



HAL
open science

Towards a flexible content adaptation of semi-structured documents

Ahlem Kalai, Ikram Amous, Corinne Amel Zayani, Florence Sèdes, Ahlem Kalai

► To cite this version:

Ahlem Kalai, Ikram Amous, Corinne Amel Zayani, Florence Sèdes, Ahlem Kalai. Towards a flexible content adaptation of semi-structured documents. 2nd International Conference on Web and Information Technologies (ICWIT 2009), Jun 2009, Kerkennah, Tunisia. pp.559-572. hal-03765453

HAL Id: hal-03765453

<https://hal.science/hal-03765453>

Submitted on 31 Aug 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards a flexible content adaptation of semi-structured documents

Ahlem Kalai* — Ikram Amous* — Corinne Amel Zayani*, ** — Florence Sèdes**

* MIRACL, ISIMS, Cité El Ons, route de Tunis Km 10, Sekit Ezzit 3021, Sfax

* ahlem.kalai@gmail.com, ikram.amous@iseecs.rnu.tn

** IRIT, 118 Route de Narbonne, 31062 Toulouse cedex 4, France

** {zayani, sedes}@irit.fr

ABSTRACT. In a generic context, adapting the documents contents is based on the user's profile. Its main objective is to reduce the cognitive overload for the user. Despite the satisfaction of the experiments of previous works, the adaptation is still limited. The flexibility in querying semi-structured documents can resolve effectively the cognitive overload. In this way, our work is oriented towards the flexible content adaptation of XML documents. Our adaptation mechanism takes into account not only user interests but also the weighted query. However, the flexibility is the use of the techniques of approximate matching of trees between the query and the documents. This matching helps afterward in the process of adapting by raising the similarity at the structure and content level. Our approach provides to the user a re-ordering of documents based on his profile. This ordering is linked to the implicit ordering of the documentary unit associated with the documents. Its objective is to provide the most relevant documentary unit satisfying the user query and interests.

RÉSUMÉ. Dans un cadre générique, l'adaptation du contenu des documents est basée sur le profil utilisateur. Son objectif principal est de lui réduire la surcharge cognitive. Malgré la satisfaction des expérimentations des travaux antérieures, cette adaptation est encore limitée. La flexibilité dans l'interrogation des documents semi-structurés peut résoudre efficacement la surcharge cognitive. C'est dans ce sens, que notre travail s'oriente et plus particulièrement vers l'adaptation flexible du contenu des documents XML. Notre mécanisme d'adaptation tient compte non seulement des intérêts de l'utilisateur mais aussi de la requête pondérée. Cependant, la flexibilité réside dans l'utilisation des techniques d'appariement approximatif d'arbres entre la requête et les documents. Cet appariement aide par la suite dans le processus d'adaptation en évoquant la similarité au niveau structure et contenu. Notre approche fournit à l'utilisateur un ré-ordonnement des documents adaptés selon son profil. Cet ordonnancement est en référence à l'ordonnancement implicite des unités documentaires associées aux documents. Le but est de fournir des unités documentaires les plus pertinentes satisfaisantes la requête et des intérêts.

KEY-WORDS: flexible content adaptation, the weighted query, the semi-structured documents, interests, the implicit ordering, structure and content similarity.

MOTS-CLES: adaptation flexible du contenu, requête pondérée, documents semi-structurés, intérêts, ordonnancement implicite, similarité structure et contenu.

1. Introduction

Generally, the adaptation of the content is a “*user-oriented*” field (Brusilovsky, 1996) (Koch, 2000) which enables him to provide relevant documents which respond to his needs in a way to reduce the problem of the cognitive overload (Brusilovsky, 1996). Based on the contents, two types of documents were distinguished (Abiteboul, 1997): (i) *non structured documents* also defined as “*oriented text*” (plain text) which can neither be integrated into any information nor a structural element, and (ii) *semi-structured/structured documents* also called “*oriented data*” which contain information linked to the structure of a document. According to every document type, two adaptations views were raised. The first vision concerns the adaptation of the unstructured documents which always depends on the domain. In this context, (Boyle, 1994) proposed a MetaDoc¹ system related to the teaching domain which enables to adapt the text to the user profile (characteristics, interests, preferences). The second view is related to the adaptation of the semi-structured documents which is generally independent of the domain (Amous, 2005), (Alilaouar, 2006) and (Zayani, 2007a).

