1-1-2007

Application of semistructured data model to the implementation of semantic content-based video retrieval system

Lilac A. E. Al-Safadi
King Saud University

Janusz R. Getta
University of Wollongong, jrg@uow.edu.au

Recommended Citation

Al-Safadi, Lilac A. E. and Getta, Janusz R.: Application of semistructured data model to the implementation of semantic content-based video retrieval system 2007, 217-222.
https://ro.uow.edu.au/infopapers/1965

Application of semistructured data model to the implementation of semantic content-based video retrieval system

Abstract

Semantic indexing of a video document is a process that performs the identification of elementary and complex semantic units in the indexed document in order to create a semantic index defined as a mapping of semantic units into the sequences of video frames. Semantic content-based video retrieval system is a software system that uses a semantic index built over a collection of video documents to retrieve the sequences of video frames that satisfy the given conditions. This work introduces a new multilevel view of data for the semantic content-based video retrieval systems. At the topmost level, we define an abstract view of data and we express it in a notation of enhanced conceptual modeling suitable for the formal representation of the semantic contents of video documents. A semistructured data model is proposed for the middle level representation of data. At the bottom level we implement a semistructured data model as an object-relational database. The completeness of the proposed approach is demonstrated through the mappings of a conceptual level into a semistructured level and into an object-relational organization of data. The paper describes a system of operations on semistructured data and shows how a sample query can be represented as an expression built from the operations.

Keywords

model, data, semistructured, application, retrieval, system, video, content, semantic, implementation

Disciplines

Physical Sciences and Mathematics

Publication Details

Al-Safadi, L. A. E. & Getta, J. R. (2007). Application of semistructured data model to the implementation of semantic content-based video retrieval system. International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, UBICOMN 2007 (pp. 217-222). USA: IEEE.

International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies

Application of Semistructured Data Model to the Implementation of Semantic Content-Based Video Retrieval System

Lilac A. E. Al-Safadi
College of Computer Sciences
King Saud University
Riyadh, Saudi Arabia
email:Lilac2k@yahoo.com

Janusz R. Getta
School of Information Technology and Computer Science
University of Wollongong
Wollongong, Australia
email:jrg@uow.edu.au

***Abstract*— Semantic indexing of a video document is a process that performs the identification of elementary and complex semantic units in the indexed document in order to create a semantic index defined as a mapping of semantic units into the sequences of video frames. Semantic content-based video retrieval system is a software system that uses a semantic index built over a collection of video documents to retrieve the sequences of video frames that satisfy the given conditions.**

This work introduces a new multilevel view of data for the semantic content-based video retrieval systems. At the topmost level, we define an abstract view of data and we express it in a notation of enhanced conceptual modeling suitable for the formal representation of the semantic contents of video documents. A semistructured data model is proposed for the middle level representation of data. At the bottom level we implement a semistructured data model as an object-relational database. The completeness of the proposed approach is demonstrated through the mappings of a conceptual level into a semistructured level and into an object-relational organization of data. The paper describes a system of operations on semistructured data and shows how a sample query can be represented as an expression built from the operations.

I. INTRODUCTION

Video documents provide the viewers with the wealth of information not available in any other media. In order to facilitate search and access to the large collections of video documents, information need to be extracted, formatted, indexed, and stored in a database system. A lot of research work has been recently devoted to the understanding and analyzing of the perceptual and semantic contents of video documents. These efforts materialize as the *semantic content-based video retrieval systems*. A concept of *semantic content-based retrieval* is commonly understood as a process of searching for the documents that satisfy the conditions expressed in the terms of semantic units such objects, associations, events, and descriptions included in the video documents. The implementations of semantic content-based video retrieval systems still need more research in the areas of formal description of semantic contents of video documents, data models and database systems capable of storing and maintaining the semantic contents, indexing of video documents, and intermediate level query languages.

