Madeus, an Authoring Environment for Interactive Multimedia Documents

Muriel Jourdan, Nabil Layaïda, Cécile Roisin, Loay Sabry-Ismaïl and Laurent Tardif
OPERA project, INRIA Rhône-Alpes.
655 avenue de l'Europe, 38330 Montbonnot, France.
e-mail: {Firstname.Lastname}@inrialpes.fr
tel: +33 4 76 61 52 00, fax: +33 4 76 61 52 52
http://opera.inrialpes.fr/OPERA/

Abstract

In this paper, we present the main results we have obtained through our experience in designing and implementing Madeus, an authoring and presentation tool for interactive multimedia documents. This system allows an efficient support for the specification of temporal scenarios and fits in an architecture that allows the integration of both authoring and presentation phases of multimedia documents. Moreover, we show that constraints are well suited for the development of both end-user graphic interfaces and efficient low-level execution mechanisms.

1 Introduction

Developing a multimedia authoring and presentation application is a difficult task as many different problems have to be solved. The temporal nature of information in multimedia documents requires new functions such as temporal synchronization between components at authoring and presentation time and scheduling of the presentation in distributed and unpredicted contexts. More traditional functions have also to be adapted to these new environments, such as graphical authoring interfaces and spatial formatting on the screen. Research in those areas has been very active in the last decade and a number of papers has been published on these topics. Unlike the web, new research models, experimental tools and document standards failed to really have an impact on existing technology (such as Apple's Quicktime) or to trigger the emergence of a new generation of tools. Although multimedia document modelling has been given a great attention, the creation or authoring process was not sufficiently covered. Nevertheless, one of the key issues is to define the level of flexibility required to make those tools usable. Perhaps the most prevalent temporal model is the timeline, which aligns all events on a single axis of time. Though simple and graphical, timelines lack the flexibility required for frequent scenario modifications carried out by the author before reaching the desired scenario. We believe that authoring and presentation demand more powerful schemes to become more accessible and effective. One such scheme is the constraint-based technique which is the key of our approach to enhance authoring and presentation of multimedia documents.

In this paper, we present the main results we have obtained through our experience in designing and implementing Madeus [6], an authoring and presentation tool for interactive multimedia documents. Madeus covers a broad range of features such as structural composition, temporal and spatial specifications, integration of edition and presentation phases. The current implementation (in C) supports Mpeg audio and video, different image formats and formatted text.

The development of Madeus is still in progress, but from its current state we can already report on the foundation of the choices that have been taken initially. In particular, we extensively discuss the usefulness of the heart of the tool which is a constraint-based system. This system allows an efficient support for the specification of temporal scenarios and fits in an architecture that allows the integration of both authoring and presentation phases of a multimedia document. Moreover, we show that constraints are well suited for the development of both end-user graphic interfaces and efficient low-level execution mechanisms.

This paper is organized in three parts. First we present the most important choices that guided the development of Madeus. Then, we outline the advantages of constraints at the authoring stage. We also explain how a presentation is carried out from a constraint representation of the document. Finally, we conclude and indicate some future research directions.

2 Related work

Research works on authoring environment for the design of (synchronized) multimedia documents are not as much developed as other multimedia areas. The two first known prototypes were Firefly [2] and CMIFed [14] which raised the debate on constraint-based versus operational environments. These two classes have been defined depending on how close the document description is to the presentation phase. With an operational approach, the author specifies the way in which scenario must be executed by means of either a script language or an operational structure (tree or Petri-net). Therefore the presentation phase directly implements the operational semantics provided by the used structure. With a constraint-based approach, the author specifies the required scenario without determining how to get the result in terms of operational actions. Constraints resolvers are then used to translate this set of declarations into an executable form. The debate between these two different ways of specification is still going on and new propositions on both sides frequently appear:

ACM Multimedia'98, Bristol, UK
 $5.00

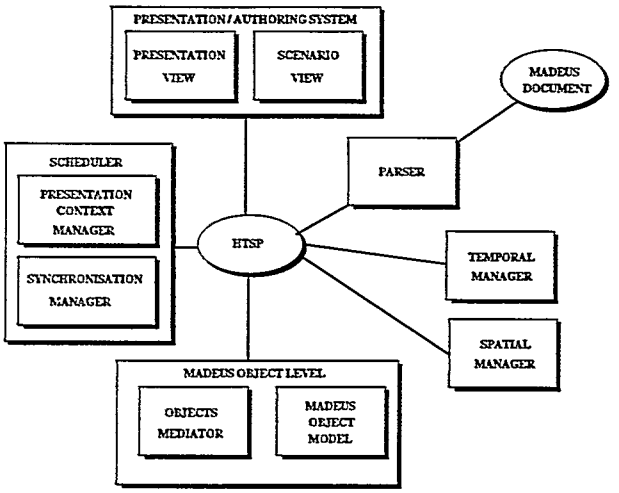

Figure 1: Madeus architecture

- ISIS [9], Madeus [6], TIEMPO [15], CHIMP [3] are constraint-based environments
- Mediadoc [8], HyperProp [11] and HPAS [18] are operational ones.

It is clear that it is easier to build an authoring environment based on an operational approach than to do the same using a constraint-based formalism. Indeed, theoretical problems associated with constraint-based approach are known to be difficult [10]. This explains why commercial tools are not based on constraint technics. The same is true for the standard promoted by the W3C for synchronized multimedia documents called SMIL [17].

We are convinced by the strong advantages that constraints could bring to authoring environments of multimedia documents. The aim of this paper is to contribute to this debate by giving a detailed presentation of all the benefits that multimedia authors could make by using such environments, through the presentation of the Madeus prototype.

3 Madeus: a constraint-based authoring tool

Madeus is a multimedia authoring and presentation system characterized by 1) a constraint-based specification of scenarios, 2) a hierarchical organization of documents, 3) the ability for the author to edit a document on multiple views, 4) a powerful internal structure that captures the temporal information, 5) an object-oriented architecture to handle media objects and 6) a plug-in mechanism to incorporate external objects in documents. These features are described in the rest of this section.

Fig. 1 gives a global view of the architecture of Madeus. We can see how the different components are organized to provide an integrated system.

In the rest of this paper, we use the expression "multimedia document" when considering the whole information attached to the document: the set of basic objects it contains (text, image, video, audio, etc.) and their logical, spatial, navigational (links) and temporal organization. When we only want to speak about its temporal aspects, we use the term "temporal scenario" or simply "scenario".

Merry *BEFORE* And
And *MEETS* Happy
Smiley *DURING* And
Merry *EQUALS* Song1
Happy *EQUALS* Song2
Merry *STARTS* Pict1
Pict2 *MEETS* Pict3
Pict3 *MEETS* Pict4
Pict4 *FINISHES* Happy

Merry: [25, 35]
Song2: [10, 60]
And: [25, 35]
Pict$_i$ [15, 120]
Happy: [25, 35]
Smiley: [5, 15]
Song1: [10, 60]

Figure 2: Temporal specification of the Christmas example

3.1 Using spatial and temporal constraints

Madeus is a constraint-based authoring tool where the author can describe the spatial and temporal organization of a document by setting constraints between basic or composite objects (see section 3.2). These constraints can express spatial and temporal synchronization such as: two videos must be vertically centered and must be presented during the same period of time. Then, the spatial and the temporal formatters compute the position of media objects in both spatial and temporal dimensions.

Madeus uses classical spatial constraints such as *align*, *center* and *shift*, on both the vertical and horizontal axes. Once a constraint is set, it holds during the whole presentation, even when the user moves manually some objects on the screen. The Madeus spatial formatter is based on DeltaBlue [13].

Temporal constraints used in Madeus are based on Allen's algebra [1] using the operators `EQUALS`, `STARTS`, `BEFORE`, etc.

Basic objects of the scenario are associated with a range of possible durations which can be defined by the author himself or automatically assigned by the system (for instance, an interval ranging from 5 to 120 seconds for images). This range can also be defined as an infinite interval.

