, documents, video sequences. Today, the UML data model is a kind of reference for complex database schema representations [4]. This conceptual model, which makes it possible to describe the semantics of business objects in an application, can, therefore, be applied to the description of Big Data databases. Concerning the processes for implementing databases on NoSQL systems, several studies have focused on schema transformation. Thus, in the context of data warehouses, the work of Chevalier et al. [5] defined rules for translating a multidimensional star model into two NoSQL physical models, a column-oriented model, and a document-oriented model. The links between facts and dimensions have been translated as nesting. Li's article [6] studied the implementation of a relational database in the HBase system. The proposed method is based on rules allowing the transformation of a relational schema into an HBase schema; the relationships between the tables (foreign keys) are translated by adding the families of columns containing references. Other studies have studied the transformation of a UML class diagram into an HBase data schema with the MDA approach [7]. The basic idea is to build meta-models corresponding to the UML class diagram and the HBase column-oriented data model and then propose transformation rules between the elements of the two built meta-models. These rules make it possible to transform a DCL directly into an implementation scheme specific to the HBase system. This state of the art shows that few studies have studied the transformation of a conceptual model of complex data into a NoSQL model. In the study [7] closest to our problematic, the schema transformation rules that have been adopted, are not independent of a technical platform. We position our work against three research articles whose problems and / or proposed solutions are close to ours. The paper by Chevalier et al. [5] fits into the context of data warehousing as it examines the rules for moving from a multidimensional schema to a physical schema; two NoSQL platforms were selected: the HBase column-oriented system and the MongoDB document-oriented system. Although the starting point of the process (a multidimensional scheme) is at the conceptual level, this scheme does not have the same characteristics as a UML DCL; in particular, it includes only Fact and Dimensions classes and a unique type of link between these two classes. The paper by C. Li [7] deals with the transformation of a relational schema into an HBase-oriented schema. This work responds well to the concrete expectations of companies who, in the face of recent developments in computing, want to store their current databases in NoSQL systems. But the source of the transformation process, here a relational schema, does not present the semantic richness that can be expressed in a DCL (notable thanks to the different types of links between classes: aggregation, composition, inheritance, etc.). Papers [8] and [9] propose a schema transformation using model-driven engineering towards Big Data technologies: Apache Hive [10], and Apache Spark [11]. The work of Y. Li et al. presented in [7] are intended to specify an MDA transformation process from a conceptual schema (DCL) to an HBase physical schema. This process does not propose an intermediate level (the logical level) that would make the result independent of a particular platform. Banane et al [12] present a schema transformation approach of RDF / XML data to MongoDB the most used NoSQL database, this choice of MongoDB is based on the comparative study [13]. We are based to perform this work on the meta-models proposed by Erraissi et al [14,15,24,25,28] in this research. They defined the meta-models of the NoSQL databases.

IV. STATE OF THE ART:

ETL transformation and load extraction processes are processes that power and refresh the data warehouse from their sources [16]. Several tools are being developed to address data integration issues such as Informatica PowerCenter SAS, IBM InfoSphere DataStage, and Microsoft SSIS. In parallel, several contributions are presented under the theme of ETL process modeling. We are interested in our paper for the modeling of ETL processes. In this regard, research has proposed models based on standards such as UML to present the conceptual model of the ETL process using the class diagram [17]. Other work has been based on specific models for modeling this process such as Vassiliadis and Simitsis who proposed a conceptual model for presenting ETL operations [18]. In our paper we follow the work of authors Atigui [16], Akkaoui [19] and Munoz [20] who adopted the IDM framework to describe the different levels of abstraction of the ETL process on the one hand, and other On the other hand, we want to continue the work that concerns data-to-data transformation and schema-schema operations. The following table shows the work already done in this direction:

Table- I: Transformation rules

Transformation	Data	NoSQL
Data	Data➔Data	-
Multidimensional	-	Multidimensional➔ NoSQL
Relational	-	-Relational➔Column Oriented (Hbase/Cassandra) -Relational➔Oriented document (MongoDB) -UML Conceptual Model➔ Oriented graph (GraphDB)
NoSQL	-	-

V. PROPOSED APPROACH:

Our work aims at first transforming data source schemas: relational, multidimensional, NoSQL and XML into one of the NoSQL model (key-value, column-oriented, document-oriented, and graph-oriented) to have a single logical data schema, and in a second time to transform the data to make them conform to a common schema.