This work considers an application of a semistructured data model as a database component of a semantic content-based video retrieval system. We also address the problems related to the conceptual modeling of semantic contents of video documents and intermediate level query language for the retrieval of indexed video documents. We propose the extensions to a traditional conceptual modeling notation used for the modeling of database domains and we present a semantics rich semistructured data model for an intermediate implementation level. A semantic content-based video retrieval system described in the paper is based on three level architecture where the first level is related to the semantic description and indexing of video documents, the second level virtually implements the first level as a semistructured database, and finally the third level implements a semistructured data model as an object-relational database system.

The paper is organized in the following way. The next section reviews the past works related to the implementation of semantic content-based video retrieval systems. Section III introduces a sample conceptual modeling notation for the video documents. A semantic rich semistructured data model is described in section IV. The mappings between the structures of conceptual video model and semistructured data model are presented in section V. Section VI introduces the operations of semistructured data model and shows how to express a sample video retrieval task the operations on semistructured data containers. Finally, section VII concludes the paper and presents the future research directions related the implementation of semantic content-based video retrieval systems.

II. RELATED WORKS

A video document contains either *perceptual* of *semantic* contents. *Perceptual* content includes all what can be percepted by the human senses, i.e. all what is seen and what is heard. *Semantic* content is the *meaning* of what has been delivered within the *perceptual* contents. Majority of the research works on the content-based retrieval systems target the analysis and

0-7695-2993-3/07 $25.00 © 2007 IEEE
DOI 10.1109/UBICOMM.2007.12

IEEE computer society

extraction of the perceptual contents. One of the current commercial perceptual content-based retrieval system widely used by the television networks is *Virage* [1]. *Virage* allows for the simultaneous automatic encoding and indexing in real time. The plug-ins to the system include face and on-screen text recognition and audio recognition. By listening to an audio track, the system identifies spoken words, speaker names, and audio types eliminating labor intensive manual annotation processes. *VIMSYS* [2] and *WebSeek* [3] are the typical systems that implement the retrievals of video clips based on colors, shapes, and sketches. The limitations of perceptual content-based retrieval are still immature processing techniques, inability to discover semantics especially when similar views have different semantics, resource hungry algorithms, and lack of support for the search techniques different from query-by-example type. The meanings like such as the generalized and specialized concepts, classifications, and subjective information like video title are beyond the perceptual level.

The *semantics based* video techniques are ignored by in many researches because of the manual annotations needed to describe the documents. Having a standard data model for video document, which is a trend consistent with the *MPEG-7* standard [4], could be an essential step towards automating video analysis and annotation process. The languages of *Resource Description Framework*(RDF) [5] and *Web Ontology Language* (OWL) [6] became the significant steps towards the conceptual modeling of semantic information. Video documents are the semistructured media. Some semantic-based retrieval systems have focused on the specific well-structured domains like for example, television news, [7] or sports [8]. The indexing of a video document requires its *logically segmentation*, i.e. the partitioning into the logically related sequences of frames with the annotations assigned to each partition. A concept of *stratification* has been proposed by [9] to identify the layers of information in the cinematic contents in a way different from the traditional segmentation. Stratification has been adopted in many further solutions and several video models based on stratification have been proposed so far. The rapid developments of object oriented technologies in early 1990s had a significant impact on these approaches. *object-Oriented Video Information* (OVID) [10] is an object-oriented data model for video retrieval. Video objects in OVID correspond to the sets of arbitrary portions of time sequential and contiguous video frames. *VideoStar* [11] is a generic data model for capturing video contents and structure based on stratification and built on enhanced Entity-Relationship (ER) model. *Video Object Description Model* (VODM) [12] takes a very similar approach of extending ER model for a database conceptual level organization. Entities in *VODM* are defined as the sequences of frames referred to video objects. Another basic element of video description is a relationship, which is an association between the objects. Both associations and objects are described by attributes. A system based on *Common Video Object Model* (CVOT) [13] is capable of automatic video segmentation and representation of temporal relationships among the video objects. *Video Information Retrieval On Notation* (VIRON) [14] is a video data model that shares and reuses annotations. The annotated objects are mapped into a unified video annotation system. The objects are used to refer to the video segments and textual annotations that are applied for the objects' descriptions. *VideoText* [15] is a simple semantic video model based on the logical video segments used for layering, video annotations, and associations between the segments. *VIdeo DAta Model* (VIDAM) [16] is a video data model that represents concepts as the semantic objects and spatiotemporal information as structural objects. Objects are defined as any description of catalogue, segment and what is seen and heard, however no formal definitions of semantic objects, relationships, or description schema are possible. *Computer Assisted Education and Training Initiative/Internet Multimedia Library* (CAETI IML) [17] supports subject-based retrieval and it is well suited for extracting visual content which can be matched with a query. *Table of Contents* (ToC) [18] concentrates on videos having story lines and structures the video streams into hierarchy of video, scene, group, shot, and key frame. *ToC* considers a scene as a semantic entity that conveys the semantic meaning of video to the viewers. The system proposes an approach to the group-based scene construction using the visual similarity and time locality. *Unified Video Retrieval System* (UVRS) [19] provides content-based, feature-based, and annotation-based queries of video data. The system is based on three layered Hybrid Object-Oriented Metadata Model which is composed of the raw-data layer for a physical video stream, the metadata layer to support the annotation-based retrieval, content-based retrieval, and similarity retrieval, and semantic layer to construct a query. [20] proposes a simple generic data model and rule based query language for content-based video access. The model allows for user-defined attributes as well as explicit relations between the objects. Objects can be linked together by the means of explicit relation names. A multilevel abstraction mechanism for capturing the spatial and temporal semantics associated with the various objects in video frames is proposed in [21]. At the finest level of granularity, video data can be indexed on mere appearances of objects and faces. At the higher levels of indexing, an object-oriented paradigm is proposed to support the domain-specific views.

