New method to parse invoice as a type the document

MOUJABBIR Mohammed[1], RAMDANI Mohamed[2]

**[1] University Hasssan II Faculty of sciences and technology
Department of computer system
Mohammedia, BP 146 Mohammedia, Morroco**

**[2] University Hasssan II Faculty of sciences and technology
Department of computer system
Mohammedia, BP 146 Mohammedia, Morroco**

Abstract

In this paper We propose a new method able to detecting and correcting errors relating to the recognition of invoice type documents. We rely on automated document readers that can read and recognize the various relevant information in a scanned document.

The process on which this method is based consists of digitizing a large volume of documents, and makes them pass through automatic readers of the documents, then carry out the correction of the various errors. The final goal is to find an electronic document reflecting the various information included in the background document. The main goal is the generation of organized electronic documents, like a data basis or files XML; for a specific use.

Our approach is based on the language theory through developing a kind of parser which is applicable to the more general case of documents and can easily detect a specific class of errors and correct them.

Keywords: *Documents dematerialization, electronic invoicing dematerialization, autorun, character automatic recognition, languages theory, compilation techniques.*

1. Introduction

The current large development and deployment of dematerialization the documents has a great effect on the research activities in the domain of recognition documents, detection errors and also correction them.

Information and communication technologies have had considerable effects on how companies do business with their business partners [3]. In a narrow sense, these effects are based on electronic commerce (e-commerce), which is the buying of products from suppliers and their selling to customers using the new information technologies.

There are several models of e-commerce, namely, business-to-business, e-commerce between companies, business-to-consumer, e-commerce between companies and consumers and business-to-government, e-commerce between companies and government organizations.

The process of dematerialization is aimed at the transition from a physical document to an electronic document (structured document[1] or not) without human intervention [2]. This is made possible by using an OCR[1] and an error processing method. This method is crucial for such a transition which can’t be, in any case, conducted in a transparent manner, i.e., without errors. Several injections of errors are due to several factors including printing quality, quality of the paper used, scanner resolution, software power (OCR) ... Hence, an error processing operation is necessary [1], that can be divided into two parts, one for detecting errors, and another to correct them.

The goal of this study is to provide a system assessing the different aspects especially relevant information contained in a physical document, into electronic textual contents.

The problem to be raised is to envisage a pretreatment of the errors [1], which ties to correct errors of recognition before reaching the learning phase. The existing solution will be presented: the arborescent[3] treatment. This solutions advance many anomalies which will be detailed in the following sections. However this paper introduces a new approach which will be compared to the XML technology, also they (approach) provide us the possibility to generate a new parser able to detect and correct errors. Practically this parser gives us very good results, specially when XML technology was not able to detect all the existing errors.

In this article we will detail the process of the document processing, with a focus on the arborescent method. In section III will present the anomalies and limitations generated by the existing method. The section VI reports the solution suggested which proposes a tally formal modeling; a language dedicated to the documents with a grammar and syntax. However the section V introduces the results obtained and a discussion of these

[1] : Optical character recognition.

results. The last section will focus on the practical implications of our study.

The text must be in English. Authors whose English language is not their own are certainly requested to have their manuscripts checked (or co-authored) by an English native speaker, for linguistic correctness before submission and in its final version, if changes had been made to the initial version. The submitted typeset scripts of each contribution must be in their final form and of good appearance because they will be printed directly. The document you are reading is written in the format that should be used in your paper.

This document is set in 10-point Times New Roman. If absolutely necessary, we suggest the use of condensed line spacing rather than smaller point sizes. Some technical formatting software print mathematical formulas in italic type, with subscripts and superscripts in a slightly smaller font size. This is acceptable.

2. The xml technology

XML(Extensible Markup Language) is becoming a dominant standard for storing and exchanging information. They use several tools such as DTD, XSLT,XSL-FO, XSLQuery, Schema-XML, each one of them has its own specification, and can be used in various domains like data warehousing ,web, e_commerce …. Since XML is used as a standard for communicating information on the Web,. Now the XML technology has become a standard format to exchange information over the Internet, and the importance of database technologies that support storage, processing, and delivery of XML is still increasing [4].

