Quality-Based Recommendation of XML Documents

Laure Berti-Équille

Quality-Based Recommendation of XML Documents

Laure Berti-Equille

IRISA, Campus Universitaire de Beaulieu
35042 Rennes cedex, France
Laure.Berti-Equille@irisa.fr

Abstract. Information quality is multidimensional and can be modeled as an ordered set of weighted criteria. For a collection of XML documents, our approach consists firstly in harvesting and generating information quality indicators and enriching the meta-description of XML documents. Quality metadata are then exploited within the query processing for metadata-driven information retrieval and filtering in order to propose quality-adaptive recommendation strategies (with a quality view as a result) of the queried XML documents. This quality view depends on the user's profile and on his specific information quality requirements. The paper describes the architecture of a quality-based recommender system for XML documents.
Keywords. Information Quality, XML, Metadata, Recommender System.

1. Introduction

Finding relevant, high-quality information in the world wide web or even in a collection of documents is a difficult task. Information quality has no consensual definition and its evaluation requires the measurement and the weighted combination of both objective and subjective quality criteria and, then, the matching between the relative perception of information quality and the users' profile in terms of quality requirements.

In this context, content-based and collaborative recommender systems [RV97] work by automatically recognizing, tallying and redistributing recommendations of the web resources. The multi-confirmed recommendations appear to be significant resources for the relevant community and finally the number of distinct recommenders of a resource is a plausible measure of resource quality. But, many collaborative recommender systems particularly ratings-based systems are built on the assumptions of role uniformity : they expect all users to do the same types of work in return for the same type of benefits. And the notion of user satisfaction and the evaluation task (rating) are very relative and should be considered in a flexible and adaptive way.

Our approach consists in taking into account the information quality evaluations and requirements in the context of collaborative annotation and quality-driven information retrieval and filtering of XML documents. We propose a modeling of document quality with various objective and subjective quality criteria. The objective crite-

ria can be quantitative and are calculated by statistical methods. The subjective criteria are defined and evaluated by a group of reviewers collaborating with the aim of reaching a consensus on the annotation of the documents. The selection of the documents is both content-based and quality-based, i.e. depending on the content and the structure of the documents relevant to the query and depending on the qualitative aspects required by the user. Our objective is to propose a multi-criteria adaptive recommendation of XML documents, and to refine the traditional selection of documents by exploiting metadata on the resource quality which are embedded into or linked to them, computed or generated automatically, or built as a result of a collaboration between individuals. From these specifications, we developed the *XDARE* system (*XML-Documents Annotation and Recommendation Environment*) for quality-driven annotation and recommendation of XML documents.

The rest of the paper is organized as follows: section 2 presents the previous works on information quality and adaptive hypermedia systems. Section 3 describes our quality metadata model and presents the recommendation process for XML documents. Section 4 describes the architecture of our system. Lastly, Section 5 concludes the paper and presents our perspectives of research and development.

2. Related Works

2.1. Metadata and Data Quality

In the context of a distributed information environments generally, metadata harvesting refers to the automatic collection of descriptive information from distributed resources. Recently, one particular way of accomplishing this collection of distributed metadata has been the subject of considerable attention in museums, archives and e-learning communities (e.g. the metadata collection proposed by the *Open Archives Initiative Metadata Harvesting Protocol* (OAI-MHP) [OAI02]). In the domains of geographical information systems [GJ98] (*ISO 15046-13, CEN prEN 287-008, FGDC)* and digital libraries *(Dublin Core, Bib-1, GILS, STARTS, Z39.50 ANSI/NISO, etc.*), most of the exchange standards propose metadata specifications for information quality, which are either automatically extracted or measured by sampling from data sets.

Many research works on information quality also proposed various definitions, conceptual models and methodologies to improve data quality in databases or in information systems [Nau02] [WSF95] [Wan98] [Red96]. The data quality dimensions most frequently mentioned in the literature are: *accuracy, completeness, actuality* and *consistency*. But many others dimensions, metrics and measurement techniques have been proposed in the literature [Red96] [FLR94] [Vas00] [MR00] [BP02] [Nau02] [NFL99] (see Table 1 for a non–exhaustive list of quality dimensions).

Most of the techniques of quality measurement are centered on various methods of imputation such as inferring missing data from statistical patterns of available data, predicting accuracy estimations based on the given data, data editing (automating detection and handling of outliers in data), error control [Red96][FLR94].

Authors	***Quality Dimensions***	
Brodie [Bro80]	6 concepts	*Integrity, Abstraction Level, Semantic Expression Power, Validity, Maintenance, Resource Use Efficiency*
Delen, Rijsenbrij [DR92]	- 4 dimensions - 21 aspects - 40 attributes	*Development and control of Information Systems, Static Properties for maintenance, Dynamic Function,* ***Information importance*** *: correctness of data, completeness, up-to-dateness, accuracy, verifiability*
Wang et al. [Wan98] [WSF95]	- 4 categories - 179 attributes	*Intrinsic Quality, Accessibility Quality, Contextual Quality, Representation Quality*
Redman [Red96]	- 4 dimensions for values - 8 dimensions for representation format	*- Accuracy, completeness, actuality, consistency* *- Appropriateness, interpretability, portability, format precision, format flexibility, null value, efficient use, consistency*
QCT-TIPS [TIPS99]	8 quality features for document quality	***Scientific quality****: correctness, completeness, originality, accuracy, currency, quality of demonstration, quality of references list, quality of methodology ;* ***Readability****: quality of the writing style, quality of the logical structure, ad-equation of illustrations, absence of repetitions, clarity of expression of ideas ;* ***Intended audience****: technical or educational level ;* ***Recency:*** *publication date* ***Authority****: reputation of author, reputation of journal or conference ;* ***Availability****: durability, printability ;* ***Popularity****: citing/reading popularity ;* ***Quality of identification****: citability*

Table 1. ***Some definitions of information quality***

The *DESIRE Project* [HBP00] also produced a detailed list of quality standards to be used for the selection of the Web resources with various categories of quality criteria: 1) criteria related on the policy of diffusion and the range of the resource, 2) criteria related to the content, 3) criteria related to the form, 4) criteria related to the management of the documentation quality.

