Galaxy-Gen: A Tool for Building Galaxy model from XML documents

Ines Ben Messaoud[1], Jamel Feki[1] and Gilles Zurfluh[2]
[1]*Laboratory Mir@cl, University of Sfax, Sfax, Tunisia*
[2] *Laboratory IRIT, University of Toulouse 1, Toulouse, France*
{Ines.benmessaoud ; Jamel.feki}@fsegs.Rnu.tn,Zurfluh@univ-tlse1.fr

Keywords: Document warehouse, XML document, Multidimensional modeling, Galaxy Model.

Abstract: A galaxy model is a multidimensional model dedicated for XML document warehouses. It can be seen as a network of entities (*i.e.,* dimensions) connected via nodes. After giving an overview of our four-steps semi-automated method for the generation of galaxy models which aims to build data marts from XML documents. This paper focuses on the software tool, called *Galaxy-Gen* that implements the proposed method. We illustrate the *Galaxy-Gen* functionalities and make its first assessment through two experiments. The first experiment is applied to a set of twenty XML documents taken from the academic domain. The second one addressed a set of 1691 XML documents issued from the Clef-2007 collection. The assessment is performed by comparing manual design galaxy models with those produced by the *Galaxy-Gen* tool. The results are very promising.

1 INTRODUCTION

The organization's documents help decision makers to understand how corporate data evolve over time. Thereby, these documents represent an important volume that should be incorporated into the decision support system. However, so far, decisional analyses are based on multidimensional databases which mainly store *numeric* business *indicators* issued from OLTP (On-Line Transaction Processing) systems. In practice, these numeric data represent less than the quarter of the whole volume of data that could be useful for decision makers. Time is coming to focus on non-numeric data stored in documents; these data are important for the decision making process. Consequently, during this process some relevant documents may be ignored while some non pertinent documents can be considered by intuition. The final result can be defective since the decision is based on incomplete information. Consequently, documents should be integrated into the decision support system (Tseng and Chou, 2006). In other terms, as advocated by the authors of (McCabe and al., 2000) and (Sullivan, 2001), these documents should be warehoused. Thus, the document warehouse (DocW) has emerged; it is defined as a collection of documents issued from internal and external data sources. Its main objective is to organize documents for effective analysis or feature extraction to enable distilled and fruitful business intelligence (Tseng and Chou, 2006).

In practice, there are several formats of documents such as XML format (eXtensible Markup Language) which allows the exchange of a wide variety of data on the Web. More accurately, there are two types of XML documents: *data-centric* and *document-centric* XML documents (Fuhr and al, 2001) (Kamps and Marx, 2004). Data-centric documents contain structured data (*e.g.*, order, invoice) as data strored or issued from databases. While, the document-centric XML documents are text-rich and then less structured (*e.g.*, scientific articles, company reports). Furthermore, an XML document is generally compliant to a generic grammar called *DTD* "*Document Type Definition*" or *XSD* "*XML Schema Definition*". In our work, we are interested in XML document-centric documents. For this latter, there are two categories of approaches for document warehousing: *contextualization of the data warehouse with XML documents* (Pérez and al., 2008), and *construction of data mart from the metadata of documents* (Krouf, 2004) (Tseng and Chou, 2006).

In general, even if they belong to the same domain, XML documents may have different structures. Consequently, a step to unify these structures is required in order to produce a global

view describing a large document set. To use this global view in their decisional processes, decision makers need a multidimensional model. Therefore, the multidimensional modeling of documents is compulsory. In addition, it provides the user with operators of the multidimensional algebra; thus, it inhibits him to write complicated queries (e.g., using XQuery). To alleviate these difficulties, we presented in (Feki and al, 2013) an approach to build a DocW; this approach is made up of two methods: (i) *Unification of XML document structures* (Ben Messaoud and al, 2011a) (Ben Messaoud and al, 2012), and (ii) *Multidimensional modeling of documents* (Ben Messaoud and al., 2011b) (Feki and al., 2013). In this paper, we focus on the second method that produces a multidimensional galaxy model for the XML DocW. More precisely, we tackle the experiments and evaluation of this method on academic and medical collections, through our developed software tool called *Galaxy-Gen* (***G****alaxy* ***G****eneration*).

This paper is organized as follows: In Section 2, we discuss related works that treat multidimensional modeling of documents. In Section 3, we give an overview of our approach for building the schema of the XML document warehouse. Then, our multidimensional modeling method is described in Section 4; while Section 5 shows the functionalities of the *Galaxy-Gen* software tool. Finally, in Section 6, we conclude the paper and address future works.

2 RELATED WORKS

Let us remember that the multidimensional modeling aims to design multidimensional models that support OLAP (On-line Analytical Processing) analyses.

In the literature, there are two types of works addressing the multidimensional modeling of XML documents: works related data-centric XML documents (Hümmer and al., 2003) (Boussaid and al., 2006) (Hachaichi and al., 2010), and others related to document-centric XML documents. The remaining of this paper concerns document-centric XML documents. Firstly we present the most relevant works related to that area where some researchers have proposed methods to model the DocW as a star model. As examples of these works we can cite (McCabe and al., 2000), (Krouf, 2004), (Tseng and Chou, 2006) and (Ravat and al., 2007). Other works such as (Tournier, 2007) and (Pujolle and al., 2011) propose the galaxy model. Secondly, we compare the literature works according to a set of criteria we have specifically established for this comparison.

Figure 1 presents an example of a star model designed for the analysis of sales. It is composed of a central fact called "*Sales*" composed of two indicators (*i.e.,* measures) namely *Quantity* and *Amounts*. These measures could be analyzed (i.e., aggregated using Sum, Avg…functions) according to the three axes: *Retail_Outlet*, *Date* and *Product*. For example, with this star model we can analyze sales amounts per product and year.

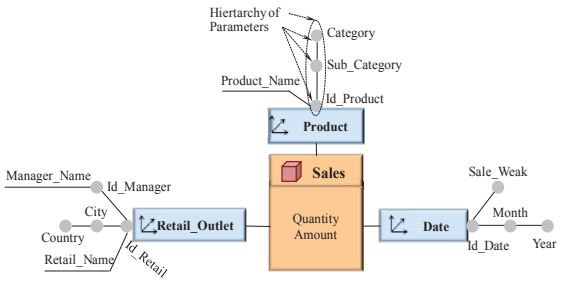


Figure 1: An example of Star model for analyzing Sales.

