Constraints Techniques for Authoring Multimedia Documents

Muriel Jourdan, Cécile Roisin 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/

1 Introduction

A multimedia document is defined as a set of objects from different media (text, image, video, audio) that are spatially and temporally organized and on which a navigational structure can be set. Such an entity can be rendered thanks to a presentation engine by means of the output channels of the computer (screen and speakers).

Numerous works [5], [19] and even standards [26], [16] have been done for the definition of languages and formats of multimedia documents, largely focusing on the temporal dimension of documents. They allow the specification of the temporal composition of media objects either by absolute placements [14], by event−based approaches [16], by the use of a hierarchy of temporal operators [19], by constraint based definitions [5], [11], [12], [2] or a combination of some of these methods [26].

In constraint based environments, the author can describe the spatial and temporal organization of a document by setting constraints between basic or composite (group of) objects. These constraints can express some spatial and temporal synchronizations 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 (using constraints solvers) compute one spatial and temporal position of media objects among the set of solutions. Techniques used in these two dimensions are different because the requirements they have to satisfy are not the same: expressive power, solution maintenance versus constraints resolution, time performances.

Constraint techniques can also be used when the authoring environment manages a graphical view of the constraints to help the author to understand the temporal organization of the document. Algorithms for solutions maintenance are required when such a view provides the author with a way to access the set of solutions by direct manipulation.

The aim of this paper is twofold: firstly, we would like to show the advantages for the author when constraints are used in an authoring environment for multimedia documents; secondly, we would like to present different technological problems emerging from the use of constraints in such context. This document is organized into four parts. The first one presents what could be an ideal constraint based specification language for multimedia documents (without any implementation issue). The second one focus on the benefits for authors when using this language. The following section presents the current state of Madeus an authoring environment based on a subset of the perfect language. Finally, the last section of this paper shows open technological problems that are still to be solved in order to provide the author with such a perfect language.

2 Specifying a multimedia document by using constraints

This section is devoted to the presentation of an ideal constraint based language which is the result of our experiments in authoring multimedia documents.

2.1 Specifying the temporal dimension

The principle is to associate basic objects of the document with a range of possible durations among which the author has the possibility to select one duration as a preferable one, and to set temporal constraints between objects.

Temporal constraints include the non−disjunctive Allen’s algebra [1], where the seven basic operators are(1):

- A *EQUALS* B : A and B start and finish at the same time
- A *STARTS* B : A and B start at the same time, and A finishes before B
- A *BEFORE* B : A is presented strictly before B

(1) the six inverse operators are written by using the 6 last operators with −1 as superscript

- A *MEETS* B : B is presented when A finishes
- A *FINISHES* B : A and B finish at the same time and A starts after B
- A *DURING* B : A is presented during B

However, some disjunctions of such operators are needed. For instance, when two objects start at the same without taking care of their ending. This is expressed by the disjunction: "A *STARTS* B or A $STARTS^{-1}$ B or A *EQUALS* B". Another representative example is given by a mutual exclusion between two objects: "A *BEFORE* B or A $BEFORE^{-1}$ B. Thus, the ideal set of temporal constraints must contain the set of Allen's Algebra with disjunction.

Figure 1 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". In addition, a "smiley" (Smiley) makes a brief apparition during the presentation of the text "and". Finally, 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" without considering their complete delivery. No preferable durations are specified.

Merry *BEFORE* And		Merry *STARTS* Pict1	
And *BEFORE* Happy		Pict1 *BEFORE* Pict2	
Smiley *DURING* And		Pict2 *BEFORE* Pict3	
Merry *EQUALS* Song1		Pict3 *BEFORE* Pict4	
Happy *EQUALS* Song2		Pict4 *FINISHES* Happy	
where:			
Merry: [25, 35]	And: [25, 35]	Happy: [25, 35]	Song1: [10, 60]
Song2: [10, 60]	$Pict_i$ [15, 120]	Smiley: [5, 15]	

Figure 1: Temporal specification of the Christmas example

Another kind of temporal operators is necessary to express interruption behavior. Suppose the author wants to provide the readers with a way to interrupt the Merry Christmas part (the text Merry with Song1) in order to see faster the remainder of the document. This kind of behavior can be expressed by the *PARMIN* operator where A *PARMIN* B means that objects A and B start together and the shortest terminates the other element. Moreover each object which is linked to either the end of A or the end of B by an equality constraint is also interrupted. For our example, a button object is added with the following behavior: its end occurs when the reader clicks on it. A new relation is then introduced: Button *PARMIN* Song1.