In the context of the *TIPS European project* [TIPS99], several services have been developed related to the reuse of evaluations performed by humans on scientific publications. The first one, called *QCT* (*Quality Control Tools*) aims at collecting human detailed evaluations of documents in order to enrich the traditional topical indexing of documents with quality-related information (see Table 1 for the quality features used in the *QCT-TIPS* project for document quality). The second one, called *SF* (*Social Filtering*) integrates a push functionality as a alternate and complementary tool to traditional pull services such as information retrieval ; documents are pushed to users with respect to the evaluations they have made in the past, and compared to other users' evaluations.

Actually, several dimensions and metrics of information/resource quality have been proposed, but no methodologies give specific or pragmatic solutions concerning:

- what are the good/appropriate quality metrics to use,
- how to measure/collect these quality metrics,
- how to verify their goodness.

A bench mark for metadata quality assessment would be an important and useful effort for the research progress in this area.

2.2. Adaptive Hypermedia and Recommender Systems

Recently a number of adaptive hypermedia systems have appeared as impersonalized systems, recommender systems |RV97] with a common goal: to learn about the implicit preferences of individual users and to use this information to serve the entire community of these users better. The early recommender systems mainly used Information Filtering (IF) techniques and individual previous behavior to produce recommendation. The major drawback of IF techniques is that they do not provide much in the way of serendipitous discovery. To cope with this drawback, Collaborative Filtering (CF) techniques have been proposed in order to recommend items based on the opinion (rating) of other users who have similar tastes. *GroupLens* [RI94] is a server-side recommendation engine for Usenet news. A user's profile is created by recording the user's explicit ratings of various articles. Automatic collaborative filtering is used to statistically compare one user's likes and dislikes with another user and to recommend articles from other similar users profiles. Various recommender systems have been created to assist users for selecting potentially interesting information and for filtering out what users may not be interested in (*PHOAKS* [TH+97], recommendation of Web resources mined from Usenet news messages, Ringo [SM95], a music recommender system). But two major limits of the CF-based techniques are:

- the "*early-rater*" problem occurring for the first rating of documents without benefit of other previous recommendations,
- the "*sparsity rating*" problem occurring when the overlap between user's ratings (or number of co-rated items) is small or null and as a consequence that the recommendation results may be not accurate or cannot be produced.

The next level of recommender system is hybrid systems combining IF and CF techniques, such as *MovieLens* [GS+99], a movie recommender system using filterbots (IF agents) as rating robots which participate as members of the CF system. Both *Personal Web Watcher* [Mla96] and *Letizia* [Lie95] are content-based systems that recommend web-page hyperlinks by comparing them with a history of previous pages visited by the user. *Personal Web Watcher* [Mla96] uses an offline period to generate a bag of words style profile for each of the pages visited during the previous browsing session. Hyperlinks on new pages can then be compared to this profile and graded accordingly. *Letizia* [Lie95] uses the unused online time when the user is actually reading a web page to search the adjoining hyperlinks. The user's profile is composed of keywords extracted from the various pages visited.

Most of the current hybrid systems still use co-rated items among users in finding correlated neighbors for an active user, and co-rated items between user and filterbot to find agreed filterbots.

On the opposite, PICS (*Platform Content for Internet Selection*) [RM96] is an infrastructure which associates labels to the contents of the documents available on the Internet in order to block the access for non-authorized users. Originally conceived to help the parents and the professors to control the navigation of their children/pupils on the Internet, it makes it possible to affect any criterion on the labels which the system interprets to authorize or block the access to the documents. Complementary to the recommendation, the advantage of this approach is that the structure of a document can be enriched by adding labels which define the conditions of viewability (blocking or full access).

SYSTEM	TYPE	MODEL
PFIT, *ifWeb* [AT97]	IF system	Bayesian networks Query-based dialogue with user
Information Lens System [MGT+87]	IF system	Rule-based construction of templates by the user
Syskill & Webert [PMB97]	IF Agents	Naïve (simple) Bayesian Classification on Boolean vectors, Nearest Neighbors, Symbolic Profile Learning on web page of interest for the user and link suggestion using decision trees
Amalthaea [Mou96]	IF Agents Discovery Agents	Economic Model with an ecosystem of agents
FAB [RV97][Bal97]	IF Agents	Hybrid recommender system
GroupLens [RI94]	CF system	
PHOAKS [TH+97]	CF system	
Personal Web Watcher [Mla96]	CF/IF system	Naïve (simple) Bayesian Classification on Boolean frequency vectors
Letizia [Lie95]	CF/IF system	Hybrid recommender system Inference from the user's browsing behavior

Table 2. ***Information Filtering and Collaborative Filtering Systems***

Another aspect of information personalization is to adapt web content to users' preferences and also to the variations of the client environment, so that web pages can be prepared suitable for the client. Adaptive hypermedia systems [BSS00] can learn about the implicit and explicit preferences of individual users and using this information to personalize information retrieval processes. In this context many adaptive hypermedia systems have been proposed, such as *OnlineAnywhere*[1], *SpyGlass*[2], *FastLane*[3], *QuickWeb*[4], *ProxyNet*[5], *Digestor* [BS97], *TranSend* [FG+98], and *Mobiware* [AC+98]. However, most of them only make adaptation of the web content under special conditions due to the lack of structural information of HTML content, and many of them focus on image conversion.

2.3. Motivation

Among these various research propositions concerning on one hand, information quality, and on the other hand, recommender, blocking and adaptive hypermedia systems,

[1] OnLineAnyWhere, FlashMap, http://www.onlineanywhere.com
[2] Spyglass, "White Paper of Prism 2.2", http://www.spyglass.com/images/Prism22.pdf
[3] FastLane, http://stage.acunet.net/spectrum/index.html
[4] QuickWeb, http://www.intel.com/quickweb
[5] ProxiNet, ProxiWare, http://www.proxinet.com/products_n_serv/proxiware/

we came to the point that there is no proposition in the literature nor system that combines resource quality for alternatively recommending/blocking and adapting digital resources on demand (in a way driven by users' profiles). Actually, in the current research works in IF or CF, the quality of the content is not considered as a key element for the users' decision in the recommendation process and we think this problem is not yet sufficiently addressed by existing approaches. Our motivation was to propose the three services (blocking/recommending and adapting information) in a flexible way and to consider the quality dimensions of the queried documents. We retained several principles which guided our approach:

- to improve quality of a search result, it is necessary to evaluate the quality of the documents and information and to exploit it in the query processing,
- it's necessary that the definition of the document quality remains flexible and the use of labels and metadata allow this flexibility for both specification and interrogation.