2.1 DTD

The solution provides a template precast DTD to validate the compliance of a new entry (like files) from the model established. In the literature there are several variants (normalized or not) of the model DTD that are well presented and offers to the user an ergonomic space well done [5].

2.1.1 Example:

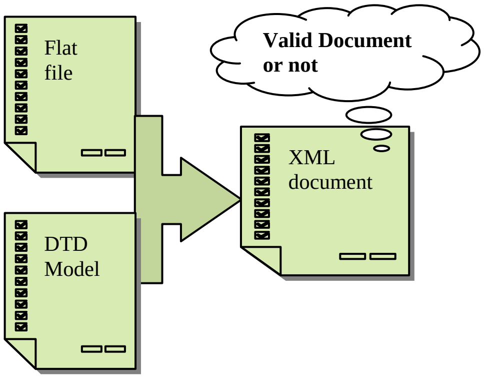


Fig. 1 Transformation from a flat file to an XML file via a model DTD

2.2 Schema-Xml and DTD

XML Schema as a recommendation published by the W3C is a language for describing XML document format for defining the structure and content type of an XML document (the syntax). This definition allows in particular to verify the validity of this document.

The XLM Schema is usually used with DTD to validate the documents and together they present a robust [6] tool for transforming a plat file into an XML document. The following figure shows the process.

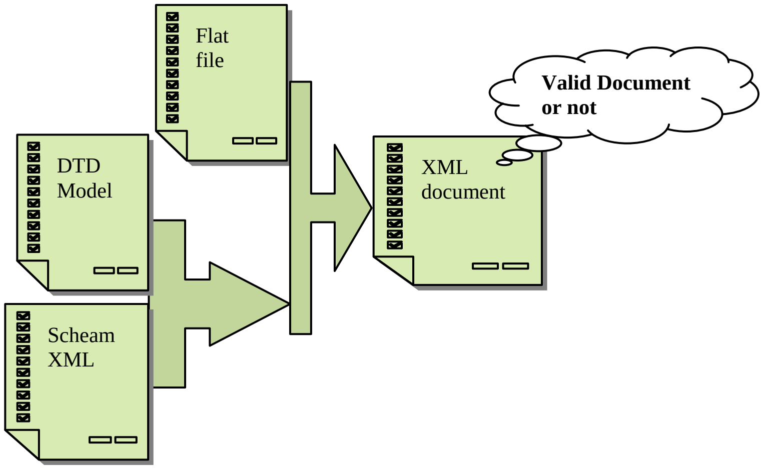


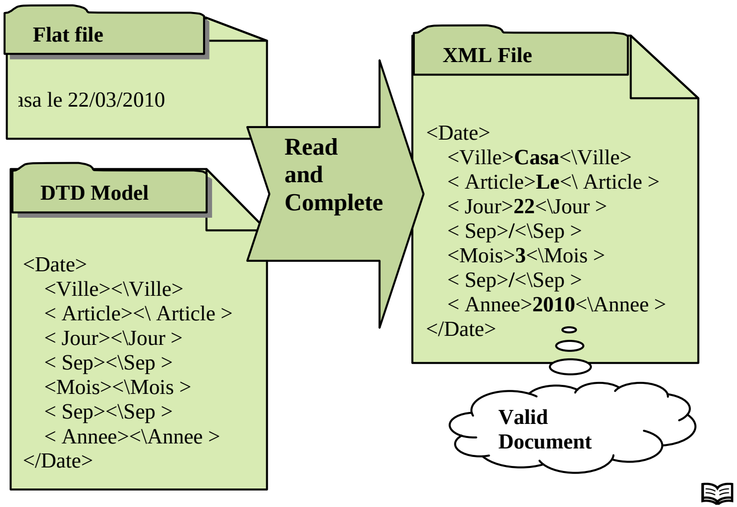
Fig 2 : Transformation from a flat file to an XML file via a model DTD and XML schema.

