Schema Matching For Integrating Multimedia Metadata

Samir Amir, Ioan Marius Bilasco, Taner Danisman, Thierry Urruty, Ismail Elsayad and Chabane Djeraba
LIFL UMR CNRS 8022, University of Lille1, Telecom-Lille1, Villeneuve d'Ascq, France
Email:{samir.amir, marius.bilasco, taner.danisman, thierry.urruty, ismail elsayad, chabane.djeraba}@lifl.fr

***Abstract*—The recent growing of multimedia in our lives requires an extensive use of metadata for multimedia management. Consequently, many metadata standards have appeared. Using these standards has become very complicated since they have been developed by independent communities. The content and context are usually described using several metadata standards. Accordingly, a multimedia user must be able to interpret all these standards. In this context, several metadata integration techniques have been proposed in order to deal with this challenge. These integrations are made by domain experts which is costly and time-consuming. This paper presents a new system for a semi-automatic integration of multimedia metadata. This system will automatically map between metadata needed by the user and those encoded in different formats. The integration process makes use of several information: XML Schema entity names, their corresponding comments as well as the hierarchical features of XML Schema. Our experimental results demonstrate the integration benefits of the proposed system.**

I. Introduction

Multimedia resources in the form of still images, audio, speech, documents and video play an increasingly pervasive role in our lives. Thus, there is a growing need to enable the interpretation and the processing of such resources for their adaptation, filtering or semantic knowledge extraction. The processing of multimedia resources is done according to the context: where and by whom these resources will be used. In order to improve multimedia processing services and allow a better description of multimedia semantics, several multimedia description standards have appeared to enhance the retrieval, utilization and delivery of multimedia data over a variety of channels (e.g., Dublin Core, MPEG-7, MPEG-21, CIDOC/CRM, FGDC, and IMS) [1]. Those standards introduce a description of the semantics in multimedia contents, the context in which the content was created and for which it was designed. These descriptions are called *metadata* as they bring a new knowledge about multimedia contents and utilization context. The metadata presented in various multimedia standards describes different kinds of multimedia contents (e.g., video, image, audio, etc.), devices consuming or transmitting these contents (e.g., networks, TV, mobile, etc.) and user characteristics (e.g., user profile, user preference, etc.).

By anticipating the increase of multimedia metadata standards in the upcoming years, we can foresee that it will become progressively more and more difficult for current multimedia services to use metadata encoded in different formats. This is due to the number of independent multimedia communities, which combine terms from multiple vocabularies and use different structures for metadata description.

Dealing with multiple independent metadata standards is one of the major challenges in multimedia domain due to the heterogeneity of information as mentioned above. The creation, delivery and consumption of rich multimedia experiences between different entities in multimedia communities (e.g., multimedia content consumers, commercial content providers, simple producer, etc.) requires that each of these entities must have a diversified prior knowledge about different standards in order to take advantage of them. However this requirement is not easy to satisfy due to the numerous existing standards and those that will appear.

In order to tackle this problem, several solutions have been proposed to integrate heterogeneous multimedia metadata, some of them are reviewed in [2] [3]. However, the integration process for these solutions is done by human experts, which is costly and time-consuming. Besides, the integration process must be updated every time a new standard appears. In this context, an intelligent multimedia metadata integration solution is needed to address the interoperability problem by providing an automatic system for mapping between metadata needed by developer and those encoded in different formats.

Existing automatic data integration approaches are not adequate with multimedia metadata characteristics due to the high semantic and structural heterogeneity. Therefore, multimedia metadata need a new automatic integration system which is proper to its characteristics. To do so, we address in this paper the issue of automatically integrating heterogeneous multimedia metadata by proposing a new matching strategy. A strategy that takes into account the different semantic resources in the XML Schemas [1] describing metadata (concept names, comments and structure).

The remainder of this paper is organized as follows: Section 2 introduces some works performed manually for integrating heterogeneous multimedia. In section 3 we present some schema matching approaches and their limitations. Section 4 describes the proposed approach. Section 5 presents the experimental results. Section 6 gives concluding remarks and future work.