III. CONCEPTUAL MODELING OF VIDEO DOCUMENTS

Conceptual modeling techniques are widely used for the formal specification of the modeled domains in the relational database design. Conceptual modeling is also a natural way to represent the semantic contents of video documents as the documents themselves are the views of real and virtual domains. From such a perspective, an enhanced conceptual modeling notation seems to be an appropriate tool for the representation of semantic structures, high-level semantic composition, and indexing of video documents.

A *user view* of a video document is his/her perception of the video contents. A user view is usually created through the identification in a video document of meaningful entities, also called as *semantic units*. Different users watching the same video clip are usually interested in a bit different semantic units and because of that the user views are different depending on the semantic contents within the scope of interests. In a consequence, does not exists a conceptual model of a video document that includes absolutely all user views and does not exist a conceptual modeling notation capable of modeling of all these views. Any conceptual modeling notation addresses only the selected aspects of user views because such a notation operates on the finite set of semantic units.

As our choice of the semantic units determines the expressiveness, completeness, and flexibility of a conceptual modeling notation then identification of the most representative semantic units is a significant issue. The present trends in the conceptual video modeling aim at the frame-based semantic units where the only type of semantic unit captured from a frame is an *object*. However, the object-based semantic models are too simple and cannot express all complex aspects of semantic video contents. Apart from the objects our set of semantic units includes *activities*, *associations*, *events* , and *composite semantic units* built from the elementary and other composite semantic units. The concepts of *descriptions* and *abstractions* apply to all categories of semantic units.

A *physical object* is an instance of a salient object captured in a video physical space and represented visually, aurally, or textually. An *object* is a physical object identified by a viewer.

To represent the interactions among the semantic units in a video document we use a concept of *association*. The semantic units are interrelated in their context, semantic structure, space, and time. The properties indicate four types of associations: *activity*, *contextual*, *structural*, *spatial*, *temporal* associations. An *activity association* is an interpretation of the continual changes in the values of object's observable attributes in a given interval time i.e. over a sequence of frames. A number of objects may be involved in an activity. The objects that play an active role in an activity are called as *actors*. A *contextual association* R is an n-ary relationship between n semantic units in a context and it is denoted by $r(a_1, \ldots, a_n)$. For example, *is-father-of(x,y)* is a contextual association between the objects x and y from a class *Person*. A *structural association* is a binary association between the instances of semantic units in a composition structure and it is denoted by $r(a, b)$ where r is an association name, e.g. *component-of*, or *part-of* and a, b are the semantic units. A *spatial association* is also a binary association between two semantic units indicating a relationship in space and denoted by $s(a, b)$ where s is an association name, e.g. *above*, *left*, *right*, etc and a, b are the semantic units. A *temporal association* is another binary association between two semantic units interpreted in time and denoted by $t(a, b)$ where t is an association name, e.g. *before*, *next*, *during*, etc and a, b are the semantic units.

