# X-ETL: A Data-Driven Method for Designing Star Schemas

https://doi.org/10.3991/ijes.v7i1.10009

Nawfal El Moukhi (✉), Ikram El Azami, Abdelaaziz Mouloudi
Ibn Tofail University, Kenitra, Morocco
elmoukhi.nawfal@gmail.com

Abdelali Elmounadi
Mohammed V University, Rabat, Morocco

**Abstract**—The data warehouse design is currently recognized as the most important and complicated phase in any project of decision support system implementation. Its complexity is primarily due to the proliferation of data source types and the lack of a standardized and well-structured method, hence the increasing interest from researchers who have tried to develop new methods for the automation and standardization of this critical stage of the project. In this paper, the authors present the set of developed methods that follows the data-driven paradigm, and they propose a new data-driven method called X-ETL. This method aims to automating the data warehouse design by generating star models from relational data. This method is mainly based on a set of rules derived from the related works, the Model-Driven Architecture (MDA) and the XML language.

**Keywords**—Data warehouse, Relational model, Multidimensional model, Design of data warehouses, MDA

## 1 Introduction

The evolution of technology and storage media in recent years has enabled companies and organizations to produce and manage a large amount of data. These data are stored in the company's operational systems and are processed by OLTP (Online Transaction Processing).

However, the use of these operational systems and transactional processes for reporting and decision-making support is described as difficult and tedious. These traditional systems have shown their limit in the operational data analysis and the extraction of strategic information that can be used as a support for decision-making [1], hence the birth of Business Intelligence and new Decision Support Systems (DSS).

After its emergence in the early 1990s [2], business intelligence was the subject of a series of researches that dealt with various issues related to: data warehousing, the integration of heterogeneous data [3], data mining techniques [4], On-line Analytical Processing (OLAP), etc. Moreover, particular attention was paid to the issue of data

warehouse design and the automation of this step given its complexity and importance for all other steps of the decision chain. Therefore, several works have focused on this issue and different design methods have been developed.

The analysis of the previous research works revealed that the design methods follow mainly three approaches, namely:

**Requirements-driven approach:** This approach gives priority to user requirements to design the data warehouse model. Thus, surveys are conducted with several user groups in order to collect requirements information. Requirements may vary over time. Consequently, these surveys should be planned periodically and a further evolution of the data warehouse model should be considered. Generally, methods following this approach require experts intervention to transform the information collected into a functional model that best meets the users' needs [5];

**Data-driven approach:** This approach is based on an analysis of the semantic structure of the OLTP data to generate a data warehouse model. Generally, the analysis aims, firstly, to detect fact tables, dimension tables and hierarchies before generating the model. The simplest methods are those aimed at generating star models, whereas the most complicated ones generate constellation schemes. The main disadvantage of this approach is that it ignores users' needs and the strategic objectives of the organization and may generate inadequate models especially when transactional data are poorly structured;

**Hybrid approach:** Combines both approaches above-mentioned to design the data warehouse from the data sources while meeting the requirements and users' needs [6].

For their part, the authors have chosen the data-driven approach simply because it allows to save an enormous amount of time and money [7], since the start of the data warehouse Design project requires only the availability of transactional data. This is a very important advantage for companies and decision-makers.

In this paper, the researchers have identified and analyzed the significant works carried out in this area (section 2). From these works, they have developed a list of rules in section 3, before presenting the new method and giving some critical observations in Section 4 and conclude with the perspectives of the research.

## 2 Related Work

There are many methods for automating the data warehouse design process. Among these methods the authors cite those that follow the data-driven approach:

**Golfarelli et al.** [8]: This work proposes a semi-automated methodology to build Dimensional Fact model from the pre-existing Entity/Relationship schemes describing a database. The conceptual model consists of tree-structured fact schemes whose basic elements are facts, attributes, dimensions and hierarchies; other features, which may be represented on fact schemes, are the additivity of fact attributes along dimensions, the optionality of dimension attributes and the existence of non-dimension attributes. Compatible fact schemes may be overlapped in order to relate and compare data. Fact schemes may be integrated with information of the conjectured workload, expressed in terms of query patterns, to be used as the input of a design phase whose output are

the logical and physical schemes of the data warehouse. However, no procedure to translate the conceptual schema into logical and physical schemas has been presented;

**Moody and Kortink** [9]: This is one of the first data-driven methods and one of the most cited. It is a formal and structured method to derive logical multidimensional schemes from the Entity / Relationship models of the company. This method is divided into three steps:

- **The first step:** Enables the classification of the entities in the data model into 3 categories, namely:
  - Transactional entities: Contain details about specific events in the company (orders, sales, etc.). These entities will form the basis of fact tables in star schemas because they represent the events that decision makers aim to analyze.
  - Component entities: Are directly related to transactional entities via one-to-many relationships. These entities will form dimension tables in star schemas.
  - Classification entities: Are related to component entities by a chain of one-to-many relationships. These entities will constitute dimension hierarchies in star schemas.
- **The second step:** Identifies existing hierarchies in the data model based on a list of rules.
- **The last step:** Consists of collapsing these hierarchies and aggregating the transactional data to form dimensional models. It also defines a new type of schema called a star cluster schema. This is a restricted form of snowflake schema, which minimizes the number of tables while avoiding overlap between different dimensional hierarchies. Individual schemas can be collected together to form constellations or galaxies;

**Vrdoljak et al.** [10]: the authors proposed a method for designing a multidimensional schema from an XML schema. The method includes the following steps:

- Pre-processing the XML schema.
- Creating and transforming XML schema into a graph by applying two transformations: Functional dependencies are explicitly stated (using key attributes) and nodes that do not store any value are rejected.
- Identifying facts among all nodes and arcs of the graph
- Constructing a multidimensional schema for each fact, which includes constructing a dependency graph.