Our experiments show us that another kind of interruption operator is also very useful to describe multimedia scenario. We call it *PARMASTER* and its semantics is such that an identified operand (the master) interrupts the other object when it ends iff it has not yet ended.

Informal specification of multimedia documents is usually given by means of a hierarchical description: a document can be decomposed into scenes, sub−scenes, etc. It would be great for the author to specify documents in such a structural way which preserves the informal description. That means putting together a set of objects into one entity, namely a *composite object*, which can thereafter be used in the scenario in the same way as a basic object.

As an example of hierarchical decomposition, let’s consider the set of objects given in figure 1. They can be grouped into a composite object named "Christmas_Card" and a "News" document can be defined as:

News = Christmas_Card *BEFORE* Family_News

The semantics 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.

2.2 Specifying the spatial dimension

Constraints can be very interesting for the incremental spatial positioning of multimedia objects (considered as rectangular boxes) [23]. First, this way of specification is quite natural and close to similar temporal relations (A *BEFORE* B replace by A *LEFT_of* B, etc.); second, it can be more powerful than classical group/ungroup operators because it allows an object to be involved in more than one relation.

In addition to conventional style specification languages [25], objects can be set into relations with constraints on both vertical and horizontal dimensions. These constraints must be able to express that two objects are aligned or centered, have the same size, one is at the right (or left) side of the other one (without specifying a precise gap) or that they are disjoint. These constraints do not introduce a master/slave relation between objects (i.e. they are multi−ways constraints).

The hierarchical structure which appears in the temporal dimension is also useful in the spatial dimension. The box associated with a composite object is the smallest one that contains all its components. It should be possible to put spatial constraints between composite objects and also between an object and the composite objects which contains it, for instance to express that a title must be centered in a scene.

The temporal dimension of multimedia documents introduces the need of spatial constraints evolving with the time dimension, for example, when a title T must be successively centered with two objects A and B sequentially displayed.

3 Benefits and difficulties for the author

A well−known advantage of using constraints in authoring environments is that they can be easily used even by computer−illiterate people. It is obvious that in our context of multimedia authoring this is a great advantage: usually designers and artists are not skillful programmers. We focus in the next sections on other advantages.

3.1 A secure and incremental design of a multimedia document

Designing an interactive multimedia document is a cyclic "specify, test and modify" process: the desired document is rarely obtained from the first specification. An important point is the easiness by which the authoring environment can help the author in adjusting his document. The use of constraint paradigms is a real advantage thanks to the following characteristic they allow: each time the author adds or deletes a (spatial or temporal) constraint, the current set of solutions is checked and updated. For instance, if the author wants to insert in the Christmas example a fifth background object between the four previously mentioned ones, it is enough to modify the existing constraints in order to have:

Pict1 *BEFORE* New_Pict; New_Pict *BEFORE* Pict2

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

3.2 Reusing multimedia specifications

The reusability of parts of − or entire− existing documents can remarkably save the time while creating multimedia documents. With constraint based approaches, it is possible to reuse a composite object into another context. The important point is that a composite object is not a fixed object but an adjustable one, i.e. neither its duration nor the temporal position of the objects it contains are precisely defined. 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 one. 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 the first version of the document. Constraint technology saves the author the trouble of modifying the temporal organization to adjust such documents.

