An Approach to Query-based Adaptation of Semi-structured Documents

Corinne Amel Zayani[1, 2] and Florence Sèdes[1, 2]

[1] IRIT, 118 route de Narbonne, 31062 Toulouse cedex 4, France
[2] LGC, 129 A, avenue de Rangueil B.P 67701, 31077 Toulouse cedex 4, France

Abstract Semi-structured documents are characterized by flexible and heterogeneous structure and content. So querying this type of documents is required to deliver easily relevant results. Despite the differences between users characteristics (interests, preferences, etc), they receive the same results delivered by the same query. So, in this paper we propose to integrate adaptation process in upstream of the querying step that consists in enriching user's queries by user's characteristics. The adaptation process aims to optimize the pertinence of results according to user requirements and offer users different results for his same query.

1 Introduction

Semi-structured documents have flexible and heterogeneous structure and content. The querying of these semi-structured documents is required to deliver easily relevant documentary units [1], [2]. Several approaches for querying semi-structured documents have been already proposed with the unique characteristics of relevance [2], [3]. Thus, users receive the same results delivered by the same query, despite the differences between their characteristics (interests, preferences, etc).

Thus, we suggest taking into account user's history according to his queries. This history that represents user's characteristics must be introduced in the user profile. On the other hand, we propose to enrich the user's queries by his histories in order to optimize the results according to the user's characteristics (interests, preferences, etc.). We consider that query enrichment introduces the adaptation process in the upstream of the query. The goal of this paper is to present how adaptation process can contribute relevant results delivered by querying process.

This paper is organized into four sections. In section two we present the architecture that combines the relevance and adaptation processes. On the other hand, we present respectively the querying process for semi-structured documents and user profile that plays a significant role in querying. In section three, we explain the upstream adaptation process by an example. We conclude with a discussion of related work.

2 Our Architecture

In figure 1 we illustrate our view of the architecture that is extended from a previous proposal in [4] encompasses two levels of research works:

− Works that offer more relevant documentary units, following the mechanisms of querying of semi-structured documents, such as [2]. But the authors don't take into account the user's history.

− Works that focus on the adaptation of full documents, while taking into account the user's history.

Therefore our contribution via this architecture consists in gathering these two complementary research works, in order to find the better relevant documentary units and to adapt them to the user.
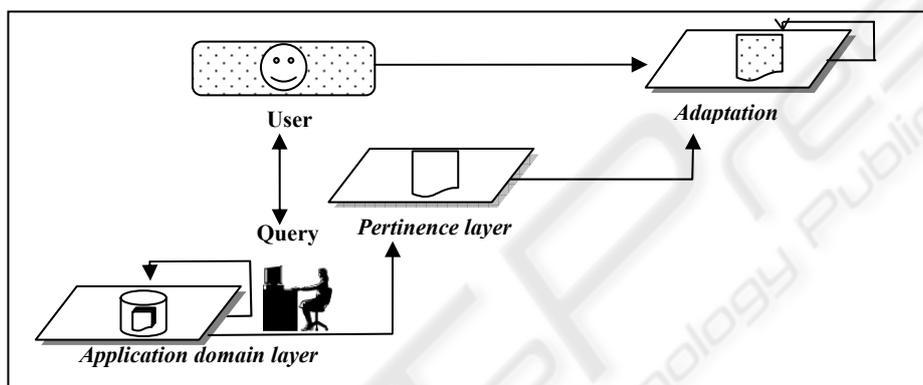

Fig. 1. Architecture gathers relevant and adaptation research works.

In our case we distinguish two adaptation processes:

− *Upstream* consists in enriching the query, firstly, and update the user profile, secondly. The aim of the adaptation process in upstream is to optimize the results relevance according to the user's requirements. It enables to offer users different results when they query semi-structured documents with same query.

− *Downstream* consists in applying adaptation techniques (e.g. stretchtext, link sorting, etc.) according to user profile [5]. This adaptation process enables to decrease cognitive overload and solve the lost in hyperspace problem.

In this paper we concentrate on the first adaptation process i.e. the "upstream" one. So, we present respectively the querying process and user profile in following section.

