AN ABSTRACT ANIMATION MODEL FOR INTEGRATING SMIL BASIC ANIMATION ELEMENTS WITH MULTIMEDIA DOCUMENTS

Tien TRAN THUONG and Cécile ROISIN

Opéra Project, INRIA Rhône-Alpes, France
(tien.tran_thuong@inrialpes.fr, cecile.roisin@inrialpes.fr)

ABSTRACT

Declarative definition of multimedia presentation such as provided by SMIL standard can be considered as the most significant advance in the multimedia integration domain. The work presented here proposes an abstract way to use animations in this domain. That provides flexibility, no redundancy, easiness to maintain and reuse of animations in authoring and presenting multimedia documents. The model defines abstractions of SMIL basic animation elements and then combines them with the intra-media temporal structuration to specify animation effects. The underlying model is Madeus multimedia model that is a flexible model based on the hierarchical structure, interval, region and relative constraints. This model includes a *sub-element* definition that allows to use the *sub-interval* element to describe the intra-media temporal structuration. We have experimented with this model in an authoring tool where the author can edit animation scenarios by directly manipulating graphical animation elements through the time line view.

1. INTRODUCTION

Most multimedia presentation models (HyTime, MHEG, CMIF [5], ZYX [2], etc.) can define not only media content but also time, spatial and hyperlink relations between these media. However they lack the important need of multimedia specification for animating individual or group of media presentation. Notice that the animations we consider are not simply autonomous media objects such as an animating image. More generally an animation is modeled as a function changing the presented value of a specific attribute of a media object over time. It gives dynamic effects on both static and dynamic media such as text, image, audio and video.

Recent releases of the new standards as SVG and SMIL 2.0 have addressed this problem with the SMIL animation modules [6]. However, the implemented features of these SMIL animation modules (in the hosts languages: XHTML+SMIL, SVG, SMIL language) remain limited [4]. In this context, our model proposes a more abstract level of animation specification. It means that the animation has to be defined just one time, and then it can be applied to animate several media objects. In addition, the author can temporally schedule each concrete application of animation on a media by either absolute or relative way. For instance, in a Karaoke presentation, the animations on the color for each word of a song are started when the audio fragments corresponding to these words take place.

On the other hand, current multimedia models do not pay correct attention to describe the intra-media structuration that allows to express fine-grained constraints between media sub-elements (video character, video shot, video scene, audio segment, region on a picture, word of a text, etc.) or to define hyperlinks on these media's sub-elements. Early works CMIF [5], HyTime and SMIL have defined media fragment synchronization with the *anchor* element or the more improved *a* and *area* elements in SMIL2.0 for posting multimedia on the Web. Unfortunately, this technique is exploited until now only for hyperlink, while its fine-grained synchronization capacity is forgotten. As a more complete approach for this problem, we have proposed a multimedia model based on *structured media* [7] and *sub-element* definitions comprising the *sub-actor*, *sub-interval* and *sub-region* elements corresponding to three axes (style, time and space) of the intra-media structuration. Thanks to the underlying relation-based model, we can use the sub-intervals definitions not only for hyperlinks but also to express fine-grained animations from the abstract level to specific level.

The rest of this paper is organized as follows: section 2 presents related work in the domain of multimedia animation. Section 3 briefly discusses about flexible features of the Madeus model that support the integration of our abstract animation model. In section 4 we introduce our abstract animation model. Section 5 illustrates the application of that model in concrete example. Finally, the current achievement of our work and some perspectives are given in the last section.

2. RELATED WORKS

Right now, when it comes to create animation for the Web, Flash is the king. It enables to create sophisticated Web applications and animation such as e-Learning while delivering low-bandwidth content. However, Flash is based on the SWF binary format that brings well known limitations to end-users and even a poor level of security (virus SWF/LFM-926[1]). The recent release of SMIL Animation [6] opens a new way for specifying lightweight animation effects by putting animation on a time line. In addition, it overcomes the limitations brought by the binary format, e.g., authoring by any textual editor; supporting textual search engines; playing directly within browsers (internet explorer 5.5 or Netscape 6); and most important supporting for the transformation and adaptation. More precisely SMIL animation uses a declarative approach based on an improved path-oriented animation model. It comprises a set of basic XML animation elements: *Animate*, *AnimateMotion*, *AnimateColor* and *Set*. These animation elements carry timing attributes (*Begin*, *End* and *Dur*) for their time layout during presentation of media object. The animation elements can be embedded in other languages such as XHTML+Time, SVG or CSS, called the host languages, to provide animation for these languages.

