Domain Knowledge Based Queries for Multimedia Data Retrieval

Samira Hammiche*, Bernardo Lopez*, Salima Benbernou*, Mohand-Saïd Hacid*, Athena Vakali**
* LIRIS – University Claude Bernard Lyon 1
43, bld du 11 Novembre 1918. 69622 Villeurbanne - France
{shammich, blopez, sbenbern, mshacid}@liris.univ-lyon1.fr
**Department of Informatics
Aristotle University of Thessaloniki. 54006 Thessaloniki – Greece
avakali@csd.auth.gr

Abstract

This paper describes an approach for semantic description and retrieval of multimedia data described by means of MPEG-7. This standard uses XML schema to define the descriptions. Therefore, it lacks ability to represent the data semantics in a formal and concise way and it does not allow integration and use of domain specific knowledge. Moreover, inference mechanisms are not provided and hence the extraction of implicit information is not (always) possible. To address these issues, we propose to add a conceptual layer on top of MPEG-7 metadata layer, where the domain knowledge is represented using a formal language. A set of mapping rules is proposed. They serve as a bridge between the two layers.

Querying MPEG-7 descriptions using XML query languages such as XPath or XQuery requires to know MPEG-7 syntax and documents structure. To provide a flexible query formulation, we exploit the conceptual layer vocabulary to express user queries. A user query, making reference to terms specified at the conceptual level, is rewritten into an XQuery expression over MPEG-7 descriptions.

Keywords: MEPG-7, Multimedia Data Descriptions, Semantic and Structural Aspects, Mapping Rules, Logic Languages, Ontology

1 Introduction and problem description

The rapid expansion of digital multimedia content has led to the need of rich descriptions and efficient retrieval tools. The MPEG-7 [1] standard has emerged with the intention of allowing for efficient searching, indexing, filtering and accessing multimedia contents. MPEG-7, formally named "*Multimedia Content Description Interface*", specifies a standard set of description tools which are : a set of Descriptors (*Ds*), a set of Description Schemes (*DSs*), a Description Definition Language (*DDL*) based on XML Schema and System Tools. An MPEG-7 description is an XML document validated in accordance to the appropriate MPEG-7 *DS*.

MPEG-7 uses XML schema as its own *DDL*. Structural, cardinality and typing constraints can be expressed. However, it displays some limitations when we need to extract implicit information. This is due to the fact that there is neither formal semantics, nor inference capabilities associated with the elements of XML documents. Consequently, MPEG-7 cannot represent the meaning (semantics) of multimedia content. Moreover, it cannot integrate and exploit domain specific knowledge. Several research works have investigated these problems and most of them propose to translate the MPEG-7 vocabulary (*Ds* and *DSs*) from the XML representation, to another formalism such as RDFS [8], OWL [5, 12, 13] or XDD [3], where the semantics can be represented, the domain knowledge can be integrated and reasoning mechanisms are provided to extract implicit information.

In this paper, we design a different approach to tackle some of MPEG-7 limitations. We propose to exploit the abilities of the MPEG-7 standard to integrate simple structures of knowledge organized as concept hierarchies, using MPEG-7 classification schemes and the semantic entities abstraction tools. These knowledge are defined in a conceptual layer, that we add on top of the metadata layer (MPEG-7 descriptions), in which reasoning mechanisms can be applied. A set of mapping rules is defined to link the vocabulary of the conceptual layer to the elements of the MPEG-7 descriptions.

To retrieve MPEG-7 descriptions, the use of XML standard query languages like XQuery [2] are not flexible and do not allow inference of implicit information. So, to allow flexibility in query formulation and inferential mechanisms, we propose to exploit the conceptual layer vocabulary to assist the query formulation process. We propose a pre-processing algorithm to rewrite the user query into several queries where implicit information are made explicit and more specific concepts are included. The resulting queries are then translated into XQuery queries using a translation algorithm and evaluated over MPEG-7 descriptions.

The rest of this paper is structured as following : Section 2 presents the architectural framework for modelling multimedia data described using MPEG-7. Section 3 deals with query processing steps to evaluate user queries over MPEG-7 descriptions. Section 4 is a brief description of the application we work. Section 5 gives a short description of related work and we conclude in Section 6.

[1] http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm/
[2] http://www.w3.org/TR/xquery/

2 Multimedia data modeling

We are interested in querying multimedia data described by means of MPEG-7. First, we have to consider the representation issues. To cope with the limitations of XML schemas as a description language, we propose to extend MPEG-7 description layer with a conceptual layer.