Fig. 2 gives the set of temporal relations which describes the following informal specification: The document starts by displaying a text "Merry Christmas" (Merry) accompanied by a Christmas song (Song1). Following these two objects, the text "and" (And) is displayed, followed by the text "Happy New Year" (Happy) that is accompanied, in its turn, by an appropriate song (Song2). Each of the three textual messages is presented for a period of about 30 seconds. In addition, a "smiley" (Smiley) makes a brief appearance during the presentation of the text "and". The background of the document is composed by a sequence of some Christmas pictures (say 4: Pict1, Pict2, ...). The two songs can be played with any duration in the range between 10 and 60 seconds without considering their complete delivery. We can notice that in this example, the author is not asked to give precise temporal information in order to obtain the desired synchronization. In addition, he can take advantage of this flexibility when reusing this document (see 4.3).

3.2 A hierarchical decomposition of the document

It is possible in Madeus to hierarchically put together a set of objects in one entity, namely a *composite object*, which could be used in the remainder of the scenario as a basic object.

As an example of hierarchical decomposition, let's consider the set of objects given in Fig. 2. They can be encapsulated in a composite object named "Christmas_Card" and a "News" document can be defined as:

News = Christmas_Card *BEFORE* Family_News

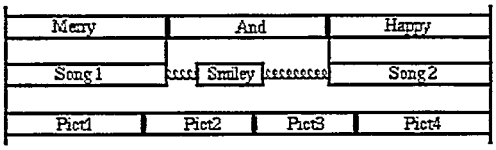

Figure 3: Scenario view of the Christmas example

The semantic of this encapsulation, in the temporal dimension, is given by the following rule: the temporal interval associated with a composite object is the shortest interval which can temporally contain all its components.

For example, hierarchic decomposition allows authors to use Madeus to design complex documents. We have created a multimedia presentation of the INRIA institute, comprising about 70 basic objects structured in 5 scenes.

3.3 An expressive textual format

Madeus uses a textual format for storing the declarative specification of the hierarchical, spatial, temporal and navigational organization of multimedia documents. A document has the same structure as a composite object: a list of basic and/or composite objects. Spatial, temporal and style attributes can be attached to basic objects (duration bounds of an object, font for a textual object, etc). Some attributes allow to express a dynamic behavior such as a moving style effect applied on a visual object. In addition, *Spatial-Rel* and *Temporal-Rel* attributes can be attached to composite objects: they contain the list of temporal and spatial relations specified between the components of the composite objects. Another kind of attributes is the hypermedia links, represented in an HTML-like way where the destination anchors are designated using their URLs. The grammar of Madeus language is based on the XML syntax [16].

3.4 A graphical view of the temporal scenario

Whatever the approach used to specify the temporal scenario, it is worth noting that the author must be provided with a graphical view of the temporal organization of documents. The "temporal axis" paradigm is known to be well-suited in such a case. In Madeus, the "scenario view" shows objects placed along a temporal axis. Moreover, the temporal constraints set by the author are also represented in this view. Fig. 3 shows the view of our Christmas card example where springs represent flexibility introduced by constraints such as before or during and vertical lines represent simultaneous instants. One of the leading points of this view is that objects are spatially aligned in the vertical dimension if and only if they are temporally aligned. This is a necessary condition to remove any ambiguities for the author.

The other interesting point of the scenario view is that the author can interact with it (i.e. moving an object along the horizontal axis or resizing it) in order to browse through the set of solutions deduced from the temporal constraints. The other objects are adjusted in real-time. When using a constraint-based approach to specify the temporal scenario, the access to the whole set of solutions is very helpful for the author to understand the global meaning of the set of constraints.

These two points (spatial alignments to denote temporal ones and dynamic manipulation of the view) are two new features which are not provided to the author in other constraint-based environments. For instance, the temporal

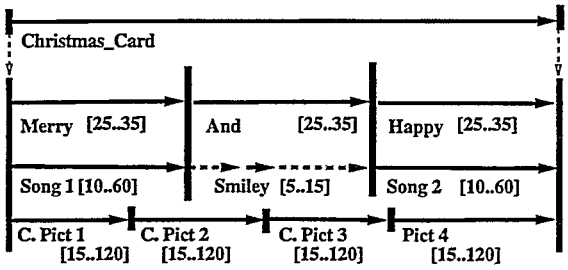

Figure 4: The internal representation of the Christmas example in Madeus

view of ISIS is a static one, in which spatial alignments are not equivalent to temporal ones.