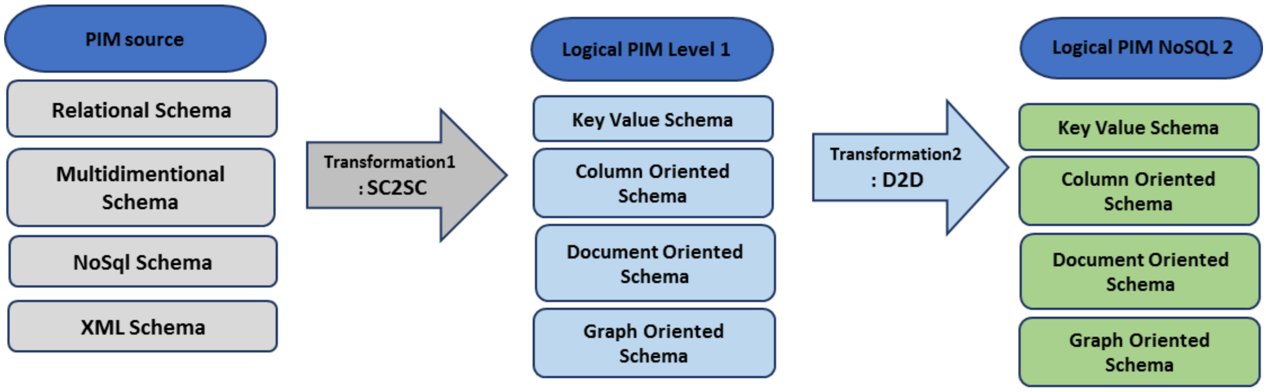

Fig. 1.Modeling Levels.

We opt for our model-driven approach (MDA), which proposes meta-models and transformations that allow us to move from one level of abstraction to another [21]. We find in our approach as illustrated in the figure above:

- A Source PIM representing the schemas of big data or other data sources.
- A first level of the Logical PIM representing the four schemas of a NoSQL database after the first transformation.
- A second level of the Logical PIM representing the four schemas of a NoSQL database after the second transformation.
- Transformation 1 SC2SC: allowing the passage of a sequence of type of data source schemas to a NoSQL schema.
- Transformation 2 D2D: to unify and standardize the Level 1 Logic PIM data and make them conform to a common schema.

A. Meta-model for PIM Source:

We now move on to define the proposed meta-models for the source PIM level. The source PIM level consists of Relational Schema, Multidimensional Schema, NoSQL Schema, and XML Schema.

Today, data sources have become numerous, to build a data warehouse to store data from these many data sources, we need the description of the schemas of these databases. We realized a meta-model implemented and validated using the Eclipse Modeling Framework. This generic meta-model encompasses five meta-models of different data sources: a multidimensional database meta-model, the second XML meta-model [15], the third is a relational database meta-model, and finally the meta-model of NoSQL database management systems. Our generic meta-model presents and provides an overview of all these data sources and also provides insight into the structure of the recording system that uses different data sources with different data schemas. Our proposed data warehouse will take this schema as input and after a series of transformations based on the Atlas Transformation Language (ATL) [22].

The following figure shows the generic meta-model for PIM Source:

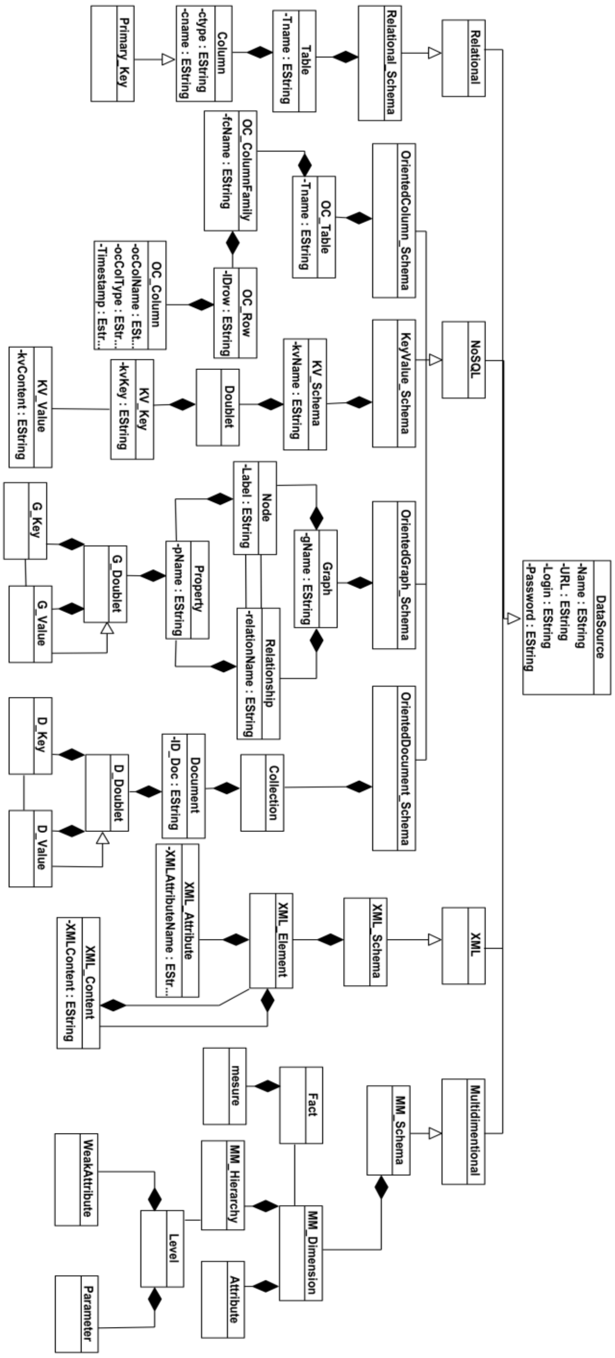

Fig. 2.The generic meta-model for PIM Source [3].

B. Meta-Model for Logical PIM

We based on the research work done by Erraissi et al [14], who proposed meta-models for the PIM Logic Level 1 components. The first proposed meta-model is that of the key-value databases, the following figure shows it:

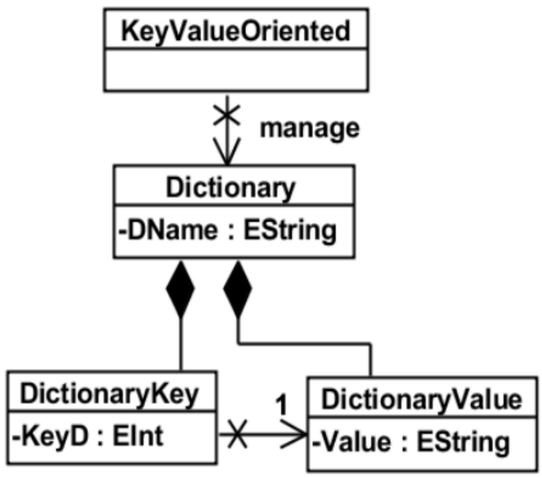

Fig. 3.Meta-model for Key/Value database [14].

Then, we define the meta-model proposed by [14] for the columns oriented databases, the following figure shows it:

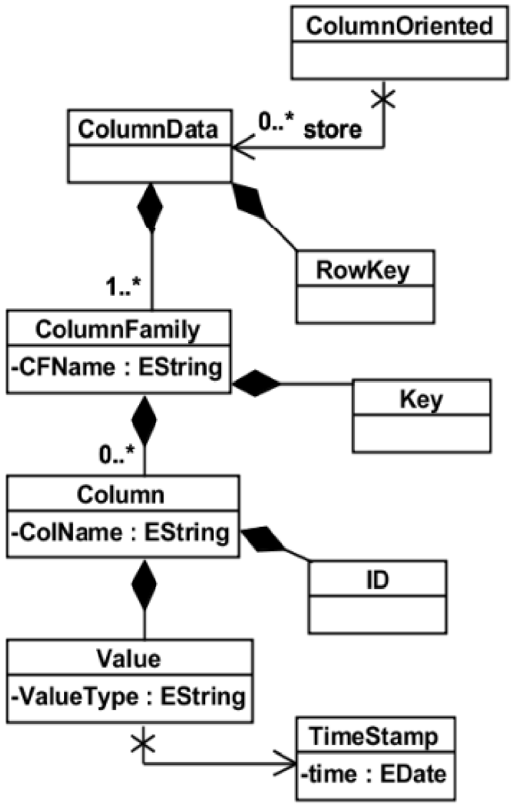

Fig. 4.Meta-model for Column-oriented database [14].

We now move on to define the meta-model that [14] proposed for document-oriented databases.

Retrieval Number: L32511081219/2019©BEIESP
DOI: 10.35940/ijitee.L3251.1081219
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-12, October 2019

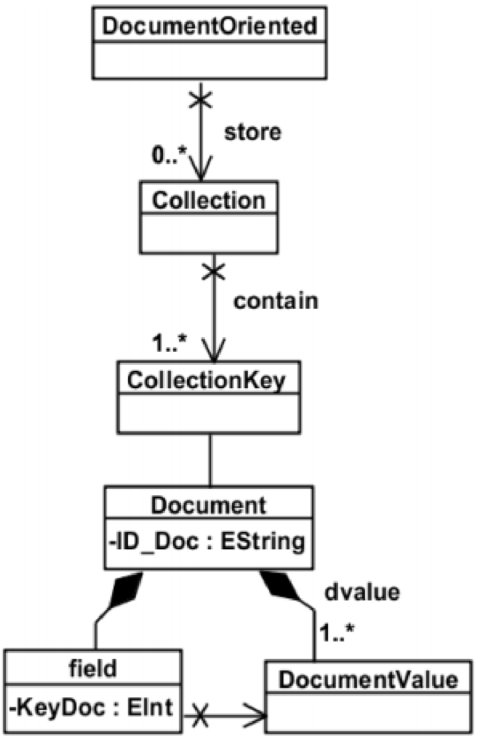

Fig. 5.Meta-model for Document-oriented database [14].

