HAL
archives-ouvertes.fr

# Presentation Services in MADEUS: an Authoring Environment for Multimedia Documents

Muriel Jourdan, Nabil Layaïda, Loay Sabry-Ismail

**To cite this version:**

Muriel Jourdan, Nabil Layaïda, Loay Sabry-Ismail. Presentation Services in MADEUS: an Authoring Environment for Multimedia Documents. [Research Report] RR-2983, INRIA. 1996. <inria-00073715>

**HAL Id: inria-00073715**

**https://hal.inria.fr/inria-00073715**

Submitted on 24 May 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

# *Presentation Services in MADEUS: an Authoring Environment for Multimedia Documents*

Muriel Jourdan, Nabil Layaïda and Loay Sabry-Ismail

**N ˚ 2983**

THÈME 3

Rapport de recherche

ISSN 0249-6399

# Presentation Services in MADEUS: an Authoring Environment for Multimedia Documents

Muriel Jourdan, Nabil Layaïda and Loay Sabry-Ismail

Thème 3 — Interaction homme-machine,
images, données, connaissances
Projet Opéra

Rapport de recherche n˚ 2983 — — 22 pages

**Abstract:**
The recent advances in multimedia systems, together with the advent of high speed networks, paved the way to a new generation of applications. In particular, authoring environments have found in multimedia the means of increasing the richness of information contained in electronic documents.

One of the goals of the Opera team is designing an authoring environment for multimedia documents, called MADEUS, which meets the following requirements: a high level of expressiveness for both spatial and temporal dimensions; a user-friendly interface allowing highly interactive design process, scriptless and structured-based editing and automatic production of spatial and temporal layout; the portability and cross-platform interchange of multimedia documents.

To achieve this research goal we first focus on finding a good representation of time for multimedia documents. This representation is required to capture the temporal dimension of media objects like video, audio, etc. and is also used to temporally organize objects with respect to each other. Time representation is clearly the main difference between multimedia authoring environment and traditional editing environments.

We present in this paper our first investigational results in our experimental authoring environment MADEUS. A large part of these results are based on the experience acquired in implementing the MADEUS prototype

**Key-words:** authoring environment , time representation, time management

*(Résumé : tsvp)*

Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN (France)
Téléphone : (33) 76 61 52 00 – Télécopie : (33) 76 61 52 52

# Services de présentation dans MADEUS : un environnement d'édition- présentation de documents multimédia

**Résumé :** L'explosion des capacités de traitements multimédia des ordinateurs rend le besoin en environnements d'édition de documents multimédia de plus en plus flagrant. L'un des objectifs du projet Opéra est de concevoir un tel environnement qui concilie les trois exigences suivantes : un haut degré d'expressivité pour exprimer l'ordonnancement temporel et l'organisation spatial d'un document; une interface d'édition interactive et déclarative (pas d'utilisation de scripts), qui prenne en charge les fonctions de formatage spatial et temporel; un environnement portable et des documents pouvant etre utilisés par d'autres applications.

Pour cela, nos efforts ont tout d'abord porté sur la définition d'une bonne représentation de la dimension temporelle d'un document multimédia. Celle-ci doit à la fois prendre en compte les différentes caratéristiques temporelles des objets de base d'un document multimédia et les différents types de composition possibles entre ces objets. De plus, elle doit servir de support à des mécanismes de présentation efficaces.

Nous présentons dans ce rapport nos premiers résultats. Une grande partie de ceux-ci ont été expérimentés au sein du prototype MADEUS.

**Mots-clé :** environnement d'édition, documents multimédia, présentation multimédia

# 1 Introduction

The recent advances in multimedia systems, together with the advent of high speed networks, paved the way to a new generation of applications. In particular, authoring environments have found in multimedia the means of increasing the richness of information contained in electronic documents.

One of the goals of the Opera team is designing an authoring environment for multimedia documents, called MADEUS, which meets the following requirements:

- a high level of **expressiveness** for both spatial and temporal dimensions;
- a **user-friendly interface** allowing highly interactive design process, scriptless and structured-based editing and automatic production of spatial and temporal layout;
- **the portability and cross-platform interchange** of multimedia documents;

To achieve this research goal we first focus on finding a good representation of time for multimedia documents. This representation is required to capture the temporal dimension of media objects like video, audio, etc. and is also used to temporally organize objects with respect to each other. Time representation is clearly the main difference between multimedia authoring environment and traditional editing environments.

We present in this paper our first investigational results in our experimental authoring environment MADEUS. A large part of these results are based on the experience acquired in implementing the MADEUS prototype. More precisely, the first section is devoted to the analysis of what are the requirements for a good representation of time in multimedia documents. At the end of this section, we present the approach used in MADEUS which is based on an extension of temporal constraint networks. In the second section, we focus on the design and architecture issues of the MADEUS application which handles both authoring and presentation stages of multimedia documents.

# 2 Representation of time in multimedia documents

## 2.1 Time representation requirements

In order to measure the quality of a given time representation used to model the temporal organization of multimedia documents and to evaluate comparatively those used in different document systems [2][3][12], we need to define some criteria. In what follows, we present key points that, in our opinion, affect significantly the quality of the temporal representation of multimedia documents.

**Expressiveness** The time representation must capture the different temporal characteristics of basic objects . Media objects may be static like text, still images, etc., or dynamic like video, audio, interaction, etc. Dynamicity is taken here in the sense that these objects have a notion of a presentation activity over time. A video or an audio are active when they are played. An interaction, like a button, is active when it is enabled so that it is possible to interact with it through the user interface. Dynamic objects have an associated duration where two cases must be distinguished according to their temporal behaviour:

- **Controllable** objects: a range of possible durations is known at the authoring stage and it is possible to choose any value within this range during the presentation stage (audio, video, ... ).;
- **Uncontrollable** objects: their durations are only known at the end of their presentation (interaction objects, uncontrollable program executions like applets).