3.5 Temporal information managed by a powerful internal structure

In Madeus, temporal information handling is based on Simple Temporal Problems (STP) [4]. An STP is a directed acyclic graph where each node represents a time point and each edge (**i**, [**min**,**max**], **j**) represents a temporal interval with a duration range from **min** to **max** between the two time points **i** and **j**. Numerous algorithms exist to answer questions like: does a solution exist (consistency checking)? Can a particular set of values be considered as a solution for the problem?

In Madeus, this formalism has been extended into a Hypergraph of STP (denoted by HSTP) in order to manage the hierarchical organization. The nodes of the HSTP represent the begin and end instants of composite and basic objects. The range of possible durations of an object is modeled by an edge between its begin and end instants labeled by its appropriate interval. Other edges in the HSTP are added to take into account the temporal constraints that the author has placed on the objects [10] (see Fig. 4 the "News" HSTP example). Moreover this structure is used by the presentation system to schedule the execution of- and navigation within- documents.

3.6 A HSTP-based scheduler

Presentation schedulers can be either purely reactive (as [5]) or predictive-reactive. In the first case, the scheduler knows nothing about the future of the presentation. It reacts when it receives an event by invoking the action associated to this event. The predictive-reactive scheduler works as the purely-reactive scheduler but in addition it uses a structure that maintains a scheme of the whole scenario which gives the scheduler hints about the future of the presentation. The Madeus presentation scheduler is of the second type: it is mainly driven by the HSTP (that gives the overall order of the scenario), but also by the user interaction events (that specify what kind of presentation service are required by the user) and the current executional state of media objects (that reflects the executional state of the whole document).

Based on the above information, the scheduler decides to start and terminate the presentation of media objects at the instants corresponding to those specified in the scenario.

3.7 Control of object progression

In Madeus, the control of the progression of each presented object is performed by an event-based algorithm using a

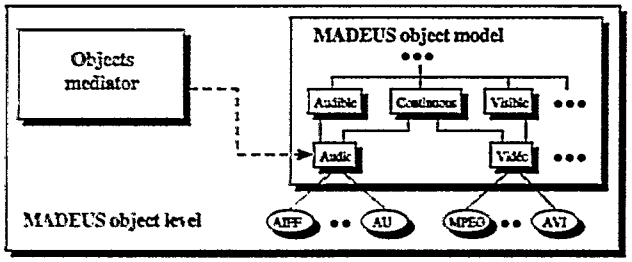

Figure 5: Madeus object level

sync notification period. The *sync* notification period is the period with which the synchronized objects are notified to align themselves temporally with respect to each other. This period corresponds to the time needed to play a whole number of the units composing every synchronized object, such as frames of video or samples of audio.

3.8 Presentation context management

The information about a presentation instant is encapsulated into a structure called a *presentation context* which holds the executional status of the media objects presented at this instant [12]. For each object, the context records its state (running or not) and, if running, its progression time relative to its beginning. Moreover, presentation instants are classified into two main types: *Tic* instants at which a set of presentation actions must be carried out; and *Tac* instants which are the Tic instants where at least one other media object starts or ends.

Moreover, Madeus performs some actions on presentation contexts, such as *store*, *update* and *calculate*, in order to achieve time-dependent operations. We have developed an appropriate algorithm for the computation of the presentation context for any Tic or Tac in the future or in the past: this algorithm takes into account both the information given by the HSTP and the temporal durations that have been statically computed by the formatter.

3.9 Object handler

Madeus proposes an object model for representing the basic media objects taking part in a multimedia presentation. Each time a new media object has to be presented, the object mediator creates an instance of the appropriate class that will manage its presentation.

Madeus handles a table of the supported media formats. The object mediator can obtain from this table all the information that it needs, such as the type of object to be created, the data stream and the available presentation services such as play, pause, resume, fast forward, fast rewind, etc.

The different object types are organized into an open generic object model hierarchy (Fig. 5) that gives powerful abstraction and inheritance features. In the hierarchy, we were keen on defining classes that reflect the perception characteristics, as in *visible* and *audible* classes, as well as the temporal characteristics, as in *continuous* and *opaque* classes. More detailed media object classes, such as audio, video, image, etc., are derived from them, while format classes, such as AU and AIFF audio classes, and MPEG and AVI video classes, are set at the lower level of the hierarchy.