In (Amous, 2005), the semi-structured documents were described by meta-data in order to facilitate the research according to the needs of the users which are explicitly defined. In (Alilaouar, 2006) a flexible model to query the XML documents was proposed using the concept of minimal size spanning tree and the fuzzy logic framework. The work purpose is to provide to the user more flexibility at the search level. In (Zayani, 2007a), a mechanism of adaptation of the XML documents is proposed. These documents are not conceived to be adapted in advance. The process of adaptation is made in a generic context independent of the domain and according to the user’s profile. This mechanism of adaptation does not take into account meta-data and flexible queries because they provide further additional information at the search level.

In this paper, we propose a mechanism of flexible content adaptation pursuant to a flexible query of XML documents. In fact, our mechanism has as an input i) *the user’s profile* (essentially their interests), ii) *the collections of XML documents* and iii) *a weighted query*. At this level, the flexible adaptation takes then into account not only the contents of the documents but also their structure. Taking into consideration the latter, implicitly effects on the relevance document/query in relation to users’ interests. At the end of the mechanism of flexible adaptation of the contents, two types of re-ordering results will be established.

This paper is organised as follows. In section 2, we present an overview of the literature. In section 3, a description of our proposal is given. An experiment of our proposal is detailed in section 4. Finally, a conclusion and some prospects are presented.

¹ A system of adaptive hypermedia information online

2. Related works

The adaptation concept consists in adapting the content of documents to the user's profile (characteristics, preferences, interests). The objective is then to focus on the information which can respond to the real need of the user. In the literature, works aim at adapting documents in a "generic" context or inversely in a "specific" context.

Let's take the example of the research works of (Balla, 2008) related to the teaching domain. The system proposed by (Balla, 2005) contains one or more models (domain, activity) and a course generator. Adapting a course to a learner is provided the *mobility of a part of the course* which depends on the evaluation of the levels of knowledge of the learner. This work is designed for heterogeneous document structures.

The adaptation of (Zayani, 2007a) is another example of content adaptation mechanism of XML documents collections which we can mention. They propose a new adaptation technique called documents *implicit re-ordering* based on the user interests. The proposed mechanism does not take into account the flexible querying based on the user interests. In this context, (Alilaouar, 2006) propose a flexible model to querying the XML documents. In fact, they use the *concept of the fuzzy logic framework* (linked to the contents), and *the concept of minimal size spanning tree* (linked to the structure) without considering the user profile.

To summarize, the adaptation of the document contents "*text oriented*" or "*data oriented*" can be applied in a "*generic*" or "*specific*" context. At this stage, research works aim at adapting the contents to the user's profile but they do not try to use the structure linked to documents to be queried. Others are interested, in addition to the contents, in the documents structure and the query to respond to the user without referring to his profile. At this level, our work consists in improving the mechanism of content adaptation (Zayani, 2007a) by a flexible querying. This improvement shows in the ordering of the results returned to the user. The flexible adaptation mechanism is carried out in a generic context and based on the user interests.

3. Flexible Adaptation Process

Our work extends the content adaptation to a flexible adaptation of the contents. Its mechanism is based on an improved reference model (Zayani, 2007a) inspired by an original model MRM proposed by (Koch, 2000).

The reference model includes (i) *a user model* which includes the user's interests' classified on a thematic context, (ii) *a document model* which represents one or several document collections which can be queried and where every collection is represented by a meta-collection in order to facilitate their querying further to a flexible query, (iii) *a query model* that defines the user's non-weighted

query expression. This query specifies the interests within the querying conditions and priorities (after modification to a weighted query). This priority is shown by attributing the weights, extracted from the profile, to the interests appearing in the query. Generally, the weight represents an interest importance degree in the same profile of a query.