2.1 Querying Process of Semi-structured Documents

The semi-structured documents adopt flexible markup language for describing data, such as HTML, XML, WML, etc. So they have flexible and heterogeneous structure. On the other hand, there are different languages of querying semi-structured document. In this paper we are interested in XQuery [6] that is used for querying XML

documents. The query result is usually a part of the full document structure, i.e. documentary units.

In figure 2 we show a query in version 0, i.e. query that offers some documentary units according to conditions defined in query. This query doesn't take into account the user's characteristics (interest, preference, etc). Therefore, our contribution via this process consists in enriching query in version 0 by the user's characteristics in order to offer most relevant documentary units and, secondly, enriching the query consists in delivering different documentary units for users that queries same query in version 0. Therefore enhancing query in version 0 leads to generate a version 1 of the query.
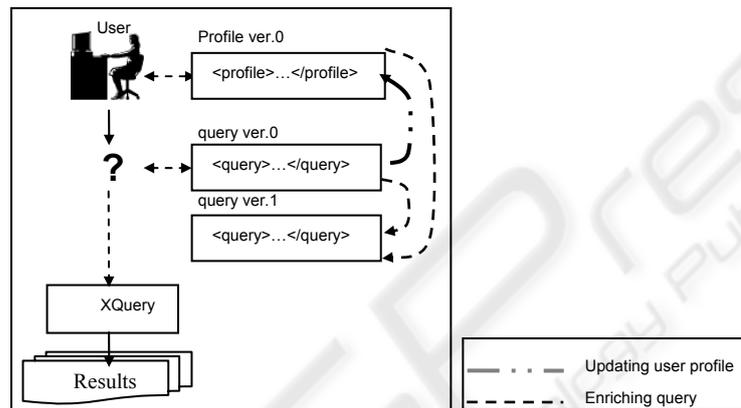

Fig. 2. Querying process.

We also define that user profile stores characteristics about each individual user. This profile is update by each user's query of version "0". After the restitution of results downstream adaptation process [5] can be applied.

2.2 User Profile

We have shown that *upstream adaptation* depends on user profile. This later contains characteristics that can be distinguished into two types [5], [7]:
- *Permanent characteristics* which can be constant over time. This type of characteristics introduces the identity of user (name, etc), demographic data (age, etc.), as soon.
- *Changing characteristics* which can evolve over time. The changing characteristics are different from permanent characteristics because they evolve over time. Generally, changing characteristics are initialized and updated implicitly by observing the user's browsing behavior.

In our research, the user has a browsing behavior while querying the semi-structured documents. So, we suggest that the changing characteristics of user profile can be updated implicitly according to user queries. Thus we define a set of records in the profile. In each record, we define the following attributes: key, value, condition,

result. The condition attribute accepts the value "yes" when the attribute "value" has a value. The result attribute has the value "yes" when the user wants to restore the attribute "key".

3 Example

In this section we propose an example which includes both semi-structured document and user profile. In this example we show how to apply the adaptation process in the upstream of user's query. We show an example of semi-structured document for application mailbox in following XML document.

```
<mailbox login="XX">
  <mailList><mailListReceived>
   … <mailReceived id_mailReceived="1">
     <type>reply</type>
     <date>04/01/06 </date>
     <sender>AA </sender>
     <address><nom> YY</nom><nom> ZZ</nom></address>
     <object>call for paper </object> …
    </mailReceived></mailListReceived>
    <mailListSend>
     <mailSend id_mailSend="1">
       <type>reply</type> <date> 14/01/06 </date>
       <sender> XX </sender>
      <address><nom> YY</nom><nom> ZZ</nom></address>
       <object>RE: </object>…
     </mailSend>…</mailListSend>
<mailList></mailbox>
```

On the other hand, the user's profile is described in the following XML document.

```
<UserProfil Id="1AA1" login="ZZ" password="hh">
  <Record>
  <key>sender</key><value>no</value>
  <condiion>no</condition><result>yes</result></Record>
 <Record>
  <key>object</key><value>multimedia</value>
  <condition>yes</condition><result>no</result>
 </Record> </UserProfil>
```

We suppose that user asks the following query: "retrieve the object of mails received by "XX" ". This query is in a version "0" represented as follow:

```
for $b in doc ("mailbox.xml") //mailReceived
where sender="XX"
return <mail> $b/object </mail>
```

Query 1. Query in version 0.

