SYNCHRONIZATION MECHANISM IN MULTIMEDIA DOCUMENTS

R. N. Jugele[1] and Dr. V. N. Chavan[2]

[1]Department of Computer Science, Science College, Congress Nagar, Nagpur. Maharashtra. rn_jugele@yahoo.com
[2]Head, Department of Computer Science, S. K. Porwal College, Kamptee, Dist : Nagpur. Maharashtra

Abstract

Synchronization is essential for delivering multimedia data correctly, the user interface means application should decide how to handle synchronization if resources are not available or sufficient for maintaining proper relationship between the media elements. Live synchronization exactly reproduces the temporal relationships as they exist during the capturing process and synthetic synchronization is artificially specified the temporal relations. In the specification phase temporal relationship between the media objects are defined and in presentation phase run-time system presents data in synchronized mode.

Synchronization mechanisms are needed to cope-up with problems to ensure the temporal ordering of streams and to maintain the presentation quality. The focus is on the synchronization problems in multimedia communications and potentially offers variety in synchronization frame work to handle synchronization specification at document level mechanisms. MHEG and HYTIME represent point and real-time synchronization which are used for synchronizing the hypermedia documents. Existing schemes lacks extensibility and flexibility that is needed in current web-based systems to accelerate the communication software development.

Index Terms :

Multimedia, synchronization, MHEG, Hytime, Temporal behavior, inter-stream synchronization, presentation scheduling, HyperODA, MHEG object, Semaphores.

---***---

1. INTRODUCTION

Synchronization is of two types, intra object refers time relation between various presentation units of one time-dependent media object where as inter object refers to Synchronization between media objects. Time dependent media objects usually consist of a sequence of information units which is a logical data units, the duration of open LDU is not predictable before the execution of the presentation.

The synchronization agency triggers one of the three synchronization mechanisms, point synchronization or real-time continuous or adaptive synchronization in order to adapt to the run-time and life-time presentation requirements of an application. Multimedia communication deals with transfer of data over the network among multimedia systems. These systems include multiple sources of various media that are either spatially or temporally related to create composite multimedia documents.

Continuous media (stream) such as audio and video are characterized by well defined temporal relationship between subsequent presentation units to be played (a presentation unit is a logical data unit that is perceivable by the user). Generally, the process of maintaining the temporal order of one or more media streams is called multimedia synchronization. The problem of maintaining continuity within a single stream is referred as intrastream or serial synchronization, whereas the problem of maintaining continuity among the streams is called as inter-stream or parallel synchronization. The intrastream and inter-stream synchronizations are necessary for both live stream(s) as well as for stored media stream(s) presentations [9].

There are several tools available to enable inter task communication and task Synchronization. Semaphores are intertask communication tools used to protect shared data resources. Tasks can call Take-Semaphore and Release-Semaphore functions. If one task has called Take-Semaphore and has not called the Release-Semaphore to release it, then any other task that calls Take-Semaphore will block until first task calls Release-Semaphore. Whenever task takes a semaphore it is potentially slowing the response of any other task that needs the same semaphore. There are two types of semaphore binary semaphore and counting semaphore. These are used when more than one task uses the same resource like in the case of a buffer pool management. Using a different semaphore for highest priority tasks ensures better response. Multiple semaphores can be used to protect different shared resources.

2. TEMPORAL RELATIONSHIPS MAINTENANCE

Within a multimedia streams it depends on following parameters :

- The delays experienced by the presentation units in the network to reach its receiver which varies according to network load.
- Delay variations of inter-arrival of presentation units at the receiver due to varying network load.
- Delay variations in presentation at the receiver due to varying CPU load and protocol processing delays.
- The clock time difference between the sender and the receiver.
- Rate of change of clock skew because of temperature differences or imperfections in crystal clocks.
- Change in generation and presentation rates due to server and receiver load variations.
- Time difference delay among the streams.
- Time interval in which the temporally related presentation units of the streams are presented.