II. Multimedia Metadata Integration

Over the last decade, researchers have taken a deep interest in the integration of heterogeneous multimedia metadata. Several integration solutions have been proposed [2] [3]. Most of these researches focus on the creation of a core ontology which contains common information on multimedia metadata. This ontology acts as a mediated schema, which is the common interface used for querying all metadata encoded in different formats [4]. After designing this core ontology, a manual mapping is performed between the latter and other metadata standards [5]. Among these works we include [6] where authors proposed a framework to integrate three different music metadata. They used the generated MPEG-7 OWL ontology [7] as an upper-ontology to integrate other music metadata. This music metadata is manually mapped to MPEG-7 ontology.

The author in [8] proposed a core top-level ontology for the integration of information from different multimedia domains. A core top-level ontology is an extensible ontology that expresses the basic concepts that are common across a variety of domains and media types. These concepts provide the basis for specialization into domain-specific concepts and vocabularies. This ontology allows the construction of well-defined mappings between several domain-specific knowledge representations (i.e., metadata vocabularies). Some other integration solutions have been proposed for multimedia metadata [9] [10] [11]. However, all integration works mentioned above are performed manually by human experts which is costly and time-consuming.

A new W3C working group was created in 2008 [12] to develop a new system called *Ontology and API for Media Object*. It addresses the inter-compatibility problem by providing a common set of properties to define the basic metadata needed for media objects and the semantic links between their values in different metadata vocabularies. It aims at circumventing the current proliferation of video metadata formats by providing full or partial translation and mapping between the existing formats. The ontology is accompanied by an API that provides uniform access to all elements defined by the ontology, which are selected elements from different formats. At the given time, the W3C working group constructs the mapping manually.

Notwithstanding the efforts of *Ontology for Media Object* working group to integrate a large number of standards, we think that the integration could benefit from automatic integration approaches to find a mapping between the common set of properties which is a mediator between user and metadata standards.

III. Motivation

Due to the high modeling flexibility enabled by the XML Schema [13] type system, component reuse/sharing, and distributed schemas, it was approved as a W3C recommendation in 2001 and since then it has been increasingly adopted especially in multimedia metadata standardization [1] . Therefore, in order to achieve metadata interoperability and help multimedia developers to integrate metadata automatically, tools and mechanisms are needed. These tools and mechanisms must resolve the semantic and the structural heterogeneity and align terms between metadata needed by the developer, defined on the mediated schema and those encoded in different formats. Schema matching plays a central role in these approaches [14].

Due to the complexity of schema matching, it was mostly performed manually by human experts. However, manual reconciliation tends to be a slow and inefficient process especially in large-scale schema (e.g., MPEG-7, MPEG-21, etc.) and dynamic environments such as multimedia where new metadata standards are appearing constantly. Therefore, the need for a highly capable automatic schema matching system has become crucial.

Several XML data integration approches have been proposed in recent decades. These approaches can be broadly classified into two categories depending on the exploited objects in similarity computation: (1) *tree-editing distance* which exploits the whole XML Schema (sub)tree or XML paths without considering elements details [15] [16] [17] [18], (2) *schema matching* which exploits the semantic and structural element properties to determine similarity among XML Schemas [19] [20] [21] [22].

The tree-editing approaches have been proposed to cluster XML documents as well as they can be very expensive rendering them impractical for huge XML data. Moreover, these approaches are more adequate with DTDs and exploit few structural and semantic characteristics.

Among the schema matching approaches that have been experienced, we can highlight the success of the work done in [19] [20]. The authors in [19] have proposed a sophisticated hybrid matching approach combining a name matcher with a structural match algorithm baptized Cupid. Cupid transforms the original XML Schemas into trees and then performs a bottom-up structure matching. The basic assumption behind the structure matching phase of Cupid is that much of the information content is represented in leaves and that the leaves have less variation between schemas then internal structures. Thus, the similarity of inter-nodes is based on the similarity of their leaf sets. This is not always true that we can find equivalent concepts occurring in completely different structures, and completely independent concepts that belong to isomorphic structures.

