http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 5 (5): 823-832, 2006
ISSN 1812-5638

A New Approach for Synchronization, Analysis and Management of Multimedia Scenarios

[1]Abdelghani Ghomari, [1]M.K. Rahmouni and [2]Chabane Djeraba
[1]Department of Computer Science, University of Es-Senia Oran,
BP 1524, El M'naouer, 31000, Oran, Algeria
[2]University of Sciences and Technologies of Lille,
Scientific City, Building M3, 59655 Villeneuve, France

Abstract: The present study propose an approach to multimedia scenarios management in an object-relational database system that considers: an object-oriented method for multimedia scenarios modeling; time-interval and causal relations models for the specification of multimedia objects with fixed or unknown playout durations; interactive relations and dependency temporal relations. Based on the user's temporal specification, a time Petri net is automatically generated and translated to the formalism of the verification tool Tina, which allows to analyse multimedia scenarios specification. Finally, the present study evaluate the approach with an experimental system prototype.

Key words: Multimedia scenarios, temporal and causal relations, time Petri net, synchronization, analysis, database

INTRODUCTION

Multimedia refers to the presentation of collections of both static and dynamic data (i.e., data with natural time dependencies e.g., audio or video) in a specified order and time. Therefore, their mutual synchronization must assure a proper temporal order of presentation events. Multimedia synchronization can be defined as a mutual assignment of data items and time instants. These time instants may be known in advance (e.g., standard consumer data players) or they can be also results of some unknown function of time (event driven synchronization) or known with some limited accuracy (e.g., random network delays).

The modelling and the presentation of multimedia scenarios are challenges of multimedia applications. Multimedia scenarios are results of temporal composition and user interactions of multimedia objects in an application domain and lot of works discussed this notion (Bertino and Ferrari, 1998; Huang and Wang, 1998).

Existing temporal models for multimedia can be divided into two classes: instant-based and interval-based (Wahl and Rothermel, 1994):

In instant-based models, the elementary units are instants in a time space. Each event in the model has its associated time instant. The time instants arranged according to some relations such as precede, simultaneous or after form complex multimedia presentations. An example of the instant-based approach is timeline, in which media objects are placed on several time axes called tracks, one per each media type. All events such as the start or the end of a segment are totally ordered on the timeline. Several approaches support instant-based models such as HyTime (ISO, 1992) or (Gibbs *et al*., 1993; Drapeau, 1993). The model is well suited for temporal composition of media segments of known durations; however it falls short for unknown durations. Some authors have proposed to use relations between interval end points for temporal composition of multimedia (temporal point nets (Buchanan and Zellweger, 1993), MME (Dingeldein, 1994)). However, their use is difficult and results in complicated, unstructured graphs. In addition to that, their use may led to an inconsistent specification in which contradictory conditions are specified for intervals. In this case, a verification algorithm (called sometimes a temporal formatter) must check for inconsistency.

Interval-based models consider elementary media entities as time intervals ordered according to some relations. Existing models are mainly based on the relations defined by Allen (1983) for expressing the knowledge about time. Giving any two time intervals, they can be arranged according to seven relations: before, meets, overlaps, finishes, during, starts, equals. However, using Allen's relations for multimedia composition faces several problems. First, the relations were designed to

---

Corresponding Author: Abdelghani Ghomari, Department of Computer Science, University of Es-Senia Oran, BP 1524, El M'naouer, 31000, Oran, Algeria

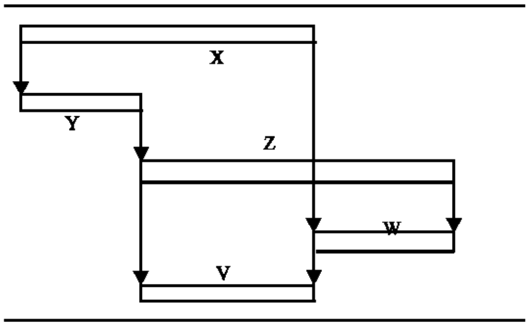

Fig. 1: Example of difficult scenario

express existing relationships between intervals of fixed duration and not for specifying relationships that must be always satisfied even when interval durations are changed. Second, the relations allow expression of an existing, a posteriori arrangement of intervals, but they do not express any causal or functional relation between intervals. The temporal relations describe existing arrangement of multimedia objects, but do not describe dependency relations between multimedia objects. For example, x meets y means that the end of multimedia object x coincides with the end of multimedia object y, but it does not describe whether multimedia object x starts multimedia object y, or whether multimedia object y stops multimedia object x.

To resolve theses disadvantages, a recent approach (Weiss *et al.*, 1994), considered in some systems such as STORM (Adiba, 1995), is proposed to allow temporal specification of dependency relations between multimedia objects of unknown duration. It defines a set of operators expressing causal relations between multimedia objects. One disadvantage of this approach is that not all scenarios can be expressed by means of those operators. For example, the scenario presented in Fig. 1 cannot be described, because of interleaved start and stop actions on parallel branches.