3. SYNCHRONIZATION SPECIFICATION IN DOCUMENT LEVEL

Multimedia data inclusion into documents leads to the specification of layout (presentation) structure of classical text/graphics documents, multimedia information spatial and temporal characteristics [10]. Presentation of a text/graphics document is defined by the structuring information established prior to its presentation. A user viewing the document may select the parts which wants to view. A multimedia document potentially offers a much greater variety regarding its presentation.

i. A simple presentation schemes will only replay a predefined scheme of multimedia data, for instance, in a music video combining parts of a song with presenting different imaginative sceneries.

ii. To allow users to make selections of multimedia data to be viewed. An interactive video presentation is a good example.

iii. The user interaction may be extended to control the presentation style of multimedia data, reversing or slowing-down a presentation fall into this category.

It can be seen, that the range covered by multimedia documents is very large and exceeds by far the scope of classical text/graphics documents.

Two core abstractions are used to describe multimedia presentation in documents. Here the terminology of [6] identifying multimedia objects and actions are used. Video and audio sequences are examples of simple multimedia objects. In general an object corresponds to a presentation which is perceived by a user as a logical unit. Compound objects may be defined out of simple objects, e.g. a movie object consisting of a video and an audio object. The presentation of an object is referred to as an action, it can be simple or compound. The presentation of multimedia data is specified in a scheme for scheduling actions which is referred as presentation scheduling[6]. Present approaches are presentation scheduling and presentation style control, all functions influencing the visible temporal behavior of an action i.e. reversing presentation and speeding it up.

3.1. Presentation scheduling: It specifies the sequence of actions which are carried out.

i. **Types of approaches :** A scheduling scheme has to specify the actions of the generation and the effect of events. This approach specification classify into three groups according to the different approaches concerning event generation.

A. **Event generation in action :** Two types of events are specified:

- Beginning of action : The beginning of an action is always triggered by the ending of another action.
- Ending of an action : The ending of an action is enforced inherently by the action itself.

In some approaches, asynchronous ending enforced by the user is also possible. The effect of ending events action triggered by them is specified by using different constructs.

In [8] the scheduling scheme is given in the form of the object composition Petri nets (OCPN) where actions are specified as nodes and associated with a fixed duration. The scheme defines 13 elementary OCPNs corresponding to the 13 basic temporal relations which can exist between time intervals including concurrence and sequentiality of presentation and more complex objects are constructed by combining the elementary OCPNs (chaining OCPNs and expanding action nodes). Figure 1 illustrates this approach where first action X is performed for Tx sec and is followed by the concurrent actions Y and Z lasting Ty and Tz sec respectively.

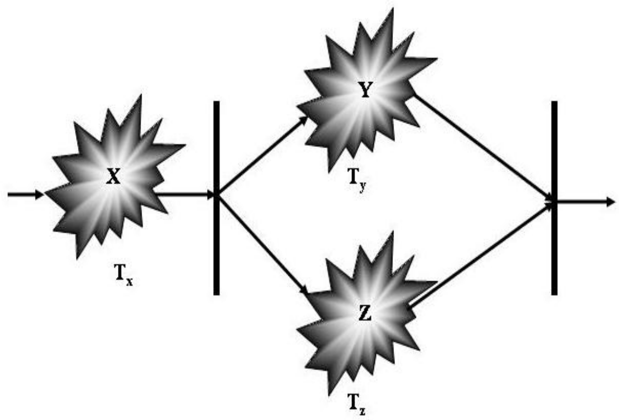

Figure 1: Petri Net - Object composition

Transitions in OCPNs are triggered by action ending events where the ending of X triggers actions Y and Z. The whole compound action is finished if both Y and Z are finished.

A more complex scheme is presented in [7] and [3]. Where path operators are introduced to describe the behaviour of actions. Six basic operators are defined, out of which more complex ones can be composed. For instance, the parallel-last operator specifies parallel execution of two actions, both started at the same time. The compound presentation ends, when both actions have ended. The parallel-first operator differs from this, in that it specifies the ending of the compound presentation as given by the first ending of an action The sequential execution of two actions is expressed by the sequence operator, the selection operator selects one out of two actions. The repetition of an action is specifiable as well as performing an action number of times simultaneously. All these operators as well as the compound ones are mappable onto Petri net structures with nodes representing actions with a deterministic or non-deterministic duration. The Petri net as shown in figure 2 specifies the parallel actions X1 and X2, started at the same time. Ending of one of these (parallel-first operator $\vee$) triggers action Y1 and ending of both actions (parallel-last operator $\wedge$) starts Y2.

