An MPEG-7 Scheme for Semantic Content Modelling and Filtering of Digital Video

MARIOS C. ANGELIDES[*] and HARRY AGIUS

Brunel University, School of Information Systems, Computing and Mathematics,

St John’s, Uxbridge, Middlesex UB8 3PH, UK

Tel: +44 1895 265990, +44 1895 265993

Fax: +44 1895 269726, +44 1895 251686

E-mail: angelidesm@acm.org, harryagius@acm.org

[*] Corresponding author.

Abstract:

Part 5 of the MPEG-7 standard specifies Multimedia Description Schemes (MDS); that is, the format multimedia content models should conform to in order to ensure interoperability across multiple platforms and applications. However, the standard does not specify how the content or the associated model may be filtered. This paper proposes an MPEG-7 scheme which can be deployed for digital video content modelling and filtering. The proposed scheme, COSMOS-7, produces rich and multi-faceted semantic content models and supports a content-based filtering approach that only analyses content relating directly to the preferred content requirements of the user. We present details of the scheme, front-end systems used for content modelling and filtering and experiences with a number of users.

Keywords:

multimedia, semantic content modelling, MPEG-7, content-based filtering, digital video

1. INTRODUCTION

The role of information filtering is to present only relevant content to the user and to eliminate irrelevant content from consideration. This is crucial with regards content management of large digital video resources since it is impractical for a user to browse through numerous and lengthy video segments, many of which they are likely to not have seen before, to track down content of interest (Day et al. 1999; Al-Safadi and Getta 2001; Löffler et al. 2002; Jaimes et al. 2004). For example, consider that the user wishes to locate segments within a digital video resource where a robbery was taking place and a white van was positioned in front of a bank; or consider that the user wishes to known which events within the video resource feature white vans. Manually searching through the video resource to obtain this information is a lengthy and time-consuming task that rises considerably as the size of the video resource increases. However, the use of filtering requires that the contents of the digital video resource have been modelled according to their semantics.

The extraction of features from video streams with which semantic content models are constructed is a lengthy, time-consuming process that therefore needs to be carried out prior to any use of the video streams (Bolle et al. 1998; Naphande and Huang 2001). Certainly, carrying out such processing 'on the fly', as and when the application demands it, is impractical, particularly for video streams. Therefore, semantic content models are used as 'surrogates' for the original video streams, which are created prior to usage and may be further populated and modified, as anticipated usage changes and as the video resource develops. The term 'surrogate' is used since all queries and responses that are required for the user and system to engage in interaction are undertaken on the semantic content model, not the video streams. Because of this, the semantic content model has to be tightly integrated with the video streams using referencing strategies. In this way, the filtering process may determine the meaning conveyed by the media within the archive so as to compare against the specified filtering criteria.

Since semantic content models serve as surrogates, the richer the representation within the model, the more useful it is to the application and thus to the user (Davis 1993; Tusch et al. 2000; Vendrig 2002; Zhao and Grosky 2002). Semantic content models require extensive and flexible vocabularies and structures to enable them to express the diverse, subjective, and multiple-perspective meanings apparent within video content. It has been advocated for some time now (Rowe et al. 1994; Hartley et al. 2000; Adami et al. 2001; Zhao and Grosky 2002) that data should be represented at multiple structural levels that map onto those structural content levels apparent within a video stream. Thus, the semantic content model must be able to express both the static content of frames and also the dynamic, temporal content of higher-level structures, such as shots and/or scenes (Golshani and Dimitrova 1998).

Part 5 of the MPEG-7 standard (ISO/IEC 2002) specifies an open format for semantic content models and thus proves appropriate in current application development (Tseng et al. 2004a) . However, while MPEG-7 specifies the format of the model, it does not prescribe how the model is to be used or, more importantly, filtered. To this end, this paper examines how an MPEG-7 implementation can be deployed for multimedia content modelling and filtering and proposes the COSMOS-7 scheme. The models produced with COSMOS-7 are rich in content detail and multi-faceted in content semantics and the content-based filtering technique developed with COSMOS-7 only analyses content that relates directly to preferred and explicitly-stated user content requirements. The paper is organised as follows. Section 2 reviews existing MPEG-7-oriented research within the literature. Section 3 discusses collectively the key semantic aspects of video content that the research community argues should be modelled by any content-modelling scheme and presents how COSMOS-7 models these semantic aspects in order to carry out content-based modelling. An end-user system, COSMOSIS, that can be used for COSMOS-7-based content modelling is also presented. Section 4 discusses the filtering clauses of COSMOS-7 and explains how filtering is undertaken within the Filtering Manager end-user system. Section 5 presents details of an evaluation undertaken with a number of users. Section 6 concludes the paper with a summary and closing remarks.

2. EXISTING MPEG-7 RESEARCH

With the emergence of MPEG-4 and, later, MPEG-7, the need for content-based approaches to multimedia was emphasised (Correia and Pereira 1998). Whereas MPEG-4 concerned itself with a content-based multimedia encoding format, MPEG-7 standardised the description of various types of multimedia content with a view to enabling fast and efficient searching, filtering and retrieval of content relevant to the user. As a result, many multimedia researchers have now started to incorporate MPEG-7 into their work, within a variety of areas and for various purposes.

Since MPEG-7 media descriptions are XML documents which conform to schema definitions expressed with the XML Schema variant MPEG-7 DDL, a number of researchers have suggested employing XML database solutions for their management. Kosch (2002) looked at XML database solutions and how MPEG-7 and multimedia database management systems could compliment each other. Westermann and Klas (2003) presented an extensive set of requirements considered important for an XML database solution that is to be employed for the management of MPEG-7 media descriptions. The requirements concern the representation of MPEG-7 media descriptions, the access to media descriptions, the ability to process media description schemes, and extensibility, as well as classic database management functionality such as transactions and concurrency control. They then evaluated a number of XML database solutions against these requirements. Their evaluation found significant deficiencies in the examined solutions which seriously affect their eligibility for the management of MPEG-7 media descriptions. They therefore proposed the need for a new generation of XML database solutions. Kang et al. (2003) proposed an XQuery Engine that can be used as an XQuery processing module in a digital library system that uses MPEG-7 descriptions. The XQuery Engine parses an input XQuery and constructs a syntax tree for the query. Then, it transforms the syntax tree into a query plan, called a Primitive Operation Tree (POT). Each node of a POT represents an atomic operation in terms of the information retrieval engine and can be interpreted and processed by the information retrieval engine. The result set is given back to the XQuery engine, which in turn transforms the result into an XML document of the form being required by the user interface. The final result in XML is returned back to the user interface. Since the current version of the XQuery specification does not define full functions for information retrieval, they have extended the XQuery syntax by adding some functions required for multimedia retrieval, such as *rankby()*. The iFinder system (Löffler et al. 2002) allows for search and retrieval of short video segments across multiple archives using a client/server architecture. The metadata is stored in an XML database and user queries are used to construct an XPath query string on the server which is then executed against the XML database. The system has been used to index speeches from the German Parliament, so that the user can search for fragments of political speeches and receives matching video segments through a video streaming service. Some researchers have advocated a relational database approach. Chu et al. (2004) propose an SM3 storage model for XML data within a relational database that overcomes the problems of the existing structure-mapping and model-mapping storage approaches. SM3 uses the model-mapping approach to store all internal XML nodes and the structure-mapping approach to store all the XML leaf nodes. Döller et al. (2002) proposed a Multimedia Data Cartridge (MDC) that implements an object-relational data model for the core part of the MPEG-7 standard. The MDC consists of two parts. The first part is a multimedia data model, which contains the MPEG-7 descriptions. The MPEG-7 schema is mapped, via Oracle's extensible type system, to a database schema of object types and tables. The second part is a multimedia indexing framework, which serves as an extensible environment for multimedia retrieval and consists of a GistService, a GistWrapper and a multimedia index type, the latter of which is an extension of the built-in indexing mechanisms of Oracle and employs high-dimensional feature vector indexing and enhanced access functionality such as k-NN search.

Other research has worked on how the MPEG-7 content model can be created. Some researchers have examined integrating image analysis techniques so that MPEG-7 descriptions can be generated automatically. For example, the Video Automatic Labelling (VideoAL) system (Lin et al. 2003b; Lin et al. 2003a) extracts semantic concepts from MPEG-1 video sequences using a set of anchor concept detectors and automatically generates MPEG-7 metadata files. The anchors are trained by users associating labels with training videos using the VideoAnnEx MPEG-7 annotation tool (Tseng et al. 2002; IBM 2005). Similarly, the MPEG-7 Metadata Authoring Tool (Ryu et al. 2002) provides a graphical environment supported by video content analysis that uses shot boundary detection and key frame extraction to input starting frames and duration indexes for video segments into an MPEG-7 description generation module. The MPEG-7 descriptions are then validated and presented to the user through an MPEG-7 metadata visualiser and editor. Descriptions can be directly edited via a tree-view editor and stored in text or binary (BiM) formats. The Video Image Retrieval System (VIRS) (Lee et al. 2003) employs ten MPEG-7 visual descriptors, including those for colour, texture, motion and shape, to deal with datasets of colour images, video clips, and object images. The system supports QBE (Query By Example) and QBD (Query By Draw). AMOS (Benitez and Chang 2003) is an MPEG-7-based video object segmentation and retrieval system, where a video object is modelled and tracked as a set of regions with corresponding visual features and spatiotemporal relation. The region-based model also provides an effective base for similarity retrieval of video objects. Manual annotation approaches have also been supported through the development of MPEG-7 authoring tools. For example, Mdefi (Tran-Thuong and Roisin 2003) uses a WYSIWYG approach for editing and presenting multimedia documents via an environment consisting of timeline, execution and hierarchical views. The first two views enable editing of temporal and spatial structures, while the latter view provides a tree-based view of the entire structure. Goularte et al. (2004) provide a method for creating annotations via pen-based interaction with Tablet PCs, which enables video streams to be annotated at capture time within a ubiquitous computing paradigm. The assumption is that drawing over images extracted from the media stream can be natural for the users and less restrictive, thus lending the annotations greater semantic meaning. The method is implemented within M4Note (MultiMedia MultiModal Annotation Tool), which uses an MPEG-7-compliant MediaObject model so that these annotations are recorded as multimedia objects. Annotations may be played back in synchrony with the video at a later date. In the VAnnotator system (Costa et al. 2002), annotations may be created for different views of the same video content. These views, termed ‘video-lenses’, provide interpretations of the multimedia content, giving different perspectives of the information, according to the specific requirements of each type of user. Thus, the same piece of information can be visualized and modified in different ways according to the video-lens being used. Annotations are made using a timeline-based interface with multiple tracks, where each track corresponds to a given video-lens. All information is stored in MPEG-7 format. Conversely, Eidenberger (2003) has undertaken an extensive evaluation of various distance measures used in MPEG-7 retrieval systems and proven that, while the distance measures recommended in the MPEG-7-standard perform extremely well, general-purpose measures can be even more effective.

Other research has looked at how to create MPEG-7 content models for different types of content and applications. Goularte et al. (2003) consider Interactive-TV documents and propose a high-level wrapper with contextual descriptions support to structure and organise MPEG-7 descriptions. The wrapper consists of a context namespace, a MediaObject schema, and an <XLinkObject> element to describe the link between objects, establish relationships between the objects, and to establish links between media objects and programmes. Regular TV broadcasting has been investigated by Pfeiffer and Srinivasan (2000), who explored the use of MPEG-7 within the TV Anytime standard. The TV Anytime standard implements a subset of MPEG-7 specifically for television broadcasting. Salembier et al. (2000) have proposed a set of MPEG-7 Description Schemes dealing with video programmes, users and devices. The Program Description Scheme is used to describe the physical structure as well as the semantic content of a video programme (visual information only). The physical structure is described via the temporal organisation of video segments, the spatial organisation of image regions, and the spatiotemporal structure of those regions in motion. The semantic content description is based around objects and events. Personal user preferences and prior viewing and listening habits are also described to enable personalised video programme delivery. Finally, a Device Description Scheme records the users of the device, available programmes, and device capabilities so that a device may prepare itself accordingly. Rehm (2000) uses MPEG-7 Description Schemes to model lnternet streaming media. A Streaming AV Description Scheme is proposed as a top-level entity to describe a piece of Internet-accessible streaming multimedia content. The Video-over-IP (VIP) system (van Setten and Oltmans 2001) encompasses various processes for digital video databases, including distributed content production for acquiring MPEG-7 metadata, distributed search engines, tools to browse and analyse videos, and playlist functionality. The system has been used in a pilot project in the educational domain. Vakali et al. (2004) propose a multi-level video representation and description scheme to cater for the description of multiple video content collections. Their model consists of the following layers: a cluster layer to provide a first partitioning of the collection according to similar images and/or video objects which may belong to different multimedia documents, a subcluster layer which contains the semantic units within the same cluster, a scene layer to support high-dimensional video indexing, and video shot and video object layers consisting of video frames and Video Object Planes respectively. More specific research has addressed the creation of ontologies. For example, the Dozen Dimensional Digital Content (DDDC) (Kuo et al. 2004) is an MPEG-7 based multimedia content description architecture aimed at personal digital photo libraries, which, it is argued, have specific characteristics compared to general-purpose image libraries. Consequently, the DDDC annotates multimedia data with 12 main attributes regarding its semantic representation: who, what, when, where (longitude, latitude, altitude), why and how the digital content was produced, together with the respective direction (two attributes), distance and duration information. A machine-understandable spatial and temporal based ontology representation for the DDDC enables a semi-automatic annotation process. On a grander scale, Hunter (2003) has proposed the ABC model as a core ontology with the potential to facilitate semantic interoperability between MPEG-7 and MPEG-21 vocabularies, as well as other metadata schemas.

