A Multimedia Synchronization Service and Implementation Architecture for Pre-orchestrated Data Communications

Ronaldo Mussauer de Lima — mussauer@gta.ufrj.br
Otto Carlos Muniz B. Duarte — otto@gta.ufrj.br
Grupo de Teleinformática e Automação (GTA)
COPPE/PEE - Programa de Engenharia Elétrica
Universidade Federal do Rio de Janeiro
C.P. 68504 - CEP 21945-970 - Rio de Janeiro - RJ - Brasil

ABSTRACT

This work defines a multimedia synchronization service, specifies the related protocol and proposes an implementation scheme for a pre-orchestrated data communication system. The defined facilities support distributed real-time multimedia applications that comprises pre-orchestrated data, stored on a remote database. The synchronization mechanisms are based on the XOCPN model. The proposed architecture is simple, modular and flexible. Furthermore, it is based on specialized modules, specific layer interfaces, and data structures to avoid memory copies. The proposed implementation seems to be very efficient.

1. INTRODUCTION

The advance in communication systems, including high-speed networks and protocols, has enabled the development and implementation of distributed multimedia applications, which integrate several types of media such as audio, video, text and graphics [1]. The need of integration derives from the dependencies among media, such as temporal, spatial and causal relations [2]. Each media type has specific presentation, storage and communication requirements and may experience random delays during transmission and retrieval. In order to support multimedia data communications and coherent presentations at the destination, it is essential to assure the synchronization of data elements.

Several synchronization techniques have been proposed [2]-[6], each one presenting advantages and disadvantages, depending on the type of application, the data sources, the communication system, the required quality of service (QoS), the available resources, etc. [7]

This work focuses on synthetic temporal relations and the target application is a communication broadcast system of multimedia educational clips. The synchronization mechanism is based on the eXtended Object Composition Petri-net (XOCPN) model [5,8], which provides a complete presentation and communication schedule for pre-orchestrated multimedia data in a distributed system.

This paper presents an implementation scheme for multimedia synchronization of pre-orchestrated data, using XOCPN-based mechanisms. It defines a synchronization service, specifies synchronization protocol procedures and proposes an implementation architecture, that includes specialized modules and specific data structures.

This work is organized as follows. Section 2 describes the basic concepts related to the multimedia synchronization. Section 3 presents the XOCPN model, the proposed synchronization service, and a diagram of the proposed modules. Section 4 presents the implementation architecture and the XOCPN data structures. Finally, conclusions are presented in section 5.

2. BASIC CONCEPTS

This section presents some definitions and classifications comprised in multimedia synchronization. Isochronous data refer to a data generated at a constant rate, from a continuous event. These data have an implicit temporal relation in its generation, requiring rate-controlled transmission (e.g., audio and video). Anisochronous data have no relation with generation, transmission or presentation rates. In this case, the spatial composition of the presentation is the most important data requirement (e.g., text and graphics). Pre-orchestrated data concern to synthetic temporal relations among data. In this case, the time dependencies are explicitly formulated (e.g., graphical presentation with explaining texts). A multimedia object is the basic multimedia element. An object refers to a multimedia information unit, complex or not, comprising data elements of a specific media. A flow is a temporal sequence of multimedia objects. Intramedia synchronization refers to the temporal relations between multimedia objects related to one media. Intermedia synchronization refers to the temporal relations among multimedia objects related to more than one media.

3. THE SYNCHRONIZATION LAYER

The proposed XOCPN-based synchronization service provides synchronized communication, including intra- and intermedia synchronization. The facilities support real-time distributed applications that communicate pre-orchestrated multimedia information stored in a remote database. Figure 1 shows the application and synchronization layers in the hierarchical architecture of the system.

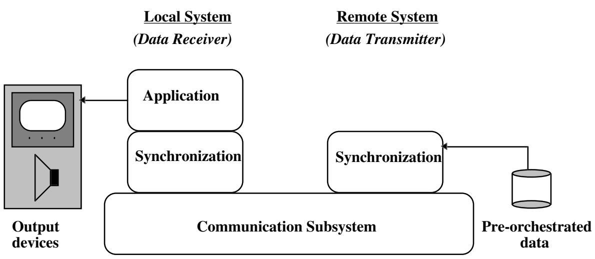

Figure 1: The hierarchical layers of the system.