Finally, the last meta-model that will be needed to apply the source PIM transformation to Level 1 logical PIM is graph-oriented databases. Still referring to the work done by Erraissi et al [14,27], we show the meta-model they proposed for graph-oriented databases:

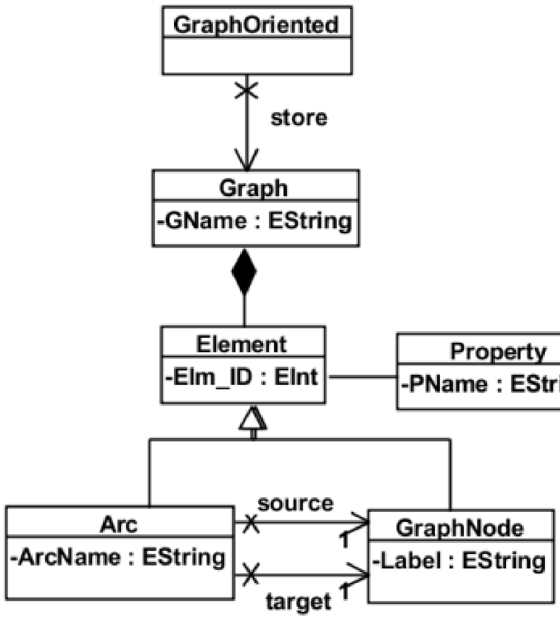

Fig. 6.Meta-model for Graph-oriented database [14].

After defining the meta-models that will be used to make our transformation, we will now present the generic meta-model for Logical PIM Level 1:

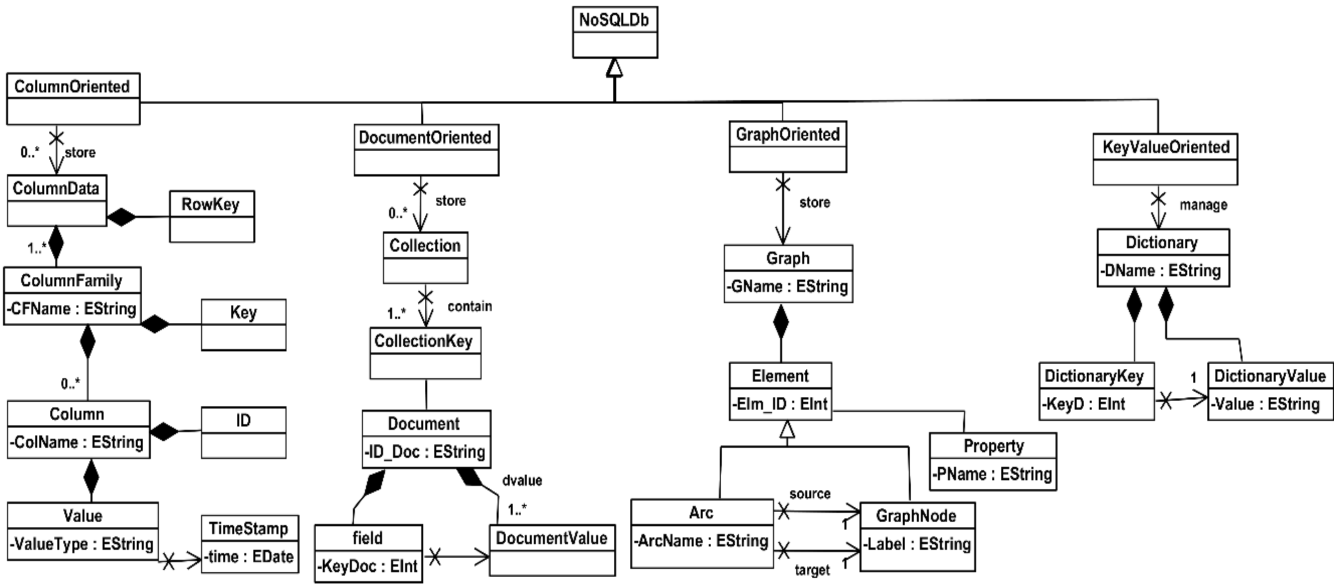

Fig. 7.Generic meta-model for Logical PIM Level 2 [14].