The authors of (McCabe and al., 2000) suggest a retrieval method in text collections; to do so, they model the global view of the documents set as a star model. In their star multidimensional model, they distinguish five types for the *dimension* concept namely: *Localization*, *Time*, *Term*, *Document* and *Category*. The *measure* concept is the number of each term occurrences within documents.

In (Krouf, 2004), a process to analyze documents of the DocW was proposed. This process relies on this star model. First, the decision maker indicates the analysis components: fact, dimensions and an aggregate function. Secondly, a document mart is generated and instantiated. Finally, the result is displayed as a multidimensional table. Nevertheless, during the DocW design phase the determination of multidimensional elements is manual. Indeed, the authors do not propose rules or algorithms to identify fact and dimensions.

As far as, authors of (Tseng and Chou, 2006) elect the star model in order to analyze documents. Their star model distinguishes three types of dimensions: *Ordinary* dimension containing keywords extracted from the document, *Metadata* dimension which describes the document with *title*, *author*, etc., and *Category* dimension that contains keywords external to the document; *i.e.*, issued from *Wordnet*. The result star model enables counting the number of documents according to these dimensions.

The result star model of (McCabe and al, 2000), (Khrouf, 2004) and (Tseng and Chou, 2006) perform only quantitative analyses because their measures are numeric. Moreover, analyses are limited since the analyses subject (*i.e.*, the fact) is defined a priori, at the design time of the star model but not at the query time.

The authors of (Ravat and al., 2007) propose to revise the constellation modeling for documents; they suggest adding a new *textual measure* and two new dimensions called *Structure* and *Complementary*. In fact, a *textual measure* can be a word, a paragraph or a whole document. The *Structure dimension* describes the structure of documents whereas the *Complementary dimension* is determined from complementary data sources (*e.g.,* data from the curriculum vitae of authors). Nevertheless, the authors did not propose rules or algorithms to assist the DocW designer elaborating the constellation schema: identification of facts, dimensions, hierarchies...

In (Tournier, 2007) and (Pujolle and al., 2011), the authors propose a hybrid design process to build a document warehouse from document-centric XML documents. Their process combines a top-down approach (*i.e.*, starting from user requirements) and a bottom-up approach (*i.e.*, relying on the source data model). In addition, they suggest a new multidimensional conceptual model called *Galaxy*. This model can be defined as a set of entities, where each entity is presented like a dimension; several dimensions could be linked by a node and then are said compatible dimensions for analyses. However, the main drawback of this work is that the authors do not define rules to assist the design phase of a galaxy model.

In order to summarize and highlight the pros and cons of the literature approaches, we compare them in Table 1, among the following set of six criteria:

C1: The approach is specific for XML document-centric document.

C2: The approach uses constellation model.

C3: The approach uses galaxy model.

C4: The approach determines multidimensional concepts manually.

C5: The approach determines multidimensional concepts semi-automatically.

C6: The approach determines multidimensional concepts automatically.

Table 1: Comparison of multidimensional modeling approaches for documents

Approach \ Criterion	C1	C2	C3	C4	C5	C6
(McCabe and al., 2000)	✓	✓	N	-	-	-
(Tseng and Chou, 2006)	✓	✓	N	-	-	-
(Krouf, 2004)	✓	✓	N	✓	N	N
(Ravat and al., 2007)	✓	✓	N	✓	N	N
(Tournier, 2007) & (Pujolle and al., 2011)	✓	N	✓	✓	N	N

✓ : Criterion supported by the approach. N: Criterion not supported by the approach. -: Not indicated by the authors.

In this section, we have presented pertinent works related to multidimensional modeling of documents. We have focused on *star* and *galaxy models*. We stress that a star model is characterized by a predefined subject of analyses (*i.e.*, fact). Whereas, within a galaxy model the fact is not predefined; it will be specified when querying the galaxy. Consequently, a galaxy model is simpler than the star model because it is based on a unique concept: *Dimension*. Furthermore, analyses expressed on a galaxy model are more flexible than analyses expressed on a star model. Considering these benefits, we have elected the galaxy model for modeling the document warehouse.

The remaining of this paper overviews our approach for building the schema of the document warehouse (*cf.*, Section 3) and then our method of multidimensional modeling of documents (*cf.*, Section 4). Experiments and evaluation are subjects of Section 5.

3 PROPOSED APPROACH FOR BUILDING XML DOCUMENT WAREHOUSE

Generally, XML documents are described by heterogeneous structures even though they belong to a same domain. Thus, when a decision maker needs to query these documents, he is constrained to write several queries (*i.e.,* as many queries as the number of different structures in the document set). To tone down this problem, we expect to provide a global view of the document set. To achieve this global view, we have proposed in (Feki and al., 2013) an approach to elaborate the schema of the DocW. This approach is composed of two methods: *Unification of XML documents structures*, and *Multidimensional modeling of documents*. Figure 2 depicts this approach. Here is a short overview of these two

methods as they are required for the readability of the remaining of this paper.

Unification of XML documents structures. This method receives as input a set of XML structures belonging to the same domain and then produces a limited number of unified trees validated by the decision makers (Ben Messaoud and al., 2011a) (Ben Messaoud and al., 2012). It consists of the four main steps namely: a) *Tree representation*, b) *Generation of unified trees*, c) *Approval of unified trees*, and d) *Correctness verification of trees*.

Firstly, *Tree representation* translates XML structures into trees by applying two rules. We choose the formalism of tree, as adopted in (Lee and al., 2002) and (Yoo and al., 2005), since it is graphical and easy to be understood by unskilled persons.

Secondly, *Generation of unified trees* step produces a limited number of *unified trees*. It treats both acronym and synonym ambiguities of tree nodes, referring to a dictionary of acronyms and the lexical database *Wordnet*. Then, it computes a triangular *Similarity Matrix* (*SM*) which has *n* trees in rows and in columns. It facilitates the identification of trees to be merged. Each pair of trees having their similarity factor higher than a given threshold (experimentally determined) is merged applying fusion-operators developed in (Ben Messaoud and al., 2011a) (*Fusion by inclusion*, *Fusion by union of sub-trees, or Fusion by merging nodes*).