Compared to existing approaches for collaborative or content-based recommender systems, the innovative aspect of our approach is to include constraints and requisites on content quality that is complementary for best recommendation services.

3. Modeling Information Quality and Processing Quality Metadata

3.1. Metadata for Document Quality

The choice of modeling for document quality has been made in order: 1) to allow a rigorous but flexible definition of each dimension of the quality of documents, as its measurement protocol, 2) to re-use the existing standards of metadata proposed in the literature including elements of quality, 3) to propose an assistance to the user who can be a reviewer into the collaborative annotation process, and so, should be able to define and evaluate himself the quality of the documents choosing and combining several specific programs for the generation/extraction of metadata, 4) to propose an assistance to the user who's searching high quality information with providing him an adaptive recommendation (as a quality view) of the retrieved documents depending both on his query and also on his quality requirements.

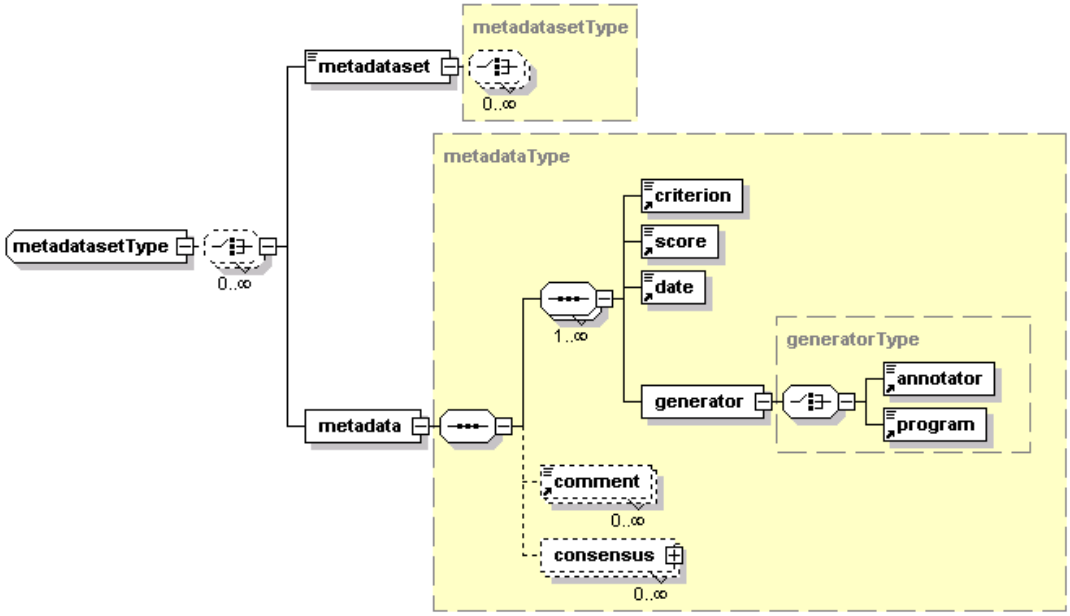
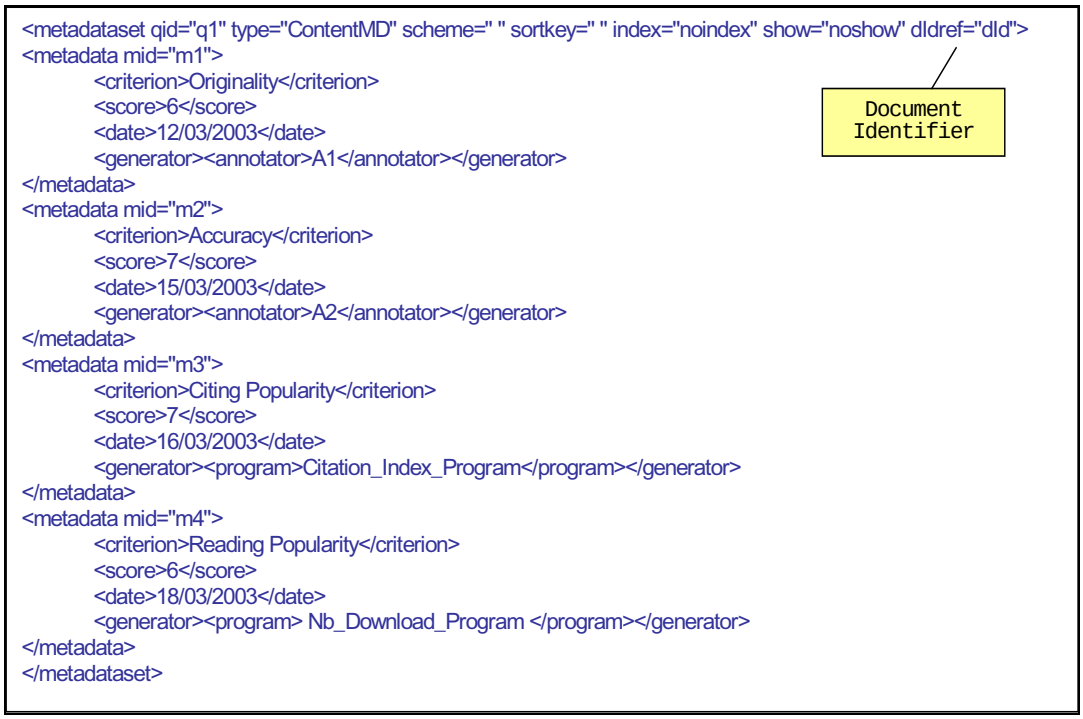


Figure 1. *Metadata description*

The quality of a document is defined by combining quantitative measurements (computed by the system) and qualitative evaluation made by one (or several) reviewer(s) ; these metadata stored in XML files are embedded or linked to the content of the XML documents. As Figure 1 shows, the metadata type associated to a document (metadatasetType) can be a set of metadata (metadataset) or one metadata (metadata) which is composed of a criterion, a scoring value for the criterion given by a human reviewer (annotator) or computed by a program (program), a creation date and a comment (comment). A consensus can be calculated for a given criterion and a date if several notations have been proposed by several reviewers. The instantiation of the quality standards for a document can be based on an adaptation of the non-exhaustive list of criteria given in [WSF95] [Red96][MR00][TIPS99].