Another research stream has dealt with the use of MPEG-7 for knowledge-based reasoning. For example, IMKA (Benitez and Chang 2003) is an intelligent multimedia knowledge application that uses the MediaNet knowledge representation framework, where multimedia information is used for exemplifying semantic and perceptual information about the world. Knowledge bases are built from collections of annotated images. Graves and Lalmas (2002) proposed a model for video retrieval based upon the inference network information retrieval model. The document network is constructed with MPEG-7 and captures information pertaining to structural (video breakdown into shots and scenes), conceptual (video, scene and shot content) and contextual aspects (context information about the position of conceptual content within the document). The retrieval process exploits the distribution of evidence among the shots to perform ranking of different levels of granularity and addresses the idea that evidence may be inherited during evaluation. It also exploits contextual information to perform constrained queries.

A large body of recent work has investigated using MPEG-7 for customisation, personalisation and adaptation. For example, Magalhães and Pereira (2004) propose a universal multimedia access oriented customisation architecture, discuss multimedia customisation processing algorithms and systems, and present a number of customisation experiments. Similarly, Martínez et al. (2002) describe a system that uses MPEG-7 to catalogue and provide access to multimedia content through annotated variations, in order to permit media content to be offered to different terminals and through different access networks. Ferman et al. (2002; 2003) propose a profiling agent for automatically determining user profiles from content usage history and a filtering agent for automatically filtering content based on the profiles. Fuzzy inferencing is used to construct and update preferences based on content interactions over a period of time. The agents support MPEG-7 or TV-Anytime-compliant descriptions. In the IndexTV system (Rovira et al. 2004), MPEG-7-based personalisation is applied to TV programmes. The system framework consists of FlowServer, a DVB play-out System, a Cataloguing Tool, to generate MPEG-7 descriptions of content in a semi-automated manner, and a Metadata Manager, to encapsulate, signal and broadcast the descriptions synchronously with content via the FlowServer. A TV Programme Assistant at the client end then uses the received descriptions to recommend content to the user. Other work has looked at personalisation and adaptation in the context of video summarisation. For example, Tseng et al. (2004a; 2004b) construct shortened video summaries that maintain semantic content within desired time constraints. To achieve this, the MPEG-7 VideoAnnEx and VideoAL tools (described previously) are integrated with MPEG-21 tools at the server end, with personalisation and adaptation engines sitting within media middleware. The VideoSue (Video Summarisation on Usage Environment) personalisation engine matches user queries and usage environments with media descriptions and rights expressions to generate personalised content. MPEG-7 descriptions along with MPEG-7 and MPEG-21 user preference declarations and user time constraints are used to output an optimised set of selected video segments that will generate the desired personalised video summary. Two adaptation engines, VideoEd and Universal Tuner, determine the optimal variation of the content for the user (in terms of format, size, and quality), according to adaptability declarations and the inherent usage environment. With VideoEd, video-audio information is directly extracted from MPEG sources and combined dynamically to generate a single MPEG file. The Universal Tuner enables the composite MPEG file to be transcoded for various devices. Similarly, Echigo et al. (2001) and Jaimes et al. (2002) propose a framework to generate video summaries from MPEG-7 descriptions personalised by user profiles. High-level semantic features (e.g. no. of offensive events) are extracted from existing metadata using feature time windows and the video summaries generated using a supervised learning algorithm which takes as input examples of important/unimportant events. They illustrate their approach by applying it to soccer footage. Fonseca and Pereira (2004) propose a mechanism for creating query-based video summaries using a relevance metric and a constraints schema based on MPEG-7 descriptions. A human skin filter, that employs the MPEG-7 Dominant Colour descriptor, allows summaries to be built based on the presence or absence of human skin.

3. SEMANTIC CONTENT-BASED MODELLING WITH COSMOS-7

Semantic content modellers argue that a number of key semantic content aspects should be represented in a semantic content model in order for that model to serve as a suitable surrogate for the original video streams. Previously (Agius and Angelides 1999, 2000, 2001) we have identified these key aspects as follows:

- *Events* within the semantic content model represent the context for objects that are present within the video stream at various structural levels. Events are therefore occurrences within the video stream that divide it into shorter content segments involving specific objects, and thus can frequently serve to represent object behaviour. As well as the participant objects, events will also typically be defined according to attributes such as the event name, its location, the time of occurrence, and so forth (Appan and Sundaram 2004). In this way, the user and the system are able to filter with regards to "What is happening here?" on both a general level (without reference to specific objects, e.g. a wedding is taking place) and a specific level (with reference to specific objects, e.g. Jim is marrying Michelle).
- *Temporal relationships between events* enable the semantic content model to express the dynamism in content that is apparent at these higher levels, thereby enabling filtering of non-static semantic content which is more in line with "What will or did happen here?" Again, this may occur on both a general and a specific level.
- The expression of *objects and object properties* within the semantic content model provides for filtering with regards to objects that are readily identifiable within the video content stream. Objects and object properties regularly form the basis of various multimedia-specific ontologies (e.g. Dasiopoulou et al. 2004). The term 'object' refers to any visible or hidden object depicted within a video frame at any necessary level of detail, from entire objects and groups of objects to the bottom-most component objects. Hidden objects are those which are not visible within a frame or segment but which are nevertheless present (Adali et al. 1996), for example, the contents of a closed box. Some research distinguishes between different classes of objects. For

example, Davis et al. (2004) distinguish between ‘persons’, ‘objects’ and ‘locations’. Because properties are also represented, other features of the objects may be used as filtering criteria, whether these are visible or not. Objects may themselves exist within a structural hierarchy thereby enabling inheritance of properties from higher level objects.

- Representations of *spatial relationships between objects* within the semantic content model enable filtering concerning the relative location of objects (rather than the absolute location that comes from a co-ordinate based representation). This enables reference to be made to the relationships between objects within the content and can provide for three-dimensional spatial representations, including those concerning hidden objects, which are difficult to derive from co-ordinate representations. Spatial relationships have a duration due to their occurrence across multiple frames. Because of object motion, spatial relationships between two objects may differ over time within the same segment.

The notion of events and objects being fundamental, interrelated building blocks for multimedia representation is one which dates back to early multimedia presentation standards such as HyTime and MHEG-1 and has been carried forward in later proposals for multimedia content modelling structures. While HyTime and MHEG-1 are for the specification of dynamic multimedia presentation content rather than the description of existing digital video content, it is useful to draw comparisons. In the event-oriented approach of HyTime (Goldfarb 1991; Newcomb et al. 1991; Newcomb 1995), events define the position and extent of occurrences of objects, such as digital video and/or audio, which occur within a finite coordinate space (FCS). An event is thus a nominal conceptual bounding box for an object. Spatial and temporal relationships are specified within event schedules that specifying the ordering of events, and thus of objects. Whether spatial or temporal relationships are specified within the event schedules depends on the chosen measurement domain for each axis of the FCS. In the object-oriented approach of MHEG-1 (Kretz and Colaïtis 1992; Meyer-Boudnik and Effelsberg 1995), content objects are those which are presented to the user. Events originate from objects. Spatial and temporal relationships can be represented through action objects and virtual views, where link objects thereby correspond to instances of events. While a simplification of the original standard, MHEG-5 (Vieira and Santos 1997) still retains this approach and indeed this structure has been used by some (e.g. Echiffre et al. 1998) as a basis for extension to enable descriptive content-based modelling.

The above semantic content aspects can be seen to have generic applicability since virtually all domains require some representation of objects and/or events, including relationships between them. For instance, entertainment-on-demand, multimedia news, and organisational content. Hence, when these semantic aspects are accommodated by a content modelling scheme the resultant model can be used to model semantic content for most mainstream domains and user groups and, consequently, facilitate filtering for those domains and user groups. COSMOS-7 therefore supports the modelling of these semantic aspects within a schema that is compliant with Part 5 of the MPEG-7 standard (ISO/IEC 2002). This consequently permits interoperability of COSMOS-7 semantic content models across a multitude of platforms and applications. Part 5 of the MPEG-7 standard contains the Multimedia Description Scheme (MDS) description tools, which include a range of *top-level types* whose purpose is to encapsulate complete descriptions of multimedia content and metadata. Thus, each top-level type contains description tools relevant for a particular media type, e.g. image or video, or additional metadata, e.g. describing usage or the scheme itself. The former top-level types are descended from the top-level abstract type *ContentDescriptionType* whereas the latter are descended from the *ContentManagementType* abstract type. Both are further descended from a *CompleteDescriptionType* which can be used to encapsulate a complete scheme. COSMOS-7 uses two of the MPEG-7 top-level types both of which are descended from the *ContentAbstractionType* abstract type, which itself is descended from the *ContentDescriptionType* abstract type:

- The *SemanticDescriptionType* is used to group the content modelled information regarding objects and object properties (including object inter-relationships), events, and temporal relationships between events.
- The *ContentEntityType* is used to group the content modelling information regarding spatial relationships between objects and provides identifiers for media files and segmentation.

Fig. 1 shows the key MPEG-7 description tools that are used within COSMOS-7 for each semantic aspect and illustrates how they are interrelated. A description tool may be either a *Description Scheme (DS)*, a *Descriptor (D)* or a *datatype*. DSs describe entities or relationships pertaining to multimedia content and specify the structure and semantics of their components, which may be other DSs, Ds, or datatypes. *Classification Schemes (CSs)* are one type of DS that define a set of standard terms describing some domain. Ds describes a feature, attribute, or group of attributes of multimedia content. Datatypes are basic reusable types employed by DSs and Ds. While MPEG-7 provides a plethora of description tools, their implementation in any modelling and filtering scheme should be dependent upon requirements. Hence, COSMOS-7 implements only those tools from MPEG-7 Part 5 which are necessary for modelling the semantic aspects discussed in the previous section, and enabling their filtering thereafter. Some description tools lent themselves naturally to a representation of the above semantic aspects, such as the *Event DS* to represent events and the *Object DS* to represent objects. However, the description of other semantic aspects required use of some generic description tools. For example, the *TemporalRelation CS* and *SpatialRelation CS* only provide the terms necessary for describing a particular temporal or spatial relationships. Therefore, the *Graph DS* was utilised to enable

