CHAPTER 15

STRUCTURED MEDIA FOR AUTHORING MULTIMEDIA DOCUMENTS

Tien Tran-Thuong and Cécile Roisin

Opéra Project,
INRIA Rhône-Alpes, Zirst - 655 avenue de l'Europe - 38330 Montbonnot
Saint Martin, France
E-mail:tien.tran_thuong@inrialpes.fr, cecile.roisin@inrialpes.fr
URL: http://opera.inrialpes.fr

This chapter proposes a new way for authoring multimedia documents. It uses the concept of structured media that allows deeper access into media objects. The proposed models can be considered as a utilization of MPEG7 for intra-media content description and an extension of hierarchical structure, interval and region based model for inter-media composition. An experiment of structuring video in our authoring and presentation environment for multimedia documents, called *VideoMadeus*, that takes advantage of that model is described. It allows the author to interactively specify video content structure descriptions that can be then used for the composition of video elements (character, shot, scene, etc.) with other media objects (text, sound, image, video, etc.) Such a composition allows to easily realize attractive multimedia presentations where media content can be synchronized in rich and flexible ways such as: tracking an object in a video, attaching hyperlinks to video objects and fine-grained synchronization (for example a piece of text can be synchronized with a video fragment).

1. Motivation

The creation of a multimedia document requires the integration of several media (such as text, image, audio, video and animation) into a single document. A multimedia document model, therefore, integrates media description models together with temporal and spatial models.[1] The media description models allow to define, to locate, to describe and to group the media that will be used to compose a multimedia document. The temporal and spatial models enable the author to organize media objects in time and space. Numerous approaches have been proposed for multimedia document modelling, including the absolute time axis, the time point temporal model, the interval temporal model and the region model.[2] One of the most interesting results is the emergence of the SMIL standard[3] for posting multimedia information on the Web. However, existing media description models mainly serve to declare the set of used media with their intrinsic spatial and temporal properties. As a consequence, a user can only express coarse-grained relationships (both temporal and spatial relationships) between the different media. But it is worth noting that most media have rich content information such as image, video, long text or included documents such as HTML or SVG. Through using subparts of that content information, the author can compose multimedia documents having more complex and sophisticated presentation scenarios. Examples of such needs are: a character in a video introduced by displaying a textual description when that character appears; a word in a text sentence highlighted when an audio stream plays out this word; a hyperlink set on a video object or on a particular region of an image. These scenarios can easily be specified if the authoring system supplies to the authors internal media information, such as: a start time of the video object in the video sequence for the first scenario; coordinates of the word in the text and time location of word pronunciation in the audio for the second scenario; or coordinates of the video objects and the image regions for the last scenario. In SMIL for instance, it is possible to specify subparts of media in terms of their time position from the beginning of the media. But it is a rather low level and limited way of specification of media subparts. Such a low semantic level of specification prevents Web analysis and search engines from processing the content of multimedia document.

Using structured media whose information content is described at a higher level will make this content information available for the composition process, indexing and retrieval services. A structured media contains not only raw data, but also a hierarchical description of this media content information. Up until now, there have been many research efforts for a standard format of content information description. Among them, the most important is the MPEG-7 standard also known as "Multimedia Content Description Interface" that aims at providing standardized core technologies, allowing the description of audiovisual data content in multimedia environments.[4] Therefore, standard structured media for multimedia authoring is not a distant goal and constitutes the first step towards allowing the editing of more complex multimedia documents.

The work presented here is carried out in the context of the development of a multimedia document authoring system called Madeus.[1] This prototype allows multimedia documents to be composed from a set of text, images, audio, video, HTML and SVG media. The Madeus document model is based on the structured, temporal interval-based and region-based models. In the first stage, we tested the structured media approach with just video media.

Based on this experiment, the discussion in this chapter is devoted to the use of structured media to edit complex multimedia documents. The rest of the paper is organized as follows: first, an overview of multimedia authoring is discussed in Section 2. A proposed multimedia model including a video content description model and an extended multimedia document model is described in Section 3. Our experiment with the proposed multimedia model is then presented in Section 4. In Section 5, we give a brief evaluation of this work through its comparison on modelling and editing aspects with existing work. Finally, the current achievements of our work and some perspectives are given.

2. Multimedia Authoring

Multimedia applications cover a wide range of services such as: media production and analysis, media storage and retrieval using multimedia databases, media integration to produce multimedia presentations, and finally multimedia broadcasting or multimedia on demand. In fact most multimedia applications require combining some of these functions in an efficient manner. For instance, media classification and indexing can benefit from media analysis as in the MAVIS system.[5] Similarly, media production and integration can use media databases for retrieving and reusing existing media. Fine-grained descriptions of media can bring valuable services for the authors who want to realize multimedia presentations with rich synchronizations between media.[6,7] The key requirement for allowing the integration of these services is the use of common media models. However most commercial tools are black boxes with proprietary and low level output formats. Standard groups have however worked to propose languages and modelling tools, such as the Mpeg groups with the MPEG4 and MPEG7 formats, or the W3C with the SMIL integration language. But it is worth noting that these standards are not yet widely used in multimedia applications.

The work presented here focuses on multimedia authoring applications and more precisely on what models and services are needed to cover all user requirements. The target application is an ideal authoring and presentation system, for which we have identified the following features:

1. Media access and browsing from a multimedia database using multiple media properties (temporal, spatial, and semantic).
2. Media analysis, content descriptors generation and media indexing.
3. Spatial, temporal or combined spatio-temporal media segments identification and access.

4. Media integration, synchronization, linking and structuring for the production of multimedia presentations.

First, let us consider how existing applications meet these requirements. Multimedia applications can be divided into three main classes: indexing, media production and multimedia document integration.