Example 1. Figure 2 gives an example of quality metadata instances that can be associated to a document with both subjective criteria (originality, accuracy) and objective criteria computed by specific programs (citing popularity, reading popularity). In this example, the notations are values in [0,10] ; the originality and the accuracy of the document are evaluated by the reviewer A1 and the citing and reading popularity computed by programs similar to the one used in *CiteSeer* (http://citeseer.nj.nec.com/).




```
<metadataset qid="q1" type="ContentMD" scheme=" " sortkey=" " index="noindex" show="noshow" dIdref="dId">
<metadata mid="m1">
        <criterion>Originality</criterion>
        <score>6</score>
        <date>12/03/2003</date>
        <generator><annotator>A1</annotator></generator>
</metadata>
<metadata mid="m2">
        <criterion>Accuracy</criterion>
        <score>7</score>
        <date>15/03/2003</date>
        <generator><annotator>A2</annotator></generator>
</metadata>
<metadata mid="m3">
        <criterion>Citing Popularity</criterion>
        <score>7</score>
        <date>16/03/2003</date>
        <generator><program>Citation_Index_Program</program></generator>
</metadata>
<metadata mid="m4">
        <criterion>Reading Popularity</criterion>
        <score>6</score>
        <date>18/03/2003</date>
        <generator><program> Nb_Download_Program </program></generator>
</metadata>
</metadataset>
```

Figure 2. *Example of quality metadata linked to the document identified by "dId"*

3.2. Annotation and Recommendation Process

The process of collaborative annotation and recommendation of the documents can be decomposed into four main operations (Figure 3): 1) harvesting and generating the quality metadata (annotating), 2) indexing the documents and their meta-descriptions (indexing), 3) matching the query (including the quality requirements of the users) with the representation of the indexed documents (matching), and 4) scoring and viewing the documents (viewing) with different recommendation strategies.

In this process, each step constitutes an elementary operation on the XML document collection. First, at the annotation step, each document of the collection is enriched by several quantitative and qualitative quality metadata (respectively, objective measures and subjective evaluations of the document quality) such as, in the example of Figure 2. Metadata, document contents and structures are then used for the indexing process in order to represent each document as a multidimensional vector on these three axes : quality (criteria), content (terms) and structure (XML elements, attributes).

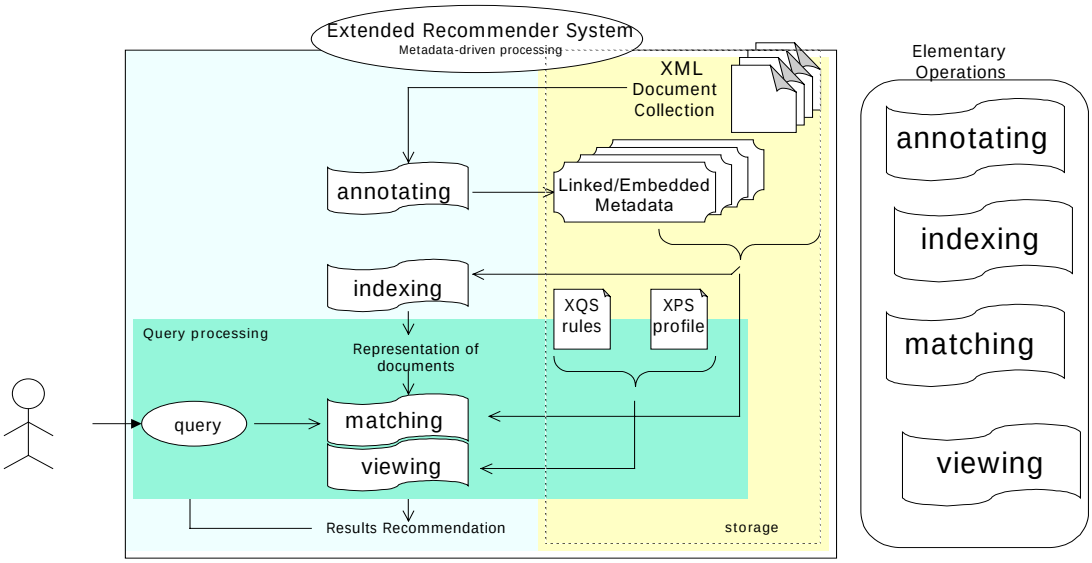


Figure 3. *Operations for Retrieving Recommended XML Documents*

For processing a query, the system finds the documents that best match the query terms and it applies to their respective metadata :

- the quality rules specified into the XQS file (*XML Quality Style sheet*)
- and the quality requirements specified into the XPS file (*XML Profile Style sheet*) that correspond to the profile of the user (or of the group of users).

At this stage, the system has to use the recommendation strategy defined into the XQS file, to verify and to adapt (i.e. soften) the quality constraints given into the XPS profile and for building the quality-driven view of the query result.