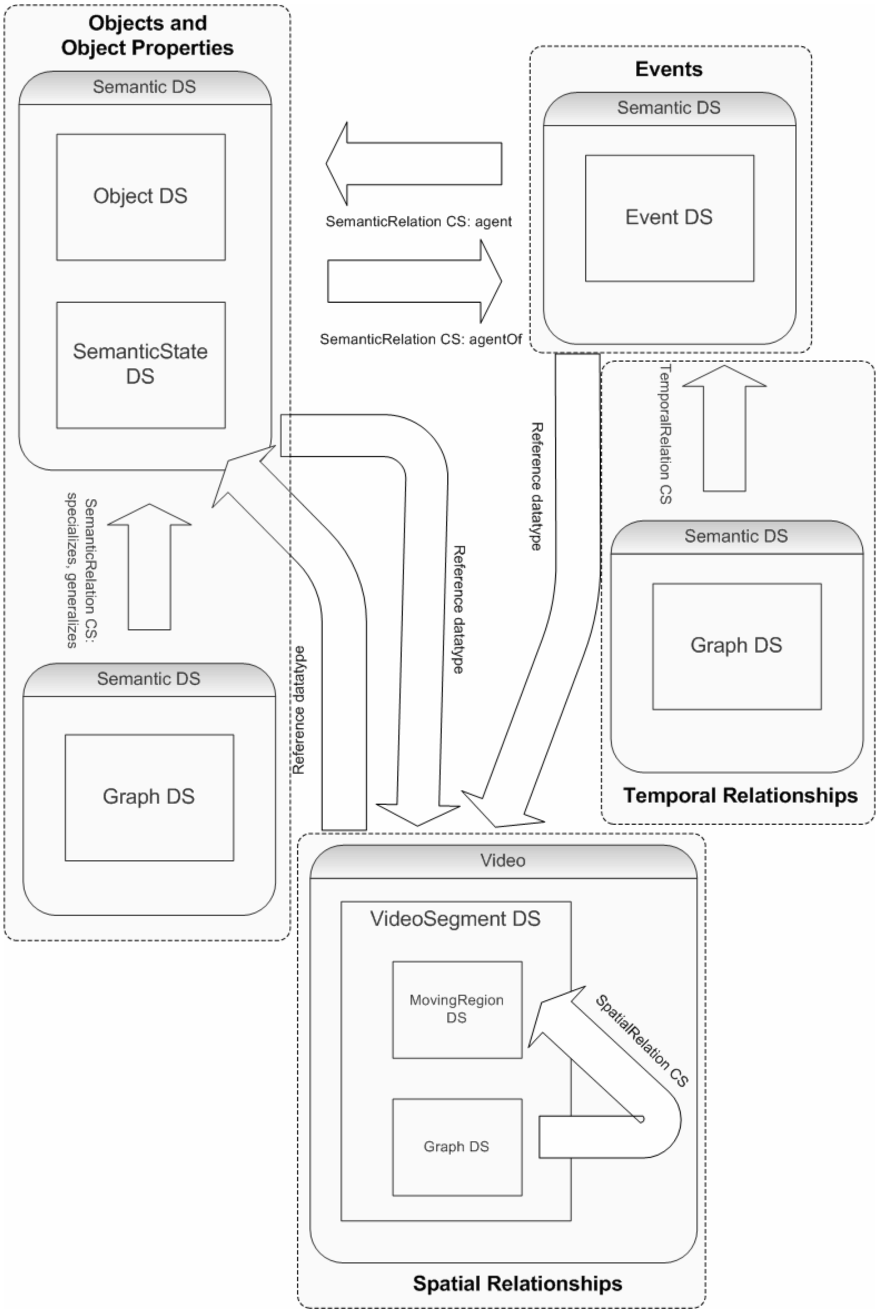


Fig. 1: Key MPEG-7 description tools used in COSMOS-7

these CSs to bind objects and events to temporal relationships via the *TemporalRelation* and *SpatialRelation CSs*. Similarly, while the *Object DS* enables instances of objects to be represented, another representation was required to model the object properties. The *SemanticState DS* is a generic DS that can be used for any type of entity to represent properties of that entity and thus was identified as suitable for supporting the modelling of object properties. Since, as stated previously, the semantic content aspects have generic applicability and the arrangement of MPEG-7 description tools within COSMOS-7 does not prescribe, and thus does not restrict, which characteristics of objects, events or relationships are to be modelled, COSMOS-7 may be seen to be useful to a wide range of domains and applications. Nevertheless, since COSMOS-7 is, like MPEG-7, an open modelling scheme, further MPEG-7 description tools may be implemented if and when necessary without the need to re-implement the entire scheme from scratch. The following sections discuss the use of MPEG-7 tools in COSMOS-7 in more detail.

3.1. Events

Events are modelled in COSMOS-7 using the *Event DS* but one or more events are grouped using the *Semantic DS* and given a suitable *id* and *Label*. This enables related events, such as those related to specific objects, to be proximate and utilised efficiently. Each event uses several elements within the *Event DS* as follows. Each event is given a *Label* that describes the event. The *MediaOccurrence* element uses the *MediaInformationRef* element to refer to the appropriate video segment and *TemporalMaskType*s within the *MediaInformationRef* element to define appropriate masks on the referred to video segment. Events are related to objects through the use of the *SemanticRelation CS* and the *agent* relation. The *agent* relation is defined as follows (ISO/IEC 2002): *A agent B* if and only if B is an agent of or performs or initiates A. The inverse, *agentOf*, is used when modelling objects (see later).

An example of COSMOS-7 events is given below. It depicts a group of Squirrel events that consist of an Eating event and a Sleeping event. Both events are depicted by the WholeSquirrel-V-VS video segment within the subintervals given. The events are related to the Squirrel-O object through the *Relation* elements.

```
<Semantics id="Squirrel-Events">
      <Label>
            <Name>Squirrel events</Name>
      </Label>
      <SemanticBase xsi:type="EventType" id="Squirrel-Eating-EV">
            <Label>
                  <Name>Eating</Name>
            </Label>
            <MediaOccurrence>
                  <MediaInformationRef idref="WholeSquirrel-V-VS" />
                  <Mask xsi:type="TemporalMaskType">
                        <SubInterval>
                              <MediaTimePoint>T00:00:00</MediaTimePoint>
                              <MediaDuration>PT4M</MediaDuration>
                        </SubInterval>
                        <SubInterval>
                              <MediaTimePoint>T00:06:05</MediaTimePoint>
                              <MediaDuration>PT3M</MediaDuration>
                        </SubInterval>
                  </Mask>
            </MediaOccurrence>
            <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:agent"
   target="#Squirrel-O"/>
      </SemanticBase>
      <SemanticBase xsi:type="EventType" id="Squirrel-Sleeping-EV">
            <Label>
                  <Name>Sleeping</Name>
            </Label>
            <MediaOccurrence>
                  <MediaInformationRef idref="WholeSquirrel-V-VS" />
                  <Mask xsi:type="TemporalMaskType">
                        <SubInterval>
                              <MediaTimePoint>T00:12:00</MediaTimePoint>
                              <MediaDuration>PT2M</MediaDuration>
                        <SubInterval>
                              <MediaTimePoint>T01:00:00</MediaTimePoint>
                              <MediaDuration>PT2M</MediaDuration>
                        </SubInterval>
                        </SubInterval>
                  </Mask>
            </MediaOccurrence>
            <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:agent"
   target="#Squirrel-O"/>
      </SemanticBase>
</Semantics>
```

3.2. Temporal relationships between events

Temporal relationships are grouped using the *Semantic DS* where “Temporal-Relationships” is the *id* and “Temporal Relationships” is the *Label*. The relationships themselves are represented as a graph via the *Graph DS*, with each temporal relationship defined using the *TemporalRelationCS*. The use of a graph in this way enables straightforward traversal of the relationships. COSMOS-7 supports the 14 binary and n-ary temporal relations specified in the MDS. These are given in Table 1, which also depicts the sub-set that map onto those originally specified by Allen (1983). The use of these relationships enables events to be logically ordered according to the requirements of a particular domain, and independently of their physical order within the video resource. This is particularly necessary where a single event can be apparent within multiple, disparate segments. Consequently, users may filter according to temporal locality in relation to known events, e.g. “Show me everything that happened before this event occurred”.

Table 1: Temporal relationships used in COSMOS-7

MPEG-7 relation	*MPEG-7 inverse relation*	*Allen’s relation*	*Conceptual example*
Binary			
Precedes	*follows*	Before	AAA BBB
CoOccurs	*coOccurs*	Equal	AAA BBB
Meets	*metBy*	Meets	AAABBB
Overlaps	*overlappedBy*	Overlaps	AAAA BBBBBB
StrictDuring	*strictContains*	During	AAA BBBBBB
Starts	*startedBy*	Starts	AAA BBBBBB
Finishes	*finishedBy*	Finishes	AAA BBBBBB
Contains	*during*	-	*Any of the above 3*
N-ary			
Contiguous	-	-	AAABBBCCC
Sequential	-	-	AAA BBBCCC
CoBeing	-	-	AAA BBBBBBB CC
CoEnd	-	-	AAA BBBBBBB CC
Parallel	-	-	AAAAAA BBBBB CCCCCCC
Overlapping	-	-	AAAAA BBBBBBB CCCCCCC

An example of COSMOS-7 temporal relationships is given below. It depicts that the squirrel Eating event occurs before the squirrel Sleeping event. Note that there is no need to relate these events to the Squirrel-O object since this has already been done in the respective events.

```
<Semantics id="Temporal-Relationships">
      <Label>
            <Name>Temporal Relationships</Name>
      </Label>
      <Graph>
            <Relation type="urn:mpeg:mpeg7:cs:TemporalRelationCS:2001:precedes"
                     source="#Squirrel-Eating-EV”
                     target="#Squirrel-Sleeping-EV"/>
      </Graph>
</Semantics>
```

3.3. Objects and object properties

As is the case with events and temporal relationships between events, each object and its properties are grouped together using the *Semantic DS* with an appropriate *id* and *Label*. There are three parts to the COSMOS-7 representation of objects and object properties within the *Semantic DS*:

- Objects are related to events through the use of the *SemanticRelation CS* and the *agentOf* relation. The *agentOf* relation is defined as follows (ISO/IEC 2002): *B agentOf A* if and only if B is an agent of or performs or initiates A.
- The *Object DS* is used to content model the objects. It includes elements describing the composition of an object from sub-objects. Thus, each sub-object is incorporated into this representation within COSMOS-7. The *MediaOccurrence* elements (together with temporal masks) are used as previously described to relate specific video segments that reflect the occurrences of objects.
- The *SemanticState DS* is used to content model object properties. To enable *MediaOccurrences* to be related to specific object properties, each property is modelled as a separate

SemanticState. The *AttributeValuePair* element is used to specify the properties themselves. The scheme makes no restrictions on which properties may be modelled, so long as they conform to the given structure. It can therefore be tailored for particular domains and user groups.

Objects are related to each other through the use of the *SemanticRelation CS*. The *Semantic DS* is used to group all object relationships together with a graph representing the relationships through the *specializes* relation (the inverse relation is *generalizes*).

An example of objects and object properties in COSMOS-7 is provided below. The example shows a squirrel object that consists of several sub-objects, one of which is its tail. The height property is shown for the squirrel and specified with a value of 16cm. It is related to two media segments that illustrate the squirrel’s height within the WholeSquirrel-V-VS segment. The object hierarchy is shown at the bottom of the example and is given the specific *id* “Object-Hierarchy” and the specific *Label* “Object Hierarchy”. The relationship shows that the Squirrel-O object is a specialisation of the Rodent-O object, thus reflecting the fact that squirrels are rodents.

```
<Semantics id="Squirrel-SEM">
      <Label>
            <Name>Description of squirrel</Name>
      </Label>
      <SemanticBase xsi:type="ObjectType" id="Squirrel-O">
            <Label>
                  <Name>Squirrel</Name>
            </Label>
            <MediaOccurrence>
                  <MediaInformationRef idref="WholeSquirrel-V-VS" />
                  <Mask xsi:type="TemporalMaskType">
                        <SubInterval>
                              <MediaTimePoint>T00:07:00</MediaTimePoint>
                              <MediaDuration>PT3M</MediaDuration>
                        </SubInterval>
                        <SubInterval>
                              <MediaTimePoint>T00:15:00</MediaTimePoint>
                              <MediaDuration>PT9M</MediaDuration>
                        </SubInterval>
                  </Mask>
            </MediaOccurrence>
            <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:agentOf"
                     target="#Squirrel-Eating-EV" />
            <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:agentOf"
                     target="#Squirrel-Sleeping-EV" />
            <Object id="Squirrel-O-tail">
                  <Label>
                        <Name>Tail</Name>
                  </Label>
                  <MediaOccurrence>
                  <MediaInformationRef idref="WholeSquirrel-V-VS" />
                  <Mask xsi:type="TemporalMaskType">
                        <SubInterval>
                              <MediaTimePoint>T00:03:30</MediaTimePoint>
                              <MediaDuration>PT2M</MediaDuration>
                        </SubInterval>
                  </Mask>
                  </MediaOccurrence>
            </Object>
      </SemanticBase>
      <SemanticBase xsi:type="SemanticStateType" id="Squirrel-O-Props-Height">
            <Label>
                  <Name>Height</Name>
            </Label>
            <MediaOccurrence>
                  <MediaInformationRef idref="WholeSquirrel-V-VS" />
                  <Mask xsi:type="TemporalMaskType">
                        <SubInterval>
                              <MediaTimePoint>T00:00:00</MediaTimePoint>
                              <MediaDuration>PT6M</MediaDuration>
                        </SubInterval>
                  </Mask>
            </MediaOccurrence>
            <AttributeValuePair>
                  <Attribute>
                        <Name>Height</Name>
                  </Attribute>
                  <Unit>
                        <Name>cm</Name>
                  </Unit>
                  <IntegerValue>16</IntegerValue>
            </AttributeValuePair>
      </SemanticBase>
</Semantics>

<Semantics id="Object-Hierarchy">
      <Label>
            <Name>Object Hierarchy</Name>
      </Label>
      <Graph>
            <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:specializes"
                     source="#Squirrel-O" target="#Rodent-O"/>
      </Graph>
</Semantics>
```

3.4. Spatial relationships between objects

Spatial relationships are specified on a per video stream basis. The video is segmented so that spatial-relationship inheritance may be deployed, whereby child segments inherit a parent segment's spatial relationships as well as specifying their own spatial relationships. COSMOS-7 therefore uses the *VideoSegment DS* and creates segments using the *VideoSegmentTemporalDecompositionType* and the *VideoSegmentSpatioTemporalDecompositionType*. *MediaTime* elements are used to delineate the segments. The *MovingRegion DS* is used to identify objects as regions within the video which are related to the content-modelled objects specified in the previous section through the use of the *SemanticRef* element (a *Reference* data type) that the *VideoSegment DS* inherits from the *Segment DS* in the MDS. The spatial relationships themselves are specified using the *SpatialRelationCS* within a graph. Table 2 lists the spatial relations that are specified in MPEG-7 and which are employed within COSMOS-7. As can be seen, some redundancy is provided through alternatives to allow for flexibility in representation, which may be needed for differing application requirements.