After that, the *Approval of unified trees* validates trees according to the analytical requirements of decision makers. In fact, they can delete and/or rename nodes. Finally, the *Correctness verification of trees* checks the syntactic validity of trees among a set of four constraints called: *Connected nodes*, *Hierarchy*, *Uniqueness of the root node* and *Acyclicity* (Aouabed and al., 2012).

Note that the input structures and the output unified trees are saved into the repository shown in Figure 3.a.

Multidimensional modeling. This method accepts the structure of the input XML documents. It can be either unified tree resulting from the previous method (*i.e.,* Unification of XML documents structures) or XML structure and then produces galaxy model. The output galaxy model is saved according to the meta-model of Figure 3.b.

The following sections detail the *semi-automatic multidimensional modeling* method, and then presents our software prototype called *Galaxy-Gen* (*Galaxy Generation*) that supports this method.

4 MULTIDIMENSIONAL MODELING OF DOCUMENTS

Our method of multidimensional modeling of documents aims to generate semi-automatically a multidimensional model for the DocW; among the existing multidimensional models, we have elected the *Galaxy* model (Tournier, 2007). For readability reasons of the paper, we first introduce the galaxy and then our proposed method.

The galaxy model can be seen as a network of entities (*i.e.,* dimensions) connected by nodes (*cf.* Figure 10). Each node denotes compatible entities which could be used together in OLAP analytical queries. In a galaxy, each entity can play a double role: an analysis subject (*i.e.,* fact) or an analysis axis (*i.e.,* dimension). As with the star model (Golfarelli, 1998), an entity is composed of one or more attributes hierarchically organized.

To generate such a galaxy model, we proposed in (Ben Messaoud and al., 2011b) and (Feki and al., 2013) a semi-automatic method composed of the four following steps:

- *Pretreatment of trees*,
- *Building galaxy models*,
- *Galaxy models approval*, and
- *Correctness verification of galaxy models.*

Figure 4 exemplifies the sequencing of these steps.

4.1 Pretreatment of trees

This step receives as input a tree that represents either the structure of a set of XML documents or the unified tree resulting from the unification of XML documents and then produces a pretreated tree. In fact, a pretreated tree has cardinalities; they are added by exploring XML documents compliant to the input structure(s).

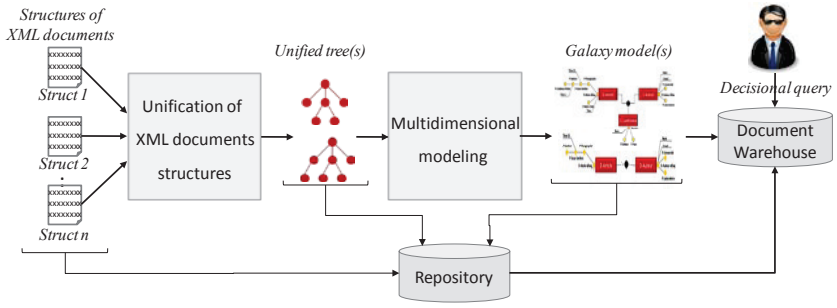


Figure 2: Approach for building the schema of the XML Document Warehouse.

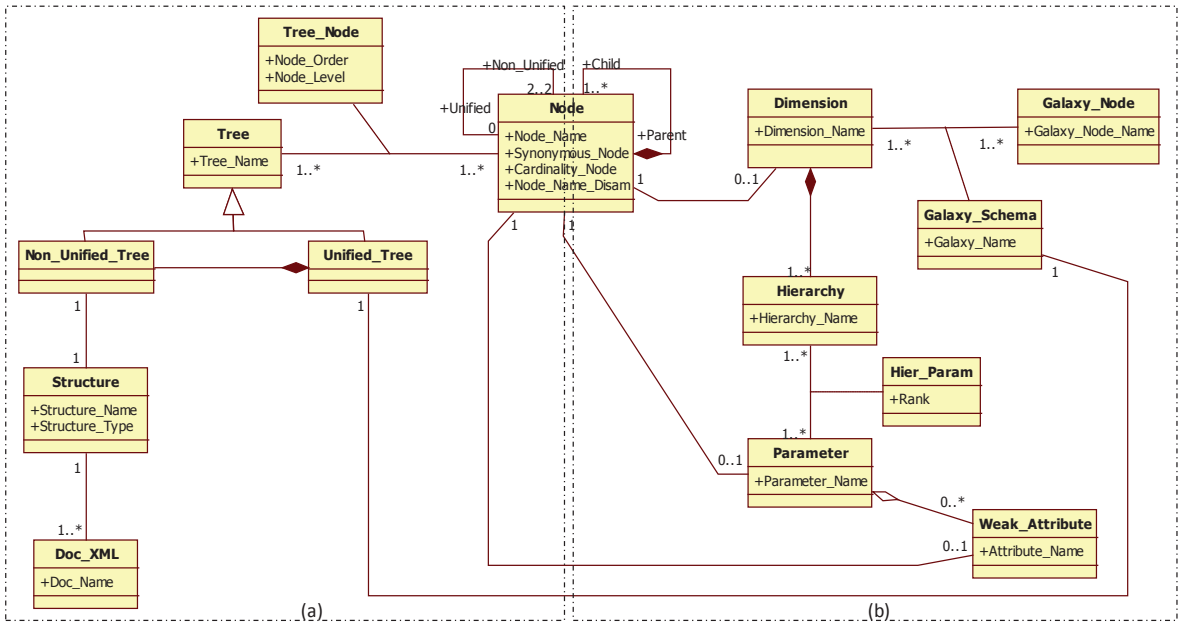


Figure 3: Meta-model of the Document Warehouse

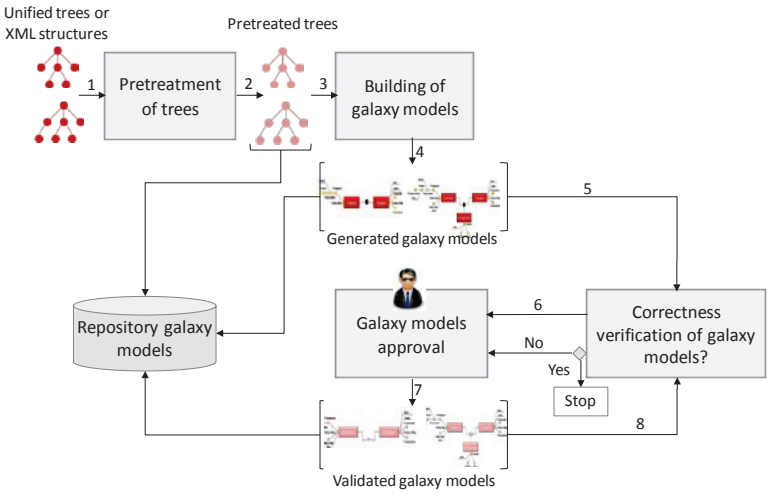


Figure 4: Galaxy model modeling steps.