The root of this graph represents the fact while dimensions and hierarchies are defined by identifying relationships with cardinalities one-to-many or many-to-many. Like all other data-driven approaches, this method remains semi-automatic since it requires the intervention of the designer who has to decide whether relationships in hierarchies are multiple and whether they are interesting for aggregation;

**Jensen et al.** [11]: The authors proposed a semi-automatic method to discover the multidimensional structure from relational databases. The discovery of the multidimensional structure consists of four general steps:

- **First step**: Aims to collect metadata (name of tables and attributes, cardinalities of attributes, etc.).
- **Second step**: Using a Bayesian network, these metadata are annotated with additional information that divides the attributes into three categories according to their potential role: measure, keys and descriptive data.
- **Third step**: Discovers the interrelations between tables by identifying functional and inclusion dependencies, which represent many-to-one relationships and that will subsequently give rise to dimensions.
- **The last step**: Derives snowflake schema from this metadata:

Fact tables are proposed based on the cardinality of tables and the number of measures identified by the Bayesian network. Then, the user has to choose those that represent an interest.

A connected graph is considered as a dimension if there is an inclusion dependency between a fact table and a graph node. This method also contains an algorithm to generate the aggregation hierarchy for each dimension;

**Song et al.** [5]: The authors presented a semi-automatic lexical method for generating star schemas from an Entity-Relationship Diagram (ERD) by analyzing its semantics and its structure. The main innovation in this method is the use of the connection topology value (CTV) to identify candidate fact and dimension tables, as well as the use of the annotated dimensional design patterns and WordNet to extend the list of dimensions. The Connection Topology value of an entity is a composite function of the topology value of direct and indirect many-to-one relationships. This composite function gives a higher weighting factor to direct relationships. Thereby, all entities with a CTV value higher than the calculated threshold are proposed as fact tables. However, this method doesn't give any information on how to identify measures and aggregation hierarchies;

**Romero and Abelló** [12]: Introduced a semi-automatic approach to support the data warehouse design by using the AMDO (Automating Multidimensional Design from Ontologies). It's an automatic method for identifying concepts that can play multidimensional roles, from an ontology representing the activity field. This approach consists of three different steps:

- **The first step** is to retain the concepts that are related to most measures and potential dimensional concepts as candidate fact tables. After that, the user can choose among these candidate tables those that represent an interest.
- **The second step** involves generating sets of concepts for each selected fact table. These sets of concepts will form the basis for generating the data cubes that will be suggested to the end-user.
- **The third step** gives rise to dimension hierarchies by rearranging the dimensional concepts. For every concept identified as a dimension, its hierarchy of levels is conformed to those concepts related to by typical part-whole relationships [13]. In this step, AMDO builds up graphs giving shape to each dimension hierarchy that the user may tune up to his needs.

The analysis of the different data-driven methods allowed the authors to make two main observations:

- All data-driven methods developed to date are semi-automatic;
- Most of the existing methods are based on the analysis of the source model structure and some rules to generate a multidimensional model.

Indeed, the analysis of the data sources and their structures is generally not sufficient to ensure a fully automatic transformation that could generate good results whatever the field of application. Such a transformation always requires manual intervention or assistance from the designer. However, it is possible to rely on the previous methods to develop a new approach that automates the transformation process as much as possible and minimizes human intervention. Thus, the authors have attempted to identify all the rules used in these methods (section 3), to refine them and integrate them into a single approach that will be fully automatic and applicable in all areas.

## 3 Rules for Data Warehouse Design

This section presents a set of rules drawn from the previous works and the methods discussed in Section 2. These rules will form the foundation of the solution to standardize the process of data warehouse design:

### 3.1 Rules for fact tables and measures

In order to extract the corresponding fact tables and measures, the proposed method is governed by a number of rules:

- The fact tables are the concepts of main interest for the decision making process. They correspond to events that always occur in the organization or company [14];
- The measures of the fact table should be numeric and additives (at worst semi-additives) [15];
- The data of a fact table are fixed and cannot be changed [16];
- A fact table represents always a particular activity and should be interrogated from a particular context (one or a few dimensions);
- No line of the fact table must contain an empty value;
- A fact table contains only the foreign keys that represent the primary keys of the dimensions, and these keys must be numeric in order to make the fact table more efficient [17];
- Each combination of dimension values defines an instance of the fact table and which is characterized by one and only one value for each measure.

A mathematical representation of the rules above-mentioned is given below:

Let $T_F$ be a fact table, $M_{TF}$ a fact table measure, $D_i$ a dimension of the fact table and m an instance of $M_{TF}$.

