

Queries of Digital Content Descriptions in MPEG-7 and MPEG-21 XML documents

Peiya Liu <pliu@scr.siemens.com>
Liang H. Hsu <lh@scr.siemens.com>

Abstract

The emerging ISO/IEC MPEG-7 and MPEG-21 multimedia standards use XML Schema for digital content descriptions and digital item declarations. They have posed an interesting challenge to XML query language design. The paper shows certain critical query specification issues for MPEG-7 and MPEG-21 XML documents and illustrates a logic method to handle the limitations in current text-oriented XML query languages for retrieving multimedia content and digital items.

Table of Contents

1. Introduction	1
2. MPEG-7 and MPEG-21 XML Documents	2
2.1. MPEG-7 XML Documents for Multimedia Content Descriptions	2
2.2. Spatial and Temporal Datatypes	5
2.3. MPEG-21 XML Documents for Digital Item Declarations	6
3. Query Specification Issues in MPEG-7 and MPEG-21	8
3.1. Predicate-based Query Specifications	9
4. MPEG-7 and MPEG-21 Structured Queries	10
5. Related Work	11
6. Concluding Remarks	11
Bibliography	11

1. Introduction

MPEG-7 is an emergent ISO/IEC standard and formally named as "Multimedia Content Description Interface". Unlike the previous MPEG[MPEG Web Site] compression standards MPEG-1, MPEG-2 and MPEG-4, MPEG-7 aims to create a standard for describing the multimedia content to enable the integration of production, distribution and content access paradigm. This MPEG-7 standard uses an XML Schema to describe multimedia objects such as video, audio images, etc. as spatial, temporal or visual XML datatypes. This type of multimedia XML documents may include descriptions about both static/spatial media (such as text, drawings, images, etc.) and time-based media (such as video, audio, animation, etc.). The content can be further organized into three major document structures: hierarchical, hyperlinked, and temporal/spatial structures.

A related ISO/IEC standard MPEG-21 is defining a multimedia framework to support the content delivery chain. This multimedia content delivery chain encompasses content creation, production, delivery and consumption. To support this, several key elements have identified: digital item declaration, identification, description, content handling, intellectual property management, digital item rights management, etc. *Digital items* are defined as structured digital objects, including representation, identification and metadata description. This paper will focus on two types of MPEG-21 XML documents (Digital Item Declaration, and Digital Item Identification and Description) and illustrate the relationships to MPEG-7 XML documents for digital content and item queries.

MPEG-7 and MPEG-21 standards have posed an interesting challenge to XML query language design in covering different XML aspects. This paper shows some critical specification issues in forming MPEG-7 and MPEG-21 XML queries. In addition, a logic formalism called *Path Predicate Calculus* [Liu2 00] is illustrated to handle the limitations in current text-oriented XML query languages for multimedia XML queries. In this formalism, the

atomic logic formulas are *element predicates* rather than relation predicates in relational calculus. The queries describe a desired document tree by specifying *path predicates* that the tree document elements must satisfy. Spatial, temporal and visual datatypes and relationships can also be described in this formalism for content retrieval and identification in MPEG-7 and MPEG-21 XML documents.

The rest of the paper is organized as follows. Section [Section 2](#) introduces MPEG-7 and MPEG-21 XML documents. Section [Section 3](#) addresses certain critical query specifications issues in MPEG-7 and MPEG-21 XML documents. We depict the proposed query language MMDOC-QL and its embedded path predicate calculus. The language is used to illustrate how to specify multimedia objects as temporal/audio/visual datatypes for content and digital item retrievals. The Path Predicate logic can be used to express document element addresses for desired digital content and items in MPEG-7 and MPEG-21 queries. Section [Section 4](#) describes more examples of structured MPEG-7 and MPEG-21 queries. Section [Section 5](#) discusses related work in multimedia document and query languages. Section [Section 6](#) provides some conclusion remarks