2.1 Multi-Layered Representation of Multimedia Content

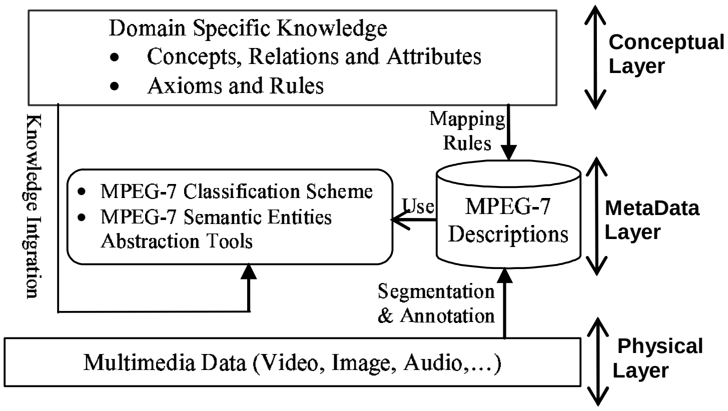

Figure 1: Architecture of the multimedia data representation framework

Figure 1 shows the different layers that we use in the process of multimedia data description using MPEG-7 standard. **The Physical Layer** contains the raw data to be described like video sequences. **The Metadata Layer** contains MPEG-7 descriptions produced after the segmentation and annotation process of multimedia data. **The conceptual Layer** contains the domain vocabulary relevant to an application.

In the sequel, we detail both the metadata and conceptual layers. We explain how to represent and formalize the domain knowledge, how to integrate these knowledge in the MPEG-7 descriptions and finally, how to connect MPEG-7 descriptions and conceptual layer in order to take the advantages of each representation type.

2.2 Metadata Layer : The MPEG-7 descriptions

In the MPEG-7 description, both structure and semantic features of multimedia content are integrated in the same description. More precisely, the structural description of the content is based on the notion of segment, represented by the abstract type *SegmentDS*. The segments can be recursively decomposed temporally, spatially or by their constituents like the audio tracks and thus, form a tree hierarchy of segments. The semantic description tools (*SemanticDS*) used to describe the content are [2]: semantic entities, semantic attributes and semantic relations. Semantic entities are partitioned into a set of Object (*ObjectDS*), Agent (*Agent DS*), Event (*EventDS*), Concept (*Concept DS*), State (*SemanticState DS*), Place (*SemanticPlaceDS*) and Time (*SemanticTimeDS*). Semantic attributes refer to the features of the semantic entity like their name and definition. Normative semantic relations describe how semantic entities are related.

```
<Mpeg7><Description xsi:type="ContentEntityType">
  <MultimediaContent xsi:type="VideoType">
    <video id="Video News">
     <! -- Creation Information --!>
     <! -- Media Information --!>
         <MediaLocator><MediaUri>news.mpg</MediaUri></MediaLocator>
         <MediaTime><MediaTimePoint>T00:00:00</MediaTimePoint>
                    <MediaDuration>PT2M30S</MediaDuration>  </MediaTime>
     <!-- Video Segmentation -->
     <TemporalDecomposition gap="false" overlap="false">
       <VideoSegment id="ID-V1">
        <MediaTime> ... ..</MediaTime>
       <!-- Semantic Description of the VideoSegment --!>
       <Semantic>
         <Label> <Name> A report dealing the subject of War in Iraq </Name></Label>
         <!-- Description of the event Iraq War --!>
           <SemanticBase xsi:type="EventType" id="IraqWar">
            <Label><Name>Iraq War</Name></Label></SemanticBase>
       </Semantic>
    </VideoSegment>
    <!--Description of the other video segments and the semantic entities implied -->
  </TemporalDecomposition> </Video> </MultimediaContent> </Description> </Mpeg7>
```

Figure 2: An MPEG-7 description example

Figure 2 is an MPEG-7 description extract of a news video sequence where the event "*IraqWar*" is illustrated. This information uses the MPEG-7 description tools : *Event DS* (for the Iraq War) and *VideoSegment DS* for the news video sequence and its components.

Let's consider the following query over the MPEG-7 description.

– Q : Find a report news sequence which illustrates a conflict.

The expression of such a query in XQuery will search for MPEG-7 *VideoSegment* elements where the event "conflict" takes place. The expected result is empty because the event "*conflict*" is not stated explicitly in the MPEG-7 description, even if "*IraqWar*" can be considered as a kind of "*conflict*". Moreover, the type of segments (report segment) is not specified in the MPEG-7 description. A video segment can be an answer without satisfying the segment type.

From this query and the expected results, we can note that MPEG-7 descriptions need to integrate knowledge about data structure and the described content of multimedia data.