$T_F = P(E_v)$
with:
$P$: Main interests
$E_v$: Company events;
Let $m_1$ and $m_2$ be two instances of $M_{TF}$
$\exists m_3 = m_1 + m_2$
with $m_3$ an instance of the same measure $M_{TF}$;
Suppose that $f$ is a change function on $T_F$
$\forall m \in M_{TF}$
$f(m)=0$;
Let F be the set of fact tables and A a particular activity of the organization
For each $T_F \in F$ we have:
$T_F = A$;
$\forall T_F$ there is at least one function $f$ which applies at least one dimension $D_i$ on $T_F$;
Let $L_{TF}$ be the set of rows of a fact table and $l$ a row of $L_{TF}$.
$\forall l \in L_{TF}$
$l \neq \emptyset$;
Let f be a foreign key and p a primary key
we have :
$\{f_1,f_2,...,f_n\}=\{p_1,p_2,...,p_n\}$
$\forall K \in \{1,2,...,n\}$
with $f_k \in T_F$ and $p_k \in D_i$
and $f_k$ and $p_k$ of type Integers;
Let C be the set of combinations of dimension values
c a combination of C and $f$ a function on $M_{TF}$:

- For each instance m of MTF, the combination f (m) ∈ C.
- For each combination c of C, the equation f (m) = c admits a unique solution (any combination c of C admits a unique antecedent MTF) f (m) is bijective.

### 3.2 Rules for dimensions and attributes

- The dimensions determine how fact instances can be aggregated significantly for decision making process;
- A fact table always contains the time dimension;
- The dimensions should have numeric primary keys;
- The primary key of each dimension table should be unique (preferably auto-increment), and fields should have an atomic value (not compound);
- The dimension hierarchies should preferably have a simple form of 1-n type, and avoid relationships of n-n type;
- A non-dimensional attribute contains additional information on an attribute of the hierarchy, and it is linked by to-one relationship [8];
- The non-dimensional attributes cannot be used for aggregation [8];
- The relationship between a fact table and a dimension is always many-to-one.

Below are the mathematical representations of the rules for dimensions and attributes:

Let $T_F$ be a fact table and $D_i$ a dimension of the fact table.

Let *f* be an aggregation function on $T_F$

*f* is significant if and only if *f* applies one or more dimensions on the instances of $T_F$.

$\forall T_F$ ($T_F \ni D_{it}$) with $D_{it}$ a time dimension;

Let $C_p$ be the set of primary keys of $D_i$

$\forall D_i$

$C_p \in \mathbb{N}$.

Let $C_p$ be the set of primary keys of $D_i$ and $p_1$ and $p_2$ two instances of $C_p$

$\forall p_1$ and $p_2$

$p_1 \neq p_2$;

Let R be a relationship between two dimensions

$\forall R$

$R \neq (\infty, \infty)$.

Let *f* be an aggregation function, A the set of its attributes, and $a_n$ a non-dimensional attribute.

For any non-dimensional attribute $a_n$ we have:

$a_n \notin A$;

Let R be a relationship between $T_F$ and $D_i$

$\forall T_F$ & $D_i$ / $R = (\infty,1)$.

## 4 The X-ETL Method

The X-ETL method is based principally on the rules developed in the previous section, the MDA techniques [18], and the XML language. The major strength of XML lies in its extensibility and its ability to describe any data field [19]. Indeed, XML allows us to structure and define the vocabulary and the syntax of the data contained in a legal document. The definition of the vocabulary and the syntax (i.e., the grammar) of a family of XML documents is described through an XSD Schema. Therefore, the authors created two XSD schemas, the first one for validating the XML file that describes the relational model (Input), and the second one for validating the generated multidimensional model (Output). The authors used the relational model because it provides several advantages compared over other models [20].

These two XSD files were developed from the meta-models presented in this paper [21] and specifically the relational meta-model (see Figure 1) and the multidimensional meta-model for the star schema (See Figure 2) since the model that will be generated by the X-ETL method is a star model. These two meta-models have been designed with Ecore