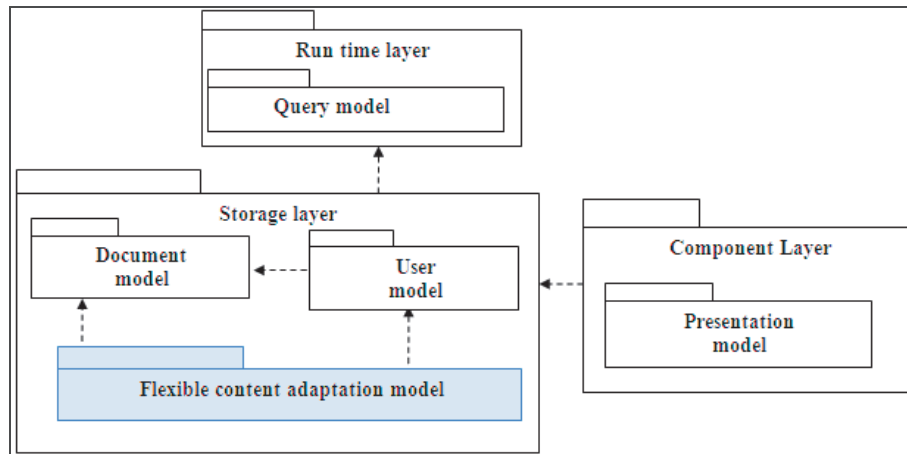


Figure 1. Improved reference model

The three models mentioned previously are closely related to the (iv) *flexible content adaptation model* which is the core of the flexible content adaptation process. It includes the adaptation mechanism starting from the query until the adaptation. This mechanism includes a set of steps which implement the operational features of the flexible content adaptation process. These steps mainly consist on: (a) implicitly modify the user's initial non-weighted query to a weighted one based on the interests saved in his profile, (b) query one or several document collections on thematic context through flexible matching techniques (c) implicitly order the obtained results in two steps: the first one order documentary unit within the documents based on the profile, the second one re-ordering the same documents and finally (d) to improve the initial query and to update the user's profile.

Another model is presented in this figure its' named a *presentation model*. This latter include the adaptive presentation mechanism of the adapted documents (results of the flexible content adaptation mechanism) according to user's preferences. The presentation mechanism consist on show different layouts of perceivable user interface elements, such as different type of media, different ordering or different colors, font size, font type or image size (Koch, 2000).

3.1. Architecture of the flexible content adaptation mechanism

The following figure shows the architecture of flexible adaptation process of the XML semi-structured documents contents which are not designed to be adapted in advance.

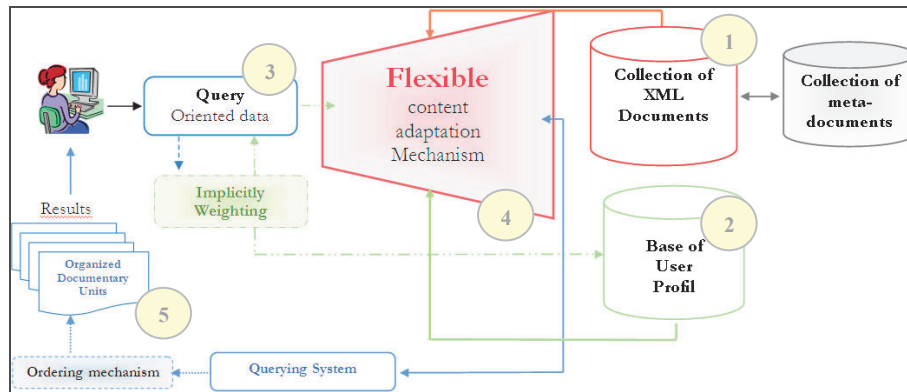


Figure 2. Layout of the flexible content adaptation mechanism

The proposed architecture includes five main elements:

- 1– **Every documents collection** contains a set of XML documents which belong to the same thematic context. Further to a user’s query, we link a meta-document to every XML document in order to make his query easier. Generally, the user has no knowledge on the structure of these document collections.
- 2– **The user’s profile** represents the user’s interest’s areas filed by topic. From the first connection (first query), an initialization of current user’s profile is conducted by linking a frequency (Zayani, 2007b) and a weight for every interest variable. Therefore, this weight represents the importance degree of interests.
- 3– **The user query.** Based on the user profile, the first expression of the query is modified implicitly in the form of a weighted tree. This new expression allows discriminating the relevance between the query nodes in general and then the nodes of the documents in particular.
- 4– **The flexible content adaptation mechanism** has three input elements (previously mentioned). In the query execution, an evaluation of the weights of nodes is made for the matching between both trees: query and document. Once sub-trees are found, a joint between nodes and a spreading of the weights to the tree root will be realized.

5– The result. The mechanism provides adapted fragmented data based on the user's interests and implicitly ordered based to their weights. A re-ordering of documents to be restored is conducted on a decreasing relevance degree.

3.2. Notations for the flexible content adaptation mechanism

Through this process, in the query step, we are looking for all the sub-trees of the XML document to be queried and which coincide with the query tree. In fact, we apply the ACM concept of minimal size spanning tree. In our context of the content adaptation, we redefine this ACM concept used by (Alilaouar, 2006). Generally, we admit three trees as an input: the user's profile, the weighted query tree and the documents collection tree. The notations used in every step of the process are:

- $D (V, E, \text{root}(D))$ the document tree; V and E represent respectively the documents nodes and the arcs connecting the nodes.
- $Q (N, A, \text{root}(Q))$ the query tree; N and A represent respectively the query nodes and the arcs connecting the nodes.
- $P (I, R, \text{root}(P))$ the user's profile tree; I and R represent respectively the interest nodes and the arcs connecting the nodes.