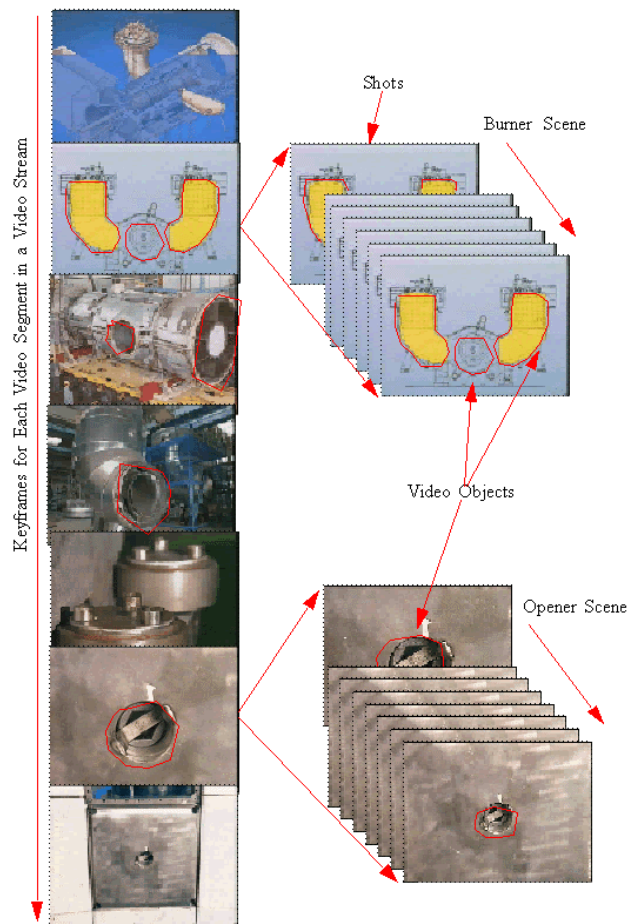
2. MPEG-7 and MPEG-21 XML Documents

MPEG committee has decided to adopt XML Schema Language for describing multimedia content in MPEG-7 and for specifying digital item declarations in MPEG-21. The structures of MPEG-7 and MPEG-21 XML documents can be complex due to characteristics of spatial, temporal and logical relationships among multimedia objects and digital items. The structures could be quite different from text-oriented documents.

2.1. MPEG-7 XML Documents for Multimedia Content Descriptions

The document *mpeg7video.xml* we used for the query is a MPEG-7 XML document for describing the content of a turbine inspection video [Figure 1](#). This video has been processed and video objects are extracted for generating this MPEG-7 description.

Figure 1. An Industrial Turbine Inspection Video



This MPEG-7 document consists of an AudioVisualContent of type "VideoType" named "TurbineVideo". The video is segmented into scenes and the scenes are described by using the "SegmentDecomposition" tag with the decomposition type "SpatioTemporal". Each segment or scene can have several objects of interest and they are described here as well. In particular, let's take a look at the second segment which has an id "BurnerScene" and is of type "MovingRegionType". We use the "MovingRegionType" tag because there are multiple objects that move over time. The detailed descriptions are as follows.

Figure 2. mpeg7video.xml document

```

<Mpeg7Main name="turbinevideo" version="1.0" copyright="Siemens">
  <ContentDescription xsi:type="ContentEntityDescriptionType">
    <AudioVisualContent xsi:type="VideoType">
      <Video id="TurbineUS">
        <TextAnnotation>
          <FreeTextAnnotation xml:lang="en-us">Turbine Inspection
          </FreeTextAnnotation>
        </TextAnnotation>
        <SegmentDecomposition gap="false" overlap="false"
          decompositionType="spatioTemporal">

          <!-- First Scene "Overview" -->
          <Segment xsi:type="MovingRegionType" id="OverviewScene">
            ... </Segment>

          <!-- The Second Scene "Burner" -->
          <Segment xsi:type="MovingRegionType" id="BurnerScene">
            <TextAnnotation>
              <FreeTextAnnotation xml:lang="en-us">Burner
              </FreeTextAnnotation>
            </TextAnnotation>
            <SegmentDecomposition gap="false" overlap="false"
              decompositionType="spatioTemporal">

              <!-- The first VideoObject -->
              <Segment xsi:type="MovingRegionType" id="MR001">
                <MediaTime> ... </MediaTime>
                <SpatioTemporalLocator>
                  <MediaTime> ...</MediaTime>
                  <ParameterTrajectory MotionModel="0">
                    <MediaTime>...</MediaTime>
                    <RegionLocator>
                      <Poly><Coords dim="4 2">
                        5 25 10 20 15 10 10</Coords></Poly>
                    </RegionLocator>
                    <Parameters KeyPointNum="25">
                      <WholeInterval>
                        <MediaIncrDuration TimeUnit="P1S">300
                        </MediaIncrDuration>
                      </WholeInterval>
                      <InterpolatedValue>
                        <!-- Total 25 interpolated points -->
                        <KeyValue Type="startPoint" dimension="2">
                          5.0
                        </KeyValue>
                        <KeyValue ... > ... </KeyValue>
                        ...
                      </InterpolatedValue>
                      <InterpolatedValue>
                        <!-- Total 25 interpolated points -->
                        <KeyValue Type="startPoint" dimension="2">
                          25.0
                        </KeyValue>
                        <KeyValue ... > ...</KeyValue>
                        ...
                      </InterpolatedValue>
                    </Parameters>
                  </ParameterTrajectory>

                  <ParameterTrajectory MotionModel="0">...
                  </ParameterTrajectory>
                  ...
                </SpatioTemporalLocator>
              </Segment>
              <!-- The Second VideoObject -->
              <Segment xsi:type="MovingRegionType" id="MR002">
                <MediaTime> ... </MediaTime>
                <SpatioTemporalLocator>... </SpatioTemporalLocator>
              </Segment>
              ...
            </SegmentDecomposition>
          <MediaTime>
            <MediaTimePoint>...</MediaTimePoint>
            <MediaDuration>...</MediaDuration>
          </MediaTime>
        </Segment>

        <!-- The "Opener" Scene -->
        <Segment xsi:type="MovingRegionType" id="OpenerScene">
          ... </Segment>

        <!-- The last Scene -->
        <Segment .. > ... </Segment>
      </SegmentDecomposition>
    <GoFGoPColor> ... </GoFGoPColor>

  </Video>
</AudioVisualContent>
</ContentDescription>
</Mpeg7Main>