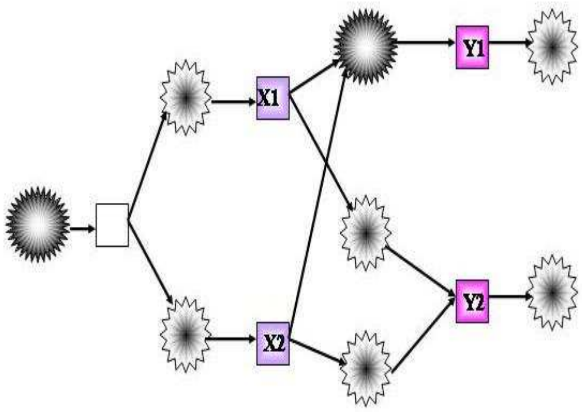

Figure 2 : Petri Net - Paths mapped

The scheme expressed by using the basic operators looks :

path (X1 $\vee$ X2) Y1 end

path (X1 $\wedge$ X2) Y2 end.

A third scheme is based on the concept of presentation frames [6]. A frame is a unit comprising several actions performed corresponding to a frame type specification. A frame is started by the ending event of one or more earlier frames. Depending on the frame type more than one ending event can be generated within a frame also a user may enforce the ending both of a single action within a frame and of a whole frame. Action durations can be specified or left indeterministic and the duration of an action can be derived from the corresponding presentation frame by using reconciliation strategies.

Five frame types are specified:

- Sequentialiser
- Parallelizer
- Splitter
- Combiner
- Brancher

Splitters are used to generate more than one ending event, while combiners and branchers specify the start of a frame depending on more than one event.

The three approaches presented differ in a number of ways.

a. The duration of an action is required in the first scheme to be deterministic, while this is not the case in the other schemes. Since the inclusion of user activities as actions implies an indeterministic action duration, the first scheme excludes this possibility.
b. The modeling power of the first scheme is rather restricted, while the other two are more powerful in that they can define the start of actions in a greater variety. Furthermore, they are conceived in a modular way, such that later extensions can be added easily.

B. **Event generation in reference point :** The approach in [2] augments the event generation which allow for reference points within actions. If the presentation of an object has reached to certain reference point, an event is generated. Reference points are inserted into the objects at arbitrary points. In a video sequence, events may be inserted to separate different scenes. The concept of reference should be avoid for handling of objects at too low-level granularity.

C. **Time system based event generation :** These approaches found in [5],[2],[1] the link action initiate to time instants generated by time coordinate systems. Start of an action is simply given as a time. This method is straight-forward and has two disadvantages:

- A clock is required for advancing the time system
- the scheme does not specify links between actions

While in the approach mentioned action terminations initiate other actions, the same is not specifiable here. Hence, the specification relations between actions with indeterministic duration is not possible. Thus, purely time based systems require knowledge of all action initiation times in advance.

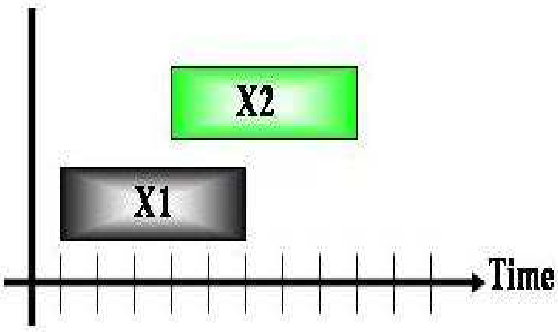

Figure 3: Referencing time system

ii Standard Approaches

a) **ODA/HyperODA :** It is the extension for temporal relationships and are basically the grouping of timed objects, called actions into composite actions. Actions have a start time and duration. Actions trigger other actions by events. Actions of a composite action can be synchronized in sequential, parallel or cyclic order and specify when to deliver an event. As actions are ordered hierarchically, the description of temporal behavior is distributed as attributes at each composite action for all of its constituents. This approach is similar to the layout specification in ODA.