Another disadvantage of this approach is its dependency aspects. It allows the expression of causal or dependent relation between multimedia objects. So, if a multimedia object fails, all the multimedia objects that depend on the failed multimedia object fail too. If x fails, the multimedia objects y, z, w, v that depend on the failed multimedia object x, fail too (Fig. 1). In the interval-based model, in which the temporal relations do not describe dependency relations between multimedia objects but describe existing arrangements of multimedia objects, the consequences of the failed multimedia object are limited to this multimedia object, during the duration associated to it.

The third problem with the Allen's relations is related to inconsistent specifications that can be introduced to a multimedia presentation. Detecting inconsistent specification requires algorithms of complexity [$O(N^2)$], where N is the number of intervals (Allen, 1983).

BACKGROUND: MULTIMEDIA SYNCHRONIZATION APPROACHES

Many approaches are based on time interval. For example:

King (1994) proposes a different formalism based on a temporal logic. He shows how the Allen's relations can be expressed using temporal logic formula. Although his formalism has solid mathematical bases, composition of multimedia presentations using declarative formula is awkward. Logic formulas are difficult to use by any author unaware of formal methods. Moreover, to be useful, the formalism must be supported by a consistency checker and an interpreter to execute a given temporal specification.

Courtiat and De Oliveira (1996a) presented a synchronization model for the formal description of multimedia documents. This model automatically translates the user formalization into a real-time LOTOS formal specification and verifies a multimedia document aiming to identify potential temporal inconsistencies.

Blakowski and Steinmetz (1996) recognized an event-based representation of a multimedia scenario as one of the four categories for modelling a multimedia presentation. Events are represented in the Hypermedia/Time-Based Structuring Language (HyTime) and Hypermedia Office Document Architecture (HyperODA). Events are defined in HyTime as presentations of media objects along with the playout specifications and finite coordinate system (FCS) coordinates. HyperODA events happen instantaneously and mainly correspond to start and end of media objects or timers.

All these approaches suffer from poor semantics conveyed by the events and moreover they do not provide any scheme for composition and consumption architectures.

Another group of approaches consider the Petri net tool for multimedia synchronization. For example:

Object Composition Petri Net Model (OCPN) (Little and Ghafoor, 1990). However, the OCPN approach do not inherently support modeling of interaction.

Hierarchical Time Stream Petri Net Model HTSPN (Senac *et al.*, 1995). However, in the HTSPN approach, it is unspecified what would happen to tokens representing streams left behind the synchronization. If those "dead tokens" were remained in the state, a semantic inconsistency between the model and the real system would be produced.

Multimedia Organization Employing a Network Approach (MORENA) (Botafogo and Moss, 1995). However, since MORENA does not fully follow Petri net theory, it may not be easy to analyze the output model using the existing techniques developed in the Petri net field.

High-Level Petri Net-Based Hypermedia System (Na Cheo and Furuta, 2001). However, the authoring environment used to specify multimedia scenarios with colored timed Petri nets is not user-friendly, like in the user community.

Dynamic Fuzzy Multimedia Petri Net DFMN (Chen and Huang, 2005). However, in the approach of Shterev (2005) using the DFMN model for modeling of interaction on multimedia streams and objects, none analysis or simulation tools of multimedia presentation are provided.

Special attention should be payed to the upcoming standard SMIL (Synchronized Multimedia Integration Language) (W3C, 2001). SMIL is a meta-language for authoring and presentation of multimedia documents. However, SMIL is a script language and does not support analysis to the same degree that Petri-net-based systems do.

In conclusion, you will find an interesting survey on authoring models and approaches elsewhere (Jourdan *et al.*, 1998; Roisin and Sèdes, 2004).

MATERIALS AND METHODS

The proposed approach consider a temporal specification that is based on both time interval of Allen (1983) and relations of Weiss *et al.* (1994). A temporal specification defines temporal relationships between temporal objects. The model for temporal composition of multimedia objects uses the causality features of the temporal relationships (i.e. the end of video A causes the start of video B). Causality enables the triggering of actions based upon the occurrence of events. Actions can take a number of forms, such as the starting, stopping, fast-forwarding, or rewinding of a media object. An event is raised by the occurrence of an action and has attached a temporal instance. Events can be basic, such as the begin event of a media object, or user-defined, such as defining a media-based event associated with the appearance of a red ball within a scene in a film. They can also be application-defined, such as a menu selection made by a user, or time events that are triggered when a timepoint associated with a specific clock occurs.

Temporal relations: Several relationships have been defined on time intervals: before, meet, equal, overlap, during, start, finish (Allen, 1983). Usually, they are binary relationships but can be easily extended to n-ary ones (Little and Ghafoor, 1993). Sequential relationships combine intervals, which share the same timeline (mutual exclusion), occurring one after the other with (before) or without delay (meet) between them. Parallel relationships relate intervals, which have their own timeline. In present model these relations are used for composing and synchronizing multimedia objects in presentations.