2.3 Conceptual Layer : The domain Knowledge

The most used structure to represent domain knowledge are ontologies. They facilitate knowledge sharing among users by providing a formal conceptualization of a given domain [7]. Besides the formal description and the vocabulary sharing of a specific domain, we choose to use ontologies in the framework mainly : to create semantically more accurate annotations of the multimedia content, to help the users to express their queries more precisely and unambiguously and to infer implicit information during the querying process since ontology languages are provided with reasoning mechanisms.

a) Domain Knowledge represented in the ontology

We consider that each domain can be represented using an ontology $O = (C_s, C_c, C_d, A_c, A_s, R_c, R_s, R_{sc})$ where:

– C_s : the set of concepts that describe the vocabulary used to characterize the media structure. For example, the structure of the different news video segments can be : *presenter sequence*, *interview sequence* or *report sequence*.

– C_c : the set of concepts that describe the multimedia content. These concepts represent the entities perceived by human, like objects, events, person, etc. We adopt the MPEG-7 semantic model in the sense that semantic content is described by a set of semantic entities, their properties and their inter-relations.

– C_d : the set of concepts that represent the different datatypes used to describe the concepts properties included in C_s and / or C_c. Examples of such datatypes are : *string*, *number* and *date*.

– A_c and A_s are the set of attributes that describe C_s and C_c concepts. They are binary relations which relate concepts of C_s and C_c to their domain datatype in C_d.

– R_s (resp. R_c) : the set of relations between the structural entities (resp. the semantic entities). These relations can be symbolic (like *equivalent*), spatial (like *left*) or temporal (like *precede*). We restrict these sets to the normative relations defined in MPEG-7.

– R_{sc} : the set of relations that allow the connection of structural entities (C_s) and their content semantic entities (C_c). For example, the relation $Event\_Occurence(Segment, Event)$ which relates an event to the segment where it occurs.

Moreover, each category of concept can be organized into a concept hierarchy, i.e. the concepts are related using the subsumption relation "is-a".

b) Representation Formalism of the domain ontology

We have to use a formalism able to describe the ontology vocabulary, to support query formulation and to provide decidable reasoning mechanisms. A family of languages which satisfy these conditions are CARIN languages [10]. In particular, CARIN-$\mathcal{ALN}$ combines two representation formalisms : the description logic (DL) $\mathcal{ALN}$ [1] and Datalog Rules. Furthermore, Datalog rules can be used as a conceptual query language to formulate user queries.

The domain knowledge base $\triangle = (\triangle_T, \triangle_R)$ of CARIN-$\mathcal{ALN}$ language is represented using two parts : the terminological knowledge ($\triangle_T$) and the deductive knowledge ($\triangle_R$).

– The Terminological Knowledge : $\triangle_T$

In this part, domain concepts, roles and attributes are defined. More complex concepts can be defined using the constructors of the $\mathcal{ALN}$ DL. Let N_C be the set of atomic concept names, N_R the set of role names (relations or attributes). The set of $\mathcal{ALN}$ concepts is defined inductively as the smallest set for which the following holds : $\top$ (the concept Top) and $\bot$ (the concept Bottom) are concepts, each atomic concept name $A \in N_C$ is a concept, if C and D are concepts, R is a role and n is an integer, then $C \sqcap D$ (concept conjunction), $\neg C$ (concept negation), $\forall R.C$ (universal quantification), $\geq nR.C$ and $\leq nR.C$ (number restriction) are also concepts.

$\triangle_T$ is a set of axioms defined using two forms : $C \sqsubseteq D$ (for concept definition and inclusion) and $R \sqsubseteq C \times D$ (for roles and attributes definition).

– The deductive knowledge : $\triangle_R$

In this part, additional knowledge, not expressible as terminologies, are expressed using Datalog rules. In particular, rules can be used to assert more complex relationships between concepts, roles and attributes.

The rules defined in CARIN-$\mathcal{ALN}$ are logical sentences of the form : $q(\bar{Y}) \leftarrow p_1(\bar{X_1}) \wedge ... \wedge p_n(\bar{X_n})$ where $\bar{X_1}$,...,$\bar{X_n}$ and $\bar{Y}$ are tuple of variables or constants. The rules must be safe, i.e. a variable that appears in $\bar{Y}$ must also appear in $\bar{X_1} \cup ... \cup \bar{X_n}$. The predicates $p_1, ..., p_n$ can be either concepts or roles names (defined in the ontology), or ordinary predicates that do not appear in the ontology vocabulary. The predicate q must be an ordinary predicate.

c) Example

Let's consider the domain knowledge of the MPEG-7 description presented in Section 2.2. The knowledge concern the structure and content of news video sequences. The segmentation of video news is a set of video segments which can be : a "*presenter sequence*", an "*interview*" or a "*report sequence*". Each segment may have some attributes (*id, name, duration*, etc.). For the content described in the segment, we can use the MPEG-7 categories, i.e. Person, Object, Event, Place and Time. The person that appears in a news sequence may be : *journalist* (*news presenter, reporter, speaker*), *interviewee* (*politic man, an event witness, an expert, ...*). The event can be of type *politic*, *cultural*, *sports*, etc. A politic event can be categorized into *war, meeting*, etc. The different categories are related using binary relations.

