Multimedia Authoring: A 3D Interactive Visualization Interface based on a Structured Document Model

Nabil Layaıda and Jean–Yves Vion–Dury

INRIA–IMAG, ZI de Mayencin, 2 rue Vignate, F–38610 Gières, France
E–mail: {Nabil.Layaida, Jean–Yves.Vion–Dury}@imag.fr

Multimedia authoring process is inherently a complex and tedious task, users have to specify all the details of a multimedia presentation (temporal coordination, user interaction and spatial placement of data on the user display). In the same time they have to keep in mind a global overview of the entire document.

Current document systems uses scripting languages or timelines to specify multimedia documents. A complete specification of all the low–level presentation details puts a heavy burden on the editing task, and the creativity of the author becomes limited by a hight cognitive overload. The complexity of editing multimedia documents is mainly related to the various tasks involved: document organization, temporal synchronization, spatial placement of multimedia objects and resource attribution. It is therefore necessary to perform an analysis of theses tasks to build an efficient authoring interface.

In this paper, we present an interface based on a structured document model and multiple interactive views. This synthetic approach relieves the user from low–level and error–prone descriptions by reducing the document complexity. We believe that it is an efficient way for enhancing the overall interface expressive power. Moreover, the structured approach eases the automatic processing of multimedia documents, allowing a rapid production of spatial and temporal layouts starting from high level logical descriptions and presentation directives.

In the first part, of the paper we present the state of the art, the multimedia data and document model. In the second part, we describe our user interface and its different views.

1 MULTIMEDIA AUTHORING: STATE OF THE ART

Most of the developments in the field of multimedia authoring systems focus on the use of programming languages, or scripts, to describe the content of multimedia documents. The main advantage of scripting is its ability to describe accurately the layout of the document (timing, spatial placement, etc.), so its easy to determine the document's behaviour at the presentation time. Scripting, by bringing some autonomy to the multimedia objects, looses the global vision of the document structure and makes the different parts of the document somehow contextual. Therefore, documents become hard to maintain and to perceive by the author since the spatial, temporal and structural information is explicitly mixed in the document representation.

Scripting does not also take advantage of document processing techniques (interactive interfaces, formatting, cutting and pasting, etc.) which is a fundamental aspect in the field of electronic documents. Finally, scripting which is based on a on–line programming lacks an efficient interactive user interface.

In some other systems [2][4][6][7], the interactivity of the interface is a central issue. It is generally based on a graphical representation of the multimedia document: timelines in MAestro [2], dependency graph in Firefly [7] and Petri Nets [4]. These systems focus on one particular aspect of the authoring process, for instance the temporal synchronization problem, therefore they remain incomplete for a final use.

The purpose of our approach is the design of an integrated authoring interface, taking into account the different aspects of a multimedia document and particularly its structuration.

2 TOWARD A STRUCTURED DOCUMENT MODEL

Multimedia documents are defined as a given set of multimedia objects and the full description of their relationships. These relationships organize the objects through different structures involving logical, spatial and temporal aspects of the document. The author's task is essentially the specification of these structures, task which can be significantly enhanced if the following functions are provided by the authoring system:

- **A composition model:** its defines a structuring facility allowing users to create multimedia documents incrementally. It allows users to produce independently different parts of a document and then link them together. This point is an important issue to build hyperlinks which requires parts of the document to be context free.

- **A model of property inheritance:** it defines a set of computation rules for object properties, for example if the author specifies a parallel presentation of a text object and a music, given the static nature of the text component, it is natural that it inherits its duration from the time–based music object. More generally, it is possible to define inheritance rules for colors, fonts, sizes and time durations. It also possible to apply constraints on a group of objects (reduction of the time duration, zoom of the window space). This facility eases the authoring by reducing the number of user specifications necessary for a given document.
- **The coherence checking:** this function helps the author to find interactively the errors introduced in the document specification. Typical examples are the introduction of impossible temporal combinations, or the simultaneous presentation of two videos placed in different pages of the the window. The later example outlines the strong dependency between the time and space dimensions.