4.2 Building galaxy models

The building galaxy step translates each pretreated tree into a galaxy model by applying a set of ten rules detailed in (Ben Messaoud and al., 2011b). It consists of the two sub-steps: *Identification of dimensions (i.e., entities) and galaxy-nodes,* and *Identification of hierarchies.*

Identification of dimensions and galaxy-nodes. This identification applies three rules to determine dimensions, and one rule for nodes. For clarity reasons, we give these rules (*cf.* (Ben Messaoud and al., 2011b) for further details).

Dimensions identification rules

Rd1: The root node *r* of a pretreated tree constitutes a dimension called *D-r*.

Rd2: Every pair of non terminal nodes *M* and *N* related through an arc *M-N* with cardinality *(+ or *)-(+ or *)* transforms into two dimensions called: *D-M* and *D-N*.

Rd3: Nodes having the same parent node and describing a date (e.g., day, month, and year) denotes the existence of a temporal dimension called *D-Date* in the resulted model. These nodes are the dimension's parameters.

Galaxy-nodes identification rule

Rn1: Each pair of nodes *M* and *N* identified as dimensions and related by an arc *M-N* in the pretreated tree constitutes two dimensions connected via a node (two compatibles dimensions).

Hierarchy identification. In a multidimensional model, dimensions are composed of attributes; some of them are organized into hierarchies that represent analyses perspectives (Ravat and Teste, 2000). Hierarchical attributes are said parameters. Within a dimensional hierarchy, the lowest granularity is the identifier of the dimension.

In our work, this identifier is a surrogate key (artificial attribute which values are sequentially generated). Parameters beyond the identifier are extracted using four rules (*Rp1*, *Rp2*, *Rp3* and *Rp4*). Sometimes, parameters can be associated with descriptive data called weak attribute (as the author name for the author Id). We extract such attributes using two rules (*Rw1* and *Rw2*).

Parameters identification rules

Rp1: Every terminal node *N* linked to a parent node *M* identified as a dimension where the arc *M-N* is not annotated with cardinalities *1-1*, transforms into a parameter *P-N* at level *2*.

Rp2: Each terminal node *N* linked to a parent node *M* identified as a parameter at level *i* where the arc *M-N* is not annotated with the cardinalities *1-1*, represents a parameter *P-N* at level *i-1*.

Rp3: Every non terminal node *N* linked to a parent node *M* identified as a parameter at a level *i* and related by an arc *M-N* annotated with the cardinalities *1-(+ or *)*, transforms a parameter *P-N* at level *i-1*.

Rp4: Each non terminal node *N* linked to a parent node *M* identified as a dimension and related by an arc *M-N* not annotated with the cardinalities *(+ or *)-(+ or *)*, transforms into a terminal parameter *P-N*.

Weak attributes determination rules

Rw1: Each terminal node *N* having its parent node *M* identified as a dimension *D* and the arc *M-N* is annotated with the cardinalities *(1 or 0)-1*, transforms into a weak attributes *W-N* for the identifier of *D*.

Rw2: Each terminal node *N* having its parent node *M* identified as a parameter *P* and the arc *M-N* is annotated with the cardinalities *(1 or 0)-1*, transforms into a weak attributes *W-N* for *P*.

4.3 Galaxy models approval

The galaxy model approval step displays models issued from the previous step to the decision maker for agreement. In fact, they adjust models according to their analytical requirements. This adjustment consists in deleting and/or renaming multidimensional elements (*i.e.,* dimension, parameter). All these changes are saved in the repository of Figure 3.b to be used later in querying the galaxy.

4.4 Correctness verification of galaxies

Galaxy models should be syntactically checked, for this purpose we define a set of constraints. Eight constraints are adapted from those defined for the star model in the literature. In addition, we define three specific constraints for the galaxy (Feki and al., 2013). We classify all these model constraints into three classes according to whether they apply on dimensions, nodes or hierarchies.

Dimension constraints.

Cd1: Identification constraint: Every dimension must have an identifier. It may be either a key extracted from the data source or a surrogate key (Hurtado and Mendelzon, 2002) (Carpani and Ruggia, 2001).

Cd2: Non empty dimension: Each dimension should have at least one hierarchy (Ben Abdallah and al., 2008).

Cd3: Non isolated dimension: In a galaxy model, every dimension has to be associated with *n (n ≥ 1)* nodes.

Node constraints.

Cn1: Non isolated node: Each node must connect at least to two different dimensions. This enables to perform multidimensional analyses on *n* (*n*≥2) axes.

Cn2: Disjunction of nodes: In a galaxy model, nodes are not directly linked; the only links between nodes are indirect via dimensions.

Hierarchy constraints.

Ch1: Hierarchical root: All hierarchies of a dimension *D* begin from the identifier of *D* (Ben Abdallah and al., 2008).

Ch2: Exclusive hierarchies: Any dimension having the minimal hierarchy (A minimal hierarchy *h* is restricted to two parameters: the identifier of the dimension of *h* directly linked to *All.*) must not have other hierarchies (Ben Abdallah and al., 2008).

Ch3: Non isolated attribute: Within a dimension *D*, each attribute must belong to at least one hierarchy of *D* (Ben Abdallah and al., 2008).

Ch4: Non empty Hierarchy: In a dimension *D*, a hierarchy must contain at least two parameters: the identifier of *D* and the *All* parameter (Ben Abdallah and al., 2008).

Ch5: Rollup: All the parameters of a hierarchy, excepting the *All* parameter, have at least a parent (Hurtado and Mendelzon, 2002).