3. Limitation the XML solution

3.1 DTD solution

The problem raised in the DTD model is that it has no lexical vision nor semantics of the processed data, so the lexeme read are inserted (in the XML file) by the first come first served without a general understanding of the sequence of tokens. And it may generate additional errors compared to the errors recognized by the OCR. The following examples illustrate the three cases document validation.

3.1.1 Example 1 : Valid document



ısa le 22/03/2010

```
<Date>
  <Ville><\Ville>
  < Article><\ Article >
  < Jour><\Jour >
  < Sep><\Sep >
  <Mois><\Mois >
  < Sep><\Sep >
  < Annee><\Annee >
</Date>
```

```
<Date>
  <Ville>Casa<\Ville>
  < Article>Le<\ Article >
  < Jour>22<\Jour >
  < Sep>/<\Sep >
  <Mois>3<\Mois >
  < Sep>/<\Sep >
  < Annee>2010<\Annee >
</Date>
```

Fig 3: The reading of the document is made successfully, because there were no errors in the flat file.

3.1.2 Example 2 : Invalid document

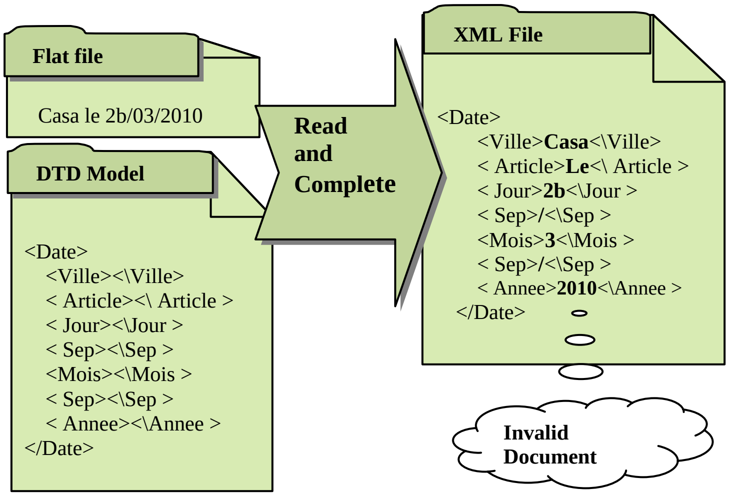


Fig 4: Transformation from a flat file to an XML file via a DTD model with errors recognition.

DTD is not able to detect that the document is invalid, since the meaning of the lexeme (Day 2b) passes unseen.

3.1.3 Example 3 : Invalid document

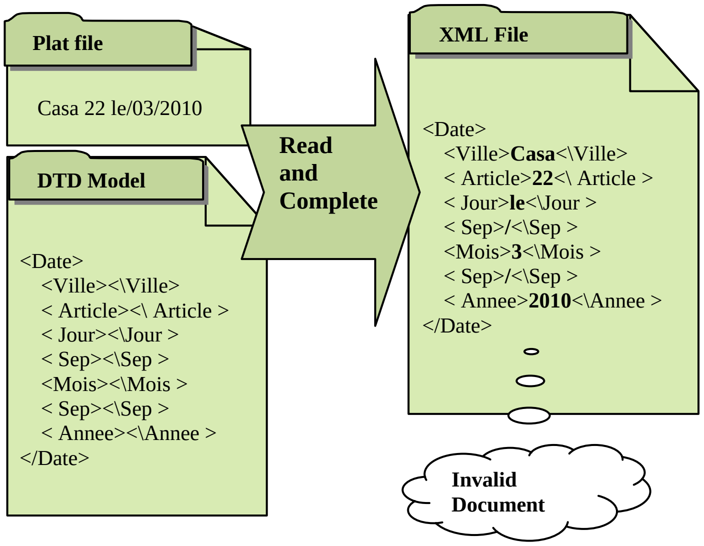


Fig 5: Transformation from a flat file to an XML file via a DTD model with errors recognition.

DDT is still unable to detect grammatical errors, as it displays no recognition error.

3.2 DTD + Schema XML solution

XML Schema offers a very good tool for the validation of documents, although it is able to detect lexical and semantic errors, but some errors can escape it because it doesn't operate on the syntactic level. Even Concatenated with the DTD tool, errors keep showing up in the system. The following examples illustrate the three cases document validation.