The authors in [20] present a structure matching algorithm called Similarity Flooding (SF). The Similarity Flooding algorithm is based on the idea of similarity propagation. Schemas are represented as directed labeled graphs. The basic concept behind the algorithm is that adjacency contributes to similarity propagation. Thus, the algorithm will perform unexpectedly in cases when adjacency information is not preserved. Furthermore, SF ignores all types of constraints while performing the structural matching. Constraints like typing and integrity constraints are used at the end of the process to filter the mapping pairs with the help of user.

Motivated by the above challenges, we present in this paper a new schema matching-based approach for XML multimedia metadata integration. In particular, we develop and implement a new matching technique which exploits the semantic and structural information in a manner that increases the matching accuracy using several types of information available on XML

Schema (semantic, syntactic and structural).

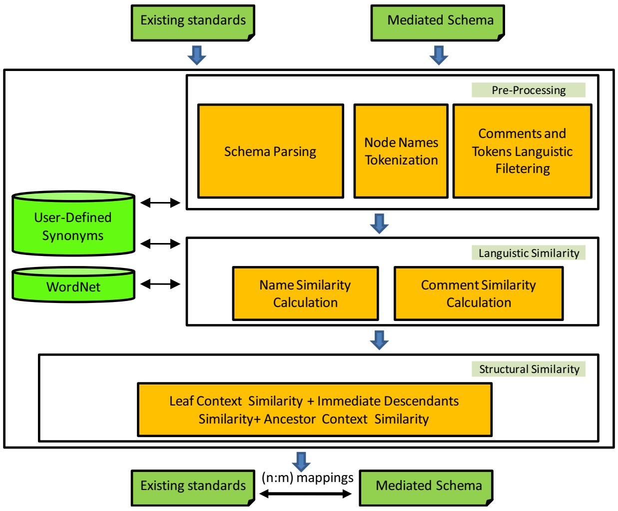

Fig. 1. Matching process phases

IV. Proposed Approach

In this section, we describe the different steps of the proposed matching system as shown in Figure 1. The system is composed of three main parts: *pre-processing*, *linguistic* and *structural similarity computation*.

We start by modeling XML Schemas as a directed labeled graph [23]. Then, all irrelevant words in both schemas to be mapped are eliminated and the useful words are normalized (Section IV-A). After the *pre-processing* step, the system calculates the linguistic similarity between nodes in both schemas by exploiting the semantic of their corresponding names and comments (Section IV-B). Finally, structural similarity is computed and the correct mappings are selected according to the linguistic and structural similarity scores (Section IV-D).

A. Pre-Processing

In this step, we start by parsing all entities involved in the matching process, including element and attribute names as well as comments corresponding to these entities (*xsd:documentation*). Then, entity names and comments are normalized in order to make their semantic useful for the linguistic similarity calculation step.

1) Node Names: Normally, each entity of an XML Schema is modeled by a node with a name. A node name is a string, without blank characters (space), that may be a word, a term, or an expression (a combination of words). In order to calculate the similarity between node names, a normalization step is necessary. First, each entity name is broken into a set of tokens M with a customizable tokenizer using punctuation, upper case, special symbols, and digits, e.g. *MediaRegionLocator* becomes (*Media*, *Region*, *Locator*). Once the tokenization step is over, tokens are lemmatized. Namely, they are morphologically analyzed in order to find all their possible basic forms. Thus, for instance, *Locations* is associated with its singular form, *Location*. A user-defined dictionary is also used to deal with acronyms and abbreviations, e.g., *ID* becomes *Identifier*.

2) Comments: As mentioned above, comments are used as other semantic information. This can be performed via information retrieval techniques. To do so, comments must be linguistically filtered by eliminating the words carrying little useful information, such as articles, prepositions, conjunctions, pronouns and modal verbs [24].