- Indexing applications such as *QBIC*, *VisualSeek* or *CueVideo* provide good solutions for the three first criteria above. Therefore authoring applications can exploit the rich information extracted from raw data to define links and synchronizations. Moreover, generic links and synchronization can be set over the media. The main limitation of these applications regarding authoring needs is their mono media approach. Each one is indeed specialized for one type of media so the last criterion is not covered at all.
- Media production applications such as *GoLive*, *Adobe Premiere*, *MediaStudio Pro*, *VideoStudio*, *DVD Movie Factory* allow to capture, split, mix, code, translate, export or render media streams. They provide analysis capabilities but at a low level only: for instance shots in video can be detected but not the semantic units such as scenes or sequences. Their media integration capabilities are limited to a flat and linear composition (the aggregation of a set of clips, for instance) and a static rendering of the resulting presentation. Therefore, the most interesting authoring features of media production applications are their abilities in editing media content, for instance to add animation effects.
- Multimedia document applications provide authoring features to integrate several types of media inside the same presentation. Examples of such authoring systems are *RealSlideShow*, *PowerPoint*, *GRiNS*, *Director* and *Madeus*. The capabilities of each tool substantially depend on their underlying media integration model. For instance *Director* has an absolute time-based approach with no hierarchical composition while *RealSlideShow*, *GRiNS* and *LimSee* implement (part of) the hierarchical and operator-based SMIL language. They are powerful to define rich temporal scenarios with interactivity and transitions but they offer few facilities for efficient and fine-grained media access, intra-media authoring and fine-grained extra-media synchronization.

We can summarize the major points of this analysis in the diagram of Fig. 1, where these three classes of applications are positioned on two axes: (1) the vertical axis measures their ability to provide high level analysis and content interpretation and (2) the horizontal axis identifies their media integration capability. Clearly, indexing applications are close to the vertical axis because they cannot easily integrate media but they can produce high level descriptors; on the other hand, multimedia authoring applications are close to the horizontal axis because they are poor in media content management but are able to richly compose scenarios; finally, media production applications are set in the bottom left corner with a low or medium rate of both criteria. The ideal multimedia application we promote is located on the top right part of the drawing.

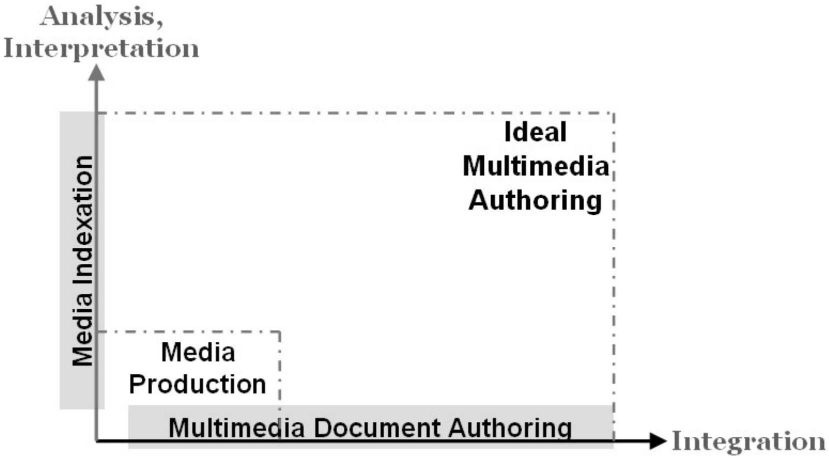


Fig. 1. Situation of multimedia applications.

Our contribution to the emergence of an authoring and presentation system that better covers user needs is based on two activities:

- A modelling activity that will allow a better integration of the different components of the system: indexing, integration and production. For that purpose we need media content models which must be consistent with the media integration model. Section 3 below is devoted to these models.
- An application architecture definition activity with its implementation in a prototype authoring tool. The application that we have developed is described in Section 4. This approach is based on the use of the media models presented in Section 3. Fig. 2 below presents the general architecture of our system.

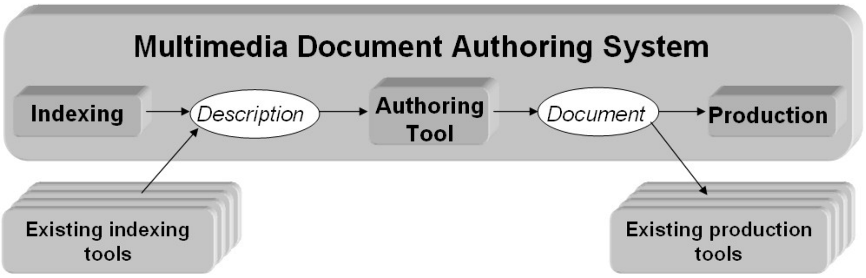


Fig. 2. Proposed architecture of a user-friendly multimedia authoring system.

3. Multimedia Modelling

The multimedia modelling mentioned in this section has two levels: the multimedia content modelling and the multimedia integration modelling. The first one can represent the media content as a semantic structure that allows more efficiency in accessing the multimedia content. The second one is aimed at authoring multimedia documents at the higher application level. Unfortunately, the gap between these components marks the limits in the existing multimedia systems, i.e., the user of an authoring system cannot deeply access the media content for a fine-grained synchronization or sophisticated composition; and the multimedia query system cannot return as a result of research a multimedia presentation instead of the individual media. This section first presents the work of modelling the video content and then presents a multimedia document model including a basic model called Madeus and its extension for using the multimedia content descriptions. We have chosen to present video content modelling because video carries rich and high-capacity information. Describing the video content allows the internal video components to be accessed. This is the key point in building more dynamic and interactive presentations in which video entities can be more finely synchronized with other media.

3.1. *Video Content Modelling*

The choices made in our video model are motivated principally by the requirements of the first application we want to realize: we want to compose space and time fragments of video (the appearance of a character (actor), the beginning of a scene or shot) together with other media objects (text, picture, sound ...) in a multimedia document. Therefore, the definition of the video structure must specify the video decomposition in terms of elements together with space and time relations among these video elements.

We chose the mark-up language XML (eXtensible Markup Language) because it is a language suited to describing structured information and its properties. Moreover, XML allows for easy integration of video structures in multimedia systems as it has already been used as a description language for multimedia documents such as in SMIL and Madeus (on which our prototype has been built). Even the Expert Group that specifies MPEG-4 envisions the use of XML for MPEG-4 descriptions.[8]

3.1.1. General Model

Our model is based on a decomposition of video information into three principal parts: Structure, Semantic and Thesaurus, which define three semantic levels of the knowledge base (Fig. 3). In[9] we can find a similar organization of the knowledge base for an intelligent multimedia system.