Table 2: Spatial relationships used in COSMOS-7

MPEG-7 relation	***MPEG-7 inverse relation***
South	*North*
West	*East*
Northwest	*Southeast*
Southwest	*Northeast*
Left	*Right*
Below	*Above*
Over	*Under*

An example of spatial relationships between objects in COSMOS-7 is given below. The example shows a video, with *id* Squirrel-V, that has a segment defined on its entirety, *id* WholeSquirrel-V-VS.

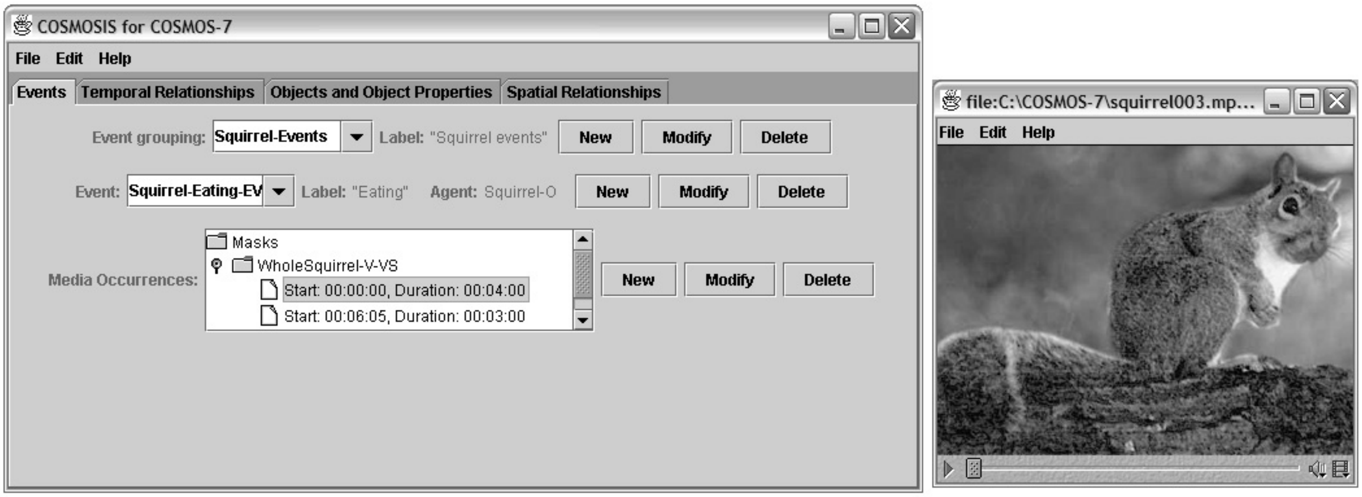
Two moving regions are then defined with *id*s Squirrel-MR and Tree-MR which are related to the Squirrel-O and Tree-O objects through the *SemanticRef* elements. The entire segment is then split into two segments. Both of these segments inherit the two moving regions and specify spatial relationships. The spatial relationship shown depicts that the squirrel is above the tree.

```
<Video id="Squirrel-V">
      <MediaLocator>
            <MediaUri>
                  squirrel003.mpg
            </MediaUri>
      </MediaLocator>
      <MediaTime>
            <MediaTimePoint>T00:00:00</MediaTimePoint>
            <MediaDuration>PT1M30S</MediaDuration>
      </MediaTime>
      <TemporalDecomposition gap="false" overlap="false">
            <VideoSegment id="WholeSquirrel-V-VS">
                  <Relation type="urn:mpeg:mpeg7:cs:SpatialRelationCS:2001:below"
                          source="#Squirrel-MR" target="#Tree-MR"/>
                  <SpatioTemporalDecomposition gap="true" overlap="false">
                        <MovingRegion id="Squirrel-MR">
                              <SemanticRef idref="Squirrel-O"/>
                        </MovingRegion>
                        <MovingRegion id="Tree-MR">
                              <SemanticRef idref="Tree-O"/>
                        </MovingRegion>
                  </SpatioTemporalDecomposition>
                  <TemporalDecomposition>
                        <VideoSegment id="WholeSquirrel-V-VS-1">
                              <Semantic>
                                    <Label>
                                          <Name>Spatial relationships</Name>
                                    </Label>
                                    <Graph>
                                          <Relation
                              type="urn:mpeg:mpeg7:cs:SpatialRelationCS:2001:above"
                              source="#Squirrel-MR" target="#Tree-MR"/>
                                    </Graph>
                              </Semantic>
                              <MediaTime>
                                    <MediaTimePoint>T00:00:00</MediaTimePoint>
                                    <MediaDuration>PT0M15S</MediaDuration>
                              </MediaTime>
                        </VideoSegment>
                        <VideoSegment id="WholeSquirrel-V-VS-2">
                              <MediaTime>
                                    <MediaTimePoint>T00:00:15</MediaTimePoint>
                                    <MediaDuration>PT0M30S</MediaDuration>
                              </MediaTime>
                        </VideoSegment>
                  </TemporalDecomposition>
            </VideoSegment>
      </TemporalDecomposition>
</Video>
```

3.5. COSMOSIS

COSMOSIS is a front-end application, developed in Java, that supports the creation and editing of COSMOS-7 files in MPEG-7 format. Validation is supported through the service provided by NIST (NIST 2005). Several existing MPEG-7 authoring tools, e.g. the MPEG-7 Metadata Authoring Tool (Ryu et al. 2002) and Mdefi (Tran-Thuong and Roisin 2003), display the authored MPEG-7 descriptions in a tree-like XML structure, but COSMOSIS instead focuses on structuring the descriptions around the semantic aspects. Consequently, four tabbed panes cater for each of the semantic video content aspects identified previously. In this way, users do not get over-involved with the intricate details of the MPEG-7 specification and the separation of semantic video content aspects via tabbed panes enables relevant information from each of the main sections of the COSMOS-7 file to be displayed together while also permitting comfortable cross-referencing between sections. All references and labels may be freely decided upon by the user, according to their requirements and the conventions of the organisation and domain of application. Prior references, such as those to segments and objects, may be chosen from a list of automatically-generated identifiers, which updates as new references are created. This maintains consistency and reduces the risk of errors during data creation and modification. COSMOSIS also supports the direct viewing and editing of the source file for users that wish to work in this manner, and changes made in this mode are also automatically reflected in the tabbed panes (with suitable error warnings issued when potential inconsistencies arise).

A screenshot of COSMOSIS is shown in Fig. 2, which depicts editing of the events given in the example in the previous section. A particular event grouping identifier is selectable from a drop down list which causes the corresponding label to be automatically displayed. The events contained within the selected event grouping may then be chosen from the second drop down list. Again, the corresponding label is displayed, together with the event's specified agent (if applicable) and the media occurrence masks. All displayed information is editable via the New, Modify and Delete buttons.



Event grouping: Squirrel-Events Label: "Squirrel events"

Event: Squirrel-Eating-EV Label: "Eating" Agent: Squirrel-O

Media Occurrences:
Masks
WholeSquirrel-V-VS
Start: 00:00:00, Duration: 00:04:00
Start: 00:06:05, Duration: 00:03:00

Fig. 2: COSMOSIS

A Java Media Framework player (shown to the right) runs simultaneously with COSMOSIS to enable the location of segment start and duration parameters. These may be entered directly by the user or retrieved in real-time from the media player. The player also serves to help the content creator visually verify and validate the semantic content information. Many existing MPEG-7 authoring tools, such as the MPEG-7 Metadata Authoring Tool (Ryu et al. 2002) and VideoAnnEx (Tseng et al. 2002; IBM 2005) take video segmentation as the first stage of the authoring process, which is automatically defined when a video file is first loaded into the authoring tool. In contrast, COSMOSIS leaves the video segmentation to the content modeller, to carry out during the content modelling process. This ensures that content modelling does not revolve around a decomposition that is tied to a technical structure but to actual content and makes it easier for multiple perspectives to be defined on the same or overlapping segments.

4. SEMANTIC CONTENT-BASED FILTERING WITH COSMOS-7

A user will very often only be interested in certain video content, e.g. when watching a soccer game the user may only be interested in goals and free kicks. Identifying and retrieving subsets of video content in this way requires user preferences for content to be stated, such that content within a digital video resource may then be filtered against those preferences. While new approaches, such as those based on agents (Wenyin et al. 2003), are emerging, filtering in video streams usually uses content-based filtering methods, which analyse the features of the material so that these may then be filtered against the user's content requirements (Ferman et al. 2002; Wallace 2002; Angelides 2003; Eirinaki and Vazirgiannis 2003). During the analysis, a set of key attributes is identified for each feature and the attribute values are then populated. The attribute values can either be simple preference ratings, such as those based on the Likert scale, or they can be weighted values, to indicate the relative success of the match of the attribute value to those preferences expressed by the user (Kuflik and Shoval 2000; van Meteren and van Someren 2000). The former type of values are commonly used in recommender systems, where content is recommended to the user based on their prior history, while the latter type of values tend to be employed in content-based retrieval domains, where the user is 'hunting' or 'exploring' content more specifically.

Content-based filtering is consequently most suitable when a computer can easily analyse the entities and where entity suitability is not subjective (Good et al. 1999; Specht and Kahabka 2000). Content-based filtering is also most effective when the content being analysed has features homogenous with the user's content preferences, since heterogeneous features across the two domains would give rise to incompatibilities, i.e. not comparing like with like. This problem is particularly apparent when new content is introduced into the resource, since this may contain new features not previously contemplated. The use of standards which are comprehensive in their consideration of semantic aspects, such as MPEG-7, overcomes this problem.