Therefore, in our mechanism we raise two types of relations. The first is between the user's initial query and his profile. The second is between the weighted query and the queried document.

–Relation Query Q / Profile P

- For every n belongs to N , there is i belongs to I ;
- $\text{label}(n) = \text{label}(f(i))$; $f: P \rightarrow N$ is a " mapping " function which assigns the Q nodes of to P nodes, label represents the name of the tree node.
- $\text{weight}(n) \leftarrow \text{weight}(i) / 0 \leq \text{weight} \leq 1$. This weight represents the degree of importance of a given node compared with other nodes of the same tree.

Through this definition, we get a new representation of the weighted query: Q_{weight} which is a tree represented as follows: $Q_{\text{weight}} (N_{\text{weight}}, A, \text{root}(Q_{\text{weight}}))$.

– Relation weighted query (Q_{weight}) / document (D)

$D' (V', E', d')$ is a sub-tree of D , is a tree including Q_{weight} by checking the following conditions:

- For every n which belongs to N_{weight} there is v belongs to V' .
- $\text{label}(n) = \text{label}(f(v))$ $f: V \rightarrow N_{\text{weight}}$ is a " mapping " function which assigns the nodes of Q_{weight} to nodes of D .

- For every $(v1, v2)$ belongs to $pred(V)^2$, $v1$ belongs to $red(v2)$; then $f(v1)$ belongs to $pred(f(v2))$; $pred$ represents the predecessor of a certain node.

In fact, all the nodes of Q_{weight} are "mapped" in nodes of D' with the same descending relations between them.

3.3. Steps for the flexible content adaptation

The process of our adaptation proposal is as shown below:

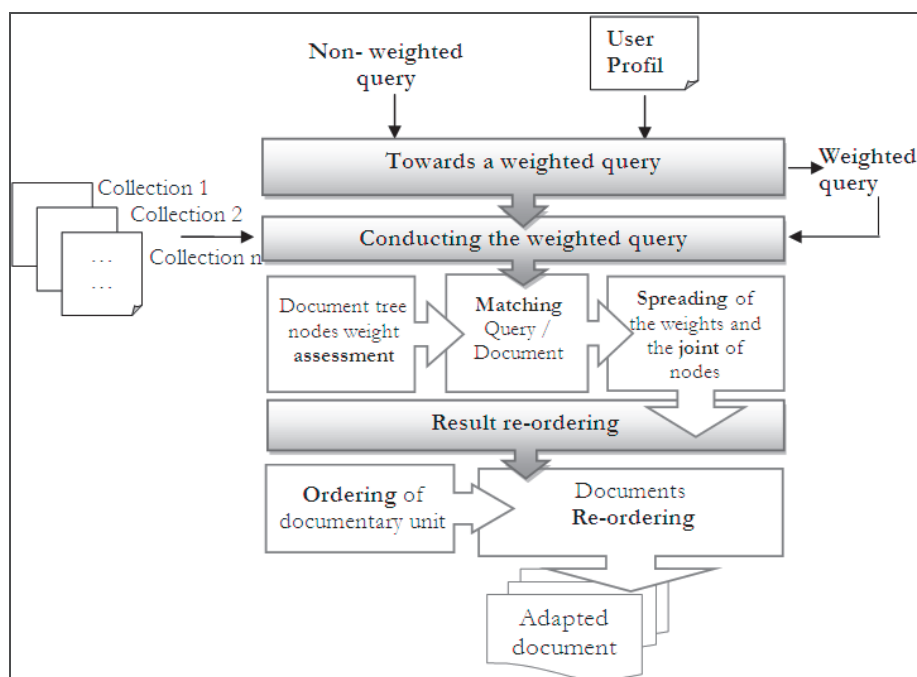


Figure 3. The steps of the flexible content adaptation mechanism

– **Step 1: Towards a weighted query.** The query provided by the user specifies his interests through querying conditions without explicitly indicating the priorities. Therefore, we suggest, first of all, extracting the general thematic context of the query in order to know the collection of documents to be queried. Then, we conduct a modification of a non-weighted initial query to a weighted query. The new query will be a weighted tree where the nodes are the interests linked to the extracted weights from the current user's profile. The consideration of the weights in the query expression allows obtaining the most satisfactory discriminated responses to

for the user. The weight of a node (interest) is calculated based on its frequency with the following formula (Zayani, 2007b).