- The *Structure* description is a low-level description that directly indexes raw video to extract the structure of the video. It is the most important part in our model. It makes it possible to describe the content of the video directly and completely.
- The *Semantic* description elements allow the description of the video contents at a higher semantic level (the characters, the events, the relations, etc).
- The *Thesaurus* description elements constitute the highest level description, which describes semantic terms and expressions to classify elements in the video content description. These terms can be located in a thesaurus or defined by the author.
- Other element descriptions: *MetaInfo*, *MediaInfo*, and *Summary* that make it possible for the author to easily identify video and to look at it quickly and throroughly. These last three elements issued from the Dublin core project[10] and from MPEG-7.[4]

In Fig. 3 we can find the general description of the video content. Because our objective is to finely synchronize structural components of video inside documents, the rest of the section is only devoted to the structure level of the video decomposition.

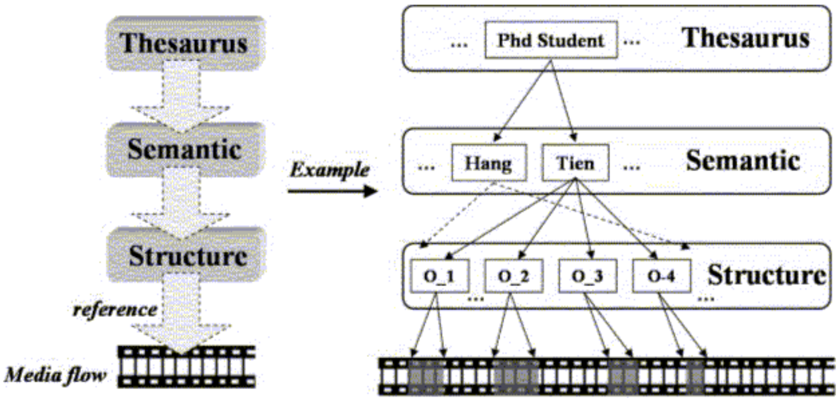

Thesaurus
Semantic
Structure
reference
Media flow
Example
... Phd Student ... Thesaurus
... Hang Tien ... Semantic
O_1 O_2 O_3 O-4 Structure
...
...

Fig. 3. General model and an example of the video content description.

3.1.2. Description of the Video Structure

At the highest level, we have defined a model of the video structure that is very similar to the classic dramatic structure of the video that can be found in a number of existing works[11, 12, 13, 14]: a video is composed of successive sequences, a sequence contains successive scenes and a scene contains successive shots (Fig. 4a). All these elements (sequence, scene and shot) can be separated by transitions. Our model especially focuses on the specification of these temporal elements and relations between them.

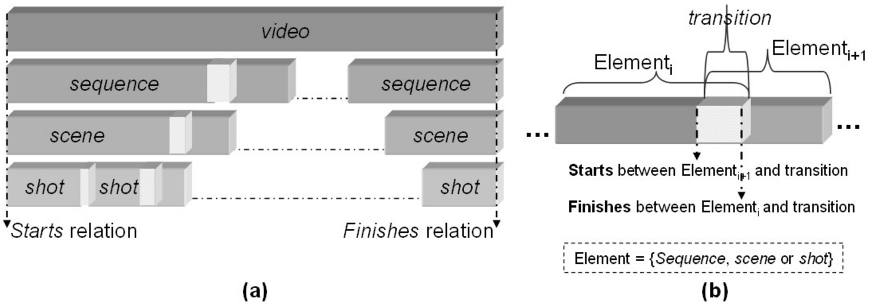

video
sequence
sequence
scene
scene
shot
shot
shot
Starts relation
Finishes relation
transition
Element$_i$
Element$_{i+1}$
...
...
Starts between Element$_{i+1}$ and transition
Finishes between Element$_i$ and transition
Element = {*Sequence*, *scene* or *shot*}
(a)
(b)

Fig. 4. Hierarchical and relational structures at high level.

These elements are organized in the hierarchical structure as follows: a *video* element contains one or more *sequence* elements; a sequence is composed of *scene* elements; and a scene includes *shot* elements. The temporal relations among these elements are defined as follows:

- A parallel *starts* relation is set between the first shot of a scene and that scene (i.e. its parent in the structure), between the first scene of a sequence and that sequence, between the first sequence and the video. (see Fig. 4a). These relations can be expressed by the following relations between *begin* instants:

$$\mathbf{Begin}_{shot} = \mathbf{Begin}_{scene}\ ;\ \mathbf{Begin}_{scene} = \mathbf{Begin}_{sequence}\ ;\ \mathbf{Begin}_{sequence} = \mathbf{Begin}_{video}$$

- A parallel *finishes* relation is set between the last shot of a scene and that scene, between the last scene of a sequence and that sequence, between the last sequence and the video. (see Fig. 4a). These relations can be expressed by the following relations between *end* instants:

$$\mathbf{End}_{shot} = \mathbf{End}_{scene}\ ;\ \mathbf{End}_{scene} = \mathbf{End}_{sequence}\ ;\ \mathbf{End}_{sequence} = \mathbf{End}_{video}$$

- The transition *T* from an element (*sequence*, *scene* or *shot*) to the following element is modeled by two relations co-starting up (*starts*) and co-ending (*finishes*) as following: the element before E_{before} has the relation *finishes* with the transition *T*; the element following E_{after} has the relation *starts* with the transition *T* (see Fig. 4b) :

$$\mathbf{End}_{Ebefore} = \mathbf{Begin}_{T}\ ;\ \mathbf{End}_{T} = \mathbf{Begin}_{Eafter}$$

The description of the former basic components allows the synchronization of coarse elements (shots, scenes or sequences) of video in multimedia documents. It can be useful to allow a more fine-grained synchronization and for that purpose it is necessary to be able to describe the content of shots such as the occurrences of characters (actors) or objects, the spatio-temporal relations between these occurrences, etc. We have defined three types of components in shots: 1) the *segment* element describes a particular situation in the video shot that is considered as an event such as a motor explosion, a plane taking off, a demonstration, a storm, etc.; 2) the *occurrence* element describes a person or an object that appears in the shot; 3) the *Spatio-TemporalLayout* element describes the spatio-temporal relationships among the composing regions of occurrences in the shot.