4.1. COSMOS-7 filters

Using a content-based filtering approach, COSMOS-7 matches the various semantic content of the digital video resource to the specified filter criteria of the user. This is achieved by building the content filter from the part of the COSMOS-7 content model that the user content requirements directly map on to. In this way, only content that relates directly to the preferred content requirements of the user is analysed and chosen. The filter is revisited every time new content is added or the user requirements change. Content filters are specified using the **FILTER** keyword together with one or more of the following: **EVENT**, **TMPREL**, **OBJ**, **SPLREL**, and **VIDEO**. These may be joined together using the logical operator AND. This specifies what the filter is to return. Consequently, the output of the filtering process may be either semantic content information and/or video segments containing the semantic content information. The criteria are specified using the **WHERE** keyword together with one or more of the following clauses: **EVENT**, **TMPREL**, **OBJ**, and **SPLREL** clauses. These clauses enable event, temporal relationship, object and their properties and spatial relationships, respectively, to be specified on the COSMOS-7 representation. Clauses may be joined together using the logical operators AND, NOT and OR and are terminated with semi-colons and grouped with braces. For example, consider that the user wishes to filter both events and video where:

- events occur during the first 30 seconds or during the 59th minute of the video footage
- the agent of the events is the squirrel object
- the events precede the squirrel sleeping event
- the video footage that reflects the events features trees and lakes

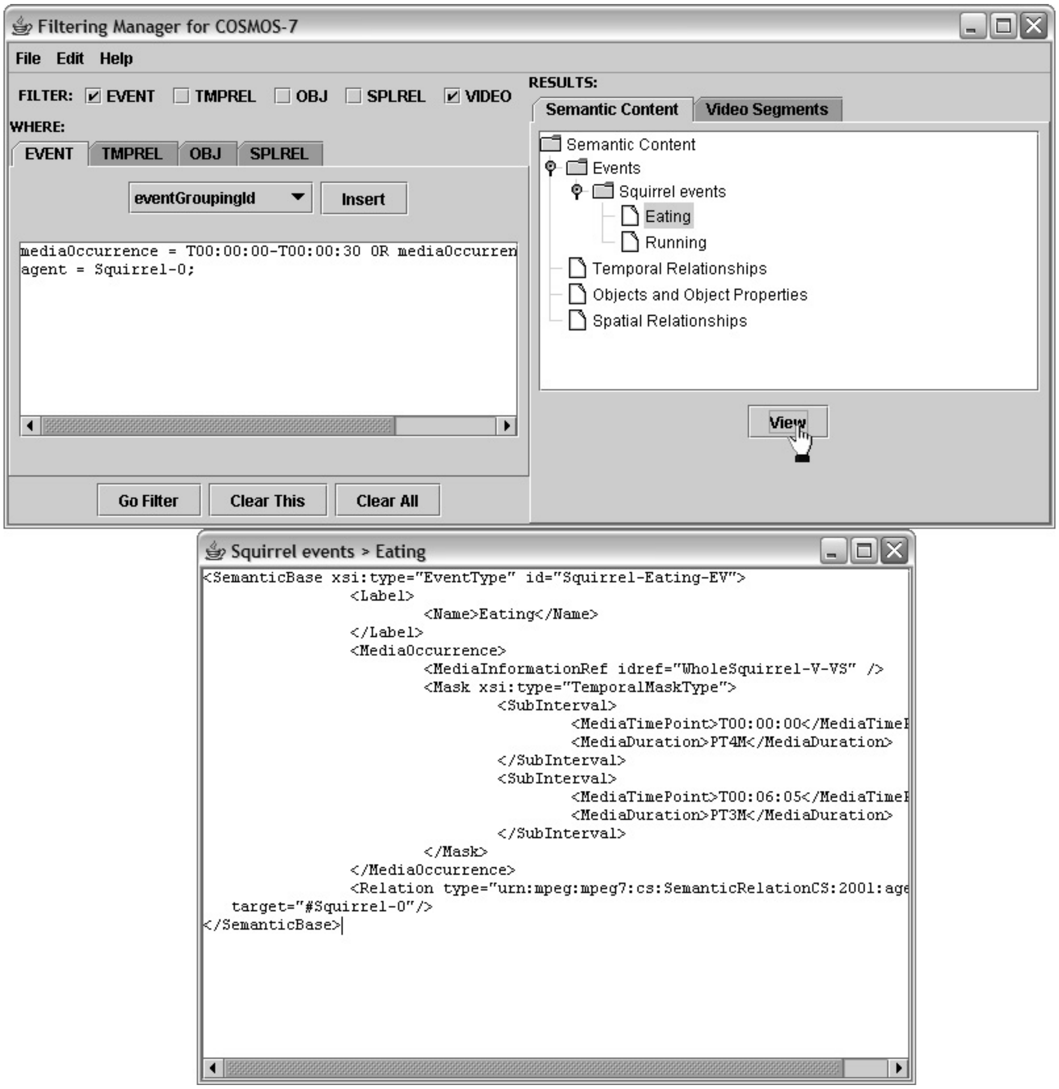
The resultant filter would look like so:

```
FILTER EVENT AND VIDEO WHERE {
      EVENT {
            mediaOccurrence = T00:00:00-T00:00:30 OR mediaOccurrence = T00:59:00;
            agent = Squirrel-O;
      }
      TMPREL {
            temporalRelationshipType = precedes;
            temporalRelationshipTarget = Squirrel-Sleeping-EV;
      }
      SPLREL {
            movingRegion = Tree-MR AND movingRegion = Lake-MR;
      }
}
```

This would return all events and all references to video that match the above criteria. Note that the names of the objects and events would be dictated by the conventions of the content modeller and/or their organisation; the use of “-O” after an object name and “-EV” after an event name are merely our current conventions following on from the previous examples. Details of all the clauses that may be used within a COSMOS-7 filter are provided in the Appendix.

4.2. Filtering Manager for COSMOS-7

Like COSMOSIS, the Filtering Manager is a Java-based front-end application. Its purpose is to construct and execute filters for COSMOS-7. The main dialog screen is divided into two areas: the left pane enables the user to specify the filtering conditions, while the right pane displays the results. This is illustrated in Fig. 3, which shows the results of the example filter given above. The filtering pane requires the user to select one or more aspects that they wish to filter on, which maps on to the filter keywords presented earlier. Various condition clauses may then be constructed in the ‘WHERE’ area. To ensure full flexibility, the clauses are stated in text, but the various clause keywords may be selected and inserted from a drop-down list box. Four tabbed panes enable the specification of the event, temporal relationship, object and object property, and spatial relationship clauses. Once specified, filters may be saved for later retrieval.



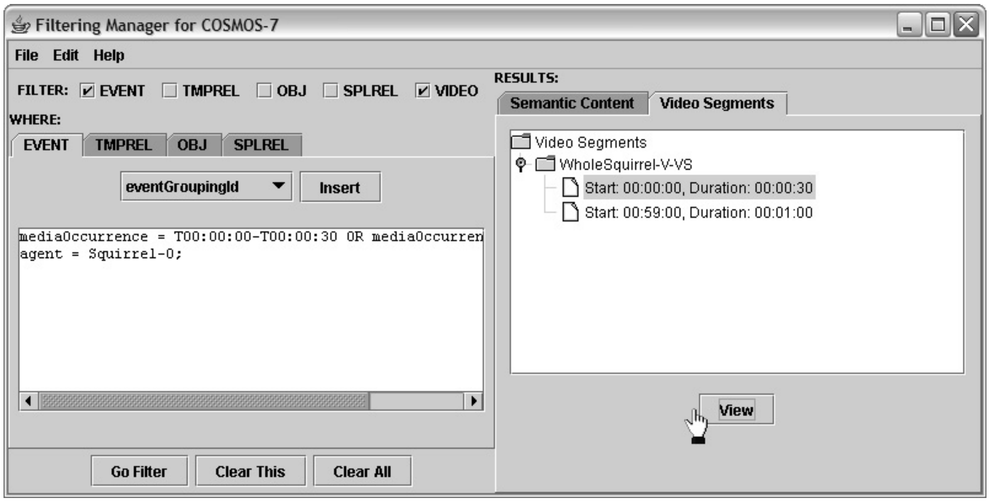
FILTER: ☑ EVENT ☐ TMPREL ☐ OBJ ☐ SPLREL ☑ VIDEO

```
mediaOccurrence = T00:00:00-T00:00:30 OR mediaOccurren
agent = Squirrel-0;
```

```
<SemanticBase xsi:type="EventType" id="Squirrel-Eating-EV">
                <Label>
                        <Name>Eating</Name>
                </Label>
                <MediaOccurrence>
                        <MediaInformationRef idref="WholeSquirrel-V-VS" />
                        <Mask xsi:type="TemporalMaskType">
                                <SubInterval>
                                        <MediaTimePoint>T00:00:00</MediaTime
                                        <MediaDuration>PT4M</MediaDuration>
                                </SubInterval>
                                <SubInterval>
                                        <MediaTimePoint>T00:06:05</MediaTime
                                        <MediaDuration>PT3M</MediaDuration>
                                </SubInterval>
                        </Mask>
                </MediaOccurrence>
                <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:age
   target="#Squirrel-0"/>
</SemanticBase>
```

Fig. 3: Returned semantic content within COSMOS-7 filtering results

Once a filter is executed, the results are displayed in the right-hand area, within two tabbed panes. The first pane contains the semantic content results, displayed as *label*s within a tree structure, grouped by semantic aspect. These may be selected for viewing in COSMOS-7 format, if required, as is illustrated by the bottom half of Fig. 3. The second pane contains the video segment results (if these were chosen by the user as part of the filter condition). Again, these are shown within a tree structure, grouped by the *idref* of the video segments. This is shown in Fig. 4. Each video segment may be selected and viewed by the user and the complete set of results may be saved for later retrieval, if required.



FILTER: ☑ EVENT □ TMPREL □ OBJ □ SPLREL ☑ VIDEO

```
mediaOccurrence = T00:00:00-T00:00:30 OR mediaOccurren
agent = Squirrel-0;
```

```
Video Segments
  WholeSquirrel-V-VS
    Start: 00:00:00, Duration: 00:00:30
    Start: 00:59:00, Duration: 00:01:00
```

Fig. 4: Returned video segments within COSMOS-7 filtering results

Once a filter is specified, it is first decomposed by the Filtering Manager. This decomposition divides the entire filter into its respective semantic content aspect clauses, i.e. those specified as criteria for EVENTS, TMPREL, OBJ and SPLREL. The filter return criteria, i.e. the keywords specified at the start of the filter, are also decomposed. Clauses are then mapped from their filter representation to equivalent COSMOS-7 representation, in terms of elements and values, in preparation for a matching process. The matching process works in conjunction with a non-validating SAX (Simple API for XML) parser to match segments of the COSMOS-7 content model against the elements and values for each of the filter clauses. A non-validating SAX parser is used due to the potentially large size of COSMOS-7 content models and the need to minimise memory requirements. SAX parsers have the advantage of reporting parsing events, such as start and end elements, through callbacks which are processed by event handlers. Memory consumption therefore does not increase with document size. Specific reasoning logic within the matching process exists for each type of semantic content aspect. For example, the matching of temporal relationships against filtering criteria requires processing of the graph structures and reasoning from these regarding temporal sequencing. Appropriate matching content and/or video streams, depending on the specified filter return criteria, are then returned. In this way, the returned content can be considered to be a subset of the full content model and thus the filtering process reduces the content model to those segments relevant to the filtering criteria. Fig. 5 illustrates these key filtering processes.