$$\text{weight}(\text{ValueInterest}_i) = \frac{\text{Frequence}(\text{ValueInterest}_i)}{\sum_{j=1}^n \text{Frequence}(\text{ValueInterest}_j)} \quad [1]$$

The algorithm of query weighting is as follows:

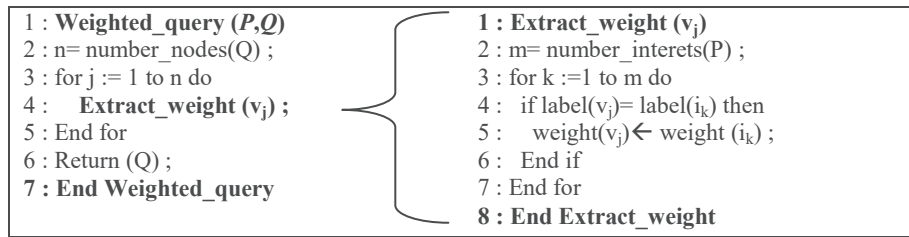


Figure 4. *Weighting algorithm of an initial query*

– **Step 2: Conducting the weighted query.** The flexibility issue is raised when conducting the query. We, therefore, use the approximate matching techniques of trees. At this step, we are looking for all the sub-trees in the document tree to be queried and which are likely to be similar, on the one side, to thematic context of the query and to the query tree, on the other side. This step is inspired from (Alilaouar, 2006) which has three phases:

Phase 1: Document tree nodes weight assessment. The weight of every document-tree node is calculated based on the query node. Two types of nodes can be distinguished (Sheet/ Intern). For every type of node, a weight calculation formula is defined. In our context, we use two formulas (Alilaouar, 2006). After calculating the nodes weight, we move to calculate the weights of sub-trees documents to be restored in order to assess the results.

Phase 2: Matching Query/Document. We are using a matching algorithm of (Alilaouar, 2006) of two trees document and query, by browsing the latter in post-order. The result of this algorithm is to build up sub-trees which are sufficiently similar to the query tree. At this level, we are looking for the structure similarity.

Phase 3: Spreading of the weights and the joint of nodes. Spreading the weights of nodes sheets to the node root in order to calculate the weight of the documentary unit to be returned to taking into account the hierarchical structure of the query. However, the weight of the sub-tree extracted from a tree-document is gradually calculated starting from the node sheet weights sheets until the root. The sub-tree weight is calculated with the following formula (Alilaouar, 2006).

$$\text{weight (T)} = \left(\frac{\text{level (u}_i) + \text{weight (u}_i)}{\text{level (u}_i) + 1} \text{weight (T[u}_i]) \right) / n \quad [2]$$

– **Step 3: Result Re-ordering.** Our objective is to order the documents to be restored based on their adequacy with the query. We propose that this re-ordering implicitly depends on the ordering of the documentary unit available in the documents to be ordered.

Phase 1: Ordering of documentary unit. We propose to carry out an ordering between the nodes which make the sub- tree. This ordering is ensured based on the weights linked to the nodes in the user’s profile. The objective is, then, to have the user implicitly focus on the fragments which might be more relevant compared to his query and his profile. The proposed algorithm is as follows:

```

1 : Ordering_sub-tree(Nodesv,n)
2 : Nodesv_order = ∅ ;
3 : k=2 ;
4 : While (k <= n) do
5 :   if weight(Nodesv[i]) > weight(Nodesv[i+2]) then
6 :     Nodesv_order = Permut_nodes(k-1,k,k+1,k+2);
7 :     k = +2 ;
8 :   else
9 :     k++;
10 : End if
11 : End While
12 : Return (Nodesv_order);
13 : End Ordering_sub-tree ;

```

Figure 5. Algorithm of sub-trees

Phase 2: Documents re-ordering. Every documentary unit has a satisfaction degree more or less relevant in relation with the query. In fact, the weight result linked to every document to be restored is equal to the sub-trees weight.

$$\text{weight (Document)} = \sum_{i=1}^n \text{weight (T[u}_i]) \quad [3]$$

In this case, our main task is to find out one or more sub-trees which most correspond to the query tree. We define a **threshold S (=0.5)** which enables to limit the number of sub-trees to be restored to the user in order to reduce the cognitive overload. The value linked to this threshold is arbitrary choice. We have therefore as a result only the documents with a maximum weight ≥ 0.5 .

4. Experiments

In this section, we try to present the assessments related to our proposal through an example. This assessment is conducted in a semi-automatic context in the PRETI platform.