Moreover, Madeus adopts a plugin mechanism to allow the utilization of a wide variety of objects directly managed by external applications. A plugin works as any other media object defined in the Madeus object model, but it is characterized by having a generic interface independent of the plugged-in application. However, from the point of view of Madeus, plugins are characterized by being opaque in nature and indeterministic in behavior. The opaqueness in nature is due to the fact that the plugin interface gives a limited set of functions to control the plugged-in application, hiding the information about the plugin presentation components. Therefore, Madeus has no fine presentation control on the media objects presented by the plugged-in application, such as the frame rate for a plugin video. As a result, a plugin object is considered to have an indeterministic duration that may conflict with the constraint-based approach of Madeus which supposes the control of object duration. The management of uncontrollable objects is one of the main research issues of our on-going works.

4 Benefits for authoring

In this paper, we would like to focus on three main authoring results obtained thanks to Madeus features:

1. A secure and incremental editing process.
2. An editing process tightly coupled with the presentation process.
3. The reusability of specifications.

4.1 A secure and incremental design of a multimedia document

Designing an interactive multimedia document is a cyclic "specify, test and modify" process: the author can rarely obtain the desired document from the first specification. One important point is the facility in which the authoring environment can help the author in adjusting his document. The use of constraint paradigms in Madeus is a real advantage as far as the following characteristic is concerned: each time the author adds or deletes a constraint (spatial or temporal), the current set of solutions is readjusted. For instance, if the author wants to insert in the Christmas example a fifth background object between the two previously mentioned ones, it is sufficient to modify the existing constraints in order to have:

Pict1 MEETS New_Pict; New_Pict MEETS Pict2

The temporal formatter automatically checks the consistency of the new scenario, thus at each editing step the author is sure of having a consistent scenario. If the scenario remains consistent, the temporal formatter adjusts the previous solution to take into account the newly inserted object. If consistency is violated, the author is aware and can adapt the document.

Working with a scenario view (see 3.4) is very helpful in this incremental process to know which constraint to delete, where an object can be inserted, etc.

This incremental support is achievable if we guarantee easy and fast switching between the editing and presentation phases. This point is developed further in the following section.

4.2 Integrating the editing and presentation phases

Due to the intrinsic dynamic characteristic of a multimedia document, the famous static WYSIWYG paradigm, which

helps greatly in writing classical documents, cannot be applied to multimedia documents: it is not possible to specify a dynamic behavior and to see its result at the same time. As a consequence, the editing and presentation phases are not the same. However, they must be really integrated to be as close as possible to the WYSIWYG paradigm. This means that, firstly, the time to toggle between the two phases must be as short as possible; secondly, the frontier between these two phases must disappear.

In Madeus, the first point is achieved thanks to the use of a unique temporal structure (HSTP) to manage both editing and presentation phases, and to the design of efficient incremental algorithms associated with this structure.

The current editing process in Madeus requires the author to create a document by editing the textual format. However, during the presentation phase the author can pause the presentation, change the spatial position of an object or add/remove a spatial constraint, and finally save the document and replay the new version. In other words, it means that the window in which the document is presented is also an editing view. Our experiment shows that this kind of integration between the editing and the presentation phases is very satisfying for the author. We plan to provide the author with the same service for the temporal dimension, but the problem is much more complex due to the difficulty for the author to have a global perception of the temporal organization. In fact, a complete authoring environment of multimedia documents must be a multi- views one, with synchronizations between views and complex navigation functionalities [7].

4.3 Reusing multimedia specifications

The reusability of parts of- or entire- existing documents can remarkably save time while creating multimedia documents. In Madeus, it is possible to reuse a composite object in another context. The important point is that a composite object in Madeus is not a fixed object but an adjustable one, i.e neither its duration nor the temporal position of the objects it contains are fixed. Once again, this is due to the use of constraints. The benefit is that the author can reuse a composite object in different contexts.