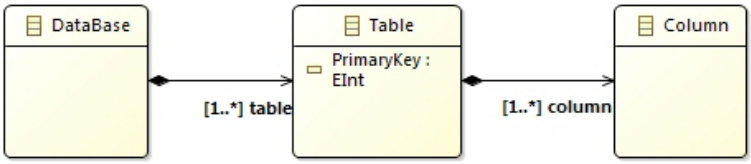


**Fig. 1.** The relational meta-model

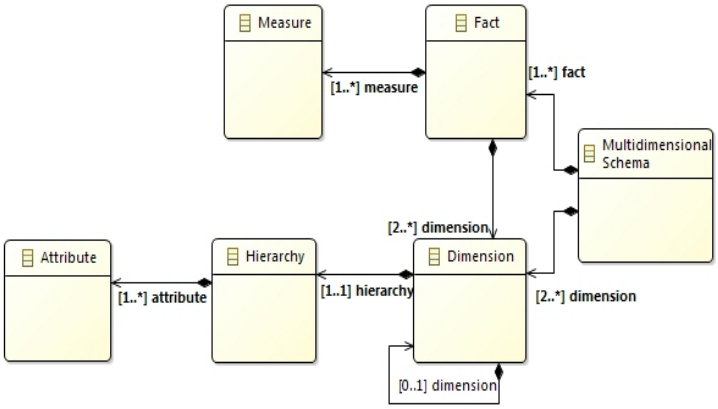


**Fig. 2.** The multidimensional meta-model for the star schema

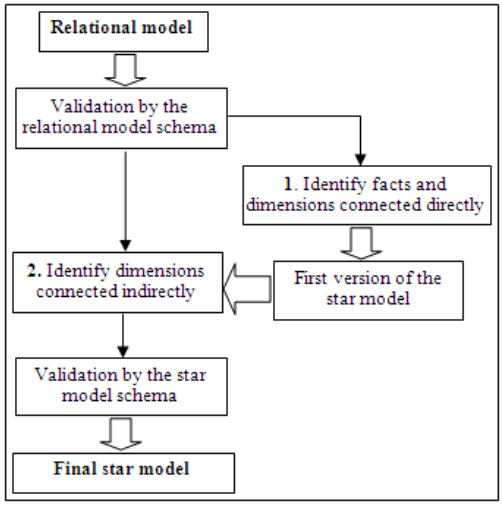


**Fig. 3.** The steps of the X-ETL method

The X-ETL method consists mainly of two steps:

- Identify fact tables and dimension tables that are directly connected;
- Identify dimension tables that are indirectly connected.

The diagram in Figure 3 illustrates the X-ETL method.

The data warehouse development starts with the detection of multidimensional elements, which are mainly facts and dimensions. This is a crucial step on which all other steps of the process will depend, and it will strongly influence the final model that will be generated. It is therefore essential to give this step a major interest in any data warehouse design method. In the data-driven approach, the detection of multidimensional elements can only be achieved through a detailed analysis of the data sources and a number of criteria that can be automated and that can be considered as solid arguments, which justify the choices made. These criteria can be drawn from the rules defined in Section 3.

### 4.1 Facts and dimensions connected directly

There is no doubt that among the multidimensional elements, the facts are the most important since they represent user's interests and form the basis of the multidimensional schema. Therefore, any method should aim, primarily, to detect these tables based on one or more criteria. According to methods discussed in Section 2, it is very likely that a table that has the highest number of relationships in the source model represents an interest for decision makers and a core activity of a company. Thereby, the number of relationships can be a good criterion to locate fact tables.

In the first step, the method allows the detection of a set of candidate fact tables by calculating the number of foreign keys in each table of the relational model (data source) and by classifying them from the largest to the smallest. Tables containing the highest number of foreign keys will be retained as the first candidate fact tables. The second candidate fact tables will be those with a number of foreign keys less than the first one and so forth.

Every table connected directly to a candidate fact table with a many-to-one relationship is automatically retained as a dimension table. Once this first step is completed, the first version of the star model containing a fact table and dimensions connected directly is generated.

### 4.2 Dimensions connected indirectly

The second step of the method aims to detect the dimensions that are indirectly connected to the fact table in the source model. To achieve this goal, the method is based on two fundamental principles:

- A fact table is always related to a dimension by a many-to-one relationship (Rules 7 in section 3);
- The transitivity of cardinalities in relational models.

So, to detect the potential dimensions, the application has to calculate the cardinalities of the tree by transitivity and every time we get the cardinality many-to-one as result, the table is retained as a candidate dimension table.

The researchers specify that the second rule of dimensions states that a multidimensional schema always contains a time dimension. Therefore, if no time dimension has been detected in the relational schema, the X-ETL engine will automatically create a new one.

In addition, the authors recall that their method is iterative. This means that for each candidate fact table, the program executes the two steps above and produces at the end a star model.

## 5 Example and Practice

In order to implement the X-ETL method, the authors take the relational model of sales in Figure 4 as the source model.

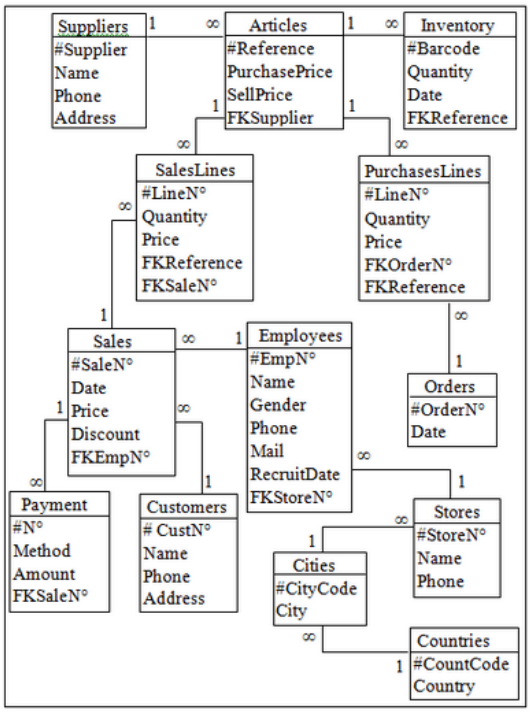


**Fig. 4.** Relational model of sales

According to the X-ETL method, candidate fact tables are selected on the basis of the number of foreign keys contained in every table. So, the result will be the 3 tables "Sales Lines", "Purchases Lines" and "Sales" since each of them contains the highest number of foreign keys in the source model (3 foreign keys). For each candidate fact table, The X-ETL engine will generate a multidimensional model by identifying, in a

first step, the dimensions connected directly and then, in a second step, those connected indirectly.