Interactive relations: The present study synchronizes the scenario with the user (i.e., an expert of the application domain). The interaction takes the form of temporal interaction (start, stop, pause, reverse and forward) and browsing interactions. Temporal interactions concern user elementary operations such as pause/resume, reverse and forward and in browsing interactions, the user branches out of the current presentations, so he/she effectively modifies the current presentation.

The study considers a program of temporal specification language that is divided into four parts: declaration, assignation, temporal and interactive relations (Fig. 2).

Example: A heart medical scenario specification: Let us consider an application of scenario temporal specification: in the framework of a service provided by a hospital, a doctor has the possibility to choose a multimedia service among several. For example, when consulting a heart medical file of a patient, the doctor is able, to visualize the multimedia file of the patient, thanks to temporal synchronization, while focusing himself on the affected bodies. The heart medical file is composed of a textual description of the heart and its behaviour, audio and video sequences describing the beats of the heart (e.g. phonocardiogram). The audio and video sequence is synchronized with two images, describing a longitudinal and transverse crosscut of the heart, respectively. These two crosscuts evolve with the audio and video sequence of the heart. At any moment, the user can stop, represent and return to the initial menu to choose medical files of other patients or leave the scenario (Fig. 3).

Multimedia P-Time Petri net (MP-RdPT) model: Petri Nets (PN) are designed to model systems with interacting concurrent components. The basic PN structure is composed of four parts: a set of paces P, a set of transitions T, an input or (backward) function B and an output or (forward) function F. The input and output

```
<Language>::= scenario<Scenario Name>declarations {<Declaration>}*assigns{<Assign>}* interactions
{<Interaction>}* relations {<Temporal Relation>}* end.
<Scenario Name>::=<Identifier>
<Declaration>::=<Multimedia Object>[(Time Interval>)]:<Type>;
        |<Scenario Object>: SCENARIO;
        |<Button Object>: BUTTON;
<Multimedia Object>::=<Object Identifier>/* OID */
<Time Interval>::= [integer, integer] /* Time */
<Type>:: =audio | video | image | text | animation | button
<Assign>:: = assign (<Multimedia Object>,<Physical Support>)
<Interaction>::= interaction (Button Object,<Scenario Object>)
<Temporal Relation>::= [<Multimedia Variable>:=] {<Equal>|<Meet>|<Before>|<Start>|<During>
|<Finish>|<Overlap>|<par_min>|<par_max>|<par_master>}
<Equal>:: = equals (<Multimedia Object Parameter>);
<Meet>::= meets (<Multimedia Object Parameter>);
<Before>::= before (<Multimedia Object Parameter>,<Delay>);
<Starts>::= starts (<Multimedia Object Parameter>,<Delay>);
<During>::= during (<Multimedia Object Parameter>,<Delay>);
<Finish>::= finishes (<Multimedia Object Parameter>,<Delay>);
<Overlap>::= overlaps (<Multimedia Objec tParameter>,<Delay>);
<par_min>::= par_min (<Multimedia Object Parameter>);
<par_max>::= par_max (<Multimedia Object Parameter>);
<par_master>::= par_master (<Multimedia Object Parameter>);
<Multimedia Object Parameter>::=<Interval>,<Interval>
<Interval>::=<Multimedia Object>|<Senario Object>|<Button Object>|<MultimediaVariable>
```

Fig. 2: The Backus-Naur form (BNF) of the temporal specification language

Scenario Heart
Declarations
HeartText (10,+∞): TEXT;
Transverse Image$_1$ (20,20),...,Transverse Image$_n$(20,20): IMAGE;
Longitudinal Image$_1$ (20,20),..., Longitudinal Image$_n$(20,20): IMAGE;
Heart Audio (10,1050): AUDIO;
Heart Video (10,1050): VIDEO;
Inter Heart (10,+∞): HYPERMEDIA; ……
Assignations
Assign (Heart Video, heart.mpg);
Assign (Heart Audio, heart.so);
Assign (Transverse Image, transheart.gif); ….
Temporal relations
Heart Image:= meets(equal (Transverse Heart Image1, Longitudinal Heart Image1),...
Meets (equal (Transverse Heart Imagen-1, Longitudinal Imagen-1),
Heart Scenario:= equal (Heart Audio Visual, Heart Text);....
Scenario Level1:= equal (during (Introduction, Inter Consultation,0), during (Introduction, Inter Buckle Introduction, 0)); ….
Interactive relations
Interaction (Inter Heart, Scenario Level3);
Interaction (Inter Buckle Heart, Scenario Level3);….
End.

Fig. 3: An example of the heart scenario edition

functions relate transitions and places (Peterson, 1981). A basic PN graph is graphically represented as a bipartite directed graph, in which the circular nodes are called places and the bar nodes are called transitions. A dot in a place represents a token and a place containing one or more tokens is said to be marked.