These three types of elements are components of shots in the hierarchical structure of the video (see Fig. 5a). They can appear at any time in the shot in which they are included and can be referred by the semantic elements such as *event*, *object* or *person*. Therefore, we use the *during* relation between these elements and that shot (see Fig. 5b) whose corresponding instant relations are:

$$\mathbf{Begin}_{element} = \mathbf{Begin}_{shot} + \mathbf{d1}\ (\mathbf{d1} = 0)\ ;\ \mathbf{End}_{shot} = \mathbf{End}_{element} + \mathbf{d2}\ (\mathbf{d2} = 0)$$

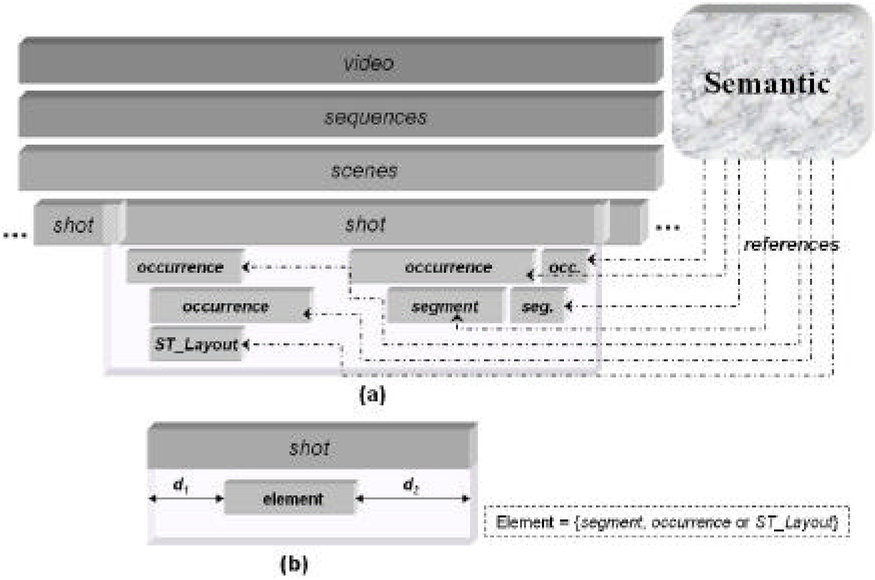


Fig. 5. Hierarchical and relational structures of the elements in a shot.

The description of the occurrences enables us to associate actions with the appearances of video objects like hyperlink, filter, seek, follow, synchronize, etc. In our model, an occurrence description is composed of (see Fig. 6) *visual features* of the occurrence such as colour layout, colour histogram, texture, shape and contour. 2) *spatio-temporal* locators 3) and finally, *sub-occurrences* inside the occurrence, for instance, the arms of a character, his clothing, etc.[15]

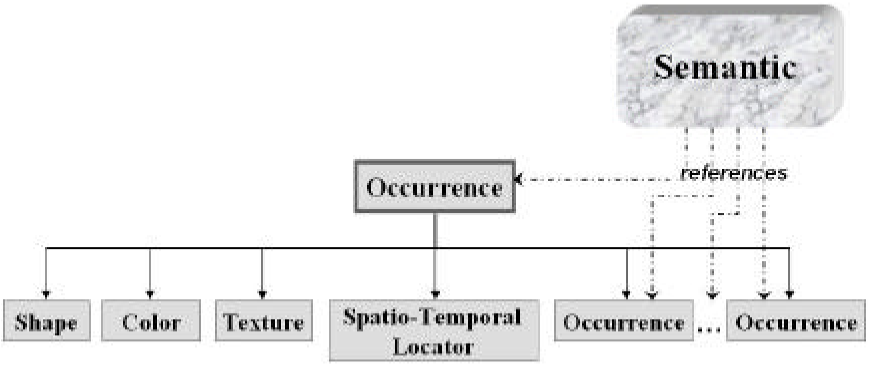


Fig. 6. Occurrence structure.

The Spatio-temporal layout defines the spatial relations among characters or things that appear at the same time in a shot, such as *A* stays behind *B*, *A* walks on the left side of *B*, etc. Note that due to the intrinsic dynamic behaviour of the video, these spatial relations can change with time. For instance, in a video shot, there is a *Taunus* car that follows a *Volvo*; when the *Taunus* approaches the *Volvo*, the *Taunus* overtakes the *Volvo* on the right side and then goes past the *Volvo*. To describe these changes, we have to define many periods in the shot corresponding to the different spatial relations among the occurrences. In the former example, we can separate the spatial relations between two cars into three sequential periods corresponding to *Taunus* behind *Volvo*, *Taunus* on the right of Volvo and *Taunus* before *Volvo* (see Fig. 7).

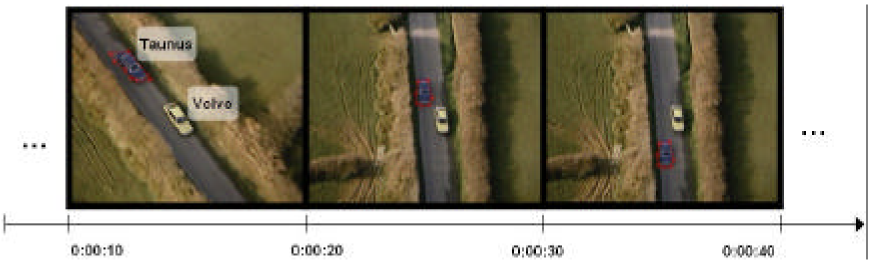


Fig. 7. Example of the spatio-temporal disposition of two cars in a video shot.

3.1.3. Extensions of MPEG-7 for the Definition of Our Model

MPEG-7 takes into account existing models to supply standard tools for multimedia content modelling: a Description Definition Language (DDL) to define sets of Descriptors (D) and Description Schemes (DS). We have opted to use these tools to describe our model. Because of that, our model is convenient for a wide range of applications and can use and adapt existing descriptions. MPEG-7 provides rich tools that can be directly used to describe information such as the metadata (*DescriptionMetadata DS*), the management of content *(UserDescription DS, CreationInformation DS, etc*), the semantics of contents (*WorldDescription DS*), the thesaurus (*ClassificationScheme DS*), the summary of the content (*SummaryDescription DS*) and even the occurrences and the relations among them through *MovingRegion DS* and *Relation DS*. Nevertheless these tools are very generic, and, therefore, it is necessary to extend them to cover the particular needs of multimedia document authoring and presentation.