In the sequel, we give an extract of the news ontology $O = (C_s, C_c, C_d, A_c, A_s, R_c, R_s, R_{sc})$ where:
– $C_s = \{VideoSegment, PresenterSequence, InterviwSequence, ReportSequence, ...\}$
– $C_c = \{Person, Journalist, Reporter, NewsPresenter, Interviewee, PoliticMan, Expert, EventWitness, Artist, Event, PoliticEvent, CulturalEvent, SportEvent, War, Place, Country, ...\}$
– $C_d = \{String, Date, Number, ...\}$
– $A_c = \{id, has\_name, has\_location, ...\}$
– $A_s = \{id, has\_title, ...\}$
– $R_c = \{AgentOf, PatientOf, ResultOf, ...\}$
– $R_s = \{Before, After, During, Depicts, ...\}$
– $R_{sc} = \{Event\_Occurence, Appear, ...\}$

The formalization of this ontology using CARIN-$\mathcal{ALN}$ formalism is given in Table 1 (due to space limitation, we present just an extract of this ontology).

Table 1: A part of CARIN-$\mathcal{ALN}$ knowledge base of the domain ontology $(\triangle_T, \triangle_R)$

The Terminological Knowledge : Structure and Content (Event) Knowledge
$VideoSegment \sqsubseteq \top, VideoSegment \sqsubseteq \forall id.String$ $PresenterSequence \sqsubseteq VideoSegment \sqcap \forall has\_presenter.NewsPresenter$ $InterviewSequence \sqsubseteq VideoSegment \sqcap \forall has\_interviewee.Interviewee$ $ReportSequence \sqsubseteq VideoSegment \sqcap \forall has\_subject.Event$ $ReportSequece \sqsubseteq \leq 1 Event\_Occurrence \sqcap \forall Event\_Occurence.Event$ $VideoSegment \sqsubseteq \leq 1 Appear \sqcap \forall Appear.Person$
$Event \sqsubseteq \top, Event \sqsubseteq \forall id.String$ $CulturalEvent \sqsubseteq Event, SportEvent \sqsubseteq Event$ $War \sqsubseteq PoliticEvent, IraqWar \sqsubseteq War$
The Deductive Knowledge
$conflict(X) \leftarrow War(X)$ $subjectof(X,Y) \leftarrow VideoSegment(X) \wedge Event(Y) \wedge Event\_Occurence(X,Y)$

A particular concept (resp. role or attribute) can be instantiated by a unary (resp. binary) predicate. To define n-ary predicates or relations between roles, the axioms of $\triangle_T$ are not enough to express such requirements. In this case, the deductive knowledge can be used. For example, the predicates "*conflict*" and "*subjectof(segment, event)*" can be defined using the structural and content predicates of $\triangle_T$.

2.4 How to integrate the domain knowledge in MPEG-7 descriptions ?

In the conceptual layer, several concept hierarchies can be defined. We need to integrate these hierarchies within MPEG-7 descriptions to allow a precise description and efficient retrieval.

a) Integration of the structural knowledge

The knowledge about the media structure has to be integrated in the MPEG-7 *Segment DS* since it is the *DS* used to represent multimedia content structural aspect. In the MPEG-7 *Segment DS* syntax, an optional element named "*StructuralUnit*" is defined. This element characterizes the segment where it is used by referencing terms defined in a **Classification Scheme** (*CS*) [2]. A *CS* defines taxonomies to classify a subject area with a set of terms, organized into a hierarchy. A term represents a well defined concept in the domain and has : an identifier, a name and a definition that describes the meaning of the term.

b) Integration of the knowledge related to the content

As for the structure, we need to integrate domain knowledge that describes the multimedia content. This knowledge is organized into categories of concept hierarchies. Each category represents one of the MPEG-7 semantic entities. Therefore, the knowledge has to be integrated in the MPEG-7 semantic entities.

The MPEG-7 $SemanticBaseDS$ (the equivalent of $SegmentDS$ for the structure) allows the definition of abstraction entities using the optional element "*AbstractionLevel*". This element has the integer attribute "*Dimension*" set to "0" if the entity described is concrete and positif (>0) if the entity is abstract. To link a concrete entity to its abstract entity, the MPEG-7 relation "*exampleOf*" is used. To define the "is-a" relation between two abstract entities, the MPEG-7 relation "*specialisationOf*" is used.

Due to space limitations, illustrating examples are not given. More details can be found in a research report version [3].