Definition 1. XML Quality Style Sheet (XQS)

A quality style sheet (XQS) (whose DTD is given in Figure 4) is a file used in the query processing that references the profile to use and the rules that have to be applied to each document node for each user profile ; each rule applied to a document node according to a user profile has a priority (between 1-lowest and 5-highest priority) and defines the access mode (access without recommendation (default), with recommendation or blocking access) and the strategy for verifying the quality constraints on the effective document quality (exact, approximate or negotiated). The exact strategy means that all the constraints must be verified on the value of every quality criteria (i.e., metadata values) of the targeted document node ; the approximate strategy allows the approximate and flexible matching between the constraints values and the effective quality criteria values (with nearest neighborhood) ; the negotiation strategy allows to soften interactively the quality constraints in order to match the document nodes that answer the query with the best quality.

```
<!ELEMENT xqs (rule*)>
<!ATTLIST xqs
  DefaultProfileFile CDATA #FIXED "profiles.xps">
<!ELEMENT rule EMPTY>
<!ATTLIST rule
  Profile CDATA #REQUIRED
  DocumentNode CDATA #REQUIRED
  Priority CDATA #REQUIRED
  Access (default | recommending | blocking) #REQUIRED
  Strategy (default | exact | approximate | negotiated) #REQUIRED>
```

Figure 4. *DTD for XQS sheet*

Example 2. Quality Style Sheet Example

In Figure 5, four rules are defined in the quality style sheet file concerning respectively three different users : the chief of the editorial board of a scientific journal, the secretary and the authors who submitted a paper for the special issue of the journal. The DTDs of the journal and the metadata set are given in Figure 8. These users will be allowed (or not) to access information according to the following rules :

R1 – the secretary access all the submitted articles ;

R2 – the authors only access their own article;

R3 – the authors are not allowed to access the papers and reviewers' comments of other authors ;

R4 - the editorial chief access the best submitted papers.

```
<xqs DefaultProfileFile="profiles.xps">
  <!-- Rule 1 -->
  <rule Profile="users/secretary"
        DocumentNode="article[@id=$user]//node()"
        Priority="2" Access="default" Strategy="exact"/>
  <!-- Rule 2 -->
  <rule Profile="users/*[name()='Authors']"
        DocumentNode=" article[@dId=$user]//node()"
        Priority="5" Access="default"  Strategy="exact"/>
  <!-- Rule 3 -->
  <rule Profile=" users/*[name()='Authors']"
        DocumentNode="article[@dId!=$user]/@qIdref//node() |
                      article[@dId!=$user]//node"
        Priority="5" Access="blocking" Strategy="default"/>
  <!-- Rule 4 -->
  <rule Profile="users/EditorialChief"
         DocumentNode="article//node() | article/@qIdref//node()"
         Priority="5" Access="recommending" Strategy="negotiated"/>
</xqs>
```

Figure 5. *Example of XQS sheet*

Definition 2. XML Profile Style Sheet (XPS)

A profile style sheet (XPS) is a file that references the user or group of users of the profile and defines the constraints on the document quality dimensions. The set of quality constraints is defined as a quality contract.
For a compact and simplified presentation, Figure 6 shows an extract of XPS in the BNF-style grammar and an example corresponding to the user profile of the Editorial Chief considering the quality metadata given previously in Figure 2.

profile ::=	PROFILE OF users { requisites }
users::=	users member | member
member::=	user_name
user_name::=	literal
requisites ::=	requisites requisite
requisite ::=	REQUIRE contractList
contractList ::=	contractList , contractElem | contractElem
contractElem ::=	contractDefintion
contractDefinition ::=	CONTRACT { constraints }
constraints ::=	constraints constraint | constraint
constraint ::=	dimName constraintOp dimValue | dimName { aspects }
constraintOp ::=	== | >= | <= | > | < | LIKE | !=
dimName ::=	literal
dimValue ::=	literal unit | literal
aspects ::=	aspects aspect
aspect ::=	NUMBER constraintOp dimValue | constraintOp dimValue | freqRange constraintOp NUMBER %
freqRange ::=	dimValue | IN lRangeLimit dimValue , dimValue rRangeLimit
lRangeLimit ::=	[| (
rRangeLimit ::=	] |)

PROFILE OF EditorialChief
{**REQUIRE** CONTRACT
{Originality > 6 ;
Accuracy > 6 ;
Citing_Popularity > 7 per year ;
Reading_Popularity in [6,9] per year} ;
} ;

Quality Contract of the Editorial Chief

Figure 6. *Extract of the XPS grammar and an example in the BNF-style*

Example 3. Profile Style Sheet Example
The example in Figure 6 presents the constraints required by the editorial chief on the document collection for what concerns the originality, the accuracy, the citing and reading popularity of the articles. In particular, this user is interested in documents with a certain level of originality and accuracy (higher than 6 in the interval [0,10]), with citing popularity higher than 7 and reading popularity between 6 and 9. Figure 6 presents the user profile in a BNF-style grammar for simplification but, actually, the user profiles are stored in XML-files using the RuleML[6] DTD.

3.3. Recommending High-Quality XML Documents

The key elements of the recommendation process are the quality-based recommendation rules (previously presented in XQS file see Figure 4) and the quality view that is generated according to two main heuristics.

[6] RuleML, http://www.dfki.uni-kl.de/ruleml/

Definition 3. Quality-based Recommendation Rule

A quality-based recommendation rule is the following quadruplet : < *Profile, DocumentNode, Access, Strategy* > with :

- *Profile* : the path expression related to the profiles of the XPS style sheet,
- *DocumentNode* : the Xpath expression evaluated inside the targeted XML document,
- *Access* : the value for recommending or for blocking the access to the information items (as document nodes),
- *Strategy* : the multi-criteria selection algorithm used to recommend the targeted node of the document. This algorithm uses Multiple Attribute Decision Making (MADM) approach [KZ00] that will not be discussed in the paper.

The semantics of the quality-based recommendation rule is not unique and depends on the type of the considered node (element, attribute, text...). Several integrity constraints may be defined in order to maintain consistency between the rules.
Instead of recommendation of entire documents based on general quality requirements, we suggest the approach of filtering XML-documents based on quality criteria in the personal profile of the requestor.

Definition 4. Quality View

A quality view of a document is the result (as the fragment(s) of the XML document) that corresponds to the query and satisfies the quality constraints and rules defined in the user's profile. Two heuristics are used in order to build the quality view of each retrieved document in conformance with the quality constraints and the recommendation strategy chosen for the user who sent the query to the system. The quality view of the XML document is built node by node (i.e., XML element by element).

Heuristic 1. If the access to a node *n* of a document is allowed for recommendation for a user *u,* then *u* can see the recommended sub-tree of the XML document whose *n* is the root node if it satisfies the quality constraints defined as quality requisites into the user profile with the chosen strategy (exact or approximate or negotiated recommendation for quality requirements).