```

The video segments (scenes) can be further broken up using the same "SegmentDecomposition" tag and is again of type "SpatioTemporal". Taking a closer look, we find that the first object has an id "MR001", and it moves over time, the trajectory of which is given here. The tag "MediaTime" provides the duration of the object. The location of the object is defined temporally using the tag "ParameterTrajectory". At the first frame or instance where the object first appears, the location is given by a 4x2 matrix defining the four coordinates of the object boundary. Any number of coordinates can be used to define the boundary. The complete interval, defined using "WholeInterval" tag, consists of 300 secs. The base time unit is 1 sec (P1S). There are 25 node points which determine the "KeyPointNum". The "InterpolatedValue" tag is used to define the corresponding coordinates of the object of interest at

each of these nodes. Each KeyValue gives the coordinate location for a single vertex. This is done for all four vertices that constitute the boundary in this case. Since the value of attribute "MotionModel" is 0, it means a linear model. For frames that lie within these nodes, a simple linear interpolation is used to determine the actual location on that frame. The rest of the example follows the above format to describe other objects and scenes in the video.

We have developed a tool based on the scene change technique [Chakraborty 99] to generate such a description from a video as follows. At first, the video is broken down temporally into scenes or shots using scene change detection algorithms that can detect both, abrupt as well as gradual changes. Next, the users identify objects of interest within these scenes and outline them. These are then tracked over time in a semi-automatic way. Wherever there is a significant motion change and a linear mode is inadequate, a node point is created. To make things simpler as described in the above example, one can also divide the interval into equal segments. At these boundaries, node points are created and the object outline is described.

2.2. Spatial and Temporal Datatypes

In previous work [Liu1 00] [Liu2 00] [Liu3 01], we have shown that multimedia objects can be described as spatial, temporal and visual datatypes by using abstract datatype technique (ADT). The composite datatypes can be constructed from more primitive ones. These datatypes can be formalized as XML element datatypes within W3C XML Schema [XML Schema Part 1: Structures] framework, particularly the datatype part [XML Schema Part 2: Datatypes]. The relationships of multimedia objects are often derived from element datatypes rather than from element hierarchical relationships. The relationships can be even predefined as another complex datatypes for multimedia XML documents. A similar technique for specifying moving objects was proposed by [Erwig 99] [Manolopoulos 00] in relational databases.

At 51th MPEG meeting in March 2000, MPEG committee has decided to adopt XML Schema Language as MPEG-7 Description Definition Language (DDL) for describing multimedia content. Since then, a comprehensive set of audio and visual datatypes is being developed based on XML datatype mechanisms. The main components of the MPEG-7 standard are: Descriptors (Ds) for describing audio and visual features, Description Schemes (DSs) for describing the structure and semantics of the relationships between components. The components can be either Ds or DSs. There is also a description definition language for allowing the creation of a new D or DS and for allowing extension of existing Ds or DSs.

MPEG-7 datatype hierarchy can be viewed as follows. The base level datatypes are: Mpeg7Type, basic datatypes, reference datatypes, unique identifier datatypes, and time datatypes. Mpeg7Type provides the main basic abstract type of MPEG-7 type hierarchy. From Mpeg7Type, DSType (Description Scheme Type) and DType (Descriptor Type) are derived. From DSType, SegmentType, RelationType, GraphType, VisualDSType and AudioDSType are derived. From DType, VisualDType and AudioDType are derived. From SegmentType, StillRegionType, VideoSegmentType, MovingRegionType, AudioSegmentType AudioVisualSegmentType, and SegmentDecompositionType are derived. Some of the temporal, audio and visual datatypes are described as follows.