In fact, MPEG-7 supplies an element root <*mpeg7:Mpeg7*> which is an extension of the complex type <*mpeg7:Mpeg7Type*> to describe either a complete multimedia element, or an information fragment extracted from a piece of media content.[4] Both cases are not convenient for our needs, because a complex description is too big to insert it into a document and, on the other hand, a unit description is too simple: it cannot thus supply enough information for editing. That is why we decided to create our element root <*MediaDescription*>. However, to remain compatible with MPEG-7 descriptions, our element root is an extension of the <*mpeg7:Mpeg7Type*> type.

The standard MPEG-7 supplies the video segment description scheme (*VideoSegment DS*) to describe the structure of video contents in time and space. However, the *VideoSegment DS* is more relevant in describing a generic video segment that can correspond to an arbitrary sequence of frames, a single frame, or even the full video sequence.[4] It does not convey the specific signification of each of the video structure levels such as the sequence, scene and shot. Therefore, we have defined three new types: *VideoSequence DS*, *VideoScene DS* and *VideoShot DS*, which inherit from the MPEG-7 *Videosegment DS* and extend it to express the specific video structure of our model (cf. Section 3.1.3).

Additionally, the *Videosegment DS* supplies the description of metadata and management. That is not needed for our model, because each *Videosegment DS* instance aims to describe the structure of only one video for which management description and metadata can be described only once at the top level of the description (see Fig. 8).

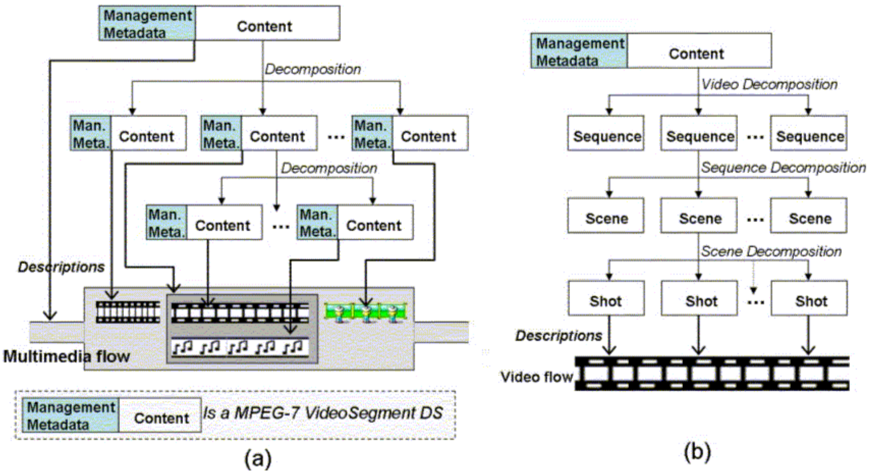


Fig. 8. Differences between (a) the MPEG-7 description model for a piece of multimedia content and (b) our description model for a structured video content.

3.2. *Document Modelling with Structured Medial*

We present in this section the basic multimedia document model and its extension to allow the inclusion of the media content description model presented above.

3.2.1. Multimedia Document Model

A multimedia document model has to realize the integration of a set of media elements through temporal, spatial and hyperlink models. Previous work on electronic documents[16,17] has stated that the use of a structure, interval and region-based model enables powerful document representation and management. SMIL,[3] the standard for bringing multimedia to the Web, ZYX[2] a powerful model for expressing adaptable multimedia presentations and Madeus,[1] our flexible and concise model are the typical models that follow the hierarchical structure of intervals and regions.

Following this decomposition approach, our Madeus model can be considered as an extension of the SMIL standard with the following additional features: 1) enhanced separation of media content location, temporal information and spatial information, 2) hierarchical, operator-based temporal model complemented with relations, 3) rich spatial specification model with relative placements. More precisely, a Madeus specification has four main parts (see Fig. 9).

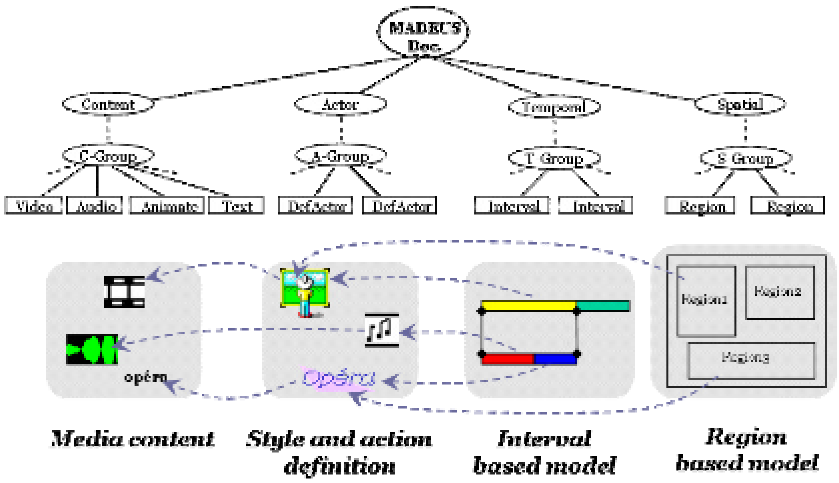


Fig. 9. Madeus document model.

The *Content* part allows the definition of a set of hierarchical fragments of the media contents that will be used to compose a multimedia document. It can be compared with the *Content* class of the MHEG[18] standard that allows the media content to be defined independently of its presentation. So the content can be reused several times for different presentations attributes.

The *Actor* part allows presentation styles and interactions on the content data such as *FillColor*, *FontSize* or *Hyperlink* to be specified through the element called *DefActor*. It can be compared with the *virtual views* concept of MHEG that allows media content to be projected onto a concrete utilisation; or the object modification of HyTime that allows media to be provided with the new presentation attributes. A *DefActor* element has to refer to a media content (see Fig. 9).

The *Temporal* part concerns the temporal presentation structure of documents. This is a hierarchical structure augmented with temporal relations set on intervals. An interval refers to one or several *DefActor* elements for presenting the associated media objects over the specified time. Each *interval* possesses the following timing attributes: *begin*, *duration* and *end* (with the constraint, *end* = *begin* + *duration*). A set of intervals can be grouped into a composite interval called *T-Group* and associated to a temporal operator (in sequence or parallel).