Ch6: Acyclicity: Each parameter, excepting the parameter *All*, cannot be parent and child of the same parameter by transitivity (Hurtado and Mendelzon, 2002), (Ghozzi and al., 2003).

Note that among the extracted attributes for a dimension, we may rarely find attributes describing the structure and others relative to the metadata of documents. To solve this problem, we split the set of attributes into two dimensions. This split relies on the usage of the *Dublin Core Metadata Initiative* (Dublin Core Metadata Initiative: www.dublincore.org.) which facilitates the identification of the metadata attributes.

5 EXPERIMENTS AND EVALUATION

In order to substantiate our method, we have implemented a software tool named *Galaxy-Gen* for the generation of multidimensional galaxy model. This generation applies a set of rules (*cf.*, Section 4). It receives as input an XML structure or a unified tree resulting from the unification of a set of DTD and/or XSD belonging to the same domain and then produces a multidimensional *Galaxy* model.

We have carried out two experiments: one on academic documents and other using medical documents.

5.1 Experiment on academic collection

This first experiment is applied to a set of twenty XML documents taken from the academic domain and compliant to four complex DTDs (*cf.* Figure 5).

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Auth ((Name, Affiliation))>
<!ELEMENT Title (#PCDATA)>
<!ELEMENT Subsection ((Para))>
<!ELEMENT Section ((Title?, Subsection+))>
<!ELEMENT Para (#PCDATA)>
<!ELEMENT Name (#PCDATA)>
<!ELEMENT Month (#PCDATA)>
<!ELEMENT Day (#PCDATA)>
<!ELEMENT Article ((Title, Auth+, Section+,
Day, Month))>
<!ELEMENT Affiliation (#PCDATA)>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Year (#PCDATA)>
<!ELEMENT Title (#PCDATA)>
<!ELEMENT Section ((Paragraph+))>
<!ELEMENT References (#PCDATA)>
<!ELEMENT Paragraph (#PCDATA)>
<!ELEMENT Name (#PCDATA)>
<!ELEMENT Month (#PCDATA)>
<!ELEMENT Institute (#PCDATA)>
<!ELEMENT Day (#PCDATA)>
<!ELEMENT Body ((Section+))>
<!ELEMENT Writer ((Name, Institute))>
<!ELEMENT Article ((Title, Writer+, Body,
References+, Day, Month, Year))>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT University (#PCDATA)>
<!ELEMENT Year (#PCDATA)>
<!ELEMENT Title (#PCDATA)>
<!ELEMENT Subsection Number (#PCDATA)>
<!ELEMENT Subsection (Subsection_Number,
Title, Paragraph+, Fig.*, Table*)>
<!ELEMENT Section_Number (#PCDATA)>
<!ELEMENT Section (Section_Number, Title,
Paragraph+, Fig.*, Table*, Subsection*)>
<!ELEMENT Paragraph (#PCDATA)>
<!ELEMENT Outline (#PCDATA)>
<!ELEMENT Name (#PCDATA)>
<!ELEMENT Fig. (#PCDATA)>
<!ELEMENT Table (#PCDATA)>
<!ELEMENT Writer ((Name, University))>
<!ELEMENT Article ((Title, Writer+,
Outline, Section+, Year))>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Tit (#PCDATA)>
<!ELEMENT Name (#PCDATA)>
<!ELEMENT Body (#PCDATA)>
<!ELEMENT Writer ((Name, Affiliation))>
<!ELEMENT Article ((Tit, Writer+, Abstract,
Body))>
<!ELEMENT Affiliation (#PCDATA)>
<!ELEMENT Abstract (#PCDATA)>
```

Figure 5: Four DTDs from the academic domain.

In the remaining of this section, we present the features of our software tool *Galaxy-Gen* while exemplifying them through the galaxy-model generated for this first experiment.

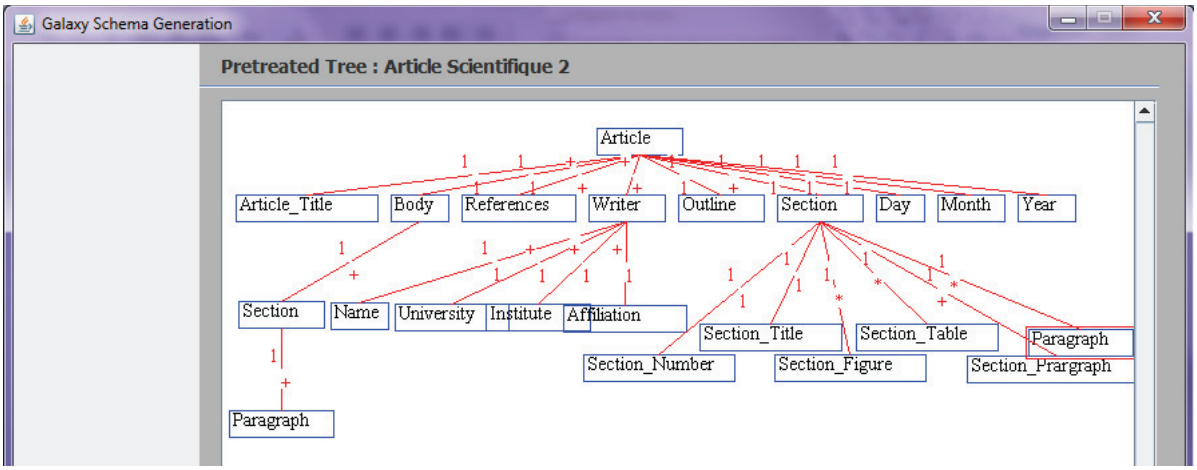