The time representation must also capture the different ways to temporally organize objects [16][13]. Firstly, it should be possible to distinguish four temporal instants when dealing with dynamic objects: when they are mapped (unmapped) at the screen (referred in the sequel as spatio-temporal instants) and when they start (finish) playing (referred in the sequel as temporal instants). Static objects are only affected by the spatio-temporal instants. Secondly, operators which express the temporal organization of the document must support logical time: relative positioning of objects (as expressed by the thirteen temporal relations introduced by Allen [1]); quantitative time: numerical constraints placed on object's duration and causal time: how instants causes other presentation action to happen (the end instant of one object causes the termination of another). For instance, the following specification (scenario 1) of a temporal organization must be supported by the time representation :

*Map the two videos A and B 2 seconds after the start of the document; Start playing video A 2 seconds after being mapped; Play video B in parallel to video A; Unmap the two videos when they are finished; Start playing audio C when video B starts to play; Play video D after video A; Map D when it is played; Unmap it at the end of the document; Start an interaction I when D starts; The end of video D causes the end of audio C if not yet finished; The first one that finishes between D and I, kills the other one. The duration of the document must be in the range between 20 and 30 seconds.*

The four objects A, B, C and D have the following temporal characteristics, defined by a tuple (min, d, max) where min and max are respectively the minimum and the maximum durations of the object, and d is the preferable one in the context of a given document: A: (4,6,8), B: (2,4,6), C : (10,11,12), D: (4,6,8).

Finally, the time representation must be able to deal with temporal hyperlinks which gives the document readers the ability to navigate freely either inside the document or outside it. Hyperlinking breaks the linearity of planned scenarios and allows semantical exploration of the document content beyond its structure.

**Interface with a declarative and hierarchic symbolic representation** Multimedia document authors do not necessarily have any particular skills in computer programming, so the editing phase must be very intuitive and as close as possible to the familiar editing style of classical documents. This goal is obviously hard to obtain when using scripting languages where the document architecturing scheme (manipulation of the document entities through a set of associated attributes) disappears in favour of the programming one. Defining global attributes on composite objects like color, font, position, etc. and applying inheritance rules on these attributes reduce the burden on document specification, increase the reusability and hide low-level presentation details. Existing document models based on declarative style [6] and descriptive mark-up of its content are more and more widely used. We think that this is the best approach in order to disseminate time-based multimedia documents. Moreover, it is also suitable for direct manipulation authoring interfaces.

The symbolic representation of time in multimedia documents must support structural composition[10]. This gives the author the ability to organize the document in nested components in order to divide its content in logical parts[14]. It helps him also in focusing on an isolated part of the document during each step of the authoring process (sub-scenes of a multimedia scenario).

**Efficient and incremental manipulation of the document** The manipulation of the temporal structure of a multimedia document in an authoring environment must be supported by efficient algorithms and data structures in order to be really interactive. Moreover, given the progressive nature of the editing process and the accesses across the network to the document content [17], the temporal organization of a document must be suitable for an incremental building process. The critical parts of an authoring environment regarding the time performances are:

- **parsing phase**: translation of the symbolic representation of the temporal structure of the document into the internal and executable one used by the presentation layer of an application engine;
- **consistency checking** : as defined in [11], three kinds of temporal inconsistencies may occur in a document: qualitative, quantitative and indeterministic. Inconsistencies may also occur from the relative positioning of spatio-temporal instants according to temporal ones. For instance a dynamic object can not be started before being mapped on the screen. All kinds of inconsistencies have to be detected at the authoring stage and the parsing one [11]. This checking phase prevents the author from writing inconsistent temporal specifications and the presentation layer from executing inconsistent scenarios.
- **edition/presentation cycle**: the transition from the editing phase to the presentation one which affects the development of integrated tools;
- **presentation monitoring**: management of the presentation phase which handles layout operations like graphics rendering, network access, temporal access control operations like pause, resume, stop and fast-forward, temporal navigation through hyperlinks, etc.

**Portability and cross-platform documents interchange** The representation of time in multimedia documents must not make particular assumptions about the format of the basic objects (video, audio, text, etc.), the operating system of the target machine like multi-threaded processing, the presence of dedicated multimedia hardware, proprietary media object formats, etc. This is the necessary condition to meet the portability requirements of multimedia documents interchange format [3].

Moreover, the representation of time must be such that documents interchange between different environments becomes easy. This is the aim of high-level symbolic formalisms as Hytime [7], MHEG [9] and more recently PREMO [8]. Documents written in these formats are preserved along time and are reusable by different people and applications.

We have seen in this section a set of requirements which serves as basis and guidelines for the choice of a time representation for multimedia documents. In the two following sections, we present the solution taken in our project, its advantages and an analysis of particular solutions used in other well known systems like the Timeline editors[5], OCPN[12] and CMIFED[3].

## 2.2 Time representation in MADEUS: an extension of temporal constraint networks

The internal representation of time in MADEUS is based on the temporal constraint networks formalism defined in the artificial intelligence area [4]. We use this formalism during the authoring stage in order to detect easily inconsistencies in a first step and to propagate numerical constraints in a second one [11]. Constraint networks are powerful because they are based on a data structuring of the temporal constraints as a graph. This graph matches the data structuring required by the presentation stage to monitor the presentation.

An intermediate operation is applied between the consistency checking and the constraint propagation stages which is called static temporal pre-formatting. This operation is necessary because constraint propagation deals only with minimum and maximum values of allowable durations (a set of solutions for the scenario). The pre-formatting phase takes into account delays and objects shrinking and stretching capabilities in order to find particular solutions. This process uses a constraint distribution policy to balance constraints between intervals taking into account the most suitable duration of the objects. Because durations of uncontrollable objects are only determined during the presentation stage, a dynamic formatting is also needed [2]. This last point will be explained in detail in section 3.2.3.