C. SC2SC Transformation:

The following table shows the proposed transformation rules for mapping relational, key-value, column-oriented, document-oriented, graph-oriented or multidimensional schema to a key-value, column-oriented, document-oriented, and graph-oriented schema.

Table- II: SC2SC transformation rules

Source	Component	Target-KV	Target-OC	Target-OG	Target-OD
Relational	Relational-Schema		Table	Graph	Collection

	Table	Schema	Family Column	Node	Document
	Column	Value	Column	Attribute	Document nested within the document
	Primary key	Key	ID Row		
	Foreign Key	Value	Column	Relationship	
	View				
Oriented Column	OC-Schema				
	Table	Schema		Graph	Collection
	Column family	Value		Node	Document
	Row	Key		Relationship (Between column & Row)	
	Column	Value		Attribute	Document nested within the document
Key-Value	KV-Schema		Table		Document
	Doublet		Family Column		Attribute
	Key		ID Row		
	Value		C		
Oriented Graph	OG-Schema		OC-Schema		
	Graph	Schema	Table		Collection
	Node	Node.Property.Doublet.Key==>Key	Node==>Family Column		Document
		Node.Property.Doublet.Value==>Value	Node.Property.Doublet.Value==>Value		
	Relationship	Relationship.Property.Doublet.Key==>Key	Relationship==>Family Column		Document nested within the document
		Relationship.Property.Doublet.Value==> Value	Relationship.Property.Doublet.Value ==>C		
			ID Row==>Concatenate Node.Property.Doublet.Key & Relationship.Property.Doublet.Key		
Oriented Document	OD-Schema	Schema	Table	Schema	
	Collection	Collection.Document.Doublet.Key==>Key	Collection==>Family Column	Graph	
		Collection.Document.Doublet.Value==> Value	Collection.Document.Doublet.Valeur ==>Column		
	Document	Document.Doublet. Key==>Key	Document==>Family Column	Node & also nested document are also Node	
		Document.Doublet. Value==>Value	Document.Doublet.Value==>Column	Relationship,peut etre stable entre nœud(2 nested document)	
			ID Row==>Concatenate Collection.Document.Doublet.Key & Document.Doublet.Key		
Mutidimensional Model	MM-Schema			Graph	Collection
	Fact	Fact.PrimaryKey==> Key	Family Column	Node	Document
	Dimension	Value	Family Column	Node	Document nested within the document (Fact)
	Mesure	Value	Column	Relationship	Attribute within a document (Fact)
	Hierarchy	Value	Column		Attribute nested within a dimenesion
	Weak attribute	Value	Column		Attribute within a dimension
	Parameter	Value	Column		Attribute within a dimension

The following code shows an ATL code snippet of our SC2SC transformation. We transformed the relational database into a column-oriented NoSQL database. Rule 1, transforms the relational schema of a table to a column-oriented table (HTable in our case of HBase), the second Rule transforms a relational table line into a column-oriented NoSQL column.

```
rule Relational2ColumnOriented {
from
        t: Table! Relational_Table (relational row)
to
        s: Schema! (
            CO_columnFamily <- t.row/relational_colu
mn1 + ' ' + t. row/relational_column2
                )
}
rule RelationalRow2ColumnOriented_colomn{
from
        r: RelationalRow!value ()
to
        cf: Column_Families!Column (
            cf.Column <- r.Row + ' ' + r.Column
            cf.timestamp< - 0000
                )
}
```

Fig. 8. Extract from ATL code transformation SC2SC.

D. D2D transformation:

To load the data into the warehouse, they must first undergo a set of transformation operations such as selection, linking, conversion, aggregation and generation through the process of extracting the data. Transformation and loading ETL [23], to describe these operations one uses the OCL language which makes it possible to express constraints and queries on the models and independently of the platform used:

- **Selection:** aims to select the data according to a well-defined criterion.
  - **Select ():** allowing the selection of a subset of data that meets a condition
  - **Collect ():** allowing the construction of a new collection containing the value of the affected expression.
  - **Reject ():** allowing the selection of a subset of data that does not respect a condition
  - **Exists ():** allowing the checking of expression for the elements of the selection.
  - **Sortedby ():** allowing sorting of the elements of the selection in the order chosen.
- **Link:** aims to link models and attributes to keep the connection between each attribute of the target and their corresponding attributes of the source and that source-target or target-source navigation is possible, which respects the interoperability that guarantees us the MDA.
  - **Link_ShSR_2_ShTR():** to establish the link between the source model and the target model.
  - **Link_AttSC_2_AttTR():** to establish the link between the source attribute and the target attribute.
- **Conversion:** aims to convert the formats and types of source attributes and to apply arithmetic operations and extract information in a predefined format.
  - **Conversion_String:** to reformat strings using the following functions:
  - Concat ()
  - To Upper ()
  - Trim ()
  - To Lower ()
  - Replace All ()
  - Replace First ()
  - SubString ()
    - **Conversion_Data_Type:** to change the data type to one of the following types:
      - ToString()
      - ToReal()
      - ToInteger()
      - ToBoolean()
      - ToDate()
    - **Conversion_Arithemetic:** to perform the following arithmetic operations on the data:
      - Add()
      - Multiply()
      - Devide()
      - Subtract()
    - **Conversion_Formate:** to derive new information from the attributes in question:
      - GetTime()
      - GetYear()
      - GetMonth()
      - GetDate()
      - GetHour()
      - GetMinute()
    - **Aggregation:** to aggregate data according to a criterion:
      - Min()
      - Max()
      - Avg()
      - Groupby()
      - Count()