However these actual uses of SMIL Animation are complex and boring because of the redundancy it implies. Indeed such a package of the animation and its timing layout (even including the target media with *targetElement* attribute) carries out a mono animated application, i.e., an animation is created to animate only one media and often is declared as a component of a media. But we can frequently observe that several media presentations are affected by the same animation function. Let see a specification of such a presentation in Figure 1. The specification tries to display five phrases in sequence and applies on each text's display the same animation that changes of display color from white to black using XHTML+TIME animations[2]. Such a specification is clearly inefficient with five-time the repetition of the same animation, and then could generate tedious work when the author wants to modify that document.

```
<t:seq>
    <p timeContainer="par" timeAction="display"> <t:animateColor
    attributeName="color" from="white" to="black" dur="3s"
    autoReverse="true" />In case you were wondering...</p>

    <p timeContainer="par" timeAction="display"> <t:animateColor
    attributeName="color" from="white" to="black" dur="3s"
    autoReverse="true" /> There is no script on this page.
    Everything you see ... </p>

    ...

    <p timeContainer="par" timeAction="display"> <t:animateColor
    attributeName="color" from="white" to="black" dur="3s"
    autoReverse="true" /> Pretty cool, isn't it?</p>
</t:seq>
```

Figure 1. Specification of SMIL animation in a XHTML+Time document

Note that, SMIL animation elements can be specified outside media elements, and then thanks to strength of full XPointer/XPath, the multi-target media elements are defined (Figure 2). It seems to address the redundancy above, but the timing specifications on the animation will break down the time layout of the presentation. The problem could be radically addressed by a cascading animation sheets[3]. However, the concept is one of the wishes for SVG2.

```
<t:animateColor  xlink:href="XPath expression" begin = "…" dur="3s" … />
```

Figure 2. Specification of a multi target animation

The animation language proposed here is another host language for SMIL animation with the objective of providing a higher level of abstraction and therefore more flexibility in the specifications. In [8] D. Vodislav has shown such flexibilities for creating an abstract animation on a graphic object. A real animation on the object is produced when the real trajectory start point is determined by an initial position of the graphic object. However, we go further in creating not only abstract but also independent animation elements that can be applied on many different concrete media objects in the same time instead of only one media object.

3. FLEXIBLE MULTIMEDIA MODEL

A multimedia document model has to realize the integration of a set of media elements through temporal, spatial and hyperlink models. In most existing models, this integration is often partially mixed, for instance, in XHTML+TIME, spacing and timing attributes are directly attached to the media elements. In SMIL, the spatial layout is separated into the *Head* part of the document. However, the temporal axis of SMIL includes the media element declaration. Such mixed specifications often carry out a complex rendering structure and often generate redundancies when the document grows up. For instance, redundancy occurs when several media elements must be presented at the same time or at the same space. On the other hand, the decomposition of document model in distinct dimensions enables to simplify the authoring and presenting process, for instance it allows to use separate formatters for the spatial and temporal axes. In addition, when the model is decomposed into several axes (object, time, space, hyperlink, etc.), the composition of multimedia presentation is more condensed and more flexible. A media object can be reused for several displaying in presentation.

Following this decomposition approach, our Madeus [3] model can be considered as an extension of SMIL standard for handling the following features: better separate the media, temporal and spatial information; complete the hierarchical temporal operator-based model with relations; provide a more elaborate spatial specification model (with relative placements)More precisely, a Madeus specification has four main parts (see Figure 3):

[1] *Newsbytes, Internet Report*

[2] demos of HTML+TIME, http://research.microsoft.com/~pschmitz/demos/H+Tdemos.html

[3] proposals for additions to the SVG-specification, pro http://www.pinkjuice.com/SVG/spec-prop.xhtml

– The *content* part is an abstract level to allow specifying the resource that will be used one or several time in the presentation. It contains not only row or structured media but also the abstract animation specification as defined in the next section.
– If the first part allows to define the content data, then the *actor* part allows to specify presentation styles on these content data (*DefActor*).
– The *Temporal* part allows conducting the *DefActor* over time. The model used for this level is an interval-based model where the placements of *intervals* can be defined by either absolute coordinates or relations among intervals [1].
– The *Spatial* structure axis organizes the document spatial layout as a 2D box hierarchy where each box is filled with one or several *DefActor*.

Interval and region based models are known to be expressive and flexible models [9]. Their main limitation is given by the granularity provided by the leaves of the structure. We have overcome this limitation by allowing intra-media definitions: *structured media* in the content part*;* *sub-interval* in the temporal part which is basically same as the *anchors* and the *area* SMIL elements; *sub-region* in the spatial part. Moreover, our *sub-interval* element can refer to the content elements (such as an abstract animation element or a video object) in the higher abstracts levels (*content* or *actor* part).