### 2.2.1 Temporal Constraint networks

A temporal constraint network is a Directed Acyclic Graph (DAG) where nodes represent time points and an 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*.

The topology of this DAG is organized as a set of temporal chains. A temporal chain $[\mathbf{e_1}, \mathbf{e_2}, .., \mathbf{e_n}]$ is a sequence of contiguous edges where each edge $\mathbf{e_j}$ , $\mathbf{1} < \mathbf{j} < \mathbf{n}$ is only

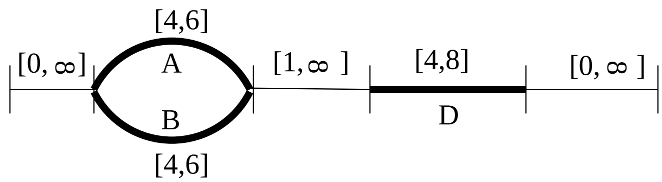



Figure 1: Constraint network of the scenario 2

related to one successor and one predecessor so that **begin** $(\mathbf{e_j})$ = **end** $(\mathbf{e_{j-1}})$ and **end** $(\mathbf{e_j})$ = **begin** $(\mathbf{e_{j+1}})$, every single interval controllable or uncontrollable.

A temporal constraint network describes the temporal ordering of media objects by capturing an information about their logical and quantitative temporal dependencies. The range of possible durations of a dynamic object is modeled by an edge labeled by its appropriate interval. Free or quantified delays are used to measure the temporal metric distance between objects or to enforce precedence relations between them. Formally, the class of problems modeled by this formalism is called Simple Temporal Problem [15]. For instance, the following scenario (scenario 2) can be managed during the authoring stage by the temporal constraint network shown in figureFig. 1.

*Play video B in parallel to video A; Play video D after video A*

In this figure interval objects are represented by bold lines and delay intervals by thin ones. A and B duration ranges have been modified to satisfy the specification.

### 2.2.2 Extended temporal constraint networks

In MADEUS, the previous definition of a constraint network has been extended to meet the requirements introduced above:

- an additional labelling of interval which indicates whether the temporal distance is controllable or not (respectively $[\text{min,max}]^c$ and $[\text{min,max}]^u$). This is the way we manage indeterministic temporal behaviour as explained latter in the dynamic formatting section.
- causality relations between temporal instants on each node to handle master-slave relationships between media streams.
- a set of *spatio-temporal* actions (map and unmap actions) on each node to control the instants when the different objects appear or disappear from the screen independently from their play time. This opens the room for the development of spatio-temporal composition and languages which are still lacking in multimedia authoring systems.
- basic objects of type hyperlinks have been added as temporal intervals corresponding to the period of time where their activation is enabled. This representation of hyperlinks

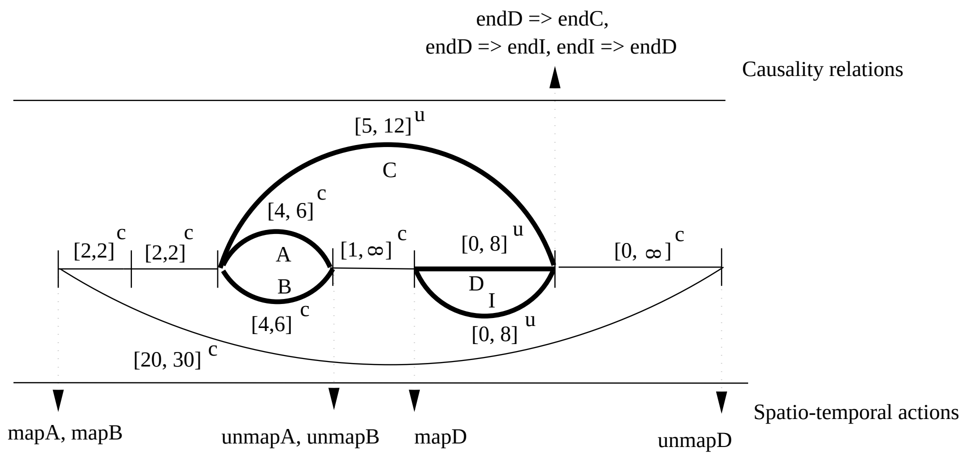



Figure 2: The constraint network of scenario 1

is homogeneous with the representation of the other types of objects in the temporal constraint network.

In addition to this constraint networks structure, we define a hierarchical structure of nested components where leaves are basics media objects and nodes are composite objects. In fact, each composite object is defined by an extended constraint network built from its enclosing components by a set of relational operators. It is worth noting that the hierarchical organization of the document does not restrict the degree of expressiveness of temporal scenarios. As a result of the orthogonality of the temporal dimension with respect to the logical organization, media objects can be involved in several relations. Because one can not predict how objects are inserted in the scenario, the hierarchical structure defines temporal boundaries to sub-components serving as an overall begin and end of the sub-scene. The network associated with composite objects constitutes the main building block of our internal representation of a multimedia document. Each composite representing a level of the hierarchy defines a new level of constraint network. Through this hyper-graph, temporal constraints can be propagated up-stream or down-stream depending on whether the editing operations modify the temporal parameters of a basic object or a composite object as a whole.

The extended constraint networks associated with the scenario 1 during the authoring stage is given in Fig. 2. As described in the example, some intervals of this scenario become uncontrollable by inheritance (like C and D due to their interaction with I). This scenario is temporally consistent since it is possible to find particular duration values of controllable intervals, such that, whatever value the duration of the uncontrollable interval I takes at the presentation stage, the last controllable interval can be dynamically adjusted to compensate the observed indeterminism [11].