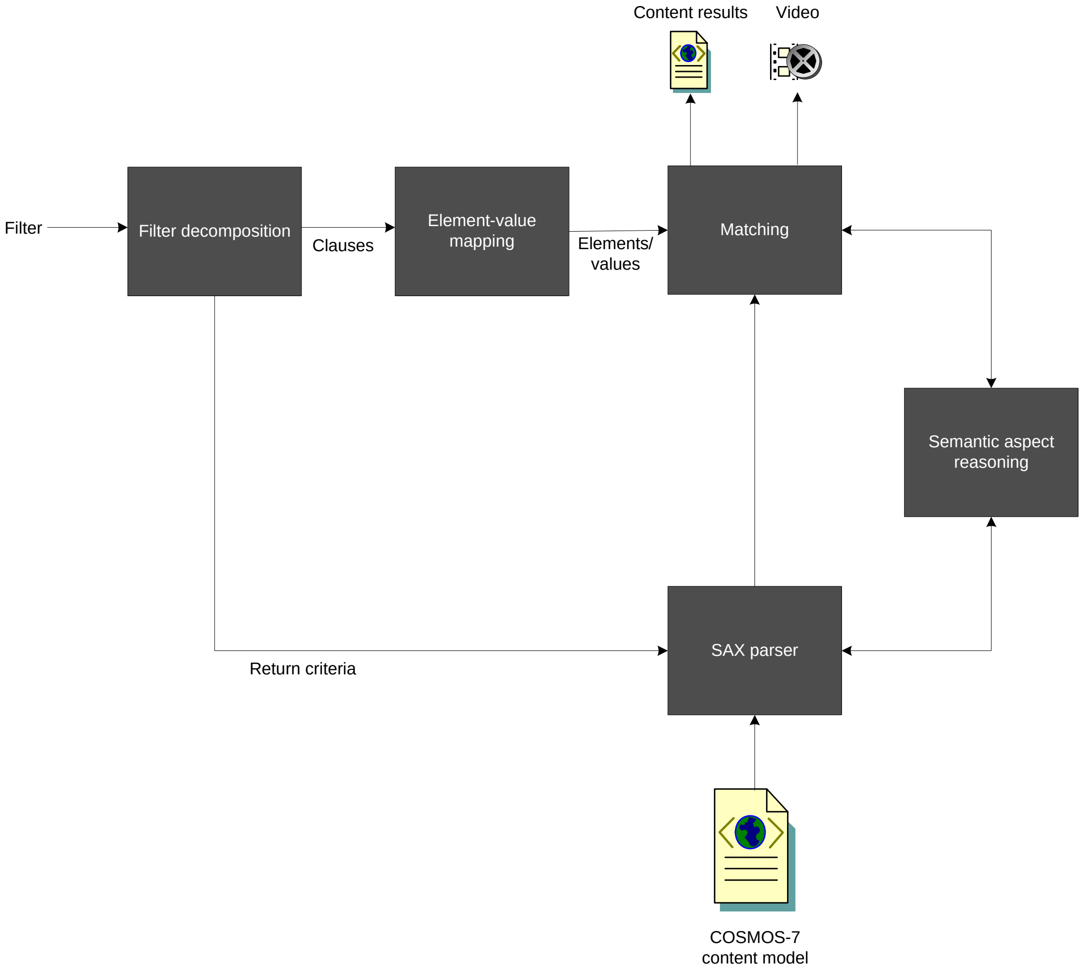


Fig. 5: Filtering processes

5. EXPERIENCES: "THREE WEEKS AND ONE DAY WITH COSMOS-7"

In presenting their MoCA Workbench, Lienhart, Pfeiffer and Effelsberg (1996) describe a typical day with users. Inspired by this approach, we chose to evaluate COSMOS-7 filtering with a group of non-specialist users over the course of a single day after they had undertaken content modelling with COSMOS-7 over a period of three weeks. Ten users were drawn from a theatre company located in north London, UK, which is serving as evaluators on another research project we are involved in. The company currently have a large database of reusable theatre material in various formats, including video, which covers the whole process of a theatrical production, such as stage designs and mock-ups, costumes, and the final show itself. As it exists at the moment, this material is indexed only via a basic keywords field within the database.

5.1. Research method

We adopted an action research method. Action research is a flexible spiral process which allows action (change, improvement) and research (understanding, knowledge) to be achieved simultaneously. Understanding allows more informed change and at the same time is informed by that change. People affected by the change are usually involved in the action research. This allows the understanding to be widely shared and the change to be pursued with commitment. At the start of the first week, the users were given a full-day training session involving an introduction to MPEG-7, COSMOS-7 and use of both COSMOSIS and the Filtering Manager software. Evaluation then proceeded within a dialectical and qualitative action group research method based on elements of Delphi (Okoli and Pawlowski 2004; Rowe et al. 2005) and eXtreme Programming (Ambler 2002; Germain and Robillard 2005), which we term *eXtreme Delphi*. Users worked together in pairs to model content within video footage from the theatre company’s existing database. Since the video footage within the database was of various durations and featured various content, we split the footage randomly into five sections of roughly equal total duration (approximately 48 hours each) which were then randomly allocated to each of the five pairs of users. This was done because it was obvious that no pair would be able to content model everything within their allocated footage and doing it this way: (1) ensured that no user pair would run out of footage to content model during the three weeks, (2) ensured that there would be no overlap in the footage that user pairs were content modelling, and (3) would provide indicators of completion progress and productivity.

During the course of each day, the pair prepared a progress report which included items they wished to discuss with the rest of the group and with ourselves as facilitators the following morning. We used structured discussion to facilitate this cyclical method where each pair of users considered, discussed and compared suggestions in light of the whole group’s views. We asked the group to suggest modifications that preserved the advantages for the group, while relieving the problems for individuals. Where user pairs had differences from the whole group, the user pair had the option of either changing to conform more nearly to the whole group or develop evidence for changing the views of the whole group in the direction of the pair. The purpose of the latter was not to persuade others to their point of view, but to present evidence which they think others may have overlooked. In light of the debate, user pairs then considered what changes they wished to implement within their own approaches to content modelling during the remainder of the day. With each meeting, individual and paired content modelling work becomes more cohesive across the entire group, objectives are better understood among the groups, user pairs and individual users, and users realise the benefits and disadvantages of various modelling approaches.

At the end of the three weeks, users were then required to undertake and discuss a number of filtering tasks over the course of a day using the Filtering Manager for COSMOS-7 to locate footage that they would typically need when working on an international theatre production.

5.2. Key results

With regards content modelling, the key issue arising from the daily user discussions was related to content modelling strategies. During the first week, all user pairs worked by choosing a piece of footage and content modelling it. Thus, their content modelling was *video-stream-led*. Typically, they started with footage of short duration, no doubt as they found it more comfortable to build their experience with. During the first few days, user discussions centred on basic details of content modelling; significantly, users discussed what should be considered an object and what should be considered an event as well as which objects and events should be considered significant enough to content model. For example, one discussion revolved around whether the entire costume that an actor wears or the individual garments themselves should be considered objects. As users moved on to other footage during the first week, most user pairs (four out of the five) began to realise that there was some redundancy in their modelling whereby they were modelling similar entities (objects and events) that had been present in previous footage. This led one user pair to present this for discussion to the group at the end of the first week and the other user pairs shared their experiences and thoughts on how to deal with this. Consequently, during the second week, content modelling became *entity-led* (objects and events) and by the end of the second week, users decided that the best strategy to adopt for entity-led content modelling was where they would browse through several hours worth of footage, decide on key objects and events common to multiple sets of footage, and then content model those entities. They would then work through the footage sequentially, referencing relevant video segments and adding pertinent additional properties, as well as the key spatial and temporal relationships. It was not until the middle of the third week that users began to discuss hierarchical issues, e.g. object inheritance. Unfortunately, by this time they were reaching the end of the evaluation period so they had only a short time in which to implement these strategies. Nevertheless, users reported that the use of hierarchies would have proven extremely beneficial had they adopted this approach much earlier. Users also reported throughout the second and third weeks that they benefited greatly from being able to model an entity once and then refer to that entity when modelling other footage. This provided an holistic approach and while they found it more time-consuming in the short-term, the benefits were felt in the long-term and over a significant period of time they felt that effort and tedium were reduced.

Throughout the three weeks, users reported that events were the easiest aspect to content model, with objects, specifically object properties, being the most difficult. We inferred that this was due to the amount of time involved in modelling object properties and the frequency of occurrence of objects within the video streams.

Fig. 6 shows the productivity of the user pairs over the three week period. Note that Day 1 is omitted since, as stated earlier, it was a training day. No work was carried out over weekends either, so these are also omitted. While the data shows that productivity differed between the pairs, the productivity of most user pairs rose steadily over the period, with Week 2 in general showing the most productivity. We can attribute this to the entity-led modelling strategy that was adopted by all pairs during that week. In the third week, all but one pair of users displayed a drop in productivity, particularly towards the end of the week. From the user meetings during that week, we believe the impending completion of the exercise to be the most likely cause of this, whereby users slowly began to ‘wind down’ as the exercise became less relevant to them and they became more interested in changing tasks, i.e. moving on to the filtering part of the evaluation. Examination of the content models created by the user pairs showed that users varied in how much detail they included and this can also be seen to account for differences in productivity between the pairs.

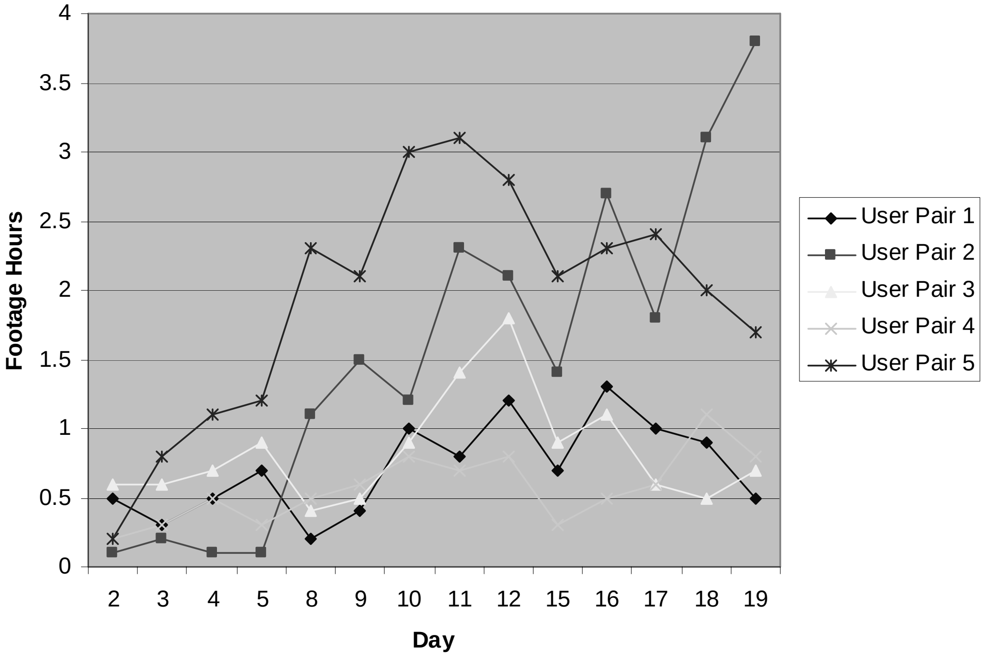


Fig. 6: Daily user-pair productivity over the full period

Also in the meetings, users discussed issues relating to the user interface of COSMOSIS. Specifically, they found it took some time to gain familiarity with the structure and layout because it assumed knowledge of COSMOS-7 which, despite the initial training day, they were still not fully comfortable with at that time. As the week progressed, users became more comfortable with both the software and COSMOS-7 and thus this became less of issue for discussion during meetings. Other than this, user comments were generally positive towards the software, with users finding the structured approach appropriate. However, some users commented that the fact that only one content aspect was visible at any one time, due to the tabs, was not the best approach and that they would have preferred to have been able to view all content aspects on the screen simultaneously. Users also commented that the use of drop-down list boxes for the various objects and events worked well initially but became cumbersome as the size of content that had already been modelled increased.

With regards filtering, we asked user pairs to choose a typical international theatrical production that they might be working on, perhaps one which they were currently working on at the company. Each user pair was then asked to create and execute several filters on the content model according to typical requirements. For this exercise, we provided access to the content models that *all* user pairs had created. This enabled more footage to be available to the filters and provided a more realistic working environment. To avoid networking issues, identical copies of the content models were installed on all workstations. We allowed users to present issues to the group and ourselves on every hour, as they deemed necessary. As with the modelling, users began by creating small filters that focused on a single semantic aspect and were only tangentially related to their chosen international production. In these short, early stages, progress was slow. However, as they became more comfortable with the Filtering Manager, their filters began to expand in both relevance and complexity and users subjectively judged their productivity and progress to have increased accordingly.

Fig. 7 shows the proportions of each semantic content aspect involved in all the user-created filters. As can be seen, objects and object properties featured most prominently, followed by events. The temporal and spatial relationships were also used but to a lesser extent. Users initially reported that this was because they had less need for this specific type of filtering. However, it emerged later that this was also because they had less experience with these types of relationships and that this had restricted how they thought about them when trying to filter content.

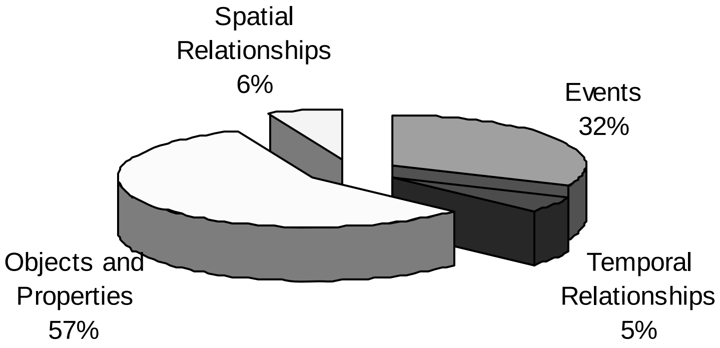


Fig. 7: Semantic aspects involved in user-created filters

Users reported that their experience with COSMOS-7 content modelling from the previous three weeks had helped to speed their learning since they could relate the filter directly to the structure of the model and also, when applicable, to the elements they had modelled. All users also reported that they had much more control over the filtering process than with the keywords-based approach that they were currently used to and commented that this was particularly beneficial for databases that contained a large number of video resources. Of note, users commented that the approach of COSMOS-7 would be beneficial in their work because, not only did it help them to locate footage, but to locate specific segments within, and content information relating to, that footage and that this was felt more when the footage was lengthy (over thirty minutes).