3.3 Modeling of adaptable multimedia documents

Multimedia documents aims to be presented on various presentation terminals with different resources. Spatial and temporal synchronizations must be preserved to guarantee the right delivery of the semantic content of the document. This is currently one of the main problems of multimedia providers. Using constraints could be a good help to answer this problem [24], since document specification does not define one presentation but a set of presentations which can be not equivalent if we take the required resources into account. Dynamic evaluation of the presentation to play would be a solution to handle in a better way the quality of the document presentation on various resources conditions.

For instance, if the system load of the presentation terminal is high, it will be great that the formatter dynamically computes a solution that does not overload the system (e.g. by reducing the size of pictures or videos, by avoiding to play two videos at the same time, etc.).

Distributed multimedia documents (on the Web for instance) will make this issue more and more important and network load will be an additional element to take into account [8].

3.4 Authoring difficulties

However, in approaches where the system automatically computes the solution (here both temporal and spatial placements), the author may be disoriented by the proposed solution. It would be useful to allow the author to tune the computed solution in order it better fits the wished result. Moreover, in an editing environment, the author must understand the current solution (more precisely, the direct and induced dependences between objects) before applying some modifications on the scenario. Finally, it has been stated that some helps for understanding incorrect specifications must be provided. All these problems are well−known when using constraints for graphics [15]. We will see in 4.1.3 that we try in Madeus to reduce them thanks to the use of a graphical view associated with temporal constraints.

4 The current state of the Madeus authoring environment

Madeus is a prototype of authoring and presentation environment [11] based on constraints techniques. It provides document authors with an efficient and flexible way to specify a multimedia document while retaining established declarative mark−up languages for temporal synchronization and spatial positioning.

4.1 Management of the temporal dimension

4.1.1 Temporal language

We restrict ourselves to the tractable Allen’s subalgebra that is equivalent to the time−point algebra [21], since (i) we can use polynomial time well−known algorithms to check consistency and (ii) numeric information like durations can be easily handled. Although this restriction reduces the expressive power of the language, it is enough to represent a large subset of interesting scenarios.

Only a simple form of hierarchical structure and interruption operators are supported in Madeus. We will see in section 5.1 some problems raised while introducing the whole semantics of such possibilities.

4.1.2 Implementation

The scenario is translated without any loss of information into a Simple Temporal Problem [6] (STP in the sequel) by translating both the Allen's relations and the duration constraints into a set of linear inequalities on time points Xi's which are the beginning and ending points of each object. Consistency and formatting phase are based upon a classical path−consistency algorithm [6]. This formalism has been extended to handle interruption operators [10].

4.1.3 The scenario view

In order to reduce authoring difficulties which arises from the use of constraints (see section 3.4), Madeus provides the author with a "scenario view". It can be seen as the projection of one document presentation into the time space, in which not only temporal information of objects (begin/end instants and duration) but also their temporal relationships are represented. Experiences with the visualization of constraints graphs have shown their limited benefits to authors [5]. More interesting is the visualization of the set of relations in conjunction with their effects i. e. the real temporal placement of the set of beginning and ending points. Selecting an object also shows the set of its possible positions in order to provide some way to anticipate further editing actions.

Figure 2 shows the view of our Christmas card example where springs represent flexibility introduced by constraints such as during and vertical lines represent simultaneous instants. The solution displayed here minimizes the total duration of the document. Working with such a scenario view is very helpful in the incremental design process: it allows to know which constraint must be deleted, where an object can be inserted, etc.

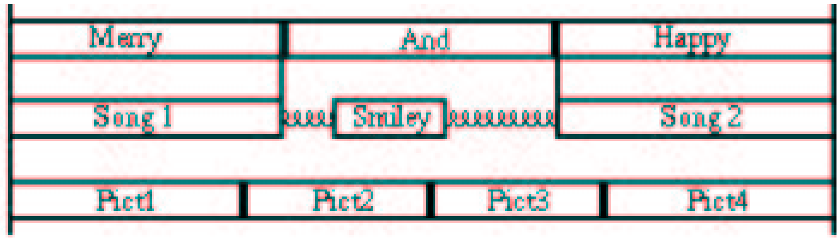

Figure 2: scenario view of the Christmas example

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 while adjusting other objects in real−time) in order to browse through the set of solutions. In addition, this view is a good support to help the author to choose one solution among the possible ones: the solution displayed in this view can be selected to be played by the presentation system.