- MPEG-7 temporal datatypes are used to specify either real world time or time used for audiovisual media. They are all from MPEG-7 time datatypes. These time datatypes are: TimeType (for real world time) and MediaTimeType (for the time used in audio and visual media data). Each one of them consists of a time point description and a time duration description. Typical 13 temporal relationships [Allen 83], such as after, before, meets, etc., can also be defined as MPEG-7 BinaryTemporalSegmentRelationType, which is derived from MPEG-7 RelationType in the type hierarchy.
- MPEG-7 visual datatypes are used to specify visual properties of multimedia objects such as spatial, color, texture, motion, location, etc. All visual datatypes are derived from VisualDType. The spatial datatypes are used to specify geometric data such as points, polylines or regions, etc. The composite visual datatypes can be constructed from these primitives. Examples are RegionShapeType, ConturShapeType, RegionLocatorType, etc. In our example, we use RegionLocatorType which consists of points in pairs of coords matrix datatype for describing video objects.
- MPEG-7 audio datatypes are used to specify audio content. Examples are SoundEffectCategoryType, SilenceType, etc. All audio datatypes are derived from AudioDType

MPEG-7 temporal, audio and visual datatypes can be further composed into more complex MPEG-7 datatypes by using XML datatype definition mechanism, predefined MPEG-7 Ds, or predefined MPEG-7 DSs. The common used DSs for composing the content are: *SegmentDecomposition DS*, *Segment DS* (e.g. *MovingRegion DS*, *StillRegion DS*, etc), *Graph DS* and *Relation DS*. Each DS or D itself is a MPEG-7 datatype. For example, MPEG-7 ParameterTrajectory datatype, SpatioTemporalLocator DS and MovingRegion DS are all spatio-temporal composite datatypes, called ParameterTrajectoryType, SpatioTemporalLocatorType and MovingRegionType, respectively for specifying spatial data changing over time. These spatio-temporal datatypes are constructed from primitive temporal datatypes (e.g., MediaTime) with spatial datatypes (e.g., RegionLocatorType) or previously defined spatio-temporal datatypes. In addition to content description DSs in MPEG-7, there are many other DSs that facilitate content navigation, content organization, content management, and user interaction. MPEG-7 DSs are used to support varieties of multimedia content retrievals such as semantics-based retrievals, structured-based retrievals, model-based retrievals, and navigation/browsing (e.g., content summary). Thus, MPEG-7 content access requires an expressive query language to support the intended information retrievals from XML-based multimedia content descriptions.

2.3. MPEG-21 XML Documents for Digital Item Declarations

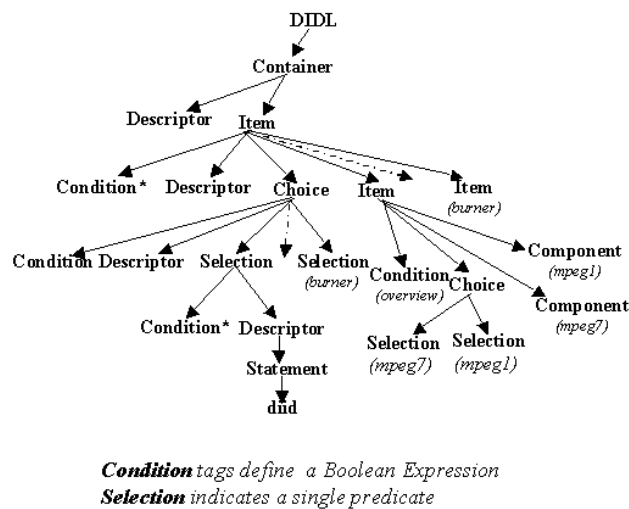
MPEG-21 committee has also decided to use a uniform and interoperable XML schema to declare digital items. A *Digital Item* is a structured and hierarchical digital object containing several multimedia elements (e.g., sounds and video clips) and meta data. In order to declare the structure of such a Digital Item to users for interacting with the content, MPEG-21 is developing an XML-based language called the Digital Item Declaration Language (DIDL) for expressing the relationships of different objects within a particular Digital item. In MPEG-21 Part 4: Digital Item Identification and Description (DII&D), XML schema is used for identification and description of any digital items regardless of its nature, type or granularity. For content identification, the DII&D provides the ability to associate Uniform Resource Identifiers (URIs) with an entire Digital Item or its parts. For content description, the DII&D framework provides the ability to include meta data from various sources and in various formats including XML or plain text. DII&D allows the binding of existing Description Schemes to meta data to allow the correct processing of such meta data. This enables to include MPEG-7 or other descriptions. Example of DIDL XML document is shown [Figure 3](#) below.