A *configuration* is a set of semantic units that describes a static view of the reality represented in a video document. For example, a configuration describes a contents of a single video frame.

An *event* is defined as a sequence of configurations. The transitions between the configurations of an event are indicated by the changes of the values of observable attributes of semantic units involved in a configuration, disappearance or appearance of new semantic units. e.g. a modification of attribute position describing an object of a class *Car* or a new association between previously not related objects of a class *Person*. For example, consider a sequence of frames representing a *room leaving* event when a person walks towards a door, opens it, and leaves a room through a door. A set of semantic units involved in the event consists of two objects p and d of class *Person* and *Door*, a unary activity associations *Walk(p)*, and binary activity associations *Open(p,d)* and *Leave(p,d)*. The initial configuration is a set of units $\{$*p:Person*, *Walk(p)*$\}$. When a person arrives at a door the initial configuration changes into $\{$*p:Person*, *Walk(p)*, *d:Door*$\}$. Next a person stops and opens a door, $\{$*p:Person*, *d:Door*, *Open(p,d)* $\}$ and finally, a person leaves a room, $\{$*p:Person*, *Walk(p)*, *Leave(p,d)*$\}$.

A *composite semantic unit* is a structure built of the instances of elementary and other composite semantic units. For example, a group of objects of class *Man* bound by an activity relationship *Collaborate* represents a complex semantic unit of class *Team*.

A *description* of a semantic unit is a set of attributes representing the features of interest in a user view. The *descriptions* provide fact oriented information usually queried by the users and because of that are the very important features of the model. A description can be *perceptual* (media dependent) such as `color` or *semantic* (media independent) such as `name`. Semantic units may appear in a video a number of times leading to two categories of attributes: *static attributes* that have fixed values like for example `date of birth` and *dynamic attributes* that change their values over time, for example `(x,y) coordinates`.

Generally a process of conceptual modeling requires three different "ways of thinking" about a modeled domain. These "ways of thinking" are commonly called as *abstractions*. *Abstractions* are the mechanisms used for the identification of classes of semantic units, grouping the instances of semantic units, and building the hierarchies of classes of semantic units. We distinguish the following types of abstraction: *classification*, *aggregation* , and *generalization*. *Classification* abstraction is performed when defining the classes of semantic units, for instance, identification of a class of objects *Person*, an activity association *Open*, an event *Lecture*, and so on. *Aggregation* abstraction is performed when assembling the complex semantic units from the elementary or composite units with *Is component of* or *Consists of* association, for instance a class of objects *Car* is an aggregation of the classes like *Engine*, *Wheel*, etc. *Generalization* abstraction is performed when defining the hierarchies of the classes of semantic units, for example an activity association *Run* is a subclass of activity association *Move* and superclass of activity association *Sprint*.

A triple $<VID, t_s, t_e>$ is called as an *observation slot* where VID is video document identifier, t_s and t_s determine the beginning and the end of a contiguous sequence of frames.

When applying the classification abstraction we have to clearly distinguish between the *instances* of semantic units and *classes* of semantic units. In the future we will abbreviate a term ”instance of semantic unit” into a simple ”semantic unit”. An instance of semantic unit may appear any number of times in many video documents. Each time an instance of the same semantic unit appears in a video document it may be described by a different set and different values of the attributes. A set of attributes and the values of attributes describing an instance of semantic unit is called *state* of the unit. Formally a state of semantic unit is a set of pairs <*attribute-name*, *attribute-value*>. As the semantic units pass through the sequences of changing states a concept of *event* can be used to represent the histories of changing states. A *index* over a collection of video documents is a partial mapping $\mathcal{I}$ of a set of all states of semantic units $\mathcal{S}$ into a powerset of observations slots $\mathcal{O}$, i.e. $\mathcal{I} : \mathcal{S} \rightarrow \mathcal{P}(\mathcal{O})$. The mapping $\mathcal{I}$ is partial because we may not wish to use the states of all elementary semantic units as the entries in an index. Indeed, the complex semantic units like for instance the events can be mapped into the observation slots without mapping the instances of units forming the events. As an example, consider an instance of activity association *Open(p,d)* where *p* is certain state of instance of class *Person*, *d* is another state of instance of class *Door*. Then, a mapping $\mathcal{I}(Open(p,d))$ provides a set of observation slots $\{s_1, \ldots, s_n\} \in \mathcal{O}$ such that a person opens a door in all sequences of frames determined by $\{s_1, \ldots, s_n\}$.