Heuristic 2. If the access to a node *n* is forbidden for a user *u,* then *u* cannot see the sub-tree of the document whose *n* is the root node.

The algorithm given in Annexes (Figure 10) is used to build the quality views according to:

- the quality-based recommendation rules given in the XQS file,
- and the user's profiles given in the XPS file.

Example 4. Query

Following the Example 2, consider the edition of a scientific journal and the associated review comments of the submitted papers. The reviewers comments are stored as metadata files. The content of the special issue of the journal has the following DTD (Figure 7) :

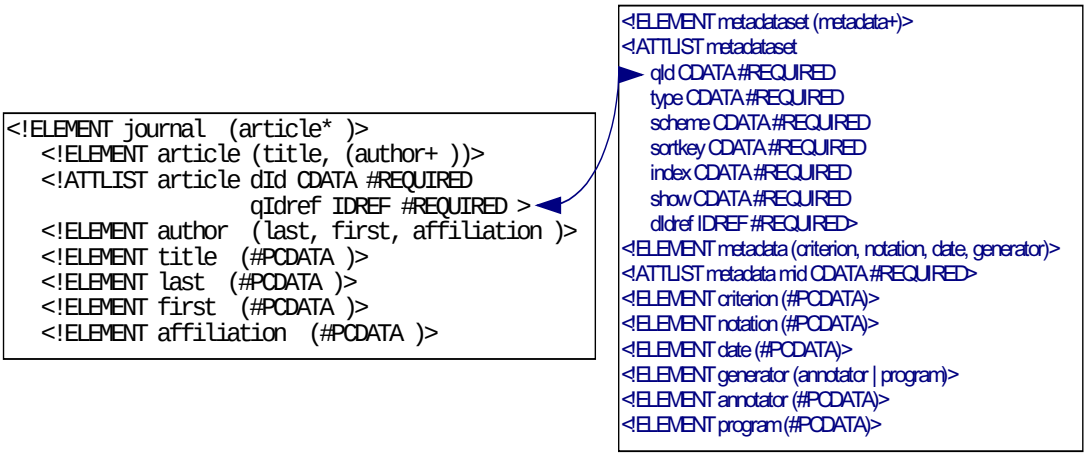


Figure 7. *The DTDs of the journal example and the reviewers comments*

Suppose the following query sent by the three users : the secretary, the authors of submitted papers and the editorial chief in order to get as a result XML element the name of authors, the title and the reviewers' comment of the submitted papers to the journal (Figure 8). The query language used is XQuery[7] [FM01] [MM+01].

```
<results>
  {for $b in document /journal/article,
           $a in $b/author,
           $t in $b/title,
           $r in document($b/@qIdref)/metadataset
           return
           <result>
                { $a }
                { $t }
                { $r }
           </result>}
</results>
```

Figure 8. *Example of query*

Result 1. Secretary's quality view. This query sent by the secretary will create as a result a flat list of all the author-title pairs of the submitted paper. Her quality requirements are given in the XQS file of Figure 5 without recommendation.

Result 2. Author's quality view. This query sent by an author will show only the title and the author name and the reviewers' comments of his own submitted article, but this author will not be allowed to see the papers and the comments of other authors.

Result 3. Editorial chief quality view. This query sent by the Editorial Chief will show the title, the author names and the reviewers' comment of the best articles.

[7] XQuery, http://www.w3.org/TR/xquery-semantics/

4. System Architecture

4.1. Document and Quality Processing

We used standard tools for implementing the quality recommendation processor of our system *XDARE* (*XML-Documents Annotation and Recommendation Environment*). The prototype has been developed in Java and implement the following operations (Figure 9):

1 **XML Document Analysis:** The document file is parsed (*Xerces Apache*) and syntactically analyzed (*SAX API*) and the corresponding internal representation is created and stored. The internal representation and manipulation has been developed with *DOM API*: the events produced by the *SAX API* during the first step of the document analysis are used to build the corresponding *DOM* tree. The *XDARE* operators use the instances of this internal representation by recopy and each query produces a new tree.

2 **Query and Xpath Analysis:** The grammar is an extended XML Query syntax. *Jlex* and *Cup* are here used to produce the corresponding Java syntactical analyzer including the Xpath expression analysis. The query tree is explicitly instantiated for future optimization.

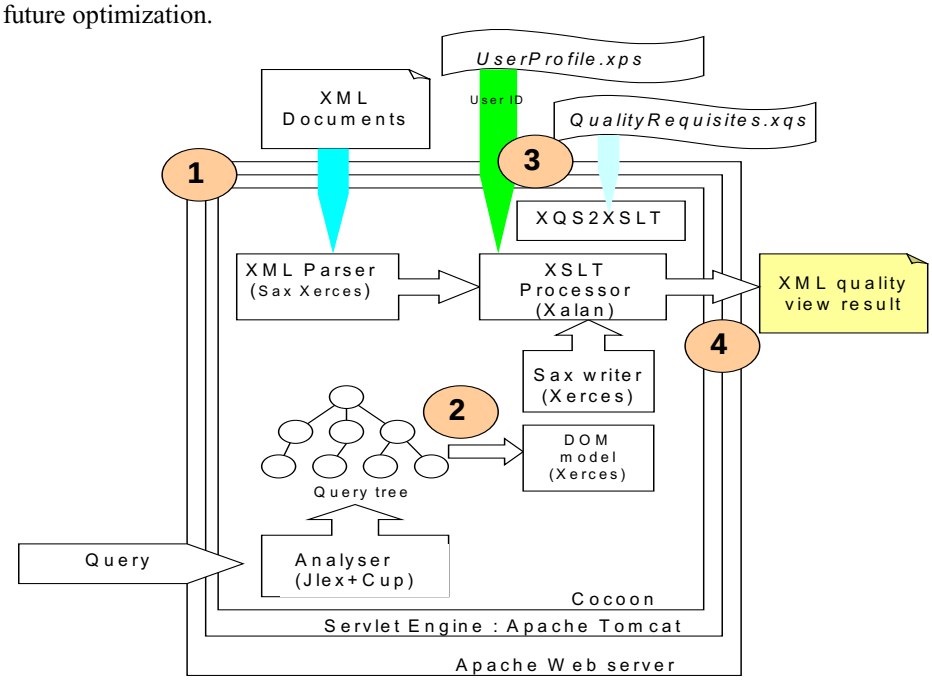


Figure 9. *Quality Recommendation Processor of XDARE from the document processing perspective*

③ **XML Document Quality Control:** In the Cocoon architecture, the *Xalan* processor applies a XSLT style sheet to the XML document. Our prototype transforms the XQS style sheet (including the quality requisites) into a set of XSLT templates. The application of XSLT style sheets enables the creation of quality views corresponding to user quality profile (defined into the XPS file).