b) **SGML/HyTime :** The temporal relationships lies in the domain of HyTime, the finite coordinate space module has to be employed. Measuring an axis in some temporal dimension allows to express high-level synchronization by placing events in a coordinate space. Placement may occur in absolute terms. In HyTime axis dimensions may be virtual or real. For both cases placement strategies are exactly the same. An event with a duration of 4 quanta takes twice as long as one with a duration of 2 quanta. Virtual quanta are abstract units and need some projection to real world units. During rendition, a virtual quantum may be mapped to ten seconds or a minute or some other real quantum. This scheme comes from the music domain, where temporal relationships are specified using full, half or quarter notes. Tempo statements then govern the actual duration of notes.

c) **MHEG :** This object is composed of other objects which may themselves be composite. Spatial and temporal relationships are handled almost the same. Within a certain object level, objects may be synchronized either in a parallel or sequential manner. In the first case, begin of the child objects is determined by some delay with respect to the begin of the parent object. In the second case begin of a child object refers to the end of the elder sibling, possibly associated with some delay.

It also supports handling of events that may be generated by user interaction or automatically by object processing, e.g. a user stopping an audio recording. For this purpose, the conditional action class is provided. An instance of this class associates an event, possibly including an event condition, with an action to be performed.

3.2. Presentation style control : It specifies for an action. The range of possible control functions are very much depending on the type of document to be viewed and the capabilities of the underlying system [4] control functions. With respect to actions these are defined: start, stop, pause and resume. Additionally, medium-dependent functions are included. For instance, "convenient" working with stored video sequences should include "by default" functions like reversing presentation, slowing it down, speeding it up, determining the start position for the presentation or jumping in the presentation. The common denominator of these functions are that they change the visible temporal behavior of the presentation of an object. They influence the behaviors of time (e.g. reversing a video presentation is often referred to as "time running backward"). The solution offered in [4] for facilitating the concept of these functions are define for each object an object time which is considered as separate from the world time. The latter is supposed to reflect the real time, advancing forever at a constant rate. In contrast, the former is employed to describe the behavior of the object presentation with respect to the world time. Besides start, stop, pause and resume, following four methods for controlling the object time are offered. Using these methods it is possible to provide the desired user control.

i. translate : A translation positions the start of an action with respect to world time.
ii. Cue : Cue positions the starting point within an object.
iii. Scale : Scale determines the advance rate of the object time in relation to the world time
iv. Invert : It specifies the advance direction, i.e. whether the object time should run forward or backward related to the world time.

HyTime does not explicitly include mechanisms or concepts to define user interaction during presentation. The basic concepts which HyTime provides is to employe in order to express schemes of presentation control. HyTime does not know about action content but only its placement. Thus, actions containing user control functions is used to read control information from the user. Such actions are linked to other actions reflecting a different presentation sequence desired by the user. The main point is that, HyTime only

provides some basic principles that employed in many different ways including specification of user interaction.
User driven presentation control in MHEG is expressible by the conditional action class and provides a set of predefined event types one of which covers user input. With this facility, other actions are triggered that change the MHEG-object's presentation in the desired way and other event types are defined by applications.

4. DISCUSSION

In presentation scheduling, the overall conclusion is that all presented approaches have deficiencies. All of them allow for specifying static presentation schemes, combining actions with fixed durations. Some of them are time system based exclude linking actions of non-deterministic duration. As a consequence, these schemes do not allow for including user-driven actions into the scheduling scheme. HyTime and MHEG also classified as time system based constitute an exception as they allow for non-deterministic duration of actions. The action driven event generation is currently restricted to small set of possible specifications. Future experience with document handling have to show, whether the corresponding expressive power is satisfactory.
Presentation scheduling concept include sequencing and synchronizing actions as a whole and presentation style control influences the temporal behavior within an action. Controlling the presentation of a multimedia object can be based easily on the separate concepts of world and object time. Controlling the relation between object and world time is a powerful means for specifying the actual temporal behavior of an action.