2.5 How to link the conceptual layer to the metadata layer?

MPEG-7 is useful since it allows the representation of multimedia data structure and content from different viewpoints. The domain ontology of the conceptual layer provides reasoning mechanisms. To take advantages of the two representations, we need to connect the conceptual layer to the MPEG-7 description. To allow that, we use a set of mapping rules of the form *R: Conceptual Knowledge → MPEG-7 Metadata* where: (i) Concept Knowledge are domain concepts, domain roles or domain attributes, (ii) MPEG-7 Metadata are elements of the MPEG-7 descriptions obtained using a set of XQuery functions that we define.

[3] http://bat710.univ-lyon1.fr/∼ shammich/Publications/SM_Paper.pdf

If C is a set of domain concepts, D a set of datatype concepts, R a set of domain relations ($R \subset C \times C$), A a set of domain attributes ($A \subset C \times D$), F a set of XQuery functions (defined to retrieve a particular element, attribute or role in an MPEG-7 description) and E a set of MPEG-7 elements, then, three types of mapping rules can be defined using the mapping function M (cf. Table 2). This function returns : (1) the MPEG-7 elements instances of a given concept, (2) the attribute values of a given concept attribute and (3) a boolean value of a binary relations, which states if the relation between the entities exists or not.

Table 2: The different types of Mapping Rules

Mapping rules for concepts
$R_{concept} : C \rightarrow E$ $c \in C \rightarrow M(c) = \{e \in E \| e \in extension(c), extension \in F\}$
Mapping rules for roles
$R_{role} : R \rightarrow \{0, 1\}$ $r(x, y) \rightarrow M(r(x, y)) = \{0$ if $f(r(x, y))$ is false, 1 if $f(r(x, y))$ is true. $f \in F\}$
Mapping rules for attributes
$R_{attribute} : A \rightarrow \{0, 1\}$ $a(x) \rightarrow M(a(x)) = \{y \in attribute(a, domain(a)) \| attribute \in F, y \in D\}$

Figure 3 illustrates the definition of two MPEG-7 functions using XQuery syntax. The first function corresponds to the concept "*VideoSegment*" of the domain ontology and returns the set of MPEG-7 *VideoSegment* elements. The second defines the MPEG-7 elements of a particular person in the person hierarchy. In these functions, the semantic abstraction tools and the classification scheme of the structural unit are used to locate the most specific element in the MPEG-7 description.

```
define function get_video_segment ($description xs:string) as element( )*
{   For $segment in $description//MPEG-7
    Let $VideoSegment := $Segment//VideoSegment
    Let $AudiovisualSegment= $Segment//AudioVisualSegment
    Return
      $VideoSegment,
      $AudioVisualSegment}

define function person_role ($role xs:string, $role_file xs:string) as element( )*
{    Let $person := doc($rolefile)//MPEG-7
     For $person_role in $person//Semantic/SemanticBase [@xsi:type="AgentObjectType"]
     Where
           $perons_role[AbstarctionLevel/@Dimension="0" ] and
           $person_role[Relation[@name="exampleOf " and  @target= $role]]
     Return
         $person_role}
```

Figure 3: XQuery functions corresponding to the domain concepts

Using the two functions of Figure 3, we can define two mapping rules :

R_1: *VideoSegment* $\rightarrow mpeg7 : get\_video\_segment(\$file\_name)$

R_2: *Presenter* $\rightarrow mpeg7 : person\_role(Presenter, \$Role\_file)$

3 Querying MPEG-7 descriptions of multimedia data

The MPEG-7 description tools allow the integration of concept hierarchies that describe the multimedia data structure and content. However, their XML representation does not allow their exploitation to infer implicit information since XML is not provided with reasoning mechanisms. To alleviate this limitation, we propose to exploit the domain knowledge formalism of the conceptual layer to *rewrite* user queries and make the implicit information explicit. The set of rewritten queries are reformulated into the XQuery formalism by using the mapping rules defined in Section 2.5. The queries obtained are then evaluated over MPEG-7 descriptions.

3.1 Query Form and Grammar

As stated previously, we use Datalog rules of the CARIN-$\mathcal{ALN}$ formalism to formulate user's queries. The Datalog rules allow the formulation of conjunctive queries of the form: $Q(\bar{X}) \leftarrow P_1(\bar{X}_1) \wedge ... \wedge P_n(\bar{X}_n)$ where : (1) P_i are unary predicates (concepts of $\triangle_T$), binary predicates (roles or attributes of $\triangle_T$) or ordinary predicates (predicates defined in $\triangle_R$), (2) the variables of $\bar{X}$ are called "*distinguished variables*" and represent the variables for which the user search the instances when the query is formulated and (3) the variables present in $\bar{X}_1 \cup ... \cup \bar{X}_n$ are used to express constraints about the distinguished variables.