For example, if the "Sales Lines" is the fact table, the program will retain "Articles" and "Sales" as dimensions connected directly since these tables are connected to the fact table by a one-to-many relationship in the source model. Thus, the first multidimensional model will include the fact table and two dimensions.

Once this step is complete, the X-ETL engine will look for dimensions connected indirectly to the fact table by using the transitivity principle and retaining only many-to-one relationships.

The table 1 summarizes the calculation of the cardinalities and shows the dimensions connected indirectly that will be retained in the example.

**Table 1.** Identification of dimensions connected indirectly to the fact table "SalesLines"

| T1 | T2 | Cardinality | Indirect Dimension table |
|---|---|---|---|
| SalesLines | Articles | (∞-1) Direct | -- |
| Articles | Suppliers | (∞-1) Direct | -- |
| SalesLines | Suppliers | (∞-1) Indirect 1 | Suppliers |
| Articles | Inventory | (1-∞) Direct | -- |
| SalesLines | Inventory | (∞-∞) Indirect 1 | Not selected |
| Articles | PurchasesLines | (1-∞) Direct | -- |
| SalesLines | PurchasesLines | (∞-∞) Indirect 1 | Not selected |
| SalesLines | Sales | (∞-1) Direct | -- |
| Sales | Payment | (1-∞) Direct | -- |
| SalesLines | Payment | (∞-∞) Indirect 1 | Not selected |
| Sales | Customers | (∞-1) Direct | -- |
| SalesLines | Customers | (∞-1) Indirect 1 | Customers |
| Sales | Employees | (∞-1) Direct | -- |
| SalesLines | Employees | (∞-1) Indirect 1 | Employees |
| Employees | Stores | (∞-1) Direct | -- |
| SalesLines | Stores | (∞-1) Indirect 2 | Stores |
| Stores | Cities | (∞-1) Direct | -- |
| SalesLines | Cities | (∞-1) Indirect 3 | Cities |
| Cities | Countries | (∞-1) Direct | -- |
| SalesLines | Countries | (∞-1) Indirect 4 | Countries |
| SalesLines | Articles | (∞-1) Direct | -- |
| Articles | Suppliers | (∞-1) Direct | -- |
| SalesLines | Suppliers | (∞-1) Indirect 1 | Suppliers |

Since the source model contains no time table, the program will create one as a dimension in the multidimensional model. Therefore, we will obtain as final result a multidimensional model which contains the fact table "Sales Lines" and 9 dimensions: Articles-Sales-Suppliers-Customers-Employees-Stores-Cities-Countries-Time.

The star model in Figure 5 represents the final result for the fact table "Sales Lines":

Subsequently, we can introduce some modifications on the final model for further improvement and adapt it to the needs, for example grouping "Stores", "Cities" and

"Countries" in a single dimension table that will be called "Place". Therefore, stores, cities and countries will be the different granularities of the dimension "Place".

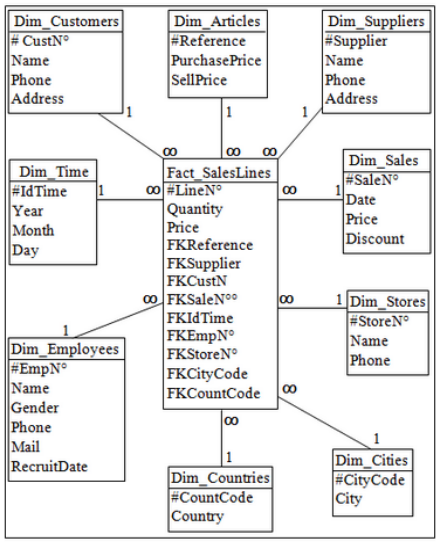


**Fig. 5.** The star model for the fact « Sales Lines »

The figure 6 is a screenshot of the X-ETL program:

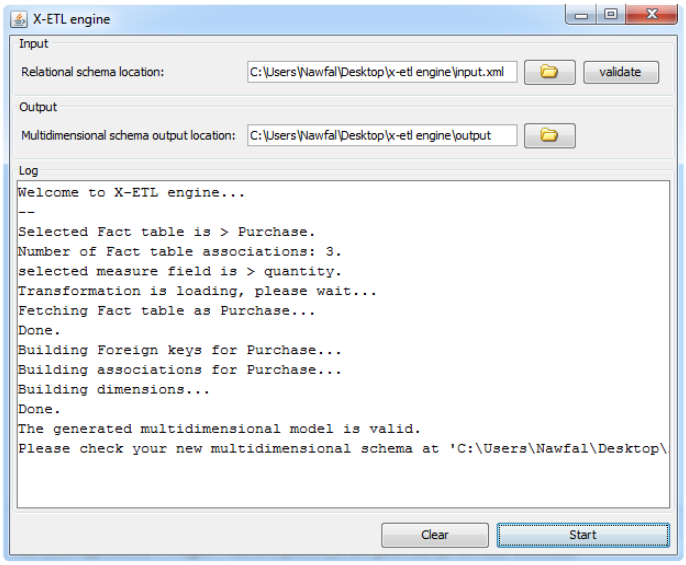


**Fig. 6.** The X-ETL program

## 6 Evaluation and Validation

In order to evaluate the X-ETL method, the authors applied it on the "hospital" example used by Golfarelli et al. [7] then they compared the results of the two methods. The figures 7 and 8 represent respectively the example used and its multidimensional model.