As the day went on, users began to raise issues relating to the software. Unlike with the comments regarding the user interface of COSMOSIS, users generally took quickly to the structure and layout of the Filtering Manager, no doubt due to the familiarity they had gained with COSMOSIS and COSMOS-7 through the content modelling part of the evaluation. User comments were, again, generally positive towards the software. However, contrary to the issues raised about the tabbed interface of COSMOSIS, users reported favourably on the tabbed interface of the Filtering Manager.

The most popular reason cited for this was that it helped to structure their thinking when creating a filter and to focus them on preparing specific clauses; it was felt that having all parts of the filter visible at the same time would have proven distracting.

Table 3 provides a summary of the results reported above.

6. CLOSING DISCUSSION

The accuracy and reliability of filtered content to user requirements relies on two factors: the richness and depth of detail used in the creation of the content model and the level of content-dependency of the filtering techniques employed. Furthermore, the interoperability of user-driven information retrieval, especially across platforms, is greatly enhanced if the underlying process is standardised. This is achieved through the various MPEG-7 parts and in our case through MPEG-7 Part 5. Part 5 allows the modelling of multimedia content using a rich set of description schemes that describe both low-level syntactic and high-level semantic concepts that can be used by any MPEG-7 compliant scheme. However, while MPEG-7 prescribes the format and structure of data, it does not specify how that data may be processed or used. COSMOS-7 is one approach to utilising MPEG-7 both for semantic-content-based modelling and filtering, which has evolved from our pre-MPEG-7 work on content modelling. Our future work will address some of the insufficiencies in the user interfaces of the COSMOSIS and Filtering Manager systems that were reported during the evaluation by adding an additional layer to further remove unnecessary complexity. We are also currently undertaking research and development in improving the visualisation and exploration of the filtering results within the Filtering Manager. In the longer term, with the openness of COSMOS-7 and the evolution of another MPEG standard, MPEG-21, the considerations of content adaptation according to both the user’s consumption environment and preferences (Vetro 2004) come to the fore.

Table 3: Summary of results

	A. Modelling			**B. Filtering**	
	A.1. Video-stream-led modelling	**A.2. Entity-led modelling**	**A.3. Advanced modelling**	**B.1. Single content aspect filtering**	**B.2. Multiple content aspect filtering**
COSMOS-7 components:	*Objects and Object Properties:* Semantic DS (Object DS, SemanticState DS) *Events:* Semantic DS (Event DS) Scattered modelling of temporal and spatial relationships.	As A.1, plus: *Object/event interrelationships:* SemanticRelation CS (agent, agentOf) *Temporal Relationships:* Semantic DS (Graph DS), TemporalRelation CS *Spatial Relationships:* VideoSegment DS (MovingRegion DS, Graph DS), SpatialRelation CS	As A.2, plus: *Objects and Object Properties:* SemanticRelation CS (specializes, generalizes)	As A.3 (all components). Use is mutually exclusive according to content aspect.	As A.3 (all components). Use is not mutually exclusive according to content aspect.
User activities:	Primary focus on objects and events appearing in video stream.	Primary focus on objects and events appearing across multiple video streams. Non-exploratory modelling of spatial and temporal relationships, as well as relationships between events and objects.	As A.2, plus non-exploratory modelling of object hierarchies.	Tangential relation to assigned task. Exploration. Objects and object properties and events most used content aspects.	High relevance to assigned task. Objects and object properties and events most used content aspects.
User productivity:	Low, rising.	Highest.	Medium.	Low, rising.	Highest.

.

REFERENCES

Adali, S., Candan, K. S., Chen, S.-S., Erol, K. and Subrahmanian, V. S. (1996). The Advanced Video Information System: data structures and query processing. *Multimedia Systems 4*(4), 172-186.

Adami, N., Bugatti, A., Leonardi, R., Migliorati, P. and Rossi, L. A. (2001). The ToCAI description scheme for indexing and retrieval of multimedia documents. *Multimedia Tools and Applications 14*(2), 153-173.

Agius, H. W. and Angelides, M. C. (1999). COSMOS - Content Oriented Semantic Modelling Overlay Scheme. *The Computer Journal 42*(3), 153-176.

Agius, H. W. and Angelides, M. C. (2000). A method for developing interactive multimedia from their semantic content. *Data & Knowledge Engineering 34*(2), 165-187.

Agius, H. W. and Angelides, M. C. (2001). Modelling content for semantic-level querying of multimedia. *Multimedia Tools and Applications 15*(1), 5-37.

Al-Safadi, L. and Getta, J. (2001). Semantic content-based retrieval for video documents. In S. M. Rahman (Ed.), *Design and Management of Multimedia Information Systems: Opportunities & Challenges*. (pp. 165-200). Hershey, PA: IDEA Group Publishing.

Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM 26*(11), 832-843.

Ambler, S. W. (2002). *Agile Modelling: Effective Practices for eXtreme Programming and the Unified Process*. New York, NY: John Wiley & Sons.

Angelides, M. C. (2003). Multimedia content modelling and personalization. *IEEE Multimedia 10*(4), 12-15.

Appan, P. and Sundaram, H. (2004). Networked Multimedia Event Exploration. *Proceedings of ACM Multimedia '04*, New York, NY, October 10-16, 40-47.

Benitez, A. B. and Chang, S.-F. (2003). Extraction, Description and Application of Multimedia Using MPEG-7. *Proceedings of the 37th Asilomar Conference on Signals, Systems & Computers*, IEEE Press, 92-96.

Bolle, R. M., Yeo, B.-L. and Yeung, M. M. (1998). Video query: research directions. *IBM Journal of Research & Development 42*(2).

Chu, Y., Chia, L.-T. and Bhowmick, S. S. (2004). Looking at Mapping, Indexing & Querying of MPEG-7 Descriptors in RDBMS with SM3. *Proceedings of ACM MMDB '04*, Washington, DC, USA, November 13, 55-64.

Correia, P. and Pereira, F. (1998). The role of analysis in content-based video coding and indexing. *Signal Processing 66*, 125-142.

Costa, M., Correia, N. and Guimarães, N. (2002). Annotations as Multiple Perspectives of Video Content. *Proceedings of ACM Multimedia '02*, Juan-les-Pins, France, December 1-6, 283-286.

Dasiopoulou, S., Papastathis, V. K., Mezaris, V., Kompatsiaris, I. and Strintzis, M. G. (2004). An Ontology Framework For Knowledge-Assisted Semantic Video Analysis and Annotation. *Proceedings of the 4th International Workshop on Knowledge Markup and Semantic Annotation (SemAnnot 2004) at the 3rd International Semantic Web Conference (ISWC 2004)*, November.

Davis, M. (1993). Media Streams: An Iconic Visual Language for Video Annotation. *Telektronikk 4.93*, 59-71.

Davis, M., King, S., Good, N. and Sarvas, R. (2004). From Context to Content: Leveraging Context to Infer Media Metadata. *Proceedings of ACM Multimedia '04*, New York, NY, October 10-16, 188-195.

Day, Y. F., Khokhar, A., Dagtas, S. and Ghafoor, A. (1999). A multi-level abstraction and modeling in video databases. *Multimedia Systems 7*(5), 409-423.

Döller, M., Kosch, H., Dörflinger, B., Bachlechner, A. and Blaschke, G. (2002). Demonstration of an MPEG-7 Multimedia Data Cartridge. *Proceedings of the 10th ACM International Conference on Multimedia (MM02)*, Juan-les-Pins, France, December 1-6, 85-86.

Echiffre, M., Marchisio, C., Marchisio, P., Panicciari, P. and Del Rossi, S. (1998). MHEG-5 - aims, concepts, and implementation issues. *IEEE Multimedia 5*(1), 84-91.

Echigo, T., Masumitsu, K., Teraguchi, M., Etoh, M. and Sekihuchi, S. (2001). Personalized delivery of digest video managed on MPEG-7. *Proceedings of the 2001 International Conference on Information Technology: Coding and Computing*, Las Vegas, NV, USA, 2-4 April, IEEE Press, Piscataway, NJ, 216-220.

Eidenberger, H. (2003). Distance Measures for MPEG-7-based Retrieval. *Proceedings of ACM MIR'03*, Berkeley, CA, USA, November 7, 130-137.

Eirinaki, M. and Vazirgiannis, M. (2003). Web mining for web personalization. *ACM Transactions on Internet Technology 3*(1), 1-27.

Ferman, A. M., Beek, J. H. E. P. v. and Sezan, M. I. (2002). Content-Based Filtering and Personalization Using Structured Metadata. *Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries*, Portland, Oregon, July, 393.

Ferman, A. M., van Beek, P., Errico, J. H. and Sezan, M. I. (2003). Multimedia content recommendation engine with automatic inference of user preferences. *Proceedings of the IEEE International Conference on Image Processing*, Vol. 3, 49-52.

Fonseca, P. M. and Pereira, F. (2004). Automatic video summarization based on MPEG-7 descriptions. *Signal Processing: Image Communication 19*, 685-699.

Germain, É. and Robillard, P. N. (2005). Engineering-based processes and agile methodologies for software development: a comparative case study. *Journal of Systems and Software 75*(1-2), 17-27.

Goldfarb, C. S. (1991). HyTime: a standard for structured hypermedia interchange. *Computer 24*(8), 81-84.

Golshani, F. and Dimitrova, N. (1998). A language for content-based video retrieval. *Multimedia Tools and Applications 6*(3), 289-312.

Good, N., Schafer, J., Konstan, J., Borchers, A., Sarwar, B., Herlocker, J. and Riedl, J. (1999). Combining collaborative filtering with personal agents for better recommendations. *Proceedings of the 16th National Conference on Artificial Intelligence*, 439-446.

Goularte, R., Cattelan, R. G., Camacho-Guerrero, J. A., Inácio Jr., V. R. and Pimentel, M. d. G. C. (2004). Interactive Multimedia Annotations: Enriching and Extending Content. *Proceedings of DocEng'04*, Milwaukee, Wisconsin, USA, October 28–30, 84-86.

Goularte, R., Moreira, E. d. S. and Pimentel, M. d. G. C. (2003). Structuring Interactive TV Documents. *Proceedings of DocEng'03*, Grenoble, France, November 20–22, 42-51.

Graves, A. and Lalmas, M. (2002). Video retrieval using an MPEG-7 based inference network. *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Tampere, Finland, August 11-15, ACM Press, New York, NY, 339-346.

Hartley, E., Parkes, A. P. and Hutchison, A. D. (2000). A conceptual framework to support content-based multimedia applications. *Lecture Notes in Computer Science*. *1629***:** 297-.

Hunter, J. (2003). Enhancing the semantic interoperability of multimedia through a core ontology. *IEEE Transactions on Circuits and Systems for Video Technology 13*(1), 49-58.

IBM (2005). IBM VideoAnnEx website. *http://www.research.ibm.com/VideoAnnEx/.*

ISO/IEC (2002). Information Technology – Multimedia Content Description Interface – Part 5: Multimedia Description Schemes. Geneva, Switzerland, International Organisation for Standardisation.

Jaimes, A., Echigo, T., Teraguchi, M. and Satoh, F. (2002). Learning personalized video highlights from detailed MPEG-7 metadata. *Proceedings of the 2002 IEEE International Conference on Image Processing (ICIP-02),* Vol. 1, Rochester, NY, 22-25 September, IEEE Press, Piscataway, NJ, 133-136.

Jaimes, A., Omura, K., Nagamine, T. and Hirata, K. (2004). Memory Cues for Meeting Video Retrieval. *Proceedings of ACM CARPE'04*, New York, NY, USA, October 15, 74-85.

Kang, J.-H., Kim, C.-S. and Ko, E.-J. (2003). An XQuery Engine for Digital Library Systems. *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries*, Houston, TX, 400.

Kosch, H. (2002). MPEG-7 and Multimedia Database Systems. *ACM SIGMOD Record 31*(2), 34-39.

Kretz, F. and Colaïtis, F. (1992). Standardizing hypermedia information objects. *IEEE Communications Magazine 30*(5), 60-70.

Kuflik, T. and Shoval, P. (2000). Generation of user profiles for information filtering - research agenda. *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 313-315.

Kuo, P.-J., Aoki, T. and Yasuda, H. (2004). Building Personal Digital Photograph Libraries: An Approach with Ontology-Based MPEG-7 Dozen Dimensional Digital Content Architecture. *Proceedings of IEEE Computer Graphics International (CGI'04)*.