Retrieving the video documents that satisfy certain properties is equivalent to filtering a database that contains the states of semantic units. A database is divided into four areas each one containing the states of all units of the same kind, e.g. *objects*, *associations*, *complex semantic units*, and *events*. A query in a high level query language is translated into an expression built from the elementary operations on the sets of states of semantic units. The evaluations of the expressions provide the sets of states of semantic units. Finally, an index is applied to find all observations slots for the semantics units found in a database. The video documents are grouped by the video document identifiers and accessed to get the sequences of frames indicated by the observation slots. A structure of a database implementing a conceptual model described above and implementation of the search processes are presented in the next section.

IV. SEMISTRUCTURED DATA MODEL

This section informally introduces the main concepts of *Object-Relationship-Attribute Semistructured* (ORA-SS) data model [22]. The model is defined around the concepts of *schema*, *schema instance*, and *set of schema instances*. A *schema* is a name followed by a sequence of attributes, schema definitions, and schema references. The *qualifications* follow the elements of schema definition and determine the cardinalities of attributes and associations among the instances of schemas. The model extends a family of typical semistructured data models, e.g. XML, to include more semantics into the storage organizations of semistructured data.

A small database that contains information about people opening the doors and entering the rooms in a building can be defined as ORA-SS schema in the following way. In order to simplify a notation the qualifications of attributes and associations are omitted.

```
PERSON(name,height,
      OPEN:DOOR(room#,
                LOCATED-IN:ref(BUILDING)));
BUILDING(area, building#);
```

An example above shows two ways of defining associations: either by ”embedding” it in a higher level definition like *activity* association `OPEN`, or by ”referencing” it in another schema like *spatial* association `LOCATED-IN`.

A *schema instance* or simply an *instance* is an elementary data component defined in the model. An *instance* is a tuple of pairs `<name,value>` where `name` is either a name of attribute, name of nested schema, or reference to an external schema. Then, `value` is a respective value of attribute, schema instance, or set of references to schema instances. The present version of the model adopts *reference by value* implementation of the references to schema instances. When a schema instance is created it obtains a unique pair `<ID,v(ID)>` where `ID` is a hidden attribute and `v(ID)` is a value of the attribute. A value `v(ID)` uniquely identifies the schema instances and it is preserved by all operations on the schema instances. Implementation of reference `ref(X)` is a set of values of attribute `ID` uniquely identifying the instances of schema `X`. A set of schema instances plays a role of a data container in ORA-SS database and it is used as an argument of all operations proposed below.

ORA-SS model is proposed for the implementation of a database that contains the conceptual descriptions of video documents because ORA-SS can be very easily implement within the latest SQL standard of object-relational data model. A number of commercial (Oracle [23], DB/2 [24]) and public domain (PostgreSQL [25]) implementations of object-relational data model are available. These systems provide the implementations of nested relational tables, tables with objects, references and collections of references to objects, which perfectly fit into the implementations of nested definitions of schemas and references to schemas of ORA-SS model. The next section shows a sample implementation of a conceptual model video database as ORA-SS database.

V. MAPPING THE CONCEPTUAL VIDEO MODEL INTO THE SEMISTRUCTURED DATA MODEL

Application of semistructured data model ORA-SS as a logical view of conceptual video model database needs the transformations of the abstract concepts of *semantic unit*, *class*, *association*, *complex object*, *configuration*, and *event* into the concepts of semistructured data model.