4.1. PRETI Platform

We illustrate our works through an application example based on concerning the Cottage hire in Midi-Pyrenees which are available on the platform PRETI. In this context, our test base contains the initial query of a user, his profile and the XML documents of “cottages” collection. For the initial query, we suppose that a user X requests “cottages” which accept animals, 4 stars and which are located in the department "Aude". The query tree is as follows:

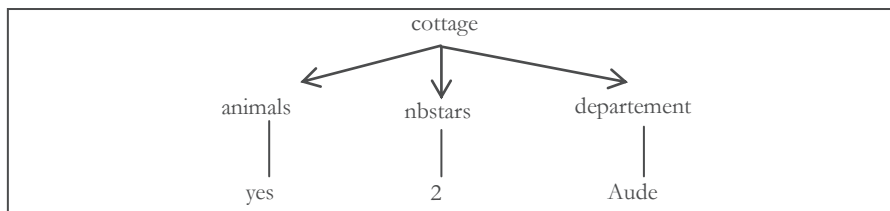


Figure 6. Non-weighted initial query

We present in the following figure an extract of a XML file representing only the users' interests.

```

<?xml version="1.0" encoding="UTF-8" ?>
- <Profil_user>
+ <CaracPermanente>
- <ListeCaracEvolutives>
- <CaracEvolutive contexte="gite">
- <ListeVarInterets>
- <VarInteret nom="animaux">
  <fréquence>2</fréquence>
  <date>12/03/07</date>
  <poids>0.28</poids>
- <interet>
  <FreqOpVal>2</FreqOpVal>
  <dateOpVal>12/03/07</dateOpVal>
  <poidsOpVal>0.5</poidsOpVal>
  <op_PU_interet>=</op_PU_interet>
  <val_PU_interet>yes</val_PU_interet>
</interet>
</VarInteret>
- <VarInteret nom="departement">
  <fréquence>3</fréquence>
  <date>19/03/08</date>
  <poids>0.5</poids>
- <interet>
  <FreqOpVal>3</FreqOpVal>
  <dateOpVal>19/03/07</dateOpVal>
  <poidsOpVal>0.5</poidsOpVal>
  <op_PU_interet>=</op_PU_interet>
  <val_PU_interet>Aude</val_PU_interet>
</interet>
</VarInteret>
- <VarInteret nom="nbstars">
  <fréquence>4</fréquence>
  <date>10/02/09</date>
  <poids>0.8</poids>
+ <interet>
+ <interet>
</VarInteret>
</ListeVarInterets>
</CaracEvolutive>
  
```

Figure 7. Example of a user's profile

We present an extract of XML documents collection with the "cottage" thematic context:

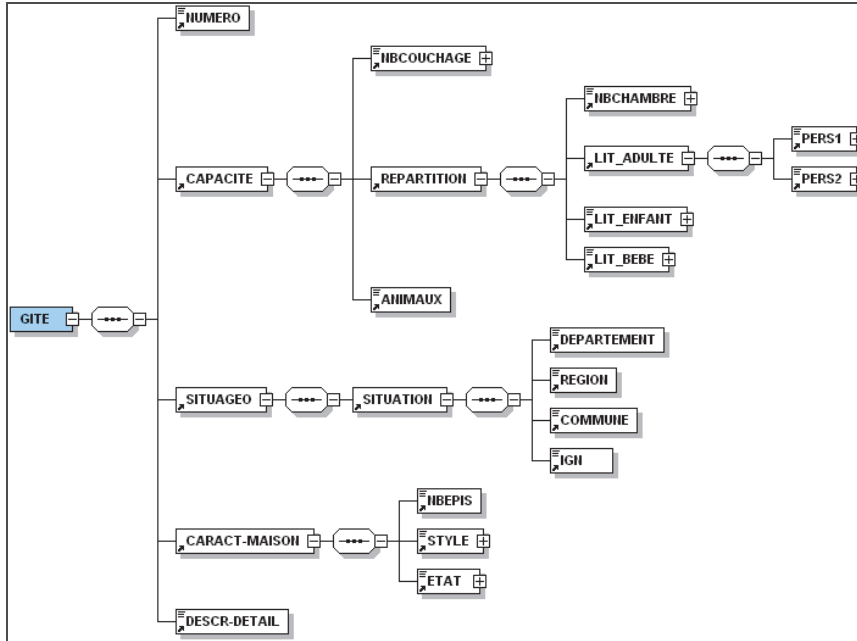


Figure 8. XML Diagram of an example of XML documents collection for cottages

4.2 Results

By browsing the query in post-order, we find: *yes, animals, 2, nbstars, Aude, department, cottage*. This last element of the list, provide us with an indication on the context of the collection of documents to be queried: [*context (query) = context (collection) = cottage*]. In our case, we query three XML documents. Thereafter, we browse the user's profile with interests list and context "cottage".