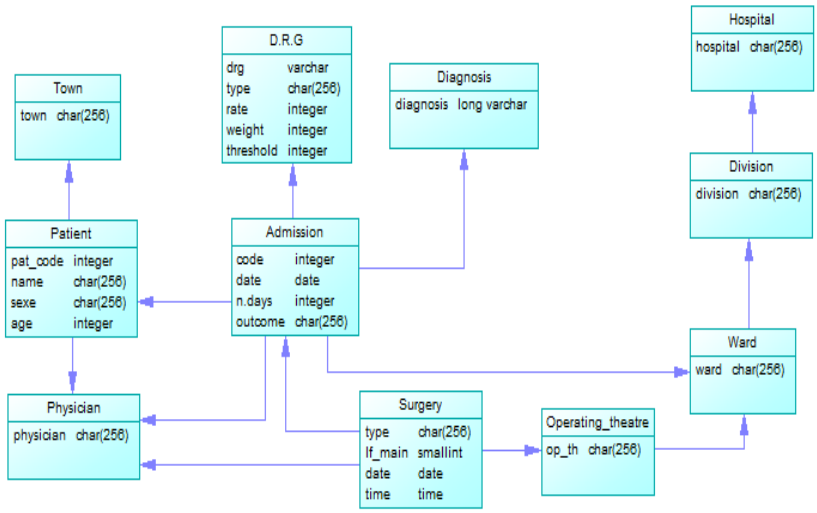


**Fig. 7.** The physical model of the "Hospital" example

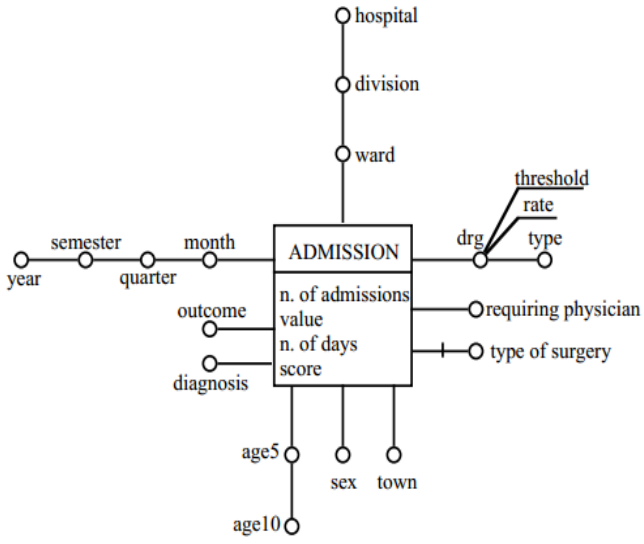


**Fig. 8.** The multidimensional model of the "Hospital" example

By applying the X-ETL method on the example of Figure 7, the following multidimensional model will be generated:

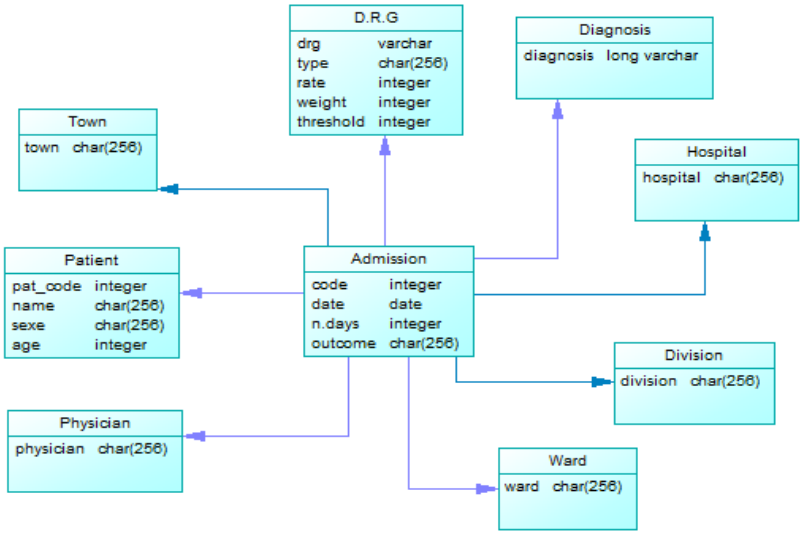


**Fig. 9.** The multidimensional model of the "hospital" example by using the X-ETL method

In order to compare the two results and measure the degree of correspondence between the model generated by the X-ETL method and the exemplary multidimensional model, the authors defined a percentage value for each multidimensional element.