A *semantic unit* being either an *object* or *association*, or *complex object*, or *event* is represented by the following schema.

```
SEMANTIC-UNIT(IS-AN-OBJECT:ref(OBJECT)|
```

```
 IS-AN-ASSOCIATION:ref(ASSOCIATION)|
 IS-A-COMPLEX-OBJECT:ref(COMPLEX-OBJECT)|
 IS-AN-EVENT:ref(EVENT));
```

A vertical bar separating the definitions of associations denotes *exclusive-or* relationship between the associations, i.e. only one of the associations listed in a schema occurs in an instance of `SEMANTIC-UNIT`.

The descriptions of all objects are represented as the instances of schema `OBJECT`.

```
OBJECT(class-name,
 HAS-INSTANCE:ref(OBJECT-INSTANCE));
```

The descriptions of all object instances are represented as the instances of schema `OBJECT-INSTANCE` in ORA-SS model. The qualifications `[0..*][1..*]` attached to `ATTRIBUTE` schema mean that object instance has none or many attributes and that each pair `(name,value)` is related to exactly one object instance.

```
OBJECT-INSTANCE(HAS-ATTRIBUTES:
 ATTRIBUTE(name, value)[0..*][1..1]);
```

The ORA-SS representations of the concepts *association* and *complex object* are similar to the schemas given above. The events are represented as the instances of schema `EVENT`.

```
EVENT(event-name,
 HAS-INSTANCE:ref(EVENT-INSTANCE));
EVENT-INSTANCE(CONSISTS-OF:
 ref(CONFIGURATION)[1..*][1..1]);
CONFIGURATION(number,CONSISTS-OF:
 ref(SEMANTIC-UNIT-INSTANCE)[0..*][0..*]);
```

Finally, an instance of semantic unit is either an instance of object or instance of association or instance of complex object or instance of event.

```
SEMANTIC-UNIT-INSTANCE(
 INCLUDES-OBJECT-INSTANCE:
  ref(OBJECT-INSTANCE)|
 INCLUDES-ASSOCIATION-INSTANCE:
  ref(ASSOCIATION-INSTANCE)|
 INCLUDES-COMPLEX-OBJECT-INSTANCE:
  ref(COMPLEX-OBJECT-INSTANCE)|
 INCLUDES-EVENT-INSTANCE:
  ref(EVENT-INSTANCE);
```

VI. OPERATIONS OF ORA-SS ALGEBRA

The system of operations proposed for ORA-SS data model is powerful enough to be used as an implementation tool for a general purpose query and data manipulation language. The system consist of three groups of operations.

The first group includes the operations of *selection*, *projection*, *extension*, *forward navigation*, and *backward navigation*. These operations act on the sets of schema instances and return the sets of schema instances. *Selection* ($\sigma_\phi(r)$) returns all instances included in an argument r that satisfy a condition ϕ. *Projection* ($\pi_x(r)$)takes on input a set of schema instances r and returns a set of instances of a subschema included in a schema x of the argument. *Extension* ($r \lhd s$) operates on two sets of schema instances r and s such that schema of s is included in a schema of r. The operation "extends" all instances in a "smaller" set s with the larger schema instances in r. In such a sense an *extension* is considered as an operation, which is opposite to *projection*. *Forward navigation* ($r_{ref(y)}{\rightarrow}s$)uses the references to move from a set of schema instances r to another set of schema instances s referenced by $ref(y)$. Finally, *backward navigation* ($r_{ref(y)}{\leftarrow}s$) operation uses the reference to move from a set of schema instances s to another set of schema instance r in the direction opposite to the references in $ref(y)$.