Allen's relations have the problem that multimedia objects expressions are dependent of multimedia objects duration. The temporal relations are designed to specify relations between multimedia objects of determined duration. Therefore, they are not appropriate for undetermined duration. If creators produce a video digest from several multimedia objects, they need to modify the temporal relations between multimedia objects after their duration changes.

In order to solve this problem, the system should represent relations between multimedia objects with unknown duration. We must study the model of temporal relations independent of duration changes. PN is one of graph representations and has some characteristics: PN considers multimedia objects with known or unknown duration and the simulation of the scenario may detect errors, such as: specification errors, graph design errors, graph configuration errors, or allocation resources errors.

The PN tool have been chosen as tool of synchronization and analysis because PN allow modelling the dynamic behaviour of multimedia scenarios that can be characterized by the qualitative properties of PN corresponding. These properties are liveness, boundedness, reversibility and consistency.

In the context of a temporal synchronization modelling, a class of enhanced PN model has been developed which assign a firing delay to each place and a type of synchronization to a transition (Ghomari and Djeraba, 2003a). This model is called MP-RdPT and it is defined as follows:

Formal definition of MP-RdPT: A MP-RdPT is a tuple $(P, T, B, F, M_0, IS, SYN, MP, R)$, where:

(P, T, B, F, M_0) defines a PN, where P is a non empty finite set of places, T is a non empty finite set of transitions, with $P \cap T = \varnothing$, $B: P \times T \to N$ is the backward function, similarly, $F: P \times T \to N$ is the forward function, $M_0: P \to N$ is the initial marking. As usual, we denote by ${}^\bullet t \overset{def}{=} \{p \in P \mid B(p, t) \geq 1\}$ the set of ingoing places and $t^\bullet =_{def} \{p \in P \mid F(p, t) \geq 1\}$ the set of outgoing places of a transition t. Similarly, ${}^\bullet p \overset{def}{=} \{t \in T \mid F(p, t) \geq 1\}$ and $p^\bullet =_{def} \{t \in T \mid B(p, t) \geq 1\}$ are the sets of ingoing transitions and outgoing transitions of a place p.

The set of markings a MP-RdPT can reach from its initial marking M_0 will be denoted as $S(M_0)$.

$\forall p \in P, \forall M \in S(M_0), M(p) \leq 1$ (a MP-RdPT is safe),

IS: is the static interval function, $IS : P \to (Q^+ \cup 0) \cup (Q^+ \cup \infty)$,

The IS function associates with each ingoing place a static validity time interval, where (a, b), associated with a place, represents respectively the earliest and the latest firing times. The firing time of a place is a timing interval during which the newly created tokens are valid to fire transition.

SYN is the synchronization function that defines the firing rule associated to a transition:

$SYN: T \to Rules$, with $Rules =_{def} \{strong\_or, weak\_and, master\}$, the set of synchronization rules. This synchronization semantics defines synchronization instants from a place statically or dynamically chosen.

MP is the function which indicates the master place of each transition from which the rule of transition requires a master, defined by: $MP : T_{master} =_{def} \{t \mid SYN(t) = master\} \to {}^\bullet t$,

- The strong_or synchronization rule is driven by the earliest media. If either one of the two multimedia objects finishes, the other one has to stop and $[Min(a_i), Min(b_i)]$ is the sensibilisation interval.
- The weak-and synchronization rule is driven by the latest media. All the multimedia objects are presented completely and $[Max(a_i), Max(b_i)]$ is the sensibilisation interval.
- The master synchronization rule is driven by the master media. If two multimedia objects are presented simultaneously, when the higher priority media finishes, the other has to stop. The multimedia presentation continues after that and $[a_m, b_m]$ is the sensibilisation interval, with p_m which indicates the master place. We define a_m, b_m by: let $MP(t) = p_m$ and $IS(p_m) = [a_m, b_m]$.

-R: $P \to \{r_1, r_2, \ldots r_n\}$, a mapping from the set of places to a set of resources (e.g., audio/video card, processor, virtual memory and others operating system resources).

MP-RdPT generation: Hamblin (1972) presents a logic of intervals which is very useful in the development of a synchronisation scheme. Given any two intervals, there are thirteen distinct ways which they can be related. These relations indicate how the two intervals relate in time; whether they overlap, abut, precede, etc. Using the representation of Allen (1983) extended to that of