B. Linguistic Similarity Computation

This phase is concerned with the linguistic similarity computation between every XML Schema node pairs (on the mediated schema and metadata standards). In order to form the linguistic similarity matrix, a string-based technique is used to map the node names. WordNet is used for the explicitation of the words meaning [25]. In addition to the node names, comments are used as a second semantic resource for the matching process. We apply the TF/IDF technique [24] to these comments in order to extract the most pertinent information. The linguistic similarity score is a weighted sum of both similarities.

1) Names Matching: The purpose of this phase is to find an initial matching by calculating the similarity distance between the names of all node pairs in the two schemas to be mapped. Each node is represented by a set of tokens. Because of the richness of natural language, we first start with the explicitation of tokens meaning by using WordNet [26]. Several synonyms can be found for a given term. This helps to resolve problems of terminological conflicts occurring when metadata standards are developed by different communities which may describe the same information using different terms. For instance, some multimedia metadata communities [27] use the term "*type*" to describe the type of a given content. Some others [12] use the terms "*format*" or "*genre*" to describe the same information. Each node n_i represented by a set of tokens M_i will have a set of synonyms *synset* for each token m_i after the explicitation step. $M_i^{'}$ is the final result that regroups all synsets returned by M_i explicitation.

$$M_i^{'} = M_i \bigcup \{m_k | \exists m_j \in M_i \bigcap m_k \in synset(m_j)\} \quad (1)$$

The name similarity S_{name} between one XML Schema node pair (n_1, n_2) is calculated by using Jaro-Winkler metric (JW) [28] between each token $m_i \in M_1^{'}$ and all tokens $m_j \in M_2$ (and vice versa) [26]. The maximum score (MJW) is taken:

$$MJW(m_i, M^{'}) = max_{m_j \in M^{'}} JW(m_i, m_j) \quad (2)$$

The Jaro-Winkler distance is given by:

$$JW(m_i, m_j) = \frac{1}{3}(\frac{r}{|m_i|} + \frac{r}{|m_j|} - \frac{r-t}{r}) \quad (3)$$

where r is the number of matching characters and t is the number of transpositions. Finally, the average of the best similarities is calculated to get the name similarity between nodes:

$$S_{name}(n_1, n_2) = \frac{\sum_{m_i \in M_1} MJW(m_i, M_2^{'}) + \sum_{m_j \in M_2} MJW(m_j, M_1^{'})}{|M_1| + |M_2|} \quad (4)$$

2) Comments Matching: Due to the use of technical vocabularies by multimedia metadata communities, node names do not always provide a sufficient semantics. The comments related to each entity are also another semantic resource. We apply the TF/IDF technique used in the information retrieval domain in order to calculate the similarity between comments [24]. To do so, all comments on two schemas to be mapped are considered as documents, each node will be represented by a vector whose coordinates are the results of TF/IDF. Hence, the similarity between two nodes is the distance between vectors corresponding to their comments.

In order to illustrate how to calculate these vectors, let us consider v = (w_1, w_2, , w_P) be a vector representing a certain node n. $P = |U|$ is the number of distinct words in all comments in two schemas to be mapped. The i_{th} element w_i in the vector v, which represents the node n in a schema, is calculated as follow:

$$w_i = tf_i * idf_i \quad (5)$$

$$idf_i = \log_2 \frac{N}{b_i} \quad (6)$$

where tf_i is the term frequency. tf_i represents the number of times that the i_{th} word in U appears in the comment corresponding to n_i. idf_i (inverse document frequency) is the inverse of the percentage of the concepts which contain the word w_i. N is the number of comments in U in both schemas. b_i is the number of comments which contain the word w_i at least one time.