Lee, J.-H., Kim, H.-J. and Kim, W.-Y. (2003). Video Image Retrieval System based on MPEG-7 (VIRS). *Proceedings of the International Conference on Information Technology: Research and Education (ITRE '03)*, 79-83.

Lienhart, R., Pfeiffer, S. and Effelsberg, W. (1996). The MoCA Workbench: Support for Creativity in Movie Content Analysis. *Proceedings of the Third IEEE International Conference on Multimedia Computing and Systems*, Hiroshima, Japan, June, 314-321.

Lin, C.-Y., Tseng, B. L., Naphade, M., Natsev, A. and Smith, J. R. (2003a). MPEG-7 Video Automatic Labeling System. *Proceedings of ACM MM'03*, Berkeley, CA, USA, November 2-8, 98-99.

Lin, C.-Y., Tseng, B. L., Naphade, M., Natsev, A. and Smith, J. R. (2003b). VideoAL: a novel end-to-end MPEG-7 video automatic labeling system. *Proceedings of the IEEE International Conference on Image Processing 2003*, Vol. 3, Barcelona, Spain, September 14-17, IEEE Press, Piscataway, NJ, III-53-56.

Löffler, J., Biatov, K., Eckes, C. and Köhler, J. (2002). iFinder: an MPEG-7-based retrieval system for distributed multimedia content. *Proceedings of the 10th ACM International Conference on Multimedia (MM02)*, Juan-les-Pins, France, December 1-6, 431-435.

Magalhães, J. and Pereira, F. (2004). Using MPEG standards for multimedia customization. *Signal Processing: Image Communication 19*, 437–456.

Martínez, J. M., González, C., Fernández, O., Garcia, C. and de Ramón, J. (2002). Towards universal access to content using MPEG-7. *Proceedings of the 10th ACM International Conference on Multimedia (MM02)*, Juan-les-Pins, France, December 1-6, 199-202.

Meyer-Boudnik, T. and Effelsberg, W. (1995). MHEG explained. *IEEE Multimedia 2*(1), 26-38.

Naphande, M. R. and Huang, T. S. (2001). A probabilistic framework for semantic video indexing, filtering, and retrieval. *IEEE Transactions on Multimedia 3*(1), 141-151.

Newcomb, S. R. (1995). Multimedia interchange using SGML/HyTime: Part II: Principles and Applications. *IEEE Multimedia 2*(3), 60-64.

Newcomb, S. R., Kipp, N. A. and Newcomb, V. T. (1991). The "HyTime" hypermedia/time-based document structuring language. *Communications of the ACM 34*(11), 67-83.

NIST (2005). NIST MPEG-7 Validation Service. *http://m7itb.nist.gov/M7Validation.html*.

Okoli, C. and Pawlowski, S. D. (2004). The Delphi method as a research tool: an example, design considerations and applications. *Information & Management 42*(1), 15-29.

Pfeiffer, S. and Srinivasan, U. (2000). TV Anytime as an application scenario for MPEG-7. *Proceedings of the ACM Multimedia Workshop*, Marina Del Rey, CA, 89-92.

Rehm, E. (2000). Representing Internet Streaming Media Metadata using MPEG-7 Multimedia Description Schemes. *Proceedings of the ACM Multimedia Workshop*, Marina Del Rey, CA, USA, 93-98.

Rovira, M., González, J., López, A., Mas, J., Puig, A., Fabregat, J. and Fernàndez, G. (2004). IndexTV: A MPEG-7 Based Personalized Recommendation System for Digital TV. *Proceedings of the 2004 IEEE International Conference on Multimedia and Expo (ICME)*, 823-826.

Rowe, G., Wright, G. and McColl, A. (2005). Judgment change during Delphi-like procedures: The role of majority influence, expertise, and confidence. *Technological Forecasting and Social Change 72*(4), 377-399.

Rowe, L. A., Boreczky, J. S. and Eads, C. A. (1994). Indices for user access to large video databases. *Proceedings of Storage and Retrieval for Image and Video Database II, Proceedings of SPIE*, Vol. 2185, February, 150-161.

Ryu, J., Sohn, Y. and Kim, M. (2002). MPEG-7 metadata authoring tool. *Proceedings of the 10th ACM International Conference on Multimedia (MM02)*, Juan-les-Pins, France, December 1-6, 267-270.

Salembier, P., Qian, R., O'Connor, N., Correia, P., Sezan, I. and van Beek, P. (2000). Description schemes for video programs, users and devices. *Signal Processing: Image Communication 16*, 211-234.

Specht, T. K. G. and Kahabka, T. (2000). Information filtering and personalisation in databases using Gaussian curves. *Proceedings of the IEEE Databases Engineering and Applications Symposium*, Yokohama, Japan, September.

Tran-Thuong, T. and Roisin, C. (2003). Multimedia modeling using MPEG-7 for authoring multimedia integration. *Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrieval*, Berkeley, CA, November 7, ACM Press, New York, NY, 171-178.

Tseng, B. L., Ching-Yung, L. and Smith, J. R. (2002). Video personalization and summarization system. *Proceedings of the IEEE Workshop on Multimedia Signal Processing*, 9-11 December, 424-427.

Tseng, B. L., Lin, C.-Y. and Smith, J. R. (2004a). Using MPEG-7 and MPEG-21 for personalizing video. *IEEE Multimedia 11*(1), 42-53.

Tseng, B. L., Lin, C.-Y. and Smith, J. R. (2004b). Video personalization and summarization system for usage environment. *Journal of Visual Communication and Image Recognition 15*, 370–392.

Tusch, R., Kosch, H. and Böszörmenyi, L. (2000). VIDEX: an integrated generic video indexing approach. *Proceedings of ACM Multimedia '00*, Los Angeles, CA, 30 October - 3 November, 448-451.

Vakali, A., Hacid, M.-S. and Elmagarmid, A. (2004). MPEG-7 based description schemes for multi-level video content classification. *Image and Vision Computing 22*, 367-378.

van Meteren, R. and van Someren, M. (2000). Using content-based filtering for recommendation. *Proceedings of the Machine Learning in the New Information Age: MLnet / ECML2000 Workshop*, Barcelona, Spain, 30 May.

van Setten, M. and Oltmans, E. (2001). Demonstration of a Distributed MPEG-7 Video Search and Retrieval Application in the Educational Domain. *Proceedings of ACM Multimedia '01*, Ottawa, Canada, September 30-October 5, 595-596.

Vendrig, J. W., M. (2002). Interactive adaptive movie annotation. *Proceedings of the 2002 IEEE International Conference on Multimedia and Expo*, Vol. 1, 93-96.

Vetro, A. (2004). MPEG-21 digital item adaptation: Enabling universal multimedia access. *IEEE Multimedia 11*(1), 84-87.

Vieira, M. T. P. and Santos, M. T. P. (1997). Content-based search on an MHEG-5 standard-based multimedia database. *Proceedings of the Eighth International Workshop on Database and Expert Systems Applications*, Toulouse, France, 1-2 September, 154-159.

Wallace, M. S., G. (2002). Towards a context aware mining of user interests for consumption of multimedia documents. *Proceedings of the 2002 IEEE International Conference on Multimedia and Expo*, Vol. 1, 733-736.

Wenyin, L., Chen, Z., Lin, F., Zhang, H. and Ma, W.-Y. (2003). Ubiquitous media agents: a framework for managing personally accumulated multimedia files. *Multimedia Systems 9*(2), 144 - 156.

Westermann, U. and Klas, W. (2003). An analysis of XML database solutions for the management of MPEG-7 media descriptions. *ACM Computing Surveys 35*(4), 331-373.

Zhao, R. and Grosky, W. I. (2002). Bridging the semantic gap in image retrieval. In T. K. Shih (Ed.), *Distributed Multimedia Databases: Techniques and Applications*. (pp. 14-36). Hershey, PA: IDEA Group Publishing.

APPENDIX: COSMOS-7 Filtering Clauses

The **EVENT** clauses enable event-related filtering criteria to be specified on the COSMOS-7 representation. The following clauses are supported:

- *eventGroupingId* = <*eventGroupingId*>, which enables filtering according to a particular event grouping, as specified by the id parameter of the <Semantics> tag of the COSMOS-7 representation.

- *eventGroupingLabel* = <*eventGroupingLabel*>, which enables filtering according to a particular event grouping label, as specified by the corresponding <Label> tag of the COSMOS-7 representation.
- *eventId* = <*eventId*>, which enables filtering according to a particular event, as specified by the id parameter of the <SemanticBase> tag of the COSMOS-7 representation.
- *eventLabel* = <*eventLabel*>, which enables filtering according to a particular event label, as specified by the corresponding <Label> tag of the COSMOS-7 representation
- *mediaOccurrence* = <*mediaId*> | <*mediaTimePoint*> | <*mediaRange*>, which enables filtering according to a particular media segment. The <mediaId>, <mediaTimePoint> and <mediaRange> elements enable correlation to the information specified within the <MediaOccurrence> element of the COSMOS-7 event representation.
- *agent* = <*objectId*>, which enables filtering according to event agents and correlates to the <Relation> element within the COSMOS-7 event representation.

The **TMPREL** clauses enable temporal-relationship-related filtering criteria to be specified on the COSMOS-7 representation. The following clauses are supported:

- *temporalRelationshipType* = <*temporalRelationshipType*>
- *temporalRelationshipSource* = <*objectId*>
- *temporalRelationshipTarget* = <*objectId*>

These all correspond with the parameters of the <Relation> element within the temporal relationship graph of the COSMOS-7 representation. However, in the case of <temporalRelationshipType>, an abbreviated version is used that omits the “urn:mpeg:mpeg7:cs:TemporalRelationCS:2001:” prefix and just states the actual temporal relationship, e.g. “precedes”, since the prefix is redundant in this context.

The **OBJ** clauses enable filtering criteria to be specified on the COSMOS-7 representation that are related to objects and their properties, respectively. The following clauses are supported:

- *objectGrouping* = <*objectGroupingId*>
- *objectGroupingLabel* = <*objectGroupingLabel*>
- *objectId* = <*objectId*>
- *objectLabel* = <*objectLabel*>

These four clauses are used to specify the same criteria as those in the **EVENT** clauses but for object groupings and objects, rather than event groupings and events.

- *subObjectId* = <*subObjectId*>
- *subObjectLabel* = <*subObjectLabel*>

These clauses are the same as the objectId and objectLabel clauses but enabling filtering of objects that are sub-objects of another object specifically. This enables the filter to match only those objects that are sub-objects of another object and will reject objects that have the same label but are main objects.

- *mediaOccurrence* = <*mediaId*> | <*mediaTimePoint*> | <*mediaRange*>, which has the same functionality as that of the **EVENT** clause of the same name.
- *agentOf* = <*objectId*>, which has similar functionality to that of the agent **EVENT** clause.

The next five clauses enable filtering according to object property identifiers, labels, attributes, units and values, respectively:

- *objectProperty = <objectPropertyId>*
- *objectPropertyLabel = <objectPropertyLabel>*
- *objectPropertyAttribute = <objectPropertyAttribute>*
- *objectPropertyUnit = <objectPropertyUnit>*
- *objectPropertyValue = <objectPropertyValue>*

These final three clauses have the same functionality as the **TMPREL** clauses but deal with the object hierarchy relationships:

- *objectHierarchyType = <objectHierarchyType>*
- *objectHierarchySource = <objectId>*
- *objectHierarchyTarget = <objectId>*

Again, the “urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:” prefix is omitted due to its redundancy in this context.

The **SPLREL** clauses deal with spatial-relationship-related filtering criteria, via the following clauses:

- *video = <videoId>* | *<mediaTimePoint>* | *<mediaRange>*
- *videoSegment = <videoSegmentId>* | *<mediaTimePoint>* | *<mediaRange>*

The above two clauses have the same functionality as that of the mediaOccurrence **EVENT** and **OBJ** clause, however here they relate to the <Video> and <VideoSegment> sections of the COSMOS-7 representation.

- *spatialRelationshipType* = *<spatialRelationshipType>*
- *spatialRelationshipSource* = *<objectId>*
- *spatialRelationshipTarget* = *<objectId>*

The above three clauses have the same functionality as the **TMPREL** clauses but deal with spatial relationships. The “urn:mpeg:mpeg7:cs:SpatialRelationCS:2001:” prefix is omitted for the same reasons as previously given.

The final clause enables filtering according to moving region identifiers or the objects related to them in COSMOS-7:

- *movingRegion* = *<movingRegionId>* | *<objectId>*