Similarly, the *Spatial* structure defines the spatial layout of documents by means of a hierarchical structure and relations over boxes called *Regions*. A region refers to one or more *DefActor* elements for presenting the associated media objects in space. The set of spatial relations available such as *left_align*, *center_align*, etc. provides relative layouts among *Region*s that are much more flexible and more comfortable than the absolute spatial layout such as in SMIL model.

Although the interval and region-based model is known to be one of the most expressive among existing models[19], the limit of this approach is mainly due to the granularity provided by the leaves of the structure. In fact, there are many media objects having rich content information such as image, video or long text for which authors want to set finer-grained synchronizations in order to produce more sophisticated presentation scenarios. The problem cannot be solved by simply using the existing model and defining deeper hierarchical structures as found in existing models with the *Anchor* and the *Area* elements. Such a solution is only a limited solution with the drawbacks of an absolute and non-significant specification. Indeed, media objects do have their own semantics, temporal and spatial organization, which the document model must consider when composing media fragments during document composition. This is why we propose extensions in the next section.

3.2.2. Model Extensions

Since our document model has to be consistent with the video content model in order to share the same representation in the different steps of our multimedia document authoring. More precisely, it is necessary to extend the components of the Madeus model to use the video content description model (and other media content models).

Thanks to the hierarchical structure-based model of Madeus we have introduced new hierarchical structures to the Madeus document model called *sub-Elements* (see Fig. 10). The extensions are done in each decomposition axis of the Madeus model (*Content*, *Actor*, *Temporal* and *Spatial*). For each axis the extension provides a specific *sub-Element* and defines precisely the constraints imposed by the element in which it is included. Therefore, the distinction between *Elements* (*DefActor, Interval, Region*) and *sub-Elements* is clearly stated.

1. The *Content* part of Madeus has been extended with new media types for structured media comprising *StructuredVideo* (specified in section 3.1), *StructuredAudio*, *StructuredText*. These new types introduce the internal structural level for the media, which was not available with the classic

media types that only represented raw data to play. They provide ease and meaningfulness while integrating the media fragments.

2. In authoring a multimedia document, the author needs to specify actions or styles on **media fragments** such as a *highlight* on a phrase or a word of a text, a *tracking* or *hyperlink* on a moving region of a video segment. A sub-element of the *DefActor* element called *subDefActor* is then provided for these purposes. It uses a *Content* attribute valued with IDs or *XPath* expression to refer to the media segments on which the action or style must be applied. The segments referred to must belong to the structured description of the media element.
3. Sub-temporal objects are necessary to carry out the *subDefActor* objects or/and the *temporal representation* of the media segment. A *subInterval* element is defined **inside** an *interval* element for that purpose. The *subInterval* element is derived from the *interval* element in our interval-based model. Therefore, as any temporal object, the sub-interval can be involved in any temporal relation of the temporal document specification. The refinement of the *subInterval* through inheritance is that the *subInterval* element has a **during** temporal constraint with its parent *interval*. The *subInterval* carries the *subActor* attribute to specify the *subDefActor* elements referring to the media fragments. The media segments can be static, such as a phrase in text media or a region of an image; in that case the time specification for static fragments must be explicit. If the *subDefActor* element refers to a temporal segment belonging to continuous media, such as an audio segment or a video segment, then the *subInterval* will be automatically scheduled thanks to the temporal information of the segment description. This *subInterval* element makes explicit a temporal fragment of media presentation for further synchronizations with others *interval/subInterval*. The key point of this model is to maintain the intrinsic time constraints (**during**) of the *subIntervals* inside their media content *interval* together. That allows temporal segments of media to be integrated into the timed schedule of the whole document.
4. In the spatial part, the *subRegion* element plays a similar role as the *subInterval* for representing a spatial segment of visual media objects. Together with its intrinsic position and dimensions, the identification of *subRegion* provides the means to specify more sophisticated spatial relations with other regions. For instance, the spatio-temporal synchronization of that region, e.g., the text bag is set on the top of a character’s occurrence by the Top-Align relation. If the character’s occurrence is a moving region, the Top-Align constraint will result in

moving the speak bullet following the movement of the occurrence in the video. The other applications of the *subRegion* element are interactions on sub areas of visual media objects such as hyperlink, tracking or displaying tip text for the area.

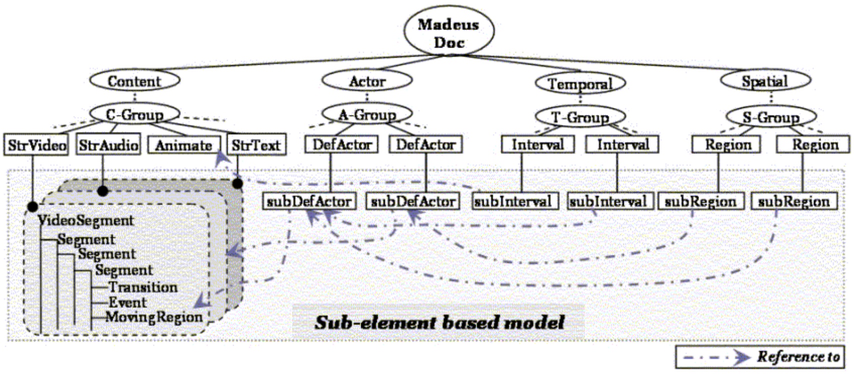


Fig. 10. A Madeus document structure with content description, *subDefActor*, *subInterval* and *subRegion* sub-elements.

The figure above summarizes the definitions of *sub-elements* and their relations. In conclusion, a *sub-element* always belongs to an element and relates to that element to express its semantic dependency in the corresponding dimension. Note that except for the content part, *sub-elements* are not recursive.

4. Multimedia Document Authoring System

This section presents an advanced environment for playing and editing multimedia documents called *VideoMadeus*. While existing tools such as *GRiNS* from *Oratrix* or *X-Smiles* are based on the SMIL standard model, ours uses the extended Madeus framework presented in the previous section, in which the internal structure of complex media such as video can be edited to be used inside spatial and temporal synchronizations in the document.