As we have mentioned previously, the similarity $S_{comment}$ between two nodes n_i and n_j is the distance between vectors corresponding to their comments v_i and v_j. This distance is a cosine similarity σ [29]. It is calculated as follow:

$$\sigma(v_i, v_j) = \frac{\sum_{k=1}^{P} w_{ik} w_{jk}}{\sqrt{\sum_{k=1}^{P} (w_{ik})^2 * \sum_{k=1}^{P} (w_{jk})^2}} \quad (7)$$

The result of above processes is a linguistic similarity matrix *lSim*, where:

$$lSim(n_i, n_j) = \mu_1 * S_{name}(n_i, n_j) + \mu_2 * S_{comment}(n_i, n_j) \quad (8)$$

where $\mu_1 + \mu_2 = 1$ and $(\mu_1, \mu_2) \geq 0$

C. Structural Similarity Computation

Linguistic similarity computation may provide several matching candidates. There can be multiple matching candidates which differ in the structure but have a high linguistic similarity value. For instance, the attributes *mpeg7:MediaRelTimePoint* and *mpeg7:MediaRelIncrTimePoint* may map to the same entity defined in mediated schema *ms:MediaTimePoint*. Thus, in order to deal with this case, the structural similarity is computed in order to prune these false positive candidates.

Convinced that the most prominent feature in an XML Schema is its hierarchical structure, our structural matching algorithm is based on the node context, which is reflected by its ancestors and its descendants. In this paper, as in [21], we consider three kinds of node contexts depending on its position in the ontology tree: *ancestor context*, *immediate descendant context* and *leaf context*. The context of a node is a combination of these three contexts.

1) Ancestor Context: The ancestor context of a node n_i is defined as the path p_i extending from the root node of the schema to n_i. The ancestor context similarity *ancSim* between two nodes (n_i, n_j) is based on the resemblance measure between their paths (p_i, p_j). This is done by calculating three scores established in [30]. These scores are combined and weighted by the linguistic similarity between (n_i, n_j) to compute the ancestor context similarity:

$$ancSim(n_i, n_j) = lSim(n_i, n_j) \\ * (\delta LCS_n(p_i, p_j) - \theta GAP(p_i, p_j) - \lambda LD(p_i, p_j)) \quad (9)$$

where δ, θ and λ are positive parameters ranging from 0 to 1. They represent the comparative importance of each factor. The three scores are: $LCS_n(p_i, p_j)$, the longest common subsequences between two paths normalized by the length of the first path, $GAP(p_i, p_j)$ used to ensure that the occurrences of two paths nodes are close to each other, and $LD(p_i, p_j)$ used to give higher values to source paths whose length is similar to target paths.

In order to relax the condition defined in [30], our parameter calculations consider that two nodes match if their *lSim* is greater than a given threshold τ_1, e.g. 0.80. This is done because *lSim* value is a combination of two similarities (S_{name} and $S_{comment}$) which may have a high values but differ from 1 (e.g., *dig35:copyright* and *dc:right*) [12].

2) Immediate Descendants Context: To obtain the immediate descendants context similarity *immSim* between two nodes (n_i, n_j), we compare their two immediate descendants context sets including attributes and subelements. This is done by using the linguistic similarity *lSim* between each pair of children in the two sets. We select the matching pairs with maximum similarity values. Finally, the average of best similarity values is taken.

3) Leaf Context: The leaf context of a node n_i is defined as the set of leaf nodes of subtrees rooted at n_i. If $l_i \in leaves(n_i)$ is a leaf node, then the context of l_i is given by the path p_i from n_i to l_i. The leaf context is given by:

$$leafSim(l_i, l_j) = lSim(l_i, l_j) \\ * (\delta LCS_n(p_i, p_j) - \theta GAP(p_i, p_j) - \lambda LD(p_i, p_j)) \quad (10)$$

To obtain the leaf context similarity between two leaves $l_i \in leaves(n_i)$ and $l_j \in leaves(n_j)$, we compute the leaf similarity *leafSim* between each pair of leaves in the two leaf sets. We then select the matching pairs with the maximum similarity values. The average of the best similarity values is taken.

D. Node Similarity

The node similarity *nodeSim* can be obtained by the combination of *ancestor context*, *immediate descendants context*, and *leaf context* similarities unless one of the two nodes being compared is a leaf node. In this case, node similarities calculation considers that the context of both nodes depends only on their ancestors. The node similarity is given by:

$$\mathit{nodeSim}(n_i, n_j) = \alpha * \mathit{ancSim}(n_i, n_j) + \beta * \mathit{immSim}(n_i, n_j) + \gamma * \mathit{leafSim}(n_i, n_j) \quad (11)$$

$\alpha + \beta + \gamma = 1$ and $(\alpha, \beta, \gamma) \geq 0$

Once the structural similarity computation is made, the system returns for each source node n_i the k node candidates that have the maximum values of *nodeSim*. In order to select the k candidates, one of the following strategies can be used:

Threshold: Returns all node pairs showing a similarity exceeding a given threshold value τ_2. This strategy may return too many matched candidates.