The Object Composition Petri-net (OCPN) model is a special kind of timed Petri-net with some enhanced characteristics [9]. Each place in OCPN is related to resource (audio, video, text, etc.) utilization and has a time duration. Firing of a transition in OCPN is an instantaneous event. Through OCPN it is possible to specify all temporal relations like 'before', 'during', 'after', etc. The OCPNs are serial-parallel nets represented by graphs and specified by a tuple. Figure 2 shows a simple OCPN.

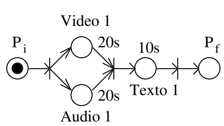

Figure 2: Representation of a OCPN.

XOCPN is an extension of the OCPN model including communication's features for distributed environments [5]. Through XOCPN it is possible to prevent and compensate the network transmission anomalies. It is suitable for continuous temporal relations among objects (isochronous data) which demands for rate-controlled communication and presentation. The XOCPN model divides objects into sub-objects called Synchronization Interval Units (SIU), in order to provide a fine-grain synchronization. XOCPNs are also represented by graphs and specified by the tuple {T, P, A, D, M, Y, Z}, where [8]:

T = {t_1, t_2,..., t_n} is a set of transitions;

P = {p_1, p_2,..., p_m} is a set of m places;

A = {TxP} U {PxT} $\rightarrow$ I, I = {1, 2,...} is a set of directed arcs;

D: P $\rightarrow$ {d_1,d_2} where d_1,d_2 represent the delay before an action and duration of an action, respectively;

R: P $\rightarrow$ {r_1, r_2,..., r_k} is a mapping from the set of places to a set of object types;

M: P $\rightarrow$ I, I = {0, 1, 2,...} is a mapping from the set of places to the number of tokens in a place;

Y: P $\rightarrow$ {Resource-setup, Resource-release, SIU-playout, SIU-transmit, Interstream-synchronize} is a mapping from the set of places to a set of action types to be performed during communication;

Z: P $\rightarrow$ {Addresses} is a mapping from the set of places to the addresses for QoS parameters, synchronization requirements and an SIU.

For the purpose of multimedia communications, a receiver and a transmitter XOCPN are generated from an OCPN specification. Each place in OCPN is replaced by a string of SIUs. Figure 3 shows the receiver and transmitter XOCPN corresponding to the OCPN of Figure 2. According to the pre-specified schedule, the receiver XOCPN playouts the multimedia objects at the user-site. To maintain the pace among related isochronous data flows, the Interstream Pacing Points (IPPs) are inserted in the receiver XOCPN. For example, it is possible to perform synchronization strategies to keep audio and video flows synchronized within 150 ms. Otherwise, the presentation quality degrades [8]. According to the pre-specified schedule, the transmitter XOCPN at the source-site delivers the multimedia objects to the network.

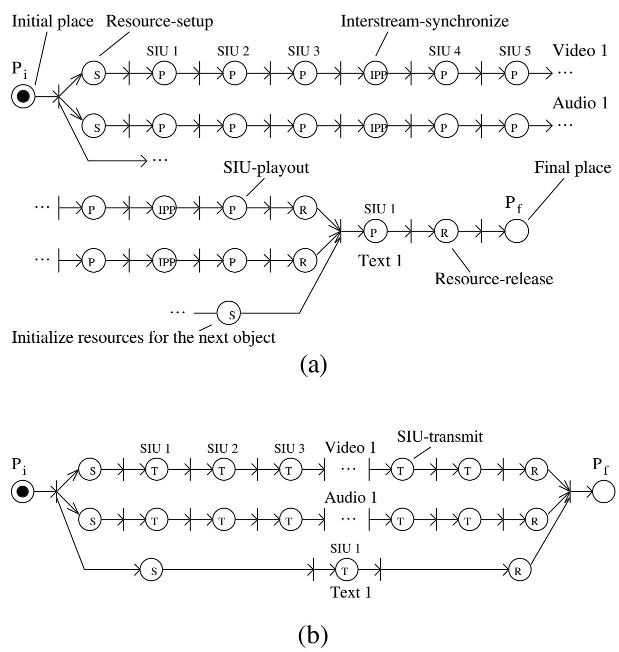

Figure 3: Representation of (a) receiver and (b) transmitter XOCPN.

The synchronization service is defined by its service primitives [10]. Table 1 shows the service primitives for the receiver synchronization entity.

Primitive	Description
SINC-INITIALIZE.Request	Request to initialize the multimedia presentation
SINC-PLAYOUT.Indication	Indication of SIU's presentation
SINC-STOP.Request	Request to stop presentation
SINC-STOP.Indication	Indication of fail or end of SIU

Table 1: Synchronization service primitives.