2.1 Multimedia data model

Multimedia data and document model's design aims to provide abstractions to the user and the application engine. At the authoring level, data abstraction reduces the number of concepts manipulated by the author by giving a uniform name space for the various MM types. At the application level, the data processing which requires a full representation of object properties (format, size, duration, etc.) must be transparent to the user. In this purpose the MHEG [8] group is defining in an object oriented fashion the MHEG standard. Objects are grouped in different classes, task which relies on the analysis of their common behaviour and properties.

One of the most important aspects of multimedia objects is their temporal dimension. The temporal formatting process is heavily dependent on the temporal model, which together with the inheritance rules of properties is considered as the heart of the multimedia authoring system.

The analysis of multimedia objects shows that three classes of objects are necessary to define a time computation policy in the application :

- **discrete objects** which have no intrinsic temporal dimension (text, images, figures),
- **continuous objects** having flow constraints and flexibility properties (sound, video),
- **indeterministic objects** (discrete or continuous) which have unpredictable durations (external programs, database queries).

A specific compound element allows to build composite objects improving the reusability and the encapsulation of parts of documents. Composite objects contributes also for a significant improvements of the user interface when manipulating large documents.

2.2 Multimedia document model

The document model is defined as the representation of the set of objects and the description of the relations between them. When editing a multimedia document, authors handle it through four different aspects:

- **The logical structure:** With the structured approach, a document is considered as an organized set of *typed elements* which are logically related to each other. In a multimedia document, typical elements are the title, paragraphs, tables, graphics, figures, 3D animation, audio, video, etc. All these elements have various structural relationships: a movie *contains* a video, some paragraphs and optionally background music; a scene *follows* another scene; a part of an image may *refer to* a section item, etc. With this approach, a document is represented in the system by its hierarchical logical structure with additional relationships that represent non–hierarchical links between elements such as hypermedia links.
- **The spatial presentation of objects:** In text editors [3], the placement of objects on the window or on the paper space is commonly called spatial formatting. This operation uses the document logical structure described above and follows the principle of embedded boxes: every object having a spatial dimension (a box) is embedded in its parents box (in the logical structure). Relative positioning rules of elements are used to solve most of the objects placements. In the case of multimedia documents, the time–based nature of some objects adds to this principle a dynamic dimension: the object box space on the window can be created and freed dynamically, therefore, spatial formatting must be performed along the time flow.
- **The temporal synchronization:** The temporal structure uses an interval based algebra for expressing and maintaining temporal constraints between objects. This algebra uses Allen[1] temporal operators (*meets, equals, overlaps, before, etc.*) to describe the objects synchronization. The temporal coherence of the document is performed at two levels: qualitative coherence is checked interactively when the author modifies a given relationship, and quantitative when the author explicitly performs a temporal formatting

operation. These two levels are necessary because formatting operations use optimization algorithms which are costly in time, thus their extensive use do not meet interactivity requirements.

- **The resource allocation:** The resource allocation problem comes from the concurrent nature of parallel presentations. Depending on the reader environment, multimedia objects requiring hardware devices, like audio and video channels may not be displayed simultaneously. For exemple, if the reader has not the necessary resources. The resource allocator component is used to serialize the resource attributions by adding special temporal precedence constraints. This operation requires the description of the reader environment, and must be performed dynamically at the presentation time.

3 AUTHORING INTERFACE

The author interfaces have to deal with a difficult trade–off: how to manage complex tasks while preserving the user creativity ? This complexity is qualitative when considering the multiple concepts involved. It can also be quantitative if we address the edition of large documents. We propose to tackle this complexity following three principles:

1. **Use of virtual images**

   Virtual Images [5] are 3D interactive graphical representations that allow to visualize and manipulate large object sets while giving a high visibility of their global relationships. Objects are represented by polyhedrons with significant shape, color and size, and their relationships are expressed using space organization. We first developed this visual concept for debugging object oriented applications running on top of the GUIDE distributed system [5]. But due to their high interactivity level and their ability to express complex relationships, we believe that their use is well adapted to handle multimedia documents.

2. **Use of multiple views**

   Multiple views allow the author to project the global multi–dimensional information space of a document on different sub–spaces. A given view is designed for a particular task according to the various dimensions of a multimedia document. This projection is a powerful way for reducing the complexity by filtering the information. Views can be completely orthogonal or complementary.