④ **Quality View Generation:** When the user wants to browse a XML document, he actually obtains a view of this document in conformance with his quality requirements.

4.2. Quality-based Recommendation Processor

Our quality processor is based upon the Cocoon java servlet. The initial documents and the quality recommendation rules (XQS) and the user profiles (XPS) are in XML format. The quality view generated in conformance with the user profile is produced with XSLT sheet that is generated from the XQS sheet using XSLT. Cocoon enables the automatic processing shown in Figure 10. An identified user asks for the access to a document *d* to the XDARE server. The document *d* refers to a XQS sheet. If the XSLT sheet corresponding to the translation of the XQS sheet is not yet in the cache, it is produced dynamically. Otherwise, the XSLT sheet is applied to *d* for producing the view of the document according to the user quality profile (XPS). Cocoon utilization offers lots of advantages : in particular the multi-level cache system. This mechanism enables to process only once the transformation from XQS to XSLT sheet. Moreover, the produced views are also put into the cache memory and the multiple access to the same filtered document quality doesn't overload the server. On the other hand, views (that are also XML documents) can be naturally retransformed with XSLT in order to be produced in HTML or PDF formats.

5. Conclusion and Perspectives

In this paper, we suggest a general architecture for quality-based recommendation of XML documents. The document quality is modeled as a set of (criteria, value) pairs collected in metadata sets, which are associated with XML documents.

A metadata schema is proposed, capable of capturing arbitrary (criterion, value) sets along with general metadata such as date and generator of the (criterion, value) assignment. We identify four basic operations to achieve quality recommendation: 1) annotation with metadata describing the documents quality, 2) indexing the documents, 3) matching queries, and 4) viewing the recommended parts of the documents. The quality requirements of each user are kept in individual user profiles (so called XPS files).

Every XML document in the document base refers to a quality style sheets (so called XQS files) which allow for specification of several matching strategies and contain matching rules relating parts (sub-trees) of XML documents to user profiles. An algorithm is described for evaluation of the quality style sheets and user profiles in

order to build an "adaptive quality view" of the retrieved XML document. Finally, we sketched the architecture of the XDARE system, which implements the proposed data structures and algorithm to support quality-based retrieval and adaptive quality views of XML documents. The system is built on standard Java components for XML processing (such as XERCES and XALAN).

The specification and the exploitation of metadata describing the quality of documents can improve the system effectiveness for information filtering and recommending. The user profiles may include requirements for quality of results.

The innovative aspect of our work is to propose the three services (blocking/recommending and adapting information) in a flexible way and to combine content-based and quality-based recommendation with considering the quality dimensions of queried XML documents.

The perspectives of our work mainly concern now: the automatic learning of user profiles (and quality requisites) by inductive logic programming (ILP module for user profiling) and the operational validation of our *XDARE* system with communities of users.

References

[AC+98] Angin, O., Campbell, A.T., Kounavis, M. E., Liao, R.-F., The Mobiware Toolkit: Programmable support for adaptive mobile networking. IEEE Personal Communication, 5(4):32-43, 1998.

[AT97] Asnicar F.A., Tasso C., ifWeb : a prototype of user model-based intelligent agent for document filtering and navigation in the Word Wide Web, Proc. of the 6th Intl. Conf. on User Modeling (UM'97), 1997.

[Bal97] Balabanovic M. , An adaptive Web page recommendation service, Proc. of the 1st Intl. Conference on Autonomous Agents, 1997.

[Bro80] Brodie M.L., Data quality in information systems, Information and Management, 3, 1980.

[BP02] Ballou D.P., Pazer H., Modeling completeness versus consistency tradeoffs in information decision contexts, IEEE TKDE, 15(1):240-243, 2002.

[BS97] BickMore T.W., Schilit B.N.: Digestor: Device-independent access to the World Wide Web, Proc. of the 6th International World Wide Web Conference, pages 655-663, 1997.

[BSS00] Brusilovsky P., Stock O., Strapparava C. (Eds.): Adaptive Hypermedia and Adaptive Web-Based Systems, Proc. of the International AH'2000Conference, LNCS 1892, Trento, Italy, August 2000.

[DR92] Delen G., D. Rijsenbrij D., The specification, engineering and measurement of information systems quality, Journal of Software Systems, 17, pages 205-217, 1992.

[FG+98] Fox A., Gribble S.D., Chawathe Y., Brewer E.A.: Adapting to Network and Client Variation Using Infrastructural Proxies: Lessons and Perspectives. IEEE Personal Communication, 5(4):10-19, 1998.

[FLR94] Fox C., Levitin A., Redman T., The notion of data and its quality dimensions, Information Processing and Management, vol. 30, no. 1, 1994.

[FM01] Fernandez M., Marsh J. XQuery 1.0 and XPath 2.0 Data Model. W3C Working Draft 2001. http://www.w3.org/TR/query-datamodel/, 2001.

[GJ98] Goodchild M., Jeansoulin R. (Ed.), Data quality in geographic information : from error to uncertainty, Hermès, 1998.

[GS+99] Good N., Schafer J., Konstan J., Borchers A., Sarwar B., Herlocker J., Riedl J., Combining Collaborative Filtering with Personal Agents for Better Recommendations. Proc. of the 1999 Conf. of the American Association of Artifical Intelligence (AAAI-99), pages 439-446, 1999.