Figure 3. Example of MPEG-21 XML document

```
<DIDL xmlns="urn:mpeg:mpeg21:20002/01-DIDL-NS"
      xmlns="urn:mpeg:mpeg21:20002/01-DIID-NS">
  <Container>
    <Item>
      <Choice minSelections="1" maxSelections="2">
        <Descriptor>
          <Statement type="text/plain">
            <diid:Identification>
              urn:mpeg:mpeg21:cid:1702.F109
            </diid:Identification>
          </Statement>
        </Descriptor>
        <Selection select_id="Overview">
          <Descriptor>
            <Statement type="text/plain">
              Burner
            </Statement>
          </Descriptor>
        </Selection>
        ...
        <Selection select_id="Opener">...</Selection>
        <Selection select_id="...">...</Selection>
      </Choice>
    </Item>
    <Item>
      <Condition require="Overview">
        <Choice minSelections="1" maxSelections="1">
          <Selection select_id="mpeg7">
            <Descriptor>
              <Statement type="text/plain"> XML
            </Statement>
          </Descriptor>
        </Selection>
        ...
      </Choice>
    </Item>
    <Component>
      <Condition require="mpeg7">
        <Resource ref="overview.xml" type="text/plain"/>
      </Component>
    <Component>
      <Condition require="mpeg1">
        <Resource ref="overview.mpg" type="video/mpg"/>
      </Component>
    </Item> ...
  </Item> ... </Item>
</Container>
</DIDL>
```

In this example, we declare the digital item of our inspection video in a MPEG-21 XML form. This digital item declaration consists of several subitems of video scenes such as overview, opener, etc. Each digital subitem has two kinds of digital item components: mpeg7 XML form or mpeg1 video form. Choice and Selection elements are used to specify the proper configuration of user-desired digital items. The *select_id* attribute value of Selection identifies a predicate such as “burner”, “overview”, etc. In general, this predicate is related to one or more Condition elements somewhere within an Item or a Component. The set of Condition elements defines a Boolean combination of predicate tests. For simplifying the issue, we only use one Condition for each Component in this example. The *require* attribute value of Condition defines an AND logic combination of predicates. The multiple Condition elements within a given parent are combined as an OR logic combination. The set of Condition elements denotes the parent element as being conditionally selected upon the truth of the Boolean combination of predicate tests. The Digital item identification and descriptions (DIID) associated with digital items are contained in Statement element of any descriptor element as shown in [Figure 4](#) below

Figure 4. Structural Hierarchy in Digital Item Declaration Documents



A digital item could be a complex structured XML document. A digital item should not be considered just as a single entity since it can hierarchically consist of multiple subitems and components. Each subitem and component could be used to declare a form of digital item representation such as file format, media type, or individual part, etc. A DIDL XML document can be viewed as a *structured catalog document* to declare and package multimedia information items with logic-based selection descriptions for exchange. DIDL documents queries can be viewed as structured catalog search based on the semantics of selection logic in the documents.

3. Query Specification Issues in MPEG-7 and MPEG-21

MPEG-7 and MPEG-21 XML documents pose an interesting challenge for XML query language design due to different aspects of XML structure and datatype usage. In the following, we address three crucial query specification issues in MPEG-7 and MPEG-21 XML document retrievals.

1. *Intensional Data and Relationship Specifications* Extensional data and relationships are those data and relationships that are explicitly stored in XML documents. Intensional data and relationships are those that are computed or deduced from extensional data and relationships in XML documents. Many relationships of multimedia objects in MPEG-7 documents are derived from stored content descriptions based on element datatypes or DS schemes rather than from XML element hierarchical relationships. Thus, the capability of expressing the relationships in query language constructs is crucial for MPEG-7 query specifications. Examples of the relationships are point-inside, region-overlap, etc.

In addition, many spatial and temporal data are represented in an implicit manner inside MPEG-7 XML documents unlike data in relational databases. For instance, an instance of MediaTime element in MPEG-7 means a time interval. It is important to express those implicit MediaTimePoints in that interval in query language since identification of multimedia objects may depend on a particular MediaTimePoint

2. *Fully Document Addressing Specifications* MPEG-7 and MPEG-21 XML documents often contain irregular document structures. For instance, An Item or Container tag can be inside another Item or Container tag in MPEG-21 XML documents. The desired digital item content is based on the semantics of selection logic. In MPEG-7 XML documents, a Segment tag can also be inside another Segment tag . MPEG-7 content structures are based on their own datatypes and description schemes (DSs) rather than on XML element hierarchy. MPEG-7 XML documents normally are not data-centered documents which are collection of almost identical structures. A full document addressing query construct is needed to precisely specify the desired document locations in recursive or contextual XML structures for retrieving information.
3. *Co-occurrence Constraints Specifications* The multimedia object descriptions have temporal and spatial synchronization constraints in nature. The scheduled structures are inherently inside multimedia documents. Thus MPEG-7 XML document elements normally have co-occurrence constraints, e.g. if one XML element for a

multimedia object description has attribute A in certain spatial location, it must have the same attribute A in another location in spatial synchronization. Another example is: two multimedia objects appear inside the same spatial region at the same time in temporal synchronization.