The second group of operations changes the schemas of the arguments. This group includes the symmetric pairs of *unnest* and *nest*, *dereference* and *reference* operations. Additionally this group includes *reordering* operation. *Unnest* operation ($\nu_y(r)$)removes the given nesting level y from all schema instances in an argument r. The operation can be used to "flatten" the given structures of schema instances. *Nest* operation ($\mu_y(r)$)acts in the opposite way by grouping the instances accordingly to the values of selected attributes and adding a new nesting level y to all instances in an argument r. *Dereference* and *reference* operations change the implementations of associations in ORA-SS model. *Dereference* ($r \prec_{ref(y)} s$) operation embeds the referenced by $ref(y)$ instances in s into all instances in a given set r. Finally, *reference* ($r \succ_{ref(y)} (s)$) operation removes the embedded instances s from all instances in a given set r, stores the extracted instances in another set, and replaces the extracted instances with the references $ref(y)$ to instances in another set.

The last group includes the operations of *Cartesian product*, and set theoretical operations of *union* ($\cup$), *intersection* ($\cap$), and *difference*(-). These operations act in a usual way on the sets of schema instances. The applications of the operations described above to implementation of semantic video retrieval are given below.

Application of ORA-SS algebra to the retrieval of video documents is demonstrated with the following example. Assume that we would like to find all video clips related to an event called as "conference" and such that video clip starts from a view of building. To find the relevant sequences of frames we have to access the areas of a database that contains information about the events and objects included in the first configuration of each event and use an index that maps objects of class building into observation slots. Consequently, the query is expressed in the following way: find all buildings included in the configurations located at the beginning of events called as "conference". Let `e` be a set of instances of schema `EVENT` in ORA-SS database. Then, finding all instances of events called as "conference" and storing them in a set of instances `ei` can be expressed as:

$\texttt{e'} := \sigma_{\texttt{name='conference'}}(\texttt{e});$

$\texttt{ei} := \texttt{e'}_{\texttt{ref(EVENT-INSTANCE)}} \rightarrow \texttt{e};$

Next, we find all initial configurations from `c` that are included in the instances of events found.

$$\mathtt{c'} := \mathtt{ei}_{\mathtt{ref(CONFIGURATION)}} \rightarrow \mathtt{c};$$
$$\mathtt{c1} := \sigma_{\mathtt{number=1}}(\mathtt{c'});$$

In the final step, we find all buildings included in the first configuration of the events.

$$\mathtt{s'} := \mathtt{c1}_{\mathtt{ref(INSTANCE\text{-}OF\text{-}SUNIT)}} \rightarrow \mathtt{s};$$
$$\mathtt{b} := \sigma_{\mathtt{name='building'}}(\mathtt{s}_{\mathtt{ref(object)}} \rightarrow \mathtt{o});$$

Whenever on object-relational system implements ORA-SS data model then the operations of ORA-SS algebra are expressed as the statement of object-relational SQL.

VII. SUMMARY, CONCLUSIONS, AND FUTURE WORK

This work addresses a question of what view of data should be used for a database component in the semantic-content video retrieval systems. A solution proposed in this paper is based on three level organization of data in video search systems. At the top level, an enhanced conceptual modeling notation is used to describe the semantic contents of video documents. A semistructured data model ORA-SS is applied at the intermediate level and at the lowest level it is implemented as an object-relational database. We show that ORA-SS model is expressive enough to represent all semantic units of the enhanced conceptual modeling notation and we provide a sample transformation of the enhanced conceptual schemas into ORA-SS schemas. An informal specification of the operations of ORA-SS algebra is followed by an example that shows how to implement a sample query as an expression of ORA-SS algebra.

The advantages from using a semistructured data model at an intermediate implementation level include the expressiveness when applied to the implementation of enhanced conceptual schemas and simplicity of mapping into the structures of object-relational database. A high level of compatibility between ORA-SS model and object-relational model allows for the fast implementation of semantic-content video retrieval system on a top of one of existing commercial or public domain available object-relational database systems. Moreover, ORA-SS model allows for the representation of nested, hierarchical, and linked data structures which are typically needed for the implementation of complex semantic concepts.

A number of problems are left for the future research. One of the important research questions is how to speed up the manual indexing of video documents. When performed by a human it is always time consuming and it provides different results when implemented by the different indexers. As it is impossible to index the video documents completely automatically an idea a computer aided tool for the construction of the index may be a practical solution of the problem. An architecture of such system and collection of the software tools supporting a human during a process of indexing are the interesting directions for the future research. As different people produce different semantic descriptions of the same video document we need a technique for merging the semantic descriptions of the same document. More practical problems include the implementation of ORA-SS model on a top of existing object-relational database management system and invention of more advanced indexing schemas than mapping of semantic units into the sequences of frames.