MaxDelta: Returns the node pair having a maximum similarity value *nodeSim* which is determined as a matching candidate plus all pairs with a similarity differing at most by a tolerance value d.

MaxN: The N node pairs with maximal similarity *nodeSim* are selected as matching candidates.

In our approach, we support considering several criteria at the same time, in particular MaxN in combination with a low threshold, e.g. 0.7.

V. Experimental Evaluation

In this section, we describe the experiments that we have carried out to evaluate our proposed method. Firstly, we describe the data sets which we have used through the evaluation. Secondly, we show our experimental results in terms of precision, recall and F-measure.

A. Data Sets

The system have been tested using several metadata standards (MPEG-7,MPEG-21, EXIF, MIX and DIG35) [1]. These standards have a significant structural and semantical heterogeneity. The mediated schema we have chosen to integrate the standards mentioned above is described in [31]. It is a part of the CAM4Home ITEA2 project [1]. A group of twenty multimedia academic and industrial practitioners from TV, 3G and Internet application fields defined a large set of metadata requirements in order to support the convergence of multimedia content in Digital Home environments. Metadata defined under CAM4Home project describes information related to content semantics, user characteristics and device profiles. Moreover, CAM4Home metadata framework contains several information which are available in the standards to be integrated. The system performance is evaluated according to the correct mappings found between equivalent concepts on CAM4Home metadata framework and different metadata standards.

[1] http://www.cam4home-itea.org/

B. Experimental Results

Table I shows a fragment of mapping results obtained from our experimentation where the first column shows the attributes from CAM4Home metadata framework that are mapped to other attributes from MPEG-7 and DIG35 respectively.

TABLE I
A FRAGMENT OF MAPPING RESULTS

CAM4Home	MPEG-7	DIG35
duration	mediaDuration	N/A
creatorReference	creator	image_creator
gpsLocation	location	location
creationDateTime	date	creationTime captureTime
camEntityVersion	version	version
copyright	copyrightString	copyright
description	abstract	caption
legalNotice	copyrightString	copyright
title	title	ipr_title
entityUID	publicIdentifier entityIdentifier	image_ID

Table II presents evaluation results in terms of precision, recall and F-measure where the second column illustrates the values of MaxN and τ_2 respectively (Section IV-D). These values have been chosen according to metadata characteristics in order to show the precision of the proposed system. For instance, the MIX standard contains a limited number of attributes and simple structure compared with MPEG-7/21 making its integration easier. MaxN and τ_2 values for MIX are (3, 0.75) which means that the system considers the 3 pairs of nodes (n_i, n_j) having maximum values of *nodeSim* which are greater than 0.75 as candidate mappings. MaxN and τ_2 are changed for other standards which contain a large number of attributes.

TABLE II
EXPERIMENTAL RESULTS

Metadata standards	$(MaxN, \tau_2)$	(α, β, γ)	Precision	Recall	F-measure
MIX	(3, 0.75)	(0.75, 0.10, 0.15)	96%	94%	95%
DIG35	(3, 0.70)	(0.60, 0.10, 0.30)	93%	88%	90%
EXIF	(4, 0.75)	(0.55, 0.20, 0.25)	94%	71%	81%
MPEG-7	(7, 0.60)	(0.70, 0.20, 0.10)	71%	65%	68%
MPEG-21	(7, 0.60)	(0.70, 0.20, 0.10)	66%	51%	58%

The third column illustrates manually selected (α, β, γ) values. The automatic choice of these values for each pair of nodes according to node positions in the schema will be a part of our future work. Our experimental evaluation shows that the greatest amount of structural information is contained in the ancestor context (note that α value is greater for all standards). This explains the interest of some matching strategies which consider that the context of nodes depend only on their ancestors [32]. The last three columns show the values of precision, recall and F-measure respectively for the matching results between each standard and CAM4Home metadata framework. Table II shows that the proposed technique successfully found a large number of correct mappings where the values of precision, recall and F-measure are higher for EXIF, MIX and DIG35. The values of precision, recall and F-measure for MPEG-7 and MPEG-21 are (71%, 65%, 68%) and (66%, 51%, 58%) which is an important score for these two large data sets compared to the high number of shared schema components used by these standards.