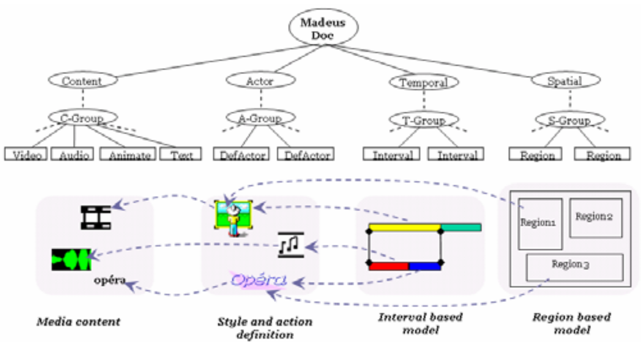

Figure 3. The *content*, *actor*, *temporal* and *spatial* structures of the Madeus model

Figure 3 above illustrates these four parts and the dependencies between them: for instance the media text "Opéra" of the content part is referred by the actor element  (with style attributes) that itself is referred both by the "blue" interval element and the *Region3* region element.

4. ABSTRACT ANIMATION MODEL

Our approach with the abstract animation is the same as the cascading animation sheets. The abstract animation refines on the basic animation elements of SMIL. For that purpose we propose independent definitions of animations from the animated media and the timing attributes. These independent definitions can be considered as the distinction of two axes: *content* and *temporal* of Madeus model. Therefore our abstract animation model can be easily expressed in Madeus. Programming animations according to our model is then performed through two steps: the abstract animation specification and the application specification (Figure 5):

– The *abstract animation specification* defines abstract animations that are presentation-neutral. This definition is a refinement of the SMIL animation in which timing attributes (*begin, dur* and *end*) and *targetElement* attribute are restricted. More importantly, we propose the use of an abstract timing between interval {0, 1} on which abstract time points corresponding to locations on the trajectory, the scaling, the translation, etc. are defined. This abstract interval can be used to map out several real timing layouts for the animation (Figure 4). Due to the presentation-neutral characteristic of the abstract animation, it is specified in the *content* part of the Madeus model.

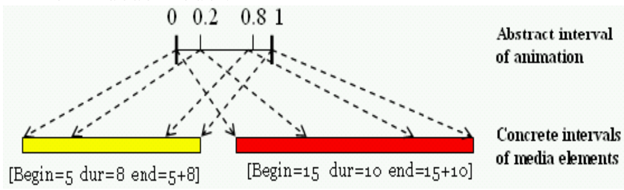

Figure 4. An animation abstract interval maps out two real intervals.

– The *application specification* defines animation's instances that consist of quantitative or/and qualitative timing layouts and target media elements effected on. In fact, like a resource of information that can have several times displaying in a presentation, an abstract animation can be also instanced in several specific animations for animating several objects at different time. Moreover, by definition an animation occurs during the presentation of the object it animates. Therefore, an animation instance is defined as a *sub-interval* element of the animated object which refers to the abstract animation element (Figure 5). The *targetElement* attribute is used to explicitly identify the target media element which is animated.

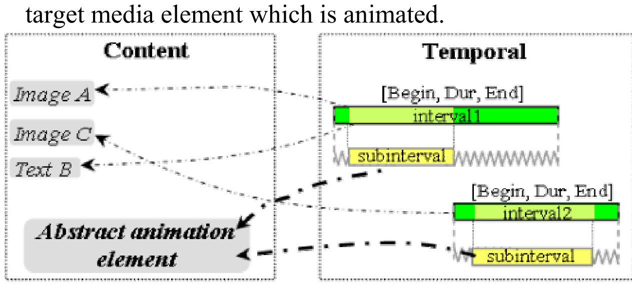

Figure 5. The abstract animation model cross the *content* and *temporal* dimensions

Our approach can be compared to the media construction abstraction (MCA) proposed by Nanard [4]. MCA aims at helping users in their design process through MCF diagrams where media composition allows event-based definitions. Therefore, it uses a box and connector paradigm. On the contrary, our animation definition has an interval-based specification where composition results from relations between intervals. Therefore, it allows the specification of synchronizations between animation instances and other objects of the document.