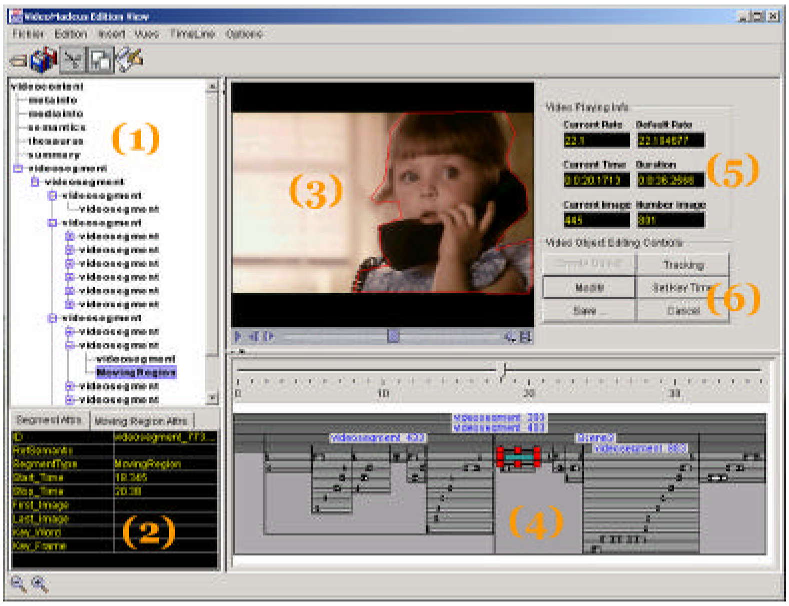
One of its main features is media fragment integration. It uses several views to display video and audio contents (see Fig. 11). These views allow the user to semi-automatically generate a media content description based on the MPEG-7 standard. This description is then used for specifying fine-grained synchronization between media objects. Using media content description in authoring multimedia documents brings such advantages as: 1) tracking an object in a video (a video object for short), 2) attaching hyperlinks to video objects (video objects are moving regions), 3) fine-grained synchronization (for example a piece of text can be synchronized with a video segment like a scene, a shot or an event), 4) spatio-temporal synchronization: a text can follow a video object, 5) extracting any part of a video/audio (even a video object) for integration with other media.

In addition, *VideoMadeus* provides a timeline view that is much more powerful than the usual flat timeline. Ours is hierarchical and supports editing of many temporal relations (meet, before, finish, during, equal, etc.). This is especially interesting in structuring the video and audio media. It allows an author to easily locate the different parts of the media and to create temporal relations between media objects and fragments of the video/audio content.

The end of the section briefly presents the video content description editing tool and the authoring of a multimedia document with a video segment in which a video object is synchronized with a text and a hyperlink is set from a moving sub-region of that video.

4.1. *Video Content Description Editing Environment*

In our system, the video content editing environment (see Fig. 12) enables information within the video medium, such as time and spatial internal structures, to be semi-automatically extracted. The interface presents the resulting video content description through several views: the hierarchical structure view (1), the attribute view (2), the video presentation view (3) and the timeline view (4). That provides a simple way for the visualization, the navigation and the modification of the video content description. More concretely, if the author wants to add a video (in the mpeg, avi or mov format) in his document, he simply selects it and the system automatically extracts its basic structure (using a "standard" shot detection algorithm). This first structure is then displayed in the video structure and the timeline views of the video content editing environment. Next, the author can adjust and add semantic media content descriptors (such as scene and sequence decomposition, character objects or spatial/personal relation) which currently cannot be automatically generated by existing content analyzers. For that purpose, some authoring functions are provided: grouping/ungrouping shots, scenes or sequences using the structure view or the timeline view, graphically selecting spatial areas containing objects or characters, attaching key positions and movement functions to these objects using the video presentation view and the attribute panels.



ID	videosegment_773...
RefSemantic	
SegmentType	MovingRegion
Start_Time	18.365
Stop_Time	20.38
First_Image	
Last_Image	
Key_Word	
Key_Frame	

Current Rate	Default Rate
22.1	22.104877
Current Time	**Duration**
0:0:20.1713	0:0:36.2598
Current Image	**Number Image**
445	801

Fig. 11. Madeus video content description editing views: (1) Video structure view. (2) Attribute view. (3) Video presentation view. (4) Video timeline structure view. (5) Video information view. And (6) Video object editing control.

In summary, the media content editing views help the user to create and modify structured media. This environment is similar to the IBM MPEG-7 Visual Annotation Tool[20], which is used for authoring audiovisual information descriptions based on the MPEG-7 Standard Multimedia Description Schemes (MDS). However, our tool is more focused on the structure description of content (we don’t yet propose enhanced features for authoring semantic level descriptions) but it allows the integration of automatic media analyzers and generators.

4.2. Authoring Multimedia Documents

The video content editing environment presented above has strong relations with other parts of the Madeus system allowing the use of video description information when composing Madeus documents. Users of Madeus can synchronize video elements of a video media with other media objects in both time and space. For instance, in the document displayed in Fig. 12, the video object “Little girl phones” of a video segment displayed in Fig. 11 has been synchronized with a text media (see the timeline document view). Authors can also apply operations and interactions on elements of the video such as tracking, hyperlink, hiding or even deletion. Thus, complex multimedia documents can be specified while maintaining the declarative approach of XML that allows the use of high-level authoring interfaces like our video content editing system.

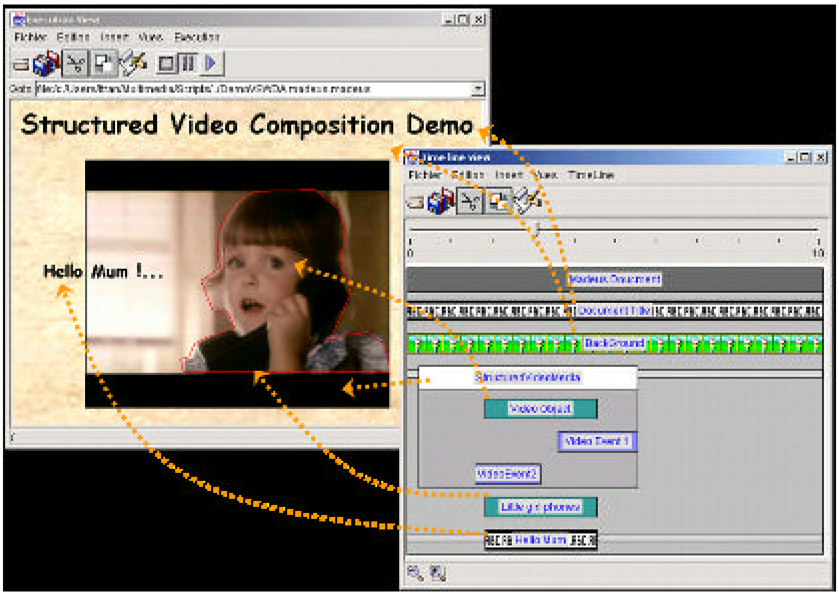


Fig. 12. The Execution and Timeline views of a Madeus document (the text media “Hello Mum” has the *equals* relationship with the video fragment “Little girl phones”).