There is no doubt that the fact tables are the most important multidimensional element. It's about central tables that group all the essential axes of analysis and their choice largely influences the choice of the other multidimensional elements. Therefore, selecting the right fact table constitute half success of the multidimensional model that will be generated. For each correct fact table selected the authors assign the value 50%. In some cases new information and measures can be solicited and since these data are available in the fact table and can be extracted from it, the authors decide to not calculate the percentages for the measures.

Regarding the dimension tables their number is undefined in a multidimensional model. Therefore, and to properly calculate the final result the authors took the remaining percentage which is 50% and divided it by the total number of the dimensions generated in the exemplary model to determine the percentage that represent each right dimension selected. If the exemplary model contains only the attribute hierarchy, the 50% will be divided by the total number of these attributes.

Thus, in the example 50% is assigned for choosing the correct fact table "Admission". Concerning the dimensions of the exemplary model which is in the form of a tree, there are 16 dimension attributes. So, the value of 2.78% is assigned for any similar attribute selected. The correctly selected tables are: Ward-Division-Hospital-

Diagnosis-Drg-Type-Rate-Threshold-Sex-Town-Age-Physician and 6 attributes were not selected. Therefore, the model generated by the X-ETL method is correct at:

50 + (2.78*12) = 83, 36%

## 7 Discussion and Criticism

The X-ETL method tries to automate as much as possible the design process of a multidimensional model from a relational model. However, this method has the following disadvantages:

- If the relational model is poorly designed, we will get a bad result or a result that doesn't make sense;
- It could generate fact tables that don't have priority;
- It could generate dimension tables that have no added value for the multidimensional model;
- The X-ETL method doesn't create hierarchies in the dimensions;
- Measures are selected manually in the fact table;
- The X-ETL method does not allow us to merge two or more different fields in order to get a single measure.

## 8 Conclusion

In this paper, the authors presented a new data-driven method for designing a multidimensional model from a relational model. This method is mainly based on a list of rules to identify the different elements of the multidimensional schema and consists of two steps. The first one aims to identify fact tables and dimensions connected directly, and the second one allows identifying dimensions connected indirectly. At the end, several multidimensional models will be generated in an automatic way. At this stage, it should be noted that the quality of these generated models depends greatly on the quality of the source model, and therefore it is very important to verify the relational source model before using the X-ETL method.

However, and like any data-driven approach, the X-ETL method has some disadvantages, among which are the risk of selecting inadequate tables and the limitations of the method in merging different tables to get a single dimension or a single fact table. These disadvantages are mainly due to the approach followed, which is limited to analyzing the data and excludes the users' and decision makers' needs. Therefore, the authors will try in the next work to remedy these disadvantages by integrating the requirements-driven approach in the X-ETL method.

## 9 References

[1] Bhagat, V., Gopal, A. (2012). Comparative Study of Row and Column Oriented Database. Fifth International Conference on Emerging Trends in Engineering and Technol-

ogy, November 5-7 2012, Himeji, Japan. pp. 196-201. https://doi.org/10.1109/ICETET.2012.56

[2] Golfarelli, M. (2005). New trends in Business Intelligence. Proceedings of the 28th International Convention MIPRO, 2005, Opatija, Croatia. pp. 15-20.

[3] Margaria, T. (2014). Leveraging Applications of Formal Methods, Verification and Validation: Specialized Techniques and Applications. 6th International Symposium on Leveraging Applications of Formal Methods, October 8-11 2014, Corfu, Greece. https://doi.org/10.1007/978-3-662-45231-8

[4] Adeel Shiraz, H., Tanvir, A. (2016). Big Data Mining: Tools & Algorithms. International Journal of Recent Contributions from Engineering, Science & IT (iJES), 4(1): 36-40. Retrieved from http://www.online-journals.org/index.php/i-jes/article/view/5350

[5] Song, I. Y., Khare, R., Dai, B. (2007). SAMSTAR: a semi-automated lexical method for generating star schemas from an entity-relationship diagram. Proceedings of the ACM tenth international workshop on Data warehousing and OLAP, November 09 2007, Lisbon, Portugal. pp. 9-16. https://doi.org/10.1145/1317331.1317334

[6] Romero, O., Abelló, A. (2011). A comprehensive framework on multidimensional modeling. 30th International Conference on Advances in Conceptual Modeling: Recent Developments and New Directions, October-November 31-03 2011, Brussels, Belgium. pp. 108-117.

[7] Golfarelli, M., Rizzi, S. (2009). Data Warehouse Design: Modern Principles and Methodologies. McGraw-Hill Education.

[8] Golfarelli, M., Maio, D., Rizzi, S. (1998). Conceptual design of data warehouses from E/R schemes. Proceedings of the Thirty-First Hawaii International Conference on System Sciences, January 9 2018, Hawaii, USA. pp. 334–343. https://doi.org/10.1109/HICSS.1998.649228

[9] Moody, D. L., Kortink, M. A. R. (2000). From enterprise models to dimensional models: a methodology for data warehouse and data mart design. Proceedings of the Second International Workshop on Design and Management of Data Warehouses, June 5-6 2000, Stockholm, Sweden. p. 5.