Based on the interest's weights (animals, department, nbstars), the query will be as structured as flows:

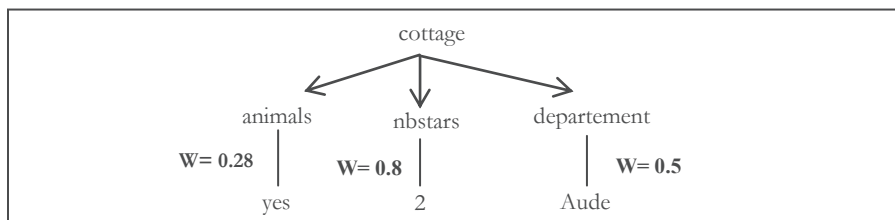


Figure 9. Weighted tree query based on to the profile.

By applying the algorithm of matching of trees ($Q_{weight,D}$) presented in the previous section (cf. 3.3), we obtain the following sub-trees:

Sub-tree 1 = {yes, <12,9>, 4, <20,17>, Aude, <15,11>, <2,22>} D_1 ,

Sub-tree 2 = {yes, <34,31>, 2, <42,39>, Aude, <37,33>, <24,44>} D_2 ,

Sub-tree 3 = {yes, <56,53>, 1, <64,61>, Aude, <57,55>, <46,66>} D_3 .

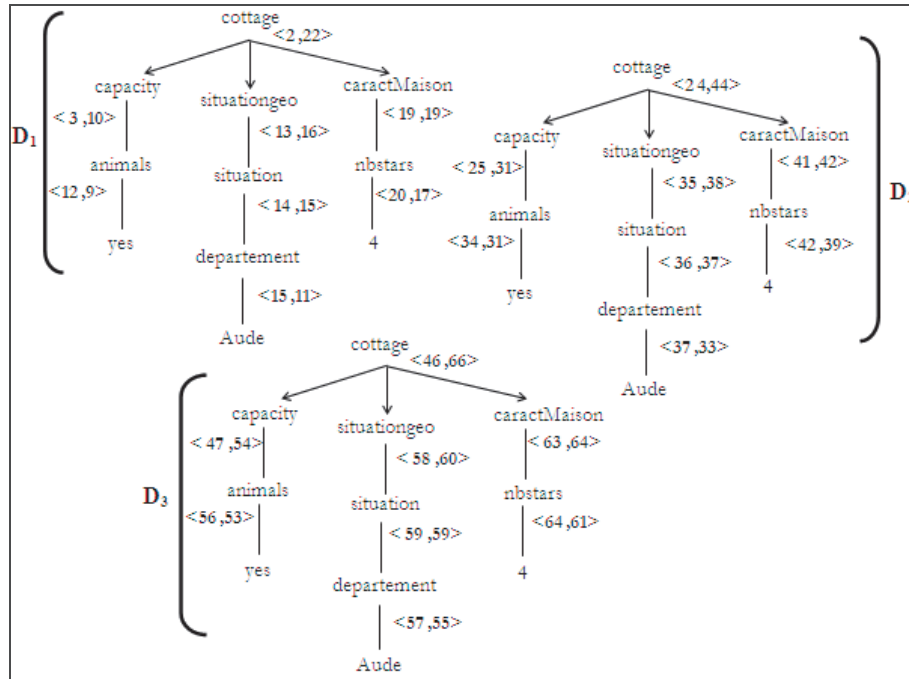


Figure10. Sub-trees results

By applying the formula [2], the results are as follows:

weight ($T[<2,22>]$) $D_1 = 0.5$,

weight ($T[<24,44>]$) $D_2 = 0.39$,

weight ($T[<46,66>]$) $D_3 = 0.47$.

Based on the weights linked to nodes in the profile [$weight(nbepis) = 0.8$, $weight(departement) = 0.5$, $weight(animals) = 0.28$], the ordering of documentary unit is as follows:

Sub-tree 1 = {4, <22,20>, Aude, <15,11>, yes, <12,9>, <2,22>} D_1 ,

Sub-tree 2 = {2, <42,39>, Aude, <37,33>, yes, <34,31>, <24,44>} D_2 ,

Sub-tree 3 = {1, <64,61>, Aude, <57,55>, yes, <56,53>, <46,66>} D_3 .