VI. Conclusion

Due to the number of existing multimedia metadata standards and their semantic and structural heterogeneity, there has been a great interest to develop an automatic multimedia metadata integration solution. The existence of such model makes the integration process faster and less expensive. We proposed and implemented a new XML Schema matching technique to automate the integration of multimedia metadata. We essentially proposed a linguistic and structural similarity measure linking metadata encoded in different formats to those defined on the mediated schema. Our experiments showed that the combination of linguistic and structural similarities plays a significant role in deriving a correct mapping. In our ongoing work, we plan to enhance the proposed matching system through a better use of structural information. This can be achieved by adding a new structural matching technique to the system. We mainly explore the use of adjacency nodes to detect other mappings that cannot be detected by the current matching strategy. We also plan to enhance the proposed approach by taking into account the mappings already validated by the user as another structural information which may help to find other mappings.

VII. Acknowledgements

This work was supported by the ITEA2 CAM4HOME European project.

References

[1] M. Hausenblas. (2005, Jul.) Multimedia vocabularies on the semantic web. W3C. Amsterdam. [Online]. Available: http://www.w3.org/2005/Incubator/mmsem/XGR-vocabularies-20070724/

[2] S. Amir, I. M. Bilasco, M. H. Sharif, and C. Djeraba, *Towards a Unified Multimedia Metadata Management Solution*. Intelligent Multimedia Databases and Information Retrieval: Advancing Applications and Technologies, IGI Global, 2010.

[3] B. Haslhofer and W. Klas, "A survey of techniques for achieving metadata interoperability," *ACM Comput. Surv.*, vol. 42, no. 2, 2010.

[4] S. Amir, I. M. Bilasco, and C. Djeraba, "A semantic approach to metadata management in sensor systems," presented at the COGnitive systems with Interactive Sensors, 2009.

[5] S. Amir, "Un système d'intégration de métadonnées dédiées au multimédia," in *INFORSID*, 2009, pp. 491–492.

[6] R. Garcìa and O. Celma, "Semantic integration and retrieval of multimedia metadata," in *2nd European Workshop on the Integration of Knowledge, Semantic and Digital Media*, 2005, pp. 69–80. [Online]. Available: URL = files/publications/d450b9-ISWC2005-GarciaCelma-SemAnnot2005.pdf, bibsource = DBLP, http://dblp.uni-trier.de

[7] D. L. McGuinness and F. v Harmelen. (2004, Feb.) Owl web ontology language overview. W3C. Amsterdam. [Online]. Available: http://www.w3.org/TR/owl-features/

[8] J. Hunter, "Enhancing the semantic interoperability of multimedia through a core ontology," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 13, no. 1, pp. 49–58, 2003.

[9] M.Brut, S. Laborie, A. Manzat, and F. Sèdes, "Integrating heterogeneous metadata into a distributed multimedia information system," presented at the COGnitive systems with Interactive Sensors, 2009.

[10] C. Tsinaraki, P. Polydoros, and S. Christodoulakis, "Integration of owl ontologies in mpeg-7 and tv-anytime compliant semantic indexing," in *CAiSE*, 2004, pp. 398–413.

[11] M. Doerr, J. Hunter, and C. Lagoze, "Towards a core ontology for information integration," *J. Digit. Inf.*, vol. 4, no. 1, 2003.