5. CONCLUSION

Each architecture has been designed for a specific set of applications which do not require total flexibility, thus the document architectures mentioned do not covers logical, layout and physical structure without any restrictions. However, due to support of new content portions, existing concepts are extended. The proposal within HyperODA allows for disaggregation of document instances leading to a richer physical structure. HyTime architectural allow for layout specification to a certain extent which introduces some kind of layout structure within the field of SGML based documents employing HyTime.
HyTime covers more flexible and richer techniques for identification and addressing of certain document portions and the establishment of relationships between them. In this respect it offers a broad framework that also contains provisions for addressing in other document architectures.
All architectures uses the object oriented approach. In ODA and SGML types of structural building blocks are defined which are similar to the notion of classes that allow for creation of multiple instances, MHEG fully follows the object oriented approach.
The ODA standard aims to allow document specification that contains all the information necessary to process the document. The receiver of the document does not need to know where the document comes from, which application it belongs to, etc. Therefore every aspect of document structuring and processing has to be standardized like content architectures or layout attributes. For support of specifying multimedia documents. Within the ODA field new content architectures are defined and the extensions discussed within HyperODA.
SGML is restricted to standardization of syntactical aspects, specifying the logical structure of a document. Users who exchange SGML documents and want them to be laid out in the same way have to agree on a layout specification i.e. agree on the semantics of the DTD which allows a certain document to be laid out differently by associating different layout specifications.. Both standards follow a different concepts. The requirements of an application decides which one is appropriate. HyTime is intended to support multimedia document specification and only standardizes the basic mechanisms, leaving many issues for the application.

REFERENCES

[1] D. P. Anderson and G. Homsy. A Continuous Media I/O Server and Its Synchronization Mechanism. *IEEE Computer*, 10 1991.

[2] G. Blakowski, J. Huebel, and U. Langrehr. Tools for Specifying and Executing Synchronized Multimedia Presentations. *2nd Intl. Workshop on Network and Operating System Support for Digital Audio and Video*, 11 1991.

[3] S. Gibbs, C. Breiteneder, L. Dami, V. de Mey, and D. Tsichritzis. A Programming Environment for Multimedia Applications. *2nd Intl. Workshop on Network and Operating System Support for Digital Audio and Video*, 11 1991.

[4] S. Gibbs, L. Dami, and D. Tsichritzis. An Object-Oriented Framework for Multimedia Composition and Synchronistion. *Eurographics Multimedia Workshop, Stockholm*, 1991.

[5] S. Gibbs. Composite Multimedia and Active Objects. *Proc. OOPSLA'91*, pages 97–112, 1991.

[6] P. Hoepner. Presentation Scheduling of Multimedia Objects and Its Impact on Network and Operating System Support. *2nd Intl. Workshop on Network and Operating System Support for Digital Audio and Video*, 11 1991.

[7] P. Hoepner. Synchronisation der Praesentation von Multimedia-Objekten - Modell und Beispiele. *Informatik-Fachberichte 293: Telekommunikation undmultimediale Anwendungen der Informatik*, pages 455–464, 10 1991.

[8] T. D. C. Little and A. Ghafoor. Synchronization and Storage Models for Multomedia Objects. *IEEE Journal on Selected Areas in Communications*, 8(3), 3 1990.

[9] S.S. Manvi, Venkataram, P., An agent based synchronization Scheme for multimedia applications. The Journal of Systems and Software 79, 2006, p.p. 701-713.

[10] R. N. Jugele and V. N. Chavan. Synchronization Scheme in Multimedia and Hypermedia Application Development. International Journal of Computer Technology & Applications, ISSN:2229-6093,Vol 3, issue 5, Sep-Oct 2012, pp. 1818-1822.

Books :

01. Principles of Multimedia By. Ranjan Parekh
    Tata McGraw Hill Companies.
02. Hypertext and Hypermedia By. J. Nielsen
    Academic Press.

BIOGRAPHIES :

R. N. Jugele received his B.Sc. degree from Nagpur University and M.Sc. degrees in Computer Science from Marathwada University, Aurangabad, Maharashtra, India in 1991 and 1993 respectively. Currently, he is working as a Associate Professor in Department of Computer Science, Science College, Congress Nagar, Nagpur. Maharashtra.

Dr. V. N. Chavan Head, Department of Computer Science, S. K. Porwal College, Kamptee, Dist : Nagpur. Maharashtra. He is research guide in Computer Science subject in various universities and have wide knowledge in research field since last 22 years. He is a member of various bodies in Computer Science subject.