As a result, the documents are classified in a descending order and by relevance (the most relevant to the least relevant). In our example, the ordering result is as follows: Sub-tree 1(D_1), Sub-tree 3(D_3) and finally Sub-tree 2(D_2). Based on the

value threshold $S (=0.5)$, the only document D_1 will be shown to the user since the weight of the documentary unit $(0.5) \geq S$.

An example of an extract of XML document D_1 before and after the ordering is presented in the following figure:

<i>Before ordering:</i>	<i>After the units ordering:</i>
<pre> <?xml version="1.0" encoding="UTF-8" ?> - <GITE> - <CAPACITE> <ANIMAUX>yes</ANIMAUX> </CAPACITE> - <SITUAGEO> - <SITUATION> <DEPARTEMENT>Aude</DEPARTEMENT> </SITUATION> </SITUAGEO> - <CARACT-MAISON> <NBEPIS>4</NBEPIS> </CARACT-MAISON> </GITE> </pre>	<pre> <?xml version="1.0" encoding="UTF-8" ?> - <GITE> - <CARACT-MAISON> <NBEPIS>4</NBEPIS> </CARACT-MAISON> - <SITUAGEO> - <SITUATION> <DEPARTEMENT>Aude</DEPARTEMENT> </SITUATION> </SITUAGEO> - <CAPACITE> <ANIMAUX>yes</ANIMAUX> </CAPACITE> </GITE> </pre>

Figure 11. An extract of the XML document before and after ordering

The experiment is conducted in a semi-automatic context, the results given are better than those achieved by (Alilaouar, 2006) and (Zayani, 2007b).

5. Conclusions and future works

Nowadays, the cognitive overload is one of the main problems encountered. Our contribution consists particularly in making improvements in the adaptation of the contents mechanisms proposed in (Zayani,2007a). Our mechanism is the flexible adaptation of the contents by taking into consideration the user profile and mainly his interests. In fact, we propose, contents which suit to the user's interests raised in his query. Thereafter, this latter will modify to weighted flexible query. This weighting provides more flexible results of search and consultation. These results are dictated by the degree of relevance of the interests of the user.

In prospects, we intend, as a first step, to implement our mechanism of flexible adaptation of contents and to assess the results obtained against other previous works. In a second step, we will consider improving this first version of mechanism, by extending the query taking into account the users interests.

6. References

- A. Alilaouar, F.Sèdes. «A fuzzy model for querying XML documents», *International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU 2006)*, Paris, EDK Editions médicales et scientifiques, pp. 252-259, 2006.
- A. Balla, K.W. Hidouci, N.Ihadadene, A. Hanouh, «Un système d'enseignement adaptatif à base du langage XML », *Information interaction intelligence*, vol. 5, n 1, pp. 9-27, 2005.
- A. Balla, A. Sarirete, «Developing Educational Applications using Adaptive e-Learning Model», livre *Innovative Techniques in Instruction Technology, E-learning, E-assessment, and Education*, Springer Netherlands, pp.13-18, 2008
- B. Boyle, D.Craig, A. O, «Metadoc: An adaptive hypertext reading system », *User Model, User-Adapt Interact*, 4(1),pp.1–19, 1994.
- C. Zayani, I. Amous. «Approche pour l'adaptation de l'interrogation de documents semi-structurés », *Ingénierie des Systèmes d'Information (ISI)*, vol. 12, pp. 61-74, 2007.
- C. Zayani, A. Péninou, M-F. Canut, F. Sèdes, «Towards an adaptation of semi-structured document querying », *International Conference on Modeling and Using Context - International Workshop on Context Based Information Retrieval (CIR 2007)*, Roskilde University, British Computer Society, 2007.
- I. Amous, A. Jedidi , F.Sèdes, «A contribution to multimedia document modeling and querying», *Multimedia Tools and Applications (MTAP)*, *Kluwer academic publishers*, vol. 25 n° 3, pp. 391-404, 2005.
- N.Koch, « Software Engineering for Adaptive Hypermedia Systems Reference Model », *Modeling Techniques and Development Process*, Thèse de doctorat, Tag der mündlichen Prüfung, 2000.
- P.Brusilovsky, «Methods and Techniques of Adaptive Hypermedia», *User Modeling and User Adapted Interaction*, vol. 6, n° 2-3, pp. 87-129, 1996.
- S. Abiteboul, « Querying semi-structured Data », *6th Conference on Database Theory (ICDT'97)*, Invited Talk, LNCS N°1186, pp. 1-18, 8-10, 1997.