3.1. Predicate-based Query Specifications

In answering to these specification issues, we have designed an experimental XML query language MMDOC-QL. This language embeds within it a logic formalism, called **Path Predicate Calculus**, to specify queries. Formulas in path predicate calculus are restricted forms of first-order predicate. For these logic-based queries and manipulations, we have designed two important predicates: **element predicates** and **path predicates** for asserting logical truth statements about document elements in a document tree. This path predicate calculus can adequately support the co-occurrence constraints and document addressing specifications for querying XML documents. To support intensional data and relationships specifications in this logical formalism, certain stereotypical logic operators are incorporated for asserting multimedia object relationships in this query language. Examples of the multimedia logic operators are, OVERLAP(element1: RegionLocatorType, element2: RegionLocatorType), TRAJECTORY(element1: MovingRegionType, element2: MediaTimePoint), etc. Another logic operator MEMBERP is also included for asserting intensional data such as MediaTimePoint in the language constructs.

In the following, we illustrate MMDOC-QL for specifying MPEG-7 XML document queries. An example of query is in the form of "finding all video object ids and show up time over a particular area".

```

GENERATE  <List>
          <Videoobject>%objectid</videoobject>
          <ShowUpTime>%t</ShowUpTime>
        </List>
PATTERN  {"HR"[0-9][0-9][0-9]/%objectid}
        (<region> ... </region>/%focusarea)
FROM     mpeg7video.xml
CONTEXT  ((<Segment> WITH xsi:type="MovingRegionType"
          id=%objectid AT %movingregion )
          CONTAINING
          (<SpatioTemporalLocator> DIRECTLY CONTAINING
           (<MediaTime> AT %x)))
AND MEMBERP(%t %x)
AND OVERLAP(TRAJECTORY(%movingregion %t) %focusarea))

```

In MMDOC-QL, there are four clauses: OPERATION clause (either GENERATE, INSERT, DELETE, or UPDATE) is used to describe the logic conclusions in the form of allowable element predicates and path predicates. In this paper, we focus on retrieval operation clause by using keyword GENERATE for MPEG-7 XML queries. PATTERN clause is used to describe the domain constraints of free logical variables including tag, attribute, content, address and datatype, by using regular expressions. FROM clause is used to describe source documents for querying. CONTEXT clause is used to describe logic assertions about document elements in allowable logic formulas in path predicate calculus. FROM and CONTEXT clauses are paired together and there could be multiple pairs for describing multiple sources. The logic variables are indicated by "%" such as "%objectid". Queries in MMDOC-QL are equivalent to finding all proofs to existential closure of logical assertions.

In this example, the path logic formula (<Segment> WITH xsi:type="MovingRegionType" ... <MediaTime> AT %x)) in CONTEXT clause asserts that element "Segment" with id equal to %objectid contains element "SpatioTemporalLocator" of which the video objects are located during MediaTime %x.

In general, (<%t> WITH attributel=%x1, ..., attributen=%xn AT %a CONTAINING %c) is an English-like notation for **element predicate** E(x1, x2, ..., xn, c, t, a) which stands for a logic assertion that element "t" at address "a" contains "c" with attributes x1, x2, ..., xn in a document tree. For brevity, we can also use short versions with only needed variables in logic queries such as (<%t> WITH attributel=%x1, ..., attributen=%xn), (<%t> CONTAINING %c), etc., if a full version can be implied clearly in the context.

A path logic formula is a composition of element predicates by XPath[XPath 99] "axis-operators". Examples are (a) parent/child relationship operators such as: INSIDE, DIRECTLY INSIDE, CONTAINING, DIRECTLY CONTAINING, etc. and (b) the sibling relationship operators such as: BEFORE, IMMEDIATELY BEFORE, AFTER, IMMEDIATELY AFTER, SIBLING, IMMEDIATELY SIBLING, etc. such as DIRECTLY CONTAINING, etc. Note that here we use a *logic form* of axis concepts defined in XPath since path formula in Path Predicate Calculus are logical statements for asserting logical truths. An example of the **path predicate** is: (<bibref>

INSIDE (<gcapaper> CONTAINING (<fname> CONTAINING "Peiya") AND (<surname> CONTAINING "Liu">)))for specifying all bibref elements inside Peiya Liu's paper.

The domain of logical variable %objectid is restricted to be strings beginning with “MR” followed by digits in PATTERN clause. The logic variable “%t” is to be used to bind the MediaTimePoint in this MediaTime interval “%x” during logic computation. TRAJECTORY operator is used to assert trajectory region from a moving region %movingregion at MediaTimePoint %t, and OVERLAP is a spatial logic operator for further asserting that the desired object region is also overlapped with the focus area.