### 2.2.3 Evaluation

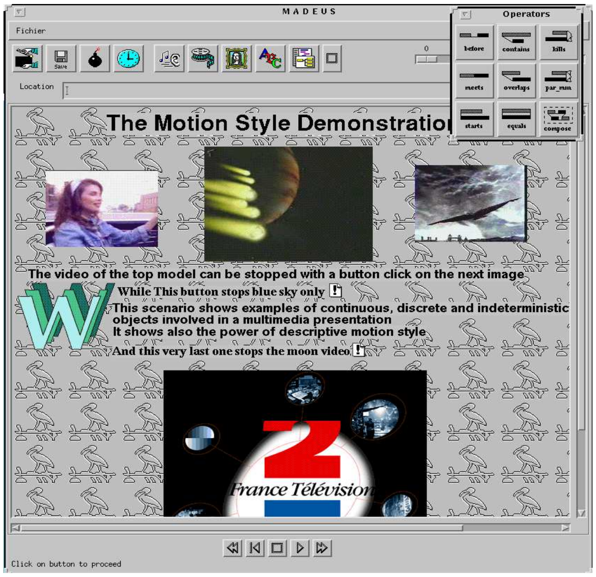
In this section, we present the advantages of the extended temporal constraint networks as a choice for a better time representation. These advantages are evaluated regarding the requirements that we have introduced earlier.

**Expressiveness** Our time representation is able to deal with static and dynamic objects, controllable and uncontrollable ones. It takes into account the spatio-temporal composition by labelling the network nodes by spatio-temporal mapping actions. The set of the thirteen primitive relations introduced by Allen are fully supported and extended to take into account the quantitative aspect of the relations. The pointizable time algebra [15] is fully implemented to cope with arbitrary complex temporal presentations, and the set of temporal operators can be redefined without modifying the temporal management layer [16]. It is also possible to express causality relations between objects like parallel presentation of two objects where the first to finish terminates the other one. Moreover, hyperlinks (navigation support) and user interactions are integrated in the temporal language in a homogeneous way as intervals.

**Interface with a declarative and hierarchic symbolic representation** Our time representation is interfaced with a declarative and hierarchic symbolic relation based on a set of relational operators (start, meet, equal, par-min, kills, ...) as illustrated by Fig. 3. A composite object is defined by three parts: a declaration part describing the enclosed objects together with their attributes like colors, fonts, motion style and video effects. A second part describing temporal relations which expresses the ordering between these objects. Finally, a third part describes the spatio-temporal actions of the different objects. All these specifications can be easily performed through the user interface by selecting objects and the temporal or spatio-temporal relations between them.

**Efficient and incremental algorithms** Our time representation is handled by an incremental editing process [11]. The consistency checking is achieved by three different methods depending on the kind of temporal inconsistencies:

- **static**: a topological sort of the constraint network is maintained in order to avoid this kind of inconsistency by detecting cycles.
- **quantitative**: by using constraints propagation techniques, each time the network is modified, we check that all chains parallel to each other still have non-empty intersection of their allowable duration ranges. Algorithms used for the constraint propagation are polynomial ones [15]. Moreover, they take advantage of the topology of the graph organized as temporal chains to achieve local propagation and fast updating techniques.
- **indeterministic**: a check is applied statically to see if it is possible to recover dynamically, at the presentation stage, the indeterministic behaviour of uncontrollable




```
<DOC>Scenario1
   <COMP>Scene1
      <DECL>
         <OBJECT>
            <NAME> A
            <URL>    http://opera.inrialpes.fr/Logo.Mpeg1
            <MOTIONSTYLE>{right_to_left : 150 pt,0}
            <POS>  252,108
            <FORMAT>1
         </OBJECT>
         ...
      </DECL>
      <TEMP_REL>
         A equals B
         D kills C
         ...
      </TEMP_REL>
      <SPATIO_TEMP_REL>
         A.start maps B
         ...
      </SPATIO_TEMP_REL>
   </COMP>
   ...
   <TEMP_REL> ...  </TEMP_REL>
   <SPATIO_TEMP_REL> ...  </SPATIO_TEMP_REL>
</DOC>
```

Figure 3: MADEUS graphic user interface and symbolic document representation

intervals. This operation is performed to ensure the dynamic formatting which is based on an observe and repair approach of the indeterminism (see section 3.2.3).

Another advantage we provide with our internal representation of the document is that it is suitable as a support for both edition and presentation support. Thus, the MADEUS application engine does not require any structural translation between the edition phase and the presentation one.

**Portability and cross-platform document interchange** One of the main advantages of our time representation is that it is interfaced with a symbolic declarative representation which can be directly used as an interchange format. Also, it can be easily translated in any other format used by another application given its higher level of expressiveness.

## 2.3 Analysis of other time representations

Timeline editors like Director [5], Object Composition Petri Nets [12] and CMIFED [3] time representations are analyzed regarding how they meet the requirements presented in the previous section. First of all, let us note that none of them distinguishes clearly the spatio-temporal aspects of a multimedia presentation from the temporal ones. Thus the kind of possible written scenarios is limited by comparison with our approach.

- **Timeline** editors: the expressiveness of timeline is not very rich since it does not capture the shrink and stretch capabilities of dynamic objects, uncontrollable objects and hyperlinks are not easy to integrate (use of a scripting language) and only quantitative dating scheme is supported. Interfaces based on timeline are rather intuitive but not scalable for large documents. The lack of structure is also one of the drawbacks of timeline editors. It puts a heavy burden on the authoring process and is error prone since any modification of the scenario have to be propagated by hand to the rest of the document. However, the simplicity of the time representation eases the application construction. The document interchange requirements are not satisfied since an exhaustive dating of the objects presentation does not capture their mutual relationships (there is no logical time).