[10] Vrdoljak, B., Banek, M., Rizzi, S. (2003). Designing web warehouses from XML schemas. Fifth International Conference on Data Warehousing and Knowledge Discovery, September 3-5 2003, Prague, Czech Republic. pp. 89-98. https://doi.org/10.1007/978-3-540-45228-7_10

[11] Jensen, M. R., Holmgren, T., Pedersen, T. B. (2004). Discovering Multidimensional Structure in Relational Data. 6th International Conference on Data Warehousing and Knowledge Discovery, September 1-3 2004, Zaragoza, Spain. pp. 138–148. https://doi.org/10.1007/978-3-540-30076-2_14

[12] Romero, O., Abelló, A. (2010). A framework for multidimensional design of data warehouses from ontologies. Data & Knowledge Engineering, 69(11): 1138-1157. https://doi.org/10.1016/j.datak.2010.07.007

[13] Taniar, D., Chen, L. (Eds.). (2011). Integrations of Data Warehousing, Data Mining and Database Technologies: Innovative Approaches. IGI Global. Retrieved from http://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/978-1-60960-537-7 https://doi.org/10.4018/978-1-60960-537-7

[14] Chandwani, G., Uppal, V. (2015). Implementation of Star Schemas from ER Model. International Journal of Database Theory and Application, 8(3): 111-130. https://doi.org/10.14257/ijdta.2015.8.3.10

[15] Akbar, K., Krishna, S. M., Reddy, T. V. S. (2013). ETL process modeling in DWH using enhanced quality techniques. International Journal of Database Theory & Application, 6(4): 179-197.
[16] Bliujute, R., Saltenis, S., Slivinskas, G., Jensen, C. S. (1998). Systematic change management in dimensional data warehousing. Third International Baltic Workshop on Databases and Information Systems, 1998. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.30.2205&rep=rep1&type=pdf
[17] Rudra, A., Nimmagadda, S. L. (2005). Roles of multidimensionality and granularity in warehousing Australian resources data. 38th Annual Hawaii International Conference on System Sciences, January 6 2005, Hawaii, USA. p. 216b. https://doi.org/10.1109/HICSS.2005.527
[18] Betari, O., Filali, S., Azzaoui, A., Boubnad, M.A. (2018). Applying a Model Driven Architecture Approach: Transforming CIM to PIM Using UML. International Journal of Online Engineering (iJOE) 14(9): 170-181. Retrieved from http://www.online-journals.org/index.php/i-joe/article/view/9137 https://doi.org/10.3991/ijoe.v14i09.9137
[19] Dkaich, R., El Azami, I., Mouloudi, A. (2017). XML OLAP Cube in the Cloud Towards the DWaaS. International Journal of Cloud Applications and Computing (IJCAC), 7(1): 47-56. https://doi.org/10.4018/IJCAC.2017010103
[20] Kraleva, R.S., Kralev, V.S., Sinyagina, N., Koprinkova-Hristova, P., Bocheva, N. (2018). Design and Analysis of a Relational Database for Behavioral Experiments Data Processing. International Journal of Online Engineering (iJOE), 14(2): 117-132. Retrieved from http://www.online-journals.org/index.php/i-joe/article/view/7988 https://doi.org/10.3991/ijoe.v14i02.7988
[21] El Moukhi, N., El Azami, I., Mouloudi, A. (2018). Towards a new method for designing multidimensional models. International Journal of Business Information Systems, 7(1): 47-56. https://doi.org/10.1504/IJBIS.2018.091161

## 10 Authors

**Nawfal El Moukhi** was born in Salé in 1987. He graduated from the School of Information Science of Rabat and he got a Master‟s degree in applied computer science from Mohammed V University in 2013. Currently, he is a phd student at Ibn Tofail University in kenitra (Morocco). His research interests are Data Warehousing, Data Mining and Big Data.

**Ikram El Azami** received his PhD degree from the University of Valenciennes and Hainaux Combrésis, France, and The University of Sidi Mohammed Ben Abdellah, Fez, Morocco. He is currently full professor in the Department of Computer Science at the Faculty of Sciences, Ibn Tofail University of Kenitra. He is a member of the MIS (Multimedia and Information Systems) research Team of MISC Laboratory (Communication Systems and Information Modeling), Faculty of Sciences of Ibn Tofail University. Dr. EL AZAMI's research interests are in the area of Information Systems Engineering, Business Intelligence, Big Data and Cloud Computing.

**Abdelaaziz Mouloudi** is a professor and a manager of MISC Laboratory at the Faculty of Sciences of Kenitra. He has a work experience of many years in teaching and research in computer science. His areas of interest include artificial neural network, artificial intelligence, computer security, and reliability.

**Abdelali Elmounadi** was born in Rabat, in 1988. He is a PhD student at Mohammed V University in Rabat, Morocco. He received a Master's degree in Computer Science from Sidi Mohammed Benabdallah University in 2012. His research interests are Software Engineering and Data Science.

Article submitted 2018-12-15. Resubmitted 2019-02-06. Final acceptance 2019-02-27. Final version published as submitted by the authors.