4. MPEG-7 and MPEG-21 Structured Queries

MPEG-7 and MPEG-21 XML documents can organize multimedia content in more structured manner to support better visual information retrievals[Del Bimbo 99] beyond feature-based content retrievals and provide a better multimedia information delivery and exchange. To benefit this, XML query language constructs need to have very expressive power about document structures and addressing specifications. In the following examples, more complex MPEG-7 and MPEG-21 structured content queries are given to illustrate document addressing specifications in this logic formalism.

In the example of MPEG-7 XML query, we add more constraints in CONTEXT clause in the form of “find out only those objects in the focus area, but shown up in a scene which appears either immediately before or after Burner scene”. This query requires an expressive power for specifying the contexts of objects by a path formula about addressing constraints about parent/ancestor/child and sibling relationships among document elements in this recursive video segment structure.

```

GENERATE  <List>
          <Videoobject>%objectid</videoobject>
          <ShowUpTime>%t</ShowUpTime>
        </List>
PATTERN  {"MR"[0-9][0-9][0-9]/%objectid}
          {<region>...</region>/%focusarea}
          {"*Scene"/%scence}
FROM      mpeg7video.xml
CONTEXT   ((<Segment> WITH xsi:type="MovingRegionType"
            id=%objectid AT %movingregion )
          INSIDE
            ((<Segment> WITH xsi:type="MovingRegionType"
              id=%scence)
             IMMEDIATELY SIBLING
              (<Segment> WITH xsi:type="MovingRegionType"
                id="BurnerScene")))
          CONTAINING
            (<SpatioTemporalLocator> DIRECTLY CONTAINING
              (<MediaTime> AT %x))
          AND MEMBERP(%t %x)
          AND OVERLAP(TRAJECTORY(%movingregion %t) %focusarea))

```

In MPEG-21 XML queries, end users can access digital item declarations in particular desired format and parts during exchange. Example of this query can be in the form of “looking-up the URIs of digital items *Overview* and *Opener* in MPEG-7 format”. This query in MMDOC-QL can be specified in the below.

```

GENERATE  <List>
          <resource comp=%comp>%uri</resource>
        </List>
FROM      mpeg21did.xml
CONTEXT   ((<Resource> WITH ref=%uri)
          IMMEDIATELY SIBLING (
            (<Condition> WITH require="mpeg7")
            DIRECTLY INSIDE
              (<Component>
                SIBLING (
                  (<Condition> WITH require=$comp)
                  AND (( $comp = "Overview")
                      OR ($comp = "Burner")))
                )
            )
          )
        ))

```

5. Related Work

Two kinds of related work are described here. One is related to XML or SGML multimedia documents and the other is related to multimedia query languages.

ISO HyTime [HyTime 97] based on SGML uses Finite Coordinate Space (FCS) to define scheduled structures and events. These event schedules are intentionally designed for HyTime document presentation. FCS defines an abstract and system-independent method of specifying spatial and temporal information separated from content to be presented as event schedules in a multidimensional coordinate space. The design motivation is based on presentation abstraction rather than information retrieval. The indexing scheme support in HyTime is limited to querying spatial/temporal media objects and structures.

W3C SMIL [SMIL 98] is based on XML to define spatial and temporal layouts for SMIL document layout. The layout information is related to media display windows on a screen and media playing time. Thus, the spatial and temporal structures provided in SMIL are also for presentation purpose rather than for storage representation to be accessed. Furthermore, there are structural differences in representation [Rutledge 98] [Liu 99]. Often, the presentation forms are not sufficient for storage representation. Spatial and temporal content descriptions are often less emphasized in presentation-oriented multimedia specifications.

SQL/MM and SQL3/Temporal [SQL Standardization Projects] are new ISO standardization projects for extending database query language capability to specify and manage multimedia objects and temporal information in the relational data model. Both are focusing on integration of time- or space- dependent multimedia objects into relational data models for query. However, multimedia document models impose requirements on querying, which are quite different from this relational table model since not only document content but also document structures must be available for retrieval. These proposed query specifications based on relational data models would limit the retrieval capability for document models.

6. Concluding Remarks

The emerging MPEG-7 and MPEG-21 standard uses XML Schema as multimedia content description and digital item declaration languages. Many proposed XML document query languages such as XML-QL, XQL, YATL, LOREL, XQuery, etc, are available, but adequate query constructs and formalisms are crucial for supporting different aspects of XML document retrievals.