- **OCPN**: this model handles logical and quantitative time but does not support the causal one. However, from our understanding it is not stated how numerical constraints are managed and propagated. Uncontrollable objects and hyperlinks are not supported. Graphical authoring interface based on petri-nets allowing hierarchical structuring can be defined. However, this temporal modelling of multimedia documents is not appropriate as an authoring paradigm. The temporal alignment of media objects and the document paradigm (entity attribute editing) disappear in favour of the transitional aspect of petri-nets.

- **CMIFed** representation of time: only two causal time operators are defined in CMIFed, the sequence operator and the parallel one where the last to finish ends the temporal construction. The CMIFed editor has a user friendly interface based on a logical structuring of the document as a hierarchy. However, this hierarchy is mixed with the temporal structure of the document where each composite element is defined by a temporal operator. Therefore, the temporal organization of the document is restricted to a tree structure, which together with the exclusively causal representation of time, limits the number of expressible scenarios. To overcome this problem, synchronization arcs have been added but inconsistencies can be easily introduced. A recent effort has been undertaken to meet the portability requirements and a translation of CMIFed documents into HyTime has been successfully achieved.

# 3 Presentation layer in MADEUS

In this section, we focus on the design and architecture issues of the MADEUS application which handles both authoring and presentation stages of multimedia documents. Particularly, we detail the presentation services related to the scheduling issues, namely the scheduler and the execution mediator levels, which are based on the hierarchy of extended temporal constraint networks.

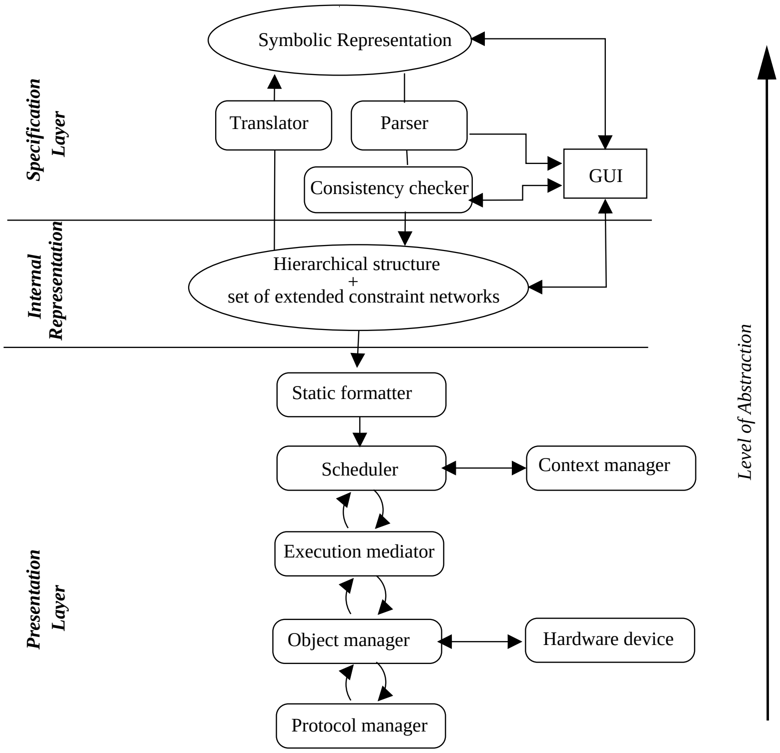



Figure 4: General architecture of MADEUS system

## 3.1 Architecture overview of MADEUS

The general architecture of the MADEUS system as shown in Fig. 4, depends on different levels of abstraction. The highest level is the **specification level**, which furnishes the system with a simple declarative specification language of high power of expressiveness to specify the required temporal relationships between different multimedia objects. The resulting set of declared multimedia objects together with the temporal relations between them, constitute a symbolic representation of the multimedia document. A **parser** is used to transform the symbolic representation shown in Fig. 3 in order to obtain the internal representation starting from which we can check consistency and monitor the presentation of the multimedia document. The author can incrementally edit and modify the symbolic representation or the internal representation of the document via a Graphical User Interface (GUI). In addition, there is a**translator** whose role is to convert the internal representation into the symbolic one which is the format in which the document will be stored on the disk.

The **scheduler level** (furtherly described in section 3.2) handles the synchronization between temporally related instants of different multimedia objects, like for example the start of the presentation of the whole document, the co-starting of a number of intervals simultaneously, the causality relationships between concurrent objects, user navigation through the hyperlinks, etc.

The **execution mediator level** (furtherly described in section 3.3) handles the state transitions of the multimedia objects as needed by the scheduler and requests the corresponding actions to be done by the executional lower level. This level guarantees the independence of the implementation layer. Also, it gives a feedback to the scheduler about the current state of the multimedia objects during their presentation life time.

The **object level** differentiates between the different types of objects while they are hidden in all of the above levels. If a certain action is to be done on an object, then the way of achieving it will be dependent on the type of this object and its implementation, so the object level helps in provoking the right procedure to execute this action based on the object's type. Also, this level manages the layout of the multimedia objects on their suitable presentation devices, such as audio channels, screen, etc.

The **network protocol level** handles the data access of multimedia objects. Firstly, it is to be specified if the object to be accessed is local or remote. This is achieved by a URL (Uniform Ressource Locator) like object naming scheme where the protocol appears as a prefix. As some media objects can tolerate loss of data like video and audio VDP, RTP and UDP based protocols are more adapted, while others cannot tolerate this loss like the source document representation, text, image and graphics so HTTP which is a TCP/IP based protocol is used. Whether the multimedia object is local or remote, in both cases the object is accessed via a stream of data managed independently from the object level. The network level manages operations like opennig, closing, buffering the network connections and callbacks for activities in the object level.