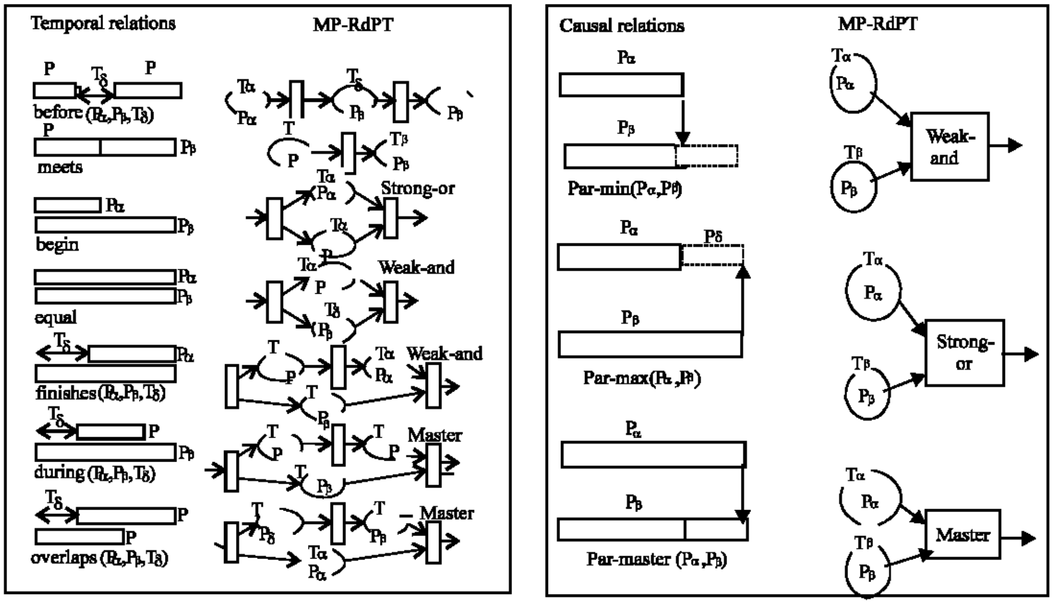

Fig. 4: MP-RdPT associated with temporal and causal relations

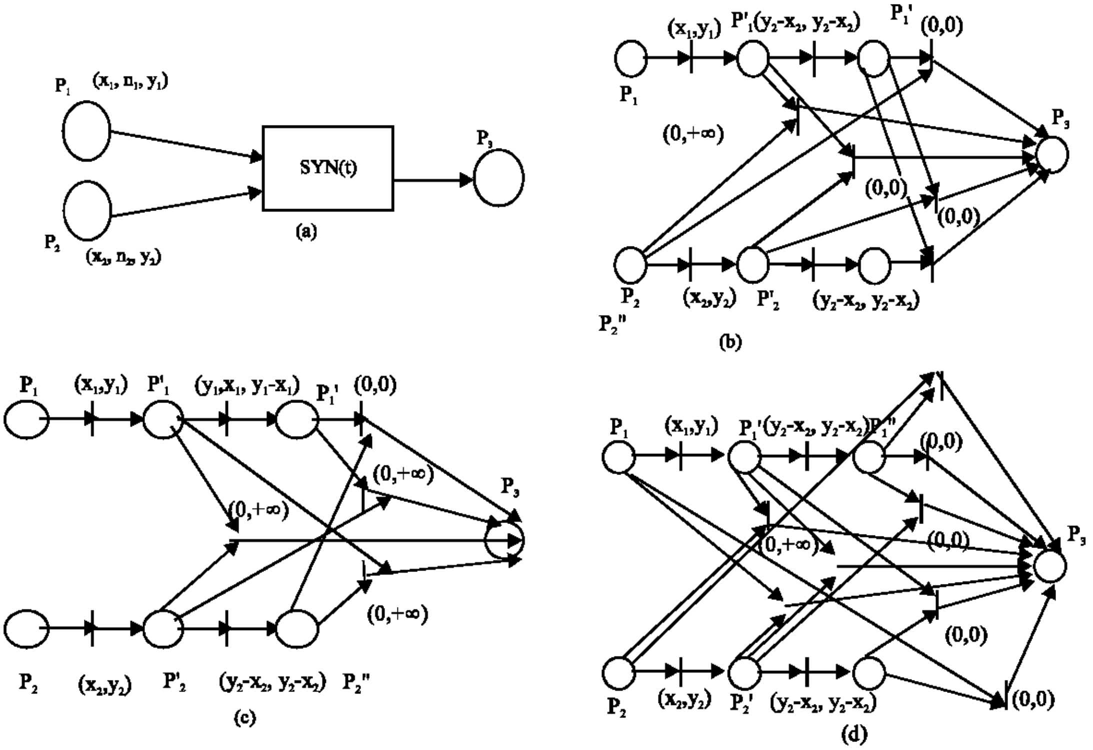

Fig. 5: Translation of an inter-media synchronization schema of a MP-RdPT net (a) in the form of a T- time Petri net (b, c, d) according to inter-media synchronization (b) transition of the type «master », (c) transition of the type « weak_and», (d) transition of the type « strong_or ».

Weiss *et al.* (1994), these relations are indicated graphically by a timeline representation (Fig. 4). Figure 4 shows intervals as processes P_α, P_β, P_δ with time intervals T_α, T_β, T_δ, respectively. One axis indicates the time dimension and the other space or resource utilization. Little and Ghafoor (1990) define an atomic process to be one which is cannot be decomposed into subprocesses, as in the case of the presentation of a single frame of a motion picture. They presents a theorem relating temporal intervals to OCPN: Given any two atomic processes specified by temporal intervals, there exists an OCPN representation for their relationship in time.