The main contributions of this paper are (1) to identify certain critical specification issues for query language design consideration to support XML retrievals in multimedia content descriptions and digital item declarations . We illustrate these issues by using MPEG-7 and MPEG-21 XML documents, and (2) to propose an alternative approach by using a logic formalism, called path predicate calculus for supporting queries about XML documents, which could have intensional data and relationships, irregular document structures, and co-occurrence structural constraints. The paper intends to show the flavors of document predicates in a logic formalism and the importance for specifying XML document retrievals. We feel that this direction of research is important for XML query language design, development and standardization.

Bibliography

[HyTime 97] ISO/IEC 10744:1997 Hypermedia/Time-based Structuring Language (HyTime), Second Edition.

[XPath 99] XML Path Language (XPath) Version 1.0, W3C Recommendations 16 November 1999

[XML Schema Part 1: Structures] XML Schema Part 1: Structures, W3C Recommendation 02 MAY 2001

[XML Schema Part 2: Datatypes] XML Schema Part 2: Datatypes: W3C Recommendation 02 MAY 2001

[SQL Standardization Projects] http://www.jcc.com/SQLPages/jccs_sql.htm (SQL Standard Reference Page)

- [Chakraborty 99] A. Chakraborty, P. Liu and L. Hsu, Authoring and Widewing Video Documents using SGML structure, 1999 IEEE International Conference on Multimedia Computing and Systems, pp 654-660 Florence, Italy,
- [Rutledge 98] L. Rutledge, L. Hardman, J. van Ossenbruggen and D. C. A. Bulterman, Structural Distinctions Between Hypermedia Storage and Presentation, in Proc. ACM Multimedia 98, September 1998, pp.145-150.
- [Liu 99] P. Liu, Y. F. Day, L. H. Hsu, Automatic Generation of DSSSL Specifications for Transforming SGML Documents into Card-Based Presentations, GCA Markup Technologies 99, PA, USA,
- [Liu1 00] P. Liu, L. H. Hsu, Spatial and Temporal Datatypes: An Approach to Specifying and Querying Multimedia Objects and Scheduled Structures in XML Documents, XML Europe 2000, Paris, 2000
- [Liu2 00] P. Liu, A. Chakraborty, L. H. Hsu, Path Predicate Calculus: Towards a Logic Formalism for Multimedia XML Query Language, Extreme Markup Languages 2000, Montreal, Canada,
- [Liu3 01] P. Liu, A. Chakraborty, L. H. Hsu, A Logic Approach for MPEG-7 XML Document Queries, Extreme Markup Languages 2001, Montreal, Canada,
- [Del Bimbo 99] A. Del Bimbo, Visual Information Retrieval, Published by Morgan Kaufmann, 1999
- [Erwig 99] M. Erwig, R. H. Guting, M. Schneider and M. Vazirgiannis, Spatio-Temporal DataTypes: Approach to Modeling and Querying Moving Objects in Databases, GeoInformatica Vol 3, No 3, 1999 .
- [Manolopoulos 00] Y. Manolopoulos, Y Theodoridis and V. J. Tsotras, Advanced Database Indexing, Kluwer Academic Publishers, 2000
- [SMIL 98] Synchronized Multimedia Integration Language (SMIL) 1.0 Specification, W3C Recommendations 15 June 1998
- [MPEG Web Site] <http://www.cselt.it/mpeg/standards.htm> for MPEG-1,2,4,7,21
- [Allen 83] J. F. Allen Maintaining Knowledge about Temporal Intervals. Comm. ACM 26(11), 1983.

Biography

Wednesday, 22 May 09.00

Peiya Liu

Senior Member of Technical Staff
Siemens Corporate Research, Inc.
Multimedia Documentation Program
Princeton
USA
Email: pliu@scr.siemens.com

Dr. Peiya Liu is a senior member of technical staff at Multimedia Documentation Program, Siemens Corporate Research, Inc. He has many years of experience in applications of multimedia documents in industrial environments. His primary interests are in the areas of multimedia document authoring/processing/management, multimedia tools, innovative industrial applications and standards. He was one of program co-chairs for the first joint IEEE International Conference on Multimedia and Expo and has served as a program committee member in many conferences. He is currently the standards section editor of IEEE Multimedia and on the editorial board of Kluwer Journal of Multimedia Tools and Applications.

Liang H. Hsu

Manager and Distinguished Member of Technical Staff
Siemens Corporate Research, Inc.
Multimedia Documentation Program

Princeton
USA
Email: lh@scr.siemens.com

Liang H. Hsu is a distinguished member of technical staff and manager of Multimedia Documentation Program, Siemens Corporate Research, Inc. Liang received his MS degree in computer science from University of Pittsburgh in 1983. Prior to joining Siemens, he worked for Digital Equipment Corporation as a field service support. His current interests include conversion of legacy documents into SGML/XML, XML-based document composition and hyperlinking for complex products, multimedia document delivery mechanisms, and browsing and navigation support for service-related applications.