E. Proposed meta-model for D2D Transformation:

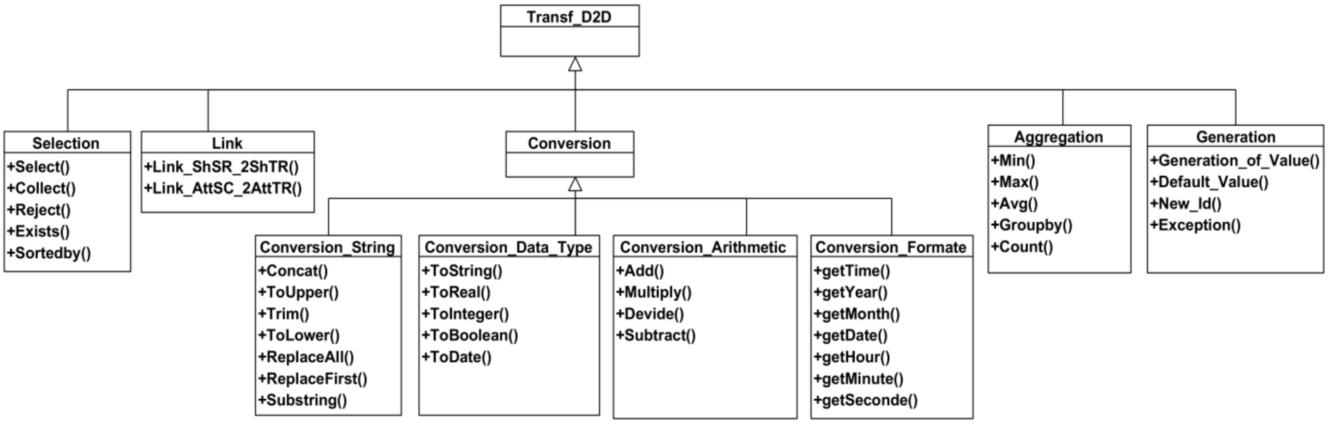

Fig. 9.Meta-model describing D2D transformation operations.

VI. EXPERIMENTS AND VALIDATION

A. Configuration and test environment

To measure the performance of our approach, we tested the execution of our SC2SC and D2D transformation, the tests are performed on a server machine with 16 GB of memory storage and 8TB of disk storage to store the relational data, the XML data, and NoSQL data stored in the HBase version 2.0.0 database, Hadoop version 3.0.3. For the dataset used in this test, is a partitioned dataset on the different data sources: the relational database, an XML file, the multidimensional data, and a large partition of 17 GB stored in column-oriented data management system. HBase ". At the transformation level, we have chosen the ATL language [26] (Atlas Transformation Language). Our choice of the transformation language was based on criteria specific to our approach. Indeed, the tool must be integrated into the EMF environment for easy use with modeling and meta-modeling tools. Thus, we used the operational QVT language.

B. Experiments results

Table 3 illustrates the data loading time in our data warehouse to combine the data that comes from different data sources.

Table- III: Loading time from different sources

	Relationnel database	XML File	Multidimentionnel schema	HBase NoSQL
Loading Time (s)	63.4	17.7	85.1	183

We present in the following the execution time results of our SC2SC and D2D transformations, noted that the D2D transformation takes a lot of time is requires a further phase of data cleaning. The results obtained show the efficiency of our transformation approach and lead to a new version of ETL, an ETL that allows using as well as traditional data sources like relational data, XML data, and multidimensional data, a new source of NoSQL data, which has become widely used.

Table- IV: transformation execution time (s) for SC2SC and D2D.

	Relationnel database	XML	Multidimentionnel	NoSQL
SC2SC	13.5	9.2	21.8	94.7
D2D	117	34.6	125.3	162.9

The following figure 10 shows graphically the execution time obtained for the two transformations SC2SC and D2D. We note that the NoSQL database management system takes a lot of time compared to other data sources since it contains the large partition of the dataset.