In this case we follow the querying process of semi-structured documents.
The user's profile can be updated by query 1 that is in version "0" as follows:

```
<UserProfil Id="1AA1" login="ZZ" password="hh">
 <Record >
   <key>sender</key><value>XX</value>
   <condition>yes</condition><result>yes </result>
 </Record>
 <Record >
   …<value> multimedia </value>…
 </Record></UserProfil>
```

On the other hand, the following query in version "1" represents the combination of query in version "0" and user's profile.

```
for $b in doc ("mailbox.xml") //mailReceived
where sender="XX" and object contains "multimedia"
return  <mail>   $b/object  </mail>
```

Query 2. Query in version 1.

This query 2 only returns to the user the objects of mails about "multimedia" received by "XX". But, it is insufficient to release this combination in this case. So, we propose to define two generic functions:

− Function named *adaptToProfile_1* that takes into account the conditions defined in the profile and its key is specified in query. The aim of this function is to show firstly these keys,

− Function named *adaptToProfile_2* aims to show the others keys and conditions don't take into account by precedent function.

Therefore we propose to enhance query 2 by including functions as follows:

```
define function adaptToProfile_1 (element $X)
returns element{
 for $y in ($X)
 where $y contains "multimedia"
 return $y/object}
define function adaptToProfile_2 (element $X )
returns element{
 for $y in ($X)
 where $y not contains "multimedia"
 return $y/object}
for $b in doc ("mailbox.xml") //mailReceived
where sender="XX"
return <mail>adaptToProfile_1($b), adaptToProfile_2($b ) </mail>
```

Query 3. Rewrite query in version 1.

In this case, the result is delivered by the query in version 1 which will be ordered accordingly to results returned by function *adaptToProfile_1*, firstly, and function *adaptToProfile_2*, secondly.

4 Conclusion

In this paper, we propose to complete the research area in semi-structured documents - that is the main interest of our research group - by the adaptation research area via an architecture extended from a previous proposal. The architecture aims at gathering relevant and adaptation research works [2] [5] through which, we have distinguished two adaptation processes: upstream of user's query and downstream from user's query. In order to validate our proposition, a perspective of our work is to develop respectively an algorithm for both adaptation processes: (i) to enrich the user's queries by the user profile that is initialized by permanent characteristics and updated by changing user's characteristics, (ii) to adapt the results according to query enrichment.

References

1. Sèdes, F.: Bases documentaires–Hyperbases Proposition d'un modèle générique et contribution à la spécification d'un langage pour l'intégration et la manipulation d'informations semi structurées. HDR de l'Université Paul Sabatier - Toulouse III, Décembre (1998)
2. Amous, I., Jedidi, A., Sèdes, F.: A Contribution to Multimedia Document Modeling and Querying. Multimedia Tools and Application, 25, 391-404, 2005
3. Albano A., Colazzo D., Ghelli G., Manghi P.: A type system for querying XML documents, in Proc. ACM SIGIR Workshop on XML and Information Retrieval, Athens, Greece, July 28, 2000
4. Zayani, C., A., Sèdes, F., Canut, M.-F., Amous, I., Péninou, A.: Vers une Approche d'Architecture d'un Système Hypermédia Adaptatif. In Proceedings of the IBIMA Internet and Information Technology in Modern Organizations: Challenges & Answers, Cairo, Egypt, 13-15 décembre (2005) 668-672.
5. Brusilovsky, P.: Adaptive hypermedia," User Modeling and User-Adapted Interaction (1:2), Novembre (2001) 87-110.
6. http://www.w3.org/TR/2005/WD-xquery-20050915/ (W3C Working Draft 15 September 2005)
7. Kobsa, A., and Koenemann, J., and Pohl, W.: Personalized Hypermedia Presentation Techniques for Improving Online Customer Relationships. In The Knowledge Engineering Review 16(2) (2001) 111-155.