REFERENCES

[1] “http://www.virage.com.”

[2] M. Yeung, B. Yeo, and B. Liu, “Extracting story units from long programs for video browsing and navigation,” in *Proceedings of the International Conference on Multimedia Computing and Systems*, 1996.

[3] J.Smith and S. Chang, “Searching for images and videos on the world wide web,” Columbia University, Tech. Rep. CRT 459-96-25, 1996.

[4] F. Nack and A. Lindsay, “Everything you wanted to know about MPEG-7 part 1 and 2,” *IEEE Multimedia*, vol. 6, 1999.

[5] “http://www.w3.org/rdf.”

[6] “http://www.w3.org/tr/owl-features.”

[7] H. Zhang, S. Tan, and S. Smoliar, “Automatic parsing of news video,” *Multimedia Systems*, vol. 1, 1995.

[8] G. Sudhir, J. Lee, and A. Jain, “Automatic classification of tennis video for high-level content-based retrieval,” in *IEEE International Workshop On Content-Based Access of image and Video Databases CAIVD’98*, 1998.

[9] G. Davenport, T. Smith, and N. Pincever, “Cinematic primitives for multimedia,” *IEEE Computer Graphics and Applications*, vol. 7, 1991.

[10] E. Oomoto and K. Tanaka, “OVID, design and implementation of a video-object database system,” *IEEE Transactions on Knowledge and Data Engineering*, 1993.

[11] R. Hjelsvold and R. Midtstraum, “Modeling and querying video data,” in *Proceedings of 20th VLDB Conference*, 1994.

[12] C. Change, K. Lin, and S. Lee, “The characteristics of digital video and considerations of designing video databases,” in *Proceedings of the 1995 International Conference on Information and Knowledge management*, 1995.

[13] J. Li, M. Özsu, and D. Szafron, “Video modeling and its integration in a temporal object model,” The University of Alberta, Tech. Rep. TR 96-02, 1996.

[14] K. Kim, K. B. Kim, and H. Kim, “VIRON, an annotation-based video information retrieval system,” in *Proceedings of the 20th International Conference on Computer Software and Applications COMPSAC’96*, 1996.

[15] H. Jiang, D. Montesi, and A. Elmagarmid, “VideoText database system,” in *Proceedings of the Fourth IEEE International Conference on Multimedia Computing and Systems*, 1997.

[16] U. Srinivasan and G. Riessen, “A video data model for content-based search,” in *Proceedings if the Eight International Workshop on Database and Expert Systems Applications*, 1997.

[17] H. Yu and W. Wolf, “A visual search system for video and image databases,” in *Proceedings if the Eight International Workshop on Database and Expert Systems Applications*, 1997.

[18] Y. Zhuang, Y. Rui, T. Huang, and S. Mehrotra, “Applying semantic associations to support content-based video retrieval,” in *International Workshop on Very Low Bitrate Video Coding, VLBV98*, 1998.

[19] M. Hee, Y. IK, and K. Kim, “Unified video retrieval system supporting similarity retrieval,” in *IEEE International Conference on Multimedia Computing and Systems*, 1999.

[20] C. Decleir, M. Hacid, and J. Kouloumdjian, “A database approach for modeling and querying video data,” in *The 15th International Conference on Data Engineering*, 1999.

[21] Y. Day, A. Khokhar, S. Dagtas, and A. Ghafoor, “A multi-level abstraction and modeling in video databases,” *Multimedia Systems*, vol. 7, 1999.

[22] G. D. Tok Wang Ling, Mong Li Lee, “An object-relationship-attribute model forsemistructured data, 2001,” National University of Singapore, Tech. Rep. TR 2001-02, 2001.

[23] “http://www.oracle.com.”

[24] “http://www.ibm.com/db2.”

[25] “http://www.postgresql.org.”