Since the input XML documents are described by heterogeneous structures, we invoke our *USD* tool (Unification of Structures of XML Documents) (Aoubed and al., 2012) in order to generate the unified tree for the four DTDs. After that, the galaxy generation process starts with picking the unified tree(s) for which the user wants to get a Galaxy model. Secondly the pretreatment step is launched to produce the pretreated tree shown in Figure 6. In this tree, cardinalities are automatically added for each parent node (except for the root), one cardinality for each outgoing edge. These cardinalities are determined by exploring the twenty input XML documents conform to the four DTDs of the running example in this experiment. We note that the structure of this pretreated tree and the structure of the unified tree are identical.

Thirdly, dimensions of the galaxy are extracted by applying three rules (*cf.*, Section 4). Thus, four dimensions are extracted: *D-Article*, *D-References*, *D-Writer* and *D-Date*. The dimension *D-Article* is identified by applying rules *Rd1* and *Rd2*; whereas *D-References* and *D-Writer* are extracted using only *Rd2*. *D-Date* is identified by applying rule *Rd3*. Among these dimensions, we assume that the decision maker has selected three ones (*D-Article*, *D-Writer* and *D-Date*) meaning that (s)he is interested in these analyses axes. Figure 7 illustrates the extracted dimensions.

This extraction of galaxy dimensions is followed by the extraction of galaxy nodes. Indeed, compatibles dimensions are linked via a node. In our running example, there is only one node connecting the three selected dimensions (*cf.*, Figure 8).

After that, hierarchies of the selected dimensions are determined. In fact, their parameters and weak attributes are extracted by applying rules defined in Section 4. Figure 9 shows the three hierarchies called *H_Writer_1, H_Writer_2* and *H_Writer_3* of the *D-Writer* dimension. The hierarchy *H_Writer_1* has two parameters *Id_D_Writer* and *P_Affiliation*. The identifier of the dimension (*i.e.*, *Id_D_Writer*) has one weak attribute *WA_Name*. We assume that the decision maker is not interested with the name of the author; thus, (s)he has not selected this attribute.

Finally, when the decision-maker checks hierarchies with their parameters and weak attributes, the galaxy model is automatically produced. Figure 10 illustrates the obtained galaxy for our running example.



Legend of added cardinalities: 1: An element can be repeated once. +: An element can be repeated (cardinality >= 1). *: An optional element can be repeated (cardinality >= 0).

Figure 6: Pretreated tree with added cardinalities.

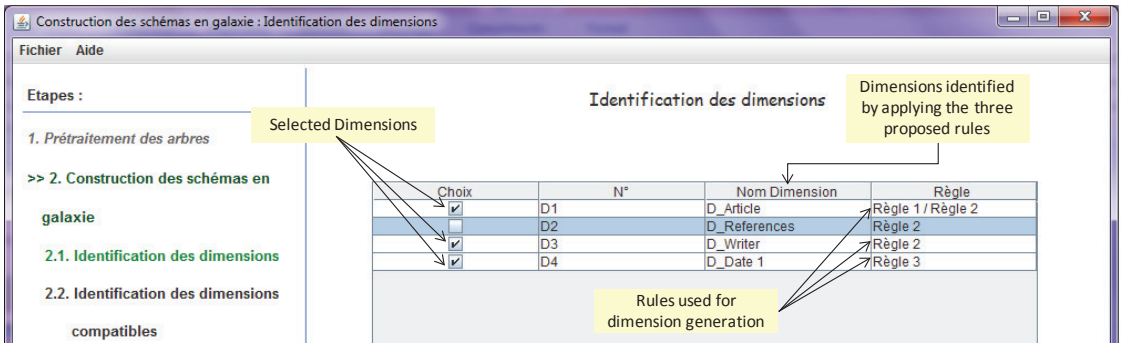


Figure 7: *Galaxy-Gen* interface for selecting identified dimensions.

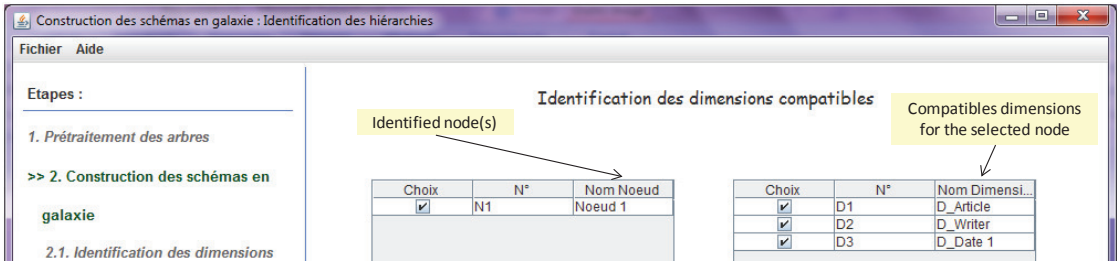


Figure 8: *Galaxy-Gen* interface for selecting compatible nodes

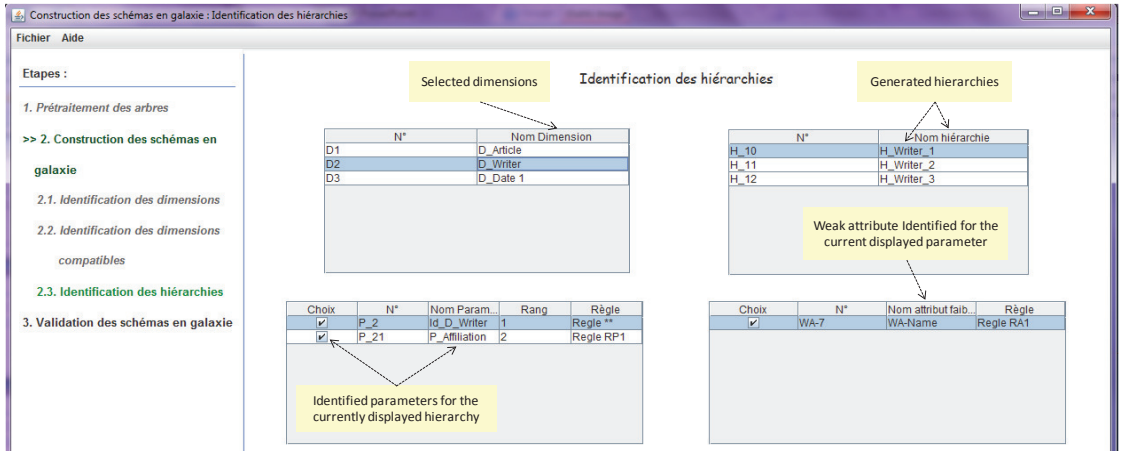


Figure 9: *Galaxy-Gen* interface for selecting hierarchies.

Compared to the galaxy model built manually on these documents, the generated galaxy model has one dimension less; whereas hierarchies are almost the same.

5.2 Experiment on medical documents

In order to better assess the *Galaxy-Gen* tool, we have conducted a second experiment; it is performed on a set of 1691 XML documents taken from the

medical collection Clef-2007. However, there were some inadequacies in these documents: for example, all *keywords* are gathered inside a unique textual tag. In order to alleviate this difficulty we have improved the DTDs of these documents (by adding the + cardinality to some elements) to obtain more accurate XML documents. Note these documents are described by three DTDs we have generated using XMLSpy; since these DTDs are long we do not include them in this paper.