We assume that the users are interested in retrieving a particular multimedia segment (image, video segment, audio segment) where constraints about the content are expressed. These constraints can concern person, events, objects, etc., their attributes and their interrelations in the segment. The attributes of the structural unit are taken into account, but we don't consider the relation between the structural segments. Given these assumptions, the grammar of multimedia queries at the conceptual layer is illustrated in the Table 3.

Table 3: Grammar of the multimedia conceptual queries

Q ::= "Q (" $Variables$ ") ← " *Query_Expression*;
Query_Expression ::= *Structure_Expression+* "∧" *Content_Expression+* "∧" *StructCont_Relation+*;
Structure_Expression ::= *Structure_Unit* | *Structure_Attribute*;
Structure_Unit ::= *Structure_Name* "("*Structure_id* ")";
Structure_Attribute ::= *Struct_Attribute_Name* "("*Structure_id* "," *Attribute_Value* ")";
Content_Expression ::= *Content_Entity* | *Content_Attribute* | *Content_Relation*;
Content_Entity ::= *Entity_Name* "("*Entity_id* ")";
Content_Attribute ::= *Entity_Attribute_Name* "("*Entity_id* "," *Attribute_Value* ")";
Content_Relation::= *Content_Relation_Name* "(" *Entity_*id_1 "," *Entity_*id_2 ")";
StructCont_Relation ::= *Event_Occurence* "(" *Structure_id* "," *Entity_id*")"| *Appear*"(" *Structure_id* ","*Entity_id*")";
Variables ::= *Structure_id* ;
Legend :
– *Structure_Name* : a terminal expression which represents the concept name of the structural unit.
– *Struct_Attribute_Name* : a terminal expression which represents the attribute name of a particular structural unit.
– *Entity_Name* : a terminal expression which represents the concept name of the semantic entity.
– *Entity_Attribute_Name* : a terminal expression which represents the attribute name of the semantic entity.
– *Content_Relation_Name* : a terminal expression which represents the relation name that rely two semantic entities.
– *Structure_id* : the structural unit identifier.
– *Entity_id* : the semantic entity identifier.
– *Attribute_Value* : the attribute value.

3.2 Query Pre-Processing Algorithm

Once the user formulates the query, pre-processing steps are performed to get a set of rewritten queries, where implicit information are made explicit and more specific concepts are included.

Given a conjunctive query Q and a knowledge base $\triangle = (\triangle_T, \triangle_R)$, the rewritten algorithm produces a set of queries $Q_1, ..., Q_m$ equivalent to Q by processing the following steps :

1. For each ordinary predicate $P_i \in body(Q)$, replace P_i by its body defined in the deductive knowledge part $\triangle_R$.
2. Iterate the step 1 until all the ordinary predicates are replaced.
3. Eliminate the redundant predicates in the obtained query.
4. Replace each concept (unary predicate) of the obtained query by its subsumer.

Illustrating Example : Let's take again the query considered in Section 2.1.
Q : *Find a report news segment which illustrates a conflict?*
This query is formulated by the following Datalog rule :
$Q(X) \leftarrow ReportSequence(X) \wedge Conflict(Y) \wedge SubjectOf(X, Y)$.
Applying the first step of the rewriting algorithm will produce the query :
$Q_1(X) \leftarrow ReportSequence(X) \wedge War(Y) \wedge VideoSegment(X) \wedge Event(Y) \wedge Event\_Occurence(X, Y)$
This query contains some form of redundancy. The conjunction of *VideoSegment(X)* and *ReportSequence(X)* will give only *ReportSequence* since *ReportSequence* concept is a specialization of the *VideoSegment* concept. The conjunction of *Event(Y)* and *War(Y)* will give only *War* instance since *War* is a specialization of *Event*. The elimination of these redundancy will give the query:
$Q_2(X) \leftarrow ReportSequence(X) \wedge War(Y) \wedge Event\_Occurence(X, Y)$.
By applying the last step of the algorithm, we can produce the new query
Q_2'$(X) \leftarrow ReportSequence(X) \wedge IraqWar(Y) \wedge Event\_Occurence(X, Y)$, where the event *War* is replaced by the event *IraqWar*.

Hence, the original query Q is rewritten into the set of queries $\{Q_2, Q_2'\}$.

3.3 Query Translation

Datalog is a relationally complete query language that can express relational algebra in terms of the relational data model and XQuery is a functional query language in which a query is generally represented as an expression in terms of the XML's tree-like data model. In most cases, Datalog queries cannot simply be represented through simple XQuery expressions, instead they usually have to be translated into sets of XQuery function calls. As the development of XQuery functions depends on individual XML data schema to be queried, in order to express the relational algebra and represent corresponding Datalog queries, we need to develop sets of XQuery functions based on the XML representation of the common data view.

In our framework, this set of XQuery functions has been developed when we defined the set of mapping rules between the conceptual layer and the MPEG-7 metadata layer.