In (Courtiat *et al.*, 1996), it has been proved that the causal operators: par-min, par-max and par-master correspond to the types of synchronization: WaitEarliest (strong-or), WaitLatest (weak-and) and WaitMatser (master), respectively. So, according to this correspondence, theorem of Little and Ghafoor (1990) is generalised for relating temporal and causal relations to their associated MP-RdPT. This mapping is helpful for automatic creation of MP-RdPT (Ghomari and Djeraba, 2003a, b).

T-time Petri net definition: A T-time Petri net (Merlin and Farber, 1976) is a tuple (P, T, B, F, M_{20}, IS), where: (P, T, B, F, M_0) defines a PN and IS: $T \rightarrow Q^+ x (Q^+ \cup \{\propto\})$ is the static time interval function. The application IS associate with each transition t of the net an interval with rational bounds $IS(t) = [min, max]$ with $0 \leq min \leq max$ and max can be $\propto$.

Rules of translations: Senac (1996) presents a theorem relating an OCPN to an HTSPN model. He also proved several rules of translating an HTSPN model to a T-time Petri net. According to this theorem, we consider three rules of translation of our MP-RdPT model to an equivalent T-time Petri net (Fig. 5). In conclusion, we can use the verification tool Tina (Berthomieu *et al.*, 2003) for analysis of multimedia scenarios.

Analysis of multimedia scenarios using the tool Tina: Tina (Time Petri Net Analyser) (Berthomieu *et al.*, 2003) is a software environment to edit and analyse PN and T-time Petri net. In addition to the usual editing and analysis facilities of such environments (computation of marking reachability sets, coverability trees, semi-flows), Tina offers various abstract state spaces constructions that preserve specific classes of properties of the concrete state spaces of the nets. Classes of properties may be general properties (reachability properties, deadlock freeness, liveness).

After generating the T-time Petri net, the author investigates the scenario specification before it is delivered to the reader by using the analysis tool Tina. Currently, the following characteristics can be verified by the analysis tool: terminate state existence (e.g., if a state m exists in which non transitions are enabled), safeness (e.g., if every place has only one token), liveness (e.g., if blocking will never occur), reversibility (e.g., if the PN come back to its initial state whatever state it reaches), consistency (is a necessary condition for the reversibility that is a difficult property to establish. There are cases where problems are: cycles and deadlocks).

Object-oriented modeling: Our multimedia framework looks to the framework proposed in (Gibbs *et al.*, 1993). It is composed of abstract classes serving to specify interfaces and suggested procedures for using the classes. The abstract classes are specialized for different multimedia platforms. So, applications using the abstract classes may adapt to variations in platform functionality. The classes of our framework belong to two distinct groups: media classes and scenario classes: (1) media classes correspond to audio, video, image, text and other media types, their basic properties and operations and (2) scenario classes model temporal composition of media objects. In this paper, we will focus on scenario classes. Gibbs *et al.* (1993) proposes at least three distinct groups: media, transform and format classes and the main difference with their framework is the scenario classes.

Scenarios are divided into types corresponding to application domains. Each type is represented by a class. These are called scenario classes and form a class hierarchy diagram such as in the UML formalism (Booch *et al.*, 1998), (Fig. 6). Nodes depict classes and edges depict super-class/subclass relationships. Instances of scenario classes are called scenario objects. A scenario class models scenario object properties and operations. The properties of the scenario object consist of internal representation of the MP-RdPT (incidence-matrix, resource vectors, time interval vectors, token vectors, etc.), descriptors and an attribute that point to the root of temporal composition hierarchy. The methods of the scenario class are divided into two categories: generation, deletion and simulation, interpretation. The method generate allows applications to generate scenarios objects using temporal specifications and the method deletion remove scenarios objects. The method simulate graphically the scenario using the associated MP-RdPT and the method interpret plays the scenario.

Architecture of the system prototype: The system prototype is implemented on Compaq Intel Pentium 4 platform using the programming language C++ and Oracle DBMS and run in a windows environment. According to