The specified synchronization protocol includes five Protocol Data Units (PDUs). Table 2 shows the PDUs and their descriptions.

PDU	Description
IREQ	Initialize request for the presentation procedures
INETDT	Initialize request for the XOCPN generation
IREADY	Confirmation of the XOCPN generation
IFAIL	Indication of initialization fail
SIUDT	Sub-object (SIU) data

Table 2: Protocol Data Units.

The non-deterministic network behavior may introduce errors and jitters during data communication. It is essential for the user-site to compensate problems related to slow flows, maintaining the temporal relations among data, specified by XOCPN, and providing the desired quality of presentation. These actions include different synchronization strategies [5], like blocking, non-blocking and restricted blocking. The choice for a specific strategy depends on the type of synchronization (intramedia or intermedia).

In the blocking strategy, the presentation process suspends until the expected SIU arrives. In the non-blocking strategy, the presentation process continues even though the expected SIU has not arrived. In the restricted blocking strategy, the presentation process suspends for a pre-specified time period. After this period, the process skips the late SIU and may, for instance, repeat the presentation of the last SIU.

In this work, we adopted the restricted blocking strategy for intramedia synchronization and for intermedia synchronization related to the transitions among different flows. On the other hand, we adopted the non-blocking strategy for intermedia synchronization related to IPP places. The blocking strategy was not used in order to avoid interrupting during playout of multimedia objects.

The State Machines and the proposed modules related to the receiver XOCPN execution are showed in Figure 4.

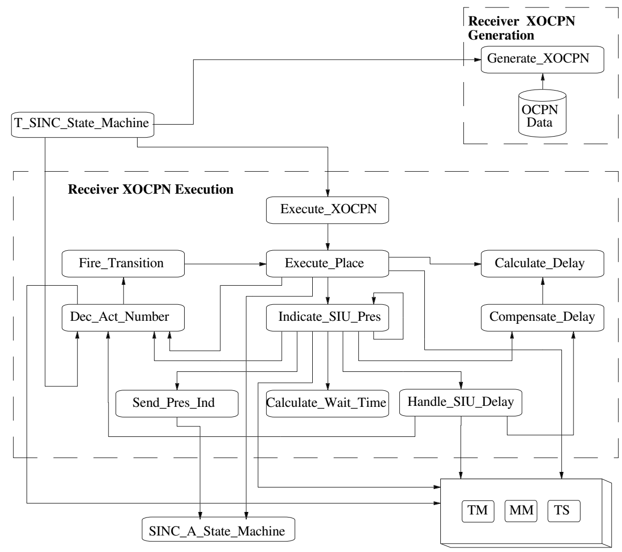

Figure 4: Procedure diagram for the receiver XOCPN.

4. IMPLEMENTATION ARCHITECTURE

An efficient implementation architecture [11] was defined to support multimedia data communication. The synchronization mechanisms' implementation comprised representation of the Petri-net's places and transitions; the automatic generation of the XOCPN from an OCPN specification; executing of the places and firing of the transitions; handling and compensating of SIU's delay; applying intermedia synchronization strategies and controlling intramedia synchronization. The implementation also included State Machines, interface between layers, data communication management (handling data structures and interfaces, connections, memory, timers and tasks). Also, multimedia data, like audio and video, comprised capture, compression, storage, retrieval and transmission procedures.

A high performance implementation architecture has been used for the purpose of synchronization mechanisms' implementation. This architecture (Figure 5) has been designed for communication of multimedia data, defining three specialized modules:

- A priority-based task scheduling (TS), that avoids unnecessary queue tests and takes advantage of required atomicity of some functions;
- An optimized memory management (MM) with fixed and variable size buffer allocation/deallocation and separated transmitter and receiver memory regions, keeping fragmentation as a minimum;

- A timer management (TM) efficiently implemented to avoid unnecessary tests in each timer clock interrupt, allowing the creation of a large number of timers.

The interface between the Synchronization Layer and adjacent layers consists of a pair of queues per connection and a set of data structure access procedures.

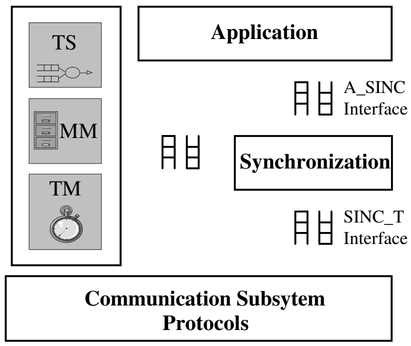

Figure 5: Implementation architecture for synchronized communications.