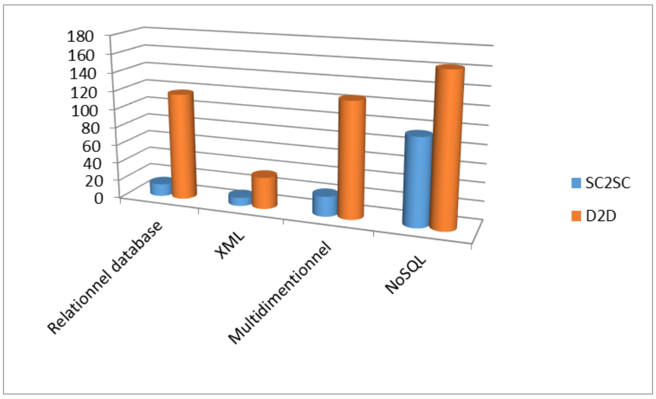

Fig. 10. SC2SC and D2D transformations time(s).

VII. CONCLUSION:

Our work is part of the development and evolution of databases and their adaptation to the Big Data phenomenon and its relationship with decision-making. Our goal is to prepare a data warehouse connected to various data sources and present its database schema under the NoSQL model. This article focuses on the transformation phase of the ETL process that aims to make all schemas and data conform to a single schema. Through our approach we presented two levels of transformation: SC2SC transformation and D2D transformation, the first allowed us to unify the schemas and the second allowed us to unify the data, these two transformations allows to obtain two levels of Logical PIM of the MDA approach. Our next work will continue with the ETL process and will focus on the loading phase, and this by presenting a suite of meta-models that will describe the three levels of abstraction after the definition of the physical platforms on which our data warehouse will be based.