5. Conclusion

Our proposition provides support for a deep access into media content in multimedia document-authoring environments, which until now have treated media content as a black box. In addition, our experimental work with video, audio and text media has provided a way to implement such a system. It should be noted that the media content description model is adapted to the composition and rendering of multimedia documents, so it makes little use of metadata descriptions defined in MPEG-7 applications mostly devoted for searching, indexing or archiving media content. Indeed, this model is focused on the structural organization of media content that is relevant to multimedia document composition. As a positive result of this first experiment, we can edit documents that contain fine-grained synchronizations (in the temporal, spatial and spatio-temporal dimensions) between basic media (text, image, audio and so on) and video elements such as scene, shot, event, video object. This result has encouraged us to continue to structure other media. As a next step, we will investigate the same approach for handling audio and text media that will allow to compose complex documents such as Karaoke document type, with which a user can sing a song where every piece of text is synchronously displayed while the associated music stream is played.

Another positive result of using description models in multimedia documents is the possibility to apply indexing and searching techniques to the whole resulting presentations. The use of SMIL technology combined with enriched media content descriptions such as proposed here will certainly permit the emergence of real multimedia documents on the Web. Indeed, these new multimedia Web documents integrate multimedia content that is no more considered as a black box such as MPEG-1/2 videos, gif images or even Flash media. Therefore Web applications will be able to fully process all the Web content.

References

1. L. Villard, C. Roisin and N. Layaïda, "A XML-based multimedia document processing model for content adaptation", *Proceedings of Digital Documents and Electronic Publishing (DDEP00),* September 2000.
2. S. Boll and W. Klas. "-ZYX- A Semantic Model for Multimedia Documents and Presentations". *Proceedings of the 8th IFIP Conference on Data Semantics*, January 1999.
3. *SMIL: Synchronized Multimedia Integration Language*, W3C Recommendation http://www.w3.org/AudioVideo/.
4. P. Beek, A. B. Benitez, J. Heuer, J. Martinez, P. Salembier, Y. Shibata, J. R. Smith and T. Walker, *Text of 15938-5 FCD Information Technology – Multimedia Content Description Interface – Part 5 Multimedia Description Schemes, ISO/IEC JTC 1/SC 29/WG 11/N3966*, Singapore, March 2001.
5. P. H. Lewis, H. C. Davis, S. R. Griffiths, W. Hall and R. J. Wilkins, "Media-based Navigation with Generic Links", *Proceedings of Hypertext96*, Washington DC, 1996.
6. L. Rutledge and P. Schmitz, "Improving Media Fragment Integration In Emerging Web Formats", *Proceedings of Multimedia Modeling Conference*, Amsterdam, 5-7 November 2001.
7. T. Tran-Thuong and C. Roisin, "A Multimedia Model Based on Structured Media and Sub-elements for Complex Multimedia Authoring and Presentation, Special

Issue on "Image and Video Coding and Indexing”, *International Journal of Software Engineering and Knowledge Engineering*, 2002.

8. M. Kim, S. Wood, L.T. Cheok, Extensible MPEG-4 textual format (XMT), ACM Press, Pages: 71 - 74 Series-Proceeding-Article, New York, NY, USA, 2000.
9. M. Bordegoni, et al, “A Standard Reference Model for intelligent Multimedia Presentation Systems”, Computer Standards & Interfaces, 18(6-7):477–496, December 1997.
10. *Dublin Core Metadata Element Set*, http://purl.oclc.org/dc/documents/recdces-19990702.htm.
11. M. Jacopo, D. Alberto, D. Lucarella and H. Wenxue, “Multiperspective Navigation of Movies”, *Journal of Visual Languages and Computing*, **7**(1996), pp. 445-466.
12. R. Hammoud, L. Chen and D. Fontaine, “An Extensible Spatial-Temporal Model for Semantic Video Segmentation”, *Proceedings of the First International Forum on Multimedia and Image Processing*, Anchorage, Alaska, 10-14 May 1998.
13. J. Hunter, “A Proposal for an MPEG-7 Description Definition Language”, MPEG-7 AHG Test and Evaluation Meeting, Lancaster, 15-19 February 1999.
14. M. Dumas, R. Lozano, M.-C. Fauvet, H. Martin and P.-C. Scholl, “Orthogonally modeling video structuration and annotation: exploiting the concept of granularity”, *Proceedings of the AAAI-2000 Workshop on Spatial and Temporal Granularity*, Austin, Texas, July 2000.
15. S. Paek, A.B. Benitez and S.K. Chang, *Self-Describing Schemes for Interoperable MPEG-7 Multimedia Content Descriptions*, Image & Advanced TV Lab, Department of Electrical Engineering, Columbia University, USA, 1999.
16. J. André, R. Furuta and V. Quint, *Structured documents*, Cambridge University Press, Cambridge, 1989.
17. G. van Rossum, J. Jansen, K. Mullender and D. Bulterman, “CMIFed: a presentation Environment for Portable Hypermedia Documents”, *Proceedings of the ACM Multimedia Conference*, California, 1993.
18. T. Meyer-Boudnik and W. Effelsberg, “MHEG Explained”, IEEE Multimedia Magazine, Volume 2, Number 1, p.p. 26-38, 1995.
19. T. Wahl and K. Rothermel, “Representing Time in Multimedia-Systems”, *Proceedings of IEEE Conference on Multimedia Computing and Systems*, May 1994.
20. B. Lugeon and J. R. Smith, *MPEG-7 Visual Authoring Tool*, IBM T. J. Watson Research Center, http://www.alphaworks.ibm.com/tech/mpeg-7.