Another kind of reuse occurs when the author wants to change some basic objects of an existing document while preserving its temporal and spatial organization. One typical case of such a situation is the translation of a document from one language into another. The author has to replace each textual message and each audio comment. There is a high probability that the durations of the new audio objects will differ from those of the existing ones. The constraint technology saves the author the effort of modifying the temporal organization to adjust such documents.

5 Benefits for presentation

5.1 Portability of multimedia documents

In the multimedia domain, portable standard formats of media objects are developing, such as MPEG audio, MPEG video, GIF, PNG and JPEG images, with players available on most existing platforms. Madeus takes advantage of these formats and combines them in documents using a high level, portable language. Thanks to its XML structure, Madeus language provides the easiness of importing from- and exporting to- other standard languages such as SMIL [17] (without any guarantee to preserve the whole semantic of the initial language). We are currently implementing in the Madeus environment an import function from SMIL documents. It allow us to use the Madeus presentation engine to play SMIL documents.

Both the language and its parser are platform independent. In this language, media objects are addressed by their URLs to comply with the designation standards used in distributed web-based applications.

Moreover, the presentation scheduler, that uses the output of the parser, is also platform independent. However, the implementation of the synchronization actions invoked by the scheduler is inevitably platform dependent as these actions require system-dependent functions such as access to system clock, control of system timers, event handler, etc.

5.2 Extensibility

Authoring environments of multimedia documents must have the ability of easily integrating newly emerging types and formats. In Madeus, extensibility is achieved by its object model that dynamically (i. e. during an editing or presentation session) accepts new types and formats. Extensibility is also provided by plugins which are represented in the object model as one type of media objects, allowing Madeus to make use of the wide variety of available plugin applications.

5.3 High level temporal navigation

Two types of temporal navigation have been introduced in Madeus for providing interactivity and global perception of documents. *Temporal navigation* falls into two main classes: Context *dependent* navigation that depends on the topic being presented and the document structure, i.e. its logical and temporal structures, and Context *independent* navigation that represents VCR-like controls to play, pause, resume, fast forward and rewind the document presentation.

Such navigation facilities allow the user to navigate through the temporal dimension of documents in order to explore rapidly their content.

These navigation facilities are implemented in Madeus by the help of the information supplied by the presentation context structure (described in 3.8). Starting from a consistent formatted scenario, the presentation context structure together with the set of Tic and Tac instants deduced from the hypergraph, allow us to implement various kinds of context-dependent navigation: *step by step* navigation where the user can navigate to-and-fro between the Tac instants, *structural* navigation where the user can navigate between objects located at the same hierarchical level of the hypergraph, and *user defined* navigation where the user, whether author or reader, can select a group of presentation instants, i.e. Tic instants, which seem interesting to him, and use them as functional navigation anchors.

6 Future Work

One of the main goals of the presentation engine of Madeus is to dynamically adapt to the current presentation conditions. The presentation conditions can be negatively affected by the indeterminism of the played media objects such as user interactions and opaque objects, and by the system and network loads. Constraints cannot statically guarantee that distributed presentations are satisfied under such conditions, but we believe that dynamic supervision can enhance the quality through temporal prediction and constraints re-evaluation. An adaptive scheme is required to control the document presentation with respect to a global clock.

The hypergraph structure, given the rich information it contains, can simplify to a great extent the design of an adaptive scheduling scheme. Information about the temporal ordering (edges in the HSTP) can be used in an observation and prediction loop: the information recorded at each observation instant can be used to predict what objects to play in the future and what objects durations to modify. Such a technique will help in minimizing effects such as jittering and skewing.

A sample of the issues that we like to address in the near future is the enhancement of the edition/presentation graphical user interface, which is currently under prototyping and the extension of scenarios to manage templates of documents.

7 Conclusion

Madeus is an authoring tool integrating authoring and presentation of multimedia documents. It provides document authors with an efficient and flexible way to specify a multimedia document while retaining established declarative languages for temporal synchronization and spatial positioning. It is currently experimented in a medical framework to build medical records.

In this paper, we have laid out the design principles of Madeus which is based on a constraint approach. We have also showed that the constraints concept is appropriate for high level document specification and provides the required transparencies and abstractions. First, at the interface level, constraints allow the combination of intuitive graphic representations of scenarios, like timelines, with the ease of modification by automatic updates through constraint propagation. Second, our model maintains documents' consistency in order to prevent authors from introducing errors when writing complex presentations. Constraints are based on a powerful representation of scenarios as graphs. Graphs are not only suitable for authoring but also for scheduling and time-based navigation.