These DTDs have some elements in common, and some different ones. They have the same root element linked to a set of elements that differs according to the DTD.

After processing these DTDs with the Galaxy-Gen we obtained a galaxy model composed of the following five dimensions: *D_Casimage_Case, D_Author, D_References, D_Keywords* and *D_Reviewer*.

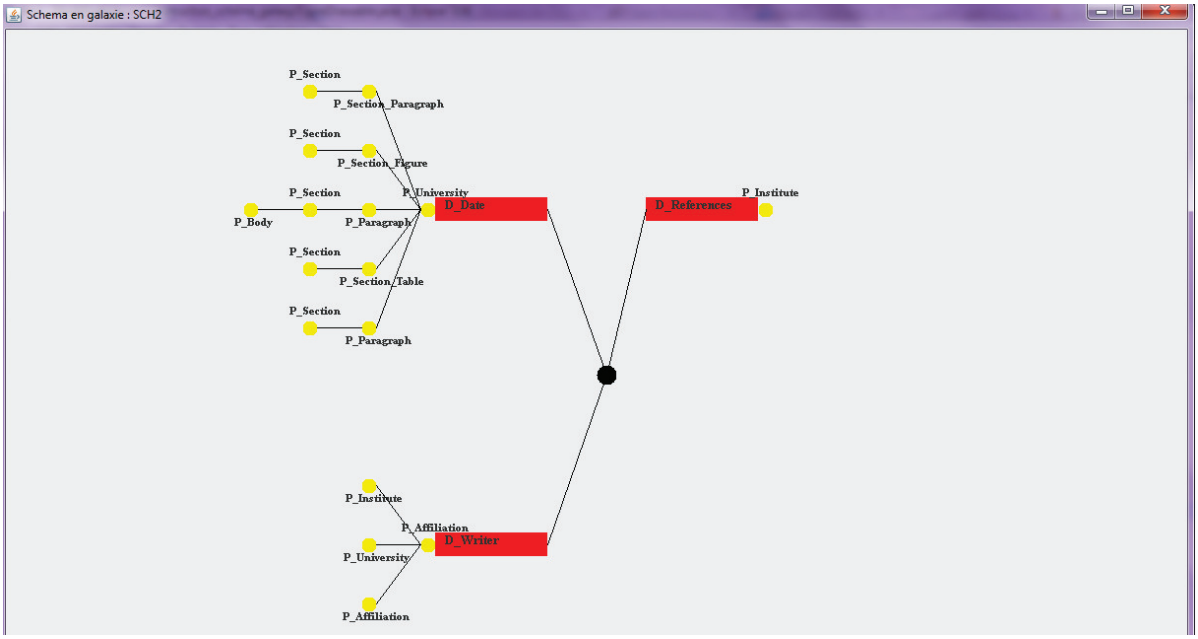


Figure 10: Result galaxy model for the first experiment.

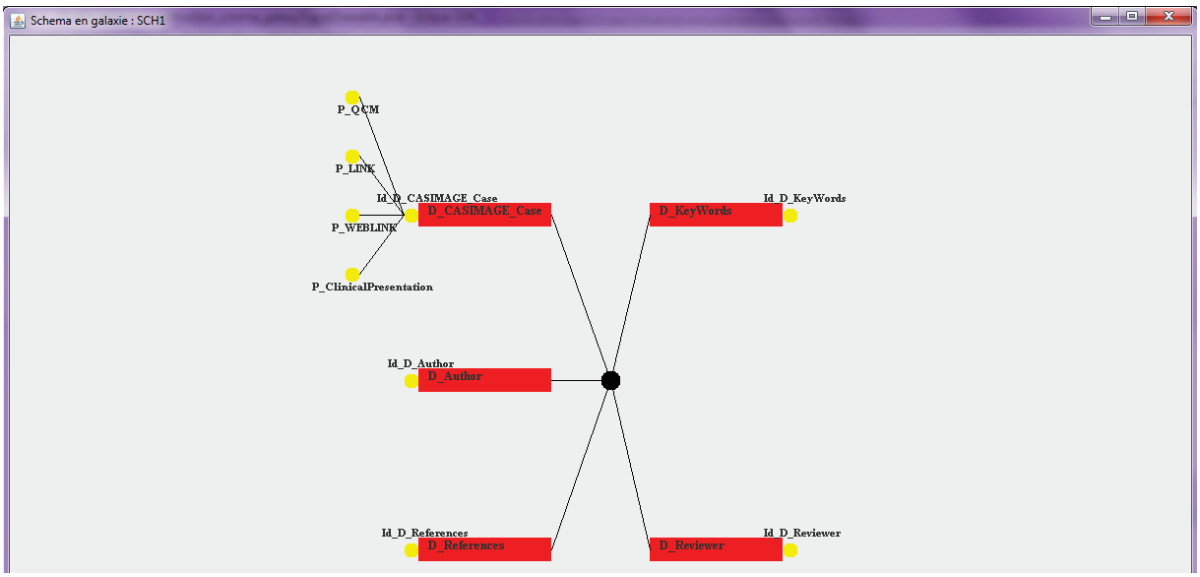


Figure 11: Result galaxy model for the second experiment.

In fact, for this second experiment, the galaxy model issued from the prototype is identical to the one built manually. This represents encouraging results.

5 CONCLUSION

Documents represent an important source for decisional analyses. They merit to be integrated in the decision support system. In this paper, our main interest was to build the schema of the document warehouse. More specifically, we gave a detailed overview of the semi-automated method for the construction of the multidimensional model for a document warehouse. We have elected the *Galaxy* model to represent this multidimensional model. Likewise, we have presented a software tool, called *Galaxy-Gen* that implements the method for the generation of galaxy models.

Furthermore, we have conducted two experiments using the Galaxy-Gen tool; they are to evaluate our proposals. The first experiment is applied on a set of XML documents taken from the academic domain. It produces a galaxy model composed of four dimensions. Whereas, the second experiment is performed on XML documents taken from the collection Clef-2007. For this experiment, we obtained a galaxy with five dimensions.

As a future work, we expect evaluate Galaxy-Gen software tool on more XML structures. Also, we are in the step of finishing the definition of a set of analytical operations dedicated to the galaxy model, and we aim implementing a query language for the galaxy based on these operations.

REFERENCES

Aouabed, H., Ben Messaoud, I., Feki, J., Zurfluh, G., 2012. USD : Un outil d'unification des structures des documents XML. *In ASD'12 Atelier des Systèmes Décisionnels*. Algeria, 83-94.

Ben Abdallah, M., Feki, J., Ben-Abdallah, H. 2008. Patrons multidimensionnels constraints. *In SIIE'08 Conférence Internationale des Systèmes d'Information et Intelligence Economique*. Tunisia, 14-16.

Ben Messaoud, I., Feki, J., Khrouf, K., Zurfluh, G., 2011a. Unification of XML document structures for Document Warehouse (DocW). *In ICEIS'11, 13th International Conference on Entreprise Information Systems*. Beijing, 85-94.

Ben Messaoud, I., Feki, J., Zurfluh, G., 2011b. Modélisation multidimensionnelle des documents XML. *In RNTI' 2011, Revue des Nouvelles Technologies de l'Information*. Vol. B-7, 55-70.

Ben Messaoud, I., Feki, J., Zurfluh, G., 2012. A First Step for Building a Document Warehouse: Unification of XML Documents. *In RCIS'12, Sixth International Conference on Research Challenges in Information Science*. Spain, 59-64.

Boussaïd, O., Ben Messaoud, R., Choquet, R., Anthoard, S., 2006. X-Warehousing: an XML-Based Approach for Warehousing Complex Data. *In ADBIS'06, 10th East-European Conference on Advances in Databases and Information Systems*. Germany LNCS, Vol. 4152, Springer, 39-54.

Carpani, F., Ruggia, R., 2001. An Integrity Constraints Language for a Conceptual Multidimensional Data Model. *In SEKE'01, 13th International Conference on Software Engineering & Knowledge Engineering*. Argentina, 220-227.