Each XOCPN place and each transition are represented by a fixed size structure. The *Place_Structure* contains fields related to the XOCPN tuple. The *Transition_Structure* contains one field to point to all output places and another field corresponding to the number of all the input places to be executed before firing of a transition. Figure 6 shows the linked structures, based on Figure 3(a).

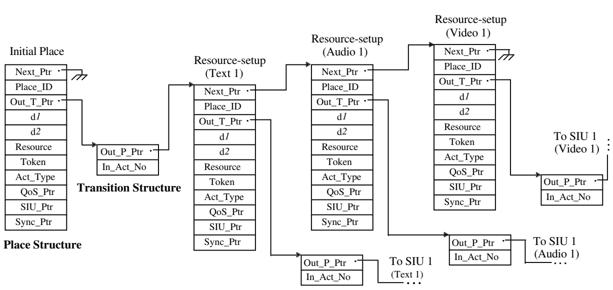

Figure 6. XOCPN place and transition structures.

The implemented system is coded in C language, using Sparc Station, Solaris operating system and a LAN environment.

5. CONCLUSIONS

This work has presented a synchronization service and an implementation architecture for pre-orchestrated multimedia communications. The synchronization mechanisms are based on the XOCPN model and take into account synthetic temporal relations among stored multimedia objects. The definition of the service primitives and the protocol data units was presented. The proposed synchronization pointer-based data structures make easy the access to a place and transition as well as the implementation of intermedia synchronization strategies. High performance is one of the most important features of the proposed implementation architecture due to the reduction of memory copies.

REFERENCES

[1] Blakowski, G. and Steinmetz, R., "A Media Synchronization Survey: Reference Model, Specification, and Case Studies", *IEEE J. Select. Areas Commun.*, vol. 14, nº 1, pp. 5-35, January 1996.

[2] Courtiat, J.-P.; Carmo, L.F.R.C., and Oliveira, R.C., "A General-Purpose Multimedia Synchronization Mechanism Based on Causal Relations", *IEEE J. Select. Areas Commun.*, vol. 14, nº 1, pp. 185-95, January 1996.

[3] Nicolaou, C., "An Architecture for Real-Time Multimedia Communication Systems", *IEEE J. Select. Areas Commun.*, vol. 8, nº 3, pp. 391-400, April 1990.

[4] Ramanathan, S. and Rangan, P.V., "Feedback Techniques for Intra-Media Continuity and Inter-Media Synchronization in Distributed Multimedia Systems", *Computer Journal*, vol. 36, nº 1, pp. 19-31, 1993.

[5] Qazi, N.U.; Woo, M., and Ghafoor, A., "A Synchronization and Communication Model for Distributed Multimedia Objects", *Proc. ACM Multimedia '93,* Anaheim, CA, pp. 147-56, August 1993.

[6] Escobar, J.; Partridge, C., and Deutsch, D., "Flow Synchronization Protocol", *IEEE/ACM Trans. on Networking*, vol. 2, nº 2, pp. 111-21, April 1994.

[7] Ehley, L.; Furht, B., and Ilyas, M., "Evaluation of Multimedia Synchronization Techniques", *Proc. of the International Conference on Multimedia Computing and Systems*, pp. 514-19, May 14-19, 1994.

[8] Woo, M.; Qazi, N.U., and Ghafoor, A., "A Synchronization Framework for Communication of Pre-orchestrated Multimedia Information", *IEEE Network*, vol. 8, nº 1, pp. 52-61, January 1994.

[9] Little, T.D.C. and Ghafoor, A., "Synchronization and Storage Models for Multimedia Objects", *IEEE J. Select. Areas Commun.*, vol. 8, nº 3, pp. 413-27, April 1990.

[10] De Lima, R.M. and Duarte, O.C.M.B., "Comunicação Sincronizada de Dados Multimídias Pré-orquestrados", *XXIII Seminário Integrado de Software e Hardware (SEMISH)*, Recife, PE, Brazil, pp. 285-96, August 1996.

[11] Albuquerque, C.V.N.; Nunes, M.D., and Duarte, O.C.M.B., "Performance Measurements in a Manufacturing Communication System", *Proc. of the International Symposium on Circuits and Systems (ISCAS)*, London, UK, pp. 69-72, May 30-June 2, 1994.