The syntax used to build XQuery queries from Datalog queries is given in Table 4. The construction of the XQuery queries requires using the Datalog query and the set of mapping rules between the conceptual layer (concepts, roles and predicates) and the MPEG-7 metadata layer (the XQuery functions defined over the MPEG-7 descriptions).

Table 4: XQuery Syntax of the Datalog queries

"**Let**" "$"*Var_Structure* ":=" *Structure_Unit_Function*;
"$"*Var_*$Concept_1$ ":= $Concept_1$*_Function*;
.....;
"$"*Var_Concept*$_n$ ":=" *Concept*$_n$*_Function*;

"**Where**" (*Condition* ("and" | "or") *Condition*)*
"**Return**" ("$" VarStructure+)

Condition := *Struct_Attr_Cond* | *Concept_Attr_Cond* | *Concept_Rel_Cond* | *Struct_Concept_Cond*
Struct_Attr_Cond := *Struct_Attr_Function Operator*$_1$ *Attr_Value* | *Operateur*$_2$ "("*Struct_Attr_Function* ","*Attr_Value*")"
Concept_Attr_Cond := *Concept_Attr_Function Operator*$_1$ *Attr_Value* | *Operateur*$_2$ "("*Concept_Attr_Function* ","*Attr_Value*")"
Concept_Rel_Cond := Concept_Rel_Function
Concept_Rel_Cond := Concept_Rel_Function
Struct_Concept_Cond := Struct_Concept_Function
$Operator_1$:= >| < | = (Arithmetic Operators)
$Operator_2$:= contains | string | substring |... (XQuery operator functions)

Legend :
Var_Structure, Var_Concept$_i$: string variables expressions.
Struct_Unit_Function, Struct_Attr_Function : XQuery functions which deal with concepts related to structure and their attributes.
Concept_Function, Concept_Attr_Function : XQuery functions related to the domain content concepts and their attributes.
Concept_Rel_Function : XQuery functions which state if a relation exists between two concepts.
Struct_Concept_Function : XQuery function which relates the structural concepts to their content concepts.

The translation of Q_2 and Q_2' into XQuery queries and their evaluation over the MPEG-7 description (cf. Figure 1) will return the video segment identified by "*ID-V1*", which is not returned using XQuery only.

4 Application of the Approach for the Tarchna Project

We are working in the context of the european project T.Arc.H.N.A (Towards Archaeological Heritage New Accessibility) [4]. The aim of this project is to develop a virtual museum by using models and tools that will allow a wide range of public to access semantically the archaeological heritage. The available archaeological resources are mainly multimedia. An image database has already been developed using PHP/MySQL plateform and a domain ontology is developed using CIDOC model [5]. The CIDOC model provides tools to build ontologies for cultural heritage information.

A prototype for the description of the archaeological resources using MPEG-7 enriched by semantic knowledge is in the process of developing. This prototype will also allow semantic access to the cultural heritage described in a virtual museum.

5 Related Work

The presented work deals with two specific research works : adding semantic knowledge to describe multimedia data using MPEG-7 and semantic retrieval of MPEG-7 descriptions.

5.1 Adding semantics to MPEG-7 descriptions

The idea of adding semantics to multimedia data has been considered in several works [8, 13, 3, 5, 12]. Most of them translate MPEG-7 *Ds* and *DSs* into knowledge representation formalism which is able to express the semantics of the descriptions and to provide reasoning mechanisms. The formalisms used are : RDFS and DAML+OIL [8], OWL [13, 5, 12] and XDD [3]. The multimedia data are then processed using these new formalisms and their appropriate query languages.

The drawbacks of such approach is that MPEG-7 is not exploited especially for the low-level features, which have the appropriate datatypes defined using the *DDL*, and the structural aspects expressed in XML. A knowledge representation formalism does not allow to deal with the low-level features appropriately and the structure of the data is generally lost when using this formalism. Moreover, most of the works propose only the translation of the MPEG-7 *Ds* and *DSs* and they do not consider the domain knowledge related to the described content or the appropriate query languages for the obtained descriptions.

In [12], OWL and MPEG-7 are combined to represent audiovisual documents. The structural aspects are expressed using the MPEG-7 *DSs*, whereas the content is described using RDF assertions. RQL is used to retrieve the RDF knowledge base. The two descriptions are linked by using XPath expressions.

Our work is quite similar to [12]. However, instead of translating the MPEG-7 vocabulary to a knowledge based language, we add a conceptual layer to cope with the representation limitations of MPEG-7 standard. This layer introduces domain knowledge related to

[4]http://www.tarchna.org
[5]http://cidoc.ics.forth.gr/

the structure and the content of the described data. These knowledge are introduced in the MPEG-7 descriptions during the annotation process and exploited during the query formulation process, instead of the evaluation step.