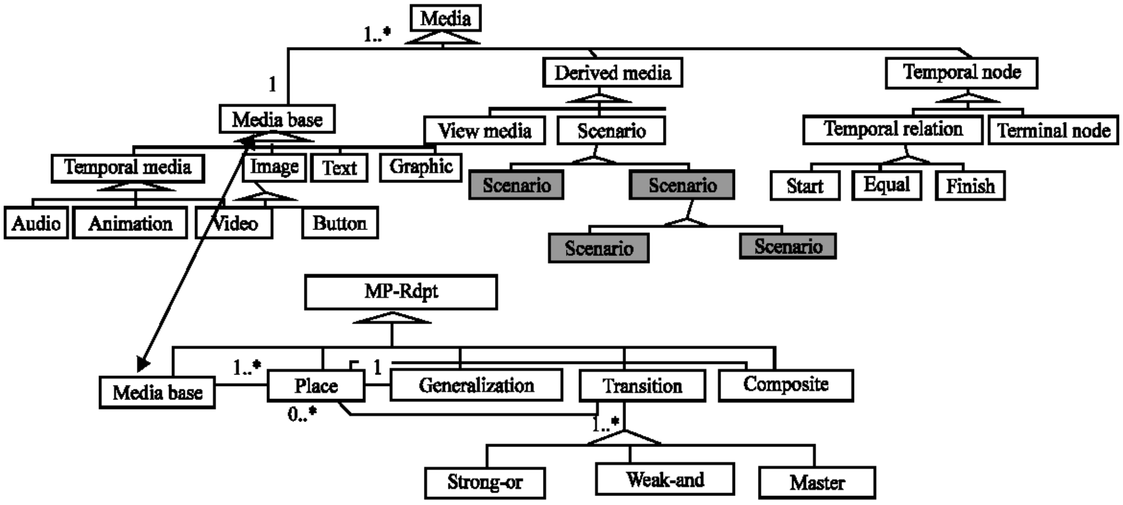

Fig. 6: A class diagram with UML formalism

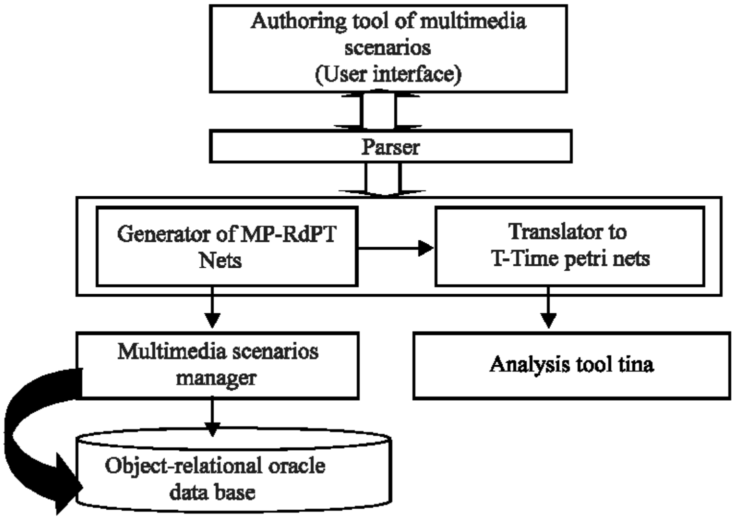

Fig. 7: System prototype architecture

the architecture of the system prototype (Fig. 7) the following components can be described: (1) Authoring tool, which use an editing language of temporal and causal relations, (2) Parser, which reads and analyses the specification file that contains the temporal specification language, (3) Generator, which generate the MP-RdPT model, (4) Translator, which translate the MP-RdPT model to T-time Petri nets, (5) Analysis Tool Tina, which analyses and verifies properties of T-time Petri nets. Manager, which manages the multimedia scenarios, such as: the generation, the simulation and the interpretation of Mp-RdPT, Object-relational Oracle Database, which stores media objects as Large Binary Objects (BLOBs) in mpeg, gif or jpeg files.

CONCLUSIONS

The present study presented a formal approach for specifying and analyzing multimedia presentation. The approach combines time-interval and causal relations models so that it supports the specification for multimedia objects with fixed or unknown durations, interactive relations and dependency temporal relations. The formalism used to model the specification is called MP-RdPT. Based on the user's temporal specification, a MP-RdPT is automatically generated according to a set of generation rules. The MP-RdPT model then can be translated to the T-Time Petri net and a set of properties then could verified against the translated model by using a third party tool called Tina. Tina allows the author to investigate the document specification before it is delivered to the reader. Finally, multimedia scenarios are stored and managed in an object-relational DBMS.

In the future, we will apply XML (Extensible Markup Language) (XML, 2001) as a format to capture and characterize multimedia information. One important property of XML for representing multimedia data is its ability to define tag as required. Moreover, XML context rules are highly customizable through definition in the DTD (Document Type Definition). Therefore, system can manage and store an enormous of structured data efficiently, not spoiling syntactic and semantic aspects.

REFERENCES

Adiba, M., 1995. STORM: Structural and Temporal Object-Oriented Multimedia Database System. In IWMDMS, Blue Mountain Lake, Newark.

Allen, J.F., 1983. Maintaining knowledge about temporal intervals. Commun., ACM, 26: 832-843.

Berthomieu, V. *et al*., 2003. The tool Tina-Construction of abstract state spaces for PN and Time Petri Nets. Intl. J. Prod. Res., 42. URL: http://www.laas.fr/Tina

Bertino E. and E. Ferrari, 1998. Temporal synchronization models for multimedia data. IEEE Trans. Knowl. Data Eng., 10: 612-630.

Blakowski, G. and R. Steinmetz, 1996. A media synchronization survey: Refrence model, specification and case studies. IEEE J. Selected Areas Commun., 14: 5-35.

Booch, G. *et al*., 1998. The Unified Modelling Language User Guide. Addison-Wesley.