The current implementation of this view is based on ad'hoc algorithms which operate on the temporal networks issued from both the consistency checking phase (minimal domains are of great importance to anticipate the consistent modifications allowed by the scenario) and the formatting phase (which gives the first displayed solution).

4.2 Management of the spatial dimension

In Madeus, we have experimented how spatial properties can take advantage of the use of constraints as stated in 2.2. We do not attempt here to present a complete language for the specification of style and layout of multimedia documents (standards still exist for that purpose).

4.2.1 Spatial constraint language

Madeus uses classical spatial constraints such as align, center and shift, on both the vertical and horizontal axes. An object can be involved in more than one spatial relation as long as it does not introduce inconsistencies. Figure 3 contains some spatial relations that can be set for the Christmas example.

Merry `Horiz−center` And

And `Horiz−center` Happy

Merry `Vert−center` Happy

And `Left−Align` Merry

`Smiley` `Right−Align` And

Figure 3: Spatial constraints of the Christmas example

Once a constraint is set, it is always hold during the life time of the document until the constraint is suppressed. Therefore, during edition or even presentation phases, the user can move any object involved in spatial relations, the constraints will be maintained.

The current implementation of Madeus allows to use a subset of the spatial relations (see [22] for a more complete list of possible spatial relations). Restrictions, such as constraints that introduce inequalities or hierarchical dependencies, are mainly due to limits of the algorithms used for constraint solution maintenance (see below): neither inequalities nor cyclic constraints are allowed.

4.2.2 Implementation

The Madeus spatial formatter is based on DeltaBlue [17]. This constraint solver uses a local propagation technique for providing efficient and incremental consistency checking. The result is very interesting for authors because they can adjust the position of objects in a very precise way: constraints are continuously maintained when objects are moved.

Its main limitations, well−known when using local propagation approaches, have been partially overcome in order to allow cycles due to redundant constraints.

5 Open technological problems

After presenting the ideal constraint based language for authoring multimedia documents and then the subset currently supported in Madeus, we now focus on different technological problems emerging from the use of constraints in such context.

5.1 Consistency of the scenario and formatting phase

A more detailed presentation of the three last points discussed below could be found in [7].

High performances of consistency checking and formatting phase

In section 3.1, we have stated that a benefit of the use of constraints is the ability to easily change the current specification of a document while being confident in its consistency. This incremental support is achievable if the algorithms used for consistency checking have high time performances. In addition, the author frequently wants to show the current state of the document by executing it. He cannot be satisfied with an environment in which switching between editing and presentation phases takes a long time. That means that algorithms used to compute the solution during the formatting phase must also have high time performances. These performance requirements can be met if we take the sublanguage defined in 4.1.1, but the difficulty is to preserve them while introducing disjunctive expressions.

Intelligent formatting phase

Authors of multimedia documents will be satisfied by constraint based approaches if and only if the automatically computed solutions are not too far from what they wish. The formatter can give priority to solutions that respect preferable duration of objects as most as possible, minimize the total duration of the document, ... Our current implementation which consists in computing the solution by several calls to a path−consistency algorithm allows the minimization of a global duration but not to take into account other criteria.

Controllable and uncontrollable durations

In classical CSPs, constraints and variables are supposed to be such that one can always assign a value from the interval domain when building a solution. Such an assumption is unrealistic in our application [13], since some durations are not under control of the application but are observed during the presentation phase. For instance, this is the case when buttons are pushed by the document reader: the corresponding duration cannot be statically decided by the formatter. Hence it is necessary to modify the consistency property: a presentation is consistent iff synchronizations are preserved *whatever the values* taken by uncontrollable durations.

Handling such uncontrollable variables in Simple Temporal Problem(s) has been studied in [20]. It has been shown that the classical consistency property must be redefined in terms of dynamic controllability. Checking this property is a highly combinatorial task in general. Alternatives to constraint based models rely on discrete−event based simulation tools but such models become generally rapidly huge, and are therefore not easy to handle, especially in an incremental context.