3.2.1 Example 1: Valid document

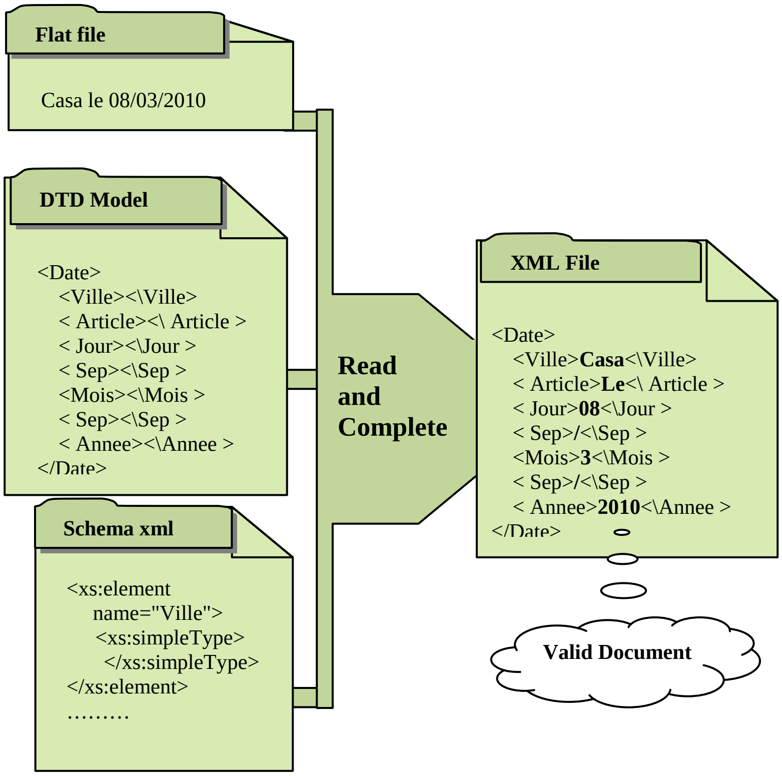


Fig 6: Transformation from a flat file to an XML file via a model DTD and schema xml without recognition errors

The reading of the document is made successfully, because there were no errors in the flat file.

3.2.2 Example 2: invalid Document

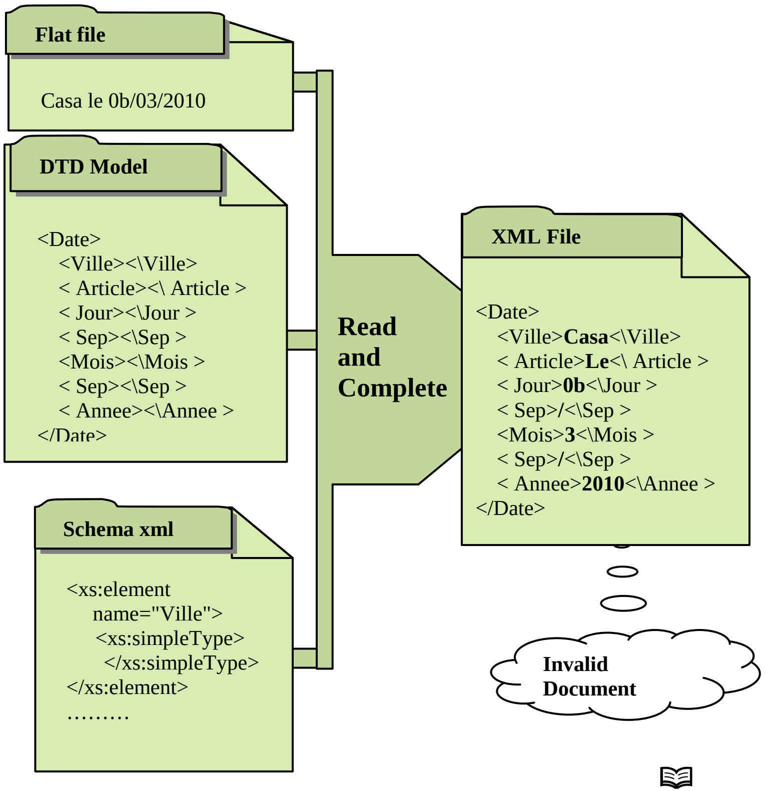


Fig 7: Transformation from a flat file to an XML file via a DTD model and an xml schema with errors recognition.