3. **Maintain of a strong coherence between the different views**

   In our opinion, the coherence checking of the author specifications is an important issue. Since documents can be modified from any particular view, all other views must be updated to give a feedback of the global effect induced on the document. This strong coherence is achieved toward a common internal information space which allows translations of modifications from one information sub–space (a view) to any other sub–space. Therefore, it seems necessary to have a common internal representation of the document, to centralize changes, and to dispatch them to the different views.

In the following sub–sections, we outline three views corresponding to the main authoring tasks.

3.1 The structural view

This view represent the temporal structure of the document as a 3D tree. Nodes represents temporal operators and leaves are multimedia objects. Both entities are represented as polyhedrons whose shapes and colors are related to their types. The space organization of the tree reflects the temporal nature of operators: branches expand in the Y=0 plan for sequential operators and in the X=0 plan for the parallel ones. The last Z=0 direction is used for showing hierarchical composition of operators describing the algebraic expression of the document. Thus, the user gets an intuitive perception of the temporal structure without being overflowed with detailed time information (the time perception is here *logical* and not *physical*). The user have interactive exploring facilities for discovering the information in this 3D workspace. He can act directly on the structure using mouse manipulations in order to change the quantity of displayed information (clicks on nodes for expanding or contracting parts of the tree) or to "open" polyhedrons that represents holophrasted composite objects. Beside the main tree structure, the user can specify non–hierarchical temporal constraints by binding objects (polyhedrons) with typed segments. This last possibility allows the specification of additional non hierarchical temporal relations, which is complementary to the algebraic description of the multimedia document.

3.2 The resource view

Resource oriented views are generally called *timelines* because multimedia documents are represented with a set of temporal axes corresponding to media channels. Multimedia objects are represented with segments whose length corresponds to their presentation duration. The user perceive the temporal sharing of physical resources, and can modify it. One major drawback of the classical approach of timelines is due to the lack of user actions feedback. Authors must maintain the global coherency of time segments "by hand" according to their own mental model. This task becomes quickly boring because of the hight number of side effect changes to be performed.

In our prototype, multimedia channels are presented as parallel time–lines wrapping around a central axe, thus forming a 3D cylinder. The linear nature of this cylinder allow to use the perspective effect for giving a global overview, as used in the *perspective wall* [5]. Multimedia objects (polyhedrons) are represented with begin–end markers (polyhedrons with the same colors and shapes) joined by a segment proportional to its duration. They can be slided on the lines with smooth mouse manipulations. Time constraints are represented as rubber–bands joining the polyhedrons. One property of virtual images is the use of a motion engine based upon gravitational laws. This metaphor produces realistic displacements of objects and allows the simulation of a force model for representing various constraints. Objects can also be bound together using springs with mechanical characteristics (default length and strength) related to abstract constraints or relationships. This technique is useful for producing spontaneous space organized structures. Graphically, it induces a perceptual continuity on the document changes (which is known as an important quality criterion for the user interaction [5]). We use these features for showing temporal constraints between multimedia objects. When the user moves an object (changes its temporal relationships), all the other objects having a temporal dependency with it move according to that modification, giving a progressive perception of the relationships dependencies.

3.3 The presentation view

This view presents the document on the author screen as it will be finally displayed to the reader. Given the temporal dimension of the document, this view is mapped in a 3D dimension space, 2D for the screen layout and 1D for the time flow. The functional core of the editing tool produces a first spatial formating, starting from the internal encoding of the document. This operation is automatically done by a formatter component which manages text, images and video placements. Other multimedia objects are mapped in respect to their correspondent channel. The presentation view also appear as a set of planes corresponding to the different scenes of the document. These planes allow the user to check the quality of the automatically produced spatial layout, and eventually, to modify it. Thus, this view is used to refine interactively the automatic formating, bringing some flexibility in the document manipulation which is considered as one of the major drawbacks of structured editors.