## 3.2 Scheduler

The scheduler can be considered as the maestro of the presentation phase of a multimedia document. It does not only handle the temporal synchronization between multimedia objects, but it handles the dynamic compensation for indeterminism as well. It also handles temporal hyperlinks and references by maintaining contexts of the document's presentation.

### 3.2.1 Presentation context and presentation stack

A presentation context, at a given instant, is the information about the current activities of a running presentation. The presentation context is maintained in the form of a list of activity identifiers. If we imagine that we have a presentation as shown by the graph in Fig. 5, and we make a temporal logical cut in this presentation represented by the curved line (say at time t). Then the presentation context will contain information about the activities of the currently active objects (A, B, C, and D). Also we can use a presentation stack in order to switch between different presentation contexts by storing the one to be suspended in the stack (after suspending all its activities). This facility enables us to free the resources allocated by a suspended presentation context as we have all the information

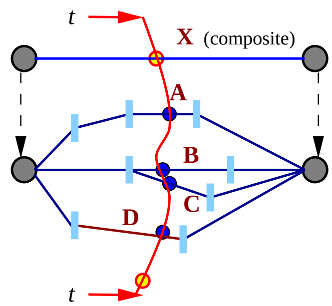



Figure 5: A temporal logical cut in a multimedia presentation

needed to reallocate resources in order to resume this context at any time (e.g. the current frame number for the video). This kind of simple check points covers most of the cases except for program executions (applets) where it is defined by the system run-time stack itself.

### 3.2.2 Scheduling principles

The scheduler uses the internal representation to monitor the document's presentation. At every node in the constraint network being reached by an ending interval, the scheduler applies a simple and straightforward rule having the following statement: "if all the incoming intervals of this node have reached their end, then execute all the spatio-temporal actions and start all the outgoing intervals". This kind of scheduling is done by the help of synchronization counters (at each node) for (1) the number of incoming and outgoing intervals at this node, and (2) the number of intervals remaining to terminate at this node during the presentation. The values of these counters and the related intervals are stored in the node structure as shown in Fig. 6. Also in this structure, we maintain tables representing the kind of dependencies (equality and causality) between instants of the incoming and outgoing intervals at this node. The equality dependency of instants implies their temporal coincidence which must be verified by the temporal consistency checking phase. For example in Fig. 6, the end instant of B coincides with start instant of interval D, and the end instant of A coincides with that of C. The causality dependency of instants implies that the occurrence of an instant leads to the occurrence of the other instant. The field called timestamp and its role in dynamic compensation for indeterminism, will be explained in a latter section.

The document's presentation starts from the starting node of the graph at the root of the hierarchical structure. A composite object is started by recursivly starting its components until basic media objects. It is ended when it receives end signals from all its components.

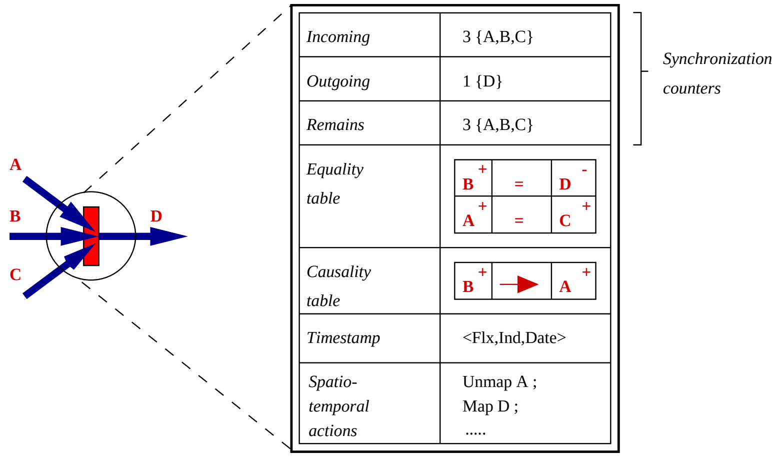


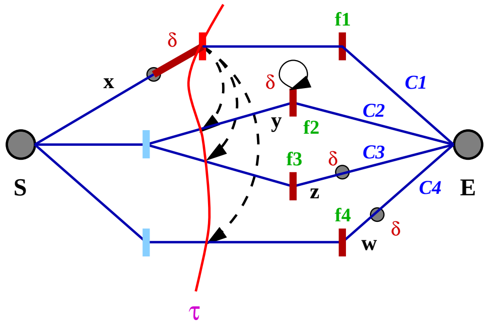

Figure 6: The structure of a node in the constraint network

### 3.2.3 Dynamic formatting

One problem that arises in the presentation of a multimedia document is the ability to handle the temporal indeterminism that occurs during the presentation. In order to deal with these situations and to meet the constraints specified in the scenarios, certain information about temporal flexibility, indeterminism and event dates, must be available at every end-of-play instant for every interval.This information is called the time stamping of instants. These instants correspond to the nodes of the temporal constraint networks.

The time stamp information at each node can be represented in the form of a tuple <FLX, IND, DATE>, where DATE is the date of termination of the last incoming interval at this node, FLX is the amount of flexibility available that can be provided by the outgoing interval of this node and along a particular temporal chain, and IND is the amount of uncompensated indeterminism intercepted in the current chain until the instant DATE. The value of indeterminism of an uncontrollable object is the difference between its effective duration and its static one determined by the static pre-formatter. The field DATE is calculated with respect to either a global document's presentation clock that measures time from the starting instant of the presentation of the document or simply a time-of-day clock. This clock is examined only when an event happens in the scenario so that the corresponding chain's clock can be checked against the clocks of the concurrent chains. This particular point of time represented by $\tau$ in Fig. 7 is a point of decision where effective duration of x can be determined being the difference between the DATE fields at this node and that of the starting node of x.