Botafogo, R. and D. Moss, 1995. The MORENA model for hypermedia authoring. In Proceedings of the International Conference on Multimedia Computing and Systems, Los Alamitos, Ca., USA. IEEE Comp. Soc. Press, pp: 42-49.

Buchanan, M.C. and P.T. Zellweger, 1993. Automatic temporal layout mechanisms. In Proceeding 1st ACM International Conference on Multimedia, Anaheim, CA, pp: 341-350.

Chen, J.N. and Y.M. Huang, 2005. Using dynamic fuzzy petri nets for navigation learning. Exploring Innovation in Education and Research. Tainan,

Courtiat, J.P. and R.C. De Oliveira, 1996a. Proving temporal consistency in a new multimedia synchronization Model. In Proceeding of ACM Multimedia Conference ACM Press, New York, pp: 141-152.

Courtiat, J.P. *et al*., 1996b. A general purpose multimedia synchronization mechanism based on causal relations. IEEE J. Selected Areas Commun., 14: 185-195.

Dingeldein, D., 1994. Modeling multimedia-objects with MME. In Proceeding Eurographics Workshop on Object-Oriented Graphics, Sintra, Portugal.

Drapeau, G.D., 1993. Synchronization in the MAEstro multimedia-authoring environment. In Proceeding 1st ACM International Conference on Multimedia, Anaheim, CA, pp: 331-339.

Ghomari, A. and C. Djeraba, 2003a. Towards a timed-petri net based approach for multimedia scenario synchronization. In Proceedings of the 5th International Conference on Enterprise Information Systems ICEIS (1), Angers, France, pp: 267-272.

Ghomari, A. and C. Djeraba, 2003b. Synchronization of multimedia scenarios-A case study: Interactive news. In Proceedings of the 3rd International Workshop on Multimedia Data and Document Engineering (MDDE) in Conjunction with 29th International Conference on Very Large DataBases (VLDB), Berlin, Germany.

Gibbs, S. *et al*., 1993. Audio/Video databases: An object-oriented approach. In Proceeding 9th IEEE Int. Data Engineering Conference, pp: 381-390.

Hamblin, C.L., 1972. Instants and intervals. In Proceeding of the 1st International Conference Society For the Study Time, Springer-Verlag, pp: 324-331.

Huang, C.M. and C. Wang, 1998. Synchronizion for interactive multimedia presentation. J. Multimedia, Vol: 5.

ISO, 1992. Information Technology Hypermedia/Time-based Structuring Language (HyTime). ISO International Standard, (ISO/IEC IS 10744).

Jourdan, M. *et al*., 1998. A Survey on Authoring Techniques for Temporal Scenarios of Multimedia Documents. Handbook of Internet and Multimedia Systems and Applications, Part 1: Tools and Standars, CRC Press, London.

King, P.R., 1994. Towards a temporal logic based formalism for expressing temporal constraints in multimedia documents. Technical Report 942, LRI, Université de Paris-Sud, Orsay, France.

Little, T.D.C. and A. Ghafoor, 1990. Synchronization and storage models for multimedia objects. IEEE J. Selected Areas Commun., 8: 413-427.

Little, T.D.C. and A. Ghafoor, 1993. Interval-based conceptual models for time-dependent multimedia data. IEEE Trans. Knowl. Data Eng., Vol. 5.

Merlin, P.M. and D.J. Farber, 1976. Recoverability of communication protocols: Implications of a Rheoretical study. IEEE Trans. Commun., 24: 1036-1043.

Na Cheo, J.C. and R. Furuta, 2001. Dynamic documents: Authoring, browsing and analysis using a high-level petri net-based hypermedia system. In Proceeding of the ACM Symposium on Document Eng., Atlanta, CA, pp: 38-47.

Peterson, J.L, 1981. Petri Net Theory and the Modeling of Systems, Prentice-Hall.

Roisin, C. and F. Sèdes, 2004. Time and Documents. Numeric Document, 8: 23-39 (in French).

Senac, P. *et al.*, 1995. Hierarchical Time Stream Petri Net: A Model for Hypermedia Systems. In Application and Theory of Petri Nets. Michel Dia (Eds.). Lecture Notes in Computer Science.

Senac, P., 1996. Contribution to multimedia and hypermedia systems modeling. Ph.D Thesis, Toulouse, France.

Shterev, J., 2005. Modeling of interaction on multimedia streams and objects by application of multimedia streams and objects by application of petri nets. In Proceeding of the International Conference on Computer Systems and Technologies. Technical University, Varna, Bulgaria.

Wahl, T. and K. Rothermel, 1994. Representing time in multimedia systems. In Proceeding IEEE International Conference on Multimedia Computing and Systems, Boston, MA, pp: 538-543.

Weiss, R. *et al.*, 1994. Content-based access to algebraic video. In IEEE Multimedia, Boston, MA.

W3C, 2001.Working draft specification of SMIL. URL: http://www.w3.org/TR/SMIL 2.0/

XML, 2001. Extensible Markup Language. Available at URL: http://www.w3.org/XML/