REFERENCES

1. Kimball, Ralph, and Joe Caserta. The data warehouse ETL toolkit: practical techniques for extracting, cleaning, conforming, and delivering data. John Wiley & Sons, 2011.
2. Colin White, Data Integration: Using ETL, EAI, and EII Tools to Create an Integrated Enterprise, academia, 2005.
3. Fatima Kalna, Abdessamad Belangour, A Meta-model for Diverse Data Sources in Business Intelligence, American Journal of Embedded Systems and Applications (AJESA), 2019.
4. Tanasescu, Adrian, Omar Boussaid, and Fadila Bentayeb. "Preparing complex data for warehousing." In The 3rd ACS/IEEE International Conference onComputer Systems and Applications, 2005., p. 30. IEEE, 2005.
5. Chevalier, Max, Mohammed El Malki, Arlind Kopliku, Olivier Teste, and Ronan Tournier. "Entrepôts de données multidimensionnelles NoSQL." 2015.
6. Li, Chongxin. "Transforming relational database into HBase: A case study." In 2010 IEEE international conference on software engineering and service sciences, pp. 683-687. IEEE, 2010.
7. Li, Yan, Ping Gu, and Chao Zhang. "Transforming UML class diagrams into HBase based on meta-model." In 2014 International Conference on Information Science, Electronics and Electrical Engineering, vol. 2, pp. 720-724. IEEE, 2014.
8. M. Banane, and Abdessamad Belangour. "New Approach based on Model Driven Engineering for Processing Complex SPARQL Queries on Hive." International Journal of Advanced Computer Science and Applications (IJACSA) 10, no. 4 (2019).
9. M. Banane, and Abdessamad Belangour. « Querying massive RDF data using Spark». International Journal of Advanced Trends in Computer Science and Engineering 8, nº 4 (2019): 1481 - 1486.
10. Huai, Yin, Ashutosh Chauhan, Alan Gates, Gunther Hagleitner, Eric N. Hanson, Owen O'Malley, Jitendra Pandey, Yuan Yuan, Rubao Lee, and Xiaodong Zhang. "Major technical advancements in apache hive." In Proceedings of the 2014 ACM SIGMOD international conference on Management of data, pp. 1235-1246. ACM, 2014.
11. Shoro, Abdul Ghaffar, and Tariq Rahim Soomro. "Big data analysis: Apache spark perspective." Global Journal of Computer Science and Technology (2015).
12. M. Banane, and Abdessamad Belangour. « RDFMongo: A MongoDB Distributed and Scalable RDF management system based on Meta-model». International Journal of Advanced Trends in Computer Science and Engineering 8, nº 3 (2019): 734 – 741
13. M. Banane, and Abdessamad Belangour. « An Evaluation and Comparative study of massive RDF Data management approaches based on Big Data Technologies». International Journal of Emerging Trends in Engineering Research. 7, nº 7 (2019): 48 – 53.
14. Erraissi, Allae, and Abdessamad Belangour. « Hadoop Storage Big Data Layer: Meta-Modeling of Key Concepts and Features ». International Journal of Advanced Trends in Computer Science and Engineering 8, nº 3 (2019): 646-53.
15. Erraissi, A., & Belangour, A. (2018). Data sources and ingestion big data layers: meta-modeling of key concepts and features. International Journal of Engineering & Technology, 7(4), 3607–3612. https://doi.org/10.14419/ijet.v7i4.21742
16. F.Atigui, Model-Driven Approach for Implementing and Reducing Data Warehouses, thesis, 2013.
17. Juan Trujillo and Sergio Luj´an-Mora. A UML Based Approach for Modeling ETL Processes in Data Warehouses. In Proceedings of the 22nd International Conference on Conceptual Modeling, ER, pages 307–320, Chicago, IL, USA, 2003.
18. Alkis Simitsis and Panos Vassiliadis. A Methodology for the Conceptual Modeling of ETL Processes. In Proceedings of the 15th Workshops on Advanced Information Systems Engineering, CAiSE Workshops, Klagenfurt/Velden, Austria, 2003.
19. Zineb El Akkaoui, Esteban Zim`anyi, Jose-Norberto Maz´on, and Juan Trujillo. A model-driven framework for ETL process development. In Proceedings of the 14th International Workshop on Data warehousing and OLAP, DOLAP, pages 45–52, Glasgow, Scotland, UK, 2011.
20. Lilia Munoz, Jose-Norberto Maz´on, and Juan Trujillo. Automatic generation of etl processes from conceptual models. In Proceedings of the 12th International Workshop on Data Warehousing and OLAP, DOLAP, pages 33–40, Hong Kong, China, 2009.
21. Fatma Abdelhedi, Amal Ait Brahim, Faten Atigui, Gilles Zurfluh, MDA-Based Approach for NoSQL Databases Modelling, International Conference on Big Data Analytics and Knowledge Discovery, 2017
22. “ATL: Atlas Transformation Language Specification of the ATL Virtual Machine.”
23. Panos Vassiliadis. A Survey of Extract-Transform-Load Technology. International Journal of Data Warehousing and Mining, IJDWM, 5(3) :1–27, 2009
24. Erraissi, Allae, and Abdessamad Belangour. « Meta-Modeling of Big Data Management Layer ». International Journal of Emerging Trends in Engineering Research 7, no 7 (15 July 2019): 36-43. https://doi.org/10.30534/ijeter/2019/01772019.
25. Erraissi, Allae, and Abdessamad Belangour. « Meta-Modeling of Big Data Visualization Layer Using On-Line Analytical Processing (OLAP) ». International Journal of Advanced Trends in Computer Science and Engineering 8, nº 4 (25 août 2019): 990-98. https://doi.org/10.30534/ijatcse/2019/02842019.
26. “ATL: Atlas Transformation Language ATL Starter’s Guide,” 2005.
27. A., Erraissi A., Belangour A. (2019) Capturing Hadoop Storage Big Data Layer Meta-Concepts. In: Ezziyyani M. (eds) Advanced Intelligent Systems for Sustainable Development (AI2SD’2018). AI2SD 2018. Advances in Intelligent Systems and Computing, vol 915. Springer, Cham.
28. A. Erraissi and A. Belangour, "Meta-modeling of Zookeeper and MapReduce processing," 2018 International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS), Kenitra, Morocco, 2018, pp. 1-5. doi: 10.1109/ICECOCS.2018.8610630.

AUTHORS PROFILE

Fatima Kalna is a Business Intelligence Engineer and a Ph.D. at the Faculty of Sciences Ben M‘Sik, Laboratory of Information Technology and Modeling (LTIM), Hassan II University of Casablanca, Morocco. His research fields: Business Intelligence (BI), Decision-making systems, Model-Driven Engineering, and Big Data.

Allae Erraissi is a Ph.D. on computer science at the Faculty of Sciences Ben M'Sik at the Hassan II University, Casablanca, Morocco. He won his Master Degree in Information Sciences and Engineering from the same University in 2016 and is currently working as a Mathematics teacher in a High school in Casablanca, Morocco. His main interests are the new technologies namely Model-driven engineering, Cloud Computing, and Big Data.

Mouad Banane is a Ph.D. at the Faculty of Sciences Ben M‘Sik, Laboratory of Information Technology and Modeling (LTIM), Hassan II University of Casablanca, Morocco. His research fields: Model-Driven Engineering, Semantic Web, and Big Data.

Abdessamad Belangour is a Full Professor at the Faculty of Sciences at the Hassan II University, Casablanca, Morocco. He is mainly working on Model-Driven Engineering approaches and their applications on new emerging technologies such as Big Data, Business Intelligence, Cloud Computing, Internet of Things, Real-time embedded systems, etc.