At time instant $\tau$, a table of chain information is handled where we have for each concurrent chain two entries:




IF $\delta > f1$

| Chain | CInd | CFlx |
|---|---|---|
| C1 | 0 | 0 |
| C2 | δ | f2 |
| C3 | δ | f3 |
| C4 | δ | f4 |

*Configuration of concurrent chains at logical time τ*

Figure 7: Dynamic formatting

1. the value of uncompensated indeterminism **CInd** resulting from **x**.

2. the available flexibility **CFlx** at the nearest future node on the chain.

Initially, the **CInd** field of each chain has a zero value and **CFlx** field is initialized with the amount of available flexibility on each chain until the next synchronisation point **E**. This information is defined also statically before starting the presentation, but it is updated dynamically during the presentation when used to compensate for indeterminism. It is essential to mark the amount of flexibility used for compensation as to be reserved in order not to be used again by another detected indeterminism in the future. In theFig. 7 an amount of indeterminism $\delta$ has been detected along the chain **C1**. Since there is not enough flexiblity **f1** along this chain ($\delta$>**f1**), the other chains are asked via the table of concurrent chains to recover from this delay. The chain **C2** notices this request at the end of the object **y** where its end is delayed (static object) while the objects **z** and **w** are stretched by the amount $\delta$ provided by their flexibilty **f3** and **f4** respectively.

We can conclude that the MADEUS's extended temporal constraint network provides a self adaptative mecanism handling unpredictable temporal behaviour. As we have seen in the example, this mecanism is based mainly on the organization of the temporal structure as a graph which in turn is organized as chains.

### 3.2.4 TAC management

By TAC (Time Access Control) management we mean all actions that may be requested by the user and lead to a change of state of a currently presented multimedia document. Examples of these actions are start, restart, pause, resume, kill, fast forward, and fast rewind of a multimedia document's presentation. To handle all these actions efficiently, the scheduler handles presentation contexts holding information about currently active multimedia objects and their associated system activities (thread, process, timer, etc.).

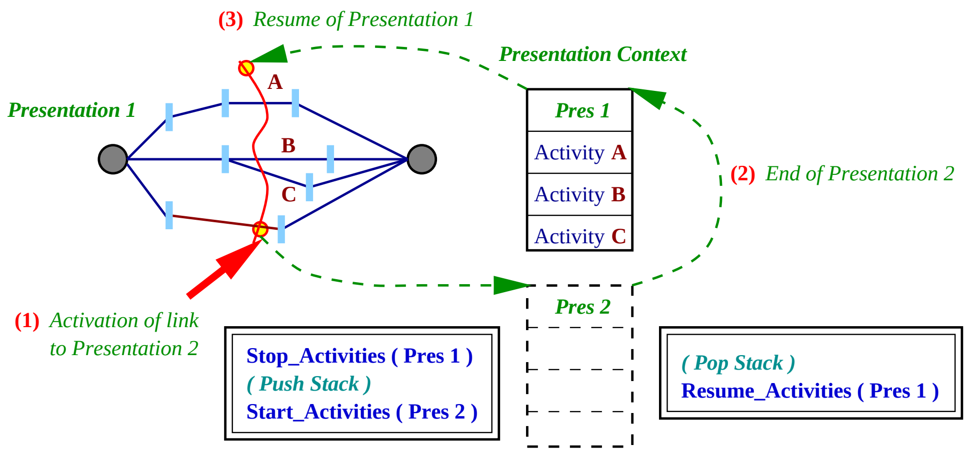



Figure 8: Management of presentation context

For example, when the user requests a pause of the presentation, the system suspends all of the concurrently running activities. It stores the value of their internal clock represented by the current presentation step and pushes the current presentation context in the presentation stack. When the user requests a resume of the presentation, the presentation context is popped out from the stack and all suspended activities belonging to this context are to be resumed once again. Also, the operations of fast forward and fast rewind are achieved similarly by modifying the speed and direction of the activities in the current context. Objects with hidden internal clocks modelled by uncontrollable intervals are simply ignored in the last two operations.

### 3.2.5 Temporal hyperlinks

Temporal hyperlinks means linking between different presentations in a context sensitive way depending on the semantics associated with this link. The linked presentations may be in the same multimedia document (internal references) or in different multimedia documents (external references). The scheduler handles this kind of referencing by managing the stack of presentation in a very simple way. Once a link is activated, the current presentation context (i.e. the presentation context of the source anchor) is pushed in the stack of presentation, and the presentation context of the destination becomes the current one. Once the presentation of the destination finishes, then the presentation context of the source is popped out from the stack of presentation in order to resume starting from the time point of link activation.

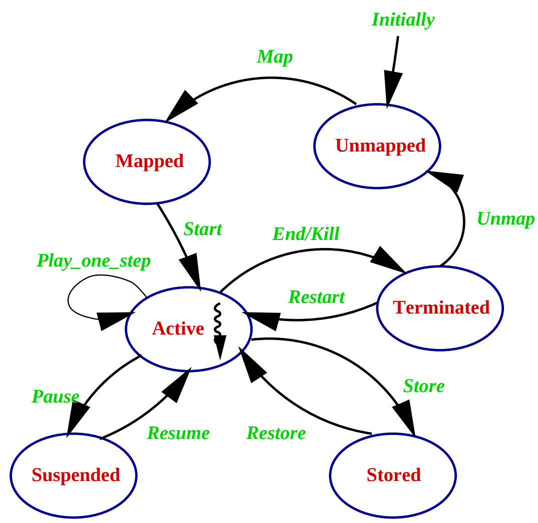



Figure 9: State transition diagram of multimedia objects

## 3.3 Execution mediator

The execution mediator maintains the states of the objects and handles their transitions between these states as shown in Fig. 9. The mediator treats all kinds of multimedia objects in the same way. The main states of the objects are: unmapped, mapped, active, terminated, suspended, and stored. The unmapped state is the initial state of all objects, i.e. the object has not yet allocated its required resources. In the mapped state, the object has allocated its required resources but its presentation has not been yet started. In the active state the system activities necessary for the object's presentation, are started. In the terminated state, the object's presentation has come to an end and it is ready to free its allocated resources. The suspended state means that the system activities associated to the object are suspended, i.e. its presentation is suspended but not yet ended.