The xml schema is able to detect the lexical errors (0b :the day field), and also the semantic errors like 23for the month field.

3.2.3 Example: invalid Document

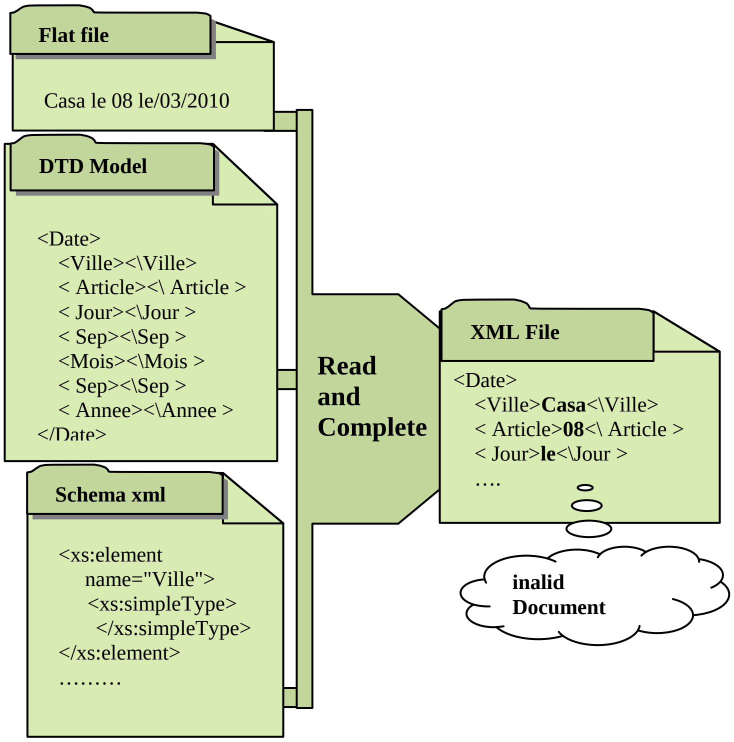


Fig 8: Transformation from a flat file to an XML file via a DTD model and an xml diagram with errors recognition.

The schema xml is not able to detect the syntax errors and they can generate an additional error because they don't have a global vision for the arrival tokens.

3.3 Summarization:

The use of xml in the dematerialization of documents is useful and practical, since this technology is capable of detecting a number of mistakes, but it is unable to detect all mistakes, besides, this method is unable to correct errors recognition

This limitation is due to the fact that XML is not a vision of the syntactic outcome studied, and it just treats lexeme by lexeme without a comprehensive framework on the content covered.

4. The method suggested

The majority of documents of the invoice type have the same structure [7], and sharing the same zones [8] like: dates zone, customer zone… and many forms such us: prices, tables, logo etc… In other words, it resembles a structure [7] which belongs to and respects a given language, i.e. grammar with a specific lexicon and syntax.

Our approach is based on the development of a grammar which will give the contents of the documents of the type invoices [9]. Thus, the set up of a grammar per customer will enable us to deduce the structure [10] of its documents and consequently will allow us to detect all the errors related to the grammar of its language and to carry out their correction if possible.

5. Grammar of the document

This approach includes three types of analyzers: lexical, syntactic and semantic. Each one of them will be concerned with a specific task which will be described in the following sections. But going further in the description of the types of analyzer, it is initially necessary to define the alphabet on which one will work.

In the literature, an alphabet is a nonempty finite whole of symbols. The latter can be unspecified letters or characters. As well says as a word is in addition to only one sequence of elements of A.(not clear)

Practically, the invoice document uses usual symbols like a…z,A…Z.., for that the alphabet [11] adopted for this kind of documents(invoices)doesn't leave this formal framework.