Hierarchical structure

As defined in section 2.1, the intuitive meaning associated with a composite object is: its starting (resp. ending) point is the minimum (resp. maximum) of the starting (resp. ending) points of the objects inside the composite. Such semantics implies to express disjunctions in the internal temporal representation. Notice that using general TCSP [6] instead of STP would not be enough: the constraints that have to be represented are not binary, but can involve more than two time points.

Interruption behavior

The semantic of `PARMIN` and `PARMASTER` relations can be defined by some equations on start and end instants but we need to distinguish between an expected end (the end computed by the formatter) and the effective one (the end caused by another object). For instance, the semantics of `PARMIN` is expressed by the following ternary constraints:

A_s=B_s, A_{ef}=B_{ef}, $A_{ef} = \min(A_{ex}, B_{ex})$, $A_{ex}-A_s \in$ [minA, maxA] and $B_{ex} - B_s \in$ [minB, maxB].

where A_s and B_s are start ponts of A and B, A_{ex} and B_{ex} their expected ends and A_{ef} and B_{ef} their effective ends.

A difficulty is to merge these equations with the equations deduced from the other constraints of the scenario: the previous algorithm which takes constraints one by one, must now be more global. Indeed, if A is the operand of a `PARMIN` relation, the end variable of A used in other equations must be either A_{ef} , otherwise the A_{ex} variable must be used.

The other difficulty is to extend consistency and controllability algorithms to take into account these new constraints.

5.2 Scenario view and spatial constraints

Problems arising when implementing an authoring environment which provides a scenario view allowing direct manipulations are the following: firstly, we have to dynamically maintain a set of solutions [4] with real−time performances: in such a visual environment response time must be about 2/10 seconds. Secondly, when adding/deleting a new constraint or object, the new displayed solution must be as "close" as possible to observe the Principle of Least Astonishment. This stability criterion is even more important if the scenario view is used by the author to tune the solution selected for the presentation. Finally, the author needs to have some information to anticipate further manipulations.

Since we have to handle cyclic constraints and inequalities, finding algorithms which meet these three requirements is a difficult task. Global approaches like [18] have not enough good time performances, local approaches like [17] cannot manage cycles and inequalities neither anticipation.

Ad'hoc algorithms currently used in Madeus are based on oversimplified hypotheses and could not be easily extended, in particular to meet the stability criterion.

Problems encountered with spatial constraints are similar to those identified in the scenario view. However, the need to spatially center objects requires to use other classes of constraints algorithms.

6 Conclusion

In this paper, we have presented some benefits that authors can obtain when using a constraint based authoring tool and we have illustrated them through the description of Madeus. As shown in the previous section, numerous technological problems are still open and cover both constraints resolution (consistency with uncontrollable variables) and solution maintenance (with anticipation needs).

Bibliography

[1] ALLEN J.F., Maintaining Knowledge about Temporal Intervals, *CACM*, Vol. 26(11), pp. 832−843, November 1983.

[2] ANDRE E. and RIST T., Coping with Temporal Constraint in Multimedia Presentation Planning, *12th Europ. Conf. on Artificial Intelligence*, edited by W.Wahlster, John Wiley & Sons Ltd, 1996.

[3] BEACKER R., ROSENTHAL A., FRIEDLANDER N., SMITH E. and COHEN a., A Multimedia System for Authoring Motion Pictures, *Proc. of ACM Multimedia 96 Conference*, pp. 31−42, Boston, November 1996.

[4] BELLICHA A. Maintenance of Solution in a Dynamic Constraint Satisfaction Problem, 8th Proc. of Applications of Artificial Intelligence in Engineering, pp. 261−274, Toulouse (FR), 1993.

[5] BUCHANAN M.C. and ZELLWEGER P.T., Automatic Temporal Layout Mechanisms, *Proc. of ACM Multimedia 97 Conference*, pp. 341−350, Anaheim, August 1993.