4 CONCLUSION

The key message of this paper is that the multimedia authoring can be approached through the synthesis of two complementary aspects strongly related: an innovative interface adapted to the complexity of multimedia documents, and a structured representation that allows automatic processing. Few works have addressed the multimedia authoring interface problem, and fewer have tested 3D visualization techniques.

This work is currently experimented in the Opera project. The goal of this project is to develop an editorial environment for the construction, manipulation and storage of complex multimedia documents. The Opera system is based on a logical document structuring scheme that was introduced and tested during the development of the Grif editor [3].

REFERENCES

1. J.F. Allen, ''Maintaining Knowledge about Temporal Intervals'', *Communications of the ACM*, Vol 26 (11), pp. 832–843, November 1983.

2. G.D. Drapeau et H. Greenfield, ''MAEstro: A distributed Multimedia Authoring Environment'', *Proceedings of summer USENIX 1991*, June 1991.

3. R. Furuta, V. Quint et J. André, ''Interactively Editing Structured Documents'', *Electronic Publishing –– Origination, Dissemination and Design*, 1(1), pp. 19–44, April 1988.

4. T.D.C. Little and A. Ghafoor, ''Interval–based Conceptual models for Time–Dependent Multimedia Data'', *Workshop on Multimedia Information Systems, Arizona*, February 1992.

5. Jean–Yves Vion–dury et Miguel Santana, ''Virtual Images: Interactive Visualization of Distributed Object–Oriented Systems'', *OOPSLA'94 Proceedings*, pp. 65–84, October 1994.

6. G. Van Rossum and al, ''CMIFed: A Presentation Environment for Portable Hypermedia Documents'', *First ACM International Conference on Multimedia*, pp. 183–188, August 1993.

7. M. C. Buchanan et P.T. Zellweger, ''Automatic Temporal Layout Mechanisms'', *First ACM International Conference on Multimedia*, pp. 341–350, August 1993.

8. R. Price, ''MHEG: An Introduction to the future International Standard for Hypermedia Interchange'', *First ACM International Conference on Multimedia*, pp. 121–128, August 1993.

Bibliographie

[1] J.F. Allen, ‘‘Maintaining Knowledge about Temporal Intervals’’, *Communications of the ACM*, Vol 26 (11), pp. 832–843, November 1983.

[2] G.D. Drapeau et H. Greenfield, ‘‘MAEstro: A distributed Multimedia Authoring Environment’’, *Proceedings of USENIX summer 1991*, June 1991.

[3] R. Furuta, V. Quint et J. André, ‘‘Interactively Editing Structured Documents’’, *Electronic Publishing –– Origination, Dissemination and Design*, 1(1), pp. 19–44, April 1988.

[4] T.D.C. Little and A. Ghafoor, ‘‘Interval–based Coceptual models for Time–Dependent Multimedia Data’’, *Workshop on Multimedia Information Systems, Tempe, Arizona*, February 1992.

[5] Jean–Yves Vion–dury et Miguel Santana, ‘‘Virtual Images: Interactive visualization of Distributed Object–Oriented Systems’’, *OOPSLA'94 Proceedings, Portlan, USA*, pp. 65–84, October 1994.

[6] G. Van Rossum and al, ‘‘CMIFed: A presentation environment for portable hypermedia documents’’, *Proceedings of the First ACM International Conference on Multimedia*, pp. 183–188, August 1993.

[7] M. Cecilia Buchanan et P.T. Zellweger, ‘‘Automatic Temporal Layout Mechanisms’’, *Proceedings of the First ACM International Conference on Multimedia*, pp. 341–350, August 1993.

[8] R. Price, ‘‘MHEG: An Introduction to the future International Standard fot Hypermedia Interchange’’, *Proceedings of the First ACM International Conference on Multimedia*, pp. 121–128, August 1993.

Multimedia Authoring: A 3D Interactive Visualization Interface based on a Structured Document Model

1 MULTIMEDIA AUTHORING: STATE OF THE ART 1

2 TOWARD A STRUCTURED DOCUMENT MODEL 2

2.1 Multimedia data model 2

2.2 Multimedia document model 3

3 AUTHORING INTERFACE 4

3.1 The structural view 5

3.2 The resource view 5

3.3 The presentation view 6

4 CONCLUSION 6