5.1 Parser Vs DTD

The syntax of an invoice document can be described by a grammar describing the arrangement of lexical units. The parser receives a sequence of lexical units from the lexical analyzer and must verify if it can be generated by the grammar of language.

5.1.1 Formal deffinition of grammar [11]

Formally, a grammar is a set denoted as G=(V_T,V_N,S,P).

- V_N is a non-empty set of terminal symbols.
- V_T is a non-empty set of non-terminal symbols with $V_N \cap V_T$ =empty.
  - S is an initial symbol (axiom).
  - P is a set of production rules.

5.1.2 In pratique

Les us examine the following example:

From a practical standpoint, we consider the following grammar:

Ident = {Casa, Casablanca, Client, Rabat,….}
Chainqqc={a|..|z,A|….|Z,0|1..|9| ;| ?|...}+
Figure=(0|1|2|3|4|5|6|7|8|9)+.
Sep=(-|/|:|.|%).

From the preestablished symbols we can form the following units:

V_T ={Ident, Figure, Sep}.

V_N={DateZone,ClientZone,QteZone,UnitPriceZone, VATZone
,TTCZone,SumInLettersZone}.
S=Zone.
P={
Zone→
DateZone|ClientZone|QteZone|UnitPriceZone| TVAZone|TTCZone|SumInLettersZone.
DateZone → Ident Figure Sep Figure Sep Figure.
ClientZone → Ident Sep Chainqqc.
QteZone → Figure.
UnitPriceZone → Figure Sep Figure.
VATZone → figure Sep
TTCZone → Figure Sep Figure.
SumInLettersZone → Chainqqc.
}

While the DTD is a limited tool because it addresses only the part shape and also some recognition errors could go undetected.

5.1.3 Lexical[11] and semantic[12] analyzer Vs schema XML

The principal task of this analyzer is reading the characters of entry and producing as a result a succession of lexemes that the parser will have to treat. Still, it is necessary to define what a lexeme is.

A lexeme is a continuation of characters which has a collective significance. Take this sentence, for example: Casa the 12/03/2009; it can be translated in the following way Ident (Keyword) Ident Chiffre (chiffre or number?!) backslash Chiffre backslash Chiffre. With such an analyzer the detection of a possible lexical error is practically easier and less expensive: both in terms of the memory occupancy rate of the processor, and the complexity of the algorithm used as well.

Generally, a grammar cannot provide a 100% description of the content of a given language, even with the use of two powerful tools: the lexical analyzer and the parser. That is why languages generally rely on a third semantic analyzer [13].

This failure [14] is due to the fact that neither the lexical analyzer nor the parser can detect an error type such as:

- A year estimated at 3000
- A month exceeds 12
- A miscalculation of the TTC.

6. Results and discussions

6.1 Results

The use of a model based on XML diagram, as well as the model based on our approach gives important results, the latter are given in the form of three fields in particular the average of the existing errors, the average of the detected errors, the average of corrigible errors, and the percentage of correction.

6.1.1 Case 1: without syntax errors

The results obtained during the test are shown in the table below:

Table 1: Average errors

Method used	*Number invoices*	*Average existing Errors*	*Average detected Errors*	*Average corrigible Errors*	*Percent-age of correction*
Diagram XML	100	35	82	30	85,72%
Our approach	100	33	70	10	30,30%

6.1.2 Case 2: with syntax errors

The results obtained during the test are shown in the table below:

Table 2: Average errors

Method used	*Number invoices*	*Average existing Errors*	*Average detected Errors*	*Average corrigible Errors*	*Percent-age of correction*
Diagram XML	100	35	82	5	14,29%
Our approach	100	33	63	26	78,79%

6.1.3 Discussion and comparison of the results

The model based on an xml diagram, proposes a multitude of choices concerning the types to be defined, in particular the kind types: string, positive integer… as well as the possibility of generating a regular expression, for that the xml solution is able to read lexeme by lexeme and to test the validity of each chain with share. On the contrary our method treats at the same time lexeme by lexeme as well as the sequence of the continuations of the lexemes according to a given order.