[12] W. Lee, T. Brger, F. Sasaki, and V. Malaisé. Use cases and requirements for ontology and api for media object. W3C. Amsterdam. [Online]. Available: http://dev.w3.org/2008/video/mediaann/mediaont-1.0/mediaont-1.0.html?content-type=text/html, month = jul, year =

[13] J. P. C. S.-M. E. M. F. Yergeau, T. Bray. (2004) Extensible markup language (xml) 1.0 (third edition) w3crecommendation. W3C. Amsterdam. [Online]. Available: http://www.w3.org/TR/2004/REC-XML-20040204

[14] J. Euzenat and P. Shvaiko, *Ontology matching*. Heidelberg (DE): Springer-Verlag, 2007.

[15] H. Su, H. A. Kuno, and E. A. Rundensteiner, "Automating the transformation of xml documents," in *WIDM*, 2001, pp. 68–75.

[16] T. Dalamagas, T. Cheng, K.-J. Winkel, and T. K. Sellis, "A methodology for clustering xml documents by structure," *Inf. Syst.*, vol. 31, no. 3, pp. 187–228, 2006.

[17] E. Bertino, G. Guerrini, and M. Mesiti, "Measuring the structural similarity among xml documents and dtds," *J. Intell. Inf. Syst.*, vol. 30, no. 1, pp. 55–92, 2008.

[18] C. C. Aggarwal, N. Ta, J. Wang, J. Feng, and M. J. Zaki, "Xproj: a framework for projected structural clustering of xml documents," in *KDD*, 2007, pp. 46–55.

[19] J. Madhavan, P. A. Bernstein, and E. Rahm, "Generic schema matching with cupid," in *VLDB*, 2001, pp. 49–58.

[20] S. Melnik, H. Garcia-Molina, and E. Rahm, "Similarity flooding: A versatile graph matching algorithm and its application to schema matching," in *ICDE*, 2002, pp. 117–128.

[21] M.-L. Lee, L. H. Yang, W. Hsu, and X. Yang, "Xclust: clustering xml schemas for effective integration," in *CIKM*, 2002, pp. 292–299.

[22] A. Boukottaya and C. Vanoirbeek, "Schema matching for transforming structured documents," in *ACM Symposium on Document Engineering*, 2005, pp. 101–110.

[23] H. H. Do and E. Rahm, "Coma - a system for flexible combination of schema matching approaches," in *VLDB*, 2002, pp. 610–621.

[24] C. J. van Rijsbergen, *Information Retrieval*, 2nd ed. London: Butterworths, 1979.

[25] C. Fellbaum, Ed., *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press, 1998.

[26] F. Lin and K. Sandkuhl, "A survey of exploiting wordnet in ontology matching," in *IFIP AI*, 2008, pp. 341–350.

[27] XMP working group. (2008, Feb.) Extensible metadata platform specification. XMPSpecificationPart2.pdf. [Online]. Available: http://www.adobe.com/devnet/xmp/pdfs/XMPSpecificationPart2.pdf

[28] M. Bilenko, R. J. Mooney, W. W. Cohen, P. D. Ravikumar, and S. E. Fienberg, "Adaptive name matching in information integration," *IEEE Intelligent Systems*, vol. 18, no. 5, pp. 16–23, 2003.

[29] D. A. Simovici and C. Djeraba, *Mathematical Tools for Data Mining: Set Theory, Partial Orders, Combinatorics*. Springer Publishing Company, Incorporated, 2008.

[30] D.Carmel, Y. S. Maarek, M. Mandelbrod, Y. Mass, and A. Soffer, "Searching xml documents via xml fragments," in *SIGIR*, 2003, pp. 151–158.

[31] I. M. Bilasco, S. Amir, P. Blandin, C. Djeraba, J. Laitakari, J. Martinet, E. Martínez-Gracía, D. Pakkala, M. Rautiainen, M. Ylianttila, and J. Zhou, "Semantics for intelligent delivery of multimedia content," in *SAC*, 2010, pp. 1366–1372.

[32] P. Bouquet, L. Serafini, and S. Zanobini, "Semantic coordination: A new approach and an application," in *International Semantic Web Conference*, 2003, pp. 130–145.