5. EXAMPLE SCENARIO

We illustrate in this section the concrete animation scenario (the demo is described in detail online[4]) used to demonstrate our prototype *AnimationMadeus* based on our abstract animation model. The demo uses two abstract animations (Figure *6*) to put smoothly in appearance/disappearance all the text and image objects by doing up/down the *alpha* component of visual source.

```
<Animate ID="UpAlphaAni" attributeName="AlphaSource"
values="0;1.0" keyTimes="0;1" calcMode="linear" additive="replace"/>
<Animate ID="DownAlphaAni" attributeName="AlphaSource"
values="1.0;0" keyTimes="0;1" calcMode="linear" additive="replace"/>
```

Figure 6. Two abstract animation specifications

Figure 7 is an excerpt code illustrating the application specification of the *UpAlphaAni* abstract animation. The animation is applied during 1.5 second at start time of each image in the sequence of eleven consecutive images. Figure 8 presents these specifications through the execution and timeline views. The author can **directly manipulate** the graphic presentations of the animations (the yellow rectangles in the timeline view) to adjust the timing layout of these animations. In the example, all the animations are placed at the start time of the image intervals, except the fifth that has been moved by the author.

```
<Interval ID="Image1" Actor="DefActor_12083" Duration="pref:5480ms">
  <SubInterval ID= " ..." Animate= " UpAlphaAni" Duration="pref:1.5s"/>
</Interval>
...
<Interval ID="Image11" Actor="DefActor_12094" Duration="pref:3039ms">
   <SubInterval ID="..." Animate="UpAlphaAni" Duration="pref:1.5s"/>
</Interval>
```

Figure 7. The application specification of the *UpAlphaAni* abstract animation.

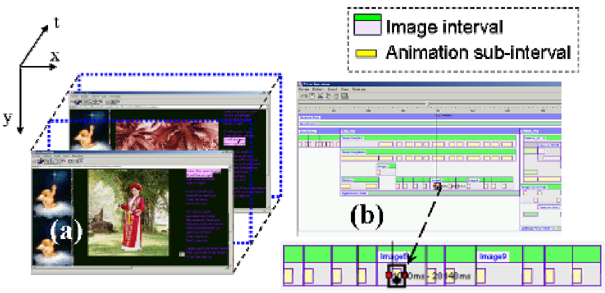

Figure 8. (a) The execution and (b) timeline view

6. CONCLUSION

This work has proved that *abstract animation* specification can be provided for XML documents. The work refines the set of basic XML animations of SMIL. In addition, using *sub-interval* element to express the specific animations allows defining the timing layout of the animation by either absolute coordinates or relative constraints among intervals. One interesting point to notice is that the abstraction level has been easily obtained thanks to neat underlying model that separately express content, spatial and temporal information.

As shown in the previous example the approach provides a good basis for the edition of animation through synchronized views that filter each dimension. As far as we know this is one of the first steps for allowing users to define animations in a non-programmatic but interactive way. A prototype implementation of this abstract animation model has given us exciting results.

REFERENCES

[1] J.F. Allen. "Maintaining Knowledge about Temporal Intervals". *Comm. ACM*, 26(11):832-843, 11 1983.

[2] S. Boll, W. Klas, U. Westermann, "Multimedia Document Models – Sealed Fate or Setting Out for New Shores?", Conference on Multimedia Computing and Systems (ICMCS) '99, June 7–11, 1999, Florence, Italy.

[3] M. Jourdan, C. Roisin, L. Tardif, "A Scalable Toolkit for Designing Multimedia Authoring Environments", *Multimedia Tools and Applications Kluwer Academic Publishers,* vol. 12, num. 2/3, pp. 257-279, November 2000

[4] M. Nanard and J. Nanard, "Towards Multimedia Computing, Lessons Learned From A Half Century of Computer Science", International Conference on Media Futures, Florence, Italy -8-9 May 2001.

[5] G. van Rossum, J. Jansen, K. Mullender, D. Bulterman, "CMIFed: a presentation Environment for Portable Hypermedia Documents", *Proc. of the ACM Multimedia Conference,* California, 1993.

[6] P. Schmitz, A. Cohen, K. Day, "The SMIL 2.0 Animation Modules", http://www.w3.org/TR/smil20/animation.html.

[7] T. Tran_Thuong, C. Roisin, "Structured Media for Multimedia Document Authoring", International Workshop on Web Document Analysis (WDA), Seattle, Sept. 2001.

[8] D. Vodislav, "Visual Programming for Animation in User Interfaces", (poster) Proceedings of the 11th IEEE Symposium on Visual Languages (VL'95), Darmstadt, Germany, September 1995.

[9] T. Wahl, K. Rothermel, "Representing Time in Multimedia-Systems", *IEEE Int. Conf. on Multimedia Computing and Systems*, May 1994.

[4] http://www.inrialpes.fr/opera/people/Tien.Tran-Thuong/Demos/multiviews.htm