5.2 Query Languages to retrieve the MPEG-7 descriptions

Currently, the approaches used for querying MPEG-7 descriptions can be classified into three main groups [4].

The first group [9] uses directly XQuery as the retrieval language of the MPEG-7 descriptions. However, XQuery requires advanced knowledge of the MPEG-7 details in order to express a precise query. Moreover, XQuery allow only exact matching during the evaluation process.

The second group [6, 11] defines new query languages to support retrieval of MPEG-7 descriptions. The disadvantages of such approach is that the query languages are specific to their application and cannot be used in another context.

The third group [4] proposes to use an adaptation of XQuery called *SVQL (Sematic Views Query Language)* to retrieve the MPEG-7 descriptions. *SVQL* is a high level query languages, which allows users to express their requirements following the semantic view model (*PhysicalView, ProductionView, ThematicView, VisualView and AudioView*) in a concise, abstract and precise way.

Our approach to retrieve multimedia data described using MPEG-7 is to use the conceptual layer to formulate the queries in a flexible and precise way. Advantages of using the conceptual layer is that users can formulate concept based queries instead of keywords based queries and a query rewriting process is performed in order to make implicit information explicit. This approach will increase the retrieval effectiveness. A query translation algorithm is used to reformulate the conceptual queries to XQuery queries, which will be evaluated over the MPEG-7 descriptions in the metadata layer.

6 Conclusion and future work

In this paper, we have presented a domain knowledge based approach to retrieve multimedia data described using the MPEG-7 standard. A conceptual layer is added on top of the MPEG-7 layer in order to provide MPEG-7 with a formal definition of domain concept and relations and to use the representation language to formulate queries in more precise and flexible way.

User queries are rewritten using the knowledge of the conceptual layer, to make explicit the implicit information on one hand, and to include more specific concepts on the other hand. The rewritten queries are then reformulated into XQuery formalism using the mapping rules and a set of XQuery functions defined over the MPEG-7 descriptions.

As future work, we will investigate the optimization of query rewriting algorithm by selecting only the most similar rewritten queries to the original query. Furthermore, we will introduce role properties (equivalence, inverse, etc.)

References

[1] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The description logic handbook: theory, implementation, and applications*. Cambridge University Press, New York, NY, USA, 2003.

[2] A.B. Benitez, D. Zhong, S.F. Chang, and J.R. Smith. MPEG-7 MDS content description tools and applications. *Lecture Notes in Computer Science*, 2124:41– 52, 2001.

[3] Ano Chotmanee, Vilas Wuwongse, and Chutiporn Anutariya. A schema language for mpeg-7. In *Proceedings of International Conference on Asian Digital Libraries (ICADL'02)*, pages 153– 164. Springer, 2002.

[4] N. Fatemi, O. Abou Khaled, and G. Coray. An XQuery adaptation for MPEG-7 documents retrieval. *XML conference and Exposition*, Decemeber 2003.

[5] Roberto Garcia and Oscar Celma. Semantic Integration and Retrieval of Multimedia Metadata. In *Proceeding of the 5th International Workshop on Knowledge Markup and Semantic Annotation (SemAnnot 2005)*, Galway, Ireland, November 2005.

[6] A.P. Graves and M. Lalmas. Video Retrieval using an MPEG-7 based Inference Network. In *The* 25^{th} *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 339– 346. ACM Press, August 2002.

[7] T.R. Gruber. A translation approach to portable ontology specifications. *Knowl. Acquis.*, 5(2):199–220, 1993.

[8] J. Hunter. Adding multimedia to the semantic web - building an MPEG-7 ontology. In *Proceedings of the First Semantic Web Working Symposium (SWWS)*, pages 261– 281, 2001.

[9] J. Kang, J. H, and al. An xquery engine for digital library systems. In *Proceedings of 3rd ACM/IEEE-CS Joint Conference on Digital Libraries*, Houston, Texas.

[10] Alon Y. Levy and Marie-Christine Rousset. Combining horn rules and description logics in carin. *Artif. Intell.*, 104(1-2):165–209, 1998.

[11] P. Liu and L. Sushu. Queries of digital descriptions in MPEG-7 and MPEG-21 XML documents. *XML Europe 2002, Barcelona, Spain*, May 2002.

[12] Raphaël Troncy. *Formalisation des connaissances documentaires et des connaissances conceptuelles à l'aide d'ontologies : application à la description de documents audiovisuels*. Computer science PHD thesis, Joseph Fourier University, Grenoble, France, 2004.

[13] Chrisa Tsinaraki, Panagiotis Polydoros, and Stavros Christodoulakis. Integration of OWL Ontologies in MPEG-7 and TV-Anytime Compliant Semantic Indexing. In *Proceedings of the 16th International Conference on Advanced Information Systems Engineering (CAiSE'04)*, pages 398–413, 2004.