References

[1] ALLEN J., "Maintaining Knowledge about Temporal Intervals", *CACM*, vol. 26, no. 11, pp. 832-843, November 1983.

[2] BUCHANAN M., ZELLWEGER P., "Automatic Temporal Layout Mechanisms", *Proc. of ACM Multimedia 97 Conference*, pp. 341-350, Anaheim, August 1993.

[3] SELCUK CANDAN K., PRABHAKARAN B., SUBRAHMANIAN V.S., "CHIMP: A Framework for Supporting Distributed Multimedia Document Authoring and Presentation", *Proc. of ACM 96 Multimedia Conference*, pp. 329-339, Boston, November 1996.

[4] DECHTER R., MEIRI I., PEARL J., "Temporal Constraint Networks", *Artificial Intelligence*, vol. 49, pp. 61-95, 1991.

[5] GEYER I., BAENTSCH M., BAUM L., ROTHKUGEL S., STURM P., "MHEG in Java - Integrating a Multimedia Standard into the Web", *Poster Proceedings 6th International World-Wide Web Conference*, April 1997.

[6] JOURDAN M., LAYAIDA N., ROISIN C., *A survey on authoring techniques for temporal scenarios of multimedia documents*, Published in Handbook of Multimedia, CRC Press, April 1998.

[7] JOURDAN M., ROISIN C., L. TARDIF, "Multiviews Interfaces for Multimedia Authoring Environments", *Proc. of 5th Conference on Multimedia Modelling*, Lausanne (Switzerland), October 1998.

[8] KHALFALLAH H., KARMOUCH A., "An architecture and a data model for integrated multimedia documents and presentation applications", *Multimedia Systems*, vol. 1995, no. 3, pp. 238-250, 1995.

[9] KIM M.Y., SONG J., "Multimedia Documents with Elastic Time", *Proc. of ACM 95 Multimedia Conference*, pp. 143-154, San Francisco, November 1995.

[10] LAYAÏDA N., SABRY-ISMAIL L., "Maintaining Temporal Consistency of Multimedia Documents Using Constraint Networks", *Multimedia Computing and Networking 1996, M. Freeman, P. Jardetzky, H. M. Vin, ed., pp. 124-135, SPIE 2667*, San-Jose, February 1996.

[11] SAADE D. C. M., SOARES L. F. G., COSTA F. R. , SOUZA FILHO G. L., "Graphical Structured-Editing of Multimedia Documents with Temporal and Spatial Constraint", *Proc. of 4th Conference on Multimedia Modelling*, pp. 279-295, World Scientific Publishing, Singapore, 18-19 November 1997.

[12] SABRY-ISMAIL L., ROISIN C., LAYAIDA N., "Navigation in Structured Multimedia Documents using Presentation Context", *Hypertextes et Hypermédias*, Hermès, Paris, September 1997.

[13] SANNELLA M., MALONEY J., FREEMAN-BENSON B., BORNING A., "Multi-way versus one way constraints in user interfaces: Experiences with the Deltablue algorithm", *Software Practice and Experience*, vol. 23, no. 5, 1993.

[14] VAN ROSSUM G., JANSEN J., MULLENDER K. and BULTERMAN D., "CMIFed: a Presentation Environment for Portable Hypermedia Documents", *Proceedings of ACM 93 Multimedia Conference*, California, 1993.

[15] WIRAG S., "Modeling of Adaptable Multimedia Documents", *Proceedings of European Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services (IDMS'97)*, Darmstadt, September 1997.

[16] W3C, *Extensible Markup Language (XML) 1.0*, W3C Recommendation, http://www.w3.org/TR/1998/REC-xml-19980210, February 1998.

[17] W3C, *Working Draft specification of SMIL (Synchronized Multimedia Integration Language)*, http://www.w3.org/TR/WD-smil, February, 1998.

[18] YU J. and XIANG Y., "Hypermedia Presentation and Authoring System", *6th International World Wide Web Conference*, April 1997.