[HBP00] Hiom D., Belcher M., Place E., People Power and the Web: Building Quality Controlled Portals, TERENA Networking Conference, 2000. http://www.desire.org/

[KZ00] Koksalan M, Zionts S. (Ed.), Multiple Criteria Decision Making in the New Millenium: Proc. of the 15th Intl. Conf. on Multiple Criteria Decision Making (MCDM), Ankara, Turkey, Lecture Notes in Economics and Mathematical Systems, 507, 2000.

[Lie95] Letizia: An Agent That Assists Web Browsing, Proc. Of the Intl. Joint Conf. Artificial Intelligence, Montreal, August 1995.

[Mla96] Mladenic, D., Personal WebWatcher: Implementation and Design, Technical Report IJS-DP-7472, Department of Intelligent Systems, J.Stefan Institute, Slovenia, 1996.

[MGT+97] Malone T., Grant K., Turbak F., Brobst S., Cohen M., Intelligent information sharing systems, Com. of the ACM, 30(5):390-402, 1997.

[MM+01] Malhotra A., Marsh J., Melton J., Robie J. (eds): XQuery 1.0 and XPath 2.0 Functions and Operators Version 1.0. W3C Working Draft 2001. Available at: www.w3.org/TR/xquery-operators/

[Mou96] Moukas A., Amalthaea : Information discovery and filtering using a multi-agent evolving ecosystem, Proc. of the Practical Application of Intelligent Agents and Multi-Agent Technologies (PAAM'96), 1996.

[MR00] Mihaila G. A., Raschid L., and Vidal M.-E., Using quality of data metadata for source selection and ranking. In Proc. of the WebDB'00 Workshop, pages 93-98, 2000.

[Nau02] Naumann F., Quality-Driven Query Answering for Integrated Information Systems, LNCS 2261, Springer 2002.

[NLF99] Naumann F., Leser U., and Freytag J., Quality-driven integration of heterogeneous information systems. In Proc. of the 25th VLDB Conference, 1999.

[OAI02] The Open Archives Initiative Protocol for Metadata Harvesting (Version 2.0), 2002. http://www.openarchives.org/OAI/2.0/

[PMB97] Pazzani M., Muramatsu J., Billsus D., Syskill & Webert : identifying interesting Web sites, Proc. of the 13th National Conference on Artificial Intelligence, 1997.

[Red96] Redman T., Data quality for the information age, Artech House Publishers, 1996.

[RI94] Resnick P., Iacovou N., Suchak M., Bergstrom P., Riedl J. GroupLens: An Open Architecture for Collaborative Filtering of Netnews. Proc. of 1994 Conf. on Computer Supported Collaborative Work, pages 175-186, 1994.

[RM96] Resnick P., Miller J., PICS: Internet access controls without censorship. Com. of the ACM, 39(10):87-93, 1996. http://wwww.w3.org/PICS

[RV97] Resnick P., Varian H., Recommender systems. Com. of the ACM, 40(3):56-89, 1997.

[SM95] Shardanand U., Maes P., Social Information Filtering: Algorithms for Automating "Word of Mouth". Proc. of ACM CHI'95 Conf. on Human Factors in Computing Systems, pages 210-217, 1995.

[TH+97] Terveen L., Hill W., Amento B., McDonald D., Creter J., PHOAKS: A System for Sharing Recommendations. Com. of the ACM, 40(3):59-62, 1997.

[TIPS99] TIPS Documentation, Quality Control Tools User Requirements, V-Framework Programme IST-1999-10419, 1999. http://tips.sissa.it

[Vas00] Vassiliadis P., Data Warehouse Modeling and Quality Issues, PhD thesis, Department of Electrical and Computer Engineering, University of Athens (Greece), 2000.

[Wan98] Wang R., A product perspective on Total Data Quality Management, Com. of the ACM, 41(2): 58-65, 1998.

[WSF95] Wang R.Y., Storey V.C., Firth C.P., A framework for analysis of data quality research, IEEE TKDE, 7(4):623-638, 1995.

Annexes

```
Algorithm Quality View Generation
 declarations
 input :
    xqs : an XQS file ;
    xps : an XPS file ;
    U : user of the view V ;
    N : a document Node (element, attribute, text);
    L : the list of XML Nodes corresponding to the considered document;
    R : the list of XML Nodes corresponding to the document view;
 output :
    V : quality view corresponding to the query of the user U
 Begin
 Initialize R and L such that R = [] and L = [] ;
 Add the root element of the answered document to L ;
 While L is not empty;
      N ← the first node of L ;
      select every quality-based recommendation rule of xqs such as :
           - the node N satisfies the pattern into the attribute DocumentNode ;
           - the user U is located with the path of the attribute Profile ;
      Apply the policy such as:
      - for the node N and the user U, if it exists a conflict between a
        set of rules
           then choose the rules with the highest priority ;
      - if there is more than one selected rule with the same priority,
           then the last rule is selected;

      if the value of the attribute "access" of the selected rule is "blocking",
           then delete N from the list L
           else add N to R ;
                Replace N by its children nodes (attributes, elements) into L ;

      if the value of the attribute "access" of the selected rule is
      "recommending", then
           case 1: the value of the attribute "strategy" is "exact":
                each quality criterion associated as metadata to the node N
                is compared to the quality constraints of the user profile xps ;
                if all the constraints are exactly satisfied by the criteria
                values then add N to the list R and delete N to L ;

           case 2: the value of the attribute "strategy" is "approximate":
                each quality criterion associated as metadata to the node N
                is weighted and compared to the quality constraints of the
                user profile xps ;
                if a quality constraints Ci (∀i) is satisfied with a weighted
                approximation ±ε_i then add N from the list R and delete N to L ;

           case 3: the value of the attribute "strategy" is "approximate":
                each quality criterion associated as metadata to the node N
                is weighted and compared to the quality constraints of the
                user profile xps ;
                while each quality constraint is not satisfied
                   - soften interactively the constraint (reduce its values);
                end-while
                add N from the list R and delete N to L ;
      Replace N by its children nodes (attributes, elements,...) into L ;
 end-while
V ← R ;
display the quality view V corresponding to xps and xqs;
end.
```

Figure 10. *Algorithm for quality-based recommendation of XML documents*