Feki, J., Ben Messaoud, I., Zurfluh, G., 2013. Building an XML Document Warehouse. *In JDS'13, Journal of Decision System*. Ed. Taylor & Francis, Vol. 22 n° 2/2013, pages 122-148, DOI: 10.1080/12460125.2013.780322

Fuhr, N., Grobjohann, Kai., 2001. XIRQL: a query language for information retrieval in XML documents. *In SIGIR'01, 24th International ACM Conference on Research and Developement in Information Retrieval*. ACM Press, 172-180.

Ghozzi, F., Ravat, F., Teste, O., Zurfluh, G., 2003. Constraints and multidimensional databases. *In ICEIS'03, 5th International Conference on Enterprise Information Systems*. France, 104-111.

Golfarelli, M., Maio, D., Rizzi, S., 1998. The dimensional fact model: a conceptual model for data warehouses. *In IJCIS'98, International Journal of Cooperative Information Systems*. 215-247.

Hachaichi, Y., Feki, J., Ben-Abdallah, H., 2010. Modélisation multidimensionnelle de documents XML centrés-données. *In JDS'10, Journal of Decision Systems,* 313-345.

Hümmer, W., Bauer, A., Harde, G., 2003. XCube—XML for Data Warehouses. *In DOLAP'03, Proc. Sixth ACM Int'l Workshop Data Warehousing and OLAP*, 33-40.

Hurtado, C. A., Mendelzon, A. O., 2002. OLAP Dimension Constraints. *In PODS'02, 21st ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*. USA, 169-179.

Kamps, J., Marx, M., De Rijke, M,. Sigurbjornsson, B., 2004. Best-Match Querying from Document-Centric XML. *In Proceedings of the Seventh International Workshop the Web and Databases*. 55-60.

Khrouf, K., 2004. Entrepôts de documents : De l'alimentation à l'exploitation. PhD thesis, University of Toulouse III, France.

Lee, M. L., Yang, L. H., Hsu, W., Yang, X., 2002. XClust: clustering XML schemas for effective integration. *In CIKM'02, Proceeding of the ACM International Conference on Information and Knowledge Management*. Virginia, 292–299.

McCabe, M. C., Lee, J., Chowdhury, A., Grossman, D., Frieder, O., 2000. On the design and evaluation of a multi-dimensional approach to information retrieval. *In SIGR'00, Proceedings of the 23th Annual International ACM SIGIR Conference*. 363–365.

Pérez, M. J. M., Berlanga, L. M. R., Aramburu, C. M. J., Pederson, T. B., 2008. Contextualizing data warehouses with documents. *In Decision Support System*. Vol. 45. Elseiver, 77-94.

Pujolle, G., Ravat, F., Teste, O., Tournier, R., 2011. Multidimensional Database Design from Document-Centric XML Documents. *In DAWAK'11, International Conference on Data Warehousing and Knowledge Discovery*. France , 51-65.

Ravat, F., Teste, O., 2000. A temporal object-oriented data warehouse model. *In DEXA'00, Database and Expert Systems Applications*. London, 583-592.

Ravat, F., Teste, O., Tournier, R., 2007. Analyse multidimensionnelle de documents via des dimensions OLAP. *In Revue Document numérique Entreposage de documents et données semi-structurées*. 85-104.

Sullivan, D., 2001. *Document Warehousing and Text Mining: Techniques for Improving Business Operations*. Marketing and Sales. John Wiley & Sons.

Tournier, R., 2007. Analyse en ligne (OLAP) des documents. PhD thesis, University of Toulouse III, France.

Tseng, F. S. C., Chou, A. Y. H., 2006. The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence. *In Decision Support Systems (DSS)*. Vol. 42. Elsevier, 727–744.

Yoo, C. S., Woo, S. M., Kim, Y. S., 2005. Unification of XML DTD for XML Documents with Similar Structure. *In ICCSA'05, Computational Science and its Applications*. Singapore, 954-963.