To sum up, the xml solution treats only the lexical part, however our method will operate beyond the lexical part respectively on the syntactic and semantic levels, which partly explains the variation observed on the level of the rate of the corrigible errors.

A positive account about this method is the ability to offer a correction when there is a syntax error, while this has been impossible through XML. This possibility is offered by the language theory, specially LL[1] and SLR[2] languages [11].

7. Conclusion and Implications

In this paper, we proposed a new method based on the theory of languages, it consists in installing a mini compiler which operates via three analyzers: The process of correction which is summarized in the following steps: Reception of the concerned zone, launch of the lexical analyzer, Launch of the parser, launch of the semantic analyzer, then Correction of the errors if possible.

A major advantage of this method is its ability to detect all errors of recognition and most (but not all) of them (all of them or most of them?). Another positive point about this method is that it is about a less expensive solution and especially an easy one to set up.

Our main goal has been to develop a learning tool capable of correcting the errors which are not detected by the mini compiler.

References

[1] Rémy Kessler, Juan Manuel, Torres-Moreno et Marc El-Bèze, Classification automatique de courriers électroniques par des méthodes mixtes d'apprentissage - Laboratoire d'Informatique d'Avignon / Université d'Avignon- 2001.

[2] Yassin Aziz REKIK : Modélisation et manipulation des documents structurés : une approche modulaire, flexible et évolutive. –Thèse N° 2396 (2001)-.

[3] Cécile Roisin Adaptation aux différents modes de lecture 2004

[4] E.J.Thomson Fredrick, G.Radhamani: INFORMATION RETRIEVAL USING XQUERY PROCESSING TECHNIQUES International Journal of Database Management Systems (IJDMS), Vol.3, No.1, February 2011.

[5] Zurinahni Zainol : XML Documents Normalization Using GN-DTD International Journal of Information Retrieval Research, 1(1), 53-76, January-March 2011 53

[6] Kinsun Tam Sanjay Goel Jagdish S. Gangolly On the design of an XML-Schema based application for business reporting: An XBRL Schema Perspective The International Journal of Digital Accounting Research Vol. 2, No. 1, pp. 83-118.

[7] Noureddine CHATTI, Sylvie CALABRETTO : MultiX : un formalisme pour l'encodage des documents multi-structurés - LIRIS-INSA de LYON - 2001.

[8] Rocio Abascal - Michel Beigbeder - Aurélien Bénel - Sylvie Calabretto -Bertrand Chabbat - Pierre-Antoine Champin - Noureddine Chatti – David Jouve - Yannick Prié - Béatrice Rumpler - Eric Thivant : Modéliser la structuration multiple des documents - LIRIS – INSA de Lyon – 2001.

[9] Rocio Abascal, Michel Beigbeder, Aurélien Bénel, Sylvie Calabretto, Bertrand Chabbat, Pierre-Antoine Champin,

[10] Noureddine Chatti, David Jouve, Yannick Prié, Béatrice Rumpler, Eric Thivant :Documents à structures multiples - LIRIS CNRS FRE-2672 INSA de Lyon 2000-.

[11] Pierre-Edouard Portier_, Sylvie Calabretto Modélisation des connaissances dans le cadre de bibliothèques numériques spécialisées - Université De Lyon, INSA de Lyon, LIRIS - 2002.

[12] Compilation et théorie des langages –Universite de bretagne occidentale-.

[13] D. Jouve a,b,*, Y. Amghar a, B. Chabbat b, J.-M. Pinon Conceptual framework for document semantic modelling: an application to document and knowledge management in the legal domain –sciencedirect 29 junuary 2003-.

[14] David JOUVE : Modélisation sémantique de la réglementation –thèse 03 ISAL 0071 28 novembre 2003-.

[15] Jean-Luc Minel, Jean-Pierre Desclé, Emmanuel Cartier. Gustavo Crispin, Slim Ben Hazez Agata Jackiewicz :

[1] Left to right scanning, and leftmost derivation.

[2] Shift/Reduce Left right