Actually, we have different kinds of resources that can be classified as follows:

- spatial resources such as windows, pixmaps which are given unique identifiers by the window manager and the color resource.
- hardware resources such as video decompression cards and audio devices.
- system resources such as processes, threads and timers.

Once a hyperlink is activated, the currently running objects of the anchor presentation enters in the stored state where their current status information and clock are stored, for example the mapped state of an object or the current frame number of a video. As a result of storing this information, the system can free the resources allocated by these objects, so that when a return from the link takes place, the system can reallocate the necessary resources in order to play the objects starting from their status and internal clock at the time of link activation.

The play_one_step transition shown in Fig. 9, represents the case of taking into account the granularity of a multimedia object (for example frames of a video object) where with each unit of granularity of the object the system can make several actions, as for example moving the enclosing box of a video on the screen, changing the degree of brightness of the displayed video, displaying a frame of this video, and visual effects like stippling and bluring. This state transition can be also used to export internal events of an active object to the scheduler. These events can be synchronized with other media objects of the document (temporal projections [8]) if their description is provided by indexation tools.

In this section, we presented the different layers handling a multimedia presentation. The MADEUS prototype has been implemented on Sun workstations with C language. It handles media streams of various format like Mpeg video, Mpeg Audio, still pictures (GIF, JPEG, ..) and formatted text. We have created and tested a variety of testing and demonstating documents and performance enhancements are currently undetaken.

# 4 Conclusion

We have presented in this paper a framework for temporal representation and synchronization of multimedia documents. We have outlined some criteria to compare other proposed time representations and their limitations. We have also shown that the extended temporal constraint networks implemented in MADEUS are powerful and suitable for both multimedia authoring and presentation.

Among the characteristics of the MADEUS prototype, we have addressed the following aspects:

- the expressiveness of the temporal representation which captures the qualitative, quantitative, causal and indeterministic aspects of multimedia objects and presentations.
- a layered application architecture which allows the production of temporal layout, temporal navigation through hyperlinks and run-time monitoring of the presentation by a dynamic formatter.

From the implementation experience of MADEUS, we came to the conclusion that the bottleneck of performance is due to presentation layout. Operations like video decompression, graphics rendering and color quantization are the main source of temporal deviation from a planned scenario. This fact shows that treating temporal uncertainty by a self-adaptative management of temporal synchronization is a central issue.

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.

# References


[1] ALLEN J.F., "Maintaining Knowledge about Temporal Intervals", *Communications of the ACM,* vol. Vol. 26, num. No. 11, pp. 832-843, november 1983.

[2] BUCHANAN M.C., ZELLWEGER P.T., "Automatic Temporal Layout Mechanisms", *Proceedings of the First ACM International Conference on Multimedia,* pp. 341-350, Anaheim, California, august 1993.

[3] BULTERMAN D.C.A., VAN ROSSUM G., VAN LIERE R., "A Structure for Transportable, Dynamic Multimedia Documents", *USENIX Conference,* pp. 137-1559, Nashville, TN, june 1991.

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

[5] DIRECTOR 4.0, "User's Guide", Macromedia.

[6] GOLDFARB C. F., *The SGML Handbook,* Oxford University Press, Oxford, 1990.

[7] ISO/IEC 10743, *HyTime Information Technology Hypermedia/Time-based Structuring Language (HyTime),* november 1992.

[8] ISO/IEC 14478-1, *Information Processing Systems – Computer Graphics and Image Processing – Presentation Environments for multimedia Objects (PREMO),* num. 1, may 1996.

[9] ISO/IEC 13552-1, *Information Tehnology Coded Representation of Multimedia and Hypermedia Information Object (MHEG)* , num. 1, october 1994.

[10] KHALFALLAH H., KARMOUCH A., "An architecture and a data model for integrated multimedia documents and presentational applications", *Multimedia Systems,* pp. 238-250, Vol. 3, 1995.

[11] 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-José, USA, february 1996.

[12] LITTLE T.D.C., GHAFOOR A, "Interval-Based Conceptual Models for Time-Dependent Multimedia Data", *IEEE Transactions on Knowledge and Data Engineering (Special Issue: Multimedia Information Systems),* vol. Vol. 5, num. 4, pp. 551-563, august 1993.

[13] PEREZ-LUQUE , LITTLE T.D.C., "A Temporal Reference Framework for Multimedia Synchronization", *Proceedings of Sync'95,* may 19, Virginia, 1995.

[14] QUINT V., VATTON I., “Grif: An Interactive System for Structured Document Manipulation”, *Text Processing and Document Manipulation, Proceedings of the International Conference,* pp. 200-213, Cambridge, 1986.

[15] VILAIN M., KAUTZ H. A., “Constraint Propagation Algorithms for Temporal Reasoning”, *Proceedings of AAAI,* pp. 377-382, Philadelphia, august 1986.

[16] WAHL T., ROTHERMEL K., “Representing Time in multimedia”, *Proceedings of the International Conference on Multimedia Computing and Systems,* pp. 538-543, Boston, 1994.

[17] WEITZMAN L., WITTENBURG K., “Automatic Presentation of Multimedia Documents Using Relational Grammars”, *Proceedings of the Second ACM International Conference on Multimedia,* pp. 443-451, San Francisco, California, october 1994.

# Contents

Unit´e de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unit´e de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unit´e de recherche INRIA Rhône-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unit´e de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unit´e de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
ISSN 0249-6399