[6] DECHTER R., MEIRI I. and PEARL J., Temporal Constraint Networks, *Artificial Intelligence*, 49, pp. 61−95, 1991.

[7] FARGIER H., JOURDAN M., LAYAIDA N., VIDAL T., Using Temporal Constraints Networks to manage Temporal Scenario of Multimedia Documents, pp. submitted to ECAI−workshop on spatial and temporal reasonning, Brighton UK, August 1998.

[8] HAFID A., BOCHMAN G., DSSOULI R, Distributed multimedia application and quality of service: a review., *Electronic Journal on Netweorks and Distributed Processing*, 6(2), 1998.

[9] HARDMAN L., BULTERMAN D.C.A and Van ROSSUM G., The Amsterdam Hypermedia Model: Adding Time and Context to the Dexter Model., *CACM*, 37(2), pp. 50−62, February 1994.

[10] JOURDAN M., LAYAIDA N. and SABRY−ISMAIL L, Time Representation and Management in MADEUS: an authoring environment for multimedia documents, *Multimedia Computing and Networking 1997*, edited by M. Freeman, P. Jardetzki, H.M. Vin, SPIE 3020, San−Jose, February 1997.

[11] JOURDAN M., LAYAIDA N., SABRY−ISMAIL L, and ROISIN C., An Integrated Authoring and Presentation Environment for Interactive Multimedia Documents, *Proc. of the 4th Conference on Multimedia Modelling*, World Scientific Publishing, Singapore, 18−19 November 1997.

[12] KIM M.Y. and SONG J., Multimedia Documents with Elastic Time, *Proc. of the ACM 95 Multimedia Conference*, pp. 143−154, San Francisco, November 1995.

[13] LAYAÏDA N. and 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, USA, February 1996.

[14] MACROMEDIA, *Director 6*, on line:http://www.macromedia.com/, 1998.

[15] MARRIOTT K., Unsolved problems os constraint based Computer Graphics , *Proc. of the International Worshop on constraints for Graphics and visualization*, Cassis (FR), 1995.

[16] PRICEe R., MHEG: An Introduction to the Future International Standard for Hypermedia Object Interchange, *Proceedings of the First ACM Conference on Multimedia*, pp. 121−128, ACM Press, Anaheim, Californie, August 1993.

[17] SANNELLA M., MALONEY J., FREEMAN−BENSON B. and BORNING A., Multi−way versus one way constraints in user interfaces: Experiences with the Deltablue algorithm, *Software Practice and Experience*, 23(5), 1993.

[18] SCHIEX T., VERFAILLIE G., Two approaches to the solution maintenance problem in dynamic constraint satisfaction problems, *Proc. of the IJCAI−93 / SOGMAN Workshop on Knowledge−based Production Planning, Scheduling and Control*, Chambery (FR), 1993 .

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

[20] VIDAL T and FARGIER H., Contingent durations in temporal CSP : from consistency to controllabilities, *4th Int. Workshop on Temporal Representation and Reasoning (TIME97)*, Daytona Beach (Florida), USA, May 10 −11 , 1997.

[21] VILAIN M., KAUTZ H. A. and VAN BEEK P., Constraint Propagation Algorithms: a revised report, *Readings in Qualitative Reasoning about Physical Systems*, Morgan Kaufman, 1989.

[22] VAZIRGIANNIS M., THEODORIS Y. and SELLIS S., Spatio−Temporal Composition in Multimedia Applications, *Proc. of the IEEE ICSE'96 Conf. on Multimedia Software Development*, Berlin, Germany, 1996.

[23] WEITZMAN L., WITTEMBURG K., Automatic presentation of multimedia documents using relational grammars, *Proc. of the second International Conference on Multimedia*, ACM, San Francisco (USA), 1994.

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

[25] W3C, *W3C Recommandation of Cascading Style Sheet Language (CSS) 1.0*, http://www.w3.org/TR/1998/REC, 1997.

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

Constraints Techniques for Authoring Multimedia Documents